DOCUMENTATION

The data that we have consisted of BAD data as following

BAD - 0 (4771), 1(1189)

Now my strategy is to bin the data using a suitable bin size and check for the WoE(Weight of Evidence) and the IV(Information Value) of the features of the data.

Hence in the following lines, we will discuss the WoE and IV for each feature:-

The code used for the following result in included in the IV.pynb file

For **CLAGE** I have used a step size of **55** while binning.

The information value of **CLAGE** - **0.23565985776546736.**

Hence **CLAGE** is a **medium predictor**.

For **YOJ** I have used a bin size of **10** while binning the data.

The information value of **YOJ** is **0.051870506679009706**.

Hence **YOJ** is a **weak predictor** which can be eliminated.

For **DEBTINC** I have used a bin size of **10** while binning the data.

The information value of **DEBTINC** is **1.87934101236577.**

Hence **DEBTINC** falls in the category of **too good to be true**.

For **DEROG** I have used a bin size of **1** while binning.

The IV is **0.3632220391752436.**

Hence **DEROG** is a **strong predictor**.

For **DELINQ** I have used a bin size of **1** while binning.

The IV is **0.4885257357648145.**

Hence **DELINQ** is a **strong predictor**.

For **VALUE** I have used a bin size of **43000** while binning

The IV is **0.44956887698919984**

Hence **VALUE** is a **strong predictor**

For **NINQ** I have used a bin size of **1** while binning

The IV is **0.17222973452562698**

Hence **NINQ** is a **medium predictor**

For **CLNO** I have used a bin size of **5** while binning

The IV is **0.08197862798796846**

Hence **CLNO** is a **weak predictor**

For **JOB**

The IV is **0.12373056571420771**

Hence **JOB** is a **medium predictor**

For **REASON**

The IV is **0.008618460238864022**

Hence **REASON** falls into the **useless for prediction** category

For **MORTDUE** I have used a bin size of **20,000** while binning

The IV is **0.07721976895021314**

Hence MORTDUE is a **weak predictor**

After getting the WoE table for each of the features as shown in IV.pynb I have combined the bins which have nearly the same values. The code for combining different bins also is present the IV.pynb file. I have combined the missing values bin to bins having a very close WoE with the missing values bin. After that, I replace all the values present in a particular bin with the WoE of that particular bin. In these ways, all the missing values are imputed and variables transformed.

We can clearly eliminate the features such as **YOJ, CLNO, REASON, MORTDUE** as these features were concluded to be weak predictors. To prove this fact I have tested my model(Logistic Regression) by taking this and without taking this. The model performs the same and the presence of the above-mentioned features does not help the prediction in any way.

MODEL:-

I have used a Logistic Regression Model for the prediction of the BAD variable. The Logistic Regression Model is chosen as it works really well with the WoE method of grouping data.

After training for 1000000000 iterations I have been able to achieve an accuracy of 88% on the validation split.

I consequently also tried the Support Vector Machine method and got an accuracy of 83% on the validation split.

VALUABLE TESTS TO CHECK THE STABILITY OF THE MODEL:-

1. **Confusion matrix:-** The confusion matrix gives us the number of false positives, number of true negatives, the number of true positives, the number of false negatives.

   False Positive = (The number of examples where the model predicts it as positive whereas actually, they are negatives)

   True Positive = (The number of examples where the model predicts it as positive and actually, they are positives)

   True Negative = (The number of examples where the model predicts it as negative and actually, they are negatives)

   False Negative = (The number of examples where the model predicts it as negative whereas actually, they are positives)

2. **Area Under ROC Curve:-** The Area under the Receiver Operating Characteristic Curve is a

good measure of the model.

0.90 - 1 (excellent)
0.80 - 0.90 (good)
0.70 - 0.80 (fair)
0.60 - 0.70 (poor)

3. **Concordant Discordant Ratio:-** This helps us in finding the model's predictive power. For a model if we have 60% concordant ratio then the model has good predictive power.

The LR.ipynb and the LR.py file consists of the Logistic regression model tested.

The SVM.ipynb and the SVM.py file consists of the SVM Model tested.

The IV.ipynb file consists of the IV and WoE displayed for each of the features.

The Rough_1.ipynb and the Rough_2.ipynb file consists of all the data visualizations and the initial ideas tested out there.

The good.csv file consists of the good data i.e it has no missing values and the WoE merged.