March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

A Predictive Analysis for Sales Forecasting in Imbalanced Grocery Retail Datasets using Various Machine Learning Algorithms

[1]**Mr.K.Prabhakar,**Asst.Prof,Department of Computer Science and Engineering, Swarnandhra College of Engineering and Technology, Narsapur, W.G Dist, A.P,India.
[2]**Dr.S.Suresh Kumar,**Principal & Prof,Department of Computer Science and Engineering, Swarnandhra College of Engineering and Technology, Narsapur, W.G Dist, A.P,India.
[3]**Dr.P.Prem Delphy**,Prof, Department of Mathematics, Swarnandhra College of Engineering and Technology, Narsapur, W.G Dist, A.P,India.

Article Info
Volume 83
Page Number: 3154 - 3163
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 21 March 2020*

Abstract:
In today's date data analysis is need for every data analytics to examine the sets of data to extract the useful information from it and to draw conclusion according to the information. Data analytics techniques and algorithms are more used by the commercial industries which enables them to take precise business decisions. It is also used by the analysts and the experts to authenticate or negate experimental layouts, assumptions and conclusions. In this paper, prediction is based on groceries data sets by analyzing or exploring the big mart sales data set. Various machine learning and data extraction models are considered for prediction are linear regression, K-means, Apriori algorithm. Before prediction we have to explore and visualize the data because data exploration and visualization is an important stage of predictive modelling. The outcome of this paper is High quality analysis, more flexibility and power as compared to others.

Keywords*: R - Studio, Linear Regression, K- means, Apriori algorithm, Random forest.*

I. INTRODUCTION

As we know that in today's era data analysis is so important to everyone to make better decisions in their field. Analyzing the big data and extracting knowledge full information from the data is little bit tough. So, for mining of complex datasets we need a powerful and effective data mining tool to extract the information and take better decisions in future. We are using R here which is an open source free data mining tool and efficient too. R has several inbuilt packages which provide us efficiency like-ggplot2, VIM etc. R is an open-source data analysis environment and programming language .

The process of converting data into knowledge, insight and understanding is Data analysis, which is a critical part of statistics. For the effective processing and analysis of big data, it allows users to conduct a number of tasks that are essential. R consists of numerous ready-to-use statistical modeling algorithms and machine learning which allow users to create reproducible research and develop data products. Although big data processing may be accomplished with other tools as well, it is when one steps on to the data analysis that R really stands unique, owing to the huge amount of built-in statistical formulae and third-party algorithms available.

To create a powerful and reliable statistical model, data transformation, evaluation of multiple model options, and visualizing the results are essential. This is the reason why the R language has

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

proven so popular: its interactive language uplifts exploration, clarification and presentation. Revolution R Enterprise gives the big-data support and speed to allow the data scientist to repeat through this process quickly.

The dataset here is of Big Mart Sales-first phase of this analysis is to impute the missing values in dataset if any,then comes the data exploration in which we have to explore the dataset and find the relation between various commodities in the big mart sales dataset, data exploration is an important key of predictive model. We cannot create a good predictive model until we know how to explore the dataset from beginning to last.

This phase creates a solid base for data manipulation. Then comes the phase of using various machine learning algorithms to predict the outlet sales (response variable) like k- means, linear regression, multiple regression, random forest, apriori algorithm, etc. The next step is to compare the prediction result of, on basis of that we will get the outlet sales for each outlet present in the city.

II. PROBLEM DESCRIPTION

Sales prediction is a very common real-life problem that each company faces at least once in its lifetime. If done correctly, it can have a significant impact on the success and performance of that company. According to a study, companies with accurate sales predictions are 10% more likely to grow their revenue year-over-year and 7.3% more likely to hit quota.

To create a powerful and reliable statistical model, data transformation, evaluation of multiple model options, and visualizing the results are essential

III. PROPOSED WORK

We have taken the dataset of big mart sales on which we are going to do the operations like exploration, visualization, cleaning, imputation and prediction after importing the dataset in R-Studio

The dataset here is of Big Mart Sales-first phase of this analysis is to impute the missing values in dataset if any, then comes the data exploration in which we have to explore the dataset and find the relation between various commodities in the big mart sales dataset, data exploration is an important key of predictive model. We cannot create a good predictive model until we know how to explore the dataset from beginning to last.

This phase creates a solid base for data manipulation. Then comes the phase of using various machine learning algorithms to predict the outlet sales (response variable) like k- means, linear regression, multiple regression, random forest, apriori algorithm, etc. The next step is to compare the prediction result of, on basis of that we will get the outlet sales for each outlet present in the city

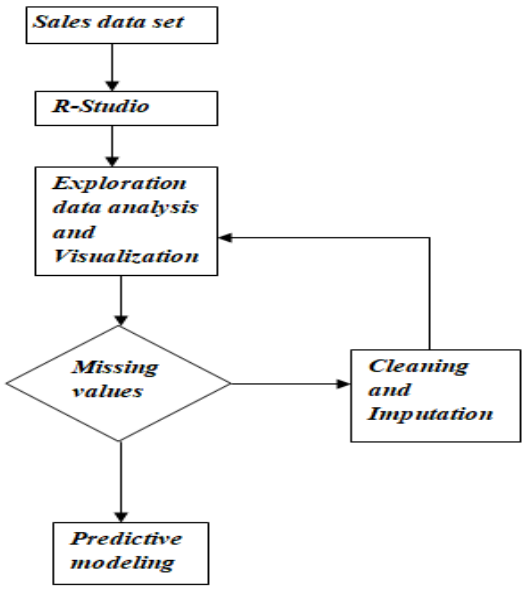

Fig 1: Architecture diagram

IV. EXPERIMENTAL SETUP AND ANALYSIS

A. Loading Packages And Data:-

In R, we take help of multiple packages or libraries to bring in extra functionalities which otherwise are not present in the base R. we will include packages for reading data, manipulation of data, visualization of data, and finally for modeling.

1)Loading packages

```
library(data.table)
library(dplyr)
```

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

```
library(ggplot2)
library(caret)
library(corrplot
library(xgboost)
library(cowplot)
```

2)Reading of Data

We are using 3 CSV files — Train, Test, and Sample Submissions collected from different stores in a city.The *Train file* contains 11 independent variables and 1 target variable, i.e., *Item_Outlet_Sales*. The *Test file* also contains the same set of independent variables, but there is no target variable because that is what we have to predict. The *Sample Submissions file* contains the format in which we have to submit our predictions.

3)Understanding the raw data

Initially we have to understand the raw data for exploring them deeply. There is certain statement for exploring the raw data. They are dim(train);dim(test).

4)Combine Train and Test

To explore data in any data science, it is advisable to append test data to the train data. Combining train and test sets saves a lot of time and effort because if we have to make any modification in the data, we would make the change only in the combined data and not in train and test data separately. Later we can always split the combined data back to train and test.

B. Exploration Data Analysis And Visualization

1)Univariate Analysis

It involves exploring variables individually. We will try to visualize the continuous variables using histograms and categorical variables using bar plots.Since our target variable is continuous, we can visualize it by plotting its histogram.It is a right skewed variable and would need some data transformation to treat its skewness.

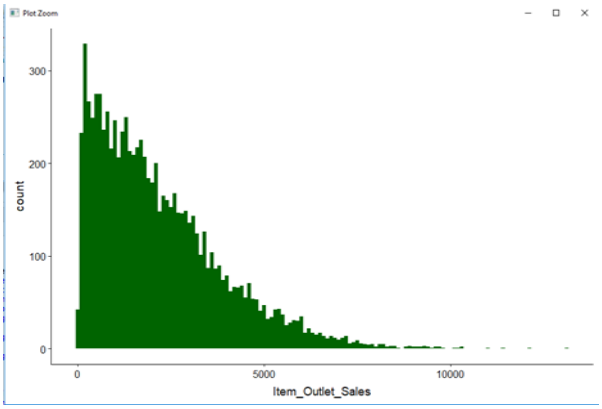

Fig 2: Visualization of continuous variables using histograms

Independent Variables (numeric variables)

Now let's check the numeric independent variables. We again use the histograms for visualizations because that will help us in visualizing the distribution of the variables.

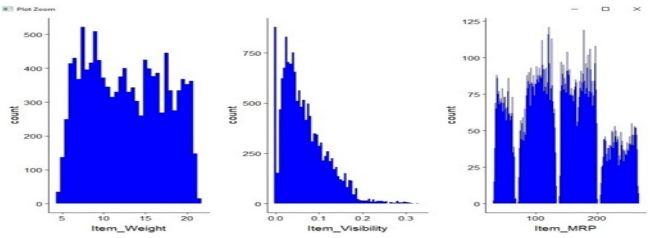

Fig 3 :Visualization of categorical variables using bar plots

Observations

- There seems to be no clear-cut pattern in Item_Weight.
- Item_Visibility is right-skewed and should be transformed to curb its skewness.
- We can clearly see 4 different distributions for Item_MRP. It is an interesting insight.

2) Bivariate Analysis:

- After looking at every feature individually, here we explore the independent variables with respect to the target variable. The objective is to discover hidden relationships between the independent variable and the target variable and use those findings in missing data imputation.

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

- We will make use of **scatter plots** for the continuous or numeric variables and **violin plots** for the categorical variables.

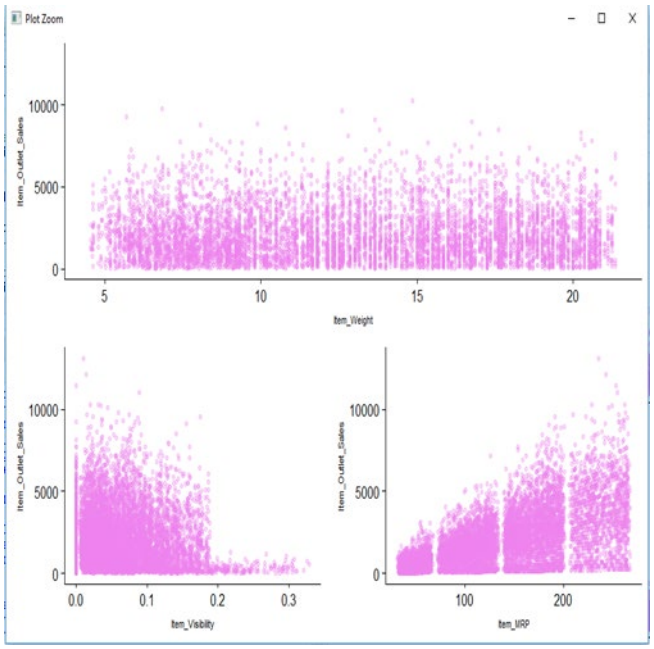

Fig 4: Visualization of scatter plots for the continuous or numeric variables

Observations

- Item_Outlet_Sales is spread well across the entire range of the Item_Weight without any obvious pattern.
- In Item_Visibility vs Item_Outlet_Sales, there is a string of points at Item_Visibility = 0.0 which seems strange as item visibility cannot be completely zero. We will take note of this issue and deal with it in the later stages.

In the third plot of Item_MRP vs Item_Outlet_Sales, we can clearly see 4 segments of prices that can be used in feature engineering to create a new variable**.**

Target Variable vs Independent Categorical Variables

Now we visualise the categorical variables with respect to Item_Outlet_Sales. We try to check the distribution of the target variable across all the categories of each of the categorical variable.

The width of a violin plot at a particular level indicates the concentration or density of data at that level.

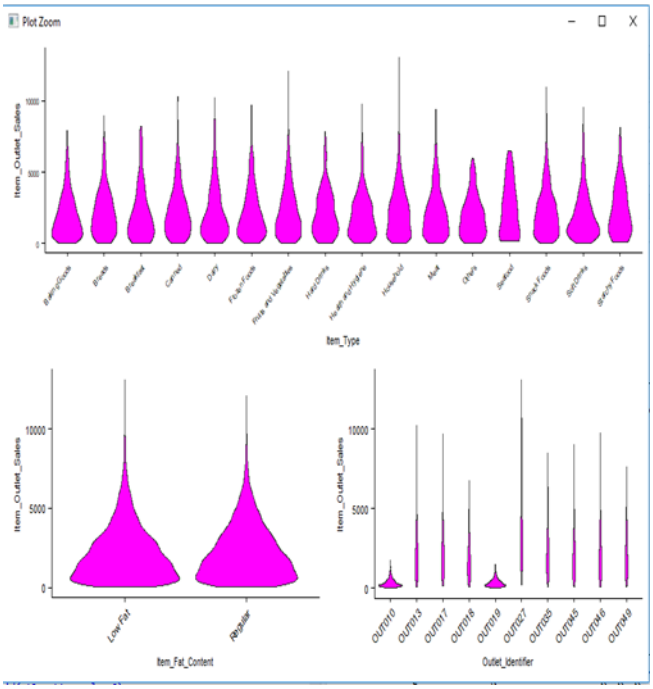

Fig 5 :Visualization of violin plot for the Target Vs Categorical variables

The distribution of 'Small' Outlet_Size is almost identical to the distribution of the blank category (first vioin) of Outlet_Size. So, we can substitute the blanks in Outlet Size with 'Small'.Please note that this is not the only way to impute missing values, but for the time being we will go ahead and impute the missing values with 'Small'.

C. Missing Value Treatment

Missing data can have a severe impact on building predictive models because the missing values might be contain some vital information which could help in making better predictions. So, it becomes imperative to carry out missing data imputation. There are different methods to treat missing values based on the problem and the data. Some of the common techniques are as follows:

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

1. **Deletion of rows**: In train dataset, observations having missing values in any variable are deleted. The downside of this method is the loss of information and drop in prediction power of model.
2. **Mean/Median/Mode Imputation**: In case of continuous variable, missing values can be replaced with mean or median of all known values of that variable. For categorical variables, we can use mode of the given values to replace the missing values.
3. **Building Prediction Model**: We can even make a predictive model to impute missing data in a variable. Here we will treat the variable having missing data as the target variable and the other variables as predictors. We will divide our data into 2 datasets—one without any missing value for that variable and the other with missing values for that variable.

Imputing Missing Value

As you can see above, we have missing values in Item_Weight and Item_Outlet_Sales. Missing data in Item_Outlet_Sales can be ignored since they belong to the test dataset. We'll now impute *Item_Weight* with mean weight based on the *Item_Identifier* variable.

Now let's see if there is still any missing data in *Item_Weight*

```
sum(is.na(combi$Item_Weight))
[1] 0
```

0 missing values! It means we have successfully imputed the missing data in the feature.

Replacing 0's in Item_Visibility variable

Similarly, zeroes in Item_Visibility variable can be replaced with *Item_Identifier* wise mean values of *Item_Visibility*. It can be visualized in the plot below.

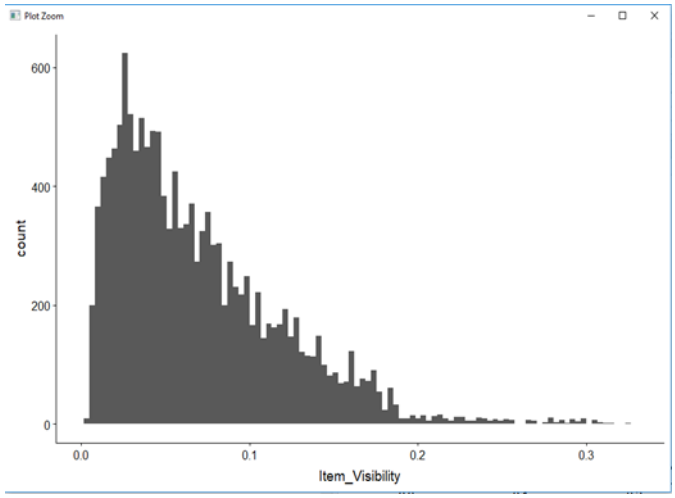

Fig 7: Visualization of Item visibility variable with replacing 0's

D. Feature Engineering

Most of the times, the given features in a dataset are not sufficient to give satisfactory predictions. In such cases, we have to create new features which might help in improving the model's performance. We try to create some new features for our dataset.The following new features:

- **Item_Type_new**: Broader categories for the variable *Item_Type*.
- **Item_category**: Categorical variable derived from *Item_Identifier*.
- **Outlet_Years**: Years of operation for outlets.
- **price_per_unit_wt**: *Item_MRP/Item_Weight*
- **Item_MRP_clusters**: Binned feature for *Item_MRP*.

We can have a look at the *Item_Type* variable and classify the categories into perishable **and** non_perishable as per our understanding and make it into a new feature.

1) Encoding Categorical Variables

Most of the machine learning algorithms produce better result with numerical variables only. So, it is essential to treat the categorical variables present in the data. One thing that can be done is to completely remove the categorical variables, but that would lead to enormous loss of information. Fortunately we

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

have smarter techniques to deal with the categorical variables.

In this stage, we will convert our categorical variables into numerical ones. We will use 2 techniques — Label Encoding and One Hot Encoding.

a)Label encoding simply means converting each category in a variable to a number. It is more suitable for ordinal variables — categorical variables with some order.

b) In One hot encoding, each category of a categorical variable is converted into a new binary column (1/0).

2) Data preprocessing-Correlation plot

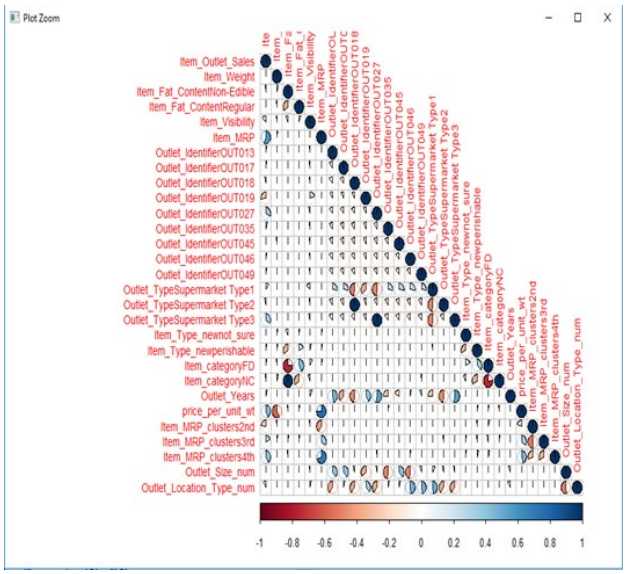

Fig 7: Visualization of Data Processing-Correlation Plot

The correlation plot above shows correlation between all the possible pairs of variables in out data. The correlation between any two variables is represented by a pie. A bluish pie indicates positive correlation and reddish pie indicates negative correlation. The magnitude of the correlation is denoted by the area covered by the pie.

Variables price_per_unit_wt and Item_Weight are highly correlated as the former one was created from the latter. Similarly price_per_unit_wt and Item_MRP are highly correlated for the same reason.

E. Model Building

Finally we have arrived at most interesting stage of the whole process — predictive modeling. We will start with the simpler linear models and then move over to more complex models like Random Forest and XGBoost. We build the following

- Linear Regression
- Lasso Regression
- Ridge Regression
- Random Forest
- XGBoost

Evaluation Metrics for Regression

The process of model building is not complete without evaluation of model's performance. That's why we need an evaluation metric to evaluate our model. Since this is a regression problem, we can evaluate our models using any one of the following evaluation metrics:

- Mean Absolute Error(MAE)
- Mean Squared Error(MSE)
- Root Mean Squared Error (RMSE)

At the competition's page, it has been mentioned that our submission data would be evaluated based on the RMSE score. Hence, we will use RMSE as our evaluation metric.

1. Linear Regression

Linear regression is the simplest and most widely used statistical technique for predictive modeling. Given below is the linear regression equation:

$$Y = \theta_1 X_1 + \theta_2 X_2 + \dots \theta_n X_n$$

where X1, X2,…,Xn are the independent variables, Y is the target variable and all thetas are the coefficients. Magnitude of a coefficient w.r.t to the other coefficients determines the importance of the

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

corresponding independent variable. For a good linear regression model, the data should satisfy a few assumptions. One of these assumptions is that of absence of multi collinearity, i.e., the independent variables should be correlated. However, as per the correlation plot above, we have a few highly correlated independent variables in our data. This issue of multi collinearity can be dealt with regularization.

We have got an RMSE of 1202.33 on the public leaderboard, but this score has been calculated by using only the 25% (public) of the test data. So, there has to be a system in place for us to check generalizability of our model, in other words, how consistently our model performs at unseen data or new data.

2. *Regularized Linear Regression*

Regularized regression models can handle the correlated independent variables well and helps in overcoming over fitting. Ridge penalty shrinks the coefficients of correlated predictors towards each other, while the **Lasso** tends to pick one of a pair of correlated features and discard the other. The tuning parameter **lambda** controls the strength of the penalty.

a)Lasso Regression

```
lasso_linear_reg_mod = train(x = train[, -c("Item_Identifier", "Item_Outlet_Sales")], y = train$Item_Outlet_Sales, method='glmnet', trControl= my_control, tuneGrid = Grid)
```

Mean validation score: 1130.02
Leaderboard score: 1202.26

b)*Ridge Regression*

```
ridge_linear_reg_mod = train(x = train[, -c("Item_Identifier", "Item_Outlet_Sales")],y=train$Item_Outlet_Sales,method='glmnet', trControl= my_control, tuneGrid = Grid)
```

Mean validation score: 1135.08
Leaderboard score: 1219.08

3. *Random Forest*

Random Forest is a tree based bootstrapping algorithm wherein a certain number of weak learners (decision trees) are combined to make a powerful prediction model. For every individual learner, a random sample of rows and a few randomly chosen variables are used to build a decision tree model. Final prediction can be a function of all the predictions made by the individual learners. In case of a regression problem, the final prediction can be mean of all the predictions.

We now build a Random Forest model with 400 trees. The other tuning parameters used here are mtry — no. of predictor variables randomly sampled at each split, and min.node.size — minimum size of terminal nodes (setting this number large causes smaller trees and reduces over fitting).

Our score on the leader board has improved considerably by using Random Forest. Now let's visualize the RMSE scores for different tuning parameters.

a)Best Model Parameters

```
plot(rf_mod)
```

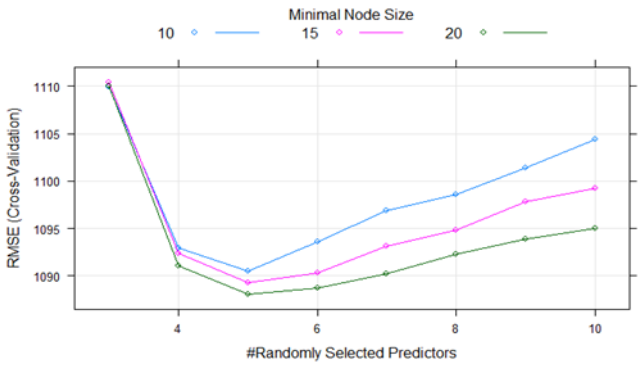

Fig 8: Visualization of RMSE scores for different tuning parameters

As per the plot shown above, the best score is achieved at mtry = 5 and min.node.size = 20

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

b)Variable Importance

Let's plot feature importance based on the RandomForest model

```
plot(varImp(rf_mod))
```

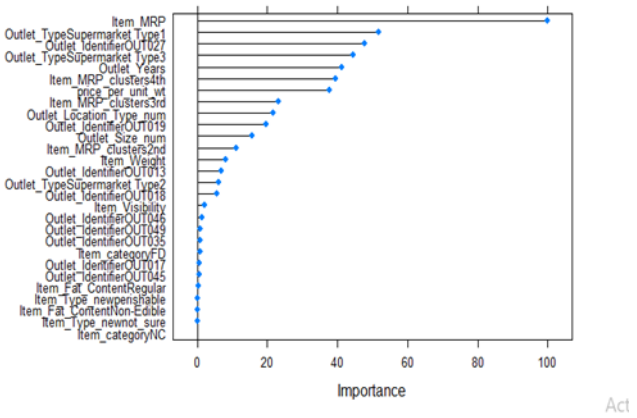

Fig 9: Visualization of feature importance based on the RandomForest model

As expected Item_MRP is the most important variable in predicting the target variable. New features created by us, like price_per_unit_wt, Outlet_Years, Item_MRP_Clusters, are also among the top most important variables. This is why feature engineering plays such a crucial role in predictive modeling.

4. XGBoost

XGBoost is a fast and efficient algorithm and has been used to by the winners of many data science competitions. It's a boosting algorithm. XGBoost works only with numeric variables and we have already replaced the categorical variables with numeric variables. There are many tuning parameters in XGBoost which can be broadly classified into General Parameters, Booster Parameters and Task Parameters.

- **General parameters** refer to which booster we are using to do boosting. The commonly used are tree or linear model
- **Booster parameters** depend on which booster you have chosen
- **Learning Task parameters** that decide on the learning scenario, for example, regression tasks may use different parameters with ranking tasks.

Let's have a look at the parameters that we are going to use in our model.

a) **eta**: It is also known as the learning rate or the shrinkage factor. It actually shrinks the feature weights to make the boosting process more conservative. The range is 0 to 1. Low eta value means the model is more robust to over fitting.
b) **gamma**: The range is 0 to ∞. Larger the gamma more conservative the algorithm is.
c) **max_depth**: We can specify maximum depth of a tree using this parameter.
d) **subsample**: It is the proportion of rows that the model will randomly select to grow trees.
e) **colsample_bytree**: It is the ratio of variables randomly chosen to build each tree in the model

i)Model Training

As per the verbose above, we got the best validation/test score at the 430th iteration. Hence, we will use nrounds = 430 for building the XGBoost model.

```
xgb_model = xgb.train(data = dtrain, params = param_list, nrounds = 430)
```

Leaderboard score: 1154.70

This model has even outperformed the RandomForest model.

ii)Variable Importance

```
var_imp = xgb.importance(feature_names = setdiff(names(train),c("Item_Identifier", "Item_Outlet_Sales")), model = xgb_model)
xgb.plot.importance(var_imp)
```

March-April 2020
ISSN: 0193-4120 Page No. 3154 - 3163

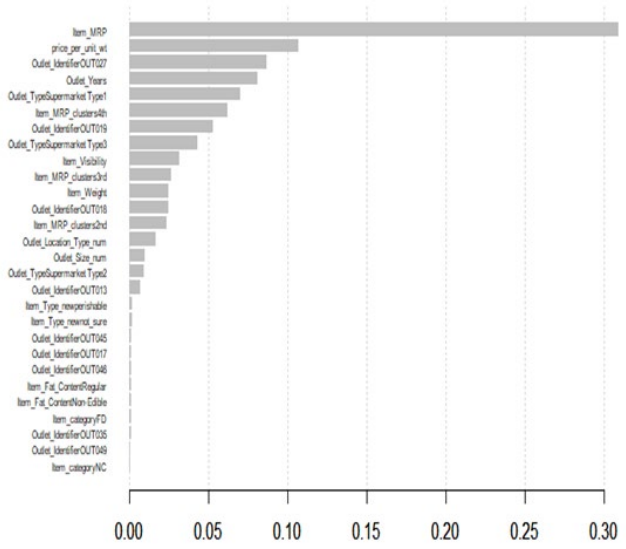

Fig 10: Visualization of Variable importance based on the XGBoost

Again the features created by us, like price_per_unit_wt, Outlet_Years, Item_MRP_Clusters, are among the top most important variables.

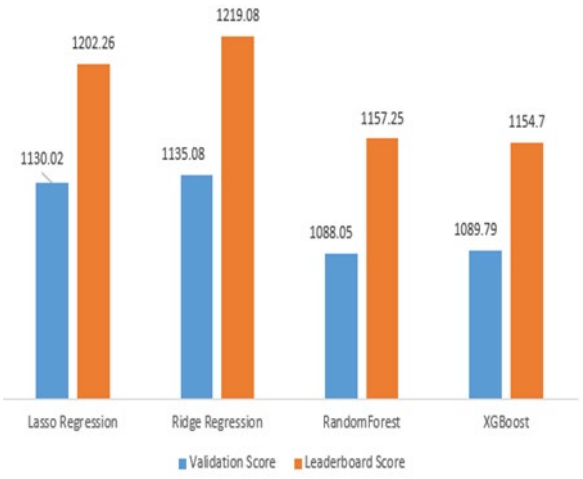

Fig 10: Comparison of Validation and leaderboard Score

V. CONCLUSION

In this paper, we have used four machine learning algorithms-linear regression, regularized linear regression, random forest, XGBoost.We have implemented linear regression for finding out the relation of item outlet sales with other variables like item mrp item visibility etc. for prediction purposes and that we have calculated the root mean square error which is equal to 1202.33for linear regression. To improve this root mean square error(r m s e) and to make decisions we have implemented and after implementation the r m s e reduces to 1219.08. As we can see that still the root mean square error is high, we can implement random forest algorithm. The best score on the public leaderboard has been achieved by XGBoost (1154.70), followed by RandomForest (1157.25). However, there are still a plenty of things that we can try to further improve our predictions.

REFERENCES

1. Pooja K., A.R. K., 'International Journal of Computer Applications', National Seminar on Recent Trends in Data Mining 2016 ,pp.0975 – 8887.
2. Eric T.,Manish G.,Praveen K,∗,1,' 'The Role of Big Data and Predictive Analytics in Retailing', journal of retailing 93,2017,pp.79-95.
3. Hilda K., J¨u., Josef K., 'Using R, WEKA and RapidMiner in Time SeriesAnalysis of Sensor Data for Structural Health Monitoring', IEEE comp. society 2011,pp.1529-4188.
4. Noorollah K., 'Analysis and predicting electricity energy consumption using data mining techniques- A case study I.R. Iran - Mazandaran province', 2015 2nd International Conference on Pattern Recognition and Image Analysis (IPRIA 2015) March 11-12, 2015,pp-4799-8445.
5. SanchitaP.,'Big Data Analytics Using R',International Research Journal of Engineering and Technology (IRJET)e-ISSN: 2395 -0056 Volume: 03 Issue: 07 | July-2016 ,pp.78-81.
6. Xindong W., Vipin K., J. Ross Q., Joydeep G., Qiang Y., Hiroshi M.,Geoffrey J.,McLachlan ,Angus Ng ,Bin g L.,Philip S. Y., Zhi-Hua Z., Michael S., David J. Hand • Dan Steinbe r., 'Top 10 algorithms in data mining', Knowl Inf

Syst (2008) 14: pp.1–37
7. Tanu Jain* Dr. A.K Dua Varun Sharma, 'Quantitative Analysis of Apriori and Eclat Algorithm for Association Rule Mining' International Journal Of Engineering And Computer Science ISSN: 2319-7242 Volume 4 Issue 10 Oct 2015, Page No. 14649-14652.
8. Iikhita a., Abhay b., 'Application of data mining classification techniques on soil data using r' International Journal of Advances in Electronics and Computer Science, ISSN:pp. 2393-2835 Volume-4, Issue-1, Jan.-2017.
9. Priyanka P., Kavita S., Oza Ass., K. Kamat., 'Outlet sales Analysis:Using R Analytical Tool',International conference on I-SMAC 2017,pp.839-842
10. Sitharthan, R., M. Geethanjali, and T. Karpaga Senthil Pandy. "Adaptive protection scheme for smart microgrid with electronically coupled distributed generations." Alexandria Engineering Journal 55, no. 3 (2016): 2539-2550.