Trim Your Sales!

Abhinandan Desai, Sarthak Gupte

6/9/2020

Problem Description

The project title is a word play on the idiom 'Trim your sails'. The idiom means to make changes to suit one's new circumstances. In business, it generally means to change the management strategy, reduce costs due to changes and unforeseen circumstances. This is a major problem in retail stores and this project aims to create a solution to predict future sales for companies so that it can enable a better strategic plan beforehand, reduce risks and handle emergency situations.

Introduction

The process of estimating future sales is called sales forecasting. Accurate sales forecasts can enable companies to make strategic business decisions and predict the performance of sales in the near future. It also gives an insight to work out the workforce, costs, and resources of a company. Past sales data, economic, seasonal and location trends are some of the major factors that help to generate accurate forecasts. Predictive sales also help companies and businesses to plan their future growth and expansion.

This project is going to focus on creating a system that will conduct regression analysis on sales data of Black Friday and try to predict accurate future sales. The dataset contains different features which will be used to create a more detailed analysis.

The project will firstly collect data, then it will focus on cleaning and preparing the data. The data will be then explored and visualized for better understanding. The project will use different regression techniques to generate machine learning models in order to carry out prediction of sales. One of the following algorithms will be worked upon for improvement to generate more accurate predictions. A comparitive analysis will be carried out between the algorithms. The algorithms that are being considered are as follows:

- Ridge Regression - It helps to mitigate the problem of multicollinearity which is common in models with many parameters. It improves parameter estimation within considerable bias.
- Lasso Regression - It is the abbreviation of **L**east **A**bsolute **S**hrinkage and **S**election **O**perator. It performs both variable selection and regularization in order to enhance the prediction accuracy.
- Elastic Net Regression - Elastic net is a combination of ridge and lasso regression. It has two tuning parameters (alpha and lambda) while lasso and ridge regression only has 1.
- Random Forest Regression - It operates by constructing a number of decision trees parallely at training time and outputting the mean prediction of the individual trees.
- XGBoost (e**X**treme **G**radient **B**oosting) Regression - It is an advanced implementation of gradient boosting algorithm. It can handle sparse data.

Such a system is a necessity nowadays in any kind of business. Large amounts of data is generated by retail stores and Walmart is one such example. The data would go to waste if not used properly and this project will try to make the most of such data. It will help find general and hidden sale trends which will further lead to better strategic plans.

Data Summary and Visualization

The dataset contains 12 different columns, each attribute representing a different characteristic of the dataset:

-User_ID: Unique identifier of the shopper.

-Product_ID: Unique identifier for the purchased product.

-Gender: Identification of the shopper (Male/Female).

-Age: Shopper's age that has been divided into bins or groups.

-Occupation: Occupation of the shopper. It is categorical as each number must be linked to a type of occupation

-City_Category: City where the shopper lives.

-Stay_In_Current_City_Years: Number of years that the shopper has stayed in current city.

-Marital_Status: Whether the shopper is married or not. value '0' corresponding to unmarried and '1' corresponding to married.

-Product_Category_1: Product category of the purchase.

-Product_Category_2: Product category of the purchase.

-Product_Category_3: Product category of the purchase.

-Purchase: Value of the product in American dollars.

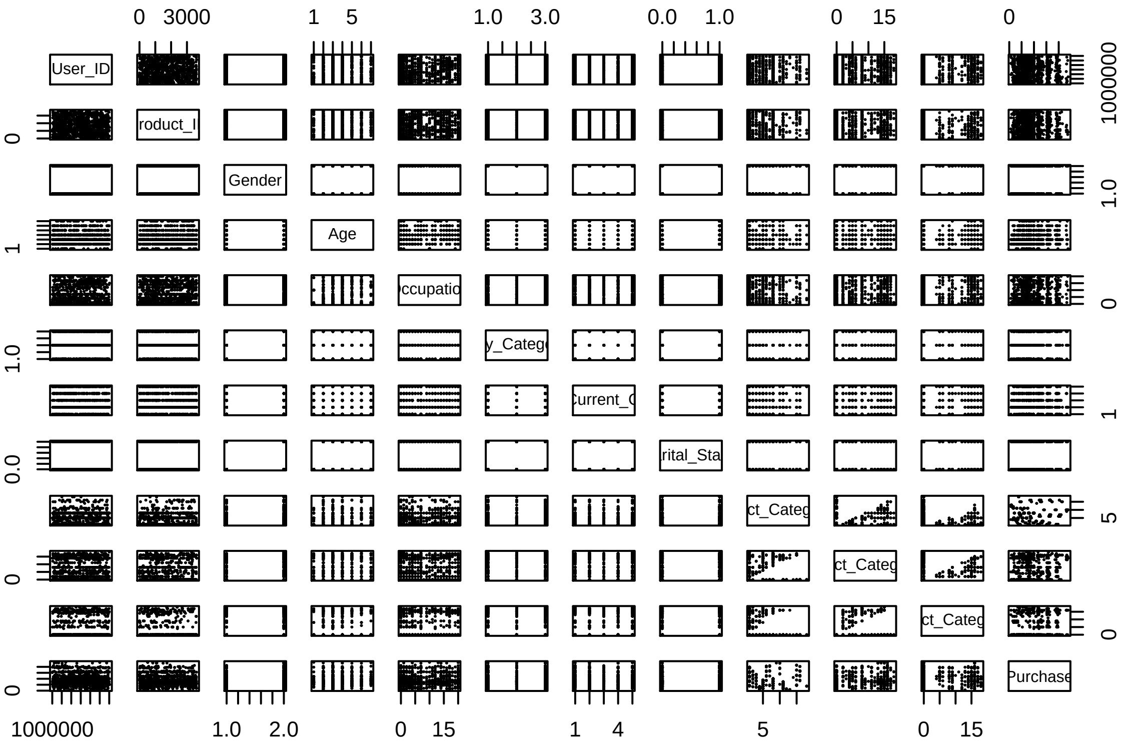
The structure of the dataset is as follows and is then followed by an example of the actual dataset:

```
## 'data.frame': 537577 obs. of 12 variables:
## $ User_ID : int 1000001 1000001 1000001 1000001 1000002 1000003 1000004
1000004 1000004 1000005 ...
## $ Product_ID : Factor w/ 3623 levels "P00000142","P00000242",..: 671 2375
851 827 2733 1830 1744 3319 3597 2630 ...
## $ Gender : Factor w/ 2 levels "F","M": 1 1 1 1 2 2 2 2 2 2 ...
## $ Age : Factor w/ 7 levels "0-17","18-25",..: 1 1 1 1 7 3 5 5 5 3 ...
## $ Occupation : int 10 10 10 10 16 15 7 7 7 20 ...
## $ City_Category : Factor w/ 3 levels "A","B","C": 1 1 1 1 3 1 2 2 2 1 ...
## $ Stay_In_Current_City_Years: Factor w/ 5 levels "0","1","2","3",..: 3 3 3 3
5 4 3 3 3 2 ...
## $ Marital_Status : int 0 0 0 0 0 0 1 1 1 1 ...
## $ Product_Category_1 : int 3 1 12 12 8 1 1 1 1 8 ...
## $ Product_Category_2 : num 0 6 0 14 0 2 8 15 16 0 ...
## $ Product_Category_3 : num 0 14 0 0 0 0 17 0 0 0 ...
## $ Purchase : int 8370 15200 1422 1057 7969 15227 19215 15854 15686 7871 ...
```

```
##   User_ID Product_ID Gender   Age Occupation City_Category
## 1 1000001  P00069042      F  0-17         10             A
## 2 1000001  P00248942      F  0-17         10             A
## 3 1000001  P00087842      F  0-17         10             A
## 4 1000001  P00085442      F  0-17         10             A
## 5 1000002  P00285442      M   55+         16             C
## 6 1000003  P00193542      M 26-35         15             A
##   Stay_In_Current_City_Years Marital_Status Product_Category_1
## 1                          2              0                  3
## 2                          2              0                  1
## 3                          2              0                 12
## 4                          2              0                 12
## 5                         4+              0                  8
## 6                          3              0                  1
##   Product_Category_2 Product_Category_3 Purchase
## 1                  0                  0     8370
## 2                  6                 14    15200
## 3                  0                  0     1422
## 4                 14                  0     1057
## 5                  0                  0     7969
## 6                  2                  0    15227
```

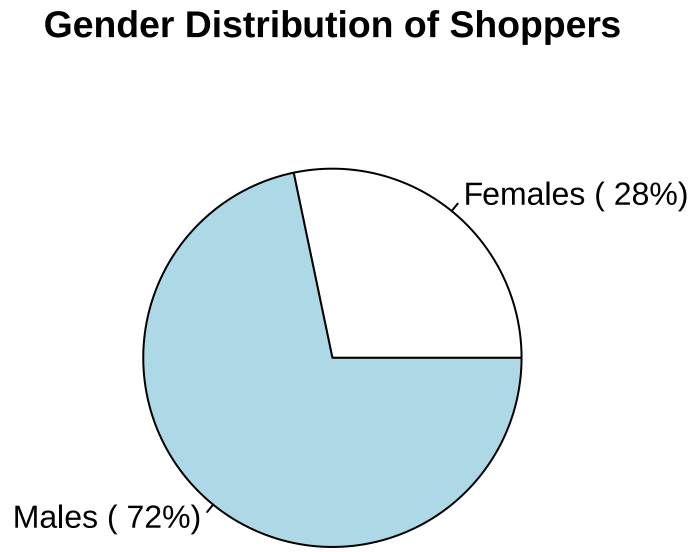
Each row of the dataset represents a different transaction or item purchased by a specific customer. The Product_ID identifies the product that has been purchased and the User_ID identifies the customer that purchased it.

In order to predict sales, the target attribute is ‘Purchase’ in the dataset. A correlation plot would help to identify attributes that have any relation or affect that attribute.



Different attributes are analyzed in order to understand the distribution of sales in the dataset. Firstly, the ‘Gender’ attribute is analyzed in order to understand the distribution of sales among males and females

```
##      F     M
## 1666 4225
```



It is visible that majority of the shoppers are Males. This is a good insight which could help understand the stores what kind of products have better chances to be sold but in order to understand it better, the average spending by each gender is analyzed to get a better insight of amount spend in sales.

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 6 x 2
##   User_ID Total_Purchase
##     <int>          <int>
## 1 1000001         333481
## 2 1000002         810353
## 3 1000003         341635
## 4 1000004         205987
## 5 1000005         821001
## 6 1000006         379450
```

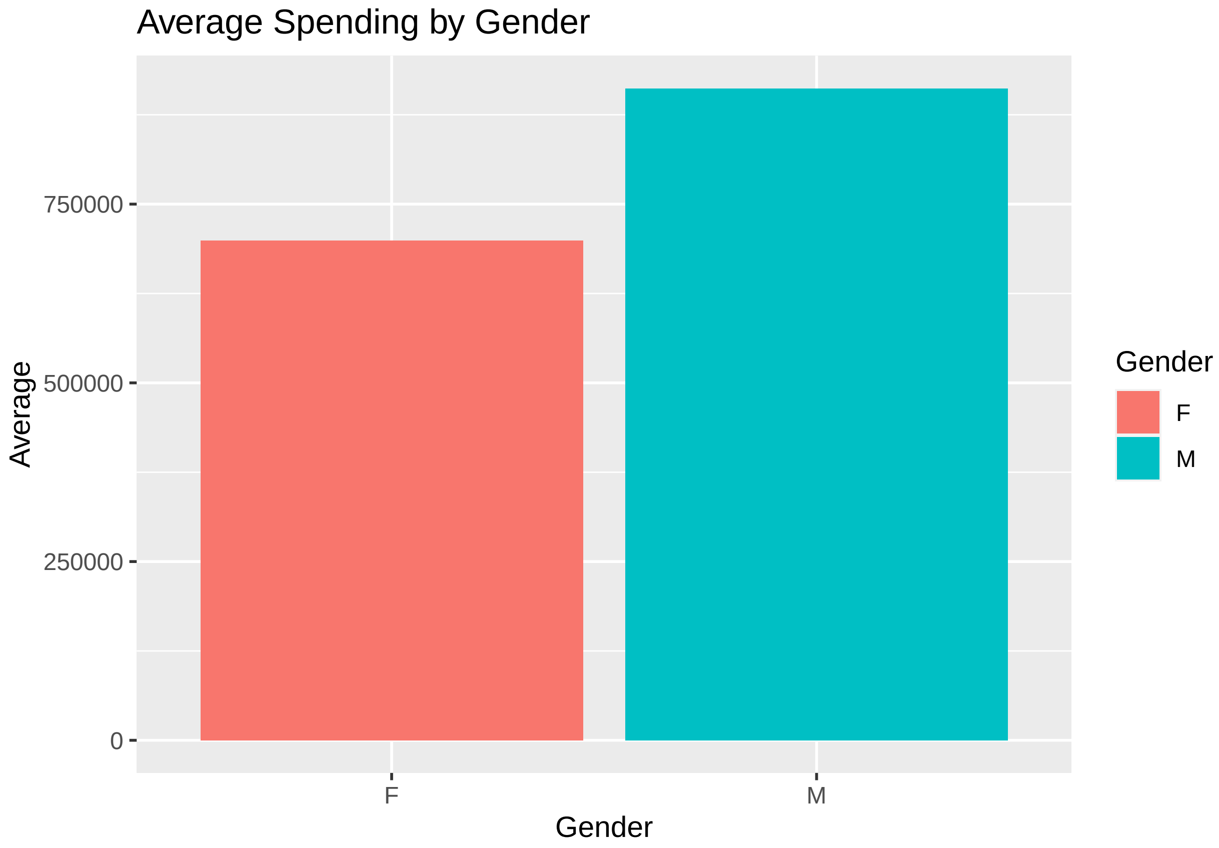
```
## # A tibble: 6 x 2
## # Groups:   User_ID [6]
##   User_ID Gender
##     <int> <fct>
## 1 1000001 F
## 2 1000002 M
## 3 1000003 M
## 4 1000004 M
## 5 1000005 M
## 6 1000006 F
```

The total amount of dollars each shopper has spent is now known and there is also a dataset that knows the gender of each shopper. Merging these datasets into one and then calculating the average spendings that each gender group has done on Black Friday is the next step.

```
## # A tibble: 6 x 3
##   User_ID Total_Purchase Gender
##     <int>          <int> <fct>
## 1 1000001         333481 F
## 2 1000002         810353 M
## 3 1000003         341635 M
## 4 1000004         205987 M
## 5 1000005         821001 M
## 6 1000006         379450 F
```

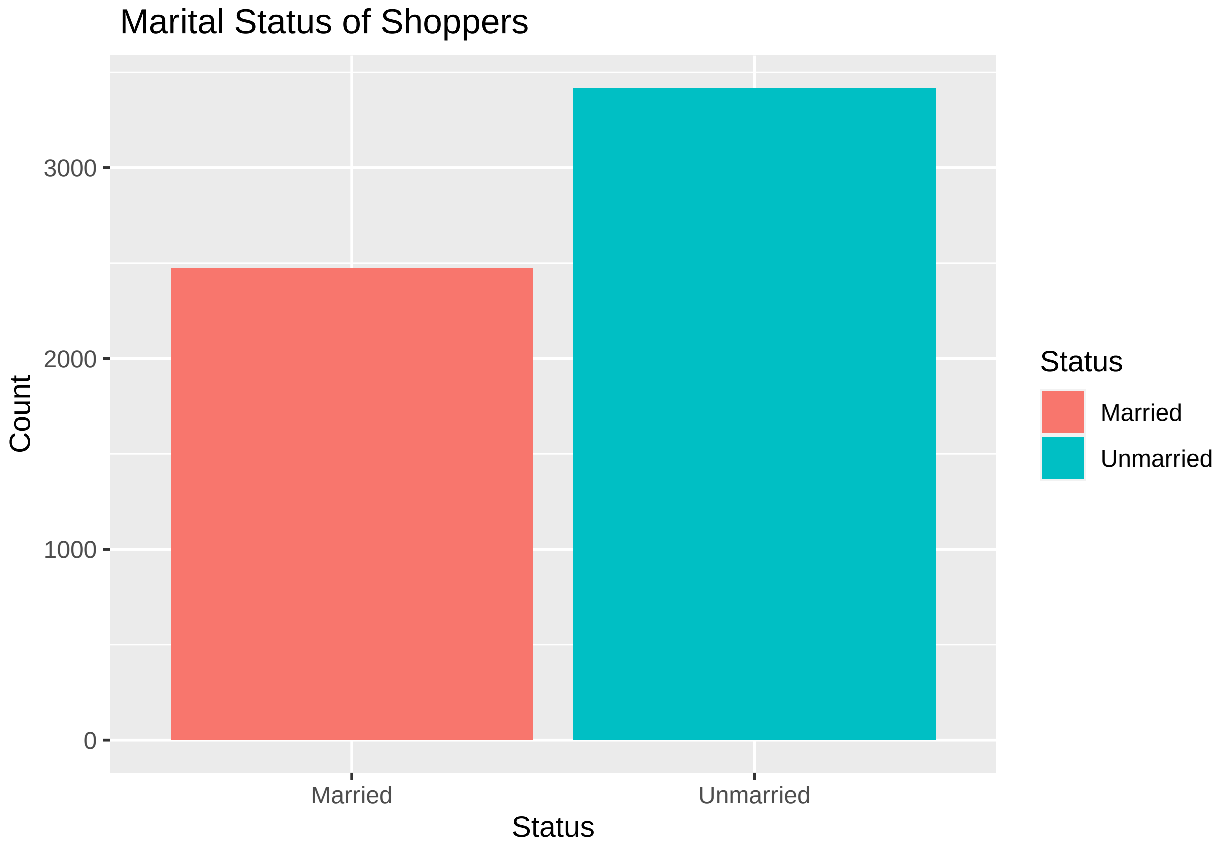
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 4
##   Gender   Purchase Count Average
##   <fct>       <dbl> <int>   <dbl>
## 1 F      1164624021  1666 699054.
## 2 M      3853044357  4225 911963.
```



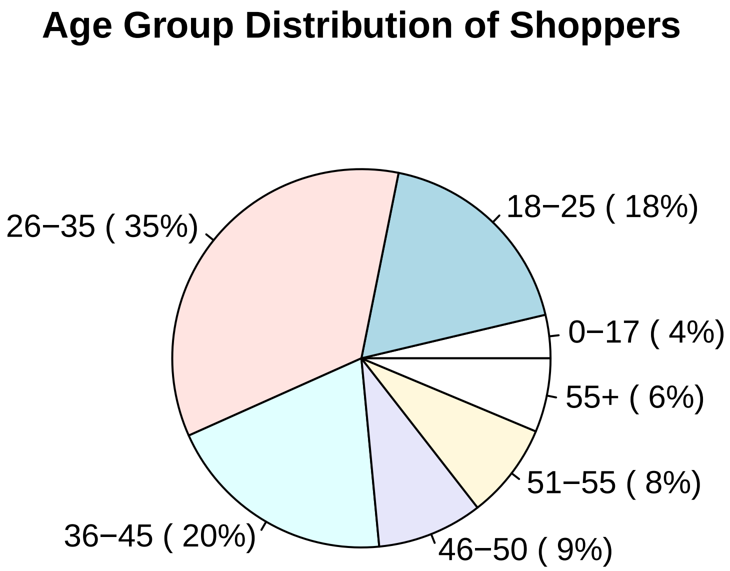
The average spending by males on Black Friday is considerably higher than the females. Further, checking the distribution of the buyers or shoppers on the basis of the marital status can give insights on the kind of products that might be more popular and whether or not being married has a major effect on spending.

```
## # A tibble: 6 x 2
## # Groups:   User_ID [6]
##   User_ID Marital_Status
##     <int>          <int>
## 1 1000001              0
## 2 1000002              0
## 3 1000003              0
## 4 1000004              1
## 5 1000005              1
## 6 1000006              0
```



Most shoppers seem to be unmarried. Similarly, visualizing the distribution of 'Age' atrribute in the dataset would help to find out which age group is dominant as shoppers.

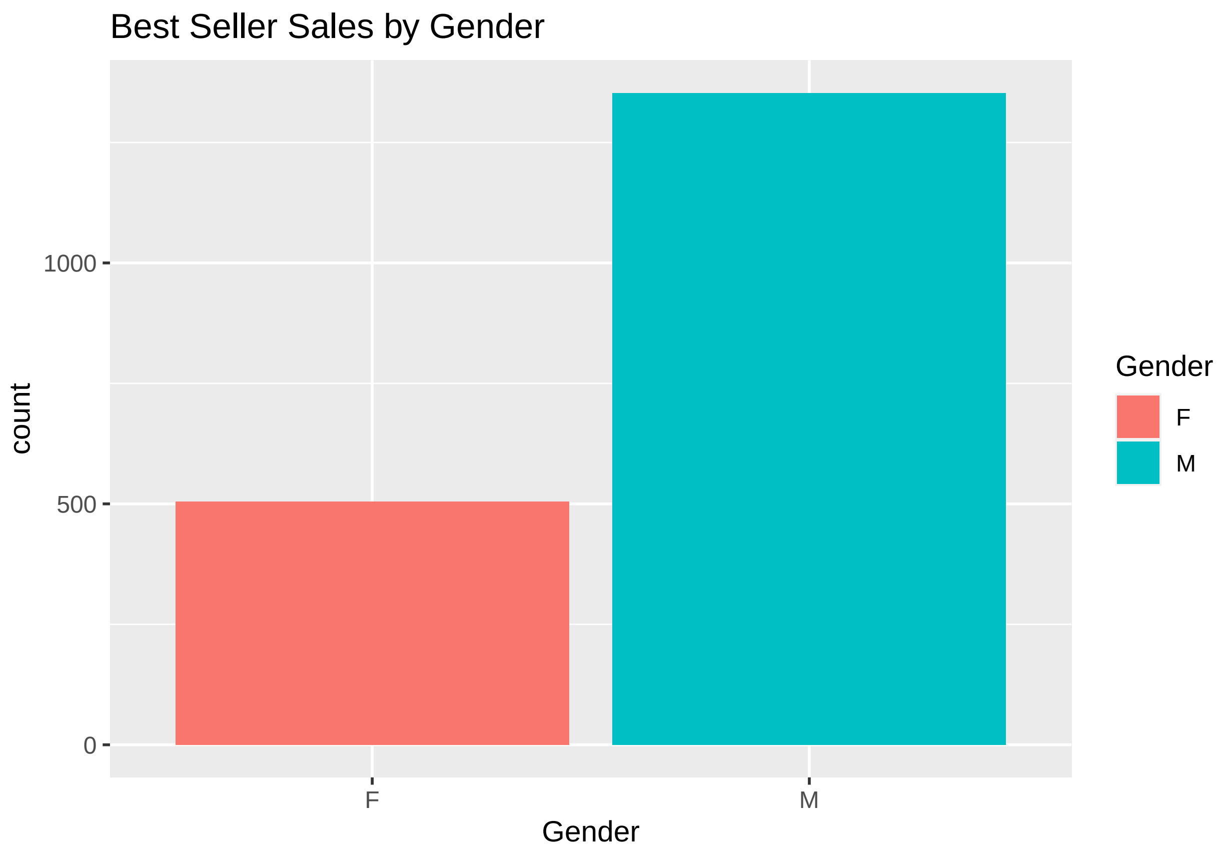
```
##     Age    n
## 1  0-17  218
## 2 18-25 1069
## 3 26-35 2053
## 4 36-45 1167
## 5 46-50  531
## 6 51-55  481
## 7   55+  372
```



From the pie chart it can be seen that the Age group of 26-35 have a majority in number of sales happening on black friday. After this, to increase purchases and sales and to understand the trends, finding out the top products that sell become helpful to understand the trends.

```
##   Product_ID    n
## 1  P00265242 1858
## 2  P00110742 1591
## 3  P00025442 1586
## 4  P00112142 1539
## 5  P00057642 1430
```

P00265242 is the top seller in the store and it is clearly sold more than any other products. Analyzing the major buyers of this product will help to understand the type of product it is.



The best selling product is mostly bought by males. These visualizations and analysis of the dataset helped to understand about the products, the trends in sales, and also the type of customers in the store. It can be deduced that most spending customers are unmarried males and are in the age group of 26-35. The average spending by males is comparatively higher than females. The best selling product is mostly bought by male customers.

Data Preprocessing

It is necessary to clean and prepare data so that it is best suited for further analysis. In order to achieve that, columns 'User_ID' and 'Product_ID' that don't add information for analysis are dropped.

```
##   Gender   Age Occupation City_Category Stay_In_Current_City_Years
## 1      F  0-17         10             A                          2
## 2      F  0-17         10             A                          2
## 3      F  0-17         10             A                          2
## 4      F  0-17         10             A                          2
## 5      M   55+         16             C                         4+
## 6      M 26-35         15             A                          3
##   Marital_Status Product_Category_1 Product_Category_2 Product_Category_3
## 1              0                  3                  0                  0
## 2              0                  1                  6                 14
## 3              0                 12                  0                  0
## 4              0                 12                 14                  0
## 5              0                  8                  0                  0
## 6              0                  1                  2                  0
##   Purchase
## 1     8370
## 2    15200
## 3     1422
## 4     1057
## 5     7969
## 6    15227
```

After this we divide the data into train and test data. 70% data as train data and 30% data as test data. The 'Purchase' attribute is the target variable.

```
## [1] 376303     10
```

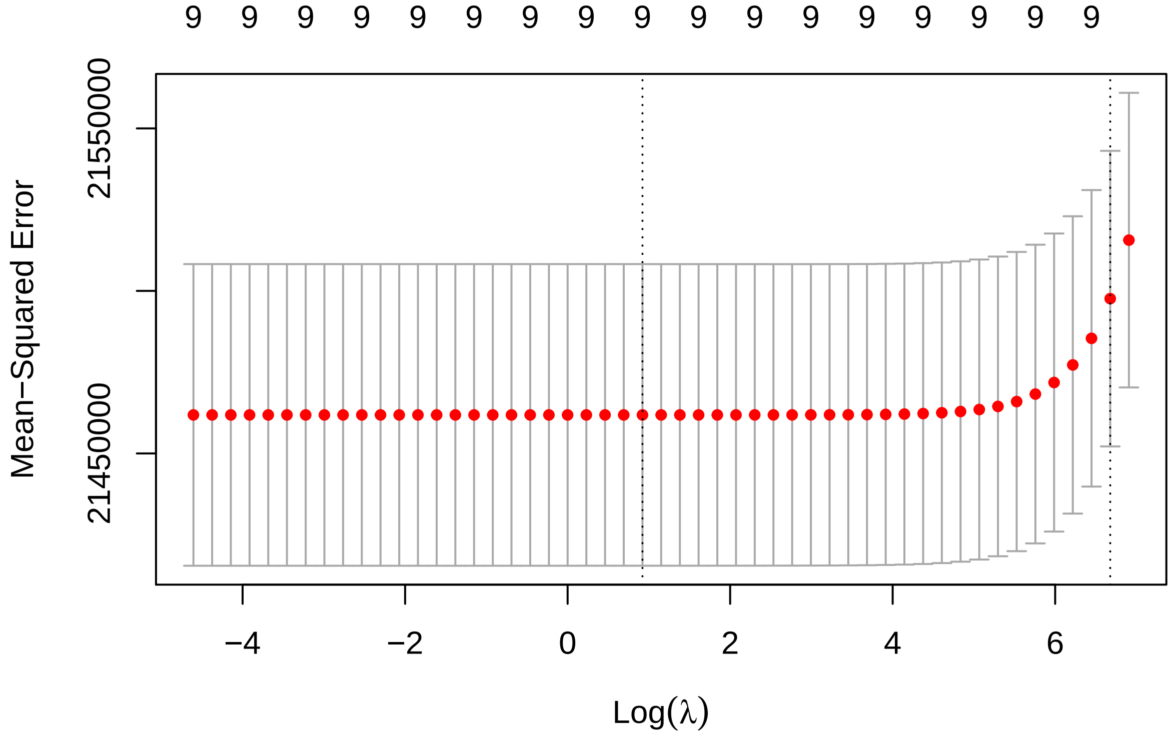
```
## [1] 161274     10
```

Once the required data is prepared, regression analysis can be carried out in order to predict the variables that affect sales the most.

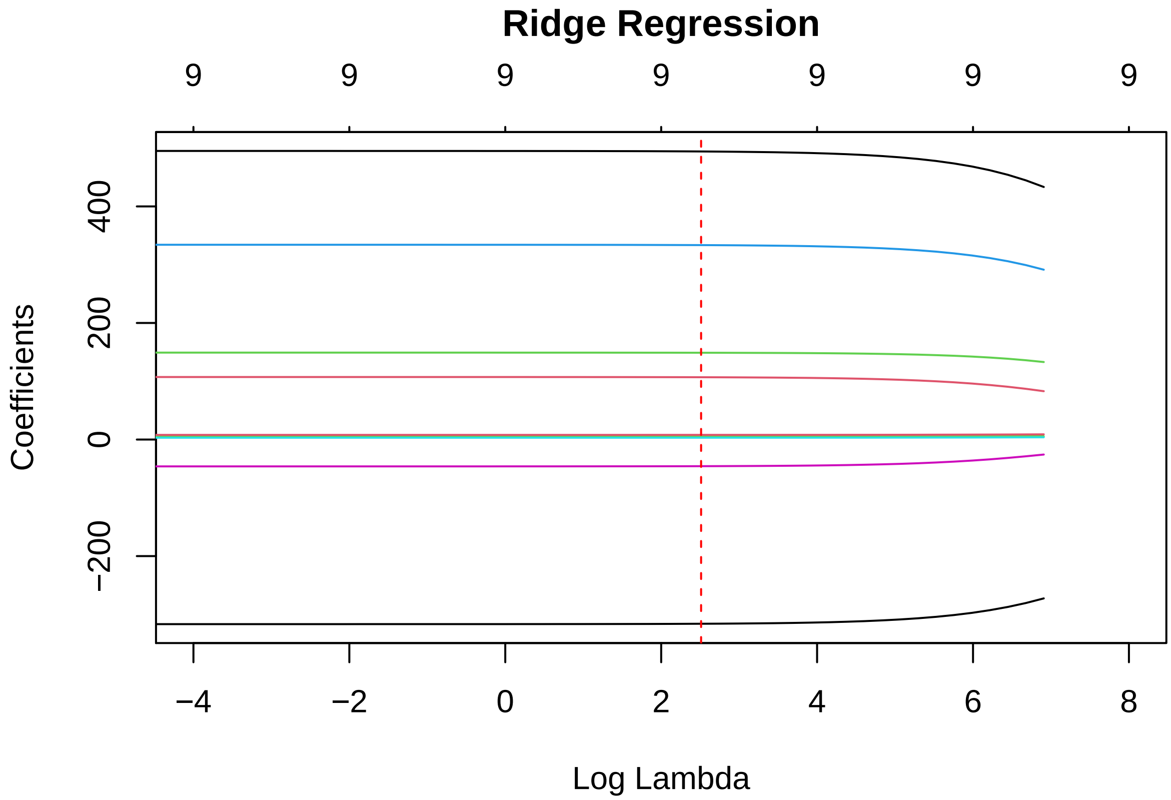
Ridge Regression

Regression is an extension of a linear regression where the loss function is modified. This minimize the complexity using alpha. Alpha value is set to zero which represents Ridge Regression. Firstly, ridge regression involves tuning a hyperparameter, lambda. An argument for a range of lambdas is created. Cross validation is carried out to find the best lambda value.

```
##           Length Class     Mode   
## a0         51    -none-    numeric
## beta      459    dgCMatrix S4     
## df         51    -none-    numeric
## dim         2    -none-    numeric
## lambda     51    -none-    numeric
## dev.ratio  51    -none-    numeric
## nulldev     1    -none-    numeric
## npasses     1    -none-    numeric
## jerr        1    -none-    numeric
## offset      1    -none-    logical
## call        6    -none-    call   
## nobs        1    -none-    numeric
```



```
## [1] 0.1353453

## [1] 4632.542

## Rsquared Value for Test data =  0.1352737

## RMSE Value for Test data =  3030.987
```

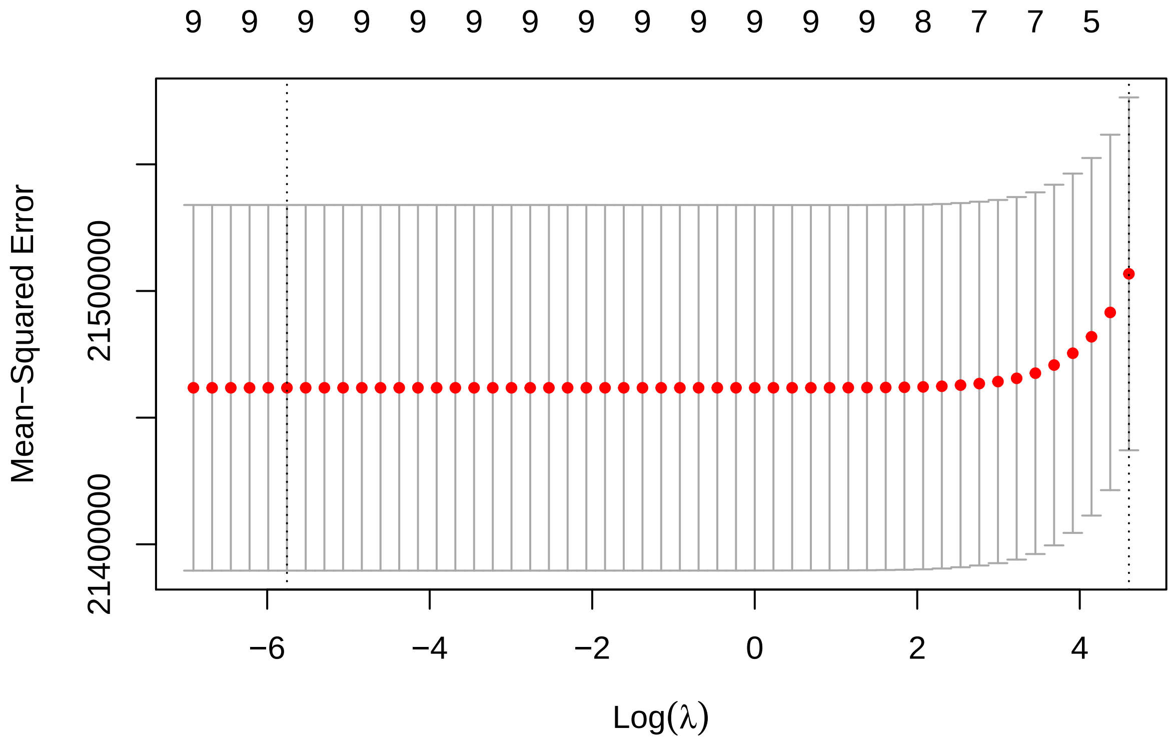


Two lambda values are selected for the model. The Coefficient VS Log Lambda contains a red dotted line which represents the lambda minimum that results in the smallest cross-validation error. This value is generally used as the penalization level for predicting outcomes in the dataset. 'lambda$1se' is the lambda at one standard error and is represented by a blue dotted line. It is used for higher penalization level for predicting outcomes in the dataset.

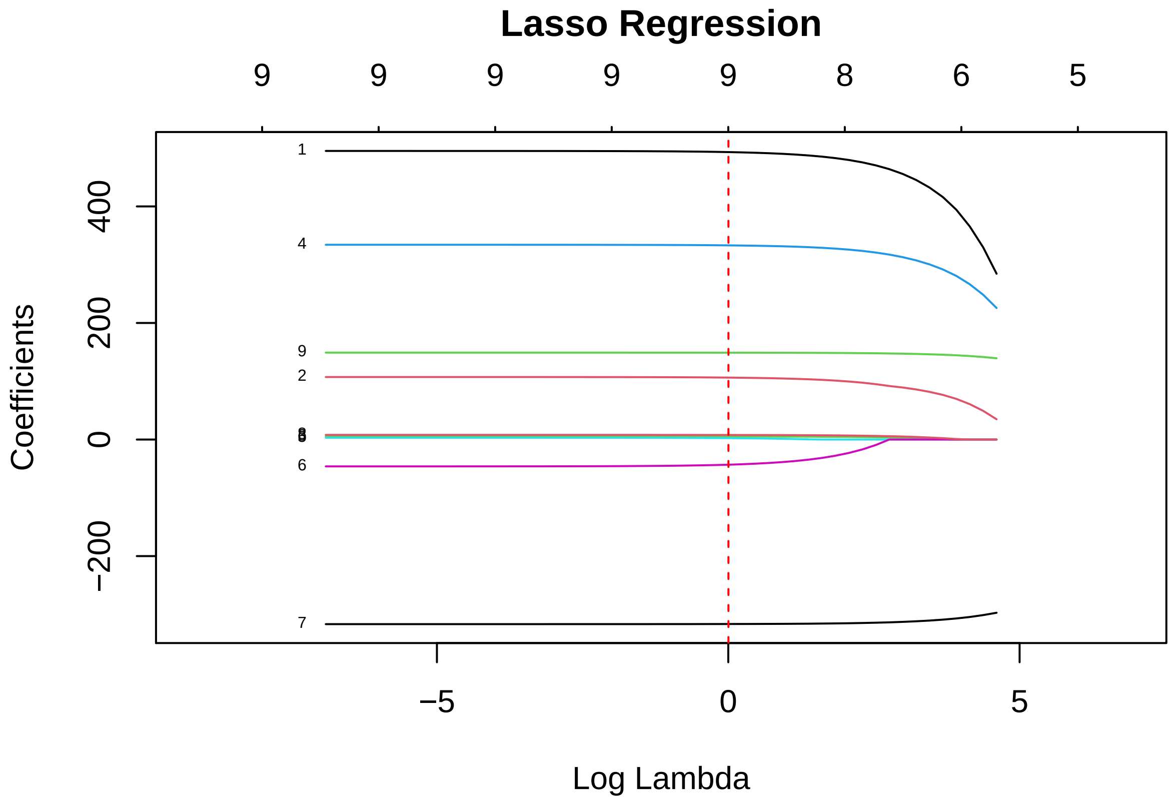
The Log Lambda value of 'lambda$1se' is around 100, hence cannot be visualized in the graph. In the graph plotted, neither of the attributes show a lot of variation. At Lambda minimum, we can say that attribute 1,4,7 and 9 could be good predictors.

Lasso Regression

After analyzing results of Ridge regression, data is analyzed using LASSO Regression. Lasso is also modified version of linear regression. Here, alpha value becomes 1 which represents Lasso regression. Firstly, the optimum lambda is found.



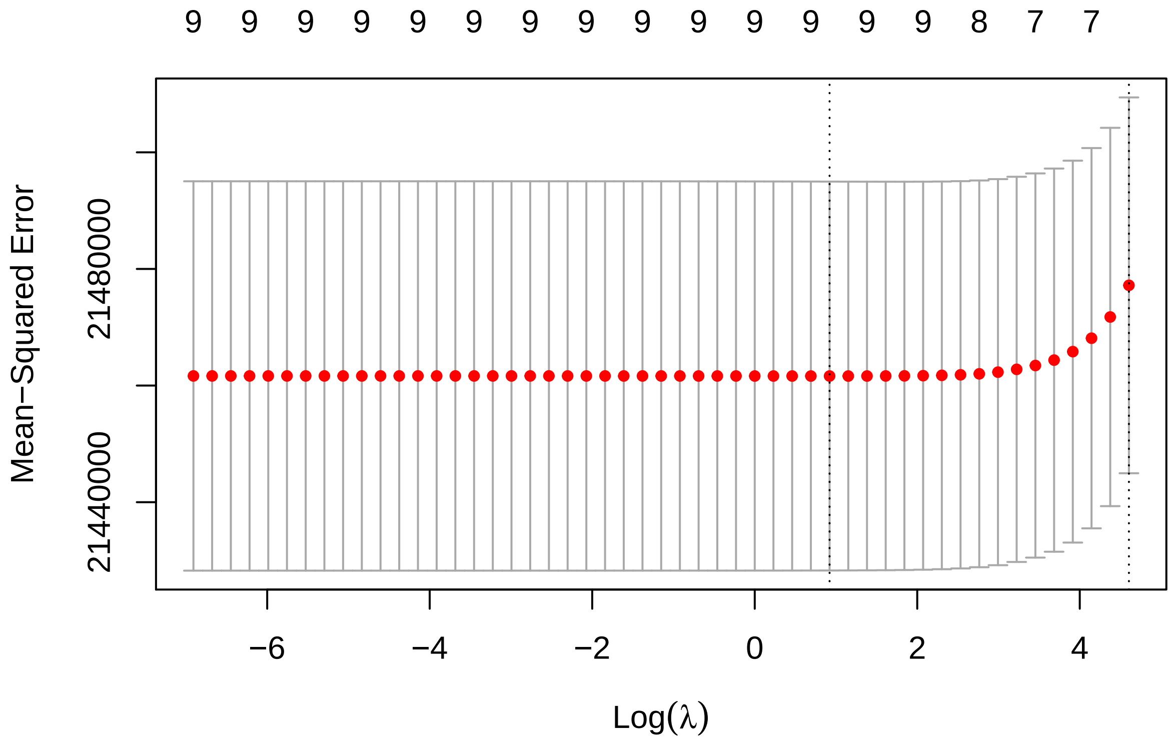
```
## [1] 0.1353453

## [1] 4632.542

## Rsquared Value for Test data =  0.1352737

## RMSE Value for Test data =  3030.987
```



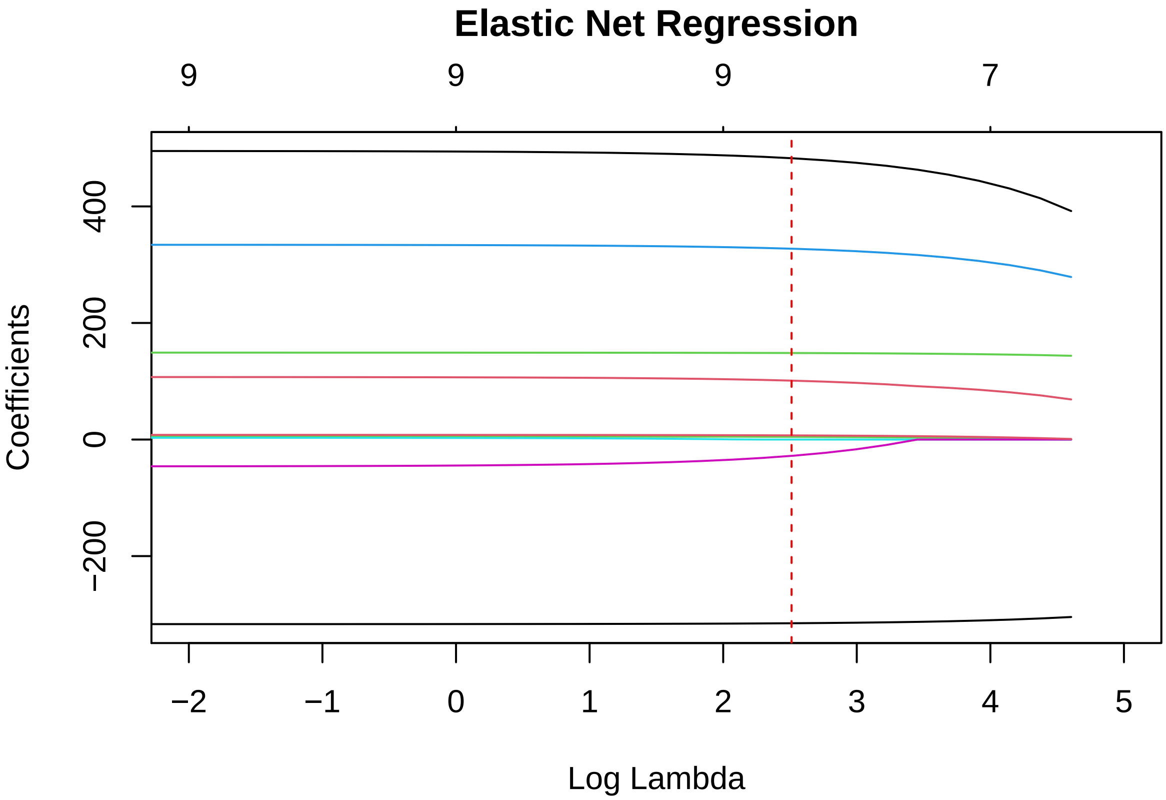
In the Coefficient VS Log Lambda plot, it can be observed that variation in the predictors is more. We can observe that just after lambda minimum the variation of the attributes 1, 4 and 7 is high. Lasso Regression uses a different penalization approach and so some coefficients become exactly zero. The attrbutes 1,4 and 7 are visibly good predictors for sales. 6 has a small coefficient but it also has a good variation.

Elastic Net Regression (Core Algorithm)

Elastic Net Regression has alpha between 0 and 1. It uses a combination of both penalization approaches from Lasso and Ridge regression techniques.



```
## [1] 0.1353447

## [1] 4632.544

## Rsquared Value for Test data =  0.1352712

## RMSE Value for Test data =  3030.991
```



Observing the Coefficent VS Log Lambda plot, we can see that at lambda minimum there is great variation in the coefficient values. It can be visualized that 1,4 and 7 are great predictors for the target attribute. we can see that attribute 6 can be removed as it almost drops to zero near lambda minimum. Elastic net regression has a scope of further improvement as there are two parameters - alpha and lambda. This algorithm can be further fine tuned and is a good candidate for core algorithm.

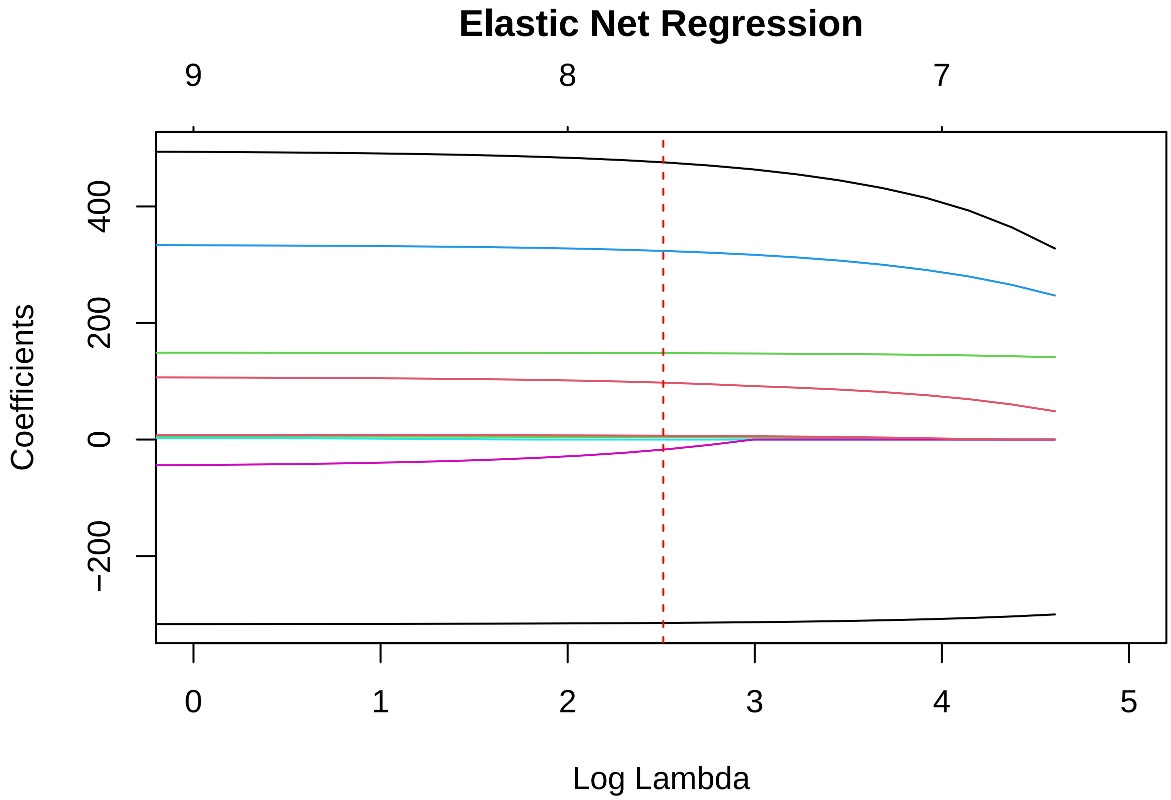
Alpha can be fine tuned and an optimum value can be found. This fine tuning would help improve the performance Elastic Net Regression. In order to fine tune alpha, the best or optimum alpha value should be calculated. Using cross validation, the optimum alpha can be figured out through using different values of alpha between 0.1 to 0.9.

```
## Warning: executing %dopar% sequentially: no parallel backend registered
```

After calculating the optimum alpha, we can use it to make a new fit with the optimized alpha value.This method is an upgrade to the core algorithm which is elastic net regression.

```
## 0.8
```

```
## 2.511886
```



```
## [1] 0.1353439
```

```
## [1] 4632.546
```

```
## Rsquared Value for Test data =  0.1352692
```

```
## RMSE Value for Test data =  3030.995
```

eXtreme Gradient Boosting (XGB) Regression

The eXtreme Gradient Boosting Regression is out of the course algortihm used in this project. Extreme Gradient Boosting (xgboost) is similar to gradient boosting framework but more efficient. It has both linear model solver and tree learning algorithms. So, what makes it fast is its capacity to do parallel computation on a single machine. In comparison to random forest regression the xgboost is more better for the same reason that xgboost perform linear model solver and the tree learning algorithms. The function intakes the training dataset and the labels.

```
## [1] 0.7985143
```

```
## [1] 2236.249
```

```
## Rsquared Value for Test data =  0.6088891
```

```
## RMSE Value for Test data =  2038.425
```

Random Forest Regression

Random Forest Regression uses ensemble classification for regression. It creates multiple decision tree and same time train them. The ouput of each tree is collected and then mode is chosen as the output of the Regression. Sometimes it is difficult to interpret the output of the random forest but it generally boosts the performance of the model.

```
## [1] 0.3951087
```

```
## [1] 3874.688
```

```
## Rsquared Value for Test data =  0.3947593
```

```
## RMSE Value for Test data =  2535.764
```

Comparitive Analysis and Conclusion

For comparing the all the 5 models used we chose RMSE value. RMSE help us to compare model well as it is used to calculate the standard deviation of the errors and how far they are from the regression line for the model. The performaces of all models for test data are summarized below -

```
## 1. RMSE value for Ridge Regression is  3030.987
```

```
## 2. RMSE value for Lasso Regression is  3030.987
```

```
## 3. RMSE value for Elastic Net Regression before tuning is  3874.688
```

```
## 4. RMSE value for Elastic Net Regression after tuning is  3030.995
```

```
## 5. RMSE value for XGBoost Regression is  2038.425
```

```
## 6. RMSE value for Random Forest Regression is  2535.764
```

In conclusion, it can be observed that the eXtreme Gradient Boost Regression gives the least RMSE score and hence it performs the best. The reason for this is that the XGBoost uses both linear regression and tree algorithm. Another observation from the results for the black friday sales dataset is that the regression tree models perform better as compared to models using linear regression. RMSE for the elastic net regression improves after tuning both the parameters - alpha and lambda. Most of the regression showed that 1, 4 and 7(Gender, City_Category and Product_Category_1) are the best predictors and the features that affect the target or predict the sales for the black friday dataset the most.