Capstone Project 3

Credit Card Default Prediction

Team Members

- Jasjot Kaur
- Venkatesh Ambore
- Parul Saini

Flow of Presentation

- Background Information
- Project Objective
- Data Summary
- Exploratory Data Analysis
- Data Preprocessing
- Classification Models
- Conclusion

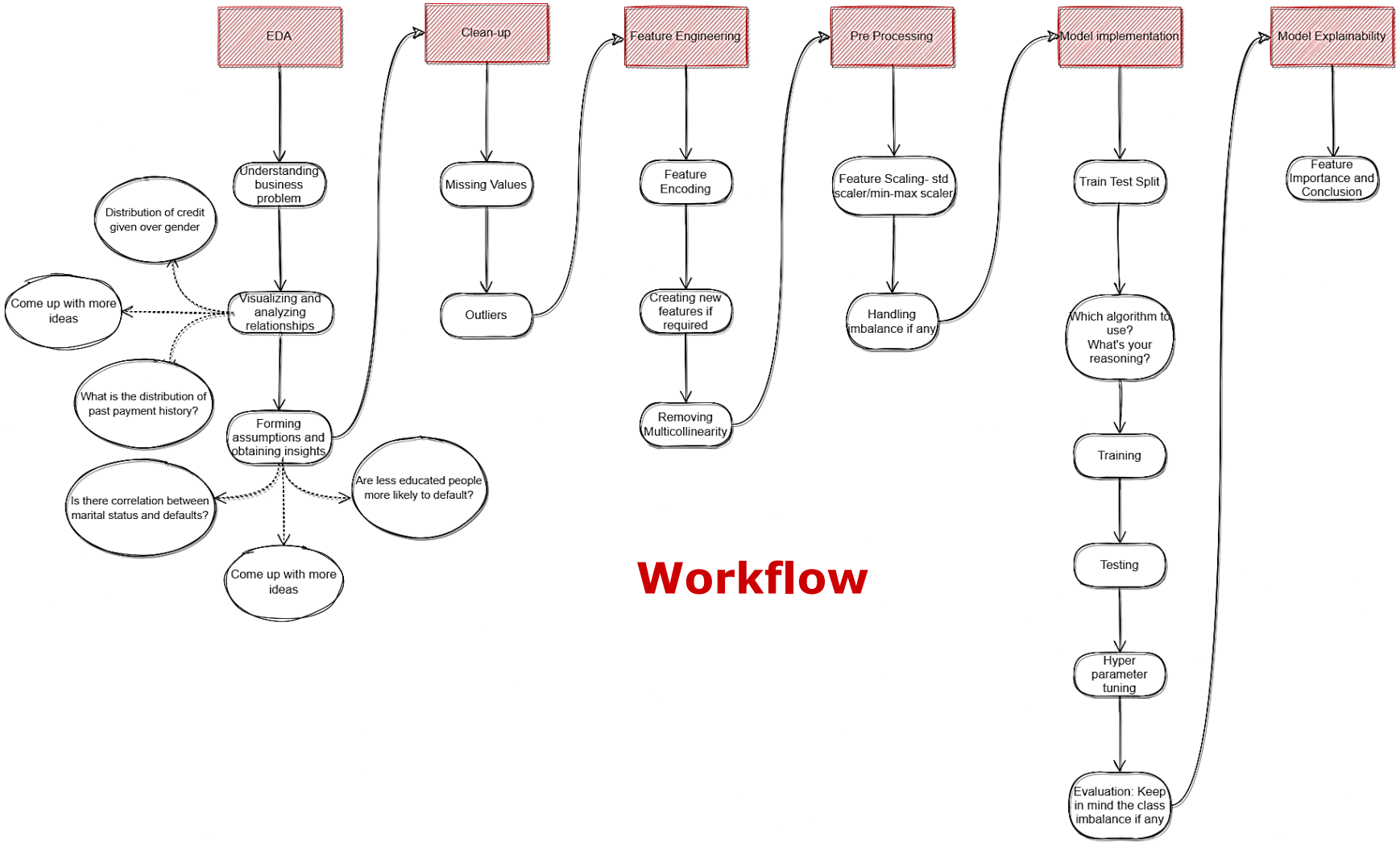
Background

- According to the Federal Reserve economic data, the default rate on credit loans across all commercial banks is at an all-time high, observed since 2016.
- With the rapid increase in delinquencies, the lending institutions, such as commercial banks, lose a significant amount of money.
- Therefore, banks must have a risk prediction model and be able to classify the most relative characteristics that are indicative of people who have a higher probability of default on credit.

Project Objective

- The objective of this project is **to predict and identify the customers who are most likely to default in the coming months.**
- From the perspective of risk management, the result of predictive accuracy of the estimated probability of default will be more valuable than the binary result of classification - credible or not credible clients.

- So, in this project:
  - identify the customers who might default by developing a model to predict the credit card defaulters well in advance
  - identify the potential customer base that can be offered various credit instruments so as to encounter least defaults.


EDA
Understanding business problem
Distribution of credit given over gender
Come up with more ideas
Visualizing and analyzing relationships
What is the distribution of past payment history?
Forming assumptions and obtaining insights
Is there correlation between marital status and defaults?
Are less educated people more likely to default?
Come up with more ideas
Clean-up
Missing Values
Outliers
Feature Engineering
Feature Encoding
Creating new features if required
Removing Multicollinearity
Pre Processing
Feature Scaling- std scaler/min-max scaler
Handling imbalance if any
Model implementation
Train Test Split
Which algorithm to use? What's your reasoning?
Training
Testing
Hyper parameter tuning
Evaluation: Keep in mind the class imbalance if any
Model Explainability
Feature Importance and Conclusion
Workflow

Workflow

- Null values
- Duplicated values
- Outliers
- Categorical data

- Answering hypothesized question
- Analyzing data visually and finding useful information and patterns

- Finding correlation
- Statistical modelling for feature selection
- Finding most important features

- Logistic Regression
- Decision tree
- Random forest
- KNN

- Confusion Matrix
- Precision
- Recall
- F1 Score

Data Summary

Our dataset contains 3000 rows and 25 columns. The description of the columns are as follows:

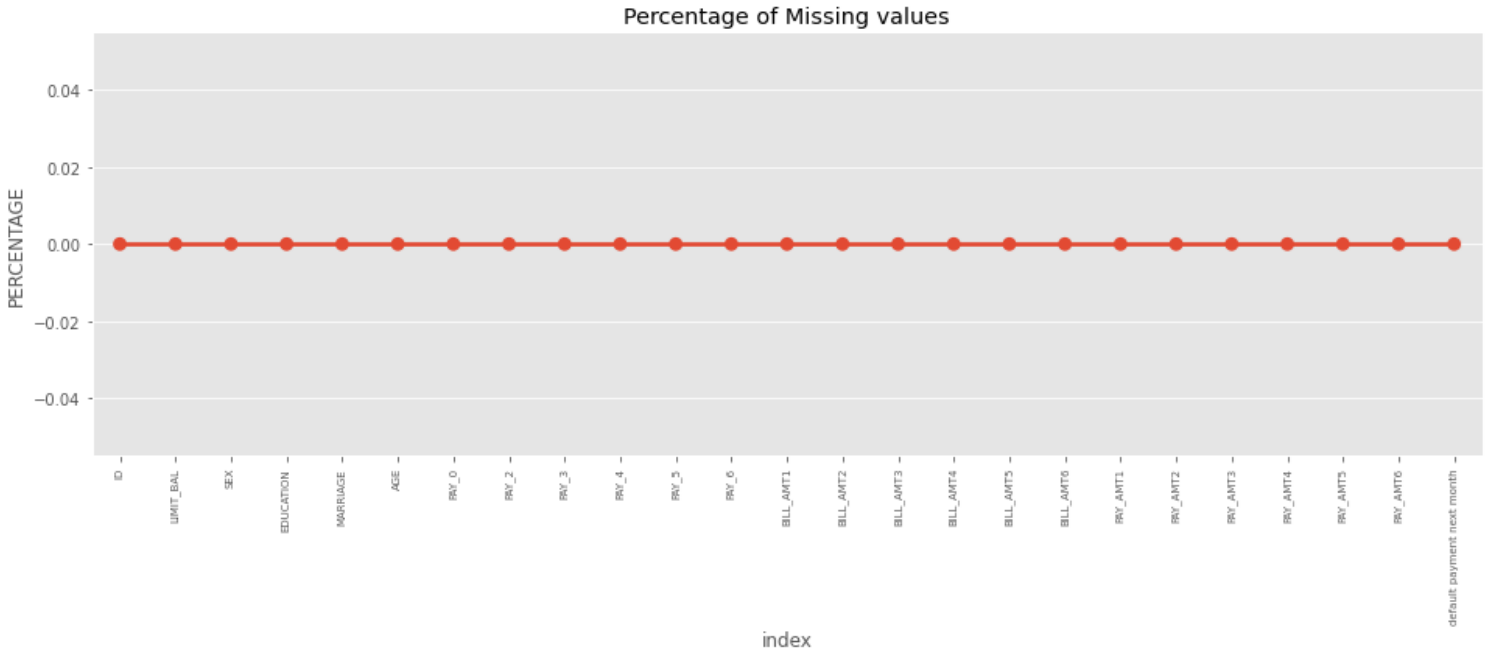
- **ID: ID of each client**
- **credit_limit: Amount of given credit in NT dollars**
- **gender: Gender (1 = male, 2 = female)**
- **education: (1 = graduate school, 2 = university, 3 = high school, 0,4,5,6 = others)**
- **marital_status: Marital status (0 = others, 1 = married, 2 = single, 3 = others)**
- **age: Age in years**
- **Scale for PAY_0 to PAY_6:(-2, -1, 0 = paid duly, 1 = payment delay for one month, 2 = payment delay for two months, ... 8 = payment delay for eight months, 9 = payment delay for nine months and above)**
- **CC_UsageSept: Repayment status in September, 2005 (scale same as above)**
- **CC_UsageAug: Repayment status in August, 2005 (scale same as above)**
- **CC_UsageJul: Repayment status in July, 2005 (scale same as above)**
- **CC_UsageJun: Repayment status in June, 2005 (scale same as above)**
- **CC_UsageMay: Repayment status in May, 2005 (scale same as above)**
- **CC_UsageApr: Repayment status in April, 2005 (scale same as above)**

Data Summary

- invoice_sept: Amount of bill statement in September, 2005 (NT dollar)
- invoice_aug: Amount of bill statement in August, 2005 (NT dollar)
- invoice_jul: Amount of bill statement in July, 2005 (NT dollar)
- invoice_jun: Amount of bill statement in June, 2005 (NT dollar)
- invoice_may: Amount of bill statement in May, 2005 (NT dollar)
- invoice_apr: Amount of bill statement in April, 2005 (NT dollar)
- total_amount_sept: Amount of previous payment in September, 2005 (NT dollar)
- total_amount_aug: Amount of previous payment in August, 2005 (NT dollar)
- total_amount_jul: Amount of previous payment in July, 2005 (NT dollar)
- total_amount_jun: Amount of previous payment in June, 2005 (NT dollar)
- total_amount_may: Amount of previous payment in May, 2005 (NT dollar)
- total_amount_apr: Amount of previous payment in April, 2005 (NT dollar)
- default_payment: Default payment, our target variable (1=yes, 0=no)
- In our dataset we got customer credit card transaction history for past 6 months, on basis of which we have to predict if customer will default or not.**

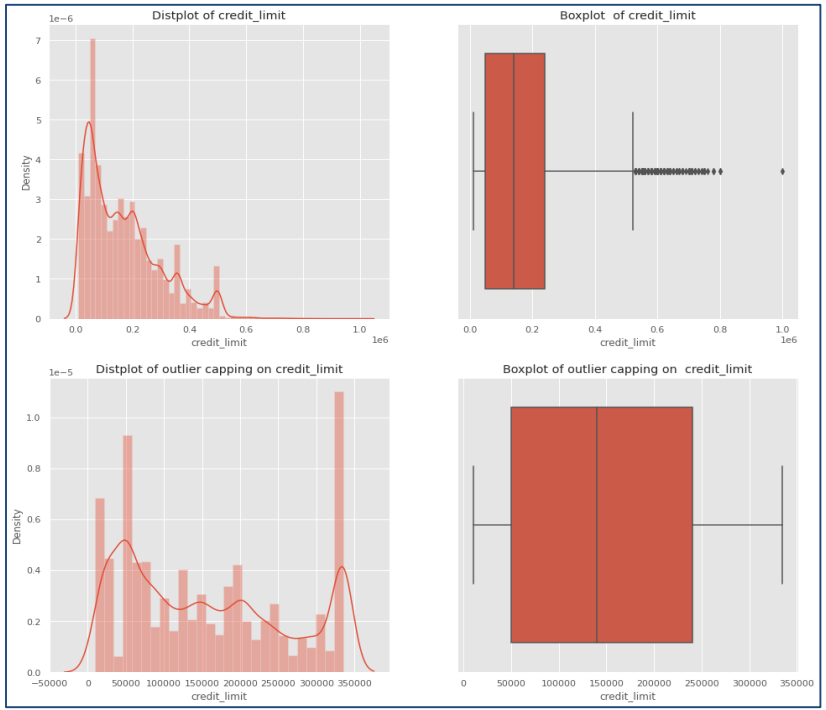
Data Cleaning

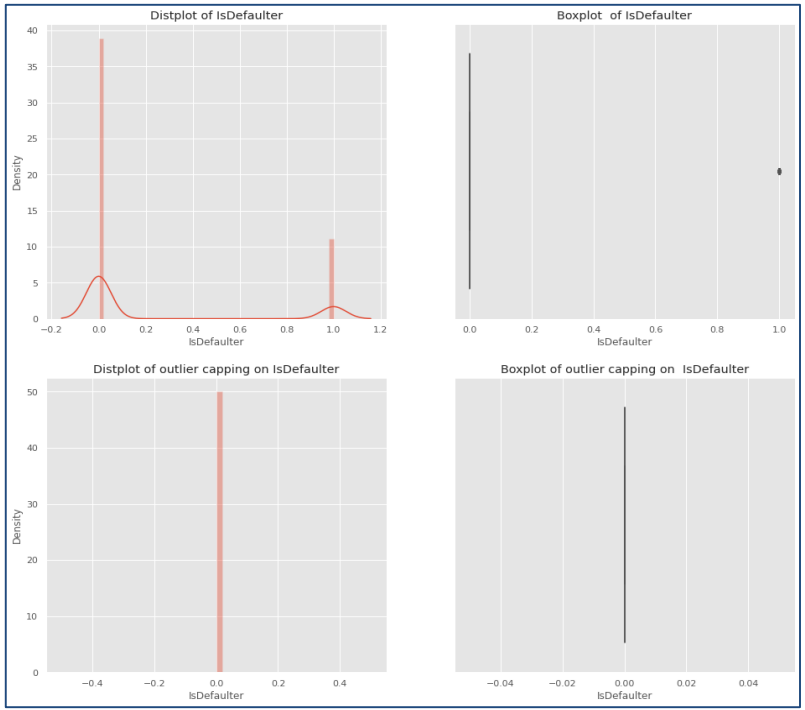
Detecting Null and Duplicate values



- **No null and duplicate values found in the dataset.**

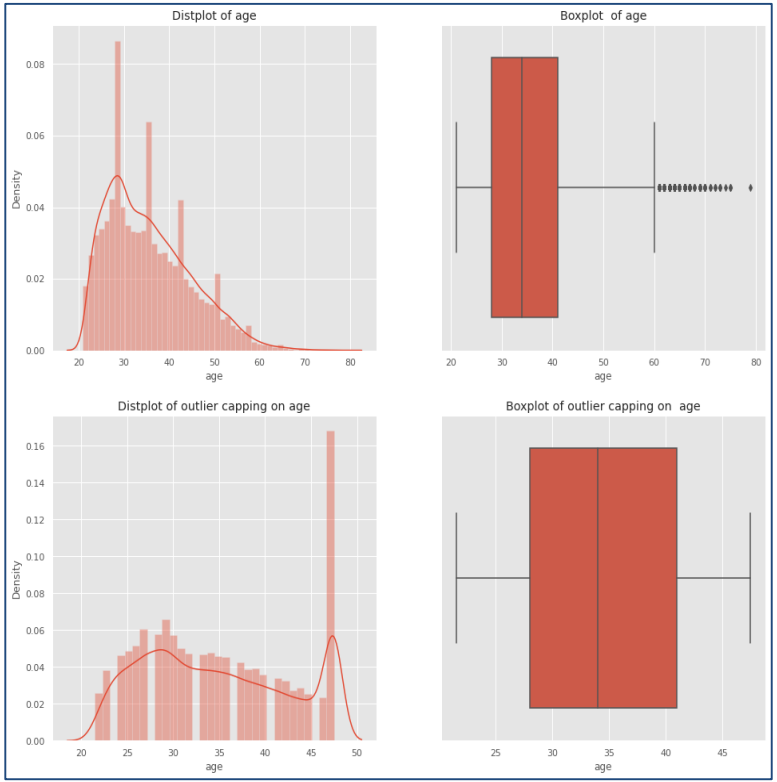
Data Cleaning: Handling Outliers

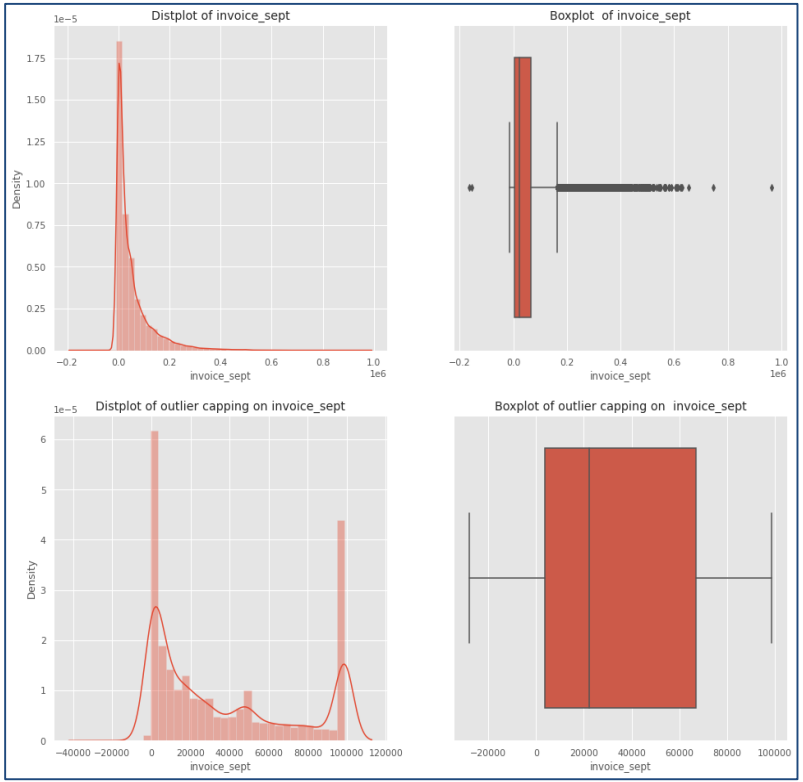


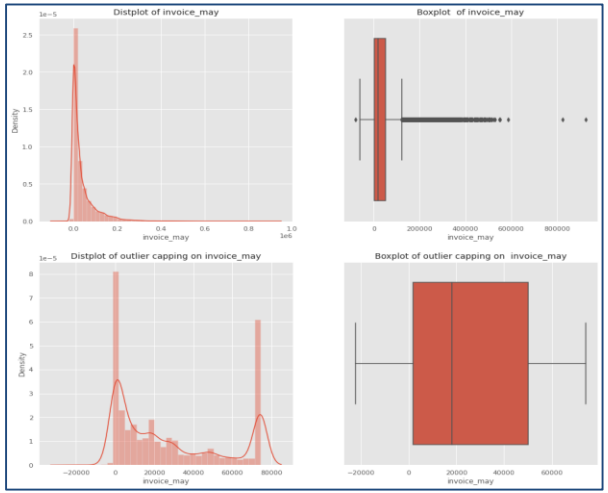


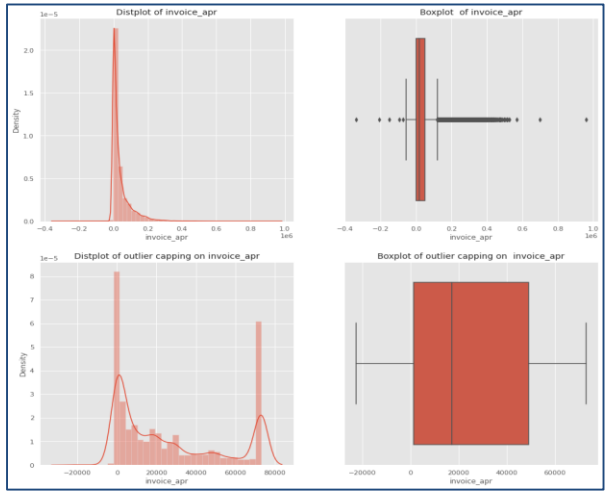
- Outliers were detected using box plot
- IQR method was used to treat outliers. These detected outlier values were replaced with upper and limits accordingly).

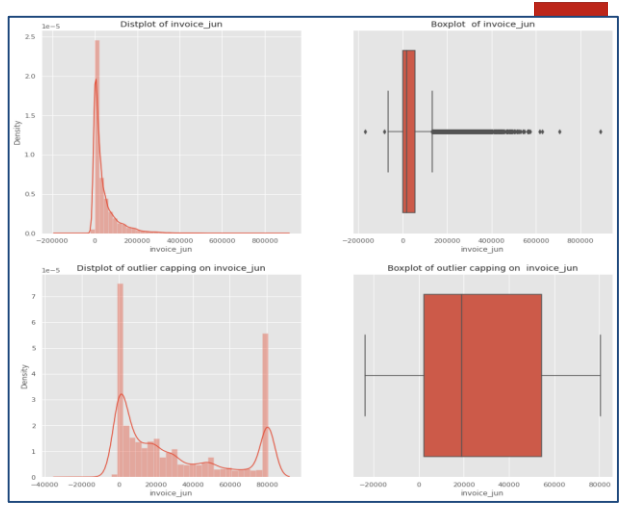
Data Cleaning: Handling Outliers

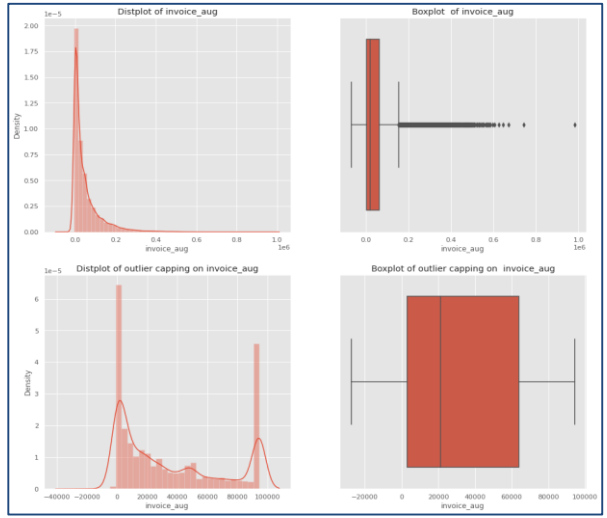


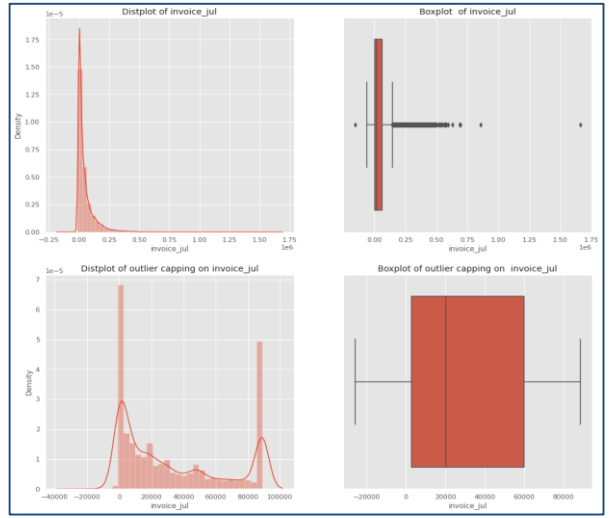


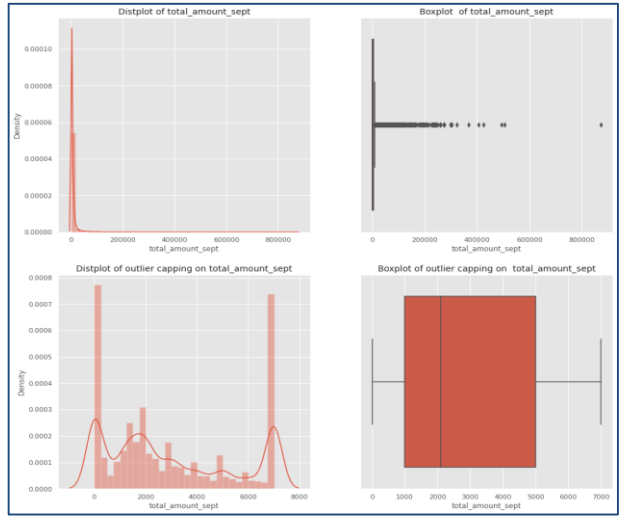

Distplot of invoice_may
Boxplot of invoice_may
Distplot of outlier capping on invoice_may
Boxplot of outlier capping on invoice_may
Density
invoice_may

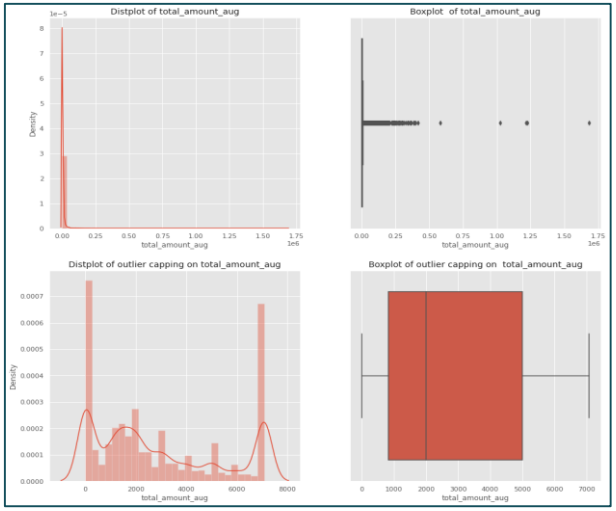

Distplot of invoice_apr
Boxplot of invoice_apr
Distplot of outlier capping on invoice_apr
Boxplot of outlier capping on invoice_apr
Density
invoice_apr

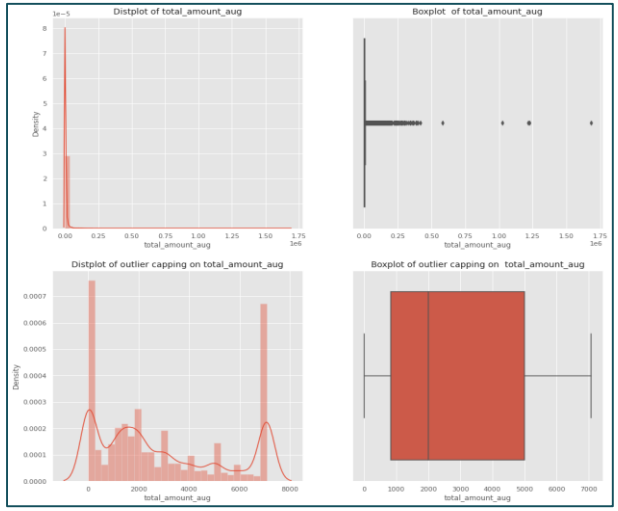

Distplot of invoice_jun
Boxplot of invoice_jun
Distplot of outlier capping on invoice_jun
Boxplot of outlier capping on invoice_jun
Density
invoice_jun

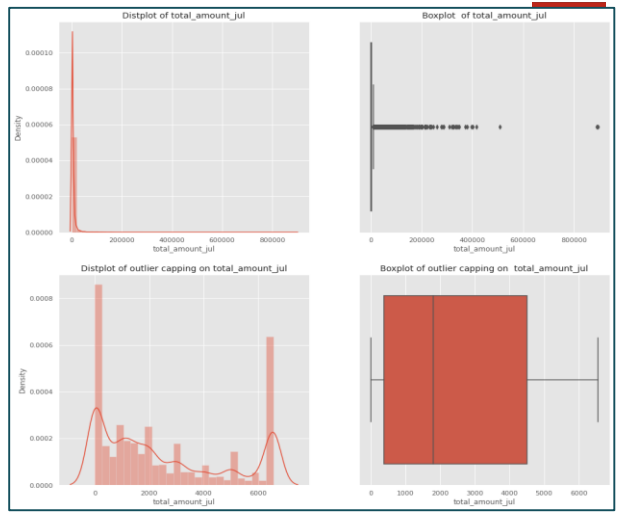

Distplot of invoice_aug
Boxplot of invoice_aug
Distplot of outlier capping on invoice_aug
Boxplot of outlier capping on invoice_aug
Density
invoice_aug

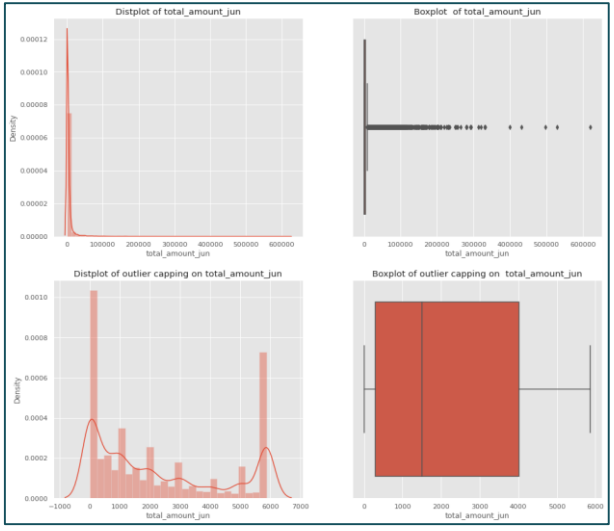

Distplot of invoice_jul
Boxplot of invoice_jul
Distplot of outlier capping on invoice_jul
Boxplot of outlier capping on invoice_jul
Density
invoice_jul

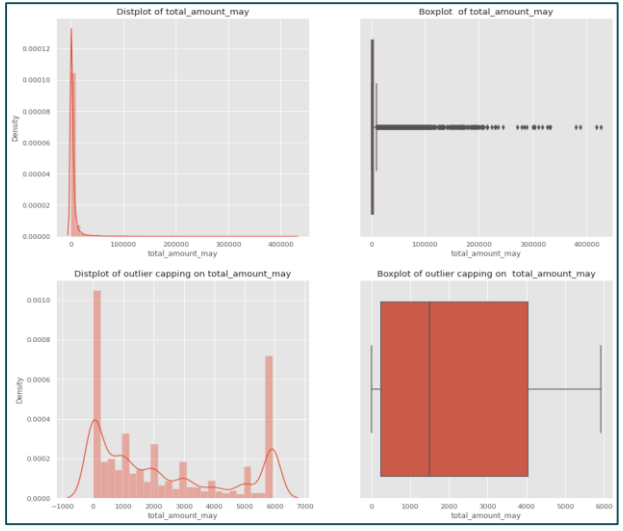

Distplot of total_amount_sept
Boxplot of total_amount_sept
Distplot of outlier capping on total_amount_sept
Boxplot of outlier capping on total_amount_sept
Density
total_amount_sept

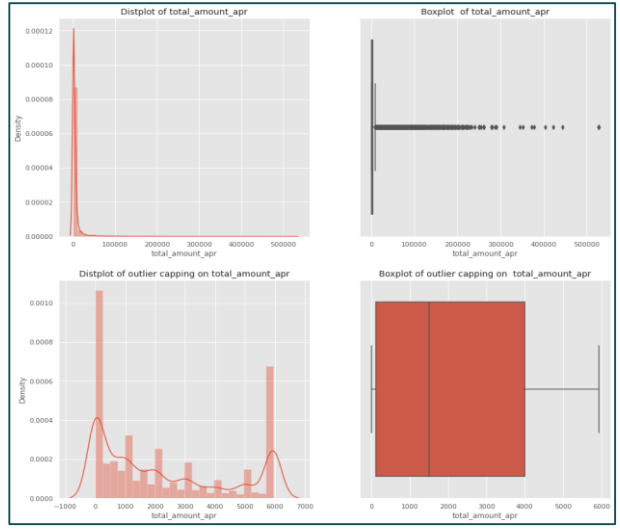

Distplot of total_amount_aug
Boxplot of total_amount_aug
Distplot of outlier capping on total_amount_aug
Boxplot of outlier capping on total_amount_aug
Density
total_amount_aug


Distplot of total_amount_aug
Boxplot of total_amount_aug
Distplot of outlier capping on total_amount_aug
Boxplot of outlier capping on total_amount_aug
Density
total_amount_aug

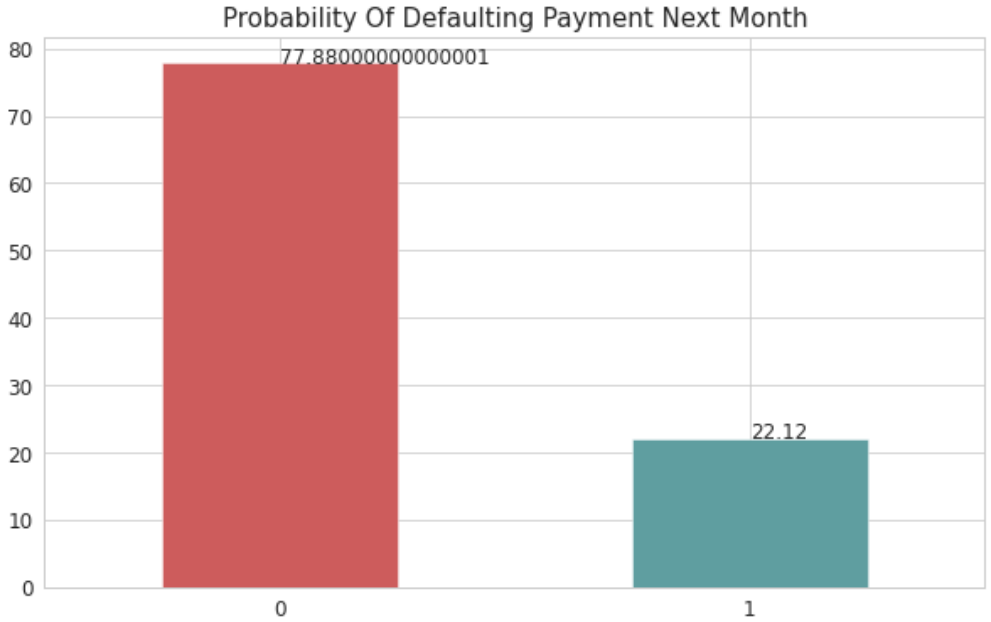

Distplot of total_amount_jul
Boxplot of total_amount_jul
Distplot of outlier capping on total_amount_jul
Boxplot of outlier capping on total_amount_jul
Density
total_amount_jul


Distplot of total_amount_jun
Boxplot of total_amount_jun
Distplot of outlier capping on total_amount_jun
Boxplot of outlier capping on total_amount_jun
Density
total_amount_jun


Distplot of total_amount_may
Boxplot of total_amount_may
Distplot of outlier capping on total_amount_may
Boxplot of outlier capping on total_amount_may
Density
total_amount_may


Distplot of total_amount_apr
Boxplot of total_amount_apr
Distplot of outlier capping on total_amount_apr
Boxplot of outlier capping on total_amount_apr
Density
total_amount_apr

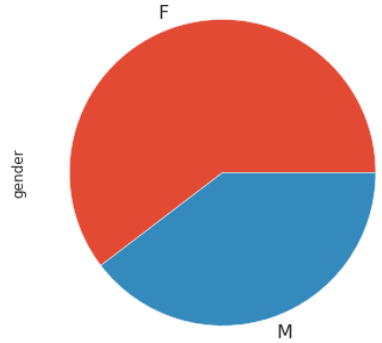
Exploratory Data Analysis

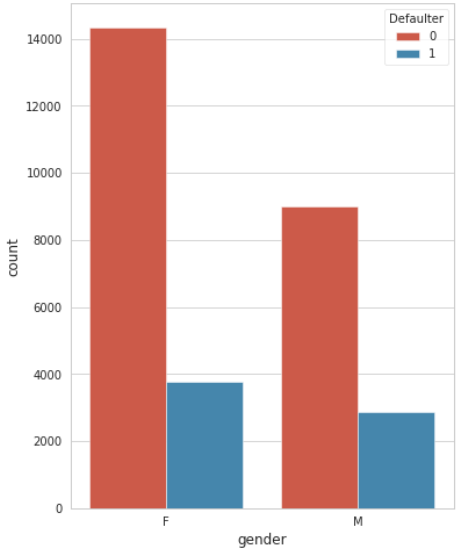


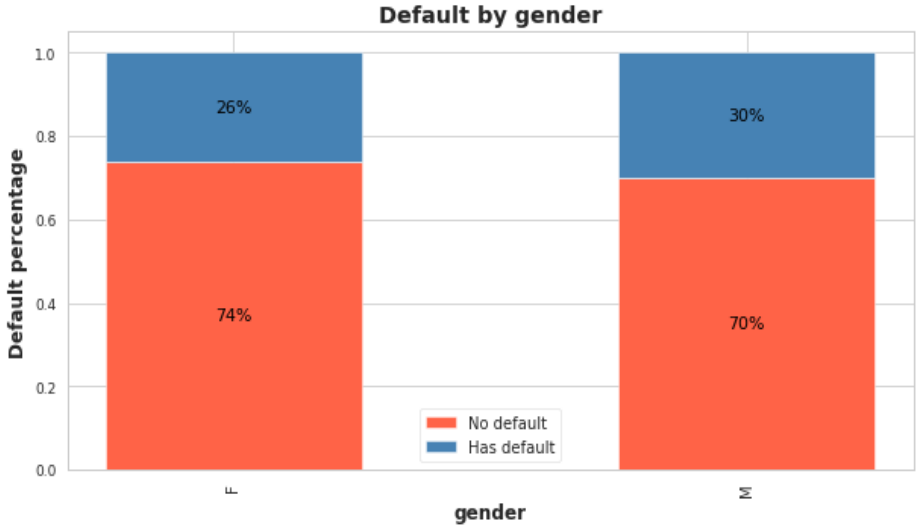
- There is a huge difference between the defaulters and non-defaulters; thus, we need to work on reducing the **class imbalance.**
- Of all the credit card holders, about **22% people are expected to default next month** and 77.8% are not expected to default.

Exploratory Data Analysis

Defaulters according to **gender**



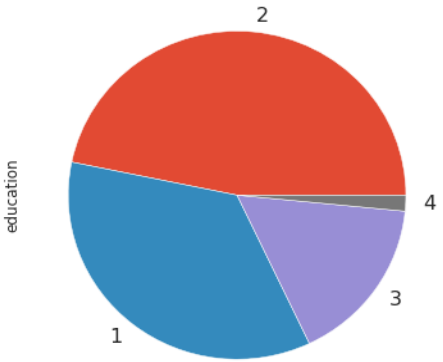


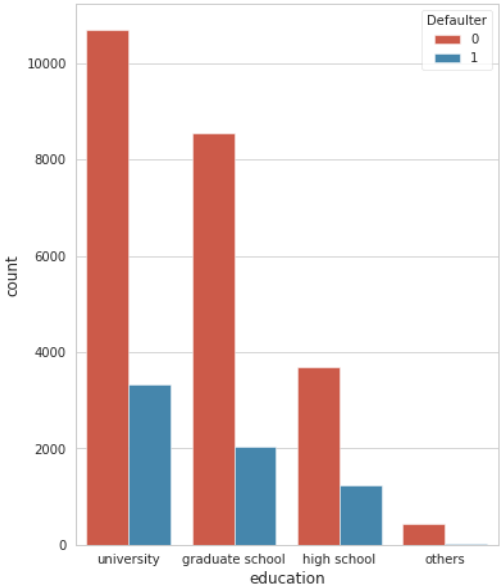


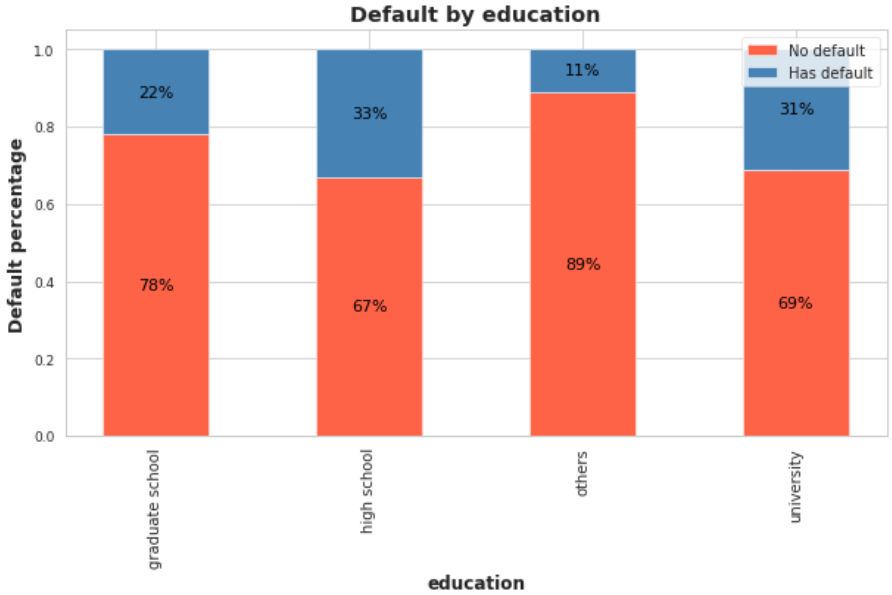
- It can be observed that there are more number of female defaulters than male defaulters;
- Among the female credit card holders, there were 26% who defaulted among the female credit card holders, while there were 30% defaulters of the male credit card holders.
- Hence, this suggests that while there are more number of female defaulters than male defaulters, and the fact that the proportion of defaulters to non-defaulters is 4% more in males than females; it can be concluded that while there are more female credit card holders, male credit card holders have higher tendency to default.

Exploratory Data Analysis

Defaulters according to **education level**



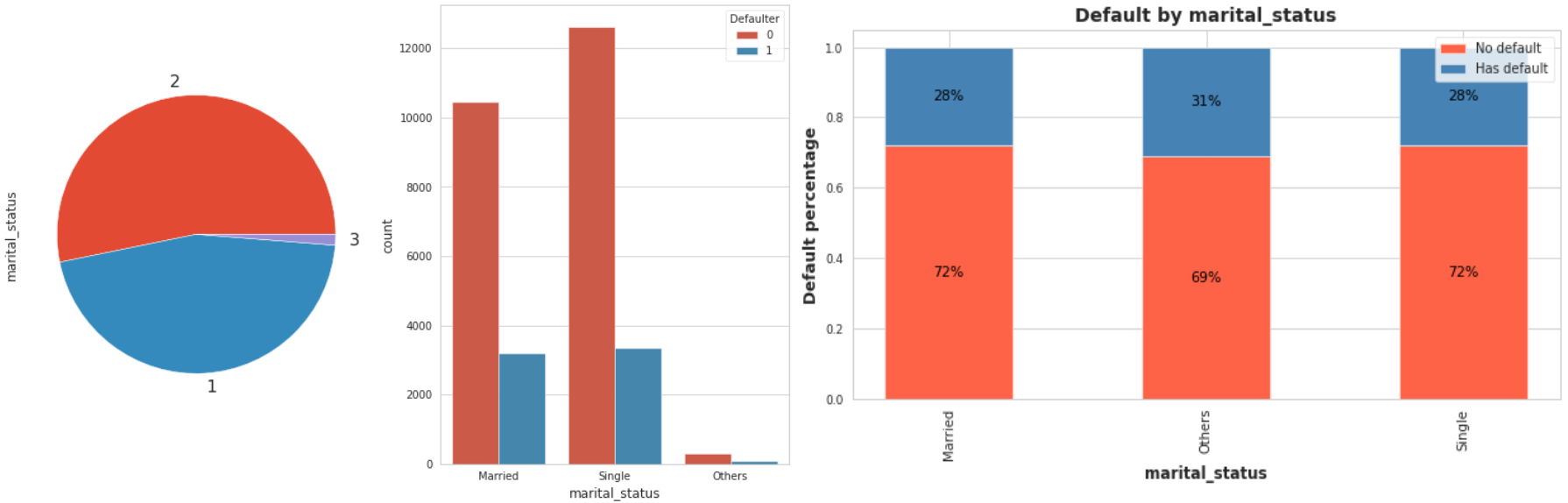




- According to the level of education, it was found that university level has the highest number of credit card holders, followed by graduate school level, then high school students and the least belonged to the ‘others’ category.
- It can also be concluded that customers with lower education levels default more. Customers with high school and university educational level had higher default percentages than customers with grad school education.

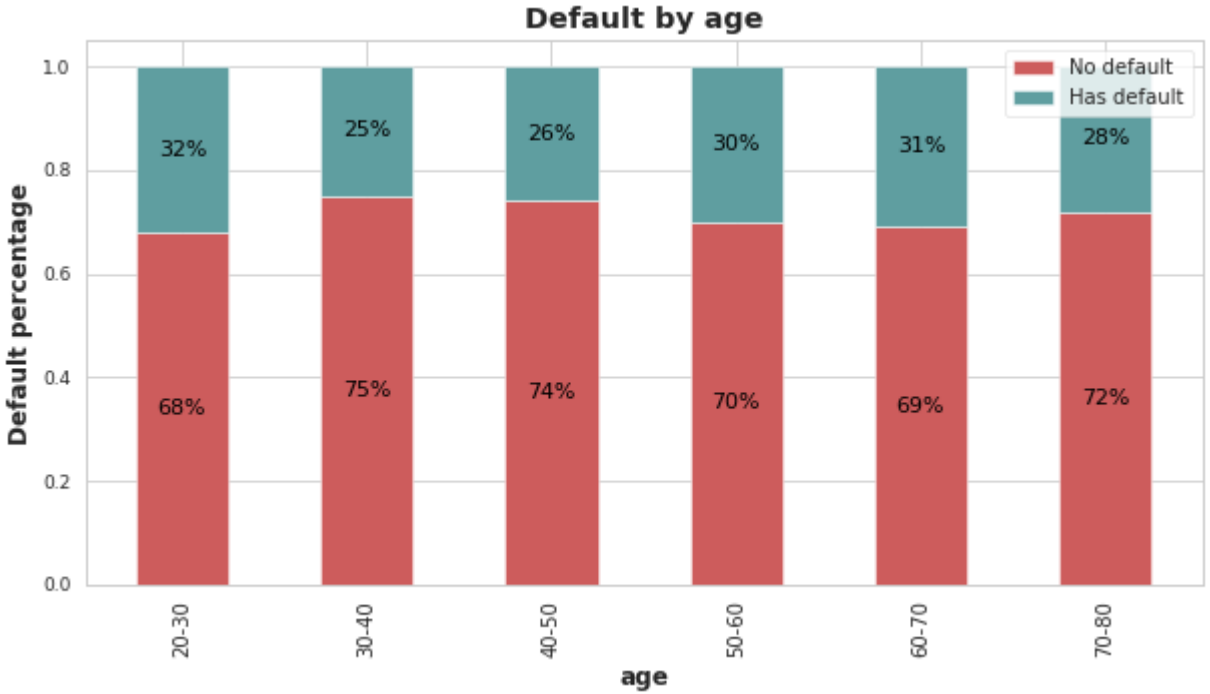
Exploratory Data Analysis

Defaulters according to **marital status**



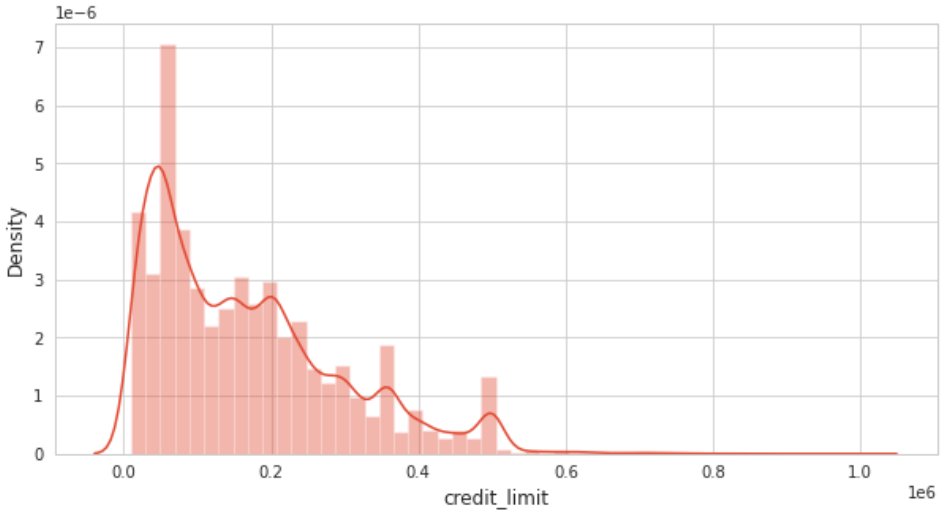
- As we can see in the charts, we can infer that there is quite similar distribution of defaulters in each category, indicating that the marital status does not influence defaults. However, ‘others’ category tend to default the most.

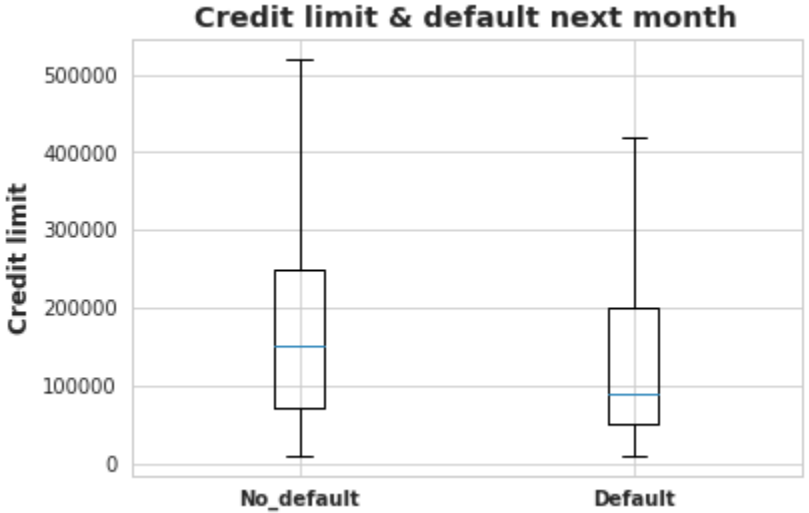
Exploratory Data Analysis



- Credit card holders aged between 30-50 had the lowest delayed payment rate, while younger groups (20-30) and older groups (50-70) all had higher delayed payment rates. However, the delayed rate dropped slightly again for customers older than 70 year.

Exploratory Data Analysis





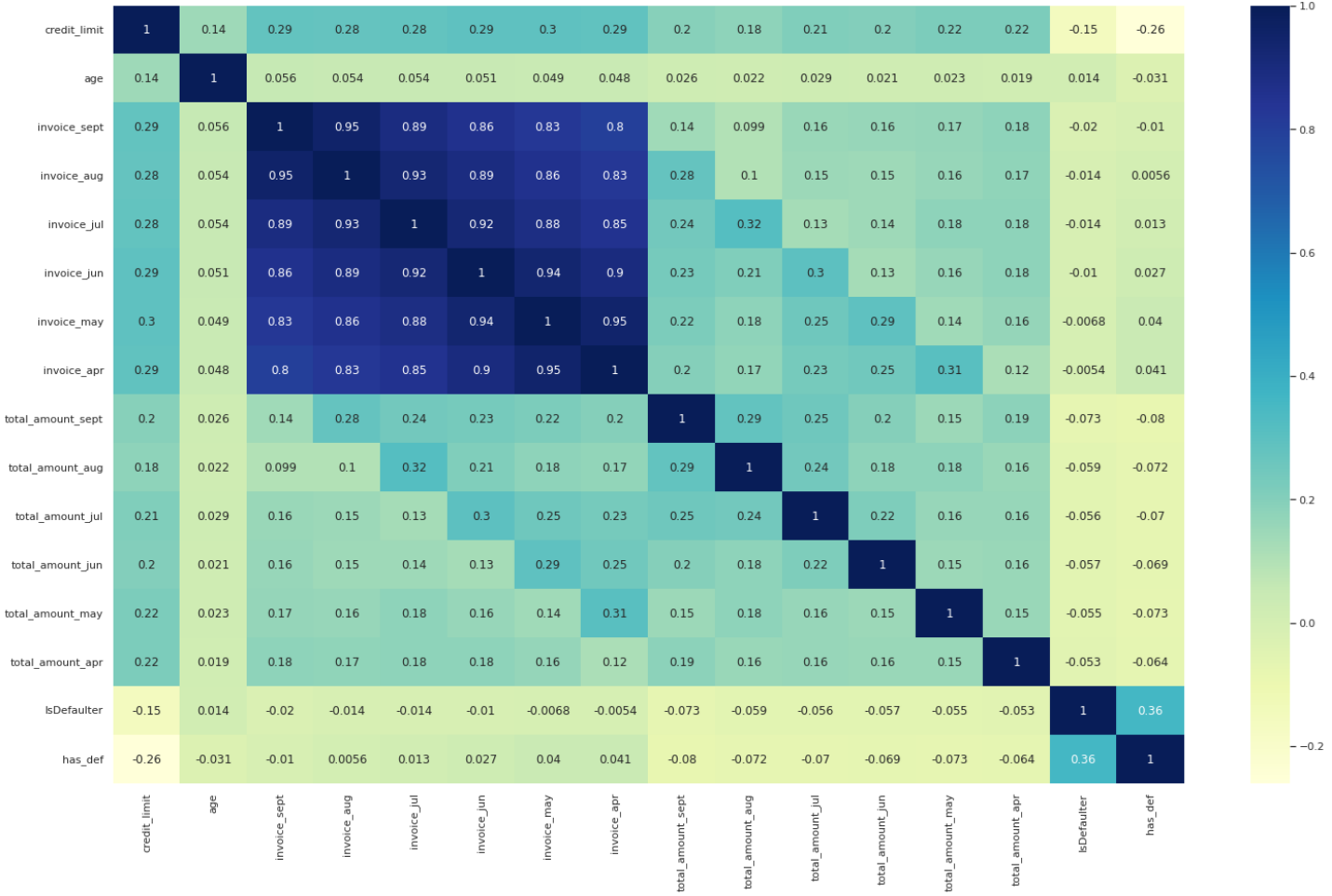
- From the density graph, it can be seen that the credit limit is positively skewed, i.e, the outliers of the distribution curve are further out towards the right and closer to the mean on the left.

- From the box plot, it can be concluded that customers with higher credit limits tend to pay the pay on time and hence, they are not likely to default.

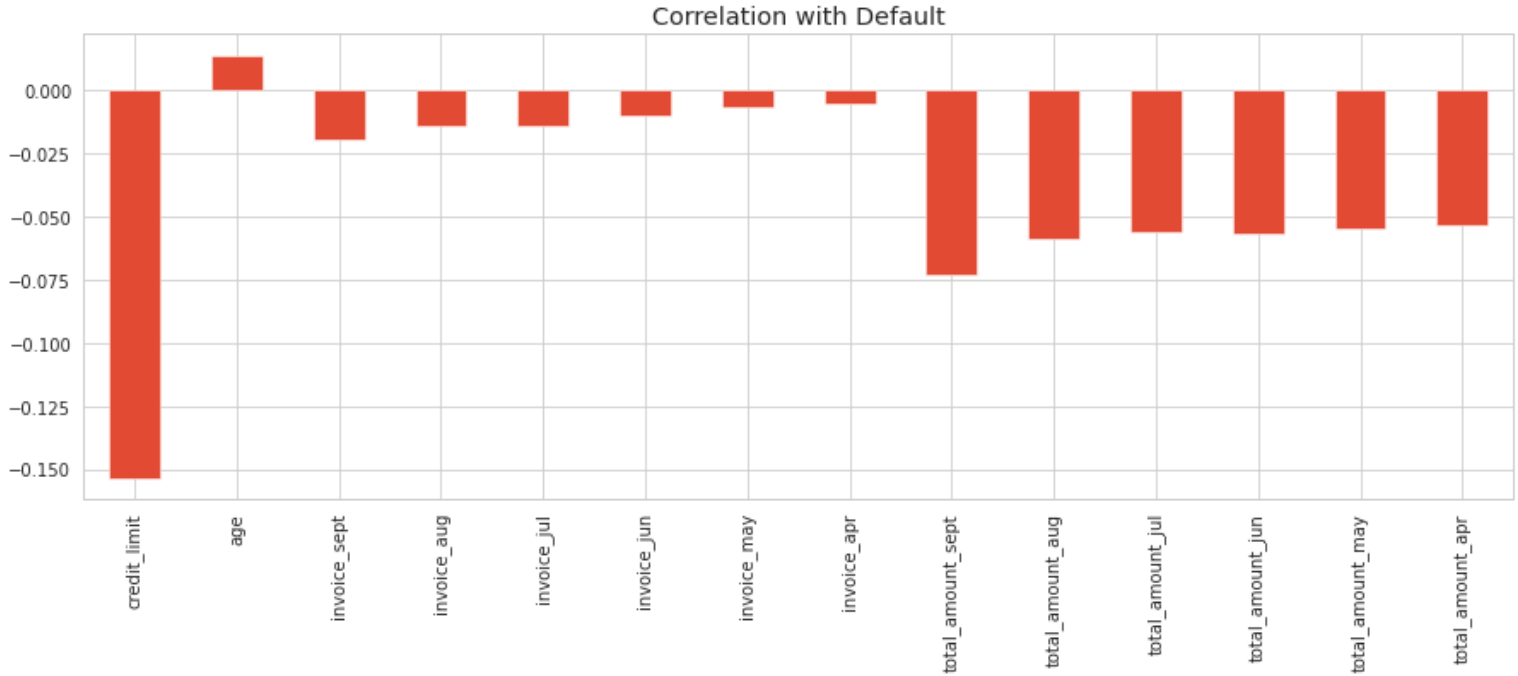
Exploratory Data Analysis

- From the above scatter plots, it can be seen that maximum of data points are closely packed along the Y-axis near to 0 on X-axis.
- Thus it can be deduced that there is a **large bill amount in all the 6 months and lesser payments are made for the same; indicating a high default rate.**

Data Visualization- Heat Map



Data Visualization



- Among all the features only age is positively correlated to default , rest all the features are negatively correlated to default.
- Credit_limit is found to be highly negatively correlated with defaulters.
- Applied stats in relation with different features to authenticate this insight.

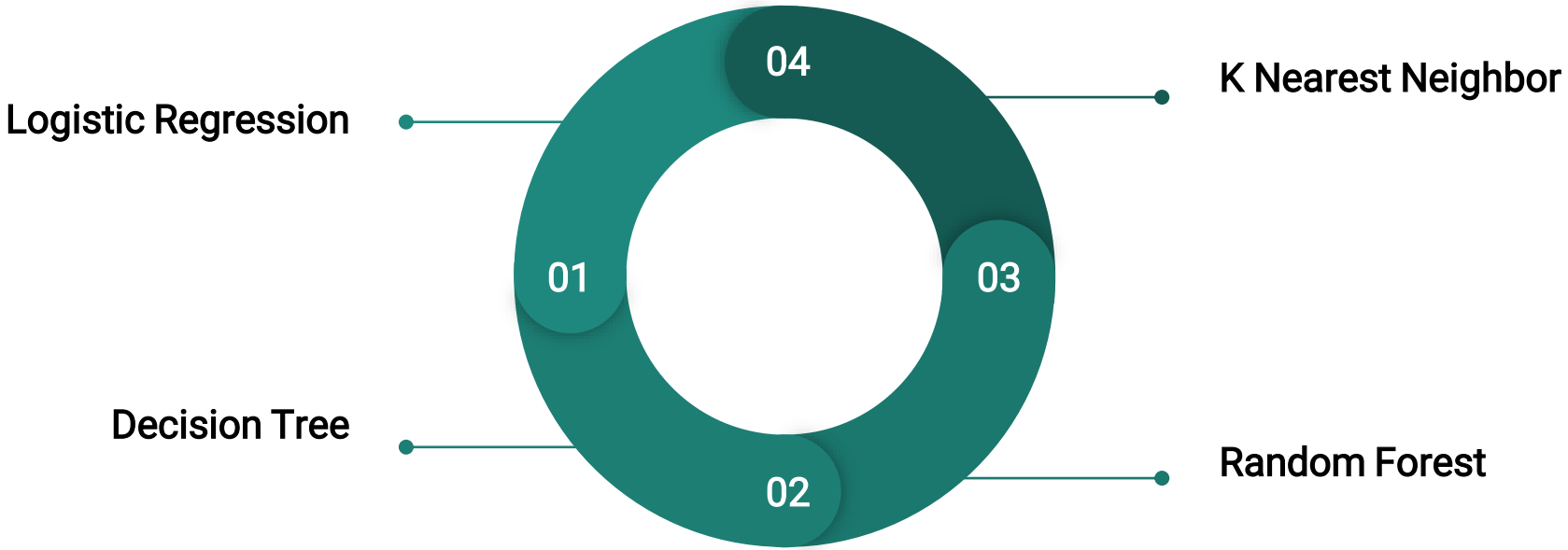
Data Preprocessing

```
# one hot encode all the categorical features
#Lets convert categorical fearures into object dtype first
df_ccd_encoded[['gender','marital_status','education','CC_UsageSept', 'CC_UsageAug', 'CC_UsageJul', 'CC_UsageJun',
       'CC_UsageMay', 'CC_UsageApr']] = df_ccd_encoded[['gender','marital_status','education','CC_UsageSept', 'CC_UsageAug', 'CC_UsageJul', 'CC_UsageJun',
       'CC_UsageMay', 'CC_UsageApr']].astype('object')
```

```
#One Hot encoding
df_ccd_encoded = pd.get_dummies(df_ccd_encoded)
df_ccd_encoded.head()
```

- Created dummy variables for categorical features using one hot encoding technique.
- The categories must be converted into numbers because many machine learning algorithms cannot work with categorical data directly.
- Hence, all the categorical input and output variables are changed to numerical data types for further application of models to the dataet.

Models used for Classification



1. Logistic Regression

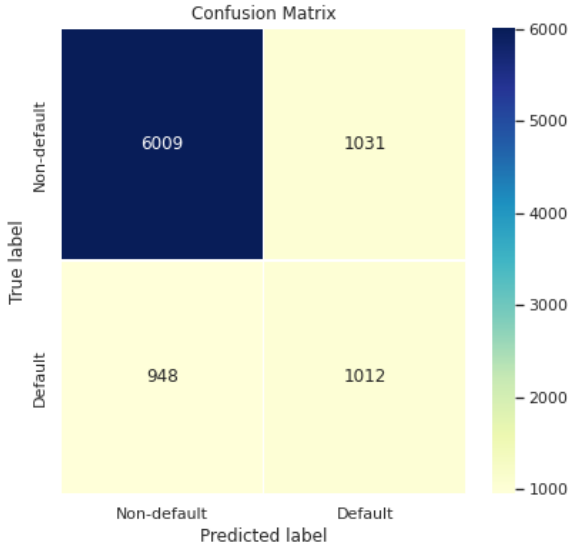
- A statistical method that is used for building machine learning models where the dependent variable is dichotomous: i.e. binary.

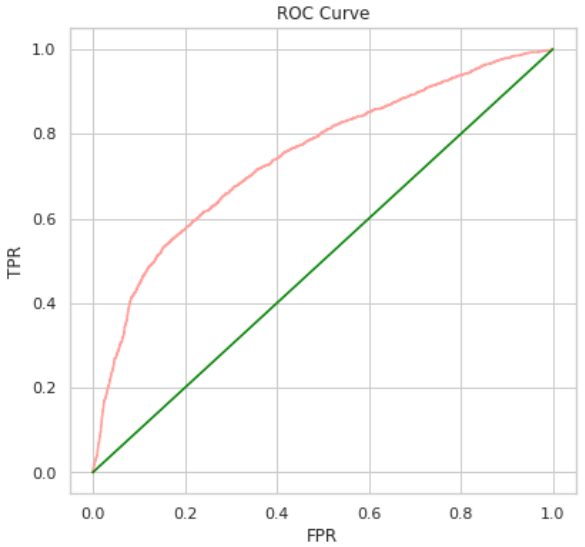
- It assigns probabilities to discrete outcomes using the Sigmoid function, which converts numerical results into an expression of probability between 0 and 1.0.

HYPERPARAMETER

```
'C': 0.0007196856730011
```

```
Data is  SMOTE  And with hyper parameter  {'C': 0.0007196856730011}
Overall Train Accuracy 0.775974025974026
Train AUC Score 0.8234617944898097
Overall Train recall 0.6948664543004166
Overall Test Accuracy
 0.7801111111111111
Test AUC Score
 0.7443705429615027
Classification Report of Test
              precision    recall  f1-score   support

           0       0.86      0.85      0.86      7040
           1       0.50      0.52      0.51      1960

    accuracy                           0.78      9000
   macro avg       0.68      0.68      0.68      9000
weighted avg       0.78      0.78      0.78      9000
```





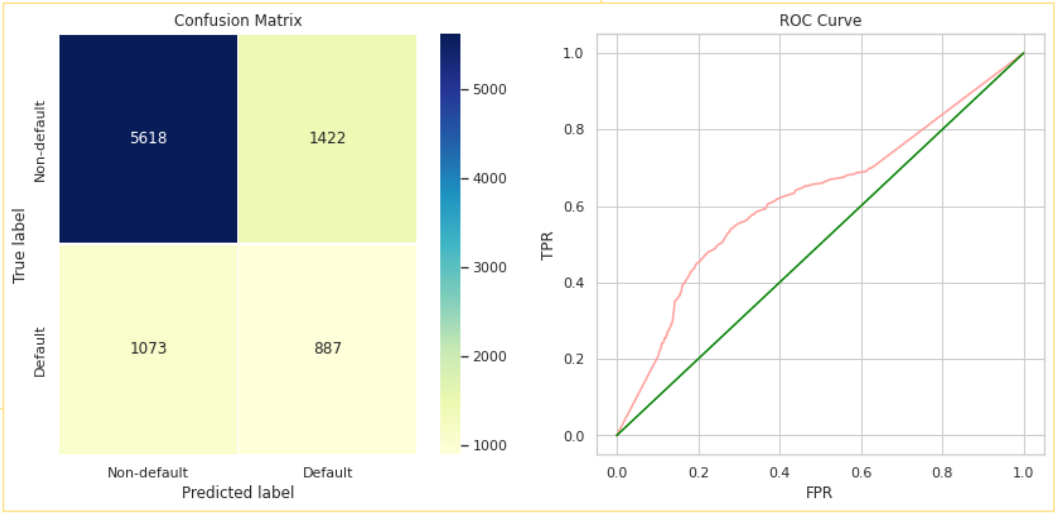
2. Decision Tree

- A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks.
- A Decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

HYPERPARAMETERS

```
'criterion': 'gini',
'max_depth': 17,
'min_samples_leaf': 1,
'min_samples_split': 8
```

```
Data is  SMOTE  And with hyper parameter  {'criterion': 'gini', 'max_depth': 17, 'min_samples_leaf': 1, 'min_samples_split': 8}
Overall Train Accuracy 0.9218022543494242
Train AUC Score 0.9808607965559705
Overall Train recall 0.9046802254349424
Overall Test Accuracy
 0.7227777777777777
Test AUC Score
 0.6209834111201299
Classification Report of Test
              precision    recall  f1-score   support

           0       0.84      0.80      0.82      7040
           1       0.38      0.45      0.42      1960

    accuracy                           0.72      9000
   macro avg       0.61      0.63      0.62      9000
weighted avg       0.74      0.72      0.73      9000
```



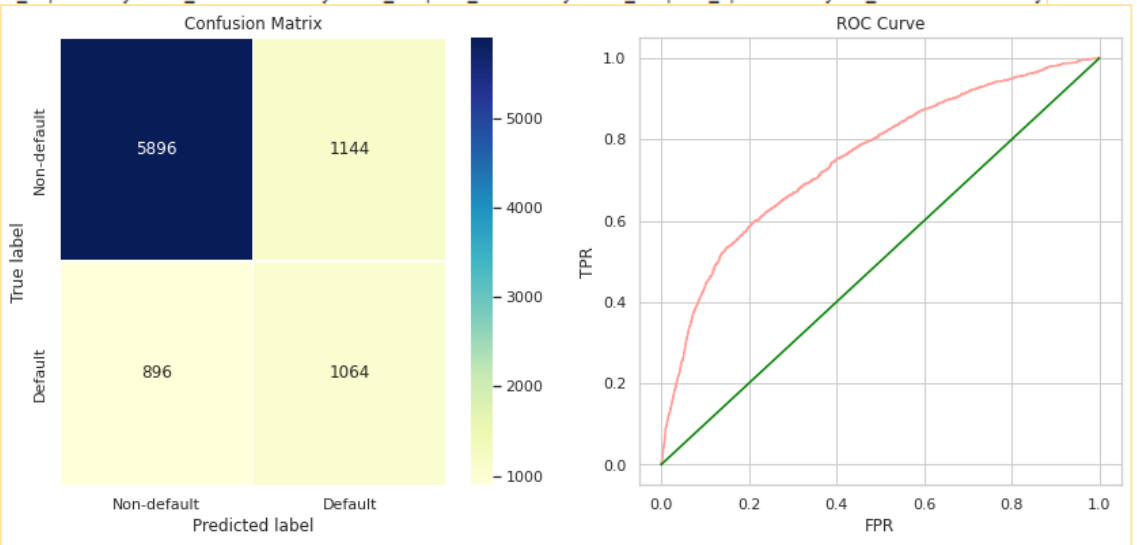
3. Random Forest

- An ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.
- Uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

HYPERPARAMETER

```
'criterion': 'gini'
'max_depth': 4,
'max_features': 1,
'min_samples_leaf': 5,
'min_samples_split': 11,
'n_estimators': 90
```

```
Data is  SMOTE  And with hyper parameter  {'criterion': 'gini', 'max_depth': 4, 'max_features': 1, 'min_samples_leaf': 5, 'min_samples_split': 11, 'n_estimators': 90}
Overall Train Accuracy 0.7775361431021809
Train AUC Score 0.8336883155461138
Overall Train recall 0.7076084293065426
Overall Test Accuracy
 0.7733333333333333
Test AUC Score
 0.7522314906076066
Classification Report of Test
              precision    recall  f1-score   support

           0       0.87      0.84      0.85      7040
           1       0.48      0.54      0.51      1960

    accuracy                           0.77      9000
   macro avg       0.67      0.69      0.68      9000
weighted avg       0.78      0.77      0.78      9000
```



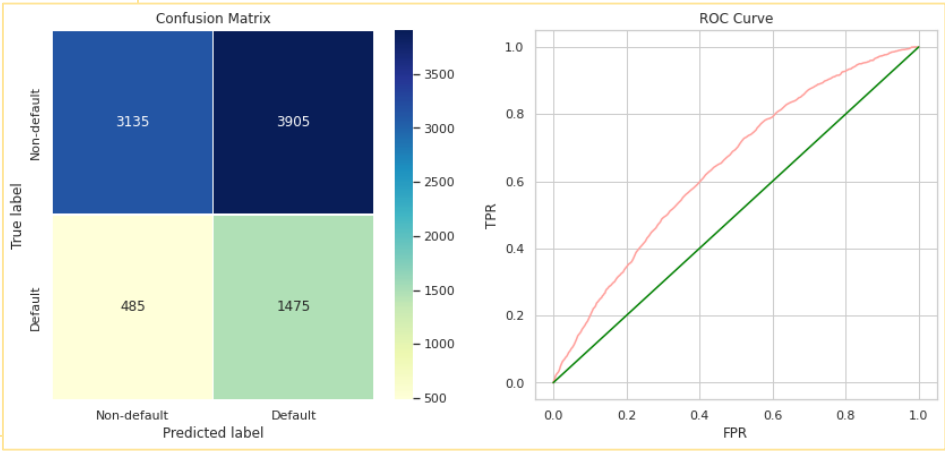
4. K Nearest Neighbor

- A non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point.
- While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

HYPERPARAMETER

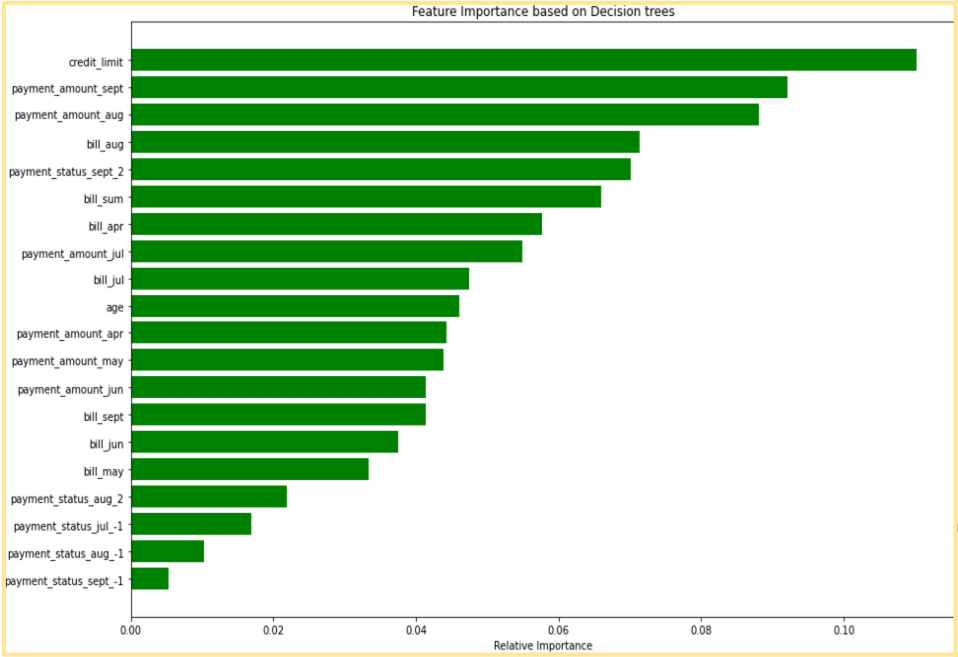
```
'n_neighbour' :'2000'
'leaf_size': '20'
```

```
Data is  SMOTE  And with hyper parameter  {'n_neighbors': 500, 'leaf_size': 10}
Overall Train Accuracy 0.6126868414604264
Train AUC Score 0.6672426343900628
Overall Train recall 0.7714408233276158
Overall Test Accuracy
 0.5122222222222222
Test AUC Score
 0.6418085067833952
Classification Report of Test
              precision    recall  f1-score   support

           0       0.87      0.45      0.59      7040
           1       0.27      0.75      0.40      1960

    accuracy                           0.51      9000
   macro avg       0.57      0.60      0.50      9000
weighted avg       0.74      0.51      0.55      9000
```

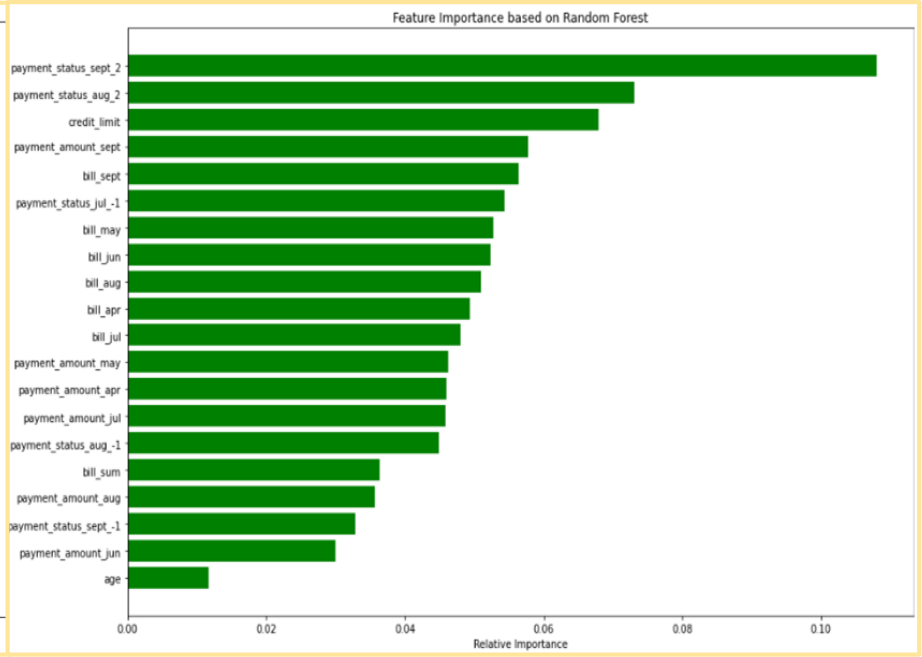


Feature Importance – Tree based model

Feature importance based on **Decision Tress**



Feature importance based on **Random Forest**



- Credit limit has given highest importance comparatively

Summary

Classifier	Train Accuracy	Test Accuracy	Precision Score	Recall Score	F1 Score
Logistic Regression	0.7798	0.7698	0.87(class 0) 0.48(class 1)	**0.83(class 0)** **0.56(class 1)**	0.85(class 0) 0.51(class 1)
Decision Tree	0.8769	0.6974	0.85(class 0) 0.34(class 1)	0.75(class 0) 0.49(class 1)	0.80(class 0) 0.40(class 1)
Random Forest	0.6885	0.6515	0.88(class 0) 0.33(class 1)	0.65(class 0) 0.66(class 1)	0.75(class 0) 0.44(class 1)
KNN Classifier	0.5983	0.5395	0.85(class 0) 0.28(class 1)	0.50(class 0) 0.68(class 1)	0.63(class 0) 0.39(class 1)

Conclusion

- The data was highly imbalanced; there was a huge difference between the defaulters and non-defaulters; thus the classes needed to be balanced.
- Of all the credit card holders, about 22% people are expected to default next month and 77.8% are not expected to default.
- It was found that there are more female credit card holders than male credit card holders; while on the contrary it as found that male credit card holders have higher tendency to default.
- According to the level of education, it was found that the highest number of credit card holders were customers with university level education, followed by graduate school level, then high school students and the least belonged to the ‘others’ category.
- Customers with high school and university educational level had higher default percentages than customers with grad school education. Hence, it can be concluded that customers with lower education levels default more.
- The marital status does not influence defaults. However, ‘others’ category was found to default the most.
- Credit card holders aged between 30-50 had the lowest delayed payment rate, while younger groups (20-30) and older groups (50-70) all had higher delayed payment rates. However, the delayed rate dropped slightly again for customers older than 70 year.

Conclusion

- There was an imbalance in the target variable which was balanced using SMOTE (Synthetic Minority Oversampling Technique).
- Logistic Regression, Decision Trees, Random Forest algorithms were implemented. The important metric to compare all the algorithms in this case is ‘Recall’. As the company can’t afford to predict False negative i.e. predict defaulter as a non defaulter. Since, company is one, who will give to money to the customers, if, for any reason giving money to defaulter is gaining more risk to getting the investment back. Hence, here identifying false negative is important.
- Logistic Regression had an imbalance in the recall score of about 83% for class 0 and 56% for class 1.
- Performance on Decision Tree and Random Forest is comparatively better. Decision Trees and Random Forest have recall scores of 75%(class 0) , 49%(class 1) and 65%(class 0), 66%(class 1) respectively.
- KNN classifier could be a good model but it needs further hypertuning.
- The features like credit limit, payment amount and bill amount are important features as per Random Forest and Decision tree algorithm.

THANK YOU