Imacharia / Default_predictor Public

generated from Imacharia/wataalam-analytics--kings-developers-project

View license

0 stars 0 forks

Star Watch

Code Issues Pull requests Actions Projects Wiki Security Insights Se

main

...

Imacharia Update README.md ... now 14

View code

Business Understanding

Overview of the Project and its Goals:

The goal of this project is to develop a credit card default prediction model using a given dataset. The dataset contains information about credit card clients, including their demographics, credit history, bill statements, and payment records. By analyzing this data, we aim to build a predictive model that can accurately predict whether a credit card client will default on their payment or not.

Problem Statement and Importance of Credit Card Default Prediction:

The problem statement revolves around predicting credit card default, which refers to the failure of a borrower to make timely payments on their credit card. Credit card default prediction is crucial for financial institutions, such as banks and credit card companies, as it helps them assess the creditworthiness and risk profile of their clients. By accurately predicting credit card default, financial institutions can take proactive measures to mitigate potential risks and make informed decisions regarding credit approvals, setting credit limits, and debt collection strategies.

Reading dataset

The data is provided by: Yeh,I-Cheng. (2016). default of credit card clients. UCI Machine Learning Repository. https://doi.org/10.24432/C55S3H.

The data attributes are as follows:

This research employed a binary variable, default payment (Yes = 1, No = 0), as the response variable. This study reviewed the literature and used the following 23 variables as explanatory variables:

> X1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.

> X2: Gender (1 = male; 2 = female).

> X3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).

> X4: Marital status (1 = married; 2 = single; 3 = others).

> X5: Age (year).

> X6 - X11: History of past payment. We tracked the past monthly payment records (from April to September, 2005) as follows: >> `X6` = the repayment status in September, 2005; `X7` = the repayment status in August, 2005; . . .; `X11` = the repayment status in April, 2005.

>> The measurement scale for the repayment status is: -1 = pay 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.

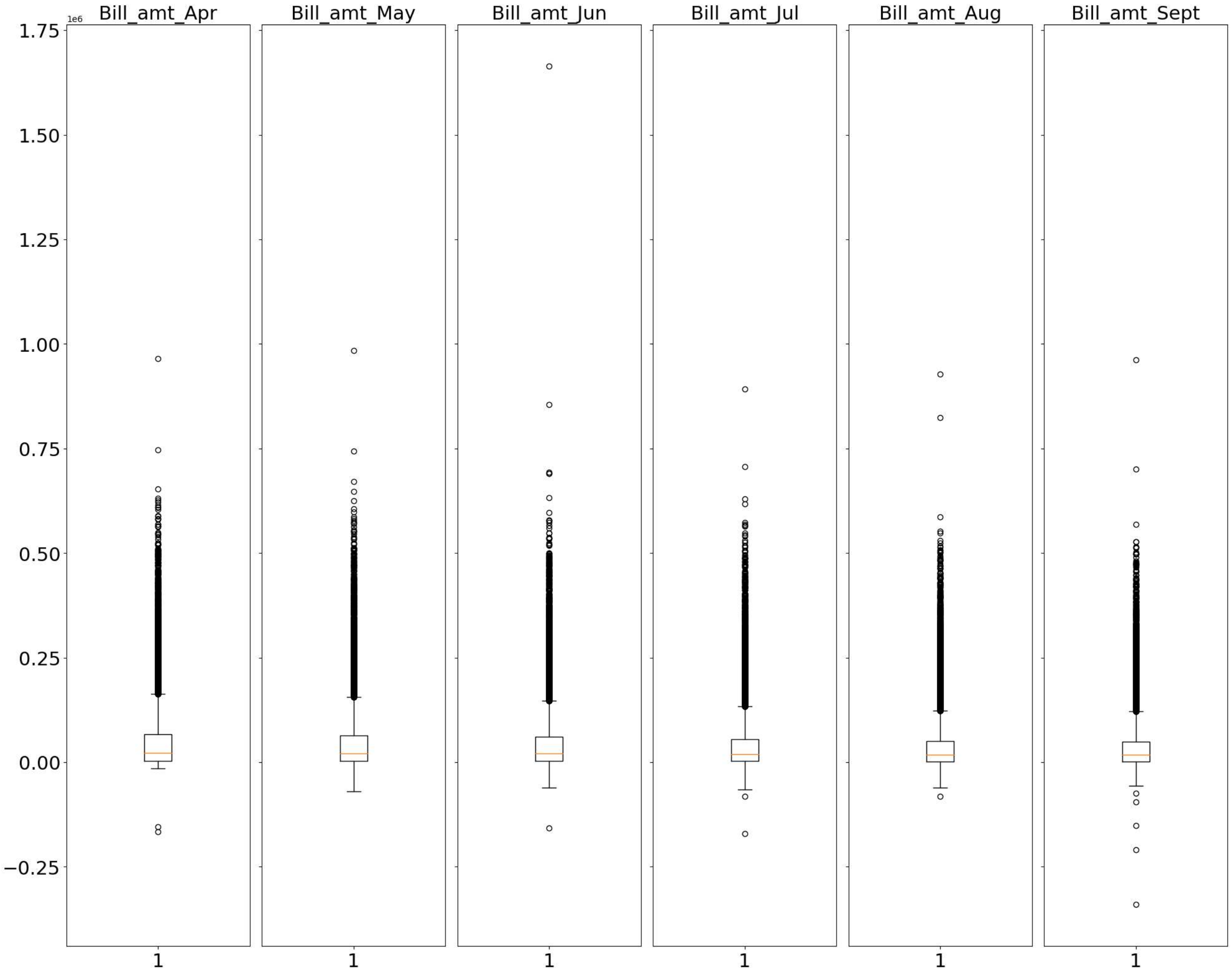
> X12-X17: Amount of bill statement (NT dollar). X12 = amount of bill statement in September, 2005; X13 = amount of bill statement in August, 2005; . . .; X17 = amount of bill statement in April, 2005.

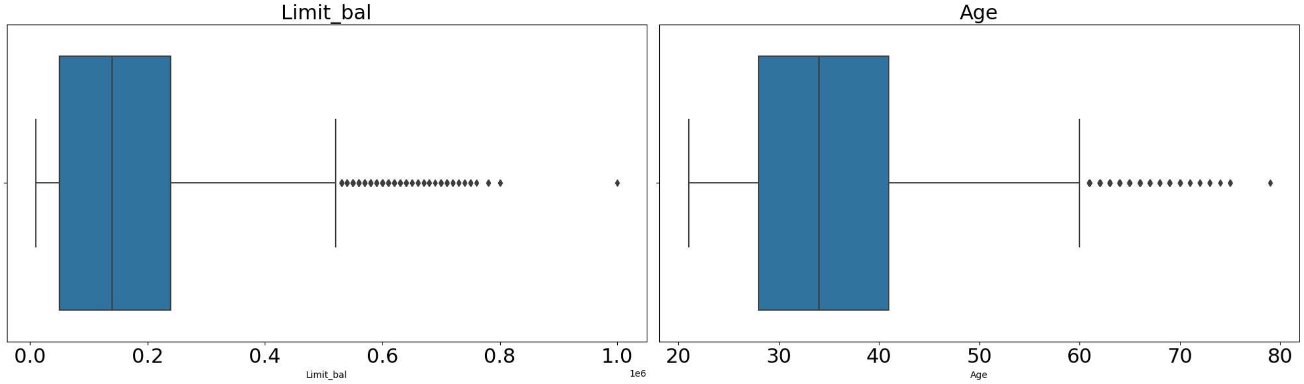
> X18-X23: Amount of previous payment (NT dollar). X18 = amount paid in September, 2005; X19 = amount paid in August, 2005; . . .;X23 = amount paid in April, 2005.

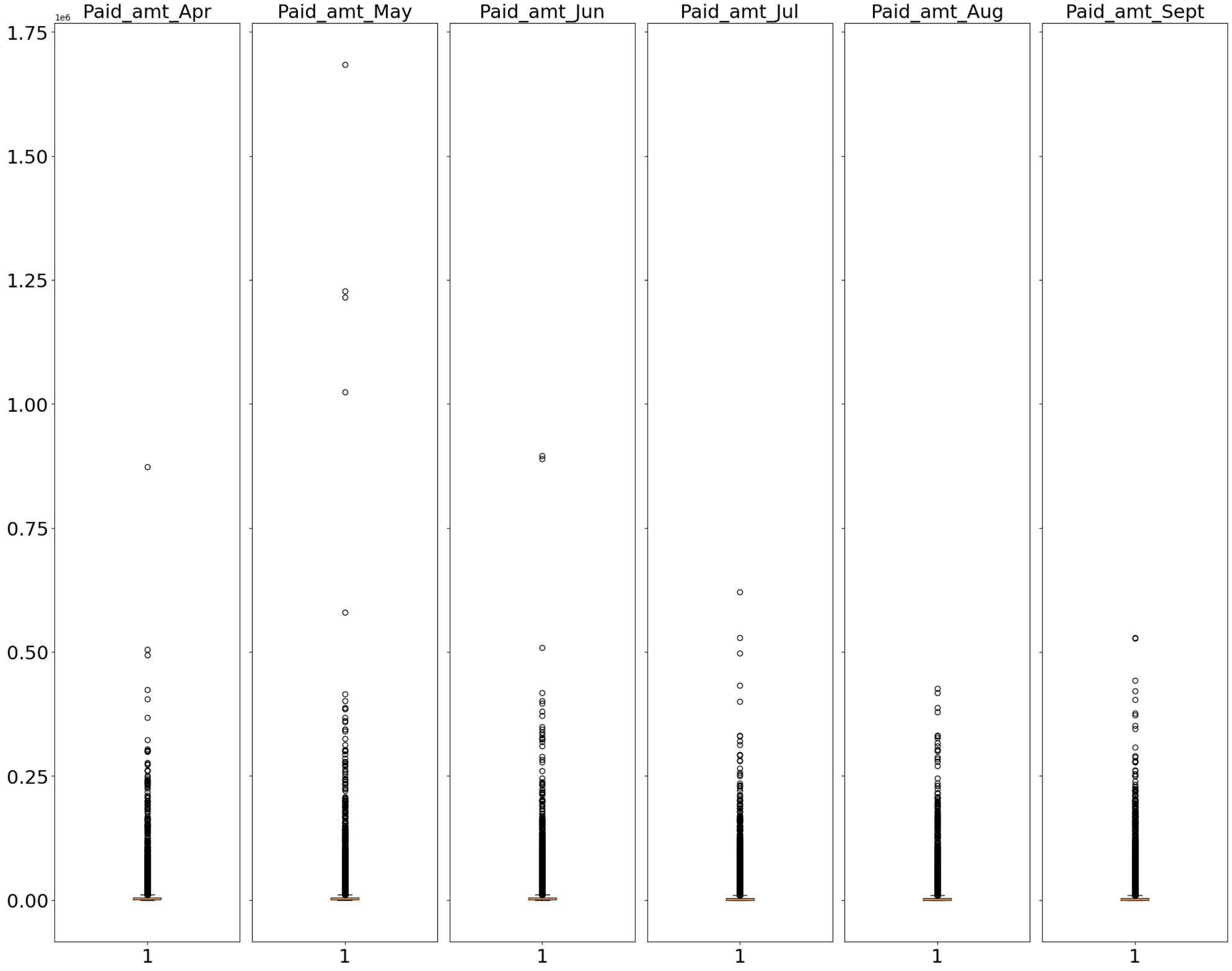
Exploratory Data Analysis (EDA)

Checking for Outliers

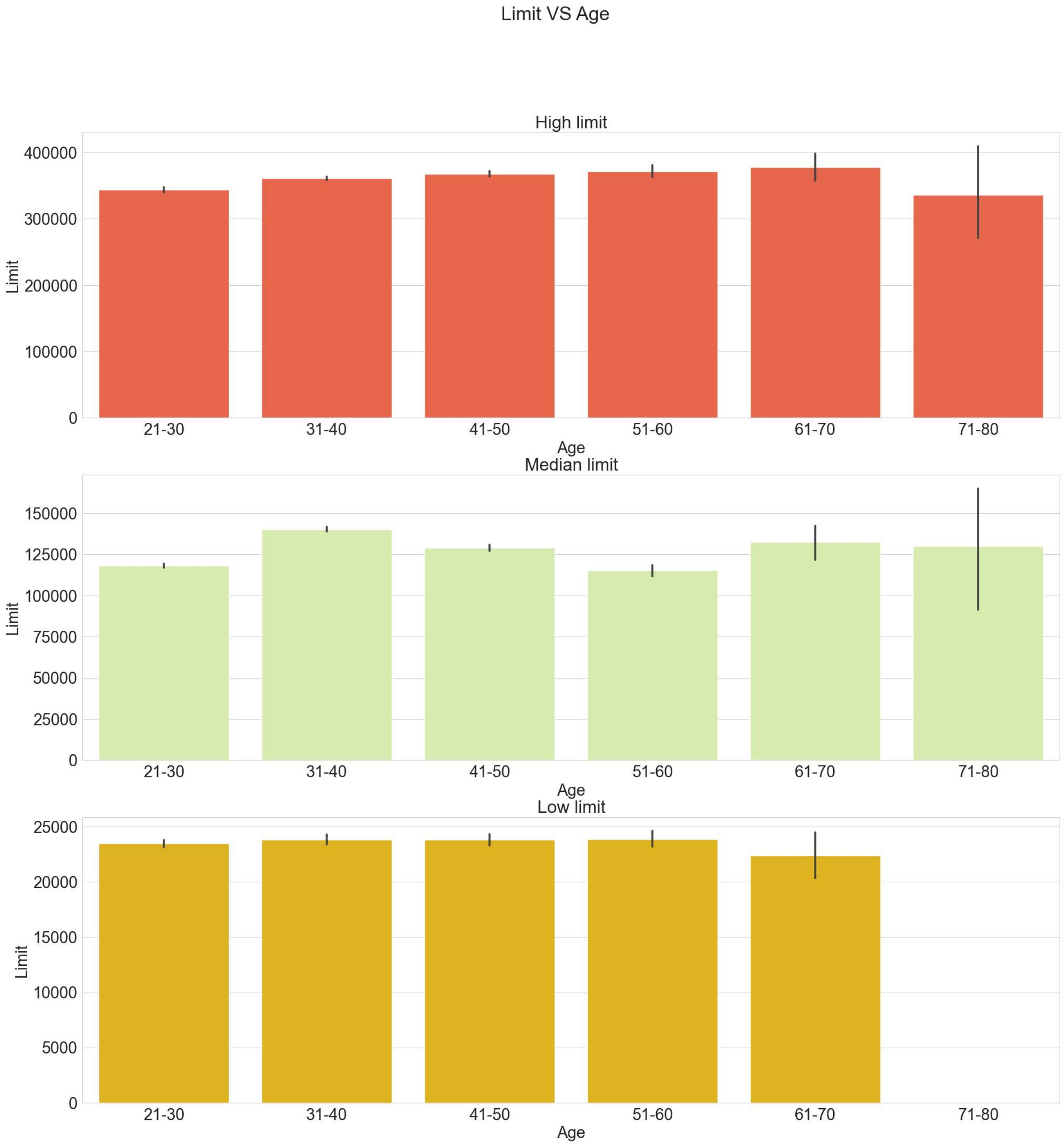
After rigourous data clean up, we the try and fine tune the data for ploting, visualization and subsequent modelling. we will begin by checking for possible outliers



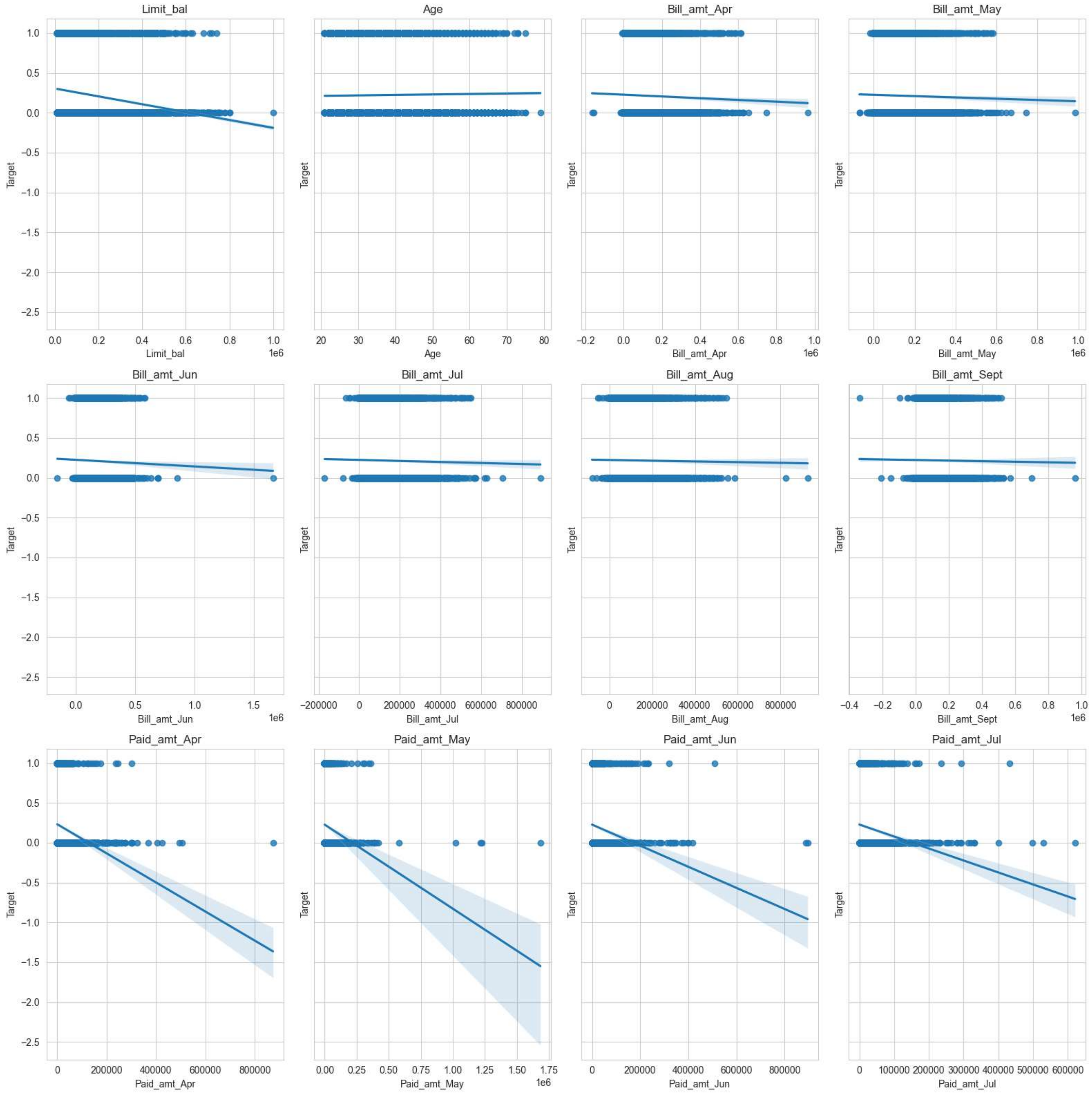

Limit_bal
Age
0.0
0.2
0.4
0.6
0.8
1.0
1e6
Limit_bal
20
30
40
50
60
70
80
Age


1e6
Paid_amt_Apr
Paid_amt_May
Paid_amt_Jun
Paid_amt_Jul
Paid_amt_Aug
Paid_amt_Sept
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
1
1
1
1
1
1

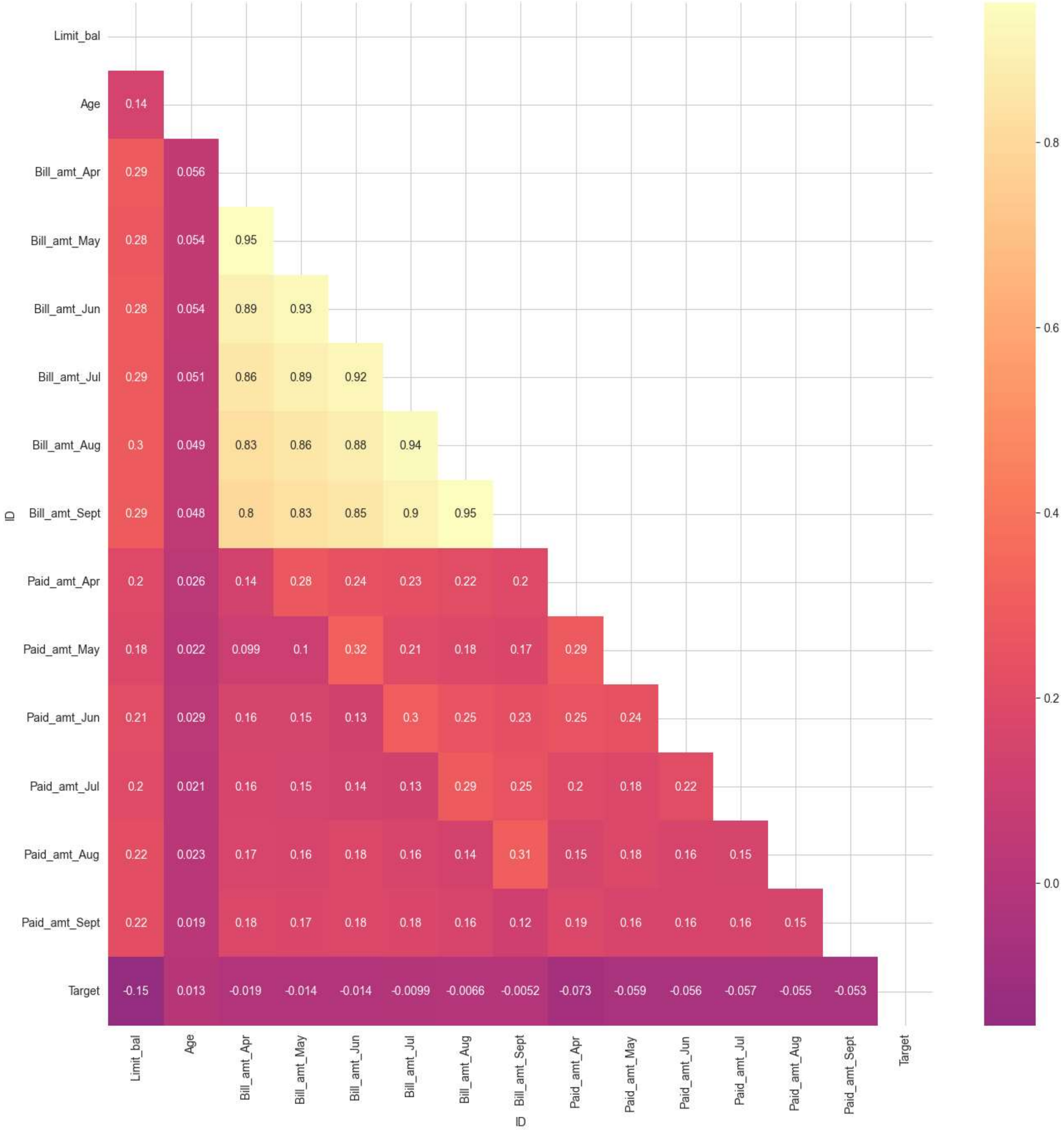
From the graphs we observe that the data is filled with outliers, but considering that they represent different clients, it provides a diversity that will be an effective representations of the whole population. We will instead normalize and standardize the data to have them in a normal distribution.



We observe that for the distribution of age, the limit is almost evenly distributed, although for the bracket 71-80, they all have loan limits above twenty five thousand, with the rest having almost an equal number of limit. We can still deduce that the outliers present in the data are in the age group 71 - 80. and it would be appropriate to assume the highest limit is also in this bracket. Next we will try and plot regression plots to better understand the relationship between the features and the target variabes


Limit_bal
Age
Bill_amt_Apr
Bill_amt_May
Bill_amt_Jun
Bill_amt_Jul
Bill_amt_Aug
Bill_amt_Sept
Paid_amt_Apr
Paid_amt_May
Paid_amt_Jun
Paid_amt_Jul
Target
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0.0 0.2 0.4 0.6 0.8 1.0
1e6
20 30 40 50 60 70 80
−0.2 0.0 0.2 0.4 0.6 0.8 1.0
0.0 0.5 1.0 1.5
−200000 0 200000 400000 600000 800000
0 200000 400000 600000 800000
−0.4 −0.2 0.0 0.2 0.4 0.6 0.8 1.0
0 200000 400000 600000 800000
0.00 0.25 0.50 0.75 1.00 1.25 1.50 1.75
0 100000 200000 300000 400000 500000 600000

The output above is a grid of regression plots, where each plot shows the relationship between a specific feature and the target variable 'Target'. The plots can help visualize the linear relationship, if any, between the features and the target variable, and provide insights into the potential predictive power of the features.



	Limit_bal	Age	Bill_amt_Apr	Bill_amt_May	Bill_amt_Jun	Bill_amt_Jul	Bill_amt_Aug	Bill_amt_Sept	Paid_amt_Apr	Paid_amt_May	Paid_amt_Jun	Paid_amt_Jul	Paid_amt_Aug	Paid_amt_Sept	Target
Limit_bal															
Age	0.14														
Bill_amt_Apr	0.29	0.056													
Bill_amt_May	0.28	0.054	0.95												
Bill_amt_Jun	0.28	0.054	0.89	0.93											
Bill_amt_Jul	0.29	0.051	0.86	0.89	0.92										
Bill_amt_Aug	0.3	0.049	0.83	0.86	0.88	0.94									
Bill_amt_Sept	0.29	0.048	0.8	0.83	0.85	0.9	0.95								
Paid_amt_Apr	0.2	0.026	0.14	0.28	0.24	0.23	0.22	0.2							
Paid_amt_May	0.18	0.022	0.099	0.1	0.32	0.21	0.18	0.17	0.29						
Paid_amt_Jun	0.21	0.029	0.16	0.15	0.13	0.3	0.25	0.23	0.25	0.24					
Paid_amt_Jul	0.2	0.021	0.16	0.15	0.14	0.13	0.29	0.25	0.2	0.18	0.22				
Paid_amt_Aug	0.22	0.023	0.17	0.16	0.18	0.16	0.14	0.31	0.15	0.18	0.16	0.15			
Paid_amt_Sept	0.22	0.019	0.18	0.17	0.18	0.18	0.16	0.12	0.19	0.16	0.16	0.16	0.15		
Target	-0.15	0.013	-0.019	-0.014	-0.014	-0.0099	-0.0066	-0.0052	-0.073	-0.059	-0.056	-0.057	-0.055	-0.053	

A correlation matrix is a table that shows the correlation coefficients between different variables. It is a useful tool for understanding the relationships between variables in a dataset. In this case, the correlation matrix includes correlations between various features.we can take a closer look at thes correlation of other features against the variable 'Target'.

Here's an explanation of the possible correlations provided:

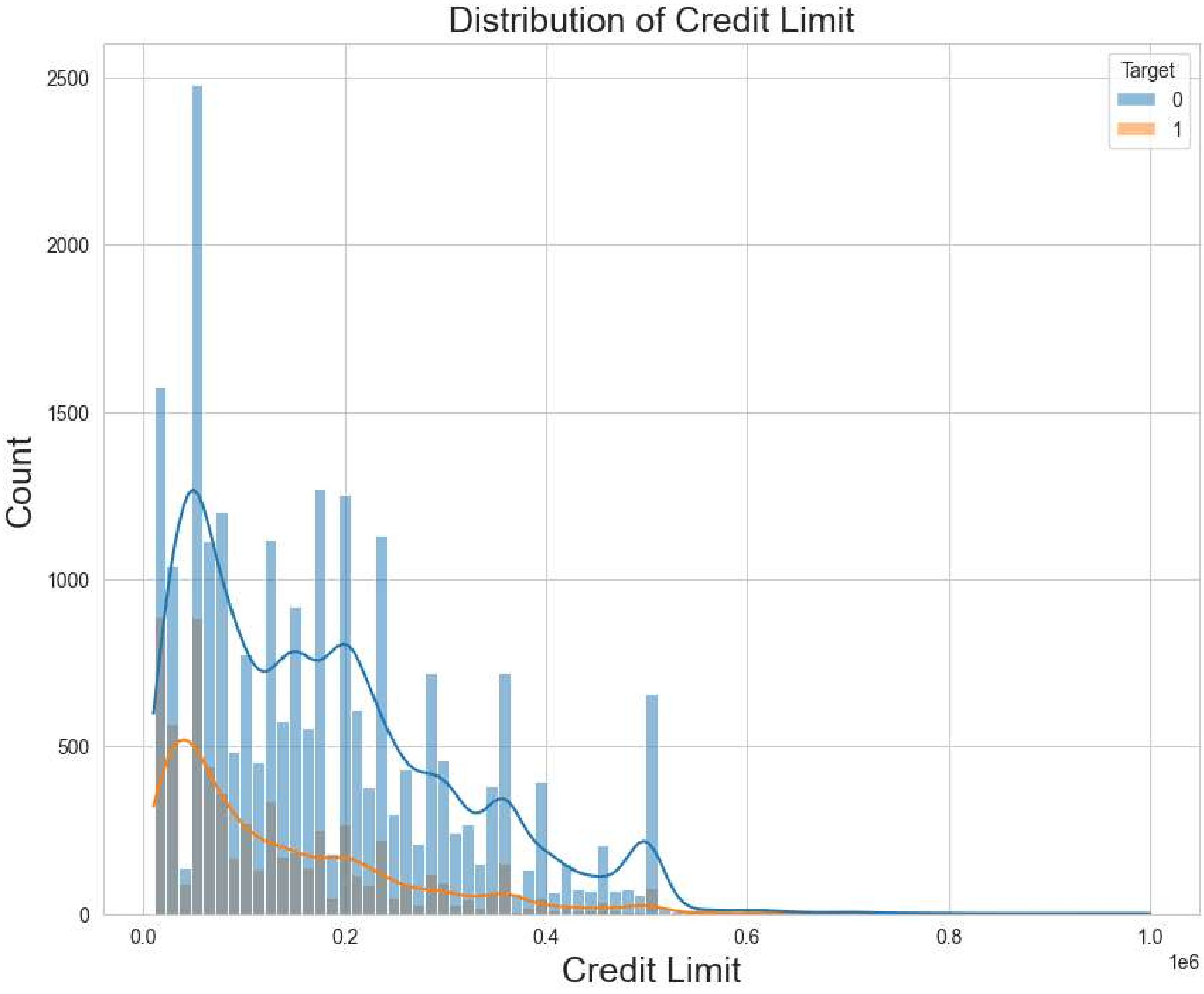
• Limit_bal: It has a negative correlation of -0.154062 with the target variable 'Target'. This suggests that as the credit limit increases, the likelihood of the target variable being positive (1) decreases, and vice versa.

• Age: It has a positive correlation of 0.013295 with the target variable 'Target'. This indicates a weak positive relationship between age and the target variable.

• Bill_amt_Apr, Bill_amt_May, Bill_amt_Jun, Bill_amt_Jul, Bill_amt_Aug, Bill_amt_Sept: These features have negative correlations ranging from -0.019437 to -0.005166 with the target variable 'Target'. The negative correlations suggest that higher bill amounts are associated with a lower likelihood of the target variable being positive.

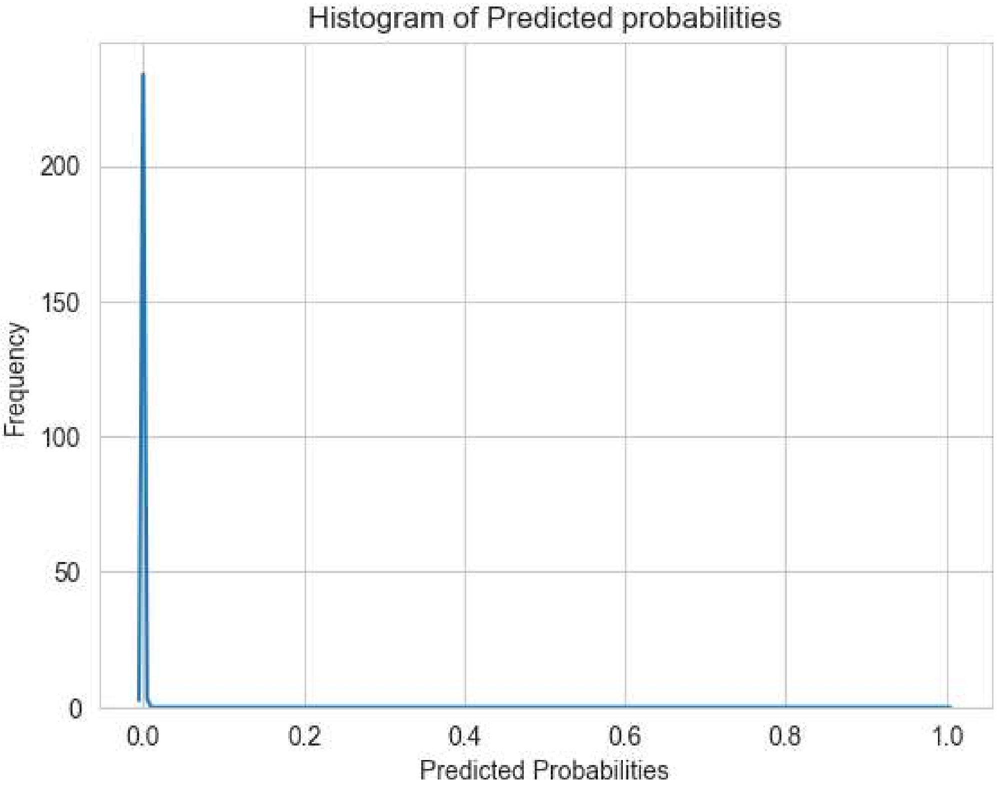
• Paid_amt_Apr, Paid_amt_May, Paid_amt_Jun, Paid_amt_Jul, Paid_amt_Aug, Paid_amt_Sept: These features have negative correlations ranging from -0.072879 to -0.053129 with the target variable 'Target'. The negative correlations suggest that higher paid amounts are associated with a lower likelihood of the target variable being positive.

• Target: It has a correlation coefficient of 1.000000 with itself, which is always 1 as it represents the correlation of a variable with itself. The correlation coefficients range from -1 to 1, with -1 indicating a strong negative correlation, 0 indicating no correlation, and 1 indicating a strong positive correlation. The provided correlations indicate the strength and direction of the linear relationship between each feature and the target variable. However, it's important to note that correlation does not imply causation, and other factors may influence the relationship between variables.


Distribution of Credit Limit
Count
Credit Limit
Target
0
1
0
500
1000
1500
2000
2500
0.0
0.2
0.4
0.6
0.8
1.0
1e6

Modelling

Since our target variable can only have one of two possibilities normal linear regression will not be possible, we will therefore use Logistic regression we will begin by separating our data into the target colum and our predictor variables. Next we will transform the non-numeric to dummy variables which is the standard way for transforming categorical variables for modelling.



The matrix above can be interpreted as:

> > TN: True Negatives (correctly predicted negatives): 7053

> > FP: False Positives (incorrectly predicted positives): 1

> > FN: False Negatives (incorrectly predicted negatives): 1946

> > TP: True Positives (correctly predicted positives): 1

> The Accuracy is the proportion of correctly classified instances of the total number of instances. Our current score show only 77.35% of the instances were classified correctly.

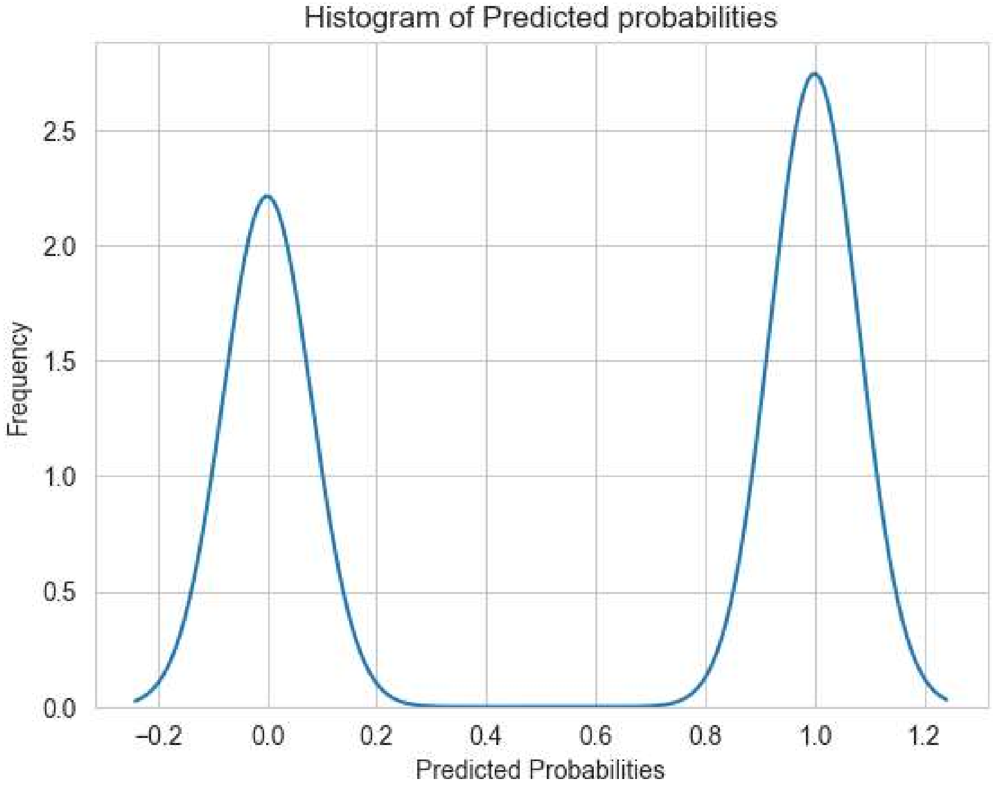
> Precision shows the proportion of true positive predictions out of the total.

> We observe a very low Recall score indicating the model only identified a small fraction of actual positive instances.

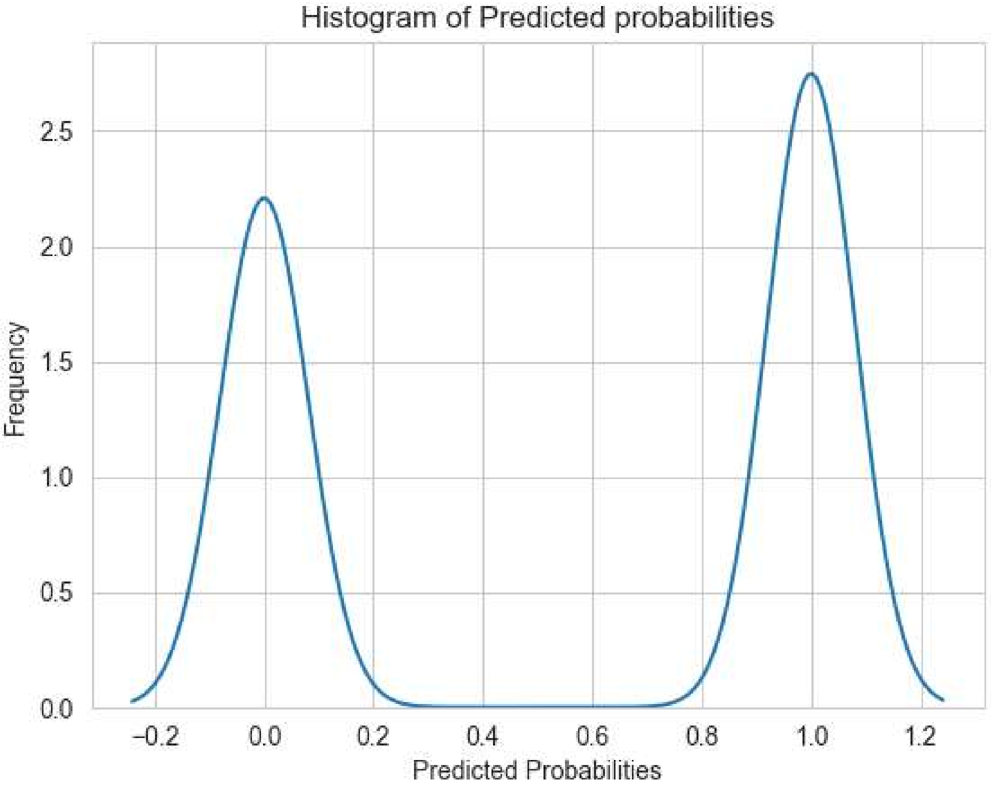
> F1-Score shows the overall performance combining both recall and precision. With this score it indicates poor performance as we deduced earlier.

> The last metric, ROC AUC(Receiver Operating, Characteristic Area Under Curve) measures the models ability to distinguish between positive and negative instances. With a score of close to 0.5, indicates the model has poor discriminatory power.

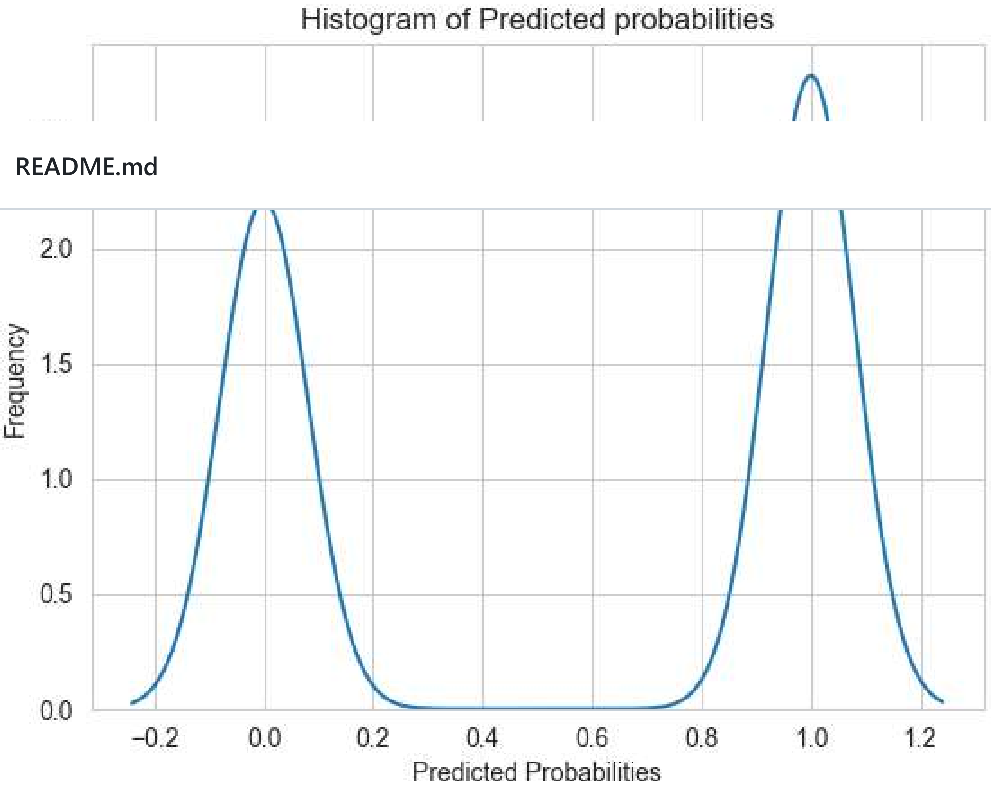
Overall, the results suggest that the model's performance is subpar. It has low recall, indicating that it fails to identify a significant portion of positive instances. The precision is also low, suggesting a high rate of false positives. The F1-score and ROC AUC further confirm the poor performance of the model. Further analysis and improvement of the model may be necessary to achieve better results. We will investigate the impact of class imbalance in our target variable, and based on the findings we will perform Oversampling of the minority class or undersampling of the majority class. We will also employ cross validation to obtain more reliable estimates of the models performance to reduce overfitting. We should also do a log transformation of the data to ensure the data is normally distributed.



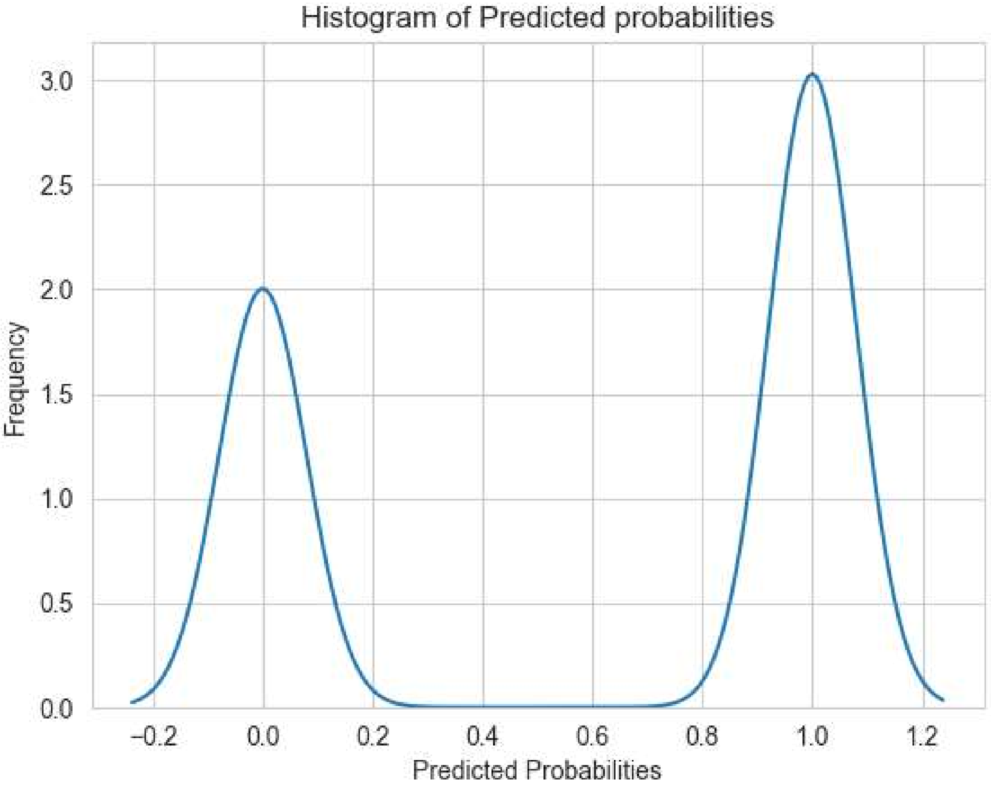
We observe that the predictions moved from a left skewed shape to a bimodal shape. we can attempt to repeat the above models but instead of applying ridge regression, we use lasso regression and observe how it will perform.



The accuracy score of 0.548 means the model correctly predicts approximately 54.8% of the default cases. Although with the low precision score of 0.297 indicates that the model is correct only about 29.7% of the time, this translates to a high number of false positives. Looking at the recall which is the sensitivity aka true positive rate of 0.727, means that the model correctly identifies 72.7% of the actual defaults, although with a relatively high rate of false negatives as well. the F1-score combines both precision and recall to a single metric. having a score of 0.421 indicates a moderate balance between recall and precision. The ROC AUC scored 0.611, with is a significant improvement from the baseline model. this means that the model's ability to discriminate between default and non default is modest.



Accuracy represents the overall correctness of the predictions, indicating that the model is accurate in approximately 54.8% of cases. A precision score of 0.2967 suggests that the model has a relatively low precision, meaning that there are a significant number of false positive predictions. The recall score of 0.7267 indicates that the model is able to capture a relatively high percentage of the true positive cases. A higher F1-Score (0.4214) indicates a better balance between precision and recall. The ROC AUC score of 0.6113 suggests that the model has some discriminative power, but it is not highly accurate in distinguishing between the two classes.



The accuracy of the classifier model is approximately 0.4537, indicating that the model correctly predicts the class of the target variable in around 45.4% of cases.

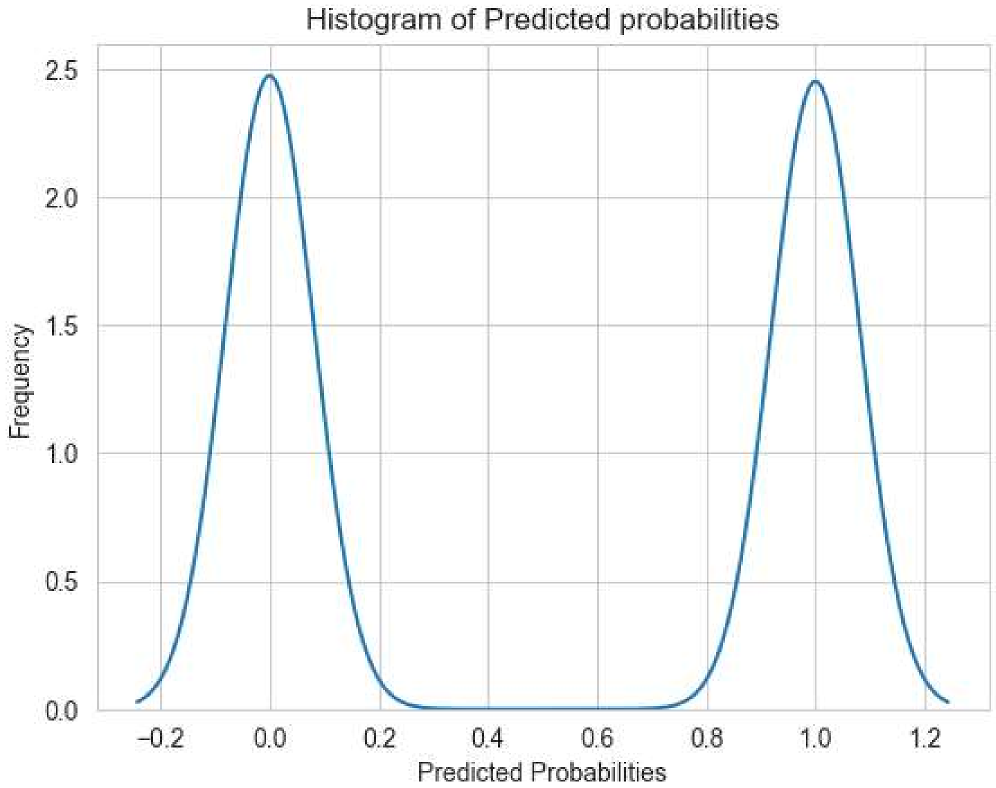
The precision score is approximately 0.2208, which suggests that out of all the instances predicted as positive, only 22.1% are actually true positives.

The recall score is approximately 0.6030, indicating that the model identifies around 60.3% of the actual positive instances.

The F1-Score, which combines precision and recall, is approximately 0.3232. This score provides a balanced measure of the model's performance in terms of both positive and negative predictions.

The ROC AUC score is approximately 0.5078, which suggests that the model's ability to distinguish between positive and negative instances is only slightly better than random chance.

These evaluation metrics indicate that the classifier model has relatively low performance in terms of accuracy, precision, recall, F1-Score, and ROC AUC. It may require further improvement or exploration of other models or techniques to enhance its predictive capabilities.



The accuracy of the classifier model is approximately 0.5156, indicating that the model correctly predicts the class of the target variable in around 51.6% of cases.

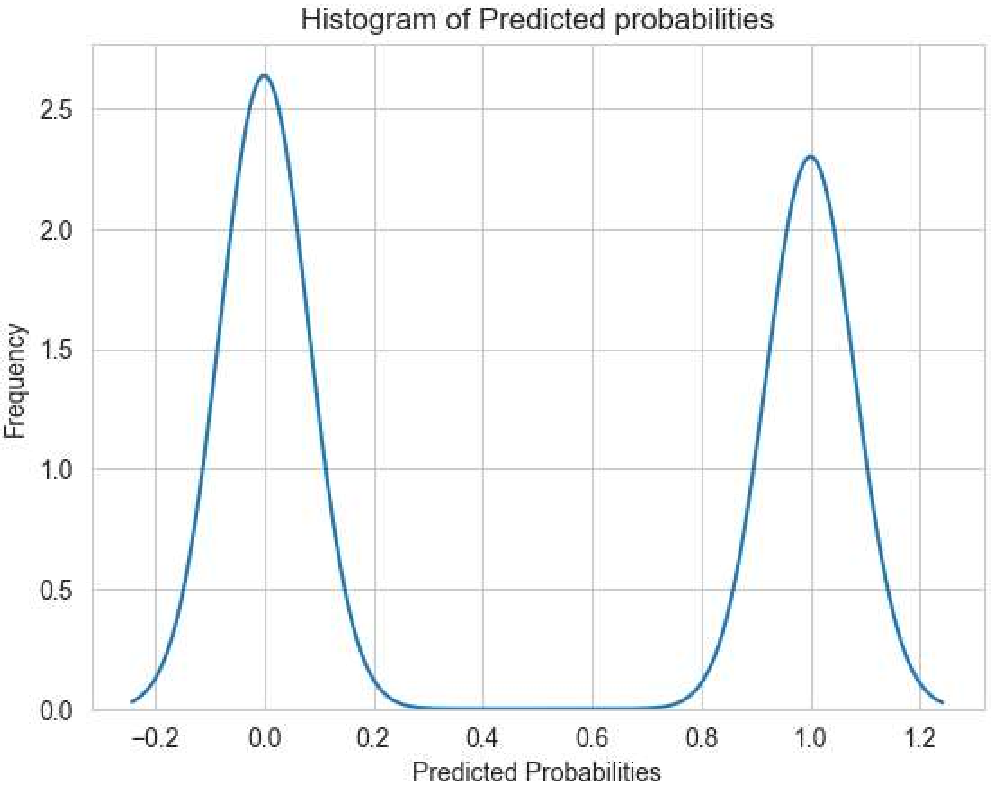
The precision score is approximately 0.2277, which suggests that out of all the instances predicted as positive, only 22.8% are actually true positives.

The recall score is approximately 0.4764, indicating that the model identifies around 47.6% of the actual positive instances.

The F1-Score, which combines precision and recall, is approximately 0.3082. This score provides a balanced measure of the model's performance in terms of both positive and negative predictions.

The ROC AUC score is approximately 0.5018, which suggests that the model's ability to distinguish between positive and negative instances is close to random chance.

These evaluation metrics indicate that the classifier model has relatively low performance in terms of accuracy, precision, recall, F1-Score, and ROC AUC. It may require further improvement or exploration of other models or techniques to enhance its predictive capabilities.



The accuracy of the classifier model is approximately 0.5295, indicating that the model correctly predicts the class of the target variable in around 52.9% of cases.

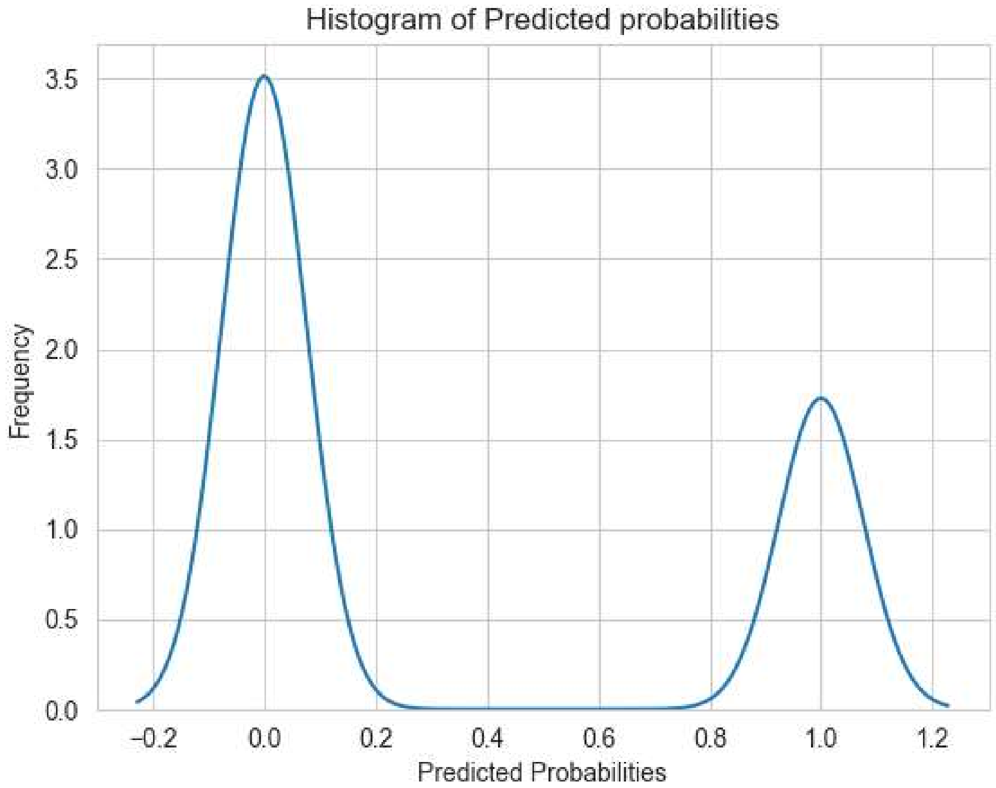
The precision score is approximately 0.2433, which suggests that out of all the instances predicted as positive, only 24.3% are actually true positives.

The recall score is approximately 0.5108, indicating that the model identifies around 51.1% of the actual positive instances.

The F1-Score, which combines precision and recall, is approximately 0.3296. This score provides a balanced measure of the model's performance in terms of both positive and negative predictions.

The ROC AUC score is approximately 0.5229, which suggests that the model's ability to distinguish between positive and negative instances is slightly better than random chance.

These evaluation metrics indicate that the classifier model has moderate performance in terms of accuracy, precision, recall, F1-Score, and ROC AUC. Further improvements could be explored to enhance its predictive capabilities.



The accuracy of the classifier model is approximately 0.6190, indicating that the model correctly predicts the class of the target variable in around 61.9% of cases.

The precision score is approximately 0.2545, which suggests that out of all the instances predicted as positive, only 25.5% are actually true positives.

The recall score is approximately 0.3538, indicating that the model identifies around 35.4% of the actual positive instances.

The F1-Score, which combines precision and recall, is approximately 0.2960. This score provides a balanced measure of the model's performance in terms of both positive and negative predictions.

The ROC AUC score is approximately 0.5252, which suggests that the model's ability to distinguish between positive and negative instances is slightly better than random chance.

RESULTS

The results of our credit card default prediction model indicate that the model's performance is subpar. The logistic regression model achieved an accuracy of 77.35%, which means that 77.35% of the instances were classified correctly. However, the precision, recall, and F1-score are relatively low, indicating room for improvement.

The precision of the model is low, suggesting a high rate of false positives. This means that the model incorrectly identifies a significant number of individuals as likely to default on their credit card payments. The recall score is also low, indicating that the model fails to identify a considerable portion of actual positive instances (individuals who will default). The F1-score, which combines precision and recall, further confirms the poor performance of the model.

The ROC AUC score, which measures the model's ability to distinguish between positive and negative instances, is close to 0.5. This indicates that the model has poor discriminatory power and is not effectively capturing the underlying patterns in the data.

CLASS IMBALANCE INVESTIGATION

We observed a class imbalance in the target variable, with a ratio of 3.52 between the majority class (non-default) and the minority class (default). Class imbalance can have a significant impact on the performance of machine learning models, particularly in classification tasks. Imbalanced classes can lead to biased predictions and a higher tendency to classify instances into the majority class.

RECOMMENDATIONS

Based on the findings of our credit card default prediction model, we make the following recommendations to improve the model's performance:

Address Class Imbalance: Given the class imbalance in the dataset, it is essential to employ techniques to address this issue. Resampling techniques, such as oversampling the minority class or undersampling the majority class, can help balance the classes and improve the model's ability to learn from both classes equally.

Feature Engineering: Explore additional feature engineering techniques to extract more meaningful information from the available data. This can include creating new features based on domain knowledge, combining existing features, or transforming variables to capture non-linear relationships.

Incorporate Additional Features: Consider incorporating additional relevant features into the model. The current dataset includes information about credit amount, demographics, payment history, bill statements, and previous payment amounts. However, there may be other variables that could provide valuable insights into credit card default prediction. Domain expertise and further research can help identify potential additional features to enhance the model's predictive power.

Advanced Modeling Techniques: Experiment with advanced machine learning algorithms specifically designed for classification tasks, such as ensemble methods (e.g., random forest, gradient boosting) or neural networks. These algorithms have the potential to capture complex relationships in the data and improve the model's performance.

Hyperparameter Tuning: Perform hyperparameter tuning to optimize the parameters of the chosen machine learning algorithms. Adjusting the hyperparameters can significantly impact the model's performance and fine-tune its ability to capture the underlying patterns in the data.

Data Quality and Representativeness: Ensure the dataset used for training the model is of high quality and representative of the target population. This includes thorough data preprocessing, handling missing values appropriately, and addressing any potential biases or data collection issues.

Cross-Validation and Model Evaluation: Implement robust model evaluation techniques, such as k-fold cross-validation, to obtain more reliable performance metrics. This helps assess the model's performance on different subsets of the data and provides a better estimate of its generalization capabilities.

Continuous Monitoring and Model Updating: Credit card default prediction is a dynamic problem influenced by changing economic conditions, customer behaviors, and external factors. It is crucial to continuously monitor the model's performance and update it as new data becomes available. Regular model evaluation and retraining will ensure its effectiveness and relevance over tim

These evaluation metrics indicate that the classifier model has moderate performance in terms of accuracy, precision, recall, F1-Score, and ROC AUC. Further improvements could be explored to enhance its predictive capabilities.

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

● **Jupyter Notebook** 99.7% ● **Python** 0.3%

Suggested Workflows

Based on your tech stack

Actions Importer

Set up

Automatically convert CI/CD files to YAML for GitHub Actions.

Pylint

Configure

Lint a Python application with pylint.

Python Package using Anaconda

Configure

Create and test a Python package on multiple Python versions using Anaconda for package management.

More workflows

Dismiss suggestions