

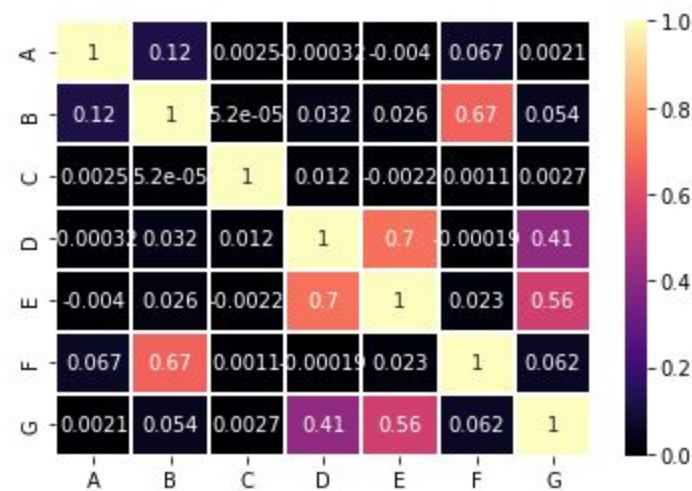
# DMG Assignment 2 Report

Navneet Agarwal  
2018348

Aditya Singh  
2018378

## Methodology: (Approach and Reasons)

- There was an imbalance in the data set; the number of instances of class 1 was 69.3 times more than the number of instances for class 0. We handled this problem using `RandomUnderSampler()` with an undersampling strategy as 'majority'. This function takes the original dataset and returns the undersampled data, where the number of instances for both classes become equal. Under the majority started the samples of minority class are not disturbed
- There were no missing values in the dataset.
- For the feature subselection, we used the data correlation matrix and we found that some of the features had correlation more than 0.5. We tried removing one of the features for the pair of features that had high correlation, but that decreased the `roc_auc` score. We also found that the feature needs to be removed if a pair of attributes had correlation more than 0.9, but we had no such pair.



Correlation matrix for the training dataset

- Decision Trees and the ensemble methods based on decision trees does not require feature scaling because they are not affected by the variance in the data.

- To make sure that the model was not overfitting, we ran 5 fold cross-validation with roc\_auc as the scoring metric. The undersample Data is split into 5 different datasets in this technique, and for each fold, we use 4 of these datasets for training and 1 for testing purpose. Since the class imbalance problem was handled by using RandomUnderSampler(), thus there were an equal number of instances for both the classes while splitting them.
  - We used Decision Trees and KNN at first without tuning any parameters for the classification, and Decision Trees gave better results. Even after tuning the KNN classifier was no match to the Decision Trees.
  - For Hyperparameter tuning, we used GridSearch(). This function takes the machine learning model, input dataset and values of different parameters to be tested. It runs k fold cross-validation and returns the training and testing score for every possible combination of parameters. We also defined the roc\_auc score to be used as the metric for scores.
  - After the grid search has been done, we extracted the best parameters for a particular model and then predicted the labels for testing data.
  - We have uploaded both continuous probability values and discrete values for “T” in the output files. For this, the best auc\_score was given by GradientBoostClassifier for 2 cases and 1 submission uses RandomForestClassifier. So, the discrete values are directly predicted from the predict() function of this model while the probabilities for class X are predicted using the predict\_proba() function of this model.
- 

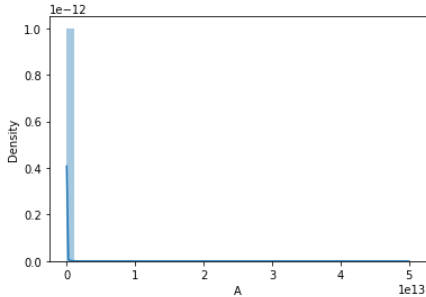
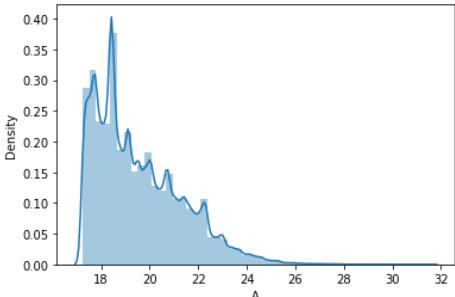
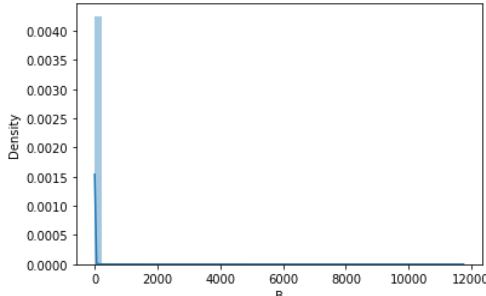
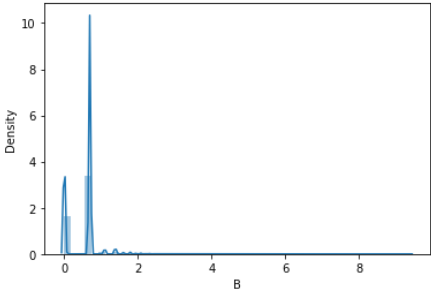
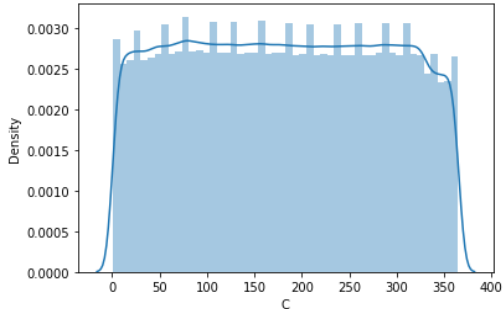
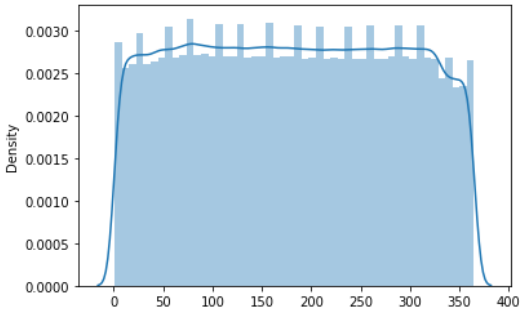
## Skewness

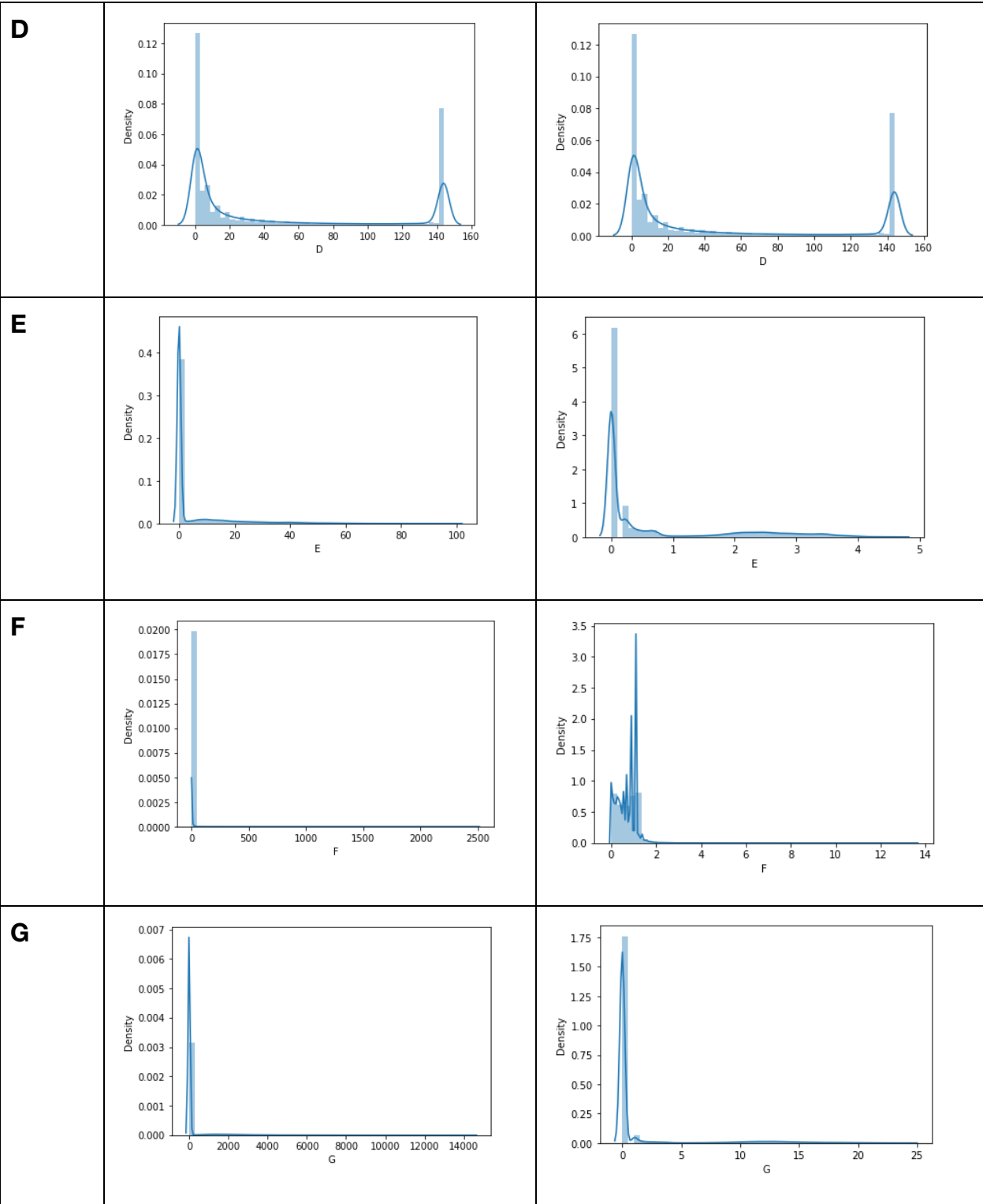
### Value for Skew for each before preprocessing

A : 235.199118 , B : 255.046242 , C : 0.010613 , D : 0.924002 , E : 2.928051 ,  
F : 247.191470, G : 5.527271

### Value for Skew for each after preprocessing

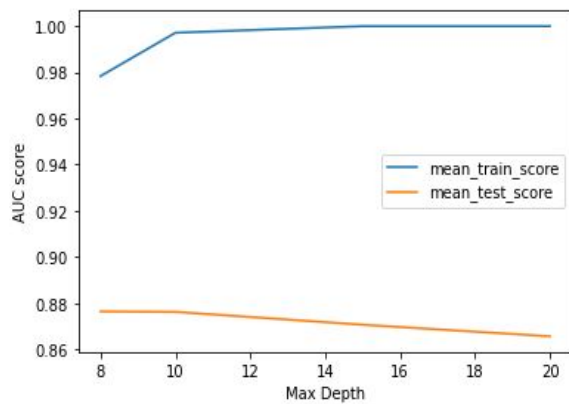
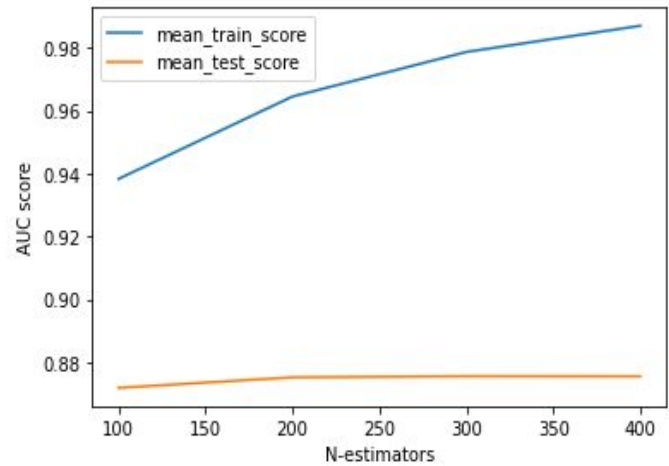
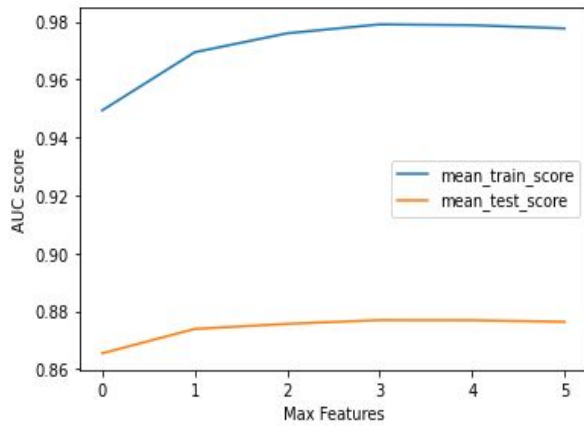
A : 0.922623 , B : 2.210409 , C : 0.010613 , D : 0.924002 , E : 1.463518 , F : 0.803372 ,  
G : 3.156695

Feature	Plots before removing Skewness	Plots before removing Skewness
A		
B		
C		

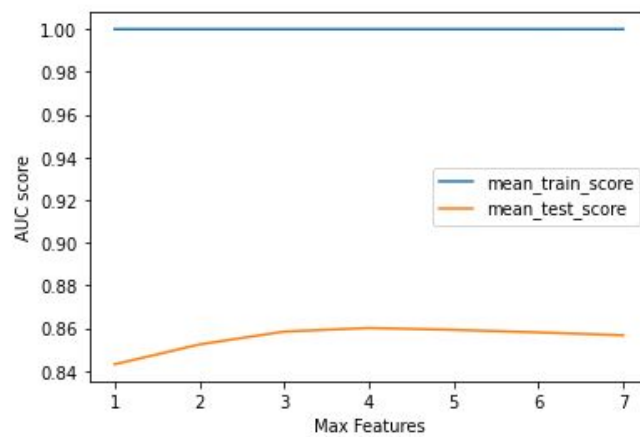
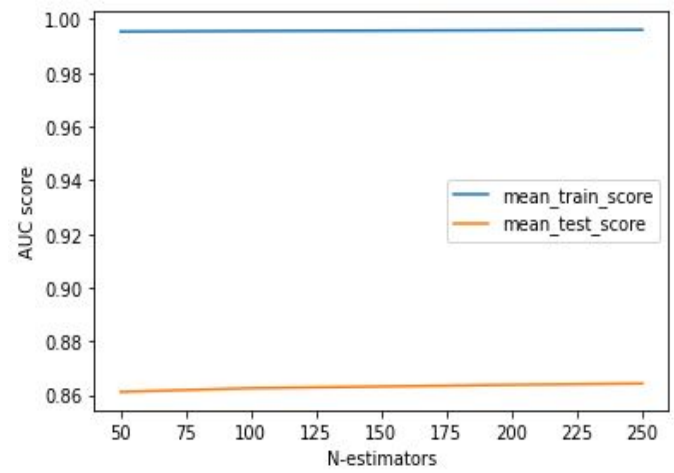
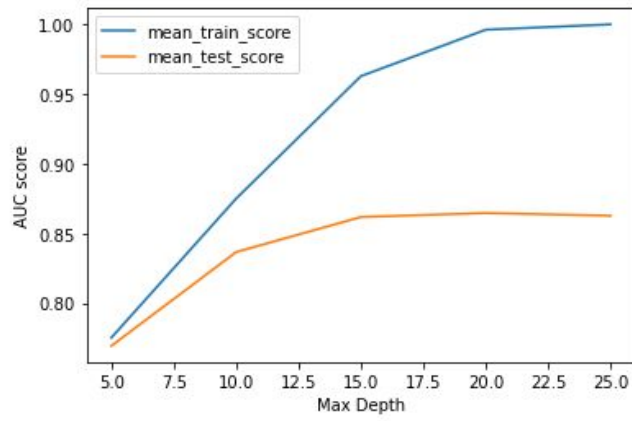


## Plot Training and Testing accuracy w.r.t. hyperparameters of the model

### For GradientBoostClassifier



## For Random Forest



## Learnings From Assignment

- The dataset was highly imbalanced, so when we predicted the results using all the dataset, then the AUC score was close to 0.50 and when we analysed the predictions, most of them were 1. This suggests a model using the imbalance datasets does not perform well while predicting the minority class, thus we need to tackle this by some technique. We learned to use `RandomUnderSampler()` to remove the imbalance in the dataset. We also learnt the use of `RandomOverSampler()`, which increases the instances of minority class by creating copies.
- We learned how we could use the correlation matrix for the feature sub-selection. The correlation values tell how closely two variables are to have a linear relationship with each other. If this value is very high for two features, then we should consider removing one of them.
- We learned how we could use `GridSearch` to train the parameters of the model and extract the best parameters of a model for a given data set.
- We also got to know the usage of K-fold cross-validation to predict the `auc_scores` locally and prevent overfitting of the out model.
- The concept of `ROC_curve()` became much more clear after doing this assignment, and we also understood the reason why continuous values give a better AUC score than the discrete values for the same model.
- We also learnt how to apply different classification models and ensemble models using the sklearn library like the `GradientBoostClassifier`, `RandomForestClassifier`, `DecisionTreesClassifier` etc.

## Link to models and CSV files

[https://drive.google.com/drive/folders/1daK--GgGiwfeJZdWJWOxfbWV5zb28y\\_B?usp=sharing](https://drive.google.com/drive/folders/1daK--GgGiwfeJZdWJWOxfbWV5zb28y_B?usp=sharing)