**Credit Card Default Prediction**

**Akash Salmuthe**

**Data Science Enthusiast**

**AlmaBetter, Bangalore**

**Abstract:**

In today’s world credit cards have become a lifeline to a lot of people so banks provide us with credit cards.

Financial threats are displaying a trend about the credit risk of commercial banks as the incredible improvement in the financial industry has arisen. In this way, one of the biggest threats faces by commercial banks is the risk prediction of credit clients. Recent studies mostly focus on enhancing the classifier performance for credit card default prediction rather than an interpretable model. In classification problems, an imbalanced dataset is also crucial to improve the performance of the model because most of the cases lied in one class, and only a few examples are in other categories. Traditional statistical approaches are not suitable to deal with imbalanced data. In this study, a model is developed for credit default prediction.

Our experiment can help understand what could be the reason for the classification of such labels by feature selection, data analysis and prediction with machine learning algorithms taking into account previous trends to determine the correct classification or finding defaulters.

***Keywords: machine learning, Credit card default prediction, classified labels, imbalanced dataset***

**1. Introduction**

As an unsecured credit facility, credit cards have huge risks behind the high re-turns of banks. The ever-increasing number of credit card circulation cards has brought about an increase in the amount of credit card defaults, and the resulting large amount of bills and repayment information data have also brought certain difficulties to the risk controllers. Therefore, how to use the data generated by users, and extract useful information to control risks, reduce default rate, and control the growth of non-performing rate has become one of the key concerns of banks. Credit card default prediction is based on the historical data of credit card customers. The use of corresponding methods to predict and analyze credit card customer default behavior is a typical classification problem.

**2. Problem Statement**

This project is aimed at predicting the case of customers' default payments in Taiwan. From the perspective of risk management, the result of predictive accuracy of the estimated probability of default will be more valuable than the binary result of classification - credible or not credible clients. We can use the [K-S chart](https://www.listendata.com/2019/07/KS-Statistics-Python.html) to evaluate which customers will default on their credit card payments

A Credit Card is a type of payment card in which charges are made against a line of credit instead of the account holder’s cash deposit. When someone uses a credit card to make purchase, that person’s account accrues a balance that must be paid off each month

The main objective is to build a predictive model, which could help them in predicting the surge pricing type proactively. This would in turn help them in matching the right cabs with the right customers quickly and efficiently.

**3. Data Description**

This dataset contains information on default payments, demographic factors, credit data, history of payment, and bill statements of credit card clients in Taiwan.

* Data set Characteristics =Multivariate
* Attribute Characteristics = Integer, Real
* Associated Task = Classification
* Number of Instances = 30,000
* Number of Attributes = 24 Number of Variables = 25

**Attribute Information:**

* **X1:** Amount of the given credit
* **X2:** Gender (1 = male; 2 = female).
* **X3:** Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
* **X4:** Marital status (1 = married; 2 = single; 3 = others).
* **X5:** Age (year).
* **X6 - X11:** History of past payment. (-1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.)
* **X12-X17:** Amount of bill statement
* **X18-X23:** Amount of previous payment
* **default.payment.next.month:** default payment (1=yes,0=no)

**4. Reasons for defaults**

**There are mainly two reasons:**

1. Intentionally they don’t want to pay
2. They don’t have enough money to pay

**5. Steps Involved:**

* **Exploratory Data Analysis**

After loading the dataset we performed this method by comparing our target variable that is Defaulters with Other independent variables. This process helped us figuring out various aspects and relationships among the target and the independent variables. It gave us a better idea of which feature behaves in which manner compared to the target variable.

* **Null values Treatment**

Our dataset contains NO null values which may tend to our accuracy at the beginning of our project in order to get a better result.

* **Encoding of categorical columns**

We used One Hot Encoding to produce binary integers of 0 and 1 to encode our categorical features are SEX, MARRIAGE, EDUCATION, because categorical features that are in string format cannot be understood by the machine and needs to be converted to numerical format.

* **Feature Selection**

In these steps we using information gaining method finding correlation coefficients between variables mostly effects that are removed from our data set i.e. "PAY\_SEPT","BILL\_AMT\_SEPT","PAY\_AMT\_SEPT" removed from the dataset.

* **SMOTE**

The given data set unbalanced data set for this we used SMOTE i.e. Synthetic Minority Oversampling Technique resampling of performing best results for randomly before after applying SMOTE and fitting and applying different algorithms to it

The basic goal was to enforce a level of consistency or uniformity to certain practices or operations within the selected environment.

* **Fitting different models**

For modelling we tried various classification algorithms like:

1. Logistic Regression
2. Random Forest Classifier
3. XGBoost Classifier

**6.1. Algorithm**

**i) Logistic Regression:**

Logistic Regression is actually a classification algorithm that was given the name regression due to the fact that the mathematical formulation is very similar to linear regression.

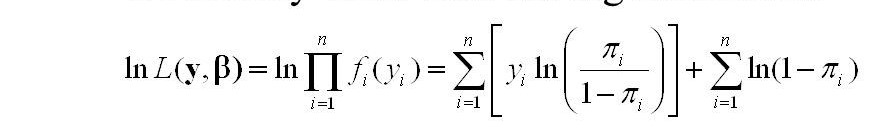
The function used in Logistic Regression is sigmoid function or the logistic function given by:

*f(x)= 1/1+e ^(-x)*



Fig 1: Logistic Regression

The optimization algorithm used is: Maximum Log Likelihood. We mostly take log likelihood in Logistic:



We have implemented logistic regression with Grid search CV. We get an accuracy score of approximately 75% and precision score approximately is 69% and f1\_score is 73%and roc auc approximately is 73%. As we have an imbalanced dataset, recall score is approximately 78% better parameter.

**ii) Random Forest Classifier:**

Random forest classifiers fall under the broad umbrella of ensemble-based learning methods. They are simple to implement, fast in operation, and have proven to be extremely successful in a variety of domains. The key principle underlying the random forest approach comprises the construction of many “simple” decision trees in the training stage and the majority vote (mode) across them in the classification stage. Among other benefits, this voting strategy has the effect of correcting for the undesirable property of decision trees to overfit training data.

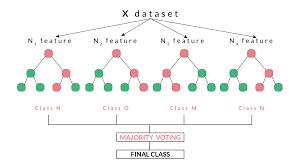


Fig 2: Random forest Classifier

Random Forest and our accuracy score is approximately 83% and recall score is approximately 85% and f1\_score is 83% and ROC\_AUC score is 91%, precision score is approximately 80% better parameter. Let's go ahead with other models and see if they can give better result.

**iii) XGBoost Classifier**

To understand XGBoost we have to know gradient boosting beforehand.

* **Gradient Boosting-**

Gradient boosted trees consider the special case where the simple model is a decision tree



Fig 3: XGBoost Classifier

In this case, there are going to be 2 kinds of parameters P: the weights at each leaf, w, and the number of leaves T in each tree (so that in the above example, T=3 and w = ([2, 0.1,-1]).

When building a decision tree, a challenge is to decide how to split a current leaf. For instance, in the above image, how could I add another layer to the (age > 15) leaf? A ‘greedy’ way to do this is to consider every possible split on the remaining features (so, gender and occupation), and calculate the new loss for each split; you could then pick the tree which most reduces your loss.

* **XGBoost**

XGBoost is an ensemble learning method. Sometimes, it may not be sufficient to rely upon the results of just one machine learning model. Ensemble learning offers a systematic solution to combine the predictive power of multiple learners. The resultant is a single model which gives the aggregated output from several models.

The models that form the ensemble, also known as base learners, could be either from the same learning algorithm or different learning algorithms. Bagging and boosting are two widely used ensemble learners. Though these two techniques can be used with several statistical models, the most predominant usage has been with decision trees.

**6.2. Model performance:**

Model can be evaluated by various metrics such as:

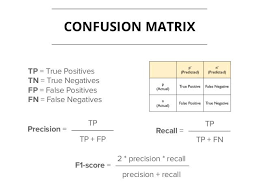


Fig 4: Confusion Matrix

* **Confusion Matrix**:

The confusion matrix is a table that summarizes how successful the classification modelis at predicting examples belonging to various classes. One axis of the confusion matrix is the label that the model predicted, and the other axis is the actual label.

* **Precision/Recall**:

Precision is the ratio of correct positive predictions to the overall number of positive predictions: TP/TP+FP

**Recall** is the ratio of correct positive predictions to the overall number of positive examples in the set: TP/FN+TP

* **Accuracy**

It is given by the number of correctly classified examples divided by the total numberof classified examples. In terms of the confusion matrix, it is given by: TP+TN/TP+TN+FP+FN

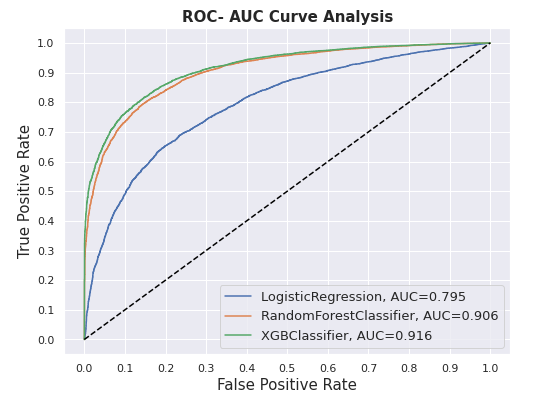
****

Fig 5: ROC-AUC Curve

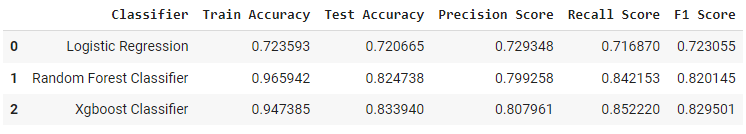
* **Area under ROC Curve (AUC)**

ROC curves use a combination of the true positive rate (the proportion of positive examples predicted correctly, defined exactly as recall) and false positive rate (the proportion of negative examples predicted incorrectly) to build up a summary picture of the classification performance.

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects performance, stability and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

* **Grid Search CV:** Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model’s performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

**7. Result:**

****

**8. Conclusion**

That's it! We reached the end of our exercise.

Starting with loading the data so far we have done EDA, null values treatment, encoding of categorical columns, feature selection and then model building.

In all of these models our accuracy revolves in the range of 73 to 91%.

And there is no such improvement in accuracy score even after hyperparameter tuning.

We used different type of Classification algorithms to train our model like, Logistic Regression, Random Forest Classifier, XGboost Classifier and also we tuned the parameters of Random forest classifier and XG boost classifier, out of them Random forest classifier with Grid search CV ( tuned hyperparameters gave) the best result.

Precision score is approximately 80%,

Recall score is approximately85%

ROC\_AUC score is approximately 91%,

Accuracy Score is approximately 83% and it’s F1\_score approximately 83%

**References:**

1. [*https://www.researchgate.net/publication/327864565\_Comparison\_of\_Several\_Data\_Mining\_Methods\_in\_Credit\_Card\_Default\_Prediction*](https://www.researchgate.net/publication/327864565_Comparison_of_Several_Data_Mining_Methods_in_Credit_Card_Default_Prediction)
2. [*https://ieeexplore.ieee.org/document/9239944*](https://ieeexplore.ieee.org/document/9239944)
3. [Simeon Kostadinov](https://towardsdatascience.com/%40simonnoff?source=post_page-----b9b810a3f38----------------------): "My Analysis from 50+ papers on the Application of ML in Credit Lending”: *[https://towardsdatascience.com/my-analysis-from-50-papers-on-the-application-of-ml-in-](https://towardsdatascience.com/my-analysis-from-50-papers-on-the-application-of-ml-in-credit-lending-b9b810a3f38)* [*credit-lending-b9b810a3f38*](https://towardsdatascience.com/my-analysis-from-50-papers-on-the-application-of-ml-in-credit-lending-b9b810a3f38)
4. [Khyati Mahendru](https://medium.com/%40khyati.ma25?source=post_page-----98e4d79fcddb----------------------): "How to Deal with Imbalanced Data using SMOTE ": [*https://medium.com/analytics-vidhya/balance-your-data-using-smote-98e4d79fcddb*](https://medium.com/analytics-vidhya/balance-your-data-using-smote-98e4d79fcddb)