CREDIT CARD DEFAULT PREDICTION

Rajakumaran S
AlmaBetter

1. Abstract:

Credit risk plays a major role in the banking industry business. Banks' main activities involve granting loan, credit card, investment, mortgage, and others. Credit card has been one of the most booming financial services by banks over the past years. However, with the growing number of credit card users, banks have been facing an escalating credit card default rate. As such data analytics can provide solutions to tackle the current phenomenon and management credit risks. This paper provides a performance evaluation of credit card default prediction. 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. There is often a significant difference between the minimum and maximum values in different features, so Standard scaler is used to scale the features within one range. Data level resampling techniques are employed to overcome the problem of the data imbalance. Various under sampling and oversampling methods are used to resolve the issue of class imbalance. Different machine learning models are also employed to obtain efficient results. This model will help commercial banks, financial organizations, loan institutes, and other decision-makers to predict the loan defaulter earlier.

2. Introduction

Credit risk has traditionally been the greatest risk among all the risks that the banking and credit card industry are facing, and it is usually the one requiring the most capital. This can be proven by industry business reports and statistical data. For example, “The Federal Reserve Bank of New York measures credit card delinquencies based on the percent of balances that are at least 90 days late. For the third quarter of 2019, that rate was about 8%, about the same level as in the previous quarter.” Thus, assessing, detecting and managing default risk is the key factor in 2 generating revenue and reducing loss for the banking and credit card industry.

Despite machine learning and big data have been adopted by the banking industry, the current applications are mainly focused on credit score predicting. The disadvantage of heavily relying on credit score is banks would miss valuable customers who come from countries that are traditionally underbanked with no credit history or new immigrants who have repaying power but lack credit history. According to a literature review report on analyzing credit risk using machine and deep learning models, “credit risk management problems researched have been around credit scoring; it would go a long way to research how machine learning can be applied to quantitative areas for better computations of credit risk exposure by predicting probabilities of default.”

3 The purpose of this project is to conduct quantitative analysis on credit card default risk by using interpretable machine learning models with accessible customer data, instead

of credit score or credit history, with the goal of assisting and speeding up the human decision-making process.

This project possesses various contributions in the domain of credit risk prediction.

1) First, latest dataset has been used to build a machine learning model for credit risk prediction.

2) Second, the data imbalance problem has been explored by comparing the different resampling techniques and evaluate the performance that which the resampling technique has given effective results with a machine learning classifier.

3) Limited work was done on resampling techniques for data balancing in this domain because only a few resampling techniques were employed and also obtained less efficient results.

4) Lastly, the interpretable model is also deployed on the web to ease the different stakeholders. This model will help commercial banks, financial organizations, loan institutes, and other decision-makers to predict the credit defaulter earlier.

3. 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.

4. Data Description

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.

5. Data Exploration

This dataset contains information on default payments, demographic factors, credit limit, history of payments, and bill statements of credit card clients in Taiwan from April 2005 to September 2005. It includes 30,000 rows and 25 columns, and there is no credit score or credit history information Overall, the dataset is very clean, but there are several undocumented column values. As a result, most of the data wrangling effort was spent on searching information and interpreting the columns. The purpose of exploratory data analysis is to identify the variables that impact payment default likelihood and the correlations between them. We use graphical and statistical data exploratory analysis tools to check every categorical variable.

6. Data Cleaning

Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset. When combining multiple data sources, there are many opportunities for data to be duplicated or mislabeled. If data is incorrect, outcomes and algorithms are unreliable, even though they may look correct. There is no one absolute way to prescribe the exact steps in the data cleaning process because the processes will vary from dataset to dataset. But it is crucial to establish a template for your data cleaning process so you know you are doing it the right way every time.

Data cleaning means fixing bad data in your data set.

Bad data could be:

- Empty cells
- Data in wrong format
- Wrong data
- Duplicates

7. Data Visualization

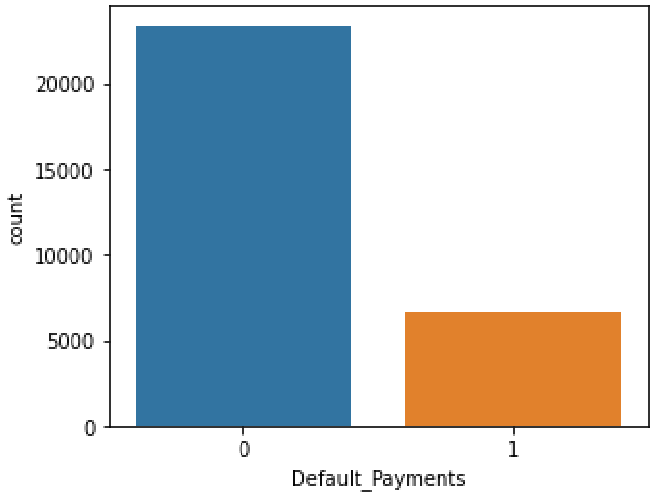
Data visualization is the discipline of trying to understand data by placing it in a visual context so that patterns, trends, and correlations that might not otherwise be detected can be exposed.

Python offers multiple great graphing libraries packed with lots of different features. Whether you want to create interactive or highly customized plots, Python has an excellent library for you.

To get a little overview, here are a few popular plotting libraries:

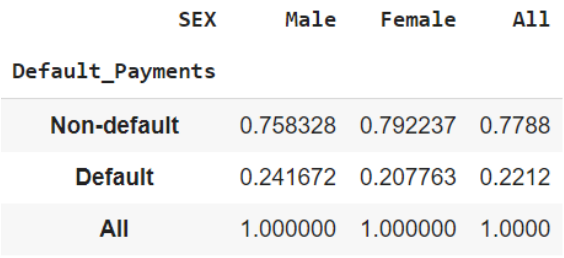
- Matplotlib: low level, provides lots of freedom
- Pandas Visualization: easy to use interface, built on Matplotlib
- Seaborn: high-level interface, great default styles

7.1 Target Variable



From the above visualization, 78% of Non-default customer and 22% of default customer. We have an imbalance dataset.

7.2 Sex variable

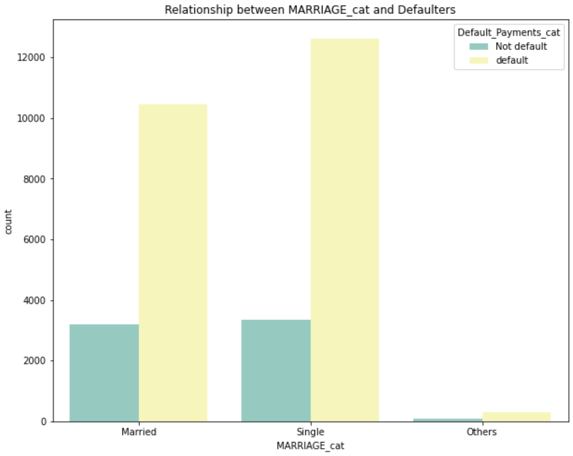


SEX	Male	Female	All
Default_Payments			
Non-default	0.758328	0.792237	0.7788
Default	0.241672	0.207763	0.2212
All	1.000000	1.000000	1.0000

As per above output, More no of female customer have credit card compared to male customer.

Male have more default customer as per ratio of data available.

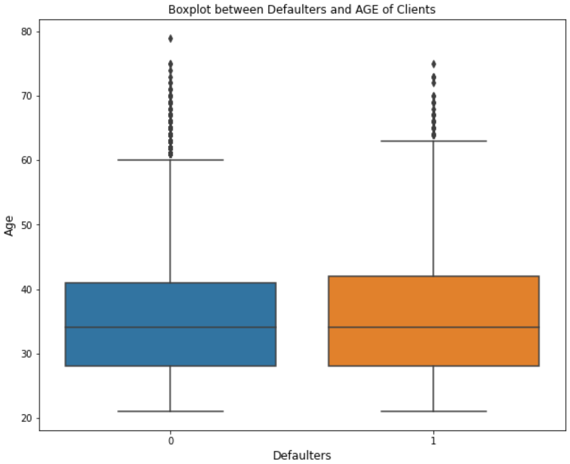
7.3 Marriage variable



From the above visualization, single have more credit card holder. But married customer are more default customer as per data available.

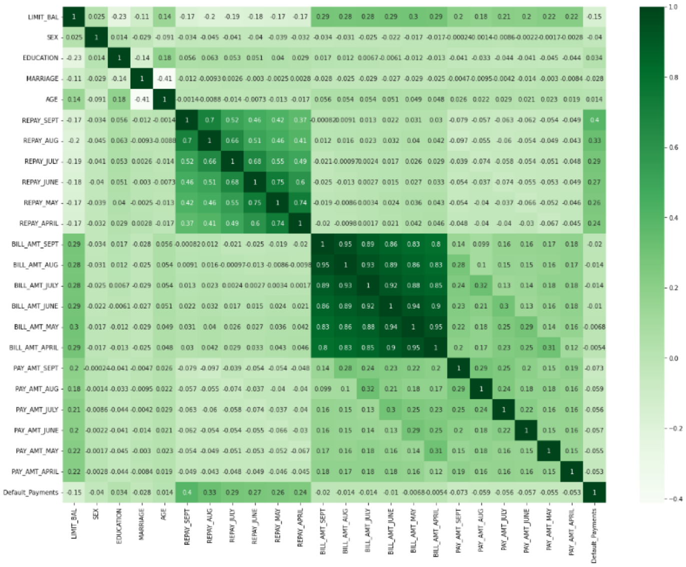
7.4 Age Variable

Most of the customer lies age between 30 to 40 of both default and non-default customer. The outlier of data above age 60 may be these customers are the default.



7.5 Correlation:

Here many features are correlated with each other, but we can't delete those features. Because it contains the past transaction details of the customers.

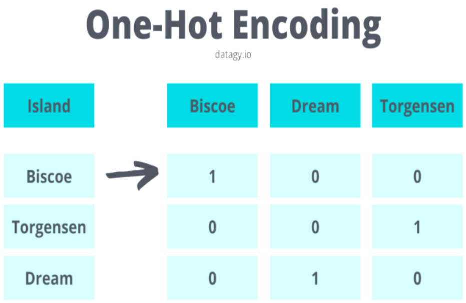


8. Feature Engineering

Feature engineering is the act of converting raw observation into desired features using statistical or machine learning approaches. Feature engineering refers to manipulation-addition, deletion, combination, mutation of our dataset to improve machine learning model training, leading to better performance and greater accuracy. Effective feature engineering is based on sound knowledge of business problem and the available data sources.

i.One hot encoder data

One-Hot encoding is used in machine learning as a method to quantify categorical data. One-hot encoding approach eliminates the order but it causes the number of columns to expand vastly. So, for columns with more unique values try using other techniques like Label Encoding



One-Hot Encoding

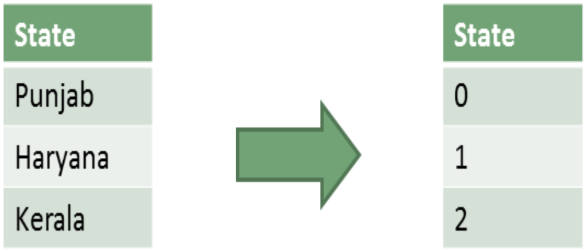
datagy.io

Island
Biscoe
Torgensen
Dream

Biscoe	Dream	Torgensen
1	0	0
0	0	1
0	1	0

ii.Label Encoder

Label Encoding refers to converting the labels into a numeric form so as to convert them into the machine-readable form. Machine learning algorithms can then decide in a better way how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.



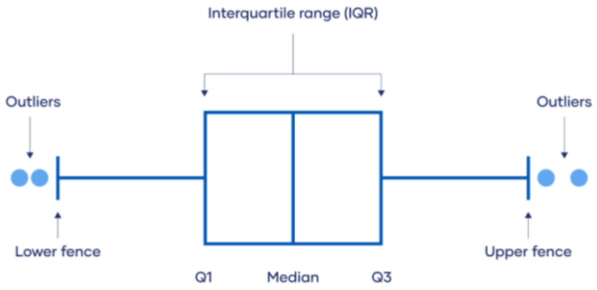
State
Punjab
Haryana
Kerala

State
0
1
2

Outlier:

Outliers is a data point in the dataset that differs significantly from the other data or observation. The thing to remember that, not all outliers are the same. Some have a strong influence, some not at all. Some are valid and important data values. Some are simply errors or noise. Many parametric statistics like mean, correlations, and every statistic based on these is sensitive to

Analysis of outlier



- **Outlier detection**

We use following methods to detect Outlier using Interquartile Range.

Standard Scaler:

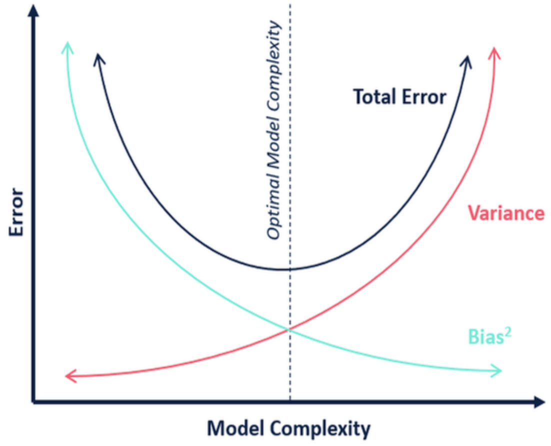
In Machine Learning, Standard Scaler is used to resize the distribution of values so that the mean of the observed values is 0 and the standard deviation is 1. Standard Scaler is an important technique that is mainly performed as a pre-processing step before many machine learning models, in order to standardize the range of functionality of the input dataset.

Overfitting: So, what is overfitting? Well, to put it in more simple terms it's when we built a model that is too complex that it matches the training data "too closely" or we can say that the model has started to learn not only the signal, but also the noise in the data. The result of this is that our model will do well on the training data, but won't generalize to out-of-sample data, data that we have not seen before.

Bias-Variance tradeoff: When we discuss prediction models, prediction errors can be decomposed into two main subcomponents we care about: error due to "bias" and error due to "variance". Understanding these two types of error can help us diagnose model results and avoid the mistake of over/under fitting. A typical graph of discussing this is shown below:

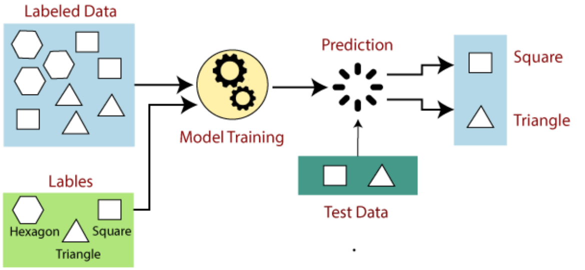
Bias: The red line, measures how far off in general our models' predictions are from the correct value. Thus, as our model gets more and more complex, we will become more and more accurate about our predictions (Error steadily decreases).

Variance: The cyan line, measures how different can our model be from one to another, as we're looking at different possible data sets. If the estimated model will vary dramatically from one data set to the other, then we will have very erratic predictions, because our prediction will be extremely sensitive to what data set, we obtain. As the complexity of our model rises, variance becomes our primary concern.



9. Supervised Learning

In supervised learning, models are trained using labelled dataset, where the model learns about each type of data. Once the training process is completed, the model is tested on the basis of test data (a subset of the training set), and then it predicts the output.



10. Fitting different models

i.Logistic Regression

ii.SVC – Support Vector Classification

iii.KNN - K-Nearest Neighbor

iv. Random Forest

v.XG Boosting

Logistic Regression:

Logistic regression is a classification algorithm that predicts the probability of an outcome that can only have two values (i.e., a dichotomy). A logistic regression produces a logistic curve, which is limited to values between 0 and 1. Logistic regression models the probability that each input belongs to a particular category.

Logistic regression is an excellent tool to know for classification problems, which are problems where the output value that we wish to predict only takes on only a small number of discrete values. Here we'll focus on the binary classification problem, where the output can take on only two distinct classes.

In Logistic Regression, the log-odds of a categorical response being "true" (1) is modeled as a linear combination of the features:

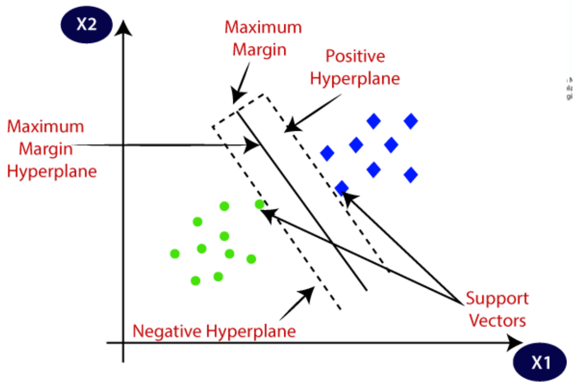
$$P = \frac{e^{a+bX}}{1+e^{a+bX}}$$

Support Vector Machine Algorithm

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

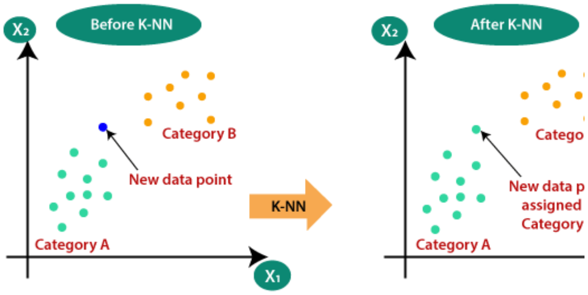
The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane



K-Nearest Neighbor (KNN)

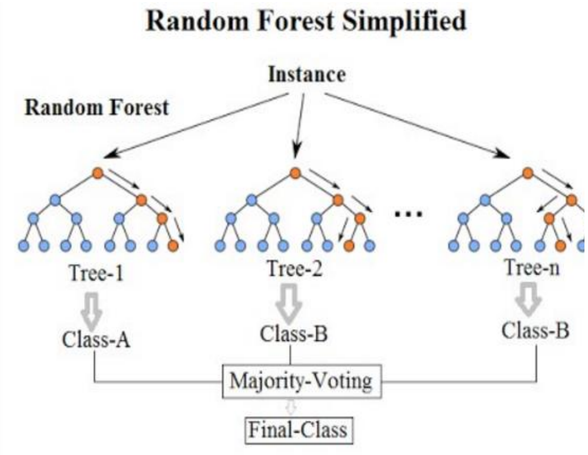
- K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.



Random Forest:

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique**.**

"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."



Extreme Gradient Boosting Machine (XGBM)

XGBM is the latest version of gradient boosting machines which also works very similar to GBM. In XGBM, trees are added sequentially (one at a time) that learn from the errors of previous trees and improve them. Although, XGBM and GBM algorithms are similar in look and feel but still there are a few differences between them as follows:

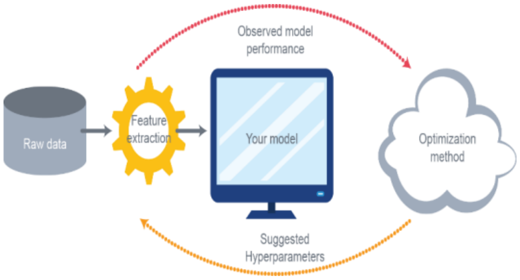
○ XGBM uses various regularization techniques to reduce under-fitting or over-fitting of the model which also increases model performance more than gradient boosting machines.

○ XGBM follows parallel processing of each node, while GBM does not which makes it more rapid than gradient boosting machines.

○ XGBM helps us to get rid of the imputation of missing values because by default the model takes care of it. It learns on its own whether these values should be in the right or left node.

Hyper parameter tuning:

A Machine Learning model is defined as a mathematical model with a number of parameters that need to be learned from the data. By training a model with existing data, we are able to fit the model parameters. However, there is another kind of parameter, known as Hyperparameters, that cannot be directly learned from the regular training process. They are usually fixed before the actual training process begins. These parameters express important properties of the model such as its complexity or how fast it should learn.



Hyperparameters are those parameters that are explicitly defined by the user to control the learning process. Some key points for model parameters are as follows:

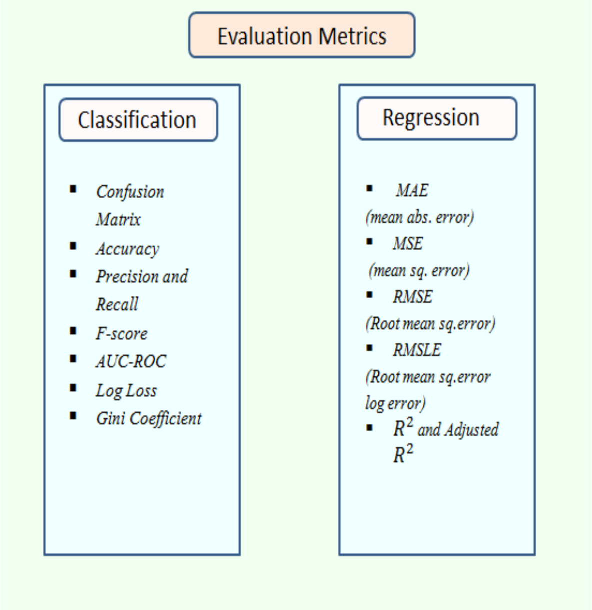
- These are usually defined manually by the machine learning engineer.
- One cannot know the exact best value for hyperparameters for the given problem. The best value can be determined either by the rule of thumb or by trial and error.
- Some examples of Hyperparameters are the learning rate for training a neural network, K in the KNN algorithm.

Grid Search CV:

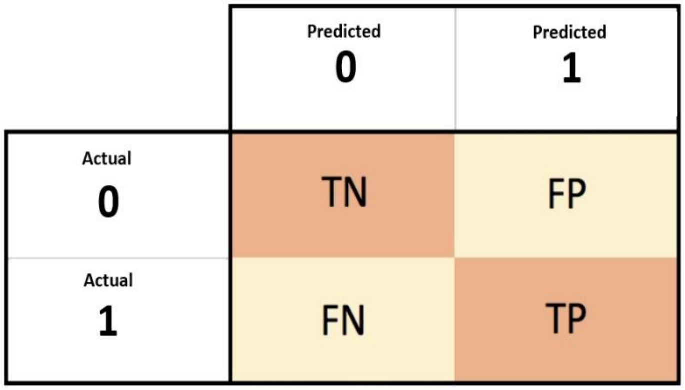
The Grid Search Method considers some hyperparameter combinations and selects the one returning a lower error score. This method is specifically useful when there are only some hyperparameters in order to optimize. However, it is outperformed by other weighted-random search methods when the Machine Learning model grows in complexity.

Grid Search is an optimization algorithm that allows us to select the best parameters to optimize the issue from a list of parameter choices we are providing, thus automating the 'trial-and-error' method. Although we can apply it to multiple optimization issues; however, it is most commonly known for its utilization in machine learning in order to obtain the parameters at which the model provides the best accuracy.

11. Model Evaluation:



Evaluation Metrics:



	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

Accuracy: Accuracy will require two inputs (i) actual class labels (ii) predicted class labels. To get the class labels from probabilities (these probabilities will be probabilities of getting a HIT), you can take a threshold of 0.5. Any probability above 0.5 will be labelled as class 1 and anything less than 0.5 will be labelled as class 0.

Precision: Precision for a label is defined as the number of true positives divided by the number of predicted positives. Report precision in percentages.

Recall: Recall for a label is defined as the number of true positives divided by the total number of actual positives. Report recalls in percentages.

F1-Score: This is defined as the harmonic mean of precision and recall.

AUC-ROC: The Receiver Operator Characteristic (ROC) curve is an evaluation metric for binary classification problems. It is a probability curve that plots the TPR against FPR at various threshold values and essentially 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 and is used as a summary of the ROC curve.

KS Statistics

It stands for Kolmogorov–Smirnov which is named after Andrey Kolmogorov and Nikolai Smirnov. It compares the two cumulative distributions and returns the maximum difference between them. It is a non-parametric test which means you don't need to test any assumption related to the distribution of data. In KS Test, Null hypothesis states null both cumulative distributions are similar. Rejecting the null hypothesis means cumulative distributions are different.

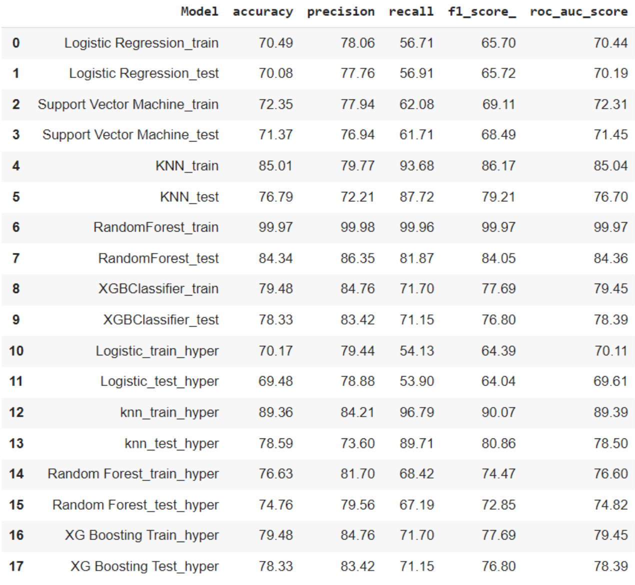
In data science, it compares the cumulative distribution of events and non-events and KS is where there is a maximum difference between the two distributions. In simple words, it helps us to understand how well our predictive model is able to discriminate between events and non-events.

Suppose you are building a propensity model in which objective is to identify prospects who are likely to buy a particular product. In this case, dependent (target) variable is in binary form which has only two outcomes : 0 (Non-event) or 1 (Event). "Event" means people who purchased the product. "Non-event" refers to people who didn't buy the product. KS Statistics measures whether model is able to distinguish between prospects and non-prospects.

12. Conclusion:

This study focused on predicting Credit Card Default Prediction using given dataset. Logistic Regression, Support Vector machine, K-Nearest Neighbor Random Forest and XG Boosting Classifier are used to predict. This statistical data analysis shows interesting outcomes in prediction method and also in an exploratory analysis.

hence the prediction from the logistic model, KNN and Support vector Machine was low performance compare to Random Forest and XG Boosting Classifier.



	Model	accuracy	precision	recall	f1_score_	roc_auc_score
0	Logistic Regression_train	70.49	78.06	56.71	65.70	70.44
1	Logistic Regression_test	70.08	77.76	56.91	65.72	70.19
2	Support Vector Machine_train	72.35	77.94	62.08	69.11	72.31
3	Support Vector Machine_test	71.37	76.94	61.71	68.49	71.45
4	KNN_train	85.01	79.77	93.68	86.17	85.04
5	KNN_test	76.79	72.21	87.72	79.21	76.70
6	RandomForest_train	99.97	99.98	99.96	99.97	99.97
7	RandomForest_test	84.34	86.35	81.87	84.05	84.36
8	XGBClassifier_train	79.48	84.76	71.70	77.69	79.45
9	XGBClassifier_test	78.33	83.42	71.15	76.80	78.39
10	Logistic_train_hyper	70.17	79.44	54.13	64.39	70.11
11	Logistic_test_hyper	69.48	78.88	53.90	64.04	69.61
12	knn_train_hyper	89.36	84.21	96.79	90.07	89.39
13	knn_test_hyper	78.59	73.60	89.71	80.86	78.50
14	Random Forest_train_hyper	76.63	81.70	68.42	74.47	76.60
15	Random Forest_test_hyper	74.76	79.56	67.19	72.85	74.82
16	XG Boosting Train_hyper	79.48	84.76	71.70	77.69	79.45
17	XG Boosting Test_hyper	78.33	83.42	71.15	76.80	78.39

Best predictions are obtained with an **Random Forest** model with a **Precision** score for train is **81%** and test score is **79%**, Recall score of for train is **74%** and test score is **72%** and KS-Statistic value is 0.502**.**

XGB Classifier model with a **Precision** score for train is **94%** and test score is **90%** and Recall score of for train is **85%** and test score is **81%** and KS-Statistic value is 0.725**.** as per the result of all model evaluation our best model **XGB Classifier.**

13. References:

i. https://stackoverflow.com/

ii. https://www.almabetter.com/

iii. https://www.javatpoint.com/

iv. https://www.geeksforgeeks.org/machine-learning/