IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

IMPLEMENTATION OF DATA ANALYSIS ALGORITHM USING JULIA

[1]Mr. Hojiwala Robin, [2]Dr. Hitesh Joshi, [3]Mrs. Ridhdhi Naik

[1]Student, [2]Principal, [3]Assistant Professor
[1]M. Tech. In Computer Engineering,
[1]Bhagwan Mahavir College of Engineering & Technology, Surat, India

Abstract: Now a days cloud computing rapidly grow in computer and IT field. Every second so many data generated by multiple devices. Most of devices connect through internet. Now a days it's require the data store and analyze for the future decision. There are many algorithms are available for the data analysis. The generated data is in the huge amount. The Big data is available in various form like images, strings, numbers etc. We need algorithm that analyze data and give prediction about future decision. The gradient boosting algorithm is one of them that analyze different format data and give result. Gradient boosting algorithm analyze big data and select effective algorithm related to best outcome. Gradient Boosting Algorithm combine all the algorithm and give efficient and accurate data. Gradient Boosting Algorithm mainly associate with the server based analytics algorithm and we can use them for real time data. In our research Julia programming language is used because Julia is so fast as compared to Python language. Julia is reliable and accuracy is more in compare to Python. Julia is useful for server based real time data analysis and it is very fast as compare to python. Here we are using medical data of America in different location.

Index Terms **– Linear Regression, Logistic Regression, NaiveBayes Regression, Classification Regression Tree (CART), K – Mean Clustering, XGBoost**

I. INTRODUCTION

Big-data analytics in deal with unclassified huge amount data for recognized specific data pattern. The Big data need to identify specific responsible source of generating data. The data analyze for the medical science, pandemic situation, future prediction, costumer behavior, product demands, complains. Big data analysis group the specific data pattern and make effective analysis. Doctors, Medical Laboratory Technician need to analyze the data for medical science technology improvement purpose and future prediction related to disease spread and we can take effective decision on proper time. We need to combine all factors related human body disease, viruses, bacteria and involve for the analysis in various medical terms. Various datamining and analytics techniques have been used for the data analytics in medical science and e-commerce companies. The Gradient Boosting Algorithms is technique for the analysis huge amount of data. The main idea of Gradient Boosting Machine is given by Jerome Friedman. In his seminal paper from 1999 (updated in 2001) called Greedy Function Approximation: A Gradient Boosting Machine, introduced the gradient boosting machine, though the idea of boosting itself was not new [1]. Ensemble machine learning methods use multiple and combine several base models and learners in order to produce an optimal predictive model and obtain a better acceptable accuracy [2] and prediction task with a high confidence level. The gradient boosting algorithm is use for variety of base modal. This algorithm is use for improve prediction and the accuracy in different level. This approach also use the reduce variance and readability in various data. Gradient Boosting Algorithm predict different model for the same data and combine different models to improve reliability and accuracy.

II. TYPES OF DATA ANALYSIS ALGORITHMS

In the recent time digital era, data is the new gold. Every organization nowadays understands the importance of having a stockpile of data at its disposal. Various medical science laboratories, scientist, researchers, WHO etc. are dominating the modern era, and a big credit for that goes to the mammoth data stores they have at their disposal. Having such huge data stores at their disposal has enabled these companies to push the boundaries of medical technology advancement in a way that was never seen before. A burning example that exhibits the power of data and what can be achieved through its proper analytics is World Health Organization works during COVID – 19 pandemic work. Built on top of data pipelines containing a huge amount of dynamic and diverse data collected by WHO from multiple sources, it is a piece of technology that seems like something straight from the future.

However, having data alone is not sufficient. Data on its own is useless and becomes meaningful only when proper analysis of that data is done. With an unprecedented increase in the amount of data generated in the last couple of years, it has become more necessary now than ever to have fast and efficient data-analytics algorithms at our disposal as the classical methods of data analysis using graphs or charts are simply not enough to keep up with this huge amount of data otherwise also known as Big Data. To solve this problem, data scientists all over the world have developed and are in the process of developing new advanced algorithms for

analyzing big data efficiently. We will keep our focus on the five most popular big-data analytics algorithms that usually form the basis of the majority of high-performance analytics models. There are five types algorithms for big data analysis:

1. Linear Regression
2. Logistic Regression
3. Naïve Bayes
4. Classification and Regression Tree
5. K –Mean Clustering

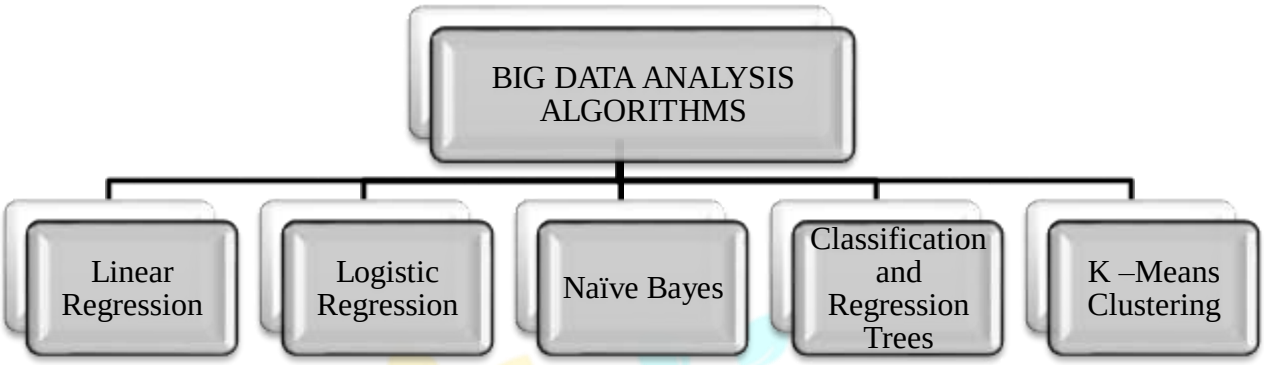

Figure 1: Types of Big data algorithm

2.1 Input Data

We are taking medical data of the USA of the various state as input. We have CSV file for analysis. This CSV file convert into Data Frame. Data frame split into various column for analysis. Input data is shown as below.

```
julia> readlines("Disease.csv")
1185677-element Vector{String}:
 "YearStart,YearEnd,LocationAbbr," … 470 bytes … "nCategoryID3,StratificationID3"
 "2014,2014,AR,Arkansas,SEDD; SID" … 123 bytes … "ST,AST3_1,NMBR,GENDER,GENM,,,,"
 "2018,2018,CO,Colorado,SEDD; SID" … 131 bytes … "ST,AST3_1,NMBR,OVERALL,OVR,,,,"
 "2018,2018,DC,District of Columb" … 123 bytes … "ST,AST3_1,NMBR,OVERALL,OVR,,,,"
 "2017,2017,GA,Georgia,SEDD; SID," … 126 bytes … "ST,AST3_1,NMBR,GENDER,GENF,,,,"
 "2010,2010,MI,Michigan,SEDD; SID" … 131 bytes … "6,AST,AST3_1,NMBR,RACE,HIS,,,,"
 "2015,2015,MT,Montana,SEDD; SID," … 144 bytes … "0,AST,AST3_1,NMBR,RACE,HIS,,,,"
 "2013,2013,OR,Oregon,SEDD; SID,A" … 122 bytes … "ST,AST3_1,NMBR,GENDER,GENM,,,,"
 "2013,2013,PR,Puerto Rico,SEDD; " … 126 bytes … "ST,AST3_1,NMBR,OVERALL,OVR,,,,"
 "2017,2017,PR,Puerto Rico,SEDD; " … 126 bytes … "ST,AST3_1,NMBR,OVERALL,OVR,,,,"
 "2010,2010,WI,Wisconsin,SEDD; SI" … 126 bytes … "ST,AST3_1,NMBR,GENDER,GENM,,,,"
 "2016,2016,WI,Wisconsin,SEDD; SI" … 133 bytes … "5,AST,AST3_1,NMBR,RACE,HIS,,,,"
 "2014,2014,AL,Alabama,NVSS,Asthm" … 109 bytes … "ST,AST4_1,NMBR,GENDER,GENM,,,,"
 "2015,2015,ID,Idaho,NVSS,Asthma," … 112 bytes … "ST,AST4_1,NMBR,OVERALL,OVR,,,,"
 "2016,2016,ID,Idaho,NVSS,Asthma," … 112 bytes … "ST,AST4_1,NMBR,OVERALL,OVR,,,,"
 "2020,2020,IL,Illinois,NVSS,Asth" … 116 bytes … "ST,AST4_1,NMBR,GENDER,GENM,,,,"
```

Figure 2: Data in CSV file

Julia 1.8.5

```
julia> Disease=CSV.read("Disease.csv",DataFrame)
1185676×34 DataFrame. Omitted printing of 28 columns
```

Row	YearStart	YearEnd	LocationAbbr	LocationDesc	DataSource	Topic
	Int64	Int64	String3	String31	String	String
1	2014	2014	AR	Arkansas	SEDD; SID	Asthma
2	2018	2018	CO	Colorado	SEDD; SID	Asthma
3	2018	2018	DC	District of Columbia	SEDD; SID	Asthma
4	2017	2017	GA	Georgia	SEDD; SID	Asthma
5	2010	2010	MI	Michigan	SEDD; SID	Asthma
6	2015	2015	MT	Montana	SEDD; SID	Asthma
7	2013	2013	OR	Oregon	SEDD; SID	Asthma
8	2013	2013	PR	Puerto Rico	SEDD; SID	Asthma
9	2017	2017	PR	Puerto Rico	SEDD; SID	Asthma
10	2010	2010	WI	Wisconsin	SEDD; SID	Asthma
11	2016	2016	WI	Wisconsin	SEDD; SID	Asthma
12	2014	2014	AL	Alabama	NVSS	Asthma
13	2015	2015	ID	Idaho	NVSS	Asthma
14	2016	2016	ID	Idaho	NVSS	Asthma

Figure 3: Data converted into Data Frame

III. LINEAR REGRESSION

Linear regression is a kind of statistical test performed on a dataset to define and find the relation between considered variables [3]. Linear regression is mostly used and popular for statistical analysis algorithms. Being a very simple yet extremely powerful algorithm for data analysis, it is used by data scientists extensively for designing simple but difficult analytical models. If the linear-regression equation contains a single dependent variable (y) and a single independent variable (x), it is known as univariate regression and is represented by equation:

$$y = \beta_1 * x + \beta_0$$

y = dependent variable: x = independent variable; β_1 = Scale factor; β_0 = Bias Coefficient

The regression model with more than one independent variable is known as multivariate regression. In a multivariate-regression model, an attempt is made to account for the variation of independent variables in the dependent variable synchronically [4]. The equation of multivariate regression is an extension of univariate regression and is represented in equation

$$y = \beta_0 + \beta_1 * x + \ldots + \beta_n * x_n + \varepsilon$$

y = dependent variable
x = independent variable
(β_1 - β_n) = scale factor
β_0 = bias coefficient
ε = error

3.1 Results:

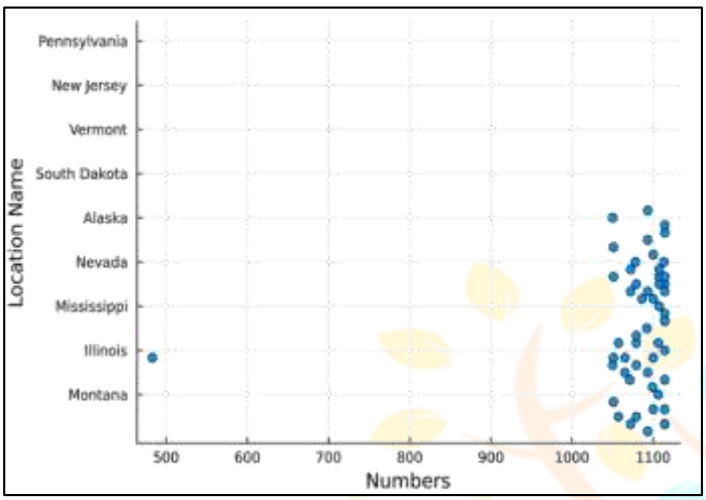

Figure 4: Scatter data "Tobacco" Group for Linear Regression

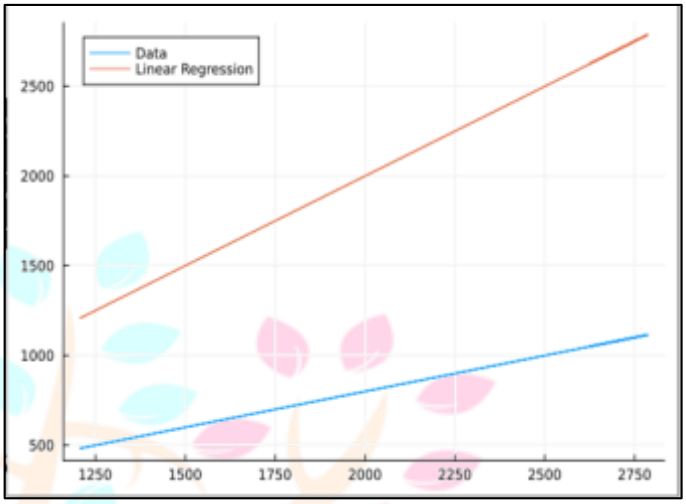

Figure 5: Linear Regression Plot

Coefficients:

	Coef.	Std. Error	t	Pr(>|t|)	Lower 95%	Upper 95%
(Intercept)	−0.66298	1.4505	−0.46	0.6495	−3.57232	2.24636
b	2.50059	0.00133909	1867.38	<1e-99	2.49791	2.50328

Figure 6: Linear Regression Data Analysis

IV. LOGISTIC REGRESSION

The technique of logistic regression in big data analytics is used when the variable to be considered is dichotomous (binary). Logistic regression works on the concept of logit—the natural logarithms of an odds ratio [5]. They proposed a category of various models (linear regression, ANOVA, Poisson regression, etc.), including logistic regression as a special case. Equation represents a general equation of logistic regression. The positive log of an odds ratio usually translates into a probability of success greater than 50%.

$$\log\log\{1 - p\} = + \beta_2 * x$$

p/(1-p) = Odd ratio
x = Independent variable
β_1 = Scale factor
β_1 = Bias coefficient

4.1 Results

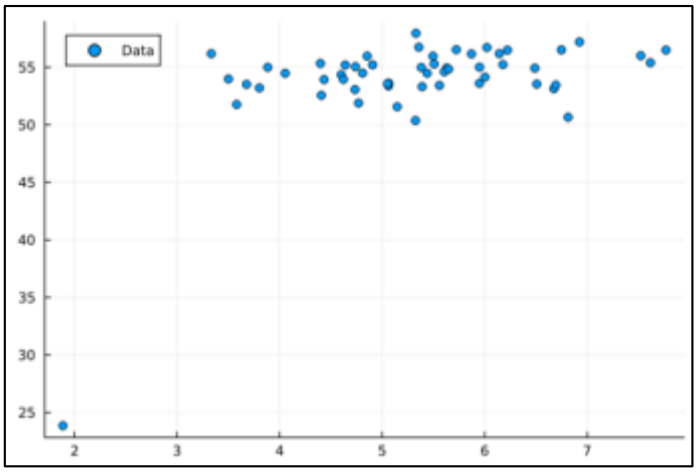

Figure 6: Scatter data "Tobacco" Group for Logistic Regression

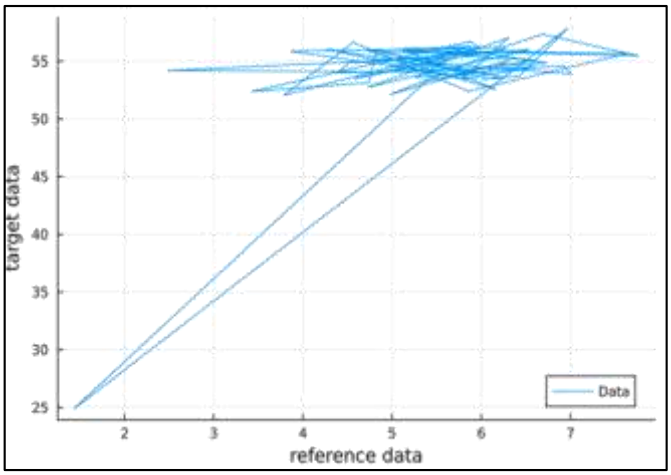

Figure 7: Logistic Regression Plot

Coefficients:

	Coef.	Std. Error	z	Pr(>|z|)	Lower 95%	Upper 95%
(Intercept)	-2.38144	1.72825	-1.38	0.1682	-5.76875	1.00587
y	0.146327	0.0317697	4.61	<1e-05	0.0840597	0.208595

Figure 8: Logistic Regression Data Analysis

V. NAIVEBAYES CLASSIFICATION

Naive Bayes uses the probabilistic approach for constructing classifiers. These classifiers can simplify learning by assuming that features are independent of given class [6]. Naive Bayes classification is a subset of Bayesian decision theory. It's called naive because the formulation makes some naive assumptions [7].

To understand the equation of Naive Bayes classifiers we need to understand Bayes' theorem, which is the fundamental theorem on which Naive Bayes classifiers work. Bayes' theorem Bayes' theorem finds the probability of the occurrence of an event, given the probability of another event that has already occurred. Bayes theorem is stated mathematically as shown in equation:

$$P\left(\frac{A}{B}\right) = \frac{P\left(\frac{B}{A}\right).P(A)}{P(B)}$$

P(A) = Probability of occurrence of event A
P(B) = Probability of occurrence of event B
P(A/B) = Probability of A Given B
P(B/A) = Probability of B given A

5.1 Result

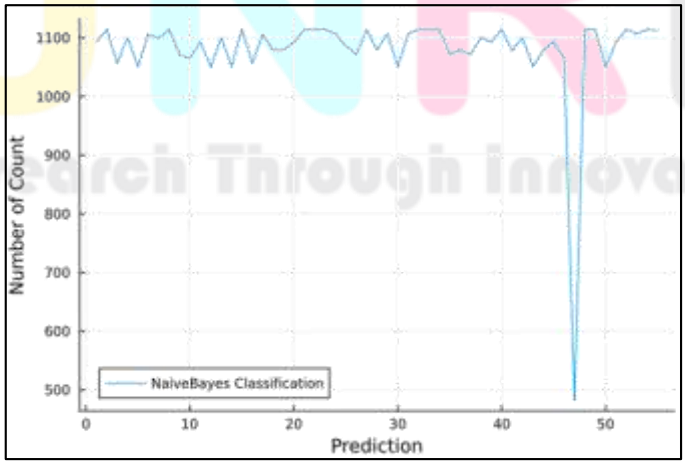

Figure 9: NaiveBayes Classification Plot

VI. CLASSIFICATION REGRESSION TREES (CART)

The CART model can be represented as a binary tree. Each node in the tree represents a single input variable (x) and a split point theorem variable, and the leaf node is represented using an output variable (y), which is utilized for forecasting. For example, suppose a dataset having two input variables (x) of disease infection and name of area with maximum infection the output variable (y) will tell whether the other infection of the person is which are. Figure 10 represents a very simple binary decision tree model.

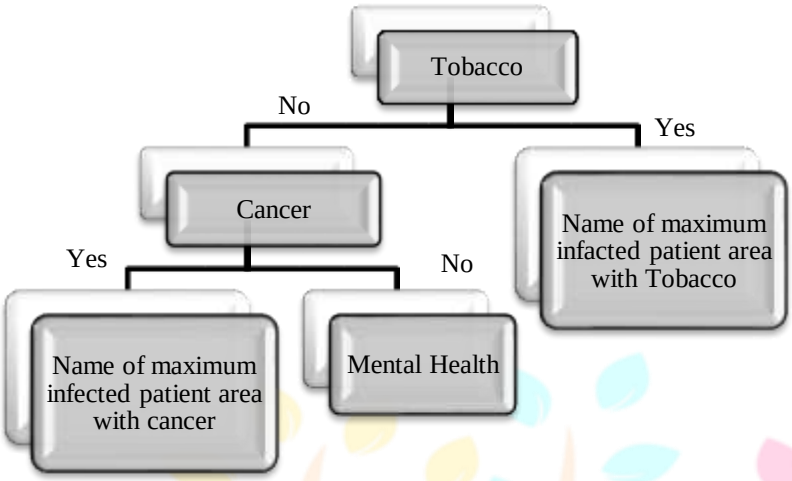

Figure 10: Representation of Binary Decision-Tree Model

6.1 Results

```
julia> model2 = build_tree(labs, feat)
Decision Tree
Leaves: 17
Depth:  7

julia> print_tree(model2, 10)
Feature 1 < 7.5 ?
├─Feature 1 < 3.5 ?
│    ├─Feature 1 < 1.5 ?
│    │    ├─Vermont : 1253/66091
│    │    └─Feature 1 < 2.5 ?
│    │         ├─Colorado : 1440/78300
│    │         └─Vermont : 1531/80342
│    └─Feature 1 < 4.5 ?
│         ├─Colorado : 3378/176339
│         └─Feature 1 < 6.5 ?
│              ├─Feature 1 < 5.5 ?
│              │    ├─Colorado : 3045/157750
│              │    └─Colorado : 472/24731
│              └─Nevada : 2976/152874
└─Feature 1 < 8.5 ?
     ├─Colorado : 2964/156808
     └─Feature 1 < 9.5 ?
          ├─Colorado : 74/3922
          └─Feature 1 < 12.5 ?
               ├─Feature 1 < 11.5 ?
               │    ├─Feature 1 < 10.5 ?
               │    │    ├─Colorado : 176/9570
               │    │    └─Colorado : 243/13200
               │    └─Texas : 1385/75418
               └─Feature 1 < 13.5 ?
                    ├─Colorado : 496/26316
                    └─Feature 1 < 15.5 ?
                         ├─Feature 1 < 14.5 ?
                         │    ├─Vermont : 431/22273
                         │    └─Colorado : 1367/73260
                         └─Feature 1 < 16.5 ?
                              ├─Vermont : 211/9086
                              └─Vermont : 1114/59396
```

Figure 11: Classification Regression Tree (CART)

VII. K –MEAN CLUSTERING

The K-means clustering algorithm follows the approach of expectation-maximization. The expectation step is assigning the data point to the closet cluster. The maximization step is finding the centroid of each of these clusters. The final goal of the K-means algorithm is to minimize the value of squared error function given as:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} (|x_i - v_j|)^2$$

$|x_i - v_j|$ = Euclidean distance between x_i and v_j

Being a high performing, unsupervised learning algorithm, K-means finds application in a wide variety of fields. Due to its popularity, researchers have created different hybrid versions of this algorithm that are being used extensively in numerous fields. Youguo & Haiyan have developed a clustering algorithm on top of K-means clustering, which provides greater dependence to choose the initial focal point [8].

7.1 Results

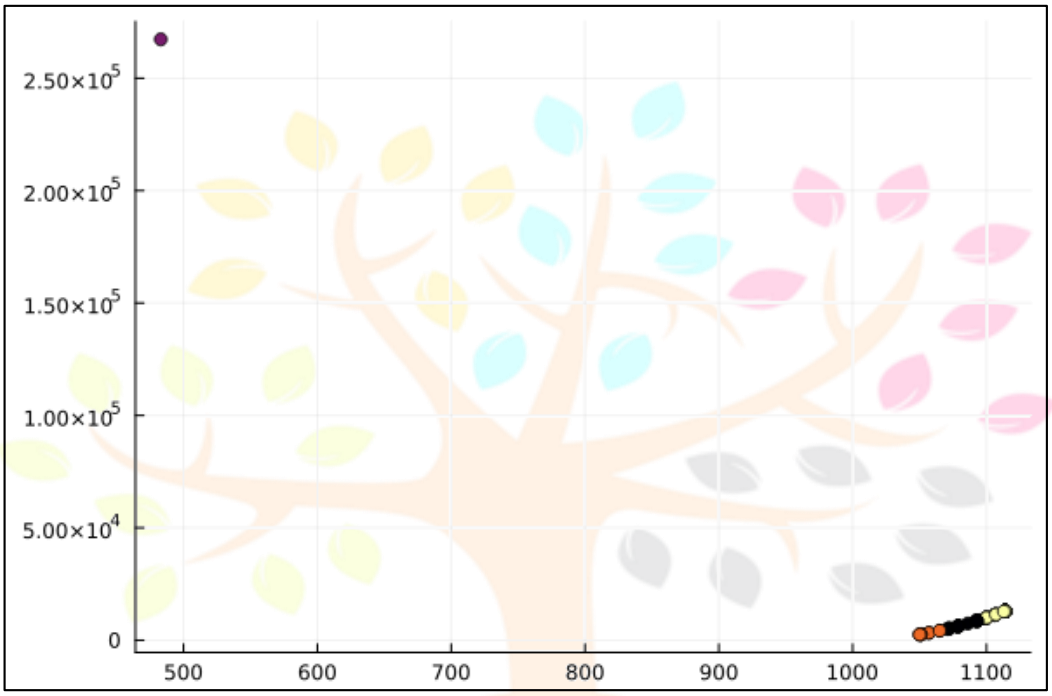

Figure 12: K – Mean Clustering

VIII. GRADIENT BOOSTING ALGORITHM

Ensemble machine learning methods use multiple and combine several base models and learners in order to produce an optimal predictive model and obtain a better acceptable accuracy [9] and prediction task with a high confidence level. By having different models and learners, this approach improves the generality of the individual classifiers. The combination of different base models and learners provides an improved prediction performance. To reduce and minimize the variance and bias from the actual and predicted values, the model is trained multiple times, and the predictions combine with several other models.

IX. MACHINE LEARNING USING H20 FOR GBM, LIGHT GBM, AND XGBOOST

Throughout this work, GBM, Light GBM, and XGBoost are used to boost the efficiency of the model by integrating a collection of weak classifiers to create such robust classifiers. Accuracy and performance at the acceptable level of training datasets and validation datasets depend on hyper parameter tuning and datasets, including the characteristics of datasets [10], [11], [12], [13], [14]. In order to leverage the machine learning for GBM families, hyper parameter tuning for 2 different models using Big Data Analytics using H20 Driverless AI is a critical aspect in building the final modelling pipeline. In this research, the number of model tuning combinations was evaluated to determine the optimal model settings for Light GBM and XGBoost.

9.1 METHOD

The ingest data resulted from the column types, and for the feature pre-processing, the output will be in the numeric form the raw features. This dissertation focuses on the model and feature tuning; specifically, on the hyper parameter tuning with feature selection and generation. Thus, the features in each iteration were updated using variable importance from the previous iteration as a probabilistic before deciding which new features to create. In order to determine the most efficient models and features, the procedures for the identification of optimal parameters for the different models have been completed through the training of models with different parameters. The best parameters are generated by the R2 on the internal validation data. For the feature evolution, the genetic algorithm was used to find the best set of model parameters and transformation features to be used in the final model.

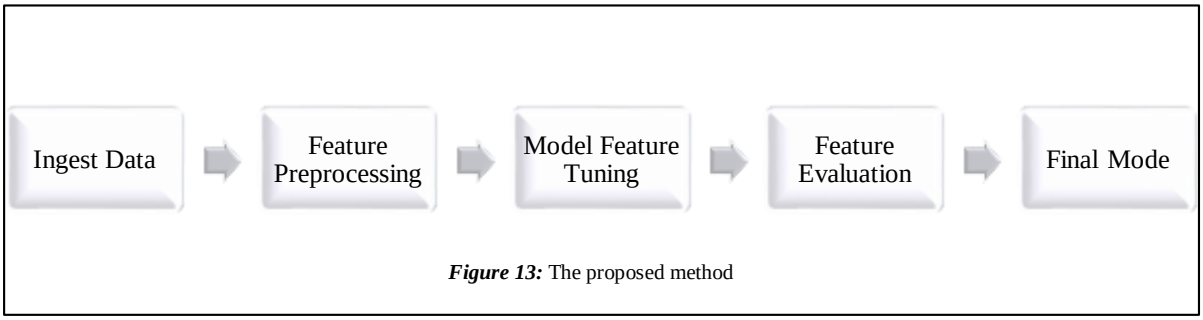

Figure 13: The proposed method

9.2 DATA MODELLING PROCESS

As highlighted in the contribution of this proposed research, predictive maintenance is performed as needed, especially before a failure occurs. The predictive maintenance problem is modelled as a supervised learning problem, where the target is the Time-To-Failure (TTF) in terms of hours. Figure 14 provides an overview of the modelling process. For every equipment, the historical dataset is split to obtain the latest three months' worth of data. From this dataset, the target is generated to create the input dataset, which will be used in training the model. For every row, the target column holds TTF value, which is the number of hours before the equipment is expected to fail. The input dataset is prepared by appending the target column to the three-month dataset.

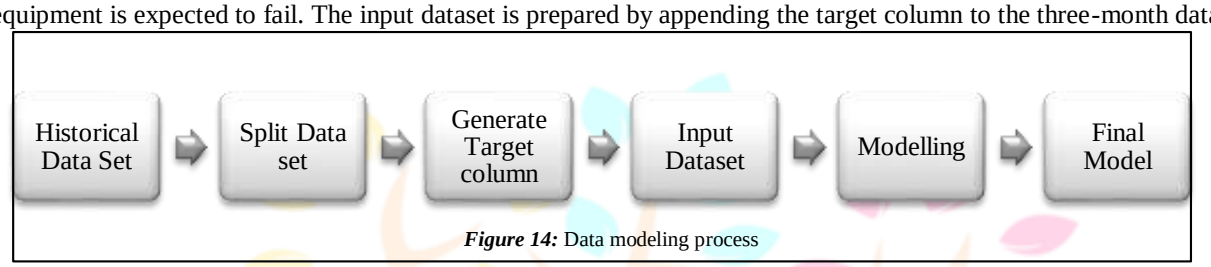

Figure 14: Data modeling process

9.3 MODEL ALGORITHMS

The final model is an ensemble of models of GBM family, which is based on decision trees. Some of the advantages of GBM include unbeatable predictive accuracy, as well as its robustness towards missing data. For each equipment, up to 50 models were trained and compared in performances based on R2 (as described in the previous section). GBM trains multiple models in an incremental, additive, and sequential manner by improving on weaker models. The variants that are used include Light GBM and XGBoost GBM, which differ in the methods used to calculate the best splitting decisions. There is not a single model that is best for all types of problems – the best model in most cases is specific to the provided training/validation dataset, hence, the variety in final models selected for each equipment. Causality is determined by running a regression analysis. The regression analysis offers a summary of the statistical relationship between one or more predictor variables and the response variable in statistical terms, a p-value tests whether a predictor variable has any effect on the response variable. The lower the p-value, the more likely it is for the predictor variable to have a significant effect on the response variable.

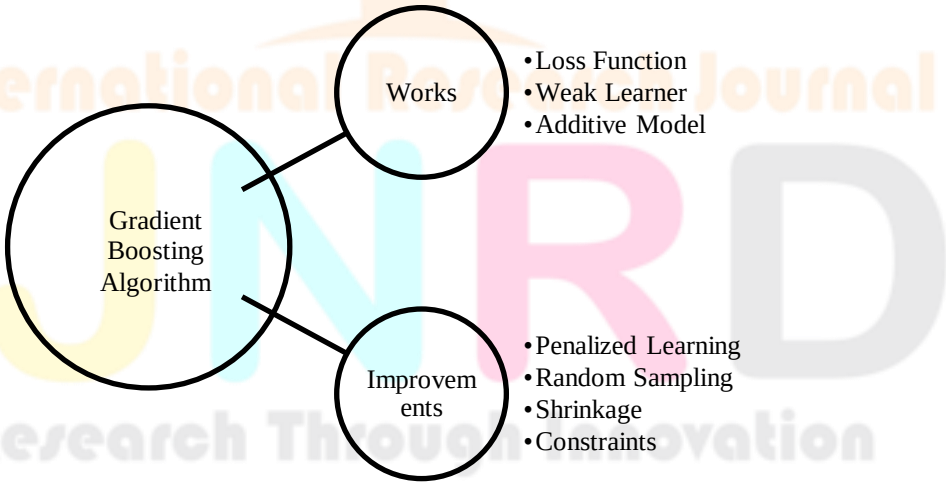

Figure 15: Gradient boosting Algorithm

Gradient boosting Algorithm involves three elements:

1. A loss function to be optimized.
2. Weak learner to make predictions.
3. An additive model to add weak learners to minimize the loss function.

9.4 MODEL TRAINING

The model training process is visualized in figure 16. The input dataset is further split into training, validation, and test datasets. The training algorithms are utilized on the training dataset, while the validation dataset is used to cross-check the robustness of the created model based on the accuracy of results. The model confidence is determined by scoring on the test dataset using the coefficient of determination method (R2). R2 indicates the correlation between the target values and values predicted by the model; the closer R2 is to 1.0, the closer the predictions are to the target values. New models are retrained until a satisfactory level of confidence is achieved with algorithm parameters, and split sizes are adjusted accordingly.

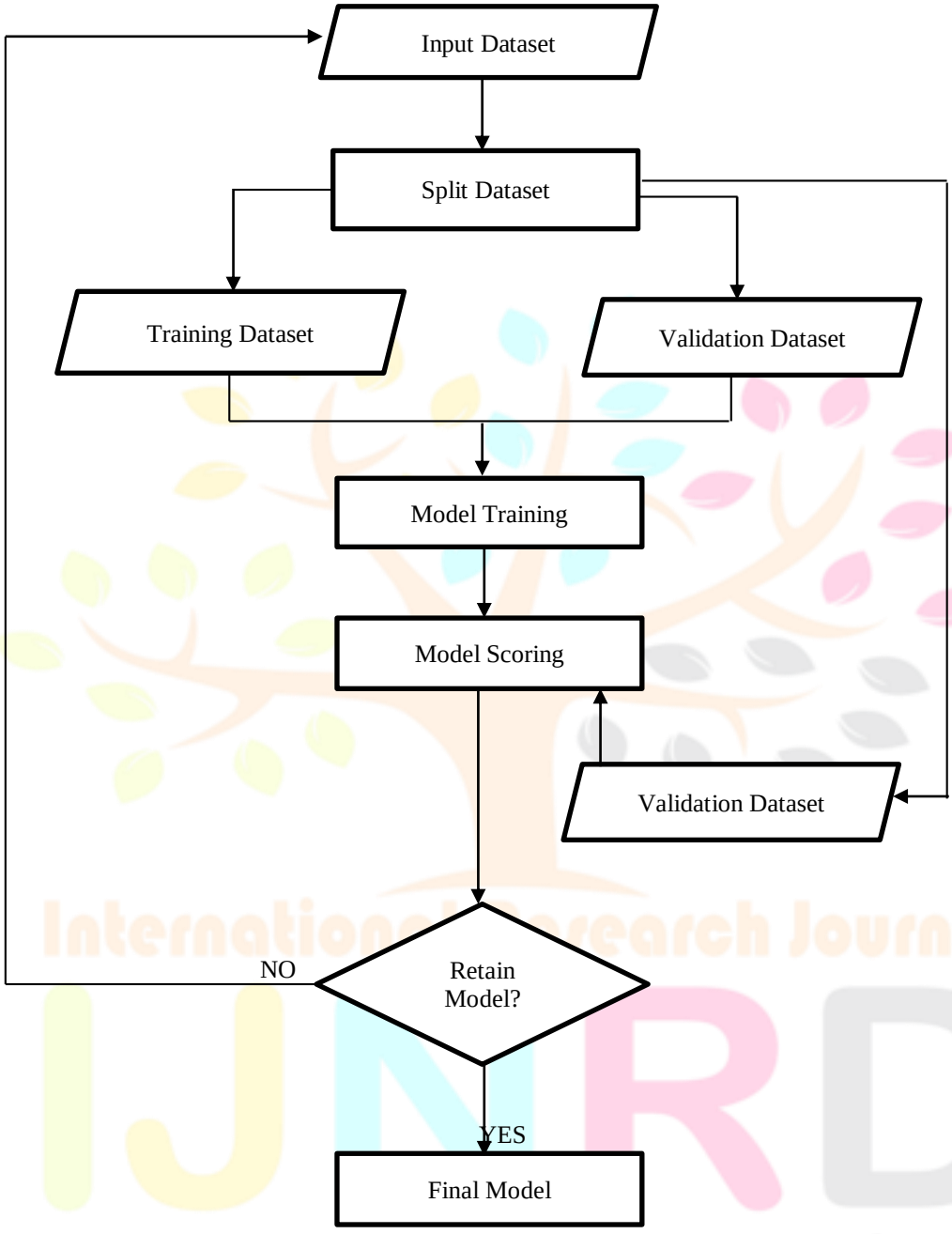

Figure 16: Model Training for the Research

X. CONCLUSION

Using gradient boosting algorithm family, we can create efficient and reliable modal for data analysis. Gradient Boosting algorithm boost the efficient algorithm for relevant data. Linear regression model is most popular and easy to use for considered variable. Linear regression is mostly suggested by scientist. Linear regression only efficient work on independent variable. If there are most dependent variable linear regression is not very efficient. Linear regression gives 70-75 % efficiency for only independent variable. Logistic regression model is efficiently used for dependent variable. In logistic regression dependent variable must be binary or data have absolute value. Logistic regression value mostly efficient for rating dataset. Logistic regression gives 50-55% efficiency. Naïve Bayes Classifier is based on dependent variable using probability. Naïve Bayes Classifier efficient on the repeated data. Naïve Bayes Classifier give efficiency 75-80% for the dependent variable. Classification and regression trees predictive model. CART model is binary tree representation. The CART starting from root nodes. The CART based on totally predictive based and binary trees. This CART model have 90-95% efficiency. K-Mean clustering is advance algorithm and very popular for data analytics. It is an unsupervised algorithm as it capable of drawing conclusions from datasets having only input variables without the requirement of having known or labeled outcomes. The centroids have stabilized using calculation. The centroid is predefined

and taking calculation around the centroid and predefined number of iterations has been reached. This K-mean clustering provides 80-85 % efficiency. This all model combined for their efficient data structure for efficient algorithm development. In gradient boosting algorithm this all model combines for efficient data analysis. The gradient boosting algorithm split all data into their respective efficient model. This technique is improving the data analysis and give efficient result.

XI. FUTURE WORK

The idea of using neural-network-based algorithms has been also proposed by data scientists. With the rise of quantum computing in the last couple of years, scientists are also looking forward to the possibility of leveraging the power of quantum computers in big-data analytics. Cloud based big-data analytics is also becoming quite popular as it can leverage the power of cloud computing for big-data analytics. With these new technological advancements on the horizon, it can be safely assumed that the future of big-data analytics is going to be bright and exciting.

REFERENCES

[1] https://explained.ai/gradientboosting/faq.html#:~:text=with%20this%20FAQ.,Who%20invented%20gradient%20boosting%20machines%3F,boosting%20itself%20was%20not%20new.

[2] E. G. Dada, J. S. Bassi, H. Chiroma, A. O. Adetunmbi, O. E. Ajibuwa and others, "Machine learning for email spam filtering: review, approaches and open research problems," Heliyon, vol. 5, no. 6, p. e01802, 2019

[3] X. Li, X. Yang, Y. Yang, I. Bennett, and D. Mba, "A novel diagnostic and prognostic framework for incipient fault detection and remaining service life prediction with application to industrial rotating machines," Appl. Soft Comput., vol. 82, p. 105564, 2019.

[4] A. Jimenez-Cortadi, I. Irigoien, F. Boto, B. Sierra, and G. Rodriguez, "Predictive Maintenance on the Machining Process and Machine Tool," Appl. Sci., vol. 10, no. 1, p. 224, 2020.

[5] A. Shirzad and M. J. S. Safari, "Pipe failure rate prediction in water distribution networks using multivariate adaptive regression splines and random forest techniques," Urban Water J., vol. 16, no. 9, pp. 653– 661, 2019.

[6] E. Yaman and A. Subasi, "Comparison of bagging and boosting ensemble machine learning methods for automated EMG signal classification," Biomed Res. Int., vol. 2019, 2019.

[7] E. G. Dada, J. S. Bassi, H. Chiroma, A. O. Adetunmbi, O. E. Ajibuwa and others, "Machine learning for email spam filtering: review, approaches and open research problems," Heliyon, vol. 5, no. 6, p. e01802, 2019.

[8] Y. Xia, C. Liu, Y. Li, and N. Liu, "A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring," Expert Syst. Appl., vol. 78, pp. 225–241, 2017.

[9] E. Al Daoud, "Comparison between XGBoost, LightGBM and CatBoost Using a Home Credit Dataset," Int. J. Comput. Inf. Eng., vol. 13, no. 1, pp. 6–10, 2019..

[10] B. Addepalli, H. Li, and D. Dueck, "Model Selection and Hyperparameter Tuning In Maps Query Auto-Completion Ranking."

[11] S. Mangalathu, H. Jang, S.-H. Hwang, and J.-S. Jeon, "Data-driven machine-learning-based seismic failure mode identification of reinforced concrete shear walls," Eng. Struct., vol. 208, p. 110331, 2020.

[12] V. A. Dev and M. R. Eden, "Formation lithology classification using scalable gradient boosted decision trees," Comput. Chem. Eng., vol. 128, pp. 392–404, 2019.

[13] J. G. de Oliveira, "A study on Gradient Boosting algorithms," 2019.