Pattern Recognition & Machine Learning

Lab-3 :- *“Random Forest & Bagging”*

Objectives

Consider the credit sample dataset, and predict whether a customer will repay their credit within 90 days or not. To tackle this problem, use Random Forest and Bagging Classifiers with Decision Tree Classifier as base estimator.

Procedures

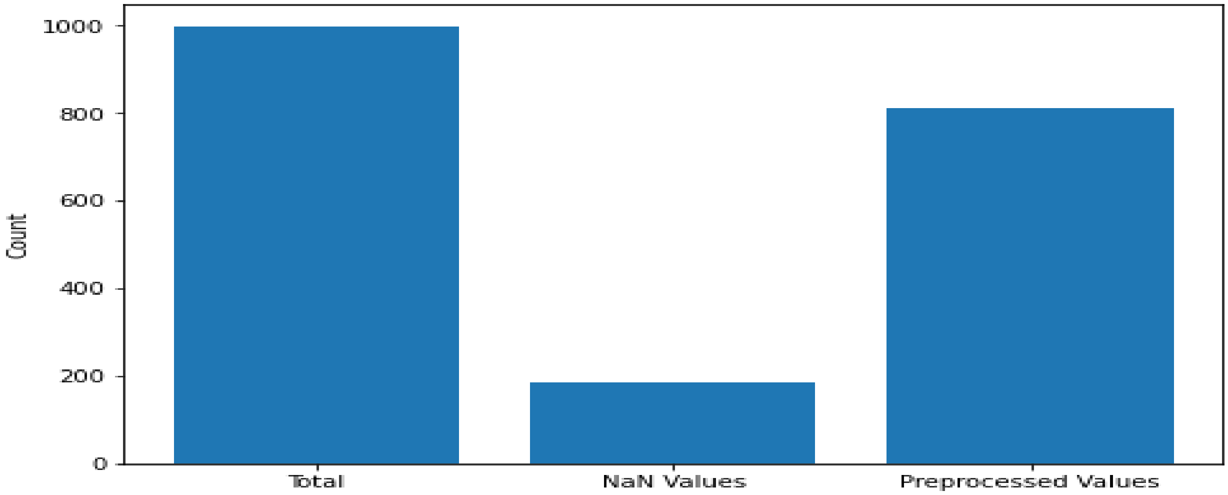
1. Remove invalid entries from Dataset and divide it into X and Y, accordingly.
2. Plot distribution curves for target variable and features, either histogram or continuous distribution curve.
3. Use Stratified K-Fold to Split Dataset into Train and Test Portions.
4. Construct Random Forests and Bagging Classifiers, with their varying parameters.
5. Use Grid-Search-CV to find best models amongst them, plot their corresponding ROC-AUC Curves and compare their performances.

Libraries / Dependencies

1. Pandas
2. Sklearn
3. Seaborn
4. Matplotlib

Inputting the Datasets

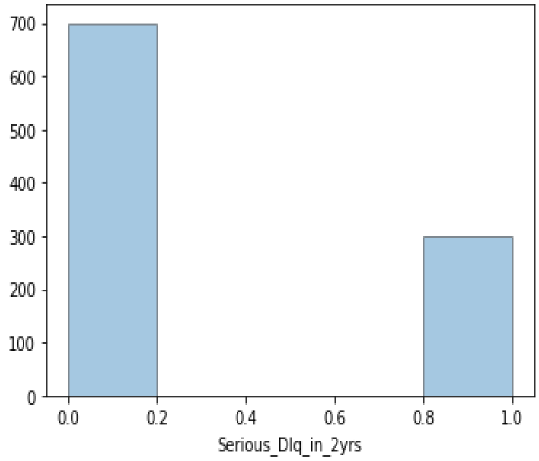
From the given link, a csv file was obtained and some of the entries were blank, as observed accordingly. In order to handle that issue, those blank entries were assigned NaN values and then dropped from the dataset. It can be clearly seen as following :-



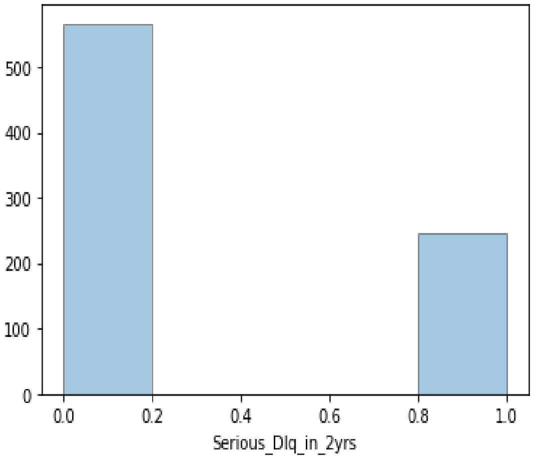
Changes observed in distribution of data points in respect to Target variable were as follows :-

Target Variable Distribution Histogram

Before Removing NaN Values



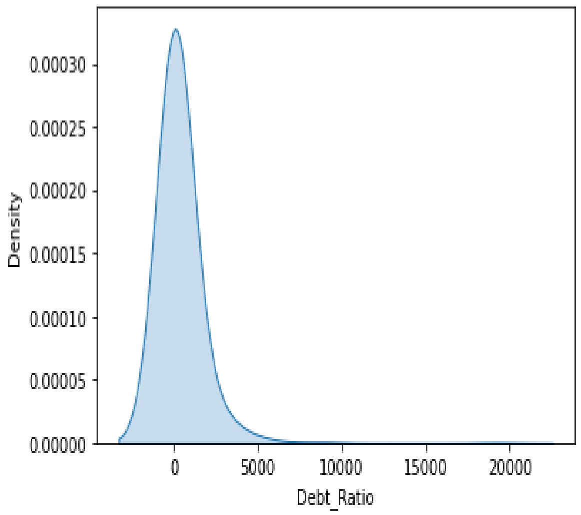
After Removing NaN Values



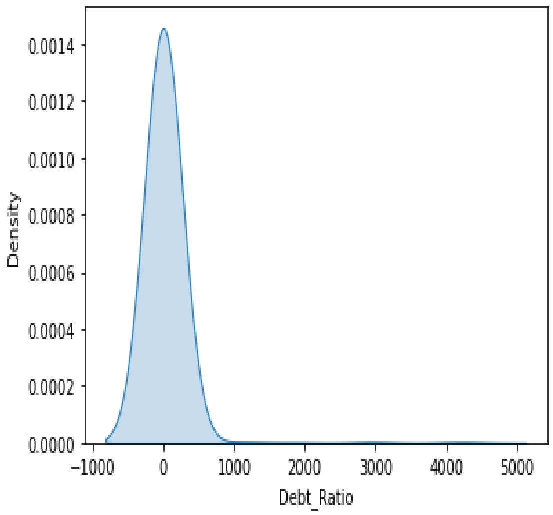
Some major changes observed in distribution of data points in respect to each feature were as follows :-

1. Debt_Ratio Distribution Curve

Before Removing NaN Values

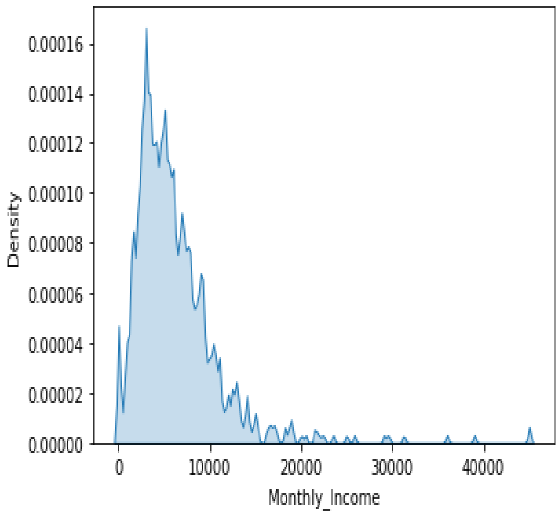


After Removing NaN Values

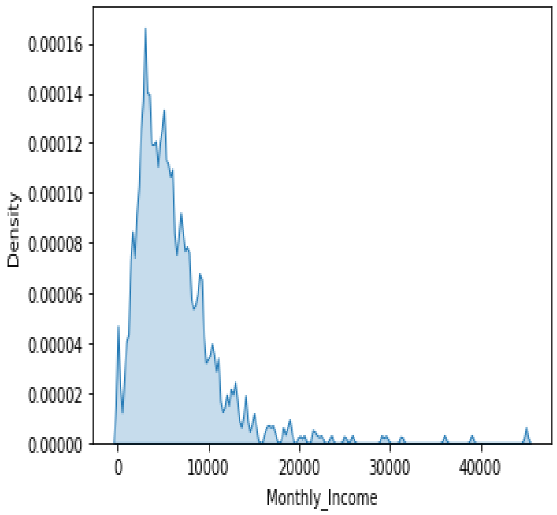


2. Monthly_Income Distribution Curve

Before Removing NaN Values

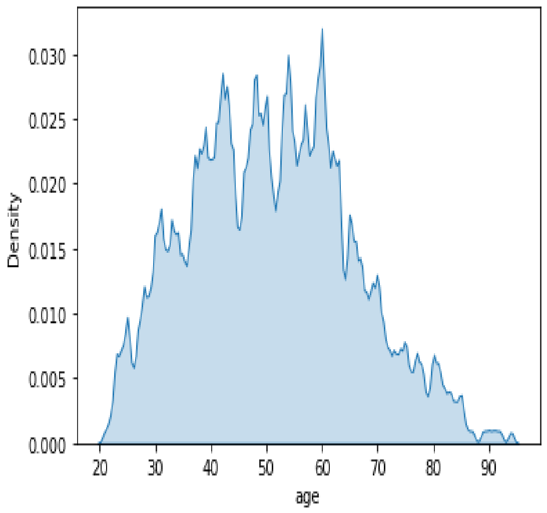


After Removing NaN Values

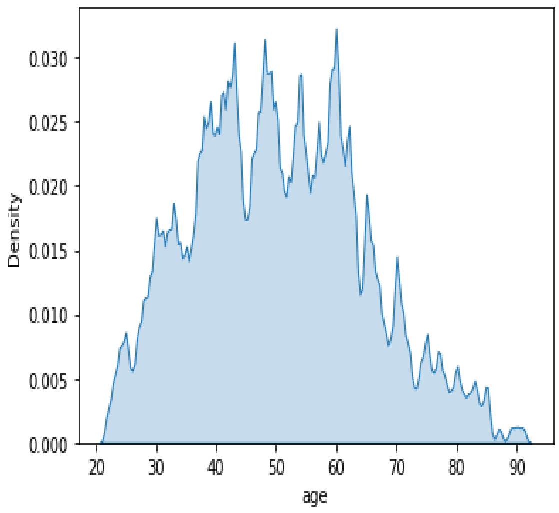


3. Age Distribution Curve

Before Removing NaN Values

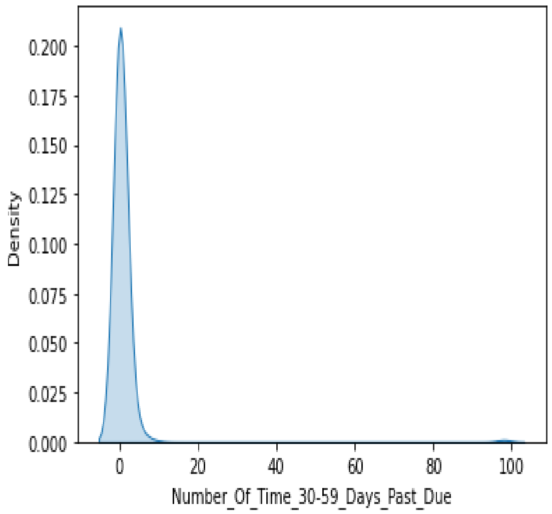


After Removing NaN Values

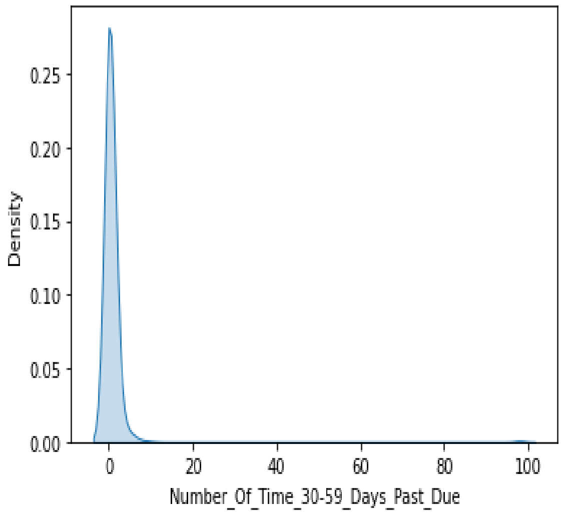


4. 30-59_Days_Due Distribution Curve

Before Removing NaN Values

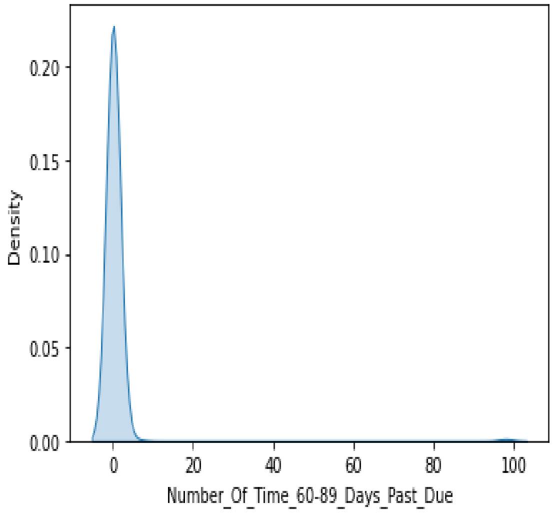


After Removing NaN Values

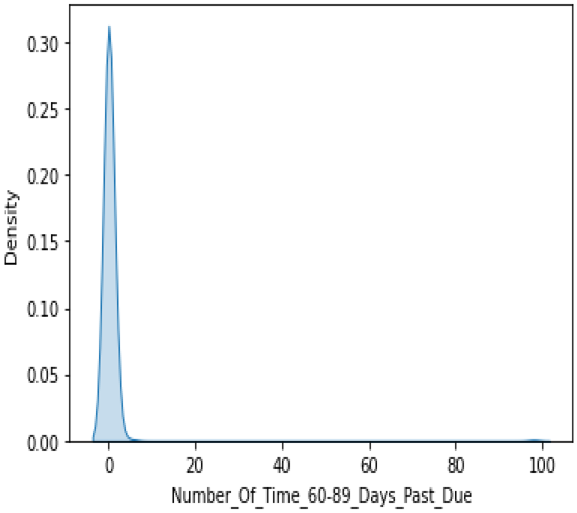


5. 60-89_Days_Due Distribution Curve

Before Removing NaN Values

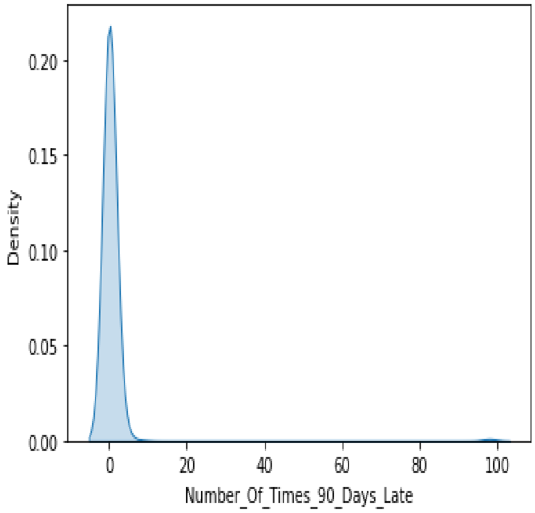


After Removing NaN Values

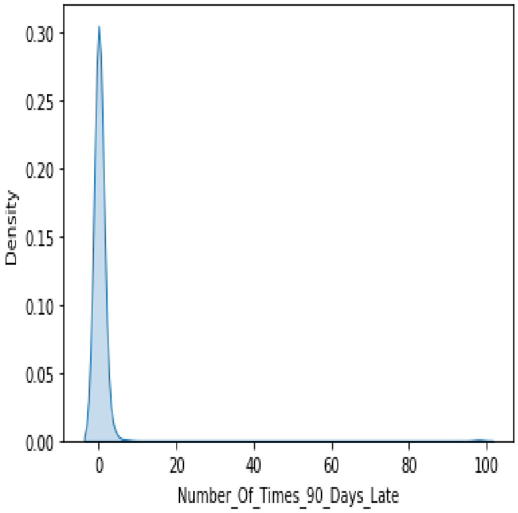


6. 90_Days_Due Distribution Curve

Before Removing NaN Values

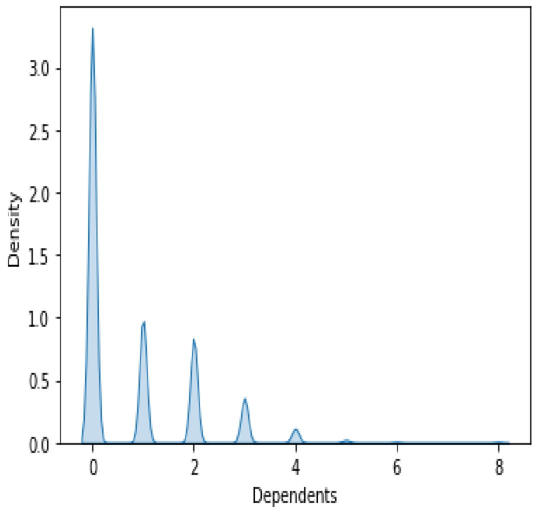


After Removing NaN Values

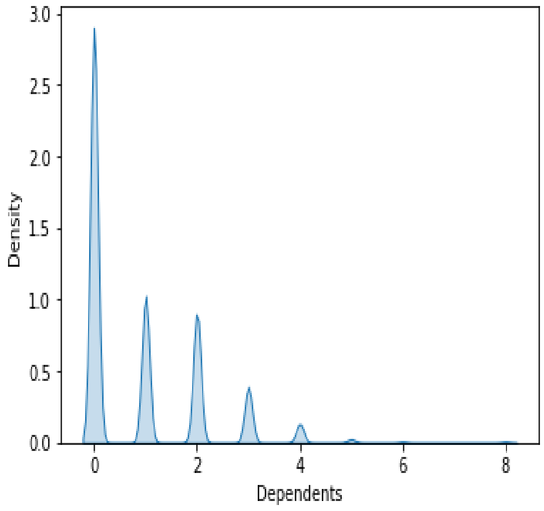


7. Dependents Distribution Curve

Before Removing NaN Values



After Removing NaN Values



Stratified K-Fold Split

Unlike the classic method of Train_Test_Split where we divide the Training and Test portions according to mentioned split ratio and random state parameter, Stratified K-Fold Split allows the user to prevent Biasness of Multi-Class Dataset, as it constitute one more parameter, i.e number of different Folds/Combinations for train-test splits. The Model/Estimator having this type of Training-Test Split, have more varieties/options to try randomized input, which will lead to lesser Biasness towards a particular Input Class, amongst other classes.

Grid-Search Cross Validation

This Type of Cross-Validation is used, when Family of Models, with different combinations of Parameters, come into picture. Instead of having Normal Cross Validation Method, where the performance of Models would be generated iteratively, Grid-Search Cross Validation first constructs Grid of Different Combinations of Parameters (enclosed in dictionary mapping), Cross-Validate their scores internally and then gives the best model amongst family, with better combination of parameters, but it takes long time for bigger number of parametric combinations.

Parameters for each type of Model

For Random Forest, 3 types of parameters are taken, as follows :-

1. Max Depth :- [2,3,4,5]

2. Max Features :- [1,2,4]

3. Min Sample Leaves :- [3,5,7,9]

For Bagging Classifier, Number of Estimators :- [2,3,4] , was the parameter for varying the details of the model.

Best Parameter Combination picked

1. Random Forest :-

   According to Grid-Search Cross Validation, Random Forest with following parameters, was best estimator:-

   1. Max Depth = 2

   2. Max Features = 1

   3. Min Sample Leaves = 7

2. Bagging Classifier (with Decision Tree Classifier as base estimator) :-

   According to Grid-Search Cross Validation, Bagging Classifier with Number of Decision Trees = 3 , was the best estimator.

Cross Validation Details of both models

Cross Validation Details, given by Random Forest and Bagging Classifier, were as follows :-

Model Type	Mean CV Score (Grid Search Cross Validation)	Standard Deviation of CV Scores (Grid Search Cross Validation)	Mean CV Score (Normal Cross Validation)	Standard Deviation of CV Scores (Normal Cross Validation)
Random Forest	0.809	0.043	0.775	0.031
Bagging Classifier	0.731	0.036	0.460	0.154

Accuracy metrics of both models

Regarding Testing Portion, the details given by Best Models from Random Forests and Bagging Classifiers, were as follows :-

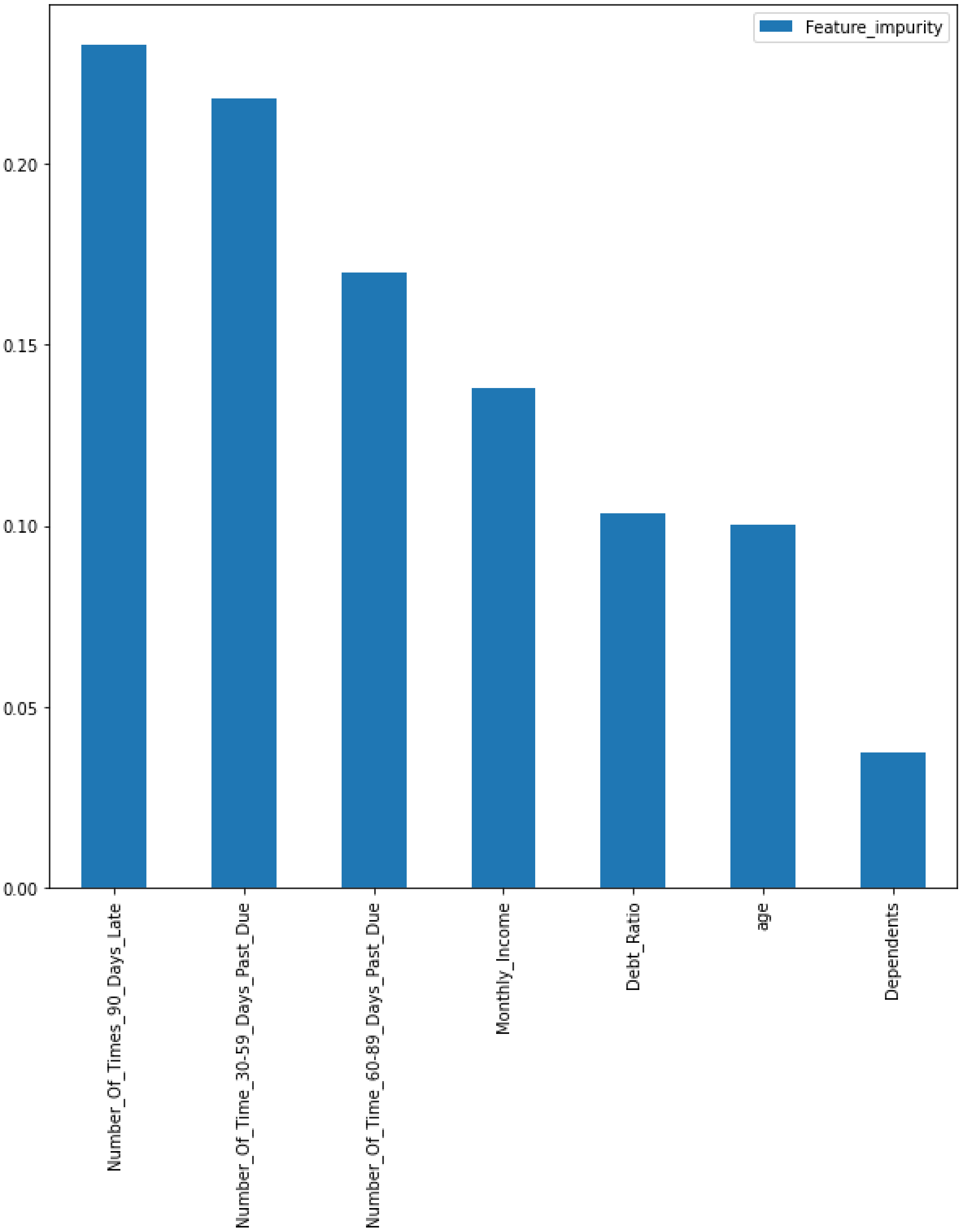
Model Type	Test Accuracy (Grid-Search Cross Validation)	Test Accuracy (Normal Cross Validation)
Random Forest	77.16%	77.77%
Bagging Classifier	70.98%	73.45%

According to the above details, Random Forest performed better than Bagging Classifier.

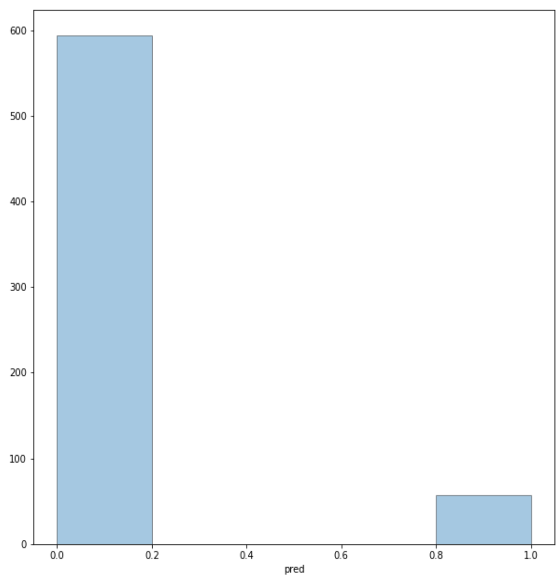
Conclusion

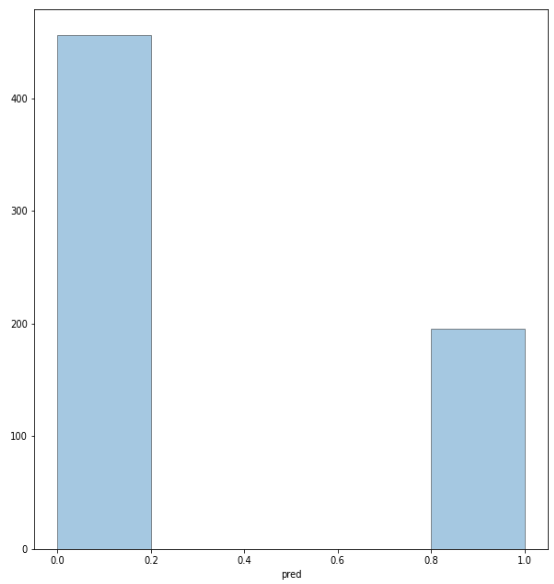
From the above tables, we can see that Cross-Validation Details and Test Details differ , as we differ the method of Cross Validation. Because the Normal Cross Validation method deals with only a single model, whereas the Grid-Search Cross Validation method deals with a family of models, alongside the type of Splitting (either Train_Test_Split or Stratified K-Fold Split), that’s why details vary.

Weakest Feature amongst Input Dataset Features, was “Dependents”, with Feature Impurity = 0.037. Lesser the value of Feature Impurity, weaker the Feature will be, more chances of misclassifications that feature will lead and lesser the extent of randomization that feature will tend to, in respect to classification model/estimator . The Magnitude of Feature Impurities of each feature can be seen as follows :-

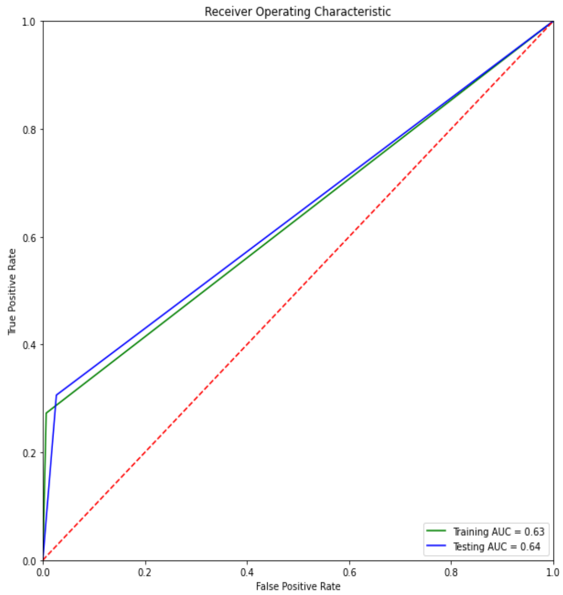

Feature_impurity
0.00
0.05
0.10
0.15
0.20
Number_Of_Times_90_Days_Late
Number_Of_Time_30-59_Days_Past_Due
Number_Of_Time_60-89_Days_Past_Due
Monthly_Income
Debt_Ratio
age
Dependents

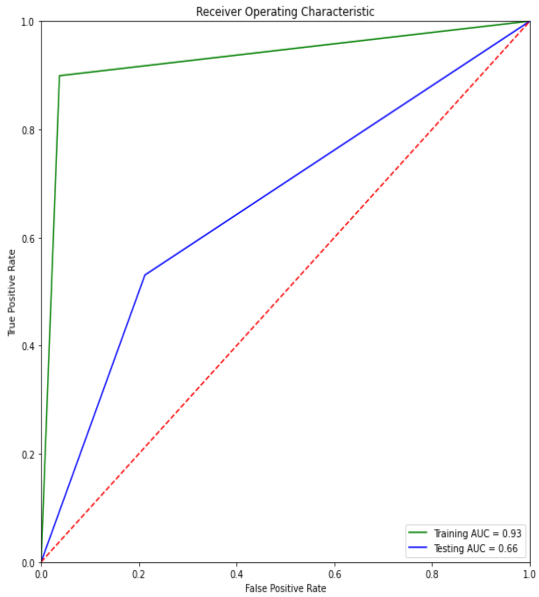
Test Predictions of each Model


Random Forest Test Predictions


Bagging Classifier Test Predictions

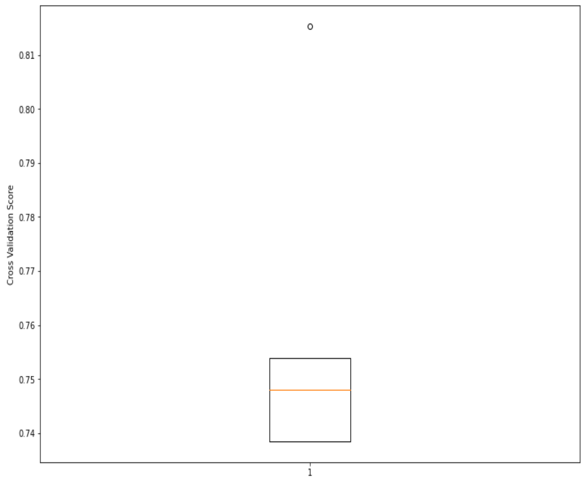
ROC-AUC Curves of each Model


Random Forest ROC-AUC Curve

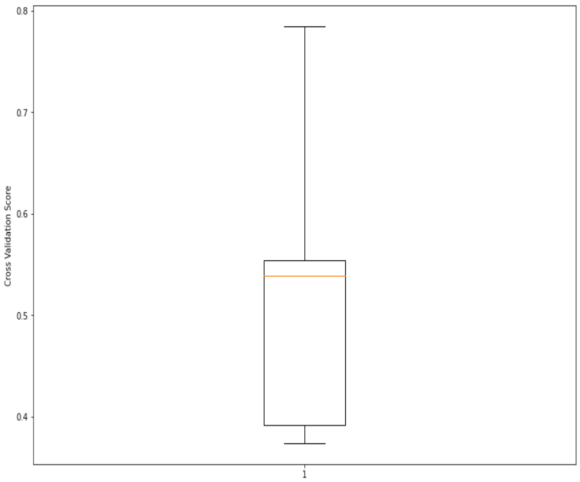

Bagging Classifier ROC-AUC Curve

Box-Plots of each Model

Random Forest Box-Plot



Bagging Classifier Box-Plot


Cross Validation Score