Data Mining Approaches for Forecasting Customer Transactions of Santander Group

***Abstract*— Every banking sector works on the principle of "client first", as clients satisfaction is essential for the growth and they are revenue generators. Providing excellent services to customers has become a new challenge and competitive within the banking sectors. However, customers are likely to do business, if provided with excellent service. Data mining techniques play a vital role in prediction techniques and every industry started their journey towards the latest technologies and to get the most out of the technologies. The Banco Santander is a wholly owned subsidiary of the Spanish Santander group is trying to take help from machine learning experts to improve their services. That is, the bank has provided anonymous potential customer transaction data, and trying to predict upcoming transactions in advance. It will help the bank to provide financial support and services to customers and in anticipating customer needs in more organised ways. The analysis embraced with, the approach towards handling sparse data, feature selection techniques and data mining forecasting techniques such as linear regression, Gradient booster and eXtreame Gradient Boosting algorithms. These techniques will help in understanding the customer needs in advance using forecasting techniques, thereby improving the services. The vast and sparse dataset obtained from ongoing Kaggle competition.**

Keywords - Data mining, Forecasting, Prediction, sparse data

I. INTRODUCTION

The Banking is the sector[1] of the economy devoted, which interact with financial assets and more likely to provide financial personalise service for essential growth. The lending activities of the bank's scripts capital markets. Around 80% of customers expect digitalisation in banks, to enable their tasks entirely and promptly. Identifying and analysing the customer's needs will help in customer retention and can benefit more in revenues. Attracting new customers and retention of existing customers involves the higher cost. Similarly, clients have more technical knowledge about the sectors which will help in oral promotions about the bank. Data mining is one of the essential techniques incorporated by all the industries to improve the services.

Santander Group started its journey towards big data in 2014. It has developed a strategy to give the most out of advanced technology to satisfy and attract the customers and to change old IT paradigms. Santander Groups[2] aims to determine the value of the customer's transactions patterns to determine the imminent value of transactions for each credible customer and scale of arts for the financial services. With the help of dynamic system developed banking sector can proactively ask for assistance to the customers, in a way to provide more services and care about the clients. In order to personalise the services at scale, the assistance of machine learning experts is indeed. Given the massive number of features with sparse data, Santander Group seeking for help by hosting the competition. The objective of Kaggle competition 2018 is to predict the value of transactions for potential customers as closely as possible. By having the model in a production environment, the Santander can take proactive steps to improve the services and happiness of customers before they would get attracted to other businesses. Most crucial steps of the system are getting information out of sparse data and imbalance of the transactions/non-transactions. The inaccurate results may provide wrong information about the customer and thereby dissatisfaction. Extracting patterns from historical data is a challenging task, and this can be achieved only by statistical methods and data mining methods. This paper will try to answer the research questions:

1. *How simple linear model predicts on sparse data?*
2. *What methodologies can be used to control the sparse data from producing inaccurate predictions?*
3. *Role of Gradient Boosting Algorithm and eXtreme Boosting algorithms in predicting sparse data.*

For the analysis, attempt has been given to apply different mining methodologies for better and accurate results. This paper consists of detail study as follows. Firstly, the understanding of data is essential before applying methodologies. Exploratory data analysis is our first step. Secondly, the data cleaning, as substantial sparse data it is important to investigate on missing values and techniques to handle sparse data. Thirdly, feature engineering to check for inconsistent data and techniques to convert into consistent data. Lastly, modelling and evaluation techniques to predict the potential customer transaction values. The paper also includes result and discussions and feature work.

II. LITERATURE REVIEW

Wichard et al., describes different mathematical methods for predicting customer credit card transition, the autoregression AR(L) model is used via different methods like Yule-Walker equations, least square fit in the time domain for the prediction and wavelet methods for determining hidden patterns or transaction trends. Here, the author has chosen the data of Mastercard transactions in the Netherlands to predict transactions of users for next fourteen months based on previous years data. Different values of L and K were selected, where L and K denote the dimension of the autoregression model and number of autocorrelation coefficient respectively. The results of these models compared and evaluated using Mean Absolute Percentage Error to increase the efficiency in forecasting of the model. The author further investigates models like nearest neighbour, perceptron, regression and radial basis using ensemble method and compared the outcomes of ensemble method with the previously described autoregression model. The results showed ensemble method outperformed the autoregression model [3].

Lipasti et al. used the last outcome scheme method in which the results were stored after the last execution of the instructions and would predict the same data values if the instructions will encounter in the future. The accuracy gained was 49% for power pc architecture which means 49% of the register result producing instruction can be predicted by storing last result of each instruction. The results showed a lot of mispredictions [4], [5].

Wang and Franklin investigate two techniques namely stride based predictor and 2-level predictor. Stride-based predict the outcomes by analysing the difference between successive values and last values whereas the 2- level model predicts by analysing and storing unique results of the instruction. The results of these two techniques have shown enhancement by generating 50% accuracy than last outcome scheme [6].

According to Spring et al., customer transaction database is used to analyse customer behaviour patterns of purchasing, which helps the banking sector to predict the potential value of the customer. Individual transactions are segregated and combined into a specific value which creates a potential value for the customers. The different models used for predicting potential customer value are, linear regression, probit model and naïve bayes. In further research, author used two methods viz test of Franses and Mean Absolute Percentage Error (MAPE) for comparing the results of the above-mentioned models to find the most efficient model. The MAPE profitability achieved using these models are 19.4%, 19.5% and 20.5% respectively [7], [8].

The self-organised neural network is used to classify bank user into profitable criteria by groups based on the bank's customer needs, transactions frequency, purchasing behaviour and present epoch. The model used to enhance profitable margin by providing exciting offers to the customers who meets to the profitable criteria is called as Behavior Scoring. For the new applicant, the score is calculated based on their age, salary and marital status and decision made whether to grant the credits. This type of model is called as Credit Scoring. These two models are essential in data mining. It helps to classify the customers by neural networks. The behaviour Scoring can be judge based on real-life examples such as frequent use of credit card. The behaviour model is comprising of an Apriori association rule inducer which helps to find relationships between scoring features of existing customers and predicting future credit status. The neural networks segmentation analysis is capable of contesting the marketers and research area [9].

Another paper describes predictability of data values which defines two predictive models namely computational and context-based predictors by which we can discover customer's potential value. The author used stride value prediction and last value prediction in the computational model to predict the future transaction value of customer whereas in the context-based model the author used recent value and previous value history to predict the transaction values. The accuracy achieved using these models were compared, and results showed context-based model gained more accuracy than the computational-based model with a margin of 20%. To sum up, context-based outperformed than computational for predicting potential customer values [10].

Verhoef et al., performed statistical as well as data mining techniques in database marketing and customer relationship management, which in turn help banking domains to discover the relationship between customers and frequent transaction to predict their future transactions. To predict the transaction values, potential customers are segmented into clusters. Numerous methods used for prediction were probit analysis, crosstab, neural network and chi-square. Here, as described by author crosstab and probit was popular among companies and widely accepted. As the neural network is black box technique, it is less widely used as compared with other models [8].

Customer loyalty is crucial factor when dealing with banking transactions. Baumann (2006) describes an efficient way of predicting satisfaction level and behavioural patterns. Based on these parameters practical implications were to predict transactions of customers and demonstrated the significance of customer satisfaction on transaction patterns. Here, the author describes SERVQUAL dimensions for predicting satisfaction level. In conclusion, three models namely short term, willingness to recommend and long-term can have high impact on predicting customer transaction in banking sector [11].

III. METHODOLOGY

This project is performed using the Cross-Industry Standard Process for Data Mining (CRISP-DM) approach consists of 5 phases. Exploratory Data Analysis, Data Cleaning, Feature selection, modelling and evaluations are the phases discussed as follows:

A. Exploratory data analysis(EDA)

The EDA will help in understanding the structure of the dataset, balance and imbalanced data and to steps for further analysis. The predictive outcome depends on the factors mentioned for EDA and are discussed as follows. The dataset contains the potential customer transaction values, which are divided into training and testing data.

Training data has a dependent variable as "target" and independent variables have huge anonymous customer data. Training data has 4459 observations and 4993 variables, the interesting part of the training dataset is number columns are more than the number of observations. As the training dataset is huge, few lines of the structure of the training dataset are as shown in figure 1.

```
Classes 'tbl_df', 'tbl' and 'data.frame':       4459 obs. of  4993 variables:
 $ ID       : chr  "000d6aaf2" "000fbd867" "0027d6b71" "0028cbf45" ...
 $ target   : num  38000000 600000 10000000 2000000 14400000 2800000 164000 600000 979000 460000 ...
 $ 48df886f9: num  0 0 0 0 0 0 0 0 0 0 ...
 $ 0deb4b6a8: int  0 0 0 0 0 0 0 0 0 0 ...
 $ 34b15f335: num  0 0 0 0 0 0 0 0 0 0 ...
 $ a8cb14b00: int  0 0 0 0 0 0 0 0 0 0 ...
 $ 2f0771a37: int  0 0 0 0 0 0 0 0 0 0 ...
 $ 30347e683: int  0 0 0 0 0 0 0 0 0 0 ...
 $ d08d1fbe3: int  0 0 0 0 0 0 0 0 0 0 ...
 $ 6ee66e115: int  0 0 0 0 0 0 0 0 0 0 ...
 $ 20aa07010: num  0 2200000 0 0 2000000 ...
 $ dc5a8f1d8: num  0 0 0 0 0 0 0 0 0 0 ...
 $ 11d86fa6a: num  0 0 0 0 0 8000 0 0 0 0 ...
 $ 77c9823f2: int  0 0 0 0 0 0 0 0 0 0 ...
 $ 8d6c2a0b2: int  0 0 0 0 0 0 0 0 0 0 ...
 $ 4681de4fd: int  0 0 0 0 0 0 0 0 22000 0 ...
 $ adf119b9a: int  0 0 0 0 0 0 0 0 0 0 ...
 $ cff75dd09: num  0 0 0 0 0 0 0 0 0 0 ...
 - attr(*, "problems")=Classes 'tbl_df', 'tbl' and 'data.frame':       6 obs. of  5 variables:
  ..$ row     : int  1545 2083 2587 2687 2815 3273
  ..$ col     : chr  "56cb93fd8" "ff3b49c1d" "c059f2574" "b77c707ef" ...
  ..$ expected: chr  "an integer" "an integer" "an integer" "an integer" ...
  ..$ actual  : chr  "2243000000" "2243000000" "2243000000" "2243000000" ...
  ..$ file    : chr  "'Downloads/train 3.csv'" "'Downloads/train 3.csv'" "'Downloads/train 3.csv'"
ads/train 3.csv'" ...
 - attr(*, "spec")=List of 2
  ..$ cols   :List of 4993
  .. ..$ ID       : list()
  .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
  .. ..$ target   : list()
  .. .. ..- attr(*, "class")= chr  "collector_double" "collector"
  .. ..$ 48df886f9: list()
  .. .. ..- attr(*, "class")= chr  "collector_double" "collector"
  .. ..$ 0deb4b6a8: list()
  .. .. ..- attr(*, "class")= chr  "collector_integer" "collector"
```

Figure1: The structure of training dataset

On the other hand, there are two challenges need to be handled with testing data. One is testing data which 5 times more than training set (49342 observations and 4991 variables). Second is the testing dataset has no dependent variable, as the challenge is given to predict the "target" variable. Hence, to handle this situation the decision is made to divide the training data into 80% and the remaining 20% is allocated to testing data to get the better results for our convenience. In training and testing datasets, all independent variables are numeric and dependent variable "target" is a continuous variable.

Dependent variable:
The dependent variable labelled as "target" and the range of the dependent variable varies from 30000 to 40000000. From the histogram plot, the dependent variable is right skewed, and the majority of values are very low value, i.e. zero.

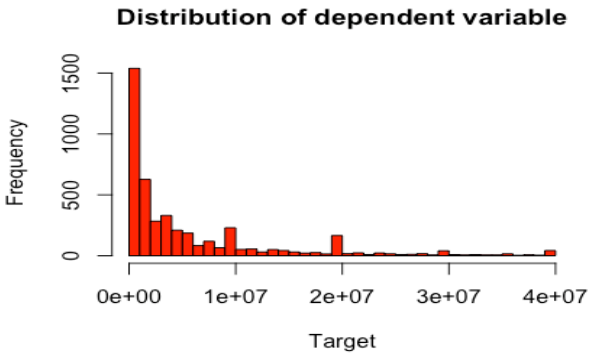

Figure 2: Distribution of dependent variable

Hence, let's plot the graph by taking the natural logarithm of the dependent variable. From figure 3 the logarithm of a dependent variable looks bell-shaped, which is much better than figure 2. The main advantage of the natural logarithm that does not affect on the values of the outcome but the magnitude of the coefficient of the dependent variable.
The log transformation of target variable will help better due to a wide range of the target.

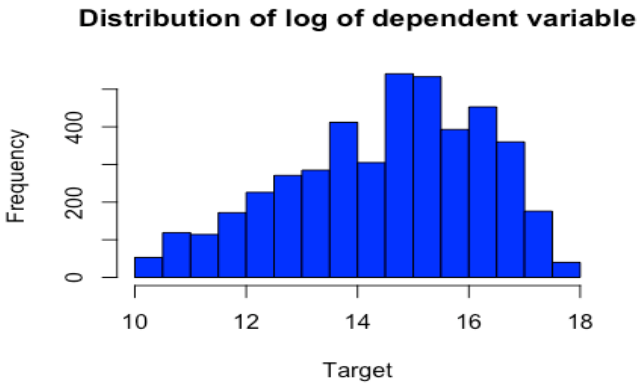

Figure 3: Distribution of log of dependent variable

Independent variables:

The count of independent variables is enormous. Total 4459 observations and 4992 variables, the primary challenge is in handling the number of columns are more than the number of observations. From the glance of the dataset, there are many zeros in the dataset. From the figure 4, it is clear that 95% of the training dataset contains zero values, in which each row contains the minimum 4000 zeros and remaining values are huge. Such datasets are known as sparse data. We will handle sparse data using feature selection section techniques.

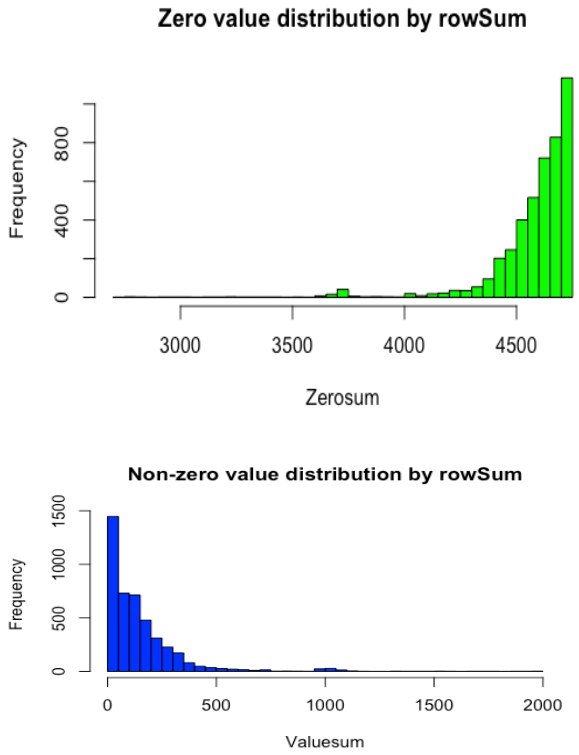

Figure 4: Zero value and Non-zero value distribution by rowSum

B. Data cleaning

The given data looks clean at first glance. However, there are numerous problems to run the machine learning algorithms. Following figure 6 indicates the cleaning steps taken for the dataset:

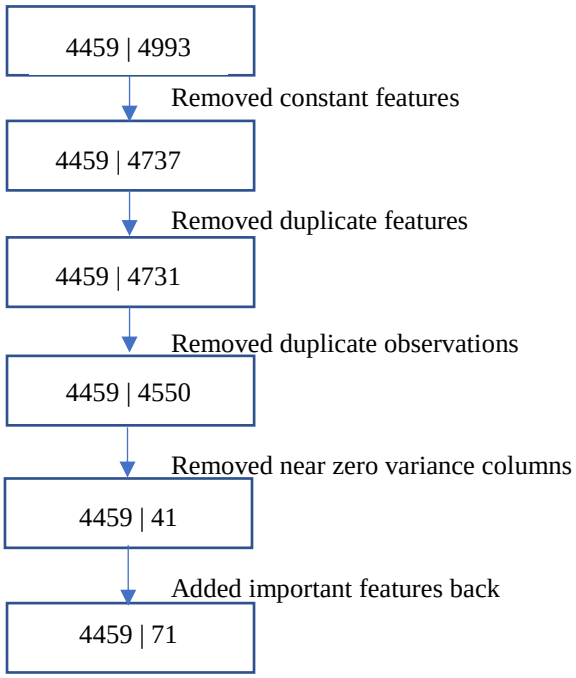

Figure 5: Steps for data cleaning

From the figure 5, initial steps are taken to reduce the number feature is by removing the constants. The number of 256 features are removed, which will not contribute much to the modelling. So all the features that have zero standard deviation are removed. 6 redundant features and 181 duplicate observations that are removed using duplicate removals.

Furthermore, near zero variance columns were removed and this made huge contribution to the data cleaning step. Near zero variance features are almost like constant features across the samples, this will lead to the inaccurate prediction of the dependent variable. It also hindrances the critical features from contributing to the model. From the figure 6, it shows that 97% of the features are near zero variance. Therefore, the count is now down from 4550 to 41.

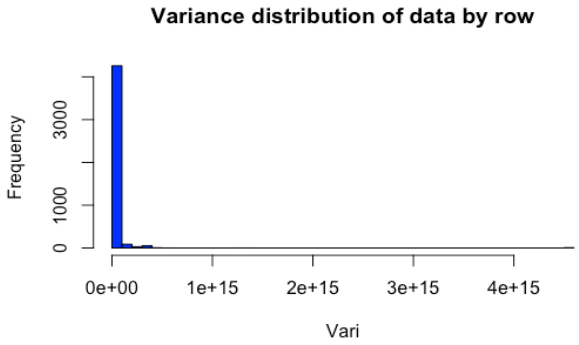

Figure 6: Near zero variance plot

Removal with these features can confirm by another two methods such as correlation analysis and important features from the base models.

Thus, our EDA has given enough information about the dataset such as frequency of zero values, non-zero values, the variance of data and distribution of the dependent variable.

C. Feature selection

Feature selection will help to remove the noise from the dataset. We chose feature selection instead of dimensionality reduction technique because dimensionality reduction did not yield good results such as PCA and K-means due to sparse data shown in figure 7 Scree plot below.

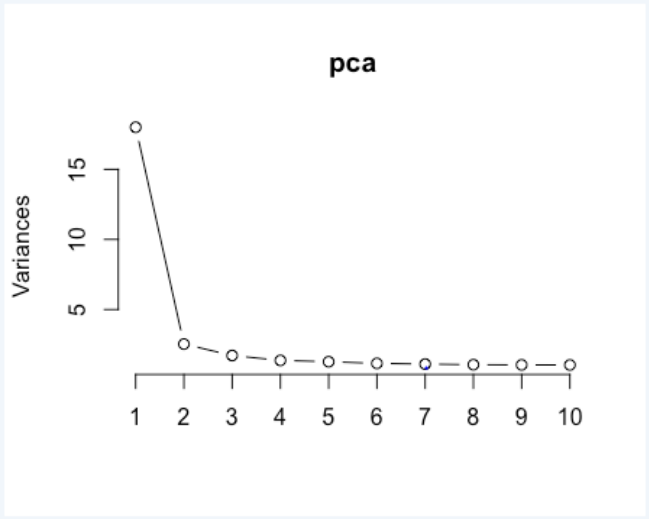

Figure 7: Scree Plot

Some of the feature selection techniques used here are:

Columns with near zero variance - Having sparse data with 95% of the variables contain zero. There are many columns with zero variance, which will not contribute much to the model. However, we may lose some important variables. Hence we cross check again using Fselector, random forest to match the columns selected by non-zero variance. As mentioned near zero variance has reduced the number of columns from 4550 to 41, which is a drastic change in the variables.

Random forest – We have extracted the important variables from random forest using "randomForest" package and the plot is as shown in figure 8.

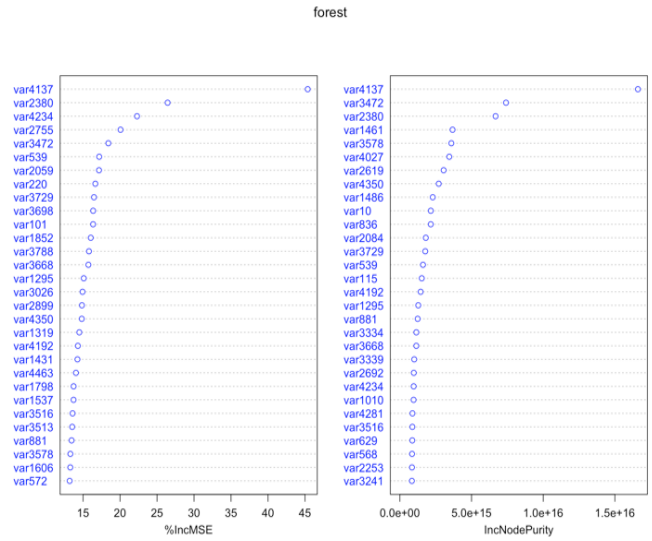

Figure 8: Extraction for important variables

The random forest has provided important features; however, we lost some features in near zero variance. Means, some near zero variance features were important contributors to the model. Union of both methods, near zero variance and important features will give 71 features in total. Hence, missing important variables by near zero variance removal is recovered by the model.

Furthermore, it is important to observe that data contains many zero's. It indicates the customer has not made any transactions but may perform the transaction in the future. Hence, sum of zero is added as an extra column to indicate the number of zeros in each row. Here, special care must be taken by not including the dependent variable (target), as this is form of data leakage and it hinders process of modelling. Similarly, added the sum of all non-zero values in each row by creating another column. With this, all the information will be provided to the model for learning process.

D. Modelling

Ever the prediction is discussed, the linear regression is the model comes to everyone's mind. So, let's start with the simple multiple linear regression, as the dependent variable is continuous. Remember, the log transformation has been taken for dependent variable as variables were not distributed linearly. Before proceeding with model building, linear regression[7] has few assumptions, which needs to be tested thoroughly else model will lead to incorrect predictions.

Assumption 1: Linear relationship between independent and dependent variable
Observation: Relationship is linear
Status: Passed

Assumption 2 : Mean of residuals should be zero
Observation: mean(lm.model$residuals) = 3.617543e-17
Mean is close to zero.
Status: Passed

Assumption 3: Homoscedasticity
Observation: From figure 10 residuals vs fitted plot shows residuals are distributed across the horizontal line.
Status: Passed

Assumption 4: Autocorrelation
Observation: Durbin Watson test is performed and failed to reject the null hypothesis. Hence, there is no autocorrelation among residuals.
Status: Passed

Assumption 5: Multicollinearity
Observation: From the figure 9, there is no strong correlation among independent variables. Maximum correlation value exists is 0.6 and is moderately strong correlation.
Status: Passed

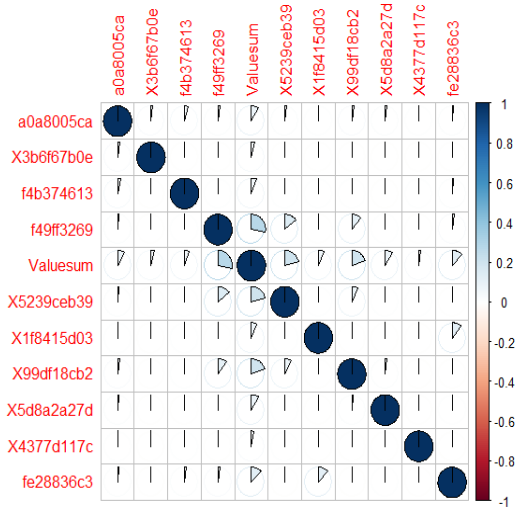

Figure 9: Check for Multicollinearity

Assumption 6: Multivariate normality
Observation: From the figure 10, the normal Q-Q plot indicates that residuals are plotted linearly. And there are no outliers.
Status: Passed

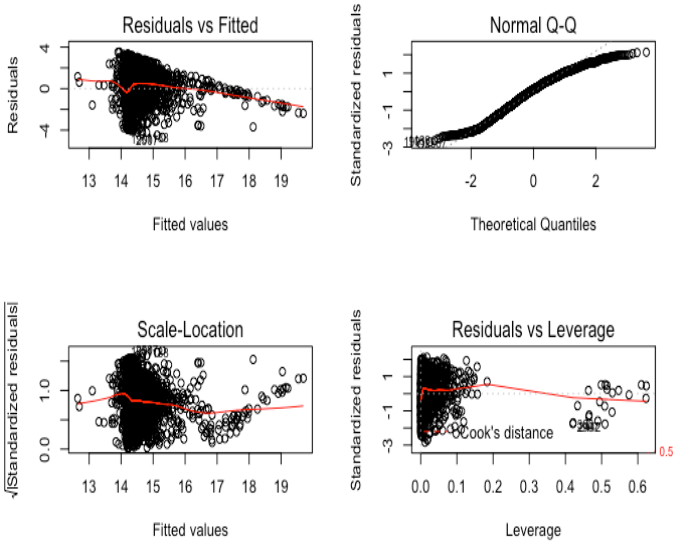

Figure 10: Linear regression model

i) Simple Linear Regression :

After passing all the assumptions, the simple Linear regression model built is used as the base model for further analysis and studies. It is relatively simple model and did not yield the best results. As the target variable is distributed randomly, the linear model was not able to get best fit for the residuals. The R squared value is 0.11, it explains 11% of the behaviour of the dependent variable and it is too less.

The figure 11 shows the Prediction vs Test plots. From the linear model, the evaluated Root Mean Square Error (RMSE) value as 1.659 and Root Mean Squared Log Error (RMSLE) as 0.112.

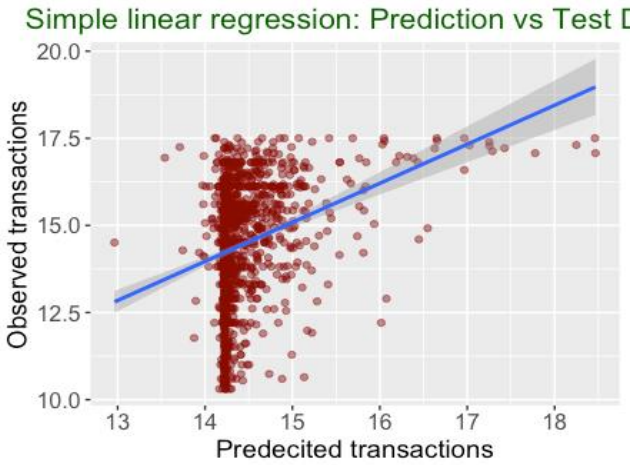

Figure 11: Prediction vs Test plots

ii) Gradient Boosting for linear model (GBM):

The enhance the performance of the regression model and reduce the model error the gradient boosting model (GBM) implemented. The goal of the Gradient boosting is it ensembles methods and enables the complex model by re-fitting sub-models like typical decision trees to pseudo-residuals. In the GBM model, the shrinkage level also known as learning rate is maintained at 0.001 attempts to control generalisation error. The training factor is for initial segmentation settled on 1.0 and interaction depth is maintained at 3 which generate relation among groups of variables. The parameters tuned to use Gaussian distribution and with cross-fold validation of 5. Another critical tuned parameter is the number of trees, initially started with 1000, after analyzing print (ntress) = 264. The model stops predicting at 264 iterations and it is the maximum point that can be produced.

In the figure 12, the graph represents evaluation between Squared error losses and number of tree iterations. The black line is the training data and the green line is testing data in the summary of model performance for fitting GBM [12]. The plotted graphs show that after applying 400 trees, its RMSE values is 1.50 and testing data stops performing at 264 trees shown by standing dotted blue line. The summary of the model builds the relative importance of the variables. The gaussian learns from its error rate at each iteration and residuals calculated at each iteration as follows:

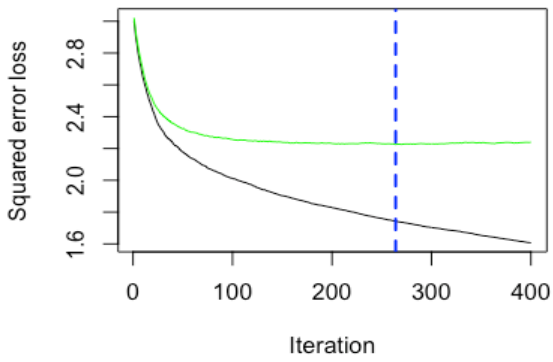

Figure 12: Evaluation between Squared error losses and number of tree iterations

[e1_predicted= y - y_predicted1]
[y_predicted2 = y_predicted1 + e1_predicted]

The test accuract is less than the training accuracy, which is called as overfitting. The main disadvantage of GBM is overfitting on the unseen data. Gradient boosting grows sequentially that means each new tree is derived based on previous decision tree and mistake are recorded then incurred in the next round. This process repeat continued till it gets low values error and weighted average of total derived predictions. But this approach mitigates the overfitting issue. The optimisation achieved using loss function generalising the model. The RMSE and RMSLE illustrate relatively great performance than Normal Linear regression, but it needs to be tuned with XGBoost for higher accuracy and to avoid the overfitting of the model.

iii) Extreme Gradient Boosting model (XGBoost):

After applying linear regression and Gradient boosting (though it is overfitting), a good accuracy has been achieved and both models working as expected. To enhance the performance and speed XGBoost is used[13]. The XGBoost implementation is gradient boosting using decision trees, pushes the limits for the computational resources. The reason behind implementing the XGBoost into our model is, it provides extra features such as Sparse Aware implementation, tree construction using all CPU cores in parallel block structure, use of clusters for our large dataset, out-of-core computing, cache optimization, efficient use of memory resources and compute time. There are some obvious factors like model performance and execution speed makes model efficient by benchmarking Random Forest implementation. It overcomes classification and regression modeling problem by dominating on out structured dataset and predict the residuals or errors of prior models which are then combined together to predict final result.

To improve the accuracy over linear model, eXtreme Gradient Booster is another option, the main advantage is it will control the overfitting of the model by tuning the parameters. Here the parameters, used are booster as "gblinear", learning rate as 0.1 which will control the overfit of the model (tuned after 0.3 and 0.2), maxdepth as 15 (decided after tuning 5, 8 and12), min_child_weight to control the tree partitioning as 0.6 (decided after tuning 1 and 0.6), subsample (tuned after 1 and 0.8), alpha value an important parameter for linear regression as 0.01 (tuned after 0.0001, 0.001 and 0.1), lastly nrounds as 1000 (tuned after 2000, 1500 and 1200). After tuning all the parameters of XGBoost, run the model and achieved the RMSE of 0.461 and RMSLE of 0.0321. Mathematically, RMSE value is in the range of 0.7.

sqrt((sum((Test$target) - (prediction))^2)/length(test1)) = 0.7073589.

Hence, the testing has achieved good accuracy than the GBM and linear model. XGBoost has controlled overfit of the model by tuning early stopping. The figure 13 shows that Extreme Gradient boosting evaluated greater prediction against the test data.

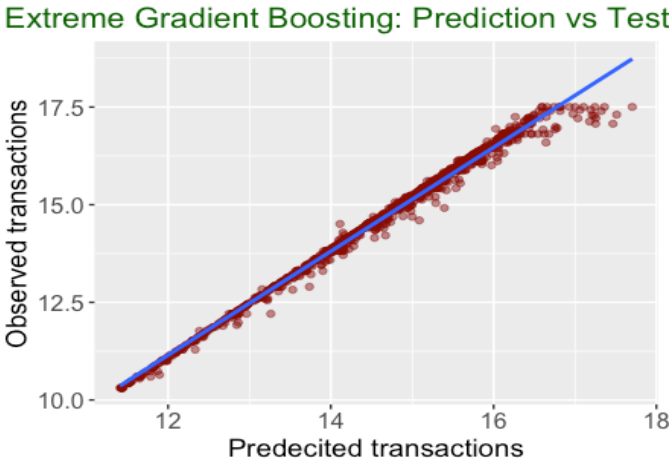

Figure 13: Observed vs Predicated transactions

Some logs of the model to show the performance

niter: 992
best_iteration : 892
best_ntreelimit : 892
best_score : 0.498451
nfeatures : 62

E. Analysis and Results

The performance measures considered as RMSE and RMSLE. In linear regression model the R square value received is 0.11, means it explains 11% of the behaviour of dependent variable and it is too less to predict the unseen data. The reason for choosing RMSE is, it refers to the standard error, allows to build the model by considering the confidential intervals around the linear regression model. It is equivalent to Z value. Similarly, RMSLE is another measure chosen, as the dependent variable "target" is log transformed. Mathematically, the RMSE value should be around 0.7 as calculated above using the formula.

In simple linear regression R square achieved is 0.11 and RMSE value is 1.6. However, our mathematical calculations shows RMSE values I 0.7. The linear model is not able to fit the best fit line due to sparse data distribution.

Residual standard error: 1.675 on 3280 degrees of freedom

Multiple R-squared: 0.1018, Adjusted R-squared: 0.08513

F-statistic: 6.096 on 61 and 3280 DF, p-value: < 2.2e-16

The stepwise linear regression tried also did not yield better results and it was tedious to remove and add columns from these huge number of the set of features. It's been handled in a better way using feature selection techniques.

Further, the analysis performed by adding boosting algorithms. The first attempt of boosting performed using GBM, which has given better performance with RMSE value of 1.5 and RMSLE of 0.1. These are the results of underfit with ntree value of 400 and tree stop growing after 264. Below is the snippet of the trained GBM model.

---

gbm(formula = train$target ~ ., distribution = "gaussian", data = train,

n.trees = 400, interaction.depth = 3, shrinkage = 0.1, cv.folds = 5, verbose = FALSE)

A gradient boosted model with gaussian loss function.

400 iterations performed.

The best cross-validation iteration was 264.

There were 71 predictors of which 59 had non-zero influence.

Finally, to control the overfit curse of GBM, the XGBoost is implemented. XGBoost performed better than GBM thereby controlling the overfit. The RMSE value of 0.46 and RMSLE value of 0.03 is achieved, which is much better than the other models.

---

params (as set within xgb.train):

booster = "gbtree", obejctive = "reg:linear", eval_metric = "rmse", eta = "0.1", max_depth = "15", min_child_weight = "1", subsample = "0.8", colsample_bytree = "0.8", alpha = "0.01", verbose = "0", silent = "1"

callbacks:cb.evaluation.log()

cb.early.stop(stopping_rounds = early_stopping_rounds, maximize = maximize, verbose = verbose)

# of features: 71

niter: 992, best_iteration : 892, best_ntreelimit : 892, best_score : 0.498451, nfeatures : 62

From all these results, we have concluded that XGBoost model performed better for sparse data and in predicting the customer transactions of Santander Group due to the values gained for RMSE. This algorithm shows lowest error between real and predicted values.

IV. Future Work

In future work, complex models such as artificial neural networks and other deep learning techniques can be implemented using multi-GPU and multi-node systems to acquire higher performance.

V. Conclusion

The analysis proposal used three main techniques for predicting the customer transactions. Before performing any model, the exploratory data analysis was performed to understand the datasets. Later, feature selection has reduced the dimensionality of the dataset, which made huge impact on our model. Further, modelling such as simple linear regression was used, which was unable to fit the best fit line. On top of the linear model, gradient boosting model and extreme gradient boosting models are used to check the prediction against testing data. XGBoost performed well by controlling the overfitting of the unseen data. However, more models such as ANN and deep learning techniques can be performed. Better the feature selection, best will be the model performance. In future, the feature selection techniques needs to be explored to tune the models.

VI. Reference

[1] S. Hatko, “The Bank of Canada 2015 Retailer Survey on the Cost of Payment Methods : Nonresponse The Bank of Canada 2015 Retailer Survey on the Cost of Payment Methods : Nonresponse,” no. 107, 2015.

[2] D. V. D. Elsen and En, “Santander Customer Satisfaction,” *Kaggle*, 2017.

[3] U. P. C. M. U. Wichard, “DFG-Schwerpunktprogramm 1114 Mathematical methods for forecasting bank transaction data,” no. January, 2003.

[4] M. H. Lipasti, C. B. Wilkerson, and J. P. Shen, “Value locality and load value prediction,” *ACM SIGPLAN Not.*, vol. 31, no. 9, pp. 138–147, 1996.

[5] M. H. Lipasti and J. P. Shen, “Exceeding the dataflow limit via value prediction,” *Proc. 29th Annu. {IEEE}/{ACM} Int. Symp. Microarchitecture, 1996. {MICRO}-29*, pp. 226–237, 1996.

[6] K. Wang and M. Franklin, “Highly accurate data value prediction using hybrid predictors,” *Proc. 30th Annu. ACM/IEEE …*, pp. 281–290, 1997.

[7] R. Huisman and C. Huurman, “Predicting Customer Potential Value,” no. 0, p. 2014, 2003.

[8] P. C. Verhoef, P. N. Spring, J. C. Hoekstra, and P. S. H. Leeflang, “The commercial use of segmentation and predictive modeling techniques for database marketing in the Netherlands,” *Decis. Support Syst.*, vol. 34, no. 4, pp. 471–481, 2002.

[9] N. C. Hsieh, “An integrated data mining and behavioral scoring model for analyzing bank customers,” *Expert Syst. Appl.*, vol. 27, no. 4, pp. 623–633, 2004.
[10] Y. Sazeides and J. E. Smith, “The predictability of data values,” *Proc. 30th Annu. Int. Symp. Microarchitecture*, pp. 248–258, 1997.
[11] C. Baumann, S. Burton, G. Elliott, and H. M. Kehr, “Prediction of attitude and behavioural intentions in retail banking,” *Int. J. Bank Mark.*, vol. 25, no. 2, pp. 102–116, 2007.
[12] M. Seeger, “Gaussian Processes for Machine Learning University of California at Berkeley,” 2004.
[13] D. Nielsen, “Tree Boosting With XGBoost Why Does XGBoost Win ‘Every’ Machine Learning Competition?,” *NTNU Tech Rep.*, no. December, p. 2016, 2016.
[14] a Liaw and M. Wiener, “Classification and Regression by randomForest,” *R news*, vol. 2, no. December, pp. 18–22, 2002.
[15] C. Bishop and N. Nasrabadi, “Pattern recognition and machine learning,” *Pattern Recognit.*, vol. 4, no. 4, p. 738, 2006.
[16] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” pp. 785–794, 2016.
[17] J. Xifra and E. Ordeix, “Managing reputational risk in an economic downturn: The case of Banco Santander,” *Public Relat. Rev.*, vol. 35, no. 4, pp. 353–360, 2009.
[18] B. Lantz, *www.allitebooks.com*. 2013.
[19] P. Romanski, L. K. Maintainer, and L. Kotthoff, “Package ‘FSelector,’” *Cran-R*, 2018.
[20] M. R. Segal, “UC San Francisco Recent Work Title Machine Learning Benchmarks and Random Forest Regression Machine Learning Benchmarks and Random Forest Regression,” 2004.
[21] M. Zięba, S. K. Tomczak, and J. M. Tomczak, “Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction,” *Expert Syst. Appl.*, vol. 58, no. October 2017, pp. 93–101, 2016.