Statistics and Applications {ISSN 2452-7395 (online)}
Volume 18, No. 2, 2020 (New Series), pp 369–381

Statistical Validity and Consistency of Big Data Analytics: A General Framework

Bikram Karmakar[1] **and Indranil Mukhopadhyay**[2]
[1] *Department Statistics, University of Pennsylvania, PA, USA*
[2*] *Human Genetics Unit, Indian Statistical Institute, India*

Received: 31 July 2020; Revised: 30 September 2020; Accepted: 05 October 2020

Abstract

Informatics and technological advancements have triggered generation of huge volume of data with varied complexity in its management and analysis. Big Data analytics is the practice of revealing hidden aspects of such data and making inferences from it. Although storage, retrieval and management of Big Data seem possible through efficient algorithm and system development, concern about statistical consistency remains to be addressed in view of its specific characteristics. Since Big Data does not conform to standard analytics, we need proper modification of the existing statistical theory and tools. Here we propose, with illustrations, a general statistical framework and an algorithmic principle for Big Data analytics that ensure statistical accuracy of the conclusions. The proposed framework has the potential to push forward advancement of Big Data analytics in the right direction. The partition-repetition approach proposed here is broad enough to encompass all practical data analytic problems.

Key words: Big Data; Data mining; Partition-repetition; Statistical inference.

AMS Subject Classifications: 62K05, 05B05

1. Introduction

'Big Data' presents itself with unique challenges in retrieving, storing and all the way to analysing the data. Technological breakthrough makes generation and collection of huge volume of data possible in many fields like genetics, genomics, health care, customer service, informatics, to name a few. Among various challenges presented by the abundance of data, analysis of the data is a well recognized hurdle. While the explosion of information allows us to know more about the process, appropriate methods or algorithms are essential to make 'correct' inference or to reveal hidden patterns.

Recent advancements of technology and targeted methods to Big Data analytics give access to ample capacity for storing the data along with the skill of parallel computing. Much effort has been dedicated to extract information from Big Data in an efficient manner. From a practical standpoint, concern remains about the validity of results from analysis of Big Data. As attested by many recent articles, in most cases the inference based on such

Corresponding Author: Indranil Mukhopadhyay
Email: indranil@isical.ac.in

data is unacceptable and unreliable. For example, High dimension conventional classification methods are no better than random guesses (Fan and Fan, 2008). Understanding the output of Big Data analytics than to fixate on the technical aspect of it is the most important issue (Fan *et al.*, 2014; Fan *et al.*, 2018; Huang, 2014), because the future decision making process depends only on this output. The aim of this article is to put forward a framework in order to establish the acceptability of the learning from the Big Data. This framework also fits to the paradigm of parallel computing and at the same time provides a robust statistical basis for practical application.

The classical statistical theory of data analysis has its roots in axioms of probability theory. By some arguments data analysis is more complex than physics, biology and even behavioural science. Formal statistics developed so far can help to tackle the analytics but it will produce realistic results only if we can keep the basic assumptions loose (Tukey, 1962). The velocity of data flow in today's world makes it more challenging for producing meaningful conclusion over time on the same problem (Efron, 2020). Naturally it is not possible to analyze the entire data at the same instance when we have so little time to produce results. Also, subsequent results can make a previous conclusion redundant. With growing complexity of Big Data, statistical theory needs to be revisited (Davidian, 2013), mainly due to the violation of probabilistic independence or exchangeability conditions. Statistics community has raised concerns about how the sound and carefully developed theory can help build a structure around it. Implicitly classical statistics is already equipped with basic mechanisms to deal with big data. Sampling and sufficiency, among other core discoveries of statistics, are extremely useful in analysis of data of large volume, a characteristic of big data (Donoho, 2017). But the analysis of such high volume data needs to be done in presence of variety and high velocity, the two main characteristics of big data. Algorithmic or computational innovations for parallel computing are not the entire solution, but they are important tools when coupled with appropriate statistical methods in order to utilise the entire available information (sufficiency) that is contained in the data flow (large sample) (Donoho, 2017). In this article we exploit an algorithmic architecture used in practice to tackle Big Data and suggest an appropriate mathematical ground for analysis of such architecture.

We propose a partition and repetition approach in a general framework for statistical analysis of Big Data. This approach expands the horizon of standard statistical methods as well as opens new avenues for novel methods to encompass and tackle the challenges arisen due to the specific characteristics of Big Data. With the help of this general framework, we prove consistency and accuracy of the analytic results thus obtained. We have explained this theory through various examples that are usually required in common data analysis paradigm in respect of many fields. We hope that such a framework would help in further development of Big Data analytics. Note that although here we mainly address the problems due to volume, velocity and variety, the 3-V's that occur simultaneously in a typical Big Data problem, the other 3-V's–veracity, validity, and volatility–must always be taken care of in any statistical analysis. A statistical model has to deal with inherent variability which is nothing but volatility especially when considered with respect to time. It is natural that with available data at hand, we have to model the variance and hence the volatility, how it is related with time or other auxiliary variables. This study of volatility would be inbuilt in our proposed framework. Similarly veracity and validity must be ensured properly with appropriate strategies, for example, using back-testing, training-testing protocol *etc.* Our proposed framework for the basis of data analytics in the Big Data paradigm considers many possible statistical fundamentals and validity requirements when analysing data with the above characteristics.

2. Developing the general framework

2.1. The divide and conquer algorithm

Abundance of digital information is one way to explain what we today understand as 'Big Data'. There are two aspects to the story. Firstly, human intuition suggests that accuracy of the answer to our question increases if we have more and more information. This intuition works backward; we start with a question, try to comprehend what data we might need to answer the question and then realize that relevant information exists somewhere in digitized format. The catch is that, this retrospective thought process assumes that the skill by which human intelligence finds this answer from the data is transferable to mechanical and algorithmic computing. Secondly, with huge volume of data we can find a question of interest from the data itself and then get the answer to the question. But the inherent complexity of available data makes this task difficult. This whole process is advertised as Big Data analytics.

Principle characteristics of Big Data are its volume, velocity, variety and complexity (Katal, 2013). All of them presents as unique challenges at a technical level of dealing with the data. At the hardware level we have reached a saturation point on the achievable clock pulse on a single processor. Rather, the growth in computing capacity is attained by increasing the number of threaded cores. Moreover, while storage capacity is fairly cheap and scalable, the Random Access Memory is not so. Recognizing this hardware restriction the state of the art algorithms (Hadoop, Amazon EC2) for Big Data analytics has adopted a partitioning based method.

However, in view of advancements in computing systems including storage and processing, need for new data analytic tools are required that are adaptive to new technologies (Petcu *et al.*, 2015). Building such statistical tools and algorithms for monitoring and analysis is needed to achieve success in Big Data analytics. Hence standard statistical methods should be revisited, modified, and validated in the light of scalability to extremely large scale data applications (Reed and Dongarra, 2015).

Fisher *et al.* (2012) have identified the standard workflow of data analysis as, (1) acquiring data, (2) choosing an architecture, (3) shaping the data to the architecture, (4) writing and editing the code, and (5) reflecting and iterating on the results. The initial struggle is to adopt a suitable architecture for the data and map the collected data to that architecture. In this article, we are not focusing on this domain of analytics job. Rather the focus is on the later part of analysing the data. To address the problem of huge volume of data, the way is to partition it into small portions that are manageable by the Random Access Memory, process the data in a parallel manner, and finally combine the processed information to produce the final output. This idea of partitioning has been used, although in a subtle way, in other areas of research, *e.g.*, data mining (Buehrer *et al.*, 2015; Calders *et al.*, 2010), Markov Chain Monte Carlo (Wang *et al.*, 2015). An extra benefit of this divide and conquer method is that such an algorithm easily adapts to the velocity of Big Data. Velocity contributes to new partitions which are to be analysed and then the inference is to be combined with the earlier output (Schifano *et al.*, 2016). The other issues relating to variety and complexity are taken into account by the statistical methods and algorithms that are used in the analysis.

2.2. The framework

Sample space structure: Classical theory of statistical analysis is a well developed area with sound theories. To establish a framework for Big Data analytics we naturally would like to fall back on those works. To begin with, we consider a sample space $(\mathcal{S}, \mathcal{A}_{\mathcal{S}})$ where $\mathcal{S}$ is the space of realized values of the data and $\mathcal{A}_{\mathcal{S}}$ is the sigma field associated with the sample space. We denote by $\mathcal{M}(\mathcal{S})$ the set of probability measures on $(\mathcal{S}, \mathcal{A}_{\mathcal{S}})$. Also let $\mathcal{M}^e(\mathcal{S})\Big(\subset \mathcal{M}(\mathcal{S})\Big)$ be the set of probability measures with finite support. An observed data $X_{n\times p}$ can be identified by a probability measure m_X on $(\mathcal{S}, \mathcal{A}_{\mathcal{S}})$, with a support having finite cardinality, defined as follows,

$$m_X(A) = \sum_{x_i\in A, i=1}^{n} \frac{1}{n},$$

for any $A \subseteq \mathcal{A}_{\mathcal{S}}$ and $x_i\,(i = 1, \ldots, n)$ is the i-th data point. To build a theory around it we would require a suitable metric on the space $\mathcal{M}(\mathcal{S})$. For example, if $(\mathcal{S}, \mathcal{A}_{\mathcal{S}})$ is a polish space then with Prokhorov metric $(d_{\mathcal{M}})$ we can put weak convergence on $\mathcal{M}(\mathcal{S})$.

Till this point we have not considered any aspect of Big Data par se. Our aim is to build the ideology of Big Data analytics on this sample space structure. Identification of the realized data with an empirical measure on some sample space gives a broader ground to work on. In a Big Data set up, we hardly have any control on the generation of data. Thus unlike in classical statistical theory, where mostly we want to build better experimental designs to apply statistical methods, be it standard or novel, here we want to construct an algorithm that would work with the data generation process. This difference in approaches is subtle but central to how these two ideologies differ.

The problem approach: Main goal of Big Data analytics is to extract information from the data, which is equivalent to getting information from an element in $\mathcal{M}^e(\mathcal{S})$. So we assume that a satisfactory data collection and mapping architecture exists. To develop a full framework, we introduce some definitions about functionality of data analysis. This is necessary to avoid the cumbersome details and technicalities of a particular scenario.

Extracted information of a data analysis can be viewed as an element in the result space $(\mathcal{R})$. A *problem approach* (ρ) is a function from $\mathcal{M}^e(\mathcal{S})$ to $\mathcal{R}$. Based on this formulation of problem approach we can consider two classes of problem approaches as follows.

Definition 1: *Inference Problem*: If the problem approach ρ can be extended to a strictly larger subset of $\mathcal{M}(\mathcal{S})$ than $\mathcal{M}^e(\mathcal{S})$, then such a problem or problem approach is called an inference problem.

Definition 2: *Mining Problem*: If the problem approach ρ can only be defined on $\mathcal{M}^e(\mathcal{S})$, then such a problem or problem approach is called a mining problem.

The usual examples of these two classes of problems are as follows. Parametric estimation and testing problems fall under the class of inference problems where the subset of $\mathcal{M}(\mathcal{S})$ under consideration is $\mathcal{M}^e(\mathcal{S})$ along with the parameter models. Clustering problem or outlier detection problem, on the other hand, are under the class of mining problems. In later sections, we shall discuss both these classes of problem approaches and their solutions in more details.

A technical assumption we need to have is that, one such problem approach is *viable* if the map

$$\rho : \Big(\rho^{-1}(\mathcal{R}), d_{\mathcal{M}}\Big) \longrightarrow \Big(\mathcal{R}, d_{\mathcal{R}}\Big) \tag{1}$$

is a continuous map, where $d_{\mathcal{M}}$ and $d_{\mathcal{R}}$ are appropriate metrics on respective spaces. A *viable problem approach (ρ)* then ensures that the problem is consistent in the number of samples and robust in the data points. This means that slight change in the data generation process ($\mathcal{M}(\mathcal{S})$) should not create substantial difference in the result ($\mathcal{R}$). Here consistency indicates the large sample property of converging results as the number of data points increases whereas robustness indicates very little or not significant change in the results from two data sets that are not too different from each other.

The existence of ρ has important implications both in statistical modelling with an underlying stochastic data generation model and also in algorithm modelling with unknown data mechanism (Breiman, 2001). We only emphasis that ρ should be judiciously chosen and it has no conflict with the "two cultures" of statistical modelling (Breiman, 2001). However, in any case, we assume that there is an underlying σ-field behind the data generation process, be it known or unknown, and hence ρ is well defined. The existence of ρ is essential for establishing the sound framework for Big Data analytics that we establish through two theorems in the next section.

2.3. Big Data Algorithm

We now discuss various components of our proposed algorithmic structure of Big Data analytics.

Partitioning: A naturally accepted strategy in analysing huge volume of data is to consider small parts of data at a time. Our formulation for Big Data analytics formulates this method of partitioning the data as a functional,

$$\begin{aligned} &H_L : \mathcal{M}^e(\mathcal{S}) \longrightarrow \mathcal{M}^e(\mathcal{S}) \times \cdots \times \mathcal{M}^e(\mathcal{S}) \quad (L \textit{ such } \mathcal{M}^e(S)\, s) \\ &H_L(m) = (m_1, m_2, \ldots, m_L), \end{aligned} \tag{2}$$

such that $(m_1, m_2, \ldots, m_L)$ is related to m by,

$$\begin{aligned} &supp(m) = \overset{L}{\underset{i=1}{\cup}} supp(m_i); \\ &supp(m_i) \cap supp(m_j) = \emptyset, \;\; 1 \leq i \neq j \leq L. \end{aligned} \tag{3}$$

where $supp(m)$ denotes the support set of m and $\emptyset$ denotes the empty set.

For convenience we write $supp(m_i) = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_{n_i}^{(i)}\right)$ for each i. For a fixed data m (or $m \equiv X$) we would be given a problem approach ρ. Then the divide and conquer strategy would choose a partitioning functional H_L.

But to reduce the error in result due to partitioning, the strategy is to repeat $K(> 1)$ times the partitioning; denote them by $H_{L,1}, H_{L,2}, \ldots, H_{L,K}$. This type of algorithm we call as the *partition-repetition algorithm*. We now formulate this partition-repetition algorithm in a comfortable manner.

Let $\mathcal{H}_L$ be the set of all partitioning functionals H_L. A σ-field $\mathcal{A}_{\mathcal{H}_L}$ can be defined as the smallest σ-field on $\mathcal{H}_L$ such that the functions $f_{i,j}(\cdot)$ on $\left(\mathcal{H}_L, \mathcal{A}_{\mathcal{H}_L}\right)$ to $(\mathcal{S}, \mathcal{A}_{\mathcal{S}})$ are measurable for any choice of $m \in \mathcal{M}^e(\mathcal{S})$, where

$$f_{i,j}\Big(H_L(m)\Big) = x_j^{(i)} \;\; j = 1, 2, \ldots, n_i; \; i = 1, 2, \ldots, L.$$

Then the strategy of analysing data of unmanageable size, in terms of volume, variety and most importantly velocity, by partition-repetition algorithm can be understood as a probability measure P_{H_L} on the measurable space $\left(\mathcal{H}_L, \mathcal{A}_{\mathcal{H}_L}\right)$. More precisely $\{H_{L,1}, H_{L,2}, \ldots, H_{L,K}\}$ would be viewed as a random sample from the probability measure space $\left(\mathcal{H}_L, \mathcal{A}_{\mathcal{H}_L}, P_{H_L}\right)$. For simplicity of notation let us denote by ρ^L the map,

$$\rho^L : (m_1, \ldots, m_L) \longmapsto (\rho(m_1), \ldots, \rho(m_L)) \;\; for \; m_i \in \mathcal{M}^e(\mathcal{S});$$

for $i = 1, 2, \ldots, L$. Then a single random sample H_L from the probability distribution P_{H_L} provides us L results $\rho^L\Big(H_L(m)\Big)$, which are L elements from $\mathcal{R}$. With a random sample $H_{L,1}, H_{L,2}, \ldots, H_{L,K}$ from the distribution, the set of results we get using the problem approach ρ is

$$\begin{aligned} \{R^*_{k,l}\}_{k=1,2,\ldots,K;\, l=1,2,\ldots,L} &= \left\{R^*_{k,1}, R^*_{k,2}, \ldots, R^*_{k,L}\right\}_{k=1,2,\ldots,K} \\ &= \left\{\rho^L\Big(H_{L,k}(m)\Big)\right\}_{k=1,2,\ldots,K}. \end{aligned}$$

This framework also encompasses the case where rather than partitioning one chooses to sub-sample. In that case we would get rid of the extra restriction in equation (3) on the functional H_L. Popular algorithms of Bootstrap and Bag-of-Little-Bootstraps (Kleiner *et al.*, 2014) are covered in this framework.

Combining: Next critical part of the algorithm is combining the results obtained above, $\left\{R^*_{k,l}\right\}_{k=1,2,\ldots,K;\, l=1,2,\ldots,L}$ in order to arrive at a final result. Let C_{KL} be the combining map that takes all the results from the collection and gives the final result. The triplet $\left(\rho, P_{H_L}, C_{KL}\right)$ can be called a *solution* to a Big Data problem.

Now it remains to understand the viability of the solution. We have put a stable condition of continuity in equation (1) on problem approach ρ as a *viable problem approach*. Proper behaviour of the pair $\left(P_{H_L}, C_{KL}\right)$ would ensure an accurate solution to the problem ρ for m.

We focus on the case where $C_{KL} := C_K^2 \circ C_L^1$ works in two stages. In the first stage C_L^1 works on each partition (k) to collect the results

$$R_k^* := C_L^1\Big(\{R_{kl}^*\}_{l=1,2,\dots,L}\Big) \quad for\ k = 1, 2, \cdots, K.$$

This K-tuple is combined by C_K^2. For a fixed data m when C_L^1 is a measurable map, the randomness of $\{H_{L,1}, H_{L,2}, \dots, H_{L,K}\}$ makes the collection $\{R_1^*, R_2^*, \dots, R_K^*\}$ an independently and identically distributed (i.i.d.) sample on the measure space $\Big(\mathcal{R}, \mathcal{A}_{\mathcal{R}}\Big)$. This formulation of the solution $\Big(\rho, P_{H_L}, C_K^2 \circ C_L^1\Big)$ provides an opportunity to use rich statistical theory in data analytics.

In the general case, the result space can be quite complicated (we shall give concrete examples in later section). Rather than dealing with the space $\mathcal{R}$ itself it would be better to work with real numbers. This is achieved by an evaluation function $ev : \mathcal{R} \longrightarrow \mathbb{R}^N$ for some fixed N belonging to the set of natural integers. Then, viability of the choice of P_{H_L} can be understood using the evaluation function of the result space $\mathcal{R}$. For a given data m and a problem approach ρ, we call a partitioning probability measure P_{H_L} to be *viable* under the first stage combining operator C_L^1 if,

$$\int ev \circ C_L^1\Big(\rho^L(H_L(m))\Big) dP_{H_L} = ev \circ \rho(m). \tag{4}$$

This condition means that the probability measure P_{H_L} and the combining method C_L^1 are compatible with each other for the problem ρ. If we do infinitely many repetitions of our partition-repetition based algorithm, the combining method C_L^1 will give equivalent performance as the one we would have got if we could apply ρ on the data m.

The second stage of combining method C_K^2 operates on the collection of first stage result by combining $R_1^*, R_2^*, \dots, R_K^*$ to get the solution

$$R_K^{**} := C_K^2\Big(\{R_k^*\}_{k=1,2,\dots,K}\Big).$$

Now the viability of C_K^2 is based on the comparison of R_K^{**} with $\rho(m) = R^*$ (say). Here we present the soundness of the algorithm of partitioning and combining through the following theorem.

Theorem 1: For a Big Data solution $\Big(\rho, P_{H_L}, C_K^2 \circ C_L^1\Big)$, if P_{H_L} is a viable partitioning method under combining method C_L^1 (*i.e.*, equation (4) is satisfied) and convergence in ev is equivalent to that of in $\mathcal{R}$, then there exists a second stage combining method C_K^2, such that $R_K^{**} \longrightarrow R^*$ almost surely in P_{H_L}.

Proof: Define C_K^2 on $\mathcal{R} \times \mathcal{R} \times \cdots \times \mathcal{R}$ (K times) as follows,

$$C_K^2(R_1, R_2, \dots, R_K) := \arg \min_{\{R_k\}_{k=1,2,\dots,K}} \Big|\Big| ev \circ R_i - ev \circ R^* \Big|\Big|.$$

Let us use the notations $Y_k = ev \circ R_k^*$, $Z_K = ev \circ R_K^{**}$ and $\mu = ev \circ R^*$. Since $\{R_k^*\}_{k \geq 1}$ is an i.i.d. sample, by strong law of large numbers as equation (4) holds, for all $\epsilon > 0$ with

$P_{\mathcal{R}} := P_{H_L} \circ C_L^{1^{-1}} \circ ev^{-1}$,

$$P_{\mathcal{R}}\Big(\cup_{k_0=1}^{\infty} \cap_{K \geq k_0} \Big(\Big|\Big| \frac{1}{K} \sum_{k=1}^{K} Y_k - \mu \Big|\Big| < \epsilon \Big) \Big) = 0.$$

Now using the fact that $|| \sum_{k=1}^{K} Y_k/K - \mu|| \geq ||Z_K - \mu||$ and definition of C_K^2, the above holds with $\sum_{k=1}^{K} Y_k/K$ replaced by Z_K. Since convergence in $(\mathcal{R}, d_{\mathcal{R}})$ is equivalent to that in $\Big(ev \circ \mathcal{R}, || \cdot ||\Big)$, rest of the argument follows as by assumption convergence in $(\mathcal{R}, d_{\mathcal{R}})$ is equivalent to that in $\Big(ev \circ \mathcal{R}, || \cdot ||\Big)$. □

The theorem above deals with the volume aspect of Big Data. It says that even if the data is unmanageable to be processed practically, we can adopt partition-repetition approach to get a good solution. It is also not passed our attention that the number of combination rules may be more than two, but the final convergence of results requires some more assumptions and strong theorems in the dependence set up.

Next we also need to answer the question which is more of classical statistical in nature. If the velocity of the data provides us more and more information of specific form, is the partition-repetition algorithm able to extract that information? The following theorem tells us if that is the case, we would be able to choose a partitioning measure and a sequence of combining methods that gives the final result.

Theorem 2: Let $\{m_n\}_{n\geq 1} \in \mathcal{M}^e(\mathcal{S})$ and $m \in$ *domain of* ρ. Suppose the problem approach ρ is viable on its domain and $m_n \longrightarrow m$. If conditions of Theorem 1 hold for the sequence of solutions $\Big(\rho, P_{H_L,n}, C_{K,n}^2 \circ C_{L,n}^1\Big)$, then there exists a sequence of integers $\{k_n : n \geq 1\}$ and a P_{H_L} such that, for $n \geq 1$, $P_{H_L,n}$ is absolutely continuous with respect to P_{H_L} with

$$\Big|\Big| ev \circ C_{k_n,n}^2 \circ C_{L,n}^1 \Big\{ \rho^L (H_{L,k}(m_n)) \Big\}_{k=1,2,\ldots,K} - ev \circ \rho(m) \Big|\Big| \longrightarrow 0,$$

as $n \to \infty$ almost surely in $P_{\mathcal{R}}$.

Proof: Define $P_{H_L}(\cdot) = \sum_{n=1}^{\infty} P_{H_L,n}(\cdot)/2^n$. Let us denote,

$$R_{K,n}^{**} = C_{K,n}^2 \circ C_{L,n}^1 \Big(\Big\{ \rho^L \Big(H_{L,k}(m_n) \Big) \Big\}_{k=1,2,\ldots,K} \Big).$$

Then for every $\epsilon (> 0)$, by Theorem 1 and equation (1) there exists a sequence $\{k_n(\epsilon) : n \geq 1\}$ and $N \geq 1$ such that for all $n \geq N$,

$$\Big|\Big| ev \circ R_{k_n(\epsilon),n}^{**} - ev \circ \rho(m) \Big|\Big| < \frac{\epsilon}{2^n},$$

almost surely in $P_{\mathcal{R},n} = P_{H_L,n} \circ C_{L,n}^{1^{-1}} \circ ev^{-1}$. Choosing ϵ as rationals, result follows from Cantor's diagonal argument. □

Both these results are of existential nature rather than being instructive for practice. Although little abstract in their formulation, these theorems form the basis of the methods that would be applied in practice. Study on combining methods is not new to statistics. This framework enforces the importance of various combining methods along with partitioning methods in the light of Big Data analytics.

The power of this kind of theory is that we do not put any hard and fast regularity condition on the data or the data generation process. Theorem 2 only requires that the data collected eventually amounts to some specific information.

3. Illustrative Examples

An analyst's job and a statistician's work differ in a crucial way. An analyst is more concerned with how to extract information from the data available. This work is referred to as number crunching. A statistician is concerned about the quality of the extracted information sometimes taking for granted the effort of extracting the information. In a Big Data scenario where importance of analyst's job comes more into the limelight, a statistician could provide support by accepting some compromise on their ideology. In this section we illustrate the formulation developed above through some standard data analytic problems.

We first consider a few problems where the solution $\rho(m_n)$ can be calculated without any error from partitioning based algorithm. Here we specify by subscript n the size of the data. In these examples it is enough to consider P_{H_L} to be some degenerate probability distribution of convenience and we only require a single sample ($K = 1$) from it.

Calculating sample mean: Here P_{H_L} can be any distribution that partitions the data into manageable balanced pieces. Then for $\rho(m_n) := (\int x\, dm_n, n)$ the combining method shall be,

$$C_L^1(\{(\bar{x}_i, n_i)\}_{i=1,2,\ldots,L}) = \left(\frac{\sum_i n_i \bar{x}_i}{\sum_i n_i}, \sum_i n_i\right).$$

A little tweak in these definitions allows us to calculate many other descriptive statistics like weighted means, dispersion measures and also some robust measures for central tendency.

Sorting: To get a Big Data solution to the sorting problem we can define a partitioning P_{H_L} as a degenerate distribution such that it divides the data m_n into L parts based on a sequence $bound_0 < bound_1 < \cdots < bound_L$ as,

$$bound_{i-1} \leq \{x_j^{(i)}\} < bound_i \;\; for \; i = 1, 2, \cdots, L.$$

The choice of the sequence $\{bound_i\}$ should be such that the individual parts are of manageable sizes. With ρ providing us with a sorted array, the combining stage should simply concatenate the ordered parts, *i.e.*,

$$C_L^1(\{R_l^*\}_{l=1,2,\ldots,L}) := (R_1^*, R_2^*, \cdots, R_L^*).$$

Similar solutions of the above type are obvious for problems like searching, calculating extreme statistics $(x_{(1)}, x_{(n)})$, constructing a histogram *etc.* Most of the time these simple problems are only intermediate steps towards more challenging problems of data analytics.

Some solutions to more standard problems of Big Data analytics are discussed in brief below. First few examples are inference problems while the later ones are mining problems.

We assume that the data are cleaned and dressed for the purpose at hand. We avoid discussing the technical aspects of implementing these algorithms in practice, though in a few examples we shall provide references to available literature that has more focus on detailed analysis of the algorithms.

Estimation: The problems of modelling (nonparametric, parametric, time series or even Bayesian) come under the radar of inference problem. Based on the requirements of the solution (*e.g.*, unbiasedness, minimum variance, consistency) there would be different Big Data solutions to the problem approach ρ. Many of the times it suffices to consider P_{H_L} as a random partitioning measure of the data, although while considering spatial and/or temporal data more clever partitioning measure would be required to satisfy viability condition like equation (4).

Let us consider the problem of finding maximum likelihood estimate for a parameter based on some algorithm (say, Expectation-Maximization algorithm or Newton-Raphson or Fisher's Scoring *etc.*). The scenario is that, we have a statistical model in mind where the number of parameters is fixed. Then partitioning the data simply breaks the objective function (log-likelihood function) into L parts. Consequently an intuitive choice of the combining method C_{KL} would be whichever of the results from partitions maximizes the whole objective function. Although this method does not ensure the MLE for the data, but in practice we are hardly concerned about theoretical properties like efficiency; the estimate found by this method is acceptable.

Testing: Consider a test function ρ that provides p-value for testing H_0 against H_1. Then based on random partitioning of the data into balanced parts, a conservative combining algorithm (Tippett, 1931) for the corresponding solution can be

$$R_k^* := C_L^1\Big(\{R_{lk}^*\}_{l=1,2,\dots,L}\Big) = \min_{l=1,2,\dots,L} R_{lk}^*, \text{for } k = 1, 2, \dots, K,$$

and

$$R^{**} := C_K^2\Big(\{R_k^*\}_{k=1,2,\dots,K}\Big) = median\{R_k^*\}_{k=1,2,\dots,K}.$$

A large part of recent statistical methods literature focuses on the regime of $p >> n$. Even when the data is not formally a Big data, because it does not inherit the various characteristics discussed in the introduction, the data set can be in this regime; for example, genome sequencing data. Note that the above discussion also encompasses the scenario when $p >> n$. Depending on the testing problem, if we get a p-value or a test statistic for each partitioned dataset, the solution is immediate in the proposed framework. It is valuable to consider specific problems in this regime in depth as they can be helpful to solve important problems in the relevant fields. But data sets solely of the $p >> n$ variety arguably represent a small part of Big data as we consider here.

Variable Selection: The context in which variable selection problem has been addressed in recent literature is sometimes too idealistic for Big Data paradigm, although there are some promising methods. The data generation process is assumed to provide information on a set of response variables and a fixed set of regressors. We might be interested in a subset of these variables which have effect on the responses. The quality of the selected variables can be assessed by proportions of the variables wrongly selected. In a situation where assumption of homoscedastic uncorrelated linear model is valid, Barbar and Candes (2015) proposed a method to select variables with a control on the proportion of falsely discovered variables. This method is no doubt computationally heavy. The partition-repetition philosophy can be used to adapt this algorithm to achieve the same goal in current context.

If the data generation process is well controlled, the above inference problems and solutions make sense. Some recent works are available in the area of regression (Battey *et al.*, 2015; Chen and Xie, 2014) focusing on divide and conquer methods. Unfortunately spurious correlations, noisy data etc. are very common in Big Data perspective. In that case these naive solutions can be hugely mis-representative of the actual truth. Data mining problems are more relevant in such a scenario. In a mining problem we are interested in the data itself without having to make any modelling assumption. Possible Big Data solutions to a few mining problems are discussed below.

Clustering: An elaborate and critical discussion on clustering problem in view of Big Data analytics can be found in recent article by the authors (Karmakar and Mukhopadhyay, 2016; Karmakar *et al.*, 2019). Karmakar and Mukhopadhyay (2016) provide a detailed example illustrating how the proposed framework fits to the class of Big Data clustering problems where it (a) demonstrates existence and evaluation of the required quantities and (b) proves validity of the final result. In brief, the combing method would identify the unique clusters from the set $\{R^*_{lk}\}_{l=1,2,\ldots,L}$ based on a decision function that tells us to combine two results when they seem to form a single data cloud. The second stage is to make stable clusters based on some measure from the K sets of clusterings $\{R^*_k\}_{k=1,2,\ldots,K}$.

Outliers Detection: Based on a random partitioning measure P_{H_L} and a problem approach ρ that separates the outliers (m^o_n) and the data (m^d_n) section, (i.e., $\rho(m_n) := (m^d_n, m^o_n)$), the combining method C^1_L would check the structure of the outliers from the individual parts and get the outliers from the whole part. The method should check if outliers from one part belongs to the data section of some other part and also if outliers from all the parts together form some data section. Second stage of combining would then pick out the stable outliers from all repetitions.

Ramaswamy *et al.* (2000) discuss another Big Data solution to this mining problem based on a different partitioning method based on clustering the data and van Stein *et al.* (2016) propose local subspace-based solution to outlier detection problem, which applies a combining strategy using global neighbourhoods. These methods can be viewed as special cases of our proposed framework.

Classification: First we consider the k-Nearest Neighbor classifier, where ρ finds the k nearest neighbours of a test data point (x) as,

$$\begin{aligned}
&\rho(m_n) := ((x_{(i)}, d(x, x_{(i)}))_{i=1,2,\ldots,k}) \\
\text{such that } \ & d(x, x_{(1)}) \leq \cdots \leq d(x, x_{(k)}) \\
&\leq min\{d(x, x_i); x_i \in X \setminus \{x_{(1)}, x_{(2)}, \ldots, x_{(k)}\}\}.
\end{aligned}$$

Based on any partitioning P_{H_L}, then the problem is exactly solvable in a single repetition with a combining operator that picks the k data points nearest to x among the $L \times k$ points.

Subsequently the classifier is contracted on a second algorithm that simply checks for the maximum number of representatives in these k data points from each of the classes.

Another celebrated class of classifiers is decision trees. A relevant combining operator of decision trees based on partition of the data is proposed by Hall *et al.* (1998).

4. Discussion

Data is the lubricant that drives the machinery of statistics. It is no longer a topic of debate that the way data is generated and collected in modern times is drastically different from what statisticians are used to deal with. Statistics should adapt to this change and thereby assist the masses of data analytic work.

The main contribution of this article is suggesting a basis of statistical theory for present day data analytic works. In composing the theory we have tried to stay true to the practical nature of a data science job. This formulation proposes a divide and conquer algorithm (either partition-repetition or subsampling method). More importantly it respects the fact that more often than not we have no control on the data generation process. We have also tried to encompass all possible data analytic problems. A range of such data analytic problems are discussed in perspective of our formulation.

5. Conclusion

Successful use of statistical theory in data analysis would require understanding the field of 'Big Data'. Rather than being insistent on developing methods and elaborate theories based on idealistic assumptions, we have kept their applicability in mind. Our proposed framework encompasses statistical analyses of majority of problems in view of complex characteristics of Big Data and can be extended further keeping its compatibility with modern advances in computational world.

References

Barber, R. F. Candes, E. (2015). Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, **43(5)**, 2055–2085. doi:10.1214/15-AOS1337.

Battey, H., Fan, F., Liu, H., Lu, J. and Zhu, Z. (2015). Distributed estimation and inference with statistical guarantees. *arXiv preprint:1509.05457v1.*

Breiman, L. (2001). The statistical modelling: the two cultures, *Statistical Science*, **16(3)**, 199-231.

Buehrer, G., de Oliveira, R. L., Fuhry, D. and Parthasarathy, S. (2015). Towards a parameter-free and parallel itemset mining algorithm in linearithmic time. *2015 IEEE 31st International Conference on Data Engineering, Seoul*, 1071–1082.

Calders, T., Garboni, C. and Goethals, B. (2010). Efficient pattern mining of uncertain data with sampling. In M. J. Zaki, J. X. Yu, B. Ravindran, V. Pudi (Eds.), *PAKDD 2010, Part I. LNCS (LNAI)*, **6118**, Springer, Heidelberg, 480–487. doi:10.1007/978-3-642-13657-3_51.

Chen, X. and Xie, M. (2014). A split-and-conquer approach for analysis of extraordinarily large data. *Statistica Sinica*, **24**, 1655–1684. doi:10.5705/ss.2013.088.

Davidian, M. (2013). Aren't *we* data science. *Amstat News*, 433-435.

Donoho, D. (2017). 50 Years of data science. *Journal of Computational and Graphical Statistics*, **26(4)**, 745–766. doi: 10.1080/10618600.2017.1384734.

Efron, B. (2020). Prediction, estimation, and attribution. *Journal of theAmerican Statistical Association*, **115**, 636–655,

Fan, J. and Fan, Y. (2008). High dimensional classification using features annealed independence rules. *The Annals of Statistics*, **36**, 2605–2637. doi:10.1214/07-AOS504.

Fan, J. Fang, H. and Han, L. (2014). Challenges of big data analysis. *National Science Review*, **1(2)**, 293–314. https://doi.org/10.1093/nsr/nwt032.

Fan, J., Shao, Q. M. and Zhou, W. X. (2018). Are discoveries spurious? Distributions of maximum spurious correlations and their applications. *The Annals of Statistics*, **46(3)**, 989–1017.

Fisher, D., DeLine, R., Czerwinski, M. and Drucker, S. (2012). Interactions with big data analytics. *Interactions with Big Data Analysis*, **19(3)**, 50–59.

Hall, L., Chawla, N. and Bowyer, K. W. (1998). Combining decision trees learned in parallel. In *Working Notes of the KDD-97 Workshop on Distributed Data Mining*, 10–15.

Huang, J. P. (2014). Big data need physical ideas and methods. *arXiv preprint:1412.6848.*

Katal, A., Wazid, M. and Goudar, R. H. (2013). Big data: Issues, challenges, tools and good practices. In *2013 Sixth International Conference on Contemporary Computing (IC3)*, 404–409. doi:10.1109/IC3.2013.6612229.

Karmakar, B., Das, S., Bhattacharya, S., Sarkar, R. and Mukhopadhyay, I. (2019). Tight clustering in Big Data Analytics with focus on microarray data. *Scirntific Reports*, **9:3053**, 1-12.

Karmakar, B. and Mukhopadhyay, I. (2016). An efficient partition – repetition approach in clustering of big data. In S. Pyne, B.L.S. Prakasa Rao, S.B. Rao (Eds.), *Big data analytics: Methods and applications*, Springer India, New Delhi, 75–93. doi:10.1007/978-81-322-3628-3_5.

Kleiner, A., Talwalkar, A., Sarkar, P. and Jordan, M. I. (2014). A scalable bootstrap for massive data. *Journal of Royal Statistical Society Series B*, **76**, 795–816. doi:10.1111/rssb.12050.

Petcu, D. et al.. (2015). On processing extreme data. *Scalable Computing: Practice and Experience*, **16(4)**, 467–489.

Ramaswamy, S., Rastogi, R. and Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. *ACM Sigmod Records*, **29(2)**, 427–438. doi:10.1145/335191.335437.

Reed, D. A. and Dongarra, J. (2015). Exascale computing and big data. *Communications of the ACM*, **58(7)**, 56–68. doi:10.1145/2699414.

Schifano, E. D., Wu, J., Wang, C., Yan, J. and Chen, M. H. (2016). Online updating of statistical inference in the Big Data setting. *Technometrics*, **58(3)**, 393-403.

Tippett, L. H. C. (1931). The methods of statistics. Williams & Norgate, London.

Tukey, J. W. (1962). The future of data analysis. *The Annals of Mathematical Statistics*, **33**, 1–67.

van Stein, B., van Leeuwen, M. and Bäck, B. (2016). Local subspace-based outlier detection using global neighbourhoods. *IEEE International Conference on Big Data*, 1136–1142.

Wang, X., Guo, F., Heller, K. A. and Bunson, D. B. (2015). Parallelizing MCMC with random partition trees. *Advances in Neural Information Processing Systems*, 451–459.

Zhao, T., Cheng, G. and Liu, H. (2016). A partially linear framework for massive heterogeneous data. *The Annals of Statistics*, **44(4)**, 1400–1437. doi:10.1214/15-AOS1410.