CREDIT CARD DEFAULT PREDICTION

Data science trainee,
AlmaBetter, Bangalore

Abstract:

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.

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 to evaluate which customers will default on their credit card payments.

Variables

Attribute Information:

This research employed a binary variable, default payment (Yes = 1, No = 0), as the response variable. This study reviewed the literature and used the following 23 variables as explanatory variables:

- ID: ID of each customer
- LIMIT_BAL: Amount of the given credit (NT dollar)
- SEX: Gender(Male = 1, Female: 2)
- EDUCATION: (1= graduate school, 2=University, 3=High School, 0,4,5,6 = Others)
- MARRIAGE: Marital status (0 = others, 1 = married, 2 = single, 3 = others)
- AGE: Age in years

Scale for PAY_0 to PAY_6 : (-2 = No consumption, -1 = paid in full, 0 = use of revolving credit (paid minimum only), 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)

- PAY_0: Repayment status in September, 2005 (same scale as given)
- PAY_2: Repayment status in August, 2005 (same scale as given)
- PAY_3: Repayment status in July, 2005 (same scale as given)
- PAY_4: Repayment status in June, 2005 (same scale as given)
- PAY_5: Repayment status in May, 2005 (same scale as given)
- PAY_6: Repayment status in April, 2005 (same scale as given)
- BILL_AMT1: Amount of bill statement in September, 2005 (NT dollar)
- BILL_AMT2: Amount of bill statement in August, 2005 (NT dollar)
- BILL_AMT3: Amount of bill statement in July, 2005 (NT dollar)
- BILL_AMT4: Amount of bill statement in June, 2005 (NT dollar)
- BILL_AMT5: Amount of bill statement in May, 2005 (NT dollar)
- BILL_AMT6: Amount of bill statement in April, 2005 (NT dollar)

- PAY_AMT1: Amount of previous payment in September, 2005 (NT dollar)
- PAY_AMT2: Amount of previous payment in August, 2005 (NT dollar)
- PAY_AMT3: Amount of previous payment in July, 2005 (NT dollar)
- PAY_AMT4: Amount of previous payment in June, 2005 (NT dollar)
- PAY_AMT5: Amount of previous payment in May, 2005 (NT dollar)
- PAY_AMT6: Amount of previous payment in April, 2005 (NT dollar)
- default payment next month: Default Payments(1 = Yes, 0 = No)

We have been provided with 6 months credit card transaction history along with the informative details of customer and their current status, on the basis of this dataset we have to predict either he/she is a defaulter or not if he/she possess with these qualities.

Introduction

This dataset contains information on default payments, demographic factors, credit data, history of payment, and bill statements of credit card clients in Taiwan from April 2005 to September 2005.

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.

Overview

In today's world credit cards have become a lifeline to a lot of people so banks provide us with credit cards. Now we know the most common issue there is in providing these kind of deals are people not being able to pay the bills. These people are what we call

"Defaulters".

Objective

The main objective of 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.

Steps involved:

- **Exploratory Data Analysis**:
  The first step of our project is performing the EDA process on the dataset so that we can get the ideaabout the dataset i.e. the number of variables, the data type of the variables , visualize the dataset for Better understanding and decide the suitable methods and algorithms that might produce desiredoutcomes

- **Data Preprocessing:**
  In EDA process we find the type of dataset and decide the approach, in this project the preprocessingsteps would removing the punctuations, stop words , generate count vectorizer and document term matrix which would help in building up the model.

- **Building Machine Learning Model**:
  After the data preprocessing is done then the data will be ready to be fit into machine learning models .For current problem statement topic modeling approach would be suitable . In topic

modeling, a topic is defined by a cluster of words with each word in the cluster having a probabilityof occurrence for the given topic, and different topics have their respective clusters of words along with corresponding probabilities. Then we use different types of evaluation metrics for classification. On the basis of score we make confusion matrix , Roc-Auc curve.

Summary:

At last the summary of the project is described to have brief look over the project.

Algorithms:

Logistic Regression Implementation

Logistic Regression is one of the simplest algorithms which estimates the relationship between one dependent binary variable and independent variables, computing the probability of occurrence of an event. The regulation parameter C controls the trade-off between increasing complexity (overfitting) and keeping the model simple(underfitting), This parameter signifies strength of the regularization and takes a positive float value. C and regularization strength are negatively correlated (smaller the C is stronger the regularization will be).

Decision Tree

- Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems. It is a tree-structured classifier, where internal nodes represent the features of adataset, branches represent the decision rules and each leaf node represents the outcome.

- In a Decision tree, there are two nodes, which are the Decision node and Leaf node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.
- The decisions or the test are performed on the basis of features of the given dataset.
- It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

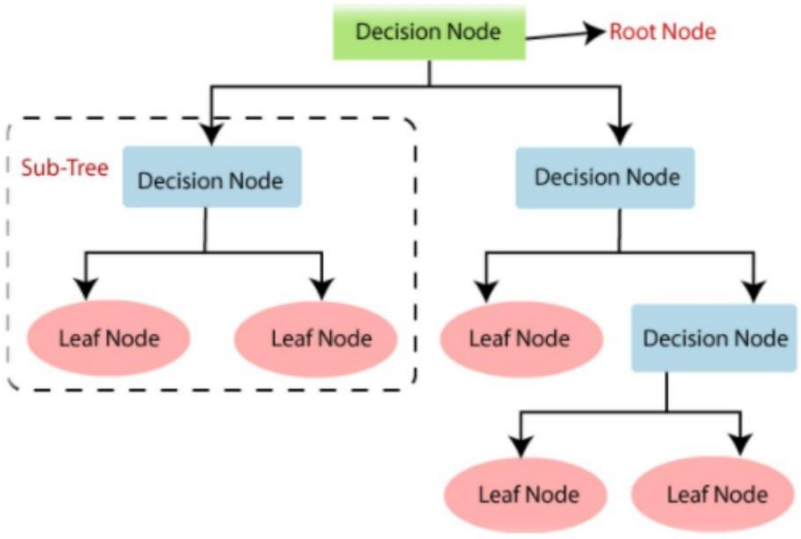


Fig 1.General structure of decision tree

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.
- 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 random forest leads to higher accuracy and prevents the problem of overfitting.

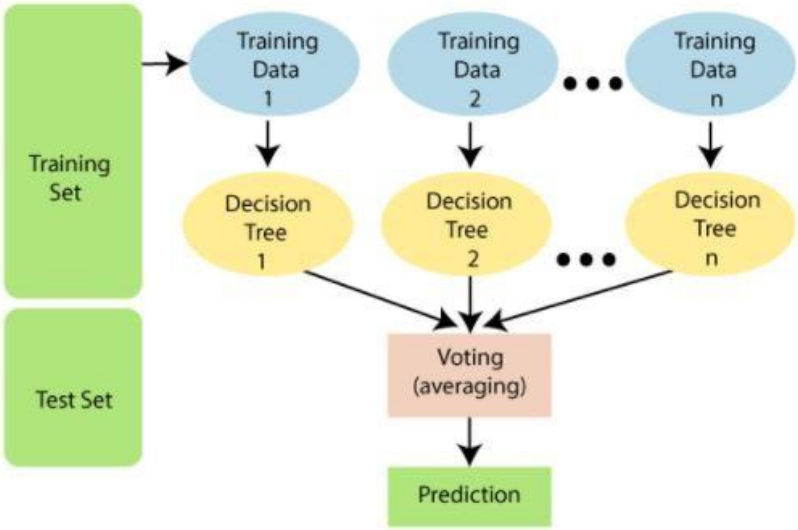


Fig 2. Working of Random Forest

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.
- XGBoost is one of the fastest implementations of gradient boosting trees. It does this by tackling one of the major inefficiencies of gradient boosted trees: considering the potential loss for all possible splits to create a new branch (especially if you consider the case where there are thousands of features, and therefore thousands of possible splits).

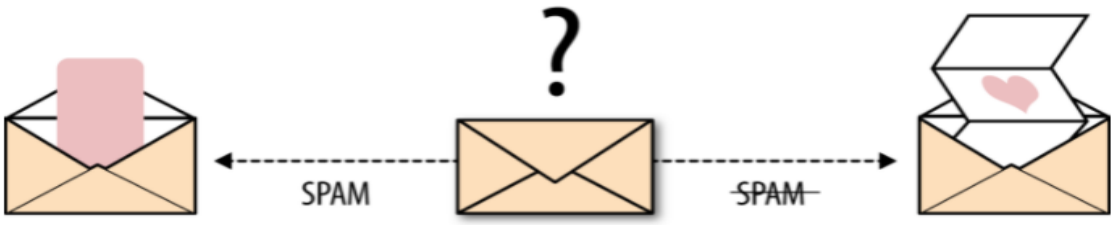
- XGBoost tackles this inefficiency by looking at the distribution of features across all data points in a leaf and using this information to reduce the search space of possible feature split.

Model performance:

Classification Metrics

Classification is about predicting the class labels given input data. In binary classification, there are only two possible output classes(i.e., Dichotomy). In multiclass classification, more than two possible classes can be present. I'll focus only on binary classification.

A very common example of binary classification is spam detection, where the input data could include the email text and metadata (sender, sending time), and the output label is either *"spam" or "not spam."* (*See Figure*) Sometimes, people use some other names also for the two classes: "positive" and "negative," or "class 1" and "class 0."



There are many ways for measuring classification performance. Accuracy, confusion matrix, log-loss, and AUC-ROC are some of the most popular metrics. Precision-recall is a widely used metrics for classification problems.

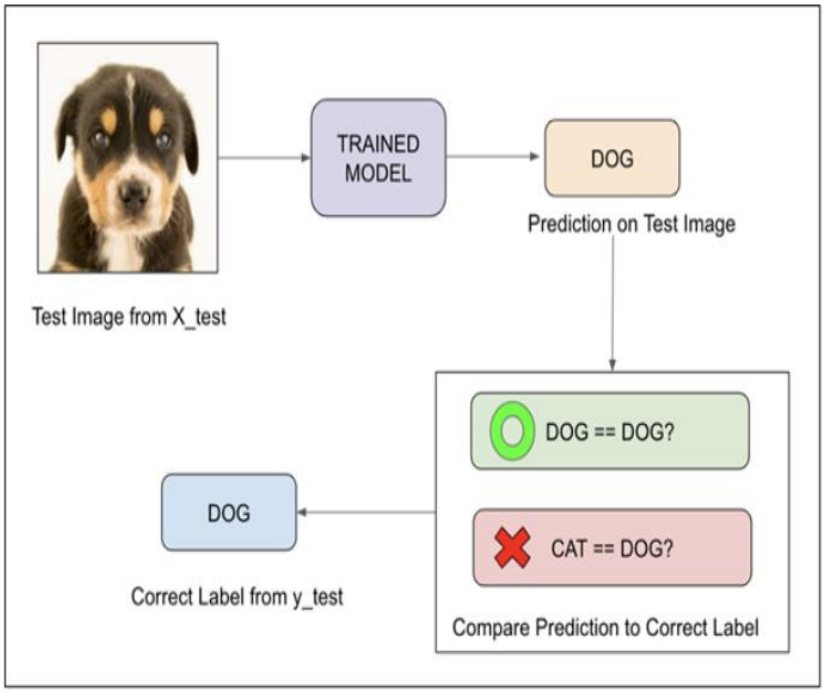
Accuracy

Accuracy simply measures how often the classifier correctly predicts. We can define accuracy as the ratio of the number of correct predictions and the total number of predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

When any model gives an accuracy rate of 99%, you might think that model is performing very good but this is not always true and can be misleading in some situations. I am going to explain this with the help of an example.

Consider a binary classification problem, where a model can achieve only two results, either model gives a **correct** or **incorrect** prediction. Now imagine we have a classification task to predict if an image is a dog or cat as shown in the image. In a supervised learning algorithm, we first **fit/train** a model on training data, then **test** the model on **testing data**. Once we have the model's predictions from the **X_test** data, we compare them to the **true y_values** (the correct labels).



We feed the image of the dog into the training model. Suppose the model predicts that this is a dog, and then we compare the prediction to the correct label. If the model predicts that this image is a cat and then we again compare it to the correct label and it would be incorrect.

We repeat this process for all images in X_test data. Eventually, we'll have a count of correct and incorrect matches. But in reality, it is very rare that all incorrect or correct matches hold **equal value**. Therefore one metric won't tell the entire story.

Accuracy is useful when the target class is ***well balanced*** but is not a good choice for the unbalanced classes. Imagine the scenario where we had 99 images of the dog and only 1 image of a cat present in our training data. Then our model would always predict the dog, and therefore we got 99% accuracy. In reality, Data is always imbalanced for example Spam email, credit card fraud, and medical diagnosis. Hence, if we want to do a better model evaluation and have a full picture of the model evaluation, other metrics such as recall and precision should also be considered.

Confusion Matrix

Confusion Matrix is a performance measurement for the machine learning classification problems where the output can be two or more classes. It is a table with combinations of predicted and actual values.

A confusion matrix is defined as thetable that is often used to describe the performance of a classification model on a set of the test data for which the true values are known.

Predicted Values \ Actual Values	Positive (1)	Negative (0)
Positive (1)	TP	FP
Negative (0)	FN	TN

It is extremely useful for measuring the Recall, Precision, Accuracy, and AUC-ROC curves.

Let's try to understand TP, FP, FN, TN with an example of pregnancy analogy.

True Positive: We predicted positive and it's true. In the image, we predicted that a woman is pregnant and she actually is.

True Negative: We predicted negative and it's true. In the image, we predicted that a man is not pregnant and he actually is not.

False Positive (Type 1 Error)- We predicted positive and it's false. In the image, we predicted that a man is pregnant but he actually is not.

False Negative (Type 2 Error)- We predicted negative and it's false. In the image, we predicted that a woman is not pregnant but she actually is.

We discussed Accuracy, now let's discuss some other metrics of the confusion matrix

1. Precision —Precision explains how many of the correctly predicted cases actually turned out to be positive. Precision is useful in the cases where False Positive is a higher concern than False Negatives. The importance of *Precision is in music or video recommendation systems, e-commerce websites, etc. where wrong results could lead to customer churn and this could be harmful to the business.*

Precision for a label is defined as the number of true positives divided by the number of predicted positives.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

2. Recall (Sensitivity)-- Recall explains how many of the actual positive cases we were able to predict correctly with our model. It is a useful metric in cases where False Negative is of higher concern than False Positive. It *is important in medical cases where it doesn't matter whether we raise a false alarm but the actual positive cases should not go undetected!*

Recall for a label is defined as the number of true positives divided by the total number of actual positives.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

3. F1 Score — It gives a combined idea about Precision and Recall metrics. It is maximum when Precision is equal to Recall.

F1 Score is the harmonic mean of precision and recall.

$$F1 = 2.\frac{Precision \times Recall}{Precision + Recall}$$

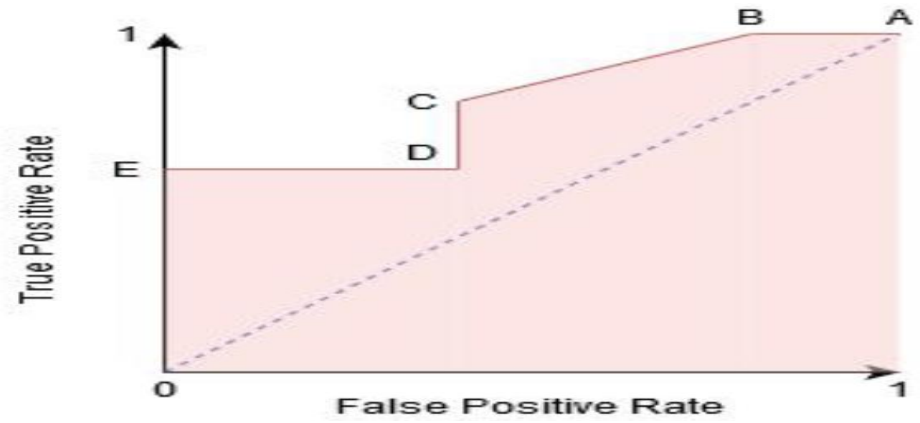
The F1 score punishes extreme values more. F1 Score could be an effective evaluation metric in the following cases:

- When FP and FN are equally costly.
- Adding more data doesn't effectively change the outcome
- True Negative is high

4. AUC-ROC — The Receiver Operator Characteristic (ROC) is a probability curve that plots the TPR(True Positive Rate) against the FPR(False Positive Rate) at various threshold values and separates the 'signal' from the 'noise'.

The **Area Under the Curve (AUC)** is the measure of the ability of a classifier to distinguish between classes. From the graph, we simply say the area of the curve ABDE and the X and Y-axis.

From the graph shown below, the greater the AUC, the better is the performance of the model at different threshold points between positive and negative classes. This simply means that When AUC is equal to 1, the classifier is able to perfectly distinguish between all Positive and Negative class points. When AUC is equal to 0, the classifier would be predicting all Negatives as Positives and vice versa. When AUC is 0.5, the classifier is not able to distinguish between the Positive and Negative classes.



Working of AUC—In a ROC curve, the X-axis value shows False Positive Rate (FPR), and Y-axis shows True Positive Rate (TPR). Higher the value of X means higher the number of False Positives(FP) than True Negatives(TN), while a higher Y-axis value indicates a higher number of TP than FN. So, the choice of the threshold depends on the ability to balance between FP and FN.

5. Log Loss — Log loss (Logistic loss) or Cross-Entropy Loss is one of the major metrics to assess the performance of a classification problem.

For a single sample with true label y∈{0,1} and a probability estimate p=Pr(y=1), the log loss is:

$$logloss_{(\mathrm{N}=1)} = y\log(p) + (1-y)\log(1-p)$$

Where *y_i* is the i'th expected value in the dataset and *yhat_i* is the i'th predicted value. The difference between these two values is squared, which has the effect of removing the sign, resulting in a positive errorvalue.

Hyper parameter tuning:

Grid Search CV

Grid Search combines a selection of hyper parameters established by the scientist and runs throughall 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.

Conclusion:

1. Distribution of defaulter vs. non defaulter - around 78% are non defaulter and 22% are defaulter. Also we check for Marriage, Education, Sex with respect to defaulter and we found in marriage more number of defaulter are Male, in Education more no. of defaulter are University Students & in Marriage more no. of defaulter are Married.
2. After that we build the Four models Logistic Regression, Decision Tree, Default XGBoost Classifier & Random Forest . The best accuracy is obtained from the Default XGBoost Classifier
3. Using a Logistic Regression classifier, we can predict with 65.97% accuracy, whether a customer is likely to default next month.
5. With Decision Tree classifier having precision 57.77%, we can predict with accuracy of 71.83%, whether customer is likely to default next month.
6. Using Random Forest, we can predict with accuracy of 81.13%, whether customer will be defaulter in next month.
7. XG Boost Classifier with recall 61.77%, accuracy of 81.73%, we can predict whether customer is likely to default next month.

From the models that are applied on the dataset, *XG Boost and Random Forest* are giving the best evaluation matrices (precision, F1-score and ROC-AUC score).

- On behalf of these matrices we can predict whether customers would be defaulter or not in the next month.
- From the **ROC-AUC** curve, Random Forest and XG Boost classifier are more able to distinguish between positive and negative class.