

MICRO CREDIT DEFAULTER

Submitted by:

D REKHA VIDHYA

**ACKNOWLEDGMENT**

This includes mentioning of all the references, research papers, data sources, professionals and other resources that helped you and guided you in completion of the project.

1.Challenges faced by Microfinance Institution

<https://www.nelito.com/blog/challenges-faced-microfinance-institutions.html>

2.Microfinance

<https://www.investopedia.com/terms/m/microfinance.asp>

3.Predictors of Microfinance default in Indian self-help groups

<https://onlinelibrary.wiley.com/doi/10.1111/apce.12259>

**INTRODUCTION**

* Business Problem Framing

A Microfinance Institution (MFI) is an organization that offers financial services to low income populations. MFS becomes very useful when targeting especially the unbanked poor families living in remote areas with not much sources of income.

We are working with one such client that is in Telecom Industry. They are a fixed wireless telecommunications network provider. They have launched various products and have developed its business and organization based on the budget operator model, offering better products at Lower Prices to all value conscious customers and who takes help from MFI to provide micro credit on mobile balances and gives consumer 5 days’ time to repay MFI. If the consumer fails to repay then they will be charged Rupiah 6 for 5 credit amount and 12 for 10.

In order to improve the selection of customers for the credit, the client wants some predictions that could help them in further investment and improvement in selection of customers.

* Conceptual Background of the Domain Problem

MFI – Micro Finance Institutions

MFS – Mobile Financial Services

* Review of Literature

This is a comprehensive summary of the research done on the topic. The review should enumerate, describe, summarize, evaluate and clarify the research done.

* Motivation for the Problem Undertaken:

This is to provide the MFI the right consumers they can reach to reduce financial risk involved of not getting Paid back for the credit amounts given

**Analytical Problem Framing**

* **Mathematical/ Analytical Modeling of the Problem**

The request from Client is to build a model for the MSI industry to predict the customers who will repay the loan amounts with 5 days of the credit received so that they can reduce the Financial risks involved in losing the money lent.

* **Data Sources and their formats**

Sample data set has been provided to us by the client from Indonesia and expectation is to create a model that can help them to predict the loan repayment status of a customer based on the factors given in the data set.

|  |  |
| --- | --- |
| **Label** | Flag indicating whether the user paid back the credit amount within 5 days of issuing the loan{1:success, 0:failure} |
| **Msisdn** | mobile number of user |
| **Aon** | age on cellular network in days |
| **daily\_decr30** | Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah) |
| **daily\_decr90** | Daily amount spent from main account, averaged over last 90 days (in Indonesian Rupiah) |
| **rental30** | Average main account balance over last 30 days |
| **rental90** | Average main account balance over last 90 days |
| **last\_rech\_date\_ma** | Number of days till last recharge of main account |
| **last\_rech\_date\_da** | Number of days till last recharge of data account |
| **last\_rech\_amt\_ma** | Amount of last recharge of main account (in Indonesian Rupiah) |
| **cnt\_ma\_rech30** | Number of times main account got recharged in last 30 days |
| **fr\_ma\_rech30** | Frequency of main account recharged in last 30 days |
| **sumamnt\_ma\_rech30** | Total amount of recharge in main account over last 30 days (in Indonesian Rupiah) |
| **medianamnt\_ma\_rech30** | Median of amount of recharges done in main account over last 30 days at user level (in Indonesian Rupiah) |
| **medianmarechprebal30** | Median of main account balance just before recharge in last 30 days at user level (in Indonesian Rupiah) |
| **cnt\_ma\_rech90** | Number of times main account got recharged in last 90 days |
| **fr\_ma\_rech90** | Frequency of main account recharged in last 90 days |
| **sumamnt\_ma\_rech90** | Total amount of recharge in main account over last 90 days (in Indonasian Rupiah) |
| **medianamnt\_ma\_rech90** | Median of amount of recharges done in main account over last 90 days at user level (in Indonasian Rupiah) |
| **medianmarechprebal90** | Median of main account balance just before recharge in last 90 days at user level (in Indonasian Rupiah) |
| **cnt\_da\_rech30** | Number of times data account got recharged in last 30 days |
| **fr\_da\_rech30** | Frequency of data account recharged in last 30 days |
| **cnt\_da\_rech90** | Number of times data account got recharged in last 90 days |
| **fr\_da\_rech90** | Frequency of data account recharged in last 90 days |
| **cnt\_loans30** | Number of loans taken by user in last 30 days |
| **amnt\_loans30** | Total amount of loans taken by user in last 30 days |
| **maxamnt\_loans30** | maximum amount of loan taken by the user in last 30 days |
| **medianamnt\_loans30** | Median of amounts of loan taken by the user in last 30 days |
| **cnt\_loans90** | Number of loans taken by user in last 90 days |
| **amnt\_loans90** | Total amount of loans taken by user in last 90 days |
| **maxamnt\_loans90** | maximum amount of loan taken by the user in last 90 days |
| **medianamnt\_loans90** | Median of amounts of loan taken by the user in last 90 days |
| **payback30** | Average payback time in days over last 30 days |
| **payback90** | Average payback time in days over last 90 days |
| **Pcircle** | telecom circle |
| **Pdate** | date |

* **Data Preprocessing Done**

**Steps done in Data preprocessing are**

1. Importing Libraries
2. Read the provided csv file
3. Check for any duplicates or null values in the dataset
4. Check for outliers
5. Check Skewness

* **Data Inputs- Logic- Output Relationships**

The Target variable here is the loan repayment status and the inputs in the Data set will be details like frequency of recharge, count of recharge, Recharge amount, Loan amount taken, amount spent on an average in 30 and 90 days etc.

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

The dataset provided has no null values, has no duplicates and it’s a Categorical dataset.

* **Hardware**

1. Modern Operating System:
2. x86 64-bit CPU (Intel / AMD architecture)
3. 4 GB RAM.
4. 5 GB free disk space

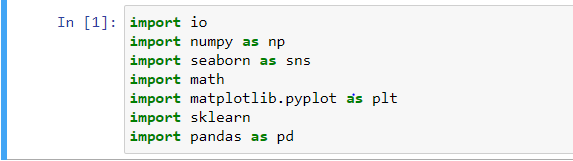
* **Software Requirements and Tools Used**

1. Anaconda Navigator
2. Launch Jupyter Notebook
3. Import Libraries

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

1.Importing Python Libraries

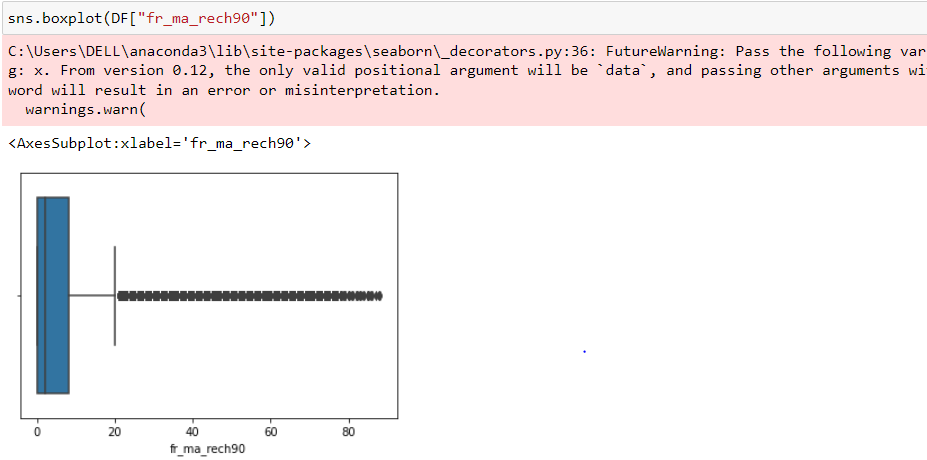


2.Read the csv file provided using read\_csv method

3.Check for the datatypes of the fields as ML can only understand Numeric data and cannot process Object datatype

4.Check for any missing values in dataset

5. **Check for outliers using boxplot** – Treat it with Z score if necessary. For this dataset we could see that there is a loss of 15% of data while trying to treat outliers whereas the project requirement allows only upto 8% data loss. So no outlier treatment performed.



6.check for class if its Balanced or imbalanced using count plot

7. check for skewness using Distplot from seaborn library and treat the skewness if necessary

8.Splitting the DataFrame into dependent and Independent Variable

9.Scaling the Data using Standard Scalar

* **Testing of Identified Approaches (Algorithms)**

**Ensemble Techniques used are:**

1.Logistic Classification

2.RF classifier

3.DTC

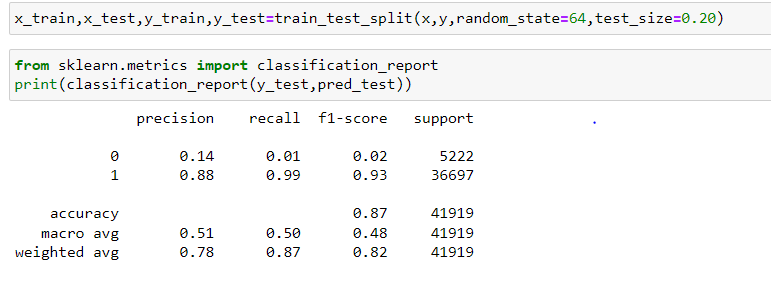
4.SVC

* **Run and Evaluate selected models**

**1.Logistic Regression-**

Logistic regression, despite its name, is a linear model for classification rather than regression. Logistic regression is also known in the literature as logit regression, maximum-entropy classification (MaxEnt) or the log-linear classifier. In this model, the probabilities describing the possible outcomes of a single trial are modeled using a [logistic function](https://en.wikipedia.org/wiki/Logistic_function).





**This Model gives us 87% Accuracy and 93% F1 Score**

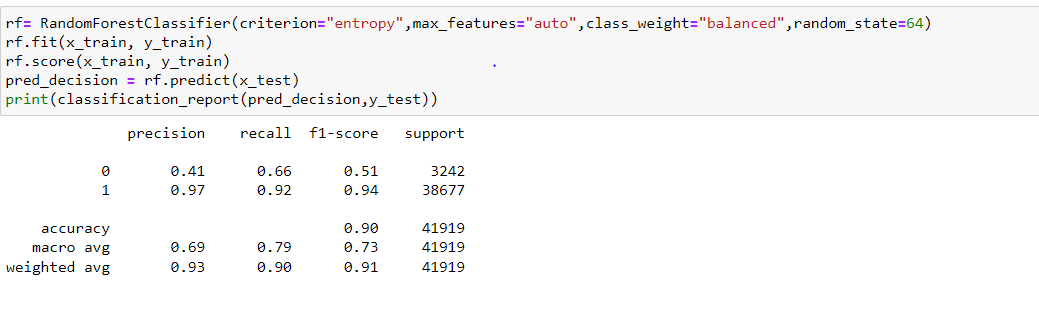
**2.Random Forest Classifier:**

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is controlled with the max\_samples parameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree.

**Hyper Parameter Tuning:**



**The result:**



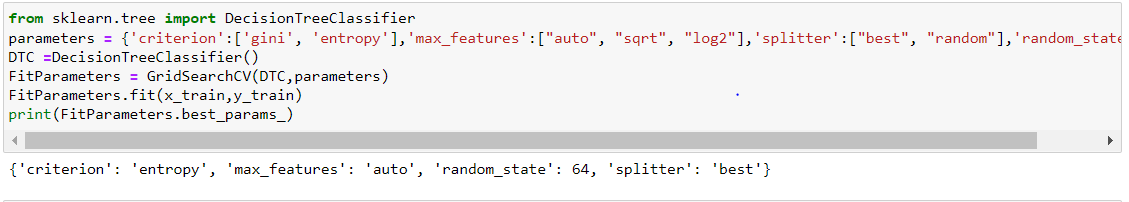
**We could see that using RFC we get 90% accuracy and 94% F1 Score**

1. **Decision Tree**

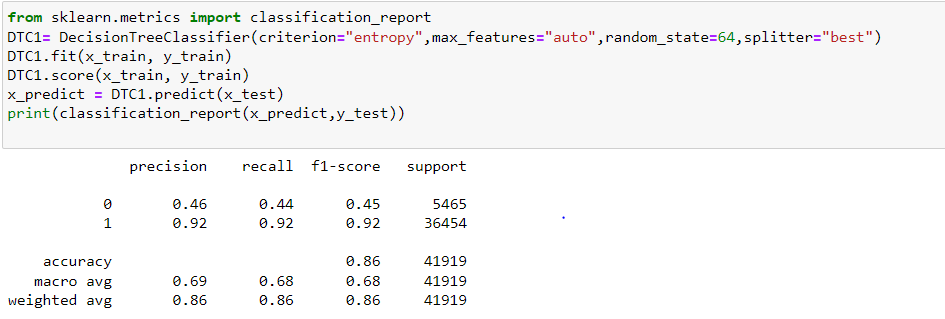
[**DecisionTreeClassifier**](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html#sklearn.tree.DecisionTreeClassifier) is a class capable of performing multi-class classification on a dataset.

As with other classifiers, **[DecisionTreeClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html" \l "sklearn.tree.DecisionTreeClassifier" \o "sklearn.tree.DecisionTreeClassifier)** takes as input two arrays: an array X, sparse or dense, of shape (n\_samples, n\_features) holding the training samples, and an array Y of integer values, shape (n\_samples,), holding the class labels for the training samples:

**Hyper Parameter Tuning:**



**Result**:



**This model gives us 86% Accuracy and 92% F1 Score.**

* Key Metrics for success in solving problem under consideration

1. **F1 Score:**

F-measure (also F-score) is a measure of a test’s accuracy that considers both the precision and the recall of the test to compute the score. Precision is the number of correct positive results divided by the total predicted positive observations. Recall, on the other hand, is the number of correct positive results divided by the number of all relevant samples (total actual positives).

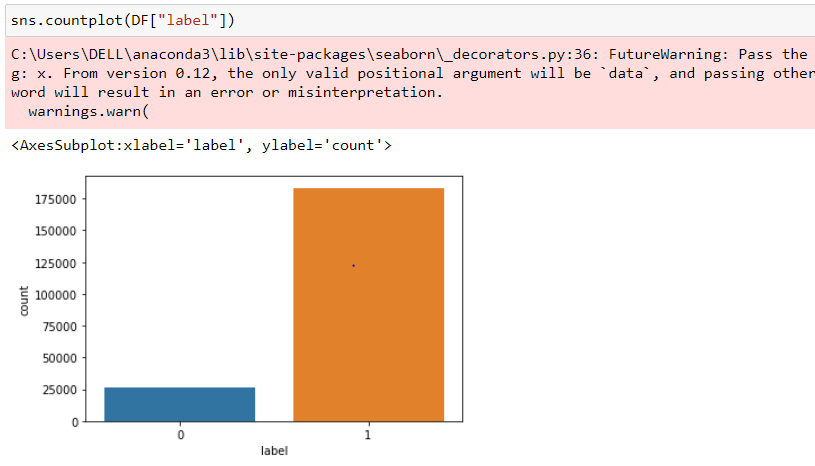
1. **Classification report has been generated to find the Accuracy score**

Accuracy is a common evaluation metric for classification problems. It’s the number of correct predictions made as a ratio of all predictions made. We use sklearn module to compute the accuracy of a classification task.

1. **ROC-AUC curve**

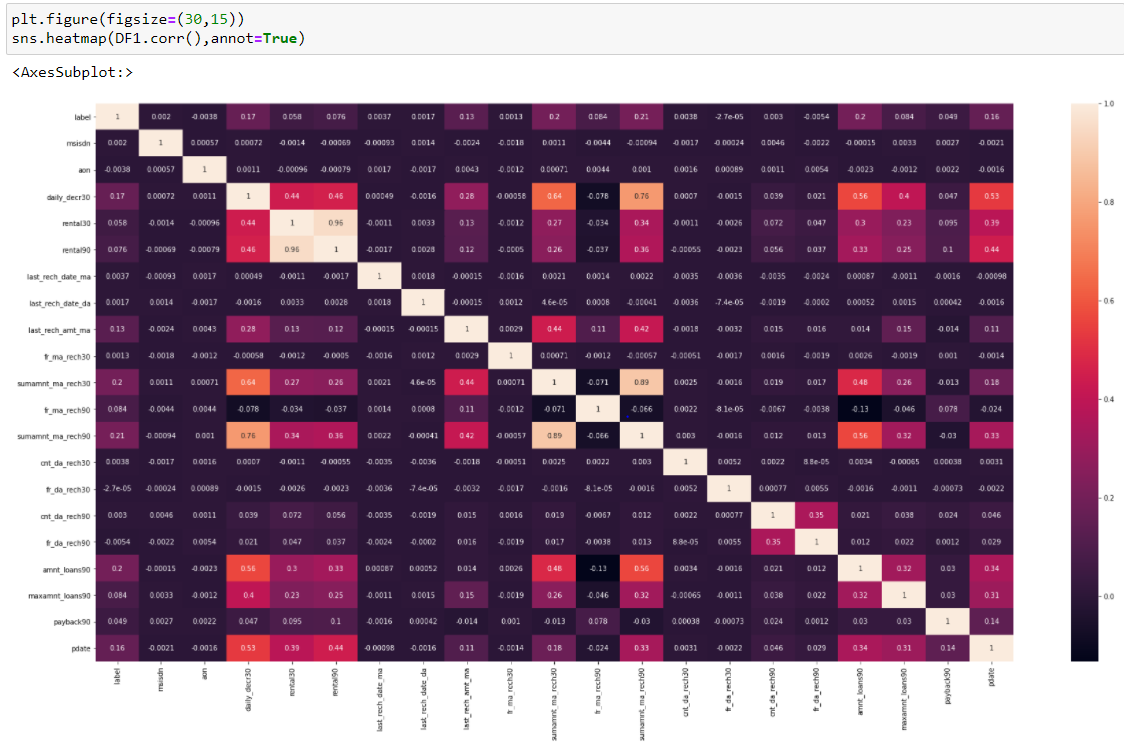
Area under ROC Curve is a performance metric for measuring the ability of a binary classifier to discriminate between positive and negative classes.

1. **Visualizations**
2. **Count Plot to understand if class imbalance exists.**



1. **Distplot to understand the skewness:**

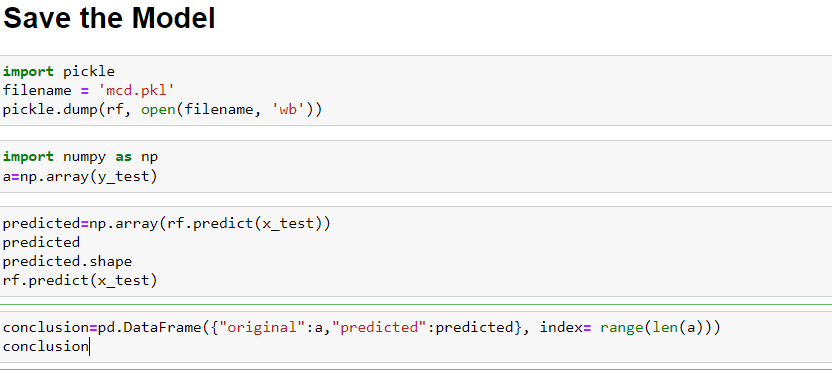
**3.Heatmap to understand Correlation**



**4.Box plot to check for Outliers**

1. **Interpretation of the Results**

From the above results we come to know that Random Forest which gives us 90% accuracy with F1 score as 94%. We have then saved the model using pickle and validated the result



**CONCLUSION**

1. **Key Findings and Conclusions of the Study**

As requested a ML model has been built for the MFUI industry to predict the defaulters and to reduce the risk of credit amount not Paid back on time

1. **Learning Outcomes of the Study in respect of Data Science**

The challenge faced is on the inter correlation as many fields were strongly inter-correlated to each other and the correlation with Target variable was less. I have removed the fields that had negative or very less correlation with Target but were highly inter correlated with independent variables.

Outliers existing in almost all the fields but while treating the same we had to encounter around 15% data loss. So outlier treatment has been skipped.

1. **Limitations of this work and Scope for Future Work**

The limitation of this work is that we may not get accurate result using this model if the outliers increase in the dataset. As outliers will have negative impact on the Model output.