Article

Performance of CatBoost classifier and other machine learning methods

Abdullahi Ibrahim[1,*], Muhammed M. Muhammed[1], Samuel O. Sowole[2], Ridwan Raheem[3], Rabiat O. Abdulaziz[4]

[1] Department of Mathematical science, Baze University, Abuja, Nigeria. abdullahi.ibrahim@bazeuniversity.edu.ng (A.I), muhammed.muhammed@bazeuniversity.edu.ng (M.M.M)
[2] African Institute for Mathematical Sciences, Mbour, Senegal. oladimeji.s.sowole@aims-senegal.org (S.O.S)
[3] African Institute for Mathematical Sciences, Ghana. ridwan@aims.edu.gh (R.R)
[4] Department of energy engineering, PAUWES, University of Tlemcen, Algeria. rabiat.abdulaziz@aims-cameroon.org(R.O.A)
* Correspondence : abdullahi.ibrahim@bazeuniversity.edu.ng (A.I); Tel.: +2348067497949

2020

Abstract: Machine learning and data-driven techniques have become very famous and significant in several areas in recent times. In this paper, we discuss the performances of some machine learning methods with case of CatBoost classifier algorithm on both loan aproval and staff promotion. We compared the algorithm's performance with other classifiers. After some feature engineering on both data, CatBoost algorithm outperforms other classifiers implemented in this paper. In analysis one, features such as *loan amount, loan type, applicant income and loan purpose* are major factors to predict mortgage loan approvals. And in the second analysis, features such as *Division, Foreign schooled, geopolitical zones, Qualification and working years* had high impact towards staff promotion. Hence, based on the performance of CatBoost in both analysis, we recommend this algorithm for better prediction of loan approvals and staff promotion.

Keywords: Machine learning algorithms, data science, CatBoost, loan approvals, staff promotion, python .

1. Introduction

Machine learning and data-driven techniques have become very significant and famous in several areas. Some of the machine learning algorithms used in practice includes; support vector machine, logistic regression,catboost, random forest, decision tree, adaboost, extreme gradient boosting, gradient boosting, naive bayes, K-nearest neighbour and many more. In supervised machine learning, classifiers have been widely used in areas such as fraud detection, spam email, loan prediction and so on. In this work, we shall look into applications of some machine learning methods in areas of loan prediction and staff promotion.

The issuance of loan is one of the many profit sources of financial institutions. However, the problems of default by applicants have been of major concern to credit providing institutions [1]. Studies conducted in the past were mostly empirical and as such the problems of default have not been definitively dealt with. The furtherance of time to the 21*st* century was accompanied by bulks of archived data collected from years of loan applications. Statistical techniques have been developed to study past data in order to develop models that can predict the possibility of defaults by loan applicants; thus, providing a score of creditworthiness. The availability of voluminous data called Big data necessitated the introduction of machine learning tools which can be used to discriminate loan applicants based on creditworthiness. This study considered some of these machine learning techniques to classify loan applicants based on available data in order to assess the probability of default and also recommend the technique that yields the best performance.
Since the advent of machine learning, several researches have been conducted to discriminate loan applicants. In [2], the authors developed an ensemble model by aggregating together Support Vector Machine (SVM), Random Forest (RF) and Tree Model for Genetic Algorithm (TMGA). The ensembled model was compared with each of these models individually and eight other machine learning techniques namely Linear Model

(LM), Neural Network (NN), Decision Trees (DT), Bagged CART, Model Trees, Extreme Learning Machine (ELM), Multivariate Adaptive Regression Spline (MARS) and Bayesian Generalized Linear Model (BGLM) and was concluded from the analysis that the Ensembled algorithm provided optimum result. Another author [3] tried to discriminate loan applicants by comparing six machine learning techniques. The study compared DT, RF, K-Nearest Neighbour (KNN), OneR (1R), Naïve Bayes (NB) and Artificial Neural Networks (ANN) in which Random Forest gave the best performance with an accuracy of 71.75%. In [1], a classifier, Binary Logistic Regression (BLR), was used to classify loan applicants after comparing Principal Components Analysis (PCA) and Factor Analysis (FA) to reduce the dimension of the data in which PCA outperformed FA. Also [4] conducted an exploratory research where the suitability of RF was tested in classifying loan applicants and an accuracy of 81.1% was achieved. In a related research by [5], RF, BLR and SVM were used to predict loan approvals and RF outperformed the other techniques with an accuracy of 88.63%. [6] predicted approvals for a peer-to-peer lending system by comparing Logistic Regression (LR), Random Tree (RT), Bayesian Neural Network (BNN), RF, Gradient Boosted Decision Trees (GBDT), XGBoost and CatBoost and the results indicated that CatBoost gave the best performance over the other classifiers. The review of past literature showed tremendous developments in the applications of machine learning classifiers and how ensembled classifiers outperform single classifiers. However, only a few researches considered Catboost classifier in loan prediction approvals; hence, this research seeks to compare eight machine learning methods namely Binary Logistic Regression, Random Forest, Ada Boost, Decision Trees, Neural Network, Gradient Boost, Extreme Gradient Boosting and Catboost algorithms in the prediction of loan approvals.

Application of machine learning in employee promotion is another area we shall look into. Employees/staff play a significant role towards the development of an enterprise. Employee promotion in an enterprise is a major concern to both the employer and employee. In the human resource management, staff promotion is very vital in order for organizations to attract, employ, retain and effectively utilize their employee's talents [7]. Promotion of staff in an organization is based on some factors among which are age [8], gender [9], education [10], previous experience [11] and communication strategy or pattern [12]. In [7], the authors applied some machine learning algorithms on chinese data to predict employee promotion. It was discovered that, among all the available features in their dataset, number of different position occupied, the highest departmental level attained and number of working years affect staff promotion. In [13], joint data clustering and decision trees was used to evaluate staff promotion. [14] carried out research on why the best and performing employees quit prematurely, and predicted performing and valuable employee likely to quit prematurely. The proposed algorithm was recommended to the human resource department in order to determine valuable employee likely to quit prematurely. Previous works showed tremendous developments in the applications of machine learning but little researches considered Catboost classifier in staff promotion.This research seeks to compare four machine learning methods namely Random Forest, Gradient Boost, Extreme Gradient Boosting and Catboost algorithms in the prediction of staff promotion.

The aim of this paper is to develop predictive machine learning model from supervised machine learning in areas of loan prediction and staff promotion. To achieve this aim, we shall set some objectives which will also be our contribution,

- Perform data science process such as exploratory analysis, perform data cleaning, balancing and transformation
- Develop predictive model from machine learning methods
- Apply some model evaluation metrics to determine the performance of the implemented models.

The rest of the paper is structured as follow: some machine learning algorithms are given in 2. Design and nomenclatures in 3. 4 present the analytical result and 5 concludes the paper.

2. Materials and Algorithms

2.1. Binary logistic regression

Consider a dataset with response variable (Y) classified into two categories, $Y =$ 'Loan approved','not approved' or $Y = \{'promoted',' notpromoted'\}$. Logistic regression models the probability of Y belongs to a specific category. With approach (1) below to predict this probability

$$p(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n \tag{1}$$

The conditions $p(X) < 0$ and $p(X) > 0$ can be predicted for values of X, except for range of X is limited. In order to keep away from this, $p(X)$ must be model with the help of a logistic function which generate between 0 and 1 values as output. The function is defined as in (2)

$$p(X) = \frac{e^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n}}{1 + e^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n}} \tag{2}$$

The *'Maximum likelihood'* method is used to fit (2). The unknown coefficients $\beta_0, \beta_1, \beta_2, \ldots, \beta_n$ in (2) should be approximated based on the data available for training the model. The intuition of likelihood function can be expressed mathematically as in (3)

$$\ell(\beta_0, \ldots, \beta_n) = \prod_{i:y_i=1} p(x_i) \prod_{i':y'_i=0} (1 - p(x'_i)) \tag{3}$$

The estimates $\beta_0, \ldots, \beta_n$ are selected to maximize this function. More explanation can be obtained in [15].

Basic Assumptions of Binary Logistic Regression

(i) The response variable must be binary.
(ii) The relationship between the response feature and the independent features does not assume a linear relationship.
(iii) Large sample size is usually required.
(iv) There must be little or no multicollinearity.
(v) The categories must be mutually exclusive and exhaustive.

2.2. Random Forest

Random forest (RF) algorithm is a well known tree based ensemble learning method and the bagging-type ensemble [16]. RF differs from other standard trees, each node is split using the best among a subset of predictors randomly chosen at that node [17]. This additional layer of randomness is what makes RF more robust against over-fitting [18]. To improve the bagged trees in RF, small tweak which de-correlate the trees are made. As in bagging, we build a number of decision trees on bootstrapped training sets. But when building these decision trees, each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p-predictors [19]. The RF approach for both classification and regression is presented in algorithm 1

Algorithm 1 Random Forests Algorithm

```
(i) Draw $m_{tree}$ boostrap samples from the initial data.
(ii) Initialize an unpruned tree, for every bootstrap samples, with the modification given as follow: instead of choosing the best-split among all predictors at each node, sample randomly $n_{try}$ of the predictors and select the best-split from among those features. Bagging can be seen as a special case of random forests which can be obtained when $n_{try} = k$, number of predictors.
(iii) new data is predicted by aggregating the predictions of the $m_{tree}$ trees.
```

2.3. Adaptive Boosting

Adaptive boosting (Adaboost) algorithm is another machine learning methods used to improve the accuracy of other algorithms. It is a boosted algorithm generated by training weaker rules to develop a boosted algorithm. In Adaboost, training sets $(x_1, y_1), \ldots, (x_m, y_m)$ is the input, where each x_i belongs to some *instance space* X, and each *feature* y_i is in some label set Y (in this case assuming that $Y = \{-1, +1\}$. This method calls repeatedly a given weak or base learning algorithm in a given series of rounds $t = 1, \ldots, T$. One of the significant and vital idea of the algorithm is to keep a distribution or set of weights over the training set. The weight of this distribution on training samples i on round t is represented by $D_t(i)$.

At initial, all weights are set equally, but on each round, the weights of mis-classified samples are increased so that the weak learner is forced to focus on the hard samples in the training set. The weak learner's job is to find a weak hypothesis $h_t : X \quad \{-1, +1\}$ appropriate for the distribution D_t [20]. A metric used to measure the goodness of a weak hypothesis is its error. The algorithm procedure is presented in algorithm 2

Algorithm 2 Adaboost Algorithm

```
Given: $(x_1, y_1), \ldots, (x_m, y_m)$ where $x_i \epsilon X, y_i \epsilon Y = \{-1, +1\}$
Initialize: $D_1(i) = \frac{1}{m}$ for $i = 1, \ldots, m$. For $t = 1, \ldots, T$ :
    • Train weak learner using distribution $D_t$.
    • Get weak hypothesis $h_t : X \quad \{-1, +1\}$ with error
    • $\epsilon_t = Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$
    • Choose $\alpha_t = \frac{1}{2} \ln(\frac{1 - \epsilon_t}{\epsilon_t})$
    • Update: $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} exp(-\alpha_t) & \text{if} \quad h_t(x_i) = y_i \\ exp(\alpha_t) & \text{if} \quad h_t(x_i) \neq y_i \end{cases} = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
      where $Z_t$ is a normalization factor (chosen so that $D_{t+1}$ will be a distribution).
      Output the final hypothesis:
          $$H(x) = sign(\sum_{t=1}^{T} \alpha_t h_t(x)) \qquad (4)$$
```

2.4. Decision Trees

Decision trees is one of the supervised learning algorithms that can be applied to both classification and regression problems [21]. We shall briefly consider regression and classification tree problems. There are two steps (as explained in [21]) for building a regression tree:

(i) Divide the set of feasible values $X_1, \ldots, X_n$ for into $I-$distinct and non-overlapping regions, $R_1, R_2, \ldots, R_i$.

(ii) For each sample that falls into R_i, same prediction is made, which is the average of the dependent feature for the training sets in R_i.

In order to construct regions $R_1, \ldots, R_i$, we elaborate on step (i) above. In theory, $R_1, R_2, \ldots, R_i$ could take any shape or dimension.

However, for simplicity, we may split the predictor space into high-dimensional boxes and for easy interpretation of predictive model. The aim is to obtain boxes $R_1, \ldots, R_i$ that minimizes the Residual Sum of Squares (RSS) as given in the mathematical expression in (5)

$$\sum_{i=1}^{I} \sum_{j \in R_i} (y_j - \hat{y}_{R_i})^2 \qquad (5)$$

Where $\hat{y}_{R_i})$ is the mean response of the training sets in the *ith* box.

The classification tree on the other hand predict a qualitative response variable. In a classification tree, we predict that every observations belongs to 'most frequently occurring' class of training sets in the region to which it belongs since we intend to allocate sample in a given region to the 'most frequently occurring' class of training sets in that region, the classification error rate is the part of the training sets in that region that do not belong to the most frequent class, as given in (6)

$$E = 1 - \max_{l}(\hat{p}_{ml}) \tag{6}$$

where $\hat{p}_{ml}$ denotes the ratio of training samples in the mth region from lth class. However, it turns out that classification error is not sensitive enough for tree-growing. The *Gini index* which is defined mathematically in (7)

$$G = \sum_{l=1}^{L} \hat{p}_{ml}(1 - \hat{p}_{ml}) \tag{7}$$

A measure of total variance over the L classes. Further details can be found in [21].

2.5. Neural Network

Artificial neural network (ANN) is an imitation of the interconnections made up in the human brain. The inputs in ANN represents the dendrites in the human brain which receives electrochemical signals from other neurons into the cell body. Every input carries a signal which is obtained by the product of its weight and the input to a hidden layer in the neuron powered by an activation function usually a sigmoid function, other activation functions like tangent hyperbolic function, linear function, step function, ramp function and gaussian function can also be used [22]. The last layer is the output layer which represents the axon extending to the synapse that connects two different neurons. A typical ANN architecture has inputs, output and a bias. The ANN architecture differs majorly by layers. The most common and simple architecture is a Perceptron which has two inputs, a hidden layer and a single output. The neural networks are mostly backpropagated to be used for classification and prediction. The back and forth movement in a neural network between the input and output layers is referred to as an epoch. A neural network undergoes several epochs until a tolerable error is achieved and thus the training of an artificial neural network is achieved. ANN architecture is shown in figure 1.

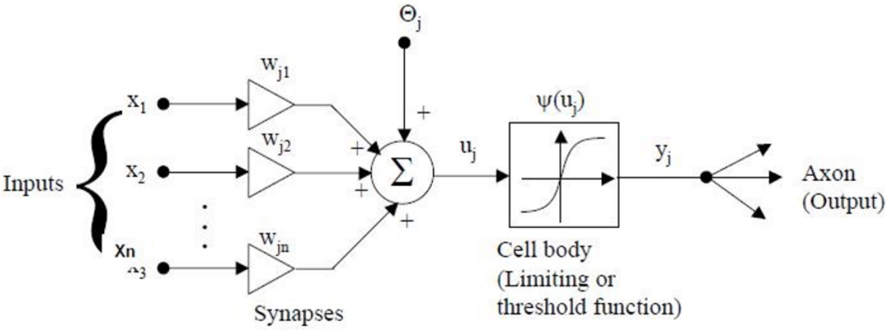


Figure 1. Architecture of Artificial Neural Network [23]

where Θ = external threshold, offset or bias w_{ji} = synaptic weights x_i = inputs y_i = output as in (8)

$$y_i = \psi\left(\sum_{i=1}^{n} w_{ji}x_i + \Theta_i\right) \tag{8}$$

2.6. Gradient boost

Gradient boost is a boosted algorithm used for regression and classification. It is derived from the combination of Gradient Descent and Boosting. It involves fitting an ensemble model in a forward stage-wise manner. The first attempt to generalise an adaptive boosting algorithm to gradient boosting that can handle a variety of loss functions was done by [24], [25]. The steps for gradient boosting algorithm is outlined in algorithm 3.

Algorithm 3 Gradient Boost Algorithm

```
Inputs:
    • Input data $(x,y)_{i=1}^{N}$
    • number of iterations M
    • choice of the loss-function $\psi(y,f)$
    • choice of the base-learner model $h(x,\theta)$
Algorithm:
    • intialize $\hat{f}_0$ with a constant
    • compute the negative gradient $g_t(x)$
    • fit a new base-learner function $h(x,\theta_t)$
    • find the best gradient descent step-size $\rho_t$ :
      $\rho_t = \arg\min_\rho \sum_{i=1}^{N} \psi[y_i, \hat{f}_{t-1}(x_i) + \rho h(x_i,\theta_t)]$
    • update the function estimate:
      $\hat{f} \leftarrow \hat{f}_{t-1} + \rho_t h(x,\theta_t)$
    • end for
```

2.7. Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is one of the boosted tree algorithm [16], which follows the principle of gradient boosting [24]. When compared with other gradient boosting algorithms, XGBoost makes use of a more regularized model formalization in other to control over-fitting of data, which gives it better performance [16]. In other to achieve this, we need to learn functions h_i, with each containing structure of tree and leaf scores [26]. As explained in [27], Given a data with m-samples and n-features, $\mathcal{D} = \{(X_j, y_j)\}(|D| = m, X_j \in \mathbb{R}^n, y_j \in \mathbb{R})$ a tree ensemble model makes use of L additive functions to predict the output as presented in (9)

$$\hat{y}_j = \phi(X_j) = \sum_{l=1}^{L} h_l(X_J), \qquad h_l \in \mathcal{H} \tag{9}$$

where $\mathcal{H} = \{h(X) = w_q(X)\}(q : \mathbb{R}^n \to U, w \in \mathbb{R}^U)$ is the space of regression trees. q denotes the structure of each tree that maps a sample to its corresponding leaf index. U denotes number of leaves in the tree. Each h_l corresponds to independent structure of tree q and leaf weights w.

To learn the set of functions used in the model, the regularized objective is minimized (10) as follows

$$\mathcal{L}(\phi) = \sum_j l(\hat{y}_j, y_j) + \sum_l \Omega(h_l), \qquad \Omega(h) = \gamma U + \frac{1}{2}\lambda||w||^2 \tag{10}$$

where l is differentiable convex loss function which measures difference between the target y_j and predicted $\hat{y}_j$. Ω penalizes the complexity of the model to avoid over-fitting. The model is trained in an additive way. A score to measure the quality of a given tree structure q is derived as given in (11)

$$\hat{\mathcal{L}}^{(u)}(q) = -\frac{1}{2}\sum_{j=1}^{U} \frac{(\sum_{i=I_j} f_i)^2}{\sum_{i=I_j} g_i + \lambda} + \gamma U \tag{11}$$

where $f_i = \partial_{\hat{y}^{(u-1)}} l(y_i, \hat{y}^{(u-1)})$ and $g_i = \partial^2_{\hat{y}^{(u-1)}} l(y_i, \hat{y}^{(u-1)})$ are the gradient and second order gradient statistics, respectively. Further explanation can be obtained in [27].

2.8. CatBoost

Another machine learning algorithm that is efficient in predicting categorical feature is the CatBoost classifier. Catboost is an implementation of gradient boosting, which makes use of binary decision trees as base predictors [28]. Suppose we observe a data with samples $D = \{(X_j, y_j)\}_{j=1,\ldots,m}$, where $X_j = (x_j^1, x_j^2, \ldots, x_j^n)$ is a vector of n features and response feature $y_j \in \mathbb{R}$, which can be binary (i.e yes or no) or encoded as numerical feature (0 or 1). Samples (X_j, y_j) are independently and identically distributed according to some unknown distribution $p(.,.)$. The goal of the learning task is to train a function $H : \mathbb{R}^n \to \mathbb{R}$ which minimizes the expected loss given in (12)

$$\mathcal{L}(H) := \mathbb{E}L(y, H(X)) \tag{12}$$

where $L(.,.)$ is a smooth loss function and (X, y) is a testing data sampled from the training data D.

The procedure for gradient boosting [24] constructs iteratively a sequence of approximations $H^t : \mathbb{R}^m \to \mathbb{R}, t = 0, 1, \ldots$ in a greedy fashion. From the previous approximation H^{t-1}, H^t is obtained in an additive process,such that $H^t = H^{t-1} + \alpha g^t$, with a step size α and function $g^t : \mathbb{R}^n \to \mathbb{R}$, which is a base predictor, is selected from a set of functions G in order to reduce or minimize the expexted loss defined in (13)

$$g^t = \text{arg min}_{g \in G} \mathcal{L}(H^{t-1} + g) = \text{arg min}_{g \in G} \mathbb{E}L(y, H^{t-1}(X) + g(X)) \tag{13}$$

Often times, the minimization problem is approached by the Newton method using a second-order approximation of $\mathcal{L}(H^{t-1} + g^t)$ at H^{t-1} or by taking a (negative) gradient step. Either of these functions are gradient descent [29,30]. Further expalanation on CatBoost algorithm can be obtained in [28].

3. Design and Nomenclatures

Some evaluation metrics such as confusion matrix, area under curve (AUC), accuracy, error rate, true positive rate, true negative rate, false positive rate, and false negative rate shall be discussed.

3.1. Confusion Matrix

A confusion matrix contains information about actual and predicted classifications from a classifier. Performance of such classifier is commonly evaluated using the data in the matrix. The table 1 shows the confusion matrix for classifier [31].

Table 1. Confusion matrix

	Predicted		
Actual		Negative	Positive
	Negative	True Negative (TN)	False Negative (FN)
	Positive	False Positive (FP)	True Positive (TP)

True Positive: The classifier predicted a true event and the event is actually true.

True Negative: The classifier predicted that an event is not true and the event is actually not true.

False Postiive: The classifier predicted that an event is true but the event is actually not true.

False Negative: The classifier predicted that an event is not true but the event is actually true.

The confusion matrix can be interpreted as: the TN and TP are the correctly classified classes while FN and FP are the mis-classified classes.

3.2. Model evaluation metrics

The Model training time, model accuracy and memory utilized are some good metrics for comparing the performance of the classifiers. In addition, the Area under the Receiver Operating Characteristics Curve (ROC-AUC) is a performance metric for classification accuracy. The AUC is another metric which checks the performance of multiple-class classification accuracy [26]. Model accuracy is the proportion of the correct predictions (True positive and True negative) from the total predictions defined in (14)

$$Accuracy = \frac{TN + TP}{TP + TN + FP + FN} \times 100\% \qquad \text{Error rate} = \frac{FP + FN}{FP + FN + TP + TN} \times 100 \tag{14}$$

Error rate is the proportion of of all incorrect predictions divided by total number of samples, given in (14)

The True Positive Rate (TPR), also called the sensitivity or recall, is the proportion of correct positive predicted class from total positive class. The best sensitivity is 1.0 and worst is 0.0. While the True Negative

Rate (TNR), also called the specificity, is the proportion of correct negative predictions from total number of negative classes. The best specificity is 1.0 and worst is 0.0. The TPR and TNR are given in (15).

$$\text{True postive rate } = \frac{TP}{FN+TP} \times 100 \qquad \text{True negative rate } = \frac{TN}{FP+TN} \times 100 \tag{15}$$

Precision is the number of correctly predicted positive value out of total number of positive class, as given in (16). False positive rate (FPR) is the number of incorrect positive prediction out of total number of negatives as in (16).

$$\text{False postive rate } = \frac{TP}{FNP+TP} \times 100 \qquad \text{False negative rate } = \frac{FP}{FP+TN} \times 100 \tag{16}$$

3.3. Calibration plots

Calibrated methods (classifers) are probabilistic classifiers for which the outcome of the predited probabilities of a pertificular classifier can be interpreted as a confidence interval. The metric is used to determine whether the predicted probablity can be interpreted as confidence interval.

3.4. System specification

All classifiers were run on jupyter notebook in python 3.7.4 on linux 19.10 version. The codes were run on 8GB HP elite book, core $i5$.

4. Results and discussion

In this section, we shall perform two analysis to in order to determine the performance of all the machine learning algorithms discussed previously. We begin by exploring the data in order to obtain the numerical statistics, identify missing values, outliers, and if the independent feature is balanced or not. After initial exploration we were able to identify missing values and outliers, the independent feature is balanced.

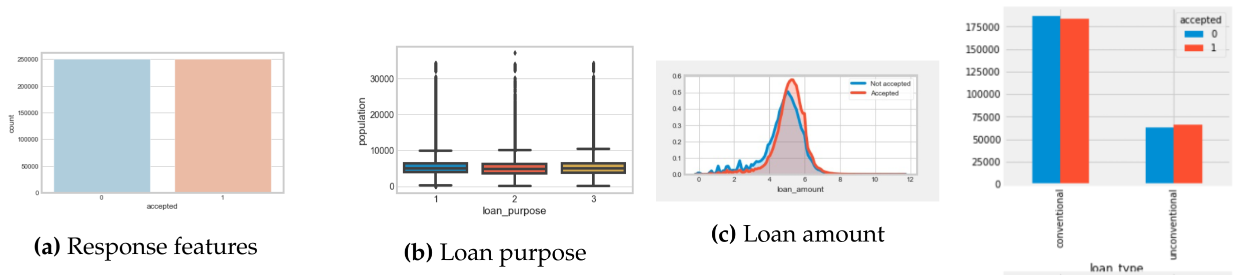
4.1. Analysis 1: Predicting Mortgage Approvals From Government Data

The analysis is based on US Government data concerning Predicting Mortgage Approvals [32]. This is a binary classification problem. Our analysis was based on 500,000 observations with 23 features from the training dataset of mortgage approvals government data, each containing specific characteristics for a mortgage application which will either get approval (”1”); or not (”0”). We tested our model on dataset with 150,000 samples.

4.1.1. Exploratory analysis

Before developing a predictive model, we need to understand the data points by exploratory analysis. In the exploratory analysis, we intend to find answers to some questions such as: (i) which features have missing values, (ii) features with outliers, (iii) is the response feature balanced? (iv) the distribution of the data points and so on. We presented some visualizations in figures 2 and 3 to answer these questions.

Figure 2(a) shows the both classes give almost the same frequency, with 250, 114 for the accepted data points and 249, 886 for not accepted data points. The data distribution is balanced and we can go on with other analysis. Figure 2(b) shows the distributions of the three classes of the loan purpose that we have. The Loan amount follows a normal distribution for both the accepted and not accepted in figure 2(c). Figure 2(d) shows the two classes of loan type. The accepted class of the conventional loan type has highest frequency with around 180, 000. In figure 3(a) state code 5 have the highest number ($\approx$ 890) of district lenders, with state code 50 having the least. Figure 3(b) shows the features “msa-md, applicant-income, number-of-owner-occupied-units, number-of-1-to-4-family-units, tract-to-msa-md-income-pct” having numbers of 76982, 39948, 22565, 22530, 22514 respectively. In summary, features such as loan amount, loan type, applicant income and loan purpose are major factors to predict mortgage loan approvals.



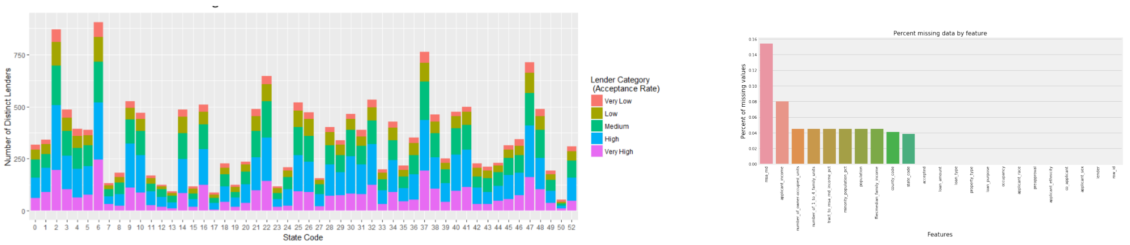
(a) Response features

(b) Loan purpose

(c) Loan amount

(d) Loan type

Figure 2. Plots for exploratory analysis



(a) District lenders by state code

(b) Features with missing info

Figure 3. More exploratory plots

4.1.2. Data pre-processing

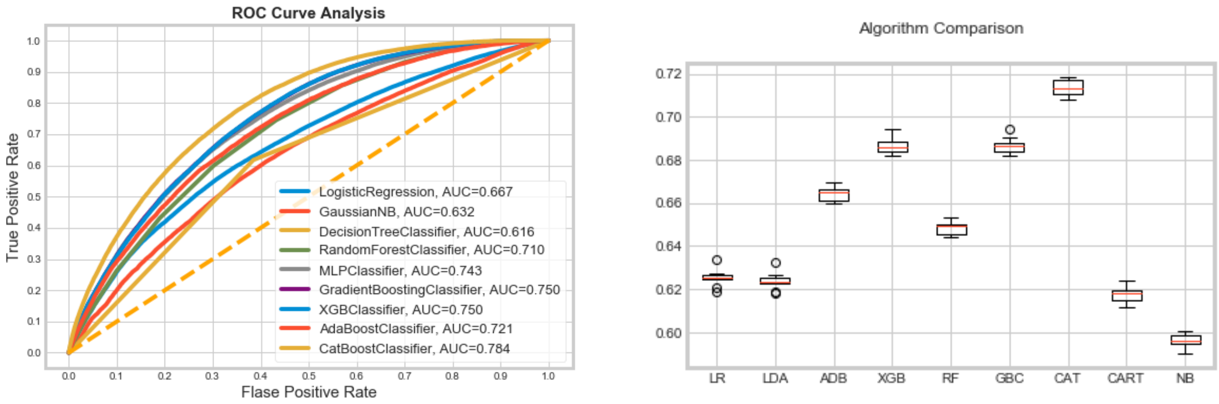
In order to replace the missing values (NA's) for both numerical and categorical features. Starting with the categorical features, the NA's encountered were replaced with mode of that feature. Also, categorical features were one-hot encoded which means each of the distinct category in a particular feature was converted to numerical fields. For numerical features, the NA's were replaced on case by case basis. Features like "applicant-income, number of owner-occupied units" were replaced with median as it handles the presence of outliers unlike mean imputation. The test set used has 150, 000 samples for each of the model.

4.1.3. Results and discussion

Table 2. Comparison of algorithms

Algorithm	**Score**	**Average time (fit)**	**Averge time (score)**(s)	**F1 score**	**AUC**	**precision**
Logistic regression	0.62	73.806	0.032	0.62	0.67	0.61
Random forests	0.69	19.045	1.271	0.64	0.71	0.68
Adaboost	0.67	28.632	1.664	0.63	0.72	0.66
XGBoost	0.69	28.322	1.342	0.65	0.75	0.68
Neural networks	0.68	27.234	1.123	0.73	0.66	0.66
Gradient Boosting	0.69	30.233	3.432	0.66	0.75	0.68
CatBoost	**0.732**	46.657	6.725	0.75	0.78	0.83
Decision trees	0.68	4.272	1.012	0.054	0.62	0.66

False positive rate is a method of committing type I error in null hypothesis testing when conducting multiple comparisons. For the problem used in this paper, false positive rate is an important metric as it would be a disaster if the system predicts a client would be given loan but in reality he was not. From table (2) the CatBoost algorithm achieved the highest accuracy. This means that the confusion metrics for CatBoost, value of correctly classified (TP + FN) is higher than the other six algorithms impleted. And with the least number of miss-classified. Other metrics such as f1 score, AUC and precision are also shown in table (2).

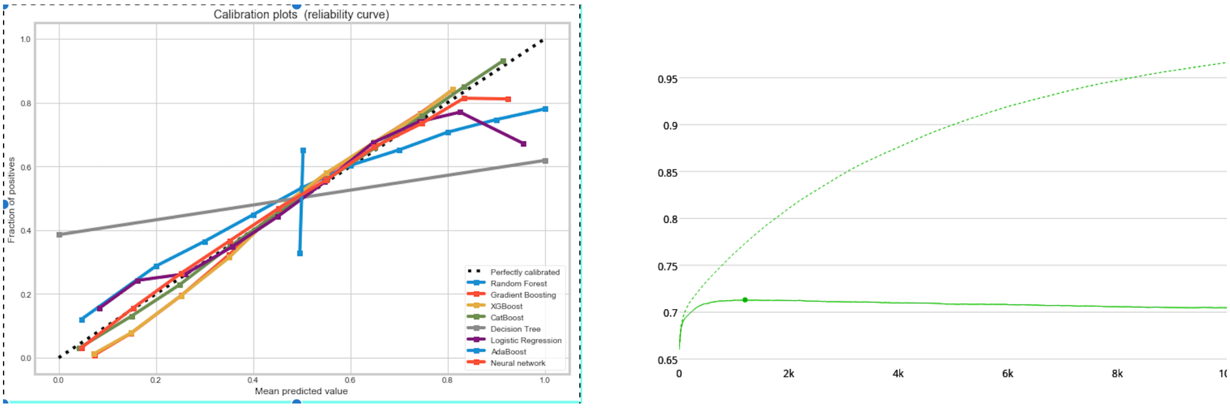


(a) ROC curves for the algorithms

(b) Boxplots for the algorithms

Figure 4. Shows the ROC curves and boxplots for the algorithms

After training the models on the training set and predicted the probabilities on the test set. We then obtain the True positive rate, False positive rate and AUC scores. From figure 4(a), CatBoost achieved highest AUC value of 0.78 which is closer to 1 than other classifiers. Also, figure 4(b) shows the comparison with other implemented algorithms. The names of the algorithms are written in short form where LR denotes Logistic regression, ADB for adaboost, RF denotes random forest,GBC for gradient boosting,CART for decision trees, NB for naive bayes, XGB denotes XGBoost and CAT for CatBoost algorithms. Box 8(b) shows the spread of the accuracy scores across each cross validation fold for each algorithm. Box 8(b) is generated based on the mean accuracy and standard deviation accuracy of the algorithms. In figure 5(a), the calibration plots for all the implemented algorithms are plotted, CatBoost method produced well calibrated predictions as it optimizes log-loss. Figure 5(b) the model performance graph for the CatBoost classifier.



(a) Calibration plots

(b) Model training graph

Figure 5. Shows the calibration and model training plots

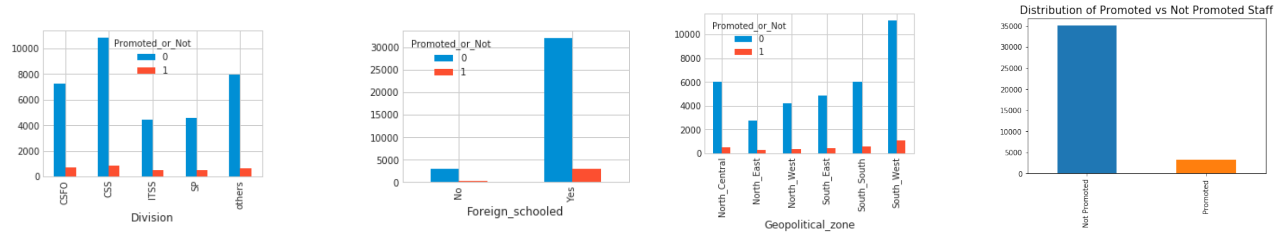
From the plot 8(b), it would suggest that CatBoost is perhaps worthy of further study on this problem due to it performance. The result has been presented in table 2 which contains the model accuracies, AUC, average time to fit and score.

4.2. Analysis 2: Staff promotion algorithm

HR analytics using machine learning will revolutionise the way human resources departments now operate. This will lead to higher efficiency and better results overall. This analysis uses predictive analytics in identifying the employees most likely to get promoted or not using historical staff promotion datasets [32]. We trained the model on dataset with 38312 samples and 19 features and tested on dataset with 16496 samples.

4.2.1. Exploratory analysis

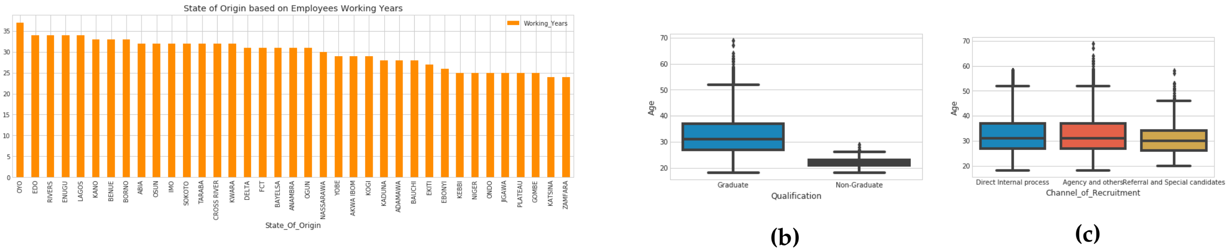
Before developing a predictive model, we need to understand the data points by exploratory analysis. In the exploratory analysis, we intend to find answers to some questions such as: (i) which features have missing values, (ii) features with outliers, (iii) is the response feature balanced? (iv) the distribution of the data points and so on. We presented some visualizations in figures 6 and 7 to answer these questions.



(a) Division **(b)** Foreign school **(c)** Geographical zones **(d)** Response feature

Figure 6. The panel of bar plots for some features

Starting with figure 6(a), the CSS class have the highest number (over 1000) of staff that was promoted in the Division feature. In figure 6(b), more than 30000 staff who got promoted were foreign-schooled and ≈ 2500 studied locally. Figure 6(c) shows the geographical zones of staffs, the South-West zone recorded the highest number of promoted staff while the North-East zone has the least number of promoted staff. Figure 6(d) shows the frequency for the two classes in the response variabe. It was observed that most of the staffs fall in the "not promoted" class, the ratio of "promoted" to "not promoted" is 8 : 92%.



(a) **(b)** **(c)**

Figure 7. Exploratory plots

Furthermore, in figure 7(a), employees from Oyo state (in South-West) appears to have the highest number of working years (38 years) while employees from Zamfara state had 24 years of working experiences. This could further support figure 6(c) with staff from South-West zone having highest number of promotion because of their working years. In figure 7(b), the graduate employees appear older than the Non-graduate employee. This could be due to the number of years spent studying before joining the workforce. The distribution of staff's channel of recruitment is shown in figure 7(c). In summary, features such as Division, Foreign schooled, geopolical zones, Qualification and working years had high impact towards staff promotion.

4.2.2. Data-preprocessing

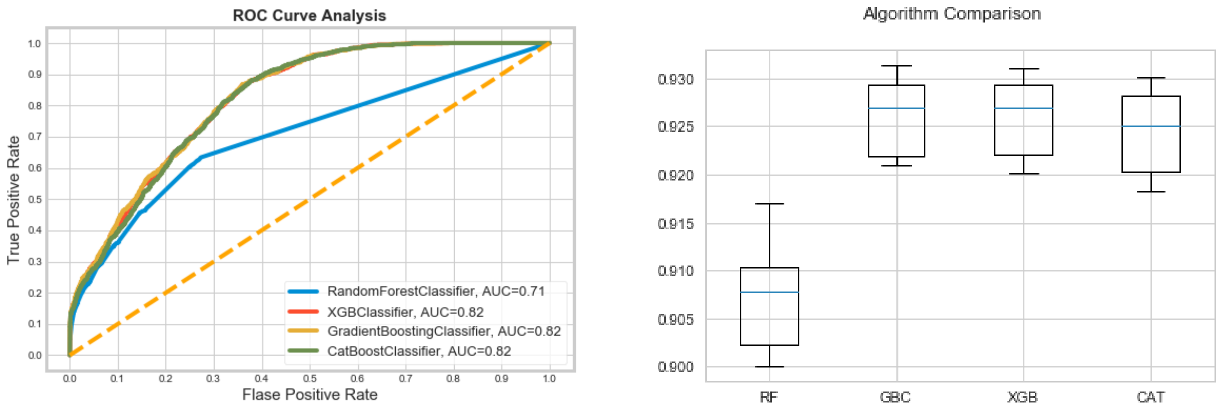
We replaced the missing values (NA's) for both numerical and categorical features. For the categorical features, the NA's encountered were replaced with mode of that feature. For Numerical features, the NA's were replaced on case by case basis. For the imbalanced response feature, it was balanced with a Resampling technique, in order to improve our prediction. After this step, we split the data and proceed to the prediction phase. The test set used has 16, 496 samples for each of the model.

4.2.3. Results and Discussion

Table (3) shows summary of the evaluation metrics for implemented algorithms. CatBost and XGBoost achieved the highest score with 94%. And when uploaded into kaggle online, we had difference of 0.01. Other metrics are also shown in this table.

Table 3. Algorithm comparison

Algorithms	**Score** (PC)	**Score** (Kaggle)	AUC	precision	**f1**
Random forest	0.93	0.88	0.71	0.70	0.94
XGBoost	0.94	0.93	0.82	0.93	0.92
Gradient Boost	0.90	0.84	0.82	0.93	0.95
CatBoost	**0.94**	0.93	0.82	0.91	0.95



(a) ROC curves for the algorithms

(b) Boxplots for the algorithms

Figure 8. Shows the ROC curves and boxplots for the algorithms

In figure 8(a) the AUC value of the applied algorithms are plotted, the RandomForestClassifier had the least value of 0.71 while other algorithms achieved 0.82. The distribution of the algorithms is shown in figure 8(b). Again, the RandomForestClassifier achieved the least value while other algorithms achieved high values. The model performance graph for the CatBoost algorithm showing how the model was trained, the number of iterations and the accuracy is shown in figure 9.

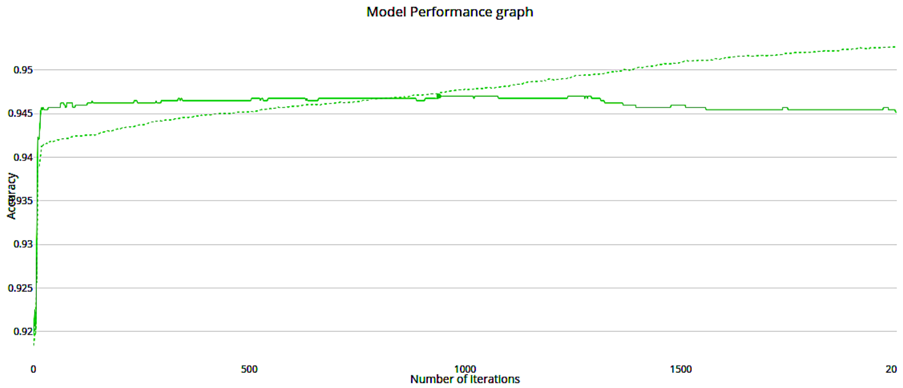
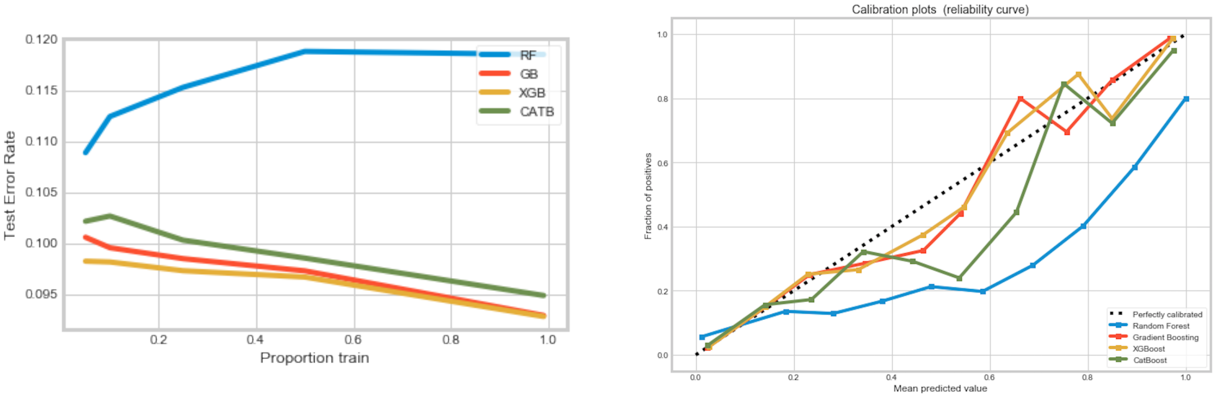


Figure 9. Model performance graph for the Catboost algorithm

The proportion of training set and the test error rate is plotted in figure 10(a). CatBoost and XGBoost had little error compared to other implemented algorithms.nIn figure 10(b) CatBoost and XGBoost methods produced well calibrated predictions.



(a) Test error rate

(b) Boxplots for the algorithms

Figure 10. Model reliability curves

5. Conclusion

Having applied all the mentioned algorithms in our methodology. Our objective is to develop predictive models. We performed two analysis: loan prediction and staff promotion. Each of the analysis started with exploratory analysis where we find insights from the data, then the data was cleaned, balanced and transformed for prediction. The machine learning algorithms discussed in this paper were then implemented and some metrics were used to evaluate the implemented models performance. CatBoost classifier did pretty good achieving the highest score (accuracy) in both applications. Other evaluation metrics also support the performance of this algorithm. We thereby recommend CatBoost classifier for predictive model.

Acknowledgements

We acknowledged the effort of all authors.

Author Contributions: All authors contributed, read and approved the final manuscript.

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] M. M. Muhammed and A. A. Ibrahim. (2019). A Comparative Analysis of Principal Components Analysis and Factor Analysis on a Binary Logistic Regression for Credit Scoring. Edited Proceedings of 3^{rd} International Conference of Professional Statisticians of Nigeria. (inpress)

[2] Anchal Goyal and Ranpreet Kaur. (2016). Loan Prediction using Ensemble Technique. International Journal of Advanced Research in Computer and Communication Engineering. Vol. 5(3)

[3] Zakaria Alomari and Dmitriy Fingerman. (2017). Loan Default Prediction and Identification of Interesting Relations between Attributes of Peer-to-Peer Loan Applications. New Zealand Journal of Computer-Human Interaction ZJCHI. Vol. 2(2).

[4] K. Ulaga Priya, S. Pushpa, K. Kalaivani and A. Sartiha. (2018). International Journal of Engineering and Technology. Exploratory Analysis on prediction of loan privilege for customers using random forest. Vol. 7(2.21) Pp. 339-341. https://doi.org/10.14419/ijet.v7i2.21.12399

[5] Li Ying. (2018). Research on bank credit default prediction based on data mining algorithm. The International Journal of Social Sciences and Humanities Invention Vol. 5(6): 4820-4823

[6] Xia Y., He L., Li Y., Liu N. and Ding Y. (2019). Predicting loan default in peer-to-peer lending using narrative data. Journal of Forecasting. Pp. 1–21. https://doi.org/10.1002/for.2625

[7] Long, Y., Liu, J., Fang, M., Wang, T., & Jiang, W. (2018, May). Prediction of Employee Promotion Based on Personal Basic Features and Post Features. In Proceedings of the International Conference on Data Processing and Applications (pp. 5-10).

[8] Machado, C. S. and Portela, M. 2013. Age and opportunities for promotion. IZA Discussion Paper No.7784.

[9] Blau, F. D. and DeVaro, J. 2007. New evidence on gender differences in promotion rates: An empirical analysis of a sample of new hires. Industrial Relations: A Journal of Economy and Society. 46, 3 (July. 2007), 511-550. DOI= https://doi.org/10.1111/j.1468-232x.2007.00479.x.

[10] Spilerman, S., and Lunde, T. 1991. Features of educational attainment and job promotion prospects. American Journal of Sociology. 97, 3 (Nov. 1991), 689-720. DOI= https://doi.org/10.1086/229817

[11] De Pater, I. E., Van Vianen, A. E., Bechtoldt, M. N., and KLEHE, U. C. 2009. Employees' challenging job experiences and supervisors' evaluations of promotability. Personnel Psychology. 62, 2 (Summer 2009), 297-325. DOI= https://doi.org/10.1111/j.1744-6570.2009.01139.x

[12] Woolley, A. W., Chabris, C. F., Pentland, A., Hashmi, N. and Malone, T. W. 2010. Evidence for a collective intelligence factor in the performance of human groups. Science. 330, 6004 (Oct. 2010), 686-688. DOI= https://doi.org/10.1126/science.1193147.

[13] Sarker, A., Shamim, S. M., Zama, M. S., & Rahman, M. M. (2018). Employee's Performance Analysis and Prediction using K-Means Clustering & Decision Tree Algorithm. Global Journal of Computer Science and Technology.

[14] Saranya, S., & Devi, J. S. PREDICTING EMPLOYEE ATTRITION USING MACHINE LEARNING ALGORITHMS AND ANALYZING REASONS FOR ATTRITION.

[15] G. James et al., An Introduction to Statistical Learning: with Applications in R, Springer Texts in Statistics 103, DOI 10.1007/978-1-4614-7138-7 8

[16] R. Punnoose and P. Ajit, Prediction of Employee Turnover in Organizations using Machine Learning Algorithms, (IJARAI) International Journal of Advanced Research in Artificial Intelligence, Vol. 5, No. 9, 2016 1-5.

[17] A. Liaw and M. Wiener, "Classification and regression by randomForest", R news, 2(3), 18-22, 2002.

[18] L. Breiman, Random forests. Machine Learning, 45(1), 5–32, 2001.

[19] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning with Applications in R. Springer textbook 2013

[20] Yoav Freud and Robert E. Schapire. (1999). A Short Introduction to Boosting. Journal of Japanese Society for Artificial Intelligence. Vol 14(5):771-780.

[21] G. James et al., An Introduction to Statistical Learning: with Applications in R, Springer Texts in Statistics 103, DOI 10.1007/978-1-4614-7138-7 8

[22] Shiruru, Kuldeep. (2016). An Introduction to Artificial Neural Network. International Journal of Advance Research and Innovative Ideas in Education. 1. 27-30.

[23] Kumamoto University http://www.cs.kumamoto-u.ac.jp/epslab/ICinPS/Lecture-2.pdf

[24] J. H. Friedman, "Greedy function approximation: a gradient boosting machine", Annals of statistics, 1189-1232, 2001.

[25] Alexy N. and Alois K. (2013).Gradient Boosting Machines: A Tutorial. Frontiers in Neurorobotics. Vol 7(21) pp 3.

[26] S. Lessmann and S. Voß, "A reference model for customer-centric data mining with support vector machines", European Journal of Operational Research 199, 520–530, 2009.

[27] T. Chen and C. Guestrin. (2015) XGBoost: Reliable Large-scale Tree Boosting System.

[28] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. In Advances in Neural Information Processing Systems (pp. 6638-6648).

[29] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. The annals of statistics, 28(2):337–407, 2000.

[30] L. Mason, J. Baxter, P. L. Bartlett, and M. R. Frean. Boosting algorithms as gradient descent. In Advances in neural information processing systems, pages 512–518, 2000.

[31] Kohavi, R. and Provost, F. (1998) Glossary of terms. Machine Learning—Special Issue on Applications of Machine Learning and the Knowledge Discovery Process. Machine Learning, 30, 271-274

[32] Microsoft capstone project https://www.datasciencecapstone.org/ (Assessed in April 2019)

[33] https://www.kaggle.com/c/intercampusai2019 (Assessed in August 2019)