World Academy of Science, Engineering and Technology
International Journal of Computer and Systems Engineering
Vol:13, No:8, 2019

Predictive Analysis for Big Data: Extension of Classification and Regression Trees Algorithm

Ameur Abdelkader, Abed Bouarfa Hafida

***Abstract*—**Since its inception, predictive analysis has revolutionized the IT industry through its robustness and decision-making facilities. It involves the application of a set of data processing techniques and algorithms in order to create predictive models. Its principle is based on finding relationships between explanatory variables and the predicted variables. Past occurrences are exploited to predict and to derive the unknown outcome. With the advent of big data, many studies have suggested the use of predictive analytics in order to process and analyze big data. Nevertheless, they have been curbed by the limits of classical methods of predictive analysis in case of a large amount of data. In fact, because of their volumes, their nature (semi or unstructured) and their variety, it is impossible to analyze efficiently big data via classical methods of predictive analysis. The authors attribute this weakness to the fact that predictive analysis algorithms do not allow the parallelization and distribution of calculation. In this paper, we propose to extend the predictive analysis algorithm, Classification And Regression Trees (CART), in order to adapt it for big data analysis. The major changes of this algorithm are presented and then a version of the extended algorithm is defined in order to make it applicable for a huge quantity of data.

***Keywords*—**Predictive analysis, big data, predictive analysis algorithms. CART algorithm.

I. Introduction

IN recent years, the big data phenomenon has drained the emergency of a large number of analysis algorithms. The importance of business analytics for this process has been well established. Almost all domains such as social management (e.g. smart cities), business (e.g. manufacturing and marketing), education, health, security, environment (e.g. monitoring, protection, and industrial symbiosis, etc.), energy, and sciences (especially in physics, astronomy, biology, ecology, and earth science) have raised the challenge of accessing, managing and analyzing rapidly expanding big data.

The aim of predictive analytics is to identify future events. Their respective probabilities are also important outputs. It uses mathematical and historical data to generate new information, to predict new situations, and derive outcomes and their respective probabilities.

Predictive analytics can be applied to a variety of fields: business processes, finance, government, etc. Predictive analysis has shown its magnitude in the big data field: More data we have, better are the results. Nevertheless, despite the efficiency of existent predictive analysis algorithms, many of them are not suitable for huge data quantity. The studies applying these algorithms on big data are not numerous and it is still a topical research field. This can be attributed to the fact that behind the simplified notion of predictive analytics the choice of a relative algorithm is also a recurrent question in the literature. The criteria allowing the choice of an algorithm depend on the size and the nature of the data. They depend also on what to do with the results, and on the type of platform deployment used, etc.

The aim of this work is to study the concept of predictive analysis in the context of big data. Since deployment platforms of data storage systems are numerous: cloud, Map-Reduce, parallel machines, etc., we have chosen to focus on the Map-Reduce platform. The remainder of this paper is organized as follows. Section II describes literature review. Section III presents the application of predictive analysis for big data. Section IV illustrates tests realized on real big data. Finally, concluding remarks are made in Section V.

II. Literature Review

Predictive analytics has its origin in the 1940s, when governments started using the first computational models. With non-linear programming and real-time analytics, data analytics and prescriptive analytics go mainstream and become available to all organizations [3].

Siegel [1] defines predictive analysis as the technology which learns from experiences (data) to predict the future individuals' behaviors in order to take the best decisions [4].

For [5], predictive analysis, often called advanced analysis is a series of analytical techniques and statistics allowing the prediction of actions or futures [3].

Gartner [7] distinguished between analytical analysis and other analysis. According to him, there are four kinds of analysis: Descriptive analysis, diagnostic analysis, predictive analysis and prescriptive analysis. Descriptive Analysis deals with the past and tries to understand the impact of the present (business analysis). Diagnostic analysis provides answers to the questions about the reasons, the effects, and the interactions or events sequences. Predictive analysis looks towards the future and gives, on the basis of data exploration, machine learning and other statistical methods the probability of future events. Prescriptive analysis goes a step further than predictive analytics. It provides recommendations on the ways to affect a particular trend.

In order to summarize: the Predictive Analysis is an exploration technique allowing extraction and analysis of data in order to predict future trends, events and reasons of

Abdelkader Ameur is with the Data Processing School, Algeria (e-mail: ameur6297@gmail.com).
Hafida Bouarfa is with the University of Blida, Algeria (e-mail: hh.bouarfa@gmail.com).

behaviors. The accuracy of the results depends enormously on the data analysis and the hypothesis quality. The heart of the predictive analysis is based on capturing relationships between data from the past and the utilization of these relations to predict futures results and to take decisions.

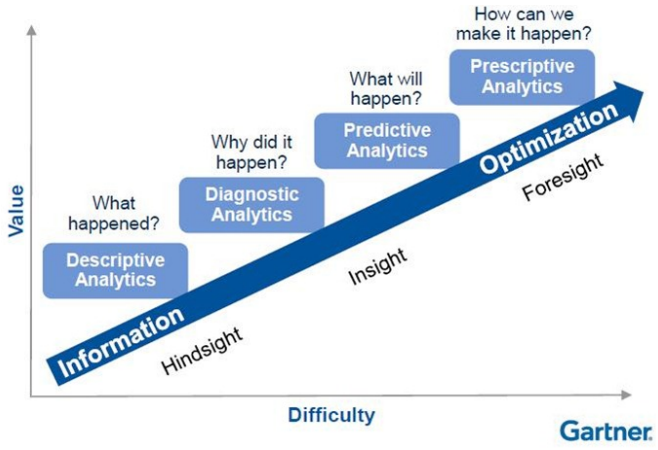

Fig. 1 Kinds of analysis [7]

A. Predictive Analysis Process

The predictive Analysis combines statistics techniques, data mining and machine learning to find a meaning to data. To design a predictive model, it is necessary to follow a cycle: Problem definition, data collection, data analysis, predictive model elaboration, validation of the model and then deployment.

In the literature, there is some interesting methods allowing the elaboration of the predictive model: Decision Trees [6], Neuronal networks [7], Support Vector Machines [8], the boosting [9]-[11], *k-nearest neighbor* [12], etc.

In this work, we have chosen to use decision trees, this choice is motivated by the fact that decision trees are easy to understand owing to their conditional structures. So, even a non-computer scientist will understand and use the model contrary to other structures which are more complex. Furthermore, we have found in them, the efficiency and the capacity to adopt the model in different situations because it can manage different type of variables.

There are many algorithms allowing building decisions trees (C4.5, CART, CHAID, etc.). In our study, CART Algorithm has been chosen: 1- It is used in the case of classification so it is in wide use. 2- It allows the utilization of variables of all types (continuous, discreet and categorical). 3- The parallel version of most of the others algorithms has been implemented, this is not the case for CART algorithm.

The aim of this work is to implement a parallel version Map-Reduce adapted to the application of the algorithm CART on the big data. In the rest of this work, we will explain the concepts Big Data and Map-Reduce. We will show the impact of these notions on the predictive analysis world. In the literature review, we can find several data structures allowing making an effective predictive model. As already mentioned, predictive analysis algorithms are relevant for the case of big data. This is done by parallelizing some instructions of these algorithms, instructions which can be executed at the same time.

There are many techniques allowing parallelizing sequential processing of an algorithm, among we can found Map-Reduce. Before presenting the concept of Map-Reduce and big Data, we have to introduce the notion of parallel programming

B. Big Data

The term “Big Data” means mega data, large data or massive data. It designates impressive volumes of data which are created daily par humans, organizations, connected objects and machines. Nevertheless, there is no universal definition to the concept of big data: The users and the providers of services do not give the same definition. For many researchers, big data refers to “the 3Vs”, Volume for the huge amount of data, Variety for the speed of data creation, and Velocity for the growing unstructured data [2].

The processing of this amount of data cannot be done with classical existing techniques, because it will generate a very expensive cost in execution time. That is why parallelism is a practical way in order to pass to the scale. In this context, we can find Map-Reduce which is one of the most used patterns allowing tasks repartition for data utilization and processing.

C. Parallel Programming

It can be defined as the simultaneous execution of a set of algorithm instructions [13]. These instructions must be executed in parallel without causing errors or deadlocks in the final results. The parallelization can be also in the execution of the same program on several fragments of a file or a system and returning results to the father system. In this work, we focus on Map-Reduce for big data.

1. Map-Reduce

Map-Reduce is a programming model allowing the development and test of programs dedicated to the analysis of distributed big data [14]. Its aim is to automate the calculation parallelization and distribution. It requires neither supervision system nor user expertise in parallel calculation. The system manages the parallel processing, the coordination and the failure tasks execution. The role of the user is to define and to implement the too functions Map and Reduce. Fig. 2 summarizes the execution steps of a Map-Reduce program:

2. CART Algorithm

CART’s principle is to build a binary tree using Gini index as a segmentation criterion and a post-pruning operation. It is a very effective algorithm, used for classification and regression. Based on a set of records, it uses Gini index to build a maximal tree. Then, it simplifies this tree owing to pos-pruning operation. In the following, Gini criterion and the phases of the algorithm are explained.

a. Gini Criterion

Gini coefficient is a statistical measurement of the dispersion of a given population [15]. Its formula is:

$$\text{Gini(S)}=\sum_{i=1}^{m} fi(1-f\text{i})=1-\sum_{i=1}^{m} fi^2$$

$f_i = \text{card}(S_i)/\text{card}(S)$

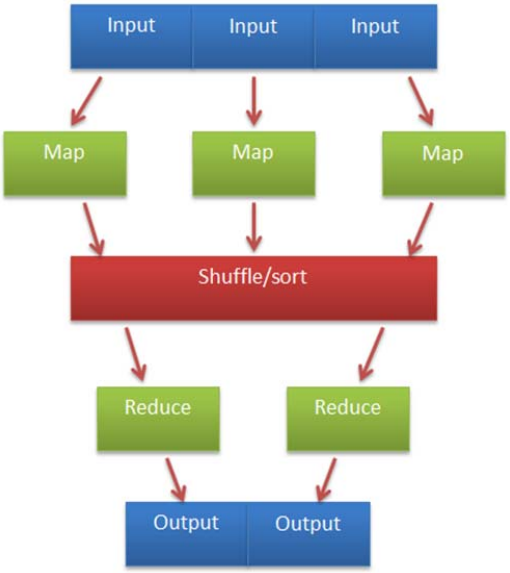

Fig. 2 The steps of a Map/Reduce *algorithm execution* [14]

Suppose, a set of data records of the shape (x_1, $x_{2...}x_n$, y), where, x_i are the prediction variables and y a variable to predict. Assuming that y takes m possible values (so we have m possible classes). S means the set of records and S_i is the group of records belonging to the class i.

b. CART Phases

The major phases of CART algorithm are: Construction of CART maximal tree and CART Post-pruning [16]. These steps are described in details in the next paragraphs.

Construction of CART Maximal Tree

CART starts with the construction of a non-simplified maximal tree which takes into consideration all the available data records, the process is as follows: The tree constructed by our algorithm is generated from a set of data records called learning data (x_1, x_2, …, x_n, y), where each xi is called an attribute. Each x_i will be an internal node or root in the maximal tree $T_{max.}$ The values taken by "y" are membership's classes and they are represented in the leaves tree.

CART starts by asking all possible binary questions for each attribute in order to determine the root of the tree. The chosen question is that which will maximize the information gain expressed as follows:

$$\text{Gain}(x_j) = \text{Gini}(S) - f_g.\text{Gini}(S_g) - f_d.\text{Gini}(S_d)$$

S: the set of records; x_j: The attribute x_j, it takes two values: x_j=a orx_j=non a. S_g: the records for which x_j=a. S_d: the records for which x_j=non a. $f_d = \text{card}(S_d)/\text{card}(S)$.

We do this process for each attribute in order to determine the question to ask for each of them. The question asked at the tree root is that, giving the maximum gain.

We do recursively the same process for the two branches of the tree. We stop when one of the three cases occurs:

1. When a node where all the examples have the same class is reached. This node becomes terminal (a leave) and it is attributed the corresponding class.
2. When a node with few examples (10% or less of the total records [15]) is reached. This node becomes terminal and it is attributed the majority class. If there is not a majority class in the considered set, all the set is taken otherwise, a random class is attributed.
3. In the case where there is no attribute for dividing, the node becomes terminal and it is attributed a class, as in 2.

CART Post-Pruning

Once the tree T_{max} is built, CART calculates for each node, a complexity criterion on the basis of the fault produced by the learning game. A leaf will replace the one having the less value. The same work is done on the resulting tree until to reach the root. At the end of this process, a succession of nested trees T_1, T_2,…,T_n are obtained, where T_n is the root of $T_{max.}$ Then, a validation set, different from the learning set is used. The tree that makes the least mistakes is returned par the algorithm.

In the next paragraph, we will explain the utilization of the algorithm CART extended for prediction in big data.

III. Application of Predictive Analysis for Big Data

Our idea was to define models of parallel programming for predictive algorithms, for using them on big data. Our proposition is then defining a Map-Reduce version of CART and applying it on big data. CART's principle is to build a binary tree using Gini index as a segmentation criterion and a post-pruning operation. It is a very effective algorithm, used for classification and regression.

The proposed parallel version is based on the parallelization of the algorithm calculation with Map Reduce. We distinguish: The calculation of the majority class at stop tests, the calculation of Gini criterion, the calculation of the criterion error of each tree node in the pruning phase, and the calculation of the number of errors made by each tree in the pruning phase. Furthermore, it is necessary to parallelize the process of each attribute (Fig. 3).

Considering that the attributes of a data set are independents from each other, different calculations can be done at the same time on these attributes: In the construction phase of the Max-Tree, binary questions on different attributes can be generated simultaneously. In the pruning phase, errors gains of the attributes can be calculated at the same time. And, also at the pruning step, the validation of the nested trees validation can be made using the validation data set. In the same way, trees tests are done in parallel. On this basis, our extended algorithm can be described as follow:

A. Max-Tree Computing

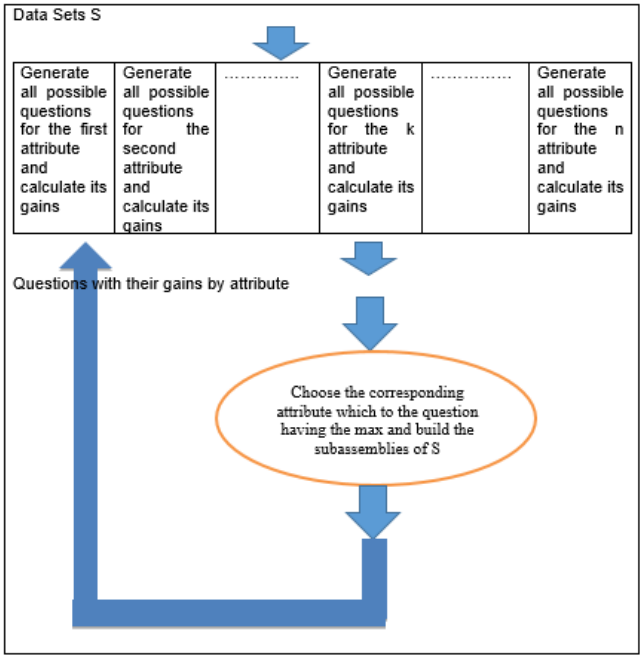

Fig. 3 Construction of CART max tree

IV. Tests and Results

Calculations in this parallel version of CART are mainly based on the Map-Reduce programming model. The algorithm performs its different calculations at the same time on all fragments of the working file. This will ensure the processing speed. In order to validate our systems, we have performed three tests. The first data tests are the parameters of motor vehicle quality parameters. In this test, the quality of a vehicle depends on a set of attributes (number of places, security, maintenance, etc.). Extended CART has learnt the prediction of a vehicle quality from these attributes.

The second test is an anonymous game; the gamers who are the attributes perform actions which are attributes' values. Each series of actions results in their game success or failure. The third test is also a game; it contains all the possibilities of scattering pieces in a part of tic-tac-toe between two gamers. The program will predict if there will be a winner according to the distribution of the pieces in the game.

In these three tests, we have divided data sets in three parts: One part has been used for the learning phase. The second part used for the pruning step in order to choose the optimal tree. And a part dedicated to prediction. The tree predicts the results and then we compare it to input data. The results are presented in Table I.

Our program is performing because its predictions are correct in more 65% of cases. On the basis of 71 tested cases, only 10 cases do not match the reality. This means it has predicted correctly 61 outputs on 71. The maximum number of errors is in the 3rd example: 22 errors on 71 lines and even in this test, it has predicted correctly more than 65% of examples (49 lines on 71).

TABLE I
Tests and Results

	Number of lines dedicated to the learning step	Number of lines dedicated to the validation	Number of lines dedicated to the prediction	Number of attributes	Execution time in minutes	Number of errors made by the classifier	Error rate
Test 1	1609	79	71	6	10	10	14.08%
Test 2	2999	115	71	36	55	2	2.8%
Test 3	410	46	71	9	25	22	30.9%

V. Conclusion

In this paper, we have tried to marry two big concepts: Predictive Analysis and Big Data. Our proposition has been materialized by CART extension which has been tested for three data sets. Our conclusion is that this extended algorithm can be used as a decision support system in various fields. From the theoretical point of view, we can say that our goal is achieved. In order to increase the speed of processing, we can parallelize all the "parallelizable" processes (not only calculations). This can be done by defining Map-Reduce models for all the program parts. Nevertheless, the experimental side can be improved as the execution of the program in a multi node Hadoop environment. This will increase significantly the processes speed and can give interesting results.

References

[1] Siegel E. (2016). PredictiveAnalysis,Library of Congress Cataloging-in-Publication Data
[2] McAfee M., E Brynjolfsson, E. (2012). Big Data: The Management Revolution, *Harvard* Business Review.
[3] Miller Thomas, R. (2013). Modeling Techniques in Predictive Analytics Business Problems and Solutions, Pearson Education, Inc.
[4] Siegel, E. (2013). *Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die.* John Wiley & Sons, p107.
[5] Gutierrez, P., Yves Gerardy, Y. (2016) Causal Inference and Uplift Modeling A review of the literature, JMLR: Workshop and Conference Proceedings 67:1–13
[6] Mitchell, T. (1997). Decision Tree Learning. Dans M. Hill (Éd.), *Machine Learning* (pp. 52-80).
[7] Gerstner, A. A. (2004). *Cognitive Navigation Based on Nonuniform Gabor Space Sampling, Unsupervised Growing Networks, and Reinforcement Learning* (Vol. 15).
[8] Cristanini, N. Shawe-Taylor, J. (2000). *Cristanini, NAn Introduction to Support Vector Machines and other kernel-based learning methods.* Cambridge University Press.
[9] Grete Heinz, L. J. (2003). Exploring Relationships in Body Dimensions. *Journal of Statistics Education, 11*(2).
[10] Sanjeev Arora, E. H. (2012). *The Multiplicative Weights Update Method: a Meta Algorithm and Applications.*
[11] Avi Levy, H. R. (2016). *Deterministic Discrepancy Minimization via the Multiplicative Weight Update Method.*
[12] Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician, 46*(3), 175-185.
[13] Culler, D. E. (1999). *Parallel Computer Architecture - A Hardware/Software Approach.* Morgan Kaufmann Publishers.
[14] Dean, J., & Ghemawat, S. (2008). Mapreduce: Simplified data

processing on large clusters. *51*(1), 107-113.

[15] *Définiitons, méthode et qualité * Indice de Gini*.(s.d.). Consulted mars 25th, 2017, on Insee: insee.fr

[16] Alain, G. (2007). *Exploration d'un algorithme génétique et d'un arbre de décision à des fins de catégorisation*. Québec: Université de Québec à trois rivières.