

**MICRO CREDIT PROJECT**

Submitted by:

SAHIL SHAH

**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.

**INTRODUCTION**

* Problem Statement

Build a model which can be used to predict in terms of a probability for each loan transaction, whether the customer will be paying back the loaned amount within 5 days of insurance of loan.

* Business Problem Framing

In this problem we have to predict whether the customer will be paying back the loaned amount within 5 days of insurance of loan or not. This classic Business problem helps Micro Financing Institutions and other Lending companies to reduce Credit risks by recognizing potential Defaulters.

* Conceptual Background of the Domain

Earlier it was very difficult to predict the defaulters, because many times a perfect candidate would display unpredictable financial and repayment behaviour after being approved for loan. For the same purpose candidates’ income, past debt and repayment behaviour are important parameters.

* Review of Literature

Studied various terms and concepts on various sites related to microfinance institutions (MFI) providing micro credits to customers.

* Motivation for the Problem

Before advancement of Data Science, loan lending companies used to risk a high rate of defaulting. By applying this Machine Learning model, it will be easier for microfinance institutions (MFI) to predict the loan defaulters, and might not face losses regarding the issue.

**METHODOLOGY**

* Data Exploration and Cleaning

On data exploration, I found that the dataset was imbalanced for the target feature (87.5% for Non-defaulters and 12.5% for Defaulters). Also, I found that the data had some very unrealistic values such as 999860 days which is not possible. Also, there were negative values for variables which must not have one (example: frequency, amount of recharge etc). All these unrealistic values were dropped which caused a data loss of 8% only.

* Feature Selection

Since there were 36 features, many of which I suspected were redundant because of the data duplication. It was imperative to select only most significant of them to make ML models more efficient and cost effective. The method used was 'Univariate Selection' using chi-square test. I selected top 20 features which were highly significant.

* Data Visualization

On visualizing data, there were two important insights I gathered.

a. Imbalance of data

b. Distribution was not normal

* Data Normalization

Since the data was not normal, I normalized all the features except the target variable which was dichotomous (Values '1' and '0').

* Oversampling of Minority class

Since the data was expensive, I did not want to lose out on data by undersampling the majority class. Instead, I decided to oversample the minority class using SMOTE. After that I have built 5 different models.

**BUILDING MACHINE LEARNING MODELS**

* Logistic Regression

Logistic Regression is a Machine Learning algorithm which is used for the classification problems, it is a predictive analysis algorithm and based on the concept of probability. We can call a Logistic Regression a Linear Regression model, but the Logistic Regression uses a more complex cost function, this cost function can be defined as the ‘Sigmoid function’ or also known as the ‘logistic function’ instead of a linear function. The hypothesis of logistic regression tends it to limit the cost function between 0 and 1. Therefore linear functions fail to represent it as it can have a value greater than 1 or less than 0 which is not possible as per the hypothesis of logistic regression.

After fitting the model, the AUC was 0.83.

* Linear SVM

Support Vector Machine (SVM) is a supervised machine learning algorithm capable of performing classification, regression and even outlier detection. The linear SVM classifier works by drawing a straight line between two classes. All the data points that fall on one side of the line will be labeled as one class and all the points that fall on the other side will be labeled as the second. Sounds simple enough, but there’s an infinite number of lines to choose from. How do we know which line will do the best job of classifying the data? This is where the LSVM algorithm comes in to play. The LSVM algorithm will select a line that not only separates the two classes but stays as far away from the closest samples as possible. In fact, the “support vector” in “support vector machine” refers to two position vectors drawn from the origin to the points which dictate the decision boundary.

After fitting the model, the AUC was 0.82.

* Decision Tree

Decision tree is one of the predictive modelling approaches used in statistics, data mining and machine learning. A decision tree is a branched flowchart showing multiple pathways for potential decisions and outcomes. The tree starts with what is called a decision node, which signifies that a decision must be made. From the decision node, a branch is created for each of the alternative choices under consideration.

Decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions. It is one of the most widely used and practical methods for supervised learning. Decision Trees are a non-parametric supervised learning method used for both classification and regression tasks.

After fitting the model, the AUC was 0.71.

* Random forest

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

After fitting the model, the AUC was 0.88.

* Gradient Boost Classifier

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. When a decision tree is the weak learner, the resulting algorithm is called gradient boosted trees, which usually outperforms random forest. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

After fitting the model, the AUC was 0.86.

**CONCLUSION**

Random Forest performs the best with accuracy of 0.88 as compared to all other models. The curve for the same is tending towards the ideal shape. Hence Random Forest is the best fit for the problem.