Capstone Project 1 – Predicting Loan Default

Machine Learning - In Depth Analysis

Contents

[Introduction 2](#_Toc3662569)

[The Lending Club Data Set 2](#_Toc3662570)

[Problem Statement 2](#_Toc3662571)

[Assumptions 2](#_Toc3662572)

[Data Modelling 2](#_Toc3662573)

[1. Logistic Regression 3](#_Toc3662574)

[2. Random Forest 3](#_Toc3662575)

[3. K Nearest Neighbors (KNN) 3](#_Toc3662576)

[4. BaggingClassifier 4](#_Toc3662577)

[5. AdaBoost classification with DecisionTree 4](#_Toc3662578)

[Results 4](#_Toc3662579)

[1. Area Under ROC Curve 4](#_Toc3662580)

[2. Root Mean Square Error 5](#_Toc3662581)

[3. Accuracy Score 6](#_Toc3662582)

[4. Precision and Recall 7](#_Toc3662583)

[5. Confusion Matrix 8](#_Toc3662584)

[Conclusion 10](#_Toc3662585)

# Introduction

The Lending Club is the world's largest peer-to-peer lending platform. Lending Club enables borrowers to create loan listings on its website by supplying details about themselves and the loans that they would like to request. On the basis of the borrower’s credit, credit history, desired loan amount and the borrower’s debt-to-income ratio, Lending Club determines whether the borrower is credit worthy and assigns to its approved loans a credit grade that determines payable interest rate and fees.

# The Lending Club Data Set

The Dataset used in this project is The Lending Club dataset available from kaggel website. (https://www.kaggle.com/wordsforthewise/lending-club/home). It is a real world data set which contains 2004126 rows of loan listing and 150 columns (attributes) of the each loan listing from year 2007 to 2018 Q2. Out of 150 columns, some columns have missing data and some columns are not needed for the analysis. After applying several data wrangling method, data set contains 2004095 rows and 83 columns. 4 columns are of data type *category*, 4 columns are of data type *datetime64*, 66 are of data type *float64*, and 9 are of data type *object*.

# Problem Statement

The Lending Club operates an online lending platform that enables borrowers to obtain a loan and investors to purchase notes backed by the payments made on loans. Investors can decide on investing on a loan applications based on the credit history, FICO score, income, current job status and many other attributes of borrowers mentioned in the application. This process comes with high risk of borrowers defaulting loans. Hence there is a need to classify each borrower as defaulter or not using the data collected when loan application is submitted. The purpose of this exercise is to predict if borrower will default the loan or not and help investors make a decision based on that.

# Assumptions

The Lending Club Dataset of all accepted loans from year 2007 to Q2 2018 is an imbalance dataset with very small number of loan applications which were defaulted. So we used the 'Charged off' loan status as an indicator of default.

Most of the features in the dataset represented information which are not available to investors at the time of loan application. These features were provided by The Lending Club after or during the loan approval process. So we used the features 'fico score' and 'dti' (Debt to Income Ratio) which are available to investors as the independent variables for our analysis.

# Data Modelling

To create a predicting model, Data has been cleaned saved in clean\_loan.csv file. We found in Exploratory Data Analysis that fico\_score and dti (Debt to Income Ratio) are two valuable features that we are going to use to model and predict if loan is going to default or not.

The data set has very small number of default loans and that’s not enough to train the classification models. There are significant amount of loans which are categorized as charged off. We used the charged off loan status as default status and trained the model and verified.

This is classification problem with 2 possible target values of 0 = no default and 1 = default. We will be using several classification algorithms from scikit-learn library.

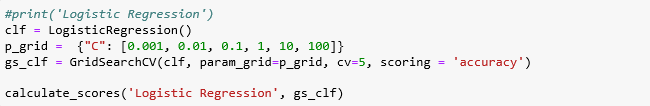
Also this is an imbalance dataset with less than 10% rows with default = 1 values. So we take a subset of the loan data which has equal number of default and no default rows.

This is a classification problem of predicting if a loan will default or not. So different models are build, fit and validated. To do the cross validation, we split the data set in training and test data sets in 80/20 ratio.

The following Simple Classification algorithms are used to model the data:

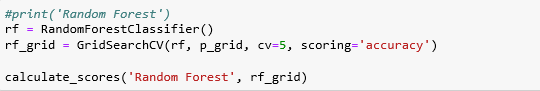
## Logistic Regression

Logistic Regression is one of the basic regression algorithm used for predicting a categorical target variable. We used a GridSearchCV() to estimate the best value of hyper-parameter 'C' and to do the cross validation with 5 folds.



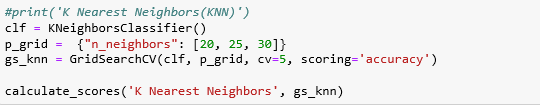
## Random Forest

A random forest is an Ensemble classifier fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement if bootstrap=True (default).



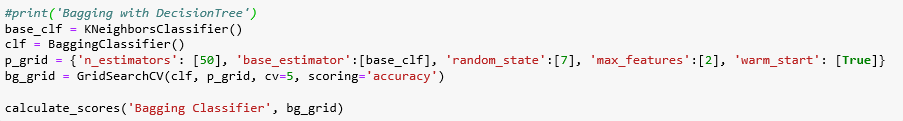
## K Nearest Neighbors (KNN)

KNN algorithm is one of the simplest classification algorithm. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. GridSearchCV() is used to estimate the best value of hyper-parameter and to perform cross validation using 5 folds.



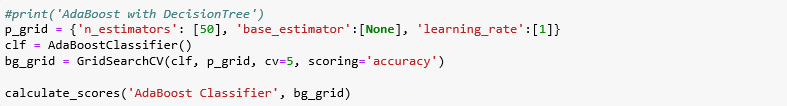
## BaggingClassifier

BaggingClassifier is an ‘ensemble’ method. Bagging constructs ‘n’ classification trees using bootstrap sampling of the training data and then combines their predictions to produce a final meta-prediction. GridSearchCV() is used to estimate the best value of hyper-parameter and to perform cross validation using 5 folds together.



## AdaBoost classification with DecisionTree

Adaptive Boosting is used with other types of learning algorithms to improve performance. The output of the other learning algorithms ('weak learners') is combined into a weighted sum that represents the final output of the boosted classifier. AdaBoost is adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers.



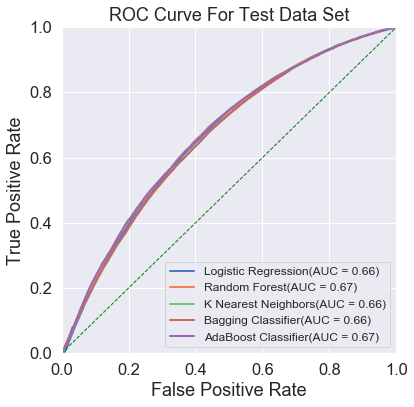
# Results

To evaluate the performance of models, following scores are measured.

## Area Under ROC Curve

The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The best possible prediction method would yield a point in the upper left corner or coordinate (0, 1) of the ROC space, representing 100% sensitivity (no false negatives) and 100% specificity (no false positives). The diagonal divides the ROC space. Points above the diagonal represent good classification results; points below the line represent bad results. A predictor with largest AUC value performs the best.

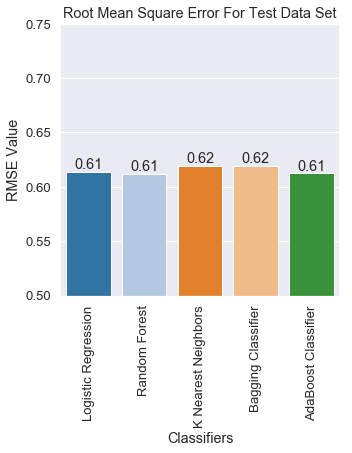
In the plot below, AdaBoost Classifier and Random Forest classifiers performed best with AUC = 0.67. Logistic Regression, K Nearest Neighbors and Bagging Classifier algorithms performed second best with AUC = 0.66



## Root Mean Square Error

RMSE of an estimator measures the square root of average of the squares of the errors—that is, the square root of average squared difference between the estimated values and actual values. The RMSE is a measure of the quality of an estimator—it is always non-negative, and values closer to zero are better.

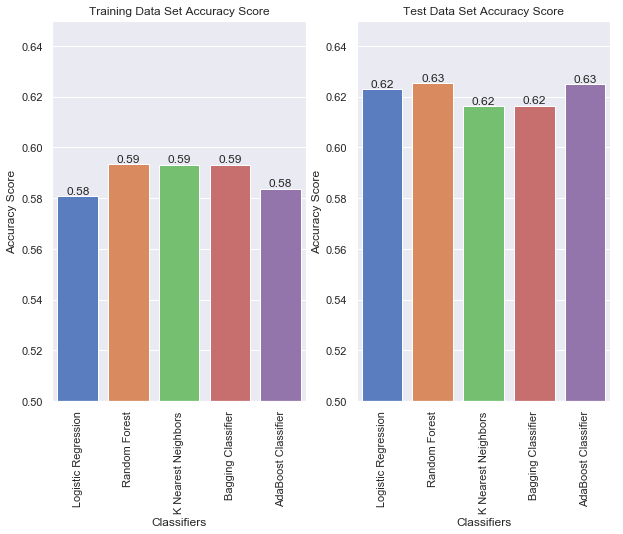
In the bar graph below AdaBoost, Random Forest and Logistic Regression performed best as minimum RMSE value of 0.61.



## Accuracy Score

Classification accuracy score is the number of correct predictions made divided by the total number of predictions made.

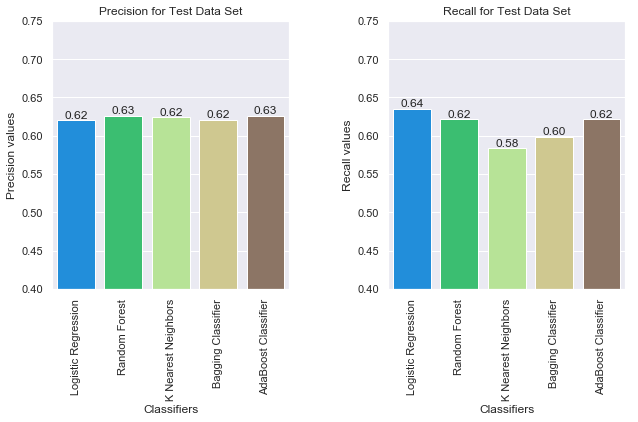
The bar graphs below show the comparison of accuracy scores for both training data set and test dataset. AdaBoost, and Random Forest classifiers performed best with accuracy score of 0.63 on test data set.



## Precision and Recall

Precision is about how precise/accurate the model is out of those predicted positive, how many of them are actual positive. Recall actually calculates how many of the Actual Positives the model capture through labeling it as Positive (True Positive). Recall shall be the model metric used to select the best model when there is a high cost associated with False Negative.

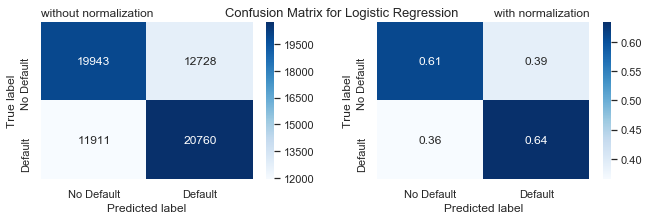
The comparison of Precision and Recall values below in the bar graphs show that Logistic Regression performed best with Recall value of 0.64.

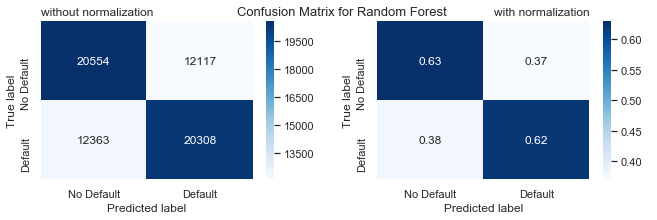


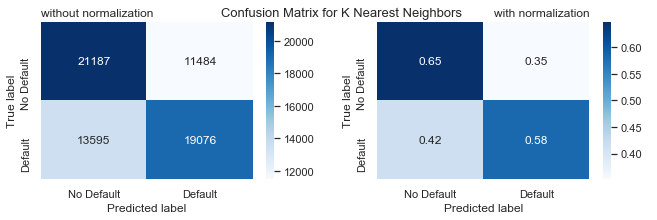
## Confusion Matrix

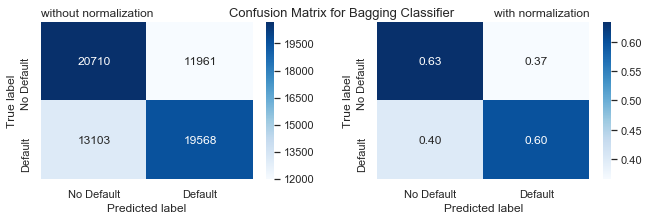
A table of confusion or confusion matrix, is a table with two rows and two columns that reports the number of false positives, false negatives, true positives, and true negatives. This allows more detailed analysis than mere proportion of correct classifications (accuracy).

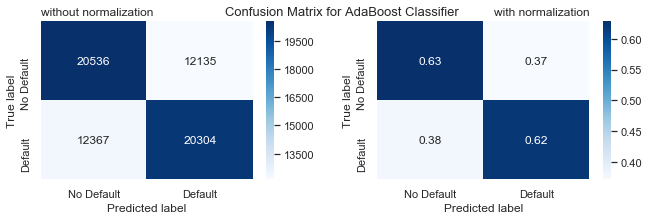
The plots below show the confusion matrix for each of the predictors.











# Conclusion

###### Predicting Loan Default in The Lending Club dataset is a real business problem. The Loan entries in the dataset have already been screened through the existing algorithms to predict the loan defaults which makes 'predicting loan default' from this dataset a highly difficult problem. This exercise tried to estimate an improvement over the existing solution.

###### I ran 5 classification algorithms and got best AUC score of 0.67, best accuracy score of 0.63, best recall value of 0.64 and best precision value of 0.63. These scores are good considering the fact that the problem is highly difficult real business problem.