BAT Bonus Assignment

Muhammad Sameer

19K-1526

8A

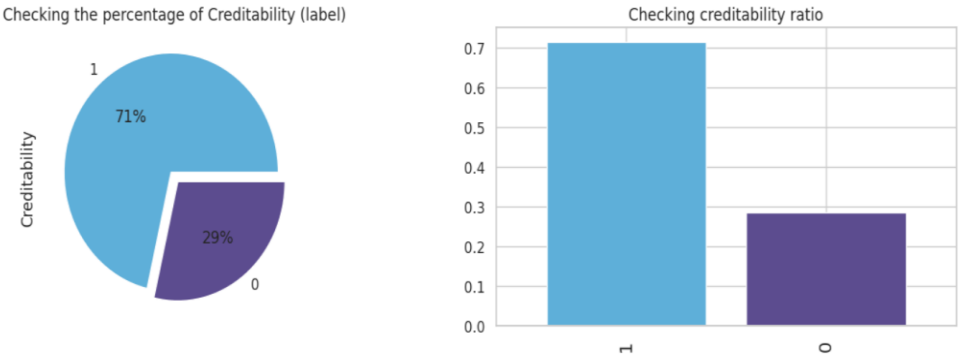
Exploratory Data Analysis (EDA)

To see the distribution of target column I have used piechart as well as countplot, it shows that majority of the examples belongs to class 1.

Interpretation ✍: We can see in piechart and barchart below, creditability for 1 has the highest ratio then that of creditability 0. Class 0 has very few examples in the data, it means our data is imbalanced we have to balance our data.

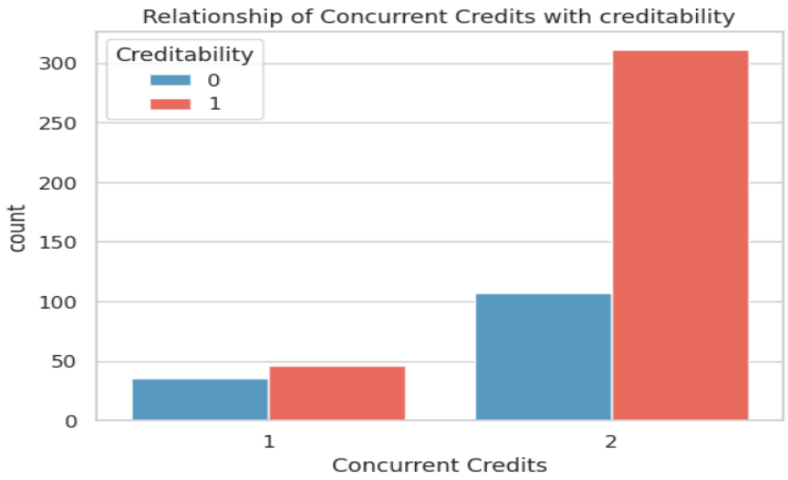
```
label_ratio(df_train,'Train',[my_colors[0],my_colors[4]])
```

```
======= Train data=======
```

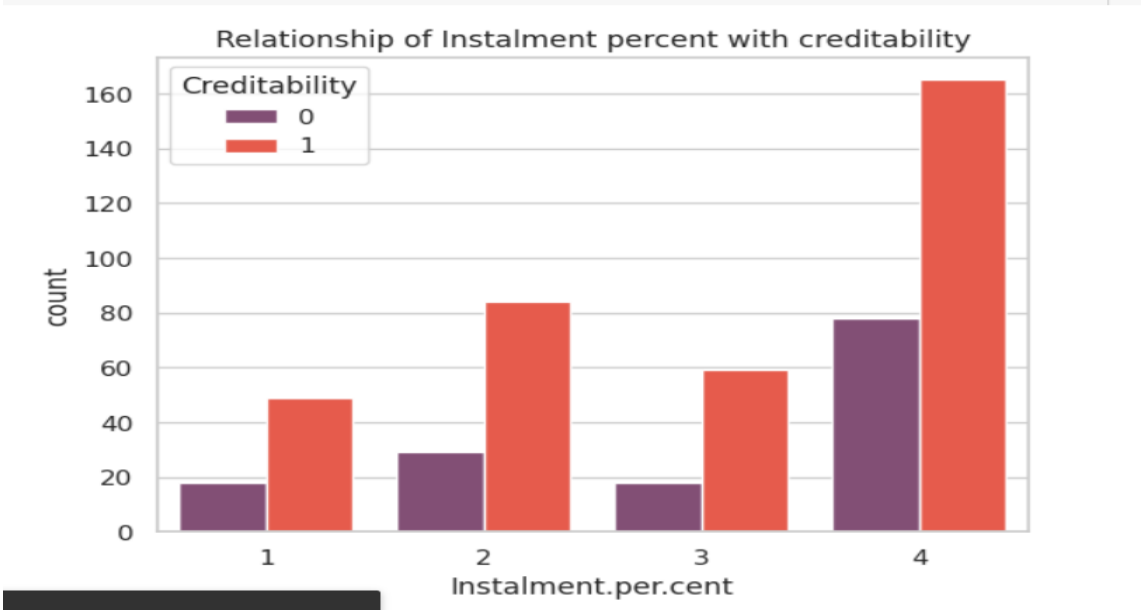


Below graph shows that value 2 of 'Concurrent.Credits' has a higher count for 'Creditability' value 1, it means that borrowers with 2 concurrent credits are more likely to be classified as creditworthy i.e., 'Creditability' value 1 compared to those with only one concurrent credit.

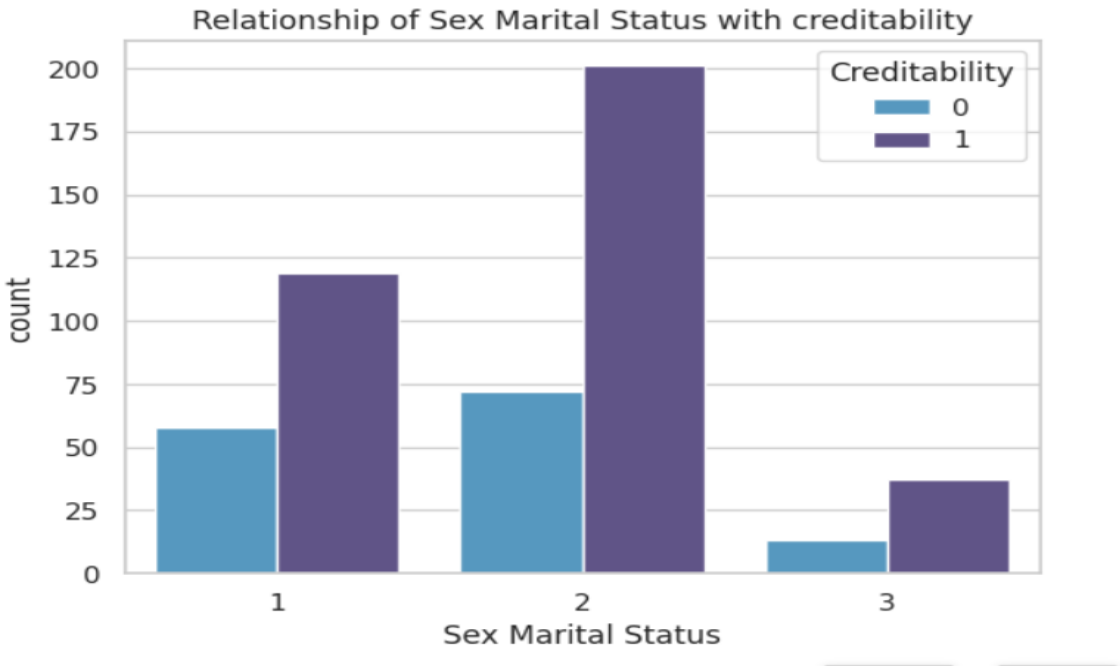
```
[354] plt.xlabel('Concurrent Credits')
      plt.ylabel('count')
      plt.title('Relationship of Concurrent Credits with creditability')
      plt.show()
```



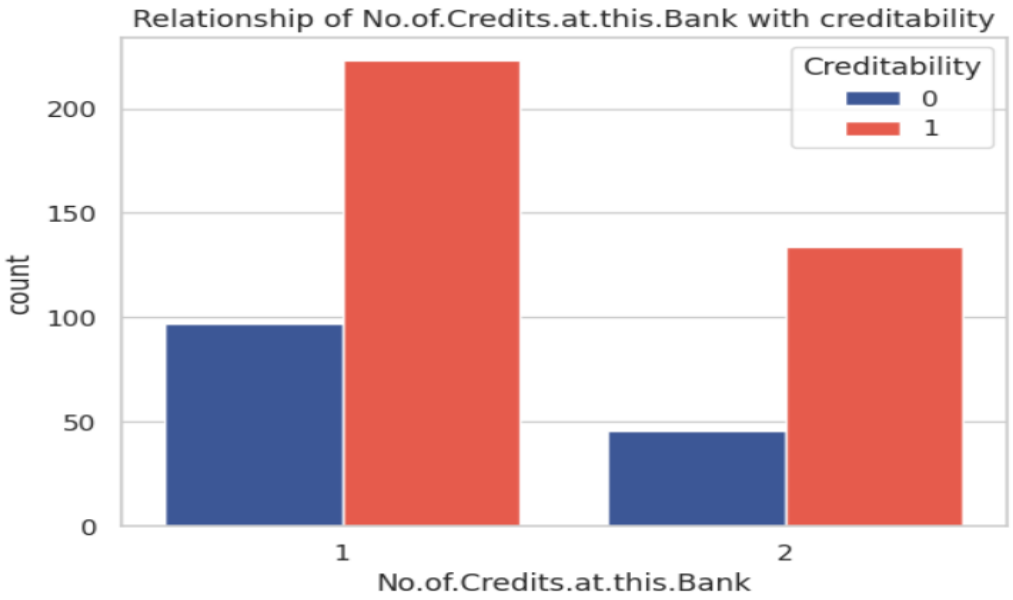
Below graph shows that Borrowers with higher instalment value are more likely to have good creditworthiness.



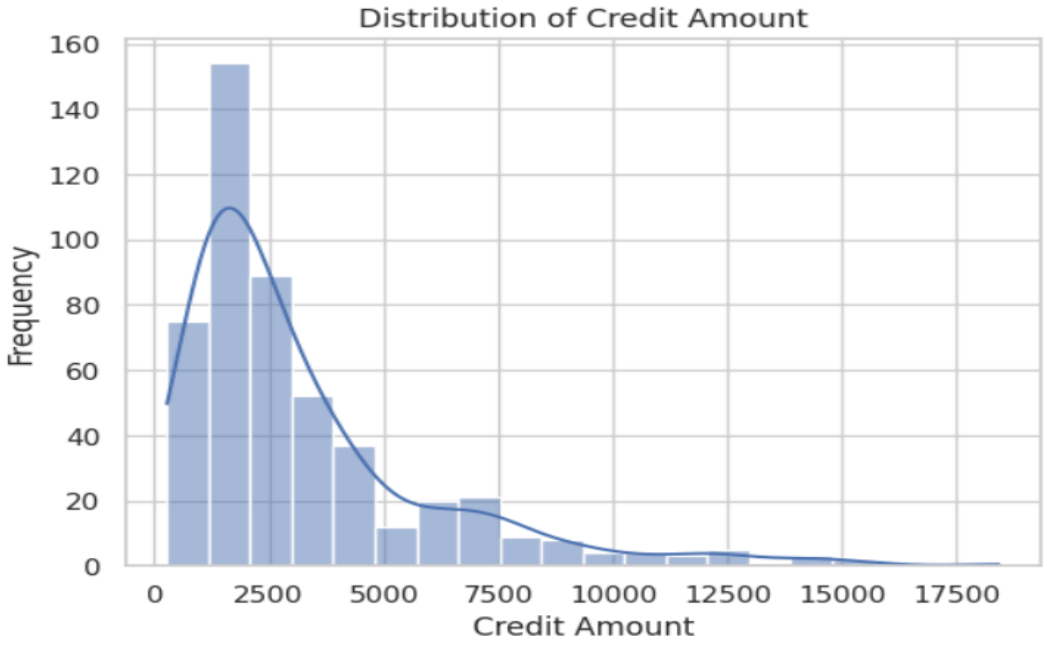
Below graph shows that Borrowers with value 2 of sex marital status are more likely to have good creditworthiness. (Interpretation is 1 belongs to ‘single’ class, 2 belongs to ‘married’ class and 3 belongs to ‘divorce’ class. This graph shows that borrowers who are married are more likely to have good creditworthiness.

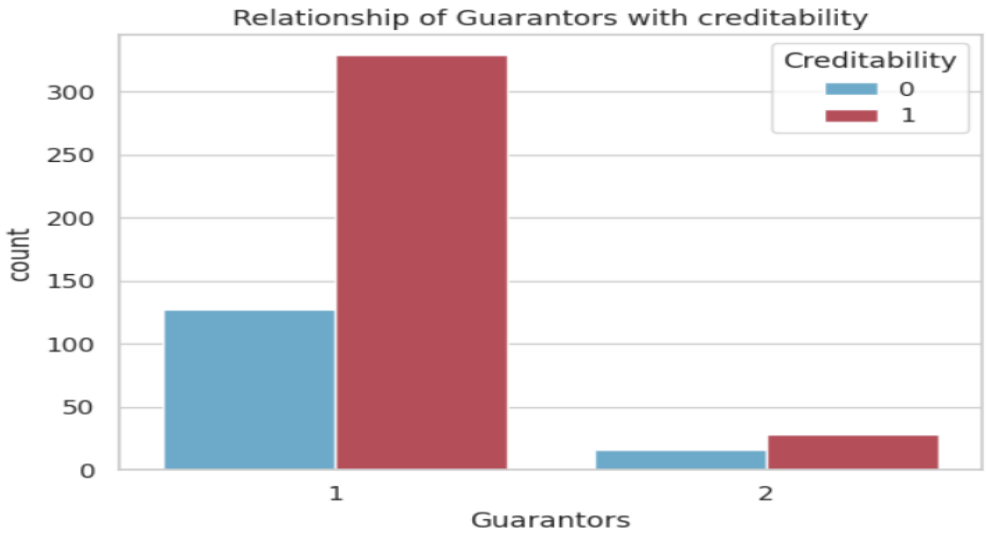


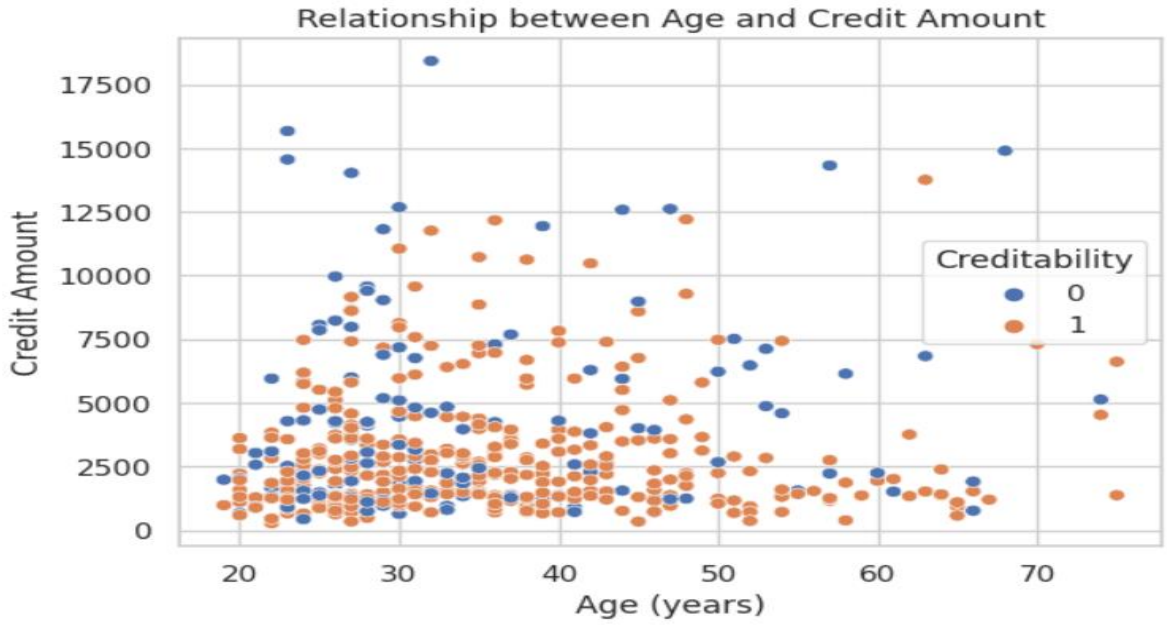
Below graph shows that Borrowers with value 1 of No of Credits at this Bank are more likely to have good creditworthiness.

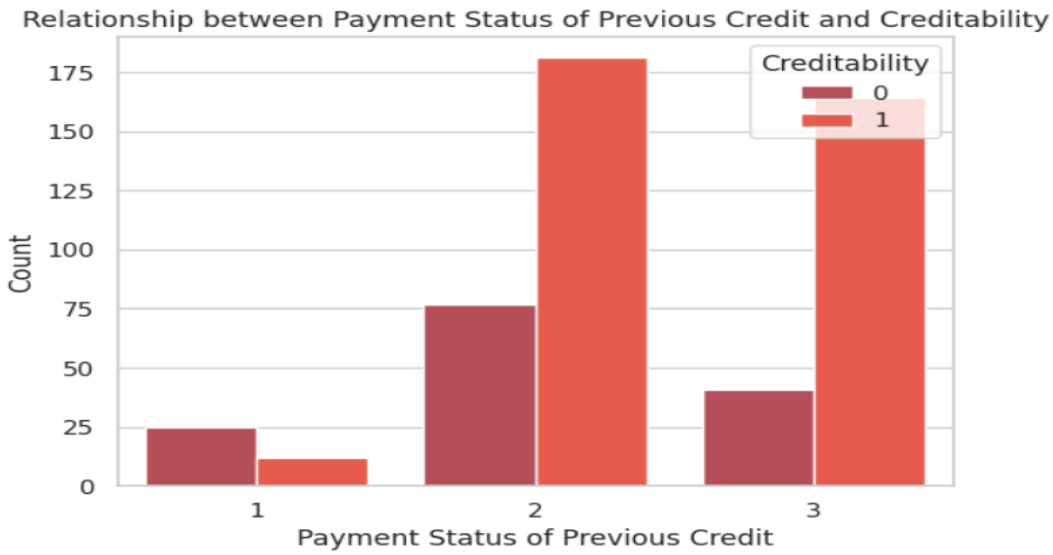


Other Visualization:




Relationship of Guarantors with creditability
Creditability
0
1
count
300
250
200
150
100
50
0
1
2
Guarantors

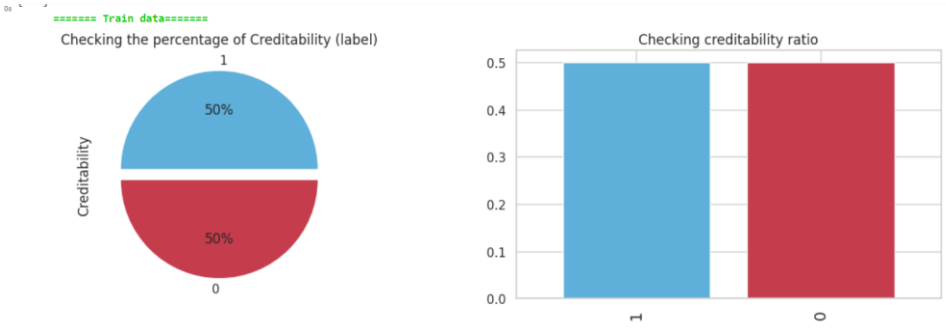

Relationship between Age and Credit Amount
Creditability
0
1
Credit Amount
17500
15000
12500
10000
7500
5000
2500
0
20
30
40
50
60
70
Age (years)



Feature engineering and Data Preprocessing

As we see in our very first graph that our data is imbalanced, we must balance it first before passing it to the classification model. To handle this problem, I have used a technique named as Oversampling which will bring minority class (0 in our case) equal to the majority class (1 in our case).

After balancing:



Following are the columns I have dropped:

i. Duration of credit month (I dropped this as it is highly related with credit amount feature which will rise the problem of multicollinearity. I have to select one column from these two so I pick credit amount as it is useful column.)
ii. Telephone (Not useful)
iii. Length of current employment (Highly related with age years column same problem of multicollinearity, also it is not that useful.)
iv. Duration in current address (Not useful)
v. Foreign Worker
vi. Occupation (contains only one value so it is good to drop it)
vii. No of dependents (Not useful)
viii. Type of apartment (correlated with age year column and with Most valuable available asset column)
ix. No of credit at this bank (correlated with Payment Status of Previous Credit column)

After that I created another column named ‘total asset’ by merging 2 columns that is ‘'Value Savings Stocks' and 'Most valuable available asset'. Both the column stores the value of asset and stock so I combined them as total asset.

After that I have scaled the credit amount column as it contains values with high range, to bring them in range of 0-1 I have used technique called StandardScaler.

Modeling

For modeling part, I have used 5 classification models and for evaluation of models I have used 2 evaluations metrics that are accuracy_score and f1_Score.

Following table below shows the results:

Model	**Training accuracy**	**Testing accuracy**	**F1 score**
XGBoost	0.99	0.716	0.797
Random Forest	0.90	0.722	0.789
Gradient Boosting	1.0	0.704	0.790
Logistic Regression	0.729	0.686	0.740
Decision Tree	1.0	0.66	0.748