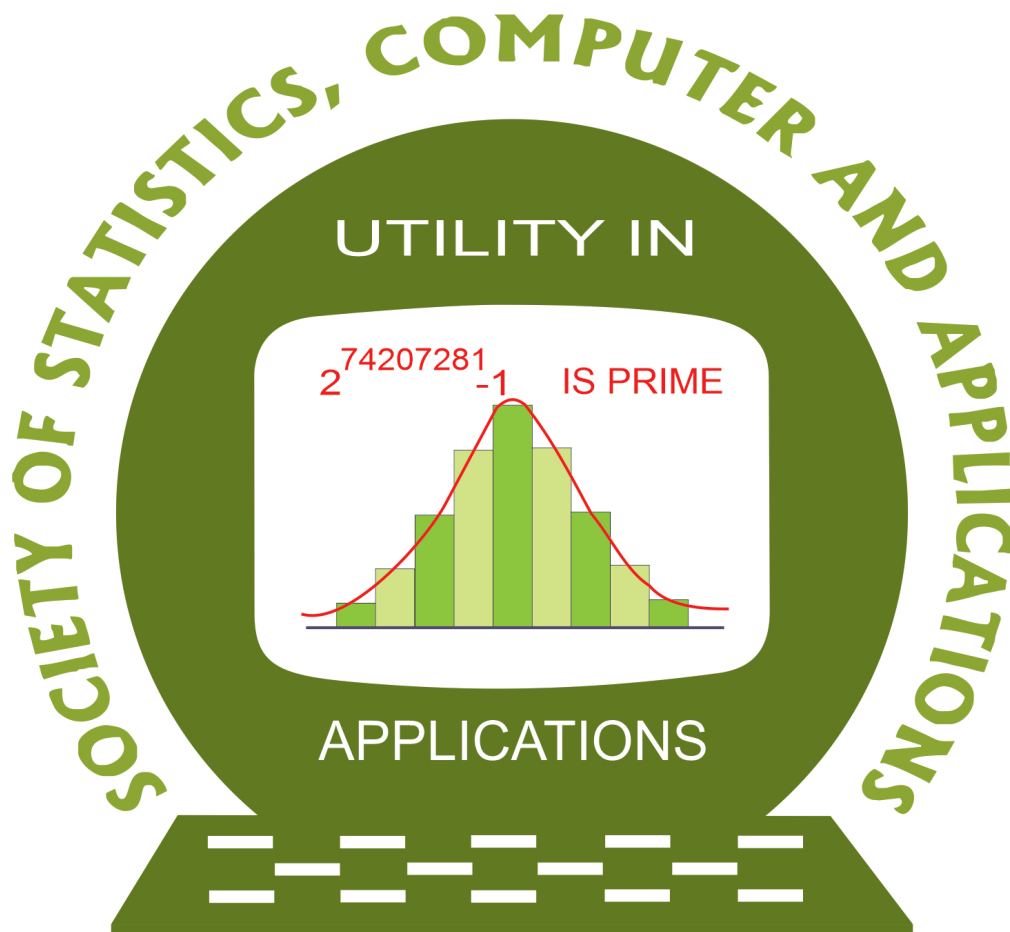


ISSN 2454-7395(online)

STATISTICS AND APPLICATIONS



FOUNDED 1998

Journal of the Society of
Statistics, Computer and Applications

<https://ssca.org.in/journal.html>
Volume 23, No. 2, 2025 (New Series)

Society of Statistics, Computer and Applications

Council and Office Bearers

Founder President

Late M.N. Das

President

V.K. Gupta

Executive President

Rajender Parsad

Patrons

A.C. Kulshreshtha

A.K. Nigam

Bikas Kumar Sinha

D.K. Ghosh

K.J.S. Satyasai

Pankaj Mittal

Prithvi Yadav

R.B. Barman

R.C. Agrawal

Rahul Mukerjee

Rajpal Singh

Vice Presidents

A. Dhandapani

Manish Kumar Sharma

Manisha Pal

P. Venkatesan

Praggya Das

Ramana D. Davuluri

S.D. Sharma

V.K. Bhatia

Secretary

D. Roy Choudhury

Foreign Secretary

Abhyuday Mandal

Treasurer

Ashish Das

Joint Secretaries

Aloke Lahiri

Shibani Roy Choudhury

Vishal Deo

Council Members

B. Re. Victor Babu

Banti Kumar

Bishal Gurung

Imran Khan

Mukesh Kumar

Parmil Kumar

Piyush Kant Rai

Rajni Jain

Rakhi Singh

Raosaheb V. Latpate

Renu Kaul

Shalini Chandra

Sukanta Dash

V.M. Chako

Vishnu Vardhan R.

Ex-Officio Members (By Designation)

Director, ICAR-Indian Agricultural Statistics Research Institute, New Delhi

Chair Editor, Statistics and Applications

Executive Editors, Statistics and Applications

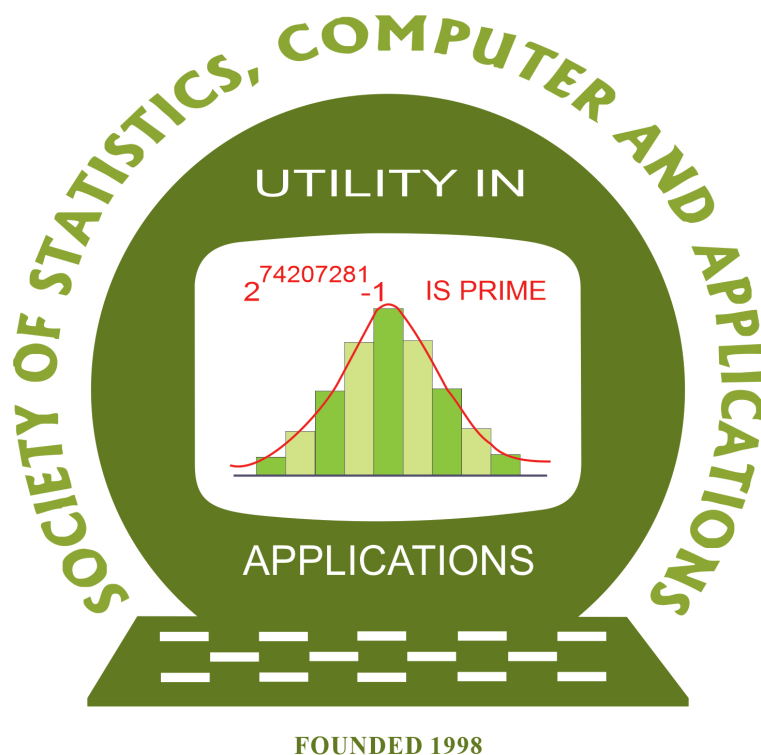
Society of Statistics, Computer and Applications

Registered Office: I-1703, Chittaranjan Park, New Delhi- 110019, INDIA

Mailing Address: B-133, Ground Floor, Chittaranjan Park, New Delhi-110019, INDIA

Statistics and Applications

ISSN 2454-7395 (online)



Journal of the Society of
Statistics, Computer and Applications
<https://ssca.org.in/journal.html>

Volume 23, No. 2, 2025 (New Series)

Statistics and Applications

Volume 23, No. 2, 2025 (New Series)

Editorial Panel

Chair Editor

V.K. Gupta, Former ICAR National Professor at IASRI, Library Avenue, Pusa, New Delhi -110012; vkgupta_1751@yahoo.co.in

Executive Editors

Durba Bhattacharya, Head, Department of Statistics, St. Xavier's College (Autonomous), Kolkata - 700016; durba0904@gmail.com; durba@sxccal.edu

Rajender Parsad, ICAR-IASRI, Library Avenue, Pusa, New Delhi - 110012; rajender1066@yahoo.co.in; rajender.parsad@icar.gov.in

Editors

Baidya Nath Mandal, Managing Editor, ICAR-Indian Agricultural Research Institute Gauria Karma, Hazaribagh-825405, Jharkhand; mandal.stat@gmail.com

R. Vishnu Vardhan, Managing Editor, Department of Statistics, Pondicherry University, Puducherry - 605014; vrstatsguru@gmail.com

Jyoti Gangwani, Production Executive, Formerly at ICAR-IASRI, Library Avenue, New Delhi 110012; jyoti0264@yahoo.co.in

Associate Editors

Abhyuday Mandal, Professor and Undergraduate Coordinator, Department of Statistics, University of Georgia, Athens, GA 30602; amandal@stat.uga.edu

Anirban Chakraborti, School of Computational and Integrative Sciences and School of Sanskrit and Indic Studies, Jawaharlal Nehru University, New Delhi 110067, India; anirban.chakraborti@gmail.com

Ashish Das, 210-C, Department of Mathematics, Indian Institute of Technology Bombay, Mumbai - 400076; ashish@math.iitb.ac.in; ashishdas.das@gmail.com

David Banks, Department of Statistical Science, Duke University, Durham, NC27708-0251 USA; david.banks@duke.edu

Deepayan Sarkar, Indian Statistical Institute, Delhi Centre, 7 SJS Sansanwal Marg, New Delhi - 110016; deepayan.sarkar@gmail.com; deepayan@isid.ac.in

Indranil Mukhopadhyay, Professor and Head, Human Genetics Unit, Indian Statistical Institute, Kolkata, India; indranilm100@gmail.com

Janet Godolphin, Department of Mathematics, University of Surrey, Guildford, GU2 7XH, UK; j.godolphin@surrey.ac.uk

Jiani Yin, Associate Director at Servier Biostatistics Project Lead Boston, MA, USA; jianiyin@gmail.com

Jyotirmoy Sarkar, Department of Mathematical Sciences, Indiana University Purdue University, Indianapolis, IN 46202-3216 USA; jsarkar@iupui.edu

K. Muralidharan, Professor, Department of Statistics, faculty of Science, Maharajah Sayajirao University of Baroda, Vadodara; lmv_murali@yahoo.com

K. Srinivasa Rao, Professor, Department of Statistics, Andhra University, Visakhapatnam, Andhra Pradesh; ksraoau@gmail.com

Lu Chen, NISS-NASS, USDA, USA, Research and Development Division, Sampling and Estimation Research Section; luchan459@gmail.com

M.R. Srinivasan, Visiting Professor, Chennai Mathematical Institute, Siruseri, Chennai-6031035, India; mrsrin8@gmail.com

Murari Singh, 65 Fulbert Crescent, Scarborough, ON M1S1C5, Canada; mandrsingh2010@gmail.com

Pranabendu Mishra, Computer Science Division, CMI, Chennai; pranabendu@cmi.ac.in

Pritam Ranjan, Indian Institute of Management, Indore - 453556; MP, India; pritam.ranjan@gmail.com

Ramana V. Davuluri, Department of Biomedical Informatics, Stony Brook University School of Medicine, Health Science Center Level 3, Room 043 Stony Brook, NY 11794-8322, USA; ramana.davuluri@stonybrookmedicine.edu; ramana.davuluri@gmail.com

Rituparna Sen, Indian Statistical Institute Bengaluru, Karnataka 560059, India; ritupar.sen@gmail.com

S. Ejaz Ahmed, Faculty of Mathematics and Science, Mathematics and Statistics, Brock University, ON L2S 3A1, Canada; sahmed5@brocku.ca

Sat N. Gupta, Department of Mathematics and Statistics, 126 Petty Building, The University of North Carolina at Greensboro, Greensboro, NC -27412, USA; sngupta@uncg.edu

Satya Prakash Singh, Department of Mathematics and Statistics, IIT Kanpur, Kanpur 208016, UP, India; snggstyprksh@gmail.com

Satyaki Mazumdar, Indian Science Education and Research Kolkata, Mohanpur, Nadia-741246, West Bengal; satyaki@iiserkol.ac.in

Saumyadipta Pyne, Health Analytics Network, and Department of Statistics and Applied Probability, University of California Santa Barbara, USA; spyne@ucsb.edu, SPYNE@pitt.edu

Shuvo Bakar, Faculty of Medicine and Health, University of Sydney, Australia; shuvo.bakar@sydney.edu.au

Snehanshu Saha, Professor, Computer Science and Information System, Head - APPCAIR 9 (All campuses), BITS Pilani K K Birla Goa Campus; snehanshus@goa.bits-pilani.ac.in

Snigdhasu Chatterjee, School of Statistics, University of Minnesota, Minneapolis, MN -55455, USA; chatt019@umn.edu

Sourish Das, Data Science Group, Chennai Mathematical Institute, Siruseri, Chennai 603103; sourish.das@gmail.com

Steffano Marchetti, Department of Economics and Management, University of Pisa, Via Ridolfi n, 10, 56124 Pisa, Italy; stefano.marchetti@unipi.it

Suman Guha, Department of Statistics, Presidency University, 86/1, College Street, Kolkata, 700073; bst0404@gmail.com

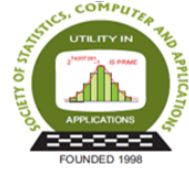
T.V. Ramanathan; Department of Statistics; Savitribai Phule Pune University, Pune; madhavramanathan@gmail.com

Tapio Nummi, Faculty of Natural Sciences, Tampere University, Tampere Area, Finland; tapio.nummi@tuni.fi

Tirupati Rao Padi, Department of Statistics, Ramanujan School of Mathematical Sciences, Pondicherry University, Puducherry; drtrpadi@gmail.com

Utkarsh Tripathi, Solventum (3M Health Care), Pittsburgh Pennsylvania, USA; utkarshbitsp@gmail.com

V. Ramasubramanian, ICAR-NAARM, Rajendranagar, Hyderabad, Telangana – 500030; ram.vaidyanathan@gmail.com



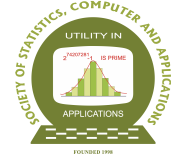
CONTENTS

1.	Gumbel Distribution: Comparison of Bayesian Estimators Under Various Loss Functions <i>P. M. Safwana and C. Chandran</i>	1–20
2.	Transient State Solution of Retrial Queueing System with Catastrophe <i>Neelam Singla and Ankita Garg</i>	21–36
3.	Finite Mixtures of Matrix Variate Log-normal Distributions for Clustering Skewed Three-Way Data <i>Shiva Kumar Kurva and Kiruthika C.</i>	37–57
4.	Balanced and Partially Balanced Semi-Latin Rectangles with Block Size Two <i>Kaushal Kumar Yadav, Sukanta Dash, Rajender Parsad, Baidya Nath Mandal, Anil Kumar and Mukesh Kumar</i>	59–70
5.	Three and Four Component Uniform Mixture Designs Based on Ellipsoidal Region - A Beginner's Training Manual <i>Poonam Singh and Himanshu Shukla</i>	71–88
6.	On Renewal Processes - Some Bounds and Characterisations <i>S. Ravi and Suman Kalyan Ghosh</i>	89–104
7.	The Modified Logistic Two-Parameter Estimators <i>Thayaparan Kayathiri, Manickavasagar Kayanan, and Pushpakanthie Wijekoon</i>	105–117
8.	Wrapped Generalized Lindley Distribution with Applications to Directional Data <i>Imliyangba and Bhanita Das</i>	119–137
9.	On the Likelihood Ordering and Tail Behavior of Certain Classes of Skew Normal Distributions <i>M. R. Anusree and C. Satheesh Kumar</i>	139–150
10.	Advancements in Power Inverse Rayleigh Modeling: Exploring Applications in Environmental and Medical Domains <i>Aadil Ahmad Mir, S. P. Ahmad and A. A. Bhat</i>	151–168
11.	VTPC: A Novel Virtual Tree Approach for Efficient Population Count in Binary Sequences <i>Samarth Godara, Prakash Kumar, Deepak Singh and Raju Kumar</i>	169–179

12.	Time Dependent Performance Analysis of Asynchronous Internet Switch - Multi Server Queueing System and Markovian QBD Process	181–189
	<i>Malla Reddy Perati, Shivaji Arepelly and Abhilash Vollala</i>	
13.	Bayesian DCC-MGARCH Models for Understanding the Price Dynamics of Volatile Series: A Case Study on Pulses Price in India	191–211
	<i>Achal Lama, Girish K. Jha, Bishal Gurung and Saurav Guha</i>	
14.	Nonparametric Tests Based on Ranks for Independence against Weighted Alternative with Missing Values	213–220
	<i>Parameshwar V. Pandit, Deepa Yogesh Kamat and Shubhashree Joshi</i>	
15.	On Log Odd Burr III Weibull Regression Model and its Application in Survival Analysis	221–236
	<i>Deepthy G. S., Lakshmi R. and Nicy Sebastian</i>	
16.	Reducing Dimensionality and Modeling Structural Dependence in Data Streams: An Approach Using Copula Matrices	237–266
	<i>Fereshteh Arad, Ayyub Sheikhi and Farshid Keynia</i>	
17.	A Framework for Building a Novel Causal Proximity Driven GNN from Biomedical Text	267–290
	<i>Samridhi Dev and Aditi Sharan</i>	
18.	Small Area Estimation Technique in Forestry Sector with Special Emphasis on National Forestry Inventory: A Review	291–316
	<i>Kamal Pandey, Sunil Chandra, Girish Chandra and Salman Khan</i>	

SHORTER COMMUNICATIONS

19.	Resolvability of Some BIB and Regular Group Divisible Designs	317–321
	<i>Shyam Saurabh, Sheet Nihal Topno and Dinesh Prasad</i>	
20.	Prabhu-Ajgaonkar's 1967 Result Revisited	323–329
	<i>Bikas K. Sinha and Manisha Pal</i>	



Gumbel Distribution: Comparison of Bayesian Estimators under Various Loss Functions

P. M. Safwana and C. Chandran

Department of Statistics, University of Calicut, Kerala, India

Received: 20 February 2024; Revised: 06 August 2024; Accepted: 10 August 2024

Abstract

Bayesian estimation is carried out for the scale parameter of Gumbel distribution under various loss functions. A new class of loss functions is introduced and an extensive Monte Carlo simulation study is carried out for the comparison of estimators under loss functions. Two different prior distributions for scale parameter are used. The posterior distributions have no closed form hence Lindley approximation as well as Markov Chain Monte Carlo methods are used for deriving estimates. The study is illustrated through a real life data.

Key words: Generalized extreme value distribution; Gumbel distribution; Bayesian estimation; Loss functions; Markov Chain Monte Carlo; Lindley approximation.

AMS Subject Classifications: 62K05, 05B05

1. Introduction

Extreme value distributions are the limiting distributions of maximum (minimum) of a large collection of independent and identically distributed random variables from an arbitrary distribution. The non-degenerate asymptotic distribution of maximum M_n must belong to one of the three possible general families of distributions called Type I, Type II and Type III extreme value distributions, also known as Gumbel, Frechet, and Weibull distributions respectively. These three distributions can be unified into a single family of distributions namely generalized extreme value distribution (GEVD), by incorporating an additional parameter. This unification work is done independently by Von Mises (1964) and Jenkinson (1955) and is a very helpful model for fitting purposes. The distribution function of GEVD has the form,

$$F(x; \alpha, \theta, \xi) = e^{-\left(1+\xi\left(\frac{x-\alpha}{\theta}\right)\right)^{-\frac{1}{\xi}}},$$

where $x > \alpha - \theta/\xi$, α , θ and ξ are the location, scale and shape parameters respectively. The Type I, Type II and Type III extreme value distributions correspond to the cases

$\xi = 0$, $\xi > 0$, and $\xi < 0$, respectively. The Gumbel case, the case $\xi = 0$ is interpreted as $\lim_{\xi \rightarrow 0} F(x; \alpha, \theta, \xi)$ and the distribution function corresponding to this can be derived as,

$$F(x; \alpha, \theta) = e^{-e^{-\left(\frac{x - \alpha}{\theta}\right)}}, \quad -\infty < x < \infty; \quad (1)$$

where $-\infty < \alpha < \infty$ and $\theta > 0$. Detailed literature on extreme value distributions and other related issues of GEVD can be seen in Leadbetter *et al.* (1983), Embrechts *et al.* (2013) and Coles (2001). The parameter estimation of extreme value distributions and GEVD are studied by many authors in a series of papers. The maximum likelihood (ML) estimation of GEVD are studied by Prescott and Walden (1983), Hosking (1985) and Smith (1985). Since the ML estimators do not exist in some cases other traditional methods of estimation were employed to estimate parameters of extreme value distributions and GEVD. Moment estimators can be derived in explicit form for location and scale parameters, however, they do not perform well in many situations. The probability weighted moments (PWM) estimators of Gumbel parameters were being studied by Landwehr *et al.* (1979). Mahdi and Cenac (2005) compared it with moment estimators and ML estimators and were seen to be favorable with the moment estimators and ML estimators. The PWM estimators of the parameters and quantiles of the GEVD were considered by Hosking *et al.* (1985). Another method of estimation similar to PWM was introduced by Hosking (1990) called method of L-moments. Hosking derived L-moment estimators for generalized extreme value distribution and showed that they are reliably efficient with those of ML estimators. Comparison of moment estimators, ML estimators, maximum entropy estimators, and PWM estimators of Gumbel distribution were studied by Phien (1987). Phien summarized that the moment estimators do not perform very satisfactorily as compared to the remaining three estimators also PWM is the best choice when considering unbiased estimators. The moment estimators have almost the same performance as the ML estimators.

Bayesian estimation techniques for estimating location and scale parameters of Gumbel distribution were also used by several authors. Effect of prior assumptions on the posterior distributions were examined by Coles and Tawn (1996). Coles and Powell (1996) discussed the modeling ideas using approximation techniques as well as exploring the feasibility of performing Bayesian analysis when ML method fails. A conjugate prior distribution for Gumbel parameters were considered by Chechile (2001) and provided the solution for posterior normalization constant along with the marginal distribution for scale parameter. Record value based analysis were studied by Mousa *et al.* (2002) for estimation of location and scale parameters of Gumbel distribution. The estimators of parameters, reliability, and failure rate functions of Gumbel distribution were derived by Al-Aboud (2009) using type II censoring and Lindley approximation. The estimators are derived under both symmetric and asymmetric loss functions such as squared error loss, linex loss, and entropy loss. They observed that linex loss and entropy loss functions are sensitive to parameter values. A non-informative prior distribution for the location parameter and three different prior distributions for the scale parameter were considered in Vidal (2014). Under these conditions the posterior distribution of location and scale parameters, posterior modes, expected values, quantiles, and credibility intervals are also derived.

Recently, Yilmaz *et al.* (2021) compared various estimation techniques to determine the best estimators of Gumbel parameters including classical and Bayesian methods. They

considered maximum likelihood, moments, least square, weighted least square, ordinary least square, percentile, L-moments, trimmed L-moments, and Bain and Engelhardt's method of estimation. Bayesian parameter estimation was considered under squared error loss function and assumed normal prior for location and gamma prior for scale parameters. To obtain Bayes estimators they used Lindley's approximation technique and Markov Chain Monte Carlo (MCMC) method. In accordance with them, the Bayes method of estimation has better performance among all other classical methods, and suggest Lindley's approximation techniques over MCMC. It is remarkable that most of the Bayesian inference approaches have been developed under squared error loss function which is symmetric in nature, and the posterior mean will be the corresponding estimate. However, squared error loss may be inappropriate in many cases. The aim of this work is to obtain the Bayes estimators of scale parameter of Gumbel distribution under a class of loss functions including the squared error loss. The Bayes estimators under various losses and various priors are also compared. The rest of the paper is arranged as follows. In Section 2, we list the various loss functions and their corresponding Bayes estimators that we come across in the later sections of this paper. A new class of loss function is introduced in this section and related estimators are derived. Bayesian estimation procedures using Lindley's approximation method and MCMC method are presented in Section 3. A Monte Carlo simulation study is performed and included in Section 4. The findings in Section 3 are also applied to a real life data in Section 5 and a brief conclusion is given in Section 6.

2. Bayesian estimators under various loss functions

The Bayesian method of estimation is an old method of estimation when more additional information about the parameter θ is known prior to observing data. This assumes that θ has some probability distribution over the parameter space Θ , known as prior distribution, $\pi(\theta)$. When data $\mathbf{X} = \mathbf{x}$ is observed, the prior distribution can be updated with the sample information to obtain posterior distribution $\pi(\theta | \mathbf{x})$ using Bayes rule as follows,

$$\pi(\theta | \mathbf{x}) = \frac{L(\mathbf{x}; \theta)\pi(\theta)}{m(\mathbf{x})},$$

where $m(\mathbf{x}) = \int_{\Theta} L(\mathbf{x}; \theta)\pi(\theta)$ is the marginal distribution of $\mathbf{x}$ and $L(\mathbf{x}; \theta)$ is the likelihood function. Another additional information about parameter θ is the consequences of our decision known as the loss function, $L(\theta, \hat{\theta})$, which is a function that assigns a penalty when $\hat{\theta}$ is an estimator for true parameter θ . Then the Bayes estimator under the loss $L(\theta, \hat{\theta})$ is that $\hat{\theta}$ which minimizes the Bayes risk function defined by,

$$r(\theta, \hat{\theta}) = \int_{\Theta} \int_{\mathbf{x}} L(\theta, \hat{\theta})L(\mathbf{x}; \theta)\pi(\theta)d\mathbf{x}d\theta.$$

Since $L(\mathbf{x}; \theta)\pi(\theta) = \pi(\theta | \mathbf{x})m(\mathbf{x})$,

$$r(\theta, \hat{\theta}) = \int_{\Theta} \int_{\mathbf{x}} L(\theta, \hat{\theta})\pi(\theta | \mathbf{x})m(\mathbf{x})d\mathbf{x}d\theta.$$

Minimum of this integral is the same as the minimum of the integral,

$$\int_{\Theta} L(\theta, \hat{\theta})\pi(\theta | \mathbf{x})d\theta, \tag{2}$$

which is known as the posterior expected loss. When doing Bayesian analysis, a number of symmetric and asymmetric loss functions are used in the Bayesian literature. The symmetric loss functions are useful when the positive and negative errors or the over estimation and under estimation are treated equally. In contrast, the asymmetric loss functions gives different weights to positive error and negative error of the same magnitude. Compared with symmetric loss, asymmetric losses have more applications in real life. See for more details, Zellner (1986), Varian (1975), Calabria and Pulcini (1994), Norstrom (1996), and Berger (2013). Some of the loss functions relevant to this paper and corresponding Bayes estimators that minimize the posterior expected loss given in (2), are given in the next section.

2.1. Various loss functions and corresponding estimators

Squared error loss function (sqlf) is an extensively used loss function in Bayesian analysis because of its simplicity and does not yield any extensive calculations. The sqlf is given in the form,

$$L_s(\theta, \hat{\theta}) = (\theta - \hat{\theta})^2. \quad (3)$$

Under this loss function mean of the posterior distribution is the Bayes estimator of θ , i.e,

$$\hat{\theta}_s = E_{\pi(\theta|\mathbf{x})}(\theta | \mathbf{x}) = E(\theta | \mathbf{x}). \quad (4)$$

The squared error loss can be generalized to the weighted squared error loss and has the form,

$$L_{ws}(\theta, \hat{\theta}) = w(\theta)(\theta - \hat{\theta})^2,$$

where $w(\theta)$ is the weight function. The Bayes estimate $\hat{\theta}_{ws}$ under this loss is given by,

$$\hat{\theta}_{ws} = \frac{E(\theta w(\theta) | \mathbf{x})}{E(w(\theta) | \mathbf{x})}.$$

Both of these loss functions are symmetric in nature. As an alternative to symmetric loss, Norstrom (1996) defined a specific class of asymmetric loss functions called precautionary loss functions (pqlf) and is given by,

$$L_p(\theta, \hat{\theta}) = \frac{(\theta - \hat{\theta})^2}{\hat{\theta}^k} w(\theta), \quad 0 \leq k \leq 2, w(\theta) > 0, \quad (5)$$

where $w(\theta)$ is an arbitrary weight function and the constant $k \leq 2$ ensures that the cost increases as the difference $\theta - \hat{\theta}$ grows, so that k is known as the precautionary index. When $k = 0$ and $w(\theta) = 1$, the precautionary loss reduces to the squared error loss and when $k = 1$ it reduces to the weighted squared error loss. Norstrom showed that the Bayes estimator, $\hat{\theta}_p$ corresponds to precautionary loss has the form,

$$\hat{\theta}_p = \frac{1}{\Psi_1(x)} \left[\bar{k} + \sqrt{\bar{k}^2 + \Psi_1(x)\Psi_2(x)} \right], \quad (6)$$

where $\Psi_1(x) = (1 + \bar{k})E(w(\theta) | \mathbf{x})/E(\theta w(\theta) | \mathbf{x})$, $\Psi_2(x) = (1 - \bar{k})E(\theta^2 w(\theta) | \mathbf{x})/E(\theta w(\theta) | \mathbf{x})$, and $\bar{k} = 1 - k$. When $k = 1$ and $w(\theta) = 1$, the loss function in equation (5) will reduces to

$$L_p(\theta, \hat{\theta}) = \frac{(\theta - \hat{\theta})^2}{\hat{\theta}}, \quad (7)$$

and the Bayes estimator corresponds to this loss is,

$$\hat{\theta}_p = \sqrt{E(\theta^2 | \mathbf{x})}, \quad (8)$$

the square root of the second moment of the posterior distribution. Varian (1975) introduced another asymmetric loss function called linex loss and Zellner (1986) studied its properties and used it for estimating the mean of normal random variate. The linex loss has the form,

$$L_l(\theta, \hat{\theta}) = b(e^{a(\hat{\theta}-\theta)} - a(\hat{\theta} - \theta) - 1), \quad a \neq 0, b > 0. \quad (9)$$

This function rises approximately exponential on one side of zero and approximately linear on the other side. The linex loss reduces to squared error loss if $|a| \rightarrow 0$. Under the linex loss, the Bayes estimator, $\hat{\theta}_l$ is given by,

$$\hat{\theta}_l = -\frac{1}{a} \log E(e^{-a\theta} | \mathbf{x}).$$

For $a = 1$,

$$\hat{\theta}_l = -\log E(e^{-\theta} | \mathbf{x}). \quad (10)$$

As an alternative to linex loss, Calabria and Pulcini (1994) considered a loss function called general entropy loss and has the form,

$$L_e(\theta, \hat{\theta}) = \left(\frac{\hat{\theta}}{\theta}\right)^p - p \log\left(\frac{\hat{\theta}}{\theta}\right) - 1, \quad (11)$$

where $p \neq 0$ is the shape parameter. This is a generalization of the entropy loss used by Dey *et al.* (1986), when the shape parameter p in equation (11) is equal to 1. When $p > 0$, a positive error causes more serious consequences than a negative error and vice versa. The Bayes estimator, $\hat{\theta}_e$ under this loss is,

$$\hat{\theta}_e = [E(\theta^{-p} | \mathbf{x})]^{-1/p},$$

and for $p = 1$,

$$\hat{\theta}_e = [E(\theta^{-1} | \mathbf{x})]^{-1}. \quad (12)$$

That is, $\hat{\theta}$ is the harmonic mean of the posterior distribution, and the Bayes estimator under this loss is the same as that of weighted squared error loss with $w(\theta) = 1/\theta$ and when $p = -1$ the estimator coincides with the estimator under squared error loss function given in equation (3). El-Sayyad (1967) considered another loss function in the form,

$$L_{el}(\theta, \hat{\theta}) = w(\theta)(\log \hat{\theta} - \log \theta)^2, \quad (13)$$

the Bayes estimator under this loss, $\hat{\theta}_{el}$ is given by,

$$\hat{\theta}_{el} = e^{\frac{E(w(\theta) \log \theta | \mathbf{x})}{E(w(\theta) | \mathbf{x})}},$$

and for $w(\theta) = 1$,

$$\hat{\theta}_{el} = e^{E(\log \theta | \mathbf{x})}. \quad (14)$$

In the discussion above we saw that mean and the square root of the second order moment of the posterior distribution are the Bayes estimator under the loss functions sqfl and pqfl respectively. In the next section we introduce a new class of loss function which is a function of sqfl and pqfl and try to get Bayes estimator in terms of the first and second order moments of the posterior distribution.

2.2. A generalized class of loss functions

Generally loss functions are non-negative functions that increase to infinity when the distance between θ and $\hat{\theta}$ increases and decrease to zero when the distance between θ and $\hat{\theta}$ decreases. Further most of them are smooth convex functions. Here we define a new class of loss functions which is a function of squared error loss defined in equation (3) and precautionary loss defined in equation (7) discussed in the Section 2.1. Let $g(\cdot)$ be a real-valued monotone function, the new class of loss functions $L_g(\theta, \hat{\theta})$ is defined as,

$$L_g(\theta, \hat{\theta}) = g^{-1}(\lambda g(L_s(\theta, \hat{\theta})) + (1 - \lambda)g(L_p(\theta, \hat{\theta}))), \quad \lambda \in [0, 1], \quad (15)$$

where $L_s(\theta, \hat{\theta})$ is the sqf in (3) and $L_p(\theta, \hat{\theta})$ is the pqlf in (7). Note that when $\lambda = 0$, $L_g(\theta, \hat{\theta})$ reduces to $L_p(\theta, \hat{\theta})$ in (7) and when $\lambda = 1$ it reduces to $L_s(\theta, \hat{\theta})$ in (3). The Bayes estimators under these losses are discussed in Section 2.1. Hence, here onward we consider only the case $\lambda \in (0, 1)$ in equation (15). We consider three cases where $g(\cdot)$ is $g_1(z) = \log z$, $g_2(z) = 1/z$ and $g_3(z) = z$ and discuss these in the following three cases.

Case 1

Let $g(z) = g_1(z) = \log z$, then $g_1^{-1}(z) = e^z$ and $L_g(\theta, \hat{\theta})$ in (15) becomes,

$$\begin{aligned} L_{g_1}(\theta, \hat{\theta}) &= g_1^{-1}(\lambda g_1(L_s(\theta, \hat{\theta})) + (1 - \lambda)g_1(L_p(\theta, \hat{\theta}))) \\ &= \frac{(\theta - \hat{\theta})^2}{\hat{\theta}^{1-\lambda}}. \end{aligned}$$

This is a special case of precautionary loss function defined in equation (5), where $k = 1 - \lambda$ and $w(\theta) = 1$. The Bayes estimator corresponds to this loss can be derive from (6) by substituting $k = 1 - \lambda$ and $w(\theta) = 1$ as,

$$\hat{\theta}_{g_1} = \frac{1}{\Psi_1(x)} \left[\lambda + \sqrt{\lambda^2 + \Psi_1(x)\Psi_2(x)} \right],$$

with $\Psi_1(x) = (1 + \lambda)/E(\theta | \mathbf{x})$ and $\Psi_2(x) = (1 - \lambda)E(\theta^2 | \mathbf{x})/E(\theta | \mathbf{x})$. That is,

$$\hat{\theta}_{g_1} = \frac{\lambda E(\theta | \mathbf{x}) + \sqrt{E(\theta^2 | \mathbf{x}) + \lambda^2((E(\theta | \mathbf{x}))^2 - E(\theta^2 | \mathbf{x}))}}{\lambda + 1}. \quad (16)$$

That is $\hat{\theta}_{g_1}$ can be expressed as the combinations of first order and second order moments of the posterior distribution. For comparison purpose and simplicity we denote these posterior moments as $u = E(\theta | \mathbf{x})$ and $v = E(\theta^2 | \mathbf{x})$ and the Bayes estimator in equation (16) can be rewritten as,

$$\hat{\theta}_{g_1} = \frac{\lambda u + \sqrt{v + \lambda^2(u^2 - v)}}{\lambda + 1}. \quad (17)$$

Case 2

Let $g(z) = g_2(z) = 1/z$, then $g_2^{-1}(z) = 1/z$ and $L_g(\theta, \hat{\theta})$ in (15) becomes,

$$\begin{aligned} L_{g_2}(\theta, \hat{\theta}) &= g_2^{-1}(\lambda g_2(L_s(\theta, \hat{\theta})) + (1 - \lambda)g_2(L_p(\theta, \hat{\theta}))) \\ &= \frac{(\theta - \hat{\theta})^2}{\lambda + (1 - \lambda)\hat{\theta}}. \end{aligned}$$

The loss $L_{g_2}(\theta, \hat{\theta})$ does not correspond to any existing form of losses discussed earlier, the Bayes estimator under this loss can be derived by minimizing posterior expected loss given in equation (2) and can be derived as,

$$\hat{\theta}_{g_2} = \frac{\lambda - \sqrt{v + 2u\lambda - 2v\lambda + \lambda^2 - 2u\lambda^2 + v\lambda^2}}{\lambda - 1}, \quad (18)$$

where $u = E(\theta \mid \mathbf{x})$ and $v = E(\theta^2 \mid \mathbf{x})$.

Case 3

Let $g(z) = g_3(z) = z$ then $g_3^{-1}(z) = z$ and $L_g(\theta, \hat{\theta})$ in (15) becomes,

$$\begin{aligned} L_{g_3}(\theta, \hat{\theta}) &= g_3^{-1}(\lambda g_3(L_s(\theta, \hat{\theta})) + (1 - \lambda)g_3(L_p(\theta, \hat{\theta}))) \\ &= \lambda(\theta - \hat{\theta})^2 + (1 - \lambda)\frac{(\theta - \hat{\theta})^2}{\hat{\theta}}. \end{aligned}$$

Again by minimizing posterior expected loss given in equation (2) we get the Bayes estimator $\hat{\theta}_{g_3}$ as,

$$\begin{aligned} \hat{\theta}_{g_3} &= \frac{1 - \lambda - 2u\lambda}{6\lambda} + \frac{1 - \lambda - 2u\lambda^2}{3 \times 2^{2/3}\lambda(H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2})^{1/3}} + \\ &\quad \frac{1}{6 \times 2^{1/3}\lambda}(H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2})^{1/3}, \end{aligned} \quad (19)$$

where $H_1(u, v, \lambda) = -2 + 6\lambda + 12u\lambda - 6\lambda^2 - 24u\lambda^2 - 24u^2\lambda^2 + 108v\lambda^2 + 2\lambda^3 + 12u\lambda^3 + 24u^2\lambda^3 + 16u^3\lambda^3 - 108v\lambda^3$, $H_2(u, v, \lambda) = -4(1 - \lambda - 2u\lambda)^6$, $u = E(\theta \mid \mathbf{x})$ and $v = E(\theta^2 \mid \mathbf{x})$.

The estimators $\hat{\theta}_{g_1}$, $\hat{\theta}_{g_2}$ have a simple form with the combination of first order moment and second order moment of posterior distribution. Although $\hat{\theta}_{g_3}$ has a lengthy expression of first order moment and second order moment, it can be calculated easily. The derivation of these estimators are included in the appended Annexures. The posterior risks under different loss functions are derived and given in Table 1, where $\hat{\theta}_{g_1}$, $\hat{\theta}_{g_2}$ and $\hat{\theta}_{g_3}$ have the expression given in equations (17), (18) and (19) respectively. In the next section we find the Bayes estimators of location and scale parameters of Gumbel distribution under these loss functions using two different prior information.

Table 1: Posterior risk under various loss functions

Loss function	Posterior risk
$L_s(\theta, \hat{\theta})$	$E(\theta^2 x) - (E(\theta x))^2$
$L_p(\theta, \hat{\theta})$	$2\left(\sqrt{E(\theta^2 x)} - E(\theta x)\right)$
$L_l(\theta, \hat{\theta})$	$\log E(e^{-\theta} x) + E(\theta x)$
$L_e(\theta, \hat{\theta})$	$E(\log \theta x) + \log E(\theta^{-1} x)$
$L_{el}(\theta, \hat{\theta})$	$E((\log \theta)^2) - (E(\log \theta))^2$
$L_{g_1}(\theta, \hat{\theta})$	$\hat{\theta}_{g_1}^{\lambda-1} E(\theta^2 x) - 2\hat{\theta}_{g_1}^\lambda E(\theta x) + \hat{\theta}_{g_1}^{\lambda+1}$
$L_{g_2}(\theta, \hat{\theta})$	$(E(\theta^2 x) - 2\hat{\theta}_{g_2} E(\theta x) + \hat{\theta}_{g_2}^2)/(\lambda + (1 - \lambda)\hat{\theta}_{g_2})$
$L_{g_3}(\theta, \hat{\theta})$	$\lambda(E(\theta^2 x) - 2\hat{\theta}_{g_3} E(\theta x) + \hat{\theta}_{g_3}^2) + ((1 - \lambda)(E(\theta x) + \hat{\theta}_{g_3}^2))/(\hat{\theta}_{g_3})$

3. Bayes estimator of Gumbel scale parameter

We derive the Bayes estimator of the Gumbel scale parameter, with the location parameter set to zero, under various loss functions given in Section 2. The probability density function of Gumbel distribution with location zero has the form,

$$f(x; \theta) = \frac{1}{\theta} e^{-\left(\frac{x}{\theta}\right)} e^{-e^{-\left(\frac{x}{\theta}\right)}}, \quad -\infty < x < \infty,$$

where $\theta > 0$ is the scale parameter. Based on a sample, $\mathbf{x} = (x_1, x_2, \dots, x_n)$, the likelihood function is given by,

$$L(\mathbf{x}; \theta) = \frac{1}{\theta^n} e^{-\sum_{i=1}^n \left(\frac{x_i}{\theta}\right)} e^{-\sum_{i=1}^n e^{-\left(\frac{x_i}{\theta}\right)}}. \quad (20)$$

For a prior distribution $\pi(\theta)$, the posterior distribution can be derived as,

$$\pi(\theta | \mathbf{x}) \propto L(\mathbf{x}; \theta) \pi(\theta).$$

Here we assume two different prior distributions for θ . As the first case, let $\pi(\theta) \propto \theta^{-1}$, a Jeffreys prior. It follows that the posterior distribution of θ is given by,

$$\pi(\theta | \mathbf{x}) \propto \frac{1}{\theta^{n+1}} e^{-\sum_{i=1}^n \left(\frac{x_i}{\theta}\right)} e^{-\sum_{i=1}^n e^{-\left(\frac{x_i}{\theta}\right)}}. \quad (21)$$

Now we assume gamma prior for θ as the second case. That is $\pi(\theta) \propto \theta^{c_1-1} e^{-c_2\theta}$, where c_1 , and c_2 are the hyper parameters. Then the posterior distribution corresponds to this prior has the form,

$$\pi(\theta | \mathbf{x}) \propto \theta^{c_1-1} e^{-c_2\theta} \frac{1}{\theta^n} e^{-\sum_{i=1}^n \left(\frac{x_i}{\theta}\right)} e^{-\sum_{i=1}^n e^{-\left(\frac{x_i}{\theta}\right)}}. \quad (22)$$

In order to find the Bayes estimator of θ , the scale parameter of Gumbel distribution, under the loss functions discussed in Section 2, we have to find the posterior expectations, $E(\theta | \mathbf{x})$, $E(\theta^2 | \mathbf{x})$, $E(e^{-\theta} | \mathbf{x})$, $E(\theta^{-1} | \mathbf{x})$ and $E(\log \theta | \mathbf{x})$. Whereas the posterior distributions

under two assumed priors have no closed form, we use two approximation techniques to evaluate these expectations, following Lindley (1980) method and the Markov chain Monte Carlo method in this section. Yilmaz *et al.* (2021) applied these methods to acquire Bayes estimates for the parameters of the Gumbel distribution under the squared error loss function and Abbas *et al.* (2019) used to derive Bayes estimates of power Lindley distribution. A brief overview of these two methods are provided, along with the derivation of the posterior expectations for the Gumbel scale parameter in the following subsections.

3.1. Lindley approximation

The work of Lindley (1980) provides an asymptotic solution for the ratio of two integrals commonly encountered while evaluating the posterior expectations. Let $q(\theta)$ be any function of θ then,

$$E(q(\theta) | x) \approx \frac{\int q(\theta) L(\mathbf{x}; \theta) \pi(\theta) d\theta}{\int L(\mathbf{x}; \theta) \pi(\theta) d\theta}, \quad (23)$$

where $\pi(\theta)$ is the prior distribution and $L(\mathbf{x}; \theta)$ is the likelihood function. According to Lindley (1980), this ratio of integrals can be approximated by,

$$E(q(\theta) | x) \approx q(\tilde{\theta}) + 0.5(\tilde{q}_{11} + 2\tilde{q}_1\tilde{\pi}_1)\tilde{\sigma}_{11} + 0.5(\tilde{q}_1\tilde{L}_{111}\tilde{\sigma}_{11}^2), \quad (24)$$

where $\tilde{\theta}$ is the ML estimate of θ ,

$$\begin{aligned} \tilde{q}_1 &= \frac{\partial q(\tilde{\theta})}{\partial \tilde{\theta}}, \quad \tilde{q}_{11} = \frac{\partial^2 q(\tilde{\theta})}{\partial \tilde{\theta}^2}, \quad \tilde{\pi}_1 = \frac{\partial \log \pi(\tilde{\theta})}{\partial \tilde{\theta}}, \\ \tilde{L}_{111} &= \frac{\partial^3 \log L(\mathbf{x}, \tilde{\theta})}{\partial \tilde{\theta}^3}, \quad \tilde{L}_{11} = \frac{\partial^2 \log L(\mathbf{x}, \tilde{\theta})}{\partial \tilde{\theta}^2}, \quad \tilde{\sigma}_{11} = \frac{-1}{\tilde{L}_{11}}. \end{aligned}$$

The approximation of posterior expectations given in Section 2 for Jeffreys prior can be achieved by utilizing equation (24) in the following manner.

$$\begin{aligned} E(\theta | \mathbf{x}) &\approx \tilde{\theta} + \tilde{\pi}_1\tilde{\sigma}_{11} + 0.5(\tilde{L}_{111}\tilde{\sigma}_{11}^2) \\ E(\theta^2 | \mathbf{x}) &\approx \tilde{\theta}^2 + (1 + 2\tilde{\theta}\tilde{\pi}_1)\tilde{\sigma}_{11} + \tilde{\theta}\tilde{L}_{111}\tilde{\sigma}_{11}^2 \\ E(e^{-\theta} | \mathbf{x}) &\approx e^{-\tilde{\theta}} + 0.5(e^{-\tilde{\theta}} - 2e^{-\tilde{\theta}}\tilde{\pi}_1)\tilde{\sigma}_{11} - 0.5(e^{-\tilde{\theta}}\tilde{L}_{111}\tilde{\sigma}_{11}^2) \\ E(\theta^{-1} | \mathbf{x}) &\approx \tilde{\theta}^{-1} + (\tilde{\theta}^{-3} - \tilde{\theta}^{-2}\tilde{\pi}_1)\tilde{\sigma}_{11} - 0.5(\tilde{\theta}^{-2}\tilde{L}_{111}\tilde{\sigma}_{11}^2) \\ E(\log \theta | \mathbf{x}) &\approx \tilde{\theta}^{-1} + 0.5(-\tilde{\theta}^{-2} + 2\tilde{\theta}^{-1}\tilde{\pi}_1)\tilde{\sigma}_{11} - 0.5(\tilde{\theta}^{-1}\tilde{L}_{111}\tilde{\sigma}_{11}^2), \end{aligned}$$

where,

$$\begin{aligned} \tilde{L}_{111} &= \frac{-2n}{\tilde{\theta}^3} + 6 \sum_{i=1}^n \frac{x_i}{\tilde{\theta}^4} + 6 \sum_{i=1}^n e^{-\frac{x_i}{\tilde{\theta}}} \frac{x_i}{\tilde{\theta}^4} - 6 \sum_{i=1}^n e^{-\frac{x_i}{\tilde{\theta}}} \frac{x_i^2}{\tilde{\theta}^5} + \sum_{i=1}^n e^{-\frac{x_i}{\tilde{\theta}}} \frac{x_i^3}{\tilde{\theta}^6}, \\ \tilde{L}_{11} &= \frac{n}{\tilde{\theta}^2} - 2 \sum_{i=1}^n \frac{x_i}{\tilde{\theta}^3} - 2 \sum_{i=1}^n e^{-\frac{x_i}{\tilde{\theta}}} \frac{x_i}{\tilde{\theta}^3} + \sum_{i=1}^n e^{-\frac{x_i}{\tilde{\theta}}} \frac{x_i^2}{\tilde{\theta}^3}, \\ \tilde{\sigma}_{11} &= \frac{-1}{\tilde{L}_{11}}, \quad \tilde{\pi}_1 = \frac{-1}{\tilde{\theta}}. \end{aligned}$$

The Bayes estimate corresponds to each loss function can now be evaluated by plug in these approximations of posterior expectations in equations given in Section 2 accordingly. Similarly one can evaluate estimates for scale parameter θ under gamma prior.

3.2. MCMC method

MCMC method was first introduced by Metropolis *et al.* (1953) and Hastings (1970). The Metropolis-Hastings algorithm is among the straightforward MCMC techniques used to generate samples from distributions that might otherwise be challenging to sample from. Here we use the Metropolis-Hastings algorithm to generate sample from posterior distribution. Let $\pi(\cdot | \mathbf{x})$ be the posterior distribution and consider normal distribution as a proposal distribution, the algorithm can be framed as follows,

1. Initialize $\theta_1 = \tilde{\theta}$ and set $t = 1$
2. Repeat : $t = 1$ to N
 - (a) Generate y from $I(\theta > 0)N(\theta_t, 0.1)$
 - (b) Generate U from Uniform $(0,1)$ and if

$$U \leq \frac{\pi(y | \mathbf{x})}{\pi(\theta_t | \mathbf{x})}$$

accept y and set $\theta_{t+1} = y$ otherwise set $\theta_{t+1} = \theta_t$

- (c) increment t .

Using this algorithm we can generate observations for θ from posteriors in equation (21) corresponds to Jeffreys prior and in equation (22) for gamma prior. Thus we can calculate the posterior expectations after discarding first b (burn in period) observations by the approximation,

$$E(q(\theta)|\mathbf{x}) = \frac{1}{N-b} \sum_{t=1}^{N-b} q(\theta_t).$$

Substituting $q(\theta)$ as θ , θ^2 , $e^{-\theta}$, θ^{-1} and $\log \theta$ the posterior expectations can be obtained. The Bayes estimates corresponding to each loss function can now be evaluated by substituting these approximations of posterior expectations into the equations provided in Section 2 accordingly.

4. Simulation and calculations

Extensive analysis is conducted through Monte Carlo simulation to estimate and compare the scale parameter of the Gumbel distribution under various loss functions and priors. We simulate 1000 samples of sizes 30, 50, and 100 using the inverse transformation method. For the gamma prior, we set the hyper parameters as $c_1 = c_2 = 1$. The estimators under the loss functions $L_{g_1}(\theta, \hat{\theta})$, $L_{g_2}(\theta, \hat{\theta})$ and $L_{g_3}(\theta, \hat{\theta})$ were derived using Mathematica software. To evaluate the estimates $\hat{\theta}_{g_1}$, $\hat{\theta}_{g_2}$ and $\hat{\theta}_{g_3}$, we assume some arbitrary values for λ ranging from 0 to 1, specifically tabulated for $\lambda = \{0.25, 0.5, 0.75\}$ for simplicity. Also we compute the Bayes estimates based on 20000 MCMC samples and burn in period b is set as 5000. The maximum likelihood estimate and all other computations are done using the R software. The simulated estimates, posterior risk and absolute bias, for $\theta = 1$ under Jeffreys prior and gamma prior using Lindley approximation are presented in Table 2 and

Table 2: The Bayesian estimates and respective simulated absolute bias and posterior risk for $\theta = 1$ under Jeffreys prior and gamma prior using Lindleys method

n	Estimator	Jeffreys prior			Gamma prior		
		Estimate	Posterior risk	Absolute bias	Estimate	Posterior risk	Absolute bias
30	$\hat{\theta}_s$	1.084661	0.009769	0.084661	0.974197	0.021422	0.025803
	$\hat{\theta}_p$	1.089181	0.009039	0.089181	0.985135	0.021876	0.014865
	$\hat{\theta}_l$	1.079026	0.004773	0.079026	0.963569	0.001344	0.036431
	$\hat{\theta}_e$	1.071072	0.082695	0.071072	0.952852	0.147326	0.047149
	$\hat{\theta}_{el}$	0.986032	0.067908	0.013967	1.104104	0.063667	0.104104
	$\hat{\theta}_{g1,.25}$	1.088051	0.009220	0.088051	0.982404	0.021784	0.017596
	$\hat{\theta}_{g1,.5}$	1.086922	0.009402	0.086922	0.979674	0.021677	0.020326
	$\hat{\theta}_{g1,.75}$	1.085793	0.009585	0.085793	0.976940	0.021557	0.023060
	$\hat{\theta}_{g2,.25}$	1.018206	0.020074	0.018206	0.982366	0.021783	0.017634
	$\hat{\theta}_{g2,.5}$	1.015716	0.020152	0.015716	0.979616	0.021675	0.020384
	$\hat{\theta}_{g2,.75}$	1.013205	0.020218	0.013205	0.976892	0.021555	0.023108
	$\hat{\theta}_{g3,.25}$	1.087983	0.009226	0.087983	0.983520	0.021823	0.016481
	$\hat{\theta}_{g3,.5}$	1.086835	0.009410	0.086835	0.979731	0.021679	0.020269
	$\hat{\theta}_{g3,.75}$	1.085729	0.009591	0.085729	0.976990	0.021558	0.023010
50	$\hat{\theta}_s$	1.048778	0.009065	0.048778	0.983729	0.012326	0.016271
	$\hat{\theta}_p$	1.053094	0.008630	0.053094	1.009107	0.012302	0.009107
	$\hat{\theta}_l$	1.043957	0.008190	0.043956	0.980286	0.009294	0.019718
	$\hat{\theta}_e$	1.038437	0.044702	0.038437	0.974213	0.082056	0.025787
	$\hat{\theta}_{el}$	0.993037	0.025964	0.006963	1.057525	0.024007	0.057525
	$\hat{\theta}_{g1,.25}$	1.052015	0.008741	0.052015	0.991104	0.012470	0.008896
	$\hat{\theta}_{g1,.5}$	1.050937	0.008850	0.050937	0.989543	0.012440	0.010457
	$\hat{\theta}_{g1,.75}$	1.049858	0.008958	0.049858	0.987982	0.012405	0.012018
	$\hat{\theta}_{g2,.25}$	1.011043	0.012073	0.011043	0.991093	0.012470	0.008907
	$\hat{\theta}_{g2,.5}$	1.009542	0.012103	0.009542	0.989527	0.012440	0.010473
	$\hat{\theta}_{g2,.75}$	1.008032	0.012129	0.008032	0.987968	0.012404	0.012032
	$\hat{\theta}_{g3,.25}$	1.051974	0.008743	0.051974	0.991114	0.012470	0.008886
	$\hat{\theta}_{g3,.5}$	1.050884	0.008852	0.050884	0.989560	0.012440	0.010440
	$\hat{\theta}_{g3,.75}$	1.049820	0.008960	0.049820	0.987996	0.012405	0.012004
100	$\hat{\theta}_s$	1.021974	0.005365	0.021974	0.991169	0.006144	0.008831
	$\hat{\theta}_p$	1.024596	0.005243	0.024596	0.994263	0.006189	0.005737
	$\hat{\theta}_l$	1.019216	0.002123	0.019216	0.988110	0.002443	0.011890
	$\hat{\theta}_e$	1.016277	0.021730	0.016277	0.985059	0.040463	0.014941
	$\hat{\theta}_{el}$	0.994431	0.018080	0.005569	1.025735	0.017843	0.025735
	$\hat{\theta}_{g1,.25}$	1.023941	0.005274	0.023941	0.993490	0.006180	0.006510
	$\hat{\theta}_{g1,.5}$	1.023286	0.005305	0.023286	0.992716	0.006169	0.007284
	$\hat{\theta}_{g1,.75}$	1.022630	0.005336	0.022630	0.991943	0.006157	0.008057
	$\hat{\theta}_{g2,.25}$	1.003335	0.006091	0.003335	0.993486	0.006180	0.006514
	$\hat{\theta}_{g2,.5}$	1.002574	0.006095	0.002574	0.992711	0.006169	0.007289
	$\hat{\theta}_{g2,.75}$	1.001813	0.006099	0.001813	0.991938	0.006157	0.008062
	$\hat{\theta}_{g3,.25}$	1.023929	0.005275	0.023929	0.993493	0.006180	0.006507
	$\hat{\theta}_{g3,.5}$	1.023271	0.005306	0.023271	0.992722	0.006169	0.007278
	$\hat{\theta}_{g3,.75}$	1.022619	0.005336	0.022619	0.991948	0.006157	0.008052

using MCMC method in Table 3. From these tables, we can observe that the posterior risk and absolute bias under Lindley method are lesser than that of MCMC method. Hence, we can conclude that Lindley approximation performs slightly better than MCMC method,

Table 3: The Bayesian estimates and respective simulated absolute bias and posterior risk for $\theta = 1$ under Jeffreys prior and gamma prior using MCMC method

n	Estimator	Jeffreys prior			Gamma prior		
		Estimate	Posterior risk	Absolute bias	Estimate	Posterior risk	Absolute bias
30	$\hat{\theta}_s$	1.024823	0.023535	0.024823	1.080640	0.029746	0.080640
	$\hat{\theta}_p$	1.036099	0.022551	0.036099	1.094007	0.026734	0.094007
	$\hat{\theta}_l$	1.013464	0.011360	0.013464	1.055327	0.014247	0.055327
	$\hat{\theta}_e$	1.003360	0.010495	0.003360	1.066392	0.011725	0.066392
	$\hat{\theta}_{el}$	1.013918	0.021217	0.013918	1.067756	0.023725	0.067756
	$\hat{\theta}_{g1,.25}$	1.033283	0.022785	0.033283	1.090669	0.027430	0.090669
	$\hat{\theta}_{g1,.5}$	1.030469	0.023027	0.030469	1.087332	0.028163	0.087332
	$\hat{\theta}_{g1,.75}$	1.027651	0.023277	0.027651	1.083991	0.028932	0.083991
	$\hat{\theta}_{g2,.25}$	1.033354	0.022749	0.033354	1.090895	0.027353	0.090895
	$\hat{\theta}_{g2,.5}$	1.030568	0.022978	0.030568	1.087652	0.028043	0.087652
	$\hat{\theta}_{g2,.75}$	1.027728	0.023240	0.027728	1.084244	0.028836	0.084244
	$\hat{\theta}_{g3,.25}$	1.033194	0.022821	0.033194	1.090403	0.027520	0.090403
	$\hat{\theta}_{g3,.5}$	1.030370	0.023075	0.030370	1.087019	0.028284	0.087019
	$\hat{\theta}_{g3,.75}$	1.027591	0.023313	0.027591	1.083782	0.029025	0.083782
50	$\hat{\theta}_s$	1.000461	0.012736	0.000461	1.058818	0.015640	0.058818
	$\hat{\theta}_p$	1.006735	0.012548	0.006735	1.066092	0.014547	0.066092
	$\hat{\theta}_l$	0.994210	0.006252	0.005791	1.044764	0.007634	0.044764
	$\hat{\theta}_e$	0.988231	0.006116	0.011769	1.051185	0.006640	0.051185
	$\hat{\theta}_{el}$	0.994293	0.012300	0.005707	1.051711	0.013385	0.051711
	$\hat{\theta}_{g1,.25}$	1.005168	0.012589	0.005168	1.064275	0.014805	0.064275
	$\hat{\theta}_{g1,.5}$	1.003601	0.012634	0.003601	1.062458	0.015073	0.062458
	$\hat{\theta}_{g1,.75}$	1.002032	0.012683	0.002032	1.060640	0.015351	0.060640
	$\hat{\theta}_{g2,.25}$	1.005177	0.012576	0.005177	1.064361	0.014780	0.064361
	$\hat{\theta}_{g2,.5}$	1.003615	0.012616	0.003615	1.062578	0.015038	0.062578
	$\hat{\theta}_{g2,.75}$	1.002045	0.012669	0.002045	1.060734	0.015324	0.060734
	$\hat{\theta}_{g3,.25}$	1.005151	0.012603	0.005151	1.064177	0.014830	0.064177
	$\hat{\theta}_{g3,.5}$	1.003586	0.012652	0.003586	1.062339	0.015107	0.062339
	$\hat{\theta}_{g3,.75}$	1.002027	0.012697	0.002027	1.060559	0.015376	0.060559
100	$\hat{\theta}_s$	0.999136	0.006171	0.000864	1.026637	0.006858	0.026637
	$\hat{\theta}_p$	1.002999	0.006126	0.002999	1.029954	0.006634	0.029954
	$\hat{\theta}_l$	0.996880	0.003055	0.003119	1.020079	0.003399	0.020079
	$\hat{\theta}_e$	0.993901	0.003018	0.006099	1.023238	0.003197	0.023238
	$\hat{\theta}_{el}$	0.996901	0.006054	0.003097	1.023345	0.006409	0.023345
	$\hat{\theta}_{g1,.25}$	1.002234	0.006136	0.002234	1.029125	0.006688	0.029125
	$\hat{\theta}_{g1,.5}$	1.001468	0.006146	0.001468	1.028296	0.006743	0.028296
	$\hat{\theta}_{g1,.75}$	1.000702	0.006158	0.000702	1.027467	0.006800	0.027467
	$\hat{\theta}_{g2,.25}$	1.002236	0.006132	0.002236	1.029144	0.006684	0.029144
	$\hat{\theta}_{g2,.5}$	1.001471	0.006142	0.001471	1.028322	0.006738	0.028322
	$\hat{\theta}_{g2,.75}$	1.000705	0.006154	0.000705	1.027487	0.006796	0.027487
	$\hat{\theta}_{g3,.25}$	1.002230	0.006140	0.002230	1.029105	0.006692	0.029105
	$\hat{\theta}_{g3,.5}$	1.001465	0.006151	0.001465	1.028271	0.006749	0.028271
	$\hat{\theta}_{g3,.75}$	1.000701	0.006161	0.000701	1.027450	0.006804	0.027449

which supports the findings of Yilmaz et al. (2021). As the sample size increases, both methods exhibit nearly identical performance. When comparing Jeffreys prior and gamma prior, Jeffreys prior outperforms the gamma prior for the parameter θ . The posterior risk

Table 4: Table for estimate, posterior risk and absolute bias for various θ under Jeffreys prior and each loss functions

Loss function	Parameter	Estimate	Posterior Risk	Absolute Bias
$L_s(\theta, \hat{\theta})$	0.1	0.15271	0.00086	0.05273
$L_p(\theta, \hat{\theta})$		0.15542	0.00536	0.05542
$L_l(\theta, \hat{\theta})$		0.15231	0.00043	0.05231
$L_e(\theta, \hat{\theta})$		0.14731	0.01827	0.04731
$L_{el}(\theta, \hat{\theta})$		0.15002	0.03628	0.05002
$L_{g1}(\theta, \hat{\theta})$		0.15408	0.00214	0.05408
$L_{g2}(\theta, \hat{\theta})$		0.15310	0.00147	0.05310
$L_{g3}(\theta, \hat{\theta})$		0.15455	0.00199	0.05455
$L_s(\theta, \hat{\theta})$	0.2	0.24213	0.00150	0.04213
$L_p(\theta, \hat{\theta})$		0.24502	0.00578	0.04502
$L_l(\theta, \hat{\theta})$		0.24138	0.00075	0.04138
$L_e(\theta, \hat{\theta})$		0.23618	0.01262	0.03618
$L_{el}(\theta, \hat{\theta})$		0.23917	0.02494	0.03917
$L_{g1}(\theta, \hat{\theta})$		0.24358	0.00293	0.04358
$L_{g2}(\theta, \hat{\theta})$		0.24272	0.00236	0.04272
$L_{g3}(\theta, \hat{\theta})$		0.24443	0.00365	0.04443
$L_s(\theta, \hat{\theta})$	0.5	0.54204	0.00243	0.04204
$L_p(\theta, \hat{\theta})$		0.54435	0.00461	0.04435
$L_l(\theta, \hat{\theta})$		0.54083	0.00122	0.04083
$L_e(\theta, \hat{\theta})$		0.53738	0.00485	0.03738
$L_{el}(\theta, \hat{\theta})$		0.53971	0.00966	0.03971
$L_{g1}(\theta, \hat{\theta})$		0.54320	0.00332	0.04320
$L_{g2}(\theta, \hat{\theta})$		0.54283	0.00314	0.04283
$L_{g3}(\theta, \hat{\theta})$		0.54357	0.00353	0.04357
$L_s(\theta, \hat{\theta})$	2	1.90446	0.00320	0.09554
$L_p(\theta, \hat{\theta})$		1.90536	0.00180	0.09464
$L_l(\theta, \hat{\theta})$		1.90286	0.00160	0.09714
$L_e(\theta, \hat{\theta})$		1.90266	0.00054	0.09734
$L_{el}(\theta, \hat{\theta})$		1.90356	0.00108	0.09644
$L_{g1}(\theta, \hat{\theta})$		1.90491	0.00238	0.09509
$L_{g2}(\theta, \hat{\theta})$		1.90503	0.00227	0.09497
$L_{g3}(\theta, \hat{\theta})$		1.90480	0.00250	0.09520
$L_s(\theta, \hat{\theta})$	5	4.76435	0.00382	0.23565
$L_p(\theta, \hat{\theta})$		4.76476	0.00081	0.23524
$L_l(\theta, \hat{\theta})$		4.76244	0.00191	0.23756
$L_e(\theta, \hat{\theta})$		4.76354	0.00009	0.23646
$L_{el}(\theta, \hat{\theta})$		4.76394	0.00018	0.23606
$L_{g1}(\theta, \hat{\theta})$		4.76455	0.00175	0.23545
$L_{g2}(\theta, \hat{\theta})$		4.76468	0.00133	0.23532
$L_{g3}(\theta, \hat{\theta})$		4.76443	0.00231	0.23558

and bias under the newly defined generalized class of loss functions fall between the posterior risk and absolute bias of squared error and precautionary loss. As λ increases from 0 to 1, the posterior risk varies from posterior risk of precautionary loss to that of squared error

loss. The bias under the new generalized class of loss function also behaves similarly. Consequently, it is observed that when squared error loss overestimates (underestimates) and precautionary loss underestimates (overestimates) the parameters, the new generalized class of loss functions outperforms squared error loss and precautionary loss in terms of absolute bias. For a comprehensive comparison of the loss functions outlined in Section 2, we computed Bayes estimates for various values of θ and determined their corresponding posterior risk and absolute bias under these loss functions. Since Jeffreys prior demonstrates superior performance compared to the gamma prior, the results are presented exclusively for Jeffreys prior under the Lindley method, as shown in Table 4. For smaller parameter values, sqf and linex loss functions exhibit lower posterior risk, while linex loss shows a slightly smaller risk to a certain extent. As the parameter value increases, entropy loss and El-Sayyad loss functions display lower posterior risk; however, entropy loss outperforms the El-Sayyad loss function. Additionally, the posterior risk under sqf increases with the increase of parameter values compared to other loss functions. It's noteworthy that we are not confined to selecting the squared error loss function or the posterior mean as an estimator.

5. Application

To demonstrate the practical implementation of the estimation techniques delineated in this paper, we analyze a dataset provided by Nelson (1982). The dataset pertains to a life-test experiment involving specimens of a specific electrical insulating fluid exposed to a constant voltage stress of 34 KV/minutes. Nelson, in his analysis, presumed a Weibull distribution for the breakdown times. Al-Aboud (2009) explored log-breakdowns times to derive Bayesian estimates for the parameters of the Gumbel distribution. The log-breakdown times are given by,

-1.66073	-0.24846	-0.04082	0.27003	1.02245	1.15057	1.42311
1.54116	1.57898	1.87180	1.99470	2.08069	2.11263	2.48989
3.45789	3.48186	3.52371	3.60305	4.28895		

The Bayes estimates and posterior risk of the scale parameter θ of Gumbel distribution under different loss functions were computed and tabulated in Table 5. It is clear from the table that posterior risk is smaller for scale parameter θ under entropy loss function for both Jeffreys prior and gamma prior. Squared error loss function has greater posterior risk in both cases. The new loss functions $L_{g1}(\theta, \hat{\theta})$, $L_{g2}(\theta, \hat{\theta})$ and $L_{g3}(\theta, \hat{\theta})$ have posterior risk smaller than that of sqf but greater than pqf.

6. Conclusion

For a better choice of loss function while estimating the scale parameters of Gumbel distribution using Bayesian approach we consider different loss functions that are available in literature. It is concluded that entropy loss function has smaller posterior risk among other loss functions. El-Sayyad loss function also performs in similar way. In most of the research work squared error loss function is considered and posterior mean will be the estimator. However here we shows that squared error loss function has greater posterior risk when the parameter values are moderately large. The generalized class of loss function has smaller bias when squared error loss function overestimate(underestimate) and precautionary

Table 5: The Bayesian estimates and posterior risk for θ under Jeffreys prior and gamma prior using for real data

Estimator	Jeffreys prior		Gamma prior	
	Estimate	Posterior risk	Estimate	Posterior risk
$L_s(\theta, \hat{\theta})$	1.1572	0.1958	1.1436	0.2068
$L_p(\theta, \hat{\theta})$	1.0692	0.1759	1.0492	0.1886
$L_l(\theta, \hat{\theta})$	1.2336	0.0764	1.2238	0.0802
$L_e(\theta, \hat{\theta})$	1.2538	0.0327	1.2450	0.0344
$L_{el}(\theta, \hat{\theta})$	1.2134	0.0771	1.2028	0.0815
$L_{g_1}(\theta, \hat{\theta})$	1.1136	0.1838	1.0969	0.1954
$L_{g_2}(\theta, \hat{\theta})$	1.1113	0.1835	1.0948	0.1952
$L_{g_3}(\theta, \hat{\theta})$	1.1161	0.1840	1.0992	0.1956

loss function underestimate(overestimate) the parameters. The class can be enlarged by considering other loss functions in the place of sqf and pqlf.

Acknowledgements

The authors would like to thank the Editor and reviewers for their valuable comments and suggestions, which helped improve the quality of this article.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Abbas, P., Ghitany, M. E., and Mahmoudi, M. R. (2019). Bayesian inference on power Lindley distribution based on different loss functions. *Brazilian Journal of Probability and Statistics*, **33**, 894–914.
- Al-About, F. M. (2009). Bayesian estimations for the extreme value distribution using progressive censored data and asymmetric loss. *International Mathematical Forum*, **4**, 1603–1622.
- Berger, J. O. (2013). *Statistical Decision Theory and Bayesian Analysis*. Springer.
- Calabria, R. and Pulcini, G. (1994). An engineering approach to Bayes estimation for the Weibull distribution. *Microelectronics Reliability*, **34**, 789–802.
- Chechile, R. A. (2001). Bayesian analysis of Gumbel distributed data. *Communications in Statistics-Theory and Methods*, **30**, 485–496.
- Coles, S. (2001). *An Introduction to Statistical Modeling of Extreme Values*. Springer.
- Coles, S. G. and Powell, E. A. (1996). Bayesian methods in extreme value modelling: a review and new developments. *International Statistical Review/Revue Internationale de Statistique*, **64**, 119–136.
- Coles, S. G. and Tawn, J. A. (1996). A Bayesian analysis of extreme rainfall data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **45**, 463–478.

- Dey, D. K., Ghosh, M., and Srinivasan, C. (1986). Simultaneous estimation of parameters under entropy loss. *Journal of Statistical Planning and Inference*, **15**, 347–363.
- El-Saayad, G. (1967). Estimation of the parameter of an exponential distribution. *Journal of the Royal Statistical Society: Series B (Methodological)*, **29**, 525–532.
- Embrechts, P., Klüppelberg, C., and Mikosch, T. (2013). *Modelling Extremal Events: for Insurance and Finance*. Springer.
- Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*, **57**, 97–109.
- Hosking, J. R. (1985). Algorithm as 215: Maximum-likelihood estimation of the parameters of the generalized extreme-value distribution. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **34**, 301–310.
- Hosking, J. R. (1990). L-moments: Analysis and estimation of distributions using linear combinations of order statistics. *Journal of the Royal Statistical Society: Series B (Methodological)*, **52**, 105–124.
- Hosking, J. R. M., Wallis, J. R., and Wood, E. F. (1985). Estimation of the generalized extreme-value distribution by the method of probability-weighted moments. *Technometrics*, **27**, 251–261.
- Jenkinson, A. F. (1955). The frequency distribution of the annual maximum (or minimum) values of meteorological elements. *Quarterly Journal of the Royal Meteorological Society*, **81**, 158–171.
- Landwehr, J. M., Matalas, N., and Wallis, J. (1979). Probability weighted moments compared with some traditional techniques in estimating Gumbel parameters and quantiles. *Water Resources Research*, **15**, 1055–1064.
- Leadbetter, M. R., Lindgren, G., and Rootzén, H. (1983). *Extremes and Related Properties of Random Sequences and Processes*. Springer.
- Lindley, D. V. (1980). Approximate bayesian methods. *Trabajos de Estadística y de Investigación Operativa*, **31**, 223–245.
- Mahdi, S. and Cenac, M. (2005). Estimating parameters of gumbel distribution using the methods of moments, probability weighted moments and maximum likelihood. *Revista de Matemática: Teoría y Aplicaciones*, **12**, 151–156.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, **21**, 1087–1092.
- Mousa, M. A., Jaheen, Z., and Ahmad, A. (2002). Bayesian estimation, prediction and characterization for the Gumbel model based on records. *Statistics: A Journal of Theoretical and Applied Statistics*, **36**, 65–74.
- Nelson, W. B. (1982). *Applied Life Data Analysis*. John Wiley & Sons.
- Norstrom, J. G. (1996). The use of precautionary loss functions in risk analysis. *IEEE Transactions on Reliability*, **45**, 400–403.
- Phien, H. N. (1987). A review of methods of parameter estimation for the extreme value type-1 distribution. *Journal of Hydrology*, **90**, 251–268.
- Prescott, P. and Walden, A. (1983). Maximum likelihood estimation of the parameters of the three-parameter generalized extreme-value distribution from censored samples. *Journal of Statistical Computation and Simulation*, **16**, 241–250.

- Smith, R. L. (1985). Maximum likelihood estimation in a class of nonregular cases. *Biometrika*, **72**, 67–90.
- Varian, H. R. (1975). A Bayesian approach to real estate assessment. In *Studies in Bayesian Econometric and Statistics in Honor of Leonard J. Savage*, 195–208.
- Vidal, I. (2014). A Bayesian analysis of the Gumbel distribution: an application to extreme rainfall data. *Stochastic Environmental Research and Risk Assessment*, **28**, 571–582.
- Von Mises, R. (1964). La distribution de la plus grande de n valeurs. reproduced, selected papers of von mises, r. *American Mathematical Society*, **2**, 271–294.
- Yilmaz, A., Kara, M., and Özdemir, O. (2021). Comparison of different estimation methods for extreme value distribution. *Journal of Applied Statistics*, **48**, 2259–2284.
- Zellner, A. (1986). Bayesian estimation and prediction using asymmetric loss functions. *Journal of the American Statistical Association*, **81**, 446–451.

ANNEXURE

Annexure A. Derivation for the Bayesian estimator under the loss $L_{g1}(\theta, \hat{\theta})$

Consider the case $g(z) = g_1(z) = \log z$ then the loss function has the form

$$L_{g1}(\theta, \hat{\theta}) = \frac{(\theta - \hat{\theta})^2}{\hat{\theta}^{1-\lambda}}.$$

Here we derive the Bayes estimator under this loss by minimizing the posterior expected loss. The posterior expected loss is then given by,

$$E(L_{g1}(\theta, \hat{\theta}) | \mathbf{x}) = \frac{1}{\hat{\theta}^{1-\lambda}} E(\theta^2 | \mathbf{x}) - 2\hat{\theta}^\lambda E(\theta | \mathbf{x}) + \hat{\theta}^{1+\lambda}.$$

Let $u = E(\theta | \mathbf{x})$ and $v = E(\theta^2 | \mathbf{x})$ then the above equation can be written as

$$E(L_{g1}(\theta, \hat{\theta}) | \mathbf{x}) = \frac{1}{\hat{\theta}^{1-\lambda}} v - 2\hat{\theta}^\lambda u + \hat{\theta}^{1+\lambda}.$$

The Bayesian parameter estimator $\hat{\theta}$, which minimizes the posterior expected loss is then the solution of the equation,

$$\frac{d}{d\hat{\theta}} E(L_{g1}(\theta, \hat{\theta}) | \mathbf{x}) = 0.$$

Implies that,

$$v(\lambda - 1)\hat{\theta}^{\lambda-2} - 2u\lambda\hat{\theta}^{\lambda-1} + (\lambda + 1)\hat{\theta}^\lambda = 0.$$

Multiplying by $\hat{\theta}^{2-\lambda}$,

$$v(\lambda - 1) - 2u\lambda\hat{\theta} + (\lambda + 1)\hat{\theta}^2 = 0$$

which is a quadratic equation in $\hat{\theta}$ and the two solutions for this equation are

$$\hat{\theta} = \frac{u\lambda + \sqrt{v + u^2\lambda^2 - v\lambda^2}}{\lambda + 1},$$

and,

$$\hat{\theta} = \frac{u\lambda - \sqrt{v + u^2\lambda^2 - v\lambda^2}}{\lambda + 1}.$$

For the Gumbel distribution, the second case is not feasible. So we consider the estimator $\hat{\theta}_{g_1}$ under the loss $L_{g_1}(\theta, \hat{\theta})$ as

$$\hat{\theta}_{g_1} = \frac{u\lambda + \sqrt{v + u^2\lambda^2 - v\lambda^2}}{\lambda + 1}.$$

Annexure B. Derivation for the Bayesian estimator under the loss $L_{g_2}(\theta, \hat{\theta})$

For $g(z) = g_2(z) = 1/z$ and the loss function $L_{g_2}(\theta, \hat{\theta})$ is given as

$$L_{g_2}(\theta, \hat{\theta}) = \frac{(\theta - \hat{\theta})^2}{\hat{\theta}(1 - \lambda) + \lambda}.$$

The posterior expected loss

$$E(L_{g_2}(\theta, \hat{\theta}) \mid \mathbf{x}) = \frac{1}{\hat{\theta}(1 - \lambda) + \lambda} E(\theta^2 \mid x) - 2\hat{\theta}E(\theta \mid x) + \hat{\theta}^2.$$

Letting $u = E(\theta \mid x)$ and $v = E(\theta^2 \mid x)$, we have,

$$E(L_{g_2}(\theta, \hat{\theta}) \mid \mathbf{x}) = \frac{1}{\hat{\theta}(1 - \lambda) + \lambda} v - 2\hat{\theta}u + \hat{\theta}^2.$$

Setting first derivative equal to zero

$$\frac{d}{d\hat{\theta}} E(L_{g_2}(\theta, \hat{\theta}) \mid \mathbf{x}) = 0.$$

Implies,

$$\frac{2\hat{\theta} - 2u}{\hat{\theta}(1 - \lambda) + \lambda} - \frac{(v - 2\hat{\theta}u + \hat{\theta}^2)(1 - \lambda)}{(\hat{\theta}(1 - \lambda) + \lambda)^2} = 0.$$

That is,

$$\hat{\theta}^2(1 - \lambda) + 2\hat{\theta}\lambda + v(\lambda - 1) - 2u\lambda = 0.$$

This is again a quadratic equation in $\hat{\theta}$ and the two solutions are,

$$\hat{\theta} = \frac{\lambda + \sqrt{v + 2u\lambda - 2v\lambda + \lambda^2 - 2u\lambda^2 + v\lambda^2}}{\lambda - 1}$$

and

$$\hat{\theta} = \frac{\lambda - \sqrt{v + 2u\lambda - 2v\lambda + \lambda^2 - 2u\lambda^2 + v\lambda^2}}{\lambda - 1}.$$

Here the first case is not feasible for the Gumbel parameters. Then we consider the estimator $\hat{\theta}_{g_2}$ under the loss function $L_{g_2}(\theta, \hat{\theta})$ as

$$\hat{\theta}_{g_2} = \frac{\lambda - \sqrt{v + 2u\lambda - 2v\lambda + \lambda^2 - 2u\lambda^2 + v\lambda^2}}{\lambda - 1}.$$

Annexure C. Derivation for the Bayesian estimator under the loss $L_{g_3}(\theta, \hat{\theta})$

Now for $g(z) = z$ and,

$$\begin{aligned} L_{g_3}(\theta, \hat{\theta}) &= \lambda(\theta - \hat{\theta})^2 + (1 - \lambda) \frac{(\theta - \hat{\theta})^2}{\hat{\theta}} \\ &= \lambda\theta^2 - 2\lambda\theta\hat{\theta} + \lambda\hat{\theta}^2 + \frac{\theta^2}{\hat{\theta}} - 2\theta + \hat{\theta} - \frac{\lambda\theta^2}{\hat{\theta}} + 2\lambda\theta - \lambda\hat{\theta}. \end{aligned}$$

Then posterior expected loss can be find as,

$$\begin{aligned} E(L_{g_3}(\theta, \hat{\theta}) \mid x) &= \lambda E(\theta^2 \mid x) - 2\lambda E(\theta \mid x)\hat{\theta} + \lambda\hat{\theta}^2 + \frac{1}{\hat{\theta}} E(\theta^2 \mid x) - 2E(\theta \mid x) + \hat{\theta} - \frac{\lambda}{\hat{\theta}} E(\theta^2 \mid x) \\ &\quad + 2\lambda E(\theta \mid x) - \lambda\hat{\theta}. \end{aligned}$$

Substituting $u = E(\theta \mid x)$ and $v = E(\theta^2 \mid x)$ we have,

$$E(L_{g_3}(\theta, \hat{\theta}) \mid x) = \lambda v - 2\lambda u\hat{\theta} + \lambda\hat{\theta}^2 + \frac{v}{\hat{\theta}} - 2u + \hat{\theta} - \frac{\lambda}{\hat{\theta}} v + 2\lambda u - \lambda\hat{\theta}.$$

To find the Bayes estimator under the loss function $L_{g_3}(\theta, \hat{\theta})$ that minimize the posterior expected loss we equate first derivative with respect to $\hat{\theta}$ equal to zero, i.e,

$$\frac{d}{d\hat{\theta}} E(L_{g_3}(\theta, \hat{\theta}) \mid x) = -2\lambda u + 2\lambda\hat{\theta} - \frac{v}{\hat{\theta}^2} + 1 + \frac{\lambda v}{\hat{\theta}^2} - \lambda = 0.$$

Multiplying both side by $\hat{\theta}^2$ we get a cubic polynomial on $\hat{\theta}$ as,

$$2\lambda\hat{\theta}^3 + (1 - \lambda - 2\lambda u)\hat{\theta}^2 + (\lambda - 1)v = 0.$$

The three solutions for $\hat{\theta}$ are given by,

$$\begin{aligned} \hat{\theta} &= \frac{1 - \lambda - 2u\lambda}{6\lambda} + \frac{(1 - \lambda - 2u\lambda)^2}{3 \times 2^{2/3}\lambda \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}} + \\ &\quad \frac{1}{6 \times 2^{1/3}\lambda} \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}, \end{aligned}$$

or,

$$\begin{aligned} \hat{\theta} &= \frac{1 - \lambda - 2u\lambda}{6\lambda} + \frac{(1 + i\sqrt{3})(1 - \lambda - 2u\lambda)^2}{6 \times 2^{2/3}\lambda \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}} - \\ &\quad \frac{1}{12 \times 2^{1/3}\lambda} (1 - i\sqrt{3}) \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}, \end{aligned}$$

or,

$$\begin{aligned} \hat{\theta} &= \frac{1 - \lambda - 2u\lambda}{6\lambda} + \frac{(1 - i\sqrt{3})(1 - \lambda - 2u\lambda)^2}{6 \times 2^{2/3}\lambda \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}} - \\ &\quad \frac{1}{12 \times 2^{1/3}\lambda} (1 + i\sqrt{3}) \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}, \end{aligned}$$

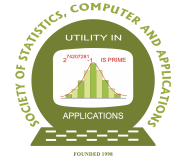
where,

$$H_1(u, v, \lambda) = -2 + 6\lambda + 12u\lambda - 6\lambda^2 - 24u\lambda^2 - 24u^2\lambda^2 + 108v\lambda^2 + 2\lambda^3 + 12u\lambda^3 + 24u^2\lambda^3 + 16u^3\lambda^3 - 108v\lambda^3,$$

$$H_2(u, v, \lambda) = -4(1 - \lambda - 2u\lambda)^6.$$

Here two of these solutions are complex conjugates. Since our parameter space is restricted to real, we can only assume real values for $\hat{\theta}$ so the estimator $\hat{\theta}_{g_3}$ under the loss $L_{g_3}(\theta, \hat{\theta})$ can be taken as

$$\hat{\theta} = \frac{1 - \lambda - 2u\lambda}{6\lambda} + \frac{(1 - \lambda - 2u\lambda)^2}{3 \times 2^{2/3}\lambda \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}} + \frac{1}{6 \times 2^{1/3}\lambda \left[H_1(u, v, \lambda) + \sqrt{H_2(u, v, \lambda) + (H_1(u, v, \lambda))^2} \right]^{1/3}}.$$



Transient State Solution of Retrial Queueing System with Catastrophe

Neelam Singla and Ankita Garg

*Department of Statistics
Punjabi University, Patiala, India*

Received: 19 May 2024; Revised: 20 August 2024; Accepted: 24 August 2024

Abstract

In this paper, we consider a single server retrial queueing system with catastrophe. If a primary customer finds the server available on its arrival, it is served immediately, else it retries for service as a secondary customer. Both primary and secondary arrivals follow Poisson processes. The time between two successive arrivals and services follow exponential distribution. Moreover, Catastrophe occurs at the system following a Poisson process. Repair of the failed system starts immediately. The repair times are also distributed exponentially. Time dependent probabilities for exact number of arrivals in the system, departures after taking service from the system and number of customers in orbit when the server is idle or busy are obtained. The probabilities of system being under repair are also obtained. Verification of results is done. Towards the end, numerical results are generated and represented graphically. Sensitivity analysis is performed to study the effect of various parameters.

Key words: Queueing; Retrial; Arrivals; Departures; Catastrophe; Repair.

AMS Subject Classifications: 60K25, 90B15, 90B22

1. Introduction

Queues or waiting lines are the systems where customers wait to get some kind of service and leave after getting service. Queues are a part of our daily lives. Everyday we encounter some form of waiting in lines. People at the checkout counter in supermarket, patients at the doctor's clinic, people to withdraw cash or to check account balance on ATM, *etc* are some examples of queueing systems. In classical queueing models, we observe customers who either wait for their service or leave the system forever. But there exist some systems in which a customer instead of leaving the system forever, retries for service after a random amount of time. Analysis of such systems introduced a new class of queueing systems which is known as retrial queueing systems. In retrial queues, if a customer on arrival finds the server free, it is served immediately else it temporarily leaves the service area and joins the virtual queue known as orbit. Thereafter it retries later from the orbit with a hope of having free server. Such kind of queueing systems play an important role to model many

practical situations like telecommunication systems, restaurants, banks, computer networks, *etc.*

A basic example of retrial queueing systems is a call center where call agent is a server who answers the calls from the customers. When a customer calls, if he is able to connect the call agent immediately, he is answered else he has to repeat the call.

There is extensive literature available on retrial queues. Among these, Cohen (1957) is the earliest work on retrial queues. The book Falin and Templeton (1997) and the survey paper Artalejo and Gómez-Corral (1999) are a great source on retrial queues in which the techniques for the computational analysis of retrial queues are covered. Yang and Templeton (1987) discussed some important retrial queueing models and presented their analytical results and the techniques used. Rajadurai *et al.* (2015) obtained the steady state probability generating function for the system size by using the supplementary variable method. The stationary characteristics of an $M/G/1$ retrial queue are investigated in Kumar *et al.* (2020). Steady state and time dependent solutions for number of customers in the system when server is idle or busy are obtained in Singla and Garg (2022).

As in real situations traffic intensity is constantly varying, so the determination of transient solution is very much essential in analyzing behaviour of the system. Transient solutions deal with behaviour of the queueing system during the initial phase after the system is started. They are useful to study the characteristics of a system on different time points. Therefore, transient analysis of queueing systems is extremely important from theoretical and practical perspective. The classical transient results for the $M/M/1$, $M/M/c$ and $M/G/1$ queue provide little insight into the behaviour of a queueing system through a fixed operation time t . The probability $P_n(t)$ gives the distribution for the number in the system at time t , but practically provides no information on how the system has regulated up until time t .

Pegden and Rosenshine (1982) was the first who introduced the concept of two-state by obtaining a closed form solution for the probability that exactly i arrivals and j services occur over a time interval of length t . Singla and Kalra (2018) used the concept of two-state in retrial queues and analyzed a single server retrial queueing model in which time dependent probabilities for exact number of arrivals and departures in the system when server is free or busy are obtained. An explanation of the retrial queueing system is shown in Figure 1.

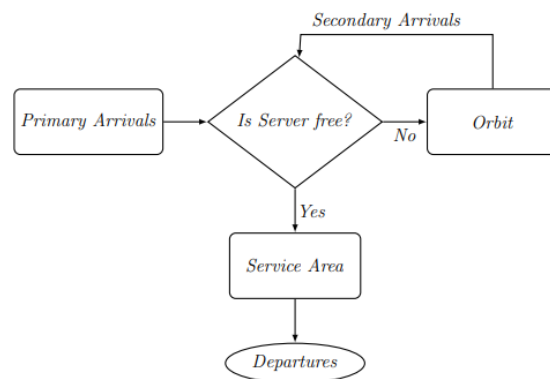


Figure 1: Basic structure of retrial queueing system

Queuing systems are often subject to sudden breakdowns that cause random failures. The sudden, unexpected breakdown of a machine, an electronic device, a computer system or a network at random time intervals is known as catastrophe. When a catastrophe occurs at a system, all the customers present in the system are deleted immediately, which also causes a breakdown of the system. Catastrophe resets the system from its current state to its failed state. The repair of the failed system starts immediately. When the system is repaired, it immediately resumes its functioning to serve new arrivals. Catastrophe models can be found very useful in computer networks and in communication systems. For example: In call centers, the sudden breakdown of power will result in loss of all the calls present at that time.

The loss of customers due to these breakdowns (also referred to as a kind of negative arrivals) was first introduced by Gelenbe (1991). Chao (1995) developed a queueing network model with catastrophes and obtained the steady-state probabilities for the model. Kumar and Arivudainambi (2000) studied ‘Transient solution of an M/M/1 queue with catastrophes’ in which behaviour of the probability of the server being free and mean queue length are discussed. Also steady state probabilities are obtained. Kumar *et al.* (2007) obtained transient solution for the system size in an M/M/1 queueing system with catastrophes, server failures and non-zero repair time. The steady state probabilities and some performance measures are also given. Bura (2022) analyzed infinite server Markovian queue with catastrophe and repairable servers. Figure 2 shows the basic structure of retrial queueing system with catastrophe.

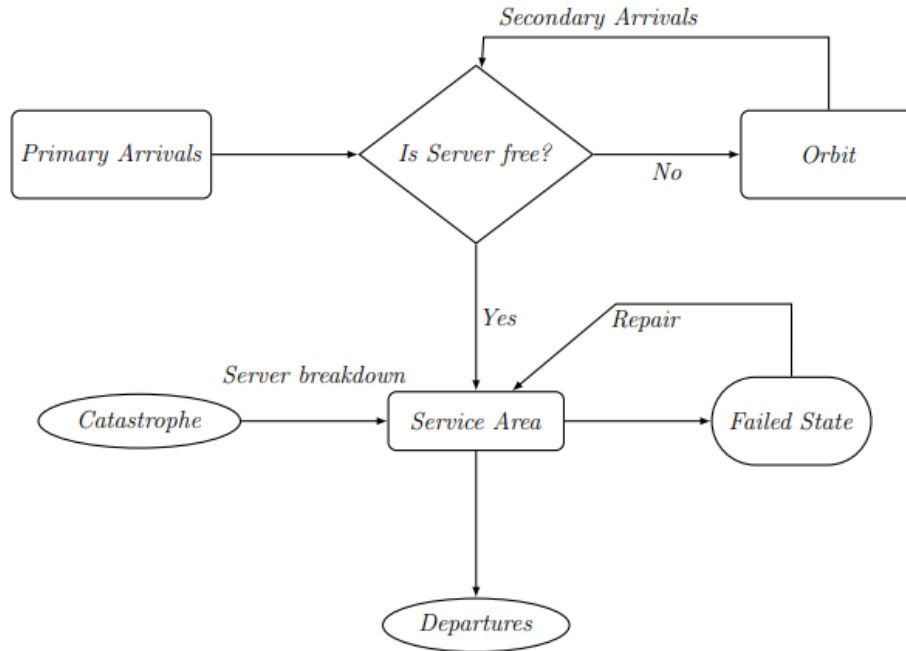


Figure 2: Basic structure of a retrial queue with catastrophe

In this paper, we derive time-dependent probabilities for exact number of arrivals in the system, exact number of departed units after taking service from the system and exact number of customers in the orbit by time t when the server is idle or busy for a single server retrial queueing system. In addition we obtain the probability of server being under repair

when the server fails due to catastrophic events at random time intervals. Besides these theoretical solutions, we present some numerical results graphically to study the behaviour of probabilities with respect to average service times. Sensitivity analysis is performed to study the effect of various parameters.

The paper is organized as follows. In section 2, the complete mathematical description of the model is defined. Also, the difference-differential equations are derived in this section. Mathematical analysis of the model is done in section 3 in which we obtained the recursive probabilities. Transient solutions are obtained in section 4. In section 5 verification of results is done. The numerical results are obtained and represented graphically in section 6. Sensitivity analysis is performed in section 7. Section 8 discusses the conclusion. Finally the references are listed.

2. Model description

We considered a single server retrial queueing system with catastrophe. Assumptions of the model are as follows:

- The primary customers arrive at the system according to Poisson process with mean arrival rate λ .
- If on arrival, a customer finds the server busy, it joins the orbit and retries later from the orbit. These retrials are considered to be secondary arrivals. These customers arrive at the system according to Poisson process with mean retrial rate θ .
- The service times follow exponential distribution with parameter μ .
- Catastrophe occurs at the system following a Poisson process with mean rate ξ .
- The repair times are distributed exponentially with parameter τ . Further it is assumed that arrival will not occur during the repair time of the system.

Also, the primary and secondary arrivals, inter-arrival times, service times, departures and catastrophes are statistically independent.

Laplace transformation $\bar{f}(s)$ of $f(t)$ is given by:

$$\bar{f}(s) = \int_0^{\infty} e^{-st} f(t) dt; \quad \text{Re}(s) > 0$$

If $L^{-1}\{\bar{f}(s)\} = F(t)$ and $L^{-1}\{\bar{g}(s)\} = G(t)$ then

$$L^{-1}\{\bar{f}(s)\bar{g}(s)\} = \int_0^t F(u)G(t-u)du = F * G$$

$F * G$ is called the convolution of F and G .

2.1. Notations

$P_{i,j,k}^{(0)}(t)$ = Probability that there are exactly i number of arrivals in the system, j number of units departed from the system after taking the service, k number of units in the orbit by time t and the server is free.

$P_{i,j,k}^{(1)}(t)$ = Probability that there are exactly i number of arrivals in the system, j number of units departed from the system after taking the service, k number of units in the orbit by time t and the server is busy.

$Q_{i,j}(t)$ = Probability that the system is under repair by time t when there are exactly i number of arrivals, j number of units departed from the system after taking service.

$P_{i,j,k}(t)$ = Probability that there are exactly i arrivals in the system, j number of units departed from the system after taking the service, k number of units in the orbit by time t .

$$P_{i,j,k}(t) = P_{i,j,k}^{(0)}(t) + P_{i,j,k}^{(1)}(t) \quad \forall i, j, k \quad i \geq j, k$$

$$\text{and} \quad P_{i,j,k}^{(1)}(t) = 0, \quad i \leq j, k; \quad P_{i,j,k}^{(0)}(t) = 0, \quad i < j, k;$$

Initially

$$P_{0,0,0}^{(0)}(0) = 1; \quad P_{i,j,k}^{(0)}(0) = 0, \quad P_{i,j,k}^{(1)}(0) = 0, \quad i, j, k \neq 0; \quad Q_{i,j}(0) = 0 \quad \forall i, j;$$

2.2. Difference-differential equations governing the system

$$\frac{d}{dt} P_{i,j,k}^{(0)}(t) = -(\lambda + \xi + k\theta)P_{i,j,k}^{(0)}(t) + \mu(1 - \delta_{0,j})P_{i,j-1,k}^{(1)}(t) + \tau\delta_{0,k}Q_{i,j}(t) \quad i \geq j, k \geq 0 \quad (1)$$

$$\frac{d}{dt} P_{i,0,k}^{(1)}(t) = -(\lambda + \mu + \xi)P_{i,0,k}^{(1)}(t) + \lambda\delta_{0,k}P_{i-1,0,k}^{(0)}(t) + \lambda(1 - \delta_{0,k})P_{i-1,0,k-1}^{(1)}(t) \quad i \geq 1, k \geq 0 \quad (2)$$

$$\begin{aligned} \frac{d}{dt} P_{i,j,k}^{(1)}(t) = & -(\lambda + \mu + \xi)P_{i,j,k}^{(1)}(t) + \lambda P_{i-1,j,k}^{(0)}(t) + \lambda(1 - \delta_{0,k})P_{i-1,j,k-1}^{(1)}(t) \\ & + (k+1)\theta(1 - \delta_{0,j})P_{i,j,k+1}^{(0)}(t) \quad i \geq 2, j > 0, k \geq 0 \end{aligned} \quad (3)$$

$$\frac{d}{dt} Q_{i,j}(t) = -\tau Q_{i,j}(t) + \xi \left(P_{i,j,0}^{(0)}(t) + \sum_{k=1}^{i-j} (1 - \delta_{0,j})P_{i,j,k}^{(0)}(t) + \sum_{k=0}^{i-j-1} P_{i,j,k}^{(1)}(t) \right) \quad i \geq j \geq 0 \quad (4)$$

where

$$\delta_{0,j} = \begin{cases} 1 & \text{when } j = 0 \\ 0 & \text{otherwise} \end{cases}$$

and

$$\delta_{0,k} = \begin{cases} 1 & \text{when } k = 0 \\ 0 & \text{otherwise} \end{cases}$$

3. Mathematical analysis of the model

Using the Laplace transformation $\bar{f}(s)$ of $f(t)$ given by

$$\bar{f}(s) = \int_0^\infty e^{-st} f(t) dt, \quad \operatorname{Re}(s) > 0$$

in the equations (1)-(4) along with the initial conditions, we have

$$(s + \lambda + \xi + k\theta) \bar{P}_{i,j,k}^{(0)}(s) = \mu(1 - \delta_{0,j}) \bar{P}_{i,j-1,k}^{(1)}(s) + \tau \delta_{0,k} \bar{Q}_{i,j}(s) \quad i \geq j, k \geq 0 \quad (5)$$

$$(s + \lambda + \mu + \xi) \bar{P}_{i,0,k}^{(1)}(s) = \lambda \delta_{0,k} \bar{P}_{i-1,0,k}^{(0)}(s) + \lambda(1 - \delta_{0,k}) \bar{P}_{i-1,0,k-1}^{(1)}(s) \quad i \geq 1, k \geq 0 \quad (6)$$

$$(s + \lambda + \mu + \xi) \bar{P}_{i,j,k}^{(1)}(s) = \lambda \bar{P}_{i-1,j,k}^{(0)}(s) + \lambda(1 - \delta_{0,k}) \bar{P}_{i-1,j,k-1}^{(1)}(s) \\ + (k+1)\theta(1 - \delta_{0,j}) \bar{P}_{i,j,k+1}^{(0)}(s) \quad i \geq 2, j > 0, k \geq 0 \quad (7)$$

$$(s + \tau) \bar{Q}_{i,j}(s) = \xi \left(\bar{P}_{i,j,0}^{(0)}(s) + \sum_{k=1}^{i-j} (1 - \delta_{0,j}) \bar{P}_{i,j,k}^{(0)}(s) + \sum_{k=0}^{i-j-1} \bar{P}_{i,j,k}^{(1)}(s) \right) \\ i \geq j \geq 0 \quad (8)$$

Solving equations (5)-(8) recursively, we obtained the following recursive solutions.

$$\bar{Q}_{0,0}(s) = \frac{\xi}{(s + \tau)(s + \lambda + \xi) - \xi\tau} \quad (9)$$

$$\bar{P}_{0,0,0}^{(0)}(s) = \frac{1}{s + \lambda + \xi} + \frac{\tau}{s + \lambda + \xi} \bar{Q}_{0,0}(s) \quad (10)$$

$$\bar{P}_{i,0,0}^{(0)}(s) = \frac{\tau}{s + \lambda + \xi} \bar{Q}_{i,0}(s) \quad i \geq 1 \quad (11)$$

$$\bar{P}_{i,0,k}^{(1)}(s) = \left(\frac{\lambda}{s + \lambda + \mu + \xi} \right)^{k+1} \frac{\tau}{s + \lambda + \xi} \bar{Q}_{i-k-1,0}(s) \quad i \geq k+2, k \geq 1 \quad (12)$$

$$\bar{Q}_{i,0}(s) = \frac{\xi}{s + \tau} \left[\bar{P}_{i,0,0}^{(0)}(s) + \sum_{k=0}^{i-1} \bar{P}_{i,0,k}^{(1)}(s) \right] \quad i \geq 1 \quad (13)$$

$$\bar{P}_{i,0,0}^{(1)}(s) = \frac{\lambda}{s + \lambda + \mu + \xi} \bar{P}_{i-1,0,0}^{(0)}(s) \quad i \geq 1 \quad (14)$$

$$\bar{P}_{i,0,i-1}^{(1)}(s) = \left(\frac{\lambda}{s + \lambda + \mu + \xi} \right)^i \bar{P}_{0,0,0}^{(0)}(s) \quad i \geq 2 \quad (15)$$

$$\bar{P}_{i,1,k}^{(0)}(s) = \frac{\mu}{s + \lambda + \xi + k\theta} \left[\left(\frac{\lambda}{s + \lambda + \mu + \xi} \right)^{k+1} \frac{\tau}{s + \lambda + \xi} \bar{Q}_{i-k-1,0}(s) \right] \\ i \geq k+2, k \geq 1 \quad (16)$$

$$\bar{P}_{i,1,i-1}^{(0)}(s) = \frac{\mu}{s + \lambda + \xi + (i-1)\theta} \left[\left(\frac{\lambda}{s + \lambda + \mu + \xi} \right)^i \bar{P}_{0,0,0}^{(0)}(s) \right] \quad i \geq 2 \quad (17)$$

$$\bar{P}_{i,1,0}^{(0)}(s) = \frac{\mu}{s + \lambda + \xi} \bar{P}_{i,0,0}^{(1)}(s) + \frac{\tau}{s + \lambda + \xi} \bar{Q}_{i,1}(s) \quad i \geq 1 \quad (18)$$

$$\begin{aligned} \bar{P}_{i,j,0}^{(0)}(s) &= \frac{\mu}{s + \lambda + \xi} \left[\frac{\lambda}{s + \lambda + \mu + \xi} \bar{P}_{i-1,j-1,0}^{(0)}(s) + \frac{\theta}{s + \lambda + \mu + \xi} \bar{P}_{i,j-1,1}^{(0)}(s) \right] \\ &\quad + \frac{\tau}{s + \lambda + \xi} \bar{Q}_{i,j}(s) \quad i \geq j \geq 2 \end{aligned} \quad (19)$$

$$\bar{Q}_{i,j}(s) = \frac{\xi}{s + \tau} \left[\sum_{k=0}^{i-j} \bar{P}_{i,j,k}^{(0)}(s) + \sum_{k=0}^{i-j-1} \bar{P}_{i,j,k}^{(1)}(s) \right] \quad i \geq j \geq 1 \quad (20)$$

$$\bar{P}_{i,j,0}^{(1)}(s) = \frac{\lambda}{s + \lambda + \mu + \xi} \bar{P}_{i-1,j,0}^{(0)}(s) + \frac{\theta}{s + \lambda + \mu + \xi} \bar{P}_{i,j,1}^{(0)}(s) \quad i > j \geq 1 \quad (21)$$

$$\bar{P}_{i,j,k}^{(1)}(s) = \sum_{p=0}^{k+1} \left(\frac{\lambda}{s + \lambda + \mu + \xi} \right)^{k+1-p} \eta_p \bar{P}_{i-k-1+p,j,p}^{(0)}(s) \quad i \geq k + j + 1, \quad k, j \geq 1 \quad (22)$$

where

$$\eta_p = \begin{cases} 1 & \text{if } p = 0 \\ 1 + \frac{p\theta}{s + \lambda + \mu + \xi} & \text{if } p = 1 \text{ to } i - k - 1 \\ \frac{p\theta}{s + \lambda + \mu + \xi} & \text{if } p = i - k \end{cases}$$

$$\bar{P}_{i,j,k}^{(0)}(s) = \frac{\mu}{s + \lambda + \xi + k\theta} \left[\sum_{p=0}^{k+1} \left(\frac{\lambda}{s + \lambda + \mu + \xi} \right)^{k+1-p} \eta_p \bar{P}_{i-k-1+p,j-1,p}^{(0)}(s) \right] \quad (23)$$

$$i \geq k + j, j \geq 2, k \geq 1 \quad (24)$$

where

$$\eta_p = \begin{cases} 1 & \text{if } p = 0 \\ 1 + \frac{p\theta}{s + \lambda + \mu + \xi} & \text{if } p = 1 \text{ to } i - k \\ \frac{p\theta}{s + \lambda + \mu + \xi} & \text{if } p = i - k + 1 \end{cases}$$

4. Transient solution of the model

Taking the Inverse Laplace transform of equations (9)–(23), we obtained the following transient solutions.

$$Q_{0,0}(t) = 2\xi \left[\left(\lambda^2 + \tau^2 + \xi^2 + 2 \left(\tau\xi + \lambda\xi - \lambda\tau \right) \right)^{1/2} e^{-1/2(\lambda+\xi+\tau)t} \sinh \left(\frac{1}{2} t \left\{ \lambda^2 + 2\lambda(\xi - \tau) + (\xi + \tau)^2 \right\}^{1/2} \right) \right]^{-1} \quad (25)$$

$$P_{0,0,0}^{(0)}(t) = e^{-(\lambda+\xi)t} + \tau e^{-(\lambda+\xi)t} * Q_{0,0}(t) \quad (26)$$

$$P_{i,0,i-1}^{(1)}(t) = \lambda^i e^{-(\lambda+\mu+\xi)t} \frac{t^{(i-1)}}{(i-1)!} * P_{0,0,0}^{(0)}(t) \quad i \geq 2 \quad (27)$$

$$P_{i,0,0}^{(0)}(t) = \tau e^{-(\lambda+\xi)t} * Q_{i,0}(t) \quad i \geq 1 \quad (28)$$

$$P_{i,0,0}^{(1)}(t) = \lambda e^{-(\lambda+\mu+\xi)t} * P_{i-1,0,0}^{(0)}(t) \quad i \geq 1 \quad (29)$$

$$P_{i,j,0}^{(0)}(t) = \mu \lambda e^{-(\lambda+\xi)t} \left[\frac{1}{\mu + \xi} - \frac{e^{-(\mu+\xi)t}}{\mu + \xi} \right] * P_{i-1,j-1,0}^{(0)}(t) + \mu \theta e^{-(\lambda+\xi)t} \left[\frac{1}{\mu + \xi} - \frac{e^{-(\mu+\xi)t}}{\mu + \xi} \right] * P_{i,j-1,1}^{(0)}(t) + \tau e^{-(\lambda+\xi)t} * Q_{i,j}(t) \quad i \geq j \geq 2 \quad (30)$$

$$P_{i,j,0}^{(1)}(t) = \lambda e^{-(\lambda+\mu+\xi)t} * P_{i-1,j,0}^{(0)}(t) + \theta e^{-(\lambda+\mu+\xi)t} * P_{i,j,1}^{(0)}(t) \quad i > j \geq 1 \quad (31)$$

$$P_{i,0,k}^{(1)}(t) = \lambda^{k+1} \tau e^{-(\lambda+\xi)t} \left[\frac{1}{(\mu + \xi)^{k+1}} - e^{-(\mu+\xi)t} \sum_{r=0}^k \frac{t^r}{r!} \frac{1}{(\mu + \xi)^{k+1-r}} \right] * Q_{i-k-1,0}(t) \quad i \geq k+2, k \geq 1 \quad (32)$$

$$Q_{i,j}(t) = \xi e^{-\tau t} * \left[\sum_{k=0}^{i-j} P_{i,j,k}^{(0)}(t) + \sum_{k=0}^{i-j-1} P_{i,j,k}^{(1)}(t) \right] \quad i \geq j \geq 1 \quad (33)$$

$$P_{i,j,k}^{(1)}(t) = \lambda^{k+1} e^{-(\lambda+\mu+\xi)t} \frac{t^k}{k!} * P_{i-k-1,j,0}^{(0)}(t) + e^{-(\lambda+\mu+\xi)t} \sum_{p=0}^{k-1} \lambda^{k-p} \frac{t^{k-p-1}}{(k-p-1)!} * P_{i-k+p,j,p+1}^{(0)}(t) + e^{-(\lambda+\mu+\xi)t} \sum_{p=0}^{k-1} \lambda^{k-p} (p+1) \theta \frac{t^{k-p}}{(k-p)!} * P_{i-k+p,j,p+1}^{(0)}(t) + (k+1) \theta e^{-(\lambda+\mu+\xi)t} * P_{i,j,k+1}^{(0)}(t) \quad i \geq j+k+1, j, k \geq 1 \quad (34)$$

$$P_{i,1,i-1}^{(0)}(t) = \mu \lambda^i e^{-(\lambda+(i-1)\theta+\xi)t} \left[\frac{1}{(\mu + \xi)^i} - e^{-(\mu+\xi)t} \sum_{r=0}^{i-1} \frac{t^r}{r!} \frac{1}{(\mu + \xi)^{i-r}} \right] * P_{0,0,0}^{(0)}(t) \quad i \geq 2 \quad (35)$$

$$P_{i,1,0}^{(0)}(t) = \mu e^{-(\lambda+\xi)t} * P_{i,0,0}^{(1)}(t) + \tau e^{-(\lambda+\xi)t} * Q_{i,1}(t) \quad i \geq 1 \quad (36)$$

$$Q_{i,0}(t) = \xi e^{-\tau t} * \left[P_{i,0,0}^{(0)}(t) + \sum_{k=0}^{i-1} P_{i,0,k}^{(1)}(t) \right] \quad i \geq 1 \quad (37)$$

$$\begin{aligned}
P_{i,1,k}^{(0)}(t) &= \mu \lambda^{k+1} \tau \left[\frac{e^{-(\lambda+\xi+k\theta)t}}{(k\theta - \mu)^{m+k}(k\theta)} \right. \\
&\quad + \frac{\sum_{l=1}^{k+1} \sum_{m=1}^l e^{-(\lambda+\mu+\xi)t} t^{k+1-l} (-1)^{m+1} \binom{l-1}{m-1} \left(\prod_{g_1=0}^{l-m-1} (1+g_1) \right) \left(\prod_{g_2=0}^{m-2} (1+g_2) \right)}{(k+1-l)!(m-1)!(k\theta - \mu)^m (-\mu)^{1+l-m}} \\
&\quad \left. + \frac{e^{-(\lambda+\xi)t}}{(k\theta)^m (\mu)^{k+1+l-m}} \right] * Q_{i-k-1,0}(t) \quad i \geq k+2, k \geq 1 \quad (38)
\end{aligned}$$

$$\begin{aligned}
P_{i,j,k}^{(0)}(t) &= \mu \lambda^{k+1} e^{-(\lambda+\xi+k\theta)t} \left[\frac{1}{(\mu + \xi)^{k+1}} - e^{-(\mu+\xi)t} \sum_{r=0}^k \frac{t^r}{r!} \frac{1}{(\mu + \xi)^{k+1-r}} \right] * P_{i-k-1,j-1,0}^{(0)}(t) \\
&\quad + \mu e^{-(\lambda+\xi+k\theta)t} \sum_{p=0}^{k-1} \lambda^{k-p} \left[\frac{1}{(\mu + \xi)^k} - e^{-(\mu+\xi)t} \sum_{r=0}^{k-1} \frac{t^r}{r!} \frac{1}{(\mu + \xi)^{k-r}} \right] * P_{i-k+p,j-1,p,0}(t) \\
&\quad + \mu e^{-(\lambda+\xi+k\theta)t} \sum_{p=0}^{k-1} \lambda^{k-p} (p+1) \theta \left[\frac{1}{(\mu + \xi)^{k+1}} - e^{-(\mu+\xi)t} \sum_{r=0}^k \frac{t^r}{r!} \frac{1}{(\mu + \xi)^{k+1-r}} \right] * \quad (39)
\end{aligned}$$

$$\begin{aligned}
&P_{i-k+p,j-1,p}^{(0)}(t) + \mu e^{-(\lambda+\xi+k\theta)t} (k+1) \theta \left[\frac{1}{\mu + \xi} - \frac{e^{-(\mu+\xi)t}}{\mu + \xi} \right] * P_{i,j-1,k+1}^{(0)}(t) \\
&i \geq k+j, j \geq 2, k \geq 1 \quad (40)
\end{aligned}$$

5. Verification of results

Taking $\xi = 0$, $\tau = 0$ and

$P_{i,j,k}^{(0)}(t) = P_{i,j}^{(0)}(t)$ where $k = \text{number of arrivals} - \text{number of departures}$.

$P_{i,j,k}^{(1)}(t) = P_{i,j}^{(1)}(t)$ where $k = \text{number of arrivals} - \text{number of departures} - 1$.

in equations (1 – 4) we get,

$$\frac{d}{dt} P_{i,j}^{(0)}(t) = -(\lambda + (i-j)\theta) P_{i,j}^{(0)}(t) + \mu(1 - \delta_{0,j}) P_{i,j-1}^{(1)}(t) \quad i \geq j \geq 0 \quad (41)$$

$$\frac{d}{dt} P_{1,0}^{(1)}(t) = -(\lambda + \mu) P_{1,0}^{(1)}(t) + \lambda P_{0,0}^{(0)}(t) \quad (42)$$

$$\begin{aligned}
\frac{d}{dt} P_{i,j}^{(1)}(t) &= -(\lambda + \mu) P_{i,j}^{(1)}(t) + \lambda(1 - \delta_{0,j}) P_{i-1,j}^{(0)}(t) + \lambda(1 - \delta_{i-1,j}) P_{i-1,j}^{(1)}(t) \\
&\quad + (i-j)\theta(1 - \delta_{0,j}) P_{i,j}^{(0)}(t) \quad i > 1, i > j \geq 0 \quad (43)
\end{aligned}$$

which coincides with the results (1 – 3) of Singla and Kalra (2018). Furthermore, the model given in equations (39) – (41) can be converted into one-dimensional model by defining the probability $U_n^{(m)}(t)$ as:

$U_n^{(m)}(t)$ = Probability that there are n customers in the orbit at time t and the server is free or busy according as $m = 0$ or 1 .

When server is free, it is defined as probability:

$$U_n^{(0)}(t) = \sum_{j=0}^{\infty} P_{j+n,j}^{(0)}(t)$$

where n = number of arrivals - number of departures.
 when server is busy, it is defined as probability:

$$U_n^{(1)}(t) = \sum_{j=0}^{\infty} P_{j+n+1,j}^{(1)}(t)$$

where n = number of arrivals - number of departures - 1.

By using above definitions in equations (39 – 41) the equations in statistical equilibrium are:

$$\begin{aligned} (\lambda + n\theta)U_n^{(0)} &= \mu U_n^{(1)} & n \geq 0 \\ (\lambda + \mu)U_n^{(1)} &= \lambda(U_n^{(0)} + U_{n-1}^{(1)}) + (n+1)\theta U_{n+1}^{(0)} & n \geq 1 \end{aligned}$$

which coincides with the results (1.5) – (1.6) of Falin and Templeton (1997)

6. Numerical solution and graphical representation

The Numerical results are generated using MATLAB programming and following Bunday (1986) for the case $\rho = \left(\frac{\lambda}{\mu}\right) = 0.7$, $\eta = \left(\frac{\theta}{\mu}\right) = 0.6$, $\tau' = \left(\frac{\tau}{\mu}\right) = 0.4$, $\xi' = \left(\frac{\xi}{\mu}\right) = 0.3$. In following tables, we observe some probabilities at various time instants whose sum approaches to 1.

Table 1: At time $t = 1$

t	$P_{0,0,0}^{(0)}$	$P_{1,1,0}^{(0)}$	$P_{2,1,1}^{(0)}$	$P_{1,0,0}^{(1)}$	$P_{2,0,1}^{(1)}$	$P_{2,1,0}^{(1)}$	$P_{3,0,2}^{(1)}$	$P_{3,1,1}^{(1)}$	$Q_{0,0}$	$Q_{1,0}$	$Q_{1,1}$
1	0.3954	0.0972	0.0161	0.1676	0.0483	0.0212	0.0102	0.0064	0.1542	0.0365	0.0122

$Q_{2,0}$	$Q_{2,1}$	Sum
0.0068	0.0035	0.9756

Table 2: At time $t = 25$

t	$P_{5,0,0}^{(0)}$	$P_{4,3,0}^{(0)}$	$P_{3,2,0}^{(0)}$	$P_{5,1,0}^{(0)}$	$P_{5,5,0}^{(0)}$	$P_{5,4,0}^{(0)}$	$P_{5,3,0}^{(0)}$	$P_{5,2,0}^{(0)}$	$Q_{5,0}$	$Q_{5,1}$	$Q_{5,2}$
25	0.0076	0.0039	0.003	0.0408	0.0898	0.1383	0.1467	0.1001	0.006	0.0315	0.0762

$Q_{5,3}$	$Q_{5,4}$	$Q_{5,5}$	Sum
0.1097	0.1014	0.0649	0.9199

Table 3: At time $t = 40$

t	$P_{5,0,0}^{(0)}$	$P_{5,1,0}^{(0)}$	$P_{5,5,0}^{(0)}$	$P_{5,4,0}^{(0)}$	$P_{5,3,0}^{(0)}$	$P_{5,2,0}^{(0)}$	$Q_{5,0}$	$Q_{5,1}$	$Q_{5,2}$	$Q_{5,3}$	$Q_{5,4}$
40	0.0082	0.0442	0.0942	0.1503	0.1611	0.1097	0.0062	0.0333	0.0823	0.1207	0.1125

$Q_{5,5}$	Sum
0.0705	0.9932

The probabilities against time are represented graphically in the following figures.

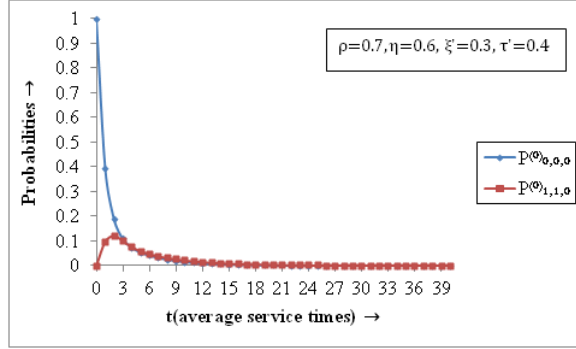


Figure 3: Probabilities $P_{0,0,0}^{(0)}$ and $P_{1,1,0}^{(0)}$ against average service times

In Figure 3, the probabilities $P_{0,0,0}^{(0)}$ and $P_{1,1,0}^{(0)}$ are plotted against average service times t for the given case. Here we observe that the probability $P_{0,0,0}^{(0)}$ decreases rapidly from its initial value 1 (at time $t = 0$) whereas the probability $P_{1,1,0}^{(0)}$ increases from its initial value 0 (at $t = 0$) in the beginning and then decreases.

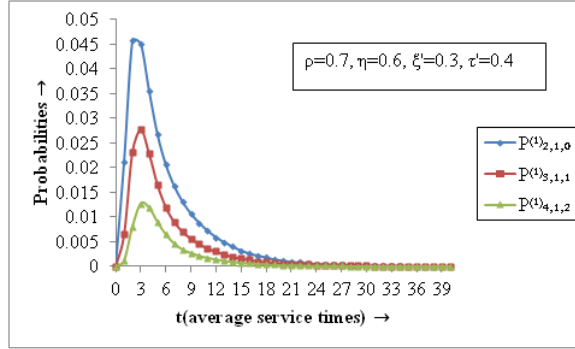


Figure 4: Comparison among the probabilities $P_{2,1,0}^{(1)}(t)$, $P_{3,1,1}^{(1)}(t)$, and $P_{4,1,2}^{(1)}(t)$

Figure 4 shows the comparison between the probabilities $P_{2,1,0}^{(1)}$, $P_{3,1,1}^{(1)}$ and $P_{4,1,2}^{(1)}$ against times t . From the graph it is observed that all the probabilities increase in the beginning and then decrease. Furthermore, the above graph shows that the probabilities achieve lower values with the increase in arrivals for the same number of departures.

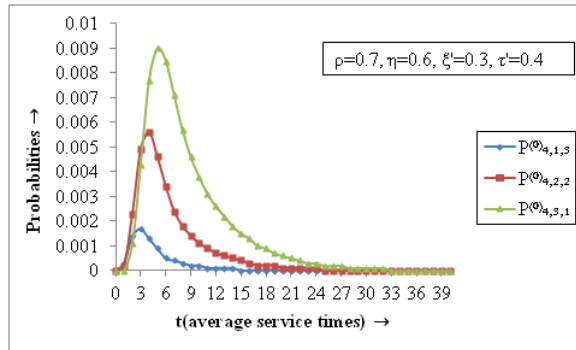


Figure 5: Comparison among the probabilities $P_{4,1,3}^{(0)}(t)$, $P_{4,2,2}^{(0)}(t)$, and $P_{4,3,1}^{(0)}(t)$

In Figure 5, the probabilities $P_{4,1,3}^{(0)}$, $P_{4,2,2}^{(0)}$ and $P_{4,3,1}^{(0)}$ are compared against average service times t for the given case. All the probabilities increase initially and then start decreasing after reaching their peak. It is also interpreted from the graph that the considered probabilities are higher when the number of departures is larger for the same number of arrivals.

7. Sensitivity analysis of the model

In this section, the effect of change in parameters of the system on the various probabilities is studied. The data is summarized in the following tables. In Table 4, the probability $P_{0,0,0}^{(0)}(t)$ is given for different values of ρ . It can be seen that in general whenever the arrival rate per unit service time increases the probability $P_{0,0,0}^{(0)}(t)$ decreases against average service time t . In Table 5, the effect of catastrophe rate per unit service time on the probability of system being under repair is studied. It is observed that the probability of system being under repair increases with the increase in ξ' . Similarly the effect of repair rate per unit service time on the probability of system being under repair is studied in Table 6. Here we notice that the probability decreases with the increase in τ' .

Table 4: Effect of arrival rate per unit service time on probability $P_{0,0,0}^{(0)}(t)$

	$\rho = 0.5$	$\rho = 0.7$	$\rho = 0.9$
t	$P_{0,0,0}^{(0)}$	$P_{0,0,0}^{(0)}$	$P_{0,0,0}^{(0)}$
0	1	1	1
3	0.1745	0.1092	0.0704
6	0.0772	0.0413	0.0243
9	0.0414	0.0196	0.0107
12	0.0227	0.0095	0.0047
15	0.0124	0.0046	0.0021
18	0.0068	0.0022	0.0009
21	0.0037	0.0011	0.0004
24	0.0021	0.0005	0.0002
27	0.0011	0.0003	0.0001
30	0.0006	0.0001	0
33	0.0003	0.0001	0
36	0.0002	0	0
39	0.0001	0	0

Table 5: Effect of catastrophe rate per unit service time on probability of system being under repair

	$\xi' = 0.3$			$\xi' = 0.6$			$\xi' = 0.9$		
t	$Q_{2,1}$	$Q_{3,1}$	$Q_{4,1}$	$Q_{2,1}$	$Q_{3,1}$	$Q_{4,1}$	$Q_{2,1}$	$Q_{3,1}$	$Q_{4,1}$
1	0.0035	0.0007	0.0001	0.0057	0.001	0.0001	0.0069	0.0012	0.0002
3	0.0317	0.0133	0.0046	0.0409	0.0165	0.0054	0.041	0.0158	0.005
6	0.0409	0.0242	0.0121	0.0572	0.0325	0.0153	0.0609	0.0329	0.0145
9	0.0328	0.0233	0.0138	0.0559	0.0388	0.0219	0.0663	0.0439	0.0234
12	0.0233	0.019	0.0128	0.0481	0.0388	0.0251	0.0636	0.0491	0.0301
15	0.0154	0.0141	0.0105	0.0385	0.0351	0.0253	0.0565	0.0495	0.034
18	0.0097	0.0098	0.008	0.0293	0.0296	0.0235	0.0477	0.0466	0.0354
21	0.0059	0.0065	0.0057	0.0215	0.0238	0.0206	0.0388	0.0418	0.0346
24	0.0035	0.0042	0.0039	0.0153	0.0185	0.0172	0.0307	0.036	0.0322
27	0.0021	0.0026	0.0026	0.0107	0.0139	0.0139	0.0237	0.0301	0.029
30	0.0012	0.0016	0.0017	0.0073	0.0102	0.0109	0.018	0.0246	0.0253
33	0.0007	0.001	0.0011	0.0049	0.0074	0.0083	0.0135	0.0197	0.0215
36	0.0004	0.0006	0.0007	0.0033	0.0052	0.0062	0.0099	0.0155	0.0179
39	0.0002	0.0003	0.0004	0.0022	0.0037	0.0046	0.0073	0.012	0.0146

Table 6: Effect of repair rate per unit service time on probability of system being under repair

	$\tau' = 0.2$			$\tau' = 0.5$			$\tau' = 0.9$		
t	$Q_{4,2}$	$Q_{4,3}$	$Q_{4,4}$	$Q_{4,2}$	$Q_{4,3}$	$Q_{4,4}$	$Q_{4,2}$	$Q_{4,3}$	$Q_{4,4}$
1	0	0	0	0	0	0	0	0	0
3	0.0038	0.0013	0.0002	0.0035	0.0012	0.0002	0.0031	0.0011	0.0001
6	0.017	0.0116	0.0035	0.0152	0.0103	0.003	0.0128	0.0087	0.0025
9	0.0228	0.0185	0.0074	0.0193	0.0156	0.0061	0.0143	0.0119	0.0046
12	0.0247	0.0207	0.0087	0.0178	0.0153	0.0063	0.0106	0.0097	0.0041
15	0.0246	0.0207	0.0086	0.014	0.0123	0.0051	0.0064	0.0061	0.0027
18	0.0232	0.0195	0.0079	0.0099	0.0089	0.0037	0.0034	0.0033	0.0015
21	0.021	0.0177	0.0071	0.0066	0.006	0.0024	0.0016	0.0016	0.0007
24	0.0185	0.0156	0.0061	0.0041	0.0038	0.0015	0.0007	0.0007	0.0003
27	0.016	0.0134	0.0052	0.0025	0.0023	0.0009	0.0003	0.0003	0.0001
30	0.0135	0.0113	0.0043	0.0015	0.0013	0.0005	0.0001	0.0001	0.0001
33	0.0112	0.0094	0.0035	0.0008	0.0008	0.0003	0	0.0001	0
36	0.0092	0.0077	0.0029	0.0005	0.0004	0.0002	0	0	0
39	0.0075	0.0063	0.0023	0.0002	0.0002	0.0001	0	0	0

This variation can also be shown graphically in the figures (6) – (8).

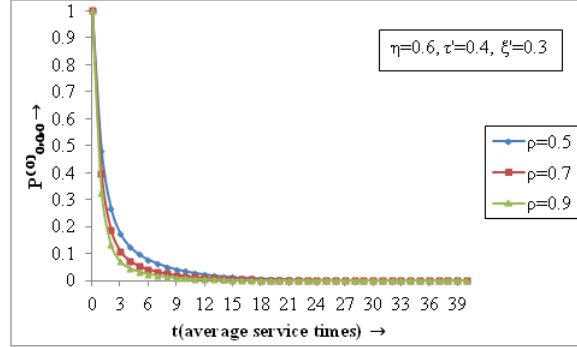


Figure 6: Probability $P_{0,0,0}^{(0)}$ for different values of ρ

Figure 6 shows the variation in probability $P_{0,0,0}^{(0)}$ by taking the different values of arrival rate per unit service time (ρ). The probability decreases with the increasing values of ρ .

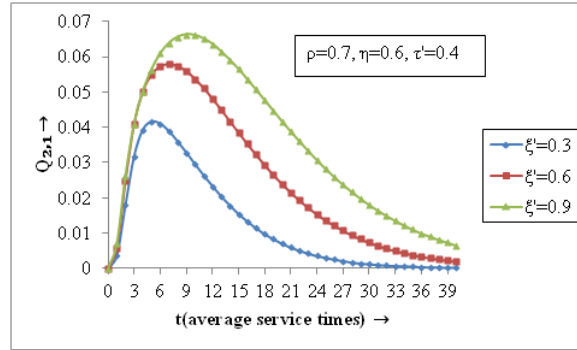


Figure 7: Probability $Q_{2,1}$ for different values of ξ'

In Figure 7, the variation is shown graphically by taking the probability of system being under repair when the number of arrivals are two and the departed units are one for different values of ξ' .

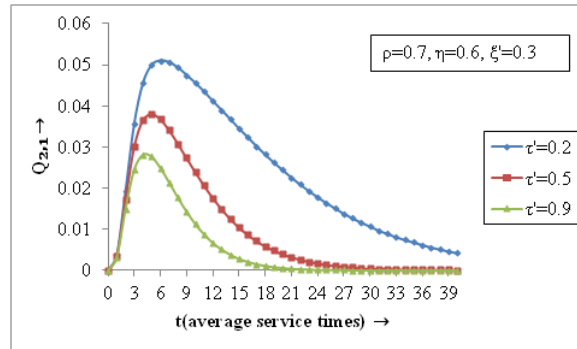


Figure 8: Probability $Q_{2,1}$ for different values of τ'

Similarly in Figure 8, the effect of τ' (change of repair rate per unit service time) on the probability $Q_{2,1}$ is studied. It is observed from the graph that the probability of system being under repair decreases with the increasing values of repair rate per unit service time which is as desired.

8. Conclusion

In this paper, we studied a single server retrial queueing system with catastrophe. It is very crucial to identify and mitigate the factors that lead to catastrophic events in designing efficient queueing systems. The proposed model is highly applicable to represent many practical situations like submitting any application online, ticket booking services using telephone facility, withdrawing cash at an ATM, manufacturing sectors, call centers, *etc.* Results are well-quantified as the obtained transient probabilities consider arrivals and departures separately rather than total number of customers in the system. The generated numerical results for the developed model and their graphical representation successfully depicts the effect of various parameters on various probabilities. Future research will aim to develop queueing models that incorporate other influencing factors, such as server heterogeneity and customer impatience.

Acknowledgments

The authors are highly thankful to the DST for developing computing facilities in the department of Statistics. The authors have made use of these facilities while doing this research work. The authors are also thankful to the anonymous referee for his valuable comments, which significantly enhanced the quality of this paper.

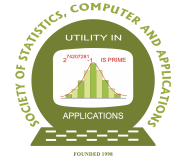
Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Artalejo, J. R. and Gómez-Corral, A. (1999). Retrial queueing systems. *Mathematical And Computer Modelling*, **30**, 1–228.
- Bunday, B. D. (1986). *Basic Queueing Theory*. Arnold.
- Bura, G. S. (2022). M/m/ ∞ queue with catastrophes and repairable servers. *Reliability: Theory & Applications*, **17**, 143–153.
- Chao, X. (1995). A queueing network model with catastrophes and product form solution. *Operations Research Letters*, **18**, 75–79.
- Cohen, J. (1957). Basic problems of telephone traffic theory and the influence of repeated calls. *Philips Telecommunication Review*, **18**, 49–100.
- Falin, G. and Templeton, J. G. (1997). *Retrial Queues*. Chapman and Hall.
- Gelenbe, E. (1991). Product-form queueing networks with negative and positive customers. *Journal of Applied Probability*, **28**, 656–663.

- Kumar, B. K. and Arivudainambi, D. (2000). Transient solution of an M/M/1 queue with catastrophes. *Computers & Mathematics With Applications*, **40**, 1233–1240.
- Kumar, B. K., Krishnamoorthy, A., Madheswari, S. P., and Basha, S. S. (2007). Transient analysis of a single server queue with catastrophes, failures and repairs. *Queueing Systems*, **56**, 133–141.
- Kumar, M. S., Dadlani, A., and Kim, K. (2020). Performance analysis of an unreliable M/G/1 retrial queue with two-way communication. *Operational Research*, **20**, 2267–2280.
- Pegden, C. D. and Rosenshine, M. (1982). Some new results for the M/M/1 queue. *Management Science*, **28**, 821–828.
- Rajadurai, P., Chandrasekaran, V., and Saravanarajan, M. (2015). Steady state analysis of batch arrival feedback retrial queue with two phases of service, negative customers, bernoulli vacation and server breakdown. *International Journal of Mathematics in Operational Research*, **7**, 519–546.
- Singla, N. and Garg, A. (2022). Single server retrial queueing system with catastrophe. *Reliability: Theory & Applications*, **17**, 376–390.
- Singla, N. and Kalra, S. (2018). Performance analysis of a two-state queueing model with retrials. *Journal of Rajasthan Academy of Physical Sciences*, **17**, 81–100.
- Yang, T. and Templeton, J. G. C. (1987). A survey on retrial queues. *Queueing Systems*, **2**, 201–233.



Finite Mixtures of Matrix Variate Log-normal Distributions for Clustering Skewed Three-Way Data

Shiva Kumar Kurva and Kiruthika C.

Department of Statistics, Pondicherry University, India

Received: 24 April 2024; Revised: 12 August 2024; Accepted: 03 September 2024

Abstract

The increasing complexity in the dimensionality of data throws new challenges in modelling. Matrix variate data with underlying subpopulations need more sophisticated models to make meaningful statistical inferences and clustering results. This work introduces the finite mixtures of matrix variate log-normal distributions to model the right skewed multi-modal matrix variate data, and its application to model-based clustering is discussed. In addition, an extended K-means algorithm is developed as an alternative to the ordinary K-means approach for clustering matrix variate data. Furthermore, it can also be a useful initialization approach for matrix variate finite-mixture models, which received much attention lately for modelling this kind of data. Using the suggested initialization approach, the Expectation-Maximization algorithm is employed to estimate the parameters. The ability of the proposed methodology is illustrated through simulations and real data studies.

Key words: Finite-mixtures; Matrix variate; Log-normal; Model-based clustering; Expectation Maximization; Extended K-means.

AMS Subject Classifications: 91G20, 62G07

1. Introduction

As the dimensionality of data increases, traditional modelling approaches often struggle to capture the intricacies and variations present in complex data sets. One such challenge is modelling three-way or matrix variate data (Viroli, 2011b) where each observation is represented as a matrix rather than a scalar or vector. The multivariate longitudinal data is an example of three-way data. On the other hand, to model and cluster the multi-modal data, the most used characterization is the finite mixture models which are a statistical approach that combines multiple probability distributions to create a more flexible and versatile model. These models are particularly useful when dealing with datasets containing distinct subpopulations or clusters with different statistical characteristics. Vast literature can be seen in finite mixture models. For example, see Everitt (2013), Peel and MacLachlan (2000), McNicholas and Murphy (2008), Bouguila and ElGuebaly (2009), Melnykov and Maitra (2010), McLachlan *et al.* (2019).

In the recent past, high-dimensional multi-modal data was modelled using finite mixtures of matrix variate distributions. Some notable works are by (Viroli, 2011a), Dođru *et al.* (2016), Gallagher and McNicholas (2018), Tomarchio *et al.* (2022), Tomarchio *et al.* (2020), and Silva *et al.* (2023). The matrix variate finite mixture models found in the literature mostly rely on symmetricity and heavy-tailed distributions. However, the observations at the tails of the distribution may be regarded as outliers in the event of highly skewed data, making it challenging to cluster these observations. Therefore, mixture models with asymmetric distributions are required. Nevertheless, the data that is right-skewed can be modelled by the log-normal distribution. Furthermore, the extreme observations—which are typically considered outliers—are also explained by this distribution. The univariate and multivariate finite mixtures of log-normal distributions are introduced by Deepana and Kiruthika (2018, 2022). Moving forward, we employ matrix variate log-normal distributions as a finite mixtures to model and cluster the matrix variate right-skewed data by utilizing the properties of log-normal distribution, such as skewness and heavy tails.

The parameter estimation of finite mixtures involves identifying latent variable problem. The widely used family of Expectation-Maximization (EM) algorithms efficiently estimate parameters in finite mixtures (Dempster *et al.*, 1977). However, the EM algorithm does not guarantee the global maximum and is sensitive to initial values. Many initialization techniques are discussed in the literature. For example, see Biernacki *et al.* (2003), Karlis and Xekalaki (2003), Michael and Melnykov (2016), Panić *et al.* (2020), and Hu (2015). However, all these techniques are restricted to univariate and multivariate finite mixture models. In recent works of finite mixtures of matrix variate distributions, a random multi-start initializations are employed in parameter estimation. On the other hand, to cluster the matrix variate data, the conventional K-means developed for multivariate data cannot be useful.

To overcome these challenges, the finite mixtures of matrix variate log-normal distributions is introduced to model and cluster the skewed matrix variate data. In addition, an extended K-means algorithm is proposed to cluster the matrix variate data and to obtain the initial values of the parameters in finite mixtures of matrix variate distributions. The rest of the paper is organized as follows: Section 2 introduces the extended K-means clustering for matrix variate data. Section 3 consists of finite mixtures of matrix variate log-normal distributions, its maximum likelihood estimation, and the EM algorithm. Section 4 provides detailed simulation studies and section 5 presents a real data study to validate the proposed methodology. The paper concludes with section 6 by summarising the proposed methodology, simulation and real data studies.

2. Extended K-means clustering

In the realm of data analysis, clustering methods are indispensable tools for uncovering hidden patterns within datasets. Among these techniques, K-means clustering stands as a cornerstone, renowned for its simplicity, efficiency, and effectiveness in partitioning data into distinct and non-overlapping clusters. The essence of K-means lies in its ability to iteratively assign data points to clusters based on the similarity of their features while striving to minimize the within-cluster sum of squares. Moreover, its computational efficiency renders it suitable for large-scale data analysis, making it a preferred choice for real-world applications.

Consider a sample of n matrix observations $X_1, X_2, \dots, X_n$ that are to be segregated into m ($\leq n$) clusters $C_1, C_2, \dots, C_m$. Let c_j ($j = 1, 2, \dots, m$) be the centre matrix of the cluster C_j . The essential part of the K-means is to calculate the distance between cluster centres and observations. Rezaei and Ahmadi (2023) have given the distance between two matrix observations X and Y which are observed from a matrix variate distribution is,

$$D(X, Y) = \sqrt{\text{tr} \{ \Psi^{-1}(X - Y)^T \Sigma^{-1}(X - Y) \}}. \quad (1)$$

where Ψ and Σ are the positive definite matrices. Here, $\text{tr}(\cdot)$ is the trace of a symmetric matrix and $(\cdot)^T$ denotes the transpose of matrix. Considering the above distance by keeping $\Psi = \Sigma = I$, the distance between a matrix observation X_i and a cluster centre matrix c_j is given as,

$$D(X_i, c_j) = \sqrt{\text{tr} \{ (X_i - c_j)^T (X_i - c_j) \}}. \quad (2)$$

Once the initial centres are identified and the distance between each observation and the centres is obtained, the observations are to be segregated into updated cluster centres. For that, the updated cluster centres are which minimize the within-cluster sum of squares, which is,

$$W = \sum_{j=1}^m \sum_{X_i \in C_j} \text{tr} \{ (X_i - c_j)^T (X_i - c_j) \}. \quad (3)$$

On minimizing the within-cluster sum of squares, the least-squares estimates of the updated cluster centres are obtained as,

$$c_j = \frac{1}{n_j} \sum_{X_i \in C_j} X_i \quad (4)$$

where n_j is the size of cluster C_j .

Since the results of the K-means depends on initial centres, the algorithm needs multiple compilations with random initial centres, typically observations of the data. The best clustering solution is the one that has the lowest within-cluster sum of squares. This method provides the clustered observations, corresponding cluster centres, and sizes. The step-by-step procedure is given in Algorithm 1. This clustering technique can be used for any matrix variate data which is reasonably well separated. Furthermore, the location parameters and mixing proportions of the finite mixtures of matrix variate distributions can be initialized with these cluster centres and cluster sizes, respectively. In further studies, we focus on alternative distance metrics and different initializations for the extended K-means technique. In the next section, the proposed technique is used as an initialization to the finite mixtures of matrix variate log-normal distributions.

3. Finite mixtures of matrix variate Log-normal distributions

For many practical reasons, it is obvious to observe data on multiple variables (say columns) at different time points or situations (say rows). As a result, there are two covariance structures, one along columns and the other along rows. Consider a random matrix $\mathcal{Y}$ of

Algorithm 1 Extended K-means algorithm

```

1: procedure K-means( $X, m$ )
2:   for  $X = [x_i]; i = 1, \dots, n$  &  $C_j; j = 1, \dots, m$  do                                ▷ Initialization
3:     Random Initialize of  $c_j$  ( $j^{th}$  Cluster Centre) from  $X$ 
4:   end for
5:   while  $c_j^{(t)} \neq c_j^{(t+1)}$  do                                                    ▷ Core
6:     for  $X = [x_i]; i = 1, \dots, n$  &  $C_j; j = 1, \dots, m$  do
7:       update  $I(x_i \in C_j) \forall j = \arg \min_j d(X_i, c_j)$  using (2)
8:       update  $c_j$  according to (4).
9:     end for
10:  end while
11:  return  $c_j, n_j, I(X_i \in C_j)$ 
12: end procedure

```

order $(r \times c)$, assumed to be distributed as matrix variate Log-normal (MVLN) distribution (Rohde *et al.*, 2012) with the mean matrix M of order $(r \times c)$, the row covariance matrix Ω of order $(r \times r)$, and the column covariance matrix Σ of order $(c \times c)$. The probability density function of $\mathcal{Y}$ is given by,

$$f(Y; M, \Omega, \Sigma) = \frac{1}{(2\pi)^{\frac{rc}{2}} |\Sigma|^{\frac{r}{2}} |\Omega|^{\frac{c}{2}} \prod_{p=1}^r \prod_{q=1}^c y_{pq}} \times \exp \left\{ -\frac{1}{2} \text{tr} \left[\Sigma^{-1} (\log Y - M) \Omega^{-1} (\log Y - M)^T \right] \right\} \quad (5)$$

where y_{pq} is the value of p^{th} row and q^{th} column of an observation matrix Y . Here, Ω and Σ are positive definite matrices. One of the advantages of this distribution is that one can obtain the closed-form expressions for the parameter estimates.

Let $\mathcal{X}$ be a random matrix that has finite mixtures of matrix variate log-normal (FMMVLN) distributions, the pdf of $\mathcal{X}$ is given as,

$$\begin{aligned} g(X; \Theta) &= \sum_{j=1}^k \pi_j f(X; M_j, \Omega_j, \Sigma_j) \\ &= \sum_{j=1}^k \pi_j \frac{1}{(2\pi)^{\frac{rc}{2}} |\Sigma_j|^{\frac{r}{2}} |\Omega_j|^{\frac{c}{2}} \prod_{p=1}^r \prod_{q=1}^c x_{pq}} \times \\ &\quad \exp \left\{ -\frac{1}{2} \text{tr} \left[\Sigma_j^{-1} (\log X - M_j) \Omega_j^{-1} (\log X - M_j)^T \right] \right\} \end{aligned} \quad (6)$$

where π_j ($0 < \pi_j \leq 1$) is the mixing proportion of the j^{th} component density $f(X; M_j, \Omega_j, \Sigma_j)$. The prior probability that an observed matrix belongs to each component density is π_j ($j = 1, 2, \dots, k$) and $\sum_{j=1}^k \pi_j = 1$. Here, $\Theta = (\pi_j, M_j, \Omega_j, \Sigma_j; j = 1, 2, \dots, k)$ is the parametric space of the mixture density.

3.1. Maximum likelihood estimation

Consider a sample $X_1, X_2, \dots, X_n$ (say $\mathbf{X}$) of size n assumed to be drawn from (6), the log-likelihood function is,

$$l(\Theta|\mathbf{X}) = \sum_{i=1}^n \log \left\{ \sum_{j=1}^k \pi_j f(X_i; M_j, \Omega_j, \Sigma_j) \right\}. \quad (7)$$

Since the summation is involved within the ‘log’ function, estimation of parameters through (7) is difficult. The EM algorithm is employed to overcome this and estimate parameters efficiently. Here, the observed data is incomplete as there is no information about the component membership of observations. Let z be a multinomial random variable that denotes from which component density each observation came. *i.e.*,

$$z_{ij} = \begin{cases} 1 & ; \text{ if } X_i \text{ came from } j^{th} \text{ component density.} \\ 0 & ; \text{ elsewhere.} \end{cases}$$

Here, $\sum_{j=1}^k z_{ij} = 1$. From the definition of finite mixture density,

$$f(z_{ij} = 1) = \pi_j \text{ and } g(X_i|z_{ij} = 1) = f(X_i; M_j, \Omega_j, \Sigma_j). \quad (8)$$

Further, using the marginal distribution of z_{ij} and the conditional distribution of $X_i|z_{ij} = 1$, the joint density of the complete data (X_i, z_i) can be written as,

$$\begin{aligned} f(X, z) &= f(z_{ij} = 1)g(X_i|z_{ij} = 1) \\ &= \prod_{j=1}^k \{ \pi_j f(X_i; M_j, \Omega_j, \Sigma_j) \}^{z_{ij}} \end{aligned} \quad (9)$$

By utilizing the information about z_{ij} , the summation in the mixture density replaced by the product, which makes the further parameter estimation easy. Using (9), for a random sample of size n , the log-likelihood function of the complete-data becomes,

$$\begin{aligned} l(\Theta|\mathbf{X}, z) &= \sum_{i=1}^n \sum_{j=1}^k z_{ij} \log \{ \pi_j f(X_i; M_j, \Omega_j, \Sigma_j) \} \\ &= \sum_{i=1}^n \sum_{j=1}^k z_{ij} \log \left\{ \pi_j \frac{1}{(2\pi)^{\frac{rc}{2}} |\Sigma_j|^{\frac{r}{2}} |\Omega_j|^{\frac{c}{2}} \prod_{p=1}^r \prod_{q=1}^c x_{pq}} \times \right. \\ &\quad \left. \exp \left\{ -\frac{1}{2} \text{tr} \left[\Sigma_j^{-1} (\log X_i - M_j) \Omega_j^{-1} (\log X_i - M_j)^T \right] \right\} \right\} \end{aligned} \quad (10)$$

3.2. EM algorithm

The maximization of (10) can be done through EM-algorithm given in Algorithm 2. Each iteration of E-step calculates $Q(\Theta, \Theta^{(t)})$ which is the expected value of the log-likelihood

Algorithm 2 EM algorithm: FMMVLN distributions

```

1: procedure FMMVLN( $X, k$ )
2:   for  $X = \{X_i\}_{i=1}^n$  &  $j = 1, \dots, k$  do ▷ Initialization
3:     Initialize  $M_j$  &  $\pi_j$  from Extended K-means.
4:     Set  $\Omega_j$  &  $\Sigma_j$  as identity matrices.
5:   end for
6:   while  $M_j^{(t+1)} \neq M_j^{(t)}$  do
7:     for  $j = 1, \dots, k$  do
8:       Update  $\gamma_{ij}$  according to (11). ▷ E-step
9:       Update  $(\widehat{M}_j, \widehat{\Omega}_j, \widehat{\Sigma}_j, \widehat{\pi}_j)$  according to (12). ▷ M-step
10:    end for
11:  end while
12:  Calculate AIC and BIC.
13: return  $M_j, \Omega_j, \Sigma_j, \pi_j, \gamma_{ij}$ , AIC, and BIC.
14: end procedure

```

of complete data w.r.t the conditional distribution of latent variable z , given the observed data, at the current value of Θ as $\Theta^{(t)}$.

Using Bayes theorem, the posterior probability that an observed matrix belongs to the j^{th} component density can be written as,

$$\begin{aligned}
 E[z_{ij} | \mathcal{X}; \Theta^{(t)}] &= f(z_{ij} = 1 | X_i; \Theta^{(t)}) \\
 &= \frac{f(z_{ij} = 1) g(X_i; \Theta^{(t)} | z_{ij} = 1)}{g(X_i; \Theta^{(t)})} \\
 &= \frac{\pi_j^{(t)} f(X_i; M_j^{(t)}, \Omega_j^{(t)}, \Sigma_j^{(t)})}{\sum_{j=1}^k \pi_j^{(t)} f(X_i; M_j^{(t)}, \Omega_j^{(t)}, \Sigma_j^{(t)})} = \gamma_{ij}^{(t)} \quad (\text{say}).
 \end{aligned} \tag{11}$$

Now, the expected value of the complete data log-likelihood, $Q(\Theta, \Theta^{(t)})$ is given as,

$$\begin{aligned}
 Q(\Theta, \Theta^{(t)}) &= E_{z|X_i; \Theta^{(t)}} \left\{ \sum_{i=1}^n \sum_{j=1}^k z_{ij} \log \{ \pi_j f(X_i; M_j, \Omega_j, \Sigma_j) \} \right\} \\
 &= \sum_{i=1}^n \sum_{j=1}^k E[z_{ij} | X_i; \Theta^{(t)}] \log \{ \pi_j f(X_i; M_j, \Omega_j, \Sigma_j) \} \\
 &= \sum_{i=1}^n \sum_{j=1}^k \gamma_{ij}^{(t)} \left\{ \log \pi_j - \frac{rc}{2} \log(2\pi) - \frac{c}{2} \log |\Omega_j| - \frac{r}{2} \log |\Sigma_j| \right. \\
 &\quad \left. - \sum_{p=1}^r \sum_{q=1}^c x_{pq} - \frac{1}{2} \text{tr} \left[\Sigma_j^{-1} (\log X_i - M_j) \Omega_j^{-1} (\log X_i - M_j)^T \right] \right\}
 \end{aligned}$$

The M-step maximizes $Q(\Theta, \Theta^{(t)})$ by obtaining derivatives with respect to the mixture parameters and equating to zero. Here, the estimates of π_j are obtained under the constraints

$0 < \pi_j < 1$ and $\sum_{j=1}^k \pi_j = 1$. The maximum likelihood estimates are given below:

$$\begin{aligned}
 \widehat{M}_j^{(t+1)} &= \frac{\sum_{i=1}^n \gamma_{ij}^{(t)} \log X_i}{\sum_{i=1}^n \gamma_{ij}^{(t)}} \\
 \widehat{\Omega}_j^{(t+1)} &= \frac{\sum_{i=1}^n \gamma_{ij}^{(t)} (\log X_i - \widehat{M}_j^{(t+1)}) \widehat{\Sigma}_j^{(t)-1} (\log X_i - \widehat{M}_j^{(t+1)})^T}{c \sum_{i=1}^n \gamma_{ij}^{(t)}} \\
 \widehat{\Sigma}_j^{(t+1)} &= \frac{\sum_{i=1}^n \gamma_{ij}^{(t)} (\log X_i - \widehat{M}_j^{(t+1)})^T \widehat{\Omega}_j^{(t+1)-1} (\log X_i - \widehat{M}_j^{(t+1)})}{r \sum_{i=1}^n \gamma_{ij}^{(t)}} \\
 \widehat{\pi}_j^{(t+1)} &= \frac{\sum_{i=1}^n \gamma_{ij}^{(t)}}{n}
 \end{aligned} \tag{12}$$

The E and M steps are iteratively repeated until the convergence is reached. The step-by-step procedure of EM algorithm is given in Algorithm 2. In the context of finite mixture models, initialization is the major challenge, as the estimates of the EM algorithm are sensitive to the initialization of parameter values. Different starting values can provide different local maxima and convergence rates of the EM algorithm. Consequently, the clustering results also vary; hence, having a unique clustering solution is difficult. The next section deals with the necessary illustrations of the proposed methodology with simulated and real data studies.

4. Simulation studies

4.1. Extended K-means clustering

To evaluate the performance of the extended K-means clustering, we considered three different sample sizes: 250, 500, and 750, with two and three clusters. The data dimension for two clusters is considered as 2×2 and 3×4 for three clusters. A total of 100 datasets were generated from matrix variate normal populations, with mean matrices chosen arbitrarily to maintain some degree of overlapping. The covariance matrices were set as identity matrices. We then varied the proportion of cluster sizes to create different scenarios for each combination. The extended K-means algorithm was applied to all 100 datasets ranging the cluster size from 1 to 6, and the within sum of squares are evaluated. The average within sum of squares of 100 datasets are plotted against the number of clusters.

The optimal number of clusters are chosen based on the ‘elbow’ of scree plot which indicates that adding more clusters does not significantly decrease the within sum of squares. The clustering performance is evaluated using Adjusted Rand Index (ARI) (Rand, 1971), a widely used metric which provides a comprehensive assessment of the clustering quality. The ARI value ranges from 0 to 1, where 0 indicates the poorest classification and 1 represents perfect classification. From Figure 1, the scree plots in the first row shows that the optimal clusters size is two while the second row reveals that there are three clusters. Hence, the results with corresponding cluster sizes are given Table 1. From these results, the estimated proportion of cluster sizes and centre matrices closely match the real parameter values, indicating that the extended K-means is adequate for parameter estimation. Besides, for a fixed overlapping between clusters, the ARI values of 2×2 data are increasing with sam-

ple size. However, ARI values of 3×4 data are substantially closer to 1 across all sample sizes, indicating that the performance of clustering improves as the sample size and data dimension increases. These outcomes demonstrate the suggested extended K-means method's effectiveness in clustering. Further, as the sample size grows, the standard deviation of every estimate drops, showing the consistency of the estimates.

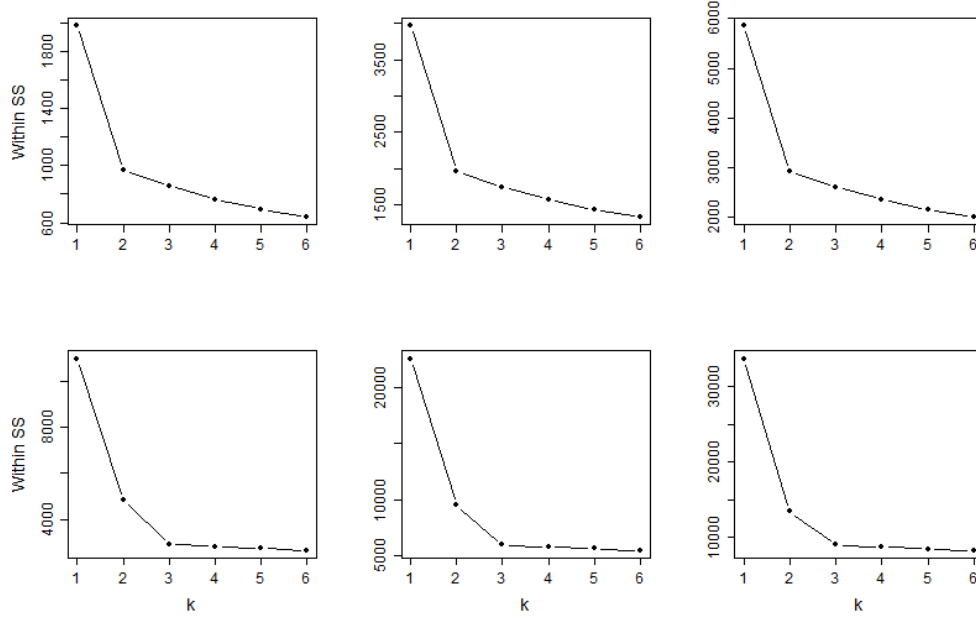


Figure 1: Scree plots for within-cluster sum of squares against number of clusters. The first row represents two cluster data and the second row represents three cluster data

4.2. Finite mixtures of MVLN distributions

In order to exhibit the behaviour of the MVLN mixture model, illustrations are given with two and three-component mixture models and component densities are generated. All the simulations under the two-component mixture model are done at 250, 500, and 750 sample sizes, whereas under the three-component mixture model, simulations are performed with 500 and 750 sample sizes. Here, 100 samples generated from each of the parameter combinations.

A two-component FMMVLN model with data dimension of (2×3) is considered and different combination of parameters are used to generate the samples for each sample size given in Table 2. The mean matrices of all models are chosen arbitrarily to maintain some degree of overlapping. For the sample size of 250, the covariance matrices within clusters considered as homogenous. For the sample size of 500, the row and column covariance matrices between clusters are kept homogenous. For the sample size of 750, the row and column covariance matrices within and between clusters considered heterogeneous. For the case of three-component FMMVLN model, two sample sizes 500 and 750 with the data dimension of (3×3) is considered. A similar approach to the two-component model is considered to generate mean and covariance matrices.

Table 1: Results of extended K-means clustering for simulated data

k	n	C	λ	c	$\hat{\lambda}$ (sd)	$\hat{c}$ (sd)	ARI (sd)
2	250	1	0.5	$\mathbf{0}_{22}$	0.499 (0.013)	$\begin{bmatrix} -0.011 & 0.004 \\ -0.007 & -0.007 \end{bmatrix} \left(\begin{bmatrix} 0.093 & 0.103 \\ 0.092 & 0.102 \end{bmatrix} \right)$	0.904 (0.038)
		2	0.5	$2 \mathbf{J}_{22}$	0.501 (0.013)	$\begin{bmatrix} 2.002 & 2.004 \\ 1.992 & 2.007 \end{bmatrix} \left(\begin{bmatrix} 0.095 & 0.094 \\ 0.093 & 0.102 \end{bmatrix} \right)$	
	500	1	0.45	$\mathbf{0}_{22}$	0.453 (0.007)	$\begin{bmatrix} -0.004 & 0.005 \\ -0.004 & 0.001 \end{bmatrix} \left(\begin{bmatrix} 0.073 & 0.078 \\ 0.073 & 0.074 \end{bmatrix} \right)$	0.911 (0.026)
		2	0.55	$2 \mathbf{J}_{22}$	0.547 (0.007)	$\begin{bmatrix} 2.020 & 2.021 \\ 2.002 & 2.015 \end{bmatrix} \left(\begin{bmatrix} 0.063 & 0.067 \\ 0.064 & 0.061 \end{bmatrix} \right)$	
	750	1	0.4	$\mathbf{0}_{22}$	0.406 (0.006)	$\begin{bmatrix} 0.006 & 0.008 \\ 0.002 & 0.007 \end{bmatrix} \left(\begin{bmatrix} 0.061 & 0.069 \\ 0.063 & 0.067 \end{bmatrix} \right)$	0.911 (0.020)
		2	0.6	$2 \mathbf{J}_{22}$	0.594 (0.006)	$\begin{bmatrix} 2.022 & 2.014 \\ 2.010 & 2.020 \end{bmatrix} \left(\begin{bmatrix} 0.052 & 0.050 \\ 0.048 & 0.051 \end{bmatrix} \right)$	
3	250	1	0.33	$\mathbf{0}_{34}$	0.329 (0.001)	$\begin{bmatrix} 0.001 & -0.009 & -0.008 & -0.013 \\ 0.004 & -0.006 & 0.011 & 0.007 \\ 0.022 & -0.013 & -0.007 & 0.005 \end{bmatrix} \left(\begin{bmatrix} 0.112 & 0.108 & 0.116 & 0.111 \\ 0.109 & 0.112 & 0.103 & 0.121 \\ 0.118 & 0.122 & 0.109 & 0.110 \end{bmatrix} \right)$	1 (0.002)
		2	0.33	$2 \mathbf{J}_{34}$	0.330 (0.001)	$\begin{bmatrix} 1.982 & 1.984 & 2.009 & 1.994 \\ 1.982 & 2.029 & 1.998 & 2.008 \\ 2.009 & 2.000 & 2.009 & 2.017 \end{bmatrix} \left(\begin{bmatrix} 0.112 & 0.123 & 0.116 & 0.110 \\ 0.104 & 0.110 & 0.108 & 0.107 \\ 0.111 & 0.117 & 0.123 & 0.094 \end{bmatrix} \right)$	
						$\begin{bmatrix} 3.992 & 4.001 & 3.974 & 4.012 \\ 3.985 & 4.014 & 4.002 & 3.988 \end{bmatrix} \left(\begin{bmatrix} 0.115 & 0.110 & 0.102 & 0.098 \\ 0.126 & 0.100 & 0.104 & 0.097 \end{bmatrix} \right)$	
						$\begin{bmatrix} 3.984 & 3.993 & 4.006 & 4.007 \end{bmatrix} \left(\begin{bmatrix} 0.107 & 0.109 & 0.109 & 0.104 \end{bmatrix} \right)$	
		3	0.34	$4 \mathbf{J}_{34}$	0.341 (0.001)	$\begin{bmatrix} -0.011 & -0.007 & -0.001 & -0.004 \\ -0.006 & 0.015 & 0.003 & 0.002 \\ 0.006 & -0.011 & 0.002 & 0.012 \end{bmatrix} \left(\begin{bmatrix} 0.070 & 0.075 & 0.073 & 0.071 \\ 0.068 & 0.069 & 0.066 & 0.068 \\ 0.079 & 0.077 & 0.073 & 0.070 \end{bmatrix} \right)$	
						$\begin{bmatrix} 2.007 & 2.009 & 1.993 & 2.010 \\ 2.001 & 1.997 & 1.994 & 1.996 \\ 2.002 & 2.000 & 2.008 & 2.004 \end{bmatrix} \left(\begin{bmatrix} 0.081 & 0.087 & 0.074 & 0.079 \\ 0.072 & 0.085 & 0.083 & 0.076 \\ 0.075 & 0.071 & 0.082 & 0.076 \end{bmatrix} \right)$	
						$\begin{bmatrix} 4.003 & 3.995 & 3.998 & 4.011 \\ 4.014 & 4.007 & 3.997 & 3.985 \\ 4.006 & 4.005 & 3.996 & 3.997 \end{bmatrix} \left(\begin{bmatrix} 0.077 & 0.076 & 0.084 & 0.083 \\ 0.086 & 0.078 & 0.079 & 0.076 \\ 0.086 & 0.077 & 0.084 & 0.085 \end{bmatrix} \right)$	
	500	1	0.4	$\mathbf{0}_{34}$	0.4 (0.000)	$\begin{bmatrix} -0.002 & -0.001 & -0.004 & 0.001 \\ -0.001 & 0.005 & 0.000 & -0.003 \\ 0.003 & -0.006 & 0.004 & 0.008 \end{bmatrix} \left(\begin{bmatrix} 0.052 & 0.059 & 0.046 & 0.055 \\ 0.050 & 0.056 & 0.051 & 0.045 \\ 0.054 & 0.053 & 0.054 & 0.048 \end{bmatrix} \right)$	0.999 (0.002)
		2	0.25	$2 \mathbf{J}_{34}$	0.251 (0.001)	$\begin{bmatrix} 1.996 & 1.996 & 1.994 & 2.018 \\ 2.009 & 2.010 & 1.996 & 1.995 \\ 2.001 & 2.003 & 1.994 & 1.995 \end{bmatrix} \left(\begin{bmatrix} 0.072 & 0.067 & 0.069 & 0.075 \\ 0.075 & 0.077 & 0.073 & 0.072 \\ 0.076 & 0.074 & 0.077 & 0.074 \end{bmatrix} \right)$	
						$\begin{bmatrix} 3.988 & 3.996 & 3.994 & 4.005 \\ 4.013 & 3.987 & 3.999 & 3.997 \end{bmatrix} \left(\begin{bmatrix} 0.068 & 0.070 & 0.072 & 0.076 \\ 0.071 & 0.074 & 0.070 & 0.071 \end{bmatrix} \right)$	
						$\begin{bmatrix} 3.989 & 3.996 & 4.000 & 3.993 \end{bmatrix} \left(\begin{bmatrix} 0.070 & 0.068 & 0.079 & 0.072 \end{bmatrix} \right)$	
		3	0.25	$4 \mathbf{J}_{34}$	0.250 (0.000)	$\begin{bmatrix} 3.988 & 3.996 & 3.994 & 4.005 \\ 4.013 & 3.987 & 3.999 & 3.997 \\ 3.989 & 3.996 & 4.000 & 3.993 \end{bmatrix} \left(\begin{bmatrix} 0.068 & 0.070 & 0.072 & 0.076 \\ 0.071 & 0.074 & 0.070 & 0.071 \end{bmatrix} \right)$	

Here, $\mathbf{0}_{ij}$ and $\mathbf{J}_{ij}$ are the null matrix and matrix of ones of order $(i \times j)$, respectively.

To assess the model fit and identify the number of components in model, we use Akaike Information Criterion (AIC) (Akaike, 1974) and Bayesian Information Criterion (BIC) (Schwarz, 1978), the effective and widely used metrics. We consider AIC and BIC formulations where higher values indicate better model fit. From (12), the estimates of location parameters of the model are the functions of ‘log X ’. Hence, the proposed K-means is fitted to ‘log X ’, and the initial values are obtained for each sample generated. From the results of extended K-means, the cluster centres and the proportion of cluster sizes are used to initialize the mean matrices and mixing proportions, respectively. The initialization for

both row and column covariance matrices is considered identity matrices of the corresponding order. Then, the FMMVLN model is fitted through the EM algorithm, varying number of components from 1 to 6. We calculated the ARI to evaluate the model's clustering performance and misclassification rate (MCR) to measure the proportion of incorrectly classified observations.

Table 2: Parameters used to simulate data from two and three-component FM-MVLN models

K	n	k^a	π	M	Ω	Σ	
2	250	1	0.35	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$	
		2	0.65	$\begin{bmatrix} 0.6 & 0.6 & 0.6 \\ 0.6 & 0.6 & 0.6 \end{bmatrix}$	$\begin{bmatrix} 0.75 & 0 \\ 0 & 0.75 \end{bmatrix}$	$\begin{bmatrix} 0.75 & 0 & 0 \\ 0 & 0.75 & 0 \\ 0 & 0 & 0.75 \end{bmatrix}$	
	500	1	0.4	$\begin{bmatrix} 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 \end{bmatrix}$	$\begin{bmatrix} 0.75 & 0.25 \\ 0.25 & 0.75 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0.5 \\ 0 & 1 & 0.25 \\ 0.5 & 0.25 & 1 \end{bmatrix}$	
		2	0.6	$\begin{bmatrix} 1.8 & 1.8 & 1.8 \\ 1.8 & 1.8 & 1.8 \end{bmatrix}$	$\begin{bmatrix} 0.75 & 0.25 \\ 0.25 & 0.75 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0.5 \\ 0 & 1 & 0.25 \\ 0.5 & 0.25 & 1 \end{bmatrix}$	
	750	1	0.5	$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0.4 \\ 0.4 & 0.75 \end{bmatrix}$	$\begin{bmatrix} 1 & 0.25 & 0 \\ 0.25 & 1 & 0.25 \\ 0 & 0.25 & 1 \end{bmatrix}$	
		2	0.5	$\begin{bmatrix} 2.5 & 2.5 & 2.5 \\ 2.5 & 2.5 & 2.5 \end{bmatrix}$	$\begin{bmatrix} 1.25 & 0.25 \\ 0.25 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.75 & 0 & 0.25 \\ 0 & 1 & 0.5 \\ 0.25 & 0.5 & 1.25 \end{bmatrix}$	
3	500	1	0.33	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$	$\begin{bmatrix} 0.75 & 0 & 0 \\ 0 & 0.75 & 0 \\ 0 & 0 & 0.75 \end{bmatrix}$	
		2	0.33	$\begin{bmatrix} 0.7 & 0.7 & 0.7 \\ 0.7 & 0.7 & 0.7 \\ 0.7 & 0.7 & 0.7 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$	$\begin{bmatrix} 0.75 & 0 & 0 \\ 0 & 0.75 & 0 \\ 0 & 0 & 0.75 \end{bmatrix}$	
				3	0.34	$\begin{bmatrix} 1.4 & 1.4 & 1.4 \\ 1.4 & 1.4 & 1.4 \\ 1.4 & 1.4 & 1.4 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$
		750	1	0.30	$\begin{bmatrix} 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
					2	0.30	$\begin{bmatrix} 1.2 & 1.2 & 1.2 \\ 1.2 & 1.2 & 1.2 \\ 1.2 & 1.2 & 1.2 \end{bmatrix}$
		3	0.40	$\begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$	
	$\begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}$			$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \end{bmatrix}$		

^a represents the component label in mixture model.

Table 3: Number of samples with best AIC and BIC values out of 100 samples for different combinations of K , n , and g

K	2						3			
n	250		500		750		500		750	
g	AIC	BIC	AIC	BIC	AIC	BIC	AIC	BIC	AIC	BIC
1	0	11	0	5	0	0	0	0	0	0
2	52	89	50	95	49	100	0	0	0	0
3	24	0	18	0	16	0	54	100	55	100
4	12	0	11	0	14	0	19	0	26	0
5	8	0	10	0	12	0	13	0	11	0
6	4	0	11	0	9	0	14	0	8	0

Table 4: Estimation results of two-component FMMVLN model

n	k^a	$\hat{\pi}$ (sd)	$\hat{M}$ (sd)			$\hat{\Omega}$ (sd)		$\hat{\Sigma}$ (sd)		
250	1	0.349	$\begin{bmatrix} -0.001 & -0.004 & -0.002 \\ 0.001 & 0.003 & 0.001 \end{bmatrix}$	$\begin{bmatrix} 0.238 & -0.001 \\ -0.001 & 0.248 \end{bmatrix}$			$\begin{bmatrix} 1.007 & -0.017 & -0.014 \\ -0.017 & 1.000 & -0.001 \\ -0.014 & -0.001 & 1.009 \end{bmatrix}$			
		(0.038)	$\left(\begin{bmatrix} 0.064 & 0.073 & 0.065 \\ 0.075 & 0.071 & 0.075 \end{bmatrix} \right)$	$\left(\begin{bmatrix} 0.031 & 0.023 \\ 0.023 & 0.030 \end{bmatrix} \right)$			$\left(\begin{bmatrix} 0.111 & 0.099 & 0.104 \\ 0.099 & 0.120 & 0.109 \\ 0.104 & 0.109 & 0.111 \end{bmatrix} \right)$			
	2	0.651	$\begin{bmatrix} 0.594 & 0.598 & 0.599 \\ 0.599 & 0.598 & 0.604 \end{bmatrix}$	$\begin{bmatrix} 0.563 & 0.000 \\ 0.000 & 0.558 \end{bmatrix}$			$\begin{bmatrix} 0.999 & -0.009 & 0.011 \\ -0.009 & 0.991 & -0.006 \\ 0.011 & -0.006 & 1.001 \end{bmatrix}$			
		(0.038)	$\left(\begin{bmatrix} 0.068 & 0.060 & 0.067 \\ 0.067 & 0.069 & 0.067 \end{bmatrix} \right)$	$\left(\begin{bmatrix} 0.041 & 0.031 \\ 0.031 & 0.043 \end{bmatrix} \right)$			$\left(\begin{bmatrix} 0.077 & 0.061 & 0.059 \\ 0.061 & 0.069 & 0.071 \\ 0.059 & 0.071 & 0.073 \end{bmatrix} \right)$			
500	1	0.403	$\begin{bmatrix} 0.508 & 0.510 & 0.505 \\ 0.502 & 0.504 & 0.496 \end{bmatrix}$	$\begin{bmatrix} 0.735 & 0.247 \\ 0.247 & 0.734 \end{bmatrix}$			$\begin{bmatrix} 1.020 & -0.005 & 0.509 \\ -0.005 & 1.024 & 0.248 \\ 0.509 & 0.248 & 1.013 \end{bmatrix}$			
		(0.040)	$\left(\begin{bmatrix} 0.095 & 0.086 & 0.089 \\ 0.097 & 0.087 & 0.088 \end{bmatrix} \right)$	$\left(\begin{bmatrix} 0.059 & 0.047 \\ 0.047 & 0.052 \end{bmatrix} \right)$			$\left(\begin{bmatrix} 0.073 & 0.070 & 0.066 \\ 0.070 & 0.071 & 0.065 \\ 0.066 & 0.065 & 0.060 \end{bmatrix} \right)$			
	2	0.597	$\begin{bmatrix} 1.802 & 1.808 & 1.804 \\ 1.806 & 1.814 & 1.809 \end{bmatrix}$	$\begin{bmatrix} 0.731 & 0.239 \\ 0.239 & 0.727 \end{bmatrix}$			$\begin{bmatrix} 1.021 & 0.006 & 0.502 \\ 0.006 & 1.026 & 0.251 \\ 0.502 & 0.251 & 1.012 \end{bmatrix}$			
		(0.040)	$\left(\begin{bmatrix} 0.073 & 0.066 & 0.070 \\ 0.071 & 0.076 & 0.073 \end{bmatrix} \right)$	$\left(\begin{bmatrix} 0.043 & 0.030 \\ 0.030 & 0.046 \end{bmatrix} \right)$			$\left(\begin{bmatrix} 0.048 & 0.043 & 0.045 \\ 0.043 & 0.055 & 0.043 \\ 0.045 & 0.043 & 0.050 \end{bmatrix} \right)$			
750	1	0.500	$\begin{bmatrix} 1.007 & 0.997 & 0.995 \\ 1.006 & 0.993 & 0.998 \end{bmatrix}$	$\begin{bmatrix} 0.992 & 0.395 \\ 0.395 & 0.740 \end{bmatrix}$			$\begin{bmatrix} 1.008 & 0.250 & 0.004 \\ 0.250 & 1.003 & 0.254 \\ 0.004 & 0.254 & 1.013 \end{bmatrix}$			
		(0.020)	$\left(\begin{bmatrix} 0.067 & 0.061 & 0.059 \\ 0.058 & 0.051 & 0.048 \end{bmatrix} \right)$	$\left(\begin{bmatrix} 0.045 & 0.028 \\ 0.028 & 0.035 \end{bmatrix} \right)$			$\left(\begin{bmatrix} 0.050 & 0.043 & 0.039 \\ 0.043 & 0.049 & 0.041 \\ 0.039 & 0.041 & 0.044 \end{bmatrix} \right)$			
	2	0.500	$\begin{bmatrix} 2.498 & 2.498 & 2.497 \\ 2.496 & 2.491 & 2.500 \end{bmatrix}$	$\begin{bmatrix} 1.203 & 0.240 \\ 0.240 & 0.959 \end{bmatrix}$			$\begin{bmatrix} 0.779 & 0.000 & 0.262 \\ 0.000 & 1.042 & 0.513 \\ 0.262 & 0.513 & 1.291 \end{bmatrix}$			
		(0.020)	$\left(\begin{bmatrix} 0.067 & 0.057 & 0.076 \\ 0.053 & 0.070 & 0.073 \end{bmatrix} \right)$	$\left(\begin{bmatrix} 0.052 & 0.041 \\ 0.041 & 0.045 \end{bmatrix} \right)$			$\left(\begin{bmatrix} 0.039 & 0.042 & 0.039 \\ 0.042 & 0.043 & 0.044 \\ 0.039 & 0.044 & 0.057 \end{bmatrix} \right)$			

^a represents the component label in mixture model.

Table 5: Estimation results of three-component FMMVLN model

n	k^a	$\hat{\pi}$ (sd)	$\hat{M}$ (sd)	$\hat{\Omega}$ (sd)	$\hat{\Sigma}$ (sd)
500	1	0.330 (0.015)	$\begin{bmatrix} 0.000 & 0.003 & 0.005 \\ 0.006 & 0.006 & -0.002 \\ 0.008 & -0.001 & 0.003 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.048 & 0.047 & 0.049 \\ 0.052 & 0.055 & 0.047 \\ 0.051 & 0.052 & 0.047 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.373 & -0.002 & 0.002 \\ -0.002 & 0.375 & 0.001 \\ 0.002 & 0.001 & 0.373 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.028 & 0.021 & 0.018 \\ 0.021 & 0.027 & 0.016 \\ 0.018 & 0.016 & 0.023 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 1.005 & -0.003 & -0.002 \\ -0.003 & 0.991 & 0.000 \\ -0.002 & 0.000 & 0.992 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.050 & 0.045 & 0.054 \\ 0.045 & 0.056 & 0.052 \\ 0.054 & 0.052 & 0.057 \end{bmatrix} \end{pmatrix}$
	2	0.332 (0.018)	$\begin{bmatrix} 0.704 & 0.700 & 0.707 \\ 0.703 & 0.701 & 0.695 \\ 0.707 & 0.712 & 0.706 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.058 & 0.057 & 0.056 \\ 0.061 & 0.061 & 0.052 \\ 0.060 & 0.057 & 0.053 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.379 & -0.001 & 0.003 \\ -0.001 & 0.380 & 0.005 \\ 0.003 & 0.005 & 0.383 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.028 & 0.019 & 0.019 \\ 0.019 & 0.027 & 0.022 \\ 0.019 & 0.022 & 0.028 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.996 & 0.000 & 0.003 \\ 0.000 & 0.990 & -0.002 \\ 0.003 & -0.002 & 0.969 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.055 & 0.055 & 0.056 \\ 0.055 & 0.059 & 0.048 \\ 0.056 & 0.048 & 0.062 \end{bmatrix} \end{pmatrix}$
	3	0.338 (0.014)	$\begin{bmatrix} 1.404 & 1.399 & 1.407 \\ 1.402 & 1.395 & 1.406 \\ 1.400 & 1.397 & 1.407 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.048 & 0.045 & 0.050 \\ 0.053 & 0.042 & 0.049 \\ 0.058 & 0.050 & 0.049 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.372 & 0.001 & -0.003 \\ 0.001 & 0.374 & 0.001 \\ -0.003 & 0.001 & 0.376 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.024 & 0.018 & 0.016 \\ 0.018 & 0.028 & 0.019 \\ 0.016 & 0.019 & 0.025 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.992 & -0.002 & 0.001 \\ -0.002 & 1.001 & 0.001 \\ 0.001 & 0.001 & 0.994 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.049 & 0.044 & 0.050 \\ 0.044 & 0.052 & 0.050 \\ 0.050 & 0.050 & 0.051 \end{bmatrix} \end{pmatrix}$
750	1	0.299 (0.018)	$\begin{bmatrix} 0.504 & 0.500 & 0.499 \\ 0.502 & 0.503 & 0.507 \\ 0.484 & 0.496 & 0.502 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.065 & 0.077 & 0.067 \\ 0.078 & 0.086 & 0.074 \\ 0.090 & 0.078 & 0.081 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.999 & -0.005 & -0.001 \\ -0.005 & 0.993 & -0.003 \\ -0.001 & -0.003 & 0.994 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.060 & 0.049 & 0.046 \\ 0.049 & 0.063 & 0.041 \\ 0.046 & 0.041 & 0.058 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.998 & -0.001 & 0.001 \\ -0.001 & 1.000 & 0.001 \\ 0.001 & 0.001 & 1.002 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.048 & 0.041 & 0.045 \\ 0.041 & 0.047 & 0.042 \\ 0.045 & 0.042 & 0.047 \end{bmatrix} \end{pmatrix}$
	2	0.301 (0.018)	$\begin{bmatrix} 1.199 & 1.198 & 1.190 \\ 1.198 & 1.204 & 1.194 \\ 1.192 & 1.203 & 1.203 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.067 & 0.058 & 0.060 \\ 0.067 & 0.062 & 0.062 \\ 0.061 & 0.063 & 0.063 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.559 & 0.004 & -0.005 \\ 0.004 & 0.560 & -0.001 \\ -0.005 & -0.001 & 0.558 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.041 & 0.026 & 0.029 \\ 0.026 & 0.037 & 0.027 \\ 0.029 & 0.027 & 0.040 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 1.003 & 0.008 & 0.005 \\ 0.008 & 0.996 & -0.002 \\ 0.005 & -0.002 & 0.998 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.057 & 0.047 & 0.051 \\ 0.047 & 0.057 & 0.047 \\ 0.051 & 0.047 & 0.056 \end{bmatrix} \end{pmatrix}$
	3	0.400 (0.005)	$\begin{bmatrix} 1.993 & 1.998 & 2.002 \\ 2.002 & 2.000 & 2.001 \\ 1.998 & 2.003 & 1.998 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.030 & 0.027 & 0.029 \\ 0.033 & 0.030 & 0.030 \\ 0.031 & 0.028 & 0.030 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.253 & 0.001 & 0.001 \\ 0.001 & 0.250 & 0.000 \\ 0.001 & 0.000 & 0.253 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.012 & 0.009 & 0.008 \\ 0.009 & 0.013 & 0.009 \\ 0.008 & 0.009 & 0.012 \end{bmatrix} \end{pmatrix}$	$\begin{bmatrix} 0.993 & 0.002 & -0.002 \\ 0.002 & 0.996 & 0.000 \\ -0.002 & 0.000 & 0.992 \end{bmatrix}$ $\begin{pmatrix} \begin{bmatrix} 0.045 & 0.029 & 0.039 \\ 0.029 & 0.041 & 0.033 \\ 0.039 & 0.033 & 0.042 \end{bmatrix} \end{pmatrix}$

^a represents the component label in mixture model.

The results in Table 3 presents the count of samples (out of 100) where different numbers of components (g) results the best AIC and BIC values. From the results, the highest number of samples achieving optimal AIC and BIC values for the two-component model occurs at $g = 2$, and for the three-component model, at $g = 3$. This indicates that the proposed FMMVLN model effectively identifies the correct number of components according to both AIC and BIC criteria. Further, the extended K-means algorithm with two and three clusters are performed. Using the results of extended K-means as initial values, two and three-component FMMVLN models are fitted and the results are presented in Tables 4 and

5, respectively. The results shows that the estimates are close to the actual parameter values. Here, the covariance matrices can only be estimated up to a multiplicative constant as they depend on each other within the component. The standard deviation of all estimates decrease as the sample sizes increase, confirming the estimates' consistency property.

Table 6: The average MCR and ARI values with standard deviations for two and three-component FMMVLN models

Model	n	MCR (sd)	ARI (sd)
2-component	250	0.100 (0.021)	0.637 (0.067)
	500	0.097 (0.017)	0.650 (0.055)
	750	0.077 (0.010)	0.717 (0.035)
3-component	500	0.065 (0.011)	0.818 (0.028)
	750	0.073 (0.009)	0.819 (0.022)

Table 7: Comparison of convergence speed (number of iterations) for random and extended K-means initialization techniques

Model	n	Random initialization Median (IQR)	K-means initialization Median (IQR)
2-component	250	60.5 (32.50)	43.0 (23.25)
	500	83.0 (38.25)	44.0 (24.25)
	750	46.0 (13.00)	28.0 (8.00)
3-component	500	67.5 (44.50)	22.5 (7.00)
	750	50.0 (37.75)	38.0 (12.00)

The model-based clustering using FMMVLN is performed for each dataset of each sample size, and observations are segregated into clusters corresponding to the maximum posterior probabilities of component densities obtained from the EM algorithm. The MCR and ARI values with their standard deviations are reported in Table 6. From these results, as the sample size increases, the MCR values decrease, and the ARI values increase. The standard deviations of MCR and ARI also decrease as the sample size increases. These results indicate that clustering is more reliable for larger sample sizes.

In order to assess the effectiveness of extended K-means as an initialization method in terms of convergence speed, we also fitted the FMMVLN models using the existing random initialization for each of the simulated datasets, noting the number of iterations required to reach convergence. The average (median) number of iterations with inter-quartile range (IQR) for random initialization and the extended K-means initialization are compared in Table 7. From these results, in all cases, the extended K-means initialization technique outperformed the random initialization by achieving convergence with considerably fewer iterations.

5. Landsat data

To assess the performance of the proposed model, we utilize the Landsat satellite data (Srinivasan, 1993), which is well-known and extensively used in image classification

tasks. This data set is available through the UCI Machine Learning Repository and was originally described by Ashwin Srinivasan in 1993. For this analysis, we focus on a test dataset consisting of 2000 samples. The dataset captures multi-spectral pixel values within a 3×3 neighbourhood in satellite images. Each neighbourhood is represented by 36 variables, corresponding to four spectral bands, with each band containing nine pixels. Hence, the dimension of the data is of 4×9 . The pixel classes are numerically coded in the range of 0 and 255 with 0 corresponding to black and 255 to white. The data is categorized into seven distinct classes: red soil, cotton crop, grey soil, damp grey soil, soil with vegetation stubble, a mixture class, and very damp grey soil. However, in this particular dataset, only six classes are present, as there are no observations for the mixture class. The objective of the model is to accurately classify these pixels into their respective categories based on the spectral information provided.

Given that the number of classes is predefined in this scenario, the extended K-means algorithm is initially applied, specifying six clusters to correspond with the six classes. The resulting cluster centers from this K-means analysis are then used to initialize the FMMVLN distributions. Following this, a six-component FMMVLN model is fitted to the data. The parameter estimates obtained from both the extended K-means and the FMMVLN distributions are summarized in Tables 10 and 11, respectively. These results reveal that the center matrices produced by the extended K-means closely align with the mean matrices derived from the FMMVLN distributions, indicating that the K-means initialization provides a solid foundation for the subsequent model fitting.

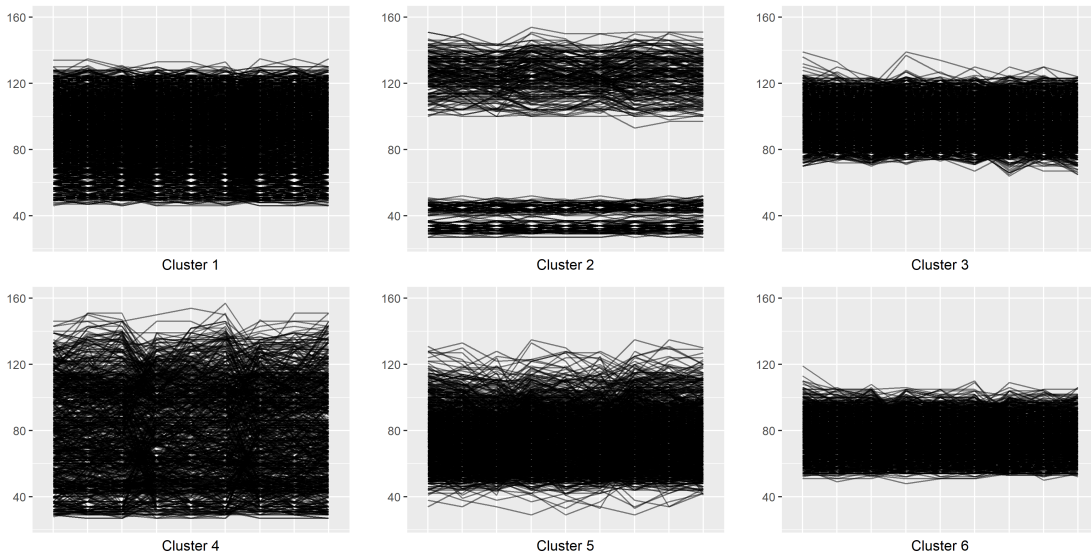


Figure 2: Parallel coordinate plot for the estimated clusters of FMMVLN distributions for Landsat data

However, it is important to note that, as previously discussed, the covariance structures within each component of the mixture model can only be estimated up to a multiplicative constant. As a result, the row covariance matrices Ω_j are estimated to have values close to zero, while the column covariance matrices Σ_j are found to be significantly different from zero. This suggests that while the K-means algorithm effectively initializes the model,

the estimation of covariance matrices requires careful interpretation, particularly given the inherent dependency structures within the model components.

Table 8: Model fit and clustering results of FMMVN, FMMVST, and FMMVLN models

Model	AIC	BIC	ARI
FMMVN	−401435	−404521	0.617
FMMVST	−417662	−421958	0.419
FMMVLN	−400145	−403231	0.638

Table 9: Cluster assignments of (a) FMMVN, (b) FMMVST, and (c) FMMVLN models

(a) FMMVN		Estimated class					
True class		1	2	3	4	5	6
Red soil		442	6	0	12	1	0
Cotton crop		0	222	0	2	0	0
Grey soil		5	3	337	52	0	0
Damp grey soil		1	17	68	29	1	95
Soil with vegetation stubble		9	44	0	62	122	0
Very damp grey soil		0	9	19	106	17	319

(b) FMMVST		Estimated class					
True class		1	2	3	4	5	6
Red soil		433	0	13	14	1	0
Cotton crop		0	197	0	26	1	0
Grey soil		6	0	368	18	5	0
Damp grey soil		1	6	186	16	2	0
Soil with vegetation stubble		22	48	26	63	71	7
Very damp grey soil		2	12	374	17	10	55

(c) FMMVLN		Estimated class					
True class		1	2	3	4	5	6
Red soil		442	0	0	5	14	0
Cotton crop		0	106	0	116	2	0
Grey soil		10	0	354	3	27	3
Damp grey soil		1	0	72	13	15	110
Soil with vegetation stubble		8	0	0	31	197	1
Very damp grey soil		1	0	18	9	78	364

To visually inspect the separability of the estimated clusters, we constructed a parallel coordinate plot presented in Figure 2. In this plot, each observation rows are represented as a line connecting points on parallel axes, with the position of each point corresponding to the value in the columns. We observed substantial overlap between clusters 1, 4, and 5 with clusters 3 and 6, indicating potential similarity or ambiguity in their feature distributions. In contrast, cluster 2 exhibited a distinct pattern with data points occupying two separate value ranges, resulting in a visible gap within the plot.

Additionally, we conducted a comparative analysis by fitting two well-known finite mixture models: the finite mixtures of matrix variate normal (FMMVN) distributions and the finite mixtures of matrix variate Skew- t (FMMVST) distributions. The results of this analysis are presented in Table 8. For this comparison, model-based clustering was performed using the posterior probabilities obtained from each of these finite mixture models, and the ARI values were calculated to evaluate the accuracy of the clustering. The AIC and BIC values were also computed for each model to assess their fit to the data.

The results in Table 8 shows that the proposed model outperformed the FMMVN and FMMVST models, as indicated by its higher AIC and BIC values. This suggests that the proposed model provides a better fit to the data compared to the other two models. Furthermore, the ARI values confirm that the proposed model is more effective in correctly classifying a greater number of pixels into their respective categories, demonstrating its superior performance in this clustering task. Despite the low increment in the ARI value, likely due to substantial overlap between clusters as visualized in the parallel coordinate plot, the proposed model demonstrates superior performance in fitting and clustering skewed matrix variate data overall. We obtained the cluster assignment tables for the three finite mixtures and presented in Table 9. Since the clusters don't follow a specific sequence, the cluster assignment table can only show which observations are assigned to each cluster. However, these tables offer information on how the observations in estimated clusters are distributed among the true classes.

6. Results and discussion

In this paper, we made an attempt to model and cluster the skewed three-way data with underlying sub-populations using FMMVLN distributions. To initialize the FMMVLN models, we introduced an extended K-means algorithm. The subsequent parameter estimation was achieved through the EM algorithm. We employed AIC and BIC to determine the optimal number of components and the model fit in the mixture models. The results demonstrated that BIC exhibited superior performance in accurately identifying the correct number of components compared to AIC. Furthermore, the clustering performance, evaluated using the MCR and ARI values. The computations of simulation and real data studies are carried out in R.

The simulation study findings revealed that the FMMVLN models effectively estimated the true parameter values, with estimation accuracy improving as sample size increased. This confirms the consistency property of the estimators. Furthermore, the clustering performance, evaluated using MCR and ARI demonstrated significant improvements with increasing sample size. This indicates that the proposed FMMVLN-based clustering approach is more reliable for larger datasets. In addition, a comparative analysis of initial-

ization methods highlighted the efficiency of the extended K-means algorithm in terms of convergence speed. The proposed initialization consistently outperformed random initialization, leading to a substantial reduction in the number of iterations required for the EM algorithm to converge.

The real data study reveals that the resulting parameter estimates from extended K-means and FMMVLN model exhibited strong alignment, indicating the effectiveness of the K-means initialization. The comparative analysis with FMMVN and FMMVST models demonstrated the superior performance of the proposed FMMVLN model in terms of model fit and clustering accuracy, as evidenced by higher AIC, BIC, and ARI values. Despite some overlap between clusters, the FMMVLN model effectively handled the skewed matrix variate data, outperforming the alternative models.

One major challenge in finite mixture models for matrix variate data is the initialization of the EM algorithm. To address this, we proposed the extended K-means algorithm to cluster the matrix variate data. Additionally, estimating covariance matrices is complex due to inherent dependencies within the model components. Furthermore, these models often require large sample sizes as the number of parameters increases with data dimensionality and the number of mixture components. We aim to explore these issues further in future research.

Acknowledgements

We thank the editors and the reviewers for their valuable comments and suggestions, which helped improve this article's work.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, **19**, 716–723.
- Biernacki, C., Celeux, G., and Govaert, G. (2003). Choosing starting values for the em algorithm for getting the highest likelihood in multivariate Gaussian mixture models. *Computational Statistics & Data Analysis*, **41**, 561–575.
- Bouguila, N. and ElGuebaly, W. (2009). Discrete data clustering using finite mixture models. *Pattern Recognition*, **42**, 33–42.
- Deepana, R. and Kiruthika (2018). Model based clustering using finite mixture models of lognormal distribution. *Research & Reviews: Journal of Statistics*, **7**, 58–67.
- Deepana, R. and Kiruthika, C. (2022). Clustering using skewed data via finite mixtures of multivariate lognormal distributions. *Statistics and Applications*, **20**, 219–237.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, **39**, 1–38.

- Doğru, F. Z., Bulut, Y. M., and Arslan, O. (2016). Finite mixtures of matrix variate t distributions. *Gazi University Journal of Science*, **29**, 335–341.
- Everitt, B. (2013). *Finite Mixture Distributions*. Springer Science & Business Media.
- Gallaugh, M. P. and McNicholas, P. D. (2018). Finite mixtures of skewed matrix variate distributions. *Pattern Recognition*, **80**, 83–93.
- Hu, Z. (2015). *Initializing the EM Algorithm for Data Clustering and Sub-population Detection*. PhD thesis, The Ohio State University.
- Karlis, D. and Xekalaki, E. (2003). Choosing initial values for the em algorithm for finite mixtures. *Computational Statistics & Data Analysis*, **41**, 577–590.
- McLachlan, G. J., Lee, S. X., and Rathnayake, S. I. (2019). Finite mixture models. *Annual Review of Statistics and Its Application*, **6**, 355–378.
- McNicholas, P. D. and Murphy, T. B. (2008). Parsimonious Gaussian mixture models. *Statistics and Computing*, **18**, 285–296.
- Melnykov, V. and Maitra, R. (2010). Finite mixture models and model-based clustering. *Statistics Surveys*, **4**, 80–116.
- Michael, S. and Melnykov, V. (2016). An effective strategy for initializing the em algorithm in finite mixture models. *Advances in Data Analysis and Classification*, **10**, 563–583.
- Panić, B., Klemenc, J., and Nagode, M. (2020). Improved initialization of the em algorithm for mixture model parameter estimation. *Mathematics*, **8**, 373.
- Peel, D. and MacLachlan, G. (2000). *Finite Mixture Models*. John Wiley & Sons.
- Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, **66**, 846–850.
- Rezaei, A. and Ahmadi, K. (2023). Generalized Mahalanobis distance and its application in detecting matrix outliers. *Filomat*, **37**, 7993–8011.
- Rohde, D., Corcoran, J., White, G., and Huang, R. (2012). Visualization of predictive distributions for discrete spatial-temporal log cox processes approximated with mcmc. In *Intelligent Data Engineering and Automated Learning-IDEAL 2012: 13th International Conference, Natal, Brazil, August 29-31, 2012. Proceedings 13*, pages 286–293. Springer.
- Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, **6**, 461–464.
- Silva, A., Qin, X., Rothstein, S. J., McNicholas, P. D., and Subedi, S. (2023). Finite mixtures of matrix variate poisson-log normal distributions for three-way count data. *Bioinformatics*, **39**, 1–8.
- Srinivasan, A. (1993). Statlog (landsat satellite). UCI Machine Learning Repository.
- Tomarchio, S. D., Gallaugh, M. P., Punzo, A., and McNicholas, P. D. (2022). Mixtures of matrix-variate contaminated normal distributions. *Journal of Computational and Graphical Statistics*, **31**, 413–421.
- Tomarchio, S. D., Punzo, A., and Bagnato, L. (2020). Two new matrix-variate distributions with application in model-based clustering. *Computational Statistics & Data Analysis*, **152**, 107050.
- Viroli, C. (2011a). Finite mixtures of matrix normal distributions for classifying three-way data. *Statistics and Computing*, **21**, 511–522.

Viroli, C. (2011b). Model based clustering for three-way data structures. *Bayesian Anal.*, **6**, 573–602.

ANNEXURE

Table 10: Estimates of cluster centres and cluster sizes of extended K-means for Landsat data

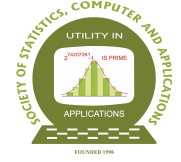
C^a	$\hat{\lambda}^b$	$\hat{c}$
1	0.17	4.13 4.13 4.13 4.13 4.12 4.12 4.13 4.12 4.12
		4.18 4.18 4.19 4.18 4.17 4.18 4.18 4.17 4.17
		4.33 4.32 4.33 4.32 4.30 4.32 4.33 4.32 4.32
		4.09 4.08 4.09 4.08 4.07 4.08 4.10 4.09 4.10
2	0.10	3.83 3.83 3.84 3.83 3.82 3.82 3.85 3.84 3.84
		3.54 3.54 3.57 3.53 3.51 3.53 3.57 3.56 3.56
		4.75 4.76 4.75 4.76 4.76 4.76 4.74 4.74 4.74
		4.81 4.81 4.79 4.82 4.83 4.82 4.80 4.80 4.79
3	0.12	4.05 4.05 4.05 4.05 4.05 4.05 4.05 4.05 4.05
		4.29 4.28 4.28 4.28 4.28 4.27 4.28 4.27 4.27
		4.56 4.55 4.55 4.55 4.55 4.56 4.55 4.56 4.56
		4.40 4.40 4.41 4.39 4.40 4.41 4.40 4.41 4.41
4	0.21	4.29 4.28 4.27 4.28 4.28 4.27 4.28 4.28 4.27
		4.43 4.41 4.40 4.42 4.42 4.41 4.42 4.41 4.41
		4.49 4.48 4.48 4.49 4.48 4.47 4.48 4.48 4.47
		4.25 4.25 4.25 4.25 4.24 4.24 4.25 4.24 4.24
5	0.16	4.21 4.21 4.20 4.21 4.20 4.19 4.19 4.19 4.19
		4.62 4.63 4.61 4.62 4.63 4.62 4.61 4.62 4.61
		4.73 4.74 4.73 4.74 4.74 4.73 4.73 4.74 4.73
		4.53 4.53 4.52 4.53 4.53 4.53 4.53 4.53 4.52
6	0.24	4.45 4.46 4.45 4.46 4.46 4.46 4.45 4.46 4.45
		4.64 4.64 4.63 4.64 4.65 4.64 4.63 4.64 4.63
		4.69 4.69 4.68 4.69 4.69 4.69 4.68 4.69 4.68
		4.45 4.45 4.45 4.46 4.46 4.45 4.45 4.46 4.45

^a represents the cluster label and ^b represents the proportion of cluster size.

Table 11: Continued

k^a	$\hat{\pi}$	$\hat{M}$										$\hat{\Sigma}$									
		$\hat{\Omega}$																			
4	0.24	4.25	4.25	4.25	4.25	4.24	4.24	4.24	4.24	4.24	4.24	0.77	0.50	0.39	0.45	0.42	0.37	0.35	0.34	0.33	0.33
		4.38	4.37	4.38	4.37	4.37	4.37	4.37	4.37	4.37	4.37	0.50	0.75	0.50	0.40	0.43	0.42	0.34	0.34	0.34	0.35
		4.42	4.42	4.42	4.42	4.41	4.42	4.42	4.42	4.41	4.41	0.39	0.50	0.77	0.33	0.36	0.41	0.28	0.29	0.33	0.33
		4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17	4.17	0.45	0.40	0.33	0.75	0.48	0.36	0.44	0.40	0.34	0.34
5	0.23	4.16	4.15	4.15	4.15	4.15	4.14	4.14	4.14	4.14	4.14	1.18	0.93	0.73	0.86	0.78	0.64	0.71	0.65	0.56	0.56
		4.55	4.55	4.55	4.55	4.55	4.55	4.55	4.55	4.55	4.54	0.93	1.17	0.91	0.81	0.82	0.77	0.67	0.66	0.63	0.63
		4.68	4.68	4.68	4.68	4.68	4.68	4.68	4.67	4.68	4.68	0.73	0.91	1.15	0.70	0.78	0.84	0.59	0.64	0.68	0.68
		4.48	4.48	4.48	4.48	4.48	4.48	4.48	4.48	4.48	4.48	0.86	0.81	0.70	1.13	0.87	0.69	0.83	0.75	0.64	0.64
6	0.22	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	0.78	0.82	0.78	0.87	1.06	0.84	0.76	0.78	0.74	0.74
		4.64	4.64	4.64	4.65	4.64	4.64	4.64	4.64	4.64	4.64	0.64	0.77	0.84	0.69	0.84	1.10	0.67	0.74	0.80	0.80
		4.69	4.69	4.68	4.69	4.69	4.69	4.69	4.68	4.69	4.68	0.71	0.67	0.59	0.83	0.76	0.67	1.10	0.84	0.70	0.70
		4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	0.65	0.66	0.64	0.75	0.78	0.74	0.84	1.03	0.82	0.82
6	0.22	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	0.56	0.63	0.68	0.64	0.74	0.80	0.70	0.82	1.06	1.06
		4.64	4.64	4.64	4.65	4.64	4.64	4.64	4.64	4.64	4.64	0.69	0.39	0.30	0.36	0.29	0.24	0.27	0.22	0.18	0.18
		4.69	4.69	4.68	4.69	4.69	4.69	4.69	4.68	4.69	4.68	0.39	0.64	0.40	0.33	0.32	0.29	0.28	0.24	0.22	0.22
		4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	0.30	0.40	0.69	0.31	0.33	0.34	0.25	0.26	0.29	0.29
6	0.22	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	0.36	0.33	0.31	0.66	0.39	0.31	0.37	0.32	0.26	0.26
		4.64	4.64	4.64	4.65	4.64	4.64	4.64	4.64	4.64	4.64	0.29	0.32	0.33	0.39	0.64	0.40	0.33	0.33	0.30	0.30
		4.69	4.69	4.68	4.69	4.69	4.69	4.69	4.68	4.69	4.68	0.24	0.29	0.34	0.31	0.40	0.68	0.31	0.35	0.38	0.38
		4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	0.27	0.28	0.25	0.37	0.33	0.31	0.72	0.40	0.30	0.30
6	0.22	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	0.22	0.24	0.26	0.32	0.33	0.35	0.40	0.65	0.42	0.42
		4.64	4.64	4.64	4.65	4.64	4.64	4.64	4.64	4.64	4.64	0.18	0.22	0.29	0.26	0.30	0.38	0.30	0.42	0.76	0.76
		4.69	4.69	4.68	4.69	4.69	4.69	4.69	4.68	4.69	4.68	0.69	0.39	0.30	0.36	0.29	0.24	0.27	0.22	0.18	0.18
		4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	0.39	0.64	0.40	0.33	0.32	0.29	0.28	0.24	0.22	0.22
6	0.22	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	0.30	0.40	0.69	0.31	0.33	0.34	0.25	0.26	0.29	0.29
		4.64	4.64	4.64	4.65	4.64	4.64	4.64	4.64	4.64	4.64	0.36	0.33	0.31	0.66	0.39	0.31	0.37	0.32	0.26	0.26
		4.69	4.69	4.68	4.69	4.69	4.69	4.69	4.68	4.69	4.68	0.29	0.32	0.33	0.39	0.64	0.40	0.33	0.33	0.30	0.30
		4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	0.24	0.29	0.34	0.31	0.40	0.68	0.31	0.35	0.38	0.38
6	0.22	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	4.46	0.27	0.28	0.25	0.37	0.33	0.31	0.72	0.40	0.30	0.30
		4.64	4.64	4.64	4.65	4.64	4.64	4.64	4.64	4.64	4.64	0.22	0.24	0.26	0.32	0.33	0.35	0.40	0.65	0.42	0.42
		4.69	4.69	4.68	4.69	4.69	4.69	4.69	4.68	4.69	4.68	0.18	0.22	0.29	0.26	0.30	0.38	0.30	0.42	0.76	0.76
		4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	4.45	0.69	0.39	0.30	0.36	0.29	0.24	0.27	0.22	0.18	0.18

^a represents the component label in mixture model.



Balanced and Partially Balanced Semi-Latin Rectangles with Block Size Two

Kaushal Kumar Yadav¹, Sukanta Dash², Rajender Parsad²,
Baidya Nath Mandal³, Anil Kumar⁴ and Mukesh Kumar²

¹*The Graduate School, ICAR-IARI, New Delhi-110012*

²*ICAR-IASRI, Pusa, New Delhi-110012*

³*ICAR-IARI, Gauria Karma, Jharkhand- 825405*

⁴*Krishi Bhavan, ICAR, New Delhi-110001*

Received: 18 January 2024; Revised: 26 August 2024; Accepted: 06 September 2024

Abstract

Balanced Semi-Latin rectangles are the subclass of Semi-Latin rectangles which are generalizations of Latin squares and Semi-Latin squares. Such types of designs are more useful in plant disease experiments where we can consider plants as columns, half-leaves as experimental units, and the height of leaves as rows. In this paper, we study partially balanced semi-Latin rectangles which have all properties of balanced semi-Latin rectangles except that each pair of distinct treatments does not appear a constant number of times in the design. We also propose four construction methods of balanced and partially balanced semi-Latin rectangles. An R package *slr* has been developed to implement the proposed methods of constructions.

Key words: Semi-Latin Rectangles; Balanced Semi-Latin rectangles; Partially Balanced Semi-Latin rectangles; Canonical efficiency factor; Average efficiency factor.

1. Introduction

In agricultural experiments, plant leaves often serve as experimental units (plots), especially in plant disease studies, as detailed by Price (1946). Variability among plants and the impact of leaf height are significant sources of heterogeneity. To address this, row-column designs like Latin squares are recommended, where columns represent plants and rows correspond to leaf height. However, the number of available plants often exceeds the usable leaves per plant. In a notable experiment on tobacco mosaic virus, Youden (1937) introduced row-column designs with fewer rows than columns, now known as Youden squares.

Often row-column designs where more than one treatment can be allocated in each row-column intersection (henceforth called as block or cell) are useful. To this effect, Bailey and Monod (2001) introduced semi-Latin rectangles (SLR) which are a class of row-column

designs where each block has the same size ($k > 1$) and each treatment appears more than or equal to once in each row and/or in each column. Uto and Bailey (2020) proposed balanced semi-Latin rectangles (BSLRs), a subclass that ensures balance among treatment pairs, and provided two algorithms for constructing such designs with block size two. Additionally, Uto and Bailey (2022) introduced regular-graph semi-Latin rectangles, which have the added property that treatment concurrences between any two distinct treatments differ by at most one and proposed two different algorithms for constructing such designs.

Limited resource situations often make it challenging to implement a full BSLR design due to its large structure and resource requirements. In such cases, partially balanced semi-Latin rectangles (PBSLRs) offer greater practicality as they retain all the properties of BSLRs except for the condition that each pair of distinct treatments appears a constant number of times in the design. This distinction sets PBSLRs apart from both BSLRs and general SLRs, making them more suitable for constrained resource scenarios. In a PBSLR design, n_i number of pairs of a treatment appear in λ_i blocks, $i = 1, 2, \dots, m$ (number of association classes). Also, each treatment appears in $hn_r (= pn_c)$ blocks (here, n_r and n_c denote the number of times a treatment appears in each row and in each column, respectively) and each block contains k distinct treatments. Hence, the sum of concurrences of a particular treatment with other treatments is $\sum_i^m n_i \lambda_i$, so

$$\begin{aligned} \sum_i^m n_i \lambda_i &= hn_r(k-1) = pn_c(k-1) \\ \sum_i^m n_i &= v-1 \end{aligned} \tag{1}$$

A little work related to Semi-Latin rectangles with block size two for balanced situation are available in literature. Also, designs available in literature are obtainable by algorithmic methods as proposed by Uto and Bailey (2020). In this paper, we introduce the concept of partially balanced Semi-Latin rectangle designs, where the number of replications is drastically reduced compared to balanced designs and regular-graph SLRs. To avoid the complexity of algorithms, four different construction methods are developed for both balanced and partially balanced situations. These methods of construction are simple and easy to use for generation of these classes of semi-Latin rectangles. Further, an R package named *slr* (Yadav *et al.*, 2023) is developed in which the proposed methods are implemented and users can generate balanced and partially balanced semi-Latin rectangles by providing only the number of treatments.

2. Experimental set up

In plant disease experiments, generally, the leaves of plants are used as experimental units. The researcher is interested in comparing a set of treatments which may be different virus preparations or different doses of a virus, *etc.* A solution containing the treatment is rubbed onto the leaves, which develop infections, and some measurements of the disease such as the number of lesions on the leaves are taken. It has been found that there are two sources of heterogeneity in these types of experiments namely, differences between plants and influence of leaf height. So, row-column designs are applicable in this type of situation.

Table 1: Semi-Latin rectangle for 10 treatments in plant disease experiment

LH 5	1, 10	2, 1	3, 2	4, 3	5, 4	6, 5	7, 6	8, 7	9, 8	10, 9
LH 4	2, 9	3, 10	4, 1	5, 2	6, 3	7, 4	8, 5	9, 6	10, 7	1, 8
LH 3	3, 8	4, 9	5, 10	6, 1	7, 2	8, 3	9, 4	10, 5	1, 6	2, 7
LH 2	4, 7	5, 8	6, 9	7, 10	8, 1	9, 2	10, 3	1, 4	2, 5	3, 6
LH 1	5, 6	6, 7	7, 8	8, 9	9, 10	10, 1	1, 2	2, 3	3, 4	4, 5
	Plant1	Plant2	Plant3	Plant4	Plant5	Plant6	Plant7	Plant8	Plant9	Plant10

Here, rows refer to leaf height and columns to plants. Since, the number of plants available for the experiment is typically more than the number of usable leaves and their positions per plant, so a row-column design which has smaller number of rows than columns is useful in such situations. The researcher is interested to compare more than two treatments in p plants each with leaves at h heights, where typically $h < p$. Generally, the two half leaves of each of the hp leaves form the plots. Considering the leaf heights as rows and the plants as columns, we have two plots in the intersection of each row and each column. Clearly, the Semi-Latin rectangles are useful in such situations. In these designs, rows and columns can be randomized and treatments within blocks can also be randomly allocated.

For a plant disease experiment involving ten plants ($p = 10$) and pairs of half-leaves ($k = 2$) at five leaf heights ($h = 5$), with ten treatments ($v = 10$) can be designed as $(5 \times 10)/2$ SLR given in Table 1.

3. Preliminaries

In the context of assuming the additive effect, the model for the present study is expressed as:

$$\mathbf{y} = \mu \mathbf{1} + \Delta' \boldsymbol{\tau} + \Phi' \boldsymbol{\rho} + \Psi' \boldsymbol{\gamma} + \mathbf{D}' \boldsymbol{\beta} + \boldsymbol{\epsilon} \quad (2)$$

where, $\mathbf{y}$ denotes $hpk \times 1$ vector of observations, $\mathbf{1}$ denotes $hpk \times 1$ vector of ones, Δ' denotes $hpk \times v$ incidence matrix of observations versus treatments, $\boldsymbol{\tau}$ denotes $v \times 1$ vector of treatment effects, Φ' denotes $hpk \times h$ incidence matrix of observations versus rows, $\boldsymbol{\rho}$ denotes $h \times 1$ vector of row effects, Ψ' denotes $hpk \times p$ incidence matrix of observations versus columns, $\boldsymbol{\gamma}$ denotes $p \times 1$ vector of column effects, $\mathbf{D}'$ denotes $hpk \times hp$ incidence matrix of observations versus blocks/cells, $\boldsymbol{\beta}$ denotes $hp \times 1$ vector of block effects, $\boldsymbol{\epsilon}$ denotes $hpk \times 1$ vector of random errors with $\text{Expectation}(\boldsymbol{\epsilon}) = \mathbf{0}$ and $\text{Dispersion}(\boldsymbol{\epsilon}) = \sigma^2 \mathbf{I}_{hpk \times hpk}$ with $(.)'$ denoting transpose of a matrix and $\mathbf{I}_{t \times t}$ representing a $t \times t$ Identity matrix. The incidence matrices Δ' , Φ' , Ψ' and $\mathbf{D}'$ are defined in the usual manner. For instance, if $\Delta' = (\delta_{ui})$, then

$$\delta_{ui} = \begin{cases} 1, & \text{if the } u^{th} \text{ observation corresponds to the } i^{th} \text{ treatment} \\ 0, & \text{otherwise} \end{cases}$$

Under model (2), the information matrix for estimating the linear functions of treatment effects is given by

$$\mathbf{C}_d = \mathbf{R}_{\boldsymbol{\tau}} - (\mathbf{N}_1 \mathbf{A}_{11} \mathbf{N}'_1 + \mathbf{N}_2 \mathbf{A}_{21} \mathbf{N}'_1 + \mathbf{N}_3 \mathbf{A}_{31} \mathbf{N}'_1 + \mathbf{N}_1 \mathbf{A}_{12} \mathbf{N}'_2 + \mathbf{N}_2 \mathbf{A}_{22} \mathbf{N}'_2 + \mathbf{N}_3 \mathbf{A}_{32} \mathbf{N}'_2 + \mathbf{N}_1 \mathbf{A}_{13} \mathbf{N}'_3 + \mathbf{N}_2 \mathbf{A}_{23} \mathbf{N}'_3 + \mathbf{N}_3 \mathbf{A}_{33} \mathbf{N}'_3)$$

where

$$\mathbf{A}_{11} = \frac{1}{pk} \mathbf{I}_{h \times h} + \frac{1}{p^2} [\mathbf{J}_{h \times p} \mathbf{A}_{22} \mathbf{J}_{p \times h} + \mathbf{J}_{h \times hp} \mathbf{A}_{32} \mathbf{J}_{p \times h} + \mathbf{J}_{h \times p} \mathbf{A}_{23} \mathbf{J}_{hp \times h} + \mathbf{J}_{h \times hp} \mathbf{A}_{33} \mathbf{J}_{hp \times h}],$$

$\mathbf{A}_{12} = -\frac{1}{p} [\mathbf{J}_{h \times p} \mathbf{A}_{22} + \mathbf{J}_{h \times hp} \mathbf{A}_{32}],$
 $\mathbf{A}_{13} = (\frac{1}{p} - \frac{1}{h}) \mathbf{J}_{h \times hp} \mathbf{A}_{33}, \mathbf{A}_{22} = \frac{1}{kh} [\mathbf{I}_{p \times p} + (pk^2 - p^3k^2) \mathbf{J}_{p \times p}],$
 $\mathbf{A}_{23} = -\frac{1}{h} \mathbf{J}_{p \times hp} \mathbf{A}_{33}, \mathbf{A}_{33} = (k \mathbf{I}_{hp \times hp} - \frac{pk}{h} \mathbf{J}_{hp \times hp})^-, \mathbf{R}_\tau = hn_r \mathbf{I}_{v \times v},$
 $\mathbf{N}_1$ denotes incidence matrix of treatments versus rows, $\mathbf{N}_2$ denotes incidence matrix of treatments versus columns, $\mathbf{N}_3$ denotes incidence matrix of treatments versus blocks /cells, $\mathbf{J}_{r \times c}$ denotes an $r \times c$ matrix of ones and $(.)^-$ denotes generalized inverse of a matrix.

As we know that connected design is necessary to estimate all treatment contrast *i.e.* Rank $(\mathbf{C}_d) = v - 1$. For a connected design with v treatments a matrix $\mathbf{M}_d$ can be defined as

$$\mathbf{M}_d = \mathbf{R}_\tau^{-1/2} \mathbf{C}_d \mathbf{R}_\tau^{-1/2}$$

Then the positive eigenvalues of the matrix $\mathbf{M}_d$ say $\mu_i, 1 \leq i \leq v - 1$ are known as the canonical efficiency factors of the design. Consider a equireplicate design with replication r , then, the canonical efficiency factors are $1/r$ times the positive eigenvalues of $\mathbf{C}_d$.

4. Methods of construction

In the present investigations, construction methods for generation of BSLR and PBSLR designs for block size two are developed. Four different construction methods are developed out of which two are for BSLR and another two are for PBSLR designs. The detailed construction methods are described in sequel.

Method 1: Construction of BSLR (v is odd)

The present approach for construction of BSLR design is based on the concept of generation of a design from an initial matrix. Once the initial matrix has been developed then the whole design can be generated from the initial matrix using specified procedure. Step 1 provides the method to get the initial matrix and Step 2 provides the procedure of generating the entire design from the initial matrix.

Step-1: Find a matrix of order $\frac{v-1}{2} \times 2$ such that elements of the first column of matrix is 1, where $v = 2n + 3$, n is a non-negative integer. For obtaining the second column of the matrix, either arrange the elements from 2 to $\frac{v+1}{2}$ in increasing order starting from up to down or v to $\frac{v+3}{2}$ in decreasing order from down to up. For example, for $n = 2$, $v = 7$ matrix will be $\begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{pmatrix}$ or $\begin{pmatrix} 1 & 5 \\ 1 & 6 \\ 1 & 7 \end{pmatrix}$.

Step-2: Take each row of the matrix one by one and add 1 to each of the element with mod v to obtain v blocks of size two. After doing cyclic rotation of these blocks, we get $v \times v$ row-column with intersection of size two from each row of the matrix. And after arranging these $v \times v$ row-column we get a $(v \times \frac{v(v-1)}{2})/2$ balanced Semi-Latin rectangles of block size two. General parameters for the obtained design are $v = 2n + 3, h = 2n + 3, p = (n + 1)(2n + 3), k = 2, n_r = 2n + 2, n_c = 2, \lambda = 2n + 3$

For example, for $v = 7, h = 7, p = 21, k = 2, n_r = 6, n_c = 2, \lambda = 7$, a $(7 \times 21)/2$ balanced semi-Latin rectangle can be obtained as shown below.

Row/Col	1	2	3	4	5	6	7	8	9	10
1	1, 2	2, 3	3, 4	4, 5	5, 6	6, 7	7, 1	1, 3	2, 4	3, 5
2	7, 1	1, 2	2, 3	3, 4	4, 5	5, 6	6, 7	7, 2	1, 3	2, 4
3	6, 7	7, 1	1, 2	2, 3	3, 4	4, 5	5, 6	6, 1	7, 2	1, 3
4	5, 6	6, 7	7, 1	1, 2	2, 3	3, 4	4, 5	5, 7	6, 1	7, 2
5	4, 5	5, 6	6, 7	7, 1	1, 2	2, 3	3, 4	4, 6	5, 7	6, 1
6	3, 4	4, 5	5, 6	6, 7	7, 1	1, 2	2, 3	3, 5	4, 6	5, 7
7	2, 3	3, 4	4, 5	5, 6	6, 7	7, 1	1, 2	2, 4	3, 5	4, 6

Row/Col	11	12	13	14	15	16	17	18	19	20	21
1	4, 6	5, 7	6, 1	7, 2	1, 4	2, 5	3, 6	4, 7	5, 1	6, 2	7, 3
2	3, 5	4, 6	5, 7	6, 1	7, 3	1, 4	2, 5	3, 6	4, 7	5, 1	6, 2
3	2, 4	3, 5	4, 6	5, 7	6, 2	7, 3	1, 4	2, 5	3, 6	4, 7	5, 1
4	1, 3	2, 4	3, 5	4, 6	5, 1	6, 2	7, 3	1, 4	2, 5	3, 6	4, 7
5	7, 2	1, 3	2, 4	3, 5	4, 7	5, 1	6, 2	7, 3	1, 4	2, 5	3, 6
6	6, 1	7, 2	1, 3	2, 4	3, 6	4, 7	5, 1	6, 2	7, 3	1, 4	2, 5
7	5, 7	6, 1	7, 2	1, 3	2, 5	3, 6	4, 7	5, 1	6, 2	7, 3	1, 4

Method 2: Construction of BSLR (v is even)

In this construction method, the process involves two steps as:

Step-1: Take $(v - 1)$ pairs of treatments such that the second treatment of each pair is v namely $(1, v), (2, v), (3, v), \dots, (v - 1, v)$, where, $v = 2n + 2, n$ is any positive integer. Now, develop a group of $\frac{v}{2}$ pairs where first pair will be from already generated $(v - 1)$ pairs and rest of the $(\frac{v}{2} - 1)$ pairs by adding 1 mod $(v - 1)$ in the first element of previous pair and second element of each pair by adding 1 mod $(v - 1)$ from the last pair to first pair in the group. For example, for $n = 2, v = 6$ pairs are $(1, 6), (2, 6), (3, 6), (4, 6)$, and $(5, 6)$ and groups of pairs from each pair are

Group 1: $(1, 6), (2, 5), (3, 4)$
 Group 2: $(2, 6), (3, 1), (4, 5)$
 Group 3: $(3, 6), (4, 2), (5, 1)$
 Group 4: $(4, 6), (5, 3), (1, 2)$
 Group 5: $(5, 6), (1, 4), (2, 3)$

Step-2: Find $\frac{v}{2} \times \frac{v}{2}$ row-column from each group by arranging in cyclic form and after arranging these $\frac{v}{2} \times \frac{v}{2}$ row-column one by one, we get a $\left(\frac{v}{2} \times \frac{v}{2}(v - 1)\right) / 2$ BSLR of block size two.

General parameters for the obtained design are $v = 2n + 2, h = n + 1, p = (n + 1)(2n + 1), k = 2, n_r = 2n + 1, n_c = 1, \lambda = n + 1$

For example, for $v = 8, h = 4, p = 28, k = 2, n_r = 7, n_c = 1, \lambda = 4$, by following the above steps, we get a $(4 \times 28) / 2$ BSLR as depicted below.

Row/Col	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1, 8	2, 7	3, 6	4, 5	2, 8	3, 1	4, 7	5, 6	3, 8	4, 2	5, 1	6, 7	4, 8	5, 3
2	4, 5	1, 8	2, 7	3, 6	5, 6	2, 8	3, 1	4, 7	6, 7	3, 8	4, 2	5, 1	7, 1	4, 8
3	3, 6	4, 5	1, 8	2, 7	4, 7	5, 6	2, 8	3, 1	5, 1	6, 7	3, 8	4, 2	6, 2	7, 1
4	2, 7	3, 6	4, 5	1, 8	3, 1	4, 7	5, 6	2, 8	4, 2	5, 1	6, 7	3, 8	5, 3	6, 2
Row/Col	15	16	17	18	19	20	21	22	23	24	25	26	27	28
1	6, 2	7, 1	5, 8	6, 4	7, 3	1, 2	6, 8	7, 5	1, 4	2, 3	7, 8	1, 6	2, 5	3, 4
2	5, 3	6, 2	1, 2	5, 8	6, 4	7, 3	2, 3	6, 8	7, 5	1, 4	3, 4	7, 8	1, 6	2, 5
3	4, 8	5, 3	7, 3	1, 2	5, 8	6, 4	1, 4	2, 3	6, 8	7, 5	2, 5	3, 4	7, 8	1, 6
4	7, 1	4, 8	6, 4	6, 4	1, 2	5, 8	7, 5	1, 4	2, 3	6, 8	1, 6	2, 5	3, 4	7, 8

Method 3: Construction of PBSLR (v is even)

In this methodology for constructing PBSLR designs, the approach revolves around generating the design from an initial matrix. After the initial matrix is established, the entire design can be derived from this matrix using a specific procedure. Step 1 outlines the method to obtain the initial matrix, and Step 2 details the procedure for generating the entire design from the initial matrix.

Step-1: First construct a matrix of order $\frac{v}{2} \times 2$ such that row sums of the matrix are constant which is $(v + 1)$, where $v = 2n + 2$, n is any positive integer. First column of the matrix is obtained by arranging the numbers from up to down as $1, 2, \dots, \frac{v}{2}$ and second column is obtained by arranging the numbers as $(\frac{v}{2} + 1), (\frac{v}{2} + 2), \dots, v$ starting from down to up. For

example, for $n = 3, v = 8$, an initial matrix of order 4×2 is $\begin{pmatrix} 1 & 8 \\ 2 & 7 \\ 3 & 6 \\ 4 & 5 \end{pmatrix}$.

Step-2: Considering the matrix of order $\frac{v}{2} \times 2$ which is obtained in step-1 as first column of size two and adding one to each element with mod v , we get total v columns of size two. Then the resulting design is $(\frac{v}{2} \times v)/2$ PBSLR of block size two.

General parameters for the obtained design are $v = 2n + 2, h = n + 1, p = 2n + 2, k = 2, n_r = 2, n_c = 1, \lambda_1 = 0, \lambda_2 = 2, n_1 = n, n_2 = n + 1$.

Association scheme for the designs constructed by this method is well known two-associate rectangular association scheme in which if we arrange treatments in two rows such that first row contain odd numbered treatments and second row contains even numbered treatments then the two treatments are (i) first associates if they lie in same row, (ii) second associates otherwise.

The parameters of the association scheme are:

$$v = 2n + 2, n_1 = n, n_2 = n + 1, P_1 = \begin{pmatrix} n-1 & 0 \\ 0 & n+1 \end{pmatrix} \text{ and } P_2 = \begin{pmatrix} 0 & n \\ n & 0 \end{pmatrix}.$$

For example, for $v = 8, h = 4, p = 8, k = 2, n_r = 2, n_c = 1, \lambda_1 = 0, \lambda_2 = 2, n_1 = 3, n_2 = 4$, a $(4 \times 8)/2$ PBSLR is obtained following above steps and is given below.

Row/Col	1	2	3	4	5	6	7	8
1	1, 8	2, 1	3, 2	4, 3	5, 4	6, 5	7, 6	8, 7
2	2, 7	3, 8	4, 1	5, 2	6, 3	7, 4	8, 5	1, 6
3	3, 6	4, 7	5, 8	6, 1	7, 2	8, 3	1, 4	2, 5
4	4, 5	5, 6	6, 7	7, 8	8, 1	1, 2	2, 3	3, 4

Now, for association scheme, arrange v treatments as

$$\begin{array}{c} 1, 3, 5, 7 \\ 2, 4, 6, 8 \end{array}$$

Then, association scheme for above designs is

Treatment	1st associates	2nd associates
1	3, 5, 7	2, 4, 6, 8
2	4, 6, 8	1, 3, 5, 7
3	1, 5, 7	2, 4, 6, 8
4	2, 6, 8	1, 3, 5, 7
5	1, 3, 7	2, 4, 6, 8
6	2, 4, 8	1, 3, 5, 7
7	1, 3, 5	2, 4, 6, 8
8	2, 4, 6	1, 3, 5, 7

with parameters of the association scheme as $v = 8, n_1 = 3, n_2 = 4, P_1 = \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}$ and $P_2 = \begin{pmatrix} 0 & 3 \\ 3 & 0 \end{pmatrix}$.

Method 4: Construction of PBSLR (v is odd)

In this construction method, the process involves two steps as given below.

Step-1: To get the initial row of p columns, first two columns are always $(v, 1)$ and $(v, 2)$ and last column is $((v+1)/2, (v-1)/2)$. Remaining $p-3$ columns are obtained as $(v-k, k), (v-k, k+2)$, where $k = 1, 2, \dots, (v-3)/2$ and $v = 2n+3, n$ is any non-negative integer. For example, if $v = 7$ then first two columns will be $(7, 1), (7, 2)$, last column will be $(4, 3)$ and remaining columns will be $(6, 1), (6, 3), (5, 2), (5, 4)$.

Step-2: By considering the initial row obtained in step-1, adding 1 mod v to each element of all pairs, one can get a $(v \times v)/2$ PBSLR of block size two.

General parameters for the design so obtained are $v = 2n+3, h = 2n+3, p = 2n+3, k = 2, n_r = 2, n_c = 2, \lambda_1 = 3, \lambda_2 = 2, n_1 = 2, n_2 = 2n$.

Association scheme for the designs constructed by this method is well known two-associate circular association scheme in which if we arrange the treatments (v) on the circle serially, then two treatments are first associates if they lie adjacently, second associates otherwise.

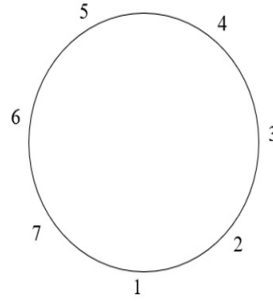
The parameters of the association scheme are:

$$v = 2n+3, n_1 = 2, n_2 = 2n, P_1 = \begin{pmatrix} 0 & 1 \\ 1 & 2n-1 \end{pmatrix} \text{ and } P_2 = \begin{pmatrix} 1 & 1 \\ 1 & 2n-2 \end{pmatrix}.$$

For example, for $v = 7, h = 7, p = 7, k = 2, n_r = 2, n_c = 2, \lambda_1 = 3, \lambda_2 = 2, n_1 = 2, n_2 = 4$, we obtained a $(7 \times 7)/2$ PBSLR as shown below.

Row/Col	1	2	3	4	5	6	7
1	7, 1	7, 2	6, 1	6, 3	5, 2	5, 4	4, 3
2	1, 2	1, 3	7, 2	7, 4	6, 3	6, 5	5, 4
3	2, 3	2, 4	1, 3	1, 5	7, 4	7, 6	6, 5
4	3, 4	3, 5	2, 4	2, 6	1, 5	1, 7	7, 6
5	4, 5	4, 6	3, 5	3, 7	2, 6	2, 1	1, 7
6	5, 6	5, 7	4, 6	4, 1	3, 7	3, 2	2, 1
7	6, 7	6, 1	5, 7	5, 2	4, 1	4, 3	3, 2

Now, for association scheme, arrange v treatments as



Then, association scheme for above design will be

Treatment	1st associates	2nd associates
1	2, 7	3, 4, 5, 6
2	1, 3	4, 5, 6, 7
3	2, 4	1, 5, 6, 7
4	3, 5	1, 2, 6, 7
5	4, 6	1, 2, 3, 7
6	5, 7	1, 2, 3, 4
7	1, 6	2, 3, 4, 5

with parameters of the association scheme as $v = 7, n_1 = 2, n_2 = 4, P_1 = \begin{pmatrix} 0 & 1 \\ 1 & 3 \end{pmatrix}$ and $P_2 = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$.

Remark 1: According to the definition of BSLR, each treatment appears an equal number of times in each row, an equal number of times in each column, and at most once in a block. Therefore, BSLR with block size two exists only for $v > 2$ because if $v = 2$, then each block, row, and column would contain the same content, which has no meaningful interpretation in the context of BSLR.

Remark 2: An R package *slr* has been developed to implement the proposed methods of constructions of balanced and partially balanced semi-Latin rectangles. The package is available on CRAN in the webpage <https://cran.r-project.org/web/packages/slr/index.html>. It comes with two main functions namely $bslr(v, k)$ and $pbslr(v, k)$ which can be used to construct a balanced SLR and a partially balanced SLR, respectively, for given number of treatments v and $k = 2$.

Remark 3: Using the proposed methods of constructions (Method 1, 2, 3 and 4), several designs were constructed and catalogued for $3 \leq v \leq 30$. Though designs can also be constructed for $v > 30$ using proposed methods, however, a general proof of the results could not be provided by us. Thus, it may be considered as a limitation of the proposed methods.

5. Efficiency of semi-Latin rectangles

In this Section, we provide the average efficiency factor of SLR designs generated using the developed methods of constructions. The average efficiency factor (John, 1992) of a design is determined by taking the harmonic mean of the canonical efficiency factors, which are obtained by multiplying the nonzero eigenvalues of its information matrix by $1/r$. Here, the lists of BSLR and PBSLR designs with their average efficiency factor ($\bar{E}$) for $v \leq 30$ are given in Table 2 and Table 3, respectively.

6. Concluding remarks

In this paper we proposed construction methods for obtaining balanced semi-Latin rectangles for any number of treatments with block size two. Another major contribution of this paper is introduction of the concept of partially balanced semi-Latin rectangles. Four methods of construction of semi-Latin rectangles are developed out of which two are for balanced and two are for partially balanced designs. The methods enable construction of balanced and partially balanced semi-Latin rectangles for all permissible number of treatments with block size two. While the proposed methods (Methods 1 and 2) are not superior to those suggested by Uto and Bailey (2020), the methods offer a simpler and easier alternative for constructing designs compared to the methods proposed by Uto and Bailey (2020). We also have provided the average efficiency factors of designs generated using the proposed methods of constructions for the number of treatments up to 30. Further, we have developed an R package called *slr* which implements the proposed methods of constructions.

Acknowledgements

The research work is a part of Ph.D. thesis of first author submitted to The Graduate School of IARI, New Delhi. The IASRI fellowship received by first author is gratefully acknowledged. The authors are thankful to the Chair Editor and anonymous reviewers for valuable suggestions which led to improvement in the presentation of the article.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

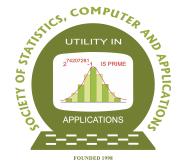
- Bailey, R. A. and Monod, H. (2001). Efficient semi-Latin rectangles: designs for plant disease experiments. *Scandinavian Journal of Statistics*, **28**, 257–270.
- John, J. A. (1992). Bounds for the efficiency factor of row-column designs. *Biometrika*, **79**, 658–661.
- Price, W. C. (1946). Measurement of virus activity in plants. *Biometrics Bulletin*, **2**, 81–86.
- Uto, N. P. and Bailey, R. A. (2020). Balanced semi-Latin rectangles: properties, existence and constructions for block size two. *Journal of Statistical Theory and Practice*, **14**, 1–11.
- Uto, N. P. and Bailey, R. A. (2022). Constructions for regular-graph Semi-Latin rectangles with block size two. *Journal of Statistical Planning and Inference*, **221**, 81–89.
- Yadav, K. K., Dash, S., Mandal, B. N., and Parsad, R. (2023). slr: Semi-Latin Rectangles. R package version 1.2.0, <https://CRAN.R-project.org/package=slr>.
- Youden, W. J. (1937). Dilution curve of tobacco-mosaic virus. *Contributions from Boyce Thompson Institute*, **9**, 49–58.

Table 2: Efficiency of balanced semi-Latin rectangles with $v \leq 30$

Method 1							
n	v	h	p	n_r	n_c	λ	$\bar{E}$
0	3	3	3	2	2	3	0.75
1	5	5	10	4	2	5	0.625
2	7	7	21	6	2	7	0.5833
3	9	9	36	8	2	9	0.5625
4	11	11	55	10	2	11	0.55
5	13	13	78	12	2	13	0.5417
6	15	15	105	14	2	15	0.5357
7	17	17	136	16	2	17	0.5313
8	19	19	171	18	2	19	0.5278
9	21	21	210	20	2	21	0.525
10	23	23	253	22	2	23	0.5227
11	25	25	300	24	2	25	0.5208
12	27	27	351	26	2	27	0.5192
13	29	29	406	28	2	29	0.5179
Method 2							
1	4	2	6	3	1	2	0.6667
2	6	3	15	5	1	3	0.6
3	8	4	28	7	1	4	0.5714
4	10	5	45	9	1	5	0.5556
5	12	6	66	11	1	6	0.5455
6	14	7	91	13	1	7	0.5385
7	16	8	120	15	1	8	0.5333
8	18	9	153	17	1	9	0.5294
9	20	10	190	19	1	10	0.5263
10	22	11	231	21	1	11	0.5238
11	24	12	276	23	1	12	0.5217
12	26	13	325	25	1	13	0.52
13	28	14	378	27	1	14	0.5185
14	30	15	435	29	1	15	0.5172

Table 3: Efficiency of partially balanced semi-Latin rectangles with $v \leq 30$

Method 3								
n	v	h	p	n_r	n_c	λ_1	λ_2	$\tilde{E}$
1	4	2	4	2	1	0	2	0.6
2	6	3	6	2	1	0	2	0.5556
3	8	4	8	2	1	0	2	0.5385
4	10	5	10	2	1	0	2	0.5294
5	12	6	12	2	1	0	2	0.5238
6	14	7	14	2	1	0	2	0.52
7	16	8	16	2	1	0	2	0.5172
8	18	9	18	2	1	0	2	0.5152
9	20	10	20	2	1	0	2	0.5135
10	22	11	22	2	1	0	2	0.5122
11	24	12	24	2	1	0	2	0.5111
12	26	13	26	2	1	0	2	0.5102
13	28	14	28	2	1	0	2	0.5094
14	30	15	30	2	1	0	2	0.5088
Method 4								
0	3	3	3	2	2	3	-	0.75
1	5	5	5	2	2	3	2	0.62
2	7	7	7	2	2	3	2	0.5799
3	9	9	9	2	2	3	2	0.5602
4	11	11	11	2	2	3	2	0.5483
5	13	13	13	2	2	3	2	0.5404
6	15	15	15	2	2	3	2	0.5348
7	17	17	17	2	2	3	2	0.5305
8	19	19	19	2	2	3	2	0.5272
9	21	21	21	2	2	3	2	0.5245
10	23	23	23	2	2	3	2	0.5223
11	25	25	25	2	2	3	2	0.5205
12	27	27	27	2	2	3	2	0.5189
13	29	29	29	2	2	3	2	0.5176



Three and Four Component Uniform Mixture Designs Based on Ellipsoidal Region - A Beginner's Training Manual

Poonam Singh and Himanshu Shukla

Department of Statistics, University of Delhi, Delhi-110007

Received: 20 January 2024; Revised: 30 August 2024; Accepted: 18 September 2024

Abstract

Talke and Borkowski (2012) discussed the generation of space-filling uniform designs in unit hypercube and gave measures of uniformity based on distance criteria. In this paper, a method is proposed for constructing uniform designs in ellipsoidal region and the designs are constructed for 2 and 3 dimensional ellipsoidal regions. The uniformity of the constructed designs is measured by distance based approaches. Further, these designs are used to obtain 3- and 4-component uniform mixture design using a transformation proposed for this purpose. The uniformity of the constructed mixture designs is measured by the DM_2 criterion.

Key words: Uniform design; Ellipsoidal Region; Mixture Experiments; Distance based criteria; DM_2 criterion.

AMS Subject Classifications: 62K20

1. Introduction

Mixture experiments involve blending of two or more ingredients to form end products. In these experiments, the response or characteristic of interest depends on the proportions of the components or ingredients present in the mixture and not on the total amount of the mixture. For example, the corrosion resistance or the strength of stainless steel depends on the proportions in which iron, copper, nickel, and chromium are mixed in the alloy. In a mixture experiment with s components, if x_i represents the proportion of the i^{th} ingredient in the mixture, then

$$0 \leq x_i \leq 1, i = 1, 2, \dots, s \quad (1)$$

with $\sum_{i=1}^s x_i = x_1 + x_2 + \dots + x_s = 1$

As a result, the factor space reduces to a $(s - 1)$ dimensional simplex $S_{(s-1)}$, where

$$S_{(s-1)} = \{x = (x_1, x_2, \dots, x_s) : \sum_{i=1}^s x_i = 1, x_i \geq 0\}. \quad (2)$$

The concept of mixture experiments was introduced by Quenouille (1953). Scheffé (1958, 1963) gave designs and models for mixture experiments. The simplex-lattice designs and simplex-centroid designs of Scheffé (1958, 1963) are boundary point designs except for the overall centroid or face centroids. For more details on mixture experiments see Cornell (2011).

Fang (1980) and Wang and Fang (1981) introduced uniform designs and discussed the use of quasi monte carlo method in their constructions. Uniform designs are space filling designs where design points are scattered uniformly over the experimental region. These designs can be easily obtained by using U-type designs defined on a unit hypercube. A U-type design can be obtained by using good lattice point (glp) method of Korobov (1959 a, b). Hua and Wang (1981) and Fang and Wang (1994) discussed the glp method and used it in the construction of uniform designs. Uniform design obtained using the glp method tends to have low discrepancy (Hua and Wang, 1981). Borkowski and Piepel (2009) introduced distance based criteria to measure the uniformity of a design. Talke and Borkowski (2016) gave a method to measure uniformity of a design in a spherical region. Ning *et al.* (2011a) gave DM_2 criterion to measure uniformity of a mixture design. For the comprehensive study on uniform design one can refer to Prescott (2008), Ning *et al.* (2011b), Fang *et al.* (2018), Adriana *et al.* (2022), Zhou and Xu (2014).

In many industrial and experimental situations, the experimentation at the vertices of the simplex region is not possible. Also, limited resources, time, and budget does not allow the exploration of entire simplex. In such cases, instead of exploring the entire simplex, the region is restricted to any subset of the simplex. In literature, spherical and ellipsoidal regions are considered. Thompson and Myers (1968) proposed a technique for obtaining mixture experimental design over some ellipsoidal region of interest within the simplex. Mandal *et al.* (2015) obtained optimum mixture designs used for parameter estimation in Scheffé (1958) models on the ellipsoidal region within the simplex. The problem of combinatorial drug experiment on lung cancer was conducted by Al-Shyoukh *et al.* (2011) for systematically qualitative characterization of cellular responses induced by multiple signals. Singh and Shukla (2023) proposed a method for obtaining the design in s -dimensional spherical region using the design in 2-dimensional spherical region given by Talke and Borkowski (2016).

In this paper, we propose a method for the construction of uniform designs in ellipsoidal region and introduce a transformation to obtain uniform mixture designs from uniform designs in ellipsoidal region. The NT-net design based on glp generators is used to obtain the U-type design for this purpose. The design is then used to obtain design in $(s - 1)$ dimensional ellipsoidal region from which uniform designs for s component mixture experiments are generated using the proposed transformation. The design for 3 and 4 component mixture experiments are obtained for illustrative purposes.

First, the designs on 2 and 3 dimensional ellipsoidal region are obtained, then from these designs uniform designs for 3 and 4 component mixture experiments are obtained. The uniformity of the designs in ellipsoidal region is measured using distance based approach and that of the mixture designs is obtained using the DM_2 criterion. DM_2 criterion has an explicit calculation formula therefore it is quite useful in measuring the uniformity of the design. Borkowski and Piepel (2009) discussed an approach using G function to obtain uniform experimental design for the s component mixtures from the design in $(s - 1)$ dimensional

unit hypercube, but due to nonlinear transformation of the G function the pattern of the uniformity is not preserved in the simplex. Here, in this paper, two transformations are used to obtain designs in the simplex and it is observed that the pattern of uniformity is preserved from the ellipsoidal region to simplex.

In Section 2, various measures of uniformity are discussed. In Section 3, a method for obtaining designs in ellipsoidal region is developed and the particular cases of 2 and 3 dimensions are discussed in detail. Section 4 includes a transformation that is used to construct uniform mixture designs in s dimensions from the uniform designs in $(s - 1)$ dimensional ellipsoidal region and the designs for 3 and 4 components are obtained. The conclusion of the paper is given in Section 5. Section 6 gives guidance to the beginners in this field.

2. Measures of uniformity

Fang (1980) and Wang and Fang (1981) were the first to apply the idea of Number-Theoretic methods (NTM) to experimental designs. Number-theoretic method or Quasi Monte Carlo method is a combination of number-theory and numerical analysis and has a variety of applications in statistics. Fang and Wang (1994) discussed the glp set and some measures of uniformity of a design. Borkowski and Piepel (2009) introduced three distance-based criteria, (i) root mean square distance (RMSD), (ii) average distance (AD) and (iii) maximum distance (MD).

Generally Monte Carlo sampling method is used to compute these values. In this method two sets a training set and a sampling set are taken and the design points of these sets are obtained using Number-Theoretic net (NT-net). From the given training set, a sampling set with smaller number of runs is obtained and the value of uniformity measure of the sampling set is evaluated based on the training set. However, in this article, we are generating different sampling sets based on different glp generator and the training set is separately generated by an appropriate glp generator. The criteria values of different sampling sets are evaluated using the training set.

Suppose $D = \{x_j = (x_{j1}, x_{j2}, \dots, x_{js}); j = 1, 2, \dots, n\}$ represents a sampling set *i.e.* a design with s components and n runs for which the distance based uniformity measures are to be computed and $T = \{t_k = (t_{k1}, t_{k2}, \dots, t_{ks}), k = 1, 2, \dots, N\}$ represent a training set consisting of $N(> n)$ runs and s components. Then D and T matrices of order $n \times s$ and $N \times s$ respectively, where

$$D = \begin{bmatrix} x_{11} & x_{12} & \cdots & \cdots & x_{1s} \\ x_{21} & x_{22} & \cdots & \cdots & x_{2s} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & \cdots & x_{ns} \end{bmatrix} \text{ and } T = \begin{bmatrix} t_{11} & t_{12} & \cdots & \cdots & t_{1s} \\ t_{21} & t_{22} & \cdots & \cdots & t_{2s} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{N1} & t_{N2} & \cdots & \cdots & t_{Ns} \end{bmatrix}.$$

Let $d_s^2(x_j, t_k)$ denote the Euclidean distance between the points x_j and t_k scaled between 0 and 1, then

$$d_s^2(x_j, t_k) = \sum_{i=1}^s (x_{ji} - t_{ki})^2$$

and the three distance-based criteria are defined as follows:

$$\begin{aligned} (1) \text{ } RMSD &= \sqrt{\frac{1}{N} \sum_{k=1}^N \min_{1 \leq j \leq n} d_s^2(x_j, t_k)} \\ (2) \text{ } AD &= \frac{1}{N} \sum_{k=1}^N \min_{1 \leq j \leq n} d_s(x_j, t_k) \\ (3) \text{ } MD &= \max_{1 \leq k \leq N} (\min_{1 \leq j \leq n} d_s(x_j, t_k)). \end{aligned}$$

The above mentioned distance based measures can also be applied for mixture experiments where $\sum_{i=1}^s x_{ji} = 1$ and $\sum_{i=1}^s t_{ki} = 1$ to measure the uniformity of the design. When the number of runs is small, the uniformity of a mixture design can be measured using the DM_2 -discrepancy which is generalization of star discrepancy (Ning *et al.*, 2011a). Star discrepancy given by Weyl (1916) is a basic and simplest measure of uniformity. Ning *et al.* (2011a) gave the following analytical expression to compute the DM_2 -discrepancy value of a design P_n with n runs defined over the s dimensional simplex.

$$\begin{aligned} DM_2(P_n) &= \\ &\left(\frac{\sqrt{s}}{(s-1)!} \right)^{1/2} \left\{ C_{n,s} - \frac{2(s-1)!}{n} \sum_{i=1}^n \sum_{(\tau_2, \dots, \tau_s) \in [0,1]^{s-1}} a_\tau \cdot (x_{i1})^{\tau_1} \cdot \prod_{j=2}^s x_{ij}^{\tau_j} + \right. \\ &\left. \frac{1}{n^2} \sum_{i=1, k=1}^n \left(\max \left(1 - \sum_{j=2}^s \max(x_{ij}, x_{kj}), 0 \right) \right)^{s-1} \right\}^{1/2} \end{aligned} \quad (3)$$

where

$$\begin{aligned} [0, 1]^{s-1} &= \{(t_1, t_2, \dots, t_{s-1}) : t_i = 0 \text{ or } 1\}, \\ C_{n,s} &= ((s-1)!)^3 2^{s-1} / (2(s-1)!) \prod_{k=0}^{s-2} (2s+k-1), \\ a_\tau &= (s-1)! / \left(2(s-1) - \sum \tau_i \right)!, \\ \text{and } \tau_1 &= 2(s-1) - \sum_{j=2}^s \tau_j. \end{aligned}$$

3. Uniform designs in ellipsoidal region

Consider an s dimensional ellipsoidal region defined as

$$\begin{aligned} E_n^s(\mathbf{y}_0) &= E_n^s(a_1, a_2, \dots, a_s) \\ &= \left\{ (y_1, y_2, \dots, y_s) : \frac{(y_1 - y_{01})^2}{a_1^2} + \frac{(y_2 - y_{02})^2}{a_2^2} + \dots + \frac{(y_s - y_{0s})^2}{a_s^2} \leq 1 \right\} \end{aligned} \quad (4)$$

where $y_0 = (y_{01}, y_{02}, \dots, y_{0s})$ is the center, n is the number of points in the region and $a_1^2 + a_2^2 + \dots + a_s^2 \leq 1$.

When $y_0 = (0, 0, \dots, 0)$ and $a_1^2 + a_2^2 + \dots + a_s^2 = 1$, it represents the standard ellipsoidal

region denoted by $E_n^s(1)$,

$$E_n^s(1) = \left\{ (y_1, y_2, \dots, y_s) : \frac{(y_1)^2}{a_1^2} + \dots + \frac{(y_s)^2}{a_s^2} \leq 1 \right\}. \quad (5)$$

To obtain uniform designs in ellipsoidal region, we first obtain U-type design for s dimensional unit hypercube using the good lattice point (glp) method and then convert this into a design in ellipsoidal region using some transformation.

Construction 1:

The steps for generating the uniform design are as follows:

Step 1: For a given number of runs n , obtain the initial generating vector $H_n = (h_1, h_2, \dots, h_k)$, with $1 \leq h_t < h_{t'} < n$ for $t < t'$, such that the greatest common divisor $\gcd(n, h_t) = 1$ for $t = 1, 2, \dots, k$.

Step 2: For a given number of factors $s(\leq k)$, select a generator $(h_{t_1}, h_{t_2}, \dots, h_{t_s})$; $h_{t_j} < h_{t_{j'}}$ for $t_j < t_{j'}$; $h_{t_1} = 1$ from H_n and generate an $n \times s$ NT-net design $Z = (z_{ij})$; $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, s$ where $z_{ij} = \frac{2iu_{ij}-1}{2n}$ with $u_{ij} = h_{ij} \pmod{n}$.

Step 3. Obtain the design $y = (y_1, y_2, \dots, y_s)$ in s dimensional ellipsoidal region $E_n^s(1)$ by using the transformation

$$\begin{aligned} y_{i1} &= a_{i1} \prod_{k=2}^s \cos(2\pi z_{ik}), \\ y_{ij} &= a_{ij} \sin(2\pi z_{ij}) \prod_{k=j}^s \cos(2\pi z_{ik}); j = 2, 3, \dots, s, \\ y_{is} &= a_{is} \sin(2\pi z_{is}), \end{aligned} \quad (6)$$

where $a_{i1}, a_{i2}, \dots, a_{is}$ are axis half lengths satisfying $\sum_{j=1}^s a_{ij}^2 = 1$.

Step 4. Compute the criteria values RMSD, AD, and MD of the designs obtained in step 3 for all the possible generators.

Step 5. Select the designs with minimum values of the RMSD, AD, and MD criteria value.

Note that there are many ways to obtain a_{ij} 's satisfying $\sum_{j=1}^s a_{ij}^2 = 1$. In this paper, Number theoretic-net (NT-net) approach is used. The NT-net design lies between 0 and 1. Some of the values are near 0 and some are near 1. The harmonic mean is one of the ways to choose a_{ij} 's as it gives less weightage to larger values and large weightage to smaller values to maintain balance and equal importance of each pair of a_{ij} 's in obtaining the a_{ij} 's. If there is availability of the normally distributed data then the axis of the ellipsoid can be obtained by using the concept of the confidence region given by Tzeng and Berns (2005).

The N points of the training set used in computation of criteria value in step 4 are generated using a generator chosen arbitrarily.

In step 5, if the generator giving minimum value of RMSD and AD is different from the generator giving minimum value of MD, then it may be considered as optimal generator whereas if the generator giving the smallest value of AD is also different from the generator giving minimum value of RMSD, then the generator for RMSD is considered as optimal generator as RMSD is least affected by variation of points near the boundary.

It may also be noted that for large n , the number of design generators are large therefore all non-equivalent design generators can be considered to generate designs and the design with minimum value of criteria is selected. For saving time, one can also use a good lattice point with a power generator in Step 4.

According to Talke and Borkowski (2016), equivalent glp generators are those generators whose design points can be found by column permutation of other glp generators. Let $H_n = (h_1, h_2, \dots, h_k)$ be the initial generating vector with $1 \leq h_t < h_{t'} < n$ for $t < t'$ such that the greatest common divisor $\gcd(n, h_t) = 1$ for $t = 1, 2, \dots, k$. In a two dimensional unit hypercube $[0, 1]^2$, the two glp generators $(1, h_i)$ and $(1, h_j)$ are said to be equivalent if

$$(h_i \times h_j)(\text{mod } n) = 1.$$

Similarly, in a three dimensional unit hypercube $[0, 1]^3$, a glp generator $(1, h_i, h_j)$ is said to be equivalent to $(1, h_k, h_l)$ and $(1, h_m, h_s)$ with $h_j > h_i$ and $h_k > h_i$, if it satisfies the five conditions given below:

- (i) $(h_i \times h_k) \text{mod}(n) = 1$
- (ii) $(h_j \times h_m) \text{mod}(n) = 1$
- (iii) $(h_i \times h_l \times h_m) \text{mod}(n) = 1$
- (iv) $(h_l \times h_s) \text{mod}(n) = 1$
- (v) $(h_j \times h_k \times h_s) \text{mod}(n) = 1.$

Remark: 1. Fang and Wang (1994) have defined a glp set as a set $P_n = \{x_k = (x_{k1}, x_{k2}, \dots, x_{ks})\}$ of lattice points with smallest discrepancy among all possible generating vectors $(n; h_1, h_2, \dots, h_s)$ for given n and $s(< n)$. They also showed that $h_1 = 1$ can always be assumed in a generator $(h_1, h_2, \dots, h_s)$. However, in general for a given n there can be more than 's' h_i 's say k in the initial generating vector $H_n = (h_1, h_2, \dots, h_k)$. This is described in Talke and Borkowski (2012). This leads to choosing a generating vector of size s out of the k elements in H_n which is possible in $m = {}^{k-1}C_{s-1}$ ways if $h_1 = 1$ is fixed. In the proposed method, we have first obtained the set $H_n = (h_1, h_2, \dots, h_k)$ with all possible h_i 's satisfying $1 \leq h_i < h_j < n$ for $i < j$ with $\gcd(n, h_i) = 1; i = 1, 2, \dots, k(< n)$ and then selected a generating vector of size s from it for a given s .

2. Fang and Wang (1994) have also discussed various methods to measure the uniformity of the design but distance-based approach of Talke and Borkowski (2016) is not discussed in this book. In the proposed method, we have used three distance-based criteria to measure the uniformity of the designs in ellipsoidal region. Obiri *et al.* (2020) has mentioned that with increase in the number of experimental factors or dimensions of the design space, it is difficult to compute discrepancy, therefore, the distance-based approach is more suitable. In the proposed method, instead of obtaining a glp set in unit hypercube, the discrepancy values

of the designs in ellipsoidal region are computed and the best design is selected. Further, as mentioned in Talke and Borkowski (2016), the design points obtained by the transformation given by Fang and Wang (1994, Section 1.5) does not preserve uniformity of the design in 2-dimensional ball B_2 despite the fact that set of points obtained from C^2 being uniformly scattered in C^2 . Thus, a different transformation is provided by Talke and Borkowski (2016) to obtain design points in B_2 . The transformation given in Fang and Wang (1994) depends on the value of (r_i, θ_i) to obtain designs. In the proposed method, the transformation used for obtaining the design in ellipsoidal region depends on the value of a_{ij} 's and z_{ij} 's for $i = 1, 2, \dots, n$; $j = 1, 2, \dots, s$ where $Z = (z_{ij})$ is the design in unit hypercube and a_{ij} 's are the axis half-length satisfying $\sum_{j=1}^s a_{ij}^2 = 1$. The value of r_i used in Fang and Wang (1994) depends only on the number of design points whereas the value of a_{ij} in the proposed transformation depends on the number of design points and the number of components. Here, a_{ij} 's are obtained by considering the harmonic mean which gives equal weightage to all the NT-net design points whereas in transformation of Fang and Wang (1994) chosen r_i 's gives higher weightage to higher value of r_i and lower weightage to smaller value of r_i .

The specific transformation for $s=2$ and $s=3$ are discussed in Section 3.1 and 3.2 respectively.

3.1. Designs in two dimensional ellipsoidal region

Consider a two dimensional standard ellipsoidal region given by

$$E_n^2(1) = \left\{ (y_1, y_2) : \frac{(y_1)^2}{a_1^2} + \frac{(y_2)^2}{a_2^2} \leq 1 \right\}$$

where $a_1^2 + a_2^2 = 1$.

Let $z_n = \{z_n(i) = (z_{i1}, z_{i2}); i = 1, 2, \dots, n\}$ be an n point design obtained after Step 1 of the Construction 1, then the uniform design in 2 dimensional ellipsoidal region $E_n^2(1)$ can be obtained using the transformation:

$$E_n^2(1) = \left\{ (y_{i1}, y_{i2}) = (a_{i1} \cos(2\pi z_{i2}), a_{i2} \sin(2\pi z_{i2})); a_{i1}^2 + a_{i2}^2 = 1; i = 1, 2, \dots, n \right\} \quad (7)$$

where $a_{i2} = \frac{2z_{i1}z_{i2}}{z_{i1}+z_{i2}}$ and $a_{i1} = \sqrt{1 - a_{i2}^2}; i = 1, 2, \dots, n$.

Example 1: Consider a design with $s=2$ and $n=21$. For $n=21$, the design can be obtained using the generators from the initial candidate generating vector H_{21} with $H_{21} = \{1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20\}$. Different generating vectors for constructing designs in unit hypercube $C^2 = [0, 1]^2$ are given in Table 1.

Table 1: List of glp generators for design in C^2

(1,2)	(1,4)	(1,5)	(1,8)	(1,10)	(1,11)	(1,13)	(1,16)	(1,17)	(1,19)	(1,20)

For each of these eleven glp generators listed in Table 1, a design is obtained in 2 dimensional ellipsoidal region. The design points of these designs are displayed in Figure 1.

From the Figure 1, it can be observed that the design points in designs obtained using glp generators (1,2), (1,4), (1,8), (1,19) are scattered more uniformly in comparison to the

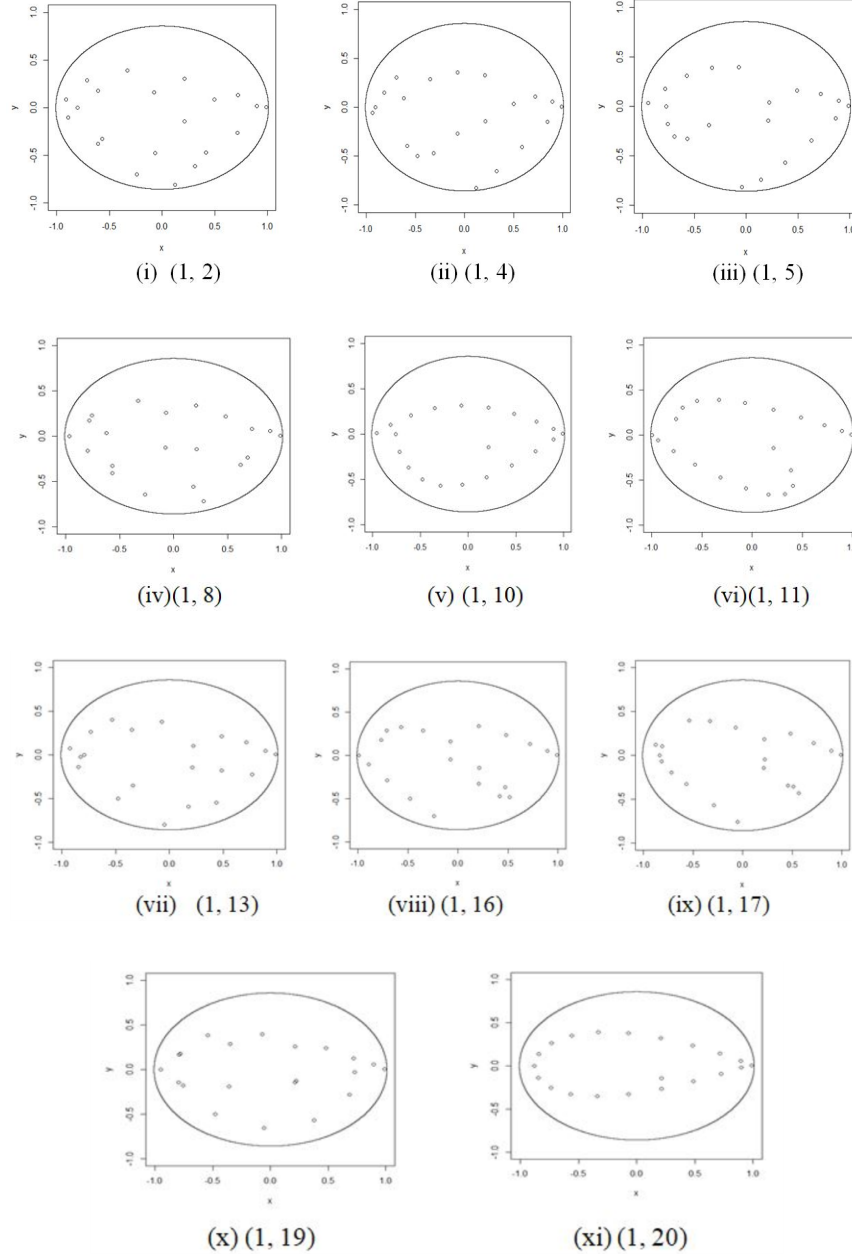


Figure 1: Plot of design points in designs obtained using generators (i) to (xi)

other designs. Scatteredness of these four designs are almost same so for choosing the most uniformly scattered design, the design with minimum value of criterion used can be selected.

Out of eleven generators only seven generators are non-equivalent. The generators (1,2), (1,4), (1,5), (1,8) are equivalent to (1,11), (1,16), (1,17), (1,19) respectively. The values of the three distance based criteria of uniformity for all the eleven designs are given in Table 2. For computing the criteria values a design, the points of the training set are obtained by using the generator (1,13) and the values are computed on the basis of 3000 evaluation points.

From the Table 2, it can be observed that the generator (1,2) gives the smallest

Table 2: Criteria values for all possible 21 run designs in 2 dimensional ellipsoidal region

Design Generator	RMSD	AD	MD
(1,2)	0.120193	0.103221	0.341422
(1,4)	0.127202	0.106910	0.328640
(1,5)	0.130708	0.105731	0.372629
(1,8)	0.122616	0.103479	0.317779
(1,10)	0.126546	0.104291	0.408531
(1,11)	0.129088	0.105089	0.394096
(1,13)	0.125127	0.105774	0.340943
(1,16)	0.143172	0.115458	0.425651
(1,17)	0.141105	0.115841	0.416019
(1,19)	0.123441	0.103987	0.350676
(1,20)	0.179616	0.132331	0.552201

value of ‘RMSD’ and ‘AD’ and the ‘MD’ value of the design obtained by generator (1,8) is minimum. Therefore, the generator (1,2) can be taken as optimal generator. It can also be observed from Figure 1 that the design points obtained by generator (1,2) are clustered less in comparison to that obtained by the generator (1,8).

3.2. Designs in three dimensional ellipsoidal region

Consider a three dimensional standard ellipsoidal region given by

$$E_n^3(1) = \left\{ (y_1, y_2, y_3) : \frac{(y_1)^2}{a_1^2} + \frac{(y_2)^2}{a_2^2} + \frac{(y_3)^2}{a_3^2} \leq 1 \right\}$$

where $a_1^2 + a_2^2 + a_3^2 = 1$.

Let $z_n = \{z_n(i) = (z_{i1}, z_{i2}, z_{i3}); i = 1, 2, \dots, n\}$ be the design in three dimensional unit hypercube obtained after step 1 of the construction 1, then the design in three-dimensional ellipsoidal region with n runs can be obtained using transformation

$$E_n^3(1) = \{(y_{i1}, y_{i2}, y_{i3}) - (a_{i1} \cos(2\pi z_{i2}) \cos(2\pi z_{i3}), a_{i2} \sin(2\pi z_{i2}) \cos(2\pi z_{i3}), a_{i3} \sin(2\pi z_{i3});$$

$$a_{i1}^2 + a_{i2}^2 + a_{i3}^2 = 1; i = 1, 2, \dots, n$$
(8)

with $a_{i2} = \frac{2z_{i1}z_{i3}}{z_{i1}+z_{i3}}$, $a_{i3} = \frac{2z_{i1}z_{i2}}{z_{i1}+z_{i2}}$ and $a_{i1} = \sqrt{1 - (a_{i2}^2 + a_{i3}^2)}$; $i = 1, 2, \dots, n$.

Example 2: Consider a design with $s=3$ and $n=34$. For $n=34$, the design can be obtained using the generators from the initial candidate generating vector H_{34} with $H_{34} = \{1, 3, 5, 7, 9, 11, 13, 15, 19, 21, 25, 27, 29, 31, 33\}$. For $n=34$, we have 35 different non-equivalent glp generators. The design obtained using all non-equivalent glp generators can be transformed into designs in 3 dimensional ellipsoidal region using the transformation (8). The criteria values of all these designs are computed and are given in Table 3. For computing the criteria values a design, the points of the training set are obtained by using the generator

(1,13,27) and the values are obtained on the basis of 8000 evaluation points.

Table 3: Criteria values for all non-equivalent designs in 3 dimensional ellipsoidal region

Design Generator	RMSD	AD	MD
(1,3,5)	0.19366	0.17805	0.48729
(1,3,7)	0.18659	0.16590	0.59109
(1,3,9)	0.19181	0.17475	0.58436
(1,3,11)	0.19181	0.17475	0.58436
(1,3,13)	0.20657	0.18613	0.55194
(1,3,15)	0.19796	0.17902	0.49603
(1,3,19)	0.20066	0.18114	0.50394
(1,3,21)	0.19043	0.17420	0.55360
(1,3,25)	0.21661	0.19547	0.52553
(1,3,27)	0.20181	0.18323	0.49835
(1,3,29)	0.20210	0.18420	0.50646
(1,3,31)	0.34922	0.28035	0.99897
(1,3,33)	0.23548	0.20737	0.70056
(1,5,7)	0.20323	0.18267	0.60436
(1,5,9)	0.19300	0.17361	0.51461
(1,5,11)	0.19723	0.17434	0.54075
(1,5,19)	0.21192	0.18817	0.32439
(1,5,21)	0.21578	0.19718	0.66106
(1,5,27)	0.20802	0.18742	0.59109
(1,5,29)	0.35220	0.28477	1.01637
(1,5,31)	0.19961	0.18250	0.51467
(1,5,33)	0.22281	0.19422	0.66455
(1,9,13)	0.20751	0.18886	0.57064
(1,9,15)	0.19687	0.18306	0.46457
(1,9,21)	0.20429	0.18215	0.57527
(1,9,25)	0.34638	0.28098	0.44235
(1,9,31)	0.20321	0.18469	0.64441
(1,9,33)	0.21291	0.18723	0.61685
(1,11,13)	0.20094	0.17888	0.40342
(1,11,25)	0.20902	0.18785	0.54557
(1,11,27)	0.20441	0.18483	0.51833
(1,11,29)	0.20375	0.18224	0.53322
(1,13,21)	0.34145	0.27485	0.99529
(1,13,25)	0.20427	0.18413	0.52374
(1,13,27)	0.22557	0.19462	0.64485

From Table 3, it can be observed that the design with generators (1,3,7) has least ‘RMSD’ and ‘AD’ value whereas ‘MD’ value of design with generator (1,5,19) is less. Therefore, we consider all the generators equivalent to (1,3,7) and (1,5,19) and out of all the equivalent design, the design with minimum criteria value should be chosen. The distance based criteria values for designs with generator equivalent to (1, 3, 7) and (1,5,19) are given in Table 4.

Table 4: Criteria value of designs equivalent to designs with generators (1,3,7) and (1,5,19) in E_{34}^3 (1)

Generator	Equivalent Generators	RMSD	AD	MD
(1,3,7)	(1,3,7)	0.18659	0.1659	0.59109
	(1,23,25)	0.19469	0.17739	0.56514
	(1,5,15)	0.21131	0.18756	0.54959
(1,5,19)	(1,5,19)	0.21192	0.18817	0.32439
	(1,7,31)	0.19808	0.17776	0.55356
	(1,9,11)	0.20558	0.18323	0.50081
	(1,9,31)	0.20321	0.18469	0.64441

From Table 4, it can be observed that the generator (1,3,7) gives the smallest value of ‘RMSD’ and ‘AD’, whereas the ‘MD’ value is minimum for the design obtained by generator (1,5,19). Therefore, the generator (1,3,7) can be taken as optimal generator. The scatter plots of the designs obtained by the generator (1,3,7) and (1,5,19) are displayed in Figure 2.

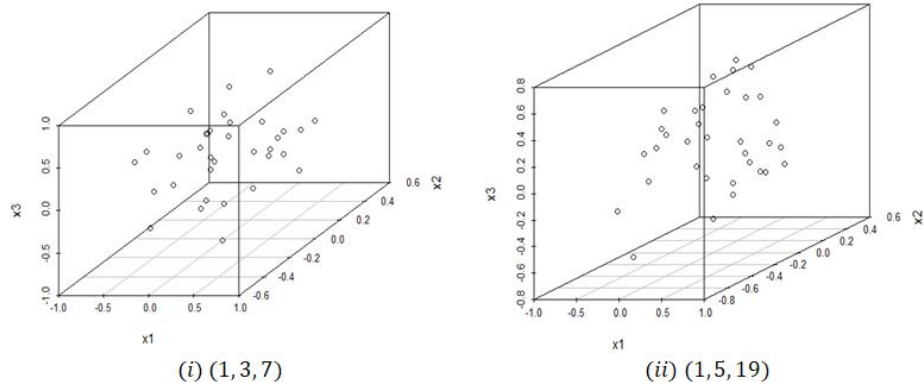


Figure 2: Scatter plot of the designs obtained by generator (i) (1,3,7) and (ii) (1,5,19)

From Figure 2, it can be observed that the design based on generator (1, 3, 7) is more uniform.

4. Uniform mixture designs for simplex

The designs in 2 and 3 dimensional ellipsoidal region can be obtain using the transformation (7) and (8) by using the methods discussed in Section 3. These designs are used to obtain the designs for mixture experiments as described in construction 2.

Construction 2:

Let $y = (y_1, y_2, \dots, y_{s-1})$ represents the design in (s-1) dimensional ellipsoidal region, then a mixture design can be obtained from this using the following transformation:

For $i = 1, 2, \dots, s - 1$, define $\xi_i = \frac{e^{y_i}}{\sum_{i=1}^{s-1} e^{y_i}}$ and make the following transformation:

$$\begin{aligned}
x_1 &= 1 - \xi_1, \\
x_i &= \prod_{j=1}^{i-1} \xi_j (1 - \xi_i); i = 2, 3, \dots, (s-1), \\
x_s &= \prod_{i=1}^{s-1} \xi_i
\end{aligned} \tag{9}$$

then $x = (x_1, x_2, \dots, x_s)$ is the required s component mixture design.

4.1. Designs for three component mixtures

Consider the 2 dimensional ellipsoidal region and the two designs obtained by using the glp generators (1,2) and (1,8). Then,, the two 3 component mixture designs corresponding to these designs are obtained by using the transformation (9). The design in 2 dimensional ellipsoidal region obtained using generator (1,2) along with corresponding mixture design is given in Table 5 and that obtained using the generator (1,8) are given in Table 6.

Table 5: Design in two dimensional ellipsoidal region and three component mixture design using generator (1,2)

Design Point	Design in Ellipsoidal Region		Mixture Design		
	y_1	y_2	x_1	x_2	x_3
1	0.90039	0.01550	0.29216	0.50103	0.20680
2	0.49749	0.08660	0.39870	0.36156	0.23974
3	-0.07372	0.16323	0.55896	0.19451	0.24652
4	-0.60717	0.17769	0.68673	0.09814	0.21513
5	-0.91427	0.08572	0.73106	0.07233	0.19661
6	-0.89357	-0.10444	0.68764	0.09757	0.21479
7	-0.56645	-0.32669	0.55965	0.19391	0.24644
8	-0.06550	-0.48002	0.39783	0.36261	0.23956
9	0.41926	-0.47187	0.29088	0.50286	0.20627
10	0.71505	-0.26396	0.27309	0.52840	0.19851
11	0.98781	0.00677	0.27269	0.52898	0.19833
12	0.71890	0.13303	0.35758	0.41270	0.22972
13	0.21118	0.30723	0.52399	0.22658	0.24942
14	-0.33192	0.38897	0.67280	0.10706	0.22014
15	-0.71054	0.28749	0.73067	0.07254	0.19679
16	-0.80287	0.00000	0.69059	0.09574	0.21368
17	-0.60784	-0.38156	0.55633	0.19684	0.24683
18	-0.23948	-0.70300	0.38615	0.37681	0.23704
19	0.12391	-0.80979	0.28218	0.51527	0.20255
20	0.31319	-0.61497	0.28330	0.51366	0.20304
21	0.21449	-0.14549	0.41096	0.34696	0.24207

For the two, 3 component mixture designs given in Table 5 and Table 6, the DM_2 values are calculated which are given in Table 7.

Table 6: Design in two dimensional ellipsoidal region and three component mixture design using generator (1,8)

Design Point	Design in Ellipsoidal Region		Mixture Design		
	y_1	y_2	x_1	x_2	x_3
1	-0.62287	0.03490	0.65876	0.11645	0.22480
2	-0.07409	-0.12989	0.48606	0.26414	0.24981
3	0.72784	0.08097	0.34370	0.43073	0.22557
4	-0.96825	0.00000	0.72477	0.07575	0.19948
5	0.68812	-0.23447	0.28443	0.51204	0.20353
6	-0.07212	0.26117	0.58256	0.17426	0.24318
7	-0.56645	-0.32669	0.55965	0.19391	0.24644
8	0.98784	0.00665	0.27266	0.52903	0.19831
9	-0.75553	0.22801	0.72781	0.07409	0.19810
10	0.18219	-0.55978	0.32257	0.45891	0.21852
11	0.48412	0.21651	0.43349	0.32093	0.24558
12	-0.79955	-0.16141	0.65433	0.11949	0.22618
13	0.62012	-0.31476	0.28194	0.51562	0.20245
14	-0.33192	0.38897	0.67280	0.10706	0.22014
15	-0.26427	-0.64275	0.40649	0.35225	0.24126
16	0.89329	0.05651	0.30221	0.48691	0.21088
17	-0.78236	0.16924	0.72144	0.07760	0.20097
18	0.27639	-0.72169	0.26932	0.53389	0.19679
19	0.20888	0.33608	0.53176	0.21925	0.24899
20	-0.56868	-0.40866	0.53992	0.21167	0.24841
21	0.21449	-0.14549	0.41096	0.34696	0.24207

Table 7: DM_2 value of the 3 component mixture designs

Design Generator	DM_2 value
(1,2)	0.20811
(1,8)	0.21414

From the Table 7, it can be observed that the criteria value of design based on generator (1, 2) has less DM_2 value and therefore it provides design with better uniformity.

4.2. Designs for four component mixtures

Consider the 3 dimensional ellipsoidal region and the designs obtained using the glp generators (1,3,7) and (1,5,19). Then, the two four component mixture designs corresponding to these two designs are obtained by using the transformation (9). The designs in 3 dimensional ellipsoidal region obtained using the generator (1,3,7) along with corresponding mixture design is given in Table 8 and that obtained using the generator (1,5,19) are given in Table 9.

For the two, 4 component mixture designs given in Table 8 and Table 9, the DM_2 values are calculated and are given in Table 10.

Table 8: Design in three dimensional ellipsoidal region and four component mixture design using generator (1,3,7)

Design Point	Design in Ellipsoidal Region			Mixture Design			
	y_1	y_2	y_3	x_1	x_2	x_3	x_4
1	0.32315	0.00440	0.02285	0.59476	0.28585	0.08356	0.03583
2	-0.41776	-0.05388	0.04178	0.75138	0.15968	0.05393	0.03501
3	0.00000	-0.10460	-0.06848	0.64720	0.24069	0.07517	0.03693
4	-0.18455	0.05609	-0.14718	0.69790	0.18601	0.07969	0.03641
5	-0.87265	0.01175	0.01864	0.82935	0.10013	0.04118	0.02934
6	-0.17402	-0.00316	0.24197	0.72989	0.18356	0.05111	0.03544
7	0.65706	0.14242	0.12940	0.54290	0.33221	0.09121	0.03368
8	0.21510	0.20547	-0.24715	0.61836	0.23739	0.10958	0.03468
9	0.08141	-0.19930	-0.32187	0.58735	0.28405	0.09315	0.03546
10	0.58516	-0.05417	0.11568	0.53553	0.35064	0.08077	0.03306
11	0.00000	0.00000	0.46683	0.72183	0.20079	0.04305	0.03433
12	-0.84944	-0.10222	0.02136	0.81818	0.11203	0.03948	0.03031
13	-0.30481	-0.18389	-0.16548	0.69494	0.20004	0.06820	0.03683
14	0.09769	0.36078	-0.20959	0.67066	0.18823	0.10693	0.03418
15	-0.29917	0.10470	0.15968	0.75489	0.15513	0.05508	0.03489
16	0.12267	-0.03837	0.41722	0.68689	0.22965	0.04837	0.03508
17	0.72112	-0.04459	0.04478	0.49328	0.38732	0.08863	0.03076
18	0.19336	0.09501	-0.50589	0.58390	0.25918	0.12447	0.03245
19	-0.18046	-0.45959	-0.35989	0.61422	0.27320	0.07628	0.03630
20	0.00000	-0.13930	0.39171	0.70145	0.22101	0.04329	0.03425
21	-0.11109	0.12545	0.65402	0.77355	0.16148	0.03335	0.03162
22	-0.29956	0.25188	-0.06935	0.74966	0.14156	0.07450	0.03428
23	-0.13234	-0.01170	-0.02828	0.69116	0.20123	0.07073	0.03688
24	0.41720	0.42709	-0.07987	0.61807	0.23461	0.11310	0.03423
25	0.22552	0.14049	0.22332	0.65710	0.23490	0.07105	0.03695
26	0.07597	-0.24125	0.34616	0.67088	0.25025	0.04486	0.03401
27	0.37766	-0.45551	-0.13673	0.50800	0.38678	0.07427	0.03095
28	0.00000	0.00000	-0.58311	0.60909	0.23810	0.11947	0.03334
29	-0.44680	-0.23511	-0.18059	0.71758	0.18385	0.06224	0.03633
30	-0.22162	-0.10609	0.62197	0.77514	0.16811	0.02709	0.02967
31	0.03699	0.34538	0.59061	0.75614	0.16291	0.04661	0.03434
32	-0.16099	0.59138	-0.38497	0.74499	0.11702	0.10987	0.02813
33	0.11301	-0.09508	-0.90978	0.53954	0.28827	0.14368	0.02851
34	0.96210	-0.09052	-0.09091	0.41104	0.46790	0.09619	0.02488

From the Table 10, it can observed that the criteria value for design using generator (1,3,7) has less DM_2 value and hence it provides more uniformly scattered design points.

5. Conclusion

In this paper, a method is proposed to construct uniform designs in ellipsoidal region and the designs in 2 and 3 dimensional ellipsoidal regions are obtained. It is observed

Table 9: Design in three dimensional ellipsoidal region and four component mixture design using generator (1,5,19)

Design Point	Design in Ellipsoidal Region			Mixture Design			
	y_1	y_2	y_3	x_1	x_2	x_3	x_4
1	-0.64748	-0.02036	-0.00724	0.79032	0.12737	0.04957	0.03274
2	-0.14593	0.04845	0.04592	0.70812	0.18841	0.06688	0.03659
3	0.46334	-0.03106	-0.10664	0.54033	0.33079	0.09538	0.03350
4	-0.16031	-0.01150	0.17158	0.71863	0.18950	0.05585	0.03602
5	-0.03201	-0.04087	-0.21982	0.64538	0.22997	0.08801	0.03663
6	-0.33227	0.08514	0.23184	0.76613	0.15084	0.04889	0.03413
7	0.75380	0.02321	-0.01646	0.48573	0.38689	0.09705	0.03032
8	-0.47411	-0.24313	0.05108	0.74687	0.17240	0.04617	0.03455
9	-0.34555	0.02579	0.02550	0.74352	0.16112	0.05992	0.03545
10	0.74003	-0.09205	-0.15444	0.45769	0.41433	0.09960	0.02837
11	-0.48051	-0.07523	0.30184	0.78661	0.14510	0.03643	0.03186
12	0.00000	0.16408	-0.43473	0.64611	0.20632	0.11376	0.03381
13	0.00000	0.00000	0.52155	0.72860	0.19774	0.03998	0.03368
14	0.29277	0.05266	-0.07405	0.59669	0.27537	0.09218	0.03575
15	-0.21416	-0.24806	0.19510	0.71201	0.20782	0.04541	0.03476
16	-0.32505	0.46508	-0.17310	0.77105	0.11344	0.08473	0.03079
17	0.72112	-0.04459	0.04478	0.49328	0.38732	0.08863	0.03076
18	-0.53092	-0.05777	0.15530	0.78219	0.14166	0.04320	0.03294
19	-0.07507	0.44871	-0.38620	0.70769	0.14804	0.11337	0.03089
20	0.31058	-0.05921	0.60236	0.66993	0.25480	0.04201	0.03326
21	-0.12218	-0.05377	-0.12884	0.67365	0.21231	0.07707	0.03697
22	-0.02910	-0.07000	0.32151	0.70413	0.21184	0.04873	0.03530
23	-0.16991	0.28774	-0.40189	0.70356	0.15756	0.10623	0.03265
24	0.50097	0.03714	0.35557	0.59897	0.29989	0.06607	0.03507
25	-0.19690	-0.67197	-0.18881	0.61977	0.29033	0.05544	0.03446
26	-0.05284	0.56681	-0.07181	0.73955	0.13439	0.09384	0.03221
27	0.42584	-0.03292	0.38274	0.61388	0.29188	0.05938	0.03485
28	-0.36576	-0.28817	-0.13497	0.70062	0.20252	0.06033	0.03652
29	0.00000	0.11247	0.35913	0.71840	0.19286	0.05295	0.03579
30	0.00000	0.00000	-0.54480	0.61240	0.23737	0.11646	0.03377
31	0.19961	0.04542	0.63163	0.70566	0.22008	0.04059	0.03367
32	-0.15955	-0.56293	-0.58508	0.56924	0.30680	0.08907	0.03489
33	-0.17986	0.44889	0.39813	0.78530	0.12826	0.05336	0.03308
34	0.96210	-0.09052	-0.09091	0.41104	0.46790	0.09619	0.02488

Table 10: DM_2 value of the 4 component mixture designs

Design Generator	DM_2 value
(1,3,7)	0.2942798
(1,5,19)	0.2977485

that the designs chosen by ‘RMSD’ and ‘AD’ criteria have better uniformity as compared to designs selected by ‘MD’ criterion. This may be because ‘MD’ is greatly influenced by points near the boundary and it fails to provide the representative points for the overall space. A transformation to construct mixture designs from the design in ellipsoidal region is also given in this paper. Designs for 3 and 4 component mixtures are constructed and the DM_2 criterion is used to measure the uniformity of these designs. The mixture design obtained by using the design selected in ellipsoidal region by ‘RMSD’ and ‘AD’ criteria has less discrepancy value as compared to that obtained by using the design selected by ‘MD’ criterion. The design which was found best in ellipsoidal region also possesses greater uniformity in simplex region. Hence, the transformation proposed in this paper preserves the uniformity when a design is transformed from the ellipsoidal region to simplex region.

6. Guidance to the beginners

The researchers who are new in this area can start with the understanding of the basic terminology and concepts provided in Section 1 and the references cited therein. They can begin with clearly identifying the number of components of the mixture and the region of interest. Here, an ellipsoidal region is considered. A suitable method needs to be developed for obtaining the design in the region of interest. Here, in Section 3, a method for obtaining the design in s -dimensional ellipsoidal region is given in steps 1-3 of Construction 1. The next step is to select a suitable measure of uniformity to obtain the criteria values for the generated designs and select the designs with lowest discrepancy. Section 2 can be referred for the distance-based approach and DM_2 -discrepancy. The constructed designs for $(s-1)$ -dimensional ellipsoidal region can then be utilized to generate designs for s component mixtures by using the construction 2 given in Section 4 and the best design is selected on the basis of DM_2 -discrepancy described in Section 2. The method discussed in the paper is provided in the form of following steps:

1. Obtain initial generating vector $H_n = (h_1, h_2, \dots, h_k)$ for a given n .
2. For the given number of factor s , obtain all possible generators of size s from H_n with $h_1 = 1$. This leads to $m = {}^{k-1}C_{s-1}$ distinct generators.
3. For each of the m generators, obtain the design $Z = (z_{ij})$, $i = 1, 2, \dots, n$; $j = 1, 2, \dots, s$ in unit hypercube.
4. For each of the m generators, obtain the design Y in the ellipsoidal region from Z .
5. Compute the RMSD, AD, and MD value of all the Y designs and select the designs with lowest criteria values and note their corresponding generators.
6. For each of m generators, obtain the mixture design X for s component mixture from all design Y in $(s-1)$ dimensional ellipsoidal region.
7. Compute the DM_2 discrepancy value for all the mixture designs and select the design with lowest value of DM_2 -discrepancy.

Acknowledgements

The authors are thankful to the editors and reviewers for their valuable comments and suggestions that helped in improvement of the papers. This research work supported by Faculty Research Programme (FRP) grant vide Ref. No./IoE/2023-24/12/FRP from Institute of Eminence (IoE), University of Delhi, Delhi.

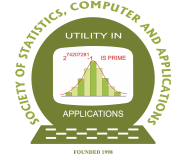
Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Adriana, M. M., Saúl, Z. M., Luis, M. G., and Carlos, A. C. C. (2022). Uniform mixture design via evolutionary multi-objective optimization, *Swarm and Evolutionary Computation*, **68**, 100979.
- Al-Shyoukh, I., Yu, F., Feng, J., Yan, K., Dubinett, S., Ho, C. M., and Sun, R. (2011). Systematic quantitative characterization of cellular responses induced by multiple signals. *BMC Systems Biology*, **5**, 1–17.
- Borkowski, J. J. and Piepel, G. F. (2009). Uniform designs for highly constrained mixture experiments. *Journal of Quality Technology*, **41**, 35–47.
- Cornell, J. A. (2011). *Experiments with Mixtures: Designs, Models, and the Analysis of Mixture Data*. John Wiley and Sons.
- Fang, K. T. (1980). The uniform design: Application of number-theoretic methods in experimental design. *Acta Mathematicae Applicatae Sinica*, **3**, 363–372.
- Fang, K. T. and Wang, Y. (1994). *Number-theoretic Methods in Statistics*. CRC Press.
- Fang, K.T., Liu, M.Q., Qin, H., and Zhou, Y.D. (2018). *Theory and Application of Uniform Experimental Designs*, Science Press and Springer, Beijing.
- Hua, L. K. and Wang, Y. (1981). *Applications of Number Theory to Numerical Analysis*. Springer and Science Press, Berlin and Beijing.
- Korobov, N. M. (1959a). The approximate computation of multiple integrals. *Doklady Akademii Nauk SSSR*, **124**, 1207–1210.
- Korobov, N. M. (1959b). Computation of multiple integrals by the method of optimal coefficients. *Vestnik Moskovskogo Universiteta Sectia de Matematica, Astronomie, Fizica, Chimie*, **4**, 19–25.
- Mandal, N. K., Pal, M., Sinha, B. K., and Das, P. (2015). Optimum mixture designs in a restricted region. *Statistical Papers*, **56**, 105–119.
- Ning, J. H., Zhou, Y. D., and Fang, K. T. (2011a). Discrepancy for uniform design of experiments with mixtures. *Journal of Statistical Planning and Inference*, **141**, 1487–1496.
- Ning, J.H., Fang, K.T., and Zhou, Y.D. (2011b). Uniform design for experiments with mixtures, *Communications in Statistics-Theory and Methods*, **40**, 1734–1742.
- Obiri, M. Y. (2020). *Space-Filling Designs for Mixture/Process Variable Experiments* (Doctoral dissertation, Montana State University).
- Prescott, P. (2008). Nearly uniform designs for mixture experiments. *Communications in Statistics—Theory and Methods*, **37**, 2095–2115.
- Quenouille, M. H. (1953). *The Design and Analysis of Experiments*. Charles Griffin and Company, London, England.
- Scheffé, H. (1958). Experiments with mixtures. *Journal of the Royal Statistical Society: Series B (Methodological)*, **20**, 344–360.
- Scheffé, H. (1963). The simplex-centroid design for experiments with mixtures. *Journal of the Royal Statistical Society: Series B (Methodological)*, **25**, 235–251.

- Singh, P. and Shukla, H. (2023). Uniform mixture designs using designs in 2-dimensional spherical region. *International Journal of System Assurance Engineering and Management*, **14**, 1888–1897.
- Thompson, W. O. and Myers, R. H. (1968). Response surface designs for experiments with mixtures. *Technometrics*, **10**, 739–756.
- Talke, I. S. and Borkowski, J. J. (2012). Generation of space-filling uniform designs in unit hypercubes. *Journal of Statistical Planning and Inference*, **142**, 3189–3200.
- Talke, I. S. and Borkowski, J. J. (2016). Measures of uniformity for space-filling uniform designs in a spherical region. *International Journal of Experimental Design and Process Optimisation*, **5**, 23.
- Tzeng, D. Y. and Berns, R. S. (2005). A review of principal component analysis and its applications to color technology. *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, **30**, 84–98.
- Wang, Y. and Fang, K. T. (1981). A note on uniform distribution and experimental design, *KeXue TongBao*, **26**, 485–489.
- Weyl, H. (1916). über die gleichverteilung der zahlen mod eins. *Annals of Mathematics*, **77**, 313–352.
- Zhou, Y. D. and Xu, H. (2014). Space-filling fractional factorial designs. *Journal of the American Statistical Association*, **109**, 1134–1144.



On Renewal Processes - Some Bounds and Characterisations

S. Ravi¹ and Suman Kalyan Ghosh²

¹*Department of Studies in Statistics*

University of Mysore, Manasagangotri, Mysuru 570006, India

²*Department of Applied Mathematics, Alliance University, Bengaluru 562106, India*

Received: 04 April 2024; Revised: 22 August 2024; Accepted: 23 September 2024

Abstract

Bounds for the distribution/ survival function and a few other characterisations associated with renewal processes based on ageing properties of the generating inter-arrival times are obtained. A few results are also obtained by comparing renewal processes with respect to some stochastic orderings.

Key words: Ageing classes; Convex transform order; Log-concavity; Renewal process.

1. Introduction

Let $\{N(t), t \geq 0\}$ be a renewal process with arrival and inter-arrival times denoted by the sequences $\{S_1, S_2, \dots\}$ and $\{X_1, X_2, \dots\}$, respectively, where $S_n = X_1 + \dots + X_n$, and, the inter-arrival times are continuous, non-negative, independent and identically distributed (iid) random variables (rvs) with common distribution function (df) $F(\cdot)$ and survival function (sf) $\bar{F}(\cdot) = 1 - F(\cdot)$. We assume that $F(0) = 0$ and the right extremity $r(F) = \sup\{x : F(x) < 1\} = \infty$. The distribution of $N(t)$ is given by

$$P(N(t) \geq n) = P(S_n \leq t) = F_n(t), n \geq 1, t \geq 0, \quad (1)$$

where F_n is the n -fold convolution of F with itself (see Ross (1995)). Unless otherwise specified, the preceding notations are used throughout this article. The n -fold convolution of a df is of great interest in applied probability theory, especially in reliability and renewal theory. However, to obtain analytical expression for the n -fold convolution is mathematically challenging and sometimes impossible. F_n can be obtained in analytical form for some dfs such as uniform, exponential and gamma with integer shape parameter. Some bounds are proposed for F_n , which, in turn, provide some bounds for the df and sf of $N(t)$, using ageing properties of the renewal df F being concave / log-concave, convex / log-convex (Bergstrom and Bagnoli (2005)), IFR / DFR, NBU / NWU (Barlow and Proschan (1975)). A few results are also obtained here by comparing renewal processes having different renewal distributions

with respect to convex (concave) transform order and super-additive / sub-additive order (for more details on stochastic orders, see Shaked and Shanthikumar (2007)). The bounds obtained here appear to be new and more strict compared to the previously obtained bounds by Barlow and Proschan (1975). This is illustrated with some numerical examples in Section 2.1.

Definitions used in this article are from Barlow and Proschan (1975) and Shaked and Shanthikumar (2007) and these have been given in Appendix for reference.

The article is organized as follows: Section 2 comprises of a few new bounds with detailed proofs and associated comments, remarks, corollary, and some numerical illustrations. Section 3 proposes a few results by comparing renewal processes with respect to some stochastic orderings. Finally Section 4 comprehensively concludes the paper with some future directions of research.

2. On bounds for renewal processes

Let $R(t) = -\ln \bar{F}(t)$ and $K(t) = -\ln F(t)$, $t > 0$, denote the hazard function (hf) and reversed hazard function (rhf) associated with the df F , respectively. We first state and prove a known lemma which will be used to prove many of the results of this section.

Lemma 1: For $\lambda_n \geq 0$, $n \geq 1$, with $\lim_{n \rightarrow \infty} \frac{\lambda_n}{n} = 0$, $\lim_{n \rightarrow \infty} \sum_{j=0}^n e^{-\lambda_n} \frac{\lambda_n^j}{j!} = 1$.

Proof: For arbitrary $k > 0$,

$$\lim_{n \rightarrow \infty} \frac{\lambda_n}{n} = 0 \Rightarrow \lim_{n \rightarrow \infty} \frac{\lambda_n + k\sqrt{\lambda_n}}{n} = 0 \Rightarrow \lambda_n + k\sqrt{\lambda_n} < n \text{ for all } n \text{ large.} \quad (2)$$

We have, for arbitrary $k > 0$,

$$\begin{aligned} 1 &\geq \lim_{n \rightarrow \infty} \sum_{j=0}^n e^{-\lambda_n} \frac{\lambda_n^j}{j!} = \lim_{n \rightarrow \infty} \sum_{j=0}^n P(X_n = j), \text{ where } X_n \sim \text{Poisson}(\lambda_n) \\ &= \lim_{n \rightarrow \infty} P(X_n \leq n) \\ &\geq \lim_{n \rightarrow \infty} P\left(X_n \leq \lambda_n + k\sqrt{\lambda_n}\right) \text{ by (2)} \\ &\geq \lim_{n \rightarrow \infty} P\left(|X_n - \lambda_n| \leq k\sqrt{\lambda_n}\right) \\ &\geq \lim_{n \rightarrow \infty} \left(1 - \frac{1}{k^2}\right) \text{ by Chebyshev's inequality} \\ &= 1 - \frac{1}{k^2}. \end{aligned}$$

Therefore $\lim_{n \rightarrow \infty} \sum_{j=0}^n e^{-\lambda_n} \frac{\lambda_n^j}{j!} = 1$. □

The following result provide some bound for the probability distribution of a renewal process when the underlying renewal distribution F is NBU/ NWU.

Theorem 1: If F is NBU (NWU), then

$$P(N(t) < n) \geq (\leq) e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!}, n \geq 1, t > 0. \quad (3)$$

In particular, if F is exponential df with mean $\frac{1}{\mu}, \mu > 0$, then equality holds as $\{N(t), t > 0\}$ is a Poisson process with rate μ .

Proof: Since NWU proof is similar, we give proof only when F is NBU. If X is NBU then $X \leq_{su} Y$, where Y has df $G(t) = 1 - e^{-\lambda t}, t > 0, \lambda > 0$, and $X \leq_{su} Y$ means $G^{-1}(F(t))$ is super-additive (su). If $W(t) = G^{-1}(F(t))$ then

$$W(t) = -\frac{\ln(1 - F(t))}{\lambda} = -\frac{\ln \bar{F}(t)}{\lambda} = \frac{R(t)}{\lambda}, t \geq 0,$$

so that $W(\cdot)$ is super-additive/ sub-additive iff $R(t)$ is super-additive/ sub-additive. If F is NBU, $R(\cdot)$ is super-additive. Therefore, $R^{-1}(\cdot)$ is sub-additive (Østerdal (2006)), that is, $R^{-1}(x + y) \leq R^{-1}(x) + R^{-1}(y), x, y \geq 0$. Let $Y_1, Y_2, \dots$ be iid rvs with df $H(x) = 1 - e^{-x}, x \geq 0$. Then, with $X_j = R^{-1}(Y_j), j \geq 1$, $X_1, X_2, \dots$ are iid rvs with df F , as $P(X_j \leq x) = P(R^{-1}(Y_j) \leq x) = P(Y_j \leq R(x)) = H(R(x)) = F(x)$. Also, $X_1, X_2, \dots$ generate the renewal process $\{N(t), t \geq 0\}$ with renewal df F whereas the iid inter-arrival times $Y_1, Y_2, \dots$ generate a Poisson process with rate 1. Therefore,

$$\begin{aligned} P(N(t) < n) &= P\left(\sum_{j=1}^n X_j > t\right) \\ &= P\left(\sum_{j=1}^n R^{-1}(Y_j) > t\right) \\ &\geq P\left(R^{-1}\left(\sum_{j=1}^n Y_j\right) > t\right) \\ &= P\left(\sum_{j=1}^n Y_j > R(t)\right) \\ &= \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!} e^{-R(t)} = e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!}, \end{aligned}$$

since $\sum_{j=1}^n R^{-1}(Y_j) \geq R^{-1}\left(\sum_{j=1}^n Y_j\right)$ as $R^{-1}(t)$ is sub-additive completing the proof. $\square$

The following result provides bounds for $P(N(t) < n)$ when F is IFR/ DFR.

Theorem 2:

(i) If F is IFR, then

$$e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!} \leq P(N(t) < n) \leq e^{-nR\left(\frac{t}{n}\right)} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!}, n \geq 1. \quad (4)$$

(ii) If F is DFR, then

$$e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!} \leq P(N(t) < n) \leq e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!}, n \geq 1. \quad (5)$$

Proof: We first prove that when F is IFR (DFR), then

$$P(N(t) < n) \leq (\geq) e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!}, n \geq 1, t > 0. \quad (6)$$

We give the proof when F is IFR and the proof is similar in the DFR case.

Let F be IFR and $G(x) = 1 - e^{-x}, x > 0$, be the df of standard exponential rv Y . With $\lambda_n = nR(\frac{t}{n}), n \geq 1$, note that $\lim_{n \rightarrow \infty} \frac{\lambda_n}{n} = 0$ as $R(0) = 0$. Hence by Lemma 1, we have

$$\begin{aligned} P(N(t) < n) &\leq e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!} \\ \Leftrightarrow P(N(t) \geq n) &\geq e^{-nR(\frac{t}{n})} \sum_{j=n}^{\infty} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!}. \end{aligned}$$

Using (1), it is enough to show that $F_n(t) \geq G_n\left(nR\left(\frac{t}{n}\right)\right), t \geq 0$, where G_n is n -fold convolution of G with itself. For $n = 1$, equality holds since $G_1(R(t)) = G(-\ln \bar{F}(t)) = 1 - \bar{F}(t) = F_1(t)$. Assuming that the inequality holds for $n - 1$, we have

$$F_n(t) = \int_0^t F_{n-1}(t-x) dF(x) \geq \int_0^t G_{n-1}\left((n-1)R\left(\frac{t-x}{n-1}\right)\right) dG(R(x)). \quad (7)$$

If X_1 is IFR, then X_1 is IFRA and by 5.4. Results. (e) in page 107 of Barlow and Proschan (1975), this is equivalent to $X \leq_c Y$, where $X \leq_c Y$ means $G^{-1}(F(t))$ is convex, that is, X is less than Y in convex order (see Appendix). We then have

$$W(t) = G^{-1}(F(t)) = -\ln(1 - F(t)) = -\ln \bar{F}(t) = R(t), t \geq 0.$$

Hence, $W(t)$ is convex/ concave iff $R(t)$ is convex/ concave. Since $R(\cdot)$ is convex, we have

$$\begin{aligned} \frac{t}{n} &= \left(\frac{n-1}{n}\right) \left(\frac{t-x}{n-1}\right) + \frac{x}{n} \\ \Rightarrow R\left(\frac{t}{n}\right) &\leq \left(\frac{n-1}{n}\right) R\left(\frac{t-x}{n-1}\right) + \frac{R(x)}{n} \end{aligned}$$

$$\begin{aligned}
\Rightarrow (n-1)R\left(\frac{t-x}{n-1}\right) &\geq nR\left(\frac{t}{n}\right) - R(x) \\
\Rightarrow F_n(t) &\geq \int_0^t G_{n-1}\left(nR\left(\frac{t}{n}\right) - R(x)\right) dG(R(x)) \quad \text{using (7)} \\
\Rightarrow F_n(t) &\geq \int_0^{R(t)} G_{n-1}\left(nR\left(\frac{t}{n}\right) - u\right) dG(u) \\
&\geq \int_0^{nR\left(\frac{t}{n}\right)} G_{n-1}\left(nR\left(\frac{t}{n}\right) - u\right) dG(u) \\
&= G_n\left(nR\left(\frac{t}{n}\right)\right),
\end{aligned}$$

as $R\left(\frac{t}{n}\right) \leq \frac{R(t)}{n}$ since $R(0) = 0$ and $R(\cdot)$ is convex. Therefore, (6) holds for n and by induction, the proof is complete. Since $IFR \Rightarrow IFRA \Rightarrow NBU$, from (6) and (3), we get (4), and since $DFR \Rightarrow DFRA \Rightarrow NWU$, from (6) and (3), we get (5). $\square$

Remark 1: Note that if F is IFR, then it is IFRA and hence $R(\cdot)$ is star-shaped which means that $\frac{R(t)}{t}$ increases as t increases. Hence, for $n \geq 1$, $nR\left(\frac{t}{n}\right) \leq R(t)$ and $e^{-nR\left(\frac{t}{n}\right)} \geq e^{-R(t)}$, $t > 0$. But, the left hand side in (4) is less than or equal to the right hand side. A similar remark holds for (5). $\square$

An interesting property for the bound of $P(N(t) < n)$ when F is IFR/ DFR is obtained as follows.

Theorem 3: If F is IFR (DFR), then

$$P(N(t) < n) \leq (\geq) P(Y_n \leq n-1), n \geq 1, t > 0, \quad (8)$$

where $Y_n \sim \text{Poisson}\left(-\ln \bar{F}^n\left(\frac{t}{n}\right)\right)$, with equality if $\bar{F}(t) = e^{-\mu t}$, $t > 0, \mu > 0$. Also,

$$Y_n \xrightarrow{d} Y \quad (9)$$

where $\xrightarrow{d}$ denotes convergence in distribution and $Y \sim \text{Poisson}(tf(0))$, provided F is absolutely continuous with probability density function (pdf) f .

Proof: Let $Y_n \sim \text{Poisson}\left(-\ln \bar{F}^n\left(\frac{t}{n}\right)\right)$, $n \geq 1$, so that

$$P(Y_n = j) = \frac{\left(-\ln \bar{F}^n\left(\frac{t}{n}\right)\right)^j}{j!} \bar{F}^n\left(\frac{t}{n}\right), j \geq 0. \quad (10)$$

If F is IFR (DFR), then from Theorem 2, $P(N(t) < n) \leq (\geq) P(Y_n \leq n-1)$, $n \geq 1$. Since $nR\left(\frac{t}{n}\right) = -n \ln \bar{F}\left(\frac{t}{n}\right) = -\ln \bar{F}^n\left(\frac{t}{n}\right)$ and $e^{-nR\left(\frac{t}{n}\right)} = \bar{F}^n\left(\frac{t}{n}\right)$, we obtain (8). If F has pdf f , for fixed $t > 0$, $\bar{F}\left(\frac{t}{n}\right) \uparrow 1$ as $n \rightarrow \infty$ so that $\lim_{n \rightarrow \infty} n \ln \bar{F}\left(\frac{t}{n}\right) = \lim_{n \rightarrow \infty} \frac{\ln \bar{F}\left(\frac{t}{n}\right)}{\frac{1}{n}} = \lim_{x \rightarrow 0} \frac{\ln \bar{F}(tx)}{x} = \lim_{x \rightarrow 0} \frac{-tf(tx)}{\bar{F}(tx)} = -tf(0)$, using L'Hospital's rule. Thus $\lim_{n \rightarrow \infty} \bar{F}^n\left(\frac{t}{n}\right) = e^{-tf(0)}$. Since $\ln \bar{F}\left(\frac{t}{n}\right) \uparrow 0$ as $n \rightarrow \infty$, $n \ln \bar{F}\left(\frac{t}{n}\right) \downarrow -tf(0)$ as $n \rightarrow \infty$ and hence $-\ln \bar{F}^n\left(\frac{t}{n}\right) \uparrow tf(0)$ as $n \rightarrow \infty$, proving (9). $\square$

Example 1: If $\bar{F}(t) = e^{-\lambda t}, t > 0, \lambda > 0$, then $P(N(t) < n) = P(Y_n \leq n-1), n \geq 1, t > 0$, where $Y_n \stackrel{d}{=} Y \sim \text{Poisson}(\lambda t)$, $\stackrel{d}{=}$ denoting equality in distribution. This follows trivially as $\{N(t), t \geq 0\}$ is a Poisson process.

Now assume that F is concave (convex). We give an upper (lower) bound for $P(N(t) \geq n)$ as the following theorem.

Theorem 4: If F is concave (convex) and $nF^{-1}\left(\frac{k}{n}\right) \leq t < nF^{-1}\left(\frac{k+1}{n}\right), 0 \leq k \leq n-1, n \geq 1$, then

$$P(N(t) \geq n) \leq (\geq) \frac{1}{n!} \sum_{j=0}^k (-1)^j \binom{n}{j} \left(nF\left(\frac{t}{n}\right) - j \right)^n, \quad (11)$$

with $F^{-1}(t) = \inf \{x : F(x) \geq t\}, 0 \leq t \leq 1$.

Proof: If F is concave, then $F\left(\frac{S_n}{n}\right) \geq \frac{F(X_1)+F(X_2)+\dots+F(X_n)}{n}$ and

$$\begin{aligned} P(N(t) \geq n) &= P(S_n \leq t) \\ &= P\left(F\left(\frac{S_n}{n}\right) \leq F\left(\frac{t}{n}\right)\right) \\ &\leq P\left(\frac{F(X_1) + \dots + F(X_n)}{n} \leq F\left(\frac{t}{n}\right)\right) \\ &= P\left(F(X_1) + \dots + F(X_n) \leq nF\left(\frac{t}{n}\right)\right) \\ &= F_U^{n*}\left(nF\left(\frac{t}{n}\right)\right), \end{aligned}$$

where F_U^{n*} is the n -fold convolution of F_U with itself, U being uniform rv on $(0,1)$ with df F_U . The last equality follows from the fact that, $F(X_i), i = 1, \dots, n$ are iid $U(0,1)$ rvs as F is assumed to be continuous. Therefore, from the well known analytical expression of the n -fold convolution of F_U with itself,

$$F_U^{n*}(t) = \begin{cases} 0, & t < 0, \\ \frac{1}{n!} \sum_{j=0}^k (-1)^j \binom{n}{j} (t-j)^n, & k \leq t < k+1, \quad 0 \leq k \leq (n-1), \end{cases}$$

we get (11). The reverse inequality in (11) can be proved using similar arguments. $\square$

Example 2: Let F be truncated Weibull df with shape parameter $= \frac{1}{2}$ and scale parameter $= 1$. Therefore,

$$F(t) = \begin{cases} 0, & t < 0, \\ \frac{1-e^{-\sqrt{t}}}{1-e^{-3}}, & 0 \leq t < 9, \\ 1, & t \geq 9. \end{cases}$$

Since $\frac{d^2 F}{dt^2} = -\frac{e^{-\sqrt{t}}}{4t(1-e^{-3})} \left(1 + \frac{1}{\sqrt{t}}\right) < 0$ for $t > 0$, F is concave. However F is neither IFR nor DFR as the failure rate function $r(t) = \frac{1}{2\sqrt{t}(1-e^{-\sqrt{t}-3})}, 0 < t < 9$ is not monotone. Also F can not be NWU as there exist a $t \geq 9$ such that $F(t) = 1$. Furthermore, it is observed that F is not NBU as

1. $0 = \bar{F}(9) = P(T > 7 + 2) < P(T > 7)P(T > 2) = \bar{F}(7)\bar{F}(2) = \frac{(1-e^{-\sqrt{7}})(1-e^{-\sqrt{2}})}{(1-e^{-\sqrt{3}})^2}$
2. $\bar{F}(2) = P(T > 1 + 1) > P(T > 1)P(T > 1) = (\bar{F}(1))^2$

Therefore, F is none of IFR, DFR, NBU and NWU distributions, although F is concave and hence bounds given in Theorem 1 and Theorem 2 can not be used for this distribution. However we get an upper bound of $P(N(t) \geq n)$ for $t \geq 0$ and $n = 1, 2, \dots$ by Theorem 4. For example, taking $n = 2$ in (11), we obtain

$$P(N(t) \geq 2) = \begin{cases} \frac{2+2e^{-2\sqrt{\frac{t}{2}}}-4e^{-\sqrt{\frac{t}{2}}}}{(1-e^{-3})^2}, & 0 \leq t < 0.8309, \\ \frac{4e^{-\sqrt{\frac{t}{2}-3}}-2e^{-2\sqrt{\frac{t}{2}}+1}-2e^{-3}-e^{-6}}{(1-e^{-3})^2}, & 0.8309 \leq t < 18, \\ 1, & t \geq 18. \end{cases}$$

We now assume that F is log-concave/ log-convex. Bounds for $P(N(t) < n)$ in this case are given below.

Theorem 5:

- (i) If F is log-concave, for $t > 0$ and $n \geq 1$,

$$e^{-nK(\frac{t}{n})} \sum_{j=n}^{\infty} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!} \leq P(N(t) < n) \leq e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!}. \quad (12)$$

- (ii) If F is log-convex, for $t > 0$ and $n \geq 1$,

$$e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!} \leq P(N(t) < n) \leq e^{-nK(\frac{t}{n})} \sum_{j=n}^{\infty} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!}. \quad (13)$$

Proof: We first prove that if F is log-concave (log-convex), for $t \geq 0$ and $n \geq 1$,

$$P(N(t) \geq n) \leq (\geq) e^{-nK(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!}. \quad (14)$$

Let F is log-concave, or equivalently, $\ln F$ is concave so that

$$\ln F\left(\frac{S_n}{n}\right) \geq \frac{\ln F(X_1) + \dots + \ln F(X_n)}{n}$$

and

$$\begin{aligned}
P(N(t) \geq n) &= P(S_n \leq t) \\
&= P\left(\ln F\left(\frac{S_n}{n}\right) \leq \ln F\left(\frac{t}{n}\right)\right) \\
&\leq P\left(\frac{\ln F(X_1) + \cdots + \ln F(X_n)}{n} \leq \ln F\left(\frac{t}{n}\right)\right) \\
&= P\left(\ln F(X_1) + \cdots + \ln F(X_n) \leq n \ln F\left(\frac{t}{n}\right)\right) \\
&= P\left(-\ln F(X_1) - \cdots - \ln F(X_n) \geq -n \ln F\left(\frac{t}{n}\right)\right) \\
&= 1 - F_Y\left(nK\left(\frac{t}{n}\right)\right),
\end{aligned}$$

where $K(t) = -\ln F(t)$, and $Y = -\ln F(X_1) - \cdots - \ln F(X_n)$ is the sum of iid standard exponential rvs as F is assumed to be continuous and hence has Gamma distribution with shape parameter n and scale parameter 1. Therefore, we get (14). When F is log-convex, the reverse inequality can be proved using similar arguments.

Since log-concave $\Rightarrow$ IFR, from (4) and (14), we get (12), and since log-convex $\Rightarrow$ DFR, from (5) and (14), we get (13), completing the proof. $\square$

The following corollaries provide stricter bounds for $P(N(t) \geq n)$ when F is log-concave or log-convex.

Corollary 1: If F is log-concave, for $t > 0$ and $n \geq 1$,

$$P(N(t) \geq n) \leq \min \left\{ e^{-nK\left(\frac{t}{n}\right)} \sum_{j=0}^{n-1} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!}, 1 - e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!} \right\}. \quad (15)$$

Proof: Since F log-concave implies that F is IFR (Lemma 5.9, page 77, Barlow and Proschan (1975)), by Theorem 5, we get

$$P(N(t) \geq n) \leq e^{-nK\left(\frac{t}{n}\right)} \sum_{j=0}^{n-1} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!}, \quad (16)$$

and by Theorem 2(i), we get

$$e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!} \leq P(N(t) < n) \Leftrightarrow P(N(t) \geq n) \leq 1 - e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!}. \quad (17)$$

Combining (16) and (17), we get (15). $\square$

Corollary 2: If F is log-convex, for $t > 0$ and $n \geq 1$,

$$P(N(t) \geq n) \geq \max \left\{ e^{-nK\left(\frac{t}{n}\right)} \sum_{j=0}^{n-1} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!}, 1 - e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!} \right\}. \quad (18)$$

Proof: Since F log-convex implies that F is DFR (Lemma 5.9, page 77, Barlow and Proschan (1975)), by Theorem 5, we get

$$P(N(t) \geq n) \geq e^{-nK(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!}, \quad (19)$$

and by Theorem 2(ii), we get

$$e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!} \geq P(N(t) < n) \Leftrightarrow P(N(t) \geq n) \geq 1 - e^{-R(t)} \sum_{j=0}^{n-1} \frac{(R(t))^j}{j!}. \quad (20)$$

Combining (19) and (20), we get (18). $\square$

Example 3: When F is Fréchet df with $F(t) = e^{-t^{-\alpha}}, t > 0, \alpha > 0$, $\frac{d \ln F}{dt} = \alpha t^{-\alpha-1}$ and $\frac{d^2 \ln F}{dt^2} = -\alpha(\alpha+1)t^{-\alpha-2} < 0$, so that F is log-concave and by Theorem 5(i), we get the bounds for the renewal process generated by F as

$$e^{-nK(\frac{t}{n})} \sum_{j=n}^{\infty} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!} \leq P(N(t) < n) \leq e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!},$$

where $R\left(\frac{t}{n}\right) = -\ln\left(1 - e^{-\left(\frac{t}{n}\right)^{-\alpha}}\right)$ and $K\left(\frac{t}{n}\right) = \left(\frac{t}{n}\right)^{-\alpha}$.

Example 4: When F is truncated Gumbel df with $F(t) = \frac{e^{-e^{-t}} - e^{-1}}{1 - e^{-1}}, t > 0$, and pdf $f(t) = \frac{dF}{dt} = \frac{e^{-(t+e^{-t})}}{1 - e^{-1}}, t > 0$, $\frac{d \ln F}{dt} = e^{-t} - 1$ and $\frac{d^2 \ln F}{dt^2} = -e^{-t} < 0$. So f is log-concave and since f is continuously differentiable, by Bergstrom and Bagnoli (2005), F is also log-concave. By Theorem 5(i), we get the bounds for the renewal process generated by F as

$$e^{-nK(\frac{t}{n})} \sum_{j=n}^{\infty} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!} \leq P(N(t) < n) \leq e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!},$$

where $R\left(\frac{t}{n}\right) = \ln\left(\frac{1 - e^{-1}}{1 - e^{-\frac{t}{n}}}\right)$ and $K\left(\frac{t}{n}\right) = \ln\left(\frac{1 - e^{-1}}{e^{-e^{-\frac{t}{n}}} - e^{-1}}\right)$.

Example 5: When F is $F(n_1, n_2)$ -distribution with pdf

$$f(t) = \frac{\Gamma\left(\frac{n_1+n_2}{2}\right)}{\Gamma\left(\frac{n_1}{2}\right)\Gamma\left(\frac{n_2}{2}\right)} \left(\frac{n_1}{n_2}\right)^{\frac{n_1}{2}} t^{\frac{n_1}{2}-1} \left(1 + \frac{n_1 t}{n_2}\right)^{-\frac{n_1+n_2}{2}}, t > 0,$$

if $n_1 \leq 2$, f is log-convex (ibid). Since f is continuously differentiable and log-convex on $(0, \infty)$, and $f(0) = 0$, F is also log-convex on $(0, \infty)$ (ibid). By Theorem 5(ii), we get the bounds for the renewal process generated by F as

$$e^{-nR(\frac{t}{n})} \sum_{j=0}^{n-1} \frac{\left(nR\left(\frac{t}{n}\right)\right)^j}{j!} \leq P(N(t) < n) \leq e^{-nK(\frac{t}{n})} \sum_{j=n}^{\infty} \frac{\left(nK\left(\frac{t}{n}\right)\right)^j}{j!},$$

where $R(t) = -\ln \bar{F}(t)$, $K(t) = -\ln F(t)$ and

$$F(t) = \frac{\Gamma\left(\frac{n_1+n_2}{2}\right)}{\Gamma\left(\frac{n_1}{2}\right)\Gamma\left(\frac{n_2}{2}\right)} \int_0^{\frac{n_1 t}{n_1 t + n_2}} x^{\frac{n_1}{2}-1} (1-x)^{\frac{n_2}{2}-1} dx.$$

2.1. Some numerical illustrations

Example 6: Consider the df

$$F(t) = \begin{cases} 0, & t < 0, \\ 2t - t^2, & 0 \leq t < 1, \\ 1, & t \geq 1. \end{cases}$$

Since $\frac{d^2 F}{dt^2} = -2 < 0$, F is concave and since the failure rate function $r(t) = \frac{2}{1-t}$, $0 \leq t < 1$ is increasing, F is also IFR. For $n = 2$, we obtain

$$F_2(t) = \begin{cases} 0, & t < 0, \\ 2t^2 - \frac{4t^3}{3} + \frac{t^4}{6}, & 0 \leq t < 1, \\ -\frac{5}{3} + \frac{16t}{3} - 4t^2 + \frac{4t^3}{3} - \frac{t^4}{6}, & 1 \leq t < 2, \\ 1, & t \geq 2, \end{cases}$$

Therefore $P(N(t) < 2) = P(S_2 > t) = 1 - F_2(t)$. Since F is IFR, the upper and lower bound for $P(N(t) < 2)$ is obtained as

$$u(t) = \left(1 - \frac{t}{2}\right)^4 \left(1 - \ln\left(1 - \frac{t}{2}\right)\right), 0 \leq t < 2 \quad (21)$$

and

$$l_1(t) = (1-t)^2(1-2\ln(1-t)), 0 \leq t < 1 \quad (22)$$

respectively, using (4). Since F is concave, the lower bound for $P(N(t) < 2)$ is obtained as

$$l_2(t) = \begin{cases} 1 - (2t^2 - t^3 + t^4/8), & 0 \leq t \leq 0.5858, \\ 2(1-t+t^2/4)^2, & 0.5858 \leq t < 2, \\ 0, & t \geq 2, \end{cases} \quad (23)$$

using (11).

The graphs of exact value $P(N(t) < 2)$, upper bound $u(t)$ in (21), lower bound $l_1(t)$ in (22) and another lower bound $l_2(t)$ in (23) are given in Figure 3. It is observed from Figure 3 that lower bound in (23) is more strict than the lower bound in (22) given by Barlow and Proschan (1975). The upper and lower bounds for $P(N(t) < 3)$ is shown in Figures 1 and 2 for the df in Example 3 and 4, respectively. It is to be noted that in both cases, the exact value for $P(N(t) < n)$, $n \geq 2$ cannot be obtained explicitly and hence the bounds provide estimates, specially so for Example 4 where the bounds are close enough. The graphs are plotted using R.

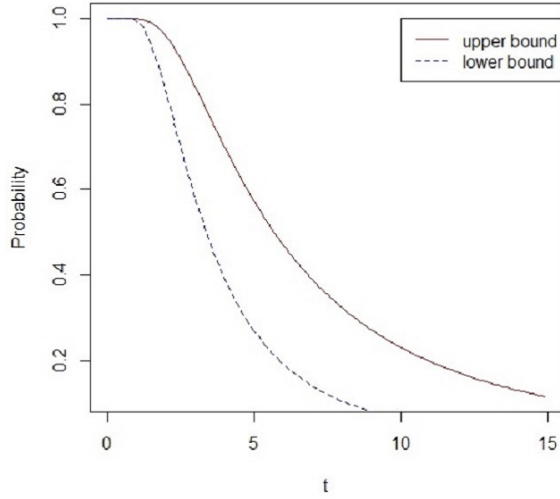


Figure 1: $P(N(t) < 3)$ in Example 3 for $\alpha = 1$

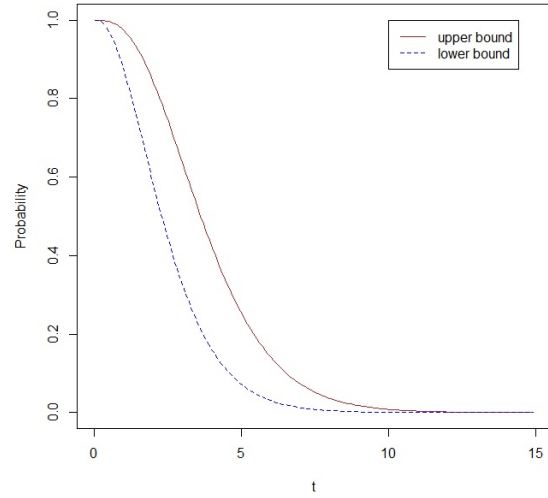


Figure 2: $P(N(t) < 3)$ in Example 4

Example 7: Let F be Weibull df with $F(t) = 1 - e^{-t^2}$, $t \geq 0$. Since $\frac{d^2 F}{dt^2} = 2(1 - 2t^2)e^{-t^2} < 0$ for $t > \frac{1}{\sqrt{2}}$ and $\frac{d^2 F}{dt^2} > 0$ for $t < \frac{1}{\sqrt{2}}$, F is not concave. However, $\frac{d^2 \ln f}{dt^2} = -2 - \frac{1}{t^2} < 0$, so f is log-concave and since f is continuously differentiable, by Bergstrom and Bagnoli (2005), F is log-concave. Therefore applying Corollary 1, we obtain the upper bound for $P(N(t) \geq 3)$

$$P(N(t) \geq 3) \leq \min \left[1 - \left(1 + t^2 + \frac{t^4}{2} \right) e^{-t^2}, \right. \\ \left. \left(1 - e^{-\frac{t^2}{9}} \right)^3 \left(1 - 3 \ln \left(1 - e^{-\frac{t^2}{9}} \right) \right) + \frac{9}{2} \left(\ln \left(1 - e^{-\frac{t^2}{9}} \right) \right)^2 \right]$$

Figure 4 displays the upper bounds from which it is observed that the upper bound in (16) is more strict than the upper bound in (17) given by Barlow and Proschan (1975).

3. Comparing renewal processes with respect to stochastic ordering

Comparison of two renewal processes generated by different renewal distributions is developed in this section on the basis of super-additive/ sub-additive order and convex/ concave transform order. Theorem 6 and Theorem 7 provide bound for the probability distribution of a renewal process $\{N_F(t), t \geq 0\}$ in the form of probability distribution of another renewal process $\{N_G(t), t \geq 0\}$ when the underlying renewal dfs F and G have super-additive/ sub-additive order and convex/ concave transform order respectively.

Theorem 6: Let $\{N_F(t), t \geq 0\}$ and $\{N_G(t), t \geq 0\}$ be two renewal processes generated respectively by dfs F and G which are continuous on $[0, \infty)$ with $F(0) = G(0) = 0$ and $F \leq_{su} (\leq_{sa}) G$, that is, $G^{-1}(F(\cdot))$ is superadditive (subadditive). If $W(t) = G^{-1}(F(t))$, $t > 0$, then, for $n \geq 1$,

$$P(N_F(t) \geq n) \leq (\geq) P(N_G(W(t)) \geq n).$$

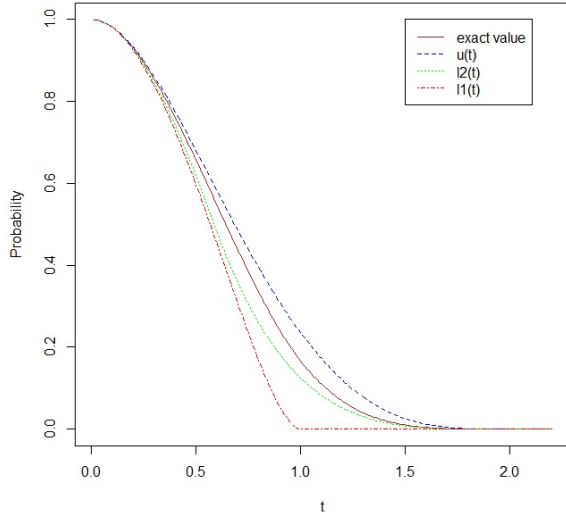


Figure 3: $P(N(t) < 2)$ in Example 6

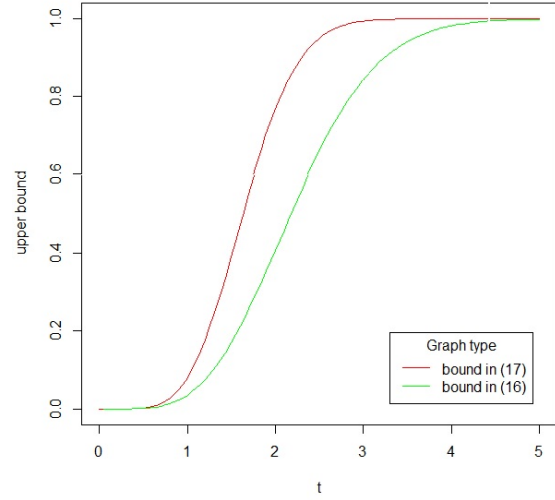


Figure 4: $P(N(t) \geq 3)$ in Example 7

Proof: Since $F \leq_{su} G$, $W^{-1}(\cdot)$ is sub-additive (Østerdal (2006)), that is, $W^{-1}(x + y) \leq W^{-1}(x) + W^{-1}(y)$, $x, y \geq 0$. Let $\{X_n, n \geq 1\}$ and $\{Y_n, n \geq 1\}$ be sequences of iid rvs denoting the inter-arrival times associated with the renewal processes $\{N_F(t), t \geq 0\}$ and $\{N_G(t), t \geq 0\}$ respectively. Then $P(N_F(t) \geq n) = P(S_n \leq t) = F_n(t)$, the n -fold convolution of F with itself, where $S_n = X_1 + \dots + X_n$ is distributed as F_n . Similarly, $P(N_G(t) \geq n) = G_n(t)$. It is enough to show that $F_n(t) \leq G_n(W(t))$. We have $P(X_j \leq t) = F(t) = G(W(t)) = P(Y_j \leq W(t)) = P(W^{-1}(Y_j) \leq t)$ so that

$$\begin{aligned}
 F_n(t) &= P(N_F(t) \geq n) \\
 &= P\left(\sum_{j=1}^n X_j \leq t\right) \\
 &= P\left(\sum_{j=1}^n W^{-1}(Y_j) \leq t\right) \\
 &\leq P\left(W^{-1}\left(\sum_{j=1}^n Y_j\right) \leq t\right) \\
 &= P\left(\sum_{j=1}^n Y_j \leq W(t)\right) \\
 &= G_n(W(t))
 \end{aligned}$$

since $W^{-1}(t)$ is sub-additive. Using similar arguments, it can be proved that, if $F \leq_{sa} G$, that is, $G^{-1}(F(\cdot))$ is sub-additive, then for $t > 0$ and $n \geq 1$,

$$P(N_F(t) \geq n) \geq P(N_G(W(t)) \geq n),$$

completing the proof. □

Theorem 7: Let $\{N_F(t), t \geq 0\}$ and $\{N_G(t), t \geq 0\}$ be two renewal processes generated respectively by two dfs F and G which are continuous on $[0, \infty)$ with $F(0) = G(0) = 0$ and $F \leq_c G$. If $W(t) = G^{-1}(F(t)), t > 0$, then, for $n \geq 1$,

$$P(N_G(W(t)) < n) \leq P(N_F(t) < n) \leq P\left(N_G\left(nW\left(\frac{t}{n}\right)\right) < n\right). \quad (24)$$

The inequalities will reverse when $G \leq_c F$.

Proof: Let $F \leq_c G$ and $W(t) = G^{-1}(F(t)), t > 0$. We first prove that

$$P(N_F(t) < n) \leq P\left(N_G\left(nW\left(\frac{t}{n}\right)\right) < n\right). \quad (25)$$

Note that $W(\cdot)$ is a convex function by definition and $P(N_F(t) \geq n) = P(S_n \leq t) = F_n(t)$, the n -fold convolution of F with itself, $P(N_G(t) \geq n) = G_n(t)$ and it is sufficient to show that $F_n(t) \geq G_n\left(nW\left(\frac{t}{n}\right)\right)$ as in the proof of Theorem 2. Equality holds for $n = 1$ by the definition of $W(\cdot)$. Assuming that the inequality holds for $n - 1$, we have

$$F_n(t) = \int_0^t F_{n-1}(t-x) dF(x) \geq \int_0^t G_{n-1}\left((n-1)W\left(\frac{t-x}{n-1}\right)\right) dG(W(x)). \quad (26)$$

Since $W(\cdot)$ is a convex function,

$$\begin{aligned} \frac{t}{n} &= \left(\frac{n-1}{n}\right) \left(\frac{t-x}{n-1}\right) + \frac{x}{n} \\ \Rightarrow W\left(\frac{t}{n}\right) &\leq \left(\frac{n-1}{n}\right) W\left(\frac{t-x}{n-1}\right) + \frac{W(x)}{n} \\ \Rightarrow (n-1)W\left(\frac{t-x}{n-1}\right) &\geq nW\left(\frac{t}{n}\right) - W(x) \\ \Rightarrow F_n(t) &\geq \int_0^t G_{n-1}\left(nW\left(\frac{t}{n}\right) - W(x)\right) dG(W(x)) \\ &= \int_0^{W(t)} G_{n-1}\left(nW\left(\frac{t}{n}\right) - u\right) dG(u) \\ &\geq \int_0^{nW\left(\frac{t}{n}\right)} G_{n-1}\left(nW\left(\frac{t}{n}\right) - u\right) dG(u) \\ &= G_n\left(nW\left(\frac{t}{n}\right)\right) \end{aligned}$$

using (26) and since $W\left(\frac{t}{n}\right) \leq \frac{W(t)}{n}$ as $W(0) = 0$ and $W(\cdot)$ is convex. Therefore, the inequality also holds for n , and by induction, the proof is complete. Using similar arguments, it can be proved that if $G \leq_c F$, equivalently, $G^{-1}F$ is concave, then for $t > 0$ and $n \geq 1$,

$$P(N_F(t) < n) \geq P\left(N_G\left(nW\left(\frac{t}{n}\right)\right) < n\right).$$

Since $F \leq_c G \Rightarrow F \leq_{su} G$, applying Theorem 6, we get

$$P(N_F(t) < n) \geq P(N_G(W(t)) < n). \quad (27)$$

Combining (25) and (27), we get the (24). Using similar arguments, it can be proved that if $G \leq_c F$ or in other words, $G^{-1}(F(\cdot))$ is concave, then for $t > 0$ and $n \geq 1$,

$$P\left(N_G\left(nW\left(\frac{t}{n}\right)\right) < n\right) \leq P(N_F(t) < n) \leq P(N_G(W(t)) < n),$$

completing the proof. $\square$

Corollary 3: If G in Theorem 7 is exponential so that $\{N_G(t), t \geq 0\}$ is a Poisson process and F is log-concave (log-convex), then the bounds in Theorem 7 are the same as those in Theorem 5(i) (Theorem 5(ii)).

Remark 2: Under the assumption that $W(t)$ is either convex for all $t > 0$ or concave for all $t > 0$, the reverse implications in Theorems 6, 7, are trivially true giving rise to characterization results.

Example 8: Let us consider two renewal processes: $\{N_F(t), t \geq 0\}$ and $\{N_G(t), t \geq 0\}$ where F is Weibull (μ_1, γ) and G is exponential (μ_2) with $\gamma > 1, \mu_1 > 0, \mu_2 > 0$. Then we obtain the following bounds: for $t > 0$ and $n \geq 1$,

$$e^{-\mu_1 t^\gamma} \sum_{j=0}^{n-1} \frac{(\mu_1 t^\gamma)^j}{j!} \leq P(N_F(t) < n) \leq e^{-\left(\frac{\mu_1 t^\gamma}{n^{\gamma-1}}\right)} \sum_{j=0}^{n-1} \frac{\left(\frac{\mu_1 t^\gamma}{n^{\gamma-1}}\right)^j}{j!}.$$

The inequalities will reverse when $0 < \gamma < 1$.

We have $F(t) = 1 - e^{-\mu_1 t^\gamma}$, $G(t) = 1 - e^{-\mu_2 t}$, $W(t) = G^{-1}(F(t)) = -\frac{\ln(1-F(t))}{\mu_2} = \left(\frac{\mu_1}{\mu_2}\right) t^\gamma, t > 0$.

Therefore, $\frac{dW}{dt} = \left(\frac{\gamma \mu_1}{\mu_2}\right) t^{\gamma-1} \Rightarrow \frac{d^2 W}{dt^2} = \left(\frac{\gamma(\gamma-1)\mu_1}{\mu_2}\right) t^{\gamma-2} \Rightarrow \frac{d^2 W}{dt^2} > (<) 0$ for $t \geq 0$,

$\gamma > (<) 1, \mu_1, \mu_2 > 0$. Therefore, $W(\cdot)$ is a convex (concave) function as $\gamma > (<) 1$ and hence we can apply Theorem 7. Since $\{N_G(t), t \geq 0\}$ is a Poisson process with rate μ_2 , by Theorem 7, lower bound $= P(N_G(W(t)) < n) = \sum_{j=0}^{n-1} \frac{(\mu_2 W(t))^j}{j!} e^{-\mu_2 W(t)} = \sum_{j=0}^{n-1} \frac{(\mu_1 t^\gamma)^j}{j!} e^{-\mu_1 t^\gamma} = e^{-\mu_1 t^\gamma} \sum_{j=0}^{n-1} \frac{(\mu_1 t^\gamma)^j}{j!}$. By Theorem 7, upper bound $= P\left(N_G\left(nW\left(\frac{t}{n}\right)\right) < n\right)$

$$= \sum_{j=0}^{n-1} \frac{(\mu_2 nW(\frac{t}{n}))^j}{j!} e^{-\mu_2 nW(\frac{t}{n})} = \sum_{j=0}^{n-1} \frac{\left(\frac{\mu_1 t^\gamma}{n^{\gamma-1}}\right)^j}{j!} e^{-\left(\frac{\mu_1 t^\gamma}{n^{\gamma-1}}\right)} = e^{-\left(\frac{\mu_1 t^\gamma}{n^{\gamma-1}}\right)} \sum_{j=0}^{n-1} \frac{\left(\frac{\mu_1 t^\gamma}{n^{\gamma-1}}\right)^j}{j!}.$$

Example 9: Let us consider two renewal processes $\{N_F(t), t \geq 0\}$ and $\{N_G(t), t \geq 0\}$ where F is Pareto (α, β) and G is exponential (μ) . Then we have the following bounds:

$$\left(1 + \frac{t}{n\beta}\right)^{-n\alpha} \sum_{j=0}^{n-1} \frac{(n\alpha \ln(1 + \frac{t}{n\beta}))^j}{j!} \leq P(N_F(t) < n) \leq \left(1 + \frac{t}{\beta}\right)^{-\alpha} \sum_{j=0}^{n-1} \frac{(\alpha \ln(1 + \frac{t}{\beta}))^j}{j!},$$

$t \geq 0, n \geq 1, \alpha > 0, \beta > 0, \mu > 0$.

We have $F(t) = 1 - \left(\frac{\beta}{\beta+t}\right)^\alpha, t > -\beta, G^{-1}(y) = -\frac{\ln(1-y)}{\mu}, 0 < y < 1, W(t) = G^{-1}(F(t)) =$

$\frac{\alpha}{\mu} \ln\left(1 + \frac{t}{\beta}\right), t > 0$. So, $\frac{dW}{dt} = \frac{\alpha}{\mu} \left(\frac{1}{\beta+t}\right), \frac{d^2 W}{dt^2} = -\frac{\alpha}{\mu} \left(\frac{1}{\beta+t}\right)^2 < 0$ for $t \geq 0, \alpha > 0, \beta > 0, \mu > 0$.

Therefore $W(\cdot)$ is concave and by Theorem 7, since $G \sim$ exponential (μ) , $\{N_G(t), t \geq 0\}$ is a Poisson process with rate μ and $P(N_G(W(t)) < n) = \left(1 + \frac{t}{\beta}\right)^{-\alpha} \sum_{j=0}^{n-1} \frac{(\alpha \ln(1 + \frac{t}{\beta}))^j}{j!}$. By

Theorem 7, $P\left(N_G\left(nW\left(\frac{t}{n}\right)\right) < n\right) = \left(1 + \frac{t}{n\beta}\right)^{-n\alpha} \sum_{j=0}^{n-1} \frac{(n\alpha \ln(1 + \frac{t}{n\beta}))^j}{j!}$.

4. Conclusions

In this paper, we have presented some bounds on the sf and df of the renewal counting variable $N(t)$ when the renewal distribution F is concave, log-concave, convex, log-convex, IFR, DFR, NBU, NWU. The bounds would be useful when the exact value for the probability distribution of a renewal process can not be obtained. The bounds also appears to be more strict than the previously obtained bounds which is illustrated with some numerical examples. Some bounds are also obtained by comparing renewal processes with respect to stochastic order relationship between the inter-arrival times. In future, we plan to develop bounds for the probability distribution of bivariate renewal process based on various forms of the baseline joint distribution function.

Acknowledgements

We are indeed grateful to the Editors for their guidance and counsel. We are very grateful to the reviewer for valuable comments and suggestions of generously listing many useful references.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Barlow, R. E. and Proschan, F. (1975). *Statistical Theory of Reliability and Life Testing: Probability Models*. Holt, Rinehart and Winston, New York.
- Bergstrom, T. and Bagnoli, M. (2005). Log-concave probability and its applications. *Economic Theory*, **26**, 445–469.
- Ross, S. M. (1995). *Stochastic Processes*. John Wiley & Sons.
- Shaked, M. and Shanthikumar, J. G. (2007). *Stochastic Orders*. Springer.
- Østerdal, L. P. (2006). Subadditive functions and their (pseudo-) inverses. *Journal of Mathematical Analysis and Applications*, **317**, 724–731.

Appendix

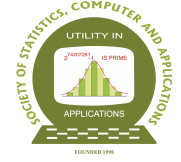
Assuming that a rv X has df $F(\cdot)$ and sf $\bar{F}(\cdot)$ with mean μ , $F(0) = 0$, we have used the following definitions (for details see Barlow and Proschan 1975, Shaked and Shanthikumar 2007, Bagnoli and Bergstrom 2005):

1. X / F is said to be new better than used (NBU) or new worse than used (NWU) accordingly as $\bar{F}(x + y) \leq (\geq) \bar{F}(x)\bar{F}(y)$ for all $x, y \geq 0$.
2. X / F is said to be new better than used in expectation (NBUE) or new worse than used in expectation (NWUE) accordingly as $\int_t^\infty \bar{F}(x)dx \leq (\geq) \mu\bar{F}(t)$ for all $t \geq 0$.

3. X / F is said to be increasing failure rate (IFR) or decreasing failure rate (DFR) if and only if $\frac{F(t+x)-F(t)}{F(t)}$ is increasing (decreasing) in t for all $x > 0$.
4. X / F is said to be increasing failure rate average (IFRA) or decreasing failure rate average (DFRA) if $(\bar{F}(x))^{\frac{1}{x}}$ decreasing (increasing) in x for $x > 0$.
5. F is said to be concave (convex) if for any $x, y \geq 0$ and for any $\alpha \in [0, 1]$

$$F((1 - \alpha)x + \alpha y) \geq (\leq) (1 - \alpha)F(x) + \alpha F(y).$$

6. F is said to be log-concave (log-convex) if $\ln F$ is concave (convex).
7. F is said to be super-additive (sub-additive) if $F(x + y) \geq (\leq) F(x) + F(y)$ for all $x, y \geq 0$.
8. For two independent rvs X and Y with respective dfs F and G , X is said to be smaller than Y is convex transform order, denoted $X \leq_c Y$, if $G^{-1}F$ is convex on the support of F .
9. For two independent rvs X and Y with respective dfs F and G , X is said to be smaller than Y is super-additive (sub-additive) order, denoted $X \leq_{su} (\leq_{sa}) Y$, if $G^{-1}F$ is super-additive (sub-additive), or equivalently, $G^{-1}F(x + y) \geq (\leq) G^{-1}F(x) + G^{-1}F(y)$ for all $x, y \geq 0$.



The Modified Logistic Two-Parameter Estimators

Thayaparan Kayathiri¹, Manickavasagar Kayanan², and Pushpakanthie Wijekoon³

¹*Postgraduate Institute of Science, University of Peradeniya, Peradeniya, Sri Lanka*

²*Department of Physical Science, University of Vavuniya, Vavuniya, Sri Lanka*

³*Department of Statistics and Computer Science, University of Peradeniya, Peradeniya, Sri Lanka*

Received: 08 June 2024; Revised: 02 October 2024; Accepted: 04 October 2024

Abstract

In this article, we proposed a new estimator, termed the Modified Logistic Two-Parameter Estimator (MLTPE), and enhanced it by modifying its coefficients, yielding three variants: Modified Logistic Two-Parameter Estimator1 (MLTPE1), Modified Logistic Two-Parameter Estimator2 (MLTPE2), and Modified Logistic Two-Parameter Estimator3 (MLTPE3). These estimators are designed for logistic regression models in the presence of multicollinearity. Theoretically, we demonstrated the superiority of the MLTPE over existing estimators, including the Maximum Likelihood Estimator (MLE), Modified Almost Unbiased Ridge Logistic Estimator (MAURLE), and Logistic Two-Parameter Estimator (LTPE), in terms of mean square error (MSE). The superiority of the estimators is examined using a simulation study and a real-world example. In the simulation study, we varied the degree of correlation and sample size. The findings revealed that the efficacy of the estimators is significantly influenced by these factors. Furthermore, we evaluated the prediction performance of these estimators using balanced accuracy. The results suggested that the new estimators, MLTPE1, MLTPE2, and MLTPE3, outperformed the others slightly in terms of balanced accuracy, with MLTPE2 exhibiting superior performance regarding both scalar mean square error (SMSE) and balanced accuracy. Finally, we validated the simulation study using the myopia dataset, which produced satisfactory results.

Key words: Balanced accuracy; Prediction performance; Simulation and Scalar mean square error.

AMS Subject Classifications: 62J07, 62J12

1. Introduction

The logistic regression model is specified as a Bernoulli distribution since y_i is a binary response variable: $y_i \sim \text{Ber}(\pi_i)$

$$P(y_i) = \pi_i^{y_i} \cdot (1 - \pi_i)^{1-y_i}$$

The general form of the logistic regression model is given by

$$y_i = \pi_i + \epsilon_i, \quad i = 1, 2, \dots, n \quad (1)$$

where,

$$\pi_i = \frac{\exp(x_i' \beta)}{1 + \exp(x_i' \beta)}, \quad (2)$$

x_i is the i^{th} row of X , which is an $n \times (p+1)$ data matrix with p predictor variables and β is a $(p+1) \times 1$ vector of coefficients, ϵ_i are independent with mean zero and variance $\pi_i(1 - \pi_i)$ of the response y_i .

The maximum likelihood estimation technique is a commonly used method to estimate the parameter vector (β), and the Maximum Likelihood Estimator (MLE) of β is given by:

$$\hat{\beta}_{MLE} = C^{-1} X' \hat{W} Z. \quad (3)$$

where, $C = X' \hat{W} X$; Z is the column vector with i^{th} element equals $\text{logit}(\hat{\pi}_i) + \frac{y_i - \hat{\pi}_i}{\hat{\pi}_i(1 - \hat{\pi}_i)}$, and, $\hat{W} = \text{diag}[\hat{\pi}_i(1 - \hat{\pi}_i)]$.

Since

$$E(\hat{\beta}_{MLE}) = \beta, \quad (4)$$

$\hat{\beta}_{MLE}$ is an unbiased estimate of β . The covariance matrix of $\hat{\beta}_{MLE}$ is

$$\text{Cov}(\hat{\beta}_{MLE}) = (X' \hat{W} X)^{-1} = C^{-1}. \quad (5)$$

Hence, the mean square error (MSE) of $\hat{\beta}_{MLE}$ is

$$\begin{aligned} \text{MSE}[\hat{\beta}_{MLE}] &= \text{Cov}[\hat{\beta}_{MLE}] + B[\hat{\beta}_{MLE}]B'[\hat{\beta}_{MLE}] \\ &= C^{-1}. \end{aligned} \quad (6)$$

Therefore, the scalar mean square error (SMSE) of $\hat{\beta}_{MLE}$ is,

$$\begin{aligned} \text{SMSE}[\hat{\beta}_{MLE}] &= \text{tr}[\text{MSE}(\hat{\beta}_{MLE})] \\ &= \text{tr}[C^{-1}]. \end{aligned} \quad (7)$$

Since C is a positive definite matrix, there exists an orthogonal matrix P such that $P'CP = \Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_p)$, where $\lambda_1 \geq \lambda_2 \geq \dots, \lambda_p > 0$ are the ordered eigen values of C . Then,

$$\text{SMSE}[\hat{\beta}_{MLE}] = \sum_{j=1}^p \frac{1}{\lambda_j}. \quad (8)$$

The Maximum Likelihood Estimator (MLE) is susceptible to the effects of multicollinearity among explanatory variables. This susceptibility results in an inflated variance of the MLE, thereby rendering the estimates inefficient. To address this issue, numerous

alternative estimators have been proposed in the literature. In this study, we considered two such estimators based on the sample information given below to mitigate the impact of multicollinearity.

The Modified Almost Unbiased Ridge Logistic Estimator (MAURLE) is one of the alternative estimators introduced by Varathan (2022), and it is defined as

$$\hat{\beta}_{MAURLE} = F_k \hat{\beta}_{MLE}, \quad (9)$$

where $F_k = [I - k^2(C + kI)^{-2}][(C + kI)^{-1}C]$, $k > 0$.

Another alternative estimator is the Logistic Two-Parameter Estimator (LTPE) proposed by Huang (2012), which is defined as

$$\hat{\beta}_{LTPE} = L_{k,d} \hat{\beta}_{MLE}, \quad (10)$$

where $L_{k,d} = (C + kI)^{-1}(C + kdI)$, $0 < d < 1$, $k > 0$.

Note that the two alternative estimators that we discussed above are a function of $\hat{\beta}_{MLE}$, and we can present them in general form as,

$$\hat{\beta}_{GLE} = J_{(i)} \hat{\beta}_{MLE}, \quad (11)$$

where $J_{(i)}$ is a positive definite matrix.

$$\hat{\beta}_{GLE} = \begin{cases} \hat{\beta}_{MLE} & \text{if } J_{(i)} = I; \\ \hat{\beta}_{MAURLE} & \text{if } J_{(i)} = F_k; \\ \hat{\beta}_{LTPE} & \text{if } J_{(i)} = L_{k,d}. \end{cases} \quad (12)$$

The asymptotic properties of the general form of estimators are

$$E[\hat{\beta}_{GLE}] = E[J_{(i)} \hat{\beta}_{MLE}] = J_{(i)} \beta. \quad (13)$$

and the dispersion matrix;

$$D[\hat{\beta}_{GLE}] = Cov[J_{(i)} \hat{\beta}_{MLE}] = J_{(i)} C^{-1} J_{(i)}'. \quad (14)$$

The bias vector and the Mean square error matrix (MSE) are

$$B[\hat{\beta}_{GLE}] = E[J_{(i)} \hat{\beta}_{MLE}] - \beta = (J_{(i)} - I) \beta. \quad (15)$$

and

$$\begin{aligned} MSE[\hat{\beta}_{GLE}] &= D[\hat{\beta}_{GLE}] + B[\hat{\beta}_{GLE}] B'[\hat{\beta}_{GLE}] \\ &= J_{(i)} C^{-1} J_{(i)}' + (J_{(i)} - I) \beta \beta' (J_{(i)} - I)', \end{aligned} \quad (16)$$

respectively. The scalar mean square error (SMSE) of the estimators can be obtained as,

$$SMSE[\hat{\beta}_{GLE}] = tr[MSE(\hat{\beta}_{GLE})]. \quad (17)$$

The superiority of the above estimators was compared in terms of mean squared error (MSE) and scalar mean square error (SMSE) in literature (Varathan, 2022; Huang, 2012). While MSE is a common metric in regression analysis, it is unsuitable for comparing logistic regression estimators. Logistic regression deals with binary classification problems, where outcomes are categorical rather than continuous. Predicted values in logistic regression represent probabilities rather than actual outcomes. Therefore, using MSE for comparison would be inappropriate as it does not consider the probabilistic nature of the predictions. Instead, evaluation metrics like balanced accuracy, the area under the receiver operating characteristic curve (ROC AUC), or log loss (cross-entropy loss) are more appropriate for assessing logistic regression model performance.

This study aimed to develop new estimators that enhance predictive performance for logistic regression models in the presence of multicollinearity. Balanced accuracy is employed as the primary metric for comparing the estimators. Balanced accuracy is an effective metric for classification problems with imbalanced datasets, where one class may predominate. It computes the average accuracy across each class, integrating sensitivity and specificity, thus providing a more comprehensive evaluation of model performance.

The paper is organized as follows: In Section 2, we propose new estimators, their properties, and the conditions for their superiority over existing estimators. Section 3 presents the simulation study for the proposed estimators, while Section 4 provides validation of the simulation study. Section 5 presents concluding remarks. Finally, the references and appendices are provided at the end of the paper.

2. Proposed estimators and their superiority

2.1. The proposed estimators

Ahmad (2020) developed the Modified New Two Parameter Estimator (MNTPE), which has shown better performance than the ordinary least squares in the linear regression model. The MNTPE is defined as:

$$\hat{\beta}_{MNTPE} = M_{k,d} \hat{\beta}_{OLSE}, \quad (18)$$

where $M_{k,d} = (X'X + I)^{-1}(X'X + dI)(X'X + kdI)^{-1}X'X$, $0 < d < 1$, $k > 0$.

Now, we propose a new estimator, named the Modified Logistic Two-Parameter Estimator (MLTPE), based on the MNTPE by replacing $\hat{\beta}_{OLSE}$ with $\hat{\beta}_{GLE}$ and $X'X$ with $X'\hat{W}X$ in equation (18) as below:

$$\begin{aligned} \hat{\beta}_{MLTPE} &= F_{k,d} \hat{\beta}_{GLE} \\ &= F_{k,d} J_{(i)} \hat{\beta}_{MLE} \\ &= L_{(i)} \hat{\beta}_{MLE}, \end{aligned} \quad (19)$$

where $F_{k,d} = (X'\hat{W}X + I)^{-1}(X'\hat{W}X + dI)(X'\hat{W}X + kdI)^{-1}X'\hat{W}X$ and $L_{(i)} = F_{k,d}J_{(i)}$. Since $C = X'\hat{W}X$, we have $F_{k,d} = (C + I)^{-1}(C + dI)(C + kdI)^{-1}C$, $0 < d < 1$, $k > 0$.

The expectation, bias, covariance, MSE, and SMSE of MLTPE can be defined as:

$$E(\hat{\beta}_{MLTPE}) = L_{(i)}\beta, \quad (20)$$

$$B(\hat{\beta}_{MLTPE}) = (L_{(i)} - I)\beta, \quad (21)$$

$$Cov(\hat{\beta}_{MLTPE}) = L_{(i)}C^{-1}L'_{(i)}, \quad (22)$$

$$\begin{aligned} MSE(\hat{\beta}_{MLTPE}) &= Cov(\hat{\beta}_{MLTPE}) + B(\hat{\beta}_{MLTPE})B(\hat{\beta}_{MLTPE})' \\ &= L_{(i)}C^{-1}L'_{(i)} + (L_{(i)} - I)\beta\beta'(L_{(i)} - I)'. \end{aligned} \quad (23)$$

and

$$\begin{aligned} SMSE(\hat{\beta}_{MLTPE}) &= tr(MSE(\hat{\beta}_{MLTPE})) \\ &= tr(Cov(\hat{\beta}_{MLTPE})) + B(\hat{\beta}_{MLTPE})'B(\hat{\beta}_{MLTPE}) \\ &= tr(L_{(i)}C^{-1}L'_{(i)}) + (L_{(i)} - I)' \beta' \beta (L_{(i)} - I). \end{aligned} \quad (24)$$

By adopting $\hat{\beta}_{MLE}$, $\hat{\beta}_{MAURLE}$ and $\hat{\beta}_{LTPE}$ in place of $\hat{\beta}_{GLE}$ in equation (19), we propose three new estimators namely, Modified logistic two parameter estimator 1 (MLTPE 1), Modified logistic two parameter estimator 2 (MLTPE 2), and Modified logistic two parameter estimator 3 (MLTPE3), respectively, and defined as,

$$\hat{\beta}_{MLTPE} = \begin{cases} \hat{\beta}_{MLTPE1} & \text{if } L_{(i)} = F_{k,d}I; \\ \hat{\beta}_{MLTPE2} & \text{if } L_{(i)} = F_{k,d}F_k; \\ \hat{\beta}_{MLTPE3} & \text{if } L_{(i)} = F_{k,d}L_{k,d}. \end{cases} \quad (25)$$

2.2. Comparison among the estimators

In this section, we compared the performance of the proposed estimator MLTPE with the existing estimator GLE in terms of the mean square error matrix criterion.

Theorem 2.2.1: When $\lambda_{max}[L_{(i)}C^{-1}L'_{(i)}(J_{(i)}C^{-1}J'_{(i)})^{-1}] < 1$, the MLTPE is superior to GLE if and only if $\delta'_{new}(D_1 + \delta'_G\delta_G)^{-1}\delta_{new} \leq 1$.

Proof

Consider

$$\begin{aligned} &MSE(\hat{\beta}_{GLE}) - MSE(\hat{\beta}_{MLTPE}) \\ &= J_{(i)}C^{-1}J'_{(i)} + (J_{(i)} - I)\beta\beta'(J_{(i)} - I)' - (L_{(i)}C^{-1}L'_{(i)} + (L_{(i)} - I)\beta\beta'(L_{(i)} - I)') \\ &= [J_{(i)}C^{-1}J'_{(i)} - L_{(i)}C^{-1}L'_{(i)}] + [(J_{(i)} - I)\beta\beta'(J_{(i)} - I)' - (L_{(i)} - I)\beta\beta'(L_{(i)} - I)']. \end{aligned} \quad (26)$$

Now consider,

$$\begin{aligned} D(\hat{\beta}_{GLE}) - D(\hat{\beta}_{MLTPE}) &= [J_{(i)}C^{-1}J'_{(i)} - L_{(i)}C^{-1}L'_{(i)}] \\ &= D_1. \end{aligned} \quad (27)$$

Note that, Since C^{-1} is a positive definite matrix, $J_{(i)}C^{-1}J'_{(i)}$ and $L_{(i)}C^{-1}L'_{(i)}$ are positive definite matrices (by Lemma 1 in Appendices A.2). Consequently, by Lemma 2 (in Appendices A.2), if $\lambda_{\max}[L_{(i)}C^{-1}L'_{(i)}(J_{(i)}C^{-1}J'_{(i)})^{-1}] < 1$ then D_1 is a positive definite matrix, where $\lambda_{\max}[L_{(i)}C^{-1}L'_{(i)}(J_{(i)}C^{-1}J'_{(i)})^{-1}]$ is the largest eigenvalue of $[L_{(i)}C^{-1}L'_{(i)}(J_{(i)}C^{-1}J'_{(i)})^{-1}]$. Further by Lemma 3 (in Appendices A.2), $MSE(\hat{\beta}_{GLE}) - MSE(\hat{\beta}_{MLTPE})$ is non negative definite if $\delta'_{new}(D_1 + \delta'_G\delta_G)^{-1}\delta_{new} \leq 1$, where $\delta_{new} = (J_{(i)} - I)\beta$ and $\delta_G = (J_{(i)} - I)\beta$. Hence, the estimator MLTPE is superior to GLE if and only if $\delta'_{new}(D_1 + \delta'_G\delta_G)^{-1}\delta_{new} \leq 1$.

Note that the Theorem 2.2.1 outlines the necessary and sufficient conditions for the superiority of the proposed estimator (MLTPE) over the general existing estimator (GLE). By substituting $J_{(i)}$ with an appropriate matrix, we can derive the following conditions for the superiority of MLTPE over the existing estimators MLE, MAURLE, and LTPE with respect to mean square error (MSE).

- If $J_{(i)} = I$; MLTPE is superior than MLE if $\lambda_{\max}[L_{(i)}C^{-1}L'_{(i)}C^{-1}] < 1$ and $\delta'_{new}[C^{-1} - L_{(i)}C^{-1}L'_{(i)}]\delta_{new} \leq 1$.
- If $J_{(i)} = F_k$; MLTPE is superior than MAURLE if $\lambda_{\max}[L_{(i)}C^{-1}L'_{(i)}(F_kC^{-1}F'_k)^{-1}] < 1$ and $\delta'_{new}[(F_kC^{-1}F'_k - L_{(i)}C^{-1}L'_{(i)} + \beta'(F_k - I)'(F_k - I)\beta)^{-1}\delta_{new} \leq 1$.
- If $J_{(i)} = L_{k,d}$; MLTPE is superior than LTPE if $\lambda_{\max}[L_{(i)}C^{-1}L'_{(i)}(L_{k,d}C^{-1}L'_{k,d})^{-1}] < 1$ and $\delta'_{new}[(L_{k,d}C^{-1}L'_{k,d} - L_{(i)}C^{-1}L'_{(i)} + \beta'(L_{k,d} - I)'(L_{k,d} - I)\beta)^{-1}\delta_{new} \leq 1$.

3. Simulation study

Following McDonald and Galarneau (1975), and Alheety *et al.* (2021), we generate the explanatory variables using the following equation.

$$x_{i,j} = \sqrt{(1 - \rho^2)}z_{i,j} + \rho z_{i,p+1} \quad ; i = 1, 2, \dots, n. \quad j = 1, 2, \dots, p. \quad (28)$$

where z_{ij} are pseudo-random numbers from a standard normal distribution, and ρ represents the correlation between any two explanatory variables. Four explanatory variables are generated using equation (28), and we choose $\rho = (0.80, 0.85, 0.90)$. Further, we considered three different sample sizes: 80, 100, and 200. The dependent variable y_i is obtained from the Bernoulli distribution with $\pi_i = \frac{\exp(x'_i\beta)}{1 + \exp(x'_i\beta)}$. The parameter values of $\beta_1, \beta_2, \dots, \beta_p$ are chosen so that $\beta'\beta = 1$. In the simulation study, we implemented a 5-fold cross-validation approach. However, due to the difficulty of achieving balanced accuracy with small sample sizes, we initiated our study with a sample size of 80. The simulation was repeated 1000 times by generating new pseudo-random numbers, and we calculated the SMSE values of the estimators using equation (29).

$$SMSE(\hat{\beta}) = \frac{1}{1000} \sum_{r=1}^{1000} (\hat{\beta}_r - \beta)'(\hat{\beta}_r - \beta), \quad (29)$$

where $\hat{\beta}_r$ denotes any estimator considered in the r^{th} simulation.

The confusion matrix is utilized to determine the balanced accuracy, a widely used measure for solving classification problems. Table 1 details the components of this matrix.

Table 1: Confusion Matrix

		Predicted Outcome	
		P	N
Actual	P	True Positive (TP)	False Negative (FN)
	N	False Positive (FP)	True Negative (TN)

The evaluation metrics are,

$$Sensitivity = \frac{TP}{(TP + FN)}$$

$$Specificity = \frac{TN}{(TN + FP)}$$

$$Balanced\ accuracy = \frac{(Sensitivity + Specificity)}{2}$$

The steps for K-fold cross-validation are:

Step 1: Split the data set into K groups.

Step 2: For each individual group $i=1,2,\dots, K$; take the one group as a test data set and the remaining $(K-1)$ groups as a training data set.

Step 3: Estimate the respective estimator $(\hat{\beta}_{k,d})$ with shrinkage parameter (k, d) using training data sets and compute its cross-validated balanced accuracy in predicting the test data set. Use initial values of k and d as 0.01, then increase the values of k and d by a small increment of 0.1. For $k > 0, 0 < d < 1$.

Step 4: Calculate the average cross-validated balanced accuracy.

Step 5: Choose the maximum value of the average cross-validated balanced accuracy of the estimators and corresponding shrinkage parameters.

The SMSE values and cross-validated balanced accuracy values of the estimators MLE, MAURLE, LTPE, MLTPE1, MLTPE2, and MLTPE3 are obtained for different shrinkage parameters (k, d) . We summarize the results of the simulation study in Table 2 - 4,

Table 2: The results of the simulation study when $n=80$

	Estimators	SMSE	Balanced Accuracy
$\rho = 0.80$	MLE	45.036111	0.517190
	MAURLE	0.574975 ($k=0.61$)	0.528781 ($k=0.01$)
	LTPE	0.704999 ($k=1.71, d=0.01$)	0.527550 ($k=1.41, d=0.01$)
	MLTPE1	0.629874 ($k=2.91, d=0.11$)	0.529005 ($k=4.21, d=0.11$)
	MLTPE2	0.566116 ($k=0.31, d=0.21$)	0.529138 ($k=0.61, d=0.11$)
	MLTPE3	0.610834 ($k=0.11, d=0.01$)	0.528975 ($k=1.51, d=0.11$)
$\rho = 0.85$	MLE	58.195287	0.520008
	MAURLE	0.542518 ($k=0.61$)	0.530163 ($k=0.01$)
	LTPE	0.671765 ($k=1.61, d=0.01$)	0.529779 ($k=2.11, d=0.01$)
	MLTPE1	0.594680 ($k=4.91, d=0.11$)	0.530442 ($k=4.31, d=0.01$)
	MLTPE2	0.536631 ($k=0.31, d=0.21$)	0.530542 ($k=0.41, d=0.99$)
	MLTPE3	0.573781 ($k=0.11, d=0.01$)	0.530594 ($k=2.11, d=0.21$)
$\rho = 0.90$	MLE	84.328874	0.520109
	MAURLE	0.522461 ($k=0.61$)	0.530471 ($k=0.01$)
	LTPE	0.643522 ($k=1.51, d=0.01$)	0.530473 ($k=2.01, d=0.01$)
	MLTPE1	0.557923 ($k=4.91, d=0.11$)	0.530814 ($k=1.81, d=0.21$)
	MLTPE2	0.520367 ($k=0.31, d=0.21$)	0.530821 ($k=1.21, d=0.01$)
	MLTPE3	0.547428 ($k=0.11, d=0.01$)	0.530921 ($k=1.71, d=0.11$)

presenting the minimum SMSE with the corresponding shrinkage value and the maximum balanced accuracy with their respective shrinkage values for each estimator. These results are also depicted in Figure 1 - 3 in Appendices A.3.

According to Table 2 - 4, it is observed that the proposed estimators MLTPE1, MLTPE2, and MLTPE3 demonstrate slightly better prediction performance (with the highest balanced accuracy) than the existing estimators MLE, MAURLE, and LTPE. Moreover, we noticed that the new estimator MLTPE2 exhibits better performance (with the lowest SMSE and highest balanced accuracy) than MLE, MAURLE, LTPE, MLTPE1, and MLTPE3 in almost all situations. An increase in the sample size positively impacts the proposed estimators in terms of SMSE but has a negative effect on balanced accuracy. Furthermore, increasing the degrees of correlation positively affects the proposed estimators in terms of both SMSE and balanced accuracy.

Figures 1 - 3 show the results of the simulation study regarding SMSE and balanced accuracy. When comparing the estimators MLE, MAURLE, LTPE, and MLTPE1, we observe that MAURLE has the smallest SMSE; however, the proposed estimator MLTPE1 exhibits slightly higher balanced accuracy. The second new estimator, MLTPE2, demonstrates the best performance in terms of both SMSE and balanced accuracy. Considering all the estimators, their performance in terms of SMSE can be ordered as follows: MLTPE2, MAURLE, MLTPE3, MLTPE1, and LTPE. In terms of balanced accuracy, MLTPE2 shows the best performance in most cases, while MLTPE3 and MLTPE1 also demonstrate comparatively good performance compared to MAURLE and LTPE.

Table 3: The results of the simulation study when $n = 100$

	Estimators	SMSE	Balanced Accuracy
$\rho = 0.80$	MLE	42.868378	0.513073
	MAURLE	0.617397 ($k = 0.61$)	0.519054 ($k = 0.01$)
	LTPE	0.741937 ($k = 1.91, d = 0.01$)	0.518463 ($k = 2.21, d = 0.01$)
	MLTPE1	0.666742 ($k = 3.21, d = 0.11$)	0.519411 ($k = 2.41, d = 0.01$)
	MLTPE2	0.607717 ($k = 0.31, d = 0.11$)	0.519457 ($k = 2.61, d = 0.21$)
	MLTPE3	0.646961 ($k = 0.21, d = 0.01$)	0.519447 ($k = 0.51, d = 0.01$)
$\rho = 0.85$	MLE	54.337142	0.513864
	MAURLE	0.589061 ($k = 0.61$)	0.520987 ($k = 0.01$)
	LTPE	0.712068 ($k = 1.71, d = 0.01$)	0.521142 ($k = 2.31, d = 0.01$)
	MLTPE1	0.635751 ($k = 2.91, d = 0.11$)	0.521016 ($k = 0.51, d = 0.99$)
	MLTPE2	0.582717 ($k = 0.31, d = 0.21$)	0.521657 ($k = 0.41, d = 0.11$)
	MLTPE3	0.617451 ($k = 0.11, d = 0.01$)	0.521297 ($k = 2.71, d = 0.11$)
$\rho = 0.90$	MLE	79.150907	0.515745
	MAURLE	0.556903 ($k = 0.61$)	0.523131 ($k = 0.01$)
	LTPE	0.676103 ($k = 1.61, d = 0.01$)	0.522414 ($k = 1.41, d = 0.01$)
	MLTPE1	0.594124 ($k = 4.91, d = 0.01$)	0.523567 ($k = 4.81, d = 0.11$)
	MLTPE2	0.554349 ($k = 0.31, d = 0.21$)	0.523503 ($k = 1.51, d = 0.01$)
	MLTPE3	0.579814 ($k = 0.11, d = 0.01$)	0.523573 ($k = 0.51, d = 0.11$)

4. Real data application

Myopia data was used to check the performance of the proposed estimators MLTPE1, MLTPE2, and MLTPE3 with existing estimators, such as MLE, MAURLE, and LTPE. This dataset is about a study of myopia taken from Hosmer Jr *et al.* (2013) and also studied by Asar *et al.* (2017). In this data, 618 subjects who were not myopic when they entered the study were followed up for at least five years, and observations were made on 17 parameters. However, following Asar *et al.* (2017), we focused our analysis on four explanatory variables: spherical equivalent refraction (SPHEQ), axial length (AL), anterior chamber depth (ACD), and vitreous chamber depth (VCD). These variables are continuous and measured on the same scale (mm). We limited our analysis to the first 100 observations. The dependent variable indicates whether a subject has myopia (coded as 1) or not (coded as 0).

The condition number, a measure of multicollinearity, is obtained as 10.4647. This result provides evidence of moderate multicollinearity among the explanatory variables in the dataset. We examined the explanatory variables with positive and negative correlations to one another and discovered that the correlation between the variables AL and VCD is very strong (0.9322). The correlation matrix of the explanatory variables, SPHEQ, AL, ACD, and VCD, is presented in Table 6 in Appendices A.1. The corresponding Variance Inflation Factor (VIF) values for these variables are 1.1322, 24.9815, 3.0186, and 22.0122. The VIF measures the extent to which the variance of the coefficient estimates is inflated due to multicollinearity among the predictor variables. As a result, variables SPHEQ and ACD have $1 < \text{VIF} < 5$, indicating moderate collinearity, while variables AL and VCD have

Table 4: The results of the simulation study when $n = 200$

	Estimators	SMSE	Balanced Accuracy
$\rho = 0.80$	MLE	38.785726	0.507707
	MAURLE	0.600115 ($k = 0.71$)	0.512734 ($k = 0.01$)
	LTPE	0.727127 ($k = 1.91, d = 0.01$)	0.511634 ($k = 3.91, d = 0.01$)
	MLTPE1	0.648386 ($k = 3.41, d = 0.11$)	0.512373 ($k = 3.51, d = 0.11$)
	MLTPE2	0.580982 ($k = 0.21, d = 0.11$)	0.513346 ($k = 0.41, d = 0.11$)
	MLTPE3	0.626576 ($k = 0.21, d = 0.01$)	0.512861 ($k = 1.01, d = 0.01$)
$\rho = 0.85$	MLE	50.044391	0.508435
	MAURLE	0.565265 ($k = 0.61$)	0.513212 ($k = 0.01$)
	LTPE	0.693498 ($k = 1.81, d = 0.01$)	0.513712 ($k = 1.51, d = 0.01$)
	MLTPE1	0.612667 ($k = 3.01, d = 0.11$)	0.513775 ($k = 1.91, d = 0.01$)
	MLTPE2	0.557344 ($k = 0.31, d = 0.21$)	0.514450 ($k = 0.61, d = 0.01$)
	MLTPE3	0.593446 ($k = 0.21, d = 0.01$)	0.513737 ($k = 1.11, d = 0.91$)
$\rho = 0.90$	MLE	72.871783	0.510181
	MAURLE	0.539267 ($k = 0.61$)	0.514214 ($k = 0.01$)
	LTPE	0.661072 ($k = 1.61, d = 0.01$)	0.514231 ($k = 0.41, d = 0.01$)
	MLTPE1	0.579958 ($k = 4.91, d = 0.01$)	0.514413 ($k = 3.71, d = 0.11$)
	MLTPE2	0.535675 ($k = 0.31, d = 0.21$)	0.515615 ($k = 0.41, d = 0.01$)
	MLTPE3	0.561105 ($k = 0.11, d = 0.01$)	0.514474 ($k = 1.11, d = 0.11$)

VIF values greater than 5, suggesting severe collinearity exists. The results of the real data application are presented in Table (5).

Table 5: The results of the real data application

	Estimators	SMSE	Balanced Accuracy
	MLE	195.6843	0.746096
	MAURLE	28.3958 ($k = 0.11$)	0.757534 ($k = 0.01$)
	LTPE	25.2972 ($k = 0.11, d = 0.01$)	0.757534 ($k = 0.01, d = 0.01$)
	MLTPE1	25.4710 ($k = 0.11, d = 0.99$)	0.751978 ($k = 0.01, d = 0.01$)
	MLTPE2	25.1819 ($k = 0.01, d = 0.61$)	0.763382 ($k = 0.01, d = 0.01$)
	MLTPE3	25.5134 ($k = 0.11, d = 0.99$)	0.751978 ($k = 0.01, d = 0.01$)

From Table 5, it is evident that the proposed estimator MLTPE2 outperforms other estimators in terms of both SMSE and balanced accuracy. Note that similar results were also observed in the simulation study.

5. Concluding remarks

Based on the findings of this study, the Modified Logistic Two-Parameter Estimators: MLTPE1, MLTPE2, and MLTPE3 demonstrated slightly better performance in terms of balanced accuracy compared to MLE, MAURLE, and LTPE. Meanwhile, the MLTPE2

significantly outperformed others in terms of SMSE and prediction accuracy. Additionally, estimators performing better in terms of SMSE do not necessarily provide superior prediction results. Therefore, when comparing the performance of logistic regression, metrics that focus on classification accuracy are more appropriate.

Acknowledgements

We express our sincere gratitude to the editorial team, especially the Chair Editor, for their valuable insights and constructive feedback, and to the anonymous reviewers for their thorough evaluation and thoughtful comments, all of which greatly improved the quality of this manuscript.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Ahmad, S. (2020). Another proposal about the new two-parameter estimator for linear regression model with correlated regressors. *Communications in Statistics - Simulation and Computation*, **51**, 1–19.
- Alheety, M., Månsson, K., and Golam Kibria, B. (2021). A new kind of stochastic restricted biased estimator for logistic regression model. *Journal of Applied Statistics*, **48**, 1559–1578.
- Asar, Y., Arashi, M., and Wu, J. (2017). Restricted ridge estimator in the logistic regression model. *Communications in Statistics-Simulation and Computation*, **46**, 6538–6544.
- Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X. (2013). *Applied Logistic Regression*. John Wiley & Sons.
- Huang, J. (2012). A simulation research on a biased estimator in logistic regression model. In *International Symposium on Intelligence Computation and Applications*, pages 389–395. Springer.
- McDonald, G. C. and Galarneau, D. I. (1975). A monte carlo evaluation of some ridge-type estimators. *Journal of the American Statistical Association*, **70**, 407–416.
- Rao, C. R., Shalabh, Toutenburg, H., and Heumann, C. (2008). *Linear Model and Generalizations*. Springer, Berlin.
- Rao, C. R. and Toutenburg, H. (1995). *Linear Models: Least Squares and Alternatives, Second Edition*. Springer, New York.
- Trenkler, G. and Toutenburg, H. (1990). Mean squared error matrix comparisons between biased estimators—an overview of recent results. *Statistical Papers*, **31**, 165–179.
- Varathan, N. (2022). An improved ridge type estimator for logistic regression. *Statistics in Transition. New Series*, **23**, 113–126.

A. Appendices

A.1. Correlation matrix

Table 6: The correlation matrix of the explanatory variables of real data

Variables	SPHEQ	AL	ACD	VCD
SPHEQ	1.0000	-0.2865	-0.2143	-0.2643
AL	-0.2865	1.0000	0.4766	0.9322
ACD	-0.2143	0.4766	1.0000	0.2007
VCD	-0.2643	0.9322	0.2007	1.0000

A.2. Lemmas

Lemma 1: (Rao and Toutenburg, 1995) Let A be positive definite and B be a regular matrix, then $B'AB > 0$.

Lemma 2: (Rao *et al.*, 2008) Let the two $n \times n$ matrices $M > 0$, $N \geq 0$, then $M > N$ if and only if $\lambda_{\max}(NM^{-1}) < 1$.

Lemma 3: (Trenkler and Toutenburg, 1990) Let $\hat{\beta}_j = A_j y, j = 1, 2$ be two competing homogenous linear estimators of β . Suppose that $D = Cov(\hat{\beta}_1) - Cov(\hat{\beta}_2) > 0$; where $Cov(\hat{\beta}_j), j = 1, 2$ denotes the covaraince matrix of $\hat{\beta}_j$. Then $\Delta(\hat{\beta}_1, \hat{\beta}_2) = MSEM(\hat{\beta}_1) - MSEM(\hat{\beta}_2) \geq 0$ if and only if $d_2'(D + d_1'd_1)d_2 \leq 1$, where $MSEM(\hat{\beta}_j), d_j; j = 1, 2$ denote the Mean Square Error Matrix and bias vector of $\hat{\beta}_j$, respectively.

A.3. Figures

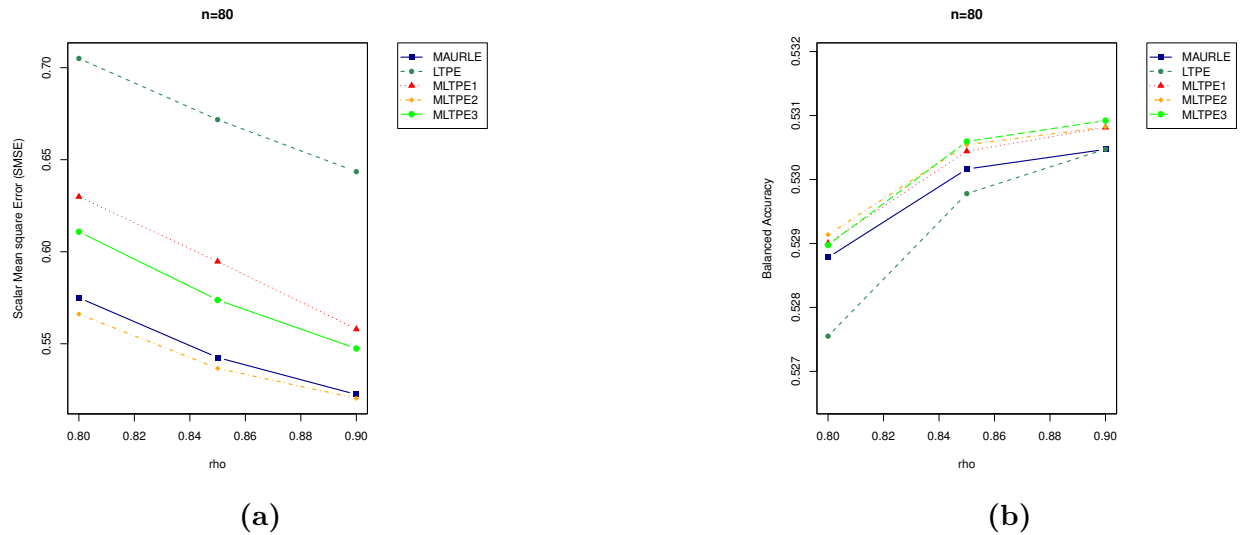


Figure 1: The result of the simulation study when $n = 80$

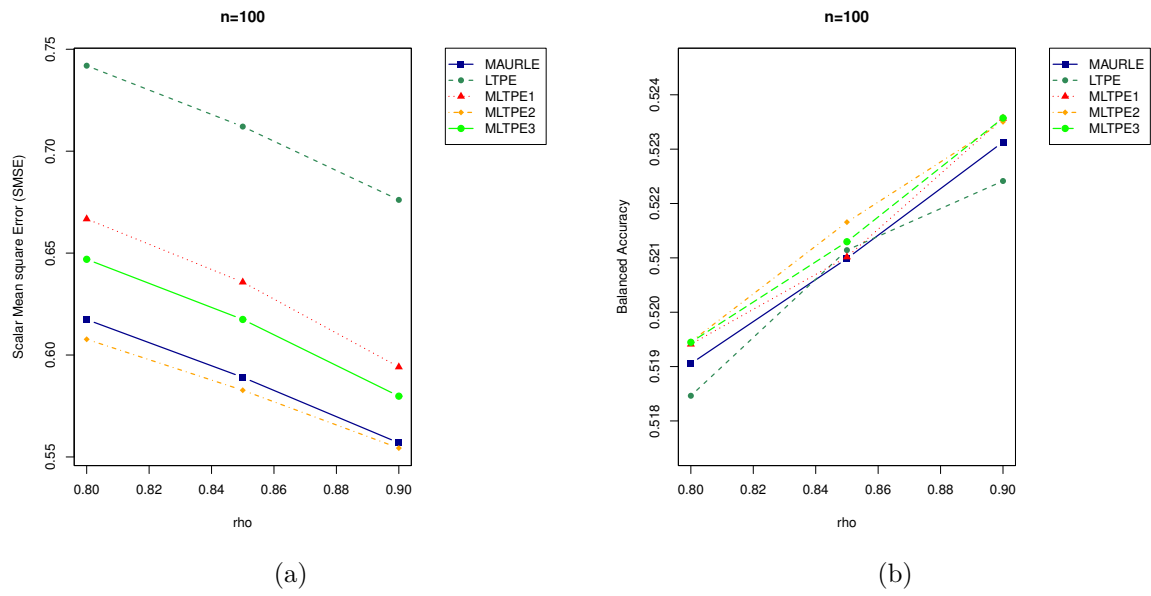


Figure 2: The result of the simulation study when $n = 100$

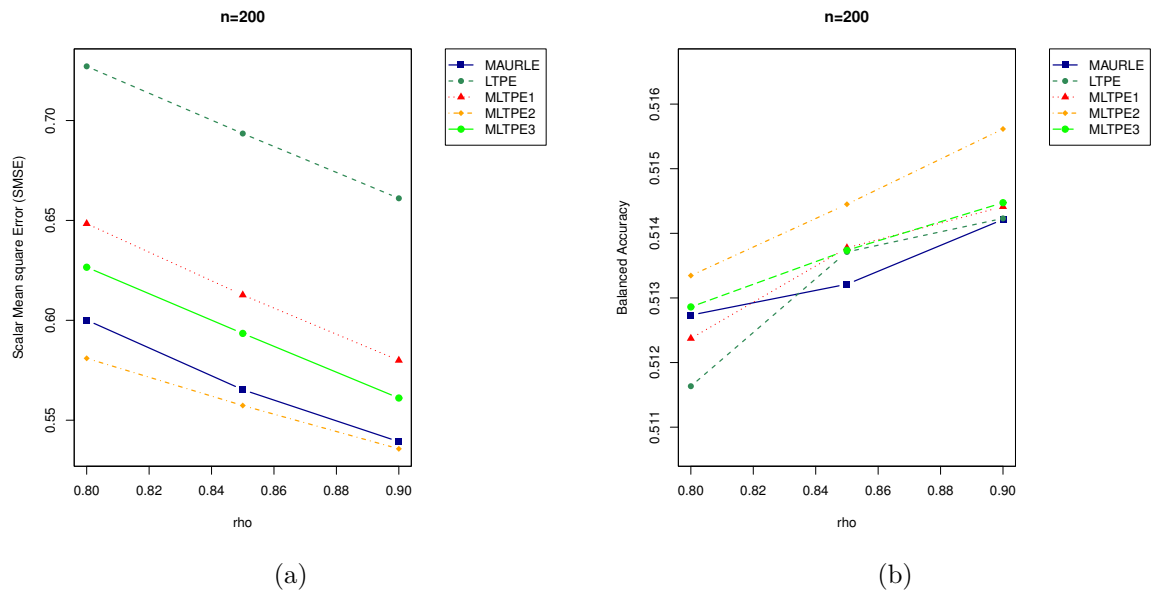
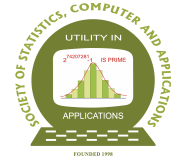


Figure 3: The result of the simulation study when $n = 200$



Wrapped Generalized Lindley Distribution with Applications to Directional Data

Imliyangba and Bhanita Das

Department of Statistics, North Eastern Hill University, Shillong-793022

Received: 27 May 2024; Revised: 06 October 2024; Accepted: 08 October 2024

Abstract

The standard statistical techniques are fit to the linear data, where data are simply presented on a straight line. However, in many diverse scientific fields, the measurements are referred to as directions. Since a direction has no magnitude, these directions can be conveniently represented as points on the circumference of a unit circle centered at the origin or as a unit vector in the plane. In this article, the wrapped generalized Lindley distribution, a new class of circular distribution based on the wrapping method was introduced. Furthermore, the trigonometric moments and fundamental properties of the new circular distribution were studied. The method of maximum likelihood estimation was used to estimate the values of the parameter. A simulation study was performed to illustrate the proposed distribution in modeling directional data. The flexibility of the proposed model was shown by analyzing real-life data and its performance was compared with other families of circular Lindley distributions.

Key words: Wrapped distributions; Lindley distribution; Trigonometric moments; Maximum likelihood estimation; Angular data analysis.

1. Introduction

In many scientific experiments such as geological (Rao and Sengupta, 1972), medical Jammalamadaka *et al.* (1986), meteorological (Fisher, 1995), biological sciences (Jammalamadaka and Sengupta, 2001), *etc.*, the data have been observed exhibiting periodic or cyclic behaviors. The standard statistical techniques are not appropriate to handle these situations where the data are circular (directional). Therefore, to deal with such data and to perform statistical analysis, many circular distributions have been introduced from the existing linear distributions by adopting a variety of techniques namely wrapping, inverse stereographic projections, rising sun function, *etc.*

Perhaps the most popular technique to analyze circular data is the idea of wrapping a linear distribution around the unit circle which gives rise to a multitude of wrapped circular distributions. PL (1939) first introduced the wrapped distribution, and since then several wrapped versions of common probability distributions on the real line have been studied

in the literature, including wrapped normal (Mardia and Jupp, 2000), wrapped Cauchy (Jammalamadaka and Sengupta, 2001), and wrapped stable distributions (Gatto and Jammalamadaka, 2003).

Further, Jammalamadaka and Kozubowski (2004) discussed the wrapped exponential and wrapped Laplace distributions and presented the explicit forms for their densities and distribution functions, as well as their trigonometric moments and related parameters. Pewsey *et al.* (2007) considered the three-parameter family of symmetric unimodal distributions obtained by wrapping the location-scale extension of Student's t distribution onto the unit circle. Coelho (2007) obtained an expression for the probability density function of the wrapped gamma distribution and illustrated both integer and non-integer shape parameters, as a mixture of truncated gamma distributions. Pewsey (2008) also considered the use of the wrapped stable family as a model for unimodal circular data. Sarma *et al.* (2011) presented certain population characteristics of the wrapped Lognormal and the wrapped Weibull distributions. In another work, Roy and Adnan (2012), Jacob and Jayakumar (2013), and Adnan and Roy (2014) explored wrapped generalized Gompertz distribution, wrapped geometric distribution and wrapped variance gamma distribution, respectively, and discussed their applications to circular data. Joshi and Jose (2018) introduced a new circular distribution called the wrapped Lindley distribution and derived expressions for characteristic function, trigonometric moments, coefficients of skewness, and kurtosis with an application to real-life datasets.

In the recent past, Shanker *et al.* (2013) introduced the generalized Lindley (GL) distribution as a special case of the one-parameter Lindley distribution (Lindley, 1958). Over the years, the GL distribution has emerged as a new lifetime model that is more applicable than the Lindley distribution. Several researchers have introduced many modified versions of the GL distribution and shown its applicability in diverse data environments through parameterization which is linear in nature. However, most works on GL distribution are limited to random variables with an infinite set of possible values. Therefore, the authors were motivated to apply the concept of circular statistics to the GL distribution by reducing its modulo 2π with the help of the wrapping technique. The main objective of this article was to propose a new generalized Lindley distribution by wrapping the density along a unit circle, called the wrapped generalized Lindley (WGL) distribution. This article also aimed to study the properties of the proposed distribution, and explore its utility as a circular model.

The rest of the paper is organized as follows. Section 2 describes the wrapping technique and introduces a new WGL distribution. Section 3 graphically illustrates the behavior of the newly developed distribution. The derivations of the characteristic function and the trigonometric moments as well as the associated parameters are discussed in Section 4 and Section 5, respectively. Section 6 examines the estimation of the parameters using the maximum likelihood method. A simulation study is discussed in Section 7, to show the consistency of the estimator. In Section 8, we fit the proposed distribution to five circular datasets and compare its performance with other competing wrapped distributions. Finally, conclusions are made in Section 9.

2. Definition and derivation

2.1. Wrapped distribution

Any linear random variable (r.v.) X on the real line may be transformed to a circular random variable by reducing its modulo 2π *i.e.*, we define

$$\theta = X(\text{mod } 2\pi)$$

This operation corresponds to taking the real line and wrapping it around the circle of unit radius, accumulating probability over all the overlapping points $x = \theta, \theta \pm 2\pi, \theta \pm 4\pi, \dots$. This is a many-to-one mapping so that if $g(\theta)$ represents the circular density and $f(x)$ the density of the real-valued r.v., we have

$$g(\theta) = \sum_{m=-\infty}^{\infty} f(\theta + 2\pi m), 0 \leq \theta < 2\pi \quad (1)$$

By this technique, both discrete and continuous wrapped distributions may be constructed.

2.2. Wrapped generalized Lindley distribution

The probability density function (*pdf*) of the generalized Lindley distribution given by Shanker *et al.* (2013) is,

$$f(x; \lambda, \alpha) = \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} (1 + \alpha x) e^{-\frac{\lambda}{\alpha} x}; \quad x > 0, \lambda > 0, \alpha > 0 \quad (2)$$

Using equations (1) and (2), the *pdf* of the WGL distribution is derived as given below,

$$\begin{aligned} g(\theta) &= \sum_{m=0}^{\infty} f(\theta + 2\pi m) \\ &= \sum_{m=0}^{\infty} \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} \{1 + \alpha(\theta + 2\pi m)\} e^{-\frac{\lambda}{\alpha}(\theta + 2\pi m)} \\ &= \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} e^{-\frac{\lambda}{\alpha}\theta} \left\{ \frac{(1 + \alpha\theta)}{1 - e^{-2\pi\frac{\lambda}{\alpha}}} + \frac{2\pi\alpha e^{-2\pi\frac{\lambda}{\alpha}}}{(1 - e^{-2\pi\frac{\lambda}{\alpha}})^2} \right\} \end{aligned} \quad (3)$$

The cumulative distribution function (*cdf*) of the wrapped distribution is given by

$$G(\theta) = \sum_{m=0}^{\infty} F(\theta + 2\pi m) - F(2\pi m)$$

Therefore, the *cdf* of the WGL distribution is obtained as

$$\begin{aligned}
 G(\theta) &= \sum_{m=0}^{\infty} \left[\left\{ 1 - \frac{\alpha(\lambda + \alpha^2) + \lambda\alpha^2(\theta + 2\pi m)}{\alpha(\lambda + \alpha^2)} e^{-\frac{\lambda}{\alpha}(\theta + 2\pi m)} \right\} - \right. \\
 &\quad \left. \left\{ 1 - \frac{\alpha(\lambda + \alpha^2) + \lambda\alpha^2 2\pi m}{\alpha(\lambda + \alpha^2)} e^{-\frac{\lambda}{\alpha}(2\pi m)} \right\} \right] \\
 &= \sum_{m=0}^{\infty} \frac{1}{\alpha(\lambda + \alpha^2)} \left[\alpha(\lambda + \alpha^2) e^{-\frac{\lambda}{\alpha}(2\pi m)} + \lambda\alpha^2 2\pi m e^{-\frac{\lambda}{\alpha}(2\pi m)} - \right. \\
 &\quad \left. \left\{ \alpha(\lambda + \alpha^2) + \lambda\alpha^2(\theta + 2\pi m) e^{-\frac{\lambda}{\alpha}(\theta + 2\pi m)} \right\} \right] \\
 &= \frac{1}{\alpha(\lambda + \alpha^2)} \left[\alpha(\lambda + \alpha^2) \sum_{m=0}^{\infty} e^{-\frac{\lambda}{\alpha}(2\pi m)} + \lambda\alpha^2 2\pi \sum_{m=0}^{\infty} m e^{-\frac{\lambda}{\alpha}(2\pi m)} - \right. \\
 &\quad \left. \left\{ \alpha(\lambda + \alpha^2) e^{-\frac{\lambda}{\alpha}\theta} \sum_{m=0}^{\infty} e^{-\frac{\lambda}{\alpha}(2\pi m)} + \lambda\alpha^2 e^{-\frac{\lambda}{\alpha}\theta} \sum_{m=0}^{\infty} (\theta + 2\pi m) e^{-\frac{\lambda}{\alpha}(2\pi m)} \right\} \right] \\
 &= \left(\frac{1}{1 - e^{-2\pi\frac{\lambda}{\alpha}}} \right) \left\{ 1 - e^{-\frac{\lambda}{\alpha}\theta} - \frac{(\lambda\alpha^2\theta e^{-\frac{\lambda}{\alpha}\theta})}{\alpha(\lambda + \alpha^2)} \right\} + \frac{2\pi\lambda\alpha^2}{\alpha(\lambda + \alpha^2)} (1 - e^{-\frac{\lambda}{\alpha}\theta}) \frac{e^{-2\pi\frac{\lambda}{\alpha}}}{(1 - e^{-2\pi\frac{\lambda}{\alpha}})^2} \quad (4)
 \end{aligned}$$

3. Graphical representations of WGL distribution

The following figures provide a linear representation of WGL distribution for some special cases.

3.1. Linear representation of WGL distribution

Case 1: When $\lambda < \alpha$, **Case 2:** When $\lambda > \alpha$, and **Case 3:** When $\lambda = \alpha$.

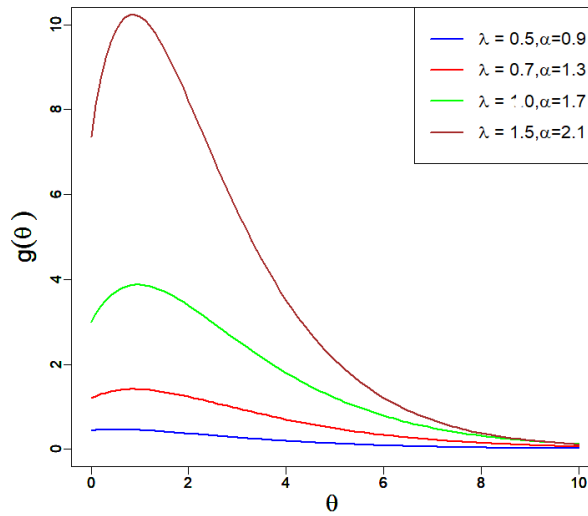


Figure 1: Graphical representation of WGL distribution when $\lambda < \alpha$

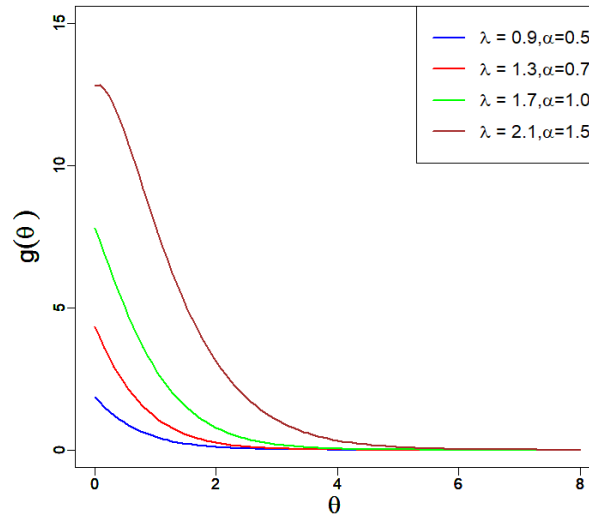


Figure 2: Graphical representation of WGL distribution when $\lambda > \alpha$

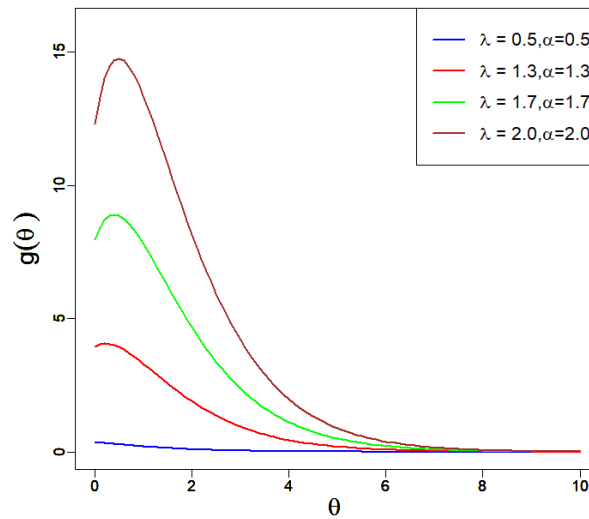


Figure 3: Graphical representation of WGL distribution when $\lambda = \alpha$

From the three graphs of WGL distribution in Figure 1, Figure 2 and Figure 3 we can make the following observations:

- Figure 1, Figure 2 and Figure 3 show some of the possible shapes of the *pdf* of WGL distribution for different values of the parameters λ and α .
- Figure 1 represents the *pdf* of WGL distribution when $\lambda < \alpha$. As the values of both the parameter increase, we get a positively high kurtosis with low skewness.
- When $\lambda > \alpha$, the curve of WGL distribution shows an exponentially decreasing behav-

ior with positively low skewness. Figure 2 also justifies that the proposed distribution inherits the properties of Exponential distribution where GL distribution can be shown as a mixture of Exponential ($\frac{\lambda}{\alpha}$) and Gamma distribution ($2, \frac{\lambda}{\alpha}$) (Shanker *et al.*, 2013).

- Figure 3 represents the shape of WGL distribution when $\lambda = \alpha$ which shows a positively high kurtosis and positively low skewness with more peakedness and lesser tail as compared to Figure 1.

The above observations (Figure 1, Figure 2 and Figure 3) justify the graphical representation of the proposed distribution. Moreover, we can conclude that the proposed distribution can be used to model the datasets with decreasing and increasing-decreasing behaviors.

3.2. Circular representation of WGL distribution

Figure 4 and Figure 5 display the circular representation of WGL distribution with different values of the parameter α , keeping $\lambda = 1.5$ in Figure 4 and with the changing values of the parameter λ , keeping α constant in Figure 5.

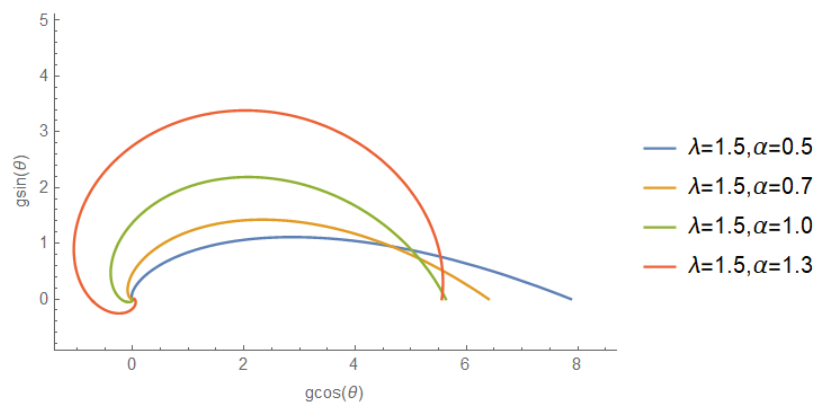


Figure 4: Circular representation of WGL distribution when λ is constant

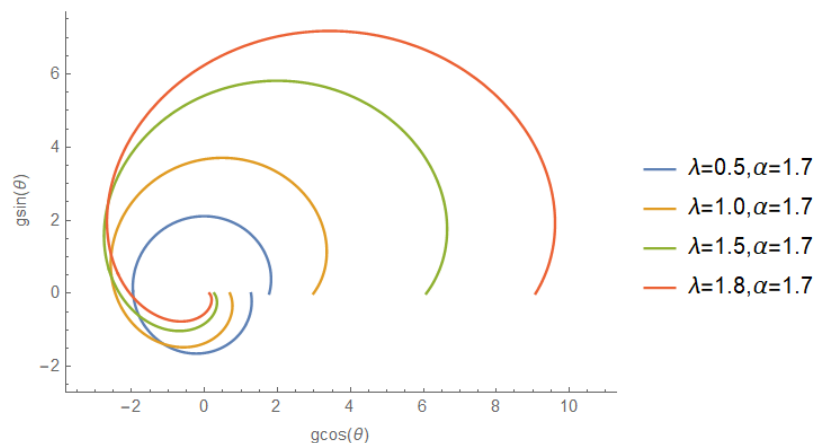


Figure 5: Circular representation of WGL distribution when α is constant

4. Characteristic function

According to Jammalamadaka and Sengupta (2001), the trigonometric moments of order p for a wrapped circular distribution corresponds to the value of the characteristic function of the unwrapped r.v. X , say $\varphi_x(t)$ at the integer value, *i.e.*,

$$\varphi(p) = \varphi_x(t)$$

The characteristic function of GL distribution is given by,

$$\begin{aligned}\varphi_x(t) &= E(e^{itx}) \\ &= \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} \int_0^\infty (1 + \alpha x) e^{-\frac{\lambda}{\alpha}x} e^{itx} dx \\ &= \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} \left\{ \frac{1}{\frac{\lambda}{\alpha} - it} - \frac{\alpha}{(\frac{\lambda}{\alpha} - it)^2} \right\} \\ &= \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} \left[\frac{\frac{\lambda}{\alpha} - \alpha - it}{(\frac{\lambda}{\alpha} - it)^2} \right]\end{aligned}$$

Therefore, the characteristic function of WGL distribution is given by,

$$\varphi(p) = \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} \left[\frac{\frac{\lambda}{\alpha} - \alpha - ip}{(\frac{\lambda}{\alpha} - ip)^2} \right] \quad (5)$$

where $i = (-1)^{\frac{1}{2}}$ and $p = \pm 1, \pm 2, \dots$

From Roy and Adnan (2012), $\forall a, b, r \in R^+$,

$$(a - ib)^{-r} = (a^2 + b^2)^{-\frac{r}{2}} \exp \left\{ ir \arctan \left(\frac{b}{a} \right) \right\}$$

The following expressions are obtained

$$\begin{aligned}\left(\frac{\lambda}{\alpha} - ip \right)^{-2} &= \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + p^2 \right\}^{-1} \exp \left\{ 2i \arctan p \left(\frac{\alpha}{\lambda} \right) \right\} \\ \left(\frac{\lambda}{\alpha} - \alpha - ip \right)^1 &= \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + p^2 \right\}^{-\frac{1}{2}} \exp \left\{ i \arctan \left(\frac{p}{\frac{\lambda}{\alpha} - \alpha} \right) \right\}\end{aligned}$$

About the above expressions, equation (5) may be finally written as,

$$\varphi(p) = \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + p^2 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + p^2 \right\}^{-1}} \exp \left\{ 2i \arctan p \left(\frac{\alpha}{\lambda} \right) - i \arctan \left(\frac{p}{\frac{\lambda}{\alpha} - \alpha} \right) \right\} \quad (6)$$

and so,

$$\varphi(p) = \rho_p e^{i\mu_p} \quad (7)$$

Comparing equation (7) to (6), we get

$$\begin{aligned}\rho_p &= \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + p^2 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + p^2 \right\}^{-1}} \\ \mu_p &= 2 \arctan p \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{p}{\frac{\lambda}{\alpha} - \alpha} \right) \\ &= \arctan p \left\{ 2 \left(\frac{\alpha}{\lambda} \right) - \left(\frac{\alpha}{\lambda - \alpha^2} \right) \right\}\end{aligned}$$

5. Trigonometric moments and related parameters

According to the definition of the trigonometric moment,

$$\Phi_p = \alpha_p + i\beta_p; \quad p = \pm 1, \pm 2, \dots$$

Therefore, the non-central trigonometric moments of the respective distribution are defined as,

$$\alpha_p = \rho_p \cos \mu_p \quad \text{and} \quad \beta_p = \rho_p \sin \mu_p$$

So, we have

$$\begin{aligned}\alpha_p &= \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + p^2 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + p^2 \right\}^{-1}} \cos \left\{ 2 \arctan p \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{p}{\frac{\lambda}{\alpha} - \alpha} \right) \right\} \\ \beta_p &= \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + p^2 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + p^2 \right\}^{-1}} \sin \left\{ 2 \arctan p \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{p}{\frac{\lambda}{\alpha} - \alpha} \right) \right\}\end{aligned}$$

Now, the central trigonometric moments are

$$\bar{\alpha}_p = \rho_p \cos(\mu_p - p\mu_1) \quad \text{and} \quad \bar{\beta}_p = \rho_p \sin(\mu_p - p\mu_1)$$

Thus, the central trigonometric moments of WGL distribution will be

$$\begin{aligned}\bar{\alpha}_p &= \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + p^2 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + p^2 \right\}^{-1}} \cos \left[2 \arctan p \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{p}{\frac{\lambda}{\alpha} - \alpha} \right) - \right. \\ &\quad \left. \left\{ 2 \arctan \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{1}{\frac{\lambda}{\alpha} - \alpha} \right) \right\} \right]\end{aligned}$$

$$\bar{\beta}_p = \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + p^2 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + p^2 \right\}^{-1}} \sin \left[2 \arctan p \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{p}{\frac{\lambda}{\alpha} - \alpha} \right) - \right. \\ \left. \left\{ 2 \arctan \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{1}{\frac{\lambda}{\alpha} - \alpha} \right) \right\} \right]$$

Since we have

$$\mu_p = \arctan p \left\{ 2 \left(\frac{\alpha}{\lambda} \right) - \left(\frac{\alpha}{\lambda - \alpha^2} \right) \right\}$$

For $p = 1$

$$\mu_1 = \arctan \left\{ 2 \left(\frac{\alpha}{\lambda} \right) - \left(\frac{\alpha}{\lambda - \alpha^2} \right) \right\}$$

Moreover, the resultant length is $\rho = \rho_1$

$$\rho = \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 1 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 1 \right\}^{-1}}$$

The mean direction is,

$$\mu = \mu_1 \\ \mu_1 = \arctan \left\{ 2 \left(\frac{\alpha}{\lambda} \right) - \left(\frac{\alpha}{\lambda - \alpha^2} \right) \right\}$$

The mean direction gives information about the mean of the distribution as an analogy of the mean in the linear distributions and the resultant length is a measure of dispersion around the mean which corresponds to the usual standard deviation or variance in linear distributions. The circular variance is,

$$V_0 = 1 - \rho \\ V_0 = 1 - \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 1 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 1 \right\}^{-1}}$$

The circular standard deviation is,

$$\sigma_0 = \sqrt{-2 \log(1 - V_0)} \\ = \sqrt{-2 \log \left(1 - 1 + \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 1 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 1 \right\}^{-1}} \right)} \\ = \sqrt{-2 \log \left(\frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 1 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 1 \right\}^{-1}} \right)}$$

The skewness $\xi_1^0 = \frac{\bar{\beta}_2}{V_0^{\frac{3}{2}}}$ is given by,

$$\xi_1^0 = \frac{\frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 4 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 4 \right\}^{-1}} \sin \left[\arctan 2 \left\{ \left(\frac{\alpha}{\lambda} \right) - \left(\frac{\alpha}{\lambda - \alpha^2} \right) \right\} - 2 \arctan \left\{ \left(\frac{\alpha}{\lambda} \right) - \left(\frac{\alpha}{\lambda - \alpha^2} \right) \right\} \right]}{\left\{ 1 - \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 1 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 1 \right\}^{-1}} \right\}^{\frac{3}{2}}}$$

And the kurtosis is given by

$$\xi_2^0 = \frac{\bar{\alpha}_2 - (1 - V_0)^4}{V_0^2}$$

where

$$\begin{aligned} \bar{\alpha}_2 = & \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 4 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 4 \right\}^{-1}} \cos \left[2 \arctan 2 \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{2}{\frac{\lambda}{\alpha} - \alpha} \right) - \right. \\ & \left. \left\{ 2 \arctan \left(\frac{\alpha}{\lambda} \right) - \arctan \left(\frac{1}{\frac{\lambda}{\alpha} - \alpha} \right) \right\} \right] \\ V_0 = & 1 - \frac{\lambda^2 \left\{ \left(\frac{\lambda}{\alpha} - \alpha \right)^2 + 1 \right\}^{-\frac{1}{2}}}{\alpha(\lambda + \alpha^2) \left\{ \left(\frac{\lambda}{\alpha} \right)^2 + 1 \right\}^{-1}} \end{aligned}$$

The values of the various descriptive measures for some particular values of α and λ are summarized in Table 1.

From Table 1, we can make the following remarks:

- The mean direction increases with the change in λ keeping α constant as well as with different values of α keeping λ as constant. However, the resultant length decreases with the change in α and λ .
- The mean direction approaches 1 as α increases.
- The circular variance increases with the increase in α and λ .

6. Maximum likelihood estimation

The method of maximum likelihood estimation is one of the most important techniques in statistics and econometrics for estimating the parameters. Let $\theta_1, \theta_2, \dots, \theta_n$ be a random sample of size n from WGL distribution. Then, the likelihood function is given by,

$$L = \left\{ \frac{\lambda^2}{\alpha(\lambda + \alpha^2)} \right\}^n e^{-\frac{\lambda}{\alpha} \sum_{i=1}^n \theta_i} \sum_{i=1}^n \left\{ \frac{1 + \alpha \theta_i}{1 - e^{-\frac{\lambda}{\alpha} 2\pi}} + \frac{2\alpha \pi e^{-\frac{\lambda}{\alpha} 2\pi}}{(1 - e^{-\frac{\lambda}{\alpha} 2\pi})^2} \right\}$$

Table 1: Values of different characteristics of the WGL distribution for various values of α and λ

α		0.5	0.5	0.5	1	1.5	2.5
λ		0.5	3.5	4.5	0.5	0.5	0.5
Mean direction	μ	0.288349	2.975447	1.799375	0.506719	0.523067	0.718435
Resultant length	ρ	0.989381	0.466179	0.182925	0.982512	0.968803	0.966841
Trigonometric moments	α_1	0.948534	-0.459759	-0.041449	0.859051	0.839266	0.727874
	α_2	0.802768	0.028768	0.031769	0.489695	0.437383	0.100793
	β_1	0.281351	0.077097	0.178167	0.476824	0.483955	0.636382
	β_2	0.522730	0.096061	-0.022027	0.792252	0.761552	0.867437
Central trigonometric moments	$\bar{\alpha}_1$	0.989381	0.466179	0.182925	0.982512	0.968803	0.966841
	$\bar{\alpha}_2$	0.957958	-0.004141	-0.018784	0.931371	0.878212	0.873128
	$\bar{\beta}_1$	0	0	0	0	0	0
	$\bar{\beta}_2$	0.000469	0.100191	0.033788	0.003477	0.002926	0.015935
Circular variance	V_0	0.010618	0.533820	0.817074	0.035918	0.064185	0.068678
Skewness	ξ_1^0	0.029535	0.146971	0.036656	1.503285	0.063031	0.323080
Kurtosis	ξ_2^0	-0.000245	-0.236381	-0.594824	-0.000510	-0.00290	-0.000740

And the log-likelihood function is

$$\begin{aligned}
 \text{Log}L = 2n \log \log \lambda - n \log \log \left\{ \alpha(\lambda + \alpha^2) \right\} - \frac{\lambda}{\alpha} \sum_{i=1}^n \theta_i + \\
 \sum_{i=1}^n \log \left\{ \frac{1 + \alpha \theta_i}{1 - e^{-\frac{\lambda}{\alpha} 2\pi}} + \frac{2\alpha \pi e^{-\frac{\lambda}{\alpha} 2\pi}}{(1 - e^{-\frac{\lambda}{\alpha} 2\pi})^2} \right\} \quad (8)
 \end{aligned}$$

The MLE of the parameters is computed by solving the maximum likelihood equations

$$\frac{\partial}{\partial \lambda} \log L = 0, \text{ and } \frac{\partial}{\partial \alpha} \log L = 0 \quad (9)$$

Since the maximum likelihood equations cannot be solved analytically, therefore, a numerical technique is to be employed to get a solution for λ and α . We use statistical packages in R to get the maximum likelihood estimator (MLE) of the unknown parameters which is explained through simulation study.

7. Simulation study

A simulation study was performed to obtain the estimates of unknown parameters *i.e.*, λ and α . Further, a random sample of different sizes ($n = 25, 50, 100, 200, 400, 500, 600$) was generated for different values of λ and α , and replicated the program $N=1000$ times to get the MLE of λ and α . Given below is the algorithm for generating data from WGL distribution.

Step 1: Generate a random variable, say u , from $U(0, 1)$.

Step 2: Equating the *cdf* given in equation (4) to u and solving it *w.r.t* θ , we get the WGL circular random variable.

Step 3: Substituting the WGL circular random variable, generated in step 2, in equation (9) and maximizing this *w.r.t* θ by using the maxLik package in R, we get the MLE of λ and α .

Step 4: To calculate the average bias and mean square error (MSE) of $\hat{\lambda}$ and $\hat{\alpha}$, we use the following formulae. Let λ^* and α^* be the true values of parameters λ and α , respectively. Then, the bias and MSE of $\hat{\lambda}$ and $\hat{\alpha}$ from true values of parameter λ^* and α^* are defined as:

$$\begin{aligned} bias(\hat{\lambda}) &= \frac{1}{N} \sum_{i=1}^N (\hat{\lambda} - \lambda^*), & MSE(\hat{\lambda}) &= \frac{1}{N} \sum_{i=1}^N (\hat{\lambda} - \lambda^*)^2 \\ bias(\hat{\alpha}) &= \frac{1}{N} \sum_{i=1}^N (\hat{\alpha} - \alpha^*), & MSE(\hat{\alpha}) &= \frac{1}{N} \sum_{i=1}^N (\hat{\alpha} - \alpha^*)^2 \end{aligned}$$

Where N is the number of replications.

For $\lambda = 4$ and $\alpha = 1.6$, the estimated values of λ and α , and average values of bias and MSE for $\hat{\lambda}$ and $\hat{\alpha}$, are obtained in Table 2.

Table 2: Estimated values of λ and α , and average values of bias and MSE for $\hat{\lambda}$ and $\hat{\alpha}$

n	$\hat{\lambda}$	$\hat{\alpha}$	$\hat{\lambda}$		$\hat{\alpha}$	
			Bias	MSE	Bias	MSE
25	4.9460	2.1640	0.037840	0.035797	0.022560	0.012724
50	4.8430	1.9960	0.016860	0.014213	0.007920	0.079680
100	4.7491	1.9829	0.007491	0.005612	0.003829	0.039319
200	4.6653	1.9340	0.003327	0.002213	0.001670	0.018702
400	4.5750	1.8663	0.001438	0.000827	0.000666	0.008708
500	4.3006	1.8241	0.000601	0.000181	0.000448	0.006655
600	4.0895	1.6868	0.000149	0.000013	0.000145	0.004742

From Table 2, it is observed that, as the sample size increases, the estimated values of the parameters approach very close to the true values of parameters used in the simulation. Moreover, we can also see that the bias and MSE values for the estimated parameters decrease and tend to zero as the sample size increases. This shows the adequacy of the estimation technique.

8. Applications to circular datasets

One of the most important ways to justify the newly proposed distribution is the applicability of the distribution in real life. Therefore, to show the flexibility of the WGL distribution, the wrapped Lindley (WL), wrapped two-parameter Lindley (WTPL), wrapped exponential (WE), and the wrapped generalized Lindley (WGL) distributions were fitted to five circular datasets given below. Various statistics like Log-likelihood, AIC, AICC, and BIC were calculated and the results are shown in Table 3, Table 4, Table 5, Table 6, and Table 7.

First dataset: The application of the proposed distribution is illustrated by fitting it to a real-life dataset. The dataset consists of orientations of 76 turtles after laying eggs (PL, 1939) which are given in direction (in degrees) clockwise from North.

8, 9, 13, 13, 14, 18, 22, 27, 30, 34, 38, 38, 40, 44, 45, 47, 48, 48, 48, 48, 50, 53, 56, 57, 58, 58, 61, 63, 64, 64, 64, 65, 65, 68, 70, 73, 78, 78, 78, 83, 83, 88, 88, 88, 90, 92, 92, 93, 95, 96, 98, 100, 103, 106, 113, 118, 138, 153, 153, 155, 204, 215, 223, 226, 237, 238, 243, 244, 250, 251, 257, 268, 285, 319, 343, 350.

The dataset is fitted to the proposed distribution and the plot is given in Figure 6. Comparing Figure 6 with that of the graph of WGL distribution in Figure 2, we can say that WGL distribution is appropriate to model the given dataset.

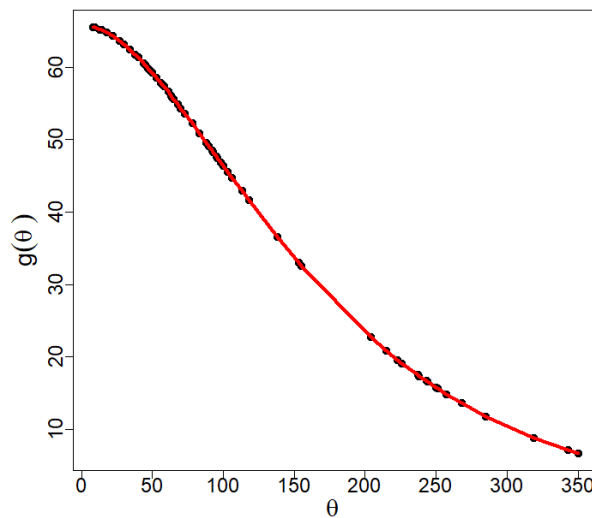


Figure 6: Graph of orientations of 76 turtles after laying eggs

Table 3: Summary of statistics for WL, WTPL, WE, and WGL distributions for the first dataset

Distribution	MLE	Log-likelihood	AIC	BIC	AICC
WL	$\lambda = 0.2163$	-62.223	-120.446	-115.759	-120.284
WTPL	$\theta = 1.151 \times 10^{-6}$ $\alpha = 0.6685$	-79.86	-155.72	-151.01	-155.56
WE	$\lambda = 1.213 \times 10^{-6}$	-83.044	-162.09	-157.4	-161.93
WGL	$\lambda = 0.0525$ $\alpha = 0.0464$	-87.355	-170.709	-165.996	-170.549

Second dataset: The second dataset includes pigeon homing experimental data (Pewsey, 2008). The experiment consists of 13 birds that were released singly in the Toggenburg valley and their vanishing angles (in degrees) were recorded. The dataset is given below:

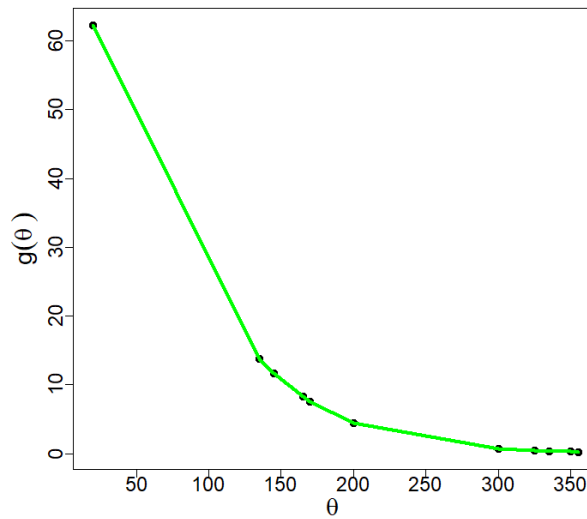


Figure 7: Graph of vanishing angles of 13 birds

20, 135, 145, 165, 170, 200, 300, 325, 335, 350, 350, 350, 350.

Figure 7 displays the graphical representation of pigeon homing experimental data which includes the vanishing angles of 13 birds. From the graph of vanishing angles of 13 birds, we get some similar shape behaviors with that of WGL distribution, given in Figure 1, Figure 2 and Figure 3. Therefore, the given dataset *i.e.*, vanishing angles of 13 birds, can be fitted with the proposed WGL distribution.

Table 4: Summary of statistics for WL, WTPL, WE, and WGL distributions for second dataset

Distribution	MLE	Log-likelihood	AIC	BIC	AICC
WL	$\lambda = 0.5178$	-0.953	2.094	3.372	3.185
WTPL	$\theta = 0.3112$ $\alpha = 1.429 \times 10^{-10}$	-4.251	-4.502	-3.086	-3.502
WE	$\lambda = 2.5750$	-2.546	-1.092	0.186	-0.001
WGL	$\lambda = 0.0890$ $\alpha = 0.0288$	-9.576	-15.152	-13.735	-14.152

Third dataset: The third dataset was taken from Fisher (1995). The dataset includes the orientations of the nest of 50b noisy scrub birds along the bank of a creek bed and the data (in degree) are as given below:

160, 145, 225, 230, 295, 295, 140, 140, 140, 205, 215, 135, 110, 240, 230, 250, 30, 215, 215, 135, 110, 240, 105, 125, 125, 130, 160, 105, 90, 130, 200, 240, 105, 125, 125, 125, 130, 160, 160, 250, 200, 200, 240, 240, 240, 250, 250, 250, 140, 140.

Figure 8 shows the graph of orientations of the nest of 50b noisy scrub birds. The following are some of the reasons for modeling and analyzing the datasets of the orientations

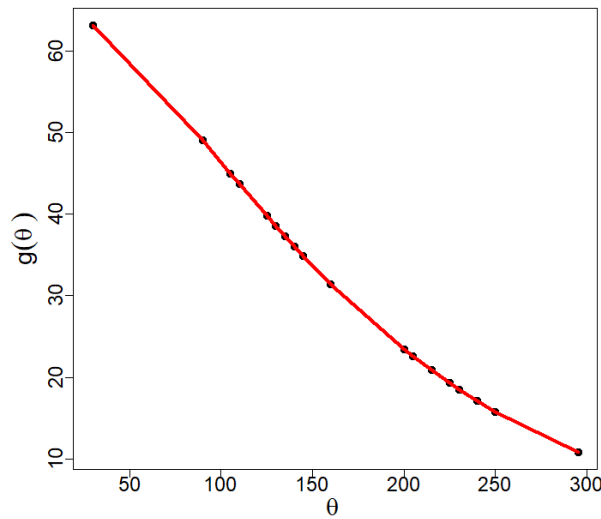


Figure 8: Graph of orientations of the nest of 50b noisy scrub birds

of the nest of 50b noisy scrub birds, directions of 22 sea stars 11 days after being displaced from their natural habitat, and the time series of 72 wind directions, with the WGL distribution. Firstly, Figure 8, Figure 9, and Figure 10 follow some of the characteristics and behaviors of the WGL distribution which can be compared with that of Figure 2. Secondly, all three datasets were measured in degrees that were circular in nature.

Table 5: Summary of statistics for WL, WTPL, WE, and WGL distributions for third dataset

Distribution	MLE	Log-likelihood	AIC	BIC	AICC
WL	$\lambda = 0.1034$	-59.128	-114.255	-110.392	-114.005
WTPL	$\theta = 0.05178$	-58.216	-112.431	-108.529	-112.186
	$\alpha = 2.102 \times 10^{-10}$				
WE	$\lambda = 2.305$	-50.891	-97.782	-93.918	-97.532
WGL	$\lambda = 0.0456$	-65.973	-127.947	-124.044	-127.702
	$\alpha = 0.0419$				

Fourth dataset: Further, to demonstrate the modeling potential of the WGL distribution, sea star movements were considered as reported by Upton (1985) and discussed later by Fisher (1995). The dataset represents the resultant directions of 22 sea stars 11 days after being displaced from their natural habitat, and the data are as below:

0, 1, 3, 3, 8, 13, 16, 18, 30, 31, 43, 45, 147, 298, 329, 332, 335, 340, 350, 354, 356, 357.

Figure 9 represents the graph of directions of 22 sea stars 11 days after being displaced from their natural habitat fitted to the proposed distribution.

Fifth dataset: The fifth dataset is taken from Fisher (1995) which presents the time series of 72 wind directions, comprising hourly measurements for three days at a site on

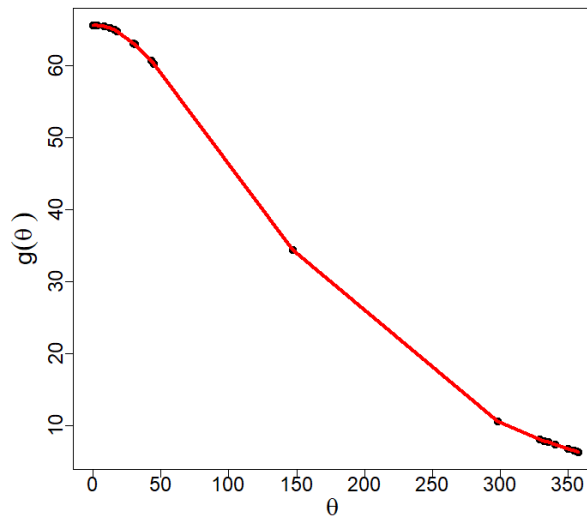


Figure 9: Graph of directions of 22 sea stars

Black Mountain, ACT, Australia. The data is provided below:

285, 285, 280, 300, 240, 255, 250, 250, 235, 240, 240, 180, 220, 265, 180, 150, 150, 150, 335, 355, 335, 305, 345, 340, 315, 0, 330, 300, 330, 330, 50, 270, 270, 270, 245, 285, 280, 270, 15, 285, 310, 330, 300, 340, 280, 300, 270, 270, 255, 90, 285, 285, 285, 270, 270, 270, 270, 270, 300, 300, 270, 300, 330, 15, 330, 300, 345, 330, 330, 300, 315, 285.

Table 6: Summary of statistics for WL, WTPL, WE, and WGL distributions for fourth dataset

Distribution	MLE	Log-likelihood	AIC	BIC	AICC
WL	$\lambda = 0.2592$	4.44	12.88	15.151	13.48
WTPL	$\theta = 0.0000371$ $\alpha = 0.0002602$	-1.691	0.617	2.973	1.189
WE	$\lambda = 1.81$	-0.465	3.069	5.34	3.669
WGL	$\lambda = 0.0865$ $\alpha = 0.0299$	-2.701	-1.401	0.955	-0.830

Figure 10 represents the graph of the fifth dataset fitted to the proposed distribution.

Firstly, it may be noted that all the five datasets taken into consideration are circular in nature which makes it appropriate for the proposed distribution to model these datasets. Secondly, all the distributions taken for comparison, are circular distributions that may be a particular case of Lindley distribution. It is to mention that the smaller values of Log-likelihood, AIC, BIC, and AICC indicate a better fit of distributions. As demonstrated in Table 3, Table 4, Table 5, Table 6, and Table 7, the Log-likelihood, AIC, BIC, and AICC have the lowest values for the WGL distribution. Hence, it can be said that the proposed

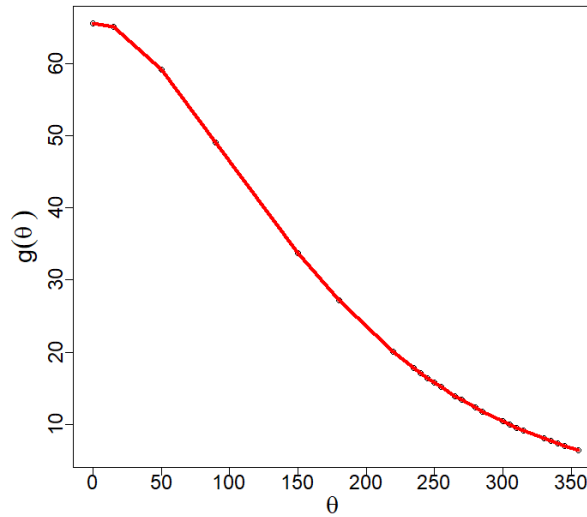


Figure 10: Graph of the time series of 72 wind directions

Table 7: Summary of statistics for WL, WTPL, WE, and WGL distributions for the fifth dataset

Distribution	MLE	Log-likelihood	AIC	BIC	AICC
WL	$\lambda = 0.0975$	-15.136	-26.271	-26.1	-21.691
WTPL	$\theta = 4.94 \times 10^{-02}$ $\alpha = 2.97 \times 10^{-11}$	-13.326	-22.651	-22.482	-18.043
WE	$\lambda = 5.16$	-36.192	-68.383	-68.212	-63.803
WGL	$\lambda = 0.0133$ $\alpha = 0.0268$	-39.869	-75.738	-71.130	-75.569

distribution fits well for all the datasets in comparison to other considered distributions.

9. Conclusion

To summarize, we have considered a new circular distribution by wrapping the generalized Lindley distribution, called the WGL distribution. The density and the distribution function of the proposed distribution were derived and expressions for characteristic functions, trigonometric moments, and other parameters have been discussed. The method of maximum likelihood estimation was used to estimate the model parameters. Further, a simulation study was conducted to show the consistency of the MLE. To show the applicability of the proposed distribution, we fit the WGL distribution to five circular datasets and compare the results with that of WL, WTPL, and WE distributions using Log-likelihood, AIC, BIC, and AICC test-statistics. Based on the above findings, it can be concluded that the WGL distribution provides the best fit for the given datasets than the other distributions taken into consideration.

Acknowledgements

The authors are indeed grateful to the Editors for their guidance and counsel. We would like to acknowledge the reviewers for their profound efforts in reviewing the manuscript and their valuable comments and suggestions, which immensely helped to improve the paper.

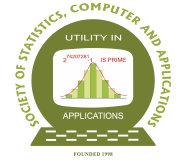
Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Adnan, M. and Roy, S. (2014). Wrapped variance gamma distribution with an application to wind direction. *Journal of Environmental Statistics*, **6**, 1–10.
- Coelho, C. A. (2007). The wrapped gamma distribution and wrapped sums and linear combinations of independent gamma and laplace distributions. *Journal of Statistical Theory and Practice*, **1**, 1–29.
- Fisher, N. I. (1995). *Statistical Analysis of Circular Data*. cambridge university press.
- Gatto, R. and Jammalamadaka, S. R. (2003). Inference for wrapped symmetric α -stable circular models. *Sankhyā: The Indian Journal of Statistics*, **65**, 333–355.
- Jacob, S. and Jayakumar, K. (2013). Wrapped geometric distribution: A new probability model for circular data. *Journal of Statistical Theory and Applications*, **12**, 348–355.
- Jammalamadaka, S. R., Bhadra, N., Chaturvedi, D., Kutty, T., Majumdar, P., and Poduval, G. (1986). Functional assessment of knee and ankle during level walking. *Data Analysis in Life Science*, 21–54.
- Jammalamadaka, S. R. and Kozubowski, T. J. (2004). New families of wrapped distributions for modeling skew circular data. *Communications in Statistics-Theory and Methods*, **33**, 2059–2074.
- Jammalamadaka, S. R. and Sengupta, A. (2001). *Topics in Circular Statistics*, volume 5. world scientific.
- Joshi, S. and Jose, K. (2018). Wrapped lindley distribution. *Communications in Statistics-Theory and Methods*, **47**, 1013–1021.
- Lindley, D. V. (1958). Fiducial distributions and Bayes' theorem. *Journal of the Royal Statistical Society. Series B (Methodological)*, **20** 102–107.
- Mardia, K. V. and Jupp, P. E. (2000). *Directional Statistics*. John Wiley & Sons.
- Pewsey, A. (2008). The wrapped stable family of distributions as a flexible model for circular data. *Computational Statistics & Data Analysis*, **52**, 1516–1523.
- Pewsey, A., Lewis, T., and Jones, M. (2007). The wrapped t family of circular distributions. *Australian & New Zealand Journal of Statistics*, **49**, 79–91.
- PL, L. (1939). Addition of random variables defined on a circumference. *Bulletin of the Mathematical Society of France*, **67**, 1–41.
- Rao, J. S. and Sengupta, S. (1972). Mathematical techniques for paleocurrent analysis: Treatment of directional data. *Journal of the International Association for Mathematical Geology*, **4**, 235–248.

- Roy, S. and Adnan, M. (2012). Wrapped generalized gompertz distribution: An application to ornithology. *Journal of Biometrics & Biostatistics*, **3**, 1–4.
- Sarma, R., Dattatreya Rao, A., and Girija, S. (2011). On characteristic functions of the wrapped lognormal and the wrapped weibull distributions. *Journal of Statistical Computation and Simulation*, **81**, 579–589.
- Shanker, R., Sharma, S., Shanker, U., and Shanker, R. (2013). Janardan distribution and its application to waiting times data. *Indian Journal of Applied Research*, **3**, 500–502.
- Upton, G. J. (1985). *Spatial Data Analysis by Example: Categorical and Directional Data*, volume 2. Wiley.



On the Likelihood Ordering and Tail Behavior of Certain Classes of Skew Normal Distributions

M. R. Anusree¹ and C. Satheesh Kumar²

¹*Department of Statistics, St. Thomas College(Autonomous), Thrissur - 680 001*

²*Department of Statistics, University of Kerala
Kariavattom, Trivandrum -695 581*

Received: 27 June 2024; Revised: 03 November 2024; Accepted: 06 November 2024

Abstract

The scalar skew normal distribution introduced has been used to model asymmetric types of data sets which are unimodal in nature. In order to provide more flexibility in modelling, several classes of asymmetric normal distributions that can accomodate plurimodal data sets has been appeared. Throughout the present note, we provide a comparison between asymmetric normal distributions based on certain stochastic orderings as well as their tail behavior.

Key words: Plurimodality; Probability density function; Skewness; Stochastic ordering; Tail behavior.

AMS Subject Classifications: MSC 60E05, MSC 60E10

1. Introduction

The normal distribution is the basis of many statistical works and it has a unique position in statistical theory and applications. The importance of normality is that, many of the distributions including sampling distributions can be approximated by normal distribution. Also, the distributions corresponding to most of the natural processes are either normally distributed or can be approximated to normal, so it is an important tool for analyzing all types of numerical data. But it is noticed that in many of the practical situations, the data set exhibit a departure from normality, and consequently unrestricted usage of normal distribution creates several types of errors. Even though to handle normal distribution is an easy task, it is not an appropriate tool to handle the quantities which are near to normal but skewed naturally.

Consequently, there has been an increased interest in constructing classes of distributions that can account skewness. Azzalini (1985) initiated a methodology to introduce skewness in normal distribution and termed this class of distributions as “the skew normal distribution (SND)”. The SND has been further studied by Liseo (1990), Ball and Mankiw

(1995), Azzalini and Dalla-Valle (1996), and several others. A detailed account of the SND is available in the books due to Genton (2004) and Azzalini (2014). Following Azzalini (1985), several generalizations came forward. For example, see Kim (2005), Sharafi and Behboodian (2008), Jamalizadeh *et al.* (2008, 2009), Gupta *et al.* (2013) etc. These generalizations can be mainly classified into three broad classes of asymmetric normal distributions those contains the existing types of SND and some of their modified versions. In this direction, Kumar and Anusree (2011, 2013, 2015) considered three classes of asymmetric normal distributions that can accomodate plurimodality and skewness.

In many practical situations, the actuarial and financial data more often exhibits asymmetrically distributed structures with extreme values resulting in tails which are heavier than that of the normal and the tails turn out to behave differently. Tail behaviour of scalar skew normal distribution has been studied in detail by Capitanio (2010), Xin *et al.* (2013). Also, for the asymmetric normal distributions tail behavior and its properties has not been deeply explored, in this context through the present note, we attempt to compare these classes of distributions with their corresponding sub-classes based on some stochastic ordering and nature of tail behavior. The paper is organized as follows. Section 2 contains the description of various asymmetric normal distributions. In Section 3, stochastic ordering of classes of distributions were discussed in detail and the tail behavior in practice of the theoretical results is analyzed in Section 4 and in Section 5, a numerical discussion is carried out.

2. The classes of skew normal distributions

Azzalini (1985) defined the SND as given below.

Definition 1: A random variable X is said to have SND with skewness parameter $\lambda \in R = (-\infty, \infty)$ denoted by $SND(\lambda)$, if its probability density function (p.d.f.) is of the following form, for $x \in R$.

$$g_1(x; \lambda) = 2f(x)F(\lambda x), \quad (1)$$

where $f(\cdot)$ and $F(\cdot)$ are respectively the p.d.f. and cumulative distribution function (c.d.f.) of a standard normal variate and this class of distribution owns the strict inclusion of the normal distribution. The density is strongly unimodal and is not suitable for the analysis of plurimodal situations. Buccianti (2005) remarked that normal and skew normal models are not adequate for describing the situations of plurimodality. There are several such phenomena which cannot be described by either the normal or the skew normal distributions. In order to accommodate plurimodality, Kumar and Anusree (2011) developed a class of asymmetric normal distribution as a modified version of the $SND(\lambda)$, namely “the modified skew normal distribution (MSND)”. They developed the MSND as a generalized mixture of standard normal and skew normal distributions, through the following p.d.f.,

$$g_1^*(x; \lambda, \alpha) = \frac{2}{\alpha + 2} f(x) [1 + \alpha F(\lambda x)], \quad (2)$$

in which $\lambda \in R = (-\infty, \infty)$ and $\alpha \geq -1$. In case of the $SND(\lambda)$, for moderate values of λ nearly all the mass accumulates either on the positive side or on the negative side depending on the sign of the parameter λ . This can be accounted as one of the limitation of the $SND(\lambda)$. In such cases, $SND(\lambda)$ closely resembles the half-normal density, with a nearly linear shape

in the side with smaller mass. To partially mitigate such a limitation, Arellano-Valle *et al.* (2004) introduced a family of asymmetric normal distributions which they termed as “the skew generalized normal distribution (SGND)” that contains the SND as a special case and it exhibits a better behavior, particularly at the sides with smaller mass. Arellano-Valle *et al.* (2004) defined the SGND as given below.

Definition 2: A random variable Y is said to have SGND with parameters $\lambda_1 \in R = (-\infty, \infty)$ and $\lambda_2 \geq 0$, if its p.d.f. takes the following form, for $y \in R$.

$$g_2(y; \lambda_1, \lambda_2) = 2f(y)F\left(\frac{\lambda_1 y}{\sqrt{1 + \lambda_2 y^2}}\right) \quad (3)$$

Following SGND and SND, Kumar and Anusree (2015) proposed a model called “the modified skew generalized normal distribution (MSGND)”, which is capable for describing asymmetric as well as plurimodal situations. Kumar and Anusree (2015) defined the MSGND through the p.d.f.

$$g_2^*(y; \lambda_1, \lambda_2, \alpha) = \frac{2}{\alpha + 2} f(y) \left[1 + \alpha F\left(\frac{\lambda_1 y}{\sqrt{1 + \lambda_2 y^2}}\right) \right], \quad (4)$$

in which $\alpha \geq -1$, $\lambda_1 \in R = (-\infty, \infty)$ and $\lambda_2 \geq 0$. Another class of SND was due to Kim (2005), which he defined as follows:

Definition 3: A random variable Z is said to have “two-piece skew normal distribution (TPSND)” with parameter λ , if its p.d.f. takes the following form, for $z \in R = (-\infty, \infty)$.

$$g_3(z; \lambda) = \frac{2\pi f(x)F(\lambda|z|)}{[\pi + 2\tan^{-1}(\lambda)]} \quad (5)$$

Note that the TPSND is found to be bimodal for all values of its parameters. The TPSND has been further studied by authors such as Jamalizadeh *et al.* (2012), Salehi *et al.* (2014). From a practical point of view, based on the sign of the shape parameter λ , there exists a symmetric behavior on either sides of the origin, which is a main drawback of the TPSND. To overcome this limitation, Kumar and Anusree (2013) considered a modified class of two-piece skew normal distribution, through the p.d.f. for $z \in R$, $\lambda \in R$ and $\rho \in [-1, 1]$.

$$g_3^*(z; \lambda, \rho) = \begin{cases} Cf(z)F(\lambda z), & z < 0 \\ Cf(y)F(\lambda \rho z), & z \geq 0, \end{cases} \quad (6)$$

where $C = 2\pi [\pi - \tan^{-1}(\lambda) + \tan^{-1}(\rho\lambda)]^{-1}$. A distribution with p.d.f. (5) we denoted as TPSND (λ) and a distribution with p.d.f. (6) we denoted as MTPSND (λ, ρ).

3. Stochastic ordering

Here, first we present the definition of likelihood ratio order relation

Definition 4: Let X and Y be two absolutely continuous random variables with p.d.f. $f(\cdot)$ and $g(\cdot)$ respectively. Then X is said to be larger than Y on the basis of likelihood ratio ordering, if $\frac{f(x)}{g(x)}$ is non-decreasing, as x increases.

Thus, for comparing different types of asymmetric normal distributions discussed in this paper, here we obtain the following theorems.

Theorem 1: Let X_1 be a continuous random variable following the MSND with p.d.f.(2) and U be a normal variate. Then X_1 is larger than U on the basis of likelihood ratio ordering if for all $\lambda > 0$, $F(\lambda x_1) \leq F(\lambda u)$.

Proof: Let X_1 follows MSND with p.d.f.(2) and $g_U(x)$ be the p.d.f. of U . Then by definition 4, the ratio

$$\frac{2}{(\alpha + 2)f(x)} f(x)[1 + \alpha F(\lambda x)]$$

is non decreasing only if for $x_1 < x_2$

$$\begin{aligned} \frac{2f(x_1)[1 + \alpha F(\lambda x_1)]}{(\alpha + 2)f(x_1)} &\leq \frac{2f(x_2)[1 + \alpha F(\lambda x_2)]}{(\alpha + 2)f(x_2)} \\ F(\lambda x_1) &\leq F(\lambda x_2), \text{ for all } \lambda > 0. \end{aligned}$$

□

Theorem 2: Let X_1 be a continuous random variable following the MSND with p.d.f.(2) and X follows $SND(\lambda)$. Then X_1 is said to be larger than X on the basis of likelihood ratio ordering if $F(\lambda x) \leq F(\lambda x_1)$ for all $\lambda > 0$.

Proof: Let X_1 follows MSND with p.d.f.(2) and $g_X(x)$ be the p.d.f. of X . Then by definition 4, the ratio $\frac{[1 + \alpha F(\lambda x)]}{(\alpha + 2)F(\lambda x)}$ is non decreasing only if for $x_1 < x_2$

$$\begin{aligned} \frac{2f(x_1)[1 + \alpha F(\lambda x_1)]}{(\alpha + 2)2f(x_1)F(\lambda x_1)} &\leq \frac{2f(x_2)[1 + \alpha F(\lambda x_2)]}{(\alpha + 2)2f(x_2)F(\lambda x_2)} \\ F(\lambda x_2) &\leq F(\lambda x_1), \text{ for all } \lambda > 0. \end{aligned}$$

□

Theorem 3: Let Y_1 be a random variable following MSGND with parameters $\lambda_1 \in R$, $\lambda_2 \geq 0$ and $\alpha \geq -1$ and U be a normal variate. Then Y_1 is said to be larger than U on the basis of likelihood ratio ordering if for all $\lambda_1 > 0$, $\lambda_2 \geq 0$, $F\left(\frac{\lambda_1 y_1}{\sqrt{1 + \lambda_2 y_1^2}}\right) \leq F\left(\frac{\lambda_1 u}{\sqrt{1 + \lambda_2 u^2}}\right)$.

Proof: Y_1 follows MSGND with p.d.f. (4) and $g_U(x)$ be the p.d.f. of U . Then by definition 4, the ratio $\frac{2}{(\alpha + 2)} f(x)[1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right)]/f(x)$ is non decreasing only if for $x_1 < x_2$,

$$\begin{aligned} \frac{2}{(\alpha + 2)} \left[1 + \alpha F\left(\frac{\lambda_1 x_1}{\sqrt{1 + \lambda_2 x_1^2}}\right) \right] &\leq \frac{2}{(\alpha + 2)} \left[1 + \alpha F\left(\frac{\lambda_1 x_2}{\sqrt{1 + \lambda_2 x_2^2}}\right) \right] \\ \left[1 + \alpha F\left(\frac{\lambda_1 x_1}{\sqrt{1 + \lambda_2 x_1^2}}\right) \right] &\leq \left[1 + \alpha F\left(\frac{\lambda_1 x_2}{\sqrt{1 + \lambda_2 x_2^2}}\right) \right] \\ F\left(\frac{\lambda_1 x_1}{\sqrt{1 + \lambda_2 x_1^2}}\right) &\leq F\left(\frac{\lambda_1 x_2}{\sqrt{1 + \lambda_2 x_2^2}}\right), \text{ for all } \lambda_1 > 0, \lambda_2 \geq 0. \end{aligned}$$

□

Theorem 4: Let Y_1 be a random variable following MSGND with parameters $\lambda_1 \in R$, $\lambda_2 \geq 0$ and $\alpha \geq -1$ and Y follows SGND(λ_1, λ_2). Then Y_1 is said to be larger than Y on the basis of likelihood ratio ordering if for all $\lambda_1 > 0$, $\lambda_2 \geq 0$, $F\left(\frac{\lambda_1 y}{\sqrt{1+\lambda_2 y^2}}\right) \leq F\left(\frac{\lambda_1 y_1}{\sqrt{1+\lambda_2 y_1^2}}\right)$

Proof: Y_1 follows MSGND with p.d.f. (4) and $g_Y(y)$ be the p.d.f. of Y . Then by definition 4, the ratio

$$\frac{\frac{2}{(\alpha+2)}f(y_1) \left[1 + \alpha F\left(\frac{\lambda_1 y_1}{\sqrt{1+\lambda_2 y_1^2}}\right)\right]}{2f(y)F\left(\frac{\lambda_1 y}{\sqrt{1+\lambda_2 y^2}}\right)}$$

is non-decreasing only if for $x_1 < x_2$,

$$\begin{aligned} \frac{2}{(\alpha+2)F\left(\frac{\lambda_1 x_1}{\sqrt{1+\lambda_2 x_1^2}}\right)} \left[1 + \alpha F\left(\frac{\lambda_1 x_1}{\sqrt{1+\lambda_2 x_1^2}}\right)\right] &\leq \frac{2}{(\alpha+2)F\left(\frac{\lambda_1 x_2}{\sqrt{1+\lambda_2 x_2^2}}\right)} \left[1 + \alpha F\left(\frac{\lambda_1 x_2}{\sqrt{1+\lambda_2 x_2^2}}\right)\right] \\ \left[\alpha + \frac{1}{F\left(\frac{\lambda_1 x_1}{\sqrt{1+\lambda_2 x_1^2}}\right)}\right] &\leq \left[\alpha + \frac{1}{F\left(\frac{\lambda_1 x_2}{\sqrt{1+\lambda_2 x_2^2}}\right)}\right] \\ F\left(\frac{\lambda_1 x_2}{\sqrt{1+\lambda_2 x_2^2}}\right) &\leq F\left(\frac{\lambda_1 x_1}{\sqrt{1+\lambda_2 x_1^2}}\right), \end{aligned}$$

for all $\lambda_1 > 0, \lambda_2 \geq 0$.

□

Theorem 5: Let Y_1 be a random variable following MSGND with parameters $\lambda_1 \in R$, $\lambda_2 \geq 0$ and $\alpha \geq -1$ and X_1 follows MSND (λ, α). Then Y_1 is said to be larger than X_1 on the basis of likelihood ratio ordering if $A(x_1, y_1) \leq B(x_1, y_1)$ where

$$\begin{aligned} A(x_1, y_1) &= \alpha F(\lambda x_1) + \alpha F\left(\frac{\lambda_1 y_1}{\sqrt{1+\lambda_2 y_1^2}}\right) + \alpha^2 F\left(\frac{\lambda_1 y_1}{\sqrt{1+\lambda_2 y_1^2}}\right) F(\lambda x_1) \text{ and} \\ B(y_1, x_1) &= \alpha F(\lambda y_1) + \alpha F\left(\frac{\lambda_1 x_1}{\sqrt{1+\lambda_2 x_1^2}}\right) + \alpha^2 F\left(\frac{\lambda_1 x_1}{\sqrt{1+\lambda_2 x_1^2}}\right) F(\lambda x_1) \end{aligned}$$

Proof: Y_1 follows MSGND with p.d.f. (4) and $g_{X_1}(x)$ be the p.d.f. of X_1 . Then by definition 4, the ratio

$$\frac{\frac{2}{(\alpha+2)}f(x) \left[1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{1+\lambda_2 x^2}}\right)\right]}{2f(x)F(\lambda x)}$$

is non-decreasing only if for $x_1 < x_2$,

$$\frac{\frac{2f(x_1)}{(\alpha+2)} \left[1 + \alpha F\left(\frac{\lambda_1 x_1}{\sqrt{1+\lambda_2 x_1^2}}\right)\right]}{\frac{2f(x_1)}{(\alpha+2)} [1 + \alpha F(\lambda x_1)]} \leq \frac{\frac{2f(x_2)}{(\alpha+2)} \left[1 + \alpha F\left(\frac{\lambda_1 x_2}{\sqrt{1+\lambda_2 x_2^2}}\right)\right]}{\frac{2f(x_2)}{(\alpha+2)} [1 + \alpha F(\lambda x_2)]}$$

$$\begin{aligned}
& \left[1 + \alpha F \left(\frac{\lambda_1 x_1}{\sqrt{1 + \lambda_2 x_1^2}} \right) \right] [1 + \alpha F(\lambda x_2)] \leq \left[1 + \alpha F \left(\frac{\lambda_1 x_2}{\sqrt{1 + \lambda_2 x_2^2}} \right) \right] [1 + \alpha F(\lambda x_1)] \\
& \alpha F(\lambda x_2) + \alpha F \left(\frac{\lambda_1 x_1}{\sqrt{1 + \lambda_2 x_1^2}} \right) + \alpha^2 F \left(\frac{\lambda_1 x_1}{\sqrt{1 + \lambda_2 x_1^2}} \right) F(\lambda x_2) \leq \alpha F(\lambda x_1) + \alpha F \left(\frac{\lambda_1 x_2}{\sqrt{1 + \lambda_2 x_2^2}} \right) + \\
& \alpha^2 F \left(\frac{\lambda_1 x_2}{\sqrt{1 + \lambda_2 x_2^2}} \right) F(\lambda x_1) \text{ holds for all } \lambda_1 \geq 0, \lambda_2 \geq 0, \alpha \geq -1, \text{ and } \lambda \geq 0. \quad \square
\end{aligned}$$

Theorem 6: Let Z_1 be a random variable following MTPSND with parameters $\lambda \in R$, $\rho \in [-1, 1]$, and U be a normal variate. Then Z_1 is said to be larger than U on the basis of likelihood ratio ordering if for $\lambda \geq 0$ and $\rho \in [0, 1]$.

$$\begin{cases} F(\lambda z_1) \leq F(\lambda u), z_1 < 0, u < 0 \\ F(\rho \lambda z_1) \leq F(\rho \lambda u), z_1 > 0, u > 0 \end{cases}$$

Proof: Z_1 follows MTPSND with p.d.f. (6) and $g_U(x)$ be the p.d.f. of U . Then by definition 4, the ratio

$$\begin{cases} \frac{Cf(x)F(\lambda x)}{f(x)}, x < 0 \\ \frac{Cf(x)F(\rho \lambda x)}{f(x)}, x \geq 0 \end{cases}$$

is decreasing only if for $x_1 < x_2$

$$\begin{cases} \frac{cf(x_1)F(\lambda x_1)}{f(x_1)} \leq \frac{cf(x_2)F(\lambda x_2)}{f(x_2)}, & x_1 < 0, x_2 < 0, x_1 < x_2 \\ \frac{cf(x_1)F(\rho \lambda x_1)}{f(x_1)} \leq \frac{cf(x_2)F(\rho \lambda x_2)}{f(x_2)}, & x_1 > 0, x_2 > 0, x_1 < x_2 \end{cases}$$

$$\begin{cases} F(\lambda x_1) \leq F(\lambda x_2), & x_1 < 0, x_2 < 0, x_1 < x_2 \\ F(\rho \lambda x_1) \leq F(\rho \lambda x_2), & x_1 > 0, x_2 > 0, x_1 < x_2 \end{cases}$$

for all $\lambda \geq 0$ and $\rho \in [0, 1]$. □

Theorem 7: Let Z_1 be a random variable following MTPSND with parameters $\lambda \in R$, $\rho \in [-1, 1]$, and Z follows TPSND (λ). Then Z_1 is said to be larger than Z on the basis of likelihood ratio ordering if for any $\lambda \geq 0$ and $\rho \in [0, 1]$,

$$\begin{cases} F(\lambda z_1) \leq F(\lambda z), z_1 < 0, z < 0 \\ F(\rho \lambda z_1) \leq F(\rho \lambda z), z_1 > 0, z < 0 \end{cases}$$

Proof: Z_1 follows MTPSND with p.d.f. (6) and $g_Z(x)$ be the p.d.f. of Z . Then by definition 4, for $x_1 < x_2$ the ratio

$$\begin{cases} \frac{Cf(x_1)F(\lambda x_1)}{f(x_1)} \leq \frac{Cf(x_2)F(\lambda x_2)}{f(x_2)} \\ \frac{Cf(x_1)F(\rho \lambda x_1)}{f(x_1)} \leq \frac{Cf(x_2)F(\rho \lambda x_2)}{f(x_2)} \end{cases}$$

$$\begin{cases} F(\lambda x_1) \leq F(\lambda x_2), & x_1 < 0, x_2 < 0, x_1 < x_2 \\ F(\rho \lambda x_1) \leq F(\rho \lambda x_2), & x_1 > 0, x_2 > 0, x_1 < x_2 \end{cases}$$

for any $\lambda \geq 0$ and $\rho \in [0, 1]$. □

4. Tail behavior of the distribution

Tail behaviors are discussed with reference to Tse (2009).

Definition 5: Let X and Y be two absolutely continuous random variables with probability density functions $f(x)$ and $g(x)$ respectively. Then, the tail behavior of the two distributions can be considered as the limiting ratio of their densities. That is, tail of the numerator density will be thinner (or thicker) than the denominator density as the ratio approaches to zero (or infinity).

Now we obtain the following theorem's, which are helpful for the comparison of different types of asymmetric normal distribution based on the nature of their tail behaviour.

Theorem 8: Let X follows $SND(\lambda)$ with p.d.f (1) and X_1 follows $MSND(\lambda, \alpha)$ with p.d.f. (2). Then X_1 has thinner tail than that of X , when $\alpha \rightarrow -1$ and $\lambda \neq 0$.

Proof: The limiting ratios of the densities of the variables X_1 and X is given by

$$\begin{aligned} \lim_{x \rightarrow \infty} \frac{g_{X_1}(x)}{g_X(x)} &= \lim_{x \rightarrow \infty} \frac{\frac{2}{\alpha+2} f(x) [1 + \alpha F(\lambda x)]}{2f(x) F(\lambda x)} \\ &= \lim_{x \rightarrow \infty} \frac{[1 + \alpha F(\lambda x)]}{(\alpha + 2) F(\lambda x)} \\ &= \frac{1 + \alpha}{2 + \alpha}, \text{ for } \lambda \neq 0 \end{aligned} \quad (7)$$

Thus, the right hand side expression of (7) tends to zero only when $\alpha \rightarrow -1$. $\square$

Theorem 9: Let Y_1 follows $MSGND(\lambda_1, \lambda_2, \alpha)$ with p.d.f. (4) and Y follows $SGND(\lambda_1, \lambda_2)$ with p.d.f. (3). Then (i) Y_1 has thinner tail than that of Y if $\alpha \rightarrow -1$, $\lambda_1 \rightarrow \infty$ or $\alpha \rightarrow -1$, $\lambda_2 \rightarrow 0$ and (ii) Y_1 has thicker tail than that of Y if $\alpha \rightarrow -1$ and $\lambda_1 \rightarrow -\infty$ or $\alpha \rightarrow 0$ and $\lambda_1 \rightarrow -\infty$.

Proof: Case (i): The limiting ratios of the densities of the variables Y_1 and Y can be written as

$$\begin{aligned} \lim_{x \rightarrow \infty} \frac{h_{Y_1}(x)}{h_Y(x)} &= \lim_{x \rightarrow \infty} \frac{\frac{2}{\alpha+2} f(x) \left[1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right) \right]}{2f(x) F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right)} \\ &= \frac{1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right)}{(\alpha + 2) F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right)} \\ &\rightarrow \frac{F\left(\frac{-\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right)}{F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right)}, \text{ as } \alpha \rightarrow -1 \\ &\rightarrow 0, \text{ as } \lambda_1 \rightarrow \infty \end{aligned}$$

Thus, Y_1 is thinner than that of Y if $\alpha \rightarrow -1$ and $\lambda_1 \rightarrow \infty$.

In a similar way,

$$\begin{aligned} \lim_{x \rightarrow \infty} \frac{h_{Y_1}(x)}{h_Y(x)} &= \frac{1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)}{(\alpha + 2)F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)} \\ &= \frac{1 + \alpha}{(\alpha + 2)}, \text{ as } \lambda_2 \rightarrow 0 \\ &\rightarrow 0, \text{ as } \alpha \rightarrow -1 \end{aligned}$$

Thus, Y_1 is thinner than that of Y if $\alpha \rightarrow -1$ and $\lambda_2 \rightarrow 0$.

Case (ii): The limiting ratios of the densities of the variables Y_1 and Y can be written as

$$\begin{aligned} \lim_{x \rightarrow \infty} \frac{h_{Y_1}(x)}{h_Y(x)} &= \lim_{x \rightarrow \infty} \frac{\frac{2}{\alpha + 2} f(x) \left[1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right) \right]}{2f(x)F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)} \\ &= \frac{1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)}{(\alpha + 2)F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)} \\ &\rightarrow \frac{F\left(\frac{-\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)}{F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)}, \text{ as } \alpha \rightarrow -1 \\ &\rightarrow \infty, \text{ as } \lambda_1 \rightarrow -\infty \end{aligned}$$

Thus, Y_1 is thicker than that of Y if $\alpha \rightarrow -1$ and $\lambda_1 \rightarrow -\infty$.

Similarly

$$\begin{aligned} \lim_{x \rightarrow \infty} \frac{h_{Y_1}(x)}{h_Y(x)} &= \lim_{x \rightarrow \infty} \frac{\frac{2}{\alpha + 2} f(x) \left[1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right) \right]}{2f(x)F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)} \\ &\rightarrow \frac{1}{2F\left(\frac{\lambda_1 x}{\sqrt{(1 + \lambda_2 x^2)}}\right)}, \text{ as } \alpha \rightarrow 0 \\ &\rightarrow \infty, \text{ as } \lambda_1 \rightarrow -\infty \end{aligned}$$

Thus, Y_1 is thicker than that of Y if $\alpha \rightarrow 0$ and $\lambda_1 \rightarrow -\infty$. □

Theorem 10: Let Z_1 follows MTPSND (λ, ρ) with p.d.f.(6) and Z follows TPSND (λ) with p.d.f.(5). Then Z_1 is thinner than Z if $\rho \rightarrow 1$ and $\lambda \rightarrow -\infty$ and is thicker than Z if $\rho \rightarrow -1$ and $\lambda \rightarrow -\infty$.

Proof: The limiting ratios of the densities of the variables Z_1 and Z can be written in the following way, in which $C = 2\pi[\pi + 2 \tan^{-1}(\lambda)]^{-1}$ and $C_0 = 2\pi[\pi - \tan^{-1}(\lambda) + \tan^{-1}(\rho\lambda)]^{-1}$.

$$\lim_{x \rightarrow \infty} \frac{g_{Z_1}(x)}{g_Z(x)} = \lim_{x \rightarrow \infty} \frac{C_0 f(x) F(\rho\lambda x)}{C f(x) F(\lambda x)}$$

$$\begin{aligned}
&= \frac{\pi + 2 \tan^{-1}(\lambda)}{\pi - \tan^{-1}(\lambda) + \tan^{-1}(\rho\lambda)} \\
&\rightarrow 0, \text{ as } \rho \rightarrow 1 \text{ and } \lambda \rightarrow -\infty
\end{aligned}$$

Thus, Z_1 is thinner than Z if $\rho \rightarrow 1$ and $\lambda \rightarrow -\infty$. Also,

$$\begin{aligned}
\lim_{x \rightarrow \infty} \frac{g_{Z_1}(x)}{g_Z(x)} &= \frac{\pi + 2 \tan^{-1}(\lambda)}{\pi - \tan^{-1}(\lambda) + \tan^{-1}(\rho\lambda)} \\
&\rightarrow \infty, \text{ as } \rho \rightarrow -1 \text{ and } \lambda \rightarrow \infty
\end{aligned}$$

Thus, Z_1 is thicker than Z if $\rho \rightarrow -1$ and $\lambda \rightarrow \infty$. $\square$

Theorem 11: Let X_1 follows MSND (λ, α) with p.d.f. (2) and Y_1 follows MSGND $(\lambda_1, \lambda_2, \alpha)$ with p.d.f.(4). Then X_1 is thinner than Y_1 when $\alpha \rightarrow -1$.

Proof: The limiting ratios of the densities of the variables X_1 and Y_1 is given by

$$\begin{aligned}
\lim_{x \rightarrow \infty} \frac{g_{X_1}(x)}{h_{Y_1}(x)} &= \lim_{x \rightarrow \infty} \frac{\frac{2}{\alpha+2} f(x) [1 + \alpha F(\lambda x)]}{\frac{2}{\alpha+2} f(x) \left[1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right) \right]} \\
&= \frac{1 + \alpha F(\lambda x)}{1 + \alpha F\left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}}\right)} \\
&\rightarrow 0,
\end{aligned}$$

when $\alpha \rightarrow -1$. Thus, X_1 is thinner than Y_1 for $\alpha \rightarrow -1$. $\square$

Theorem 12: Let X_1 follows MSND (λ, α) with p.d.f.(2) and Z_1 follows MTPSND (λ, ρ) with p.d.f.(6). Then X_1 is thinner than Z_1 either if $\alpha \rightarrow 0$ or $\alpha \rightarrow -1$ and $\rho \rightarrow -1$.

Proof: The limiting ratios of the densities of the variables X_1 and Z_1 is

$$\begin{aligned}
\lim_{x \rightarrow \infty} \frac{g_{X_1}(x)}{g_{Z_1}(x)} &= \lim_{x \rightarrow \infty} \frac{\frac{2}{\alpha+2} f(x) [1 + \alpha F(\lambda x)]}{C f(x) F(\rho \lambda x)} \\
&= \frac{[\pi - \tan^{-1}(\lambda) + \tan^{-1}(\rho \lambda)]}{\pi(\alpha + 2)} \\
&\rightarrow 0,
\end{aligned}$$

when $\alpha \rightarrow 0$ or $\alpha \rightarrow -1$ and $\rho \rightarrow -1$. Thus, then X_1 is thinner than Z_1 when $\alpha \rightarrow 0$ or $\alpha \rightarrow -1$ and $\rho \rightarrow -1$. $\square$

Theorem 13: Let Y_1 follows MSGND $(\lambda_1, \lambda_2, \alpha)$ with p.d.f.(4) and Z_1 follows MTPSND (λ, ρ) with p.d.f.(6). Then Y_1 is thinner than Z_1 if for $\lambda_1 \rightarrow \infty$ either $\alpha \rightarrow -1$, $\rho \rightarrow 1$ or $\alpha \rightarrow -1$, $\rho \rightarrow -1$.

Proof: The limiting ratios of the densities $h_{Y_1}(x)$ and $k_{Z_1}(x)$ of the variables Y_1 and Z_1 is

$$\begin{aligned} \lim_{x \rightarrow \infty} \frac{h_{Y_1}(x)}{k_{Z_1}(x)} &= \lim_{x \rightarrow \infty} \frac{\frac{2}{\alpha+2} f(x) \left[1 + \alpha F \left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}} \right) \right]}{C f(x) F(\rho \lambda x)} \\ &= \frac{[\pi - \tan^{-1}(\lambda) + \tan^{-1}(\rho \lambda)] \left[1 + \alpha F \left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}} \right) \right]}{\pi(\alpha + 2) F(\rho \lambda x)} \\ &= \begin{cases} \frac{1 + \alpha F \left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}} \right)}{(\alpha + 2) F(\lambda x)}, & \text{as } \rho \rightarrow 1 \\ \frac{[\pi - 2 \tan^{-1}(\lambda)] \left[1 + \alpha F \left(\frac{\lambda_1 x}{\sqrt{1 + \lambda_2 x^2}} \right) \right]}{\pi(\alpha + 2) F(-\lambda x)}, & \text{as } \rho \rightarrow -1 \end{cases} \end{aligned}$$

$\rightarrow 0$, as $\alpha \rightarrow -1$, $\lambda_1 \rightarrow \infty$

Thus, Y_1 is thinner than Z_1 if for $\lambda_1 \rightarrow \infty$ either $\alpha \rightarrow -1$, $\rho \rightarrow 1$ or $\alpha \rightarrow -1$, $\rho \rightarrow -1$. $\square$

5. Numerical discussion

The tail behaviour is influenced by its skewness parameter where one tail becomes heavier relative to the other depending shape parameter. It is very crucial in the study of rare events, risk assessment, risk management and insurance. Here for numerical illustration we have compared various distributions based on their Hill estimator value for particular values of its parameters and is shown in Table 1. In a similar manner, one can compare heavier tails using this method this value will typically be close to zero, indicating light tails or tail is not heavy. Also, the tail behaviour of classes of distribution for various simulated values of its parameter are obtained is as shown in Table 2.

Table 1: Simulated Hill Estimator values for SND and MSND, MSGND and SGND, MTPSND and TPSND, MSND and MTPSND, MSGND and MTPSND for particular values of $(\alpha = 0.5, \lambda = 0.5, \rho, \lambda_1 = 0.2$ and $\lambda_2 = 0.4)$

Distributions	Hill Estimator
MSND	0.0045
MSGND	0.002
SGND	0.0048
MTPSND	0.0098
TPSND	0.56
MSND	0.432
MTPSND	0.0029
MSGND	0.395
MTPSND	0.23

Table 2: Ratio of the densities and its behaviour for various values of its parameters

Distributions	Ratio				Behaviour
SND(0.5), MSND(0.5,-1)	0	0	0	0	X_1, X are Comparable
SND(0.5), MSND(0.5,0.55)	0.7828275	0.6817935	0.6359176	0.6169725	X_1 thinner, X thicker
SND(0.5), MSND(0.5,0.65)	0.7910227	0.6938013	0.6496566	0.6314263	X_1 thinner, X thicker
SND(0.5), MSND(0.5,0.75)	0.7986219	0.7049358	0.6623963	0.6448290	X_1 thinner, X thicker
SND(0.5), MSND(0.5,0.85)	0.8056878	0.7152889	0.6742421	0.6572911	X_1 thinner, X thicker
MSGND(0.5,0.6,-1), SGND(0.5,0.6)	0	0	0	0	Y_1, Y are Comparable
MSGND(0.5,0.6,0.55), SGND(0.5,0.6)	0.8156047	0.7709895	0.7577999	0.7525272	Y_1 thinner, Y thicker
MSGND(0.5,0.6,0.65), SGND(0.5,0.6)	0.8225630	0.7796314	0.7669395	0.7618658	Y_1 thinner, Y thicker
MSGND(0.5,0.6,0.75), SGND(0.5,0.6)	0.8290153	0.7876448	0.7754144	0.7705253	Y_1 thinner, Y thicker
MSGND(0.5,0.6,0.85), SGND(0.5,0.6)	0.8350148	0.7950959	0.7832946	0.7785770	Y_1 thinner, Y thicker
MSGND(0.5,0.6,-1), MSND(0.5,alpha)	0	0	0	0	X_1, Y_1 are Comparable
MSGND(0.5,0.6,0.55), MSND(0.5,alpha)	0.8350148	0.7950959	0.7832946	0.7785770	X_1 thinner, Y_1 thicker
MSGND(0.5,0.6,0.65), MSND(0.5,alpha)	0.8250121	0.7930989	0.7832946	0.7685571	X_1 thinner, Y_1 thicker
MSGND(0.5,0.6,0.75), MSND(0.5,alpha)	0.8250248	0.7940959	0.7852546	0.7685670	X_1 thinner, Y_1 thicker
MSGND(0.5,0.6,0.85), MSND(0.5,alpha)	0.8250314	0.7850959	0.7732946	0.7589670	X_1 thinner, Y_1 thicker

Acknowledgements

We are indeed grateful to the Editors for their guidance and counsel. We are very thankful to the reviewer for valuable comments and suggestions of generously listing many useful references.

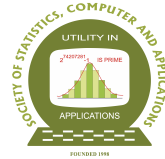
Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Arellano-Valle, R. B., Gomez, H. W., and Quintana, F. A. (2004). A new class of skew normal distribution. *Communications in Statistics - Theory and Methods*, **33**, 1465–1480.
- Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scandinavian Journal of Statistics*, **12**, 171–178.
- Azzalini, A. (2014). *The Skew Normal and Related Families*. Cambridge University Press.
- Azzalini, A. and Dalla-Valle, A. (1996). The multivariate skew normal distribution. *Biometrika*, **83**, 175–176.
- Ball, L. and Mankiw, N. G. (1995). Relative price changes as aggregate supply shocks. *The Quarterly Journal of Economics*, **110**, 161–193.
- Buccianti, A. (2005). Meaning of the parameter of skew normal and log skew normal distributions in fluid geo chemistry. In *CODAWORK'05*, pages 1–15.
- Capitanio, A. (2010). On the approximation of the tail probability of the scalar skew-normal distribution. *Metron - International Journal of Statistics*, **LXVIII**, 299–308.
- Genton, M. G. (2004). *Skew-elliptical Distributions and Their Applications: A Journey Beyond Normality*. Chapman and Hall/CRC, London.
- Gupta, A. K., Aziz, M. A., and Ning, W. (2013). On some properties of the unified skew normal distribution. *Journal of Statistical Theory and Practice*, **7**, 480–495.

- Jamalizadeh, A., Arabpour, A. R., and Balakrishnan, N. (2009). A generalized skew two-piece skew-normal distribution. *Statistical Papers*, **79**, 99–113.
- Jamalizadeh, A., Arabpour, A. R., and Balakrishnan, N. (2012). A generalized skew two-piece skew-normal distribution. *Statistical Papers*, **52**, 431–446.
- Jamalizadeh, A., Behboodian, J., and Balakrishnan, N. (2008). A two-parameter generalized skew-normal distribution. *Statistics and Probability Letters*, **78**, 1722–1726.
- Kim, H. J. (2005). On a class of two-piece skew-normal distributions. *Statistics*, **39**, 537–553.
- Kumar, C. S. and Anusree, M. R. (2011). On a generalized mixture of standard normal and skew normal distributions. *Statistics and Probability Letters*, **81**, 1813–1821.
- Kumar, C. S. and Anusree, M. R. (2013). A generalized two-piece skew normal distribution and some of its properties. *Statistics*, **47**, 1370–1380.
- Kumar, C. S. and Anusree, M. R. (2015). On an extended version of skew generalized normal distribution and some of its properties. *Communications in Statistics - Theory and Methods*, **44**, 573–586.
- Liseo, B. (1990). The skew-normal class of densities, inferential aspects from a Bayesian viewpoint. *Statistica*, **50**, 71–82.
- Salehi, M., Jamalizadeh, A., and Doostparast, M. (2014). A generalized skew two-piece skew elliptical distribution. *Statistical Papers*, **55**, 409–429.
- Sharafi, M. and Behboodian, J. (2008). The Balakrishnan skew-normal density. *Statistical Papers*, **49**, 769–778.
- Tse, Y. K. (2009). *Non-life Actuarial Models, Theory Methods and Evaluation*. Cambridge University Press.
- Xin, L., Zuoxiang, P., and Saralees, N. (2013). Tail properties and asymptotic expansions for the maximum of the logarithmic skew-normal distribution. *Journal of Applied Probability*, **50**, 900–907.



Advancements in Power Inverse Rayleigh Modeling: Exploring Applications in Environmental and Medical Domains

Aadil Ahmad Mir¹, S. P. Ahmad¹ and A. A. Bhat²

¹*Department of Statistics, University of Kashmir, Srinagar, Jammu and Kashmir, India*

²*Department of Mathematical Sciences, Islamic University of Science and Technology,
Awantipora, Jammu and Kashmir, India*

Received: 02 September 2024; Revised: 16 November 2024; Accepted: 19 November 2024

Abstract

In this manuscript, a novel probability distribution, known as the *MTI* inverse power Rayleigh distribution (*MTI-IPRD*), is obtained through the use of the *MTI* transformation. This new distribution fits data better than many existing models. A number of statistical features are examined in detail. The maximum likelihood estimation (*MLE*) procedure is employed to estimate the unknown parameters. An extensive simulation study is carried out to illustrate the behaviour of *MLEs* on the basis of mean square errors. In addition, the suggested distribution's flexibility and importance are assessed in comparison to existing probability distributions using three real data sets.

Key words: *MTI* transformation; Rayleigh distribution; Moments; Renyi entropy; Stress strength reliability; Maximum likelihood estimation.

AMS Subject Classifications: 62K05, 05B05

1. Introduction

Lifetime phenomena modelling and analysis are essential components of statistical work in many scientific and technological fields. Lifetime data analysis has had tremendous growth and progress in terms of theory, applications, and technique. Various generalisation or transformation techniques together with continuous probability distributions have been presented to simulate real-life situations. The model's acceptability and applicability are enhanced by these generalisations for a variety of uses. These can be achieved by altering the distribution's functional form or by including one or more shape factors.

One popular lifetime probability model for explaining skewed data is the Rayleigh distribution (RD). Lord Rayleigh (1880) is the inspiration behind its name. The statistical literature contains numerous proposals for extending the Rayleigh distribution (RD). Surles and Padgett (2001) studied exponentiated RD (ERD). Using various estimating techniques, Kundu and Raqab (2005) investigated and evaluated the parameters of the generalised RD . Abd Elfattah *et al.* (2006) investigated the effectiveness of MLE under different censored sampling strategies for the RD . The odd Lindley power RD was extended by Bhat *et al.* (2023), who also examined its characteristics and assessed parameter estimate approaches using Bayesian and classical approaches. Using the power transformation technique, Bhat and Ahmad (2020) presented a new generalisation of the RD . The sine power RD was presented and its features and applications were explored by Mir and Ahmad (2024). By Voda (1972), the inverse RD (IRD) was proposed. Additionally, the parameters were estimated by Gharrahi (1993) and Soliman *et al.* (2010) utilising various classical and Bayesian estimation techniques. For instance, Merovci (2013) and Merovci (2014) created transmuted inverse RD and transmuted RD , respectively, using the $QRTM$.

This paper's primary goal is to suggest and investigate a new lifetime model based on the MTI approach, known as the MTI inverse power Rayleigh distribution ($MTI-IPRD$). The main goal of the new model is to give the hazard and density functions more flexibility and other desirable qualities through the addition of a parameter. Furthermore, when applied to three real data sets, the suggested model performs better than a few well-known models. This article's remaining sections are organised as follows. The $MTI-IPRD$ is introduced in Section 2. The reliability analysis is covered in Section 3. Its statistical properties are discussed in Section 4. The objective of Section 5 is to estimate the unknown parameters using the maximum likelihood technique. The outcomes of the simulation are shown in Section 6. In Section 7, the model's applicability is illustrated. Lastly, Section 8 presents conclusions.

2. MTI inverse power Rayleigh distribution ($MTI-IPRD$)

The CDF and PDF of the MTI transformation method proposed by Lone *et al.* (2022) are defined by

$$F_{MTI}(y) = \frac{\eta F(y)}{\eta - \log \eta \bar{F}(y)} \quad ; \quad y \in \mathbb{R}, \eta \in \mathbb{R}^+ \quad (1)$$

where $\bar{F}(y) = 1 - F(y)$.

$$f_{MTI}(y) = \frac{\eta(\eta - \log \eta)f(y)}{(\eta - \log \eta \bar{F}(y))^2} \quad ; \quad y \in \mathbb{R}, \eta \in \mathbb{R}^+ \quad (2)$$

Where the CDF and PDF of the baseline distribution are denoted by $F(y)$ and $f(y)$ in the Eqs. (1) and (2) respectively.

Bhat *et al.* (2022) proposed the inverse power Rayleigh distribution (*IPRD*) with PDF and CDF, respectively, given by

$$g(y; \zeta, \delta) = \frac{\zeta}{\delta^2} y^{-(2\zeta+1)} \exp\left(-\frac{y^{-2\zeta}}{2\delta^2}\right); \quad \zeta, \delta > 0 \quad (3)$$

$$G(y; \zeta, \delta) = \exp\left(-\frac{y^{-2\zeta}}{2\delta^2}\right); \quad \zeta, \delta > 0 \quad (4)$$

We now present the *MTI* technique. The CDF of the *MTI-IPRD* can be obtained by using (4) in (1), where $G(y; \zeta, \delta)$ is considered to represent the CDF of the *IPRD* and is given by

$$F(y; \eta, \zeta, \delta) = \begin{cases} \frac{\eta e^{-\frac{y^{-2\zeta}}{2\delta^2}}}{\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}}\right)} & ; y > 0, \eta \neq 1, \eta, \zeta, \delta > 0 \\ \exp\left(-\frac{y^{-2\zeta}}{2\delta^2}\right) & ; \eta = 1, \zeta, \delta > 0 \end{cases} \quad (5)$$

The corresponding PDF of *MTI-IPRD* is obtained as:

$$f(y; \eta, \zeta, \delta) = \begin{cases} \frac{\eta \zeta}{\delta^2} y^{-(2\zeta+1)} \frac{(\eta - \log(\eta)) e^{-\frac{y^{-2\zeta}}{2\delta^2}}}{\left(\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}}\right)\right)^2} & ; y > 0, \eta \neq 1, \eta, \zeta, \delta > 0 \\ \frac{\zeta}{\delta^2} y^{-(2\zeta+1)} \exp\left(-\frac{y^{-2\zeta}}{2\delta^2}\right) & ; \eta = 1, \zeta, \delta > 0 \end{cases} \quad (6)$$

Figure 1 shows the PDF of the *MTI-IPRD* for various parameter values of η , ζ , and δ . This example highlights the versatility of the *MTI-IPRD* by showing PDFs that can have increasing and decreasing density functions, be symmetric, or be right-skewed. These variants demonstrate how well the model may represent a variety of data patterns found in lifespan distributions.

3. Reliability analysis of *MTI-IPRD*

3.1. Reliability function

The Reliability function for *MTI-IPRD* is given as

$$R(y; \eta, \zeta, \delta) = \frac{(\eta - \log(\eta)) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}}\right)}{\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}}\right)} \quad (7)$$

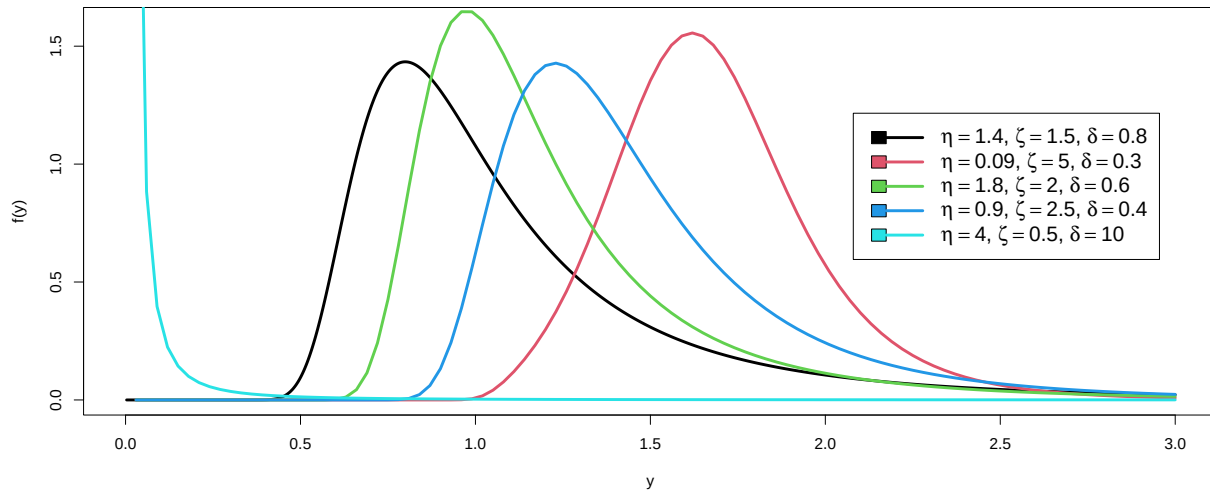


Figure 1: Plots of the pdf of the *MTI-IPRD*

3.2. Hazard rate

The hazard rate for *MTI-IPRD* is obtained as

$$h(y; \eta, \zeta, \delta) = \frac{\eta \zeta y^{-(2\zeta+1)} e^{-\frac{y^{-2\zeta}}{2\delta^2}}}{\delta^2 \left(\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right) \right) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right)} \quad (8)$$

The hazard rate graphs for the *MTI-IPRD* for various parameter values are shown in Figure 2. The proposed distribution is extremely flexible and can take on a number of shapes, as shown in Figure 2.

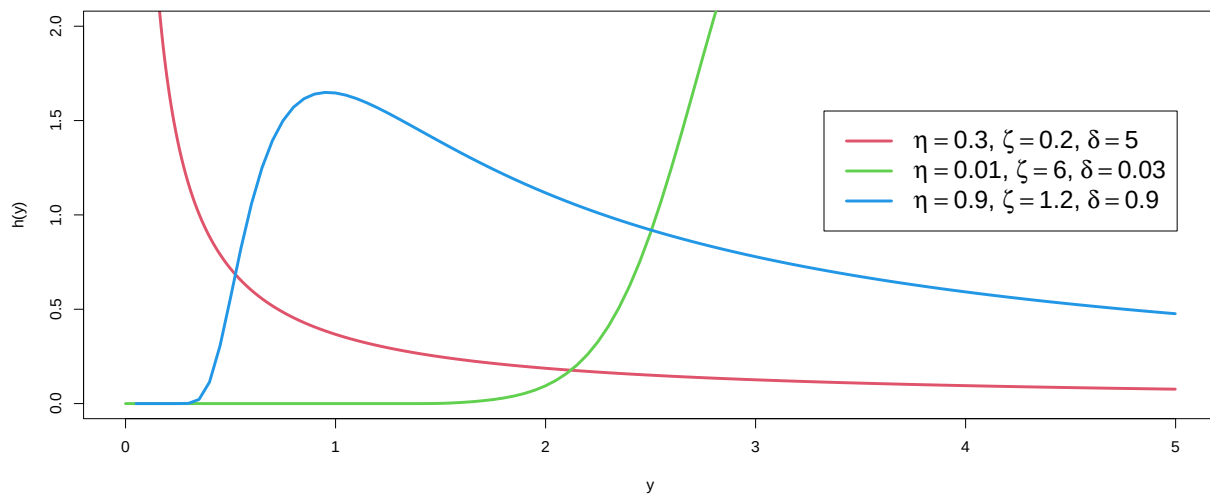


Figure 2: Plots of the hazard rate of the model *MTI-IPRD*

3.3. Reverse hazard rate

The reverse hazard rate for *MTI-IPRD* is obtained as

$$h_r(\eta, \zeta, \delta) = \frac{\zeta(\eta - \log(\eta)) y^{-(2\zeta+1)}}{\delta^2 \left(\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right) \right)} \quad (9)$$

3.4. Cumulative hazard function

The cumulative hazard function for *MTI-IPRD* is determined as

$$\Lambda(y; \eta, \zeta, \delta) = \log \left\{ \frac{\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right)}{\eta (\eta - \log(\eta)) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right)} \right\} \quad (10)$$

3.5. Mills ratio

The mills ratio for *MTI-IPRD* is obtained as

$$M.R = \frac{\eta e^{-\frac{y^{-2\zeta}}{2\delta^2}}}{\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right)} \quad (11)$$

4. Statistical properties of *MTI-IPRD*

The main statistical and distributional aspects of the suggested model are examined in this Section.

4.1. Quantile function

Theorem 1: If $Y \sim MTI-IPRD(\eta, \zeta, \delta)$, then the quantile function of Y is given as

$$y = \left[-2\delta^2 \log \left(\frac{m(\eta - \log(\eta))}{\eta - m \log(\eta)} \right) \right]^{\frac{-1}{2\zeta}} \quad (12)$$

where M is a uniform random variable, $0 < m < 1$.

Proof: Let $F(y; \eta, \zeta, \delta) = m$. The quantile function of the *MTI-IPRD* can be obtained by using (4) in (1) as follows:

$$\frac{\eta e^{-\frac{y^{-2\zeta}}{2\delta^2}}}{\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right)} = m$$

Taking the logarithm on both sides and simplifying further, we obtain the required quantile function as

$$y = \left[-2\delta^2 \log \left(\frac{m(\eta - \log(\eta))}{\eta - m \log(\eta)} \right) \right]^{\frac{-1}{2\zeta}} \quad (13)$$

By setting $m = \frac{1}{4}$, $\frac{1}{2}$, and $\frac{3}{4}$ in (13), respectively, one can derive the first quartile (Q_1), median (Q_2), and third quartile (Q_3). $\square$

4.2. Moments

Theorem 2: If $Y \sim MTI-IPRD(\eta, \zeta, \delta)$, then the r^{th} moment of the $MTI-IPRD$ about the origin is given as

$$\mu'_r = \frac{(\eta - \log(\eta))(2\delta^2)^{\frac{-r}{2\zeta}}}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\Gamma\left(1 - \frac{r}{2\eta}\right)}{(q+1)^{1-\frac{r}{2\eta}}}; \quad r < 2\eta \quad (14)$$

Proof: Utilising the subsequent series representation, the r^{th} moment of the $MTI-IPRD$ is found.

$$(1-v)^{-2} = \sum_{q=0}^{\infty} (q+1)v^q; \quad |v| < 1, \quad (15)$$

$$(1-v)^p = \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} v^q \quad (16)$$

The r^{th} moment of Y is given by

$$\mu'_r = \int_0^{\infty} y^r \frac{\eta \zeta}{\delta^2} y^{-(2\zeta+1)} \frac{(\eta - \log(\eta)) e^{-\frac{y-2\zeta}{2\delta^2}}}{\left(\eta - \log(\eta) \left(1 - e^{-\frac{y-2\zeta}{2\delta^2}} \right) \right)^2} dy. \quad (17)$$

By substituting $\frac{y-2\zeta}{2\delta^2} = z$ in (17), we get

$$\mu'_r = \frac{(\eta - \log(\eta))(2\delta^2)^{\frac{-r}{2\zeta}}}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \left(\int_0^{\infty} z^{1-\frac{r}{2\eta}-1} e^{-(q+1)z} dz \right) \quad (18)$$

The final expression is

$$\mu'_r = \frac{(\eta - \log(\eta))(2\delta^2)^{\frac{-r}{2\zeta}}}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\Gamma\left(1 - \frac{r}{2\eta}\right)}{(q+1)^{1-\frac{r}{2\eta}}}; \quad r < 2\eta \quad (19)$$

Setting $r = 1$ in (19), the mean of the $MTI-IPRD$ is computed as

$$\mu'_1 = \frac{(\eta - \log(\eta))(2\delta^2)^{-\frac{1}{2\zeta}}}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\Gamma\left(1 - \frac{1}{2\eta}\right)}{(q+1)^{1-\frac{1}{2\eta}}}; \quad \eta > \frac{1}{2} \quad (20)$$

Similarly, substituting $r = 2, 3, 4$ in (19), the second, third, and fourth moments about the origin of the $MTI-IPRD$ are obtained, respectively. $\square$

4.3. Moment generating function of *MTI-IPRD*

Theorem 3: If $Y \sim \text{MTI-IPRD}(\eta, \zeta, \delta)$, then the moment generating function, $M_Y(t)$, is

$$M_Y(t) = \frac{(\eta - \log(\eta))}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} \sum_{r=0}^{\infty} \frac{t^r}{r!} (-1)^q \binom{p}{q} (p+1) (2\delta^2)^{\frac{-r}{2\zeta}} \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\Gamma\left(1 - \frac{r}{2\eta}\right)}{(q+1)^{1-\frac{r}{2\eta}}}; \quad r < 2\eta \quad (21)$$

Proof: The *MTI-IPRD* moment generating function is described as

$$M_Y(t) = \int_0^{\infty} e^{ty} f(y; \eta, \zeta, \delta) dy \quad (22)$$

By utilizing the series representation of e^{ty} , we have

$$M_Y(t) = \sum_{r=0}^{\infty} \frac{t^r}{r!} E(Y^r) \quad (23)$$

Substituting (19) in (23), we get

$$M_Y(t) = \frac{(\eta - \log(\eta))}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} \sum_{r=0}^{\infty} \frac{t^r}{r!} (-1)^q \binom{p}{q} (p+1) (2\delta^2)^{\frac{-r}{2\zeta}} \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\Gamma\left(1 - \frac{r}{2\eta}\right)}{(q+1)^{1-\frac{r}{2\eta}}}; \quad r < 2\eta \quad (24)$$

□

4.4. Conditional moments and associated measures

Lemma 1: Assume we have a random variable $Y \sim \text{MTI-IPRD}(\eta, \zeta, \delta)$ with PDF given in (6), and let $\phi_r(z) = \int_0^z y^r f(y; \eta, \zeta, \delta) dy$ denote the r^{th} incomplete moment, then we have

$$\phi_r(z) = \frac{(\eta - \log(\eta))(2\delta^2)^{\frac{-r}{2\zeta}}}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\gamma\left(\left(1 - \frac{r}{2\eta}\right), \frac{q+1}{2\delta^2} z^{-2\zeta}\right)}{(q+1)^{1-\frac{r}{2\eta}}}; \quad r < 2\eta \quad (25)$$

where $\gamma(m, n) = \int_0^n z^{m-1} e^{-z} dz$ represents the lower incomplete gamma function.

Proof: By utilizing the PDF of *MTI-IPRD* provided by (6), we have

$$\phi_r(z) = \int_0^z y^r f(y; \eta, \zeta, \delta) dy = \frac{\eta \zeta (\eta - \log(\eta))}{\delta^2} \int_0^z y^r y^{-(2\zeta+1)} \frac{e^{-\frac{y-2\zeta}{2\delta^2}}}{\left(\eta - \log(\eta) \left(1 - e^{-\frac{y-2\zeta}{2\delta^2}} \right) \right)^2} dy \quad (26)$$

On simplification, we obtain the r^{th} incomplete moment as

$$\phi_r(z) = \frac{(\eta - \log(\eta))(2\delta^2)^{\frac{-r}{2\zeta}}}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\gamma\left(\left(1 - \frac{r}{2\eta}\right), \frac{q+1}{2\delta^2} z^{-2\zeta}\right)}{(p+1)^{1-\frac{r}{2\eta}}}; \quad r < 2\eta \quad (27)$$

Setting $r = 1$ in (27) will provide the first incomplete moment as given by

$$\phi_1(z) = \frac{(\eta - \log(\eta))(2\delta^2)^{\frac{-1}{2\zeta}}}{\eta} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{\gamma \left(\left(1 - \frac{1}{2\eta}\right), \frac{q+1}{2\delta^2} z^{-2\zeta} \right)}{(q+1)^{1-\frac{1}{2\eta}}} ; \eta > \frac{1}{2} \quad (28)$$

□

4.4.1. Lorenz and Bonferroni inequality curves

An important use of the first incomplete moment is found in the curves of Lorenz and Bonferroni inequality. For a specific probability distribution, they are defined as follows:

$$L_w = \frac{1}{E(Y)} \int_0^t y f(y; \eta, \zeta, \delta) dy = \frac{\phi_1(t)}{E(Y)}$$

$$L_w = \frac{\sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \gamma \left(\left(1 - \frac{1}{2\zeta}\right), \frac{q+1}{2\delta^2} t^{-2\zeta} \right)}{\sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \Gamma \left(1 - \frac{1}{2\zeta}\right)} ; \quad \eta > \frac{1}{2}$$

Similarly,

$$B_w = \frac{1}{wE(Y)} \int_0^t y f(y; \eta, \zeta, \delta) dy = \frac{\phi_1(t)}{wE(Y)}$$

$$B_w = \frac{\sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \gamma \left(\left(1 - \frac{1}{2\zeta}\right), \frac{q+1}{2\delta^2} t^{-2\zeta} \right)}{w \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \Gamma \left(1 - \frac{1}{2\zeta}\right)} ; \quad \eta > \frac{1}{2}$$

4.4.2. r^{th} Conditional moment and r^{th} reversed conditional moment of *MTI-IPRD*

The r^{th} conditional moment of the *MTI-IPRD* is calculated by

$$E[Y^r | y > t] = \frac{1}{R(t)} \int_t^{\infty} y^r f(y; \eta, \zeta, \delta) dy = \frac{1}{R(t)} [E(Y^r) - \phi_r(t)]$$

Inserting the value's of (7), (19), and (27), we obtain

$$E[Y^r | y > t] = \frac{\left(\eta - \log(\eta) \left(1 - e^{-\frac{t^{-2\zeta}}{2\delta^2}}\right) \right)}{\eta \left(1 - e^{-\frac{t^{-2\zeta}}{2\delta^2}}\right)} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{(2\delta^2)^{\frac{-r}{2\zeta}}}{(q+1)^{1-\frac{r}{2\zeta}}} \times$$

$$\left[\Gamma \left(1 - \frac{r}{2\zeta}\right) - \gamma \left(\left(1 - \frac{r}{2\zeta}\right), \frac{q+1}{2\delta^2} t^{-2\zeta} \right) \right] ; \quad r < 2\eta \quad (29)$$

Likewise, the *MTI-IPRD*'s r^{th} reversed conditional moment is determined by

$$E[Y^r | y \leq t] = \frac{1}{F(t)} \int_0^t y^r f(y; \eta, \zeta, \delta) dy = \frac{\phi_r(t)}{F(t)}$$

$$E[Y^r | y \leq t] = \frac{(\eta - \log(\eta)) \left(\eta - \log(\eta) \left(1 - e^{-\frac{t-2\zeta}{2\delta^2}} \right) \right)}{\eta^2 e^{-\frac{t-2\zeta}{2\delta^2}}} \times$$

$$\sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{(2\delta^2)^{\frac{-r}{2\zeta}}}{(q+1)^{1-\frac{r}{2\zeta}}} \gamma \left(\left(1 - \frac{r}{2\zeta} \right), \frac{q+1}{2\delta^2} t^{-2\zeta} \right); r < 2\eta$$

4.4.3. Mean residual life (textitMRL) and mean waiting time (MWT)

The *MRL*, say $\mu(t)$, of *MTI-IPRD* can be acquired in the manner described below:

$$\mu(t) = \frac{1}{R(t)} \left[E(Y) - \int_0^t y f(y; \eta, \zeta, \delta) dy \right] - t = \frac{1}{R(t)} [E(Y) - \phi_1(t)] - t$$

Following the insertion of the value's of (19), (20), and (28), we derive the necessary mean residual life expression as

$$\mu(t) = \frac{\left(\eta - \log(\eta) \left(1 - e^{-\frac{t-2\zeta}{2\delta^2}} \right) \right)}{\eta \left(1 - e^{-\frac{t-2\zeta}{2\delta^2}} \right)} \times$$

$$\sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{(2\delta^2)^{\frac{-1}{2\zeta}}}{(q+1)^{1-\frac{1}{2\zeta}}} \left[\Gamma \left(1 - \frac{1}{2\zeta} \right) - \gamma \left(\left(1 - \frac{1}{2\zeta} \right), \frac{q+1}{2\delta^2} t^{-2\zeta} \right) \right] - t$$

(30)

The *MWT* of Y , say $\bar{\mu}(t)$, is determined by

$$\bar{\mu}(t) = t - \frac{1}{F(t)} \int_0^t y f(y; \eta, \zeta, \delta) dy = t - \frac{\phi_1(t)}{F(t)}$$

$$\bar{\mu}(t) = t - \frac{(\eta - \log(\eta)) \left(\eta - \log(\eta) \left(1 - e^{-\frac{t-2\zeta}{2\delta^2}} \right) \right)}{\eta^2 e^{-\frac{t-2\zeta}{2\delta^2}}}$$

$$\times \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{p}{q} (p+1) \left(\frac{\log(\eta)}{\eta} \right)^p \frac{(2\delta^2)^{\frac{-1}{2\zeta}}}{(q+1)^{1-\frac{1}{2\zeta}}} \gamma \left(\left(1 - \frac{1}{2\zeta} \right), \frac{q+1}{2\delta^2} t^{-2\zeta} \right)$$

4.5. Renyi Entropy

Theorem 4: If $X \sim MTI-IPRD(\eta, \zeta, \delta)$, then the $MTI-IPRD$'s Renyi entropy is provided by

$$RE_X(v) = \frac{1}{1-v} \log \left\{ \left(\frac{\zeta(\eta - \log(\eta))}{\eta\delta^2} \right)^v \frac{1}{2\eta} \right\} \\ + \frac{1}{1-v} \log \left\{ \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{2v}{p} \binom{p}{q} \left(\frac{\log(\eta)}{\eta} \right)^p \left(\frac{2\delta^2}{v+q} \right)^{\frac{(2\eta+1)v-1}{2\eta}} \Gamma \left(\frac{(2\eta+1)v-1}{2\eta} \right) \right\}$$

Proof: The Rényi entropy, say $RE_X(v)$, of $MTI-IPRD$ can be defined as

$$RE_X(v) = \frac{1}{1-v} \log \left(\int_0^{\infty} f^v(y) dy \right); \quad v > 0, \quad v \neq 1 \quad (31)$$

Substituting (6) in (31), we get

$$RE_X(v) = \frac{1}{1-v} \log \left\{ \left(\frac{\zeta(\eta - \log(\eta))}{\eta\delta^2} \right)^v \int_0^{\infty} y^{-v(2\zeta+1)} \frac{e^{-v\frac{y^{-2\zeta}}{2\delta^2}}}{\left(\eta - \log(\eta) \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta^2}} \right) \right)^{2v}} dy \right\} \quad (32)$$

Using (16), the Renyi entropy of $MTI-IPRD$ is given by

$$RE_X(v) = \frac{1}{1-v} \log \left\{ \left(\frac{\zeta(\eta - \log(\eta))}{\eta\delta^2} \right)^v \frac{1}{2\eta} \right\} \\ + \frac{1}{1-v} \log \left\{ \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} (-1)^q \binom{2v}{p} \binom{p}{q} \left(\frac{\log(\eta)}{\eta} \right)^p \left(\frac{2\delta^2}{v+q} \right)^{\frac{(2\eta+1)v-1}{2\eta}} \Gamma \left(\frac{(2\eta+1)v-1}{2\eta} \right) \right\}$$

□

4.6. Stress Strength Reliability

Theorem 5: When $Y_1 \sim MTI-IPRD(\eta_1, \zeta, \delta_1)$ and $Y_2 \sim MTI-IPRD(\eta_2, \zeta, \delta_2)$, where Y_1 and Y_2 denote the independent random variables for stress and strength, respectively, then the $MTI-IPRD$'s stress strength reliability $P(Y_1 > Y_2)$ is

$$SSR = \frac{\delta_2^2(\eta_1 - \log(\eta_1))}{\eta_1} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} \sum_{r=0}^{\infty} \sum_{s=0}^{\infty} (-1)^{r+s} \binom{p}{q} \binom{q}{s} \left(\frac{\log(\eta_1)}{\eta_1} \right)^p \left(\frac{\log(\eta_2)}{\eta_2} \right)^q \\ \frac{(p+1)}{(s+1)\delta_1^2 + (r+1)\delta_2^2}$$

Proof: The stress strength reliability $P(Y_1 > Y_2)$ of the *MTI-IPRD*, say *SSR*, can be acquired as

$$SSR = \int_0^{\infty} f_1(y) F_2(y) dy. \quad (33)$$

Using (5) and (6) in (33), *SSR* can be acquired as

$$\begin{aligned} SSR = & \frac{\zeta(\eta_1 - \log(\eta_1))}{\delta_1^2 \eta_1} \int_0^{\infty} y^{-(2\zeta+1)} e^{-\frac{y^{-2\zeta}}{2\delta_1^2}} e^{-\frac{y^{-2\zeta}}{2\delta_2^2}} \left(1 - \frac{\log(\eta_1)}{\eta_1} \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta_1^2}}\right)\right)^{-2} \\ & \times \left(1 - \frac{\log(\eta_2)}{\eta_2} \left(1 - e^{-\frac{y^{-2\zeta}}{2\delta_2^2}}\right)\right)^{-1} dy. \end{aligned}$$

Using the same method as in (19), we obtain the stress strength reliability for *MTI-IPRD* in its final expression as

$$\begin{aligned} SSR = & \frac{\delta_2^2(\eta_1 - \log(\eta_1))}{\eta_1} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} \sum_{r=0}^{\infty} \sum_{s=0}^{\infty} (-1)^{r+s} \binom{p}{q} \binom{q}{s} \left(\frac{\log(\eta_1)}{\eta_1}\right)^p \left(\frac{\log(\eta_2)}{\eta_2}\right)^q \\ & \frac{(p+1)}{(s+1)\delta_1^2 + (r+1)\delta_2^2} \end{aligned}$$

□

5. Estimation

5.1. Maximum likelihood estimation

Assuming a random sample $y_1, y_2, \dots, y_n$ from *MTI-IPRD* with parameters $\eta, \zeta, \delta > 0$, the likelihood function of *MTI-IPRD* may be expressed as follows:

$$\begin{aligned} l = & n \log(\zeta) + n \log(\eta(\eta - \log(\eta))) - 2n \log(\delta) - (2\zeta + 1) \sum_{a=1}^n \log(y_a) - \frac{1}{2\delta^2} \sum_{a=1}^n y_a^{-2\zeta} \\ & - 2 \sum_{a=1}^n \log \left(\eta - \log(\eta) \left(1 - e^{-\frac{y_a^{-2\zeta}}{2\delta^2}}\right) \right) \end{aligned} \quad (34)$$

The partial differentiation (34) of η , ζ , and δ leads to their *MLEs* by equating to zero

and partially differentiating with regard to the required parameters. As we have

$$\frac{\partial l}{\partial \eta} = \frac{n(2\eta - \log(\eta) - 1)}{\eta(\eta - \log(\eta))} - \frac{2}{\eta} \sum_{a=1}^n \left[\frac{e^{-\frac{y_a^{-2\zeta}}{2\delta^2}} + \eta - 1}{\eta - \log(\eta) \left(1 - e^{-\frac{y_a^{-2\zeta}}{2\delta^2}}\right)} \right] = 0 \quad (35)$$

$$\frac{\partial l}{\partial \zeta} = \frac{n}{\zeta} + \frac{\sum_{a=1}^n y_a^{-2\zeta} \log(y_a)}{\delta^2} - 2 \sum_{a=1}^n \log(y_i) - 2 \log(\eta) \sum_{a=1}^n \left[\frac{y_a^{-2\zeta} \log(y_a) e^{-\frac{y_a^{-2\zeta}}{2\delta^2}}}{\eta - \log(\eta) \left(1 - e^{-\frac{y_a^{-2\zeta}}{2\delta^2}}\right)} \right] = 0 \quad (36)$$

$$\frac{\partial l}{\partial \delta} = \frac{1}{\delta^3} \sum_{a=1}^n y_a^{-2\zeta} - \frac{2n}{\delta} + \frac{2 \log(\eta)}{\delta^3} \sum_{a=1}^n \left[\frac{y_a^{-2\zeta} e^{-\frac{y_a^{-2\zeta}}{2\delta^2}}}{\eta - \log(\eta) \left(1 - e^{-\frac{y_a^{-2\zeta}}{2\delta^2}}\right)} \right] = 0 \quad (37)$$

The solutions to the nonlinear equations 35, 36, and 37 are not in closed form. To estimate the parameters, these equations will be solved using the *R* software.

6. Simulation study

Here, *R* software is utilised to perform a simulation study to examine how *MLEs* behave for various sample sizes. We generate random samples in the following sizes: 25, 75, 150, 300, and 500 using *MTI-IPRD*. The procedure is then carried out a thousand times using *R* software. Various combinations of parameters are chosen in relation to the standard order (η, ζ, δ) , such as $(0.55, 0.35, 0.25)$ and $(0.25, 0.85, 0.65)$. For every scenario, mean squared errors (*MSEs*), bias, and average *MLE* values were calculated. Tables 1 and 2 display the outcomes. The estimations are consistent and fairly close to the real parameter values, as these Tables show. Furthermore, in every scenario, the *MSE* falls as sample size increases.

Table 1: *MLE*, *Bias*, and *MSE* for the parameters η , ζ , and δ

Sample size n	Parameters			MLE			Bias			MSE		
	η	ζ	δ	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$
25	0.55	0.35	0.25	1.00576	0.37845	0.25182	0.70285	0.07355	0.05670	1.10630	0.00898	0.00710
75				0.79346	0.36203	0.25491	0.45981	0.04886	0.03320	0.58870	0.00370	0.00233
150				0.69982	0.35489	0.25209	0.31398	0.03863	0.02263	0.30394	0.00221	0.00088
300				0.63625	0.35171	0.25137	0.21134	0.02780	0.01704	0.13579	0.00121	0.00052
500				0.58196	0.35025	0.25055	0.13613	0.02123	0.01277	0.04363	0.00072	0.00028

Table 2: *MLE, Bias, and MSE for the parameters η , ζ , and δ*

Sample size	Parameters			MLE			Bias			MSE		
n	η	ζ	δ	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$
25	0.25	0.85	0.65	0.62927	0.89377	0.68999	0.46417	0.14567	0.24876	0.69750	0.03404	0.12298
75				0.37205	0.88736	0.66497	0.20624	0.10320	0.17098	0.15497	0.01667	0.05673
150				0.29298	0.86108	0.66084	0.11332	0.07061	0.12175	0.03297	0.00797	0.02595
300				0.27159	0.85840	0.65621	0.07223	0.04908	0.08378	0.00980	0.00388	0.01193
500				0.25904	0.85237	0.65178	0.05570	0.03895	0.06778	0.00541	0.00238	0.00802

7. Applications to real life data

In this section, three actual data sets have been used to assess the newly created distribution's efficacy. As the new distribution is compared to alpha-power exponentiated inverse *RD* (*APEIRD*) Ali *et al.* (2021), inverse *RD* (*IPRD*) Bhat *et al.* (2022), modified *IRD* (*MIRD*) Khan (2014), and transmuted *IRD* (*TIRD*) Ahmad *et al.* (2014), it turns out that the newly created distribution provides a suitable fit. *AIC*, *BIC*, *AICC*, and *KS* are only a few of the metrics used to compare the fitted models. The *p*-value for each model is also given. It is considered better to have a distribution with a high *p*-value and lower *AIC*, *BIC*, *AICC*, and *KS*.

Data set 1 : The first data set consists of thirty observations for the rainfall (in inches) of March in Minneapolis/St. Paul Hinkley (1977). The data are as follows:

0.77 , 1.74 ,0.81, 1.20, 1.95 ,1.20, 0.47, 1.43 ,3.37 ,2.20 ,3.00 ,3.09 ,1.51 ,2.10 ,0.52, 1.62, 1.31, 0.32 ,0.59, 0.81, 2.81, 1.87, 1.18 ,1.35, 4.75 ,2.48, 0.96 ,1.89, 0.90 and 2.05

Data set 2 : The second data set consists of average annual percent change in private health insurance premiums. The data has been previously used by Malik and Ahmad (2024). and is given as follows:

14.4, 14.0, 15.4, 9.4, 11.7, 15.0, 24.9, 20.7, 12.5,14.9, 12.6, 16.7, 13.8, 11.0, 12.9, 10.1, 1.9, 8.5, 16.5, 15.3, 13.3, 9.8, 8.4, 7.9, 3.7,5.1, 4.6, 4.4, 5.4, 6.1, 8.0, 10.0, 11.2, 10.1, 6.4, 6.7, 5.7, 5.8.

Data set 3: The third data set consists of 150 observations and is related to the Reddit advertising data. The data has been previously used by Shen *et al.* (2022). and is given as follows:

11.340, 6.296, 5.136, 7.292, 6.700, 3.648, 7.272, 5.140, 2.980, 6.336, 5.128, 7.564, 6.180, 5.588, 8.760, 11.560, 5.192, 10.032, 6.660, 7.932, 7.536, 4.436, 17.580, 9.836, 5.580, 6.092, 7.912, 6.760, 10.096, 5.948, 11.156, 6.936, 5.424, 8.532, 4.980, 6.536, 13.012, 6.668, 5.540, 11.184, 7.052, 9.336, 10.624, 7.148, 4.892, 6.584, 4.436, 11.696, 7.868, 4.040, 6.748, 5.336, 9.056, 11.496, 11.548, 12.392, 3.636, 7.380, 8.940, 9.068, 4.536, 13.060, 8.940, 7.900, 9.972, 5.740, 5.560, 6.136, 10.904, 7.960, 9.380, 6.484, 3.512, 4.792, 8.980, 5.512, 2.392, 5.336, 3.440, 4.580, 6.704, 7.296, 4.600, 5.592, 9.292, 7.816, 7.068, 8.492,7.556, 8.836, 5.960, 3.696, 9.816, 10.908, 5.636, 8.536, 6.260, 7.912, 12.492, 8.880, 6.188, 11.900, 7.692, 7.496, 10.340, 9.636, 3.784, 5.068, 2.940, 9.992, 7.252, 10.956, 7.512, 8.108, 7.796, 6.928, 6.236, 4.924, 8.056, 3.468, 7.904, 3.780, 5.912, 7.756, 9.900, 5.472, 3.956, 5.044, 12.676, 5.376.

When compared to other competing models, the *MTI-IPRD* achieves the lowest values

for AIC , BIC , and $AICC$, as shown in Tables 6, 7, and 8. As a result, the $MTI-IPRD$ performs better than the other competing models listed as well as the base model. Additional evidence for these conclusions may be found in Figures 3, 4 and 5.

8. Conclusion

In this paper, we present a new model for data analysis with real support, which extends the $IPRD$ and we call it $MTI-IPRD$. More modelling flexibility for real-world data is the primary objective of generalising a standard distribution. Key statistical properties of the proposed model have been inferred. The new distribution's hazard rate function exhibits more flexibility and complex forms. The MLE performs well, as demonstrated by a simulation analysis that shows it to be accurate and consistent. To estimate the parameters, the MLE approach is applied. In comparison to other competing distributions, the suggested model might offer a superior fit given its application. In the field of statistics, we believe that the suggested method will find wide usage.

Acknowledgements

I am indeed grateful to the Editors for their guidance and counsel. I am very grateful to the reviewer for valuable comments and suggestions of generously listing many useful references.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Abd Elfattah, A., Hassan, A. S., and Ziedan, D. (2006). Efficiency of maximum likelihood estimators under different censored sampling schemes for Rayleigh distribution. *Interstat*, **1**, 1–16.
- Ahmad, A., Ahmad, S., and Ahmed, A. (2014). Transmuted inverse Rayleigh distribution: A generalization of the inverse Rayleigh distribution. *Mathematical Theory and Modeling*, **4**, 90–98.
- Ali, M., Khalil, A., Ijaz, M., and Saeed, N. (2021). Alpha-power exponentiated inverse Rayleigh distribution and its applications to real and simulated data. *PloS one*, **16**, e0245253.
- Bhat, A., Ahmad, S., and Mudasar, S. (2022). A novel extension of power Rayleigh distribution with properties. *International Journal of Statistics and Reliability Engineering*, **9**, 518–529.

- Bhat, A., Ahmad, S. P., Almetwally, E. M., Yehia, N., Alsadat, N., and Tolba, A. H. (2023). The odd Lindley power Rayleigh distribution: properties, classical and Bayesian estimation with applications. *Scientific African*, **20**, e01736.
- Bhat, A. A. and Ahmad, S. P. (2020). A new generalization of Rayleigh distribution: Properties and applications. *Pakistan Journal of Statistics*, **36**.
- Gharraph, M. (1993). Comparison of estimators of location measures of an inverse Rayleigh distribution. *The Egyptian Statistical Journal*, **37**, 295–309.
- Hinkley, D. (1977). On quick choice of power transformation. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **26**, 67–69.
- Khan, M. S. (2014). Modified inverse Rayleigh distribution. *International Journal of Computer Applications*, **87**, 28–33.
- Kundu, D. and Raqab, M. Z. (2005). Generalized Rayleigh distribution: different methods of estimations. *Computational Statistics & Data Analysis*, **49**, 187–200.
- Lone, M. A., Dar, I. H., and Jan, T. (2022). An innovative method for generating distributions: Applied to weibull distribution. *Journal of Scientific Research*, **66**.
- Malik, A. S. and Ahmad, S. (2024). Generalized inverted Kumaraswamy-Rayleigh distribution: Properties and application. *Journal of Modern Applied Statistical Methods*, **23**.
- Merovci, F. (2013). Transmuted Rayleigh distribution. *Austrian Journal of statistics*, **42**, 21–31.
- Merovci, F. (2014). Transmuted generalized Rayleigh distribution. *Journal of Statistics Applications & Probability*, **3**, 9.
- Mir, A. A. and Ahmad, S. (2024). Modeling and analysis of sine power Rayleigh distribution: Properties and applications. *Reliability: Theory & Applications*, **19**, 703–716.
- Rayleigh, L. (1880). Xii. on the resultant of a large number of vibrations of the same pitch and of arbitrary phase. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, **10**, 73–78.
- Shen, Z., Alrumayh, A., Ahmad, Z., Abu-Shanab, R., Al-Mutairi, M., and Aldallal, R. (2022). A new generalized Rayleigh distribution with analysis to big data of an online community. *Alexandria Engineering Journal*, **61**, 11523–11535.
- Soliman, A., Amin, E. A., and Abd-El Aziz, A. A. (2010). Estimation and prediction from inverse Rayleigh distribution based on lower record values. *Applied Mathematical Sciences*, **4**, 3057–3066.
- Surles, J. and Padgett, W. (2001). Inference for reliability and stress-strength for a scaled burr type x distribution. *Lifetime Data Analysis*, **7**, 187–200.
- Voda, V. G. (1972). On the inverse Rayleigh distributed random variable. *Reports of Statistical Application Research*, **19**, 13–21.

ANNEXURE

Table 3: MLEs of *MTI-IPRD* and competing models with corresponding SE (given in parentheses) for data Set 1

Model	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$	$\hat{\lambda}$
MTI-IPRD	0.05897 (0.10150)	1.35728 (0.22783)	3.04861 (3.23500)	
APEIRD	35.35513 (67.35165)	0.59188 (43.43911)	0.61802 (45.35775)	
IPRD	-	0.77479 (0.10132)	0.69834 (0.06737)	
MIRD	0.35977 (0.37454)	0.58811 (0.29747)		
TIRD	-	-	0.85914 (0.21365)	0.00100 (0.40164)

Table 4: MLEs of *MTI-IPRD* and competing models with corresponding SE (given in parentheses) for data Set 2

Model	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$	$\hat{\lambda}$
MTI-IPRD	0.01321 (0.01977)	1.68518 (0.22194)	0.27327 (0.25462)	-
APEIRD	168.36874 (375.38918)	4.11696 (1498.48126)	3.12876 (1138.8015)	-
IPRD	-	0.82199 (0.08939)	0.14022 (0.02346)	-
MIRD	0.00100 (2.949399)	44.25819 (17.271966)		
TIRD	-	-	44.28455 (13.91961)	0.00100 (0.56670)

Table 5: MLEs of *MTI-IPRD* and competing models with corresponding SE (given in parentheses) for data Set 3

Model	$\hat{\eta}$	$\hat{\zeta}$	$\hat{\delta}$	$\hat{\lambda}$
MTI-IPRD	0.00266 (0.00040)	2.09448 (0.19230)	0.57909 (0.23767)	
APEIRD	0.06374 (0.03970)	8.10089 (107.03792)	7.90391 (104.43518)	
IPRD	-	1.30989 (0.08066)	0.07160 (0.00950)	
MIRD	0.00100 (2.41996)	36.17014 (9.48393)		
TIRD	-	-	54.68699 (11.3787766)	1.00000 (0.69597)

Table 6: Comparison of *MTI-IPRD* and competitive models for data set 1

Model	$-2ll$	AIC	BIC	AICC	K-S	p-value
MTI-IPRD	77.5967	83.5967	87.8003	84.5198	0.0737	0.9968
APEIRD	80.8098	86.8098	91.0134	87.7329	0.1224	0.7593
IPRD	83.8340	87.8340	90.6364	88.2784	0.1523	0.4893
MIRD	87.2598	91.2598	94.0622	91.7043	0.1832	0.2662
TIRD	88.28373	92.28373	95.08613	92.72818	0.2398	0.0634

Table 7: Comparison of *MTI-IPRD* and competitive models for data set 2

Model	$-2ll$	AIC	BIC	AICC	K-S	p-value
MTI-IPRD	228.8583	234.8583	239.7711	235.5642	0.0901	0.9172
APEIRD	236.2966	242.2966	247.2094	243.0025	0.1551	0.3196
IPRD	243.6060	247.6060	250.8812	247.9488	0.1673	0.2377
MIRD	247.3570	251.3570	254.6322	251.6999	0.2111	0.0674
TIRD	247.3682	251.3682	254.6434	251.7110	0.2113	0.0671

Table 8: Comparison of *MTI-IPRD* and competitive models for data set 3

Model	$-2ll$	AIC	BIC	AICC	K-S	p-value
MTI-IPRD	616.3947	622.3947	630.9973	622.5851	0.0577	0.7781
APEIRD	639.0906	645.0906	653.6932	645.2810	0.0875	0.2724
IPRD	640.2807	644.2807	650.0158	644.3752	0.1020	0.1331
MIRD	656.4957	660.4957	666.2308	660.5902	0.1834	0.0003
TIRD	625.8180	629.8180	635.5531	629.9125	0.0748	0.4599

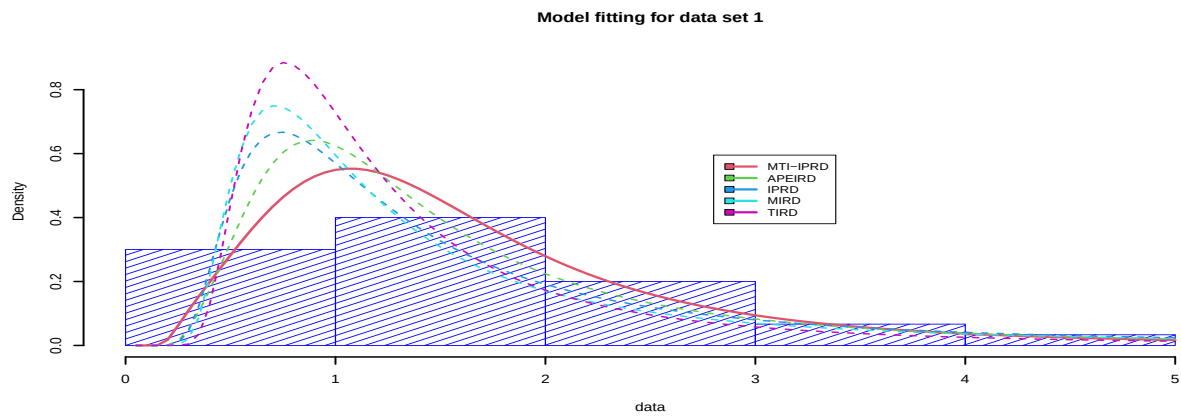


Figure 3: Fitted density plots for data set 1

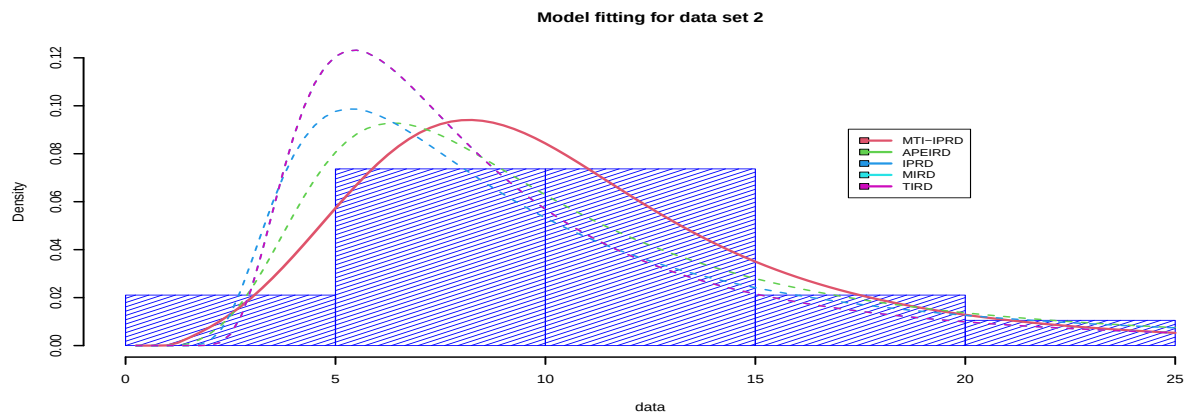


Figure 4: Fitted density plots for data set 2

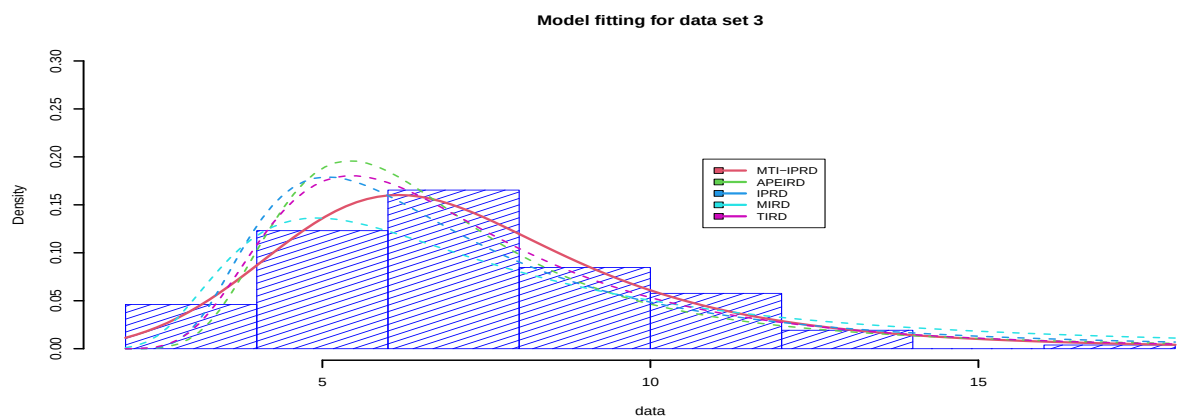
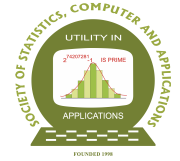


Figure 5: Fitted density plots for data set 3



VTPC: A Novel Virtual Tree Approach for Efficient Population Count in Binary Sequences

Samarth Godara, Prakash Kumar, Deepak Singh and Raju Kumar

ICAR-Indian Agricultural Statistics Research Institute, New Delhi-110012, India

Received: 04 October 2024; Revised: 03 December 2024; Accepted: 06 December 2024

Abstract

The task of counting the number of ones in a bit sequence, known as the “Population Count,” is essential in various domains such as cryptography, database information retrieval, and machine learning. Traditional methods, whether hardware or software-based, often require direct interaction with the binary representation of data, which can be computationally intensive. This paper introduces a new technique, the Virtual Tree for Population Count Method (VTPC), which can calculate the population count of a number by leveraging a virtual tree structure without converting the number to its binary form. This method significantly reduces time complexity compared to existing algorithms. Our approach is evaluated against traditional methods, demonstrating its efficiency and accuracy in computing the population count. By eliminating the need to convert numbers to their binary form, VTPC streamlines processes and can lead to faster, more scalable implementations in various high-performance computing environments.

Key words: Population count; Virtual tree method; Bit counting; Cryptography; Algorithm efficiency.

AMS Subject Classifications: 62K05, 05B05

1. Introduction

Bit-counting, a crucial computer operation, involves determining the number of ‘1’s in a bit sequence. This operation is fundamental for various applications that utilize bitsets or bitmaps, necessitating frequent population counts. The term “population count” (often abbreviated as ‘popcount’) also refers to the cardinality of a bitset, indicating the number of ‘1’ bits representing elements in the set. The efficiency of bit-counting operations significantly impacts the performance of file systems, databases, machine learning algorithms, search engines, and information security systems.

In cryptography, population count functions are employed for randomness tests and generating pseudo-random permutations, aiding in the identification of duplicated web pages. Additionally, they are widely used in bioinformatics, ecology, chemistry, and various other

fields. As noted by Mula *et al.* (2018), recent developments have seen population count instructions integrated into fast sorting algorithms by researchers like Gueron and Krasnov.

Population count techniques are broadly classified into two categories: hardware implementations and software-based techniques. Hardware implementations involve specialized circuitry to execute popcount functions efficiently, whereas software-based techniques encompass a range of algorithms that do not require additional hardware. This paper focuses on proposing a novel software-based technique for bit-counting, which predicts the population count of a number directly from its decimal representation without converting it into binary form. The proposed method aims to improve the time complexity of population count operations compared to existing algorithms. The following is a detailed overview of some prominent existing methods for population count.

1. Naïve Method: As shown in Rosen (2019), the naïve approach involves examining each bit in a binary sequence individually. This method iterates through all bits of an integer, checking if each bit is set, and increments a count if it is. For a 32-bit integer, this approach may require up to 32 iterations in the worst case, making it relatively inefficient. The time complexity is $O(M)$, where M is the number of bits in the integer.
2. Table Lookup: A more efficient method involves precomputing the population counts for all possible byte values (0 to 255) and storing them in a lookup table, as explained by Fateman (1989). Each byte of the input number is then checked against this table, and the results are summed. This method reduces the number of required operations significantly. For larger chunks of data, a 16-bit or even larger lookup table can be used, though this increases memory usage. The time complexity is $O(M/L)$, where L is the number of bits processed in each lookup operation.
3. Divide and Conquer Strategy: Knuth (2014) introduced the divide and conquer method, that breaks down the problem into smaller parts, summing bits in a hierarchical manner. Initially, pairs of bits are summed, then these results are summed in pairs, and so on. This technique leverages the hierarchical nature of binary numbers and can perform the population count in $\log_2(M)$ steps, where M is the number of bits. This method is efficient for systems that can perform parallel operations on bits.
4. Carry-Save Adder (CSA): The CSA technique uses a bitwise parallel carry-save adder, a hardware construct often used in binary multipliers, as shown by Mula *et al.* (2018). This approach processes elements in groups, reducing three words to two and then applying the population count operation to these two words. The final population count is obtained by summing the outputs. This method is particularly useful for handling large arrays of data efficiently.
5. Population Count Operations on Logical Results: In many practical applications, the population count is performed based on the result of logical operations, such as AND, OR, or XOR between two bitsets, as shown by Marton *et al.* (2017). This involves loading input bytes, generating temporary words through logical operations, and then determining their population count. These operations can be performed by either software methods or hardware implementations, depending on the specific application requirements.

Moreover, Polychroniou *et al.* (2015) noted that multiple recent advancements include vectorized algorithms that leverage modern microprocessor capabilities to process multiple bits in parallel. These algorithms can significantly enhance performance by utilizing SIMD (Single Instruction, Multiple Data) instructions available in many modern processors. Vectorized methods are especially effective for large datasets and high-performance computing applications.

Furthermore, many solutions apply hardware implementations of population count functions involving specialized circuitry designed to perform bit-counting efficiently, as shown by Klarqvist (2021). Many modern processors include dedicated instructions for population count, such as the POPCNT instruction in x86 architectures. These implementations offer high-speed bit-counting by leveraging hardware-level parallelism and optimized circuitry.

These existing techniques highlight the diversity of approaches to the population count problem, each suited to different scenarios and performance requirements. The proposed Virtual Tree for Population Count Method (VTPC) aims to offer a novel software-based solution that improves upon the time complexity and practical implementation challenges associated with these traditional methods.

2. Virtual tree for population count method (VTPC)

Algorithm 1 Population count algorithm using VTPC

Require: n (a positive integer in decimal number system)

Ensure: m (the number of 1's present in the binary representation of n)

```

1: mask_size  $\leftarrow 4$ 
2: mask  $\leftarrow [0, 1, 1, 2]$ 
3:  $m \leftarrow 0$ 
4:  $i \leftarrow 0$ 
5:  $d \leftarrow \lceil \log_{\text{mask\_size}}(n) \rceil$ 
6:  $e \leftarrow \text{mask\_size}^d$ 
7:  $r \leftarrow n$ 
8: while  $d \geq 0$  do
9:    $i \leftarrow \lfloor (r/e) \times \text{mask\_size} \rfloor$ 
10:   $r \leftarrow r - \left( \frac{e}{\text{mask\_size}} \times i \right)$ 
11:   $e \leftarrow e/\text{mask\_size}$ 
12:   $m \leftarrow m + \text{mask}[i]$ 
13:   $d \leftarrow d - 1$ 
14: end while
15: return  $m$ 
```

The VTPC method calculates the population count of a given number using a search operation on a virtual tree structure as given in Algorithm 1. Here, each node within this tree is represented by a vector of integers. The root node is termed the “mask vector,” and each element within this vector corresponds to a child node. The mask vector has a length of 2^L , where L is any natural number. The elements of the mask vector are the first 2^L numbers in the sequence of population counts.

For example, using a mask vector of $\langle 0, 1, 1, 2 \rangle$, which represents the first four numbers

in the population count sequence of integers (population count of numbers 0, 1, 2 and 3), each child node is generated by adding the corresponding integer from the parent node to every integer in the mask vector. This resultant vector forms the child node. The leaf nodes of this virtual tree represent the population counts of the integers.

The algorithm uses a variable, i , to track and generate the necessary child node by skipping the appropriate number of elements to reach the desired position in the sequence. The algorithm operates in two main phases: selection and generation. During the selection phase, items from the current node are chosen for expansion. In the generation phase, the next node is generated by adding the selected element to the mask vector. This process iterates $\log(n) + 1$ times, where s is the length of the mask vector, and n is the input number. With each iteration, a new child node is generated, replacing its parent node, and the cycle continues until a leaf node is reached.

3. Case studies and simulation setup

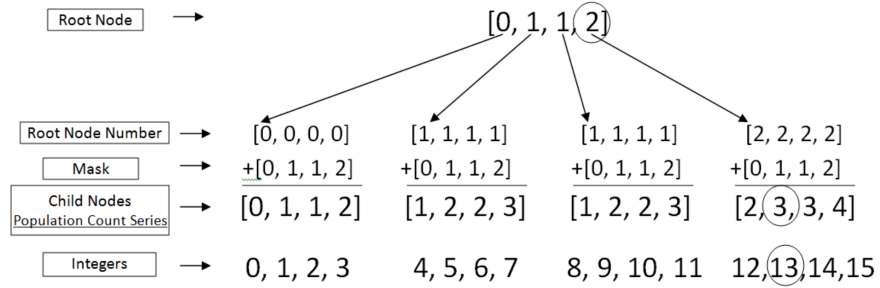


Figure 1: Complete virtual tree of depth 1 generated to compute population count of number 13. Integers of leaf nodes represent the population count series. The bottom most series in the figure represents the corresponding natural numbers from 0 to 15.

To illustrate the VTPC Method, we consider a case study to examine how the algorithm computes the population count for the number 13. Figure 1 presents the virtual tree for this problem. In the first iteration, the algorithm selects the fourth integer from the root node, which is 2. The child node is then generated by adding 2 to each integer in the mask vector, resulting in the vector $\langle 2, 3, 3, 4 \rangle$. Although Figure 1 displays all potential leaf nodes for demonstration purposes, the algorithm only generates the required child node at each step. During the second iteration, the algorithm selects the second integer from the current node $\langle 2, 3, 3, 4 \rangle$, which is 3. In the final iteration, the algorithm recognizes that the current node is a leaf node, indicating the end of the process. Consequently, the algorithm terminates its execution. The whole calculation process can be described as follows:

1. Initialization:

- (a) The mask vector is $\langle 0, 1, 1, 2 \rangle$.
- (b) The input number n is 13.
- (c) Initialize m (the population count) to 0.

- (d) Calculate d as $\lceil \log_4(13) \rceil$, which determines the depth or number of iterations. For $n = 13$, d is 2 because $\log_4(13)$ is approximately 1.85, and the ceiling function rounds it up to 2.
 - (e) Calculate e as $4^d = 4^2 = 16$.
 - (f) Initialize r (remainder) to 13.
2. First Iteration:
- (a) Calculate the index i using the formula: $i = \left\lfloor \frac{r}{e} \times 4 \right\rfloor$.
 - (b) Substituting the values: $i = \left\lfloor \frac{13}{16} \times 4 \right\rfloor$.
 - (c) Simplifying further: $i = \lfloor 0.8125 \times 4 \rfloor = \lfloor 3.25 \rfloor = 3$.
3. Selection: According to the mask vector $\langle 0, 1, 1, 2 \rangle$, the element at index 3 (0-based index) is 2.
4. Updating Values:
- (a) Update r by subtracting $\left(\frac{e}{4} \times i\right)$ from it: $r = 13 - \left(\frac{16}{4} \times 3\right) = 13 - 12 = 1$.
 - (b) Update e by dividing it by 4: $e = \frac{16}{4} = 4$.
 - (c) Update m by adding the selected mask element '2' to it: $m = 0 + 2 = 2$.
 - (d) Decrease d by 1: $d = 2 - 1 = 1$.
5. Second Iteration: In the second iteration, again the value of 'm' is updated using the above equations, and the resultant value '13' is outputted.

The selection of the fourth integer '2' in the first iteration is driven by the calculated index i . This calculation ensures that the algorithm navigates correctly through the virtual tree to determine the population count of the input number. This step-by-step process ensures that the population count of the number 13 is accurately determined by navigating through the virtual tree and selecting the appropriate child nodes at each iteration.

The time complexity of the proposed algorithm is $O(\log_s(n))$, where s is the size of the mask vector and n is the input number. This is because the while loop in the algorithm executes $\log_s(n) + 1$ times. An important aspect of the algorithm is its efficient space utilization. To search through the entire virtual tree, only the space for a single node is required at any given time. The algorithm uses the variable m to keep track of the integer to be added to the mask vector to generate the next child node. By increasing the size of the mask vector, the execution time can be reduced, as larger masks decrease the number of iterations needed. However, this improvement in time complexity comes at the cost of increased space consumption. To further illustrate this, consider an example where the input number is 368. Figure 2 demonstrates the virtual tree generated by the algorithm to find the population count of 368. This example shows how the algorithm navigates and constructs the tree efficiently, balancing time and space requirements based on the chosen mask size.

Figure 2 also illustrates the values of the algorithm's variables after each iteration. Initially, before the first iteration, the mask vector is set to the root node. The value of d is calculated to be 4, and it decrements by one with each iteration.

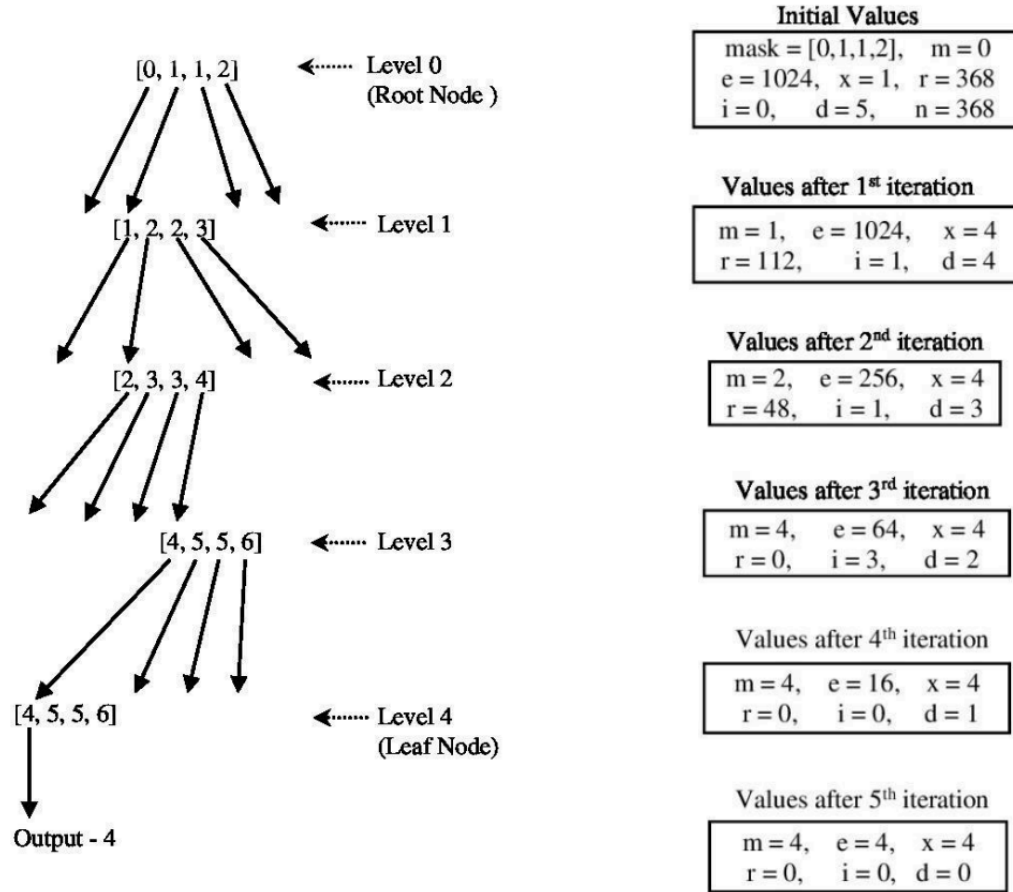


Figure 2: Virtual tree generated to find the population count of 368 (execution starts with the mask as root node). Execution Sequence of the algorithm to find the population count of the number 368. ‘i’ gives the position of the selected integer for next generation, ‘d’ controls the iterations, ‘m’ keeps track of the generated node and after the execution, it carries the output.

In the first iteration, the algorithm selects the integer ‘1’ from the root node for expansion. This selection is based on the calculation described in the algorithm. After generating the first child node, the current node is updated to $\langle 1, 2, 2, 3 \rangle$. Using the formula outlined in the algorithm, the next integer chosen from the current node is ‘2’. This selection leads to the generation of the child node $\langle 2, 3, 3, 4 \rangle$. In the following iteration, the algorithm selects ‘3’, resulting in the generation of the node $\langle 3, 4, 4, 5 \rangle$. Finally, in the last iteration, ‘4’ is selected from the current node, generating the node $\langle 4, 5, 5, 6 \rangle$. In the last step, the algorithm identifies ‘4’ as the first element of the final node, which is chosen as the population count result. This detailed step-by-step process shows how the algorithm navigates through the virtual tree, updating nodes and selecting elements based on predefined calculations to arrive at the final population count.

The proposed VTPC Method is evaluated through a simulation that verifies its accuracy across a range of integers from 0 to 10,000. The evaluation process involves the following steps (Algorithm 2). First, for each integer in the range, the population count is first calculated using the naïve method, which involves counting the number of set bits (‘1’s)

in the binary representation of the integer. Next, the population count for the same integer is then computed using the proposed VTPC method. Later, the results from both methods are compared. If the population counts match for every integer in the range, the evaluation algorithm returns ‘1’. If there is any mismatch, the algorithm returns ‘0’. In the conducted simulation, the evaluation algorithm consistently returns ‘1’, indicating that the proposed VTPC algorithm produces accurate population count results for all tested integers. This validation confirms the correctness and reliability of the VTPC method.

Algorithm 2 Simulation for evaluation of the proposed VTPC approach

Require: R (number of simulation cycles to be executed)

Ensure: 1 if VTPC approach passes the evaluation, otherwise 0

```

1:  $i \leftarrow 0$ 
2: while  $i \leq R$  do
3:    $d_1 \leftarrow \text{Naïve\_pop\_count}(i)$ 
4:    $d_2 \leftarrow \text{VTPC\_pop\_count}(i)$ 
5:   if  $d_1 \neq d_2$  then
6:     return 0
7:   end if
8:    $i \leftarrow i + 1$ 
9: end while
10: return 1

```

4. Time complexity comparison

The naïve method for population count involves examining each bit of the register individually, resulting in a time complexity of $O(M)$, where M is the total number of bits representing the input number. The lookup table method improves upon this by checking chunks of bits at a time. In this approach, the worst-case time complexity is $O(M/L)$, where L is the number of bits evaluated in each lookup operation. The Divide and Conquer technique offers further efficiency, with a time complexity of $O(\log_2(M))$. This is because the number of calculations required is halved at each step. The Carry-Save Adder (CSA) method operates similarly to the Divide and Conquer approach. It reduces the number of calculations by half after every iteration, achieving comparable efficiency. The proposed Virtual Tree for Population Count (VTPC) method has a worst-case time complexity of $O(\log_S(N))$, where N is the input number in decimal number system, and S is the size of the mask vector. Table 1 presents a comparison of the worst-case time complexities for these different algorithms, including the proposed VTPC algorithm, highlighting the efficiency gains of each method.

The line plot in Figure 3 illustrates the theoretical trade-off between space complexity (on the x-axis) and time complexity (on the y-axis) for the VTPC algorithm, across different values of N (8, 32, 128, and 512). Here N represents the decimal number corresponding to which the population count is to be calculated. From the figure, it is noted that for all values of N , as space complexity increases, time complexity decreases. This indicates a typical trade-off where more memory usage (space complexity) can reduce the time required to run the algorithm. Moreover, the decrease in time complexity becomes less significant as space complexity continues to increase. After a certain point (around 10–15 units of

Table 1: Comparison with different techniques based on their time complexity. Here, ‘M’ is the number of bits used to represent the input number ‘N’, ‘L’ is the number of bits used to address the elements of the lookup table, and ‘S’ is the size of the mask vector used in VTPC.

S. No.	Algorithm	Time Complexity
1	Naïve Method	$O(M)$
2	Look-up Table	$O(M/L)$
3	Divide and Conquer	$O(\log_2(M))$
4	CSA	$O(\log_2(M))$
5	VTPC	$O(\log_S(N))$

space complexity), the time complexity levels off, suggesting that additional space does not significantly improve time efficiency. In addition, higher values of N (larger data sizes) result in a higher baseline for time complexity at lower space complexities. However, the overall shape of the curve remains similar for each N , with each curve flattening at higher space complexity levels. As N increases, the benefit of increasing space complexity is more pronounced at lower levels of space complexity. This suggests that for larger problem sizes, allocating additional memory initially has a larger impact on reducing time complexity, but this benefit diminishes as more space is allocated.

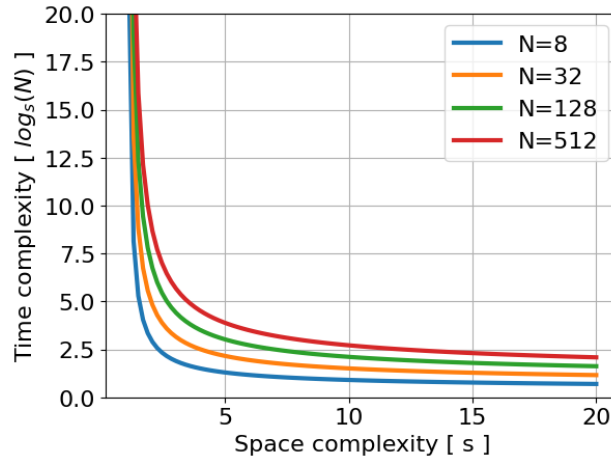


Figure 3: The trade-off between theoretical space complexity and time complexity by the VTPC algorithm.

In this context, Figure 4 shows a line plot that illustrates the relationship between mask size (representing a form of space complexity on the x-axis) and execution time (time complexity on the y-axis, measured in seconds) for different values of N (1000, 10,000, 100,000, and 1,000,000). The simulation for this study was implemented and executed using a Python 3.0 script on the Google Colab platform, running on a system equipped with a dual Intel® Xeon® CPU @ 2.20 GHz, 13 GB of RAM, and 108 GB of disk space.

From the figure, it is observed that for all values of N , there is a steep drop in execution time as the mask size increases from 0 to around 20. This suggests that increasing mask size significantly improves execution time initially, likely because additional memory allows for more efficient processing. Beyond a mask size of approximately 20, the execution time

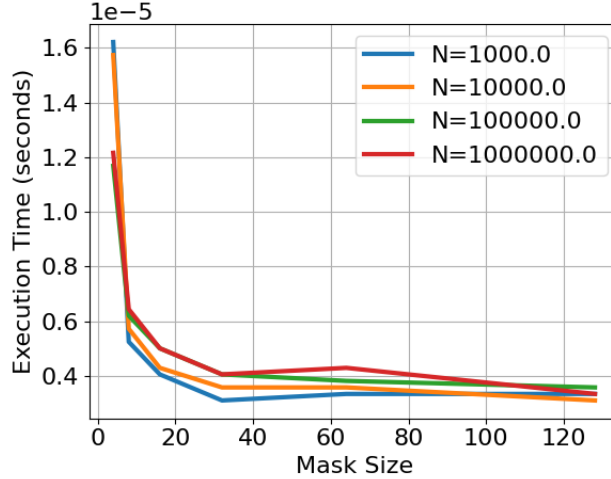


Figure 4: The trade-off between space complexity and time complexity observed through the simulation results.

stabilizes and does not show significant changes as the mask size continues to increase up to 120. This indicates diminishing returns, where increasing the mask size further does not result in noticeable improvements in execution time. Moreover, similar to Figure 3 observations, in Figure 4 also it is observed that the higher values of N tend to have a slightly higher baseline execution time, but the general pattern of improvement with increasing mask size is similar across all values of N . This suggests that the relationship between mask size and execution time is consistent across different data sizes, with similar diminishing returns for larger datasets.

In summary, the simulation results demonstrate that while increasing mask size (space complexity) initially improves execution time (time complexity), the benefits taper off beyond a certain mask size. This relationship is corroborated by the theoretical time and space complexity graph shown in Figure 3. Furthermore, this trend is consistent across different dataset sizes, showing that the simulation reaches a point of optimal efficiency where further increases in mask size no longer reduce execution time.

The VTPC algorithm, as a software-based population counting method, falls within the complexity class P , meaning it can be solved in polynomial time. By aiming to improve time complexity over traditional population counting methods, VTPC could potentially offer a more efficient approach within this class, especially for large data sets. While it does not shift the problem into a different complexity class, its optimizations could impact practical applications by making previously infeasible or slower computations more accessible and efficient, possibly narrowing the performance gap in applications that require rapid bit-counting, such as cryptography and big data processing.

The VTPC method could also be adapted for parallelization by distributing segments of the bit sequence across multiple cores, allowing each core to independently compute partial population counts. Leveraging multi-core processors, the VTPC algorithm can perform bit-counting concurrently across different parts of the data, aggregating results at the end for a final count. This parallel approach would significantly improve processing speed, especially for large datasets or high-throughput applications.

Furthermore, the VTPC method could be adapted for quantum algorithms by leveraging quantum parallelism to evaluate multiple branches of the virtual tree simultaneously, potentially enhancing population count speed. Quantum algorithms like Grover's search or amplitude amplification might be incorporated to efficiently locate and count '1' bits within superposition states, potentially achieving a faster, more scalable solution for large-scale population count tasks.

5. Conclusion

The need for efficient population counting arises across cryptography, database management, and machine learning, prompting the development of the Virtual Tree for Population Count Method (VTPC) as a faster, hardware-independent solution, streamlining processes and enhancing scalability in high-performance computing environments. The proposed VTPC method offers an innovative approach to calculating the population count of integers. By leveraging a virtual tree structure and mask vector, the VTPC method efficiently navigates through the population count sequence, reducing the time complexity to $O(\log_S(N))$, where N is the input number and S is the size of the mask vector. This stands in contrast to traditional methods such as the naïve bit-by-bit approach, lookup tables, and Divide and Conquer techniques, all of which exhibit higher time complexities under certain conditions. The accuracy and reliability of the VTPC algorithm were validated through a comprehensive simulation, which compared its results with those of the naïve method across a wide range. The consistent match of results underscores the correctness of the VTPC method. Moreover, the VTPC algorithm balances time and space complexity effectively. While increasing the mask size can further reduce execution time, it is essential to consider the corresponding increase in space consumption. The method's flexibility allows it to be tailored to specific application needs, ensuring optimal performance. Overall, the VTPC method represents a significant advancement in population count algorithms, providing a robust, efficient, and accurate solution suitable for various computational tasks. Future work can explore further optimizations and extensions of this approach to other domains requiring efficient bit-level operations.

Acknowledgements

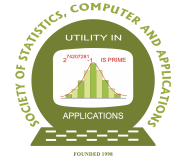
We extend our sincere gratitude to Dr. Anil Kumar and Dr. Sudeep Marwaha, ICAR-IASRI, New Delhi, India, and Dr. Pradip Basak, Uttar Banga Krishi Viswavidyalaya, West Bengal, India, for their invaluable guidance and support throughout this work. We are indeed thankful to the Editors for their guidance and counsel. We are very grateful to the reviewer for valuable comments and suggestions of generously listing many useful references.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Fateman, R. J. (1989). Lookup tables, recurrences and complexity. In *Proceedings of the ACM-SIGSAM 1989 International Symposium on Symbolic and Algebraic Computation*, pages 68–73.
- Klarqvist, M. (2021). Efficient analysis and storage of large-scale genomic data. In *University of Cambridge (Doctoral Dissertation)*.
- Knuth, D. E. (2014). *The Art of Computer Programming: Seminumerical Algorithms, Volume 2*. Addison-Wesley Professional.
- Marton, K., Raluca, B., and Suci, A. (2017). Counting bits in parallel. In *2017 16th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, pages 1–6. IEEE.
- Muła, W., Kurz, N., and Lemire, D. (2018). Faster population counts using avx2 instructions. *The Computer Journal*, **61**, 111–120.
- Polychroniou, O., Raghavan, A., and Ross, K. A. (2015). Rethinking simd vectorization for in-memory databases. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1493–1508.
- Rosen, K. H. (2019). *Discrete Mathematics and Its Applications*. McGraw-Hill.



Time Dependent Performance Analysis of Asynchronous Internet Switch - Multi Server Queueing System and Markovian QBD Process

Malla Reddy Perati, Shivaji Arepelly and Abhilash Vollala

Department of Mathematics, Kakatiya University, Warangal- 506009

Received: 14 February 2024; Revised: 25 November 2024; Accepted: 29 December 2024

Abstract

In this paper network nodes namely routers and switches with self similar input traffic are modelled as multi-server finite buffer queueing system. Time dependent Markovian Arrival Process (MAP) is employed for input process, as it emulates the nature of traffic. The network traffic is asynchronous, and of variable packet lengths, service times(packet lengths) are assumed to follow Phase type (PH) distribution, The Phase type distribution is more general, in the sense, it includes all kinds of exponential distributions namely Erlang, Hyper exponential and Coxian distribution. The transition rate matrix of buffer occupancy of the system is obtained using Quasi Birth and Death (QBD) process. Here, time dependent behaviour of queueing system is evaluated through the performance metrics, namely, packet loss, and mean waiting time. This kind of investigation is useful in dimensioning the network nodes to guarantee Quality of Service (QoS).

Key words: Internet switch; Self-similarity; Multi-server queue; Markov modulated Poisson process; Loss probability.

AMS Subject Classifications: 60G18, 60K25, 68M20

1. Introduction

In this era of technology, performance of communication networks play a significant role in all aspects of day to day life. Typically, network systems are characterized by queueing systems using Markovian approach. Due to inflated demand, network traffic is bursty, which causes congestion, thereby loss of information. For the said reasons, network traffic modeling and performance analysis have become crucial issues to guarantee the QoS. The networks namely LAN and WAN traffic display fractal-like characteristics, which can significantly impact network performance. The aggregation of this traffic reveals a self-similar pattern, and the degree of self-similarity measured in terms of the Hurst parameter, which is used for measuring the burstiness of traffic (Paxson and Floyd (1995); Crovella and Bestavros (1996)). Self-similar traffic exhibits the same statistical behaviour over a wide range of time scales (Leland *et al.* (1994)), hence, it is necessary to consider the said nature in modeling.

To model self-similar behaviour efficiently, various traffic models have been proposed earlier, like chaotic maps, Fractional Brownian Motion (FBM), and FARIMA, *etc.* These models characterize the traffic patterns in a parsimonious manner. Later on, several Markovian models were proposed to emulate the self-similar nature of traffic based on second order characteristics over certain ranges of time scales. However, they are not realistic because, the traffic parameters in the models were assumed to be homogeneous with respect to time and the performance metrics are computed at the steady state (Shao *et al.* (2005); Chang *et al.* (2006); Chen *et al.* (2007)). In this regard, a new procedure was proposed recently, based on the second order characteristics for fitting the time dependent Markovian process, and presented that it emulates self-similar traffic (Abhilash and Perati (2022)). On the other hand, Wavelength Division Multiplexing (WDM) is a promising technology to guarantee the Quality of Service (QoS) and provides a highly efficient way to increase network capacity by transmitting multiple signals over a single fiber. In WDM systems, converting asynchronous data (such as IP traffic) into synchronous traffic is often done through statistical multiplexing, it is used to dynamically allocate bandwidth based on traffic demand, and is suitable for circuit-switched networks (Chen *et al.* (2007); Perati *et al.* (2007); Venkataramani *et al.* (1997)). In asynchronous networks, packets typically have variable lengths, and routers must be capable of handling variable length packets. Using an MMPP/D/1/C queueing system (with deterministic service time) to model router performance is not suitable for variable-length packets. A better approach is the MMPP/M/1/C system, where service time is exponentially distributed. However, it is often more appropriate to model the service time (packet length) using a more general distribution, such as a Phase type distribution. This approach is particularly relevant in Broadband Integrated Services Digital Networks (B-ISDN), where communication services are designed to offer differentiated services (Diff-Serv) and ensure Quality of Service (QoS). The asynchronous router consists of N input fiber lines, N output fiber lines, and each fiber line has K (say) wavelength channels, and a wavelength converter pool of size c (say), ($0 \leq c \leq K$) dedicated to each output fiber line. In the papers Krieger and Wagner (1992); Artalejo *et al.* (2001); Kim *et al.* (2016) network node is modelled as multi-server queueing system, and considered underlying Markov chain with its quasi birth-death structure. However, performance evaluation in the above papers was made in steady state which is again not realistic. For the above reasons, in the current work, asynchronous switch with self-similar input traffic is modeled into MMPP(t)/PH/c/K queueing system and transient performance analysis is made by using QBD process. The performance measures, namely, mean waiting time, packet loss are presented numerically. The paper is organized in the following way. The fundamental definitions of self-similar and Long Range Dependent(LRD) processes are placed in section 2. The queueing model description is presented in section 3. In section 4, performance analysis is given, and numerical results are presented in section 5. Finally, conclusions are given in section 6.

2. Self-similar and long range dependent (LRD) process

Self-similarity is a key characteristic in fractals. It is a property, wherein a certain feature of the object is preserved with respect to scaling in space and time. It is statistically defined as follows. Let X be a second order process with variance σ^2 , and the time axis is splitted into disjoint sub intervals of unit length. Let $X = \{X_t : t = 1, 2, \dots\}$ be the points (packet arrivals) in the t^{th} interval. Let $X^{(m)} = \{X_t^{(m)}\}$ be a new sequence obtained by

averaging the original sequence over non-overlapping blocks of size m . *i.e.*,

$$X_t^{(m)} = \frac{1}{m} \sum_{i=1}^m X_{(t-1)m+i}, \quad t = 1, 2, 3, \dots \quad (1)$$

The obtained sequence is a second order process, and is called exactly second order self similar with Hurst Parameter $H = 1 - \beta/2$, if

$$\text{Var}(X^{(m)}) = \sigma^2 m^{-\beta}, \quad \forall m \geq 1 \quad (2)$$

If the parameter H lies in $(0, 0.5)$, then the process X is called Short Range Dependant (SRD). The process X is called Long Range Dependant (LRD), if $0.5 < H < 1$. Self-similar traffic displays burstiness across a wide range of time scales, which distinguishes it from traditional SRD traffic models. However, it is possible to replicate the self-similar behaviour over a specified time-scale range by using Markov Arrival Processes (MAPs) to model, and by matching the second-order self-similarity characteristics of the process accurately (Shao *et al.* (2005); Abhilash and Perati (2022)).

3. Queueing model

Network switch under consideration is the WDM asynchronous $N \times N$ router, where each output port with self-similar input traffic is modelled into MMPP(t)/PH/c/K multi-server queueing system. Self-similar input traffic here is modeled as MMPP with time dependent sinusoidal arrival rates *i.e.*, $a + b_i \sin t$, where a, b_i are constants and $0 < b_i < 1$. The arrival process here is assumed to be superposition of d two state IPPs, and a classical Poisson process. The i^{th} IPP is characterized by following matrices:

$$Q_i = \begin{bmatrix} -d_{1i} & d_{1i} \\ d_{2i} & -d_{2i} \end{bmatrix}, \quad \Lambda_i(t) = \begin{bmatrix} \lambda_{1i}(t) & 0 \\ 0 & 0 \end{bmatrix}, \quad (i = 1, 2, \dots, d). \quad (3)$$

The superposition of IPPs is given by

$$Q = Q_1 \oplus Q_2 \oplus \dots \oplus Q_d, \\ \Lambda(t) = \Lambda_1(t) \oplus \Lambda_2(t) \oplus \dots \oplus \Lambda_d(t) \oplus \lambda_p(t) \quad (4)$$

where $\oplus$ indicates kronecker's sum, and $\lambda_p(t)$ denotes arrival rate of classical Poisson process at time t . The mean arrival rate $\lambda(t)$ is $\frac{1}{t} (\pi \int_0^t \Lambda(t) dt e)$, where e is the vector of ones with appropriate dimension, and π is steady state vector of Q . Here, each server follows continuous-time Phase type service distribution characterized by (β, S) with dimension k , where β is vector of order $1 \times k$, $S^0 = -S e$. The mean service time is $\mu = -\beta S^{-1} e$. Service discipline is assumed to be FIFO, and the service time (here packet lengths) are independent, identically distributed random variables governed by Phase type distribution of order k . System capacity is finite, and is taken to be K . The resultant MMPP(t)/PH/c/K queueing system can be described by using stochastic process, $Z(t) = (R(t), H(t), Y(t)), t \geq 0$. Here, $R(t)$ denotes the number of packets in the system at time t , and $H(t)$ denotes the number of customers just served in different phases of the service process at time t . According to assumptions, the process $Z(t)$ is a CTMC on the finite state space $S = \{z = (r, h_1, h_2, \dots, h_k, y)\}, 0 \leq r \leq K, 0 \leq h_j \leq c, 1 \leq j \leq k, 1 \leq y \leq m$. In the case of quasi

birth and death process with lexicographical ordering of state space S , transition rate matrix is given by

$$A(t) = \begin{bmatrix} A_{00}(t) & A_{01}(t) & 0 & \cdots & \cdots & \cdots & 0 \\ A_{10} & A_{11}(t) & A_{12}(t) & \ddots & \cdots & \cdots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \cdots & \vdots \\ \vdots & \ddots & A_{cc-1} & A_{cc}(t) & A_{cc+1}(t) & \ddots & \vdots \\ \vdots & \cdots & \ddots & A_{c+1c} & A_{c+1c+1}(t) & \ddots & 0 \\ \ddots & \cdots & \cdots & \ddots & \ddots & \ddots & A_{K-1K}(t) \\ \vdots & \cdots & \cdots & \cdots & 0 & A_{KK-1}(t) & A_{KK}(t) \end{bmatrix}$$

where each block matrix $A_{ij}(t)$ is an $n_i \times n_j$ matrix. The block matrix $A_{ii+1}(t)$ corresponds to arrival of IP packet, Q_{ii-1} associate with departure of IP packet and, $A_{ii}(t)$ are associate with internal phase changes with no arrival and service completion. The block matrices are as follows:

$$A_{00}(t) = D_0(t)$$

$$A_{ii+1}(t) = D_1(t) \otimes \beta \otimes I, \quad i = 0, 1, \dots, c-1$$

$$A_{ii-1} = I \otimes S^{\oplus i}, \quad i = 0, 1, \dots, c-1$$

$$A_{ii}(t) = D_0(t) \otimes S^{\oplus i}, \quad i = 0, 1, \dots, c-1$$

$$A_{ii}(t) = D_0(t) \otimes S^{\oplus c}, \quad i = c, c+1, \dots, K-1$$

$$A_{KK}(t) = D_0(t) \otimes S^{\oplus K} + D_1(t) \otimes I$$

$$A_{ii+1}(t) = D_1(t) \otimes I, \quad i = c, c+1, \dots, K$$

$$A_{ii-1} = I \otimes (S^0 \beta)^{\oplus c}, \quad i = c, c+1, \dots, K,$$

where $D_0(t) = Q - \Lambda(t)$, $D_1(t) = \Lambda(t)$ and I is the identity matrix of appropriate order.

4. Performance analysis

Let $\pi(t) = (\pi_0(t), \pi_1(t), \pi_2(t), \dots, \pi_K(t))$ be transient state probability vector of $A(t)$, that is, $\pi(t)$ satisfies Stewart (1994)

$$\frac{d}{dt} \pi(t) = A(t) \pi(t), \quad (5)$$

$$\implies \pi(t) = \pi(0) + \int_0^t A(s) \pi(s) ds \quad (6)$$

By applying Picard's iterative method, one can get k^{th} iteration, and is given by

$$\pi_k(t) = \pi(0) \left(I + \sum_{r=1}^k \int_0^t \int_0^{s_r} \dots \int_0^{s_2} A(s_r) \dots A(s_1) ds_r \dots ds_1 \right) \quad (7)$$

The solution of Eq. (7) is given by $\pi(t) = \lim_{k \rightarrow \infty} \pi_k(t)$, that is,

$$\pi(t) = \pi(0) \left(I + \sum_{r=1}^{\infty} \int_0^t \int_0^{s_r} \dots \int_0^{s_2} A(s_r) \dots A(s_1) ds_{r+1} \dots ds_1 \right) \quad (8)$$

If each element of arrival rate matrix $\Lambda(t)$ is Riemann integrable, then each element of $A(t)$ is Riemann integrable. By using Theorem 2.4.3 in Slavík (2007), one can get

$$\pi(t) = \pi(0) \prod_{k=0}^n (I + A(t_k)h), \quad (9)$$

where $h = t_k - t_{k-1}$, n is the number of partitions of the interval $(0, t]$, and $\pi(0)$ is state probability vector at time $t = 0$. The performance measures, namely, Mean Waiting Time (MWT) and Packet loss Probability are derived from Zhao *et al.* (2015) as follows:

By Little's Law, The Mean Waiting Time (MWT) is given by

$$MWT = \frac{E[N_L(t)]}{\lambda(t)} = \frac{\sum_{i=c+1}^K (i - c) \pi_i(t)}{\lambda(t)}.$$

where $E[N_L(t)]$ represents the average queue length of the system. In a finite buffer queueing system, there is a chance of loss of packets due to buffer is being full. The packet loss in Δ time is given by

$$P_{loss}(\Delta) = \frac{1}{\lambda(t + \Delta)} \left(\pi_K(t) \left[\int_t^{t+\Delta} (D_1(x) \otimes I) dx \right] e \right).$$

5. Numerical results

In this study, a real-time Internet switch is considered for the analysis. Wireshark, an open-source software tool, is commonly used for network protocol analysis. It enables the capture and real-time inspection of data packets transmitted across the network. The data collected (which is presented in Appendix) using Wireshark serves as the basis for numerical analysis, from which key network traffic parameters such as the arrival rate of the whole process (λ_w), the Hurst parameter (H), and the variance (σ^2) are calculated using the NumXL tool in MS-Excel. The NumXL tool in MS-Excel is used to calculate the traffic parameters. It is a statistical and time series analysis add-in that offers advanced functions for computing metrics, making it easy to analyze network traffic directly within Excel. Based on the tool, the values are obtained as follows $\lambda_w = 27.964$, $\sigma^2 = 138.39$, $H = 0.72$ in the interval of $[10, 10^3]$. Since $H > 0.5$, the input traffic exhibits long-range dependence (self-similar nature). Based on the obtained network parameters, the transient MMPP fitting has been done using the procedure given in the paper Abhilash and Perati (2022), *i.e.*, by

matching variance of MMPP and Self-similar traffic (real time). The number of superposed MMPPs (IPPs) are taken to be two, *i.e.*, $d = 2$. The sinusoidal arrival rates are considered for modelling the asynchronous input traffic, and is given in the form of $a + b \sin t$, where b varies in between $(0, 1)$. The arrival rates referenced are drawn from the work of Eick *et al.* (1993) and used to analyze the behaviour of $Mt/G/\infty$ queue. Based on the fitting procedure for the given input parameters the following values are obtained, which are used in the current numerical analysis. The transition rates of packet arrivals are $d_{11} = 0.0401126$, $d_{21} = 0.0087$, and arrival rates are $\lambda_1(t) = 1 + 0.3 \sin t$, $\lambda_2(t) = 1 + 0.7 \sin t$. Here, it is assumed that service time follows phase distributions with two phases, namely, Erlang (E_2), Coxian (C_2), and Hyper Exponential (H_2) with varying rates. The numerical results were obtained using MATLAB programming and are presented in the form of figures. In which, the comparison of performance measures, namely, Mean Waiting time and Packet Loss Probability is presented with respect to traffic intensity pertaining to E_2 , C_2 , and H_2 .

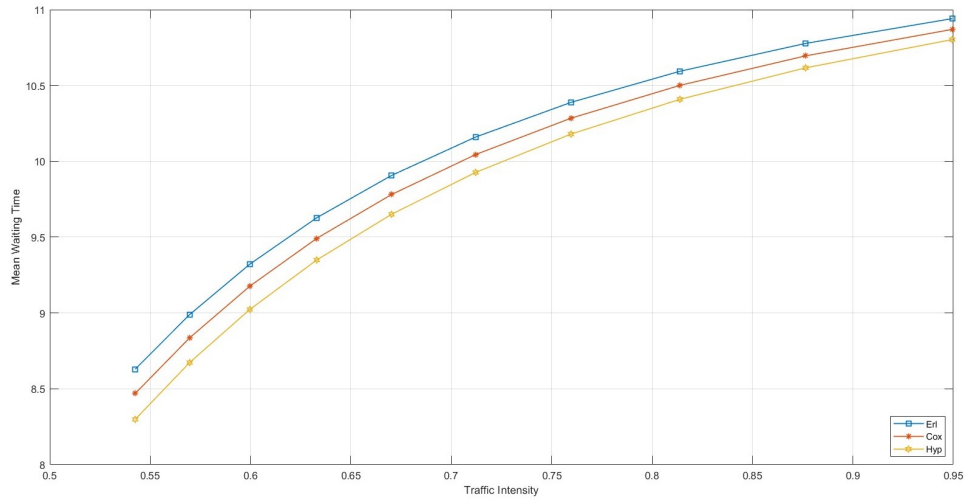


Figure 1: Mean waiting time versus traffic intensity for $t=5$, $K=20$, $c=3$

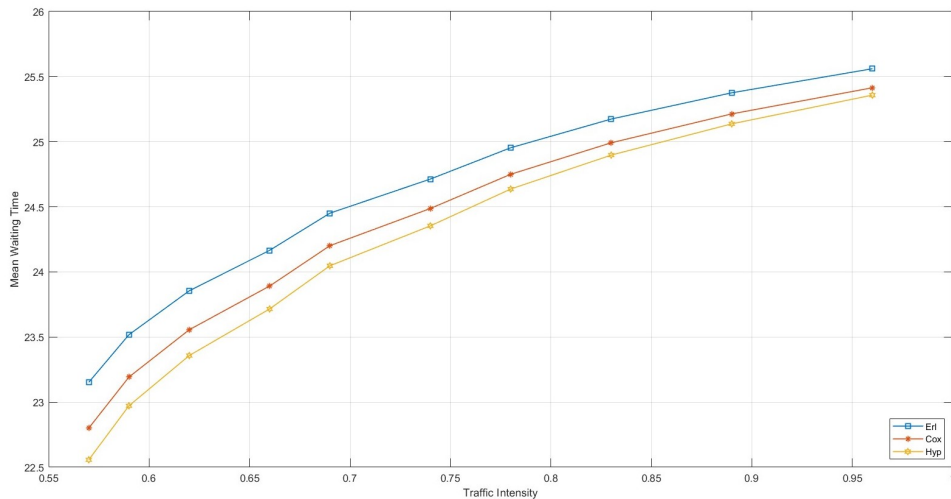


Figure 2: Mean waiting time versus traffic intensity for $t=10$, $K=20$, $c=3$

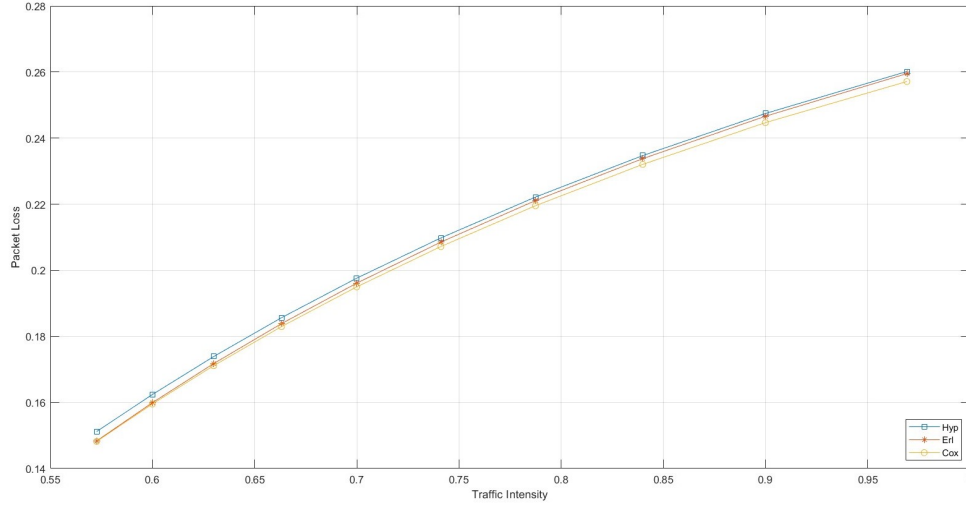


Figure 3: Packet loss probability versus traffic intensity for $\Delta = 1$, $t=10, K=20$, $c=3$

The figures show that as traffic intensity increases, both the average waiting time and the Packet loss probability increases. This suggests that, under heavier traffic conditions, the system faces longer delays and a greater chance of buffer overflow. The results presented in the figures reveals that the average waiting time for the $E2$ service type is consistently higher than $C2$ and $H2$, which means that $E2$ tends to experience more delays under similar traffic conditions. Also, packet loss probability, varies across different service types, *i.e.*, $H2$ has the lowest loss probability, followed by $E2$, and $C2$ experiences the highest packet loss. This indicates that, as traffic volume increases, $H2$ handles the load with the least packet loss, while $C2$ is more prone to packet loss. The insights gained from these results are crucial for understanding how each service type copes with varying traffic intensities. These type of analysis is useful for designing and optimizing networks to reduce delays and packet loss, particularly when traffic levels are high.

In this paper, the work is distinguished by its integration of time-dependent analysis, asynchronous behaviour, and Markovian models in a multi-server queueing system. The Internet switches with self-similar input traffic are modelled as multi-server Phase type queueing system, and the queueing behaviour is studied using time-dependent input parameters which obtained for a real time network traffic. Transient performance measures are obtained at different time instants using numerical and analytical techniques which are complimentary to steady state analysis. This approach offers a more accurate method for modeling and forecasting the performance of Internet switches, which is vital for modern networks dealing with fluctuating and unpredictable traffic. By using advanced queueing models like the QBD process, the analysis captures the complex dynamics between servers and queues, providing valuable inputs to improve network performance and reliability. This sort of analysis is appropriate in dimensioning the asynchronous internet switch.

Acknowledgements

Second author wishes to acknowledge the Ministry of Social Justice and Empowerment, Government of India (MOSJE, GoI) for their funding under the scheme NFSC (Na-

tional Fellowship for Scheduled Caste) (UGC-Ref. No.:4545/(CSIR-UGCNET JUNE2019).

Conflict of interest

Second author received financial support from the Ministry of Social Justice and Empowerment, Government of India (MOSJE, GoI) under National Fellowship for Scheduled Caste (NFSC) scheme for this study. The authors declare that this funding did not influence the study design, data analysis, or manuscript preparation, and that no other conflicts of interest exist.

References

- Abhilash, V. and Perati, M. R. (2022). Fitting model for self-similar traffic-time dependent markovian process and second order statistics. *Statistics and Applications*, **20**, 297–309.
- Artalejo, J. R., Gomez Corral, A., and Neuts, M. F. (2001). Analysis of multiserver queues with constant retrial rate. *European Journal of Operational Research*, **135**, 569–581.
- Chang, C. H., Shao, S. K., Perati, M., and Wu, J. (2006). Performance study of various packet scheduling algorithms for variable-packet-length feedback type wdm optical packet switches. In *2006 Workshop on High Performance Switching and Routing*.
- Chen, C. Y., Chang, C. H., Perati, M. R., Shao, S. K., and Wu, J. (2007). Performance analysis of WDM optical packet switches employing wavelength conversion under markovian modeled self-similar traffic input. In *2007 Workshop on High Performance Switching and Routing*, pages 1–6.
- Crovella, M. and Bestavros, A. (1996). Self-similarity in world wide web traffic: Evidence and possible causes. *IEEE/ACM Transactions on Networking*, **5**, 835–846.
- Eick, S. G., Massey, A. W., and Whitt, W. (1993). Mt/G/infty queues with sinusoidal arrival rates. *Management Science*, **39**, 241–252.
- Kim, C., Dudin, A., Dudin, S., and Dudina, O. (2016). Analysis of two-server queueing model with phase-type service time distribution and common phases of service. In *Distributed Computer and Communication Networks*, pages 19–29, Cham. Springer International Publishing.
- Krieger, U. R. and Wagner, D. (1992). Analysis and application of an MMPP/PH/n/m multi-server model. In *Performance of Distributed Systems and Integrated Communication Networks*, pages 323–342. North-Holland, Amsterdam.
- Leland, W., Taqqu, M., Willinger, W., and Wilson, D. (1994). On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, **2**, 1–15.
- Paxson, V. and Floyd, S. (1995). Wide area traffic: the failure of poisson modeling. *IEEE/ACM Transactions on Networking*, **3**, 226–244.
- Perati, M. R., Shao, S. K., Chang, C. H., and Wu, J. S. (2007). An efficient approximate markovian model for optical packet switches employing partial buffer sharing mechanism under self-similar traffic input. In *2007 IEEE Workshop on High Performance Switching and Routing*, doi:10.1109/HPSR.2007.4281232.

- Shao, S. K., Perati, M. R., Tsai, M. G., Tsao, H. W., and Wu, J. (2005). Generalized variance-based markovian fitting for self-similar traffic modelling. *IEICE Transactions on Communications*, **E88B**.
- Slavík, A. (2007). *Product Integration, Its History and Applications*. Matfyz press.
- Stewart, W. J. (1994). *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, Princeton, New Jersey.
- Venkataramani, B., Bose, S. K., Srivathsan, K., and R, K. (1997). Queuing analysis of a non-pre-emptive MMPP/D/1 priority system. *Computer Communications*, **20**, 999–1018.
- Zhao, N., Lian, Z., and Wu, K. (2015). Analysis of a MAP/PH/1 queue with discretionary priority based on service stages. *Asia-Pacific Journal of Operational Research*, **32**, 1550042.

ANNEXURE

The data presented below is used for performing numerical analysis and for calculating key traffic parameters like arrival rate of whole processes, Variance, and Hurst Parameter.

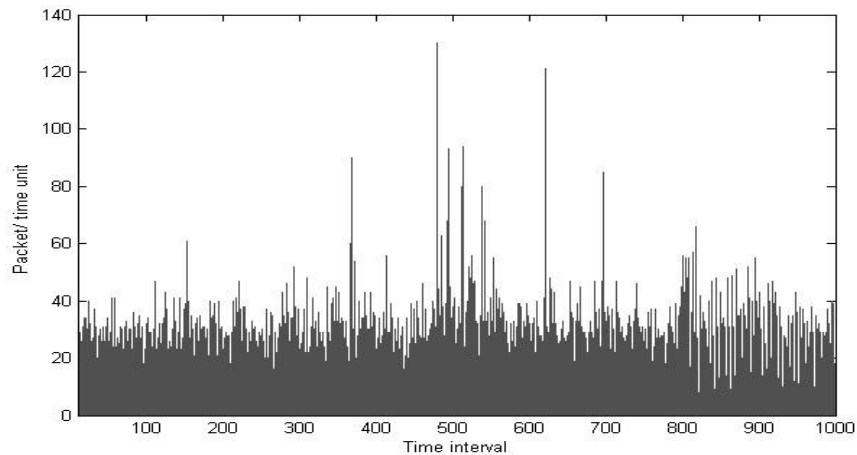
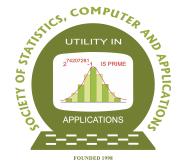


Figure 4: Packets per unit time of real-time data



Bayesian DCC-MGARCH Models for Understanding the Price Dynamics of Volatile Series: A Case Study on Pulses Price in India

Achal Lama¹, Girish K. Jha¹, Bishal Gurung² and Saurav Guha^{3,4}

¹ICAR – Indian Agricultural Statistics Research Institute, New Delhi, India

²North-Eastern Hill University, Shillong, India

³Bihar Agricultural University, Sabour, Bhagalpur, India

⁴Health Analytics Network, Pittsburgh, PA, USA

Received: 27 July 2024; Revised: 07 January 2025; Accepted: 10 January 2025

Abstract

This article explores the price movement and dependency among major pulses in India namely Moong, Masur, Urad and Arhar. To achieve our objective we have considered the well-known DCC-MGARCH model. A two-step estimation procedure, first, identifying the appropriate mean model using VAR and then the variance model using DCC-MGARCH has been performed. An attempt has also been made to document the comparative performance of the DCC-MGARCH model under both classical and Bayesian framework. The model parameters have been estimated using MLE and MCMC methods. The robustness of the Bayesian DCC-MGARCH model is evaluated under different prior settings and the number of MCMC samples drawn. Empirical studies indicate the superiority of Bayesian DCC-MGARCH model with errors satisfying multivariate t distribution for modelling the pulses series. The results suggest that the Bayesian DCC-MGARCH models outperforms the competing models in terms of parameters standard error, information criteria and its interpretability. This study also revealed the existence of co-movement of pulse prices to be dynamic, where prices of Moong and Urad exhibit the highest degree of dependency among them. The findings of this study can better assist the stakeholders in formulating policy decisions focusing on enhancing income of the pulse cultivators of India.

Key words: Pulses; VAR; MGARCH; DCC; Bayesian.

AMS Subject Classifications: 62M10

1. Introduction

India is one of the largest importer, producer, processor and consumer of pulses in the world. Thus pulses are an integral part of Indian agricultural system. Indian agriculture till date is largely dependent on monsoons. Pulses being no exception rely largely on monsoons. To aggravate the situation further, pulses are grown mainly on marginal lands. These factors

together contribute to the high level of fluctuations in pulse production and prices in India. Leading to the scenario that Indian farmers are not readily taking up cultivation of pulses despite of government ensuring high wholesale prices in recent years. Among the pulses, Moong, Masur, Urad and Arhar together are cultivated at around 50% of the total pulse growing area and account for more than 40% of the total production. In addition to it, these pulses face the inherent problem of fluctuation in production and prices. The movement of prices among the commodities can be short lived or long lived, depending upon the nature of the commodities. The fluctuation in prices leads to volatility and heterocedasticity is an important characteristic of volatility. There are various models reported in literature to model the volatility of a series. In recent years due to liberalisation of the markets in domestic as well as international markets the prices of the commodities move together over markets and time. It is due to this very reason that the researchers were compelled to look for models that can model more than one series together, the multivariate modelling approach. Different multivariate time-series models are used for forecasting, but after the pioneering work of Sims (1977), Vector Autoregressive (VAR) model have become the most popular among them for correlated series. This phenomenon of correlation is also observed in agricultural data series. The VAR model is useful only for modelling the mean or the first order moment of the series. Thus to have a better understanding of the series, modelling and forecasting volatility has been a major area of time series research for years.

Traditional econometric models assume a constant one-period forecast variances. To generalize this implausible assumption, Engle (1982) introduced a new class of stochastic process called Autoregressive Conditional Heteroscedasticity (ARCH) which is very useful where underlying forecast variances may change over time and is predicted by the past forecast errors. However, ARCH model has some drawbacks such as high number of unknown parameters and rapid decay of unconditional autocorrelation function of squared residuals *etc.* To overcome these difficulties, Bollerslev (1986) proposed the Generalized ARCH (GARCH) model in which conditional variance is also a linear function of its own lags. This model is also a weighted average of past squared residuals, but it has declining weights that never go completely to zero. It provides flexible lag structure and it permits parsimonious descriptions in most of the situations. The ability of GARCH model to capture volatility has been widely studied in literature (Lama *et al.*, 2015). A multivariate GARCH (MGARCH) model framework was proposed to model more than one series at a time. A class of MGARCH models have been developed over time. Engle and Kroner (1995) introduced a multivariate structure of GARCH model known as BEKK (Baba, Engle, Kraft and Kroner) model which is the direct generalization of univariate GARCH model and have huge flexibility. Bollerslev (1990) developed a relatively flexible approach known as Constant Conditional Correlation (CCC) model which allowed for combination of univariate GARCH model, with an assumption of constant correlation among the series over time. Engle (2002) proposed a new class of multivariate GARCH model known as Dynamic Conditional Correlation (DCC) model which has the flexibility of the univariate GARCH models coupled with parsimonious parametric model for the correlations. DCC model allows the conditional correlation to vary over time, adding dynamicity to the model. These models along with other variants of MGARCH models are explained lucidly by Bauwens *et al.* (2006). The use of these models for modelling the degree of interactions among various volatile commodities and markets can be widely seen in literature (Chevallier, 2012; Lean and Teng, 2013; Lin and Li, 2015; Gupta *et al.*, 2019). It is also interesting to note that most of the financial series exhibits leptokurtic behaviour

which has been documented well. There are number of ways to deal with this problem, such as the use of stable Paretian distribution with a characteristic exponent less than two to replace the normal error setting (Fama, 1965), discrete mixture of normal distributions can be used to explain the observed patterns of significant kurtosis and a positive skewness of data (Hull and White, 1998), but the use of Student- t distribution to model the error term is a promising alternative among them (Ku, 2008).

In recent times the use of Bayesian paradigm in time series literature has gained popularity. Researches have harnessed the benefits of Bayesian inference to interpret time series models explaining volatility process efficiently. Univariate as well as multivariate models have been estimated using Bayesian framework (Fioruci *et al.*, 2014). But, the use of Bayesian DCC-MGARCH model is very limited till date (Shiferaw, 2019). Furthermore, in majority of the cases the application is based on financial markets, very little efforts have been made to analyse volatile agricultural commodities. Hence, we undertake this study with two major objectives; first we model the multivariate agricultural price series in two step process using combination of VAR and DCC-MGARCH model. In the first step VAR has been used to capture the linear dependency among the series and then the DCC-MGARCH model has been used in the second step upon the residuals obtained from VAR model to capture the dynamic movement of non-linear volatile component among the series (Mohammadi and Tan, 2015; Lama *et al.*, 2016; Chen *et al.*, 2020). Secondly, this study also compares the performance of DCC-MGARCH model under Maximum Likelihood Estimation (MLE) and Bayesian framework using different error distributions for appropriate modelling of the inherent kurtosis nature of the series. Such comparative analyses are very few in multivariate time series literature (Shiferaw, 2019). The driving force behind implementation of Bayesian framework in the present study can be attributed largely to these three properties of Bayesian estimation:

1. Constraints on the model parameters to ensure positive variance are achieved in Bayesian framework by assigning priors to the model parameters.
2. The Bayesian framework also provides reliable results for finite samples and heavy tailed data sets (Hall and Yao, 2003).
3. The comparison of models is more consistent in Bayesian framework as it makes use of not only the estimates as in MLE, but also the marginal likelihood and Bayes factor or posterior model probabilities (Miazhyńska and Dorffner, 2006).

The very basic assumption of a Bayesian framework is that the data is not exhaustive to explain all the underlying behavior of the series. Thus priors are to be assigned to the parameters of the model and then posterior is estimated under that prior information. Bayesian analysis also provides the density of the parameters of the model unlike the point or interval estimates provided by the classical approaches. Further, in Section 2 brief details of the VAR, MGARCH models and testing of MARCH effect have been described, followed by empirical results in Section 3. Section 4 of this paper deals with the discussion of the results. Finally, the paper is concluded in Section 5.

2. Theoretical background

2.1. VAR model

Let $Y_t = (y_{1t}, y_{2t}, \dots, y_{Nt})^T$ denote an $(N \times 1)$ vector of time series variables. The basic p -lag vector autoregressive VAR(p) model has the form:

$$Y_t = A + B_1 Y_{t-1} + B_2 Y_{t-2} + B_3 Y_{t-3} + \dots + B_p Y_{t-p} + \varepsilon_t \quad (1)$$

where, A is $N \times 1$ vector of intercepts, $B_i (i = 1, 2, \dots, p)$ is $N \times N$ matrices of parameters and $\varepsilon_t \stackrel{iid}{\sim} N(0, \Sigma)$. The number of parameters to be estimated in the VAR model is $N(1 + Np)$ which increases with the number of variables (N) and number of lags (p).

2.2. MGARCH models

For a multivariate time series $y_t = (y_{1t}, y_{2t}, \dots, y_{Nt})^T$ the MGARCH model is given by:

$$y_t = H_t^{1/2} \varepsilon_t \quad (2)$$

where, $H_t^{1/2}$ is $N \times N$ positive-definite matrix and of the conditional variance of y_t . N is the number of series and $t = 1, 2, \dots, n$ (number of observations). It is with the specification of conditional variance that the MGARCH model changes. The core issues in MGARCH model is to construct the conditional variance-covariance matrix H_t . A relatively easy estimation approach is the CCC model introduced by Bollerslev (1990). This model assumes the conditional correlations to be constant. This restriction strongly reduces the number of unknown parameter and thus simplified the estimation. In case of CCC model the H_t represented as follows:

$$H_t = D_t R D_t \quad (3)$$

where, $D_t = \text{diag}(h_{11,t}^{1/2}, \dots, h_{NN,t}^{1/2})$ and R is a symmetric positive-definite matrix whose elements are (constant) conditional correlations:

$$R(i, j) = \begin{cases} 1 & \text{for } i = j (i, j = 1, 2, \dots, N); \\ \rho_{ij} & \text{otherwise.} \end{cases}$$

Here $h_{11,t}^{1/2}, \dots, h_{NN,t}^{1/2}$ are conditional variances of each series estimated from univariate GARCH models. Thus each conditional covariance is given by (Bollerslev, 1990):

$$h_{ij,t} = \rho_{ij} \sqrt{h_{ii,t} h_{jj,t}}. \quad (4)$$

In case of DCC the R matrix is also time varying thus making it dynamic. The representation of the model is as follows:

$$H_t = D_t R_t D_t \quad (5)$$

where, $R_t = \text{diag}(Q_t)^{-1/2} Q_t \text{diag}(Q_t)^{-1/2}$ and $Q_t = (1 - \alpha - \beta) R + \alpha u_{t-1} u_{t-1}^T + \beta Q_{t-1}$ and $u_t = D_t^{-1} y_t$. R is the unconditional covariance matrix of u_t . And the conditional covariances are given by:

$$h_{ij,t} = q_{ij,t} \sqrt{h_{ii,t} h_{jj,t}} / \sqrt{q_{ii,t} q_{jj,t}}. \quad (6)$$

Q_t is written as GARCH(1, 1) type equation and then transformed to get R_t .

2.3. Parameter estimation

2.3.1. Maximum likelihood estimation

The log likelihood estimator (Engle, 2002) is as follows:

$$y_{t|t-1} \sim N(0, H_t)$$

$$\begin{aligned} L &= -\frac{1}{2} \sum_{t=1}^T (n \log(2\pi) + \log |H_t| + y_t^T H_t^{-1} y_t) \\ &= -\frac{1}{2} \sum_{t=1}^T (n \log(2\pi) + \log |D_t^T R_t D_t| + y_t^T D_t^{-1} R_t^{-1} D_t y_t) \\ &= -\frac{1}{2} \sum_{t=1}^T (n \log(2\pi) + 2 \log |D_t| + \log |R_t| + \varepsilon_t^T R_t^{-1} \varepsilon_t) \\ &= -\frac{1}{2} \sum_{t=1}^T (n \log(2\pi) + 2 \log |D_t| + y_t^T D_t^{-1} D_t y_t - \varepsilon_t^T \varepsilon_t + \log |R_t| + \varepsilon_t^T R_t^{-1} \varepsilon_t) \end{aligned} \quad (7)$$

Now this equation is to be maximized over the parameters of the model. The main advantage of this estimation procedure is that it allows the model to be estimated even when the covariance matrix is very large. The essence of estimation of DCC model lies in the fact that it is a two-step estimation procedure. The first one being the estimation of the volatility part and the next one is the correlation part. Thus the likelihood can be written as the sum of the volatility and correlation components. To do so we denote the D by φ and R by ϕ , now the likelihood can be written as :

$$L(\varphi, \phi) = L_v(\varphi) + L_c(\varphi, \phi).$$

The volatility component is represented as:

$$L_v(\varphi) = -\frac{1}{2} \sum_t \left[n \log(2\pi) + \log |D_t|^2 + y_t^T D_t^{-2} y_t \right]$$

and the correlation part as:

$$L_c(\varphi, \phi) = -\frac{1}{2} \sum_t \left[(\log |R_t| + \varepsilon_t^T R_t^{-1} \varepsilon_t - \varepsilon_t^T \varepsilon_t) \right].$$

The two-step maximization is done and the estimates are obtained as follows:

$$\hat{\varphi} = \arg \max \{L_v(\varphi)\}$$

and then take this value as given in the second stage as $\max_{\phi} \{L_c(\hat{\varphi}, \phi)\}$.

In this estimation procedure maximum at second step is a function of the first step parameter estimates. Thus to ensure the consistency, first step estimates' consistency will ensure the consistency of the second step as long as the function is continuous in the neighborhood of the true parameters. The error distribution can be multivariate normal distribution or the multivariate Student-t distribution depending on the kurtosis of the data. The density function of the multivariate t distribution is given by:

$$p(\varepsilon_t) = \frac{\Gamma((v+k)/2)}{\Gamma(v/2) [\pi(v-2)]^{k/2}} \left[1 + \frac{\varepsilon_t^T \varepsilon_t}{v-2} \right]^{-(v+k)/2} \quad (8)$$

where, v is the degree of freedoms of t distribution and $\Gamma(\cdot)$ is the Gamma function.

2.3.2. Bayesian estimation

The MLE method is the classical approach for parameter estimation of the GARCH models. The Bayesian estimation technique is different from MLE and is growing rapidly. This technique is based on the generation of the posterior distribution based on the prior distribution through various techniques such as Markov Chain Monte Carlo (MCMC). In our study we have adopted Metropolis-Hastings (MH) algorithm where the GARCH parameters are updated by blocks (one block for α and one block for β) while the degrees of freedom parameter is sampled using an optimized rejection technique from a translated exponential source density. The basis of Bayesian estimation is the Bayes' Theorem. Let us consider the parametric space θ which is the vector of the parameters of the model with a prior density function $\pi(\theta)$ and Y is the data vector. According, to Bayes' rule the posterior density

$$\pi(\theta|Y) \propto L(Y|\theta)\pi(\theta) \quad (9)$$

where, $L(Y|\theta)$ is the likelihood function same as in 7.

At first model is fitted using MLE technique and the obtained log likelihood value is noted. Then the priors are defined for the parameters. Incorporating both log likelihood and prior, facilitating the algorithm to draw samples from the posterior distribution following the Bayes' rule. It generates Markov chain for each parameters based on the initial value. The ML estimates of parameters are taken as initial value for sampling in the present study. Based on the extensive review of literature Gaussian proposal density is taken for the current study. A new value of each parameter δ^{t+1} , where $\delta \in \theta^* = (\omega, \alpha, \beta, DCC_\alpha, DCC_\beta)$ is obtained with the acceptance probability $\alpha(\delta^{t+1}|\delta^t) = \min(1, r(\delta^{t+1}, \delta^t))$ for each $\delta \in \theta^* = (\omega, \alpha, \beta, DCC_\alpha, DCC_\beta)$ where, $r(\delta^{t+1}, \delta^t) = \pi(\delta^{t+1}|Y) / \pi(\delta^t|Y)$. The new value is either accepted or rejected and this will be continued until a sufficient number of iterations are obtained.

2.3.3. Model evaluation criteria

Standard model evaluation criteria, such as Akaike information criterion (AIC) and Bayesian information criterion (BIC), are used to compare the performance of different models. These criteria penalize the decrease in the degrees of freedom when more variables are added. The AIC and BIC values for GARCH model with Gaussian distributed errors are computed by:

$$AIC = 2\log(\text{likelihood}) + 2T$$

$$BIC = 2\log(\text{likelihood}) + \log(Tk)$$

where, k is model degrees of freedom. In case of models estimated using Bayesian technique these AIC and BIC evaluation criteria are not applicable as it involves samples estimated through MCMC from posterior distribution. Hence, Deviance Information criterion (DIC) is used as we have assumed posterior distributions to be normally distributed. The formula is as follows:

$$DIC = 2E[D(\theta M)] - D(E[\theta M])$$

where θM is the set of parameters in model M and $D(\cdot)$ is the deviance function defined as minus twice the log-likelihood function. So, given a sample from the posterior distribution of θM , it is straightforward to approximate the DIC. The implementation of the models to the dataset was done using R software and a schematic representation is given in Figure 1.

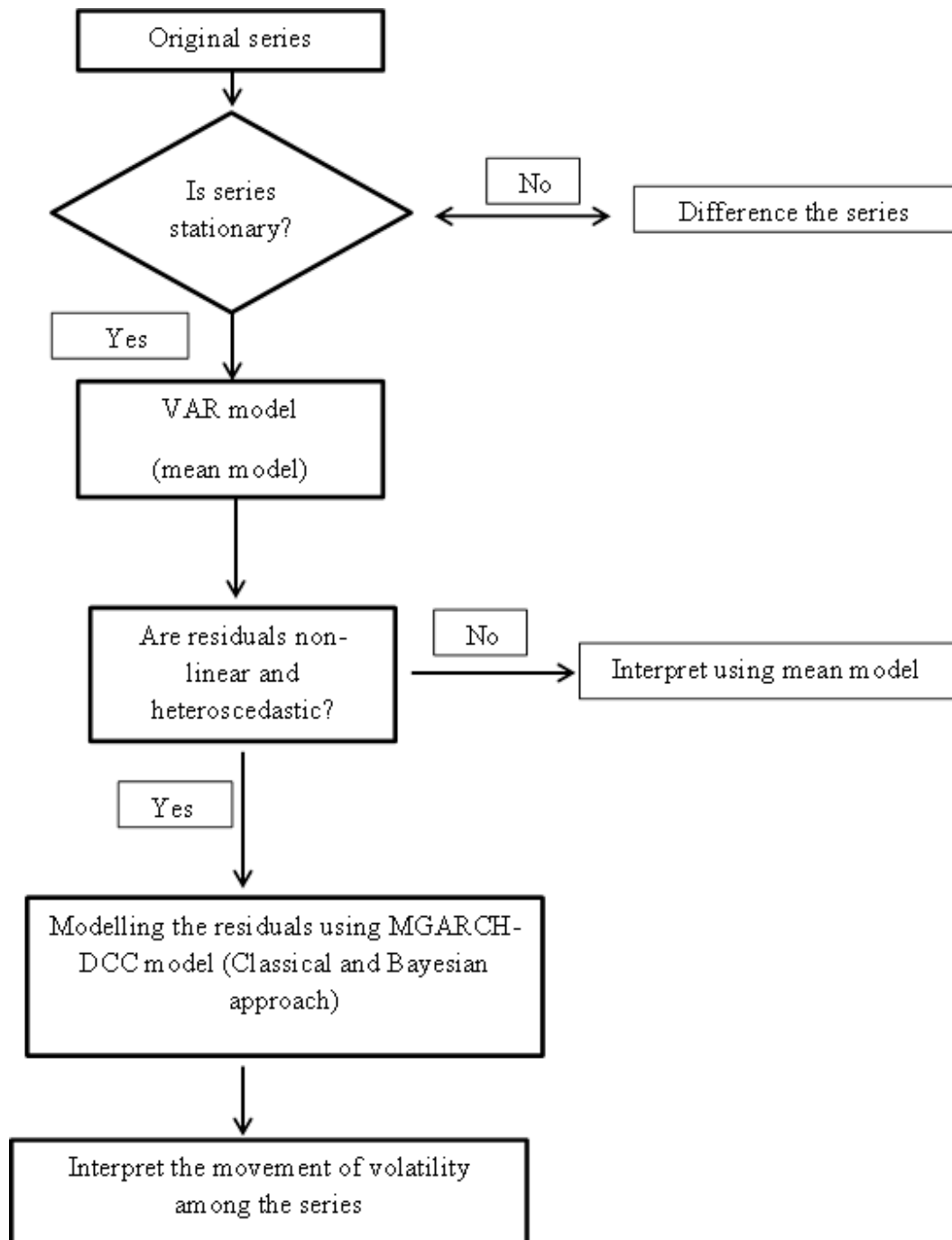


Figure 1: Flowchart of the employed methodology

3. Empirical results

3.1. Data description

The four major pulses series namely price index of Moong, Masur, Urad and Arhar are used in this study. The source of the data being Office of the Economic Adviser, Ministry of Commerce and Industry, Government of India (<https://eaindustry.nic.in/>). These series very well depict the properties of typical agricultural price data with inherent fluctuations as shown in Figure 2. Each series (monthly) contained 312 data points (April, 1994 to March, 2020) and the entire series was used for model development purpose. The statistical properties of the dataset are presented in the Table 1. We also explore the stationarity of the series by implementing the augmented Dickey–Fuller (ADF) test and we find in Table 1 that the series are stationary after first order differencing. The visual inspection from Figure 3 of the stationary series clearly indicated the presence of volatility at several time-epochs. The data set was tested for the presence of seasonality using the correlogram. The ACF (auto correlation function) in Figure 4 for all the series was devoid of any oscillatory pattern, indicating absence of any seasonality. In addition, skewness and kurtosis of the stationary series were computed along with the correlation coefficients. Table 2 shows a fair degree of linear dependency was observed among the series ranging from 0.364 to 0.593. It is evident from Table 3 that the stationary series tends to be non-skewed apart from Urad showing moderate skewness and leptokurtosis being a common feature to all.

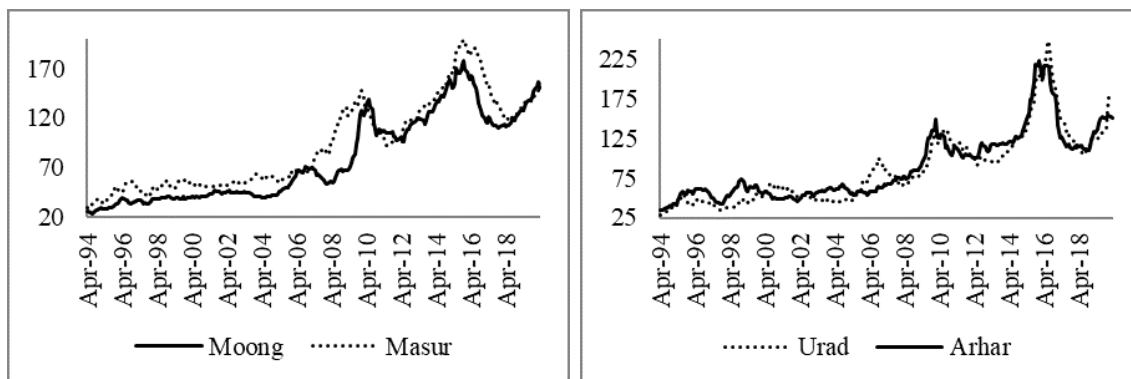


Figure 2: Time plot of the series

Table 1: Summary statistics of different pulses

Properties	Moong	Masur	Urad	Arhar
Mean	78.49	92.25	86.88	89.66
Median	61.90	74.95	76.00	70.10
Standard Deviation	43.32	44.97	45.69	42.91
Kurtosis	-1.18	-0.86	0.84	0.64
Skewness	0.52	0.57	1.07	1.06
Minimum	23.66	28.84	28.60	34.60
Maximum	178.10	198.80	247.60	222.80
C.V(%)	55.20	48.75	52.58	47.86
ADF (Level)	-3.32	-2.93	-2.89	-3.20
ADF (1st Differenced)	-4.90*	-4.51*	-4.71*	-4.72*

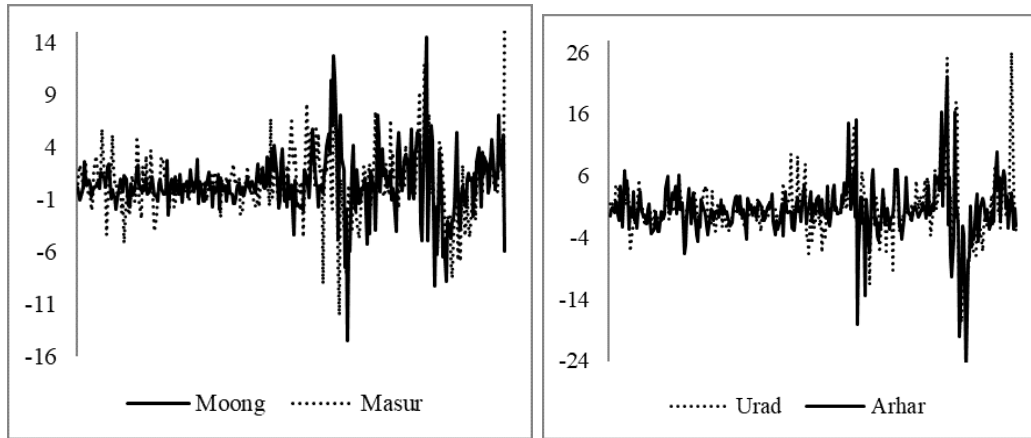


Figure 3: Plot of the first differenced series

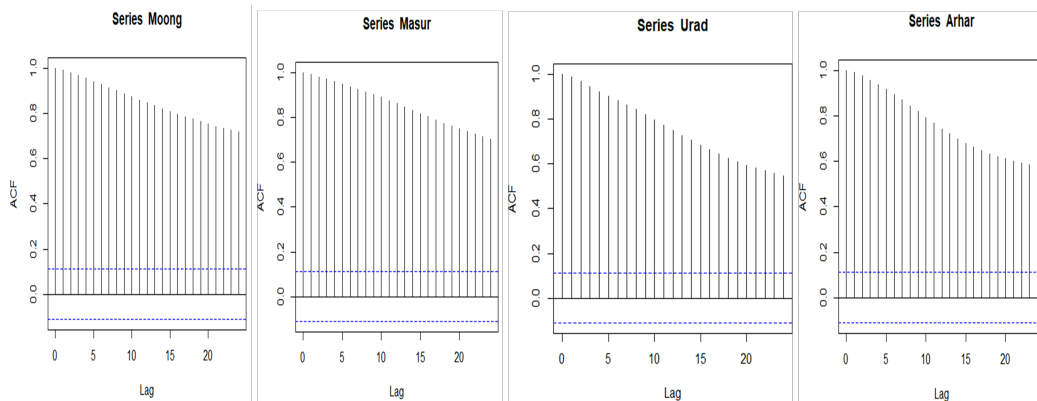


Figure 4: ACF plots of the original series

3.2. Fitting VAR model

The VAR model as discussed earlier was fitted to the stationary dataset. The appropriate order of the VAR model was selected based on Bayesian information criterion (BIC) and Hannan and Quinn (HQ) information criterion. We focused on these two criteria as they penalize depending on the sample size, unlike AIC which penalizes each parameter by a factor of 2. The lag order with lowest values for both the criteria was selected (BIC: 8.893; HQ: 8.777). We obtained VAR (1) model for the present investigation. The estimates of the model given in Table 4 reveal linear dependency within and between the series at first lag. Within series dependency ranged between 0.219 to 0.428 and 0.003 to 0.293 for between the series. The strongest linear dependency within series was found for Urad (0.428) and Arhar and Masur (0.293) exhibited highest between series association. As a standard statistical practice we carried out the diagnostic check of the residuals to determine the suitability of the fitted model. To this end, we start with obtaining the Q-Q plot given in Figure 5 for the residuals of each series to verify the normality assumption. The Q-Q plot in Figure 5 indicates the residuals violating normality assumption as they do not align with the straight line. Further, we moved on with studying the non-linearity of the residuals and presence of heteroscedasticity each at univariate (individual residuals) and multivariate (all residuals together) framework. The results in Table 5 point that residuals series are non-linear in

Table 2: Correlation matrix of the stationary series

	Moong	Masur	Urad	Arhar
Moong	1	0.364	0.519	0.457
Masur	0.364	1	0.412	0.550
Urad	0.519	0.412	1	0.593
Arhar	0.457	0.550	0.593	1

Table 3: Descriptive statistics of the stationary series

Statistics	Moong	Masur	Urad	Arhar
Mean	0.403	0.429	0.476	0.371
Standard Deviation	2.942	3.066	4.601	4.388
Kurtosis	5.337	3.597	7.579	9.003
Skewness	0.119	0.251	1.098	-0.393

nature as the null hypothesis of linearity is not accepted for BDS test. Then, the presence of heteroscedasticity is confirmed for both univariate and multivariate framework by the ARCH- Lagrange multiplier (LM) and Multivariate ARCH-LM tests respectively. For both the tests, null hypothesis of homocedasticity is not accepted as evident from Table 5. Finally, we conclude the residuals to be non-normal, non-linear and heteroscedastic (volatility clustering) in nature. These characteristics of the residuals demands for further modelling using MGARCH model in general and DCC-MGARCH model in particular as we aim at understanding the dynamic co-movement in the volatility structure.

Table 4: Estimates of $VAR(1)$ model

Series	Moong	Masur	Urad	Arhar
Moong (-1)	0.347 (0.064)	0.003 (0.066)	-0.022 (0.044)	0.091 (0.049)
Masur (-1)	0.153 (0.065)	0.316 (0.067)	0.012 (0.045)	0.058 (0.049)
Urad (-1)	0.202 (0.088)	-0.130 (0.091)	0.428 (0.061)	0.190 (0.067)
Arhar (-1)	0.293 (0.095)	0.178 (0.097)	-0.027 (0.065)	0.219 (0.072)
C	0.231 (0.1565)	0.217 (0.158)	0.156 (0.215)	0.112 (0.231)
AIC	8.598			

Values in the parenthesis are Standard Errors.

3.3. Fitting DCC-MGARCH models

The stationary series were tested for the presence of dynamic conditional correlation using Engle and Sheppard Test of Dynamic Correlation (Engle and Sheppard, 2001). The test statistic value (9.257) was rejected at 5% level of significance, confirming the presence of dynamic properties of the series. Thus we decided for the application of DCC model. To begin the implementation of the DCC model we have to first identify the volatility model,

in this study we have used DCC-MGARCH model in 5 with an assumption of the errors to be multivariate normal and the multivariate t distribution. The parameters were estimated both by the MLE and Bayesian framework.

The results of the DCC model estimated using MLE with multivariate Normal and multivariate t errors are reported in Table 6. It is clear from the Table 6 that ARCH (α) parameters for Masur, Urad and Arhar along with the DCC_α are not significant at 5% when estimated using MLE for both the error distributions. However, the lower value of BIC for DCC model with multivariate t error suggests its superiority over the other model for modelling the pulses series. This result cast a doubt in the appropriateness of the model fitted.

Then, we turned our attention into the Bayesian estimation procedure. Essentially, Bayesian estimation starts with defining prior distribution for all the parameters of the model. In this study we have defined the priors based on results obtained by Fioruci *et al.* (2014) and Shiferaw (2019). We use truncated Normal priors on the GARCH parameters α and β as follows:

$$p(\alpha) \sim \phi N_2(\alpha|\mu, \Sigma) 1\{\alpha \in R_+^2\}$$

$$p(\beta) \sim \phi N_1(\beta|\mu, \Sigma) 1\{\beta \in R_+\}$$

where μ and Σ are the hyperparameters, $1\{\cdot\}$ is the indicator function and ϕN_d is the d -dimensional Normal density. The values of the hyperparameters are kept fixed. The prior distributions are normal owing to its conjugate nature. The ω , DCC_α and DCC_β are assigned $N \sim (0, 10)$ priors, whereas the priors for α and β are varied with a total of 5 different combinations. The prior combinations are as follows: The prior combinations are as follows:

Prior 1 : $\alpha \sim N(0.1, 0.01), \beta \sim N(0.4, 0.01)$

Prior 2 : $\alpha \sim N(0.2, 0.03), \beta \sim N(0.5, 0.03)$

Prior 3 : $\alpha \sim N(0.3, 0.05), \beta \sim N(0.6, 0.05)$

Prior 4 : $\alpha \sim N(0.4, 0.07), \beta \sim N(0.7, 0.07)$

Prior 5 : $\alpha \sim N(0.5, 0.09), \beta \sim N(0.8, 0.09)$.

These combinations allow us to test the sensitivity of parameter estimates of the model with regards to change in distribution or the parameters of the distribution. As, in our investigation we have used normal priors, hence we judge the sensitivity by varying mean and variance within a specified range. The prior settings used for simulation study can be broadly classified into two groups. In the first group (Prior 1, Prior 2 and Prior 3) where the sum of α, β are < 1 and the second group (Prior 4 and Prior 5) where the sum of α, β are > 1 . These groups are formed intentionally to study the impact of α, β on the overall parameter estimation of the model. Along with the sensitivity of the priors we have also evaluated the effect of various sample sizes of MCMC draws on parameter estimates. The results reported in Table 6 are of prior 2 specifications with 10,000 MCMC draws. Results are encouraging as parameters estimate along with its standard error clearly indicates its statistical significance. Moreover, the DCC model with multivariate t errors has lower DIC values as compared with the multivariate Normal errors. This indicates the superiority of the model over the other one. The superiority of DCC-MARCH model with multivariate t

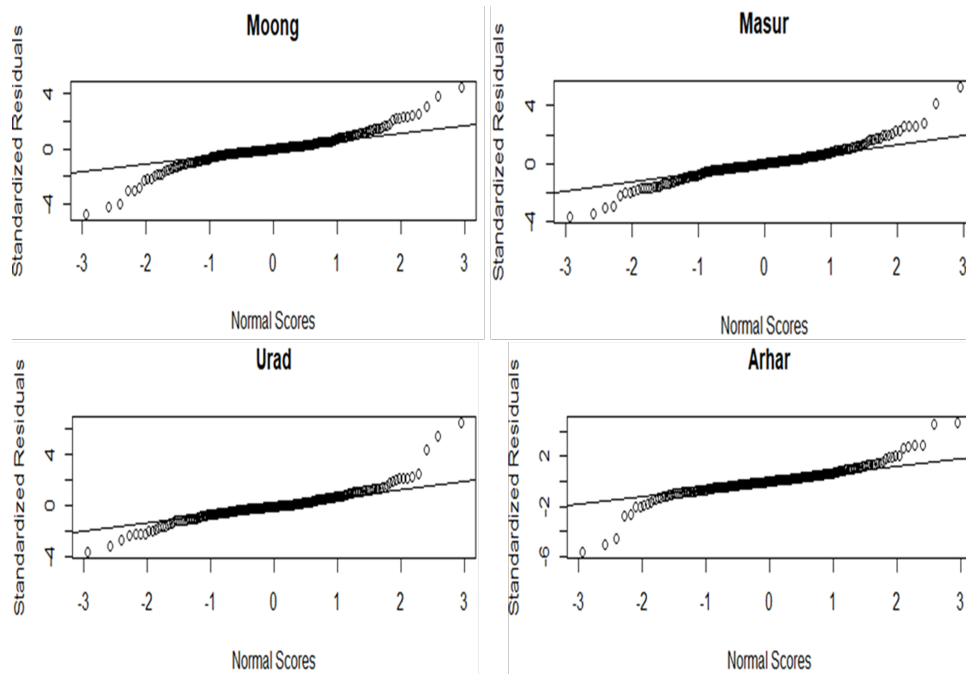


Figure 5: Q-Q plots of the residuals after fitting VAR(1) model

distribution to model leptokurtic pulses price series goes with the findings in literature (Ku, 2008).

Table 5: Residual diagnostics of the VAR (1) model

Test	Moong	Masur	Urad	Arhar
ARCH-LM	90.942*	13.482*	23.776*	23.776*
BDS	6.290*	3.726*	5.441*	4.161*
MARCH		241.467*		

*Significant at 5% level of significance

4. Discussions

The series under consideration were modelled following standard statistical procedures with VAR-DCC-MARCH models. VAR model was first applied for capturing the mean dependency of the multivariate framework. Residuals obtained from VAR model was used as an input for the DCC-MGARCH model for further analysis. In this study we have used the variant of DCC model after obtaining a positive result of DCC test. The DCC model parameters were estimated using the classical MLE method as well as the Bayesian method. The results from MLE estimates were not encouraging as all the parameters were not statistically significant. Thus result raises a very vital question upon the reliability of the model to explain the dynamic nature of correlation present between the series. Thus, to have further insight we decided to estimate the parameters using the Bayesian method. The results obtained through the Bayesian framework were very positive. As, looking into the parameters value along with its standard errors we can validate its statistical significance. The standard errors of each parameter under Bayesian framework are smaller as compared

Table 6: Parameter estimates of the DCC-MGARCH (1, 1) model

Parameters	Multivariate Normal error distribution		Multivariate t error distribution	
	MLE	Bayesian	MLE	Bayesian
ω_{Moong}	0.020 (0.024)	0.169 (0.007)	0.020 (0.024)	0.213 (0.014)
α_{Moong}	0.150 (0.037)	0.193 (0.005)	0.150 (0.037)	0.153 (0.011)
β_{Moong}	0.846* (0.046)	0.785 (0.005)	0.846* (0.046)	0.821 (0.010)
ω_{Masur}	0.059 (0.141)	0.573 (0.042)	0.059 (0.141)	0.511 (0.036)
α_{Masur}	0.06 (0.048)	0.229 (0.017)	0.06 (0.048)	0.163 (0.012)
β_{Masur}	0.937* (0.068)	0.729 (0.017)	0.937* (0.068)	0.774 (0.012)
ω_{Urad}	0.109 (0.506)	0.723 (0.029)	0.109 (0.505)	0.839 (0.072)
α_{Urad}	0.067 (0.099)	0.151 (0.006)	0.067 (0.099)	0.177 (0.009)
β_{Urad}	0.930* (0.152)	0.813 (0.007)	0.931* (0.151)	0.75 (0.013)
ω_{Arhar}	0.051 (0.078)	0.753 (0.037)	0.051 (0.078)	1.504 (0.089)
α_{Arhar}	0.189 (0.044)	0.285 (0.015)	0.189 (0.044)	0.247 (0.020)
β_{Arhar}	0.805* (0.043)	0.668 (0.014)	0.805* (0.043)	0.654 (0.011)
DCC_{α}	0.003 (0.007)	0.029 (0.002)	0.003 (0.008)	0.02 (0.002)
DCC_{β}	0.914* (0.024)	0.282 (0.011)	0.923* (0.043)	0.31 (0.037)

*Significant at 5% significance level. Values in the parenthesis are standard errors.

to that of MLE estimates as reported in Table 6.

Though we have modelled four series together, we can isolate the bivariate combinations for better interpretation of the DCC among them. A total of 6 such combinations were formed between them, namely Moong-Masur, Moong-Urad, Moong-Arhar, Masur-Urad, Masur-Arhar and Urad-Arhar. The dynamic correlation between each of the six combinations was studied under MLE and Bayesian framework. The mean, minimum and maximum values for each dynamic correlation are reported in Table 7 obtained from DCC-MGARCH model with multivariate t error distribution. The mean values for each combination are similar to 1 decimal place for MLE and Bayesian estimates but the maximum and minimum values differ noticeable for the two estimation procedure. The wider ranges of the dynamic correlation estimated under Bayesian DCC model indicates the efficiency of the model in capturing the dynamic co-movement among the series. Figure 6 depicts the DCC plots for all 6 combination of the series estimated under both techniques using multivariate t error. The figure highlights the inability of MLE estimates in capturing the co-movement efficiently over time between all the series as the correlation estimated do not fluctuate much over

time, unlike the fluctuations as noticed in the original series. This is mainly due to model parameters not being statistically significant and most importantly the DCC_{α} . Whereas the dynamic correlation exhibited by the Bayesian model seems to have explained the inherent fluctuations present in the original series. For instance, the Moong-Masur prices have fluctuated considerably between the period April, 2010 and March, 2016 similarly if we look into the DCC plot in Figure 6 a similar pattern of highly variable correlation is found between the series for the period. In Table 2 the dynamic correlation was found to be highest between Moong-Urad series (0.600) followed by Urad-Arhar (0.529), Masur-Arhar (0.483) and lowest among Moong-Masur (0.288). These dynamic correlation patterns fit with the linear correlation obtained between the stationary series.

Bayesian estimation comes with obvious advantages over the classical approach and has been put forward by our empirical illustration. As the Bayesian estimates are dependent upon the prior distribution and MCMC algorithm for generating posterior estimates, it is important to assess the sensitivity of the estimates with respect to the priors used and the MCMC samples drawn. In our implementation we have considered both these aspects. A total of 5 different prior settings under normal distribution has been tried with an increment of 0.1 in mean and 0.02 in the variance of the normal distribution. This has yielded us two fold results, first the sensitivity has been tested and second we could identify the optimal prior setting for our implementation. We document that the model estimates are quite robust with respect to the priors as we could only obtain slight changes in the estimates for prior 1 and prior 2 and rest of the prior settings (prior 3, prior 4 and prior 5) provided similar results as that of prior 2. Thus we have reported in Table 8 the estimates obtained from prior 1 and prior 2 settings only. After obtaining the optimal prior settings (prior 2) based on lower DIC value, we concentrate our efforts towards the sample size of MCMC draws. Two different sample sizes were tried 10,000 and 15,000 with 20% burn-in samples *i.e.*, 2000 and 3000 respectively. In the similar lines we could not find any significant difference among the estimates obtained. Hence, all the results presented for Bayesian estimates are obtained from 10,000 MCMC draws with 2,000 samples as burn-in.

As forecasting is essential for time series problems, however assessing forecasting performance in volatility problems is difficult as we are forecasting conditional covariance or correlation (Livingston and Nur, 2023). Hence we can compare the forecasted values of conditional variances with the true values and compute correlation among them as a measure of association among them (Table 9). The correlation between the original series and the forecasted conditional variances were found to be 0.67, -0.98, -0.67 and 0.89 respectively for Moong, Masur, Urad and Arhar.

5. Conclusions

In this study we have employed the VAR-DCC-MARCH model to volatile pulses price. In the first step the multivariate series is modelled using VAR model to capture the linear dependency and DCC-MGARCH model for capturing the dynamic co-movement among them. We document the superiority of the multivariate t innovations for modelling the leptokurtic behaviour of the series over the multivariate normal assumption of the errors. Our comparative analysis between the MLE and Bayesian estimation approach for DCC-MGARCH model provided comprehensive results highlighting the appropriateness of the Bayesian DCC-MGARCH model in terms of lower DIC values and standard errors of the

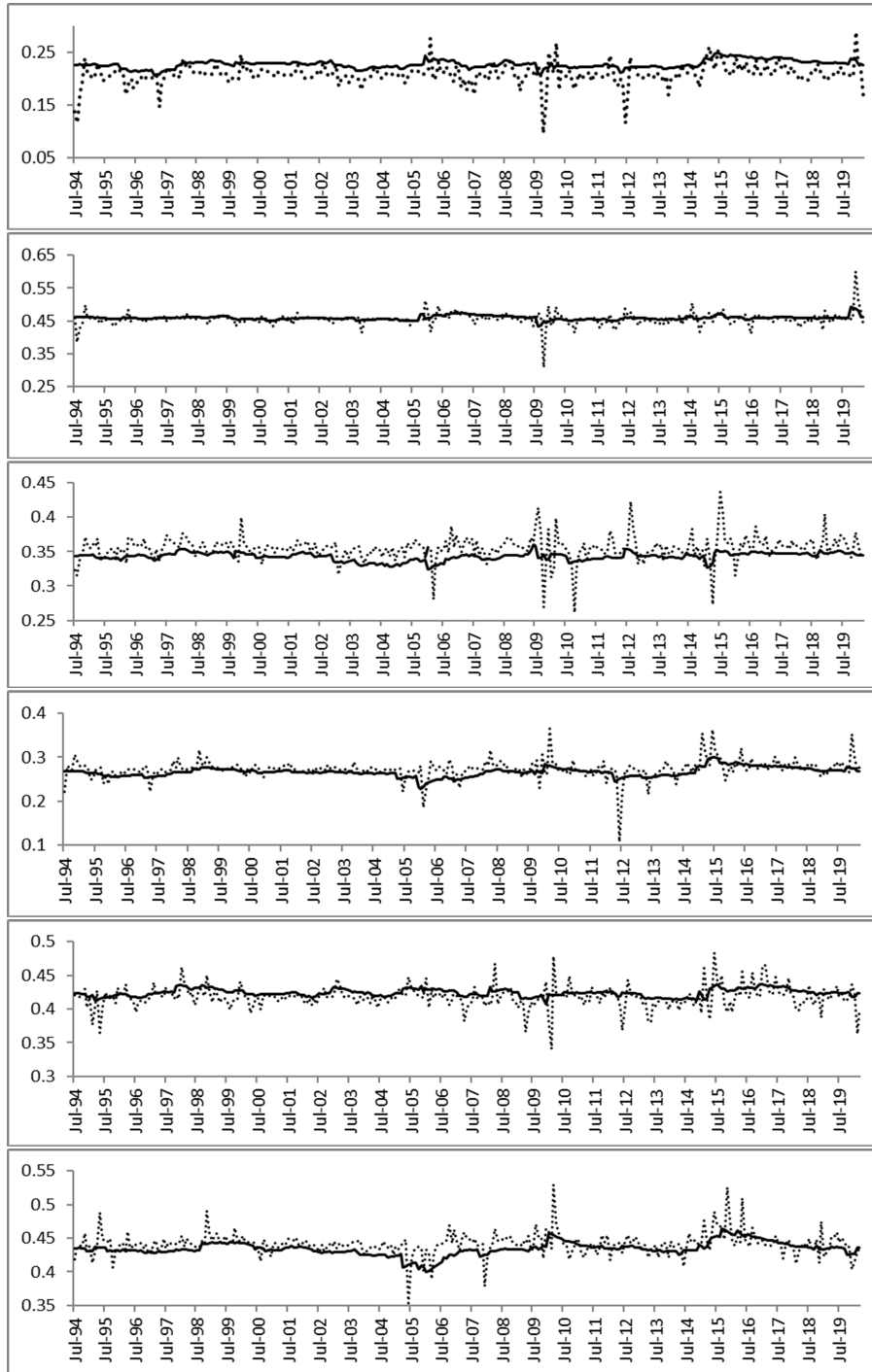


Figure 6: Dynamic conditional correlation plots obtained from MLE (in bold) and Bayesian (in dotted) using multivariate t error distribution of Moong-Masur, Moong-Urad, Moong-Arhar, Masur-Urad, Masur-Arhar, Urad and Arhar (top to bottom)

Table 7: Descriptive statistics of the dynamic conditional correlation

Series	Descriptive Statistics					
	Mean		Minimum		Maximum	
	MLE	Bayesian	MLE	Bayesian	MLE	Bayesian
Moong-Masur	0.227	0.208	0.205	0.095	0.249	0.288
Moong-Urad	0.458	0.455	0.433	0.307	0.492	0.600
Moong-Arhar	0.342	0.354	0.324	0.261	0.359	0.436
Masur- Urad	0.266	0.273	0.229	0.107	0.301	0.365
Masur- Arhar	0.424	0.417	0.407	0.340	0.437	0.483
Urad-Arhar	0.434	0.441	0.399	0.353	0.463	0.529

Table 8: Individual GARCH (1,1) estimates under different prior settings for Bayesian DCC-MGARCH model

Prior	Distribution	α_{Moong}	β_{Moong}	α_{Masur}	β_{Masur}	α_{Urad}	β_{Urad}	α_{Arhar}	β_{Arhar}	DIC
Prior 1	MVN	0.208	0.771	0.248	0.717	0.146	0.818	0.279	0.660	5708.382
Prior 2		0.193	0.785	0.229	0.729	0.151	0.813	0.285	0.668	5705.396
Prior 1	MVT	0.153	0.821	0.163	0.774	0.177	0.750	0.247	0.654	5553.357
Prior 2		0.153	0.821	0.163	0.774	0.177	0.750	0.247	0.654	5553.357

Table 9: Values of the original and forecasted conditional variance

Months	Original Values				Forecasted Conditional variance			
	Moong	Masur	Urad	Arhar	Moong	Masur	Urad	Arhar
Apr-20	161.9	166.6	180.4	155.4	20.1	43.8	42.6	12.7
May-20	167.1	167.8	187.5	156.7	19.5	41.6	40.6	13.6
Jun-20	163.7	172.6	183.8	158.3	19.0	39.7	38.9	14.3
Jul-20	161.4	175.3	181.5	159.8	18.5	37.8	37.3	14.9
Aug-20	158.4	175.9	180.7	160.6	18.1	36.2	35.9	15.4
Sep-20	152.7	177.0	177.4	165.5	17.7	34.7	34.7	15.8
Oct-20	151.2	181.2	181.3	173.3	17.3	33.3	33.6	16.1
Nov-20	153.5	185.1	187.9	174.2	16.9	32.0	32.6	16.5
Dec-20	156.6	182.3	190.8	168.8	16.6	30.8	31.7	16.8
Jan-21	155.0	184.1	193.8	166.5	16.3	29.7	30.9	17.0
Feb-21	157.8	185.1	196.4	172.9	16.0	28.7	30.1	17.3
Mar-21	157.3	187.4	196.5	174.7	15.7	27.7	29.4	17.5

model parameters. To add into the results, we also obtained the optimal prior setting for the dataset which is robust to the change in the distributional parameters and the sample size of the MCMC algorithm. The pulses series were found to have first order regressive relation among themselves. DCC-MGARCH models provided exciting insights into the dynamicity of the pulses. Results of this investigation have revealed the existence of dynamic co-movement among them with highest between Moong-Urad and lowest in Moong-Masur combinations. The findings of this study can better assist the stakeholders in formulating policy decisions focusing on enhancing income of the pulse cultivators of India. This study can be extended using different advanced MCMC algorithms and other multivariate volatility models.

Acknowledgements

The authors are grateful to the editor and anonymous reviewers for their valuable comments and suggestions.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Bauwens, L., Laurent, S., and Rombouts, J. V. K. (2006). Multivariate GARCH models: a survey. *Journal of Applied Econometrics*, **21**, 79–109.
- Bollerslev, T. (1986). Generalized autoregressive conditional heteroscedasticity. *Journal of Econometrics*, **31**, 307–327.
- Bollerslev, T. (1990). Modeling the coherence in short-run nominal exchange rates: a multivariate generalized ARCH model. *Review of Economic and Statistics*, **72**, 498–505.
- Chen, Y., Zheng, B., and Qu, F. (2020). Modeling the nexus of crude oil, new energy and rare earth in China: an asymmetric VAR-BEKK (DCC)-GARCH approach. *Resources Policy*, **65**, 101545.
- Chevallier, J. (2012). Time-varying correlations in oil, gas and CO2 prices: An application using BEKK, CCC and DCC-MAGARCH models. *Applied Economics*, **44**, 4257–4274.
- Engle, R. (2002). Dynamic conditional correlation. *Journal of Business & Economic Statistics*, **20**, 339–350.
- Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of U.K. inflation. *Econometrica*, **50**, 987–1008.
- Engle, R. F. and Kroner, K. F. (1995). Multivariate simultaneous generalized ARCH. *Econometric Theory*, **11**, 122–150.
- Engle, R. F. and Sheppard, K. (2001). Theoretical and empirical properties of dynamic conditional correlation multivariate garch. Working paper, National Bureau of Economic Research.
- Fama, E. F. (1965). The behavior of stock-market prices. *The Journal of Business*, **38**, 34.
- Fioruci, J. A., Ehlers, R. S., and Filho, M. G. A. (2014). Bayesian multivariate GARCH models with dynamic correlations and asymmetric error distributions. *Journal of Applied Statistics*, **41**, 320–331.
- Gupta, R., Kanda, P. T., Tiwari, A. K., and Wohar, M. E. (2019). Time-varying predictability of oil market movements over a century of data: The role of US financial stress. *The North American Journal of Economics and Finance*, **50**, 100994.
- Hall, P. and Yao, Q. (2003). Inference in ARCH and GARCH models with heavy-tailed errors. *Econometrica*, **71**, 285–317.
- Hull, J. and White, A. G. (1998). Incorporating volatility updating into the historical simulation method for value at risk. *Journal of Risk*, **1**, 5–19.

- Ku, Y.-H. H. (2008). Student-t distribution based VAR-MGARCH: an application of the DCC model on international portfolio risk management. *Applied Economics*, **40**, 1685–1697.
- Lama, A., Jha, G. K., Gurung, B., Paul, R. K., and Sinha, K. (2016). VAR-MGARCH models for volatility modelling pulses prices. *Journal of the Indian Society of Agricultural Statistics*, **70**, 145–151.
- Lama, A., Jha, G. K., Paul, R. K., and Gurung, B. (2015). Modelling and forecasting of price volatility: An application of GARCH and EGARCH models. *Agricultural Economics Research Review*, **28**, 73–82.
- Lean, H. H. and Teng, K. T. (2013). Integration of world leaders and emerging powers into the Malaysian stock market: A DCC-MGARCH approach. *Economic Modelling*, **32**, 333–342.
- Lin, B. and Li, J. (2015). The spillover effects across natural gas and oil markets: Based on the VEC-MGARCH framework. *Applied Energy*, **155**, 229–241.
- Livingston, G. C. and Nur, D. (2023). Bayesian inference of multivariate-GARCH-BEKK models. *Statistical Papers*, **64**, 1749–1774.
- Miazhynskaia, T. and Dorffner, G. (2006). A comparison of Bayesian model selection based on MCMC with an application to GARCH-type models. *Statistical Papers*, **47**, 525–549.
- Mohammadi, H. and Tan, Y. X. (2015). Return and volatility spillovers across equity markets in mainland China, Hong Kong and the United States. *Econometrics*, **3**, 215–232.
- Shiferaw, Y. A. (2019). Time-varying correlation between agricultural commodity and energy price dynamics with Bayesian multivariate DCC-GARCH models. *Physica A: Statistical Mechanics and its Applications*, **526**, 120807.
- Sims, C. A. (1977). Macroeconomics and reality. *Econometrica*, **48**, 1–48.

ANNEXURE

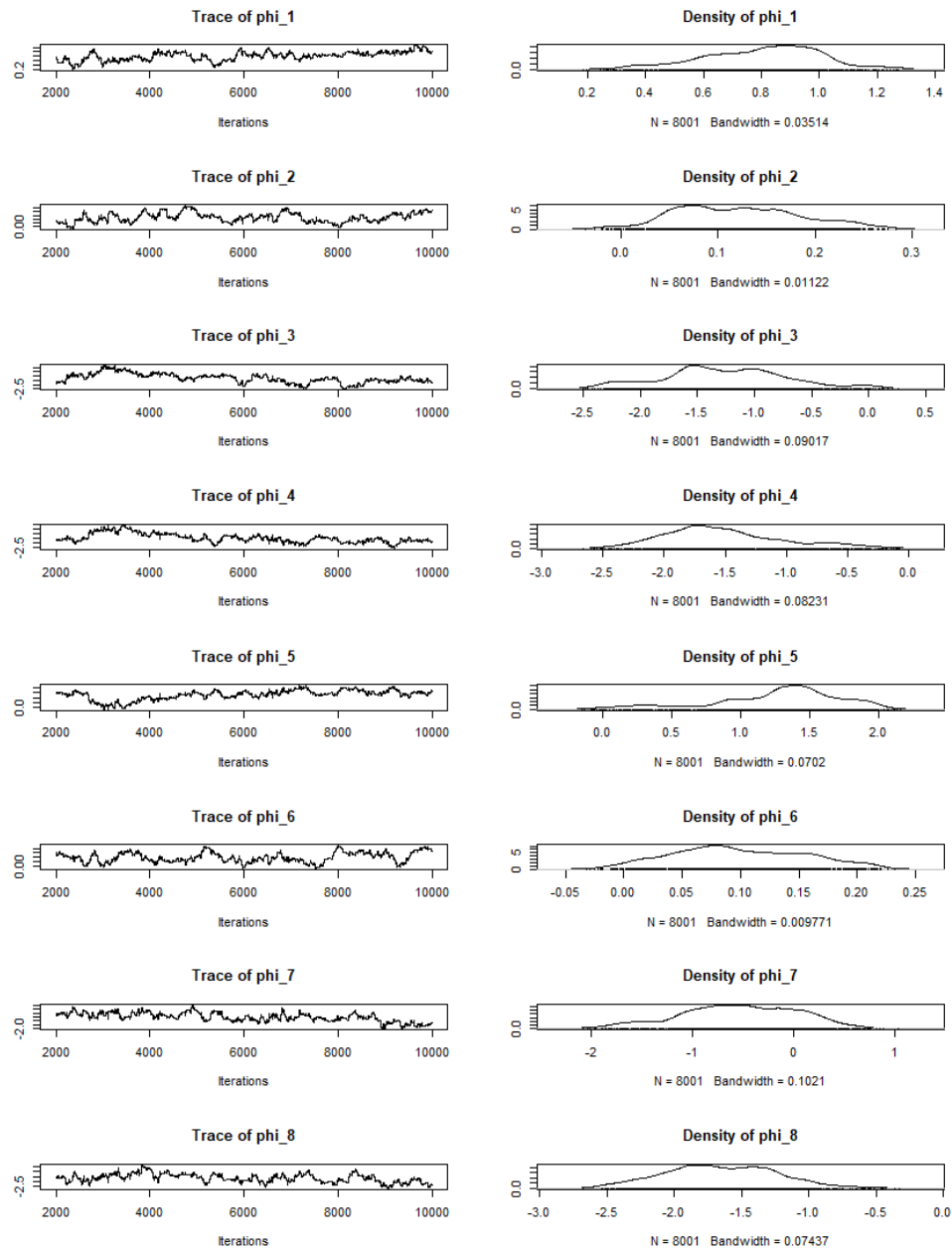


Figure 7: Trace plots

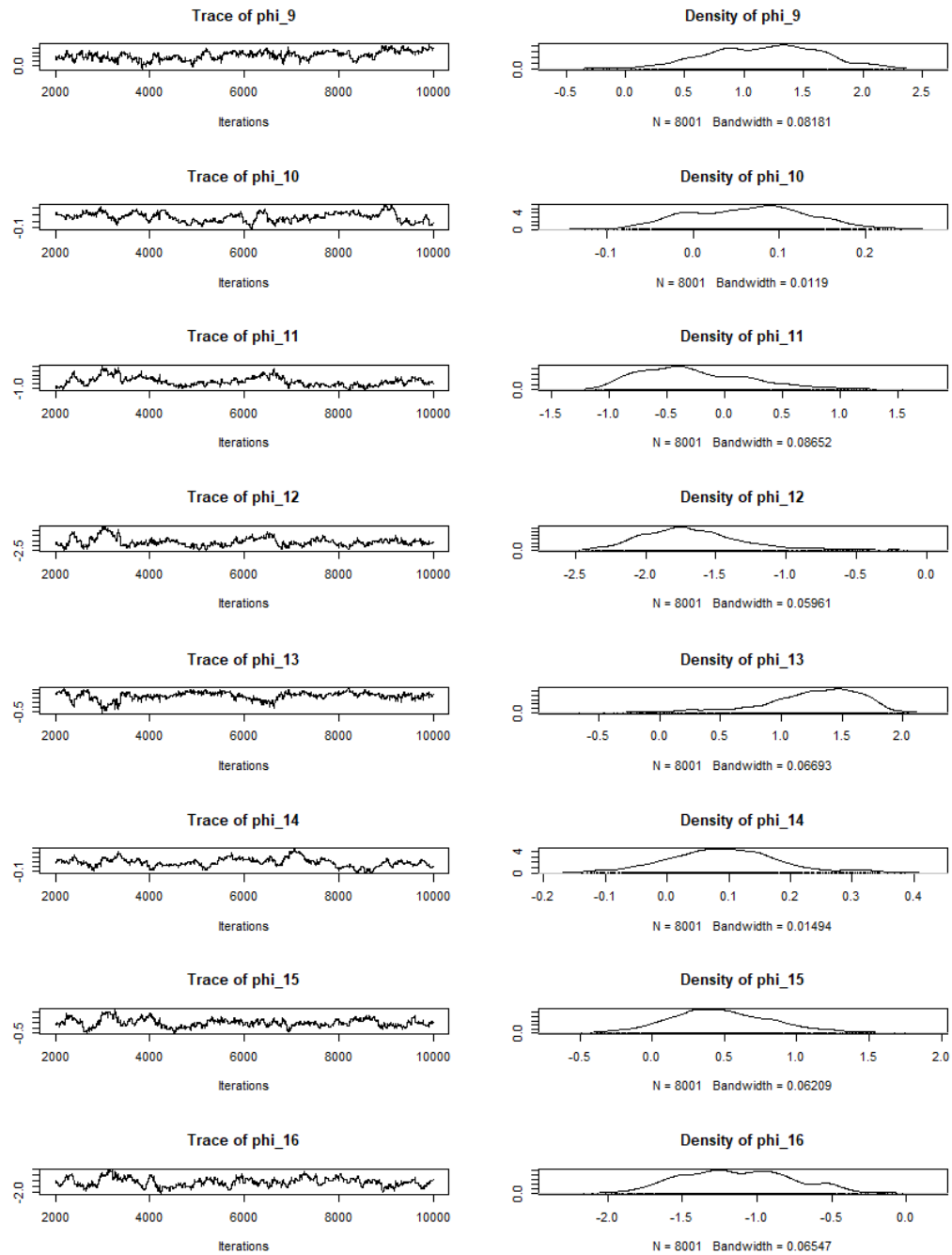


Figure 8: Trace plots

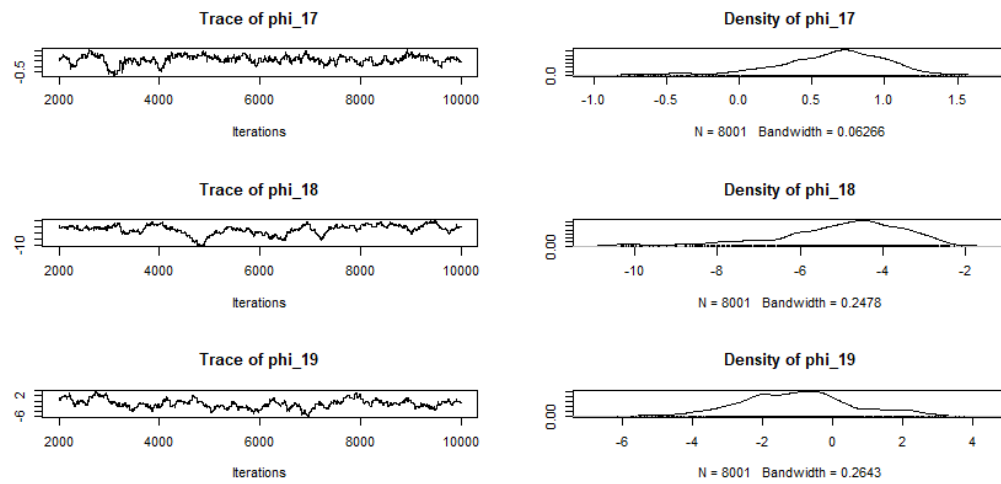
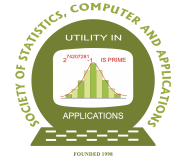


Figure 9: Trace plots



Nonparametric Tests Based on Ranks for Independence against Weighted Alternative with Missing Values

Parameshwar V. Pandit¹, Deepa Yogesh Kamat² and Shubhashree Joshi³

¹*Department of Statistics, Bangalore University, Bangalore-560056, India*

²*Department of Statistics, Nrupathunga University, Bengaluru-560001, India*

³*Department of Statistics, St. Joseph's University, Bengaluru-560027, India*

Received: 17 March 2024; Revised: 12 August 2024; Accepted: 03 February 2025

Abstract

The problem of testing independence of bivariate random variables against a weighted alternative model with possible missing values on both responses is considered in this study. Shie, Bai and Tsai (2000) studied the problem of independence with a contaminated alternative. Pandit (2006) considered the problem of independence against weighted alternatives. In this paper, a generalization of Pandit (2004) is considered the alternative with weighted contamination on both the variables for testing independence against dependence. A new rank test based on ranks is proposed and its asymptotic normality is established. Also, locally most powerful test for the above stated model due to Pandit (2004) is derived when the sample has missing values. The asymptotic null distributions of the test statistics are also provided for the purpose of practical use.

Key words: Independence tests; Locally most powerful rank test; Weighted dependence alternative; Missing values.

AMS Subject Classifications: 62G10

1. Introduction

One of the earliest problems that has attracted many statisticians working in the area of nonparametric testing is the problem of independence in bivariate situations. Many researchers have attempted to quantify the concept of stochastic dependence for bivariate distributions. Researchers such as Spearman (1904), Kendall (1962), Bhuchongkul (1964), Puri and Sen (1971), Shirahata (1974) among others proposed measure of dependence in bivariate distributions to develop rank test for independence. Pearson's correlation coefficient based on Savage (1956) scores was proposed by Iman and Conover (1987) with a view to give more weight on top ranks. In testing independence, the locally most powerful rank test (LMPT) against weighted alternatives was studied by Shei, Bai and Tsai (2000) along with some new rank tests. Pandit (2006) proposed some rank tests with different weighted

alternative and LMPR was derived. In practical situations to handle missing observations the tests mentioned above cannot be applied. For this set up a locally most powerful rank test (LMPR) for independence was derived with alternative due to Hajek and Sidak (1967). Pandit and Savitha Kumari (2015) proposed some new rank tests for testing independence against the alternative due to Hajek and Sidak (1967) and Shei, Bai and Tsai (2000). They also derived LMPR test for independence against the alternative due to Shei, Bai and Tsai (2000). Some of the alternatives for independence considered in the literature are stated below. The first family which is due to Hajek and Sidak (1967) is given by

$$(1) X = X^* + \Delta Z, Y = Y^* + \Delta Z \text{ where } X^*, Y^* \text{ and } Z \text{ are mutually independent.}$$

A similar model due to Shei, Bai and Tsai (2000) with some weights on the first component is

(2) $X = X^* + u(X^*)\Delta Z, Y = Y^* + \Delta Z$ where X^*, Y^* and Z are mutually independent and $u(x)$ is monotone in x and y respectively. However, a more general model which is due to Pandit (2006) is given by

(3) $X = X^* + u(X^*)\Delta Z, Y = Y^* + w(Y^*)\Delta Z$ where X^*, Y^* and Z are mutually independent and $u(x), w(y)$ are monotone in x and y respectively.

In all the models considered above, Δ is a dependence parameter and $\Delta = 0$ implies that X and Y are independent. The model (1) is a straight forward model which makes both X and Y identically distributed if X^* and Y^* are identically distributed where as in (2) and (3), even if X^* and Y^* are identically distributed, X and Y need not be identically distributed. Here, Models (2) and (3) are weighted models. However, model (3) is the most general which includes models (1) and (2) as its particular cases. If $u(x) = w(y) = 1$, then (3) reduces to (1) and if $w(y) = 1$, (3) reduces to (2). In other words, the models due to Hajek and Sidak (1967) and Shei, Bai and Tsai (2000) are particular cases of model due to Pandit (2006). Based on a random sample $((X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n), (X_{n+1}, \cdot), \dots, (X_{n+m}, \cdot), (\cdot, Y_{n+1}), \dots, (\cdot, Y_{n+k}))$ from a bivariate distribution function $F(x, y)$, we consider the problem of testing $H_0 : F(x, y) = F_1(x).F_2(y)$, for all (x, y) against $H_0 : F(x, y) \neq F_1(x).F_2(y)$, for some (x, y) . Here, the alternative considered is a more general weighted alternative (3) due to Pandit (2006).

Under (3), it is clear that if $\Delta = 0$, X and Y are independent and larger the Δ is more dependent are X and Y . Thus, the constant Δ may be regarded as a dependence or mixing coefficient. The alternatives stated in (3) indicate the positive dependence of the random variables X and Y . If we assume negative dependence between X and Y , then the model (3) is (3*) $X = X^* + u(X^*)\Delta Z, Y = Y^* - w(Y^*)\Delta Z$ and $X = X^* - u(X^*)\Delta Z, Y = Y^* + w(Y^*)\Delta Z$ where X^*, Y^* and Z are mutually independent and $u(x), w(y)$ are monotone in x and y respectively.

The first coordinates $X_1, X_2, \dots, X_{n+m}$ of the sample having the ranks $R_1, R_2, \dots, R_{n+m}$ and the the second coordinates $Y_1, Y_2, \dots, Y_{n+k}$ of the sample have the ranks $Q_1, Q_2, \dots, Q_{n+k}$. We present rank tests for testing independence against the alternative (3) due to Pandit (2006). Naturally the test statistics proposed here are the functions of R_i 's and Q_j 's, $i = 1, 2, \dots, n + m, j = 1, 2, \dots, n + k$.

In Section 2, a new test based on rank statistic for testing independence is proposed and the LMPR test for the alternative (3) is considered in Section 3. Power comparison is

given in Section 4. Section 5 is devoted to some remarks and conclusions.

2. Testing independence based on rank statistic

The model of dependence considered is that considered in Pandit (2006). The model for a random sample with missing values is as specified below. Let

$$X_i = X_i^* + \Delta u(X_i^*) Z_i, \quad Y_i = Y_i^* + \Delta w(Y_i^*) Z_i, \quad i = 1, 2, \dots, n$$

$$X_j = X_j^* + \Delta u(X_j^*) Z_j, \quad Y_l = Y_l^* + \Delta w(Y_l^*) Z_{m+l}, \quad j = 1, 2, \dots, m; l = 1, 2, \dots, k.$$

The variables X^* , Y^* and Z are independent and Δ is a real nonnegative parameter.

Spearman (1904) introduced the correlation coefficient-based ranks of (X_i, Y_i) , $i = 1, 2, \dots, n$. Further, Shei, Bai and Tsai (2000) proposed a statistic based on rank correlation with weights. In this paper, the statistic is proposed by extending it to suit the above model (3). The proposed test statistic for testing bivariate independence against model (3) is given by,

$$S_P = \sum_{i=1}^n u_i w_i \left(R_i - \frac{n+m+1}{2} \right) \left(Q_i - \frac{n+k+1}{2} \right),$$

where, $u_i = I(i \leq n_1^*)$, $w_i = I(i \leq n_2^*)$, where $n_1^* = (n+1)p$, $n_2^* = (n+1)q$ and $0 \leq p, q \leq 1$, where p, q roughly the proportion of the observed items in X and Y respectively. The choice of p, q is to have less loss in level of significance. (Pandit and Savitha Kumari (2015)). The asymptotic distribution of the statistic S_P is given in the following theorem. Let $I(f)$ denote the Fisher information, $I(f) = \int_{-\infty}^{\infty} \left(\frac{f'(x)}{f(x)} \right)^2 f(x) dx$

2.1. Theorem 1

Assume that H_0 holds, $I(f_{10}) < \infty$ and $I(f_{20}) < \infty$. Then, $\frac{S_p}{\sqrt{\frac{n(n^2-1)}{12}}}$ converges in distribu-

tion to normal variate with mean zero and variance σ^2 where $\sigma^2 = \begin{cases} p((1-p)^3 + p^2) & p < q \\ q((1-q)^3 + q^2) & p \geq q \end{cases}$

Proof: Here the statistic S_p is a linear rank statistic with $E(S_p) = 0$. Assuming that $I(f_{10}) < \infty$ and $I(f_{20}) < \infty$, the asymptotic variance of $\frac{S_p}{\sqrt{\frac{n(n^2-1)}{12}}}$ under H_0 is given by,

$$\text{Var} \left(\frac{S_p}{\sqrt{\frac{n(n^2-1)}{12}}} \right) \rightarrow \sigma^2 = \begin{cases} p((1-p)^3 + p^2) & p < q \\ q((1-q)^3 + q^2) & p \geq q \end{cases} \text{ as } n \rightarrow \infty. \text{ By applying theorem 1.6a of}$$

chapter V in Hajek and Sidak (1967), $\frac{S_p}{\sqrt{\frac{n(n^2-1)}{12}}}$ converges in distribution to normal variate

with mean zero and variance σ^2 where $\sigma^2 = \begin{cases} p((1-p)^3 + p^2) & p < q \\ q((1-q)^3 + q^2) & p \geq q \end{cases}$

3. Locally most powerful rank test for weighted alternative

To develop locally most powerful rank (LMPR) test we consider the model due to Pandit (2006). For that, X^* and Y^* are the absolutely continuous random variables with

probability density functions (p.d.f) $f_{10}(x)$ and $f_{20}(y)$ respectively and the distribution of Z is arbitrary. For given $(x, (\Delta)z)$ the equation $x = x^* + w(x^*)\Delta z$ has the unique solution which is given by $X^* = t(x, \Delta z)$ and for $(y, \Delta z)$ the equation $y = Y^* + u(Y^*)\Delta z$ has the unique solution which is given by $Y^* = t(y, \Delta z)$. Then the Jacobian is

$$|J| = \left| \frac{\partial(x^*, y^*, z)}{\partial(x, y, z)} \right| = \{(1 + \Delta z u(x^*))(1 + \Delta z w(y^*))\}^{-1}$$

The joint p.d.f., g_Δ , of $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n), (X_{n+1}, \cdot), \dots, (X_{n+m}, \cdot), (\cdot, Y_{n+1}), \dots, (\cdot, Y_{n+k})$ is given by

$$g_\Delta = \prod_{i=1}^n h_\Delta(x_i, y_i) \cdot \prod_{j=n+1}^{n+m} f_{10}(x_j^*) \cdot \prod_{l=n+1}^{n+k} f_{20}(y_l^*)$$

where $h_\Delta(x, y) = \int_{-\infty}^{\infty} f_{10}(x^*) \cdot f_{20}(y - \Delta z) \cdot dM(z)$ and $M(z)$ is the distribution function of Z with mean μ_z and finite variance σ_z^2 . Let $X_{(i)}$ and $Y_{(i)}$ be the i -th order statistic of $\{X_1, X_2, \dots, X_{n+m}\}$ and $\{Y_1, Y_2, \dots, Y_{n+k}\}$, respectively.

Further, let $a_{n+m}(r_i, f_{10}) = E \left\{ - \left(\frac{u f_{10}}{f_{10}} \right)' (X_{(i)}) \right\}$ and $a_{n+k}(q_i, f_{20}) = E \left\{ - \left(\frac{w f_{20}}{f_{20}} \right)' (Y_{(i)}) \right\}$ denote the score functions corresponding to f_{10} and f_{20} respectively. In order to obtain LMPR test we assume the following conditions:

- (i) The derivatives $(u f_{10})'$ and $(w f_{20})'$ are continuous,
- (ii) $\int_{-\infty}^{\infty} |(u f_{10})'(x)| dx < \infty$ and $\int_{-\infty}^{\infty} |(w f_{20})'(x)| dx < \infty$,
- (iii) The missing observations on either the first coordinate or the second coordinate occur at random.

The following theorem states the LMPR test.

3.1. Theorem 2

Under the conditions (i) to (iii) and for the model (1), the test statistic $D_1 = \sum_{i=1}^n a_{n+m}(r_i, u, f_{10}) \cdot a_{n+k}(q_i, w, f_{20})$ with critical region $D_1 > c$, where c is a constant, is locally most powerful rank test for testing $H_0 : \Delta = 0$ against $H_1 : \Delta > 0$ at corresponding level of significance.

Corollary 1: If f_{10} and f_{20} are from Logistic family, then the test based on S_P with critical region $S_P \geq c$, where c is a constant, is asymptotic LMPR test for testing $H_0 : \Delta = 0$ against $H_1 : \Delta > 0$ for model (3).

4. Power comparison

In this Section, we compare the powers of the proposed test S_P with those of top-down statistic r_T and Kendall's τ . The alternative used is as in (1). For power comparisons, the alternatives considered are

1. X_i, Y_i and Z_i follow normal with mean zero and variance one.
2. X_i, Y_i and Z_i follow logistic $(0, 1)$.

The correlation coefficient between X and Y denoted by ρ in terms of Δ is

$$\rho = \frac{pq\Delta^2\sigma_z^2}{\sqrt{(\sigma_x^2 + p\Delta^2\sigma_z^2)(\sigma_y^2 + q\Delta^2\sigma_z^2)}}$$

Here it is to be noted that $\rho = 0$ implies the independence. For simulation selected values of ρ are considered when $p = 0.9$.

Table 1: Empirical powers of new test S_P top-down statistic r_T and Kendall's τ for $n = 20$

m	k	ρ	$N(0, 1)$			$Logistic(0, 1)$		
			S_p	r_T	τ	S_p	r_T	τ
2	2	0.1	0.1325	0.0994	0.0812	0.1625	0.0872	0.0905
		0.4	0.3242	0.1089	0.1582	0.3255	0.0951	0.1567
		0.7	0.5122	0.1279	0.2665	0.5441	0.1056	0.2381
2	3	0.1	0.2624	0.1552	0.1652	0.2847	0.2311	0.2153
		0.4	0.6234	0.557	0.5284	0.7995	0.6732	0.5932
		0.7	0.9132	0.9081	0.8973	0.9653	0.8973	0.8693
3	3	0.1	0.2925	0.1072	0.0929	0.317	0.1991	0.2125
		0.4	0.6356	0.1187	0.1696	0.717	0.5862	0.5685
		0.7	0.9212	0.1286	0.2765	0.9664	0.8984	0.8791
3	4	0.1	0.3514	0.1645	0.1673	0.3558	0.2732	0.2053
		0.4	0.6531	0.5483	0.5096	0.8019	0.6578	0.5937
		0.7	0.9437	0.9104	0.8973	0.9762	0.9073	0.8774

Table 2: Empirical powers of new test S_P top-down statistic r_T and Kendall's τ for $n = 50$

m	k	ρ	$N(0, 1)$			$Logistic(0, 1)$		
			S_p	r_T	τ	S_p	r_T	τ
2	2	0.1	0.1365	0.1024	0.0836	0.1674	0.0898	0.0932
		0.4	0.3339	0.1122	0.1629	0.3352	0.098	0.1614
		0.7	0.5276	0.1317	0.2745	0.5605	0.1088	0.2452
2	3	0.1	0.2703	0.1599	0.1702	0.2932	0.238	0.2218
		0.4	0.6421	0.5737	0.5443	0.8235	0.6934	0.611
		0.7	0.9406	0.9353	0.9242	0.9943	0.9242	0.8954
3	3	0.1	0.3013	0.1104	0.0957	0.3265	0.2051	0.2189
		0.4	0.6547	0.1223	0.1747	0.7385	0.6038	0.5856
		0.7	0.9488	0.1325	0.2848	0.9954	0.9254	0.9055
3	4	0.1	0.3619	0.1694	0.1723	0.3664	0.2814	0.2115
		0.4	0.6727	0.5647	0.5249	0.8259	0.6775	0.6115
		0.7	0.972	0.9377	0.9242	1	0.9345	0.9037

Table 3: Empirical powers of new test S_P top-down statistic r_T and Kendall's τ for $n = 100$

m	k	ρ	$N(0, 1)$			$Logistic(0, 1)$		
			S_P	r_T	τ	S_P	r_T	τ
2	2	0.1	0.1391	0.1044	0.0853	0.4365	0.0916	0.0950
		0.4	0.3404	0.1143	0.1661	0.8221	0.0999	0.1645
		0.7	0.5378	0.1343	0.2798	1.0000	0.1109	0.2500
2	3	0.1	0.2755	0.1630	0.1735	0.5345	0.2427	0.2261
		0.4	0.6546	0.5849	0.5548	0.8751	0.7069	0.6229
		0.7	0.9589	0.9535	0.9422	1.0000	0.9422	0.9128
3	3	0.1	0.3071	0.1126	0.0975	0.6674	0.2091	0.2231
		0.4	0.6674	0.1246	0.1781	0.9843	0.6155	0.5969
		0.7	0.9673	0.1350	0.2903	1.0000	0.9433	0.9231
3	4	0.1	0.369	0.1727	0.1757	0.7203	0.2869	0.2156
		0.4	0.6858	0.5757	0.5351	0.9546	0.6907	0.6234
		0.7	0.9909	0.9559	0.9422	1.0000	0.9527	0.9213

In the Tables 1-4, m is the number of missing observations corresponding to x -values and k is the number of missing observations corresponding to y -values. From the above tables it is easily seen that the test S_P is more powerful than the top-down statistic r_T due to Iman and Conover (1987) and Kendall's test τ .

5. Remarks and conclusion

1. The problem of independence against a weighted alternative (3) when the data have missing values is studied. Pandit (2006) considered the alternative (3) which is the generalization of Hajek and Sidak (1967), accommodating weighted contamination in both coordinates of the pair.
2. A statistic, S_P is proposed for the above stated problem which is the generalization of Spearman's rank correlation coefficient accommodating missing values. The asymptotic normality of the test statistic S_P is established.
3. Locally most powerful rank (LMPR) test for testing independence against the alternative due to Pandit (2006) accommodating missing values is derived.
4. The LMPR for the alternative (1) due to Hajek and Sidak (1967) considered in Wei (1983), and that for the alternative due to Shai, Bai and Tsai (2000) established in Pandit and Savitha Kumari (2015) can be obtained as particular cases of the LMPR derived here.
5. The small sample powers of tests based on S_P , r_T , τ are evaluated using simulation study. It is observed that the power of the test based on S_P is more as compared to the tests due to Iman and Conover (1987) r_T and Kendall's test τ , when the marginal distributions of X^* and Y^* belong to normal and logistic families.
6. The test based on S_P proposed here is shown to be LMPR when the marginal distributions of X^* and Y^* belong to logistic family. It is also observed that test based on

Table 4: Empirical powers of new test S_P top-down statistic r_T and Kendall's τ for $n = 200$

m	k	ρ	$N(0, 1)$			$Logistic(0, 1)$		
			S_P	r_T	τ	S_P	r_T	τ
2	2	0.1	0.1405	0.1054	0.0861	0.4496	0.0924	0.0959
		0.4	0.3437	0.1154	0.1677	0.8468	0.1008	0.1661
		0.7	0.5429	0.1356	0.2825	1.0000	0.1119	0.2524
2	3	0.1	0.2781	0.1645	0.1751	0.5505	0.2450	0.2282
		0.4	0.6608	0.5904	0.5601	0.9014	0.7136	0.6288
		0.7	0.968	0.9626	0.9511	1.0000	0.9511	0.9215
3	3	0.1	0.3101	0.1136	0.0985	0.6874	0.2110	0.2253
		0.4	0.6737	0.1258	0.1798	0.9765	0.6214	0.6026
		0.7	0.9765	0.1363	0.2931	1.0000	0.9523	0.9318
3	4	0.1	0.3725	0.1744	0.1773	0.7419	0.2896	0.2176
		0.4	0.6923	0.5812	0.5402	0.9832	0.6973	0.6293
		0.7	0.9956	0.9650	0.9511	1.0000	0.9617	0.9300

S_P has better performance in terms of small sample power.

Acknowledgements

The authors are indeed grateful to the Editors for their guidance and counsel. The authors also express their gratefulness to the reviewer for valuable comments and suggestions of generously listing many useful references.

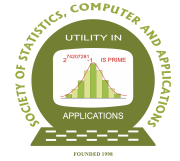
Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Hajek, J. and Sidak, Z. (1967). *Theory of Rank Tests*. Academic Press; New York.
- Iman, R. L. and Conover, W. J. (1987). A measure of top-down correlation. *Technometrics*, **29**, 351-357.
- Kendall, M. G. (1962). *Rank Correlation Methods*. Griffin: London.
- Pandit, P. V. (2006). Locally most powerful and other rank tests for independence-with a contaminated weighted alternative. *Metrika*, **64**, 379-387.
- Pandit, P. V. and Savitha Kumari (2015). Rank tests for independence against weighted alternatives with missing values. *American Journal of Mathematics and Statistics*, **5**, 196-199.
- Puri, M. L. and Sen, P. K. (1971). *Non-parametric Methods in Multivariate Analysis*. Wiley: New York.
- Savage, I. R. (1956). Contributions to the theory of rank order statistics- two-sample case. *Annals of Mathematical Statistics*, **27**, 590-615.
- Shieh, G. S. (1998). A weighted Kendal's statistic. *Statistics and Probability Letters*, **39**, 17-24.

- Shei, G. S., Bai, Z., and Tsai, W. Y. (2000). Rank tests for independence – with a weighted contamination alternative. *Statistica Sinica*, **10**, 577-593.
- Shirahata, S. (1974). Locally most powerful rank tests for independence. *Bulletin of Mathematical Sciences*, **16**, 11-21.
- Spearman, C. (1904). The proof and measurement of association between two things. *American Journal of Psychiatry*, **15**, 72-101.
- Wei, L. J. (1983). Tests for independence in the presence of missing values. *Austrian Journal of Statistics*, **25**, 85-90.



On Log Odd Burr III Weibull Regression Model and its Application in Survival Analysis

Deepthy G. S., Lakshmi R. and Nicy Sebastian

Department of Statistics

*St. Thomas College (Autonomous), Thrissur, affiliated to the University of Calicut,
680 001, Kerala, India*

Received: 01 September 2024; Revised: 18 February 2025; Accepted: 18 March 2025

Abstract

This paper focuses on parametric forms, particularly the location-scale regression model using the odd Burr III Weibull distribution. This model serves as an alternative to the log odd log-logistic Weibull and log-Weibull regression model for analyzing data with decreasing, increasing, unimodal, and bathtub-shaped failure rate functions. Several mathematical properties of the log-transformed distribution have been established. The suggested distribution has the benefit that it encompasses various classical distributions as submodels found in the literature. Maximum likelihood and jackknife estimation techniques are used to assess the model parameters. We conduct simulations under different parameter settings to guarantee model robustness. In our proposed model, we employ diagnostic techniques that employ case deletion known as global influence to determine the influential observations. Additionally, we present martingale and modified deviance residuals to figure out the outliers and assess the model assumptions. The performance of the newly developed regression model is exhibited using a real life dataset.

Key words: Regression model; Censored data; Residual analysis; Sensitivity analysis.

AMS Subject Classifications: 62F10, 60E05, 62N01

1. Introduction

In order to develop more reliable and significant models, the trend in recent statistics literature has changed towards introducing novel methods that incorporate additional parameters into the extensive range of continuous univariate distributions. A distribution that has faced a major hurdle is the Weibull distribution, which is well-known for usage in lifetime data analysis. The drawback of this distribution is its ineffectiveness to consider the non-monotone hazard rates, such as bathtub-shaped hazard rates. To address this, researchers are seeking for generalizations of Weibull distribution that might fit the datasets better than the traditional two parameter Weibull models. For instance, exponentiated Weibull distribution proposed by Mudholkar *et al.* (1995), the generalized modified Weibull distribution

presented by Carrasco *et al.* (2008a), and the beta Weibull distribution explored by Lee *et al.* (2007) represent some recent advancements in Weibull generalizations, demonstrating the continued relevance of the Weibull distribution in lifetime data modeling.

In everyday scenarios, a number of factors might influence survival time, which can explain the variability in the time of survival. To assess how these factors influence survival time, it is important to employ proper regression model for censored and time-to-failure data. Thus, describing a probabilistic model for survival time is crucial to design regression models. Different types of regression models exist for this purpose. The location-scale regression model (LSRM), as highlighted by Lawless (2011) stands out among these and is often utilized in clinical trials. In these models, applying a log transformation serves to linearize multiplicative relationships, making the model well-suited for time-to-event analysis. A linear combination is assumed for the log-lifetimes models preserving the characteristics of standard regression models and also incorporating the advantage of handling censored data, effectively managing skewed distributions, and stabilizing variance. Numerous research studies have been carried out employing log-LSRM. Among these, regression analysis using bivariate location-scale models was discussed by He and Lawless (2005) with applications to lifetime data, Silva *et al.* (2010) suggested the log-Weibull extended regression model for censored data, whereas Carrasco *et al.* (2008b) applied log-modified Weibull regression models to evaluate censored data. Cruz *et al.* (2016) investigated the log-odd log-logistic Weibull regression model, whereas Elbatal *et al.* (2022) introduced a new log LSRM based on the odd Perks-Weibull distribution. Furthermore, Zamanah *et al.* (2022) used survival time data from hypertensive patients to illustrate the use of the log-harmonic mixed Weibull-Weibull distribution, a log-modified alpha power-transformed Burr XII regression model was proposed by Anafo (2024) and then analyze a stock market liquidity dataset. The relevance of these distributions also extends to their potential to incorporate non-monotonic failure rate functions in lifetime analysis, a regular occurrence in both survival analysis and reliability studies.

In this study, we discuss a new log-location regression model based on the logarithm of the odd Burr III Weibull distribution, which we call the log odd Burr III Weibull (LOBIIIW) distribution. The modification in the current distribution results in a LSRM that is apt for modelling censored survival times exhibiting bathtub shaped hazard rates. The logarithm of the lifetimes is carried out to regulate variance and establish a more linear relationship with the covariates. This model provide an adequate alternative for the log-logistic regression model. Our suggested distribution should therefore become beneficial in a variety of applications for evaluating survival and reliability data.

Following model execution, it becomes necessary to illustrate the model's assumptions and perform out robustness checks in order to detect any significant observations that may influence the results of the analysis. A study conducted by Cook (1977) established an influence diagnostic approach that uses case deletion technique to evaluate the significance of each observation on parameter estimates. Our proposal suggests using case deletion based diagnostic measures to identify influential data points in the analysis of censored data using LOBIIIW regression models.

The OBIIIW and LOBIIIW distributions are described in Section 2, along with the structural characteristics of the LOBIIIW distribution. In Section 3, we present an location-

scale LOBIIIW regression model. Section 4 presents estimation methods like maximum likelihood estimation and jackknife method and discusses the findings from the simulation studies. Two different kinds of residuals are defined in Section 5 for evaluating deviations from the error assumptions and the existence of outliers. An evaluation of a real-data set in Section 6 highlights the significance of the new model. Conclusions are provided at the end of Section 7.

2. Log Odd Burr III Weibull distribution

Jamal *et al.* (2017) defines the cumulative distribution function (CDF) for the odd Burr III Generating (OBIII-G) family as follows:

$$F(t; c, k, \gamma) = \left\{ 1 + \left[\frac{1 - G(t, \gamma)}{G(t, \gamma)} \right]^c \right\}^{-k},$$

where $c > 0$ and $k > 0$ denotes the shape parameters and γ represents the parameters of the baseline distribution.

A novel extension of the Weibull distribution has been introduced using the OBIII-G family. This novel distribution named the odd Burr III Weibull (OBIIIW) distribution is established by incorporating the Weibull distribution as base distribution in the OBIII-G family.

Consider the random variable T following OBIIIW distribution, characterized by the parameters c, k, a and b . Then, the survival function is given as follows:

$$S(t; c, k, a, b) = 1 - \left\{ 1 + \left[\frac{e^{-(at)^b}}{1 - e^{-(at)^b}} \right]^c \right\}^{-k}, \quad (1)$$

and the corresponding probability density function (PDF) defined by,

$$f(t; c, k, a, b) = ckba^bt^{b-1} \frac{[e^{-(at)^b}]^c}{[1 - e^{-(at)^b}]^{c+1}} \left\{ 1 + \left[\frac{e^{-(at)^b}}{1 - e^{-(at)^b}} \right]^c \right\}^{-k-1}, \quad c, k, a, b > 0, \quad (2)$$

where c, k and b are the shape parameters, and a is the scale parameter.

Usman and Haq (2019) undertook an extensive investigation of the characteristics and applications of the above proposed model. Their conclusions revealed that the model offers the benefit of accommodating various hazard rate patterns including constant, decreasing, increasing, unimodal and bathtub shaped hazard rates (see Figure 1). This flexibility holds particular significance in reliability analysis, where non-monotone hazard rates are frequently encountered in real world data sets.

If the random variable T has the OBIIIW density function (2), then the random variable $Y = \log(T)$ has the log odd Burr III Weibull (LOBIIIW) distribution. Its density function, parameterized by $a = e^{-\mu}$ and $b = \sigma^{-1}$, can be formulated as follows:

$$f(y; c, k, \mu, \sigma) = \frac{ck}{\sigma} \frac{\exp\left[\left(\frac{y-\mu}{\sigma}\right) - c \exp\left(\frac{y-\mu}{\sigma}\right)\right]}{\left[1 - \exp(-\exp(\frac{y-\mu}{\sigma}))\right]^{c+1}} \left\{ 1 + \left(\frac{\exp[-\exp(\frac{y-\mu}{\sigma})]}{1 - \exp(-\exp(\frac{y-\mu}{\sigma}))} \right)^c \right\}^{-k-1}, \quad (3)$$

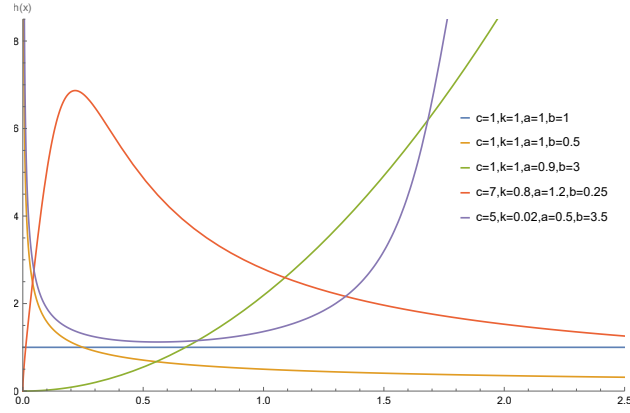


Figure 1: Hazard plot of OBIIIW distribution

where $-\infty < y < \infty$, $c > 0$, $k > 0$, $\sigma > 0$ and $-\infty < \mu < \infty$. We refer to Equation (3) as representing the distribution of $Y \sim LOBIIIW(c, k, \mu, \sigma)$, where μ indicates the location parameter, σ is scale parameter, and c and k are the shape parameters. Figure 2 displays graphs depicting the density function (3) for various parameter values. These findings suggest that this distribution serves as a suitable choice for effectively modeling both left-skewed, right-skewed, and symmetric datasets.

The survival function associated with Equation (3) can be expressed as,

$$S(y; c, k, \mu, \sigma) = 1 - \left\{ 1 + \left[\frac{\exp[-\exp(\frac{y-\mu}{\sigma})]}{1 - \exp[-\exp(\frac{y-\mu}{\sigma})]} \right]^c \right\}^{-k}. \quad (4)$$

Then the standardized random variable $Z = \frac{Y-\mu}{\sigma}$, which is characterized by the density function is given as follows:

$$\pi(z; c, k) = ck \frac{\exp[z - c \exp(z)]}{[1 - \exp(-\exp(z))]^{c+1}} \left\{ 1 + \left(\frac{\exp[-\exp(z)]}{1 - \exp(-\exp(z))} \right)^c \right\}^{-k-1}, \quad -\infty < z < \infty. \quad (5)$$

3. Log Odd Burr III Weibull regression model

Researchers and statisticians have been investigating flexible regression models aimed at effectively capturing non-monotone failure rates, frequently encountered in fields such as reliability and biology. To overcome the limitations of the Weibull distribution in modelling such patterns, a LSRM utilizing the LOBIIIW distribution can be used instead. This model is particularly beneficial for analyzing data characterized by increasing, decreasing and bathtub shaped hazard failures. The importance for the newly developed regression model is its capability to extend the commonly used Weibull distribution to more complex scenarios through continuous extension. This generalization opens up new possibilities for applications that require more sophisticated statistical analysis. Moreover, it is crucial to examine the impact of various explanatory variables on lifetimes such as cholesterol levels, weight, blood pressure *etc.* Regression models, in particular location scale models, give researchers important insight regarding the relationship between lifetime and the explanatory factors. We have now established a linear regression model to analyze the relationship between the

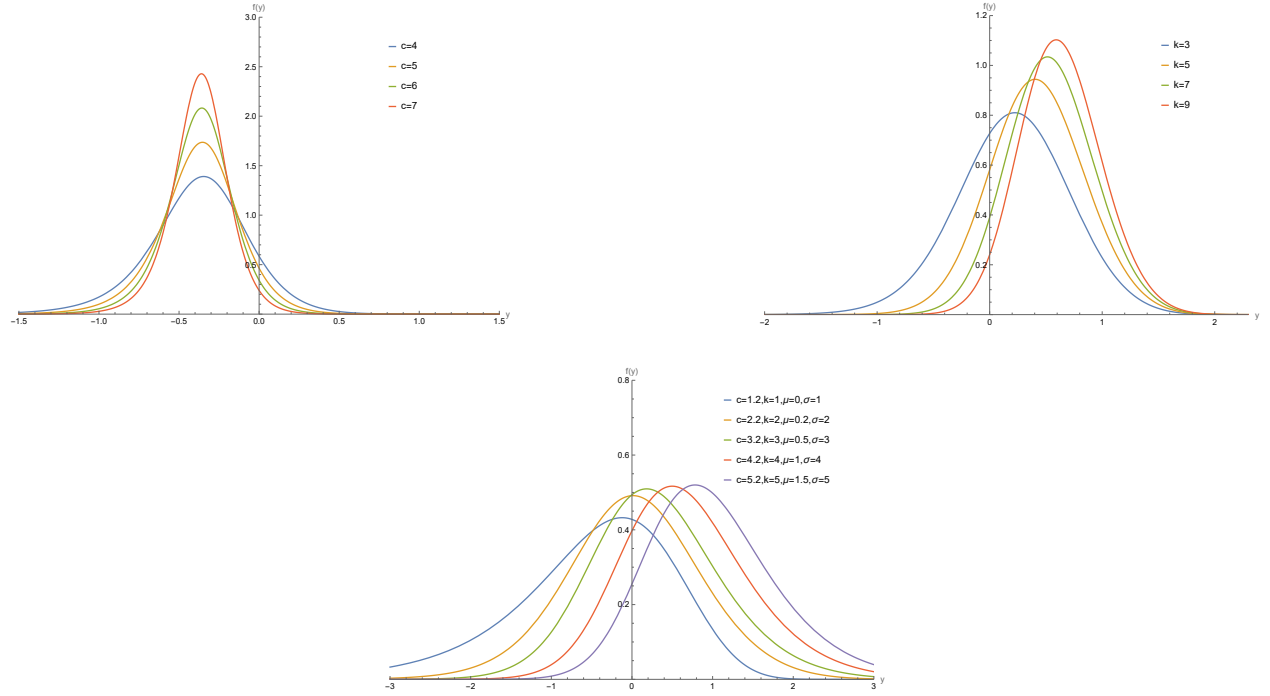


Figure 2: Plots of the LOBIIIW pdf for parameter values (a) $k = 1$, $\mu = 0$, $\sigma = 1$ (b) $c = 1.5$, $\mu = 0$, $\sigma = 1$

response variable y_i and the explanatory variable vector $\mathbf{x}_i = (x_1, x_2, \dots, x_p)^T$ using the LOBIIIW distribution, in the following way:

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \sigma z_i, i = 1, 2, \dots, n. \quad (6)$$

where the error term z_i follows the density function (5), $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_p)^T$, $c > 0$ and $k > 0$ are the unknown parameters. The vector $\mathbf{x}_i^T = (x_{i1}, x_{i2}, \dots, x_{ip})$ are linked with the explanatory variables. The vector $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^T$ of LOBIIIW regression model represents the location parameters and is indicated as linear model $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$ where $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^T$ is a pre-defined model matrix. New opportunities for fitting several kinds of data are made possible by the LOBIIIW model (6).

Using the log linear model given in Equation (6), the survival function $Y_i|\mathbf{x}$ simplifies to,

$$S(y_i|\mathbf{x}) = 1 - \left\{ 1 + \left[\frac{\exp[-\exp(\frac{y_i - \mathbf{x}_i^T \boldsymbol{\beta}}{\sigma})]}{1 - \exp[-\exp(\frac{y_i - \mathbf{x}_i^T \boldsymbol{\beta}}{\sigma})]} \right]^c \right\}^{-k}. \quad (7)$$

The regression model (6) reduces to the log-odd logistic regression model when $k = 1$, and to the log-Weibull regression model when $c = k = 1$.

4. Estimation of the LOBIIIW regression model

4.1. Maximum likelihood estimation method

Let the sample consists of n independent observations $(y_1, \mathbf{x}_1), (y_2, \mathbf{x}_2), \dots, (y_n, \mathbf{x}_n)$, where $y_i = \min\{\log(T_i), \log(C_i)\}$, and $\mathbf{x}_i$ denotes the vector of explanatory variables related

to the i^{th} individual. We consider that the observed lifetimes ($\log(T_i)$) and censoring times (C_i) are independent and the censorship is non-informative. The log-likelihood function for the vector of parameters $\boldsymbol{\eta} = (c, k, \sigma, \boldsymbol{\beta}^T)^T$ is given by,

$$l(\boldsymbol{\eta}) = \sum_{i \in F} l_1(c, k, z_i) + \sum_{i \in C} l_2(c, k, z_i) \quad (8)$$

where,

$$l_1(c, k, z_i) = \log \left[\frac{ck}{\sigma} \frac{\exp[z_i - c \exp(z_i)]}{[1 - \exp(-\exp(z_i))]^{c+1}} \left\{ 1 + \left(\frac{\exp[-\exp(z_i)]}{1 - \exp(-\exp(z_i))} \right)^c \right\}^{-k-1} \right],$$

$$l_2(c, k, z_i) = \log \left[1 - \left\{ 1 + \left[\frac{\exp[-\exp(z_i)]}{1 - \exp[-\exp(z_i)]} \right]^c \right\}^{-k} \right].$$

and $z_i = \frac{y_i - \mathbf{x}_i^T \boldsymbol{\beta}}{\sigma}$. The set F represents individuals for whom y_i corresponds to log-lifetime, whereas C designates individuals subject to right censoring.

Maximizing the likelihood function (8) yields the maximum likelihood estimates (MLEs) for the parameter vector $\boldsymbol{\eta}$. The estimates are obtained by using the R software, specifically utilizing the MAXLIK package.

Furthermore, the LOBIIW model can be compared to particular sub-models using the Likelihood Ratio (LR) test. The maximum log-likelihood values for the unconstrained and constrained models can be used to build the LR statistic for testing specific models within the LOBIIW regression model.

4.2. Simulation

We implement a Monte-Carlo simulation study for examining the behavior of maximum likelihood estimators (MLEs) of c , k , σ , β_0 , and β_1 in a finite sample ($n = 50, 100$, and 300). The samples are obtained using the true parameter values: $c = 2$, $k = 4$, $\sigma = 1.5$, $\beta_0 = 3$ and $\beta_1 = 2$ with varying levels of censoring generally at censoring percentages, $cp = 0\%$, 10% and 30% . The log-lifetimes $\log(T_1), \dots, \log(T_n)$ are simulated from the LOBIIW regression model (6), where $\mathbf{x}_i^T \boldsymbol{\beta} = \beta_0 + \beta_1 x_i$ and x_i are sampled from a uniform distribution within the interval $(0, 1)$. The censored times $C_1, C_2, \dots, C_n$ are chosen from a uniform distribution over the interval $(0, \tau)$, where τ has been altered until the required censoring percentages are attained. Every combination of n , c , k , σ , β_0 , β_1 , and censoring percentages is then used to generate 1000 samples. Each dataset generated was fitted with the LOBIIW regression model (6), where $\mu_i = \beta_0 + \beta_1 x_i$. The mean squared errors (MSEs) and average estimates (AEs) for the MLEs of c , k , σ , β_0 , β_1 computed from simulation results are shown in Table 1. We can see from the data in Table 1 that MSEs increase along with an increase in the censoring percentage. Additionally as predicted, the MSEs decline with sample size increases. These facts indicate that the asymptotic normal distribution accurately approximates the distribution of the estimates from a finite sample.

5. Residual analysis

A key step after fitting a statistical model to the dataset is residual analysis. It serves for various purposes, including validating the data, evaluating the data to unveil valuable

Table 1: AEs and MSEs (in parentheses) of MLEs of c, k, σ, β_0 and β_1

Percentage	Parameter	Sample Size		
		50	100	300
0%	c	2.0489(0.0479)	2.0195(0.0210)	2.0086(0.0072)
	k	4.1593(0.5196)	4.0676(0.2080)	4.0195(0.0682)
	σ	1.3882(0.1946)	1.4689(0.0856)	1.4981(0.0384)
	β_0	3.1711(0.1313)	3.2008(0.0937)	3.2220(0.0688)
	β_1	2.3465(0.2200)	2.3294(0.1647)	2.3203(0.1273)
10%	c	2.0174(0.0494)	1.9894(0.0228)	1.9824(0.0078)
	k	4.3935(0.7953)	4.2926(0.3593)	4.2337(0.1380)
	σ	1.3016(0.3028)	1.3745(0.1242)	1.4457(0.0507)
	β_0	3.1143(0.1505)	3.1500(0.0777)	3.1571(0.0497)
	β_1	2.3528(0.2171)	2.3168(0.1409)	2.2673(0.0913)
30%	c	1.8797(0.0593)	1.8464(0.0437)	1.8328(0.0348)
	k	4.9652(1.8927)	4.8063(0.9988)	4.7374(0.6627)
	σ	1.0040(0.8286)	1.0962(0.4321)	1.0828(0.3557)
	β_0	3.0172(0.2901)	3.0948(0.1148)	3.1736(0.0844)
	β_1	2.4716(0.3867)	2.4157(0.2480)	2.3849(0.1784)

information, and confirming the assumptions of the model. Additionally, residual analysis can assist in identifying outliers and reviewing any deviations from the error assumption. Several types of residual analysis have been mentioned in the literature to achieve these goals. For instance, Collett (2023), McCullagh (2019), Fleming and Harrington (2013) and Ortega *et al.* (2008) have discussed about various residual analyses. This study assesses two types of residuals: martingale-type residual and deviance component residual. The martingale residual (MR) in parametric lifetime models may be defined as $r_{Mi} = \delta_i + \log[S(y_i, \hat{\boldsymbol{\eta}})]$, where $\delta_i = 0(\delta_i = 1)$ signifies censored (uncensored) observation and $S(y_i, \hat{\boldsymbol{\eta}})$ refers to the estimated survival function as specified by Equation 7. In fact, r_{Mi} spans from a minimum of $-\infty$ to maximum of $+1$. Thus, the MR for the LOBIIIW model assumes the following expression:

$$r_{Mi} = \begin{cases} 1 + \log \left\{ 1 - \left(1 + \left[\frac{\exp[-\exp(\hat{z}_i)]}{1 - \exp[-\exp(\hat{z}_i)]} \right]^{\hat{c}} \right)^{-\hat{k}} \right\} & \text{if } \delta_i = 1, \\ \log \left\{ 1 - \left(1 + \left[\frac{\exp[-\exp(\hat{z}_i)]}{1 - \exp[-\exp(\hat{z}_i)]} \right]^{\hat{c}} \right)^{-\hat{k}} \right\} & \text{if } \delta_i = 0, \end{cases}$$

where $\hat{z}_i = \frac{y_i - \mathbf{x}_i^T \hat{\boldsymbol{\beta}}}{\hat{\sigma}}$.

The drawback of the MR is its significant skewness, indicating that it does not closely resemble a normal distribution. To address this issue, Therneau *et al.* (1990) proposed the modified deviance residual (MDR), which transforms the MR to reduce skewness. The advantage of the MDR is that its distribution approximates a normal distribution as closely

as possible, facilitating more effective residual analysis, which can be formulated as:

$$r_{Di} = \text{sign}(r_{Mi}) \{-2[r_{Mi} + \delta_i \log(\delta_i - r_{Mi})]\}^{1/2},$$

where r_{Mi} is the MR.

5.1. Simulation studies

A study using simulation was executed to examine the empirical distributions of residuals r_{Di} for the values $n = 50, 100$ and 300 and censoring percentage 0% , 10% , and 30% , following the procedure outlined in Section 4.2. Following conclusions can be drawn by observing Figure 3:

1. The empirical distribution of the MDR closely matches the standard normal distribution, indicating a strong agreement.
2. The empirical distribution of the MDR tends to converge towards the standard normal distribution as the censoring percentage decreases or the sample size increases.

6. Application

The heart transplant dataset, which includes records up to April 1st, 1974, is detailed in the book by Kalbfleisch and Prentice (2011). During this time frame, certain patients passed away before a suitable heart became available. Out of the 103 patients, 69 underwent heart transplant surgeries, with 75 deaths reported. The remaining 28 patients provided censored survival time data. The survival times from patient entry are represented by the response variable y_i (the number of days from acceptance into the transplantation program to both the transplant and death was recorded for each patient). The failure indicator is shown by the variable “survival status” (0 - censored, 1 - observed). Each patient ($i = 1, 2, \dots, 103$) was related to the following explanatory variables.

- x_{i1} = Age at acceptance(in years).
- x_{i2} = Prior surgery (marked as 1 for yes and 0 for no).
- x_{i3} = Transplant (0 = no; 1 = yes).

In many cases, it is possible to qualitatively evaluate the form of the failure rate function, which is useful when choosing a model. The total time on test (TTT) plot is a useful tool in this context Aarset (1987). Figure 4 demonstrates the TTT plot for this data, which shows a failure rate function has a monotonically increasing pattern. Hence, the LOBIIIW distribution is appropriate for modeling this dataset.

Now, we demonstrate the results obtained from fitting the LOBIIIW regression model:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \sigma z_i.$$

where the errors z_i are independent random variables with a density function (5)

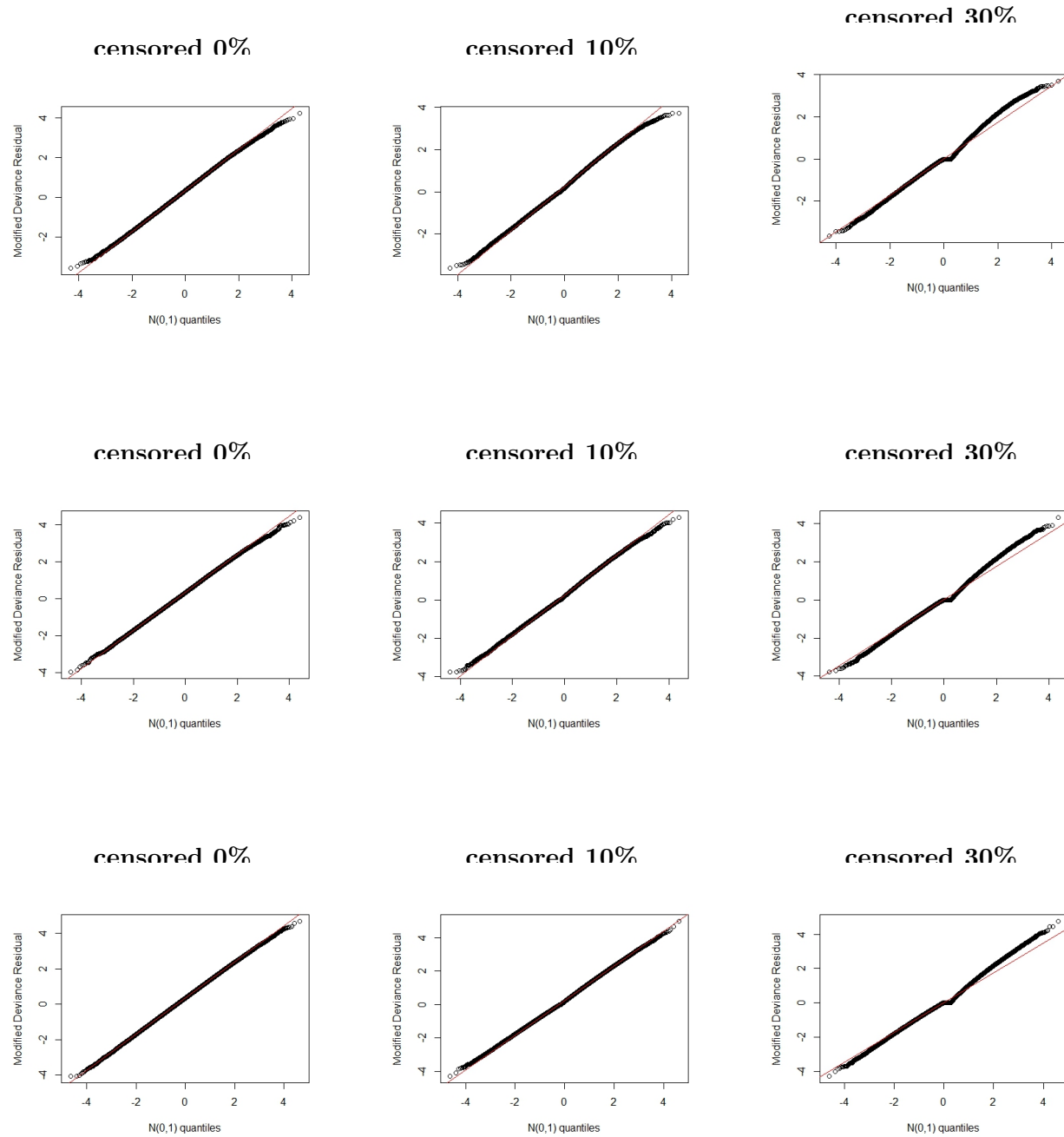


Figure 3: Normal probability plots for the residuals r_{Di} for parameter values $c = 2$, $k = 4$, $\sigma = 1.5$, $\beta_0 = 3$, $\beta_1 = 2$

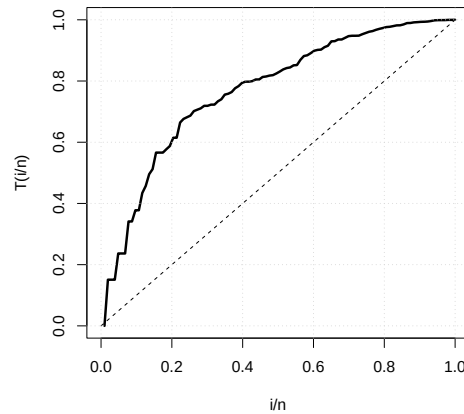


Figure 4: TTT plot of Stanford heart transplant data

6.1. Results of maximum likelihood and jackknife estimation

The statistical program R has been employed for estimating the maximum likelihood estimates of the LOBIIIW, Log Odd Log Logistic Weibull (LOLLW), and Log Weibull (LW) regression models. Table 2 presents the parameter estimates along with their standard errors, p -values, and model selection criteria (AIC, CAIC, and BIC statistics). The smaller standard errors represent more precise parameter estimates and lower p - values reveal that the covariates x_1 , x_2 , and x_3 have a significant impact. A comparative analysis of these models indicates that the LOBIIIW model consistently yields lower values across the model selection criteria. Also, we can analyze the estimated coefficients in the following way: the expected survival time is expected to decreases by approximately 94.08% ($e^{-0.061} \times 100\%$) when the age at acceptance (x_1) increases by one unit. Variables x_2 and x_3 might also be interpreted in the same way. The jackknife method was also used along with maximum likelihood estimation (MLE) to validate the results. The jackknife serves as a diagnostic tool to verify parameter stability since MLE can be sensitive to influential observations and small sample size. Table 3 provides the jackknife estimates of the model parameters. For both estimation procedures, the LOBIIIW regression model's findings indicate that the explanatory variables x_1 , x_2 , and x_3 are statistically significant at the 5% level. Both methods presented nearly identical estimates.

The Cox proportional hazards model was employed as a comparative technique since it is a useful regression model for analyzing censored failure times. Further, estimates, standard errors (SE), and p - values for the Cox regression model are displayed in Table 4. Aligning with the findings of the LOBIIIW regression model, the Cox regression model also shows that all explanatory variables are marginally significant at the 5% level.

Furthermore, the LR statistic is employed to compare the LOBIIIW and LW regression models by testing the hypotheses $H_0 : c = k = 1$ against $H_1 : H_0$ is not true. The test yielded a test statistic value of $w = 2(-166.3611 - (-171.7112)) = 10.7002$, with a corresponding p -value of 0.0047. These results signify a favorable indication towards the

LOBIIIW regression model.

Table 2: The parameter estimates, standard errors (given in parentheses) and p -values in [·], for the LOBIIIW, LOLLW and LW regression models fitted to the heart transplant data

Models	c	k	σ	β_0	β_1	β_2	β_3
LOBIIIW	1.813 (0.722)	2.378 (1.114)	3.506 (1.153)	5.604 (1.254) [< 0.001]	-0.061 (0.019) [0.001]	1.450 (0.589) [0.013]	2.580 (0.378) [< 0.001]
		AIC = 346.722		BIC = 365.165		CAIC=347.901	
LOLLW	4.628 (3.530)		6.203 (4.685)	8.744 (1.760) [< 0.001]	-0.076 (0.019) [< 0.001]	1.405 (0.574) [0.016]	2.591 (0.388) [< 0.001]
		AIC = 347.595		BIC = 363.404		CAIC = 348.470	
LW			1.465 (0.1314)	7.974 (0.933) [< 0.001]	-0.092 (0.020) [< 0.001]	1.214 (0.647) [0.063]	2.537 (0.373) [< 0.001]
		AIC = 353.420		BIC = 366.594		CAIC = 354.039	

Table 3: Jackknife estimates fitted to the heart transplant data

Parameter	Estimates	SE	95%CI	p -value
c	1.8790	0.7775	(0.3550, 3.4028)	0.0156
k	2.6486	1.2789	(0.1419, 5.1553)	0.0383
σ	3.7520	1.2375	(1.3264, 6.1775)	0.0024
β_0	5.6791	1.3454	(3.0420, 8.3161)	< 0.001
β_1	-0.0649	0.0192	(-0.1297, -0.0271)	< 0.001
β_2	1.5309	0.5878	(0.3788, 2.68290)	0.0091
β_3	2.5511	0.3804	(1.8055, 3.2967)	< 0.001

Table 4: Cox regression model estimates fitted to the heart transplant data

Parameter	Estimate	SE	p -value	95%CI
β_1	0.05919	0.01494	<0.0001	(0.0299, 0.0884)
β_2	-0.74266	0.44225	0.0931	(-1.6094, 0.1241)
β_3	-1.66121	0.27588	<0.0001	(-2.2019, -1.1204)
AIC		BIC	CAIC	
554.3522		561.3047	554.6902	

6.2. Results of sensitivity and residual analysis

The global influence measures $GD_i(\eta)$ and $LD_i(\eta)$ have been evaluated to identify the influential observations in the LOBIIW regression model fitted to the present dataset. Figure 5 shows the index plots of influence measures, which signifies that the possible influential observations are 18 and 99. Another way for detecting potential outliers in the LOBIIW regression model involves plotting the MDR (r_{D_i}) against the adjusted values, as shown in Figure 6(a). Upon examining the MDR plot, a residual analysis indicates that observations 8 and 99 are identified as potential outliers.

Both sensitivity and residual analysis determined observations 99 as the most frequently occurring possible influential points. Observation 99 indicates the longest survival time within the present censored dataset. We modified the model by eliminating the above observations in order to determine whether they had a significant effect on the parameter estimates. The Table 5 illustrates the relative change in the estimated parameters, given by the formula $R = \frac{\hat{\eta}_j - \hat{\eta}_{j(i)}}{\hat{\eta}_j}$. Here, $\hat{\eta}_{j(i)}$ represents the MLE estimates calculated excluding the i^{th} record. We evaluate the robustness of the parameters in the LOBIIW regression model, considering that the covariates x_1 , x_2 and x_3 have significance in the fitted model. Thus, the inference remains unchanged when the observation identified as potentially influential in the diagnostic plots is removed.

Also, we display the normal probability plot for MDR with the simulated envelope in Figure 6(b). The plot in Figure 6(b) recommends that the LOBIIW regression model appears to adequately fit the dataset, with no observations occurring as possible outliers.

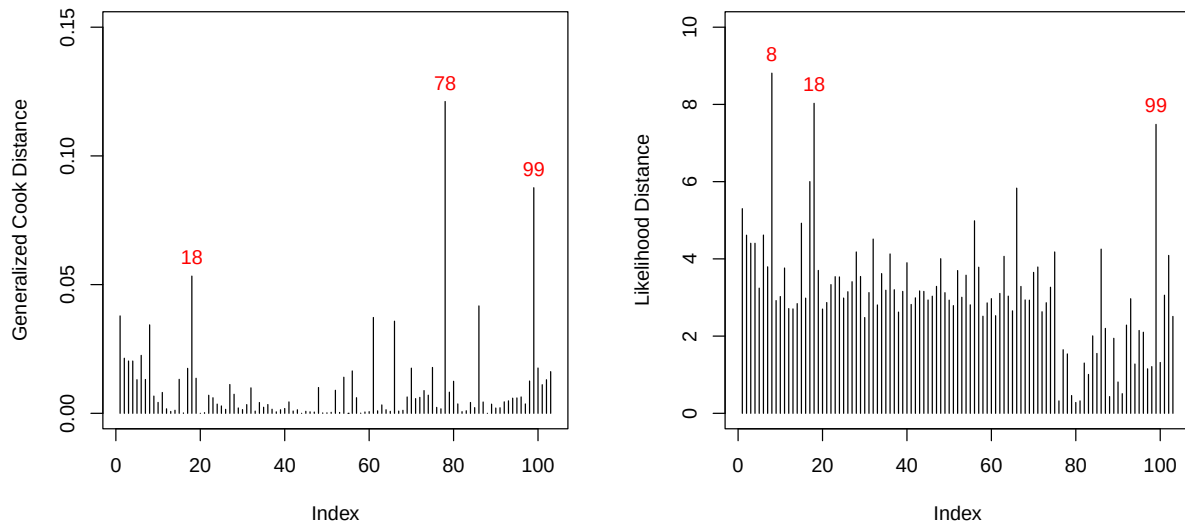


Figure 5: The index plot of (a) $GD_i(\eta)$ and (b) $LD_i(\eta)$ for the LOBIIW regression model

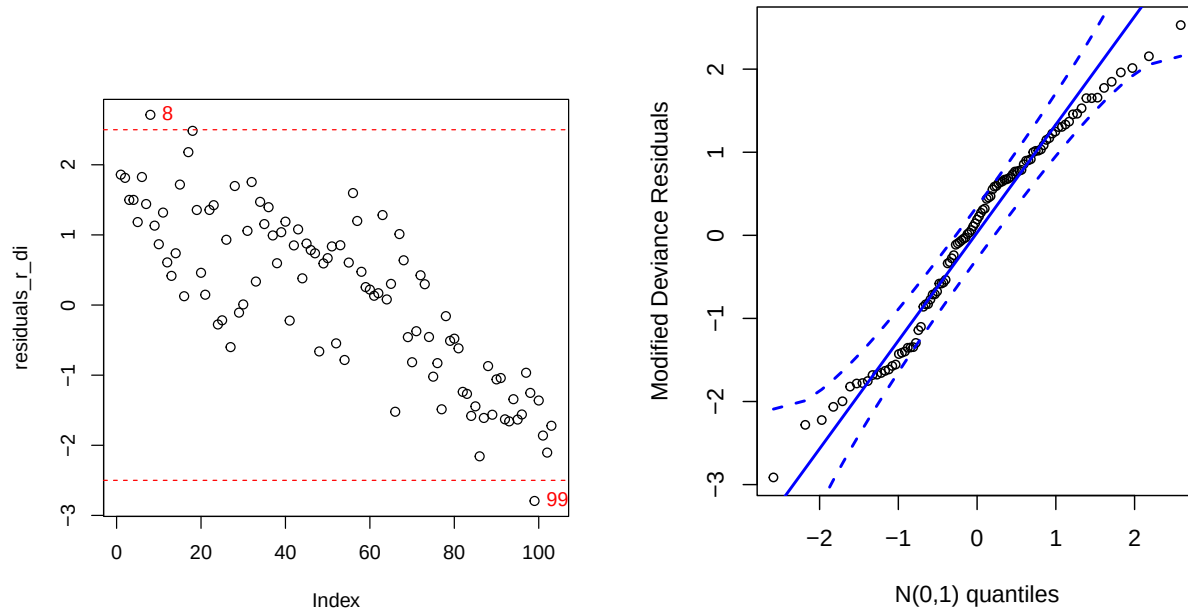


Figure 6: (a) Index plot of the deviance residual and (b) Normal probability plot of r_{Di} with envelopes

Table 5: Relative changes (expressed in percentages) along with the estimates and corresponding p -values for the datasets

Dropped Observation	$\hat{c}$	$\hat{k}$	$\hat{\sigma}$	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$
None	-	-	-	-	-	-	-
	1.8136	2.3785	3.5062	5.6040	-0.0610	1.4501	2.5804
				(< 0.001)	(0.0013)	(0.0138)	(< 0.001)
#99	[-2.2618]	[-12.2317]	[-2.6506]	[3.8708]	[0.2569]	[-6.734]	[-1.4363]
	1.8546	2.6694	3.5991	5.3871	-0.0609	1.5478	2.6175
				(< 0.001)	(< 0.001)	(0.0066)	(< 0.001)

7. Conclusion

This study introduces a novel LOBIIIW regression model for analyzing survival data, particularly when dealing with censored observations. The model performs effectively when the hazard function displays a bathtub form. We used two methods to estimate the parameters of the proposed model: the maximum likelihood and jackknife estimator. The assumptions of the model were verified by utilizing MDR. Simulation research has shown that the empirical distribution of MDR approximates a standard normal distribution. In addition, GCD and LD measures were introduced to detect influential observations in the regression model. We also demonstrate the effectiveness of the model by analyzing a real dataset.

Acknowledgements

The authors express their gratitude to the reviewer and the chief editor for their valuable suggestions, which led to considerable improvements in the research paper.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Aarset, M. V. (1987). How to identify a bathtub hazard rate. *IEEE Transactions on Reliability*, **36**, 106–108.
- Carrasco, J. M., Ortega, E. M., and Cordeiro, G. M. (2008a). A generalized modified Weibull distribution for lifetime modeling. *Computational Statistics and Data Analysis*, **53**, 450–462.
- Carrasco, J. M., Ortega, E. M., and Paula, G. A. (2008b). Log-modified Weibull regression models with censored data: Sensitivity and residual analysis. *Computational Statistics and Data Analysis*, **52**, 4021–4039.
- Collett, D. (2023). *Modelling Survival Data in Medical Research*. Chapman and Hall/CRC.
- Cook, R. D. (1977). Detection of influential observation in linear regression. *Technometrics*, **19**, 15–18.
- Cruz, J. N. d., Ortega, E. M., and Cordeiro, G. M. (2016). The log-odd log-logistic Weibull regression model: modelling, estimation, influence diagnostics and residual analysis. *Journal of Statistical Computation and Simulation*, **86**, 1516–1538.
- Elbatal, I., Alotaibi, N., Almetwally, E. M., Alyami, S. A., and Elgarhy, M. (2022). On odd perks-G class of distributions: properties, regression model, discretization, Bayesian and non-Bayesian estimation, and applications. *Symmetry*, **14**, 883.
- Fleming, T. R. and Harrington, D. P. (2013). *Counting Processes and Survival Analysis*, volume 625. John Wiley & Sons.
- He, W. and Lawless, J. F. (2005). Bivariate location–scale models for regression analysis, with applications to lifetime data. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, **67**, 63–78.
- Jamal, F., Nasir, M. A., Tahir, M., and Montazeri, N. H. (2017). The odd Burr-III family of distributions. *Journal of Statistics Applications and Probability*, **6**, 105–122.
- Kalbfleisch, J. D. and Prentice, R. L. (2011). *The Statistical Analysis of Failure Time Data*. John Wiley & Sons.
- Lawless, J. F. (2011). *Statistical Models and Methods for Lifetime Data*. John Wiley & Sons.
- Lee, C., Famoye, F., and Olumolade, O. (2007). Beta-Weibull distribution: some properties and applications to censored data. *Journal of Modern Applied Statistical Methods*, **6**, 173–186.
- Lipsitz, S. R., Laird, N. M., and Harrington, D. P. (1990). Using the jackknife to estimate the variance of regression estimators from repeated measures studies. *Communications in Statistics-Theory and Methods*, **19**, 821–845.

- Manly, B. F. (2018). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Chapman and Hall/CRC.
- McCullagh, P. (2019). *Generalized Linear Models*. Routledge.
- Mudholkar, G. S., Srivastava, D. K., and Freimer, M. (1995). The exponentiated Weibull family: A reanalysis of the bus-motor-failure data. *Technometrics*, **37**, 436–445.
- Ortega, E. M., Paula, G. A., and Bolfarine, H. (2008). Deviance residuals in generalised log-gamma regression models with censored observations. *Journal of Statistical Computation and Simulation*, **78**, 747–764.
- Silva, G. O., Ortega, E. M., and Cancho, V. G. (2010). Log-Weibull extended regression model: Estimation, sensitivity and residual analysis. *Statistical Methodology*, **7**, 614–631.
- Therneau, T. M., Grambsch, P. M., and Fleming, T. R. (1990). Martingale-based residuals for survival models. *Biometrika*, **77**, 147–160.
- Usman, R. M. and Haq, M. A. u. (2019). Some remarks on odd Burr III Weibull distribution. *Annals of Data Science*, **6**, 21–38.
- Zamanah, E., Nasiru, S., and Luguterah, A. (2022). Harmonic mixture Weibull-G family of distributions: Properties, regression and applications to medical data. *Computational and Mathematical Methods*, **2022**, 2836545.

ANNEXURE

A. Jackknife method

Jackknifing is a technique that converts the task of estimating a population parameter to that of estimating a population mean. Applying this strategy takes a different way to estimating a mean value. By suggesting another robust estimator for the covariance matrix, Lipsitz *et al.* (1990) made a substantial contribution to the implementation of the jackknife method. This estimator relies on the jackknife method and is valid for analyzing data from repeated measure analyses. This method can be used as an alternative method to estimate population parameters.

Suppose $X_1, X_2, \dots, X_n$ be a random sample of size n and $\bar{X} = \sum_{i=1}^n \frac{X_i}{n}$ be the sample mean used to estimate the population mean. Let $\hat{\eta}$ be the estimated parameter vector of η based on each of the n observations and $\hat{\eta}_{-i}$ for $i = 1, 2, \dots, n$ be the estimated η value after the i^{th} observation was removed from the sample. Hence, the pseudo-values are obtained by;

$$\hat{\eta}_i^* = n\hat{\eta} - (n-1)\hat{\eta}_{-i}, \quad i = 1, 2, \dots, n.$$

Thus, the jackknife estimator of η is given by,

$$\hat{\eta}^* = \frac{\sum_{i=1}^n \hat{\eta}_i^*}{n}.$$

Manly (2018) proposed that an approximate $100(1-\gamma)\%$ confidence interval for each η is provided by $\hat{\eta}^* \pm t_{\frac{\gamma}{2}, n-1} \frac{s}{\sqrt{n}}$, which eliminates bias of order n^{-1} . Here s represents the standard deviation of the pseudo-values and $t_{\frac{\gamma}{2}, n-1}$ is the upper $(1 - \frac{\gamma}{2})$ point of t distribution with $n - 1$ degrees of freedom.

B. Sensitivity analysis: Global influence

The initial technique for performing sensitivity analysis involves global influence through case - deletion as outlined in Cook (1977). Case deletion is a widely utilized technique for assessing the effect of removing the i^{th} record from the dataset. In the context of model (6), the case - deletion process is defined as follows:

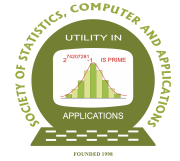
$$Y_l = \mathbf{x}_l^T \boldsymbol{\beta} + \sigma Z_l, \quad l = 1, 2, \dots, n, \quad l \neq i. \quad (9)$$

The term expressed by “ i ” describes the actual expression with the exclusion of the i^{th} record. The log-likelihood function for model (9) is represented as $l_{(i)}(\boldsymbol{\eta})$ and let $\hat{\boldsymbol{\eta}}_i = (\hat{c}_i, \hat{k}_i, \hat{\sigma}_i, \hat{\boldsymbol{\beta}}_i^T)^T$ represent the corresponding estimate of $\boldsymbol{\eta}$. The basic idea is to evaluate, the difference between $\hat{\boldsymbol{\eta}}_i$ and $\hat{\boldsymbol{\eta}}$ in order to figure out the impact of the i^{th} observation on the maximum likelihood estimate $\hat{\boldsymbol{\eta}} = (\hat{c}, \hat{k}, \hat{\sigma}, \hat{\boldsymbol{\beta}}^T)^T$. More consideration should be provided to an observation, if its exclusion substantially affects the estimates. Thus, if $\hat{\boldsymbol{\eta}}_i$ is regarded as an influential observation if it differs considerably from $\hat{\boldsymbol{\eta}}$. The generalized Cook distance (GCD), a frequently employed measure to assess the global influence, is defined as,

$$GD_i(\boldsymbol{\eta}) = (\hat{\boldsymbol{\eta}}_i - \hat{\boldsymbol{\eta}})^T \{-\ddot{\mathbf{L}}(\hat{\boldsymbol{\eta}})\}(\hat{\boldsymbol{\eta}}_i - \hat{\boldsymbol{\eta}}).$$

Another established measure used to quantify the distinction between $\hat{\boldsymbol{\eta}}_i$ and $\hat{\boldsymbol{\eta}}$ is the likelihood displacement (LD) defined as:

$$LD_i(\boldsymbol{\eta}) = 2\{l(\hat{\boldsymbol{\eta}}) - l(\hat{\boldsymbol{\eta}}_i)\}.$$



Reducing Dimensionality and Modeling Structural Dependence in Data Streams: An Approach using Copula Matrices

Fereshteh Arad¹, Ayyub Sheikhi¹ and Farshid Keynia²

¹*Department of Statistics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran*

²*Department of Energy Management and Optimization, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology, Kerman, Iran*

Received: 01 September 2024; Revised: 27 March 2025; Accepted: 31 March 2025

Abstract

Dimension reduction techniques are effective in dealing with high-volume data. Considering streaming data, in this study we introduce a dimension reduction algorithm when non-linear relationships exist between variables. We propose a copula-based feature extraction algorithm to extract streaming most informative components. Using a simulation study we show that our proposed algorithm has a better performance compared to other traditional feature extraction methods. We also use the extracted components to make a prediction analysis. A real-time analysis of online Apple stock data has been provided to illustrate the merits of our approach. Moreover, we carry out feature extraction as well as a prediction analysis in the *COVID-19* and Real-time Bitcoin dataset and show that our method benefits the prediction accuracy.

Key words: Data stream; Copula; Dimension reduction; Feature extraction.

AMS Subject Classifications: 68T05, 68W27.

1. Introduction

Recent advancements in information technology have led to the generation of a large volume of data that requires online analysis and intelligent processing to extract useful information. One key concept that holds significant importance in this field is the data stream, which is an endless collection of countable elements that enter the system continuously and are analyzed by various processes to generate new results (*c.f.*, Margara and Rabl (2019)).

The data stream has created various research challenges, especially yielding big data, so, reducing the dimension of the data has received more attention. Generally, dimension reduction refers to the process of reducing the number of random variables under considera-

tion by obtaining a set of principal variables, and is commonly used in fields such as pattern recognition, data mining, and machine learning. As some instances, we refer to van der Maaten *et al.* (2009) and Sarveshwaran *et al.* (2019), Jia *et al.* (2022), Salem and Hussein (2019).

In the context of the data stream, researchers have used dimension reduction methods to simplify the analysis of complex streaming data sets, improve the performance of data mining algorithms, and reduce storage and computational requirements. Yan *et al.* (2006) have proposed an effective dimensionality reduction for a larger scale and streaming data; Upadhyay *et al.* (2013) have investigated dimension reduction techniques in data stream mining; Xie *et al.* (2020) have introduced an efficient dimension reduction for streaming multi-view data and Zheng *et al.* (2022) have used graph diffusion for a streaming feature selection. In this work we have we propose a feature extraction algorithm as a subcategory of dimension reduction in the data stream. Bahri *et al.* (2020) have studied feature transformation techniques for data streams. Park and Lee (2020) have studied a linear dimension reduction method for streaming data. Moreover, since time series are the special case of the data stream models, dimension reduction techniques have been carried out by the researchers. For further study in this area, we refer to Wang and Megalooikonomou (2008) and Park *et al.* (2010), Krawczak and Szkatuła (2014).

As another challenge of the data stream, the relationship between variables varies over time by increasing the amount of data. The concept of dependence between variables can be explained using copula functions which can address the linear and nonlinear dependencies of the variables (*c.f.*, Sheikhi *et al.* (2022a)). Houari *et al.* (2016) have introduced a new technique for reducing the dimensionality of multi-dimensional data using Copulas. Qu (2012) have refereed to the feature extraction using the copula function. Simard and Remillard (2013) utilized the copula function to model time series model also presented a prediction method for the time series using a copula function.

Also, nowadays, GPU graphics processors are used to perform complex calculations in fields such as machine learning, scientific simulations, and image and video processing. These processing units are very efficient due to the ability to perform a large number of operations simultaneously and help improve graphics quality and processing speed (*c.f.*, Misic *et al.* (2012), Ram (2023)).

In this article, we propose our Stream Copula Feature Extraction (SCFE) algorithm to reduce the dimensionality of data in data streams by extracting the main principal components. Then, we apply the extracted components as new informative inputs for prediction purposes several studies have been done in this context. Sheikhi *et al.* (2022b) have been working on the topic of dimension reduction in neural networks using copula function, Zeng and Wang (2022) have introduced a dimension reduction approach called Neural Copula. Carrillo *et al.* (2021) have presented a machine learning prediction algorithm based on a copula function. Toharudin *et al.* (2019) have used a combination of copula and neural networks for forecasting. Qu (2012) in their article focuses on feature extraction using Archimedes copula. However, those methods were inefficient in dealing with streaming data. The structure of this work is as follows. Section 2 introduces a couple-based feature extraction algorithm and uses the extracted components to carry out a prediction analysis. Section 3 is devoted to providing numerical analysis of the proposed algorithm by simulated and real

datasets, and finally, Section 4 brings some concluding remarks.

2. Stream copula feature extraction

One of the most important methods in dimensionality reduction is feature extraction. Feature extraction leads to dimensionality reduction by combining the main features. From this perspective, constructing a new set of components is usually more compact and has a greater distinguishing property (*c.f.*, Mutlag *et al.* (2020)). The main question here is what to do if the variables are not necessarily related linearly? This question will become an important issue when these non-linear connections vary over time.

Feature extraction methods may consider non-linear relationships but do not consider the dependency between variables (*c.f.*, Fukunaga and Short (1978), Zhu (2010)), so there is a need for a method to consider the dependency between variables. On the other hand, in the data stream, variables may show different behaviors at each moment, and exploring these behaviors will help researchers extract the most informative components. In this study, we attempt to copula function to study the dependency between variables in streaming data and we present a method for feature extraction.

Consider a continuous random vector $(X_1, X_2, \dots, X_n)$. Let F_j be the marginal cumulative distribution function (CDF) of X_j for $j = 1, 2, \dots, n$, and F be the joint CDF. We apply the probability integral transform and define $U_j := F_j(X_j)$, $j = 1, 2, \dots, n$. Since X_j is assumed to be continuous, $U_j \sim U(0, 1)$ follows a uniform distribution. Then the CDF of $(U_1, U_2, \dots, U_n)$ is the copula of $(X_1, X_2, \dots, X_n)$ (*c.f.*, Nelsen (2006)). Denoting this copula function as C , we have

$$C(u_1, u_2, \dots, u_n) = F(F_1(x_1), F_2(x_2), \dots, F_n(x_n))$$

Assume that at the time t , the vectors $\mathbf{x}_1^{(t)}, \mathbf{x}_2^{(t)}, \mathbf{x}_3^{(t)}, \dots, \mathbf{x}_n^{(t)}$, are pairwise copula related random vectors. In this case, the Kendall's association is as $\boldsymbol{\tau}^{(t)} = (\tau_{i,j})^{(t)}$, $i, j = 1, 2, \dots, n$, $t = 1, 2, \dots$, where

$$(\tau_{i,j})^{(t)} = \tau_{X_i^{(t)}, X_j^{(t)}} = 4 \int_0^1 \int_0^1 C_{X_i^{(t)}, X_j^{(t)}}(u_i^{(t)}, u_j^{(t)}) dC_{X_i^{(t)}, X_j^{(t)}}(u_i^{(t)}, u_j^{(t)}) - 1, \quad i, j = 1, 2, \dots, n.$$

It is known that the copula association measures such as Spearman's ρ and Kendall's τ can better capture the non-linear dependencies in contrary to Pearson's correlation (*c.f.*, Zeng and Wang (2022)). As a motivated example for implementing the time, consider z consist of $t = 25$ observation from a normal distribution and z_1, z_2, z_3, z_f and z_e are respectively, linear, quadratic, cubic, cumulative normal, and an exponential transform of z . Figure 4-a depicts the differences between the values of pairwise Pearson's correlation and the values of pairwise Kendall's association of variables z, z_1, z_2, z_3, z_f and z_e for the first five time points; Figure 3-b shows this difference for the first 10 time points, and so on, Figure 3-f compares these two measures for all 25 time points.

As Figure 3 reveals, due to the small number of observations, *i.e.*, $t = 5$, there may be a linear relationship between the variables, but with the passage of time and the increase of observations, *e.g.*, $t = 25$, the intensity of a linear relationship decreases while the Kendall's

Table 1: Pairwise Pearson's and Kendall measures at time 1 (first five observations), time 2 (first ten observations),..., time 5 (all twenty five observations)

time	Pearson's correlation matrix					Kendall's association matrix				
time1	1	0.867	0.870	0.934	0.910	1	0.800	1	1	1
	0.867	1	0.995	0.636	0.993	0.800	1	0.800	0.800	0.800
	0.870	0.995	1	0.638	0.995	1	0.800	1	1	1
	0.934	0.636	0.638	1	0.704	1	0.800	1	1	1
	0.910	0.993	0.995	0.704	1	1	0.800	1	1	1
time 2	1	0.731	0.819	0.950	0.864	1	0.280	1	1	1
	0.731	1	0.968	0.498	0.968	0.280	1	0.280	0.280	0.280
	0.819	0.968	1	0.601	0.993	1	0.280	1	1	1
	0.950	0.498	0.601	1	0.668	1	0.280	1	1	1
	0.864	0.968	0.993	0.668	1	1	0.280	1	1	1
time 3	1	0.523	0.782	0.959	0.807	1	0.270	1	1	1
	0.523	1	0.871	0.305	0.907	0.270	1	0.270	0.270	0.270
	0.782	0.871	1	0.577	0.987	1	0.270	1	1	1
	0.959	0.305	0.577	1	0.619	1	0.270	1	1	1
	0.807	0.907	0.987	0.619	1	1	0.270	1	1	1
time 4	1	0.519	0.770	0.961	0.805	1	0.400	1	1	1
	0.519	1	0.865	0.310	0.903	0.400	1	0.400	0.400	0.400
	0.770	0.865	1	0.566	0.984	1	0.400	1	1	1
	0.961	0.310	0.566	1	0.623	1	0.400	1	1	1
	0.805	0.903	0.984	0.623	1	1	0.400	1	1	1
time 5	1	0.400	0.752	0.967	0.783	1	0.013	1	1	1
	0.400	1	0.810	0.206	0.859	0.013	1	0.013	0.013	0.013
	0.752	0.810	1	0.562	0.982	1	0.013	1	1	1
	0.967	0.206	0.562	1	0.704	1	0.013	1	1	1
	0.783	0.859	0.982	0.613	1	1	0.013	1	1	1

association measure remains stable. Moreover, see Table 1 for the comparison of these two pairwise measures at five time points.

The main aim of this section is carrying out a competitive feature extraction by extracting more informative components of features when the features do not have necessarily linear relationships. In this regard, inspired by (*c.f.*, Witten *et al.* (2009)), we obtain the ordered copula-based principal components by solving the following equation.

$$(\boldsymbol{\tau}^{(t)} - \hat{\lambda}^{(t)} \mathbf{I}_n) \hat{\mathbf{g}}^{(t)} = \mathbf{0}, \quad t = 1, 2, \dots, T, \quad (1)$$

where $\mathbf{I}_n$ is an identity matrix of dimension n . Denote $\hat{\lambda}_1^{(t)}, \hat{\lambda}_2^{(t)}, \dots, \hat{\lambda}_n^{(t)}$ are descending ordered eigenvalues of the matrix $\boldsymbol{\tau}^{(t)}$ at time t and their corresponding eigenvectors are $\hat{\mathbf{g}}_1^{(t)}, \hat{\mathbf{g}}_2^{(t)}, \dots, \hat{\mathbf{g}}_n^{(t)}$.

Let define the matrix of all observations at time t as $\mathbf{X}^{(t)} = (\mathbf{x}_1^{(t)}, \mathbf{x}_2^{(t)}, \mathbf{x}_3^{(t)}, \dots, \mathbf{x}_n^{(t)})$ and matrix of all eigenvectors as $\hat{\mathbf{G}}^{(t)} = (\hat{\mathbf{g}}_1^{(t)}, \hat{\mathbf{g}}_2^{(t)}, \dots, \hat{\mathbf{g}}_n^{(t)})$ then by multiplying these two matrices we obtain the new set of observation as $\mathbf{Q}^{(t)} = \mathbf{X}^{(t)} \hat{\mathbf{G}}^{(t)}$. In the sequel, regarding to extracting more informative principal component, we use the matrix $\mathbf{Q}^{(t)}$. The first column

of this matrix includes the a larger portion of the total variance, the second column stands in the second rank, and so on. If we also want to calculate the cumulative explained variance of these columns, we can evaluate the explained variance of the l -th columns as

$$P_l^{(t)} = \frac{\hat{\lambda}_l^{(t)}}{\sum_{j=1}^n \hat{\lambda}_j^{(t)}}, \quad l = 1, 2, \dots, n. \quad (2)$$

Example 1: Suppose there are three variables at three times $X_1^{(t)}$, $X_2^{(t)}$ and $X_3^{(t)}$, $t = 1, 2, 3$ in which the relationship between these variables vary over time. Suppose at time 1, variables $X_1^{(1)}$ and $X_2^{(1)}$ are related through the Clayton copula (*c.f.*, Cuvelier and Noirhomme-Fraiture (2005)) with $\theta^1 = 0.5$, and variables $X_1^{(1)}$ and $X_3^{(1)}$ are related through the Gumbel copula (*c.f.*, Wang *et al.* (2010)) with $\eta^1 = 0.8$, and variables $X_2^{(1)}$ and $X_3^{(1)}$ are related through the Gaussian copula with $\rho^1 = 0.3$. Using the equation (1), we obtain $\boldsymbol{\tau}^{(1)}$ as the first row of Table 2 and hence using 2 the cumulative explained variance as $P_1^{(1)} = 0.41$, $P_1^{(1)} + P_2^{(1)} = 0.80$ and $\sum_{l=1}^3 P_l^{(1)} = 1$. Similarly, see the second and the third row of this table for different assumptions and results at times 2 and 3. As Table 2 reveals, as time moves forward, the association measure shows stronger relationship. For example, the sum of the explained variances of the first two extracted components at time 1 is 0.81, while for time 2 and 3 will be 0.91 and 0.93 respectively.

Table 2: Pairwise Pearson's and Kendall measures at time 1 (first five observations), time 2 (first ten observations),..., time 5 (all twenty observations)

Time	Assumptions	$\boldsymbol{\tau}^{(t)}$	$\sum_{l=1}^k P_l^{(t)}$
1	$X_1^{(1)}, X_2^{(1)} \sim Cl(\theta^{(1)} = 0.5)$	$\boldsymbol{\tau}^{(1)} = \begin{bmatrix} 1 & 0.20 & -0.25 \\ 0.20 & 1 & 0.19 \\ -0.25 & 0.19 & 1 \end{bmatrix}$	$P_1^{(1)} = 0.41$
	$X_1^{(1)}, X_3^{(1)} \sim G(\eta^{(1)} = 0.8)$		$P_1^{(1)} + P_2^{(1)} = 0.80$
	$X_2^{(1)}, X_3^{(1)} \sim Ga(\rho^{(1)} = 0.3)$		$\sum_{l=1}^3 P_l^{(1)} = 1$
2	$X_1^{(2)}, X_2^{(2)} \sim Cl(\theta^{(2)} = 0.6)$	$\boldsymbol{\tau}^{(2)} = \begin{bmatrix} 1 & 0.23 & -0.42 \\ 0.23 & 1 & 0.49 \\ -0.42 & 0.49 & 1 \end{bmatrix}$	$P_1^{(2)} = 0.51$
	$X_1^{(2)}, X_3^{(2)} \sim G(\eta^{(2)} = 0.7)$		$P_1^{(2)} + P_2^{(2)} = 0.91$
	$X_2^{(2)}, X_3^{(2)} \sim C^t(\zeta_1^{(2)} = 0.7)$		$\sum_{l=1}^3 P_l^{(2)} = 1$
3	$X_3^{(3)}, X_2^{(3)} \sim Cl(\theta^{(3)} = 0.6)$	$\boldsymbol{\tau}^{(3)} = \begin{bmatrix} 1 & 0.23 & 0.59 \\ 0.23 & 1 & 0.71 \\ 0.59 & 0.71 & 1 \end{bmatrix}$	$P_1^{(3)} = 0.68$
	$X_1^{(3)}, X_3^{(3)} \sim Ga(\rho_1^{(3)} = 0.8)$		$P_1^{(3)} + P_2^{(3)} = 0.93$
	$X_2^{(3)}, X_3^{(3)} \sim Ga(\rho_2^{(3)} = 0.9)$		$\sum_{l=1}^3 P_l^{(3)} = 1$

The following theorem states that we reach faster to the higher levels of percentage of the variance's explanation by passage of the time.

Theorem 1: Let $\lambda_1^{(t)} > \lambda_2^{(t)} > \dots > \lambda_n^{(t)}$ be the eigenvalues of Kendall's association matrix of time $t = 1, 2, 3, \dots$ and define $f(c)^{(t)} = 1 - \sum_{j=1}^k P_j^{(t+c)}$ and $f^{(t)} = 1 - \sum_{j=1}^k P_j^{(t)}$ then $f(c)^{(t)} = o(f^{(t)})$.

Proof: According to the definition of 2, $\exists k_0$ s.t $k_0 = \underset{u \in N}{argmax} \sum_{j=1}^u P_j^{(t)}$ and $\forall k > k_0$ we have $0 < \sum_{j=1}^k P_j^{(t)} < \sum_{j=1}^k P_j^{(t+c)} \leq 1$ and $\lim_{t \rightarrow \infty} \sum_{j=1}^k P_j^{(t+c)} = 1$, therefore, it can be said that $0 < \lim_{t \rightarrow \infty} 1 - \sum_{j=1}^k P_j^{(t)} < 1$. Also, as $\lim_{t \rightarrow \infty} 1 - \sum_{j=1}^k P_j^{(t)} \neq 0$ the limit ratio can be considered equal to the ratios and $\lim_{t \rightarrow \infty} 1 - \sum_{j=1}^k P_j^{(t+c)} = 0$ so we have

$$\lim_{t \rightarrow \infty} \frac{f(c)^{(t)}}{f^{(t)}} = \lim_{t \rightarrow \infty} \frac{1 - \sum_{j=1}^k P_j^{(t+c)}}{1 - \sum_{j=1}^k P_j^{(t)}} = 0$$

□

One result of the above theorem is the following corollary.

Corollary 1: When faced with a data stream as time moves forward, $\lim_{t \rightarrow \infty} \sum_{j=1}^k P_j^{(t+c)}$ for $k_0 < k \leq n$ In such a way $k_0 = \underset{u \in N}{argmax} \sum_{j=1}^u P_j^{(t)}$ and $c = 1, 2, 3, \dots$ reaches one earlier than previous times $\lim_{t \rightarrow \infty} \sum_{j=1}^k P_j^{(t)}$.

Proof: Considering theorem 1 and the relationship $f(c)^{(t)} = o(f^{(t)})$, since $\lim_{t \rightarrow \infty} 1 - \sum_{j=1}^k P_j^{(t+c)}$ approaches zero faster than $\lim_{t \rightarrow \infty} 1 - \sum_{j=1}^k P_j^{(t)}$, it can be said that $\lim_{t \rightarrow \infty} \sum_{j=1}^k P_j^{(t+c)}$ converges faster than $\lim_{t \rightarrow \infty} \sum_{j=1}^k P_j^{(t)}$ to one. □

The above corollary indicates that over time, as the behaviors of variables become clear, we can extract features more accurately using Kendall's correlation matrix between variables. This approach can explain a higher percentage of variance with fewer components. After extracting the relevant components based on the explained variance percentage, they can be used instead of the original features in the inputs of a prediction model such as neural networks, regression, *etc.*

Regarding to a prediction task, we can use the primary components obtained at each time as inputs of a neural network and update the weight matrix of the artificial neural network to minimize the prediction error below a certain threshold δ . This process involves obtaining the first hidden layer for each time t as follows

$$\mathbf{h}_1^{(t)} = f_w^{(t)}(\mathbf{w}_{hu}^{(t)} \mathbf{u}_d^{(t)} + \mathbf{b}_1^{(t)}) \quad t = 1, 2, \dots, \quad (3)$$

where $\mathbf{w}_{hu}^{(t)}$ stands for the weights between the input values and the nodes of the first hidden layer at time t , $f_w^{(t)}$ is an activation function and $\mathbf{b}_1^{(t)}$ is the bias term at time t . Also, in the next layers

$$\mathbf{h}_k^{(t)} = f_w^{(t)}(\mathbf{w}_{k,k-1}^{(t)}\mathbf{h}_{k-1}^{(t)} + \mathbf{b}_k^{(t)}), \quad t = 1, 2, \dots, \quad k = 2, 3, \dots, \quad (4)$$

in which, $f_w^{(t)}$ represents an activation function at time t , $\mathbf{w}_{k,k-1}^{(t)}$, $k = 2, 3, \dots$, are the connecting weights between the k -th and $(k-1)$ -th layers at time t , and $\mathbf{b}_k^{(t)}$ is the bias term at time t . The subsequent equation in the neural network utilizes the final layer and certain weights to generate the output $\hat{\mathbf{y}}^{(t)}$ as

$$\hat{\mathbf{y}}^{(t)} = g_w^{(t)}(\mathbf{w}_{hy}^{(t)}\mathbf{h}_d^{(t)} + \mathbf{b}_y^{(t)}) \quad t = 1, 2, \dots, \quad (5)$$

where $g_w^{(t)}$ can be a nonlinear function and $\mathbf{w}_{hy}^{(t)}$ is the weight associated with the hidden layers and outputs at time t . Additionally, $\mathbf{b}_y^{(t)}$ represents the bias term at time t .

Algorithm 1: Stream Copula Feature Extraction Neural Network (*SCFE-NN*) algorithms for all times.

Data: streaming data $\mathbf{X}^{(t)}$ and $\mathbf{y}^{(t)}$, minimum percentage of explained variance α , and threshold value $\delta^{(t)}$

Result: Updating the weights of the neural network for the extracted components

- 1 Initialization; Obtain the copula matrix $\boldsymbol{\tau}^{(t)}$ and compute eigenvectors $\hat{\mathbf{g}}_1^{(t)}, \dots, \hat{\mathbf{g}}_n^{(t)}$
 - 2 **SCFE:**
 - 3 set $l = 1$ and obtain $\mathbf{Q}^{(t)} = \mathbf{X}^{(t)}\hat{\mathbf{g}}_1^{(t)}$ and it's $P_l^{(t)}$ using **2**
 - 4 **while** $\sum_{k=1}^l P_k^{(t)} < \alpha$ **do**
 - 5 $l = l + 1$
 - 6 construct $\mathbf{Q}^{(t)} = \mathbf{X}^{(t)}(\hat{\mathbf{g}}_1^{(t)}, \dots, \hat{\mathbf{g}}_l^{(t)})$ and obtain it's $P_l^{(t)}$
 - 7 **SCFE-NN:**
 - 8 Set the j -th input of *NN* as $\mathbf{u}_j^{(t)} = \mathbf{X}^{(t)}\hat{\mathbf{g}}_j^{(t)}$ and and construct $\mathbf{U}^{(t)} = (\mathbf{u}_1^{(t)}, \mathbf{u}_2^{(t)}, \mathbf{u}_k^{(t)})$
 - 9 **while** $\|\hat{\mathbf{y}}^{(t)} - \mathbf{y}^{(t)}\|_2 < \delta^{(t)}$ **do**
 - 10 Update the *NN*'s weights and biases using **3-5**.
-

Based on the results of theorem 1, by passage of time, the values of $\sum_{k=1}^l P_k^{(t)}$ will be increased, so we may perform the algorithm 1 for initial values of time and increase t until $\sum_{k=1}^l P_k^{(t)}$ meets the threshold of minimum explained variance (α).

3. Numerical study

In this Section we assess the performance of our approach using a simulation and we apply our contribution to analyze two real world data sets. Two main approaches have been used to optimize the model. First, to optimize the parameters of the copula function, the `Bicopselect` command is used in the copula library in R software, this command uses the *MLE* method, this method finds the best fit for the data and allows us to accurately model the dependence between the variables. In the next step, a neural network is used for more complex modeling, and a piece of code written in R software is used to optimize the network parameters such as units, stepmax, number-layers using mean absolute error (*MAE*) criterion. To evaluate and ensure the generalizability of the models, the 5-fold cross-validation method has been used. The codes of the article have been implemented in Google colab's gpu¹ space, the output of the results is given below.

3.1. Simulation study

In order to perform a simulation analysis, we consider Gaussian, t and Gumbel copulas for generating data set. Assume that $X_1, X_2, \dots, X_5 \sim G_5(\mathbf{R}_1)$ and $X_6, \dots, X_{10} \sim t_5(\mathbf{R}_2, \nu)$, where G_5 and t_5 are respectively five dimensional Gaussian and t copulas where

$$\mathbf{R}_1 = \begin{bmatrix} 1 & 0.5 & 0.6 & 0.65 & 0.8 \\ 0.4 & 1 & 0.5 & 0.4 & 0.5 \\ 0.6 & 0.5 & 1 & 0.5 & 0.6 \\ 0.65 & 0.4 & 0.5 & 1 & 0.6 \\ 0.8 & 0.5 & 0.6 & 0.6 & 1 \end{bmatrix}, \mathbf{R}_2 = \begin{bmatrix} 1 & 0.3 & 0.4 & 0.65 & 0.6 \\ 0.3 & 1 & 0.3 & 0.5 & 0.4 \\ 0.4 & 0.3 & 1 & 0.4 & 0.5 \\ 0.65 & 0.5 & 0.4 & 1 & 0.6 \\ 0.6 & 0.4 & 0.5 & 0.6 & 1 \end{bmatrix}$$

are respectively their correlation matrices. Also, we assume that $X_{11}, X_{12}, \dots, X_{15}$ follow Gumbel's copula with $\theta = 15$. We generate 1000 random samples from these 15 variables considering their marginals come from a normal distribution.

In the interest of creating the target variable, y , assuming that ε is a white noise, we employ the equation $Y = \sum_{i=1}^{15} X_i + e^{\sum_{i=1}^5 X_i} + \varepsilon$ which provides linear and nonlinear relations between inputs and output variables. Finally, suppose we have 5 time point, so, we divide the observation into 5 overlapping segments., *i.e.*, Segment 1: the first 200 cases, Segment 2: the first 400 cases,..., Segment 5: all 1000 cases.

By estimating the pairwise connection between the variables in these five segments ($t = 1, 2, \dots, 5$), and calculating the Kendall copula correlation matrix ($\hat{\mathbf{T}}^{(t)}$) and the Pearson correlation matrix ($\hat{\mathbf{R}}^{(t)}$), we can reduce the dimension of the variables via implementing algorithm 1. The results are summarized in Figure 4. As it can be seen from Figure 4, the Kendall's association matrix easily resolves non linear relationship between variables over time.

To assess our *SCFE-NN* approach, we compare its results with the results of the other traditional feature extraction methods. The cumulative percentage of explained variance by our approach as well as other approaches such as Linear *PCA*, polynomial Kernel *PCA*, Gaussian Kernel *PCA*, Laplacian Kernel *PCA*, and *ANOVA* Kernels *PCA* are provided in

¹<https://colab.research.google.com/drive/1xHgYWQxfIkNtNx6YU4CXEQSzaDoriYr8I?usp=sharing>

Table 3: The cumulative percentage of explained variance by six *PCA* methods at different times

time 1	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8	comp 9	comp 10	comp 11	comp 12	comp 13	comp 14	comp 15
polynomial Kernel <i>PCA</i>	0.65	0.94	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
Gaussian Kernel <i>PCA</i>	0.24	0.41	0.52	0.60	0.67	0.72	0.76	0.81	0.84	0.87	0.90	0.93	0.95	0.97	1
Laplacian Kernel <i>PCA</i>	0.28	0.43	0.52	0.60	0.66	0.71	0.76	0.80	0.84	0.88	0.90	0.93	0.95	0.98	1
ANOVA Kernels <i>PCA</i>	0.27	0.44	0.54	0.61	0.68	0.73	0.77	0.81	0.84	0.87	0.90	0.93	0.95	0.97	1
Copula <i>PCA</i>	0.50	0.64	0.75	0.86	0.91	0.93	0.96	0.96	0.97	0.98	0.98	0.99	0.99	0.99	1
Linear <i>PCA</i>	0.60	0.73	0.83	0.88	0.92	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1
time 2	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8	comp 9	comp 10	comp 11	comp 12	comp 13	comp 14	comp 15
polynomial Kernel <i>PCA</i>	0.45	0.72	0.87	0.94	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1
Gaussian Kernel <i>PCA</i>	0.26	0.44	0.55	0.63	0.70	0.75	0.80	0.83	0.87	0.90	0.93	0.95	0.96	0.98	1
Laplacian Kernel <i>PCA</i>	0.29	0.45	0.55	0.63	0.69	0.75	0.79	0.82	0.86	0.88	0.91	0.94	0.96	0.98	1
ANOVA Kernels <i>PCA</i>	0.29	0.48	0.58	0.66	0.72	0.77	0.81	0.84	0.87	0.90	0.93	0.95	0.96	0.98	1
Copula <i>PCA</i>	0.52	0.65	0.76	0.85	0.94	0.95	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	1
Linear <i>PCA</i>	0.57	0.70	0.79	0.86	0.93	0.95	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1
time 3	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8	comp 9	comp 10	comp 11	comp 12	comp 13	comp 14	comp 15
polynomial Kernel <i>PCA</i>	0.40	0.70	0.84	0.93	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1
Gaussian Kernel <i>PCA</i>	0.26	0.45	0.55	0.63	0.70	0.75	0.80	0.83	0.87	0.90	0.93	0.95	0.96	0.98	1
Laplacian Kernel <i>PCA</i>	0.30	0.47	0.56	0.64	0.71	0.76	0.80	0.83	0.87	0.90	0.92	0.94	0.96	0.98	1
ANOVA Kernels <i>PCA</i>	0.29	0.48	0.58	0.66	0.73	0.77	0.81	0.85	0.88	0.91	0.93	0.95	0.97	0.98	1
Copula <i>PCA</i>	0.53	0.70	0.84	0.93	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1
Linear <i>PCA</i>	0.57	0.70	0.79	0.86	0.91	0.95	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1
time 4	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8	comp 9	comp 10	comp 11	comp 12	comp 13	comp 14	comp 15
polynomial Kernel <i>PCA</i>	0.49	0.77	0.90	0.93	0.95	0.96	0.96	0.97	0.97	0.98	0.99	0.99	0.99	0.99	1
Gaussian Kernel <i>PCA</i>	0.25	0.45	0.56	0.64	0.71	0.76	0.80	0.84	0.87	0.90	0.93	0.95	0.96	0.98	1
Laplacian Kernel <i>PCA</i>	0.31	0.48	0.57	0.65	0.71	0.77	0.81	0.84	0.87	0.90	0.92	0.94	0.96	0.98	1
ANOVA Kernels <i>PCA</i>	0.29	0.49	0.59	0.67	0.73	0.78	0.82	0.85	0.88	0.91	0.93	0.95	0.97	0.98	1
Copula <i>PCA</i>	0.63	0.83	0.94	0.95	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
Linear <i>PCA</i>	0.63	0.82	0.89	0.94	0.95	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1
time 5	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8	comp 9	comp 10	comp 11	comp 12	comp 13	comp 14	comp 15
polynomial Kernel <i>PCA</i>	0.72	0.86	0.91	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1
Gaussian Kernel <i>PCA</i>	0.32	0.45	0.56	0.64	0.71	0.75	0.80	0.83	0.87	0.90	0.93	0.95	0.96	0.98	1
Laplacian Kernel <i>PCA</i>	0.33	0.48	0.58	0.65	0.71	0.77	0.81	0.84	0.87	0.90	0.93	0.95	0.96	0.98	1
ANOVA Kernels <i>PCA</i>	0.29	0.48	0.59	0.67	0.73	0.78	0.81	0.85	0.88	0.91	0.94	0.95	0.97	0.98	1
Copula <i>PCA</i>	0.76	0.87	0.95	0.96	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
Linear <i>PCA</i>	0.70	0.85	0.90	0.93	0.96	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1

Table 3 for all 5 time points. As you can see, at the first time, the polynomial Kernel *PCA* reduced the dimension of the variables better than the other methods, so that by considering the first 4 components, it can explain (97%) of the total variance, but with the passage of time and the behavior of the variables being determined, other methods perform better. As you can see in time 5, the Copula *PCA* method explains (95%) of the total variance by considering the first 5 components, and it performs much better than the other two methods. See also, Figure 1, as a visualization representation.

In summary, it can be said that at the initial times when the behavior of the data is not known, the polynomial Kernel *PCA* and Linear *PCA* dimension reduction methods work well, but as time passes and the behavior of the variables becomes clear, the Copula *PCA* method superforms the dimension reduction in contrast of other methods.

Next, to check the stated algorithm on the generated data set, suppose we want the components to explain (95%) of the total variance ($\alpha_t = 0.95$, $t = 1, 2, 3, 4, 5$), in this case, the results of dimension reduction on this data set are shown in Table 4.

As seen from Table 4 the best methods are Kernel *PCA*, Copula *PCA*, and Linear *PCA*. More precisely, for the first time, using the polynomial Kernel *PCA* method, the first 3 components account for 95% of the total variance while the number of components for this percentage are 7 and 6 respectively for Copula *PCA* and Linear *PCA*. However, with the passage of time and the increase of data, for example at the 5th time, in the polynomial Kernel *PCA* method, 5 components explain, 95% of the total variance, but the Copula *PCA* method, explains 95% of the total variance by 3 components and, and the Linear *PCA* needs 5 components to explain 95% of the total variance.

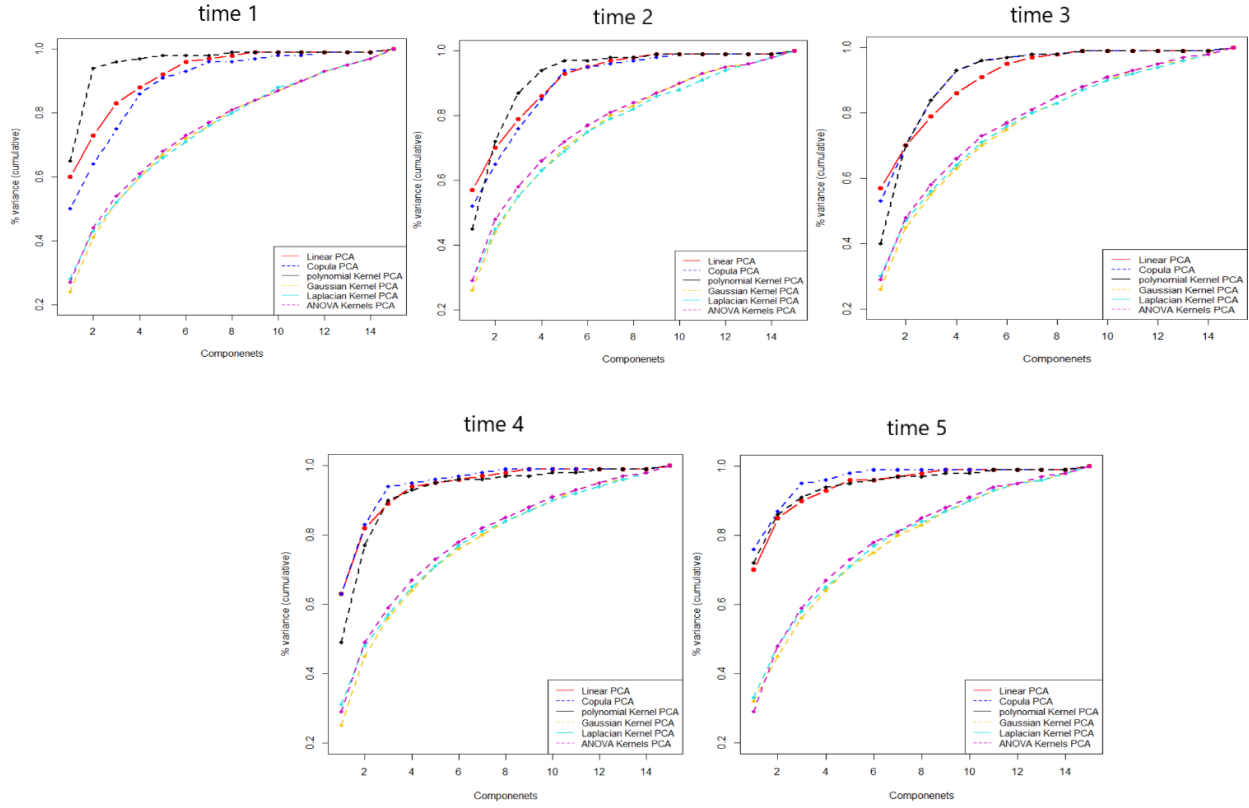


Figure 1: Percents of explained variance using six *PCA* methods at different times

Table 4: The number of components that explain 95% of the total variance in each method and at different times

time	1	2	3	4	5
polynomial Kernel <i>PCA</i>	3	5	5	5	5
Gaussian Kernel <i>PCA</i>	13	12	12	12	12
Laplacian Kernel <i>PCA</i>	13	13	13	13	12
<i>ANOVA</i> Kernels <i>PCA</i>	13	12	12	12	12
Copula <i>PCA</i>	7	6	5	4	3
Linear <i>PCA</i>	6	6	6	5	5

To evaluate model performance across different time points, we can use two key metrics: Reconstruction-Mean-Squared-Error (*RMSE*) (c.f. Milosevic *et al.* (2021)), and *Mean-Corr* (mean of correlations). *RMSE* quantifies how well a model reconstructs its input data, with lower values indicating better performance. It is calculated using the formula:

$$RMSE = \sqrt{\frac{1}{n} \sum_{l=1}^n (x_l - \hat{x}_l)^2}$$

where n is the number of values, x_l is the original value, and $\hat{x}_l$ is the reconstructed value. *Mean-Corr*, on the other hand, assesses the relationship between original and reconstructed data, where higher values (closer to 1) show that the model preserves the patterns in the data well. It's calculated as:

$$Mean-Corr = \frac{1}{n} \sum_{l=1}^n Cor(x_l, \hat{x}_l)$$

where $Cor(x_l, \hat{x}_l)$ is the Pearson correlation between original and reconstructed data for each sample. By calculating these metrics for different models at various time intervals and organizing them into a table, you can track changes in performance, identify trends, and determine which model best maintains its effectiveness across time.

According to the output of Table 5, it can be said that Copula *PCA* has more *RMSE* and less *Mean-Corr* in the early times, but as time increases and the volume of data increases, it has a lower *RMSE* value than other models and *Mean-Corr* increases.

To examine the algorithm *SCFE-NN* in a prediction, we input variables of the neural network by the principal components that explain 95% of the total variance at each time, using the components obtained in the previous step using the Linear *PCA*, Copula *PCA*, polynomial Kernel *PCA* method. The 5-Fold cross-validation method was used for unbiased evaluation. In each fold, the parameters of the neural network were optimized separately and the obtained results are given below.

The results of the comparisons conducted are presented in the table below, As mentioned, in this research.

As shown in Table 7, the *SCFE-NN* algorithm consistently exhibits less overfitting compared to other methods. In fact, relying on the *SCFE-NN* algorithm is advisable as it provides better predictions based on the number of components it selects at any given time. It may initially select more components than other methods, but it delivers more accurate predictions. As time progresses and the function behaviors change, the *SCFE-NN* algorithm with fewer components offers better predictions.

3.2. Real data

We use our approach to carry out a feature extraction and make a prediction in online Apple stock dataset as well as *COVID-19* dataset and the Real-time Bitcoin dataset.

Table 5: Output *RMSE* and *Mean-Corr* values for each method and at any time

time 1	<i>RMSE</i>	<i>Mean-Corr</i>
polynomial Kernel <i>PCA</i>	1.09	0.08
Gaussian Kernel <i>PCA</i>	1.09	0.06
Laplacian Kernel <i>PCA</i>	1.09	0.04
<i>ANOVA</i> Kernels <i>PCA</i>	1.09	0.03
Copula <i>PCA</i>	1.11	0.13
Linear <i>PCA</i>	1.24	0.16
time 2	<i>RMSE</i>	<i>Mean-Corr</i>
polynomial Kernel <i>PCA</i>	1.07	0.10
Gaussian Kernel <i>PCA</i>	1.07	0.08
Laplacian Kernel <i>PCA</i>	1.07	0.07
<i>ANOVA</i> Kernels <i>PCA</i>	1.07	0.05
Copula <i>PCA</i>	1.09	0.15
Linear <i>PCA</i>	1.18	0.14
time 3	<i>RMSE</i>	<i>Mean-Corr</i>
polynomial Kernel <i>PCA</i>	1.05	0.12
Gaussian Kernel <i>PCA</i>	1.05	0.09
Laplacian Kernel <i>PCA</i>	1.05	0.09
<i>ANOVA</i> Kernels <i>PCA</i>	1.05	0.07
Copula <i>PCA</i>	1.04	0.17
Linear <i>PCA</i>	1.15	0.11
time 4	<i>RMSE</i>	<i>Mean-Corr</i>
polynomial Kernel <i>PCA</i>	1.04	0.14
Gaussian Kernel <i>PCA</i>	1.04	0.11
Laplacian Kernel <i>PCA</i>	1.04	0.11
<i>ANOVA</i> Kernels <i>PCA</i>	1.04	0.11
Copula <i>PCA</i>	1.03	0.18
Linear <i>PCA</i>	1.13	0.11
time 5	<i>RMSE</i>	<i>Mean-Corr</i>
polynomial Kernel <i>PCA</i>	1.03	0.15
Gaussian Kernel <i>PCA</i>	1.03	0.12
Laplacian Kernel <i>PCA</i>	1.03	0.12
<i>ANOVA</i> Kernels <i>PCA</i>	1.03	0.12
Copula <i>PCA</i>	1.00	0.19
Linear <i>PCA</i>	1.10	0.12

Table 6: Values of optimized neural network parameters in each fold of each method

time	Method	Fold	units	stepmax	number-layers	MAE
1	linear- <i>PCA-NN</i>	1	6	$1e+06$	3	0.003
		2	4	$1e+05$	1	0.003
		3	4	$1e+06$	2	0.003
		4	4	$1e+06$	2	0.002
		5	6	$1e+06$	1	0.003
	polynomial Kernel <i>PCA-NN</i>	1	4	$1e+07$	2	0.232
		2	4	$1e+05$	1	0.259
		3	4	$1e+07$	1	0.263
		4	6	$1e+07$	1	0.295
		5	4	$1e+07$	2	0.232
	<i>SCFE-NN</i>	1	4	$1e+05$	2	0.003
		2	6	$1e+05$	1	0.002
		3	4	$1e+05$	1	0.003
		4	4	$1e+06$	2	0.003
		5	4	$1e+07$	1	0.003
2	linear- <i>PCA-NN</i>	1	8	$1e+06$	1	0.001
		2	6	$1e+05$	3	0.001
		3	4	$1e+05$	1	0.002
		4	8	$1e+06$	3	0.002
		5	4	$1e+06$	2	0.002
	polynomial Kernel <i>PCA-NN</i>	1	4	$1e+05$	3	0.66
		2	4	$1e+05$	1	0.57
		3	6	$1e+06$	3	0.47
		4	6	$1e+05$	3	0.55
		5	4	$1e+07$	2	0.59
	<i>SCFE-NN</i>	1	4	$1e+06$	2	0.001
		2	4	$1e+05$	2	0.001
		3	4	$1e+06$	2	0.002
		4	8	$1e+05$	1	0.001
		5	6	$1e+05$	1	0.001
3	linear- <i>PCA-NN</i>	1	6	$1e+05$	2	0.002
		2	6	$1e+06$	2	0.002
		3	8	$1e+07$	1	0.001
		4	6	$1e+05$	2	0.001
		5	6	$1e+05$	1	0.001
	polynomial Kernel <i>PCA-NN</i>	1	4	$1e+07$	3	0.005
		2	4	$1e+06$	2	0.001
		3	6	$1e+05$	3	0.003
		4	4	$1e+07$	3	0.013
		5	6	$1e+07$	3	0.0164
	<i>SCFE-NN</i>	1	8	$1e+06$	3	0.001
		2	6	$1e+07$	2	0.001
		3	6	$1e+06$	2	0.001
		4	6	$1e+06$	1	0.002
		5	4	$1e+05$	2	0.001

time	Method	Fold	units	stepmax	number-layers	MAE
4	linear- <i>PCA-NN</i>	1	6	$1e + 05$	3	0.003
		2	8	$1e + 06$	2	0.001
		3	6	$1e + 05$	2	0.002
		4	8	$1e + 05$	1	0.001
		5	4	$1e + 05$	2	0.001
	polynomial Kernel <i>PCA-NN</i>	1	8	$1e + 05$	2	0.015
		2	4	$1e + 07$	3	0.017
		3	6	$1e + 07$	2	0.001
		4	4	$1e + 06$	1	0.001
		5	6	$1e + 05$	2	0.003
	<i>SCFE-NN</i>	1	8	$1e + 06$	3	0.001
		2	8	$1e + 05$	2	0.002
		3	8	$1e + 05$	2	0.003
		4	8	$1e + 06$	2	0.003
		5	6	$1e + 07$	2	0.002
5	linear- <i>PCA-NN</i>	1	6	$1e + 05$	2	0.004
		2	4	$1e + 06$	2	0.003
		3	8	$1e + 05$	1	0.003
		4	4	$1e + 05$	2	0.002
		5	6	$1e + 05$	1	0.002
	polynomial Kernel <i>PCA-NN</i>	1	8	$1e + 06$	3	0.015
		2	8	$1e + 05$	2	0.016
		3	8	$1e + 07$	3	0.017
		4	8	$1e + 07$	3	0.016
		5	8	$1e + 06$	3	0.016
	<i>SCFE-NN</i>	1	6	$1e + 06$	3	0.003
		2	6	$1e + 07$	2	0.001
		3	4	$1e + 06$	2	0.003
		4	4	$1e + 06$	1	0.004
		5	6	$1e + 05$	2	0.004

3.2.1. Apple stock dataset

A real-time streaming dataset for implementing our *SCFE-NN* algorithm is the stock data of Apple Inc². This dataset is updated daily. We consider 6 variables: “AAPL.Open”, “AAPL.High”, “AAPL.Low”, “AAPL.Close”, “AAPL.Volume”, “AAPL.Adjusted” from this dataset from the date of 1-4-2008 to 20-2-2024, and the data is segmented into 4 timeframes. Assuming AAPL.Open, AAPL.High, AAPL.Low, AAPL.Volume and AAPL.Adjusted as predictors and as the response variable, we implemented algorithm 1. We first extract the most informative components from the predictor variables. Table 8 expresses the percentage of explained variance using our algorithm as well as other alternative methods.

As shown in this table, Copula *PCA* gradually explains a higher percentage of variance with fewer components over time.

²<https://finance.yahoo.com/quote/AAPL/history>

Table 7: Comparison results of algorithm model *SCFE-NN* with other models at different times

time	Estimation	linear- <i>PCA-NN</i>	polynomial Kernel <i>PCA-NN</i>	<i>SCFE-NN</i>	<i>PCR</i>
1	mean of <i>RMSE</i> train	0.011	0.300	0.008	0.690
	SD of <i>RMSE</i> train	0.001	0.016	0.001	0.070
	mean of <i>RMSE</i> test	0.210	0.602	0.136	0.600
	SD of <i>RMSE</i> test	0.197	0.500	0.164	0.150
2	mean of <i>RMSE</i> train	0.006	0.180	0.005	0.990
	SD of <i>RMSE</i> train	0.001	0.027	0.000	0.050
	mean of <i>RMSE</i> test	0.073	0.662	0.057	0.840
	SD of <i>RMSE</i> test	0.120	0.429	0.066	0.180
3	mean of <i>RMSE</i> train	0.008	0.025	0.001	2.240
	SD of <i>RMSE</i> train	0.004	0.022	0.043	0.600
	mean of <i>RMSE</i> test	0.092	0.159	0.003	1.590
	SD of <i>RMSE</i> test	0.045	0.378	0.044	0.710
4	mean of <i>RMSE</i> train	0.007	0.011	0.001	2.020
	SD of <i>RMSE</i> train	0.009	0.000	0.003	0.620
	mean of <i>RMSE</i> test	0.102	0.232	0.050	1.810
	SD of <i>RMSE</i> test	0.097	0.078	0.042	0.950
5	mean of <i>RMSE</i> train	0.008	0.027	0.008	0.70
	SD of <i>RMSE</i> train	0.003	0.002	0.001	0.280
	mean of <i>RMSE</i> test	0.092	0.201	0.078	0.340
	SD of <i>RMSE</i> test	0.064	0.039	0.061	0.170

For example, using the first extracted component, at time 1, the best method is Linear *PCA* (83.3%), and the second rank is for Copula *PCA* (81.2%). However, as time moves forward, the Copula *PCA* gradually will be better until at the final time, Copula *PCA* is the superior (90.7%). Also, if we aim to explain 99.9% of the total variance, we can specify the least number of needed components for each method. Table 11 summarizes the number of components that we need to explain 99.9% of the total variance using polynomial Kernel *PCA*, Gaussian Kernel *PCA*, Laplacian Kernel *PCA*, *ANOVA* Kernels *PCA*, Copula *PCA* and Linear *PCA*. In a nutshell, at time 1, Linear *PCA* is the best while at times 3 and 4 Copula *PCA* is highly accomplished.

Finally, due to the close competition between Copula *PCA*, Linear-*PCA*, and polynomial Kernel *PCA*, for the implementation of the *SCFE-NN* algorithm, we utilize the components extracted by these 3 methods. We execute the algorithm *SCFE-NN* and predict the response variable AAPL.Close. The differences between the predicted values and the true values will be our criterion to assess the algorithm. For this purpose, we split the data into train and test sets with respectively 70% and 30% of cases. The prediction results are stated in Table 11. As can be realized from this table, the *SCFE-NN* algorithm possesses less overfitting and provides better predictions with any number of components it extracts. Also, the values of the optimized parameters of the neural network in each fold of each method in this dataset are given in Table 10.

Table 8: The results of the percentage of variance explained by six *PCA* methods at different times in real dataset (Apple stock)

time 1 (2008/04/01-2012/03/30)	comp 1	comp 2	comp 3	comp 4	comp 5
polynomial Kernel <i>PCA</i>	0.711	0.922	0.975	0.999	1
Gaussian Kernel <i>PCA</i>	0.554	0.775	0.887	0.956	1
Laplacian Kernel <i>PCA</i>	0.561	0.753	0.866	0.948	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.503	0.732	0.854	0.932	1
Copula <i>PCA</i>	0.812	0.985	0.998	0.999	1
Linear <i>PCA</i>	0.833	0.994	0.999	0.999	1
time 2 (2008/04/01-2016/03/31)	comp 1	comp 2	comp 3	comp 4	comp 5
polynomial Kernel <i>PCA</i>	0.644	0.932	0.977	0.999	1
Gaussian Kernel <i>PCA</i>	0.565	0.818	0.915	0.964	1
Laplacian Kernel <i>PCA</i>	0.588	0.774	0.886	0.957	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.470	0.754	0.862	0.948	1
Copula <i>PCA</i>	0.857	0.997	0.998	0.999	1
Linear <i>PCA</i>	0.883	0.993	0.997	0.999	1
time 3 (2008/04/01-2020/03/31)	comp 1	comp 2	comp 3	comp 4	comp 5
polynomial Kernel <i>PCA</i>	0.770	0.971	0.989	0.999	1
Gaussian Kernel <i>PCA</i>	0.544	0.817	0.903	0.964	1
Laplacian Kernel <i>PCA</i>	0.552	0.773	0.874	0.945	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.463	0.764	0.866	0.944	1
Copula <i>PCA</i>	0.887	0.998	0.999	0.999	1
Linear <i>PCA</i>	0.893	0.995	0.996	0.999	1
time 4 (2008/04/01-2024/02/20)	comp 1	comp 2	comp 3	comp 4	comp 5
polynomial Kernel <i>PCA</i>	0.688	0.981	0.996	0.999	1
Gaussian Kernel <i>PCA</i>	0.612	0.864	0.942	0.984	1
Laplacian Kernel <i>PCA</i>	0.582	0.814	0.900	0.965	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.523	0.794	0.897	0.964	1
Copula <i>PCA</i>	0.907	0.998	0.999	0.999	1
Linear <i>PCA</i>	0.864	0.996	0.998	0.999	1

Table 9: The number of components that explain 99.9% of the total variance in each method and at different times in the real data set (Apple stock)

time	1	2	3	4
polynomial Kernel <i>PCA</i>	4	4	4	4
Gaussian Kernel <i>PCA</i>	5	5	5	5
Laplacian Kernel <i>PCA</i>	5	5	5	5
<i>ANOVA</i> Kernels <i>PCA</i>	5	5	5	5
Copula <i>PCA</i>	4	4	3	3
Linear <i>PCA</i>	3	4	4	4

Table 10: Values of the optimal parameters of the neural network in the Apple stock dataset

time	Method	Fold	units	stepmax	number-layers	MAE
1	linear- <i>PCA-NN</i>	1	4	$1e+06$	1	0.001
		2	4	$1e+07$	1	0.001
		3	4	$1e+06$	2	0.001
		4	4	$1e+07$	2	0.000
		5	6	$1e+07$	2	0.001
	polynomial Kernel <i>PCA-NN</i>	1	6	$1e+07$	3	0.004
		2	6	$1e+06$	2	0.005
		3	6	$1e+05$	2	0.004
		4	6	$1e+05$	2	0.004
		5	8	$1e+05$	2	0.005
	<i>SCFE-NN</i>	1	4	$1e+07$	3	0.001
		2	8	$1e+06$	3	0.001
		3	8	$1e+05$	1	0.001
		4	6	$1e+05$	3	0.000
		5	4	$1e+07$	2	0.001
2	linear- <i>PCA-NN</i>	1	6	$1e+06$	2	0.001
		2	4	$1e+05$	1	0.001
		3	4	$1e+06$	1	0.001
		4	8	$1e+05$	3	0.001
		5	6	$1e+05$	2	0.001
	polynomial Kernel <i>PCA-NN</i>	1	6	$1e+05$	2	0.003
		2	6	$1e+07$	2	0.003
		3	6	$1e+07$	2	0.003
		4	8	$1e+07$	2	0.003
		5	8	$1e+06$	2	0.004
	<i>SCFE-NN</i>	1	6	$1e+07$	2	0.001
		2	8	$1e+06$	1	0.001
		3	6	$1e+07$	1	0.001
		4	6	$1e+05$	1	0.001
		5	8	$1e+05$	2	0.001
3	linear- <i>PCA-NN</i>	1	6	$1e+05$	3	0.001
		2	8	$1e+05$	3	0.001
		3	4	$1e+05$	2	0.000
		4	6	$1e+05$	3	0.001
		5	8	$1e+07$	1	0.001
	polynomial Kernel <i>PCA-NN</i>	1	6	$1e+06$	2	0.004
		2	6	$1e+05$	3	0.003
		3	8	$1e+05$	2	0.003
		4	6	$1e+05$	2	0.003
		5	8	$1e+05$	2	0.004
	<i>SCFE-NN</i>	1	6	$1e+06$	1	0.001
		2	8	$1e+06$	2	0.000
		3	8	$1e+05$	2	0.000
		4	8	$1e+05$	2	0.001
		5	6	$1e+05$	2	0.001

time	Method	Fold	units	stepmax	number-layers	MAE
4	linear- <i>PCA-NN</i>	1	8	$1e+05$	3	0.002
		2	8	$1e+07$	3	0.001
		3	6	$1e+07$	2	0.000
		4	8	$1e+06$	1	0.001
		5	6	$1e+06$	2	0.002
	polynomial Kernel <i>PCA-NN</i>	1	6	$1e+06$	2	0.003
		2	6	$1e+05$	3	0.003
		3	8	$1e+07$	2	0.003
		4	6	$1e+05$	2	0.004
		5	8	$1e+05$	2	0.004
	<i>SCFE-NN</i>	1	6	$1e+05$	2	0.001
		2	4	$1e+06$	2	0.001
		3	8	$1e+05$	2	0.000
		4	8	$1e+06$	2	0.001
		5	6	$1e+05$	3	0.000

Table 11: Comparison results of algorithm model *SCFE-NN* with other models at different times in real time (Apple stock)

time	Estimation	linear- <i>PCA-NN</i>	polynomial Kernel <i>PCA-NN</i>	<i>SCFE-NN</i>	<i>PCR</i>
1	mean of <i>RMSE</i> train	0.003	0.007	0.003	4.98
	SD of <i>RMSE</i> train	0.001	0.001	0.000	0.01
	mean of <i>RMSE</i> test	0.003	0.014	0.003	4.92
	SD of <i>RMSE</i> test	0.001	0.009	0.001	0.01
2	mean of <i>RMSE</i> train	0.002	0.007	0.002	6.3
	SD of <i>RMSE</i> train	0.000	0.004	0.000	0.02
	mean of <i>RMSE</i> test	0.001	0.012	0.002	6.3
	SD of <i>RMSE</i> test	0.000	0.000	0.000	0.05
3	mean of <i>RMSE</i> train	0.002	0.009	0.001	25.44
	SD of <i>RMSE</i> train	0.001	0.005	0.000	0.15
	mean of <i>RMSE</i> test	0.003	0.011	0.001	25.13
	SD of <i>RMSE</i> test	0.001	0.001	0.000	0.3
4	mean of <i>RMSE</i> train	0.004	0.010	0.002	27.77
	SD of <i>RMSE</i> train	0.000	0.005	0.000	0.13
	mean of <i>RMSE</i> test	0.003	0.014	0.002	27.64
	SD of <i>RMSE</i> test	0.001	0.007	0.000	0.21

3.2.2. COVID-19 dataset

The second dataset we considered and implemented our algorithm is the *COVID-19* dataset. This dataset has been recorded daily from 27-7-2020 to 22-11-2020³. In this experiment, we considered 8 variables including “Confirmed”, “Deaths”, “Recovered”, “Active”, “New.cases”, “New.recovered”, “Recovered/100.Cases”, “Deaths/100.Recovered”. Suppose we aim to perform dimension reduction on this considered dataset. To express the method described in this article well, we considered 3 times of 11 days from this data set and implemented the described methods on them. The results are presented in Table 12 and Figure 2.

Table 12: The results of the percentage of variance explained by Linear *PCA* and Copula *PCA* and *PCA* methods at different times in real dataset

time 1 (2020/01/22-2020/01/31)	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8
polynomial Kernel <i>PCA</i>	0.723	0.923	0.958	0.975	0.988	0.994	0.997	1
Gaussian Kernel <i>PCA</i>	0.432	0.718	0.848	0.911	0.947	0.976	0.994	1
Laplacian Kernel <i>PCA</i>	0.426	0.648	0.755	0.826	0.878	0.926	0.966	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.407	0.684	0.824	0.890	0.928	0.962	0.989	1
Copula <i>PCA</i>	0.783	0.886	0.931	0.969	0.996	0.997	0.998	1
Linear <i>PCA</i>	0.768	0.940	0.975	0.995	0.995	0.998	0.999	1
time 2 (2020/01/22-2020/02/10)	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8
polynomial Kernel <i>PCA</i>	0.843	0.913	0.948	0.978	0.989	0.995	0.998	1
Gaussian Kernel <i>PCA</i>	0.503	0.750	0.864	0.932	0.957	0.979	0.992	1
Laplacian Kernel <i>PCA</i>	0.505	0.695	0.803	0.867	0.912	0.946	0.976	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.473	0.716	0.837	0.915	0.947	0.971	0.988	1
Copula <i>PCA</i>	0.775	0.928	0.957	0.983	0.996	0.997	0.998	1
Linear <i>PCA</i>	0.827	0.953	0.985	0.994	0.999	0.999	0.999	1
time 3 (2020/01/22-2020/02/20)	comp 1	comp 2	comp 3	comp 4	comp 5	comp 6	comp 7	comp 8
polynomial Kernel <i>PCA</i>	0.596	0.779	0.921	0.966	0.981	0.992	0.997	1
Gaussian Kernel <i>PCA</i>	0.469	0.697	0.779	0.847	0.901	0.950	0.981	1
Laplacian Kernel <i>PCA</i>	0.481	0.661	0.749	0.814	0.872	0.925	0.973	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.452	0.691	0.773	0.836	0.892	0.936	0.976	1
Copula <i>PCA</i>	0.760	0.942	0.978	0.999	0.999	0.999	0.999	1
Linear <i>PCA</i>	0.769	0.914	0.958	0.987	0.999	0.999	0.999	1

As can be understood from Table 12, aiming to reach 99% of the explained variance in the first time, polynomial Kernel *PCA* needs 6 components, Gaussian Kernel *PCA* method needs 7 components, the Laplacian Kernel *PCA*, *ANOVA* Kernels *PCA*, Copula *PCA*, and Linear *PCA* need respectively, 8, 8, 5 and 4 components. But at the last time, these numbers changed into 6, 8, 8, 8, 8, 4, and 5 for polynomial Kernel *PCA*, Gaussian Kernel *PCA*, Laplacian Kernel *PCA*, *ANOVA* Kernels *PCA*, Copula *PCA*, and Linear *PCA* respectively, which shows a significance improvement of Copula *PCA*.

Regarding constructing the *SCFE-NN* algorithm in this dataset, we assume New.deaths as the response variable and predict this variable using the principal components that explained 99% of the total variance in the previous best methods, *i.e.*, polynomial Kernel *PCA*, Linear *PCA*, and Copula *PCA*. We set the extracted components as inputs of a neural network, and the results of comparing different models at the specified times are presented in Table 15, Also, the values of the optimized parameters of the neural network in each fold

³<https://www.kaggle.com/datasets/imdevskp/corona-virus-report>

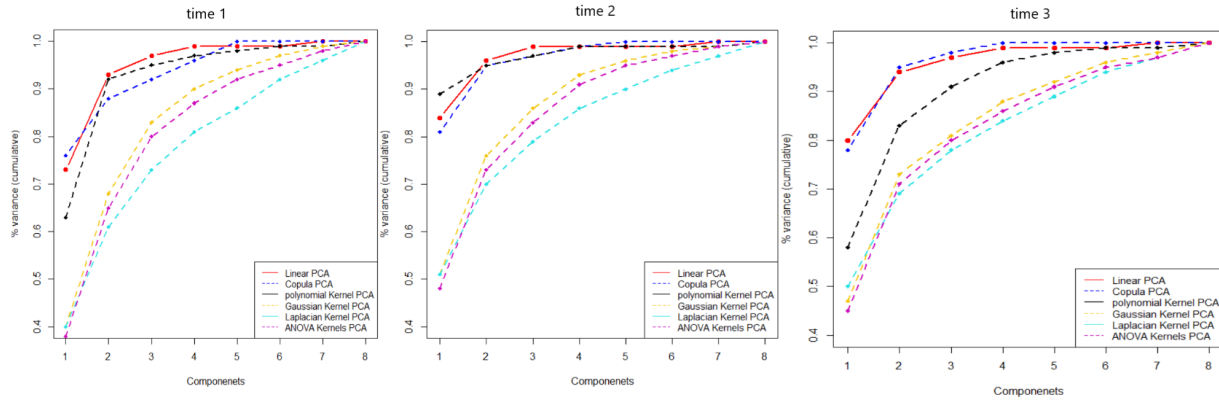


Figure 2: Percents of explained variance using linear *PCA*, Copula *PCA* and kernel *PCA* at different times in real-data (*COVID-19*)

Table 13: The number of components that explain 99% of the total variance in each method and at different times in the real data set

time	1	2	3
polynomial Kernel <i>PCA</i>	6	6	6
Gaussian Kernel <i>PCA</i>	7	7	8
Laplacian Kernel <i>PCA</i>	8	8	8
<i>ANOVA</i> Kernels <i>PCA</i>	8	8	8
Copula <i>PCA</i>	5	5	4
Linear <i>PCA</i>	4	4	5

of each method in this dataset are given in Table 14. Based on this table, the *SCFE-NN* algorithm consistently exhibits less overfitting compared to other methods and provides a more accurate prediction.

3.2.3. Bitcoin dataset

The third data set that we implemented our algorithm on is the Bitcoin data set, which in this study focuses on daily data from 1-1-2025 to 5-1-2025⁴ and we have also considered 5 variables “Bitcoin.Price”, “Bitcoin.Open”, “Bitcoin.High”, “Bitcoin.Low” and “Bitcoin.Change” from this data set. In this regard, 3 time periods 4-1-2025 to 6-1-2025, 4-1-2025 to 7-1-2025 and 4-1-2025 to 8-1-2025 have been considered for this research. The results are presented in Table 16 and Figure 6. Considering that the purpose of presenting this dataset is to implement the described method on daily data, for this reason, the bootstrap method has been used to increase the volume of data for a more accurate analysis.

As can be understood from Table 16, aiming to reach 99% of the explained variance in the first time, polynomial Kernel *PCA* needs 5 components, Gaussian Kernel *PCA* method needs 5 components, the Laplacian Kernel *PCA*, *ANOVA* Kernels *PCA*, Copula *PCA*, and Linear *PCA* need respectively, 5, 5, 3 and 2 components. But at the last time, these numbers changed into 5, 5, 5, 5, 3, and 3 for polynomial Kernel *PCA*, Gaussian Kernel *PCA*, Laplacian

⁴<https://www.investing.com/crypto/bitcoin/historical-data>

Table 14: Values of the optimal parameters of the neural network in the *COVID-19* dataset

time	Method	Fold	units	stepmax	number-layers	MAE
1	linear- <i>PCA-NN</i>	1	8	$1e+06$	2	0.215
		2	8	$1e+05$	1	0.109
		3	68	$1e+06$	3	0.070
		4	6	$1e+05$	3	0.076
		5	8	$1e+05$	3	0.165
	polynomial Kernel <i>PCA-NN</i>	1	4	$1e+05$	3	0.046
		2	8	$1e+07$	1	0.010
		3	8	$1e+05$	2	0.008
		4	4	$1e+07$	3	0.005
		5	4	$1e+05$	1	0.001
	<i>SCFE-NN</i>	1	6	$1e+07$	2	0.347
		2	4	$1e+05$	1	0.492
		3	6	$1e+06$	1	0.507
		4	4	$1e+05$	1	0.362
		5	8	$1e+06$	2	0.140
2	linear- <i>PCA-NN</i>	1	8	$1e+06$	3	0.677
		2	8	$1e+06$	1	0.650
		3	8	$1e+07$	1	0.705
		4	6	$1e+06$	2	0.641
		5	8	$1e+07$	1	0.558
	polynomial Kernel <i>PCA-NN</i>	1	4	$1e+06$	3	0.027
		2	8	$1e+07$	1	0.022
		3	6	$1e+06$	3	0.040
		4	8	$1e+07$	3	0.015
		5	8	$1e+05$	2	0.016
	<i>SCFE-NN</i>	1	4	$1e+05$	3	0.639
		2	8	$1e+06$	3	0.557
		3	6	$1e+07$	3	0.545
		4	4	$1e+05$	3	0.700
		5	8	$1e+05$	3	0.486
3	linear- <i>PCA-NN</i>	1	8	$1e+05$	3	0.663
		2	8	$1e+07$	3	0.688
		3	4	$1e+07$	2	0.639
		4	8	$1e+05$	3	0.558
		5	8	$1e+07$	1	0.738
	polynomial Kernel <i>PCA-NN</i>	1	6	$1e+06$	2	0.040
		2	4	$1e+06$	3	0.033
		3	8	$1e+07$	3	0.178
		4	4	$1e+05$	2	0.160
		5	8	$1e+05$	2	0.125
	<i>SCFE-NN</i>	1	4	$1e+06$	1	0.506
		2	8	$1e+06$	2	0.527
		3	8	$1e+05$	2	0.491
		4	8	$1e+05$	2	0.523
		5	8	$1e+07$	3	0.440

Table 15: Comparison results of algorithm model *SCFE-NN* with other models at different times in real time

time	Estimation	linear- <i>PCA-NN</i>	polynomial Kernel <i>PCA-NN</i>	<i>SCFE-NN</i>	<i>PCR</i>
1	mean of <i>RMSE</i> train	0.764	0.032	0.866	1.79
	SD of <i>RMSE</i> train	0.270	0.021	0.120	0.33
	mean of <i>RMSE</i> test	0.977	0.999	0.925	1.17
	SD of <i>RMSE</i> test	0.502	0.703	0.180	0.97
2	mean of <i>RMSE</i> train	0.970	0.080	0.970	1.58
	SD of <i>RMSE</i> train	0.028	0.026	0.127	0.14
	mean of <i>RMSE</i> test	1.05	1.022	0.982	1.30
	SD of <i>RMSE</i> test	0.140	1.437	0.136	0.64
3	mean of <i>RMSE</i> train	0.980	0.979	0.957	1.39
	SD of <i>RMSE</i> train	0.068	0.251	0.071	0.17
	mean of <i>RMSE</i> test	0.933	1.02	0.962	1.51
	SD of <i>RRMSE</i> test	0.308	0.601	0.263	0.85

Table 16: The results of the percentage of variance explained by Linear *PCA* and Copula *PCA* and *PCA* methods at different times in real dataset

time 1 (2025/01/04-2025/01/06)	comp 1	comp 2	comp 3	comp 4	comp 5
polynomial Kernel <i>PCA</i>	0.522	0.721	0.845	0.965	1
Gaussian Kernel <i>PCA</i>	0.468	0.686	0.839	0.969	1
Laplacian Kernel <i>PCA</i>	0.401	0.607	0.774	0.926	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.404	0.625	0.793	0.949	1
Copula <i>PCA</i>	0.861	0.970	0.996	0.998	1
Linear <i>PCA</i>	0.952	0.999	0.999	0.999	1
time 2 (2025/01/04-2025/01/07)	comp 1	comp 2	comp 3	comp 4	comp 5
polynomial Kernel <i>PCA</i>	0.441	0.797	0.890	0.961	1
Gaussian Kernel <i>PCA</i>	0.308	0.576	0.792	0.918	1
Laplacian Kernel <i>PCA</i>	0.269	0.541	0.741	0.886	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.304	0.530	0.744	0.890	1
Copula <i>PCA</i>	0.667	0.980	0.999	0.999	1
Linear <i>PCA</i>	0.714	0.996	0.999	0.999	1
time 3 (2025/01/04-2025/01/08)	comp 1	comp 2	comp 3	comp 4	comp 5
polynomial Kernel <i>PCA</i>	0.416	0.643	0.815	0.932	1
Gaussian Kernel <i>PCA</i>	0.291	0.540	0.722	0.896	1
Laplacian Kernel <i>PCA</i>	0.270	0.519	0.697	0.866	1
<i>ANOVA</i> Kernels <i>PCA</i>	0.296	0.503	0.686	0.862	1
Copula <i>PCA</i>	0.636	0.957	0.995	0.998	1
Linear <i>PCA</i>	0.583	0.945	0.999	0.999	1

Kernel *PCA*, *ANOVA* Kernels *PCA*, Copula *PCA*, and Linear *PCA* respectively, which shows a significance improvement of Copula *PCA*.

Table 17: The number of components that explain 99% of the total variance in each method and at different times in the real data set

time	1	2	3
polynomial Kernel <i>PCA</i>	5	5	5
Gaussian Kernel <i>PCA</i>	5	5	5
Laplacian Kernel <i>PCA</i>	5	5	5
<i>ANOVA</i> Kernels <i>PCA</i>	5	5	5
Copula <i>PCA</i>	3	3	3
Linear <i>PCA</i>	2	2	3

Regarding the construction of the *SCFE-NN* algorithm for this dataset, we consider Volume as the response variable. This variable is predicted using the principal components that account for 99% of the total variance, derived from the previously best-performing methods: polynomial Kernel *PCA*, Linear *PCA*, and Copula *PCA*. The extracted components are utilized as inputs for a neural network. The results comparing different models at specified times are presented in Table 19. Additionally, the optimized parameters of the neural network for each fold of each method in this dataset are provided in Table 18. According to this table, the *SCFE-NN* algorithm consistently demonstrates less overfitting compared to other methods, resulting in more accurate predictions.

4. Conclusion

When dealing with streaming data, quick decision-making and data management is of great importance because when faced with such data, the number of observations is high and variables are often interdependent. Additionally, non-linear relationships may become apparent over time with an increase in the number of observations in the variables. Therefore, we were looking for a method that takes these issues into account and improves prediction accuracy. Our proposed Stream Copula Feature Extraction (*SCFE*) algorithm benefits using Kendall’s τ association measure to extract the best informative components. We investigate the performance of our algorithm using a simulation dataset, the online streaming real dataset of Apple Inc., and the recorded dataset of *COVID-19*. Compared to other alternative feature extraction methods, we showed that our approach yields more percentage of explained variance and our *SCFE-NN* approach provides good results in neural network prediction analysis, also showed in the Bitcoin data set that when the amount of data is small, the proposed algorithm is equal to previous methods such as *PCA*, and as time goes on, the algorithm explains a higher percentage of variance with a smaller number of components.

Our method can be extended in many fashions. In this study, the copula function was used to extract features in data streams. It is recommended for future research to utilize the copula function in other dimension reduction methods such as feature selection. Also, although we have used a neural network algorithm in the prediction part, one may consider using other machine learning methods such as a deep learning approach in the prediction analysis, on the other hand, to increase the accuracy of the models, the number of layers,

Table 18: Values of the optimal parameters of the neural network in the Bitcoin dataset

time	Method	Fold	units	stepmax	number-layers	MAE
1	linear- <i>PCA-NN</i>	1	6	$1e + 06$	1	24.61
		2	4	$1e + 05$	1	26.82
		3	4	$1e + 05$	3	26.27
		4	6	$1e + 05$	1	24.03
		5	8	$1e + 05$	3	26.49
	polynomial Kernel <i>PCA-NN</i>	1	6	$1e + 07$	1	0.0014
		2	6	$1e + 07$	3	0.0003
		3	6	$1e + 06$	2	0.000
		4	6	$1e + 05$	3	0.001
		5	6	$1e + 07$	3	0.003
	<i>SCFE-NN</i>	1	4	$1e + 05$	1	26.82
		2	6	$1e + 06$	2	22.87
		3	4	$1e + 05$	1	25.95
		4	4	$1e + 05$	2	21.32
		5	6	$1e + 07$	2	21.32
2	linear- <i>PCA-NN</i>	1	8	$1e + 05$	1	25.80
		2	8	$1e + 06$	2	29.97
		3	8	$1e + 06$	1	28.61
		4	6	$1e + 07$	2	12.36
		5	8	$1e + 05$	1	25.78
	polynomial Kernel <i>PCA-NN</i>	1	8	$1e + 06$	3	0.000
		2	4	$1e + 06$	1	0.011
		3	8	$1e + 05$	3	0.006
		4	8	$1e + 07$	3	0.000
		5	6	$1e + 05$	2	0.004
	<i>SCFE-NN</i>	1	6	$1e + 05$	3	22.29
		2	4	$1e + 05$	3	24.73
		3	6	$1e + 05$	1	27.98
		4	4	$1e + 06$	3	22.60
		5	8	$1e + 06$	2	29.64
3	linear- <i>PCA-NN</i>	1	6	$1e + 05$	1	23.49
		2	8	$1e + 05$	2	26.87
		3	6	$1e + 07$	1	25.52
		4	6	$1e + 07$	2	19.41
		5	4	$1e + 05$	2	29.81
	polynomial Kernel <i>PCA-NN</i>	1	8	$1e + 07$	2	0.005
		2	8	$1e + 06$	1	0.002
		3	6	$1e + 05$	1	0.013
		4	8	$1e + 06$	1	0.003
		5	6	$1e + 07$	3	0.004
	<i>SCFE-NN</i>	1	8	$1e + 07$	1	13.22
		2	4	$1e + 07$	1	22.00
		3	6	$1e + 07$	1	26.72
		4	8	$1e + 07$	3	22.98
		5	6	$1e + 06$	3	27.94

Table 19: Comparison results of algorithm model *SCFE-NN* with other models at different times in real time

time	Estimation	linear- <i>PCA-NN</i>	polynomial Kernel <i>PCA-NN</i>	<i>SCFE-NN</i>	<i>PCR</i>
1	mean of <i>RMSE</i> train	26.394	0.008	22.94	34.258
	SD of <i>RMSE</i> train	0.554	0.012	2.37	2.752
	mean of <i>RMSE</i> test	32.297	27.89	25.58	37.003
	SD of <i>RMSE</i> test	3.225	11.727	3.021	3.985
2	mean of <i>RMSE</i> train	30.324	0.019	28.162	37.658
	SD of <i>RMSE</i> train	1.205	0.052	1.632	2.702
	mean of <i>RMSE</i> test	37.747	44.121	34.362	48.256
	SD of <i>RMSE</i> test	5.643	13.910	5.323	9.602
3	mean of <i>RMSE</i> train	29.618	8.65	27.325	42.369
	SD of <i>RMSE</i> train	1.401	12.43	1.39	2.963
	mean of <i>RMSE</i> test	35.239	58.49	30.026	49.521
	SD of <i>RMSE</i> test	3.168	24.725	3.125	10.325

units and stepmax of the neural network have been optimized.

Also, besides using Kendall's τ association measures to extract components, one may interested in extracting these components using other copula-based concordance measures such as Spearman's ρ , Spearman's footrule ϕ and Gini's γ , see *e.g.* Mesiar *et al.* (2024). We aim to extend these results using a linear combination of concordance measures in our next ongoing work but over on going work will be extending this result to sum predictive models such as multiple regression or principal component regression (*PCR*).

Acknowledgments

The authors are thankful to the editor and referees for their constructive and helpful comments which have significantly improved the article.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Bahri, M., Bifet, A., Maniu, S., and Gomes, H. M. (2020). Survey on feature transformation techniques for data streams. In Bessiere, C., editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2020)*, pages 4796–4802. Survey track.
- Carrillo, J., Nieto, M., Velez, J., and Velez, D. (2021). A new machine learning forecasting algorithm based on bivariate copula functions. *Forecasting*, **3**, 355–376.
- Cuvelier, E. and Noirhomme-Fraiture, M. (2005). Clayton copula and mixture decomposition. In *Applied Stochastic Models and Data Analysis (ASMDA 2005)*, pages 699–708. ENST Bretagne, Brest, France.
- Fukunaga, K. and Short, R. D. (1978). Nonlinear feature extraction with a general criterion function. *IEEE Transactions on Information Theory*, **24**, 600–607.

- Houari, R., Bounceur, A., Kechadi, M.-T., Tari, A.-K., and Euler, R. (2016). Dimensionality reduction in data mining: A copula approach. *Expert Systems with Applications*, **64**, 247–260.
- Jia, W., Sun, M., Lian, J., and Hou, S. (2022). Feature dimensionality reduction: A review. *Complex & Intelligent Systems*, **8**, 2663–2693.
- Krawczak, M. and Szkatuła, G. (2014). An approach to dimensionality reduction in time series. *Information Sciences*, **260**, 15–36.
- Margara, A. and Rabl, T. (2019). Definition of data streams. In Sakr, S. and Zomaya, A. Y., editors, *Encyclopedia of Big Data Technologies*, pages 648–652. Springer International Publishing.
- Mesiar, R., Kolesárová, A., Sheikhi, A., and Shvydka, S. (2024). Convex weak concordance measures and their constructions. *Fuzzy Sets and Systems*, **478**, 108841.
- Milosevic, S., Frank, P., Leike, R. H., Müller, A., and Enßlin, T. A. (2021). Bayesian decomposition of the galactic multi-frequency sky using probabilistic autoencoders. *Astronomy & Astrophysics*, **650**, A100.
- Misic, M., Durdevic, D., and Tomasevic, M. (2012). Evolution and trends in gpu computing. In *Proceedings of MIPRO 2012, 35th International Convention*, pages 289–294.
- Mutlag, W., Ali, S., Mosad, Z., and Ghrabat, B. H. (2020). Feature extraction methods: A review. *Journal of Physics: Conference Series*, **1591**, 012028.
- Nelsen, R. B. (2006). *An Introduction to Copulas*. Springer, New York, NY, USA.
- Park, C. and Lee, G.-H. (2020). Comparison of incremental linear dimension reduction methods for streaming data. *Pattern Recognition Letters*, **135**.
- Park, J.-H., Sriram, T., and Yin, X. (2010). Dimension reduction in time series. *Statistica Sinica*, **20**, 747–770.
- Qu, X. (2012). Feature extraction by combining independent subspaces analysis and copula techniques. In *Proceedings of the 2012 International Conference on System Science and Engineering (ICSSE 2012)*, pages 326–329.
- Ram, J. (2023). A review on gpu architectures and programming. In *Proceedings of the 2nd National Conference on Engineering Applications of Emerging Technology (in association with International Journal of Scientific Research in Science, Engineering and Technology)*.
- Salem, N. and Hussein, S. (2019). Data dimensional reduction and principal components analysis. *Procedia Computer Science*, **163**, 292–299.
- Sarveshwaran, V., Sevugan, A., and Swamidason, I. T. J. (2019). A review of dimensionality reduction techniques for efficient computation. *Procedia Computer Science*, **165**, 104–111.
- Sheikhi, A., Arad, F., and Mesiar, R. (2022a). A heteroscedasticity diagnostic of a regression analysis with copula dependent random variables. *Brazilian Journal of Probability and Statistics*, **36**, 1–19.
- Sheikhi, A., Mesiar, R., and Holeňa, M. (2022b). A dimension reduction in neural network using copula matrix. *International Journal of General Systems*, **52**, 1–16.
- Simard, C. and Remillard, B. (2013). Forecasting time series with multivariate copulas. SSRN Electronic Journal, doi:10.2139/ssrn.2295120.

- Toharudin, T., Caraka, R., Bachrudin, A., Abu Bakar, S., and Ambarsari, D. (2019). Copula-based feedforward neural network genetic algorithm in cargo forecasting. *International Journal of Advances in Soft Computing and its Applications*, **11**, 22–33.
- Upadhyay, D., Jain, S., and Jain, A. (2013). Comparative analysis of various data stream mining procedures and various dimension reduction techniques. *International Journal of Advanced Research in Computer Science*, **4**.
- van der Maaten, L., Postma, E. O., and van den Herik, H. J. (2009). Dimensionality reduction: A comparative review. *Journal of Machine Learning Research*, **10**, 66–71.
- Wang, L., Guo, X., Zeng, J.-C., and Hong, Y. (2010). Using gumbel copula and empirical marginal distribution in estimation of distribution algorithm. In *3rd International Workshop on Advanced Computational Intelligence (IWACI 2010)*, pages 583–587.
- Wang, Q. and Megalooikonomou, V. (2008). A dimensionality reduction technique for efficient time series similarity analysis. *Information Systems*, **33**, 115–132.
- Witten, D., Tibshirani, R., and Hastie, T. (2009). A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, **10**, 515–534.
- Xie, L., Guo, W., Wei, H., Tang, Y., and Tao, D. (2020). Efficient unsupervised dimension reduction for streaming multiview data. *IEEE Transactions on Cybernetics*, **52**, 1772–1784.
- Yan, J., Zhang, B., Liu, N., Yan, S., Cheng, Q., Fan, W., Yang, Q., Xi, W., and Chen, Z. (2006). Effective and efficient dimensionality reduction for large-scale and streaming data preprocessing. *IEEE Transactions on Knowledge and Data Engineering*, **18**, 320–333.
- Zeng, Z. and Wang, T. (2022). Neural copula: A unified framework for estimating generic high-dimensional copula functions. arXiv preprint arXiv:2205.15031 [stat.ML].
- Zheng, W., Chen, S., Fu, Z., Li, J., and Yang, J. (2022). Streaming feature selection via graph diffusion. *Information Sciences*, **618**, 150–168.
- Zhu, Q. (2010). Reformative nonlinear feature extraction using kernel MSE. *Neurocomputing*, **73**, 3334–3337.

ANNEXURE

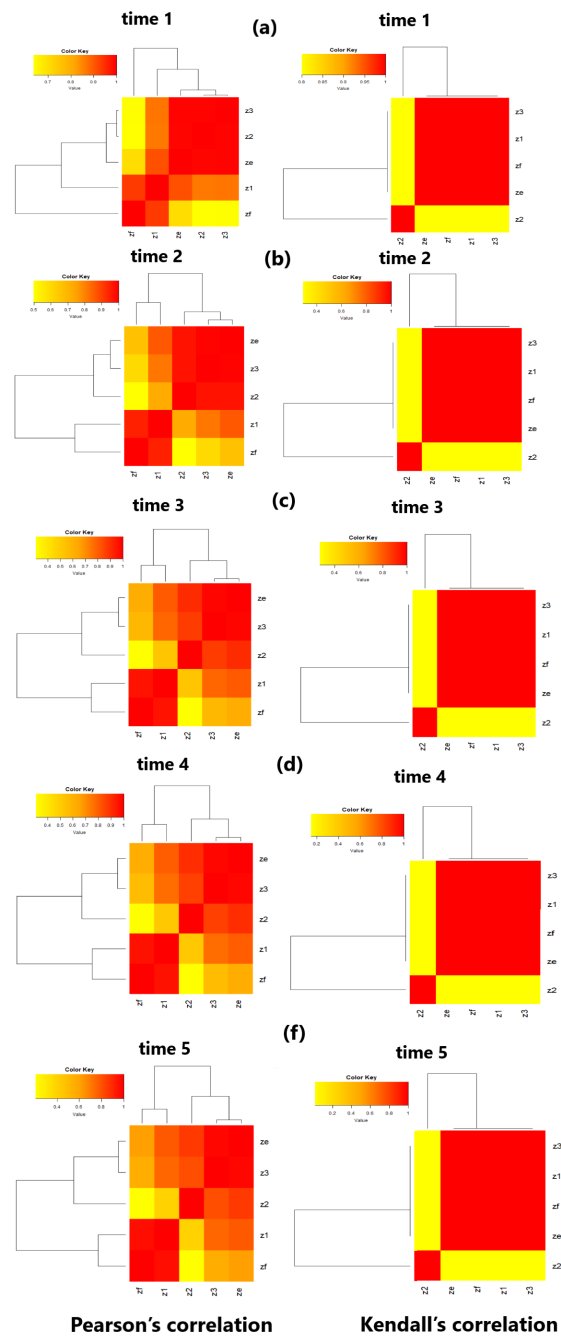


Figure 3: Heatmap plots: Left side) Pearson's correlation from time one to time five, Right side) Kendall's association from time one to time five

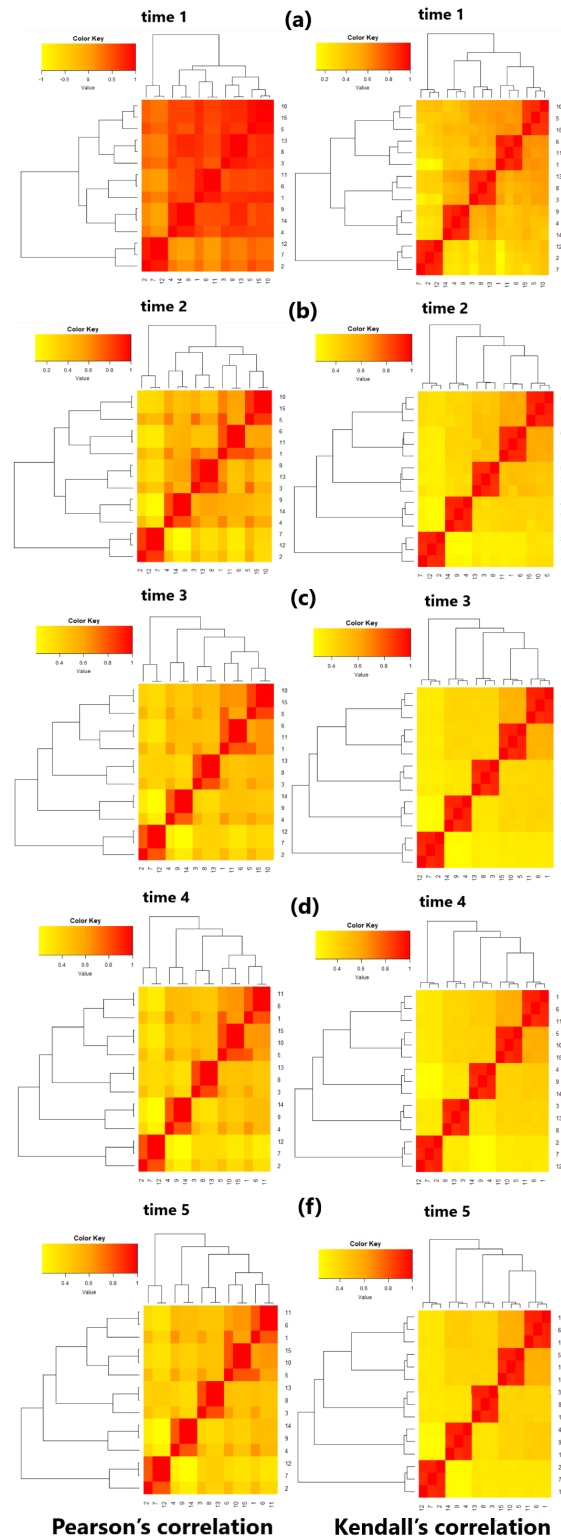


Figure 4: Heatmap plots: Left side) Pearson's correlation from time one to time five in simulated data, Right side) Kendall's association from time one to time five in simulated data

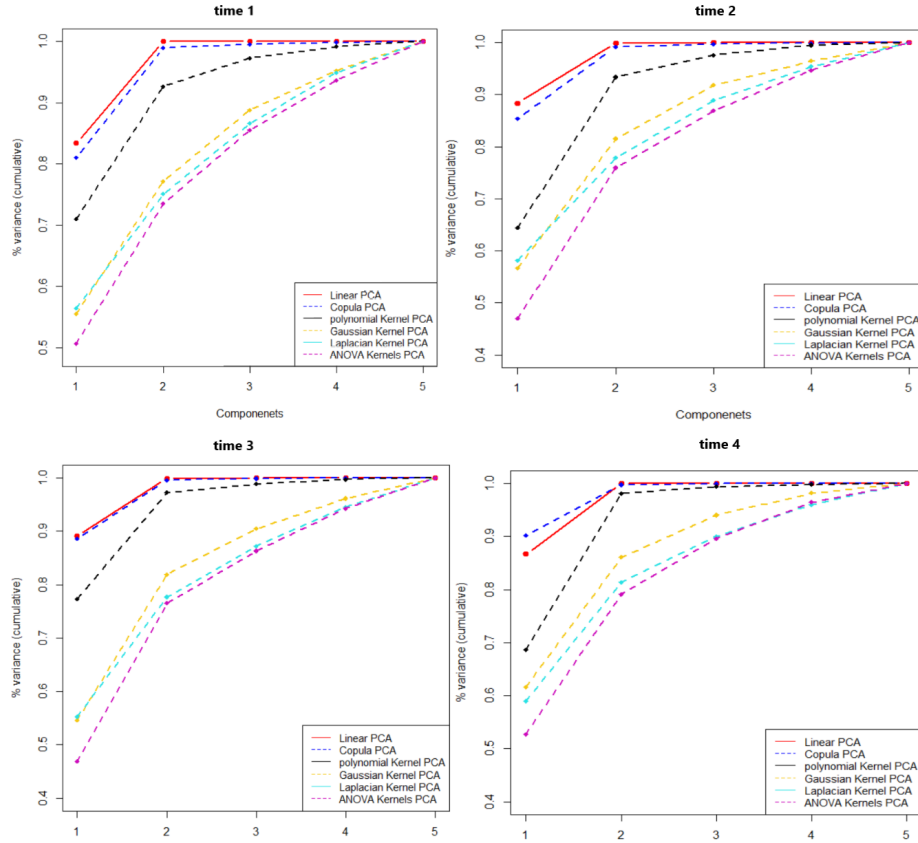


Figure 5: Percents of explained variance using linear *PCA*, Copula *PCA* and kernel *PCA* at different times in real-data (Apple stock)

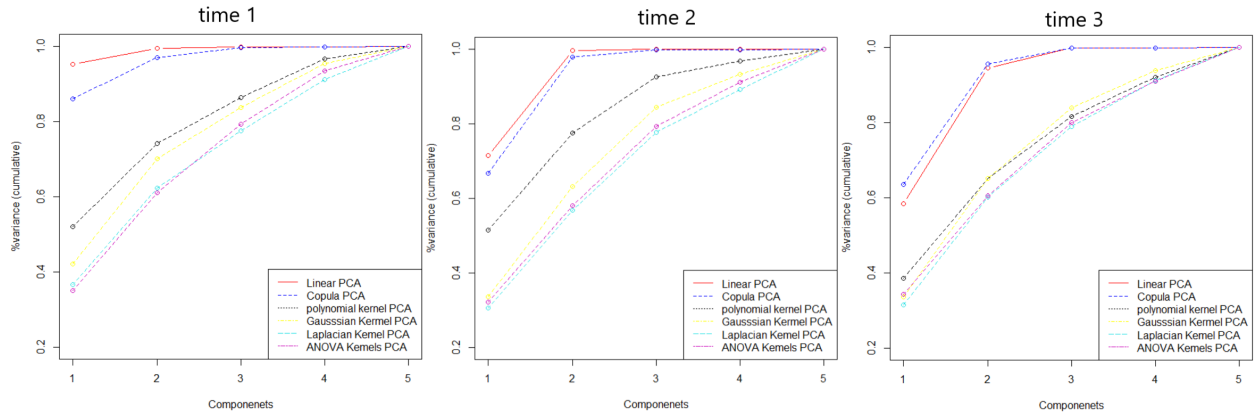
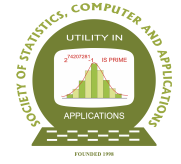


Figure 6: Percents of explained variance using linear *PCA*, Copula *PCA* and kernel *PCA* at different times in real-data (Bitcoin)



A Framework for Building a Novel Causal Proximity Driven GNN from Biomedical Text

Samridhi Dev and Aditi Sharan

*School of Computer and Systems Sciences
Jawaharlal Nehru University, New Delhi 110067*

Received: 11 February 2025; Revised: 25 March 2025; Accepted: 02 April 2025

Abstract

In the digital era, the abundance of biomedical text, along with the advanced Graph Neural Network (GNN) based algorithm, provides a forum for unrevealing significant information hidden in the text. In this context, researchers have tried to predict Adverse Drug Reactions (ADRs) in cancer treatments using GNN. However, ADR prediction in cancer is still a challenging research problem. Current GNN based methodologies face critical limitations, including over-smoothing, equal weighting of neighboring nodes, and inability to capture intersentential, causal, and high-granular indirect information. Additionally, the field is hindered by limited annotated data and the absence of automated methods for corpus construction, making it challenging to scale the models for effective ADR prediction. To overcome these barriers, we propose a two-model framework. The first is a linguistics-driven, rule-based approach designed to automate corpus construction and causal relation extraction process, thereby generating a cancer-specific ADR corpus with annotated entity types and causal relationships. This automated corpus construction leverages sophisticated linguistic rules to ensure accurate and consistent causality annotation, significantly enhancing available resources. Building on this, the second neural network-based model, termed the Causality and Proximity-based Relational Multi-head Attention Model (CPRMAM), is trained on the constructed corpus, integrating both causal semantics and proximity considerations. Together, these models address key limitations in existing GNN frameworks, advancing ADR prediction by enabling the extraction of complex causal structures and enhancing interpretability within oncological contexts.

Key words: Graph neural network; Adverse drug reactions; Aggregation function; CPRMAM; Knowledge graph completion; Data mining; Healthcare; Neural networks.

1. Introduction

With the digitization of health data, a significant amount of textual information is being generated from biomedical journals, research articles, and case reports. However, much

of this data remains untapped due to its unstructured nature. Graph neural network based models offer a systematic approach to extract valuable insights from this information, transforming unstructured medical/biomedical text into structured graphs, leading to meaningful findings. These models assist in maximizing the potential of previously underutilized medical data. By improving data-driven inferences, the model assist healthcare professionals in making informed clinical decisions.

Data-driven graph neural networks have been studied widely to infer meaningful insights for complicated diseases such as cancer. Cancer is a complicated disease mainly because it is caused by changes in genes that control functioning of multiple cells and involve complex progression pathways. In spite of a lot of progress in detecting and treating cancer, cancer-related Adverse Drug Reactions (ADRs) remain a significant challenge in healthcare and biomedical research. To address this, we have chosen to construct a cancer-related corpus specifically for ADR prediction, enabling a more structured and data-driven approach to analyze adverse drug effects in cancer treatment. Effective extraction of cancer-related entities and relations from textual data is crucial for understanding the complexities of drug interactions in oncology. Considering various types of relations that can be extracted from textual data, causal relations extraction is an open research problem. However, the success of such extraction processes hinges on accurate and comprehensive annotation of biomedical data, which is inherently challenging due to its complexity, terminologies, and ambiguity. In the context of ADR prediction, earlier deep learning and machine learning models have shown limitations, especially in capturing complex, intersentential, and indirect relationships within biomedical knowledge graphs. Graph Neural Networks (GNNs) have a better potential of capturing these complicated relations.

This paper proposes a framework to build causal proximity based GNN. The study introduces two models; the first model is a novel causal linguistic driven framework for complex entity and causal relation annotation, leveraging linguistic and grammatical properties alongside a rule-based approach. The framework consists of key subtasks: entity extraction, coreference resolution, rule generation, relation extraction, and the formation of triplets in a novel format that surpasses the limitations of state of art triplet representations. By combining semantic and linguistic dependencies, this approach ensures robust and efficient relation extraction in biomedical texts.

Our second proposed model is the Causality and Proximity-based Relational Multi-head Attention Model (CPRMAM). It is trained on a cancer-specific ADR knowledge graph constructed using the first model. The model integrates causal reasoning and proximity for better knowledge discovery, addressing critical gaps in current methods and improving the robustness and accuracy of predictions. The paper is structured as follows: it begins with the automatic construction of the required corpus, where entities and causal relations are systematically annotated. Afterward, we design and implement the GNN model, leveraging the constructed dataset to predict ADRs in oncology.

2. Related work

This section briefly reviews and summarizes the recent studies related to our work. Adverse drug reaction prediction has been approached using various methodologies, encompassing statistical, machine learning (ML), deep learning (DL), graph-based, and similarity-

based techniques. In statistical-based approaches, researchers have developed drug-pair protein interaction profiles using data from the STITCH and TWOSIDES databases, employing the Laplacian-corrected estimator to predict drug-induced effects [Pauwels *et al.* (2011)]. Another group of researchers modeled ADR-drug relationships with a three-layer hierarchical Bayesian model [Bate *et al.* (1998)], utilizing latent Dirichlet allocation to uncover biochemical mechanisms linking ADRs to drug structures [Liu *et al.* (2017)]. Additional statistical methods include using proportional reporting ratio, reporting odds ratio, and empirical Bayes geometric mean algorithms to identify drug-associated adverse event [Kwak *et al.* (2020)], and extracting biomedical knowledge via MetaMap and SemRep to support reasoning about drug-effect relationships [Yildirim *et al.* (2014)]. Moreover, researchers collected data from seven databases and implemented methods such as Bayesian confidence propagation neural network, gamma Poisson shrinker, proportional reporting ratio, and reporting odds ratio to detect ADRs [Szarfman *et al.* (2002) and Schuemie *et al.* (2012)]. Extended LRT methods based on Poisson models were used to identify ADR signals with disproportionately high reporting rates [Zhao *et al.* (2018)], while other studies leveraged EudraVigilance data [Monaco *et al.* (2017)], tree-based scan statistics [Wang *et al.* (2018)], and disproportionate methods to detect unknown causal associations [Lerch *et al.* (2015)]. In the realm of ML/DL-based approaches, researchers focused on integrating data from DrugBank, DrugCentral, CTD, and TWOSIDES databases, proposing the HCNS-ADR machine learning method to predict ADRs from combined medication [Xiao *et al.* (2017)]. Another team developed a structure-enhanced line graph convolutional network to learn comprehensive representations of drug-disease pairs, transforming the task into a node classification problem using the SEAL architecture for link prediction [Zheng *et al.* (2018) and Shang *et al.* (2014)]. A convolutional framework was also proposed to construct chemical fingerprint features and assess their associations with ADRs [Mantripragada *et al.* (2021)]. Graph-based approaches have seen significant advancements, with authors proposing a GNN model trained on clinical data using SIDER database labels to predict side effect signals between drug-disease pairs [Gao *et al.* (2023)]. Another innovative method, the contextualized graph embedding model (CGEM), captures cause-effect relations for ADR detection by combining contextualized embeddings, convolutional GNNs, and BertGCN for classification [Liu *et al.* (2024) and Dey *et al.* (2018)]. Additional graph-based techniques include analyzing non-directional heterogeneous healthcare data for triad prediction [Zhang *et al.* (2021)], combining deep learning with a biomedical tripartite network to predict drug-ADR associations [Xue *et al.* (2020)], and inferring new associations through drug-ADR network topological features [Manzi and Reis (2011)]. An external link concept was also proposed to infer associations in a heterogeneous network by connecting drugs sharing common ADRs [Ji *et al.* (2011) and Lin *et al.* (2013)]. Similarity-based approaches involve predicting drug side effects by combining canonical correlation analysis with network-based diffusion [Atias and Sharan (2011)], identifying significant correlations between chemical fragments and side effects for ADR prediction [Pauwels *et al.* (2011)], and using a naive Bayesian model to infer drug-ADR associations based on known drug-protein and drug-ADR associations [Xiang *et al.* (2015)]. These diverse methodologies illustrate the multifaceted nature of ADR prediction, leveraging a broad spectrum of data sources and analytical techniques to enhance the safety and efficacy of pharmaceutical treatments.

3. Proposed work

We propose two models; the first model is to construct an automatically annotated relational corpus by extracting entities of interest and the causal relationship between them. In this model, we propose an algorithm and an extended version of the triplet structure. The second proposed model is a causal proximity based multi-head attention relational graph neural network that enhances biomedical knowledge discovery. This model is trained on the constructed corpus. The proposed work section is divided into three parts: the construction of the corpus, the development of the GNN-based CPRMAM knowledge discovery model, and the training of the CPRMAM model with the corresponding joint probability density function.

3.1. Automatic annotation model

This sub-section provides a brief overview of the workflow illustrated in Figure 1, that outlines the process for automatic construction of the relational corpus and the steps taken to annotate the same. The proposed algorithm begins by focusing on coreference resolution within raw textual data derived from case reports that have been pre-processed. Once coreference resolution is achieved, entities are carefully extracted and annotated, followed by a normalization process.

Using the extracted entities along with grammatical dependencies and linguistic properties, rules for identifying complex and causal relationships are systematically developed. Subsequently, these relationships are extracted and organized into a novel triplet structure based on the established rules. The generation of novel triplets involves assigning relevant properties to the entities and establishing connections that link triplets that occur in close contextual proximity.

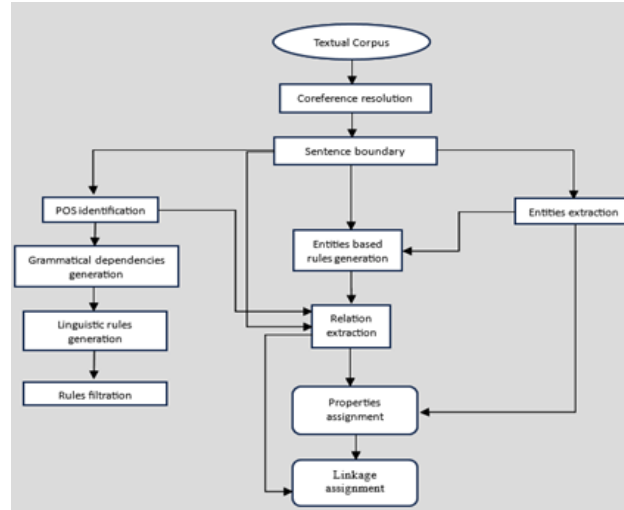


Figure 1: Workflow of first model

3.1.1. Raw corpus collection

Case reports, acknowledged for their ability to capture realistic and practical knowledge, serve as the primary data source for the constructed corpus. The curation process

involved the selection and gathering of case reports from PubMed by executing the query, with the overarching objective of semantically capturing pertinent and realistic medical data [Dev and Sharan (2023b)]. PubMed executed query is responsible for collecting only relevant case reports (cancer specific case reports).

3.1.2. Coreference resolution

The objective of Coreference Resolution is to effectively cluster expressions denoting the same entity within a document, thereby mitigating textual ambiguity. To achieve this, SpanBERT, a self-supervised method meticulously designed to optimize the prediction of text spans, is employed for coreference resolution. SpanBERT adopts a training approach where a single contiguous segment of text is sampled for each training example. Figure 3 elucidates the architectural representation of SpanBERT. Figure 2a and 2b exemplify an instance of both raw and coreferenced corpus, explaining the imperative need for such a resolution in comprehensive linguistic analysis. Figures 2c and 2d, respectively, depict the extracted pattern with and without including the SpanBERT algorithm. In the absence of coreference resolution using SpanBE, the extraction of relations from a paragraph would follow the pattern depicted in Figure 2c. Highlighting the significance of coreference, once coreference resolution is applied, it becomes evident that without this process, the relationship highlighted in Figure 2d would remain uncaptured. Coreference resolution plays a pivotal role in ensuring complete latent semantics encapsulation.

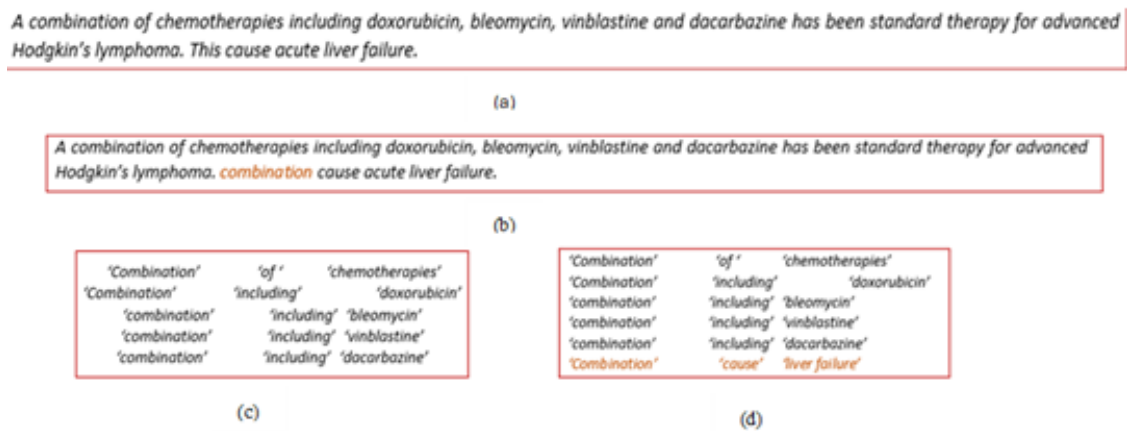


Figure 2: Instance of coreference resolution

3.1.3. Entity extraction

Entity extraction procedure is compartmentalized into two segments: the extraction of nine primary entities and the seventeen secondary entities delineated in Table 1. Primary entities represent the key entities considered as nodes (subjects, or objects) while secondary entities encompass properties assigned to the nodes and relationships. Primary entities include the following categories: drugs, adverse drug reactions, cancer, and dysn. These entities were extracted using METAMAP and the variant settlement algorithm [Dev and Sharan (2023a)]. The extraction of the remaining entities was accomplished using clinical BERT [Alsentzer *et al.* (2019)], a model trained through the integration of three datasets: BC2GM, Jnlpba, and Maccrobat. Earlier, BERT models did not produce the desired results

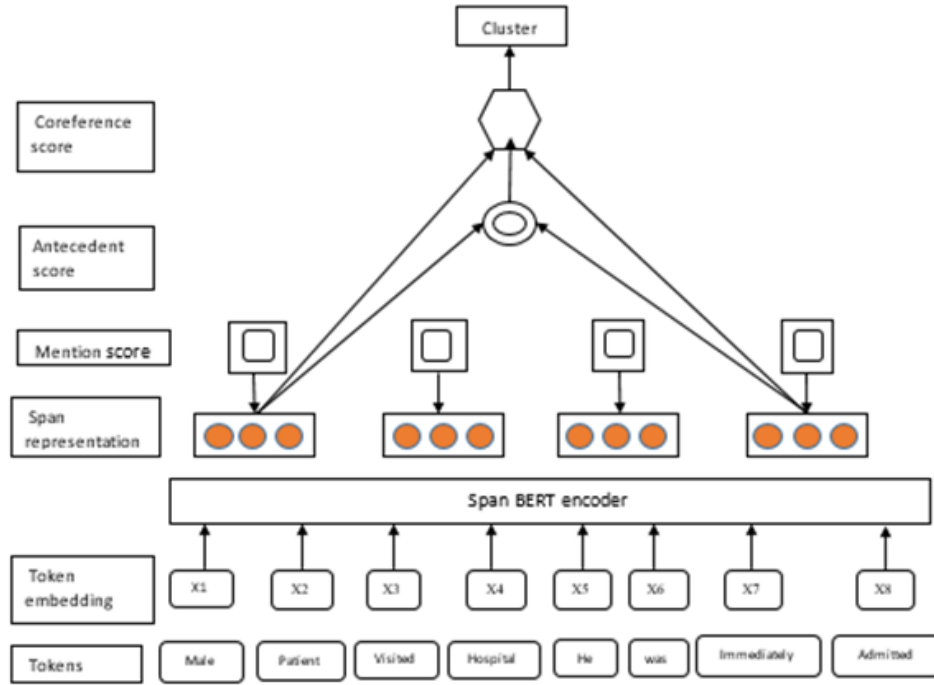


Figure 3: Span BERT architecture

in the medical domain because they were not pre-trained on medical data. This led to the realization that a model needed to be trained on a medical corpus, resulting in the development of medical domain specific BERT models such as MedBERT and ClinicalBERT. Clinical BERT is pre-trained on clinical reports and electronic health records making it suitable for extraction of secondary entities. Clinical BERT enhances the accuracy of secondary entity extraction, as clinical reports contain complex sentences filled with medical jargon and rare terms.

In Clinical BERT, input sentence is represented by vector $X = \{x_1, x_2, \dots, x_N\}$, where x_i is the i -th word and N represents the length of the sentence. The Clinical BERT model aims to classify word $x_i \in X$ and assign it a corresponding label $y \in Y$ by fine-tuning the pre-trained embeddings. The model is tested using test data. The testing accuracy obtained was 0.951. Following the testing of the model, it is deployed to extract entities from the case report corpus, subsequently mapping them to their respective entity types as illustrated in Table 1. These identified entities are then employed in subsequent tasks related to relation extraction. The clinical BERT architecture is depicted in Figure 4.

3.1.4. Relation extraction

An annotated document is represented as $D = [d_1, \dots, d_m]$, words set of each document as $W = [w_1, \dots, w_n]$, set of tagged entities as $E = [e_1, \dots, e_n]$ and $R(w_i, w_j)$ describes the relation between pair of entities w_i and w_j . Extraction of relation is grounded in dependency relations and grammatical properties to delineate specific relation patterns. Dependencies between the words in a document required for relation extraction were extracted using BiLSTM based deep bi-affine neural network dependency parser. Resultant relations are subse-

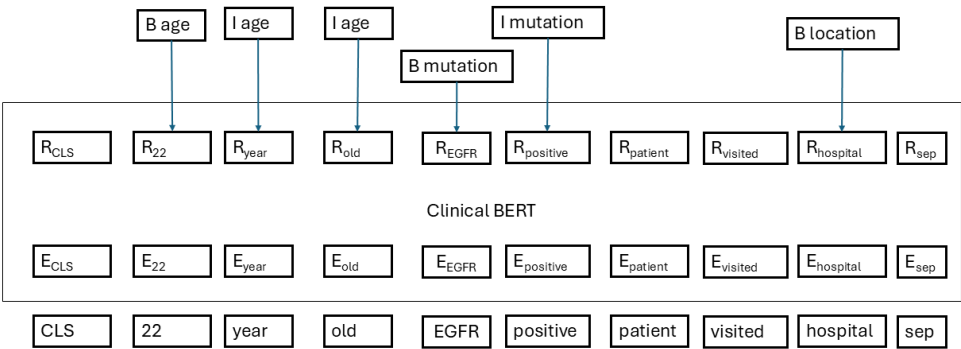


Figure 4: Workflow of first model

Table 1: Primary and secondary entities

Category	Entities
Primary Entities	Drug, Cancer type, Adverse drug reaction, Dysn, Protein, Gene, Gene mutation, Cell type
Secondary Entities	Age, Date, Detailed description, Family history, Dosage, Duration, Height, History, Non-biological location, Lab value, Biological structure, Activity, Personal background, Gender, Severity, Clinical event, Area

quently transformed into triplets, a format conveying semantics in text. It includes subject, object, and predicate. They can be either three words or three phrases. The assignment of triplet elements is contingent on their respective grammatical roles and dependencies.

This baseline triplet structure proves insufficient in capturing the semantic nuances inherent in complex sentences. To address this limitation, we introduce a novel extended triplet structure, as illustrated in Figure 5. This structure builds upon the base triplet format by incorporating properties for each triplet component and proximity links between them. By augmenting triplet elements with properties, we encapsulate detailed and vital information. Concurrently, the inclusion of proximity links establishes connectivity among proximate triplets, facilitating the extraction of comprehensive and meaningful relational data.

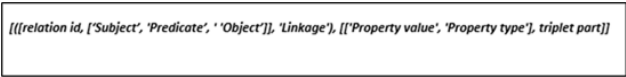


Figure 5: Proposed triplet format

For example, consider a sentence of a case report. ‘Severe rashes were caused due to bleomycin during cycle 1 treatment of metastasis breast cancer’. In this sentence, an extracted relation triplet was

[‘Rashes’, ‘caused due to’, ‘bleomycin’].

But in this triplet format, most of the information is lost, such as:

- Rashes are severe
- Bleomycin was given during breast cancer cycle 1 treatment
- Breast cancer was metastasis

The introduced novel extended triplet format serves as a robust solution to circumvent this constraint. We showcase in the given example how the relational data was extracted and significantly enriched through the utilization of our innovative triplet structure.

(1, ['rashes', 'caused due to', 'bleomycin'], [2, 3], [severe, severity, subject])

(2, ['bleomycin', 'during', 'treatment'], [3], [cycle 2, process, object])

(3, ['treatment', 'of', 'breast cancer'], [Metastasis, characteristic, object])

For instance, within Sentence 1, the individual triplets, each denoted by IDs 1, 2, and 3, may initially appear devoid of substantial significance. However, when considered collectively and augmented with incorporated properties and proximity links, they synergistically yield a profound, comprehensive, and contextually rich array of relational information.

3.1.5. Relation categorization

Relations extracted in the form of triplets can be classified into two distinct types: regular relations and causal relations. Figure 6 visually represents the categorization of the extracted relation types.

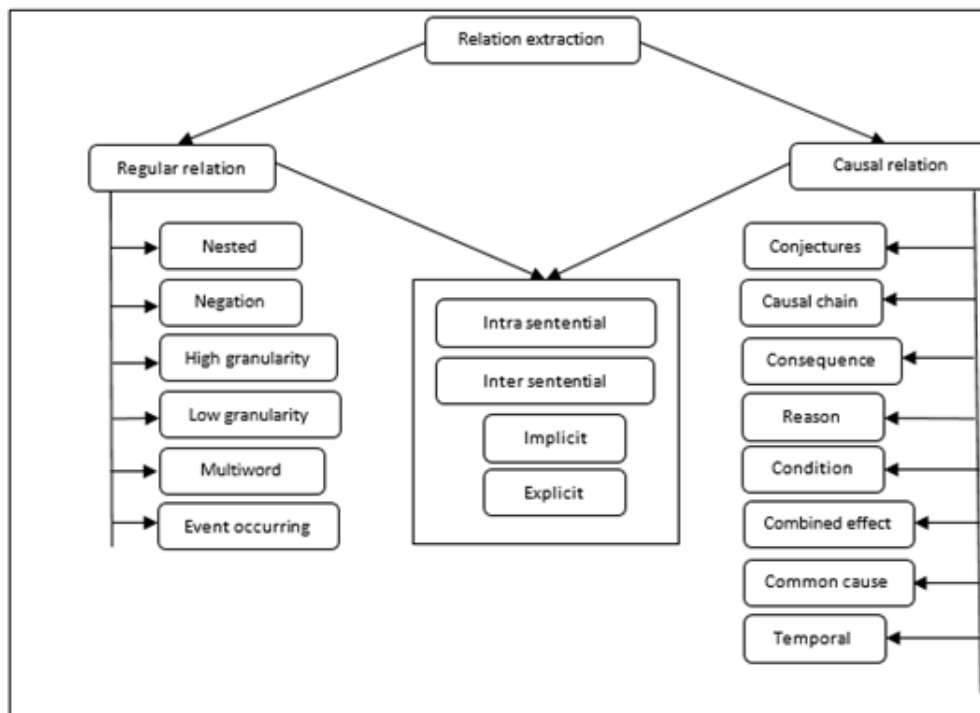


Figure 6: Relations classification

Regular relation: Regular relations are typically based on various semantic associations between entities. These associations can be diversely based on biomedical information.

These relations can be further divided into binary, complex relations, negation, conjectures, implicit relations, explicit relations, event-occurring relations, high granularity relations, nested relations, and multiword relations.

Causal relations: Causal relations are typically more complex and explicitly concerned with cause-and-effect connections and focus on understanding the factors or events that lead to a specific outcome or result. Causality or causation is a concept that expresses how one variable contributes to the creation of another variable [Szarfman *et al.* (2002)]. Causal relations were further classified into intra-sentential relations, inter-sentential relations, causal chains, consequence, reason, condition, common cause, combined effect, common effect, and action. Extracting these causal relations from medical literature is integral for constructing a knowledge graph. Such a graph serves to assist healthcare professionals in swiftly identifying causality patterns, such as diseases causing symptoms, diseases leading to complications, treatments improving conditions, and ultimately tailoring personalized treatment plans.

3.2. CPRMAM model development

This section outlines the development of the second proposed model, the Causal Proximity-based Multihead Attention Relational Graph Neural Network (CPRMAM), that is trained on triplets in the proposed triplet structure extracted by the first model. CPRMAM is designed to predict adverse drug reactions (ADRs) specific to cancer treatment drugs as a case study. To enhance message passing and mitigate over-smoothing, we introduce a novel aggregation function incorporating a proximity vector. Our approach aims to improve GNN performance in heterogeneous graph learning by effectively capturing comprehensive node roles and relationships. Each subsection in this section defines a step in the process, presented sequentially, to provide a clear and structured overview of the model development workflow. The proposed model is elucidated in Figure 7 and an algorithm for the proposed model is described in Algorithm 1.

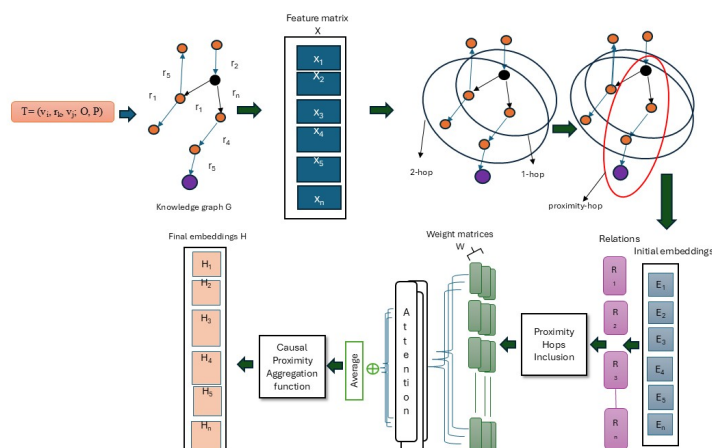


Figure 7: Proposed CPRMAM model

Figure 7 illustrates the Causal Proximity-based Multihead Attention Relational Graph Neural Network (CPRMAM) framework, designed for ADR prediction in oncology by lever-

aging relational information and proximity-aware aggregation. The process begins with a heterogeneous knowledge graph representation of triplets, where nodes (such as drugs, ADRs, and genes) are connected through different relationships. Mathematically, given a set of triplets T , the feature matrix is generated as: $X^{|n| \times d}$ where d is the feature dimension and n is the node size. To enhance information propagation, proximity hops were added. A weight matrix W is applied to encode relational dependencies, followed by the multihead attention mechanism. The final embeddings were generated using the proposed aggregation function.

3.2.1. Initial node embedding generation

The efficacy of graph machine learning tasks, such as link prediction, hinges upon the acquisition of a useful feature representation for nodes within the graph. Conventionally, node embedding generation methods rooted in structural equivalence have predominated in literature. However, these methods suffer from two primary drawbacks. Firstly, they are inherently transductive, thereby impeding their generalization to unseen nodes. Secondly, there exists no standardized approach for seamlessly integrating node attributes into the network representation. Consequently, in these methods, proximity between nodes fails to guarantee semantic similarity.

To circumvent these challenges, we propose an extension to the role2vec [Ahmed *et al.* (2018)] algorithm, designed to learn a function capable of capturing the semantic role and behavior of nodes within a graph. The basic idea of role2vec is to introduce the notion of attributed random walks that are not tied to structural similarity but tied to a function that maps a node to a role in a graph.

The goal of the proposed algorithm is to generate the embedding of a node based on role that it plays in a graph. The role of a node is defined by its features that are described in Table 2. Notion is that the nodes having similar roles must have similar embedding. It includes structural properties as well as semantic properties. Role2vec framework integrates vertex mapping and attributed random walks.

Consider a knowledge graph $G = (V, R, T)$

Where V = set of entities, R = set of relations, T = set of triplets in extended format

Now V_N and R_L can be represented as follows

$$V_N = \{v_1, v_2, v_3, \dots, v_n\}$$

$$R_L = \{r_1, r_2, r_3, \dots, r_l\}$$

Now extended triplet t can be represented as follows

$$\text{Extended triplet } t = \{(v_i, r_k, v_j; O, P); v_i, v_j \in V_N \text{ and } r_k \in R_L\}$$

Where O is the element properties set and P is the proximity link set.

$$a_{ijr_k} \in A^{N \times N} \quad (1)$$

Consider $A^{N \times N}$ as the relational adjacency matrix for G , where a_{ijr_k} defines the existence of a relation type r_k between entities v_i and v_j . Let X be the set of node attributes defining the semantic role of each node. Let $F^{N \times P}$ be the feature matrix for the set of

Algorithm 1 CPRMAM Model

Input: Extended triplets $t = (v_i, r_k, v_j; O, P)$ where $v_i, v_j \in V$; $r_k \in R$; O = element properties set and P = proximity link set

Output: Missing link triplets $M = (v_i^m, r_k^m, v_j^m)$

1. Initialize feature set for a node v
2. Generate feature matrix $F^{N \times P}$ for V

$$X_V = \{x_1, x_2, x_3, \dots, x_{N_x}\}$$

3. Calculate initial node embeddings e for each node v

$$E_d = \{e_1, e_2, e_3, \dots, e_{N_e}\}$$

4. Set initial relational weight matrix W
 5. Initialize Proximity nodes and embedding set Y
 6. For each $r_k \in R$ do
 - (a) For each $v_n \in V$ do
 - i. Initialize causal proximity aggregation function
 - ii. Calculate Multi-head attention vector for different heads
 - iii. Calculate final embedding for v based on (7) and (8)
 7. Initialize Decoder (scoring function)
 8. Initialize final embeddings for V
 9. Generate negative embeddings for V using Negative sampling
 10. Compute triplet loss and update weight W
 11. Train relational model
 12. For $t \in T$ do
 - (a) Generate corrupted triple set T^c
 - (b) Initialize final embeddings for V
 - (c) Initialize scoring function $f_{r_k}(h_i, h_j)$
 - (d) Calculate scores for testing triplet t and T^c
 - (e) Generate rank for t against T^c
 13. Evaluate model
 14. Predict missing links
-

Table 2: Node semantic features

Feature	Description
Entity_name	Name of the entity
Entity_class	Class or type of the entity
Entity_CUI	Concept Unique Identifier (CUI) of the entity
Entity_property	Properties associated with the entity
Entity_Synonyms	Synonyms of the entity
Entity_normalized version	Normalized version of the entity

entities having N entities and P features. Additionally, e_{id} denotes the initial embedding of a node v_i , generated by the $\phi(v_i)$ function using the extended Role2Vec algorithm.

$$E_d = \{e_1, e_2, e_3, \dots, e_n\} \quad \text{and} \quad \{x_1, x_2, x_3, \dots, x_k\} \in X \quad (2)$$

$$\phi(v_i) \rightarrow e_i v_i \quad (3)$$

$$\phi(v_i) = x_1^i \circ x_2^i \circ x_3^i \circ \dots \circ x_k^i \quad \text{where } \circ \text{ denotes concatenation.} \quad (4)$$

Extended Role2Vec focuses on minimizing the cosine distance d or maximizing cosine similarity s that relates a node to the nodes having similar roles using the mapping function $\mathbb{Y}$. For v_i and $v_j \in V$, which are two nodes having similar semantic roles, the goal should be:

$$\mathbb{Y}(e_i^v | e_j^v) \rightarrow \text{Max}(S(e_i^v | e_j^v)) \quad \text{or} \quad \text{Min}(d(e_i^v | e_j^v)) \quad (5)$$

$$S(v_i | v_j) = \frac{e_i^v \cdot e_j^v}{\|e_i^v\| \|e_j^v\|} \quad (6)$$

$$d(v_i | v_j) = 1 - \frac{e_i^v \cdot e_j^v}{\|e_i^v\| \|e_j^v\|} \quad (7)$$

3.2.2. Attention vector calculation

Attention weight α_{uv} is the result of the attention mechanism, which defines how much attention or weightage must be given to a neighbor u of node v . In graph convolutional methods, it is defined as $\frac{1}{|N(v)|}$, where a weighing factor for all nodes is the same in GCN. The basic graph attention network is given by:

$$h_u^{(L)} = \sigma \left(\sum_{u \in N(v)} \alpha_{uv} W^{(L)} h_u^{(L-1)} \right) \quad (8)$$

$$a_{cv_u} = Q(W^{(L)} h_u^{(L-1)}, W^{(L)} h_v^{(L-1)}) \quad (9)$$

$$\alpha_{uv} = \frac{e^{a_{cvu}}}{\sum_{k \in N(v)} e^{a_{cvk}}} \quad (10)$$

Multiple α_{uv} values with different parameters for each attention head are used in multi-head attention. These are calculated as:

$$h_u^{r(L)}[1] = \sum_{u \in N(v)} \alpha_{uv1} W^{(L)r} h_u^{(L-1)} \quad (11)$$

$$h_u^{r(L)}[2] = \sum_{u \in N(v)} \alpha_{uv2} W^{(L)r} h_u^{(L-1)} \quad (12)$$

$$h_u^{r(L)}[n] = \sum_{u \in N(v)} \alpha_{uvn} W^{(L)r} h_u^{(L-1)} \quad (13)$$

Finally, the embedding of node u after multi-head attention is obtained by aggregating the results from all attention heads:

$$H_{u,r}^{(MH^L)} = \text{AGG}(h_u^L[1], h_u^L[2], \dots, h_u^L[n]) \quad (14)$$

3.2.3. Proposed aggregation function

After the initial embedding generation, we introduce a novel aggregation function tailored for message passing and final node generation. This pioneering aggregation function is specifically designed to integrate a proximity vector, a crucial addition to address the over-smoothing dilemma pervasive in Graph Neural Networks (GNNs). Typically, GNN architectures limit the number of layers to 2-5 hops to mitigate over-smoothing. However, this restricted depth fails to capture the intricate semantics and contextual intricacies inherent in complex relations.

To overcome this limitation, we propose the inclusion of a proximity vector, which aggregates all nodes interconnected and proximate to a given relation. These proximity relations, derived from the training set produced by our novel triplet format, enrich the model's understanding by incorporating contextual information from neighboring nodes. In general, the Graph Neural Network (GNN) framework consists of layers, where each layer L includes a message function M and an aggregation function η . The message function for a node u is defined as:

$$M_L^u = mf^L(h_u^{(L-1)}) \quad (15)$$

where

$$mf^L(h_u^{(L-1)}) = W^{(L)} h_u^{(L-1)} \quad (16)$$

The proposed aggregation function is defined as:

$$\eta = H^L = ReLU \left(\sum_{r \in R} \sum_{u \in N(v)} \frac{1}{c(v, r)} H_{u,r}^{MH^L} + w_0^L h_v^{(L-1)} + \Delta \sum_{a \in Y} P_a \right) \quad (17)$$

where for v_i ,

$$Z_i = H_i^1 + H_i^2 + \dots + H_i^L \quad (18)$$

Here, H^L represents the updated embedding at layer L , and Z is the final embedding. The relational node degree is given by:

$$c_{v,r} = |N_{v,r}|, \quad N_{v,r} = \text{relational node degree} \quad (19)$$

$$P = \sum_Y Z, \quad Y = \text{set of proximity triplets} \quad (20)$$

The proximity function is defined as:

$$P_a = ReLU \left(\sum_{r \in R} \sum_{b \in N(a)} \frac{1}{c(a, r)} \alpha_{uv} w_r^L h_b^{(L-1)} + w_0^L h_a^{(L-1)} \right) \quad (21)$$

The message function is given by:

$$M = mf^L(h_u^{(L-1)}) \quad (22)$$

which can also be written as:

$$M = mf^L(H_u^L, \theta_G) \quad (23)$$

where θ_G represents the shared network parameters. The normalization function is denoted by:

$$\Delta = \text{Normalization function} \quad (24)$$

Each relation has L relational weight matrices. For a given relation r , the relational weight matrices are:

$$W_r^1, W_r^2, \dots, W_r^L \quad (25)$$

3.3. CPRMAM model training

3.3.1. Positive sampling training

We trained the model using the proposed aggregation function on the constructed ADR corpus. We set the number of hidden layers to two ($k=2$). The final embedding for each node has been encoded. We trained four distinct translational and deep learning graph

models on our cancer-specific triplet corpus. Our proposed model, the Causal Proximity Relational Multihead Attention Model (CPRMAM), was compared against four existing models: TransE, DistMult, ComplexE, and RAGAT. CPRMAM produced the best results because its encoders are trained to handle complex and indirect relationships, thereby enhancing the efficiency of knowledge discovery to generate accurate and precise results. The training data contains 132184 triplets in the proposed novel format.

3.3.2. Negative sampling training

Negative sampling is also important to reduce false positive cases. We opted 1:N negative sampling strategy [Qian *et al.* (2021)].

3.3.3. Loss computation and weight updation

We trained our CPRMAM model on two different loss functions; cross entropy loss and pairwise hinge loss. It was seen that model performed better with the cross entropy loss function. Then we applied gradient descent to update the weights.

$$\text{Cross Entropy Loss} = -\log(\text{ReLU}(f_{r_k}(h_i, h_j))) - \log(1 - \text{ReLU}(f_{r_k}(h_i, h_j))) \quad (26)$$

3.3.4. Selection of most suitable parameters

Efficient model training depends on the selection of correct parameters and avoiding overfitting and underfitting. To make sure that the model gives the best results, we used grid search to find the suitable embedding size z , learning rate lr , number of corruptions that have to be made for each triplet eta , and maximum epochs to run.

3.3.5. Triplet scoring

After training the encoder part and generation of final embeddings, the decoder plays an important role in scoring the sample triplets and generates non-existent knowledge. We used four different scoring functions with each different encoder. Table 4 shows the possible combinations of encoders and decoders, or we can say aggregation function and scoring functions. It was noticed that the proposed aggregation function as encoder and ConvE as decoder/ scoring function gave the best results.

3.3.6. Corrupt triplet generation

To test the model against false positives, we generated corrupted triplets in test data using the filter strategy. In the filter strategy, only one element of a triplet will be corrupted by replacing that element with random entities and relations. Corruption parameter eta defines the total number of corruptions that should be generated for each triplet. It was found that the most suitable value for eta is 20. Practically generating corruption for every triplet is not feasible. Therefore, we corrupted triplets that contain the relations among entities of interest: drug, ADR, cancer types, food, genes, protein, and allele.

$$t = (v_i, r_k, v_j), \text{ create corrupted triplets } t_c = (v_i, r_k, v_j^c) \quad (27)$$

3.3.7. Scoring of triplets

After generating corrupted triplets in test data, every triplet will now be scored based on scoring functions/decoder. We used four available scoring functions: TransE, DistMult, Complex, and ConvE. Score tells how likely two nodes relate to a particular relation. Each scoring function mentioned have been used in possible combination with the available algorithms and proposed model. It was noticed that ConvE performed better as shown in Table 5.

Compute the score for the triplet t using the function f that takes the relationally weighted final layer embeddings h and the transformation matrix $\odot$.

$$f_{r_k}(h_i, h_j) = f_{r_k}(W_{r_k}, h_i, W_{r_k}, h_j) \quad (28)$$

$$f_{r_k}(h_i, h_j) = ReLU\left(\text{vec}\left(ReLU\left([H_i; e_r] * \odot\right) W_r\right) H_j\right) \quad (29)$$

4. Triplet ranking and model evaluation

Since there is no benchmark dataset available for model validation, the performance of the model has been evaluated using several metrics: Mean Reciprocal Rank (MRR) [Craswell (2009)], Hits@1, Hits@3, Hits@10, and Hits@100 [Khan *et al.* (2024)], as shown in Tables 3 and 4. Each triplet in the test data is ranked against its corrupted triplet variants based on the generated score. In Table 3, the left side presents the available and proposed encoders for final embedding generation, while the top side lists the available decoders used to score triplets based on the translation method. These encoders and decoders are combined, and the model's performance is evaluated for each possible combination using the mentioned metrics. Table 4 provides the evaluation metrics for the GNN-based model and the proposed model when combined with translational and convolutional decoders. The proposed model, CPRMAM, achieved the best results when paired with the ConvE scoring function. We have used dense ranking strategy for triplet ranking. These metrics assess the overall quality of the model in ranking the correct triplet for ADR prediction, which is particularly crucial in high-stakes domains like oncology, where incorrect ranking can have severe consequences.

$$\text{rank} = COUNT(\text{corruption score} \geq \text{hypothesis score}) + 1 \quad (30)$$

The MRR (equation 31) measures the ranking quality by computing the average reciprocal rank of the first correct triplet among the predicted scores. It is defined as:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i} \quad (31)$$

where Q is the set of all queries, and rank_i represents the position of the first correct triplet for the i -th query. A higher MRR value indicates that the model ranks correct predictions closer to the top, which is critical in clinical settings where lower-ranked correct predictions may be overlooked.

The Hits@K metric (equation 32) evaluates the proportion of test samples where the correct entity appears within the top K ranked predictions. It is formulated as:

$$\text{Hits@K} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \mathbf{H}(\text{rank}_i \leq K) \quad (32)$$

where $\mathbf{H}$ is an indicator function that returns 1 if the condition inside holds, otherwise it will be 0. A higher Hits@K value signifies that the model can reliably place correct ADR predictions among the top K results, ensuring that critical drug interactions are not missed.

These metrics are particularly significant in the domain of oncology, where precise ADR prediction is essential for patient safety. A high MRR ensures that correct predictions appear early in ranked lists, making them more accessible for decision-making. Similarly, a high Hits@K ensures that crucial ADR predictions are consistently included in the top-ranked results, reducing the risk of missing life-threatening interactions. By optimizing these metrics, models can improve clinical decision support systems, leading to safer and more effective treatment strategies.

Table 3: Evaluation metrics for translational-based models

Decoder	Encoder	MRR	Hits@1	Hits@3	Hits@10
TransE	TransE	0.236	0.004	0.087	0.125
	DistMult	0.284	0.005	0.051	0.150
	ComplEx	0.297	0.008	0.101	0.189
DistMult	TransE	0.284	0.005	0.471	0.145
	DistMult	0.290	0.007	0.018	0.124
	ComplEx	0.295	0.052	0.907	0.105
ComplEx	TransE	0.297	0.008	0.101	0.189
	DistMult	0.295	0.007	0.018	0.128
	ComplEx	0.307	0.052	0.087	0.287

5. ADR related knowledge discovery

After evaluating the CPRMAM model on testing data, we try to predict all possible adverse drug reactions for all cancer related drugs that were not earlier mentioned in the training text data by predicting the missing edges in the knowledge graph. Our model predicted 200 missing links that have been ranked based on the score generated by the scoring function for each triplet. The following triplets are generated through the knowledge discovery process and were not present earlier in the corpus.

(colon cancer, undergo, galantamine)

Table 4: Evaluation metrics for GNN-based models

Decoder	Encoder	MRR	Hits@1	Hits@3	Hits@10
RAGAT	TransE	0.244	0.144	0.344	0.414
	DistMult	0.244	0.144	0.344	0.414
	ComplEx	0.234	0.154	0.354	0.398
	ConvE	0.498	0.301	0.341	0.398
CPRMAM	TransE	0.348	0.258	0.548	0.878
	DistMult	0.348	0.258	0.548	0.877
	ComplEx	0.375	0.281	0.581	0.818
	ConvE	0.658	0.538	0.621	0.695

(Ipilimumab, adverse effects, fever)

(Ipilimumab, adverse effects, increased orthostatic hypotensive activities)

Table 6 provides a knowledge discovery analysis of ADRs identified by CPRMAM and the baseline models (TransE, DistMult, and ComplEx). While all models detect common ADRs, CPRMAM uniquely incorporates critical medical contexts, such as affected body organs, age-related factors, and dose dependency key aspects for clinical relevance.

For example, baseline models identify "peeling" as an ADR of 5-Fluorouracil, but only CPRMAM specifies "peeling at palms and soles," enhancing clinical interpretability. Likewise, CPRMAM uniquely captures age-related nausea and constipation (more prevalent in patients aged 65+) and dose-dependent seizures, improving ADR specificity. These refinements enhance real-world applicability in oncology pharmacovigilance.

6. Model comparison and validation

6.1. Validation of constructed ADR corpus

To ensure the accuracy and quality of the constructed ADR corpus, we conducted validation at four stages.

First, we validated the extracted entities by normalizing and mapping them to standard medical terminologies using SNOMED CT. In the second stage, we validated the relationships between drugs and the adverse drug reactions associated with cancer and its various types, using established pharmaceutical knowledge databases: SIDER and DrugBank. Third, any relationships not found in these databases were confirmed by a domain expert. Finally, in the fourth stage, we validated the missing links generated by the CPRMAM model, again in consultation with medical experts.

6.2. Model comparison

The results obtained through the proposed model 1 were systematically compared with two state-of-the-art models, namely ScispaCy and BERT, for relation extraction. Notably, the proposed approach demonstrated a superior capability in incorporating additional semantics, evidenced by its extraction of a greater number of relations compared to the aforementioned methods. Figure 8 visually presents the relations extracted by the proposed

approach and also provides an illustration of the relations extracted by the existing methods. Proposed model 2 CPRMAM addresses the existing limitations in the literature and performs better than existing models. Different GNN algorithms were implemented in combination with different scoring functions. We applied different scoring function combinations and found proposed aggregation function with ConvE scoring function gave the best results in evaluation. Our model achieved 0.658 MRR, 0.538 Hits@1, 0.621 hits@3, 0.695 hits@10 and 0.877 Hits@100. The proposed model was compared with the state-of-the-art model; RAGAT. It was noticed that the proposed model performed better. Comparison can be seen in Table 5.

Table 5: Model comparison based on MRR and Hits@K metrics

Model	MRR	Hits@1	Hits@3	Hits@10	Hits@100
RAGAT	0.498	0.301	0.341	0.398	0.414
CPRMAM	0.658	0.538	0.621	0.695	0.877

Epidermal growth factor receptor-tyrosine kinase inhibitors (EGFR-TKIs) are a strain of small molecule inhibitors mainly used to treat metastatic non-small cell lung cancer. Their predominant adverse effect is skin toxicity, usually manifested as acneiform rash, skin fissure, xerosis, and paronychia. Severe epidermal necrosis and exfoliation rarely occur. As one of the new generation of epidermal growth factor receptor-tyrosine kinase inhibitors, AZD-9291 is claimed to have better efficacy and fewer side effects, particularly appropriate for patients with EGFR T790M mutation, a 51-year-old man who developed a large area of skin necrosis and was diagnosed with toxic epidermal necrolysis after AZD-9291 ingestion. In October 2017, a 51-year-old man, suffering from loose blisters, large-area skin peeling (nearly involving the entire body), mucosal erosion and fever, visited The First Affiliated Hospital with Nanjing Medical University. He had been diagnosed with stage IV EGFR-mutation-positive advanced NSCLC with pleural metastasis in 2016 and received treatments at a thoracic hospital regularly. After chemotherapy (pemetrexed plus cisplatin for six cycles) was completed, icotinib continued to be used till August 2017 when the drug resistance was revealed. Once EGFR T790M mutation was diagnosed, he started taking AZD-9291 with a dose of 80 mg/day to delay the disease progression. Twenty-one days later, he presented with several oral blisters and scattered erythematous maculopapular rashes, which rapidly developed into mucosal erosion and general epidermal detachment involving his entire body with exfoliation on his back skin (Figure 1A and B). When admitted into our department, he was unable to get up owing to severe pain, extreme febrility, fever, and lower limb edema."

(a) Coreferenced corpus instance

```
[[0, ('AZD', 'claimed', 'efficacy better appropriate')),
(1, ('AZD', 'appropriate', 'patients')),
(2, ('EGFR', 'mutation', 'T790M')),
(3, ('AZD', 'claimed', 'fewer side effects')),
(4, ('man', 'diagnosed', 'fewer side effects')),
(5, ('mutation', 'starved', 'body entire')),
(6, ('man', 'suffering', 'blisters')),
(7, ('man', 'suffering', 'erosion')),
(8, ('man', 'suffering', 'blisters')),
(9, ('man', 'suffering', 'peelings')),
(10, ('peelings', 'involving', 'body')),
(11, ('NSCLC', 'diagnosed', 'man')),
(12, ('man', 'received', 'treatments')),
(13, ('tyrosine kinase inhibitors', 'treat', 'lung cancer')),
(14, ('AZD', 'treat', 'lung cancer'))]
```

(b) Relations extracted through proposed approach

```
[[{'head': 'AZD-9291', 'type': 'medical condition treated', 'tail': 'non-small cell lung cancer'}, {'head': 'non-small cell lung cancer', 'type': 'drug used for treatment', 'tail': 'AZD-9291'}, {'head': 'non-small cell lung cancer', 'type': 'drug used for treatment', 'tail': 'AZD-9291'}, {'head': 'AZD-9291', 'type': 'medical condition treated', 'tail': 'non-small cell lung cancer'}]]
```

(c) Relations extracted through BERT model

```
[[{'patient', 'suffering', 'blisters'}, {'AZD-9291', 'treated', 'lung cancer'}, {'lung cancer', 'drug', 'AZD-9291'}, {'patient', 'unable to', 'get up'}]]
```

(d) Relations extracted through SciSpacy

Figure 8: Results comparison

Table 6: Analysis of adverse drug reaction related knowledge discovery with CPRMAM and other models

Drug	Adverse drug event (ADE)	CPRMAM	TransE	DistMult	CompEx	Clinical Relevance
Erlotinib	<i>Hepatorenal syndrome</i>	No	No	No	No	Rare ADR
	<i>Internal bleeding</i>	No	No	No	No	Severe ADR
5-Fluorouracil	<i>Swelling</i>	No	No	No	No	Common ADR
	<i>Palmar-plantar erythrodysesthesia</i>	No	No	No	No	Drug-specific ADR
	<i>Redness</i>	No	No	No	No	Common ADR
	<i>Peeling (palms and soles)</i>	Yes	No	No	No	Body location-enhanced ADR (unique to CPRMAM)
	<i>Blisters (skin)</i>	Yes	No	No	No	Body location-enhanced ADR (unique to CPRMAM)
Cabazitaxel	<i>Nausea (increased in 65+ years)</i>	No	No	No	No	Age-related ADR
	<i>Constipation (increased in 65+ years)</i>	No	No	No	No	Age-related ADR
Prednisone	<i>Weight gain (belly)</i>	Yes	No	No	No	Body location-enhanced ADR (unique to CPRMAM)
Doxorubicin	<i>Pain (belly)</i>	Yes	No	No	No	Location-enhanced ADR (unique to CPRMAM)
Fludarabine	<i>Seizures (with high doses)</i>	No	No	No	No	Dose-dependent ADR
	<i>Stroke</i>	No	No	No	No	Severe ADR
	<i>Heart attack</i>	No	No	No	No	Severe ADR
	<i>Blindness</i>	No	No	No	No	Rare ADR

7. Conclusion

In conclusion, this study presents two models designed to enhance adverse drug reaction (ADR) prediction in oncology, addressing the limitations in existing approaches. Our first model focuses on automated corpus construction, employing rule-based entity extraction, coreference resolution, and relation extraction to annotate entities of interest from cancer-specific case reports. By introducing a novel triplet format, it captures regular and causal relations with improved semantic clarity. The second model, CPRMAM, leverages this annotated corpus and integrates causal and proximity-based information into Graph Neural Networks (GNNs). By overcoming challenges like over-smoothing and equal neighbor weightage, CPRMAM captures complex, indirect, and nested relations, significantly enhancing the quality of knowledge graphs and message passing in GNNs. Together, these models offer a robust framework for ADR prediction and knowledge discovery, with potential applications in broader biomedical domains. Future work will focus on expanding the models' scope, incorporating additional data sources, and further enhancing their generalizability across diverse biomedical contexts.

Acknowledgements

We would like to express our heartfelt gratitude to Dr. Neera Samar, Professor of General Medicine, RNT Medical College and Hospital, Udaipur, for her invaluable expertise and validation of our findings. Her insightful guidance and expert validation have significantly contributed to the rigor and relevance of this research. We would also like to acknowledge the University Grants Commission (UGC), India, for providing financial support through the Junior Research Fellowship (JRF) under the UGC-JRF scheme. Finally, we thank the reviewers for their constructive feedback and suggestions, which have greatly enhanced the quality of this manuscript.

Conflict of interest

No financial or non-financial conflicts of interest are associated with this research work included in this article.

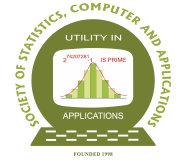
References

- Ahmed, N. K., Rossi, R., Lee, J. B., Willke, T. L., Zhou, R., Kong, X., and Eldardiry, H. (2018). Learning role-based graph embeddings. In *ArXiv*. /abs/1802.02896.
- Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jindi, D., Naumann, T., and McDermott, M. (2019). Publicly available clinical BERT embeddings. In Rumshisky, A., Roberts, K., Bethard, S., and Naumann, T., editors, *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.
- Atias, N. and Sharan, R. (2011). An algorithmic framework for predicting side effects of drugs. *Journal of Computational Biology*, **18**, 207–218.
- Bate, A., Lindquist, M., Edwards, I. R., Olsson, S., Orre, R., Lansner, A., and DeFreitas (1998). A Bayesian neural network method for adverse drug reaction signal generation. *European Journal of Clinical Pharmacology*, **54**, 315–321.

- Craswell, N. (2009). *Mean Reciprocal Rank*, pages 1703–1703. Springer US, Boston, MA.
- Dev, S. and Sharan, A. (2023a). Annotated and normalized causal relation extraction corpus for improving health informatics. In *Proceedings of ICON 2023, 20th International Conference on Natural Language Processing*. Presented at ICON 2023.
- Dev, S. and Sharan, A. (2023b). Automatic construction of named entity corpus for adverse drug reaction prediction. In *Lecture Notes in Computer Science: Innovations in Data Analytics (ICIDA)*, volume 1442. Springer.
- Dey, S., Luo, H., Fokoue, A., Hu, J., and Zhang, P. (2018). Predicting adverse drug reactions through interpretable deep learning framework. *BMC Bioinformatics*, **19 Suppl 21**, 476.
- Gao, Y., Ji, S., Zhang, T., Tiwari, P., and Marttinen, P. (2023). Contextualized graph embeddings for adverse drug event detection. In *Lecture Notes in Computer Science: Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, volume 13714. Springer.
- Ji, Y., Ying, H., Dews, P., Mansour, A., Tran, J., Miller, R. E., and Massanari, R. M. (2011). A potential causal association mining algorithm for screening adverse drug reactions in postmarketing surveillance. *IEEE Transactions on Information Technology in Biomedicine*, **15**, 366–372.
- Khan, M., Mello, G. B. M., Habib, L., Engelstad, P., and Yazidi, A. (2024). Hits-based propagation paradigm for graph neural networks. *ACM Trans. Knowl. Discov. Data*, **18**.
- Kwak, H., Lee, M., Yoon, S., Chang, J., Park, S., and Jung, K. (2020). Drug-disease graph: Predicting adverse drug reaction signals via graph neural network with clinical data. In *Advances in Knowledge Discovery and Data Mining*, volume 12085, pages 633–644. Springer.
- Lerch, M., Nowicki, P., Manlik, K., and Wirsching, G. (2015). Statistical signal detection as a routine pharmacovigilance practice: effects of periodicity and resignalling criteria on quality and workload. *Drug Safety*, **38**, 1219–1231.
- Lin, J., Kuang, Q., Li, Y., Zhang, Y., Sun, J., Ding, Z., and Li, M. (2013). Prediction of adverse drug reactions by a network based external link prediction method. *Analytical Chemistry*, **5**, 6120.
- Liu, B.-M., Gao, Y.-L., Li, F., Zheng, C.-H., and Liu, J.-X. (2024). SLGCN: Structure-enhanced line graph convolutional network for predicting drug–disease associations. *Knowledge-Based Systems*, **283**, 111187.
- Liu, R., AbdulHameed, M. D. M., Kumar, K., Yu, X., Wallqvist, A., and Reifman, J. (2017). Data-driven prediction of adverse drug reactions induced by drug-drug interactions. *BMC Pharmacology and Toxicology*, **18**, 44.
- Mantripragada, A. S., Teja, S. P., Katasani, R. R., Joshi, P., V, M., and Ramesh, R. (2021). Prediction of adverse drug reactions using drug convolutional neural networks. *Journal of Bioinformatics and Computational Biology*, **19**, 2050046.
- Manzi, S. and Reis, B. (2011). Predicting adverse drug events using pharmacological network models. *Science Translational Medicine*, **3**, 114ra127.

- Monaco, L., Melis, M., Biagi, C., Donati, M., Sapigni, E., Vaccheri, A., and Motola, D. (2017). Signal detection activity on eudravigilance data: analysis of the procedure and findings from an italian regional centre for pharmacovigilance. *Expert Opinion on Drug Safety*, **16**, 271–275.
- Pauwels, E., Stoven, V., and Yamanishi, Y. (2011). Predicting drug side-effect profiles: a chemical fragment-based approach. *BMC Bioinformatics*, **12**, 169.
- Qian, J., Li, G., Atkinson, K., and Yue, Y. (2021). Understanding negative sampling in knowledge graph embedding. *International Journal of Artificial Intelligence and Applications*, **12**, 71–81.
- Schuemie, M. J., Coloma, P. M., Straatman, H., Herings, R. M., Trifirò, G., Matthews, J. N., Prieto-Merino, D., Molokhia, M., Pedersen, L., Gini, R., *et al.* (2012). Using electronic health care records for drug safety signal detection: a comparative evaluation of statistical methods. *Medical Care*, **50**, 890–897.
- Shang, N., Xu, H., Rindflesch, T. C., and Cohen, T. (2014). Identifying plausible adverse drug reactions using knowledge extracted from the literature. *Journal of Biomedical Informatics*, **52**, 293–310.
- Szarfman, A., Machado, S. G., and O'Neill, R. T. (2002). Use of screening algorithms and computer systems to efficiently signal higher than-expected combinations of drugs and events in the US FDA's spontaneous reports database. *Drug Safety*, **25**, 381–392.
- Wang, S. V., Maro, J. C., Baro, E., Izem, R., Dashevsky, I., Rogers, J. R., Nguyen, M., Gagne, J. J., Patorno, E., Huybrechts, K. F., *et al.* (2018). Data mining for adverse drug events with a propensity score-matched tree-based scan statistic. *Epidemiology*, **29**, 895–903.
- Xiang, Y.-P., Liu, K., Cheng, X. Y., Cheng, C., Gong, F., Pan, J. B., and Ji, Z. L. (2015). Rapid assessment of adverse drug reactions by statistical solution of gene association network. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **12**, 844–850.
- Xiao, C., Zhang, P., Chaowalitwongse, W. A., Hu, J., and Wang, F. (2017). Adverse drug reaction prediction with symbolic latent dirichlet allocation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI'17)*, pages 1590–1596. AAAI Press.
- Xue, R., Liao, J., Shao, X., Han, K., Long, J., Shao, L., Ai, N., and Fan, X. (2020). Prediction of adverse drug reactions by combining biomedical tripartite network and graph representation model. *Chemical Research in Toxicology*, **33**, 202–210.
- Yildirim, P., Majnarić, L., Ekmekci, O., and Holzinger, A. (2014). Knowledge discovery of drug data on the example of adverse reaction prediction. *BMC Bioinformatics*, **15 Suppl 6**, S7.
- Zhang, F., Sun, B., Diao, X., Zhao, W., and Shu, T. (2021). Prediction of adverse drug reactions based on knowledge graph embedding. *BMC Medical Informatics and Decision Making*, **21**, 38.
- Zhao, Y., Yi, M., and Tiwari, R. C. (2018). Extended likelihood ratio test-based methods for signal detection in a drug class with application to FDA's adverse event reporting system database. *Statistical Methods in Medical Research*, **27**, 876–890.

Zheng, Y., Peng, H., Zhang, X., Zhao, Z., Yin, J., and Li, J. (2018). Predicting adverse drug reactions of combined medication from heterogeneous pharmacologic databases. *BMC Bioinformatics*, **19 Suppl 19**, 517.



Small Area Estimation Technique in Forestry Sector with Special Emphasis on National Forestry Inventory: A Review

Kamal Pandey¹, Sunil Chandra¹, Girish Chandra² and Salman Khan¹

¹*Forest Survey of India, Kaulagarh Road, Dehradun, Uttarakhand-248195*

²*Department of Statistics, University of Delhi-110007*

Received: 26 September 2024; Revised: 16 April 2025; Accepted: 20 April 2025

Abstract

The National Forest Inventory (NFI) focuses on gathering and analyzing data regarding forest resources, including tree species, sizes, age, density, and distribution, to assess forest conditions and inform management decisions. While traditional forest inventories offer detailed statistics at larger scales, they often fail to provide precise estimates for smaller areas such as forest divisions and ecologically sensitive regions. Small Area Estimation (SAE) techniques bridge this gap by employing statistical methods to enhance estimates for sub-populations with limited sample sizes. These methods incorporate auxiliary data, such as remote sensing, to enhance the precision of estimates. SAE techniques, including regression models and Bayesian methods, have proven effective in forestry and other sectors, providing robust estimates crucial for localized forest management and policy-making. Substantial theoretical work has been carried out in these techniques; however, in comparison, less work is seen in the application aspect. It is seen that SAE is being used in biomass estimation, carbon sequestration estimation, biodiversity assessment, and various other domains of research. This review highlights the application of SAE in forestry, emphasizing its potential to deliver accurate estimates for small areas and supporting informed forest management practices. Different auxiliary variables required for getting higher precision are also discussed in detail.

Key words: Auxiliary information; Domain; Forestry parameters; National Forest Inventory; Small Area Estimation.

1. Introduction

Forest inventory is the process which involves gathering and analyzing data about forests and their resources, such as tree types, growth sizes, age, density, distribution, *etc.* (Vidal *et al.* (2016); Knoke *et al.* (2021)). Forest inventory includes field surveys, followed by validating through remote sensing and other concomitant techniques, and thereafter statistical analysis to ensure accurate and detailed information about forested areas (Wulder *et al.*

(2008); Pandey *et al.* (2019)). It is done systematically to evaluate forest conditions, forest increment, volume estimation, forest dynamics, and guide forest management decisions (Tomppo *et al.* (2008a); Corona *et al.* (2011)). These assessments help in understanding the forest ecosystems, biodiversity change, carbon storage from all five carbon pools (above ground, below ground, litter, deadwood and soil), timber availability and economic evaluation (Heym *et al.* (2021)). Forest inventory provides details about the forestry statistics up to the major geographical scale, such as country, state or district level. It does not provide results at regional or small area levels such as forest division, range, compartment, section, ecological sensitive areas, *etc.* Importantly, these small areas are crucial segments of forest management, because they are the preliminary site where the management practices start. To solve this issue, researchers are utilizing different estimation techniques based on inventory and other auxiliary data of the concerned area (Breidenbach and Astrup (2012)). One such method, popularly known as the Small Area Estimation (SAE) technique, which has been recently used in various sectors including forestry. In almost all the studies, it has been proven that SAE methods provide statistically precise estimates of the variables in small domains.

SAE technique refers to statistical methods used to estimate parameters for small sub-populations or geographic areas where traditional survey methods may not provide accurate or reliable estimates due to limited or even zero sample sizes (Rao (2003); Rao and Molina (2015); ADB (2020); Sugawara and Kubokawa (2020)). However, Guldin (2021) mentioned and described Small Domain Estimation (SDE), which is a relatively similar conceptual term as SAE and further, it was suggested that SAE should be more appropriate to use over SDE due to its spatial domain. These techniques are crucial to obtain robust estimates for localized regions or domains within a larger population (Rao (2003)). In this context, SAE methodologies might find utility not only in scenarios characterized by a small geographic or temporal scope of the domain of interest with consequently limited sample sizes but also in circumstances necessitating enhanced precision in estimates across any such domain (Coulston *et al.* (2021)). In the realm of SAE, the specific level being assessed, whether it be stands, districts, or municipalities, and for which the estimation is desired, is termed a small area or domain (Breidenbach *et al.* (2018)). Small area models leverage information from both within and outside the domain of interest, as well as other supporting data, to augment the precision of parameter estimates (Lehtonen *et al.* (2003)). There are two distinct categories of SAE models exist: unit-level and area-level. Unit-level models pertain to estimators formulated at the level of sample units (such as field inventory plots in NFI), where auxiliary information is directly linked to these sample units (Rao and Molina (2015); Coulston *et al.* (2021)). Whereas, Area-level models generate estimates at aggregated geographic levels, such as administrative regions, forest range/beat/compartments *etc.*, where auxiliary information is linked to entire areas rather than individual sample units, making them suitable when unit-level data are limited or unavailable (Rao and Molina (2015)). More precisely, SAE methods typically involve combining information from both survey data and auxiliary information, such as census data or remote sensing data, to improve the precision of estimates for smaller areas (Jeong and Son (2009); Marchetti *et al.* (2015); Jiang and Rao (2020)). Recent focus on remote sensing-assisted forest inventories has shifted towards unit-level estimators for population parameter inference and associated uncertainty assessment. Numerous studies (McRoberts *et al.* (2014); Saarela *et al.* (2016); Chen *et al.* (2016); Mauro *et al.* (2016)) highlight this trend, while area-level estimators are also gaining atten-

tion (Goerndt *et al.* (2011); Magnussen *et al.* (2017)). However, only a few research reports are available for comparative analysis between unit- and area-level estimators.

SAE includes model-based estimation methods like regression models and Bayesian methods, as well as design-based methods like stratified random sampling and cluster sampling (Noble *et al.* (2002); Rao and Molina (2015)). The application of SAE, for providing reliable estimates for policy-making and decision support, has been widely seen in almost all the important sectors like public health, economics, forestry and environmental studies, demography analysis, social research, hydrological research, agricultural sector, economic analysis *etc.* (Jiang and Rao (2020); Chandra and Chandra (2015)). Within forest statistics, SAE studies focus on particular environmental data within limited geographic regions. Although national forest inventories (NFIs) provide precise assessments at larger scales, local forest officials encounter difficulties amid evolving market dynamics and financial limitations. SAE investigations act as a conduit between forest biometricians and policymakers, furnishing enhanced estimations tailored to smaller areas, thereby facilitating informed management choices (Guldin (2021)). The review serves as a crucial resource by synthesizing existing knowledge on Small Area Estimation techniques, highlighting their transformative potential in forestry research and management. By bridging the gap between traditional forest inventory methods and localized data requirements, this paper provides a roadmap for enhancing precision in small-area estimates, thereby supporting more informed policy-making, sustainable forest management, and biodiversity conservation. It also suggests the forestry researchers to utilize the SAE for generation of results based on SAE models.

This review paper is structured to provide a comprehensive exploration of SAE techniques and their significance in the forestry sector. It begins by presenting an overview of SAE methods, their classifications, and statistical approaches. It also includes detailed discussion on the diverse applications of SAE in forestry, such as biomass estimation, carbon sequestration, biodiversity assessment, and water resource management. A separate section delves into the integration of SAE techniques into national forest inventories, highlighting key methodologies, advancements, and regional case studies. Additionally, the paper examines the use of auxiliary data sources, including satellite imagery and LiDAR (Light Detection and Ranging), to enhance the precision of SAE estimates and discusses the challenges and limitations associated with these approaches. The review concludes by offering insights into future research opportunities and recommendations for the broader application of SAE in forestry and allied fields.

2. Search stratagem and data extraction

The information quoted in this review on "Forest Resource Assessment and Use of SAE Technique in Forestry Sectors" was searched and obtained using databases such as online databases subscribed, Access e- Resources like PubMed, Science Direct, Scopus, Research Gate and Google Scholar. All these resources were searched with the keyword "SAE" along with "techniques", "models", "auxiliary variables", "in forestry sectors", "in national forest inventories", "in NFI", "forest resource assessment", "Indian context" and "uses". No language restrictions were imposed. Research papers were assessed for the information about the reviews, methodologies and results and discussions on SAE and its application in forestry, of that particular study. To enhance search sensitivity, specific search techniques such as the use of quotation marks, parentheses, and asterisks were employed to identify exact terms or

expressions, as well as to capture related search terms and variations. Additionally, a manual screening of reference lists from selected studies and previous reviews was conducted to identify additional relevant publications. All the searched literature was studied thoroughly. The information from the full-text research papers and review articles of appropriate studies were extracted. Among the vast suggestions, published articles listed in the reference section were found in the databases that contain SAE techniques and their application in forestry sectors in general and the national forest inventory (NFI) in particular. The observed information in the included literature have been summarized in the present review.

3. Overview of SAE methods

SAE is an important technique for estimating the resources for smaller areas to reduce the standard error and coefficient of variations of the estimators under interest. The direct estimation of a small area yields large standard error and this eventually decreases the quality and accuracy of the estimates (Ghosh and Rao (1994)). The important factor and basis for the SAE technique is auxiliary variables. Breidenbach and Astrup (2012) mentioned that SAE techniques offer the solution to the statistical problem (low precision in estimation) if the correlated auxiliary variables are available. SAE methodologies, as delineated by Rao (2003), are typically classified into three categories: direct domain estimation, indirect domain estimation, and small area model-based domain estimation (Figure 1). Direct domain estimation focuses solely on sample data within a small area, particularly for estimating variance or mean squared error (MSE). Whereas, the indirect domain estimation objective is to enhance MSE by utilizing the estimates of the external domain through statistical modelling (Breidenbach *et al.* (2018)). Moreover, small area model-based domain estimation incorporates random effects to address unexplained variation between small areas, distinguishing between area-level and unit-level models. Area-level models, where auxiliary information pertains only to the small area level, can be advantageous in forest inventory contexts, particularly when sample plot locations are not geographically recorded (Goerndt *et al.* (2011); Breidenbach and Astrup (2012)). Auxiliary variables are the backbone of SAE and improve its precision and accuracy. In forestry, auxiliary variables can be either satellite (remote sensing) data, LiDAR (Airborne Laser Scanning) or other forestry parameters. Researchers have identified various auxiliary variables and have utilized them in estimating the characteristics of small domains. Like, Coulston *et al.* (2021) for their forest inventory and analysis program, used country-level timber products output and tree cover loss data along with SAE. Tree cover loss data was collected from global forest watch and Landsat imagery. Such timber product output summaries could be used for country and survey unit-level forest health and stand-level forest health models, aiding in understanding forest dynamics and management. Earlier, Breidenbach and Astrup (2012) and Breidenbach *et al.* (2018) in their experiment, considered the canopy height from photogrammetric models as an important auxiliary variable, revealing a strong linear correlation with forest biomass. The use of remote sensing data as an auxiliary variable is also mentioned by Goerndt *et al.* (2013) who compared different SAE models using 16 LANDSAT variables along with field data (tree density, basal area, cubic volume, quadratic mean diameter, and the average height of the top 100 trees/ha), land cover classification, tree cover, and elevation as the supportive variables. Thereafter, Breidenbach *et al.* (2018) used digital remote sensing data as an auxiliary variable in the area-level and unit-level estimate models under heteroscedasticity conditions. Airborne laser scanning is also proven auxiliary information, which enhances the quality and

eventually leads to achieving greater precision in SAE. For instance, Green *et al.* (2020) have identified and mentioned the use of LiDAR (Light Detection and Ranging) height percentile and stand thinning status as auxiliary variables in SAE of Loblolly Pine-dominated areas in the Southern US. They informed that Area-level SAE models that incorporated both LiDAR height percentiles and stand-thinning status demonstrated substantial improvements in precision, outperforming models that relied solely on LiDAR data. Conclusively, Georgakis (2019) have also mentioned the importance and utilization of auxiliary variables in SAE. Thus, it is identified that the LiDAR canopy height model and Landsat imagery are the two most used auxiliary variables in SAE for national forest inventories.

3.1. Direct estimation methods

Direct estimate utilizes summary statistics derived from a domain or small area, to predict a particular attribute of interest associated with that area. It also includes sample means, sample proportions, variance, MSEs, and the products of sample means and the size of the domain's population, assuming the population size is known (Jiang and Rao (2020)). Direct estimates do not incorporate information from other domains or external sources when making predictions for a specific area using model-based or design-based estimation (Figure 1). According to Breidenbach *et al.* (2018), direct estimate (population mean of timber volume) is derived from the sample unit of each domain of a small population unit. Considering simple random sampling within each domain, the estimator was calculated using the Horvitz-Thompson estimation method (survey sampling method). In this method, the weights are assigned to sampled units based on their inclusion probabilities to estimate population parameters, say population total (Berger (1998)). These weights ensure that each unit's contribution to the estimate reflects its representation in the population, useful for complex sampling designs.

3.2. Indirect estimation methods

The indirect estimation approach is divided into indirect domain estimation and SAE Models (Figure 1). In the indirect domain estimation approach, the researcher (knowingly or unknowingly) uses the survey data to generate estimates using design-based or model-based estimations without keeping checks on optimum sample size. Thus, the generated estimates are not precise enough because of small or zero sample sizes. Therefore, to improve the precision of estimates, researcher may use additional information from related domains or external auxiliary data sources which is possible through Small Area Estimation (SAE) models. SAE models, under the indirect estimation, are applied when there is insufficient or no direct data for a specific domain, such as a small geographic area or demographic group. It integrates domain-specific data with auxiliary sources or utilizes information from similar domains to enhance accuracy. Common methods include synthetic estimation, SAE model-based approaches, and composite estimation techniques. For, example, in estimating health statistics for a district one may merge local survey data with broader regional or national trends.

3.2.1. SAE model-based estimation methods

As already discussed, Rao and Molina (2015) and Coulston *et al.* (2021) have mentioned that there are two types of SAE models, Unit-level and Area level (Figure 1). However, according to Jiang and Lahiri (2006) and Jiang and Rao (2020), SAE models could be of three different types: area level model (Fay III and Herriot (1979)), unit level model (Battese *et al.* (1988)), and mixed logistic model (Jiang and Lahiri (2001)). The difference in classification arises due to variations in methodological perspectives and specific applications. The mixed logistic model, as mentioned by Jiang and Lahiri (2006), is often considered a specialized extension of unit-level models, particularly for categorical data and binary response variables. In contrast, studies that classify SAE models into only two types typically focus on continuous variable estimation, where unit-level and area-level models sufficiently cover most applications. Thus, the distinction is not contradictory but rather a reflection of different contexts and modeling approaches used in SAE research. In the SAE model, mathematical equations are used to predict values for different small areas having survey data. These predictions are based on some factors that can influence the outcome. The model also includes random factors that account for differences that can't be explained with predictors. Sometimes, predictions contain errors which are independent of random factors. Although exact values for all errors are not available, they can be estimated accurately. These estimations help to improve the reliability and precision of the prediction (Jiang and Rao (2020)). The assumptions of these models are considered robust as they completely validate the data distribution. This enables inference methods like maximum likelihood (ML) or restricted maximum likelihood (REML) (Jiang and Nguyen (2021)). However, when these assumptions are violated, REML may not be reliable. Alternatives to ML or REML, such as consistent estimators of variance components (Prasad and Rao (1990)), can be used for computing the empirical best linear unbiased predictor (EBLUP). Nonetheless, Jiang and Rao (2020) mentioned that measures of uncertainty are more affected by distributional assumptions. Breidenbach and Astrup (2012) presented theoretical and empirical evidence supporting the superiority of the EBLUP estimator over simple random and generalized regression models. They mentioned that the EBLUP estimator adjusts the bias correction factor based on model variance and the number of observations within the domain, enhancing its accuracy compared to the generalized regression estimator. Also, the EBLUP estimator consistently exhibits smaller mean square errors than both the simple random sampling and generalized regression estimators, especially in domains with few sample plots. However, the validity of the EBLUP estimator more rely on the robust mixed-effects model.

Various researchers in forestry have worked on SAE and predicted the different parameters. They utilized different models supported with the required auxiliary variables and forest inventory data. For example, in Mexico, Reich and Aguirre-Bravo (2009) estimated the reliability and accuracy of synthetic and regression estimators by SAE estimation of forest stand. For their experiment, the modelling process involved using multiple linear regressions to capture large-scale spatial variation, while a tree-based stratified design was adopted to address small-scale variability linked to site-specific differences in forest stand structure. Various independent variables, including Landsat-7 *ETM+* bands, and climatic, and topographic data, were considered for the models. Later, the use of model-based approaches along with NFI by different countries rises following 2016 for smaller spatial estimates (provinces, forest compartments and cities) of forestry parameters (Guldin (2021)). As depicted by Breidenbach *et al.* (2018), the nested-error linking model was used for unit-

level estimation and the mixed-effects model for area-level estimation. In their study, for unit-level modelling, the mixed effects model focusing timber volume per hectare calculated on sample plots. The model included the mean height observed on a sample plot and its square as explanatory variables, resulting in three different parameters. Also, REML was employed for parameter estimations, to depict the variations. The selection of the constant was optimized using Akaike's information criterion and Breusch-Pagan tests (Breusch and Pagan (1979)). For the area-level model, the response variable was the direct mean timber volume estimate of sample plots, while the sole explanatory variable was the stand-level mean of aerial photogrammetry mean height observed on all grid cells having parameter estimation using REML (Breidenbach *et al.* (2018)). They found that both the estimators (unit level and area level EBLUP) were indicating similar estimates with some differences in their standard errors.

Area-level and unit-level SAE models using relative error ratios were compared by Green *et al.* (2020) to enhance the accuracy of forest characteristic assessments in pine plantations. They discovered that area-level models incorporated with LiDAR height percentiles and stand thinning status were showing significant precision gains, especially when both types of auxiliary information were used. Whereas, unit-level models offer higher precision but have few limitations due to the complexity and the lack of insufficiency of precise spatial data. Furthermore, Guldin (2021) observed that in Switzerland's NFI, the design-based model-assisted approach is being employed for SAE, relying on probability samples for validity. Usually, a model-assisted method involves implementing a model to support estimation following after probability sampling. This approach ensures unbiased estimators of population parameters. Apart from the model-assisted approach, researchers compared different inferences in SAE. In particular, Brewer (2013) has compared design-based inference with randomization-based inference, contrasting it with model-based inference, which was associated with prediction-based inference. Gregoire *et al.* (2016) exemplified this with a forest inventory scenario, demonstrating how statistical regression can link airborne LiDAR height measurements to estimates of forest biomass. However, Pulkkinen and Zell (2019) mentioned that the Swiss NFI prefer the design-based and model-assisted approach as a method in SAE. Subsequently, Coulston *et al.* (2021) in their experiment, they examined different variables in forest health and stand-level forest health models, to estimate removal volumes categorized by total, species group, and merchantability class. The models integrated relevant connections between auxiliary data and the parameters under the experiment. Their experiment revealed that simpler models encountered fewer convergence challenges, with certain models relying on either a single explanatory variable or a combination of two variables and their interaction. It was also reported that their methodologies resulted in the development of 168 unique model parameterizations spanning diverse variables, methodologies, time frames, and spatial scopes in SAE. Conclusively it was advised that SAE techniques can meaningfully increase the temporal and spatial resolution of removal estimates, benefiting broad-scale inventory programs.

In model-based approaches, the regression models are the most used comprised of linear, generalized linear models, *etc.* Many researchers have contributed to developing regression models to be utilized in forestry. Figure 1 depicts different approaches of small area estimation and various tools and models for forestry application. Some other models are also paved their utilization is SAE such as the logistic mixed model which can be useful to improve the accuracy of small area proportion estimation. Authors like, Jeong and Son

(2009) introduced and described the logistics mixed model for the estimation of the small area. They considered the best linear unbiased predictor (BLUP) for the experiment. Parametric bootstrap and linear approximation were equated using the Monte Carlo approach to study the random variation across small areas. They found that linear approximation overestimated the Mean Squared Error (MSE) in comparison to bootstrap. Additionally, Reich and Aguirre-Bravo (2009) assessed the use of synthetic and regression methods in small-area analyses for estimating forest stand characteristics by taking forest inventory as auxiliary data in Jalisco, Mexico. Also, it was reported that the regression method, leveraging spatial models, showed better performance, with the accuracy of estimates heavily dependent on the spatial resolution and correlation of the models with the variable of interest. Moreover, Breidenbach and Astrup (2012) employed SAE in the forestry sector (NFI) in Norway by utilizing the photogrammetric model as an important auxiliary variable. They utilized the generalized regression estimation model for SAE and compared the outcome with simple random sampling and EBLUP estimator models. Also, the goodness-of-fit for the models was assessed through the root mean squared deviation.

Many researchers like Guldin (2021) provided a detailed description of the application of different models in the forestry sector of different countries and found that where there was not enough information from field surveys, countries relied on imputation models which combined various spatial datasets with the available field data, to create maps and calculate local estimates. It was also reported that few countries use a variety of techniques for these predictions. Some used supervised methods like maximum likelihood, discriminant analysis, and various types of regression. Others used nonparametric methods like k-NN and unsupervised approaches such as neural networks or clustering methods.

Researchers have also provided different models for SAE estimation due to its dependence on probability samples and statistical models for unbiased estimation. For example, the sampling in the Switzerland forest inventory program, had two phases: randomly selecting sample points and collecting auxiliary information, then using a simple random sample from these points to locate field plots for data collection. Further, the Swiss NFI follows this design-based Monte Carlo approach for sampling, with the fifth cycle continuing systematic measurements across the country (Pulkkinen and Zell (2019)). Similarly, Guldin (2021) hinted at new design-based SAE model-assisted estimators developed by incorporating non-exhaustive auxiliary data and different linear models. Thus, these estimators can compute the averages of model predictions using auxiliary data points and residuals from field plots within the area. In Norway, most of the research on SAE has been done with a special focus on national forest inventory *e.g.* Breidenbach and Astrup (2012) employed simple random sampling for SAE of sub-population in NFI of Norway. Conclusively, design-based models are important for the SAE point of view because simple random sampling is a type of direct estimator and utilizes the data of sample plots falling within the area of interest. Despite, various researchers having carried out growing stock estimation using remote sensing and GIS, still application of SAE in forestry research is found limited. The efforts are made to identify and provide a detailed review of various research on applications of the SAE technique in the forestry sector in general and NFI in particular.

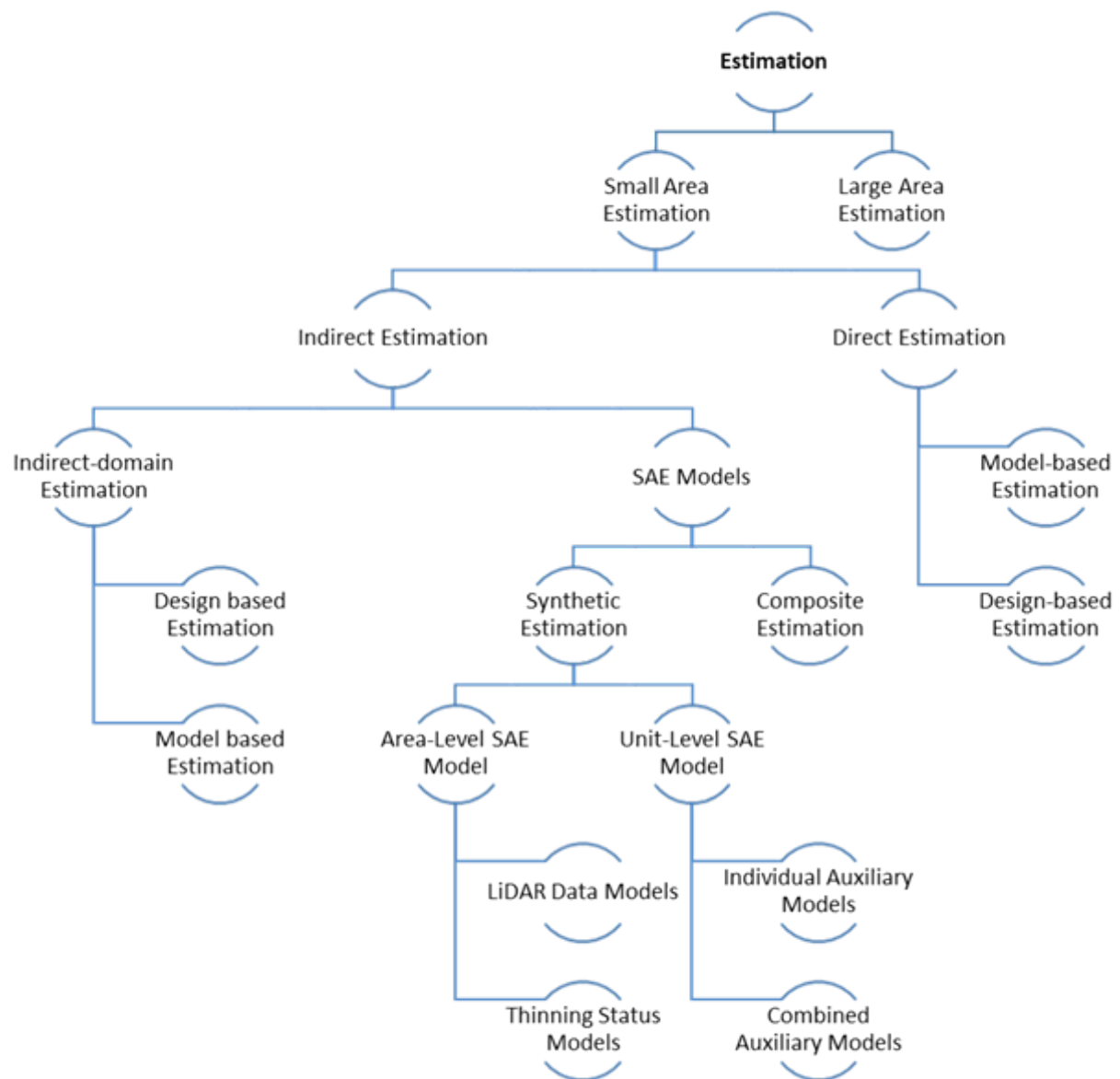


Figure 1: Small area estimation approaches and their utilization in forestry

4. Utilization of SAE techniques

Following section provides detail on application of small area estimation approaches in various sector including forestry, natural resource estimation (biomass, carbon sequestration, biodiversity assessment, water resource assessment), national forest inventories and others.

4.1. Forestry sector

SAE techniques in forestry utilize a combination of data sources, including satellite imagery, remote sensing data, and ground-based surveys, to produce detailed estimates of forest attributes such as tree density, biomass, and carbon sequestration rates at finer spatial resolutions (McRoberts (2012); Chandra and Chandra (2015)). These data sources are applied in the unit-level or area-level SAE models as auxiliary variables or additional information to generate precise estimates (Figure 1).

SAE is being used in various domains but in forestry, it has great importance as it reduces the cost and time. Guldin (2021) discussed that despite, forest inventories providing credible estimates at the national level; their spatial sampling often lacks the precision required for smaller scales or in areas with sparse or unevenly distributed forest resources (Chandra and Chandra (2015)). Worldwide, forest officials resort to auxiliary datasets like digital aerial photography to enhance inventory data accuracy through imputation models. These models employ parametric and non-parametric approaches and assign values to pixels corresponding to forest attributes. Subsequently, these imputed values are aggregated within defined small areas, employing algorithms to generate SAE estimates with acceptable accuracy (Goerndt *et al.* (2011); Breidenbach *et al.* (2018)). This bridging of statistical methods with practical forestry aims to facilitate sustainable forest management practices, support biodiversity conservation efforts, and assess the impacts of climate change on forest ecosystems. Additionally, SAE enables policymakers and stakeholders to make evidence-based decisions by providing accurate information on forest resources and their spatial distribution (Guldin (2021)).

Researchers started utilizing SAE in the estimation of forestry parameters. The majority of applications have concentrated on unit-level models (Coulston *et al.* (2021)). For instance, McRoberts (2012) employed unit-level methodologies to refine the precision of volume per acre estimates, leveraging Landsat Thematic Mapper imagery as auxiliary data. On a similar pattern, Mauro *et al.* (2017) incorporated LiDAR-based auxiliary information in their SAE approach, where LiDAR-derived metrics such as mean canopy height and biomass density were aggregated at the regional level. These variables served as key auxiliary data in area-level models, aligning with the classification in Figure (1), which places "LiDAR Data Models" under area-level SAE approaches. Similarly, Goerndt *et al.* (2013) utilized Landsat-derived spatial variables, such as land cover and tree cover, which, when aggregated over predefined areas, enhance area-level SAE predictions.

Likewise, Breidenbach and Astrup (2012) investigated the utilization of photogrammetric canopy heights within a unit-level approach to enhance the precision level of mean canopy height estimations. Their study compared unit-level and area-level SAE approaches, proposing methodologies to handle the inherent heteroscedasticity in forest parameters using photogrammetric methods.

On a similar trend, Reich and Aguirre-Bravo (2009) explored the efficacy of synthetic and regression methods for estimating forest stand characteristics in small geographic regions within Jalisco, Mexico. They interpret that despite, the synthetic method being user-friendly, its accuracy is contingent on meeting specific model assumptions. Whereas, the regression method employs spatial models, and offers improved accuracy and precision, though reliant on model quality and spatial resolution. They highlighted the importance of valid and precise data for informed decision-making in ecosystem sustainability. They concluded that the regression-based approach demonstrates notable improvements over synthetic methods, offering consistent estimates across spatial scales and aiding decision-making processes for resource management. Moreover, Goerndt *et al.* (2013) compared several SAE estimators in 12 counties of the northern Oregon Coast range. Multiple linear regression (MLR), gradient nearest neighbor imputation (GNN), most similar neighbor imputation (MSN), and five composite estimators combining MLR, MSN, and GNN with county-level direct estimates were tested and compared. The supportive data related to forest inventory such as tree density, basal area, cubic volume, quadratic mean diameter, and the average height of the top one hundred trees per hectare belonging to about 680 forest inventory plots were considered for testing with the SAE model. Besides, the tree characteristics, 16 Landsat variables, land cover classification, tree cover, and elevation were also utilized as auxiliary variables. They found that the composite estimators were better than any other method, by offering a high level of precision and minimum bias.

Thumaty *et al.* (2016) estimated Above Ground Biomass (AGB) for central Indian deciduous forests in Madhya Pradesh, India using 2010 ALOS-PALSAR L-band data and field-based AGB estimates through empirical models. Data from a survey of 415 sampling plots (0.1 ha each) collected in 2009-10 were used as auxiliary variables. They found that plot-level AGB estimates were modelled with PALSAR backscatter data, showing the HV backscatter relation ($R^2 = 0.51$) with field-based AGB estimates. Their study estimated the total AGB (AGB) for Madhya Pradesh forests to be substantial, with validation results showing a low root mean square error (RMSE), indicating robust model accuracy. Though, their experiment did not fully satisfy the SAE approach they have a similar model which is equally important. Aardt *et al.* (2006) worked on SAE for predicting timber volume under heteroscedasticity conditions using satellite imagery. Reich and Aguirre-Bravo (2009) investigated the reliability and accuracy of synthetic and regression estimators in small-area analyses, using forestry data from Jalisco's state of Western Mexico. Chandra and Chandra (2020) estimated total basal cover of trees, herbs and shrubs using SAE methods.

4.2. SAE in natural resource estimation

Effective management and planning of natural resources often necessitate precise estimates for small geographic regions. SAE methods have become increasingly important for delivering reliable estimates in these contexts, especially when direct survey estimates are inadequate due to limited sample sizes. SAE techniques have been effectively utilized in multiple areas of natural resource estimation such as:

4.2.1. Biomass estimation

Biomass is an integral part of any natural system including forests, oceans *etc.* Researchers have tried to estimate the biomass in forestry using SAE and successfully assessed

the precise and accurate biomass. The choice of SAE model depends on whether auxiliary information is available at the unit level (*e.g.*, individual sample plots) or the area level (*e.g.*, forest compartments, districts). Unit-Level Model (Nested-Error Regression Model or EBLUP) or Area-Level Model (Fay III and Herriot (1979)) may be utilized for biomass estimation (Figure 1). For illustration, Breidenbach *et al.* (2018) estimated the forest biomass of different areas in Norway. For this, they considered the average height of the tree canopy (acquired from the canopy height model) as an auxiliary variable. Further, the same was compared with three different SAE techniques, *viz.* simple random sample mean, generalized regression and unit level EBLUP. They concluded that the SAE estimation of biomass through the SAE technique by unit-level EBLUP estimator was significantly more precise than the other three estimators. Similarly, Green *et al.* (2020) estimated the biomass of Loblolly pine in USA using the SAE model at both area level and unit level and found that auxiliary variable LiDAR and stand thinning status can generate precise estimates of the forest biomass and volume. In the USA, Gaines III and Affleck (2021) suggested the SAE model for postfire forest regeneration. However, the MSE estimates exhibited frequent negativity and variability, which suggests the limited operational utility of this model.

4.2.2. Carbon sequestration estimation

Utilization of SAE in carbon sequestration was reported by Guldin (2021) who reviewed and informed that SAE application in forestry at the global level has increased since 2010. Carbon sequestration can also be expressed in terms of AGB (Above Ground Biomass). It was reported that the estimation of AGB volume has gained significant attention in SAEs, surpassing the earlier focus on timber volume estimations. This rise in SAE application in carbon assessment can be attributed to recent global policies like greenhouse gas reporting and REDD+ (Reducing Emission from Deforestation and Degradation+), which emphasize the importance of monitoring forest carbon stocks and fluxes (Bhattarai *et al.* (2015)). Additionally, there is growing interest from both public and governmental sectors in evaluating the capacity of forests to sequester carbon at both national and regional scales. In SAE models, information such as soil samples, land use data, and environmental variables are incorporated to estimate the soil carbon content for small areas of interest (Prisley *et al.* (2021) Stanke *et al.* (2022)). As described in Figure (1) both unit-level and area-level model may be employed for carbon sequestration estimation through LiDAR models or auxiliary models (Battese *et al.* (1988)).

4.2.3. Biodiversity assessment

Biodiversity includes the variety of life forms within the ecosystems, which is very essential for sustainable ecosystem functioning and resilience. SAE methods overcome the limitation of a small area by integrating information from various origins and accounting for spatial variability, thus allowing for more accurate biodiversity estimates at smaller resolutions. Researchers such as Packalén and Maltamo (2007) mentioned the use of different canopy height models for the identification of different forest species through satellite imagery; Breidenbach *et al.* (2018) highlighted that these imageries can be utilized in biodiversity assessment in small areas of a survey unit. Prisley *et al.* (2021) highlighted that SAE techniques such as machine learning, spatial interpolation and hierarchical Bayesian models are useful in various fields of biodiversity such as species distributional modelling, commu-

nity composition estimations, rare species and endemic assessments. Both Fay-Herriot Model (unit-level) and Nested-error regression model (area-level) may be utilized for biodiversity assessment if species richness, abundance, frequencies and other variables are available as auxiliary variable (Figure 1).

4.2.4. Water resource assessment

SAE can also be utilized in estimating groundwater levels and surface water availability by integrating survey data with hydrological models and remote sensing data (Ferreira *et al.* (2022)). By leveraging the Fay-Herriot model, their study provided district-specific estimates for indicators such as household hunger, access to medical supplies, and piped water availability. Earlier, Opsomer *et al.* (2003) had applied small area estimation (SAE) techniques to evaluate soil erosion within the Rathbun Lake Watershed in Iowa, USA. They utilized survey data with auxiliary information to enhance the precision of erosion estimates at the sub-watershed level. The methodology involved integrating direct survey estimates with model-based predictions, accounting for spatial correlations and measurement errors. This approach yielded more reliable and detailed erosion assessments, facilitating targeted soil conservation efforts and informed watershed management decisions.

Later, Nguyen (2023) reported the use of SAE in water hygiene and sanitation coverage in Vietnam. Study integrates survey data with auxiliary information to produce disaggregated estimates of water, sanitation, and hygiene (WASH) coverage at provincial and district levels in Vietnam. This approach involved combining data from national household surveys with additional sources, such as census data and administrative records through regression, to enhance the precision of estimates in smaller geographic areas. By applying small area estimation techniques, the study was able to construct detailed geographic maps of WASH coverage, revealing significant disparities, particularly in poorer provinces and districts, which exhibited notably lower access to safely managed sanitation and water services.

4.3. SAE in national forest inventories

National Forest Inventory (NFI) is the systematic program designed to collect, analyze, and report data on forest resources within a whole area of any specific region, state or country to meet international reporting requirements for agreements such as the Kyoto Protocol and the United Nation's Food and Agriculture Organization. NFIs also fulfil intergovernmental mandates, as seen in initiatives like the Montreal Process, the Ministerial Conference on the Protection of Forests in Europe, and COST Action E43 (Tomppo *et al.* (2008b); McRoberts (2010); Georgakis (2019)). This structure ensures that NFIs offer critical data for domestic management and international compliance, aiding sustainable forest management and adherence to global environmental commitments. As NFI is large-scale work in which different forestry parameters such as the composition of tree species, their size and age distribution, density, volume, biomass, carbon sequestration and biodiversity are measured and calculated to monitor the health and changes in forest conditions and to evaluate forest dynamics, there is greater cost and time involved in it. So, to overcome this, SAE, become a helpful technique which is an economical and time-saving method and also provides greater accuracy and precision in NFI estimates for small domains. For NFI estimates, Unit-level combined auxiliary model (if data is available at individual plots) or

composite models (if sample size is very low) or area level model (if field data is at the area level like range/compartments/beat/others) can be employed (Figure 1). Various researchers belonging to the forest services and universities worldwide have developed precise estimates for NFI of the concerned regional forests area using SAE methodology.

In Norway, Breidenbach and Astrup (2012) estimated the mean above-ground forest biomass of the sub-population using SAE techniques by employing auxiliary variables, specifically canopy height from a photogrammetric model, correlated with other variables of interest. They recommended that the unit-level EBLUP yielded the most stable and accurate estimates.

Likewise, Næsset (2014) highlighted that operational forest management inventories now often use individual tree crown approaches and area-based approaches because both the variable of interest and the explanatory variables are accessible at the level of individual population units, such as geographically located trees or field sample plots. On the other hand, domain-level approaches have predominantly found application in research investigations, as demonstrated by studies conducted by Aardt *et al.* (2006) and Goerndt *et al.* (2011). Chandra and Chandra (2015) obtained the small area estimates for the three forestry parameters frequency, density, and total basal cover of trees, shrubs, and herbs for the state of Maharashtra in India. The auxiliary data, the percentage of forest cover at the small area level, was used for the purpose. They showed that the forest type-wise estimates of all three parameters are reliable as compared to direct estimates. Moreover, Mauro *et al.* (2017) conducted a comparison in an airborne laser scanning supported forest inventory. These findings underscore the importance of incorporating diverse auxiliary data sources in forest inventories to enhance the reliability of estimates.

Similarly, Barrett *et al.* (2016) provided an overview of the practical application of remotely sensed data in NFIs, drawing on insights from experts representing 45 countries, which collectively cover 65 percent of global forest areas. Their analysis revealed the extensive use of remotely sensed data from various sensors to refine the estimates across numerous forestry parameters in smaller areas. However, they highlighted the need for estimation through more effective integration of remotely sensed data with field inventories dataset. This comprehensive approach ensures that forest inventories remain accurate and reflective of current forest conditions. Various researchers in Spain have experimented with combining NFI and LiDAR data, yielding significant results. Condés and McRoberts (2017) focused on updating NFI-based estimates by using models to predict annual plot-level volume change and assess uncertainties. Novo-Fernández *et al.* (2019) researched to generate a fine-resolution database of forest yield variables. They described estimation methods that combined Spanish NFI data with Airborne Laser Scanning (ALS) data to predict the volume of growing stock for three main commercial tree species found in northwestern Spain. Additionally, Durante *et al.* (2019) studied the vast expanse of approx. 2.8 million acres in southwestern Spain. They integrated field plot data from the Spanish NFI, high-precision ALS, and bio-geophysical spectral variables sourced from MODIS. They explored a two-stage up-scaling approach for biomass estimation, leveraging ALS data calibrated with NFI field plots to create regional biomass maps. Esteban *et al.* (2019) described a model-assisted inference approach using random forests, comparing different bootstrap estimators and constructing change maps. In a similar experimental format during 2012, Breidenbach and Astrup (2012) continued working on NFI and reported that Norway employs permanent sample plots for

the estimation of forestry parameters, yet precise estimates for small sub-populations could not be achieved due to very limited data. In their study, they evaluated simple random sampling, generalized regression, and unit-level EBLUP for estimating above-ground forest biomass across various locations. They discovered that EBLUP produced more stable estimates in areas with numerous sample plots. Ultimately, they emphasized that increasing the sample size enhances estimation precision.

Similarly, Breidenbach *et al.* (2018) carried out a forest survey in parts of Vestfold county in southeastern Norway, utilizing complete digital aerial photogrammetry data. The survey, based on Norway's NFI methodology, aimed to estimate the mean timber volume per hectare for stands sampled using fixed-area plots. Trees were recorded, and volume was predicted from diameter and height using standard models. Stand samples were collected from areas with available forest management inventories in Norway, with explanatory variables including aerial photogrammetry heights calculated for sample plots and grid cells.

Germany is also not lagging in developing their NFI using SAE techniques. Like, Hill *et al.* (2018) mentioned a double-sampling extension for the German NFI to create design-based SAEs at the forest district level. It was suggested to utilize airborne laser scan (LiDAR) based derived canopy height models and tree species classification maps as auxiliary data with regression models to predict timber volumes (Mandallaz *et al.* (2013)). In addition to this, Wagner *et al.* (2017) conducted a study in Germany aimed at developing cost-efficient SAE methods to address the limitations of traditional forest inventories in a dense forest which require forest management information at various administrative levels. They investigated and tested different design-based regression estimators, considering laser data of varying temporal resolution. Subsequently, they found a reduction in the significant error of estimate at the district level and suggested more research is required for sub-district-level estimations. Further, the SAE method expanded to other German states for their NFI data estimates.

In Nordic countries, Kangas *et al.* (2018) reviewed the scientific advancements in NFI, focusing on the integration of remotely sensed data to boost the accuracy of NFI and Forest Management Inventory (FMI) datasets and to minimize uncertainties in parameter estimations at both national and small area levels. These FMIs are important sources to enhance the estimate precision thus, various other researchers started to utilize FMIs in their experiments. For example, Jiang and Rao (2020) explored the utilization of remotely sensed data to complement NFI and FMI datasets, aiming to improve the precision of parameter estimates at a small area level.

Furthermore, Georgakis (2019) provided a detailed review of SAE in forestry and explained that SAE methods, including synthetic estimation and the Fay-Herriot model, addressed the scarcity of terrestrial sample data and improved accuracy in the NFI. His review highlighted how these statistical techniques are crucial for enhancing the reliability of forest resource estimates, especially in regions where terrestrial data is limited.

In their study of the application of SAE in the NFI of Switzerland, Pulkkinen and Zell (2019) explored how the Swiss forest inventory utilizes a design-based Monte Carlo approach for sampling. This method involves sampling an infinite population of points within a designated region of interest to estimate spatial means of tree population totals. Field data collection employed a simple random technique. Estimation was carried out using a specific

model incorporating auxiliary data and a linear model based on probability sampling, ensuring an unbiased nature in the estimation process. Whereas, Breidenbach *et al.* (2021) utilized Sentinel-2 mosaics in conjunction with NFI data to develop models and cartographic representations of various Norwegian conifer types. These models were then leveraged to generate species-specific distribution maps tailored to smaller geographic areas, such as municipalities. Further, Astrup *et al.* (2019) detailed the integration of photogrammetric point cloud data with NFI point cloud data to produce a raster map featuring selected modelled attributes applicable to forest management inventories. The pair of studies by Haakana *et al.* (2020) investigated post-stratification as an alternative method for utilizing auxiliary information to estimate parameters for municipalities based on Finland's NFI data.

In India, Chandra and Chandra (2020) experimented with the SAE approach to generate estimates of total basal cover (m^2/ha) for trees, shrubs, and herbs in small areas of Maharashtra, India. Seven different forest types were considered as small areas for the study. Survey data (nested quadrates of $10\text{ m} \times 10\text{ m}$, $3\text{ m} \times 3\text{ m}$, and $1\text{ m} \times 1\text{ m}$ for tree, shrub, and herb layers, respectively) of the year 2011-12 were utilized as one of the important variables for the estimation. India's State of Forest Report (ISFR) depicting NFI and remote sensing-based forest cover and forest change data were utilized as auxiliary information. Their results indicated that estimates of total basal cover for trees, shrubs, and herbs, generated using the SAE approach, were reliable and precise compared to direct survey estimates.

In the Southeastern US, Coulston *et al.* (2021) in their experiment, utilized SAE methods in the NFI, considering Landsat-based tree cover change information and mill survey data as auxiliary variables. In their study, more than 35,000 permanent sample plots across forest and non-forest areas were assessed. The result suggested that improvements in the precision of estimates can be achieved when NFI data is measured at considerably fine spatial and temporal scales. They observed that the precision of estimates in tree species groups and size classes improves when SAE methods are used compared to design-based assessments. They utilized the Forest Inventory and Analysis (FIA) program's rotating panel design for data collection and followed the recommendation of Bechtold and Patterson (2005), where removal estimates were calculated based on midpoint assumptions within the re-measurement period. Parameters of interest included total removals, hardwood and softwood removals, and removals based on tree size categories.

Strîmbu *et al.* (2021) addressed inconsistency in aggregating parameter estimates for SAEs in Norway. They proposed a method to align model-based and model-assisted estimators, ensuring consistency in AGB estimates. The evaluation was conducted through simulated sampling in a 50 km^2 forest area, adjusting estimates within confidence intervals using heuristic optimization. Artificial forest populations were generated from airborne laser scanning data. Comparing adjusted AGB estimators with unadjusted ones showed minimal bias introduction, with decreased RMSE for stand-level and property-level estimators, showcasing the potential for consistency in complex estimator systems.

For NFI estimates in France, Vega *et al.* (2021) experimented with and introduced a distinct estimation algorithm aiming to balance the estimate precision in the Oak-dominated areas. The algorithm relies on NFI field data along with auxiliary information such as forest cover maps and canopy height models to achieve the desired precision in forest characteristics

estimates. However, Guldin (2021) mentioned that NFI in France is based upon a two-phase sampling strategy: Photo-interpretation of the plot for land use and land cover identification followed by drawing of a sample based on land cover. Previously, Fortin (2020) investigated the challenge of annual sampling of NFI plots and its impact on variance over the period. To address this, older plot measurements were updated by a forest growth model. However, that introduced uncertainty from both sample design and growth model and suggested to use of a forest growth model to improve the estimate precision in NFI. Likewise, Irulappa-Pillai-Vijayakumar *et al.* (2019) utilized 3D variables from photogrammetric estimated canopy height models and auxiliary attributes (forest type maps, variables related to vegetation and characteristics, satellite maps, *etc.*) to enhance French NFI data precision in the forested region. Their experiment demonstrated a significant reduction in errors, specifically in volume estimates, suggesting potential for improved forest attribute estimation in smaller areas using SAE. Dettmann *et al.* (2022) and Wilson *et al.* (2023) reviewed the extensive use of SAE in forest inventory and management, including the terminology, methods, concerns, data sources, research findings, challenges, and future opportunities. They also informed about the various methodologies such as direct, indirect, and composite estimation within design-based and model-based frameworks of SAE and are supplemented with remote sensing and geospatial data, to enhance precision in small domains, avoiding instability from small sample sizes.

All these research studies constantly show improved precision compared to direct estimates based solely on field data, highlighting the potential of SAE in national forest inventory for future research and development.

5. Data sources and challenges-remote sensing data

In SAE, the supported data and auxiliary data are important factors which determine the precision and accuracy of the model. In the case of NFI estimation, these data can be satellite imagery, airborne laser scanning (LiDAR) data or any other data. The description of these are as follows:

5.1. Satellite imagery

Satellite-borne sensor data such as LANDSAT (Land Satellite) is an important approach for integration with SAE for utilization in the small domain forestry estimation. Various other spatial data like DEM (Digital Elevation Model), soil maps, NDVI (Normalized difference Vegetation Index), watershed maps, forestry maps, geology maps, species maps, forest boundary maps, *etc.* are being considered in forestry research including SAE. These spatial data are used as auxiliary variable in SAE models to generate estimates for various theme such as biodiversity, carbon, forest volume/growing stock *etc.* (Figure 1). Pioneer work is done by Tomppo and Katila (1991) by combining satellite data with NFI data. Furthermore, Reich and Aguirre-Bravo (2009) for their experiment in Mexico's NFI utilized Landsat-7 imagery having *ETM+* along with climatic and topographical information. Different bands of satellite data were identified for specific uses *viz.* Goerndt *et al.* (2013) utilized Landsat 5 TM (Thematic Mapper) data focusing on six bands (*B1-B5, B7*) of 30×30 m spatial resolution. Normalized Difference Vegetation Index (NDVI), calculated from bands 3 and 4, and various band ratios (*B4/B3, B5/B4, B7/B5*) were used to describe forest characteristics. After identification, the imagery was processed for geometric recti-

fication, radiometric correction and several transformations, including tasselled-cap (TC) transformation, optimizing brightness, greenness, and wetness. These transformations aid in visualizing the forest structure and health in a better way. However, researchers like Hill *et al.* (2018) had taken tree species maps as auxiliary variables for their estimation in Germany's NFI. Other satellite imagery like Moderate Resolution Imaging Spectroradiometer (MODIS) is also commonly used imagery in NFI. In particular, Durante *et al.* (2019) reported the use of MODIS data for bio-geophysical spectral attributes. Likewise, Astrup *et al.* (2019) utilized photogrammetric point cloud data along with NFI data to generate raster maps depicting forestry parameters/attributes. High-resolution satellite data was also suggested to be utilized by Breidenbach *et al.* (2021)). They employed Sentinel-2 mosaics in conjunction with NFI for modelling conifers in Norway. The use of satellite data in forestry estimation was also suggested by Guldin (2021)). They mentioned the worldwide practice of utilising satellite data or aerial photography as an auxiliary variable for the estimation of timber volume, tree cover and land use change identification and classification.

5.2. LiDAR (Light Detection and Ranging) data

Earlier, Airborne Laser Scanning (ALS) data such as LiDAR was found to be strongly useful in forestry but due to higher cost and availability of free satellite data, it is less commonly used nowadays. Various LiDAR models can be utilized for SAE based estimation along with other auxiliary variable (Figure 1). In the review, Guldin (2021) informed that Switzerland is working on SAE in the forestry sector for their country and utilizing the LiDAR-based terrain model. They are utilizing the vegetation height and other forestry parameters as auxiliary variables. This system estimates various parameters for cantons, forest districts, and municipalities in Switzerland. Also, Gregoire *et al.* (2016) defined the process of regression modelling using the point cloud data of airborne LiDAR-based height measurements to estimate forest biomass. Various researchers have supported the argument that the airborne laser scans integrated with NFI data are more useful and economical when compared with the traditional inventory (Haakana *et al.* (2020); Rahlf *et al.* (2021); Strimbu *et al.* (2021)). Other important researchers include Mandallaz (2013) who tested the SAE approach based on the NFI and auxiliary data such as LiDAR and floral information. Also, Wagner *et al.* (2017) applied SAE methods along with laser data for estimating the timber in Germany's selective forest. Similar to this, Hill *et al.* (2018) utilized airborne laser scan data-based canopy height for SAE modelling. Furthermore, Durante *et al.* (2019) constructed an experiment to measure biomass through the Laser data and created a model based on it. Additionally, Novo-Fernández *et al.* (2019) estimated the yield of the plantation forest using laser data in Spain. Alongside, Breidenbach *et al.* (2018) highlighted the SAE status in Norway and focused on LiDAR-based SAE estimation. Apart from this, Green *et al.* (2020) mentioned that LiDAR data are very important as an auxiliary variable despite their age and utilized 4-5-year-old LiDAR data and projected this with current year forest inventory data. However, if the LiDAR data are of the same year of forest inventory then the result would be more precise as the chances of variation decrease significantly. Additionally, Geogarkis informed that SAE using LiDAR data shows superior results compared to those based on photogrammetry. For extensive areas like whole counties, remote sensing data from sources like Landsat, which are freely available, can provide comprehensive coverage. However, due to the high costs and sometimes limited availability of LiDAR data, its use is often restricted to smaller areas of interest, such as forest districts, where its higher accuracy can be better

justified (Goerndt (2010)). This highlights the trade-off between cost and precision when choosing appropriate data sources for SAE in different spatial contexts.

6. Some work in other sectors and future recommendations

SAE is already being considered in various sectors including agriculture, fisheries, medical, big-data analytics and others. According to Chandra and Chandra (2015), early developments in SAE for crop yield began in the late 1960s with Panse *et al.* (1966) who used a double sampling approach at the Block level. Later, Srivastava *et al.* (1999) and Sisodia and Chandra (2012) applied a synthetic method for Block-level crop estimation, utilizing extensive auxiliary information from crop-cutting experiments. Singh and Goel (2000) further refined crop yield estimation using remote sensing data. In the context of India's National Agricultural Insurance Scheme, alternative approaches for scaling down estimates to the Gram Panchayat level were explored by Sharma *et al.* (2004) and Sud *et al.* (2001)). In contrast, Sud *et al.* (2011) also applied a random effect model to account for small area dissimilarities. Beyond forestry, Srivastava *et al.* (2007) applied SAE techniques to estimate district-level economic parameters, such as loan amounts and poverty measures, in Uttar Pradesh using survey data. These studies demonstrate the precision and representativeness of SAE in various agricultural and socio-economic contexts. Additionally, Marchetti *et al.* (2015) mentioned the SAE-based estimator using big data sources. This is also validated by Kordos (2016) who advocated the utilization of big data which provides opportunities to understand complex socioeconomic phenomena such as poverty and resource distribution. When combined with SAE methods, big data offers faster access to auxiliary variables compared to traditional data sources. Despite advantages, challenges arise in ensuring data quality and representativeness, especially concerning privacy and self-selection biases.

Methodological advancements are necessary to integrate this with survey data effectively, revolutionizing scientific research and policymaking by providing precise, continuous insights into societal well-being (Rao and Molina (2015)). Using SAE, Pramanik *et al.* (2015) predicted 2011 vaccination coverage rates for 26 states not covered by the 2010-11 Annual Health Survey (AHS) in India. The model-based estimates provided almost similar results as to AHS results which ensures the reliability of the SAE method. Similarly, Jiang and Rao (2020) discussed the SAE models being utilized in the field of medical science. SAE is also being utilized in the happiness index through social media. For example, Aziz and Ubaidillah (2021) studied SAE models, EBLUP Fay-Herriot and Error Measurement, utilizing auxiliary variables from Twitter data. Conclusively, the SAE technique is an important statistical method which helps in estimating precise and accurate information for small areas and eventually reduces time and cost and increases reliability. Various models can be utilized for SAE depending upon the need and budgets. Auxiliary variables are also very important for generating precise estimates. SAE has shown enormous uses in the fields of forestry, medical science, and agriculture and its application is still under use in various other fields.

7. Conclusion

To conclude, Small Area Estimation (SAE) techniques offer significant potential to transform forestry practices by overcoming the challenges associated with traditional forest inventories. SAE provides accurate and dependable estimates at finer spatial scales, such as forest divisions or ecologically sensitive regions, which are essential for informed decision-

making in forest management and conservation. Its flexibility is evident through diverse applications, ranging from estimating biomass and carbon storage to assessing biodiversity and water resources. By incorporating advanced tools like satellite imagery, LiDAR, and remote sensing, SAE not only improves the precision of forest assessments but also offers a practical, cost-effective solution. This review brings together key developments, methodologies, and examples, shedding light on how SAE can advance sustainable forestry. By connecting theoretical advancements with real-world applications, the paper emphasizes the value of SAE as a vital tool for enhancing forest management strategies on a global scale.

Acknowledgement

The authors express their gratitude to the Director General of the Forest Survey of India (FSI), Dehradun, Uttarakhand, India, for his invaluable support and guidance. Special appreciation is extended to the Director, Forest Research Institute, Dehradun, for providing the necessary facilities and permissions during this work. The authors are also deeply indebted to Sh. S.P. Singh, IFS (Retd.), Senior Consultant at FSI Dehradun, for his unwavering support and continuous encouragement throughout the writing process.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Aardt, J. A. V., Wynne, R. H., and Oderwald, R. G. (2006). Forest volume and biomass estimation using small-footprint LiDAR-distributional parameters on a per-segment basis. *Forest Science*, **52**, 636–649.
- ADB (2020). *Introduction to Small Area Estimation Techniques: A Practical Guide for National Statistics Offices*. Asian Development Bank.
- Astrup, R., Rahlf, J., Bjørkelo, K., Debella-Gilo, M., Gjertsen, A. K., and Breidenbach, J. (2019). Forest information at multiple scales: development, evaluation and application of the norwegian forest resources map sr16. *Scandinavian Journal of Forest Research*, **34**, 484–496.
- Aziz, S. D. and Ubaidillah, A. (2021). Big data for small area estimation: Happiness index with twitter data. In *Proceedings: The International Conference on Data Science and Official Statistics*.
- Barrett, F., McRoberts, R. E., Tomppo, E., Cienciala, E., and Waser, L. T. (2016). A questionnaire-based review of the operational use of remotely sensed data by national forest inventories. *Remote Sensing of Environment*, **174**, 279–289.
- Battese, G. E., Harter, R. M., and Fuller, W. A. (1988). An error-components model for prediction of county crop areas using survey and satellite data. *Journal of the American Statistical Association*, **83**, 28–36.
- Bechtold, W. B. and Patterson, P. L. (2005). *The Enhanced Forest Inventory and Analysis Program: National Sampling Design and Estimation Procedures*. U.S. Department of Agriculture, Forest Service, Southern Research Station.

- Berger, Y. G. (1998). Rate of convergence to normal distribution for the Horvitz-Thompson estimator. *Journal of Statistical Planning and Inference*, **67**, 209–226.
- Bhattarai, T., Skutsch, M., Midmore, D., and Shrestha, H. L. (2015). Carbon measurement: an overview of forest carbon estimation methods and the role of geographical information system and remote sensing techniques for REDD+ implementation. *Journal of Forest and Livelihood*, **13**, 69–86.
- Breidenbach, J. and Astrup, R. (2012). Small area estimation of forest attributes in the Norwegian national forest inventory. *European Journal of Forest Research*, **131**, 1255–1267.
- Breidenbach, J., Magnussen, S., Rahlf, J., and Astrup, R. (2018). Unit-level and area-level small area estimation under heteroscedasticity using digital aerial photogrammetry data. *Remote Sensing of Environment*, **212**, 199–211.
- Breidenbach, J., Waser, L. T., Debella-Gilo, M., Schumacher, J., Rahlf, J., Hauglin, M., Puliti, S., and Astrup, R. (2021). National mapping and estimation of forest area by dominant tree species using Sentinel-2 data. *Canadian Journal of Forest Research*, **51**, 365–379.
- Breusch, T. S. and Pagan, A. R. (1979). A simple test for heteroscedasticity and random coefficient variation. *Econometrica*, **47**, 1287–1294.
- Brewer, K. (2013). Three controversies in the history of survey sampling. *Survey Methodology*, **39**, 249–263.
- Chandra, H. and Chandra, G. (2015). An overview of small area estimation techniques. In Chandra, G., Nautiyal, R., Chandra, H., Roychoudhury, N., and Mohammad, N., editors, *Statistics in Forestry: Methods and Applications*. TFRI Jabalpur, India.
- Chandra, H. and Chandra, G. (2020). Small area estimation for total basal cover in the state of Maharashtra in India. In G. Chandra, R. N. and Chandra, H., editors, *Statistical Methods and Applications in Forestry and Environmental Sciences*. Springer.
- Chen, Q., McRoberts, R. E., Wang, C., and Radtke, P. J. (2016). Forest aboveground biomass mapping and estimation across multiple spatial scales using model-based inference. *Remote Sensing of Environment*, **184**, 350–360.
- Condés, S. and McRoberts, R. E. (2017). Updating national forest inventory estimates of growing stock volume using hybrid inference. *Forest Ecology and Management*, **400**, 48–57.
- Corona, P., Chirici, G., McRoberts, R. E., Winter, S., and A., B. (2011). Contribution of large-scale forest inventories to biodiversity assessment and monitoring. *Forest Ecology and Management*, **262**, 2061–2069.
- Coulston, J. W., Green, P. C., Radtke, P. J., Prisley, S. P., Brooks, E. B., Thomas, V. A., Wynne, R. H., and Burkhardt, H. E. (2021). Enhancing the precision of broad-scale forestland removals estimates with small area estimation techniques. *Forestry*, **94**, 427–441.
- Dettmann, G. T., Radtke, P. J., Coulston, J. W., Green, P. C., Wilson, B. T., and Moisen, G. G. (2022). Review and synthesis of estimation strategies to meet small area needs in forest inventory. *Frontiers in Forests and Global Change*, **5**, 1–16.
- Durante, P., Martín-Alcón, S., Gil-Tena, A., Algeet, N., Tomé, J. L., Recuero, L., Palacios-Orueta, A., and Oyonarte, C. (2019). Improving aboveground forest biomass maps: from high-resolution to national scale. *Remote Sensing*, **11**, 1–20.

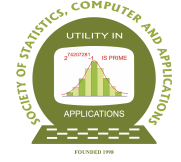
- Esteban, J., McRoberts, R. E., Fernández-Landa, A., Tomé, J. L., and Næsset, E. (2019). Estimating forest volume and biomass and their changes using random forests and remotely sensed data. *Remote Sensing*, **11**, 1–21.
- Fay III, R. E. and Herriot, R. A. (1979). Estimates of income for small places: an application of James-Stein procedures to census data. *Journal of the American Statistical Association*, **74**, 269–277.
- Ferreira, T., Stone, W., Vercuil, E., Lourens, M., Made, N., and Madonsela, T. (2022). Small area estimation for south african resource distribution and policy impacts during COVID-19. *Open Research Africa*, **5**, 1–15.
- Fortin, M. (2020). Updating plots to improve the precision of small-area estimates: the example of the Lorraine region, France. *Canadian Journal of Forest Research*, **50**, 648–658.
- Gaines III, G. C. and Affleck, D. L. (2021). Small area estimation of postfire tree density using continuous forest inventory data. *Frontiers in Forests and Global Change*, **4**, 761509.
- Georgakis, A. (2019). Small area estimation in forest inventories. In *Proceedings of the Seventh International Conference on Environmental Management, Engineering, Planning & Economics*.
- Ghosh, M. and Rao, J. N. K. (1994). Small area estimation: An appraisal. *Statistical Science*, **9**, 55–76.
- Goerndt, M. E. (2010). *Comparison and analysis of small area estimation methods for improving estimates of selected forest attributes*. PhD thesis, Oregon State University. Doctoral Dissertation, Doctor of Philosophy in Forest Resources.
- Goerndt, M. E., Monleon, V. J., and Temesgen, H. (2011). A comparison of small-area estimation techniques to estimate selected stand attributes using LiDAR-derived auxiliary variables. *Canadian Journal of Forest Research*, **41**, 1189–1201.
- Goerndt, M. E., Monleon, V. J., and Temesgen, H. (2013). Small-area estimation of county-level forest attributes using ground data and remote sensed auxiliary information. *Forest Science*, **59**, 536–548.
- Green, P. C., Burkhart, H. E., Coulston, J. W., and Radtke, P. J. (2020). A novel application of small area estimation in loblolly pine forest inventory. *Forestry: An International Journal of Forest Research*, **93**, 444–457.
- Gregoire, T. G., Næsset, E., McRoberts, R. E., Ståhl, G., Andersen, H. E., Gobakken, T., Ene, L., and Nelson, R. (2016). Statistical rigor in LiDAR-assisted estimation of aboveground forest biomass. *Remote Sensing of Environment*, **173**, 98–108.
- Guldin, R. W. (2021). A systematic review of small domain estimation research in forestry during the twenty-first century from outside the United States. *Frontiers in Forests and Global Change*, **4**, 1–15.
- Haakana, H., Heikkinen, J., Katila, M., and Kangas, A. (2020). Precision of exogenous post-stratification in small-area estimation based on a continuous national forest inventory. *Canadian Journal of Forest Research*, **50**, 359–370.
- Heym, M., Uhl, E., Moshhammer, R., Dieler, J., Stimm, K., and Pretzsch, H. (2021). Utilising forest inventory data for biodiversity assessment. *Ecological Indicators*, **121**, 107196.

- Hill, A., Mandallaz, D., and Langshausen, J. (2018). A double-sampling extension of the German national forest inventory for design-based small area estimation on forest district levels. *Remote Sensing*, **10**, 1–26.
- Irulappa-Pillai-Vijayakumar, D. B., Renaud, J. P., Morneau, F., McRoberts, R. E., and Vega, C. (2019). Increasing precision for french forest inventory estimates using the k-NN technique with optical and photogrammetric data and model-assisted estimators. *Remote Sensing*, **11**, 1–19.
- Jeong, K. M. and Son, J. (2009). Estimation of small area proportions based on logistic mixed model. *The Korean Journal of Applied Statistics*, **22**, 153–161.
- Jiang, J. and Lahiri, P. (2001). Empirical best prediction for small area inference with binary data. *Annals of the Institute of Statistical Mathematics*, **53**, 217–243.
- Jiang, J. and Lahiri, P. (2006). Mixed model prediction and small area estimation. *Test*, **15**, 1–96.
- Jiang, J. and Nguyen, T. (2021). *Linear and Generalized Linear Mixed Models and Their Applications (Vol. 1)*. Springer.
- Jiang, J. and Rao, J. S. (2020). Robust small area estimation: An overview. *Annual Review of Statistics and Its Application*, **7**, 337–360.
- Kangas, A., Astrup, R., Breidenbach, J., Fridman, J., Gobakken, T., Korhonen, K. T., Maltamo, M., Nilsson, M., Nord-Larsen, T., Næsset, E., and Olsson, H. (2018). Remote sensing and forest inventories in Nordic countries—roadmap for the future. *Scandinavian Journal of Forest Research*, **33**, 397–412.
- Knoke, T., Kindu, M., Schneider, T., and Gobakken, T. (2021). Inventory of forest attributes to support the integration of non-provisioning ecosystem services and biodiversity into forest planning—from collecting data to providing information. *Current Forestry Reports*, **7**, 38–58.
- Kordos, J. (2016). Development of small area estimation in official statistics. *Statistics in Transition. New Series*, **17**, 105–132.
- Lehtonen, R., Särndal, C. E., and Veijanen, A. (2003). The effect of model choice in estimation for domains, including small domains. *Survey Methodology*, **29**, 33–44.
- Magnussen, S., Mauro, F., Breidenbach, J., Lanz, A., and Kändler, G. (2017). Area-level analysis of forest inventory variables. *European Journal of Forest Research*, **136**, 839–855.
- Mandallaz, D. (2013). Design-based properties of some small-area estimators in forest inventory with two-phase sampling. *Canadian Journal of Forest Research*, **43**, 441–449.
- Mandallaz, D., Breschan, J., and Hill, A. (2013). New regression estimators in forest inventories with two-phase sampling and partially exhaustive information: a design-based monte carlo approach with applications to small-area estimation. *Canadian Journal of Forest Research*, **43**, 1023–1031.
- Marchetti, S., Giusti, C., Pratesi, M., Salvati, N., Giannotti, F., Pedreschi, D., Rinzivillo, S., Pappalardo, L., and Gabrielli, L. (2015). Small area model-based estimators using big data sources. *Journal of Official Statistics*, **31**, 263–281.
- Mauro, F., Molina, I., García-Abril, A., Valbuena, R., and Ayuga-Téllez, E. (2016). Remote sensing estimates and measures of uncertainty for forest variables at different aggregation levels. *Environmetrics*, **27**, 225–238.

- Mauro, F., Monleon, V. J., Temesgen, H., and Ford, K. R. (2017). Analysis of area level and unit level models for small area estimation in forest inventories assisted with LiDAR auxiliary information. *PloS One*, **12**, e0189401.
- McRoberts, R. E. (2010). Probability-and model-based approaches to inference for proportion forest using satellite imagery as ancillary data. *Remote Sensing of Environment*, **114**, 1017–1025.
- McRoberts, R. E. (2012). Estimating forest attribute parameters for small areas using nearest neighbors techniques. *Forest Ecology and Management*, **272**, 3–12.
- McRoberts, R. E., Andersen, H. E., and Næsset, E. (2014). Using airborne laser scanning data to support forest sample surveys. In Maltamo, M., Næsset, E., and Vauhkonen, J., editors, *Forestry Applications of Airborne Laser Scanning. Managing Forest Ecosystems*. Springer, Dordrecht.
- Nguyen, C. V. (2023). Coverage of safely managed water, sanitation and hygiene: An application of small area estimation. *Science of The Total Environment*, **896**, 165277.
- Noble, A., Haslett, S., and Arnold, G. (2002). Small area estimation via generalized linear models. *Journal of Official Statistics*, **18**, 45–60.
- Novo-Fernández, A., Barrio-Anta, M., Recondo, C., Cámara-Obregón, A., and López-Sánchez, C. A. (2019). Integration of national forest inventory and nationwide airborne laser scanning data to improve forest yield predictions in north-western Spain. *Remote Sensing*, **11**, 1693.
- Næsset, E. (2014). Area-based inventory in Norway – from innovation to an operational reality. In Maltamo, M., Næsset, E., and Vauhkonen, J., editors, *Forestry Applications of Airborne Laser Scanning. Managing Forest Ecosystems*. Springer, Dordrecht.
- Opsomer, J. D., Botts, C., and Kim, J. Y. (2003). Small area estimation in a watershed erosion assessment survey. *Journal of Agricultural, Biological, and Environmental Statistics*, **8**, 139–152.
- Packalén, P. and Maltamo, M. (2007). The k-MSN method for the prediction of species-specific stand attributes using airborne laser scanning and aerial photographs. *Remote Sensing of Environment*, **109**, 328–341.
- Pandey, P. C., Srivastava, P. K., Chetri, T., Choudhary, B. K., and Kumar, P. (2019). Forest biomass estimation using remote sensing and field inventory: a case study of Tripura, India. *Environmental Monitoring and Assessment*, **191**, 1–15.
- Panse, V. G., Rajagopalan, M., and Pillai, S. S. (1966). Estimation of crop yields for small areas. *Biometrics*, **66**, 374–388.
- Pramanik, S., Muthusamy, N., Gera, R., and Laxminarayan, R. (2015). Vaccination coverage in India: a small area estimation approach. *Vaccine*, **33**, 1731–1738.
- Prasad, N. N. and Rao, J. N. (1990). The estimation of the mean squared error of small-area estimators. *Journal of the American Statistical Association*, **85**, 163–171.
- Prisley, S., Bradley, J., Clutter, M., Friedman, S., Kempka, D., Rakestraw, J., and Sonne Hall, E. (2021). Needs for small area estimation: Perspectives from the US private forest sector. *Frontiers in Forests and Global Change*, **4**, 746439.
- Pulkkinen, M. and Zell, J. (2019). *Overview of Research Carried out on Small-Area Estimation Around Swiss NFI Since 2010*. Swiss Federal Institute for Forest, Snow, and Landscape Research (WSL), Birmensdorf.

- Rahlf, J., Hauglin, M., Astrup, R., and Breidenbach, J. (2021). Timber volume estimation based on airborne laser scanning-comparing the use of national forest inventory and forest management inventory data. *Annals of Forest Science*, **78**, 1–14.
- Rao, J. N. K. (2003). Some new developments in small area estimation. *Journal of the Iranian Statistical Society*, **2**, 145–169.
- Rao, J. N. K. and Molina, I. (2015). *Small Area Estimation*. John Wiley and Sons Inc, Hoboken, NJ, USA, 2nd edition.
- Reich, R. M. and Aguirre-Bravo, C. (2009). Small-area estimation of forest stand structure in Jalisco, Mexico. *Journal of Forestry Research*, **20**, 285–292.
- Saarela, S., Holm, S., Grafström, A., Schnell, S., Næsset, E., Gregoire, T. G., Nelson, R. F., and Ståhl, G. (2016). Hierarchical model-based inference for forest inventory utilizing three sources of information. *Annals of Forest Science*, **73**, 895–910.
- Sharma, S. D., Srivastava, A. K., and Sud, U. C. (2004). Small area crop estimation for crop yield estimates at Gram Panchayat level. *Journal of the Indian Society of Agricultural Statistics*, **57**, 26–37.
- Singh, R. and Goel, R. C. (2000). Use of remote sensing satellite data in crop surveys. Technical report, Indian Agricultural Statistics Research Institute, New Delhi.
- Sisodia, B. V. S. and Chandra, H. (2012). Estimation of crop production at smaller geographical level in India. *Journal of the Indian Society of Agricultural Statistics*, **66**, 313–319.
- Srivastava, A. K., Ahuja, D. L., Bhatia, D. K., and Mathur, D. C. (1999). Small area estimation techniques in agriculture. In *Souvenir on Advances in Agricultural Statistics and Computer Application*, New Delhi. Indian Agricultural Statistics Research Institute.
- Srivastava, A. K., Sud, U. C., and Chandra, H. (2007). Small area estimation - an application to national sample survey data. *Journal of the Indian Society of Agricultural Statistics*, **61**, 249–254.
- Stanke, H., Finley, A. O., and Domke, G. M. (2022). Simplifying small area estimation with rFIA: a demonstration of tools and techniques. *Frontiers in Forests and Global Change*, **5**, 745874.
- Strîmbu, V. F., Ørka, H. O., and Næsset, E. (2021). Consistent forest biomass stock and change estimation across stand, property, and landscape levels. *Canadian Journal of Forest Research*, **51**, 848–858.
- Sud, U. C., Chandra, H., and Srivastava, A. K. (2011). Crop yield estimation at district level using improvement of crop statistics scheme data - an application of small area estimation technique. *Journal of the Indian Society of Agricultural Statistics*, **66**, 321–326.
- Sud, U. C., Mathur, D. C., Srivastava, A. K., Bathla, H. V. L., and Jha, G. K. (2001). Crop yield estimation at small area level using farmers' estimates. Technical report, Indian Agricultural Statistics Research Institute, New Delhi.
- Sugasawa, S. and Kubokawa, T. (2020). Small area estimation with mixed models: a review. *Japanese Journal of Statistics and Data Science*, **3**, 693–720.
- Thumaty, K. C., Fararoda, R., Middinti, S., Gopalakrishnan, R., Jha, C. S., and Dadhwal, V. K. (2016). Estimation of above ground biomass for central indian deciduous forests using ALOS PALSAR L-band data. *Journal of the Indian Society of Remote Sensing*, **44**, 31–39.

- Tomppo, E., Haakana, M., Katila, M., and Peräsaari, J. (2008a). *Multi-source National Forest Inventory: Methods and applications*. Springer Science & Business Media.
- Tomppo, E. and Katila, M. (1991). Satellite image-based national forest inventory of Finland for publication in the IGARSS 91 digest. In *Proceedings IGARSS'91 Remote Sensing: Global Monitoring for Earth Management*. IEEE.
- Tomppo, E., Olsson, H., Ståhl, G., Nilsson, M., Hagner, O., and Katila, M. (2008b). Combining national forest inventory field plots and remote sensing data for forest databases. *Remote Sensing of Environment*, **112**, 1982–1999.
- Vega, C., Renaud, J. P., Sagar, A., and Bouriaud, O. (2021). A new small area estimation algorithm to balance between statistical precision and scale. *International Journal of Applied Earth Observation and Geoinformation*, **97**, 1–11.
- Vidal, C., Alberdi, I., Hernández, L., and Redmond, J. J. (2016). *National forest inventories*, volume 10. Springer Science+ Business Media.
- Wagner, J., Münnich, R., Hill, J., Stoffels, J., and Udelhoven, T. (2017). Non-parametric small area models using shape-constrained penalized b-splines. *Journal of the Royal Statistical Society Series A: Statistics in Society*, **180**, 1089–1109.
- Wilson, B., Coulston, J., Prisley, S., and Radtke, P. (2023). *Small area estimation in forest inventories: New needs, methods, and tools*. Frontiers in Forests and Global Change.
- Wulder, M. A., White, J. C., Fournier, R. A., Luther, J. E., and Magnussen, S. (2008). Spatially explicit large area biomass estimation: three approaches using forest inventory and remotely sensed imagery in a GIS. *Sensors*, **8**, 529–560.



Resolvability of Some BIB and Regular Group Divisible Designs

Shyam Saurabh¹, Sheet Nihal Topno² and Dinesh Prasad³

¹Department of Mathematics, Tata College, Kolhan University, Chaibasa, Jharkhand, India

²University Department of Mathematics, Ranchi University, Ranchi, Jharkhand, India

³Department of Mathematics, Gossner College, Ranchi, Jharkhand, India

Received: 03 October 2024; Revised: 28 October 2024; Accepted: 29 December 2024

Abstract

$(\mu_1, \mu_2, \dots, \mu_t)$ -resolvable solutions of some BIB and regular group divisible designs are obtained by decomposing the $v(=mn) \times b$ incidence matrix into smaller circulant submatrices of orders $m \times t$.

Key words: Regular group divisible design; Balanced incomplete block (BIB) design; Permutation circulant matrix; Resolvability.

AMS Subject Classifications: 62K10, 05B05

1. Introduction

Let the incidence matrix N of a block design $D(v, r, k, b)$ may be decomposed into submatrices as $N = (N_1|N_2|\dots|N_t)$ such that each row sum of N_i ($1 \leq i \leq t$) is μ_i . Then the design is $(\mu_1, \mu_2, \dots, \mu_t)$ -resolvable [see Kageyama (1976), Saurabh (2024b)]. If $\mu_1 = \mu_2 = \dots = \mu_t = \mu$ then the design is μ -resolvable. Such designs are also denoted as A -resolvable in combinatorial design theory [see Ge and Miao (2007)]. A practical application of such designs may be found in Kageyama (1976). The resolvable solutions obtained here are not earlier reported in the Tables of Clatworthy (1973) and Saurabh *et al.* (2021) [see Table 1].

Let $v = mn$ elements be arranged in an $m \times n$ array. A *regular group divisible* (RGD) design is an arrangement of the $v = mn$ elements in b blocks each of size k such that:

1. Every element occurs at most once in a block;
2. Every element occurs in r blocks;
3. Every pair of elements, which are in the same row of the $m \times n$ array, occur together in λ_1 blocks whereas every other pair of elements occur together in λ_2 blocks;
4. $r - \lambda_1 > 0$ and $rk - v\lambda_2 > 0$.

Further let N be the incidence matrix of a GD design then the structure of NN^T is given as [see Saurabh and Sinha (2023) for GD association schemes]:

- (i) $NN^\top = (r - \lambda_1)(I_m \otimes I_n) + (\lambda_1 - \lambda_2)(I_m \otimes J_n) + \lambda_2(J_m \otimes J_n)$; or
(ii) $NN^\top = (r - \lambda_2)(I_n \otimes I_m) + \lambda_2(J_n \otimes J_m) + (\lambda_1 - \lambda_2)\{(J_n - I_n) \otimes I_m\}$,
where J_n is an $n \times n$ matrix all of whose entries are 1.

Notations: I_n is the identity matrix of order n , $A \otimes B$ is the Kronecker product of two matrices A and B , $0_{m \times n}$ is a null matrix of order $m \times n$ and $N^\top$ is transpose of the matrix N . In Section 2, N represents incidence matrix of the design whereas RX and RXa numbers are from Clatworthy (1973) and Freeman (1976) respectively.

2. Resolvable solutions

2.1. $(\mu_1, \mu_2, \dots, \mu_t)$ -resolvable GD designs

Here $\alpha = circ(010\dots 0)$ is a permutation circulant matrix of order m such that $\alpha^m = I_m$.

Table 1: $(\mu_1, \mu_2, \dots, \mu_t)$ -resolvable RGD designs

No.	Design	v	r	k	b	λ_1	λ_2	m	n	Present Status
1	$R89a$	18	10	3	60	4	1	9	2	(1,3,3,3)-resolvable
2	$R109a$	12	7	4	21	1	2	6	2	(2,2,3)-resolvable
3	$R113a$	14	10	4	35	6	2	7	2	(2,2,2,4)-resolvable
4	$R123a$	18	10	4	45	0	2	6	3	(2,4,4)-resolvable
5	$R124a$	22	8	4	44	4	1	11	2	(2,2,4)-resolvable
6	$R126a$	24	9	4	54	5	1	12	2	(1,2,2,4)-resolvable
7	$R128a$	26	10	4	65	6	1	13	2	(2,2,2,4)-resolvable
8	$R152a$	22	10	5	44	11	2	0	2	5-resolvable
9	$R163$	45	10	5	90	0	1	9	5	5-resolvable
10	$R167a$	12	9	6	18	5	4	6	2	(3,6)-resolvable

1. $R89a$: $v = 18$, $r = 10$, $k = 3$, $b = 60$, $\lambda_1 = 4$, $\lambda_2 = 1$, $m = 9$, $n = 2$.

Consider a block matrix

$$N = (N_1|N_2|N_3|N_4) =$$

$$\left(\begin{array}{cc|cc|cc|cc} P & 0_{9 \times 3} & \alpha + \alpha^2 & \alpha & \alpha + \alpha^8 & \alpha^4 & \alpha + \alpha^6 & \alpha \\ 0_{9 \times 3} & P & \alpha & \alpha + \alpha^2 & \alpha & \alpha + \alpha^8 & \alpha^4 & \alpha + \alpha^6 \end{array} \right),$$

where $P = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \otimes circ(010)$. Then $NN^\top = 9(I_2 \otimes I_9) + (J_2 \otimes J_9) + 3\{(J_2 - I_2) \otimes I_9\}$.

Hence N represents the incidence matrix of $R89a$. Further since each row sum of N_1 , N_2 , N_3 and N_4 is 1, 3, 3 and 3 respectively, the design is (1, 3, 3, 3)-resolvable.

2. $R109a$: $v = 12$, $r = 7$, $k = 4$, $b = 21$, $\lambda_1 = 1$, $\lambda_2 = 2$, $m = 6$, $n = 2$.

$$N = (N_1|N_2|N_3) = \left(\begin{array}{cc|cc} \alpha + \alpha^3 & \alpha + \alpha^4 & \alpha + \alpha^2 + \alpha^3 & 0_{6 \times 3} \\ \alpha + I_6 & \alpha^2 + I_6 & \alpha^4 & P \end{array} \right),$$

where $P = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \otimes circ(011)$. Since each row sum of N_1 , N_2 and N_3 is 2, 2 and 3 respectively, the design is (2, 2, 3)-resolvable.

3. *R113a*: $v = 14, r = 10, k = 4, b = 35, \lambda_1 = 6, \lambda_2 = 2, m = 7, n = 2$.

$$N = (N_1|N_2|N_3|N_4) = \left(\begin{array}{c|c|c|c} \alpha + \alpha^2 & \alpha + \alpha^3 & \alpha + \alpha^4 & \alpha + \alpha^2 + \alpha^4 \\ \hline \alpha + \alpha^2 & \alpha + \alpha^3 & \alpha + \alpha^4 & I_7 \\ \hline & & & \alpha + \alpha^2 + \alpha^4 \end{array} \right).$$

Since each row sum of N_1, N_2, N_3 and N_4 is 2, 2, 2 and 4 respectively, the design is $(2, 2, 2, 4)$ -resolvable.

4. *R123a*: $v = 18, r = 10, k = 4, b = 45, \lambda_1 = 0, \lambda_2 = 2, m = 6, n = 3$.

$$N = (N_1|N_2|N_3) =$$

$$\left(\begin{array}{c|c|c|c} P_{6 \times 9} & 0_6 & \alpha^4 & \alpha + \alpha^2 + \alpha^3 \\ \hline Q_{6 \times 9} & \alpha + \alpha^2 + \alpha^3 & 0_6 & \alpha^4 \\ \hline R_{6 \times 9} & \alpha^4 & \alpha + \alpha^2 + \alpha^3 & 0_6 \\ \hline & & & \alpha^4 \end{array} \right),$$

where $P_{6 \times 9} = \left(\begin{array}{c|c|c} I_3 & 0_3 & \beta^2 \\ \hline I_3 & 0_3 & \beta^2 \end{array} \right)$, $Q_{6 \times 9} = \left(\begin{array}{c|c|c} \beta^2 & I_3 & 0_3 \\ \hline \beta^2 & I_3 & 0_3 \end{array} \right)$, $R_{6 \times 9} = \left(\begin{array}{c|c|c} 0_3 & \beta^2 & I_3 \\ \hline 0_3 & \beta^2 & I_3 \end{array} \right)$ and $\beta = \text{circ}(010)$.

Since each row sum of N_1, N_2 and N_3 is 2, 4 and 4 respectively, the design is $(2, 4, 4)$ -resolvable.

5. *R124a*: $v = 22, r = 8, k = 4, b = 44, \lambda_1 = 4, \lambda_2 = 1, m = 11, n = 2$.

$$N = (N_1|N_2|N_3) = \left(\begin{array}{c|c|c} \alpha + \alpha^7 & \alpha + \alpha^8 & \alpha + \alpha^3 + \alpha^4 \\ \hline \alpha + \alpha^8 & \alpha + \alpha^7 & \alpha \\ \hline & & \alpha + \alpha^3 + \alpha^4 \end{array} \right).$$

Since each row sum of N_1, N_2 and N_3 is 2, 2 and 4 respectively, the design is $(2, 2, 4)$ -resolvable.

6. *R126a*: $v = 24, r = 9, k = 4, b = 54, \lambda_1 = 5, \lambda_2 = 1, m = 12, n = 2$.

$$N = (N_1|N_2|N_3|N_4) = \left(\begin{array}{c|c|c|c} P & \alpha + \alpha^4 & \alpha + \alpha^{11} & \alpha + \alpha^2 + \alpha^9 \\ \hline P & \alpha + \alpha^{11} & \alpha + \alpha^4 & \alpha \\ \hline & & & \alpha + \alpha^2 + \alpha^9 \end{array} \right),$$

where $P = \left(\begin{array}{c} \text{circ}(010000) \\ I_6 \end{array} \right)$. Since each row sum of N_1, N_2, N_3 and N_4 is 1, 2, 2 and 4 respectively, the design is $(1, 2, 2, 4)$ -resolvable.

7. *R128a*: $v = 26, r = 10, k = 4, b = 65, \lambda_1 = 6, \lambda_2 = 1, m = 13, n = 2$.

$$N = (N_1|N_2|N_3|N_4) = \left(\begin{array}{c|c|c|c} \alpha + \alpha^2 & \alpha + \alpha^4 & \alpha + \alpha^5 & \alpha + \alpha^6 + \alpha^8 \\ \hline \alpha + \alpha^4 & \alpha + \alpha^2 & \alpha + \alpha^5 & \alpha \\ \hline & & & \alpha + \alpha^6 + \alpha^8 \end{array} \right).$$

Since each row sum of N_1, N_2, N_3 and N_4 is 2, 2, 2 and 4 respectively, the design is $(2, 2, 2, 4)$ -resolvable.

8. *R152a*: $v = 22, r = 10, k = 5, b = 44, \lambda_1 = 0, \lambda_2 = 2, m = 11, n = 2$.

$$N = (N_1|N_2) = \left(\begin{array}{c|c} \alpha + \alpha^2 + \alpha^3 + \alpha^6 & \alpha^9 \\ \hline \alpha^9 & \alpha + \alpha^2 + \alpha^3 + \alpha^6 \end{array} \right).$$

Since each row sum of N_1 and N_2 is 5, the design is 5-resolvable [see Saurabh (2024a)].

9. *R163*: $v = 45, r = 10, k = 5, b = 90, \lambda_1 = 0, \lambda_2 = 1, m = 9, n = 5$.

A 5-resolvable solution of the design is presented in Table 2:

Table 2: Resolution classes of $R163$

Resolution class I				
(2, 3, 10, 22, 25)	(15, 17, 19, 32, 36)	(4, 7, 29, 30, 37)	(1, 14, 18, 42, 44)	(23, 25, 30, 40, 44)
(1, 3, 11, 23, 26)	(13, 18, 20, 33, 34)	(5, 8, 28, 30, 38)	(2, 15, 16, 40, 45)	(20, 27, 31, 39, 44)
(1, 2, 12, 24, 27)	(14, 16, 21, 31, 35)	(6, 9, 28, 29, 39)	(3, 13, 17, 41, 43)	(21, 25, 32, 37, 45)
(5, 6, 13, 19, 25)	(11, 18, 22, 30, 35)	(1, 7, 32, 33, 40)	(4, 12, 17, 38, 45)	(19, 26, 33, 38, 43)
(4, 6, 14, 20, 26)	(12, 16, 23, 28, 36)	(2, 8, 31, 33, 41)	(5, 10, 18, 39, 43)	(21, 23, 34, 38, 42)
(4, 5, 15, 21, 27)	(10, 17, 24, 29, 34)	(3, 9, 31, 32, 42)	(6, 11, 16, 37, 44)	(19, 24, 35, 39, 40)
(8, 9, 16, 19, 22)	(12, 14, 25, 29, 33)	(1, 4, 35, 36, 43)	(7, 11, 15, 39, 41)	(20, 22, 36, 37, 41)
(7, 9, 17, 20, 23)	(10, 15, 26, 30, 31)	(2, 5, 34, 36, 44)	(8, 12, 13, 37, 42)	(24, 26, 28, 41, 45)
(7, 8, 18, 21, 24)	(11, 13, 27, 28, 32)	(3, 6, 34, 35, 45)	(9, 10, 14, 38, 40)	(22, 27, 29, 42, 43)
Resolution class II				
(11, 12, 19, 31, 34)	(6, 8, 10, 23, 27)	(5, 9, 33, 35, 37)	(5, 11, 17, 40, 42)	(20, 21, 28, 40, 43)
(10, 12, 20, 32, 35)	(4, 9, 11, 24, 25)	(6, 7, 31, 36, 38)	(6, 12, 18, 40, 41)	(19, 21, 29, 41, 44)
(10, 11, 21, 33, 36)	(5, 7, 12, 22, 26)	(4, 8, 32, 34, 39)	(7, 10, 13, 44, 45)	(19, 20, 30, 42, 45)
(14, 15, 22, 28, 34)	(2, 9, 13, 21, 26)	(3, 8, 29, 36, 40)	(8, 11, 14, 43, 45)	(23, 24, 31, 37, 43)
(13, 15, 23, 29, 35)	(3, 7, 14, 19, 27)	(1, 9, 30, 34, 41)	(9, 12, 15, 43, 44)	(22, 24, 32, 38, 44)
(13, 14, 24, 30, 36)	(1, 8, 15, 20, 25)	(2, 7, 28, 35, 42)	(1, 13, 16, 38, 39)	(22, 23, 33, 39, 45)
(17, 18, 25, 28, 31)	(3, 5, 16, 20, 24)	(2, 6, 30, 32, 43)	(2, 14, 17, 37, 39)	(26, 27, 34, 37, 40)
(16, 18, 26, 29, 32)	(1, 6, 17, 21, 22)	(3, 4, 28, 33, 44)	(3, 15, 18, 37, 38)	(25, 27, 35, 38, 41)
(16, 17, 27, 30, 33)	(2, 4, 18, 19, 23)	(1, 5, 29, 31, 45)	(4, 10, 16, 41, 42)	(25, 26, 36, 39, 42)

Remark 1: Juxtaposing the resolution classes I and II of $R163$ with the resolution class III given below, we obtain a $(1,5,5)$ -resolvable solution of BIB design with parameters: $v = 45$, $r = 11$, $k = 5$, $b = 99$, $\lambda = 1$. The blocks of resolution class III are rows of the GD scheme. This BIB design is listed as $T98$ in the Table of Takeuchi (1962).

Resolution class III: $(1, 10, 19, 28, 37)$; $(6, 15, 24, 33, 42)$; $(2, 11, 20, 29, 38)$; $(7, 16, 25, 34, 43)$; $(3, 12, 21, 30, 39)$; $(8, 17, 26, 35, 44)$; $(4, 13, 22, 31, 40)$; $(9, 18, 27, 36, 45)$; $(5, 14, 23, 32, 41)$.

10. $R167a$: $v = 12$, $r = 9$, $k = 6$, $b = 18$, $\lambda_1 = 5$, $\lambda_2 = 4$, $m = 6$, $n = 2$.

$$N = (N_1|N_2) = \begin{pmatrix} \alpha + \alpha^2 + \alpha^4 & I_6 + \alpha + \alpha^2 + \alpha^4 & \alpha^2 + \alpha^3 \\ \alpha + \alpha^2 + \alpha^4 & \alpha^2 + \alpha^3 & I_6 + \alpha + \alpha^2 + \alpha^4 \end{pmatrix}.$$

Since each row sum of N_1 and N_2 is 3 and 6 respectively, the design is $(3, 6)$ -resolvable.

2.2. $A(2,2,2,2,6)$ -resolvable BIB design

The following solution of BIB design $T49$: $v = 22$, $r = 14$, $k = 4$, $b = 77$, $\lambda = 2$ using the method of differences may be found in Takeuchi (1962):

$[(0_0, 1_0, 3_0, 10_1), (0_1, 1_1, 3_1, 10_0), (0_0, 1_0, 5_0, 8_1), (0_1, 3_1, 5_1, 8_0), (0_0, 4_0, 5_0, 6_1), (0_1, 4_1, 5_1, 6_0), (0_0, 4_0, 0_1, 4_1)] \bmod 11$.

The incidence matrix N of the design may be decomposed into block submatrices as follows:

$$N = (N_1|N_2|N_3|N_4|N_5) =$$

$$\left(\begin{array}{cccccc} I_{11} + \alpha^2 & I_{11} + \alpha^3 & I_{11} + \alpha^4 & I_{11} + \alpha^5 & I_{11} + \alpha^3 + \alpha^9 + \alpha^{10} & I_{11} \\ \alpha^5 + \alpha^8 & \alpha^4 + \alpha^7 & \alpha^3 + \alpha^9 & \alpha^2 + \alpha^6 & 0_{11} & I_{11} + \alpha^2 + \alpha^7 \\ & & & & & I_{11} + \alpha^9 + \alpha^{10} \end{array} \right),$$

where $\alpha = \text{circ}(010 \dots 0)$ is a permutation circulant matrix of order 11. Since each row sum of N_1, N_2, N_3, N_4 and N_5 is 2, 2, 2, 2 and 6 respectively, the design is $(2, 2, 2, 2, 6)$ -resolvable.

Further repeating the above solution three times, we obtain a 6-resolvable solution of the BIB design with parameters: $v = 22, r = 42, k = 4, b = 231, \lambda = 6$. The 6-resolvable solution is not reported in the Tables of Kageyama and Mohan (1983) and Subramani (1990).

Acknowledgements

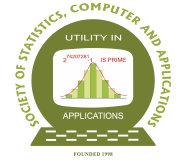
The authors are grateful to anonymous reviewers and Editor-in-chief for their insightful comments and suggestions.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Clatworthy, W. H. (1973). Tables of two-associate-class partially balanced designs. *National Bureau of Standards (U.S.), Applied Mathematics*, **63**.
- Freeman, G. H. (1976). A cyclic method of constructing regular group divisible incomplete block designs. *Biometrika*, **63**, 555–558.
- Ge, G. and Miao, Y. (2007). *PBDs, Frames, and Resolvability*. Chapman & Hall/ CRC, New York, 2nd edition.
- Kageyama, S. (1976). Resolvability of block designs. *The Annals of Statistics*, **4**, 655–661.
- Kageyama, S. and Mohan, R. N. (1983). On μ -resolvable BIB designs. *Discrete Mathematics*, **45**, 113–122.
- Saurabh, S. (2024a). A note on cyclic and partial cyclic solutions of block designs. *Gujarat Journal of Statistics and Data Science*, **40**, 45–50.
- Saurabh, S. (2024b). Resolvability of a BIB design of Takeuchi (1962). *Statistics and Applications*, **22**, 361–362.
- Saurabh, S. and Sinha, K. (2023). Matrix approaches to constructions of group divisible designs. *Bulletin of the ICA*, **97**, 83–105.
- Saurabh, S., Sinha, K., and Singh, M. K. (2021). A survey of resolvable solutions of partially balanced designs. *Communications in Statistics-Theory and Methods*, **52**, 1946–1962.
- Subramani, J. (1990). A note on μ -resolvable BIB designs. *Journal of the Indian Society of Agricultural Statistics*, **XLII**, 226–233.
- Takeuchi, K. (1962). A table of difference sets generating balanced incomplete block designs. *Review of International Statistical Institute*, **30**, 361–366.



Prabhu-Ajgaonkar's 1967 Result Revisited

Bikas K. Sinha¹ and Manisha Pal²

¹*Retired Professor, Stat-Math Unit, Indian Statistical Institute, Kolkata, India*

²*Department of Statistics, St. Xavier's University, Kolkata, India*

Received: 13 March 2025; Revised: 30 March 2025; Accepted: 02 April 2025

Abstract

This short communication revisits the work of Ajgaonkar (1967), wherein the author claimed that the precision of an estimator of population total (or population mean) decreases on increasing the sample size, if the estimator under consideration is an average function, and the population elements are drawn with varying probabilities of selection at each draw. Prabhu-Ajgaonkar (1967) justified this claim through some examples. However, this paper disproves the claim so made!

Key words: Finite population; Unequal probability sampling; Unbiased estimation of population total; Prabhu-Ajgaonkar's claim; Lanke's estimator.

1. Introduction

In survey sampling theory, it is a common belief that an increase in sample size will result in increase of precision of a linear unbiased estimator of the population total (or population mean) under a given sampling scheme. However, Prabhu-Ajgaonkar (1967) made a claim that looked counter-intuitive. He stated that when the population units are drawn with varying probabilities of selection at each draw and the estimator/parameter under consideration is an average function, the statement does not hold good. Examples were presented, whereby he compared unbiased estimators based on sample sizes 1 and 2 to emphasize his point. His study did not raise any question/concern.

Meanwhile, Lanke (1975) published his Ph.D. thesis which received much attention of the researchers in the area of statistical inference, particularly in finite population inference. For any given unbiased estimator (linear or non-linear) of a population parameter, Lanke (1975) defined an improved unbiased estimator by Rao-Blackwellization. The improved estimator is now frequently termed as “Lanke's Estimator”. By virtue of the result, we have that in comparison to any homogeneous linear unbiased estimator of a finite population total based on a given sampling scheme, the Lanke's estimator is superior!

In this short communication, we revisit Prabhu-Ajgaonkar's claim. We consider an example of finite population cited in Prabhu-Ajgaonkar (1967), and show that under the same

sampling design with sample size $n = 2$, the Lanke's unbiased estimator of the population total is not only more precise than the unbiased estimator considered by Prabhu-Ajgaonkar, but also better than the unique unbiased estimator for $n = 1$. Further, we establish the superiority of Lanke's estimator for any general finite population size(N).

2. Illustration as in Prabhu-Ajgaonkar (1967)

To illustrate his claim, Prabhu-Ajgaonkar (1967) used the three populations furnished by Yates and Grundy (1953), each having $N = 4$ units with unequal probabilities of selection in a draw. For sample size $n = 1$, sampling is done with unequal probabilities, while for $n = 2$, sampling is done without replacement, and the first unit is drawn with unequal probabilities while in the second draw the remaining units are assumed to have equal probabilities of selection.

We consider the first population, for which the values (Y_i) of the study character Y and the selection probabilities (p_i) of the units are as given in Table 1.

Table 1: Population of size $N = 4$

Unit i	1	2	3	4	Total
Y_i	0.5	0.2	0.1	0.2	7.0
p_i	0.1	0.2	0.3	0.4	1.0

The population total is $T(Y) = 7.0$.

For a sample of size $n = 1$, the unique unbiased estimator $\hat{T}_1(Y)$ of $T(Y)$ gives the estimate

$$e(s_i) = \frac{Y_i}{p_i}, \quad \text{for sample } s_i = \{i\}, \quad i = 1, 2, 3, 4.$$

It is readily verified that $E[\hat{T}_1(Y)] = T(Y) = 7.0$, and $\text{Var}[\hat{T}_1(Y)] = 1.00$.

Next, the sampling design is extended to the case of $n = 2$ by drawing exactly one of the remaining 3 units at random. In effect, therefore, one is dealing with the extended design specified in Table 2 below.

Table 2: Extended sampling design for $n = 2$

Sample $s(i, j)$	(1,2)	(1,3)	(1,4)	(2,3)	(2,4)	(3,4)
$p^*(s(i, j))$	$\frac{0.3}{3}$	$\frac{0.4}{3}$	$\frac{0.5}{3}$	$\frac{0.5}{3}$	$\frac{0.6}{3}$	$\frac{0.7}{3}$

In the above table, $p^*(s(i, j)) = \frac{p_i + p_j}{3}$, for $i, j = 1, 2, 3, 4$ with $i < j$.

The unbiased estimator $\hat{T}_2(Y)$ considered by Prabhu-Ajgaonkar (1967) for $n = 2$ is

the Horvitz-Thompson estimator (HTE). It is given by

$$\hat{T}_2(Y|s(i, j)) = \left(\frac{Y_i}{\pi_i} + \frac{Y_j}{\pi_j} \right),$$

where π_i denote the inclusion probabilities of the population units.

The computation of the estimates $e[s(i, j)]$ of the population total are indicated in Table 3 below.

Table 3: HTE of the population total for $n = 2$

Sample $s(i, j)$	(1,2)	(1,3)	(1,4)
$e[s(i, j)]$	$3 \left(\frac{0.5}{1.2} + \frac{1.2}{1.4} \right)$	$3 \left(\frac{0.5}{1.2} + \frac{2.1}{1.6} \right)$	$3 \left(\frac{0.5}{1.2} + \frac{3.2}{1.8} \right)$
Sample $s(i, j)$	(2,3)	(2,4)	(3,4)
$e[s(i, j)]$	$3 \left(\frac{1.2}{1.4} + \frac{2.1}{1.6} \right)$	$3 \left(\frac{1.2}{1.4} + \frac{3.2}{1.8} \right)$	$3 \left(\frac{2.1}{1.6} + \frac{3.2}{1.8} \right)$

Thus,

$$E[\hat{T}_2(Y)] = 7.00 = T(Y), \quad \text{and} \quad \text{Var}[\hat{T}_2(Y)] = 2.8833 > 1 = \text{Var}[\hat{T}_1(Y)].$$

Note: Prabhu-Ajgaonkar (1967) wrongly reported the variance as 6.32 – it was, of course, a computational error. This was his contention to claim that sample size $n = 1$ fares better than sample size $n = 2$. However, the above estimator is far from being Lanke's estimator!

3. Lanke's estimator

Lanke's unbiased estimator $\hat{T}_L(Y)$ of the population total, based on the extended sampling design given in Table 2, gives the sample estimates of the population total as:

$$e^*[s(i, j)] = \frac{Y_i + Y_j}{p_i + p_j}, \quad i, j = 1, 2, 3, 4, \quad i < j.$$

We note that:

1. $E[\hat{T}_L(Y)] = 7.00$
2. $E[\hat{T}_L^2(Y)] = 49.3629 \Rightarrow \text{Var}[\hat{T}_L(Y)] = 49.3629 - 7.00^2 = 0.3629$

which implies:

$$\text{Var}[\hat{T}_L(Y)] = 0.3629 < 1 = \text{Var}[\hat{T}_1(Y)].$$

Thus, it transpires that Lanke's estimator for sample size $n = 2$ fares better than the unique unbiased estimator for $n = 1$, as it should.

Remark 1: It may be noted that **Lanke's estimator is different from the usual Horvitz-Thompson Estimator (HTE)**, which was taken by Prabhu-Ajgaonkar (1967) in his study. The HTE uses inclusion probabilities in the denominator, while Lanke's estimator uses the sum of selection probabilities of the sample.

4. Variance inequality for any finite population

In this section, the variance inequality

$$\text{Var}(\hat{T}_L(Y)) \leq \text{Var}(\hat{T}_1(Y))$$

is established, where $\hat{T}_1(Y)$ and $\hat{T}_L(Y)$ denote, respectively, the unbiased estimators of the population total $T(Y)$ based on sample sizes $n = 1$ and $n = 2$. Here, $\hat{T}_1(Y)$ is the Horvitz-Thompson estimator (HTE), and $\hat{T}_L(Y)$ is Lanke's estimator under the sampling design indicated in Prabhu-Ajgaonkar (1967).

Let $Y_i, i = 1, 2, \dots, N$ denote the values of the study variable, and p_i the corresponding probabilities of selection for each unit. Then:

$$\text{Var}(\hat{T}_1(Y)) = \sum_{i=1}^N \frac{Y_i^2}{p_i} - [T(Y)]^2.$$

$$\text{Var}(\hat{T}_L(Y)) = \sum_{i < j} \frac{(Y_i + Y_j)^2}{\frac{p_i + p_j}{2(N-1)}} - [T(Y)]^2$$

A sufficient condition for

$$\text{Var}(\hat{T}_1(Y)) \geq \text{Var}(\hat{T}_L(Y))$$

is that for all pairs (i, j) with $i < j$:

$$(N-1) \left[\frac{Y_i^2}{p_i} + \frac{Y_j^2}{p_j} \right] \geq \frac{(Y_i + Y_j)^2}{p_i + p_j}$$

This inequality is equivalent to:

$$(N-2) \left[\frac{Y_i^2}{p_i} + \frac{Y_j^2}{p_j} \right] + \frac{p_j Y_i^2}{p_i(p_i + p_j)} + \frac{p_i Y_j^2}{p_j(p_i + p_j)} - \frac{2Y_i Y_j}{p_i + p_j} \geq 0$$

which simplifies to:

$$(N-2) \left[\frac{Y_i^2}{p_i} + \frac{Y_j^2}{p_j} \right] + \frac{1}{p_i p_j (p_i + p_j)} (p_j Y_i - p_i Y_j)^2 \geq 0 \quad \text{for all } i < j.$$

Since the left-hand side is a sum of non-negative quantities and $N \geq 2$, this inequality always holds. Hence, we conclude

$$\text{Var}(\hat{T}_1(Y)) \geq \text{Var}(\hat{T}_L(Y)).$$

Thus, the claim made by Prabhu-Ajgaonkar (1967) is not tenable.

Remark 2: It may be mentioned here that Pal and Sinha (2024) devoted a chapter to the use of additional resources for the improvement of an estimator. This area of research has not yet been duly exposed to the researchers in the broad area of statistical inference.

5. Estimation of $\text{Var}(\hat{T}_L(Y))$

We proceed to unbiasedly estimate $\text{Var}(\hat{T}_L(Y))$ in the general set-up described in Section 4. Since

$$\text{Var}(\hat{T}_L(Y)) = E[\hat{T}_L(Y)^2] - T^2(Y),$$

it suffices to find an unbiased estimator of $T^2(Y)$.

Now,

$$T^2(Y) = \left(\sum_{i=1}^N Y_i \right)^2 = \sum_{i=1}^N Y_i^2 + \sum_{\substack{i,j=1 \\ i \neq j}}^N Y_i Y_j.$$

Note that the cross-product term can be written as

$$\sum_{\substack{i,j=1 \\ i \neq j}}^N Y_i Y_j = \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{Y_i^2 + Y_j^2}{2(N-1)} + \sum_{\substack{i,j=1 \\ i \neq j}}^N Y_i Y_j,$$

which simplifies the expression.

Consider the following estimator of $T^2(Y)$ based on the sample $s(i, j)$:

$$\hat{T}^2(Y) |_{s(i,j)} = \frac{1}{p_i + p_j} [Y_i^2 + Y_j^2 + 2(N-1)Y_i Y_j].$$

It is easily verified that this is an unbiased estimator of $T^2(Y)$.

Hence, an unbiased estimate of $\text{Var}(\hat{T}_L(Y))$ for the sample $s(i, j)$ is:

$$\widehat{\text{Var}}(\hat{T}_L(Y)) |_{s(i,j)} = \left(\frac{Y_i + Y_j}{p_i + p_j} \right)^2 - \frac{1}{p_i + p_j} [Y_i^2 + Y_j^2 + 2(N-1)Y_i Y_j].$$

This can also be expressed as a quadratic form:

$$\widehat{\text{Var}}(\hat{T}_L(Y)) |_{s(i,j)} = \frac{1}{(p_i + p_j)^2} \begin{pmatrix} Y_i & Y_j \end{pmatrix} A(i, j) \begin{pmatrix} Y_i \\ Y_j \end{pmatrix},$$

where

$$A(i, j) = \begin{pmatrix} 1 - (p_i + p_j) & \{1 - (p_i + p_j)\} - (p_i + p_j)(N-2) \\ \{1 - (p_i + p_j)\} - (p_i + p_j)(N-2) & 1 - (p_i + p_j) \end{pmatrix}.$$

Clearly, $\widehat{\text{Var}}(\hat{T}_L(Y))|_{s(i,j)} \geq 0$ provided $A(i, j)$ is positive semi-definite. This is ensured when

- (i). $1 - (p_i + p_j) \geq 0$, and
- (ii). $\det A(i, j) \geq 0$.

Condition (ii) is equivalent to:

$$\frac{1 - (p_i + p_j)}{p_i + p_j} \geq \frac{N - 2}{2}, \quad \text{for } N > 2. \quad (2)$$

5.1. Numerical illustration

For the example in Section 2 with $N = 4$, and

$$(Y_1, p_1) = (0.5, 0.1), \quad (Y_2, p_2) = (1.2, 0.2), \quad (Y_3, p_3) = (2.1, 0.3), \quad (Y_4, p_4) = (3.2, 0.4),$$

the unbiased estimates of $\text{Var}(\hat{T}_L(Y))$ are shown in Table 4.

Table 4: Unbiased estimates of $\text{Var}(\hat{T}_L(Y))$ for various pairs (i, j)

Sample $s(i, j)$	(1,2)	(1,3)	(1,4)	(2,3)	(2,4)	(3,4)
$p^*(s(i, j))$	$\frac{0.3}{3}$	$\frac{0.4}{3}$	$\frac{0.5}{3}$	$\frac{0.5}{3}$	$\frac{0.6}{3}$	$\frac{0.7}{3}$
$\widehat{\text{Var}}(\hat{T}_L(Y))$	14.478	14.850	14.580	1.620	-4.089	-21.202

Remark 3. The estimates for samples (2, 4) and (3, 4) are negative since the left-hand side of condition (2) is

$$\frac{1 - 0.6}{0.6} = \frac{2}{3}, \quad \text{and} \quad \frac{1 - 0.7}{0.7} = \frac{3}{7},$$

which are both less than the right-hand side value 1 (since $N = 4$ implies $(N - 2)/2 = 1$). Hence, matrix $A(i, j)$ is not positive semi-definite in those cases, leading to negative variance estimates.

Acknowledgement

The authors gratefully acknowledge the anonymous reviewer for the insightful comments and constructive suggestions, which significantly improved the clarity and presentation of the manuscript.

Conflict of interest

The authors do not have any financial or non-financial conflict of interest to declare for the research work included in this article.

References

- Lanke, J. (1975). *Some Contributions to the Theory of Survey Sampling*. Unpublished Doctoral Thesis, Lund University, Sweden.
- Pal, M. and Sinha, B. K. (2024). *Selected Topics in Statistical Inference: Theory and Applications*. Springer.
- Prabhu-Ajgaonkar, S. (1967). The effect of increasing sample size on the precision of an estimator. *American Statistician*, **21**, 26–28.
- Yates, F. and Grundy, P. (1953). Selection without replacement from within strata with probability proportional to size. *Journal of the Royal Statistical Society. Series B*, **15**, 253–261.

Publisher
Society of Statistics, Computer and Applications
Registered Office: I-1703, Chittaranjan Park, New Delhi-110019, INDIA
Mailing Address: B-133, Ground Floor, C.R. Park, New Delhi-110019, INDIA
Tele: 011-40517662
<https://ssca.org.in/>
statapp1999@gmail.com
2025

Printed by : Galaxy Studio & Graphics
Mob: +91 9818 35 2203, +91 9582 94 1203