**Problem Statement**

**Loan Delinquency Prediction**

Loan default prediction is one of the most critical and crucial problem faced by financial institutions and organizations as it has a noteworthy effect on the profitability of these institutions. In recent years, there is a tremendous increase in the volume of non – performing loans which results in a jeopardizing effect on the growth of these institutions.

Therefore, to maintain a healthy portfolio, the banks put stringent monitoring and evaluation measures in place to ensure timely repayment of loans by borrowers. Despite these measures, a major proportion of loans become delinquent. Delinquency occurs when a borrower misses a payment against his/her loan.

Given the information like mortgage details, borrowers related details and payment details, our objective is to identify the delinquency status of loans for the next month given the delinquency status for the previous 12 months (in number of months)

**Approach:**

1. At first I tried to analysis all the data; I used very interesting tool pandas profiling to analyses the data.
2. I found that **number\_of\_borrowers** was highly co-related with **co-borrower\_credit\_score** so i deleted that column from train and test data.
3. I checked many date columns which was but could be considered as categorical columns as there were very few unique values
4. I found many outliers but I didn’t handle those as it may be a good indicator of Loan Delinquency
5. I plotted many scatter plot and histograms to check if I can get any intuition about new feature engineering
   1. I came to conclusion to try 2 features , sum of all M1 to M12
   2. **Total credit** score by adding **borrower\_credit\_score** and **co-borrower\_credit\_score**
6. I found that the data is very highly imbalance so decided to go with under sampling and oversampling.

**Model :**

1. At first I have to accept that all my models was overfitting and was giving zero score due to one of my mistake.
2. I then tried look into the data by applying PCA and try and visualize the data.
3. Then I tried to apply oversampling SMOTE which gave to around 0.3 in LB
4. I applied SMOTE followed by TOMEK link removal which gave me almost similar result.
5. I tried many variant like SMOTE followed by TOMEK link , SMOTE Followed by ENN , SMOTE followed by NeighbourhoodCleaningRule.
6. I also tried providing aggregate features if there is any features which is important.
7. After trying many variant I decided to keep the simple model with SMOTE SVM followed by TOMEK link removal and changing the threshold to 0.4 which gave me 0.34 on LB.

**Final model:**

1. I used no additional features, used SMOTE SVM followed by TOMEK Link removal.
2. Used boosting algorithm LightGBM with low **min\_data\_in\_leaf** and high depth (as it was giving good results)

**Key Takeaway:**

1. Understand how your data looks like, apply relevant techniques which will work on that data.
2. Try and keep your model simple enough and always check the model output and try and analyses the result.
3. Try and generalize your model instead of looking for LB score.

**Top 5 things to focus on:**

1. Try and generalize your model instead of looking for LB score.
2. Understand your data
3. If you are not aware then do proper research from community on techniques used in such cases which will work like using imbalance library or using low ***min\_data\_in\_leaf*** for highly imbalance class.
4. Find ways to analyses the model after applying it , use tools like SHAP or feature importance.
5. Don’t give up until end , keep working on the problem unless you are satisfied ☺