|  |
| --- |
| Project- Report |
| Santander Customer  Transaction Prediction |
| Kunal pandya |

**COURSE - DATA SCIENCE (Edwisor)**

**Table of contents**

**• Introduction…………………………………………………………..…...3**

**1. Understanding data……………………………………………..…….4**

**2. Data Pre processing…………………………………………..……….5**

2.1 Missing Value Analysis………………………………...…………5

2.2 Outlier Analysis………………………………………..……… ……7

2.3 Feature Selection…………………………………….………….. .9

2.4 Feature Scaling…………………………………………………….12

**• Visualization…………………………………………………………..…14**

**3. Modelling…………………………………………………………………15**

3.1 Logistic Regression…………………………………..…………..15

3.2 Random Forest………………………………………..…………..16

3.3 Naïve bayes……………………………………………..…………..17

**4. Model Selection…………………………………………….……..…19**

**5. Model Fitting & Conclusion………..…..………………………20**

Introduction

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**​ - In this challenge, we need to identify which customers will make a specific transaction in the future, irrespective of the amount of money transacted.

**1. Understanding data**

Two data-sets are provided namely **“train”** & **“test”** both of which contains numeric feature variables & a string **ID\_code** column. The train dataset consists of a binary target column & the task is to predict the value of target column in the test dataset.

In addition to ID code & target column, the train data consists of **200 variables (var\_0 to var\_199)** with **200000** observations which impacts the target variable. The test data also contains 200 **(var\_0 to var\_199) variables** in addition to ID code with 200000 observations. There is no target column & the objective is to predict the target values.

A machine learning model has to be built on the train data and to be applied on test data to predict the target column. As this is a classification problem, models such as **Logistic regression, Random forest & Naïve bayes algorithms** can be applied. To choose the best model, the train dataset can be divided into subset train & subset test data. After building models on the subset train, it can be applied to the subset test data to check the accuracy & other evaluation metrics. The best model can be selected considering these metrics and it can be applied to the main test dataset to find the target values.

**2. Data Pre processing**

**2.1 Missing Value Analysis**

**Theory**

Missing values occur when no data is recorded for an observation; it was intended to make an observation, but because of some reason did not. Missing data are a common occurrence and can have a significant effect on the statistical analysis. The concept of missing values is important to understand in order to successfully manage data. If the missing values are not handled properly by the researcher, then he/she may end up drawing an inaccurate inference about the data. Due to improper handling, the result obtained by the researcher will differ from ones where the missing values are present.

**If the total missing values of a feature is less than 30% of the total observations, then the missing observations may be removed from original data. If they are greater than 30%, then it is suggested to impute them with methods such as mean, mode, median, knn imputation methods.** Mean & median imputation works by calculating the mean/median of the non-missing values in a column and then replacing the missing values within each column separately and independently from the others. It can only be used with numeric data. Mode imputation is used for categorical variables. KNN stands for k nearest neighbors and works by finding ‘k’ no. of nearest neighbors & then imputing based on those neighbors.

**Methodology**

**• Create a data frame with total no of missing values for each variable • Calculate percentage of missing values against each variable**

**• Sort in descending order**

**• In this project, missing values are decided to be imputed irrespective of their percentage, to avoid information loss**

**Observations & Results**

No missing values were found in the train dataset, both in R & Python. **2.2 Outlier Analysis**

**Theory**

An outlier is an observation that is abnormal compared to other observations in that dataset. One of the most important tasks from large data sets is to find an outlier because outliers can significantly alter the results even though they are present in small proportions.

To find an outlier, inter quartile range (IQR) is found first. IQR represents the middle 50% of the data. The position of first quartile can be found using formula (N+1)/4 & third quartile can be found by 3\*(N+1)/4 where N is the total no. of observations. The difference of the values in the first & third quartiles is the IQR**. If any observation falls below 1.5 times IQR from the first quartile value, or if it falls above 1.5 times IQR from the third quartile value, then the value can be qualified as an outlier.**

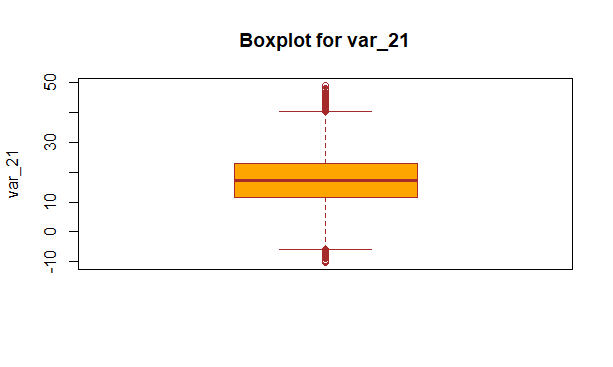
Outliers can be found using box plot method which can be plotted in both R & Python. After finding the outliers, they can be removed from the dataset or they can be imputed by KNN method.

Observations & Results

Original train data contained a total of 200000 observations. After removing the outliers, **it reduced to 175073**. So, **a sum of 24927 outliers were detected from all variables and removed from the train data.**

In this project, KNN imputation was not applied as KNN algorithm was found to be time consuming.

one box plot is shown below:



2.3 Feature Selection

**Theory**

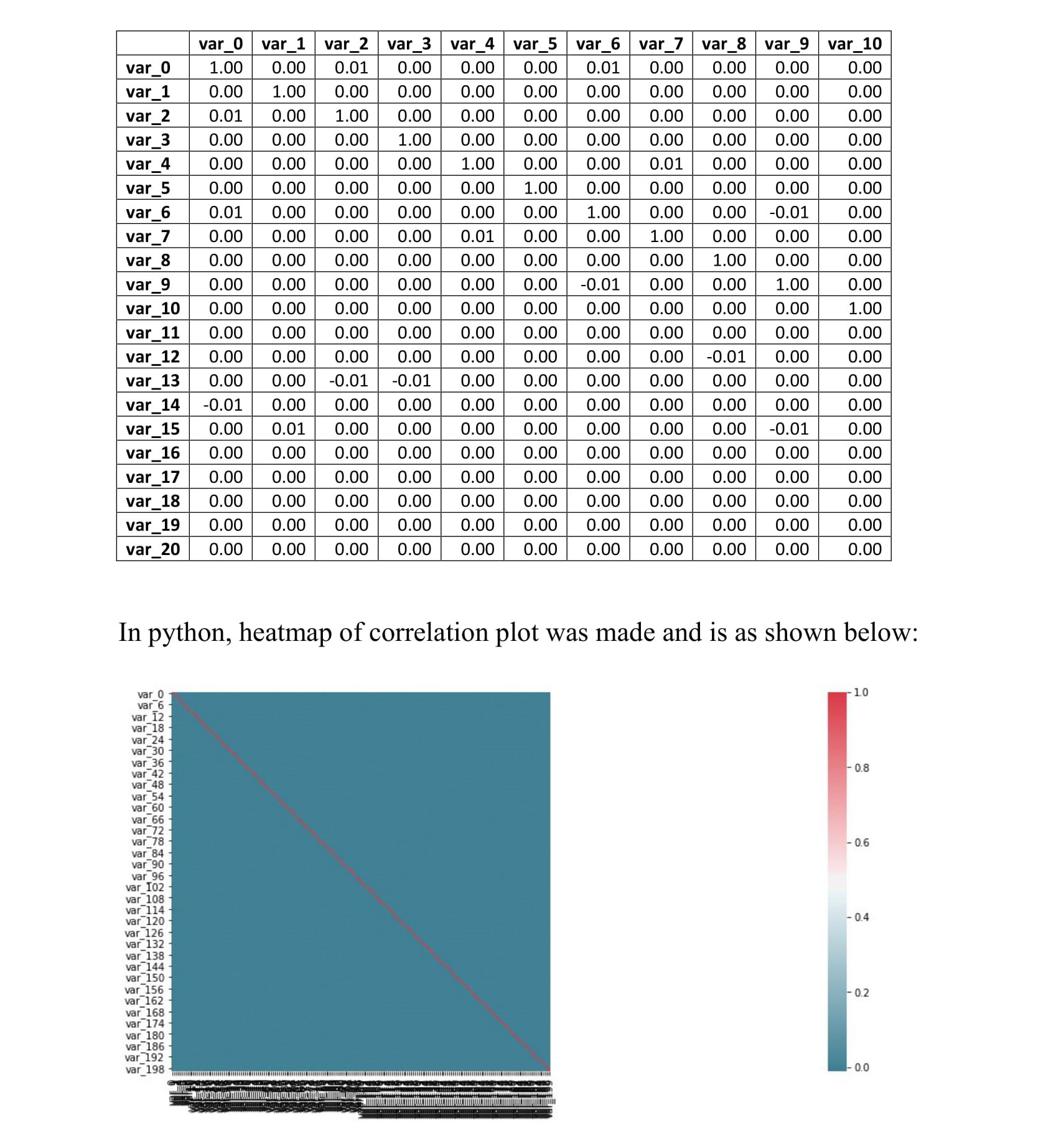
Feature Selection is the process of selecting those features which contribute most to the prediction variable. Having irrelevant features in data can decrease the accuracy of the models and make the model learn based on irrelevant features. For this, relation between different variables are evaluated and if two variables are strongly correlated with each other, then one of them may be dropped. Correlation analysis is used in feature selection for numerical variables and chi square test is used for categorical variables.

**In correlation analysis, correlation coefficient is calculated between two variables which ranges from -1 to +1**. Correlation coefficient approaching -1 or +1 means that both the variables are strongly correlated (negatively & positively correlated respectively). While value close to 0 implies little or no correlation.

In chi square test, a null hypothesis is formulated which states that the given two variables are independent of each other and an alternate hypothesis states that the two variables are not independent. A critical value is found using chi square value and degree of freedom, which if less than chi square value, alternate hypothesis is accepted and if greater than chi square, null hypothesis is accepted.

Observations & Results

In R , **correlation matrix** was prepared and a subset of the matrix is as shown below:



**CORELATION MATRIX**

**If correlation value is greater than 0.8 or less than -0.8, then it can be safely assumed that the two variables in consideration are highly correlated and one of them may be dropped. But, from correlation matrix in R, all the values were close to 0, which indicates that they are independent of each other.**

**HEAT MAP**

**Similar conclusion can be drawn from the heat map, in which no shade of red color which indicates dependency of the variables were found. Hence, none of the features were eliminated from the train data. Correlation plot in R , was not plotted as the process was time consuming.**

**2.4 Feature Scaling**

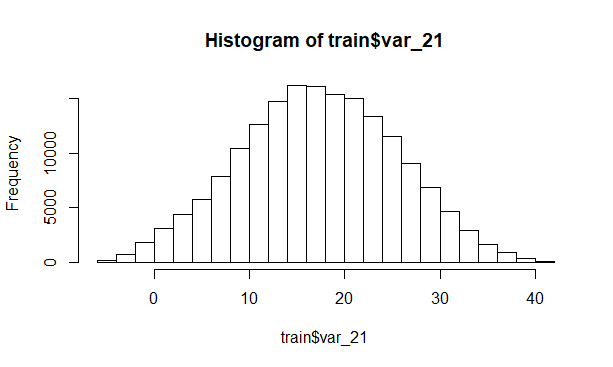
**Theory**

**Feature scaling means adjusting data that has different scales into the same range.** Feature scaling is an important technique in Machine Learning and it is one of the most important steps during the preprocessing of data before creating a machine learning model. Most of the times, the dataset contains features highly varying in magnitudes, units and range. The two most important scaling techniques is Standardization and Normalization.

Normalization is the process of rescaling the features to the range of 0 to 1. Standardization is the process of rescaling data to have a mean of 0 and a standard deviation of 1. This is usually applied to the dataset which is normally distributed.

**Observations & Results**

**Histograms were plotted to check the normality of the data and the plots of some of the variables are as follows:**

****

From the above histogram, it can be concluded that the data is normally distributed. So, to scale the data, standardization was applied.

**Range of the data before standardization = ( -76.5496, 61.1035 )**

**Range of the data after standardization = ( -3.1157, 3.1105 )**

**• Visualization (Bar-graph)**

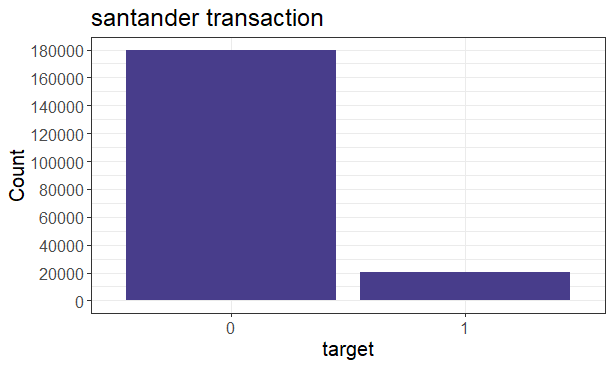
Visualization is a technique used to see insights of the data with the help of plot. It is a graphical representation with effective plot.

Our data set contains two categories of **target** **class (0 & 1).**

**Class 0 = customer who have not transacted**

**Class 1 = customer who have transacted**

Here , Bar graph is showing **that 90% of the data is falling under category 0 (no) and 10% of the data is falling under category 1(yes).**

****

3. Modeling

**3.1 Logistic Regression**

**Theory**

Logistic Regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable (0 or 1). In this, the independent variables should not be correlated with each other. That is, the model should have little or no multi-collinearity. Logistic Regression is one of the most popular ways to fit models for categorical data, especially for binary response data in Data Modeling. Logistic Regression is used when the dependent variable is categorical & it requires quite large sample sizes. Even though logistic regression is frequently used for binary variables, which is called Binary logistic regression, it can also be used for categorical dependent variables with more than 2 classes. In this case it’s called Multinomial Logistic Regression.

Results

|  |  |  |
| --- | --- | --- |
| Model – R | Accuracy | FNR |
| Logistic  Regression | 91.78 | 73.56 |

|  |  |  |
| --- | --- | --- |
| **Model-python** | Accuracy | FNR |
| Logistic Regression | 59.98 | 92.94 |

3.2 Random Forest

**Theory**

Random forest consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest gives a class prediction and the class with the most votes becomes the model’s prediction. Random forest works on the principle that a large number of relatively uncorrelated models (trees) operating as a group will outperform any of the individual constituent models. While some trees may be wrong, many other trees will be right, so as a group the trees are able to move in the correct direction. The predictions (and therefore the errors) made by the individual trees need to have low correlations with each other.

Results

|  |  |  |
| --- | --- | --- |
| Model – R | Accuracy | FNR |
| Random – Forest | 92.23 | 79.50 |

|  |  |  |
| --- | --- | --- |
| Model – python | Accuracy | FNR |
| Random – Forest | 90.58 | 100 |

3.3 Naïve Bayes

**Theory**

Naive Bayes is a probabilistic machine learning algorithm that can be used in a wide variety of classification tasks. The name naïve is used because it assumes the features that go into the model is independent of each other. That is changing the value of one feature, does not directly influence or change the value of any of the other features used in the algorithm.

The fundamental Naive Bayes assumption is that each feature makes an independent & equal contribution to the outcome. The Naïve Bayes classifier is based on the Bayes theorem which is given as

If, P (A | B) = P ( A & B ) / P(B)

& P(B | A) = P ( A & B ) / P(A)

Then, P (A | B) = [ P ( B | A) \* P(A) ] / P(B)

Using Bayes theorem, we can find the probability of A happening, given that B has occurred. Here, B is the evidence and A is the hypothesis. The assumption made here is that the predictors/features are independent. That is presence of one particular feature does not affect the other. Hence it is called naive.

Result

|  |  |  |
| --- | --- | --- |
| Model – R | Accuracy | FNR |
| Naïve Bayes | 92.46 | 64.79 |

|  |  |  |
| --- | --- | --- |
| Model – python | Accuracy | FNR |
| Naïve Bayes | 92.47 | 64.08 |

4. Model Selection

Here is the comparison of all the models we have developed.

|  |  |  |
| --- | --- | --- |
| M.L. Model – R | Accuracy | FNR |
| Logistic regression | 91.78 | 73.56 |
| Naïve Bayes | 92.46 | 64.79 |
| Random Forest | 92.23 | 79.50 |

|  |  |  |
| --- | --- | --- |
| M.L. Model - Python | Accuracy | FNR |
| Logistic regression | 59.98 | 92.94 |
| Naïve Bayes | 92.47 | 64.08 |
| Random Forest | 90.58 | 100 |

**As the random forest model was not convincing with the results due to some unidentified errors, it was considered to be a dummy model. So, the selection was to be done among Logistic Regression & Naïve Bayes model.**

**In R & Python , naïve bayes was performing well among all the models so, we will freeze naïve bayes and Naïve Bayes was chosen for the prediction of target variable in test data.**

**5. Model Fitting & Conclusion**

* **After the selection of the best possible model, it was fit to the large test dataset for which the target variable was to be predicted. Data pre-processing was also done on the test data for maximum accuracy.**
* **No missing observation was found in missing value analysis.**
* **Feature scaling was also done because the original train dataset was trained on the scaled data, thus the predicted results would be accurate only if the model fitting is done on the scaled test data.**
* **After fitting the model, the predicted target results were saved against the ID\_code numbers. If the ID\_code represents an individual, it was predicted that out of 200000 customers, a total of 7423 & 7430 would make the transaction according to R model and Python model respectively.**