**Credit card default prediction**

Jayesh

**ABSTRACT**

Aiming at the problem that the credit card default data of a financial institution is unbalanced, which leads to unsatisfactory prediction results, this paper proposes a prediction model based on *k-*means SMOTE and BP neural network. In this model, *k-*means SMOTE algorithm is used to change the data distribution, and then the importance of data features is calculated by using random forest, and then it is substituted into the initial weights of BP neural network for prediction. The model effectively solves the problem of sample data imbalance. At the same time, this paper constructs four common machine learning models, KNN, logistics, random forest, and Decision tree, and compares the classification performance of these four prediction models..

**PROBLEM STATEMENT**

Recently, the state vigorously promotes the economic construction of large- and medium-sized cities, which not only improves people’s living standards but also changes people’s consumption concept and consumption mode. People are more and more inclined to spend ahead of time and mortgage their “credit” to the bank to enjoy certain things in advance. However, when consuming, people often lack rational thinking and overestimate their ability to repay loans to banks in time. On the one hand, it increases the loan risk of banks; on the other hand, it increases the credit crisis of consumers themselves. With a large number of banks selling credit cards, the phenomenon of credit card default emerges one after another. It is very important for banks to effectively identify high-risk credit card default users. Generally speaking, compared with the credit card customers who have not paid their loans overdue, there are fewer overdue repayments

**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:

### X1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) 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. We tracked the past monthly payment records (from April to September, 2005) as follows: X6 = the repayment status in September, 2005; X7 = the repayment status in August, 2005; . . .;X11 = the repayment status in April, 2005. The measurement scale for the repayment status is: -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 (NT dollar). X12 = amount of bill statement in September, 2005; X13 = amount of bill statement in August, 2005; . . .; X17 = amount of bill statement in April, 2005.

### X18-X23: Amount of previous payment (NT dollar). X18 = amount paid in September, 2005; X19 = amount paid in August, 2005; . . .;X23 = amount paid in April, 2005.

**INTRODUCTION**

As we are moving towards the digital world — cybersecurity is becoming a crucial part of our life. When we talk about security in digital life then the main challenge is to find the abnormal activity. When we make any transaction while purchasing any product online — a good amount of people prefer credit cards. The credit limit in credit cards sometimes helps us me making purchases even if we don’t have the amount at that time. but, on the other hand, these features are misused by cyber attackers. To tackle this problem we need a system that can abort the transaction if it finds fishy. Here, comes the need for a system that can track the pattern of all the transactions and if any pattern is abnormal then the transaction should be aborted. Today, we have many machine learning algorithms that can help us classify abnormal transactions. The only requirement is the past data and the suitable algorithm that can fit our data in a better form. In this article, I will help you in the complete end-to-end model training process — finally, you will get the best model that can classify the transaction into normal and abnormal types.

**STEPS INVOLVED**

In order to go ahead for data visualization upon key factors we need to go for certain extra steps before proceeding to the main segment. In this part we are going with the following steps:

1. Importing Analytical necessary library classes for future analysis.
2. Reading the csv data file from Google drive.
3. Setting figure size for future visualization.
4. Removing future warnings in seaborne plots.
5. Visualizing all the columns of the respective Data frame.
6. Viewing all data information
7. Checking the Unique values in the column ( if any)
8. Converting the data types to similar objects as the Analysis Demands.
9. Formatting the “size” column into a single column in the dataset.
10. Eradicating special characters from the dataset columns.

* **EXPLORATORY DATA ANALYSIS**

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypotheses and to check assumptions with the help of summary statistics and graphical representations. It gives us better idea of which feature behaves in which manner compared to target variable. After loading the dataset we performed this method by comparing our target variable that is Rented Bike count with other independent variables. This process helped us figuring out various aspects and relationships among the target and the independent variables.

* **EXAMINING** **NULL** **VALUES**

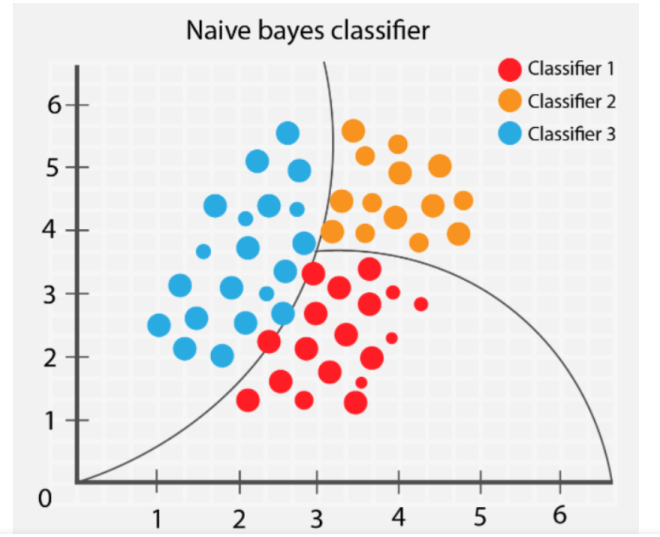
The most critical thing from which we can draw some observations is Dataset, however data comes with unexpected values too i.e. sometimes it may be Null or missing in other words the space might be blank. Thus, at the time of analysing the first thing which we will do is to examine the null or missing values on the Dataset. It is the first step that will make the results “more” accurate &should be handled before it affects the performance of the models that predict the outcome.

* Random Forest
* Decision Tree
* Logistic Regression
* KNN

**ALGORITHMS**

# I. Naive Bayes

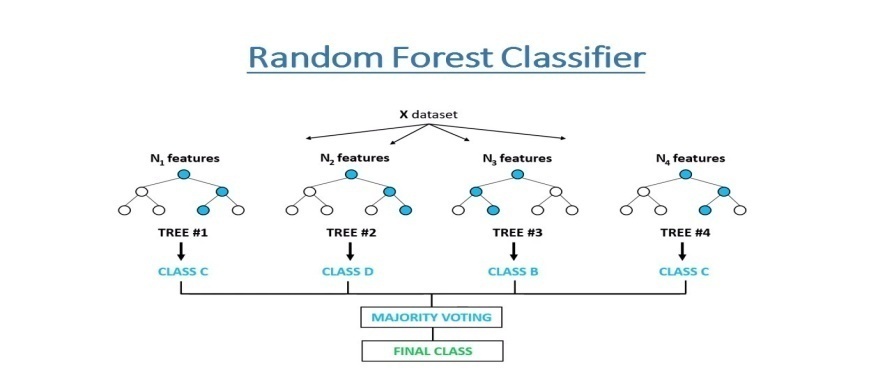
The Naive Bayes classification algorithm is a probabilistic classifier. It is based on probability models that incorporate strong independence assumptions. The independence assumptions often do not have an impact on reality. Therefore they are considered as naive. You can derive probability models by using Bayes' theorem (credited to Thomas Bayes). Depending on the nature of the probability model, you can train the Naive Bayes algorithm in a supervised learning setting.



Stochastic Gradient Descent (SGD) is a simple yet efficient optimization algorithm used to find the values of parameters/coefficients of functions that minimize a cost function. In other words, it is used for discriminative learning of linear classifiers under convex loss functions such as SVM and Logistic regression. It has been successfully applied to large-scale datasets because the update to the coefficients is performed for each training instance, rather than at the end of instances.

**III. Random Forest Classifier**

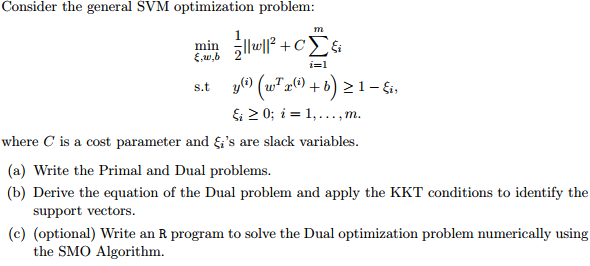
Random Forest is a bagging type of Decision Tree Algorithm that creates a number of decision trees from a randomly selected subset of the training set, collects the labels from these subsets and then averages the final prediction depending on the most number of times a label has been predicted out of all.

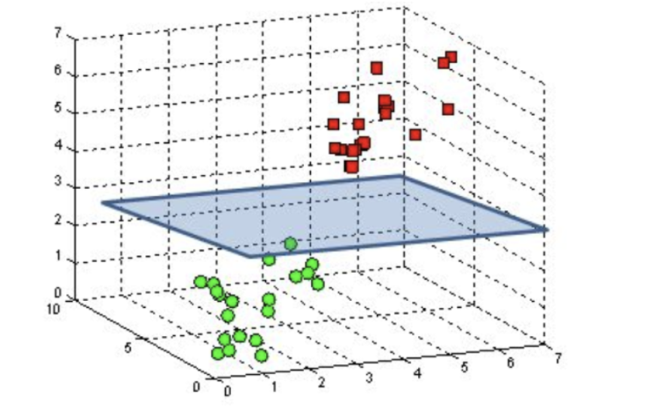


**IV. Support Vector Machine Classifier**

SVM is used mostly when the data cannot be linearly separated by logistic regression and the data has noise. This can be done by separating the data with a hyperplane at a higher order dimension.

In SVM we use the optimization algorithm as:





We use hinge loss to deal with the noise when the data isn’t linearly separable.

Kernel functions can be used to map data to higher dimensions when there is inherent non linearity.

**V.Logistic Regression**

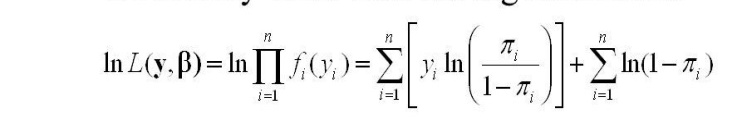
Logistic Regression is actually a classification algorithm that was given the name regression due to the fact that the mathematical formulation is very similar to linear regression.

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

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



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



**MODEL PERFORMANCE**

**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-**

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

**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.

**HYPER PARAMETER TUNING**

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.

We used Grid Search CV, Randomized Search CV and Bayesian Optimization for hyperparameter tuning. This also results in cross validation and in our case we divided the dataset into different folds. The best performance improvement among the three was by Bayesian Optimization.

**Grid Search CV**

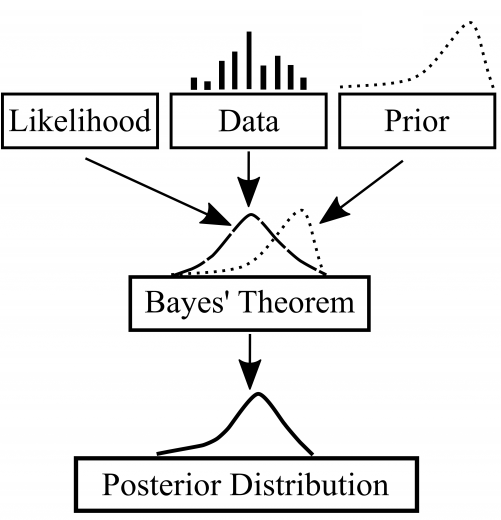
Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model’s performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific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

**Randomized Search CV**

In Random Search, the hyperparameters are chosen at random within a range of values that it can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters, since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist’s control

**Bayesian Optimization**

Bayesian hyper parameters optimization is a very efficient and interesting way to find good hyper parameters. In this approach, in naive interpretation way is to use a support model to find the best hyper parameters. A hyper parameter optimization process based on a probabilistic model, often Gaussian Process, will be used to find data from data observed in the later distribution of the performance of the given models or set of tested hyper parameters.



As it is a Bayesian process each iteration, the distribution of the model’s performance in relation to the hyperparameters used is evaluated and a new probability distribution is generated. With this distribution it is possible to make a more appropriate choice of the set of values that we will use so that our algorithm learns in the best possible way.

**CONCLUSION**

In this paper we developed a novel method for fraud detection, where customers are grouped based on their transactions and extract behavioural patterns to develop a profile for every cardholder. Then different classifiers are applied on three different groups later rating scores are generated for every type of classifier. This dynamic changes in parameters lead the system to adapt to new cardholder's transaction behaviours timely. Followed by a feedback mechanism to solve the problem of concept drift. We observed that the Matthews Correlation Coefficient was the better parameter to deal with imbalance dataset. MCC was not the only solution. By applying the SMOTE, we tried balancing the dataset, where we found that the classifiers were performing better than before. The other way of handling imbalance dataset is to use one-class classifiers like one-class SVM. We finally observed that Logistic regression, decision tree and random forest are the algorithms that gave better results.