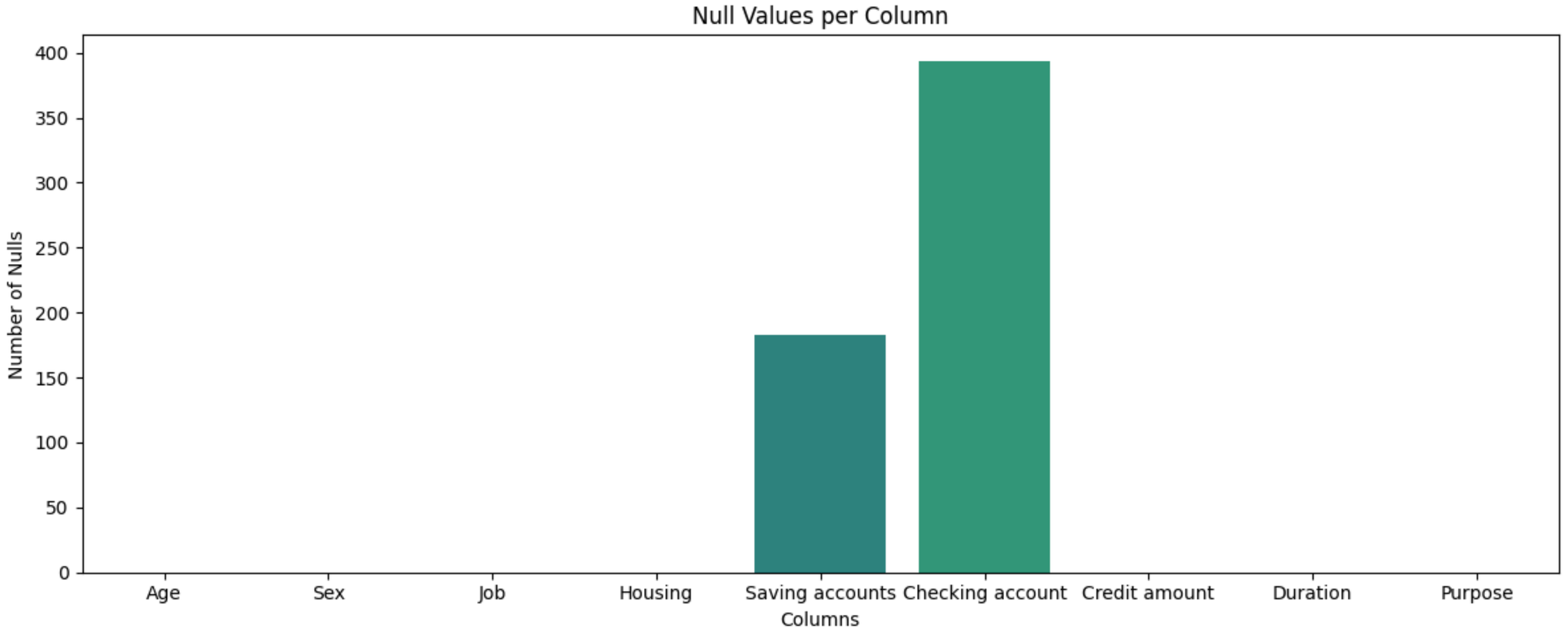
Predicting Credit Risk for Loan Applicants

By:- Daniel Lawrence - DT20234270647

Null Values



This bar chart shows the number of null (missing) values for each column in the dataset.

Null Values

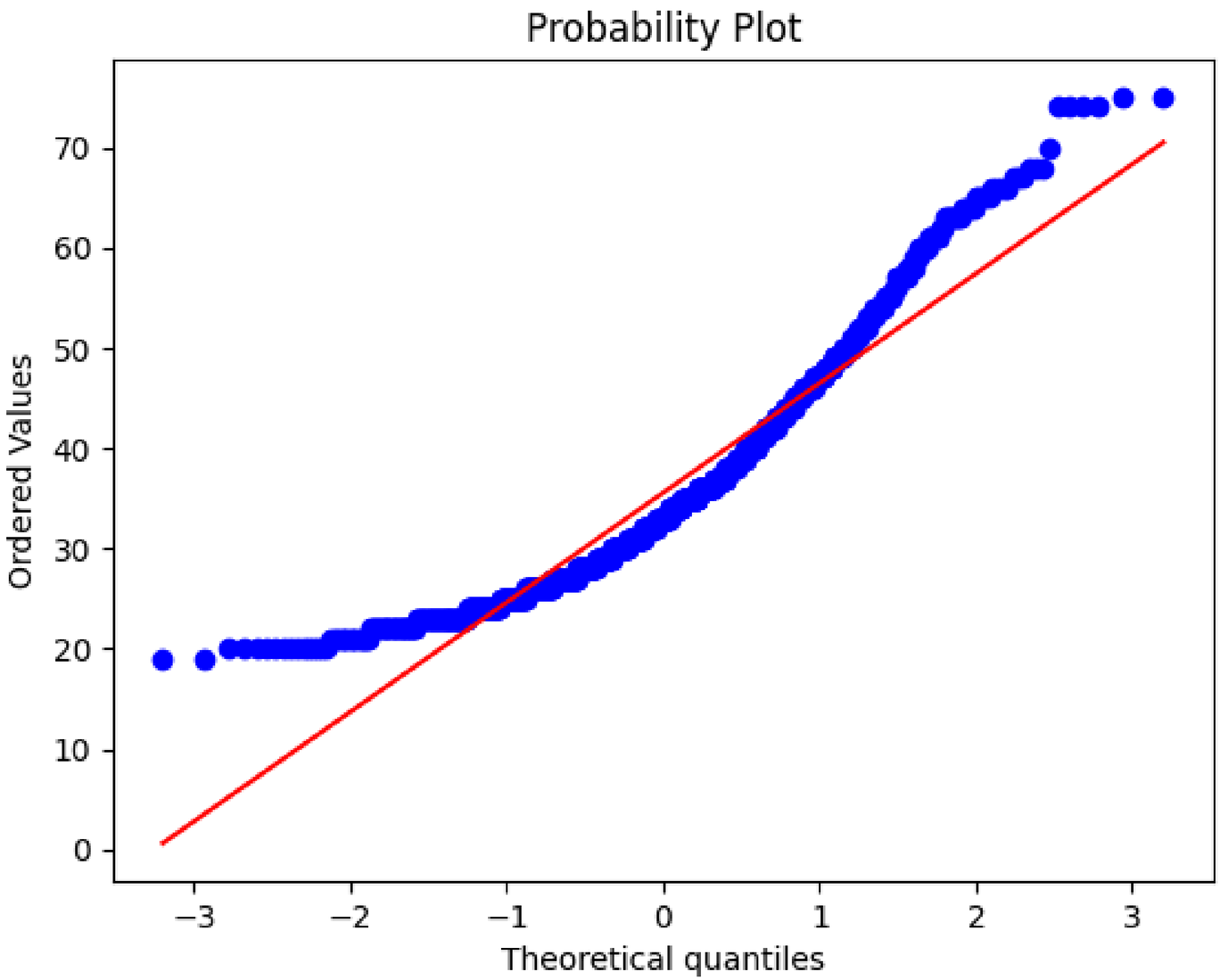
Only two columns have missing data:

- Saving accounts has 183 missing values.
- Checking account has the most, with around 394 missing values out of 1000 entries - nearly 40% missing.

Handling the null values

Instead of leaving them as NaN (which can cause issues with some models), a placeholder category "**unknown**" is used.

This is done to preserve the missingness as potentially meaningful - maybe people with "unknown" account info are riskier?

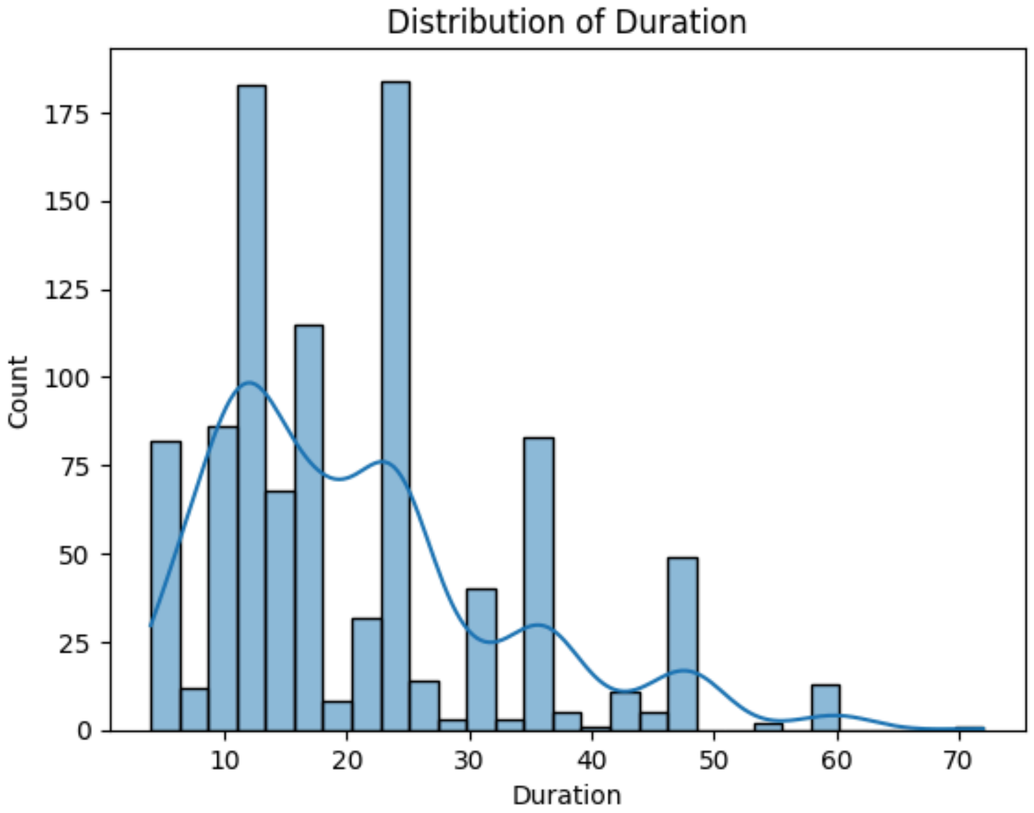


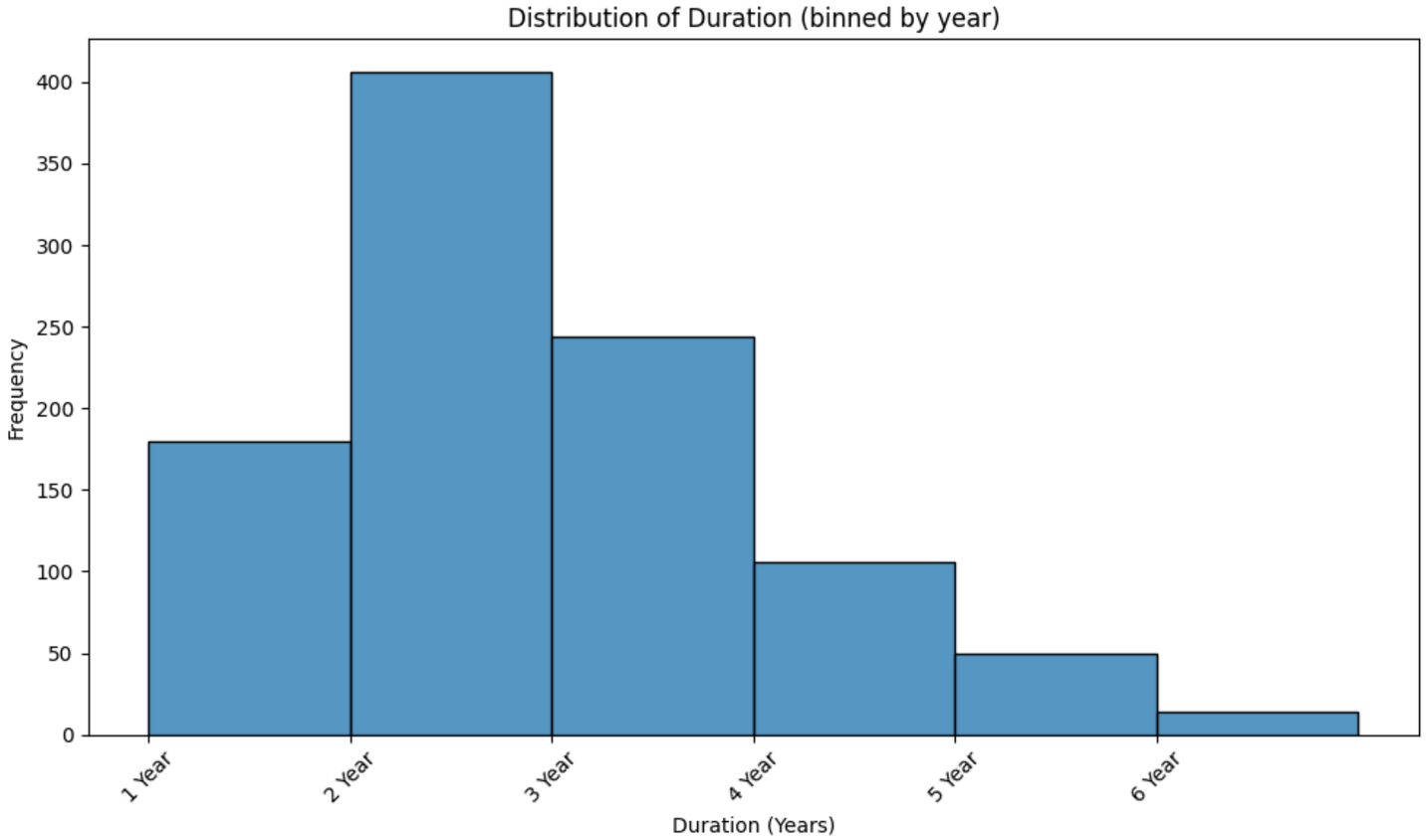
A normal probability plot, or more specifically a quantile-quantile (Q-Q) plot, shows the distribution of the data against the expected normal distribution.

For normally distributed data, observations should lie approximately on a straight line.

Possible outliers are points at the ends of the line, distanced from the bulk of the observations.

In our observation, as we can see that the distribution is not quite normal.



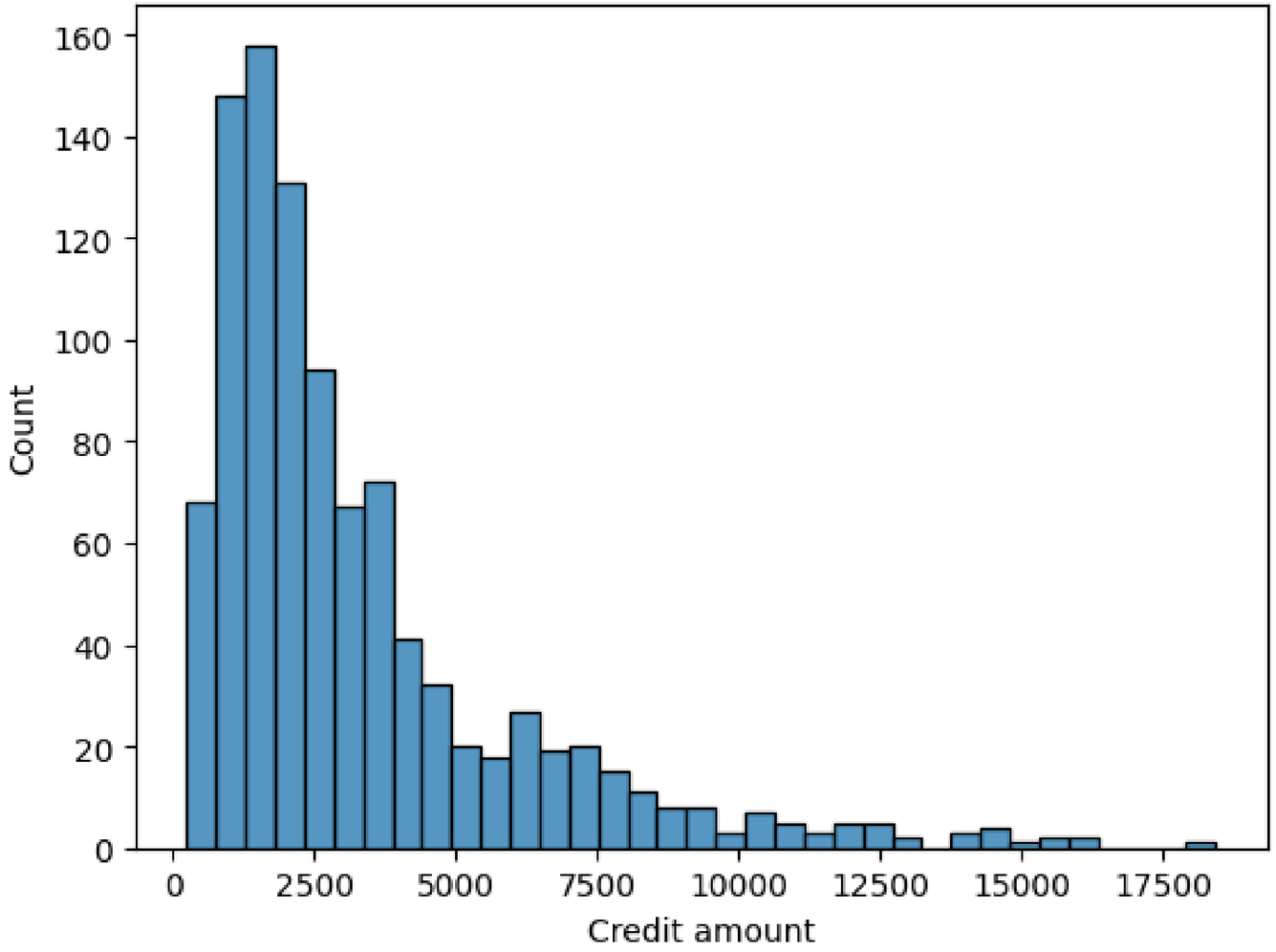


Bin to 12 months to convert duration from months to years

Most loans are short-term:

- The 2-year duration is the most common, with over 400 loans.
- Followed by 3-year and 1-year durations.

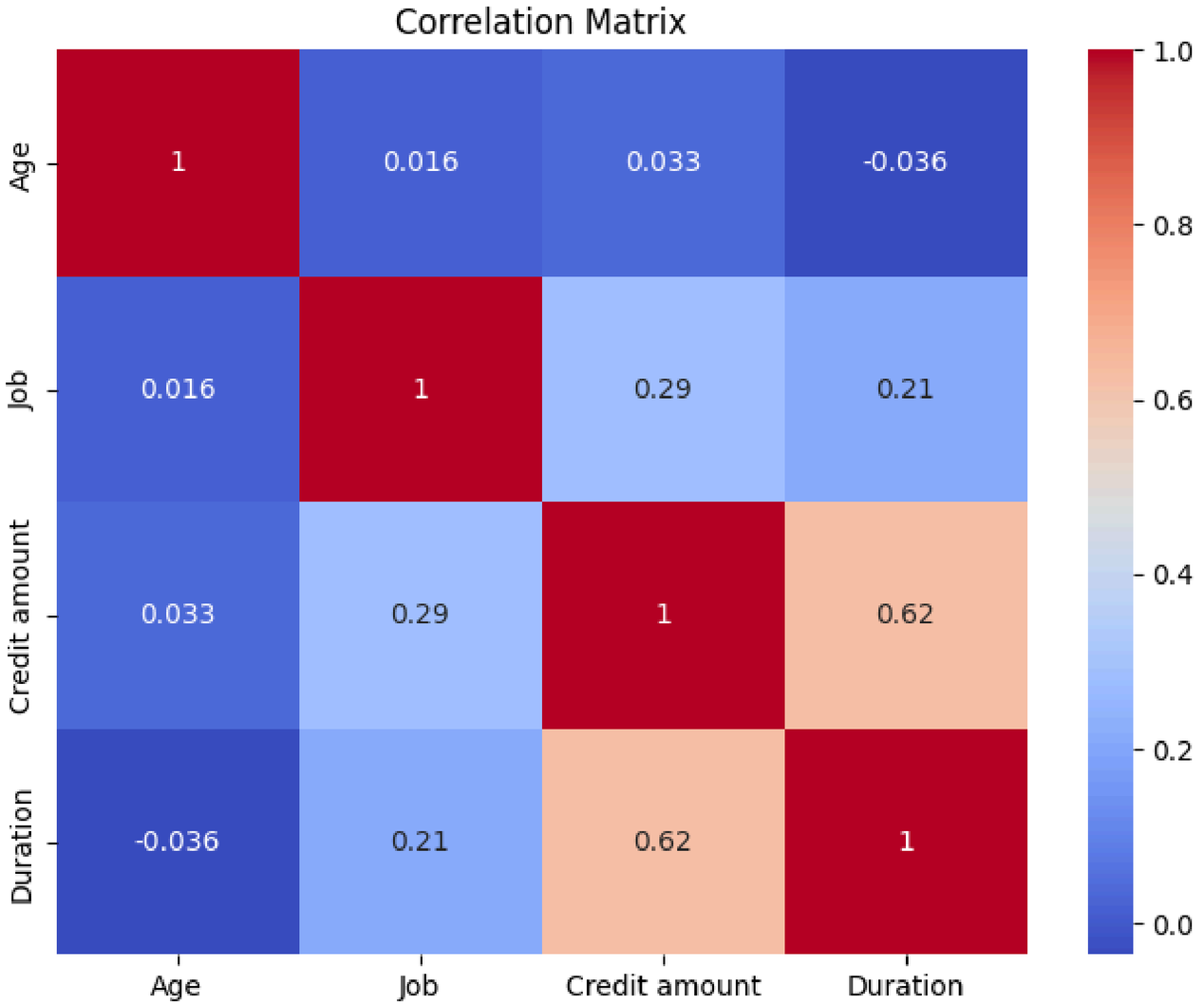
As the loan duration increases, the frequency drops significantly



This distribution is typical in consumer finance, where smaller loans are more common due to lower risk and easier approval.

Larger credit amounts are:

- Less common
- Likely require better creditworthiness



1. Loan Duration is most influenced by Credit Amount. Bigger loans usually need more time to repay.
2. Age doesn't seem to affect credit behavior much in this dataset.
3. Job type matters a bit when it comes to credit amount and duration, maybe more stable or higher-paying jobs get better terms.

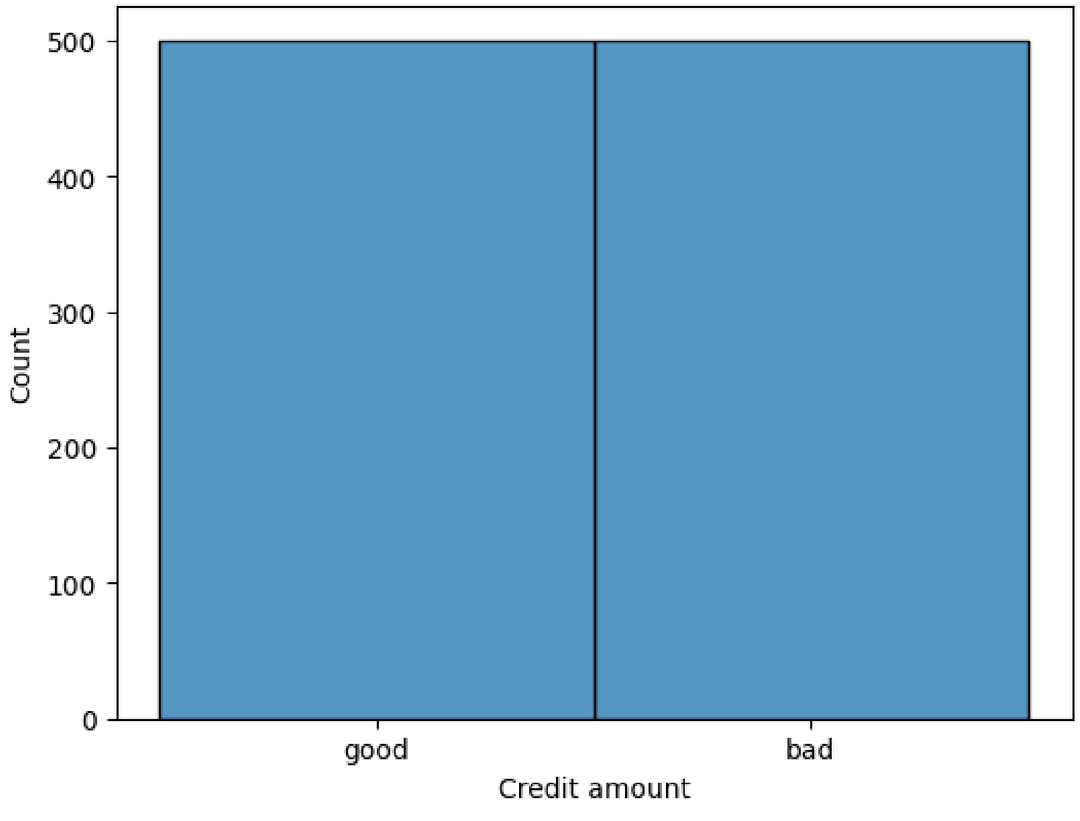
Credit amount is replaced with the values ‘good’ and ‘bad’ values based on Credit amount to represent good credit risk and bad credit risk.

$$\text{Let } \{x_i\}_{i=1}^{n} \text{ be your observed credit amounts.}$$

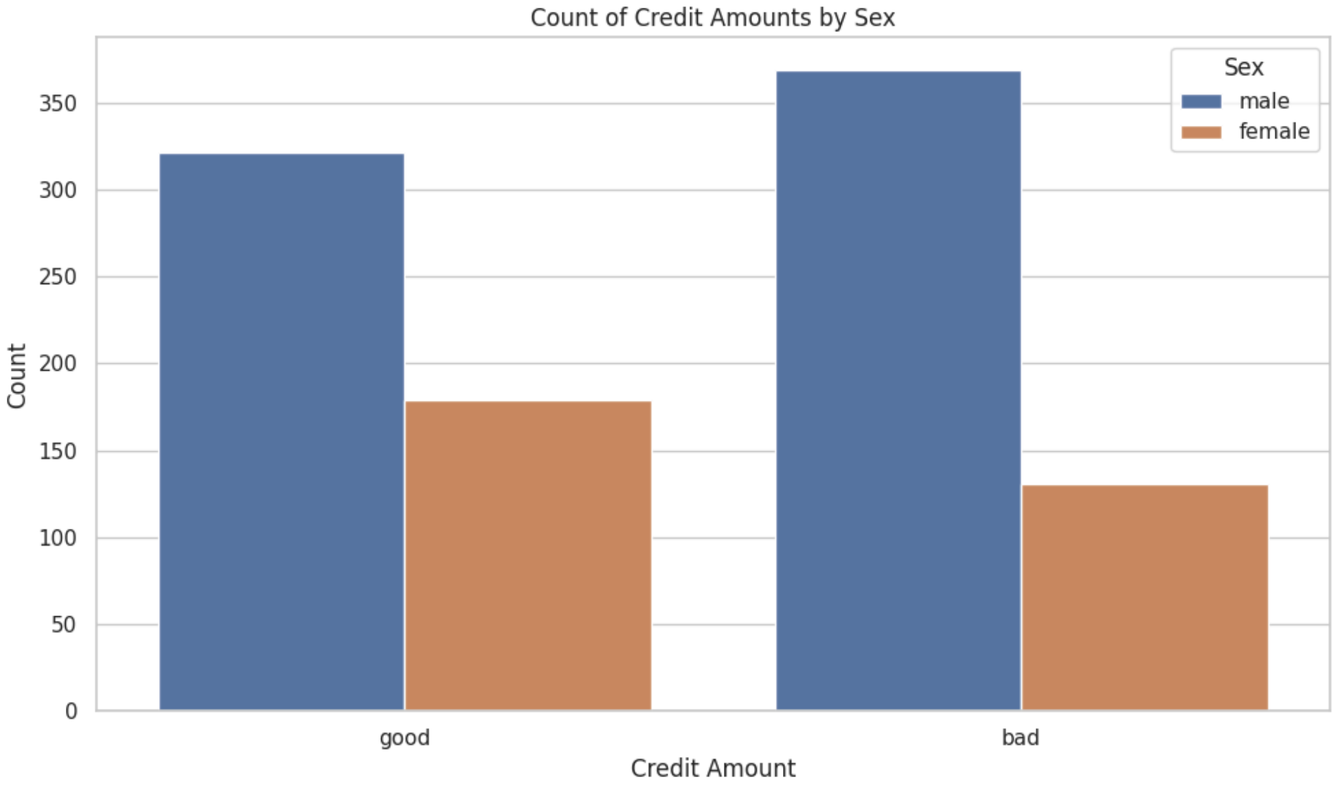
$$good = \{\, i : x_i < m \},\quad bad = \{\, i : x_i \geq m \}$$

$$\text{where m is median}$$

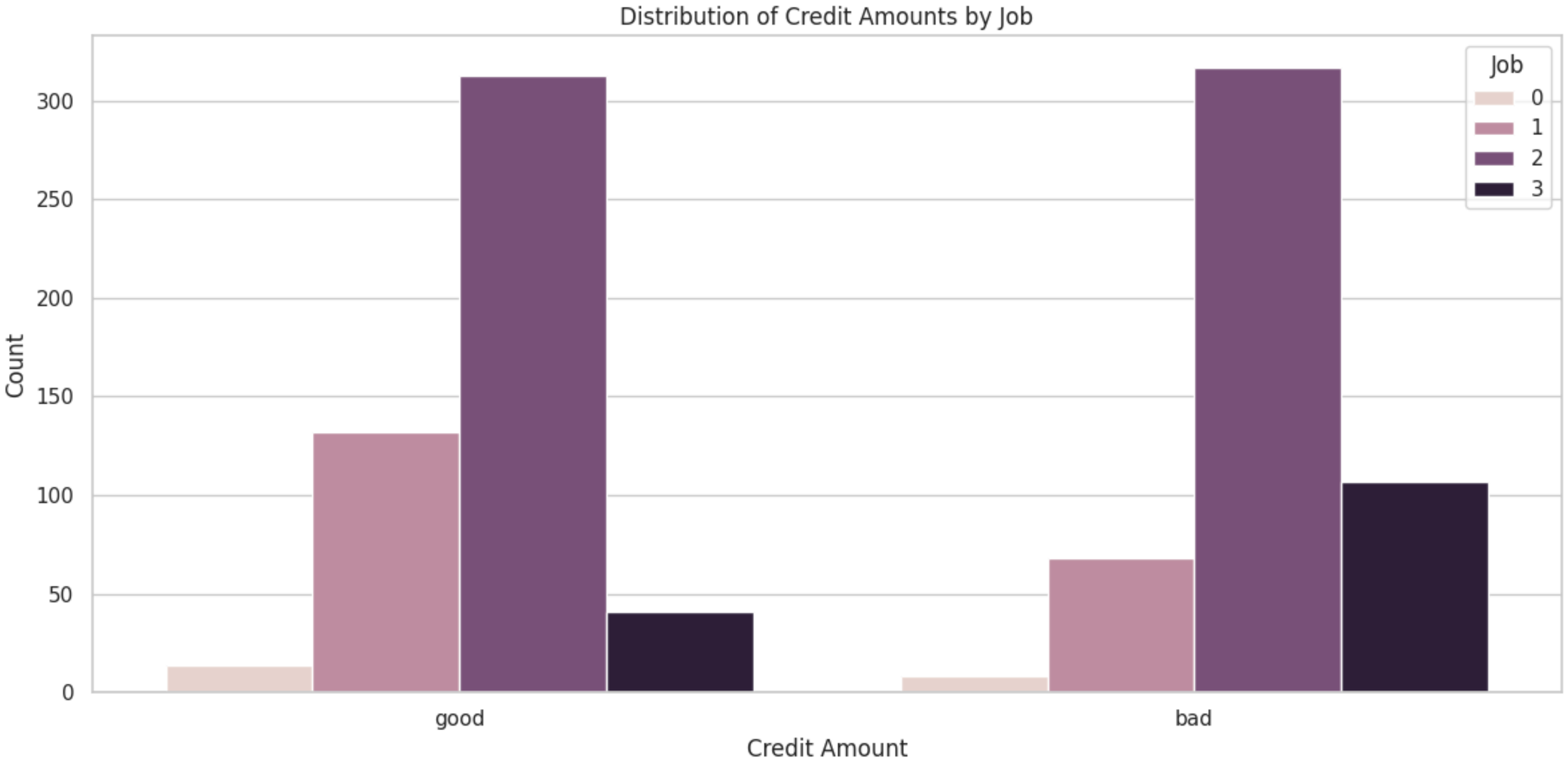
So all credit amounts above median is tagged as bad and below as good



Because of this type of tagging the classes are balanced

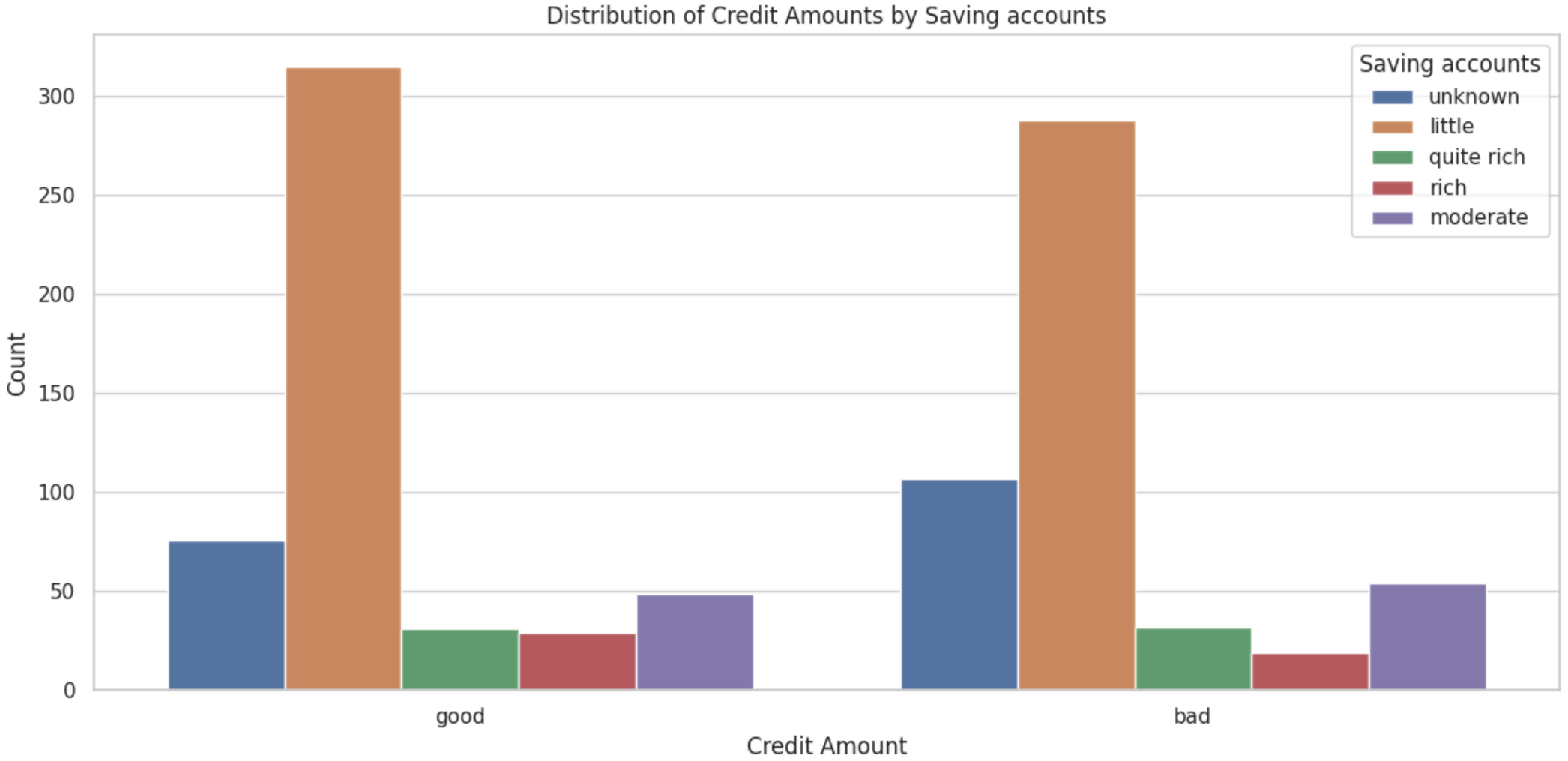


1. More males than females had good credit
2. More males also had bad credit, but the count of bad credit is noticeably higher than good for males



1. **The bad credit risk has more job 3 types (highly skilled jobs).**
2. **The good credit risk has more job 1 types (unskilled and resident jobs).**

1. **The bad credit risk has more free housing.**
2. **The good credit risk has slightly more own housing.**



1. **Good credit risk has more little savings.**
2. **Bad credit risk has more unknown values.**
3. **The rich tend to have a better credit score.**

Preprocessing the data

Encoding

- For the Sex column, I have mapped 'male' to 1 and 'female' to 0
- For the Credit amount column, I have mapped 'good' to 1 and 'bad' to 0
- For columns Saving accounts, Checking account, Housing, and Purpose, I have processed using One-Hot Encoding.

Splitting

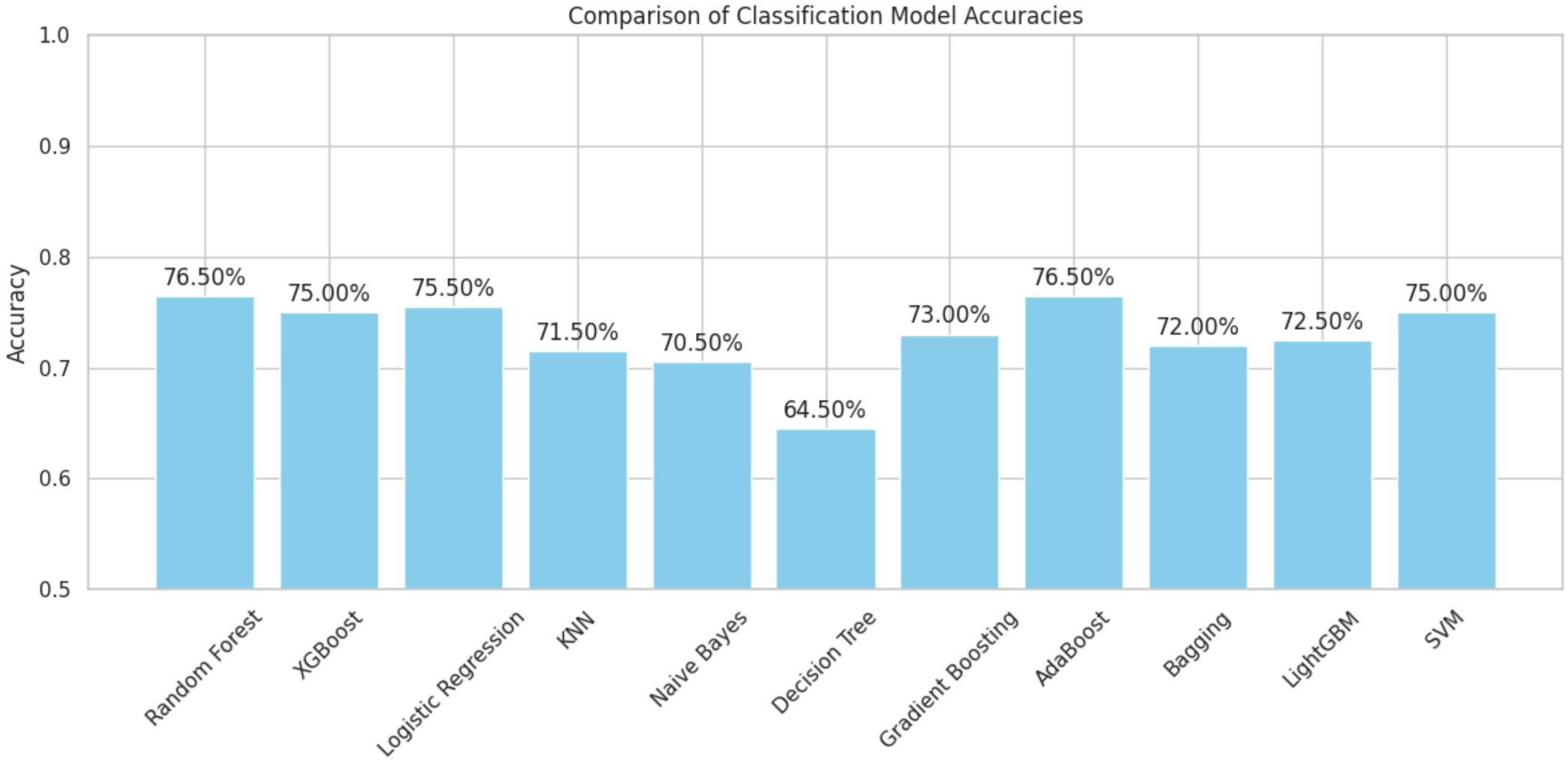
- Data split into X and y, where X is everything except Credit amount and y is Credit amount
- Then data is split to training and test set with 8:2 ratio

Model Building Stage

I trained and evaluated multiple classification models to determine the most accurate algorithm for our dataset. The models included Random Forest, XGBoost, Logistic Regression, K-Nearest Neighbors (KNN), Naive Bayes, Decision Tree, Gradient Boosting, AdaBoost, Bagging, LightGBM, and Support Vector Machine (SVM).

Each model was trained on the training dataset and evaluated on the test dataset using accuracy as the performance metric.

For models like XGBoost and Logistic Regression, hyperparameter tuning was performed using GridSearchCV to ensure optimal performance.



The highest performing models were AdaBoost and Random Forest, so the Random Forest Model is saved.

Links

You will find all the relevant documents in the Github.
You can find the hosted interface in the streamlit link.

Streamlit link: https://credit-risk-for-loan.streamlit.app/

Github Link: https://github.com/DL4150/Credit-Risk-for-Loan-Applicants

Colab Code Link:
https://colab.research.google.com/drive/1DqkOasBa4ykOP05PbxLLIQftcdxJGrHZ?usp=sharing

THANK YOU