Santander Customer Transaction Prediction Project

* Venkatesh Kypa

Table of Contents

1. Busines BAckground and Problem statement-----------------------------------------3
2. DATA DESCRIPTION ---------------------------------------------------------------------------------------------4
   1. Attributes of Data------------------------------------------------------------------------------------------4
   2. Importing Data----------------------------------------------------------------------------------------------4
3. METHODOLOGY------------------------------------------------------------------------------------------------- 5
   1. Data Pre-Processing---------------------------------------------------------------------------------------5
   2. Missing Value Analysis-----------------------------------------------------------------------------------12
   3. Outlier Analysis--------------------------------------------------------------------------------------------12
   4. Correlation -------------------------------------------------------------------------------------------------13
4. SPLITTING TRAIN AND TEST DATA SET--------------------------------------------------------------------15
5. MODEL DEVELOPMENT---------------------------------------------------------------------------------------16
   1. Model Performance--------------------------------------------------------------------------------------16
6. ACCURACY IMPROVEMENT----------------------------------------------------------------------------------18
7. FINALIZING THE MODEL AND PREDICTING TEST DATA SET -----------------------------------------19
8. Business Background and Problem Statement:

Back ground of the business problem here is

“At Santander, mission is to help people and businesses prosper. We are always looking for ways to help our customers understand their financial health and identify which products and services might help them achieve their monetary goals.

Our data science team is continually challenging our machine learning algorithms, working with the global data science community to make sure we can more accurately identify new ways to solve our most common challenge, binary classification problems such as: is a customer satisfied? Will a customer buy this product? Can a customer pay this loan? ”

Problem statement is

“In this problem goal is to predict the customer who will make the transaction and who will not irrespective of the money he/she spent in each transaction”

1. Data Description

Here this project contains 2 datasets one is train dataset and another one is test dataset

* 1. Attributes of the data:

Here both train and test datasets are anonymized datasets containing numeric feature variables, binary target column and the transaction ID column.

* 1. Importing data:

Including Transaction ID column, binary target column train dataset contains 202 columns and 2, 00,000 rows and same as test dataset

As mentioned before here data doesn’t contain names of the numeric independent variables which are indexed as ‘var\_0’, ‘var\_1’, ‘var\_2’……………………. ‘var\_200’,’var\_201’,’var\_202’.

As per the description the data contains missing values which we will look into while later.

Data type:

1. Transaction \_ID: object/character
2. Binary Target variable: Integer type
3. Numeric feature variable: Numeric/float type

This is a binary classification problem target variable is numeric having 0 and 1 values and 200 features.

Since customer transaction conveys nothing but a unique ID and it will not be considered for the further analysis

1. Methodology
   1. Data Pre-Processing:

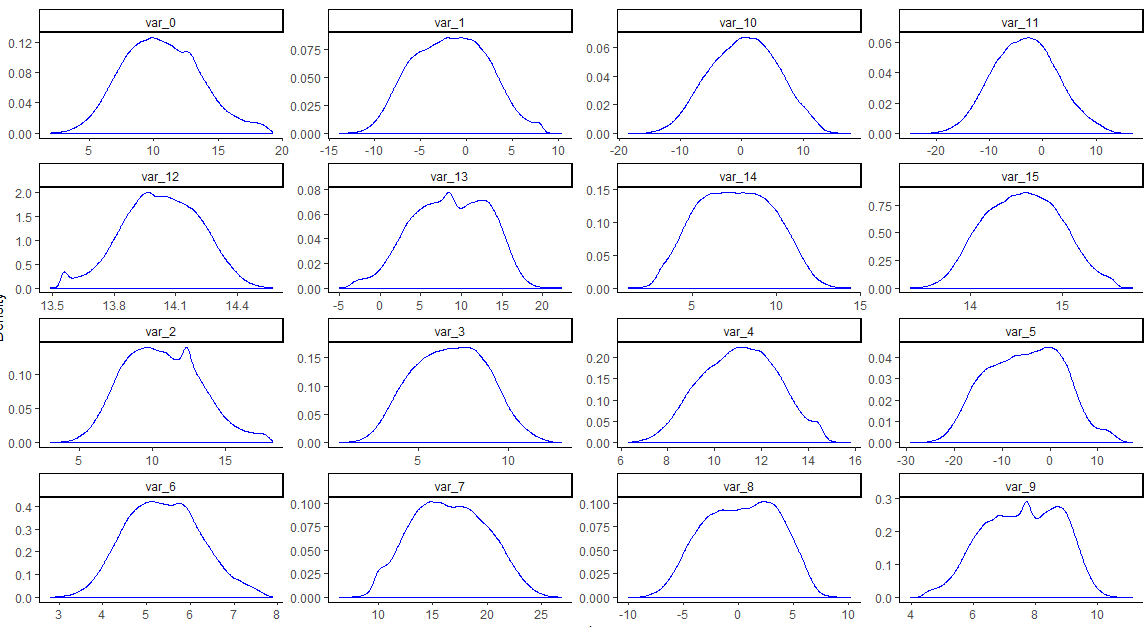
Data pre-processing is first stage of any type of project. In this stage we first do the exploratory data analysis through which we get a sense of distribution of data. If it’s numeric variable we go for distribution and if it’s categorical variable we go for histogram to get understand and sense of data. This stage of the project is generally involves data cleaning if the looks messy, looking for missing values and Imputing missing values with suitable technique like mean / mode/median/KNN ….etc., looking for outliers and dealing the outlier values.

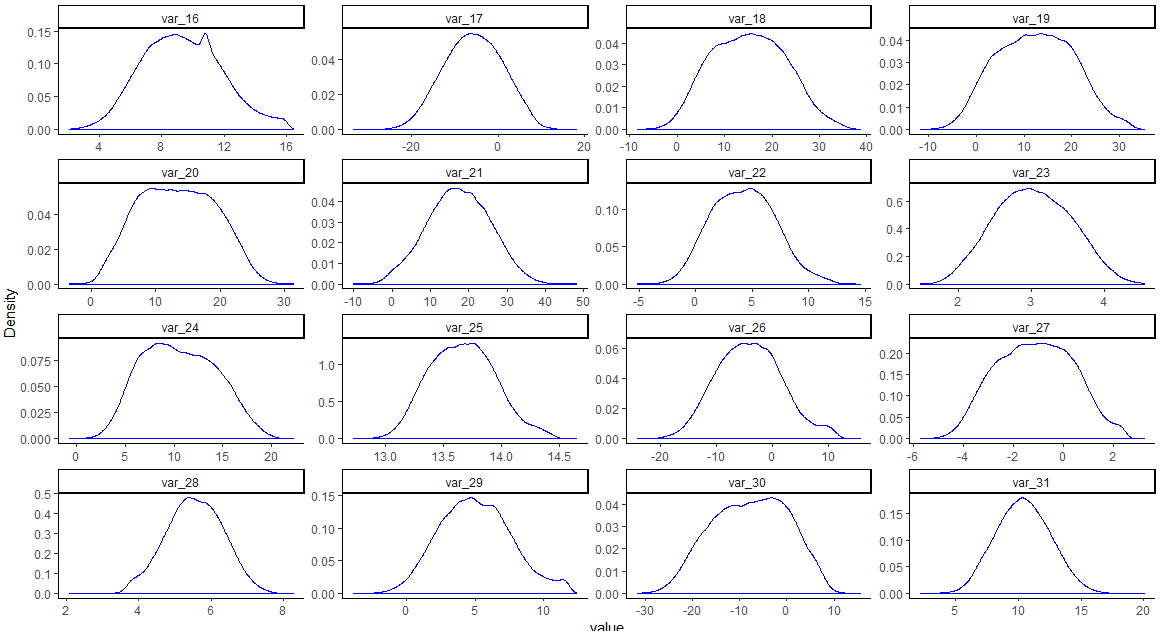
Furthermore we will look into as we move to the pre-processing method.

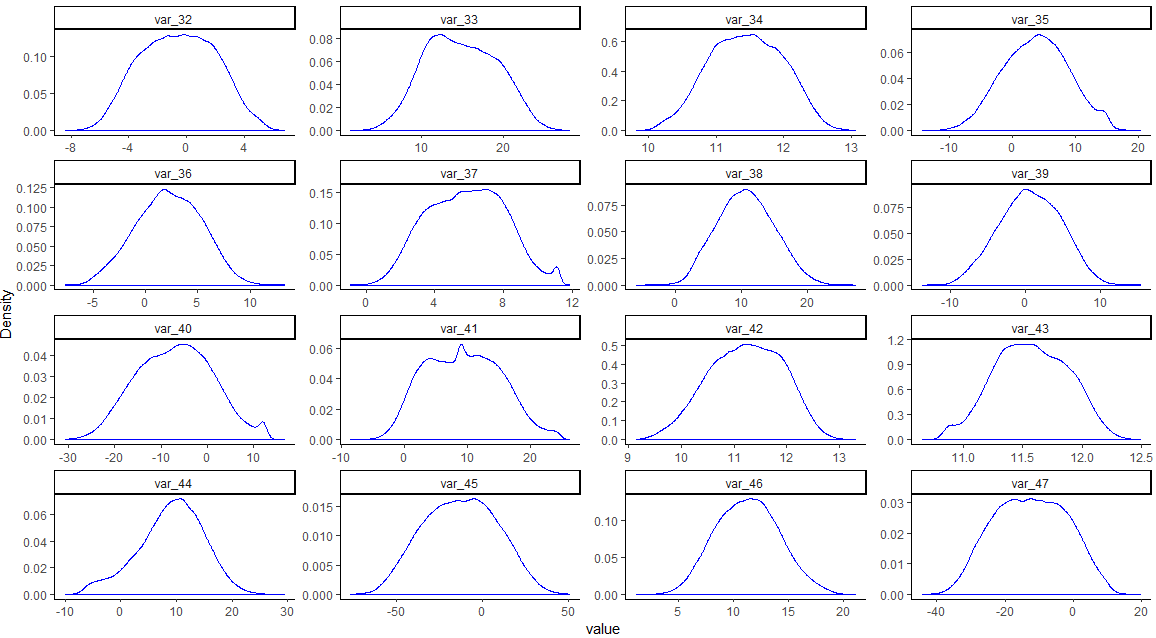
Since all the Independent variables are numeric type we will draw distribution chart

Plotting the distributions for the first 47 numeric variable

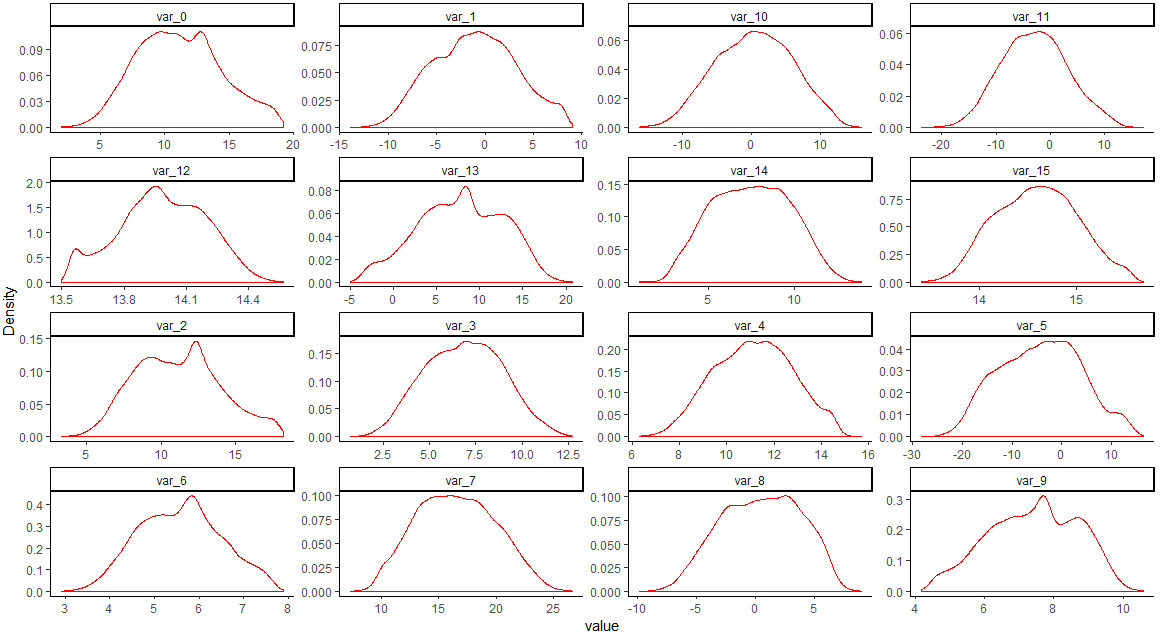
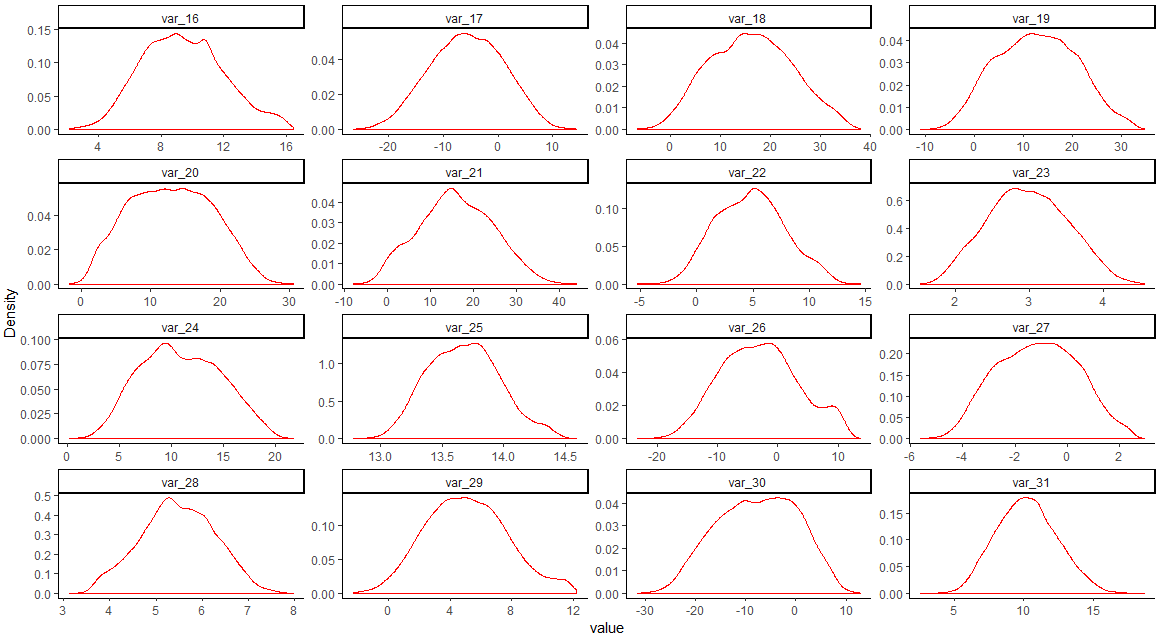
1. With target value: ‘0’

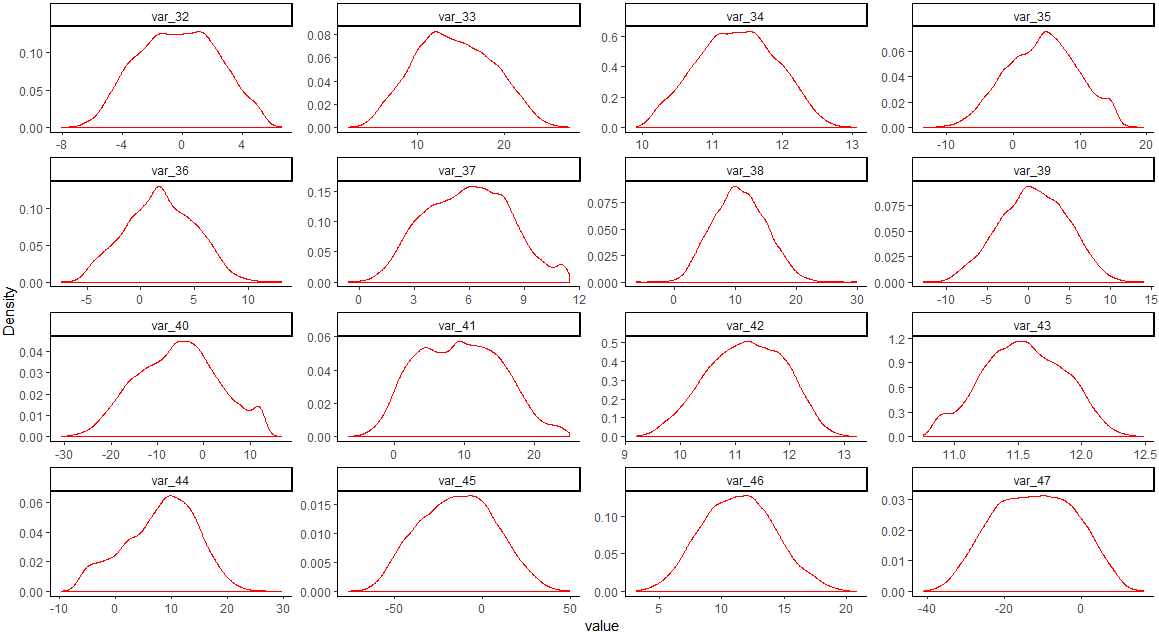






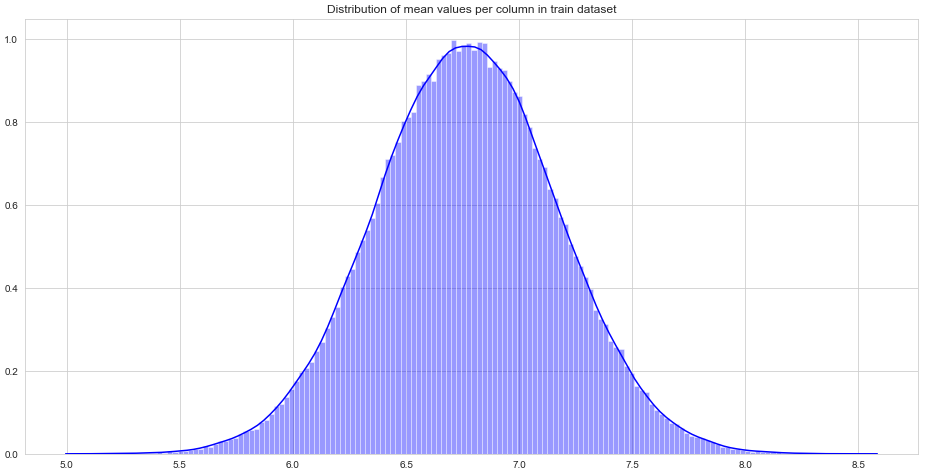
1. With target value: ‘1’





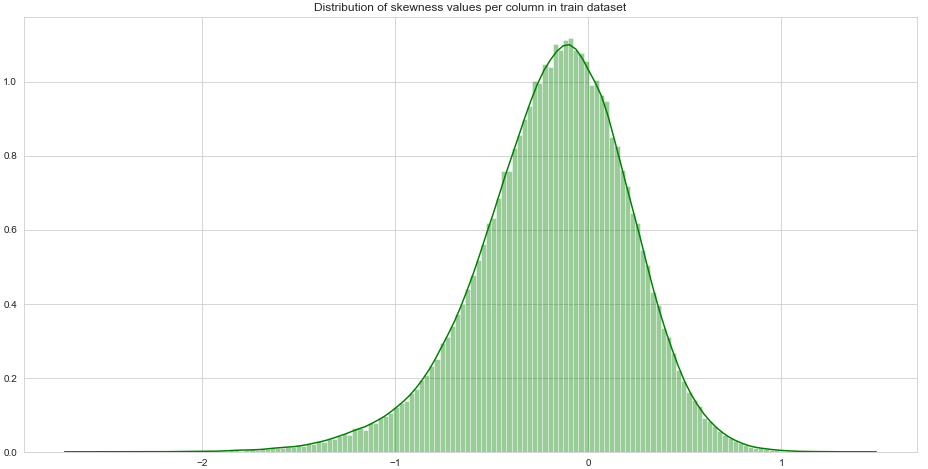
Numeric variables either with target variable 0 or 1 follow more or less normal distribution with little skewedness.

Just by looking at the description of data mean values of each numeric variable are different and varies in a considerable range now we look at the distribution of mean data of each numeric variable:

Mean values distribution follows normal distribution and ranges from

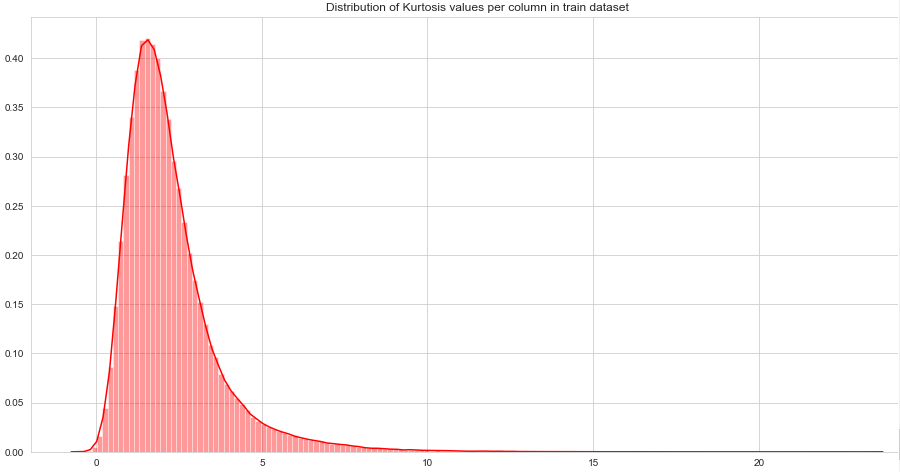
Similarly we look at the distribution data for skewness and kurtosis for each numeric variable in the train data set

Skewness Distribution:



Skewness distribution shows that it follows normal distribution with slightly skewed to left and ranges from close to -3 to 2 which shows that the all numeric variables follow normal distribution

Kurtosis distribution:



Target Class Count:

Let us find the count of target class:



Here in this data the target class is not balance i.e. ‘0’ or customer not making the transaction is in higher number than ‘1’ or customer making the transaction. If the class is not balanced then the model we create on imbalance data will be biased to ‘0’ therefore accuracy of the model can be affected badly. Therefore with proper techniques target class to be balanced and considered for building the model.

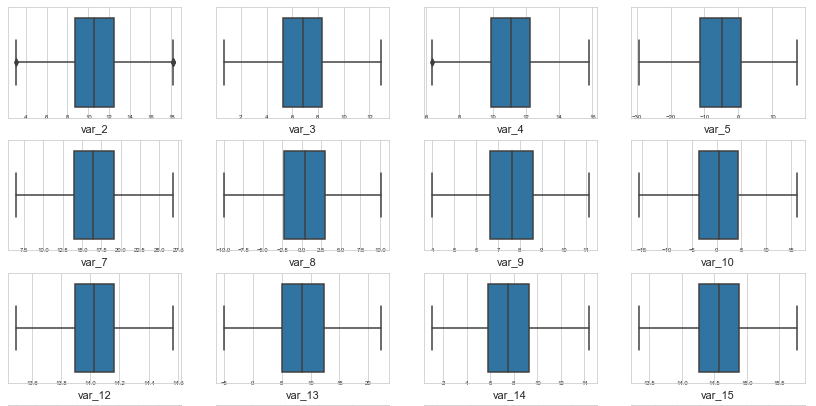
* 1. Missing Values Analysis

Though in the description of data says missing values are present but during the data pre-processing didn’t find any values are missing. Therefore no such missing data is present and moved on to the next stage of pre-processing i.e. Outlier analysis

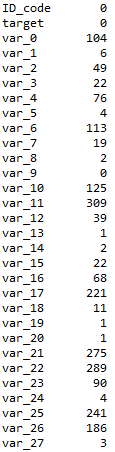
* 1. Outlier Analysis

In general we use boxplot method to find if any outliers are present in the numeric variables. Here in this data also I have plotted boxplot chart to get a sense of whether outliers are present or not.

Sample boxplot chart for Numeric variables:



Outliers found in first 27 numeric variables shown below



I found that total 200000 rows 24,896 outliers are present that constitutes ~12% of the data. Since it’s only 12% of data are outliers so I decided to remove from the dataset and continue on further data preparation

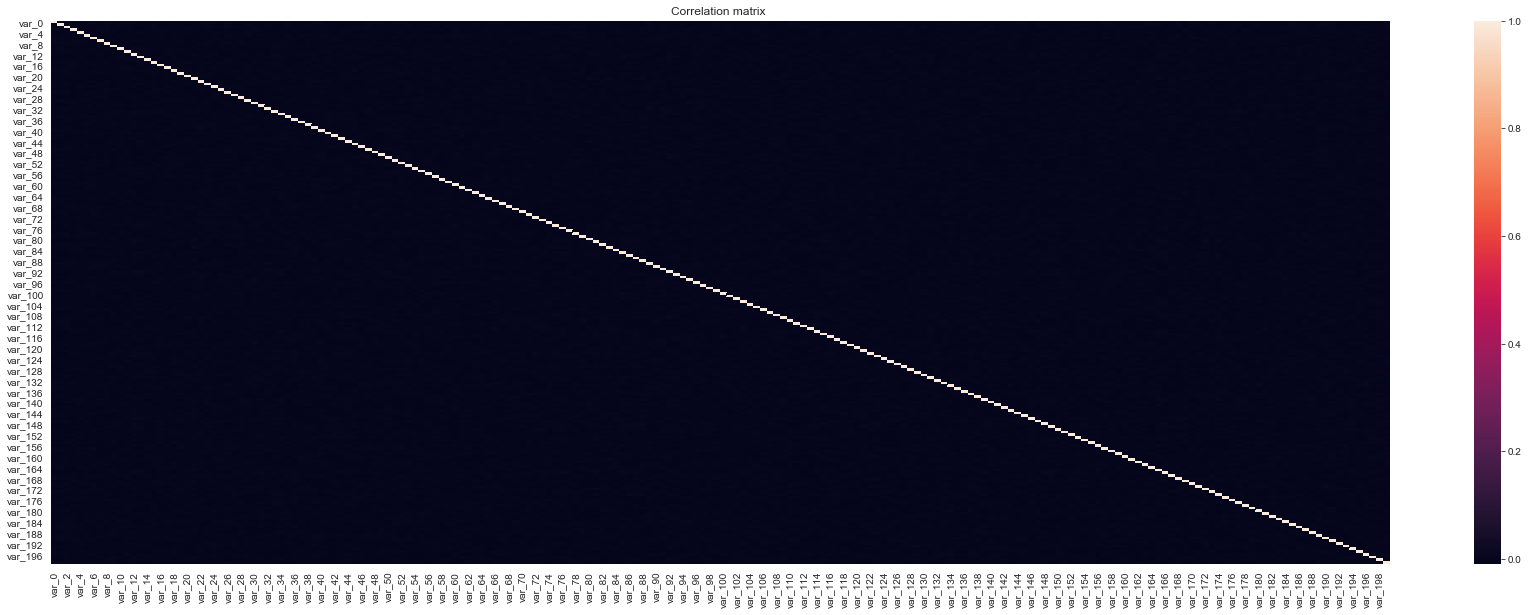
I did remove all outliers present each of the Independent numeric variables and the final training data set contains 1, 75,104 rows and 202 columns

I also decide to remove customer transaction ID which I stated it before that it is non-significant in predicting customer transaction success rate

* 1. Correlation

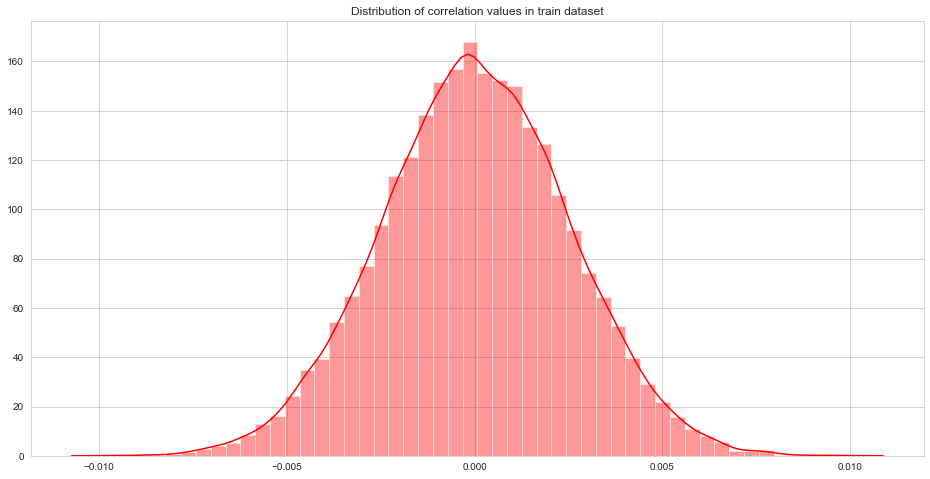
Since this require only numeric variables for analysis we consider only numeric variables i.e. from ‘var\_0’ to ‘var\_200’. We do this by plotting correlation plot for all numeric variables. This correlation analysis helps us to find if there is any multicollinearity among the numeric variables.

Here below chart shows the correlation plot for numeric variables:



The above chart shows that there is no correlation between among the numeric variables as the correlation values ranges between 0 and 0.2

Distribution of correlation value:



1. Splitting Train and Test Data set
2. Here I have used sklearn train\_test\_split in python and general sample with 75% data as train data and 25% as test data i.e. 1,31,328 as train data and 43,776 as test data
3. Here later in the case I used SMOTE technique to balance the target class
4. Model Development

Our problem statement wants us to predict the customer’s transaction irrespective of the money he/she spend on transaction. So here it is classification problem and I have used below mentioned techniques/algorithms/methods to predict customer transaction

1. Logistic Regression (without class balance)
2. Logistic Regression (with class balance) # Using SMOTE
3. Decision Tree Classifier
4. Random Forest Classifier
5. XG Boost Classifier

Here in this classification problem I have considered 2 important performance metrics to select the model

1. ROC: AUC (Area Under the Curve)
2. FNR (False Negative rate)

Since it is customer prediction problem FNR has important significance and costs the business if not controlled.

FNR tells us that predicting that customer will not make transaction when there is instance where actually customer makes a transaction which is detrimental to any business and have effect on its sales. Therefore I considered FNR. As I mentioned before since it has class imbalance and we are approaching it with specific techniques taking ‘Accuracy’ as a performance metric will not make sense rather its better if we take ROC’s AUC value

* 1. Model Performance

1. Logistic Regression (without class balance)

Performance Metrics:

1. AUC: 0.612
2. FNR: 73%
3. Accuracy: 91.62% (optional)
4. Logistic Regression (with class balance)

Performance Metrics:

1. AUC: 0.80055
2. FNR: 18.85%
3. Accuracy: 79.76% (optional)
4. Decision Tree Classifier (with class balance)

Since Decision tree is also sensitive to class balance I have decided to build model after balancing the target class

Performance Metrics:

1. AUC: 0.6753
2. FNR: 27.53%
3. Accuracy: 68.38% (optional)
4. Random Forest Classifier

Random forest classifier is not sensitive to target class balance therefore I used target class imbalance data for building Random Forest Classification

Performance Metrics:

1. AUC: 0.7021
2. FNR: 52.11%
3. Accuracy: 90.31% (optional)
4. Improving Accuracy

For improving accuracy we have used XGBoost as an ensemble technique

Xbboost hyperparameters tuned parameters:

Tuned xgboost parameters = {"eta":0.3, "gamma":10, "max\_depth":5,

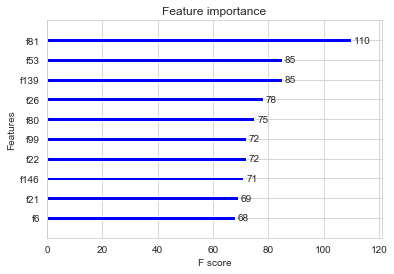
"min\_child\_weight":2,"booster":"gbtree", "subsample":0.5, 'objective': 'binary:logistic', max\_depth=6, n\_estimators=100, learning\_rate=0.3, gamma=3}

XG Boost Classifier (Gradient Boosting Technique)

Performance Metrics:

1. AUC: 0.931
2. FNR: 6.81%
3. Accuracy: 93.04% (optional)

Important top 10 Features in Predicting Customer Transaction:



Here in above chart f81 stands for ‘var\_81’, f53 stands for ‘var\_53’. Therefore ‘f’ feature stands for actual variable in the data

Performance Metrics for various models used:

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Model | AUC | FNR |
| 1 | Logistic Regression (without target class balance) | 0.612 | 73% |
| 2 | Logistic Regression (with target class balance) | 0.801 | 18.85% |
| 3 | Decision Tree Classifier | 0.675 | 27.53% |
| 4 | Random Forest Classifier | 0.702 | 52.11% |
| 5 | XGBoost Classifier | 0.931 | 6.81% |

Clearly XG Boost is the ‘Winner’ therefore we consider that XBBoost classifier performs well

1. Finalizing the Model and Predicting The Test data

Now we finalized XGBoost classifier as our final model for this problem and now we will load test data given in the problem statement and try to predict the target class

We have trained XGBoost model on entire training dataset and used that model to predict on test data. Also, we have saved model for later use.