

Micro-Credit Defaulter Model

Submitted by:

Sourov Sahoo

**ACKNOWLEDGMENT**

**Guidance personals:-**

1. DataTrained Deepika mam

2. DataTrained Vishal sir.

3. Fliprobo Astha mam.

**Literature study:-**

<https://pypi.org/project/sweetviz/>

<https://www.youtube.com/watch?v=YMPMZmlH5Bo&t=3s>

<https://www.youtube.com/watch?v=OJedgzdipC0>

<https://github.com/sourovsahoo/Evaluation-Projects-11-to-15/blob/master/Project-11-correction%201-Automobile%20insurance%20fraud%20%26%20claims..ipynb>

**INTRODUCTION**

* Business Problem Framing

The problem in the business is the defaulter category (0) which is the main cause of leakage of revenue.

In this data set of 9months the company have incurred a loss of **148956** Indonesian Rupiah.(considered Zero’s in feature”amnt\_loans90” are customers who never paid or might have also left the network)

I have explained this in the jupyter note book better with full codes.

**Solution is**:- We can stop giving credit to such defaulter customers

Or we can offer them some other deal on trial basis if the company wants.

My model can predict the customer will be a defaulter (0) or non-defaulter (1) by analyzing the customers past transaction records .Hence we find ways to deal with such customer more effectively and try to retain the customer.

e.g.-We can increase the interest rate for such customers so as to recover old amount.

Case history🡪Insurance fraud/not dataset.

<https://github.com/sourovsahoo/Evaluation-Projects-11-to-15/blob/master/Project-11-correction%201-Automobile%20insurance%20fraud%20%26%20claims..ipynb>

Conceptual Background of the Domain Problem

If a customer who has paid the loan amount is falsely marked as defaulter (0).This would not be much problem to the company but if a defaulter (0) is not detected and appropriate measures were not taken by the company then it will result in loss of revenue.

This model can surely help to know overcome the above problem also. We can know our customer properly beforehand and then deal our offers.

It’s rightly said “prevention is better than cure”.

* Review of Literature

Data is imbalanced, balancing the dataset is main concept here.

<https://www.youtube.com/watch?v=YMPMZmlH5Bo>

<https://www.youtube.com/watch?v=OJedgzdipC0>

* Motivation for the Problem Undertaken

Revenue is what is motivation and the base of any business.

In this project I am trying to minimize the revenue leakage in the form of defaulter customer.

”Every customer is important”. So taking precaution before retaining such customers who all from now on will be treated differently by some other credit plans.

Reference:-

<https://github.com/sourovsahoo/Evaluation-Projects-11-to-15/blob/master/Project-11-correction%201-Automobile%20insurance%20fraud%20%26%20claims..ipynb>

**Analytical Problem Framing**

* Mathematical/ Analytical Modelling of the Problem

1) Feature generation technique used on pdate feature.  
2) Removing 'I' from 'msisdn' column  
3) Adding 25000 to all numerical feature to cancel out the max -ve value in the dataset (-24720.580000-rental90), then i will take log1p(). This will not affect the distribution of the dataset.  
4) Over sampling (RandomOverSampler) as it is faster than SMOTETomek.

* Data Sources and their formats

Primarily data was in .csv file format with no missing values in the data set. The dataset shape is Rows🡪209593

Columns/features🡪36

Out of 36 feature 35 are independent and one is dependent variable (label).

I deleted a unknown extra column in the csv format and then exported to jupyter notebook.

Count of both categories are:-

Defaulter (0)-26162

Non defaulter (1)-183431

Clearly it’s a imbalanced data. I have applied oversampling technique to overcome such situation.

Data Pre-processing Done

1) EDA by sweetviz library.

2) Removing unwanted rows for prediction(customers with no credit history past 90days)

3) Feature Engineering/ generation from “pdate” feature.

4) Removing ‘pdate’ feature as we have extracted our required data.  
5) Removing 'I' from 'msisdn' column and converting it into integer.

\*Then Dropping the original feature’msisidn’.New feature “new\_msisdn”

6) Adding 25000 to all numerical feature to cancel out the max -ve value in the dataset (-24720.580000-rental90), then i will take log1p().This will not affect the distribution of the dataset.  
7) Over sampling (RandomOverSampler) as it is faster than SMOTETomek.  
8) Removing skewness.

9) Applying StandardScaler.

10) Applying XGBoostClassifier here, And checking scores.  
11) Finding best random state for XGBoostClassifier  
12) GridsearchCV on XGBoostClassifier  
13) Cross validating the model.  
14) Saving the final model in joblib-->'Micro-Credit Defaulter.obj'  
15) Loading the model  
16) Testing the model on test data.  
17) Saving final predictions in file.csv format  
18) Conclusion

* Data Inputs- Logic- Output Relationships

The Xgboost classifier algorithm Used here is a ensemble technique boosting algorithm where the models (tree)are built sequentially upon the errors of the previous model. The final model is taken for hypertuning and saved in .obj format.

The predictions on test data set are saved in local drive in .csv format.

* State the set of assumptions (if any) related to the problem under consideration

Assumption -Max value of ‘aon’ feature is absurd (2739years)this is not possible in years. I have assumed it exists and carried on. So starting year of the company is a missing value here.We can remove some absurd data and make our model more generic.

* Hardware and Software Requirements and Tools Used

**Hardware**-64bit, 12GB RAM, 240GB SSD.

**Software-**Excel, Anaconda,jupyter notebook,python 3.6

Libraries used:-

1. numpy

2. pandas

3. matplotlib

4. sweetviz

5. seaborn

6. sklearn

7. Xgboost

8. joblib

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

Data balancing technique (RandomoverSampler). Scores improved drastically after this approach.

* Testing of Identified Approaches (Algorithms)

1. StandardScaler

2. KNN

3. Logistic regression

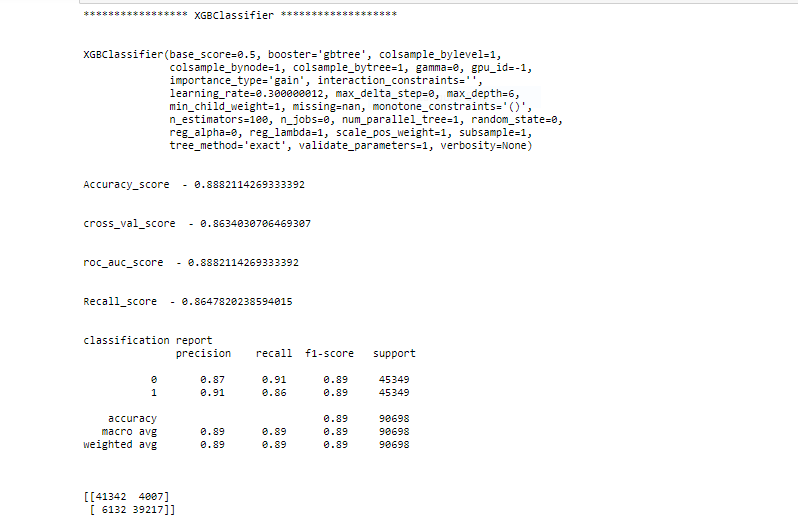
4. Xgboost

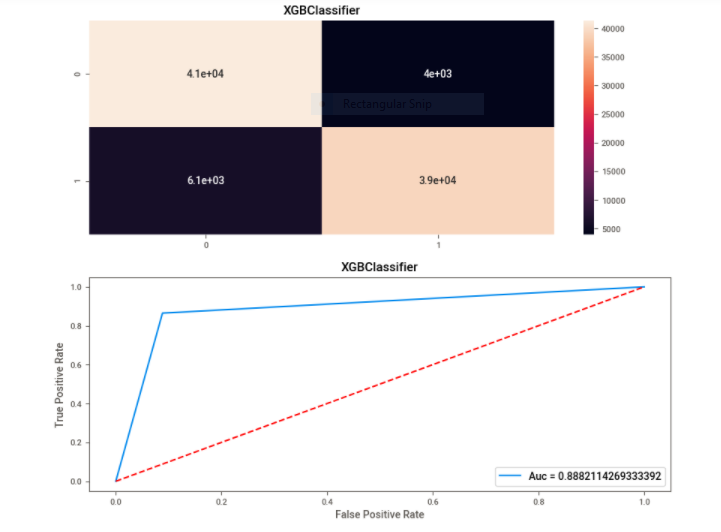
5. Gradient boosting machine(GBM)

6. Log transformation technique used to reduce skewness.

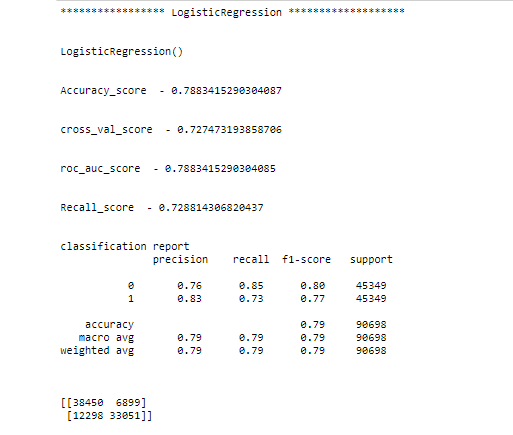
* Run and Evaluate selected models

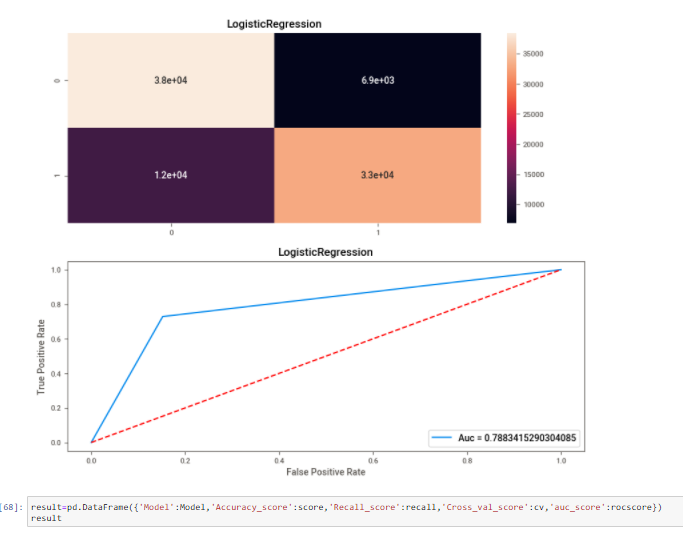
1.Xgboost:-



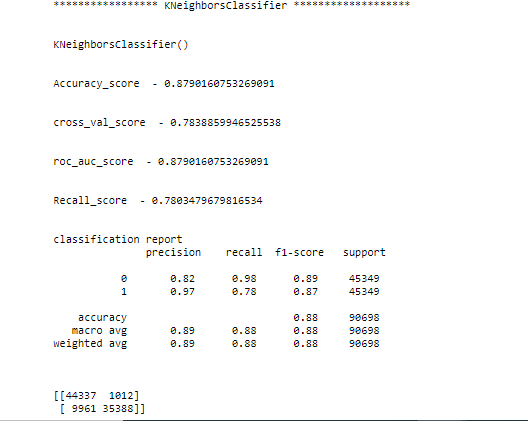


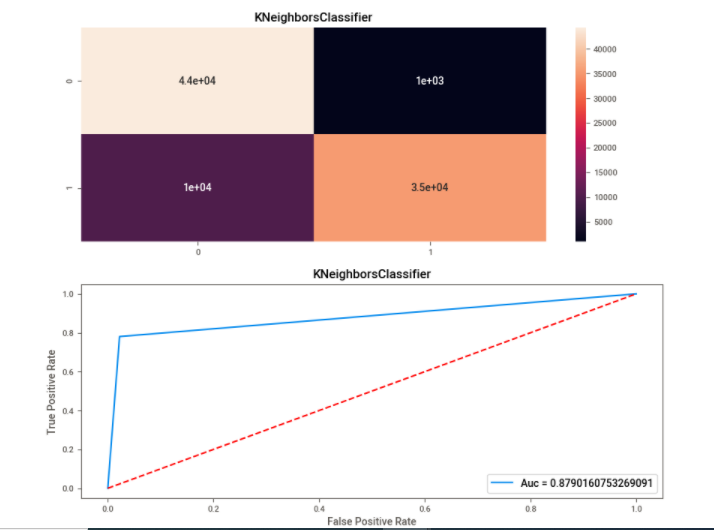
2.Logistic regression:-





3.KNN:-





From All the Algorithm we can see XgboostClassifier is working the best as the recall scores for class 0 & 1 are upto the mark.

* Key Metrics for success in solving problem under consideration.

Here we have to predict the ‘label’, whether the customer returned the loan within 5days(1) or not(0) . This is a binary class classification problem,

We could use the following 2 popular metrics:

1:Recall

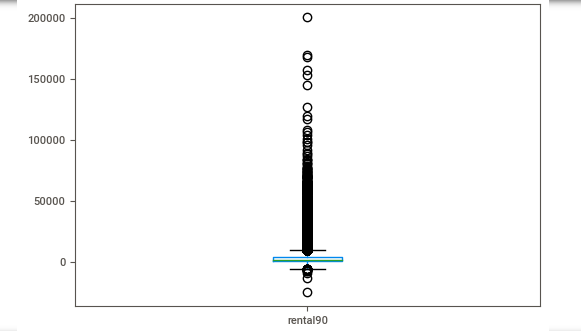
2:AUC-ROC(Area under the receiver operating characteristic curve)

The ROC\_AUC is the area under the curve when plotting the(normalized)true positive rate(x-axis) and the false positive rate(y-axis) Our main metric here would be Recall value, while AUC-ROC score would take care of how well predicted probabilities are able to differentiate between the two classes Here ,we are looking at the recall value because,

If a customer who has paid the loan amount is falsely marked as defaulter(0) that would not be much problem to the company but if a defaulter(0) is not detected and appropriate measures were not taken by the company then it will result in loss of revenue.

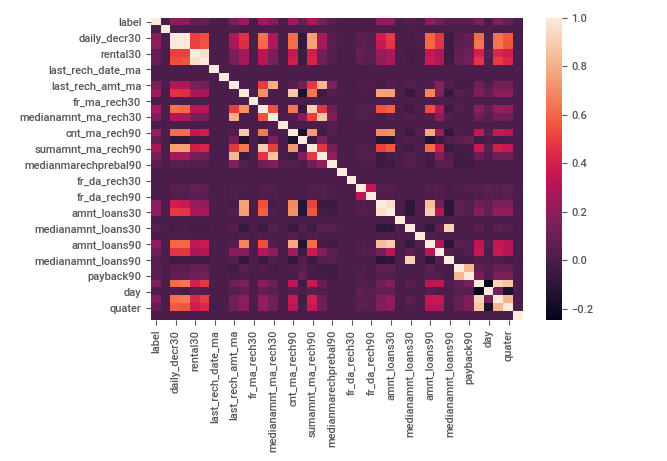
.# We are to predict the customer is defaulter-0 or Paid-1

* Visualizations



We can see –ve data outliers, these have to be treated. These –ve data are erroneous data as rental90 cannot be negative.

Important features can be seen below:-



* Interpretation of the Results

We can see –ve data outliers, these have to be treated. These –ve data are erroneous data as rental90 cannot be negative.

Pre-processing:-

1) EDA by sweetviz library.

2) Removing unwanted rows for prediction(customers with no credit history past 90days)

3) Feature Engineering/ generation from “pdate” feature.

4) Removing ‘pdate’ feature as we have extracted our required data.  
5) Removing 'I' from 'msisdn' column and converting it into integer.

\*Then Dropping the original feature’msisidn’.New feature “new\_msisdn”

6) Adding 25000 to all numerical feature to cancel out the max -ve value in the dataset (-24720.580000-rental90), then i will take log1p().This will not affect the distribution of the dataset.  
7) Over sampling (RandomOverSampler) as it is faster than SMOTETomek.  
8) Removing skewness.

9) Applying StandardScaler.

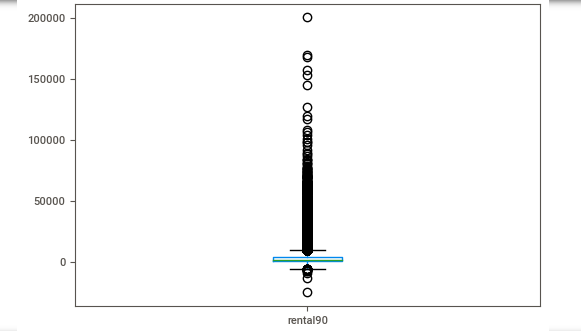
Best performing (XGBoostClassifier):-

10) Applying XGBoostClassifier here, And checking scores.  
11) Finding best random state for XGBoostClassifier  
12) GridsearchCV on XGBoostClassifier  
13) Cross validating the model.  
14) Saving the final model in joblib-->'Micro-Credit Defaulter.obj'  
15) Loading the model  
16) testing the model on test data.  
17) Saving final predictions in file.csv format  
18) Conclusion

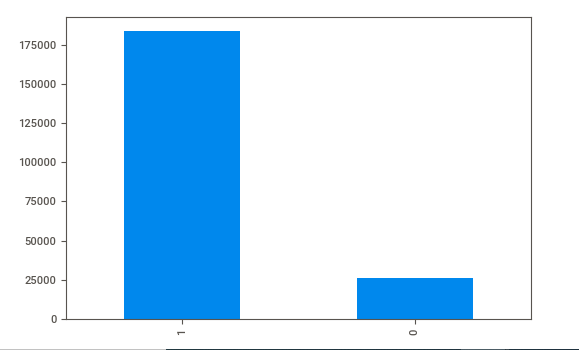
**CONCLUSION**

* Key Findings and Conclusions of the Study

1. We can see –ve data outliers, these have to be treated. These –ve data are erroneous data as rental90 cannot be negative.



2. Imbalanced dataset:-



* Learning Outcomes of the Study in respect of Data Science

New things I implemented here is the sweetviz library for EDA process,this automated process really saves a lot of time.

Balancing the data by Upsampling technique (RandomOverSampler).

* Limitations of this work. And Scope for Future Work

Computational complexity:

Model building and GridsearchCV takes too much time for algorithms like KNN and SVC.

Hardware problem:-

Need more powerful system, but this time managed by working more time. My maximum time went in GridsearchCV and model building which is just opposite for a data scientist working hours.

With a upgraded system next time I will be concentrating and spending more time in EDA and data analysis, finding out some insights within data.

There is much EDA to be done with data and find some business solution to contribute towards the revenue model of the company.

**Problems I faced during project**

Unbalanced data set.

I could not perform SVC algorithm as gridsearchcv took lot of time(>15hours)

I could not perform GBM algorithm as gridsearchcv took lot of time(>15hours)

I could not play around the values while hypertuning due to low computational power.

Maximum time went in computing rather than analysis of the data.

**Future works**

1. Analysing the graphs obtained from sweetviz library more deeply.

2. Do outlier removal, there are very few columns, one by one analysis can also be done.

3. Try different skewness removal technique.

4. Try different combinations for hypertuning.