DETAILED PROJECT REPORT
Credit Card Default Prediction

1. Project Overview:

The credit card industry faces the challenge of identifying customers who are likely to default on their credit card payments. Defaulting on credit card payments can lead to financial losses for the credit card company. To mitigate this risk, predictive models can be developed to identify customers who are at a higher risk of defaulting.

Credit card companies face financial risks when customers default on their payments. Predicting default risks helps identify high-risk customers, enabling proactive measures to mitigate potential losses and manage overall risk exposure.

Defaulted credit card accounts result in significant costs for credit card companies, including collection efforts, legal actions, and write-offs. By predicting defaults accurately, companies can reduce these costs by implementing preventive actions or early intervention strategies.

Effective default prediction enables credit card companies to optimize their credit underwriting processes, including credit limit setting and interest rate determination. By accurately assessing default risks, companies can ensure appropriate risk-adjusted pricing, thereby improving profitability. By implementing a credit card default prediction solution, credit card companies can proactively identify customers at a higher risk of default. This enables them to take appropriate actions such as adjusting credit limits, offering financial counselling, or implementing early collection efforts to reduce default rates and associated financial losses. Ultimately, the solution helps improve risk management, customer relationships, and overall profitability for credit card companies.

2. Scope of Detailed Project Report (DPR):

Detailed Project Reports (DPRs) are the outputs of the planning and design phase of a project. DPR is a very detailed and elaborate plan for a project indicating overall project. A DPR is a final, detailed appraisal report on the project and a blueprint for its execution and eventual operation. It provides details of the basic program, the roles and responsibilities, all the activities to be carried out and the resources required and possible risk with recommended measures to counter them.

3 **What causes credit card default**?

1. Financial Hardship: One of the primary reasons for credit card default is financial hardship. This can arise from factors such as job loss, reduced income, medical emergencies, or unexpected expenses. When individuals face financial difficulties, they may struggle to meet their credit card payment obligations, leading to default.

2. Poor Financial Management: Inadequate financial management skills and poor budgeting can contribute to credit card default. If individuals consistently overspend, carry high credit card balances, or fail to prioritize debt repayment, they may find it challenging to make timely payments, ultimately leading to default.

3. Excessive Debt: Accumulating high levels of debt can increase the risk of credit card default. When individuals carry multiple credit cards with significant balances, they may face difficulty managing their overall debt load, resulting in missed payments and eventual default.

4. Interest Rates and Fees: High interest rates and fees associated with credit cards can make it challenging for cardholders to keep up with their payments. If individuals have credit cards with high annual percentage rates (APRs), the interest charges can quickly accumulate, making it difficult to pay off the balance and leading to default.

5. Lack of Financial Literacy: Limited knowledge or understanding of credit card terms, payment obligations, and interest calculations can contribute to default. If individuals are not aware of the consequences of missed payments or do not understand how interest accrues, they may unintentionally fall into default.

6. Life Events: Significant life events such as divorce, job loss, or serious illness can disrupt financial stability and contribute to credit card default. These events often result in reduced income, increased expenses, and heightened financial stress, making it challenging for individuals to keep up with their credit card payments.

3. The 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 must evaluate which customers will default on their credit card payments. 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 faced by commercial banks is the risk prediction of credit clients. To analyze and predict the above given database, the current project is developed. This project is an attempt to identify credit card customers who are more likely to default in the coming month.

4. Data Exploration:

5.1 Data Overview

The dataset of this project work has been published in Kaggle. The dataset is divided into the training and testing datasets. Data Source: https://github.com/prachi24rai/Credit_card_default_prediction

This project is an attempt to identify credit card customers who are more likely to default in the coming month. A lot of credit card issuing companies are working on predictive models which would help them predict the payment status of the customer ahead of time using the customer's credit score, credit history, payment history and other factors. This project is aimed at using customer's personal and financial information like credit line, age, repayment, and delinquency history for the past 6 months to predict the probability of the customer to become default next month. Many statistical and data mining techniques will be used to build a binary predictive model. If the credit card issuing companies can effectively predict the imminent default of customers beforehand, it will help them to pursue targeted customers and take calculated efforts to avoid the default, to overcome future losses efficiently. The data, in any sense, does not directly reveal the identity of any individual or provide information that could be decrypted to connect to an individual. In this project, the plan is to predict the probability of credit-card holders to go into default in the next month by using payment data from October 2015 to March 2016. Among the total 30,000 observations, to determine the binary variable – default payment in April 2016 (Yes = 1, No = 0), as the response variable. Some of the key attributes consisting of those variables which are used for this project are listed below:

Attributes and their 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.

6. Data Pre-Processing:

Before jumping to the exploration stage, we need to perform basic data pre-processing steps like null value imputation and removal of unwanted data.

6.1 Library used:

Pandas ,NumPy , Matplotlib, Seaborn, Sk-learn

The dataset contains 30000 rows and 25 columns. But we only need information about important columns. and dropping other unwanted columns.

6.2 Handling Null Values:

There are a few null values in the dataset. The major part of the dataset is null values which triggers many stumbling blockages in processing the necessary columns, therefore it is required to identify the columns containing lump sum null values and get dropped. So, let’s drop these null values columns which contain more than 80 percent of null values.

7. EXPLORATORY DATA ANALYSIS(EDA):

7.1 Shape:

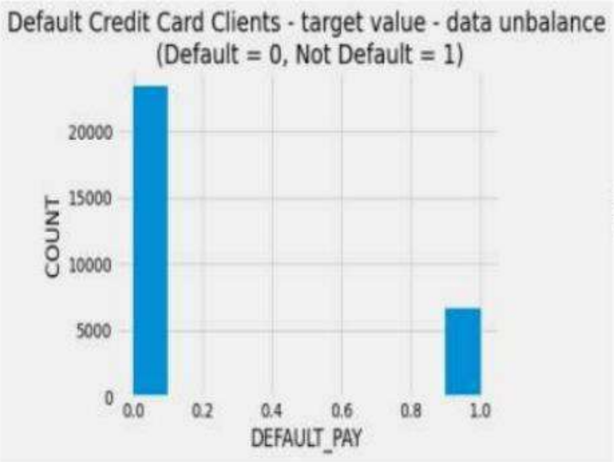
The shape attribute tells us a number of observations and variables we have in the data set. It is used to check the dimension of data. The given data set has more than 30000 observations and 23 variables in the data set.

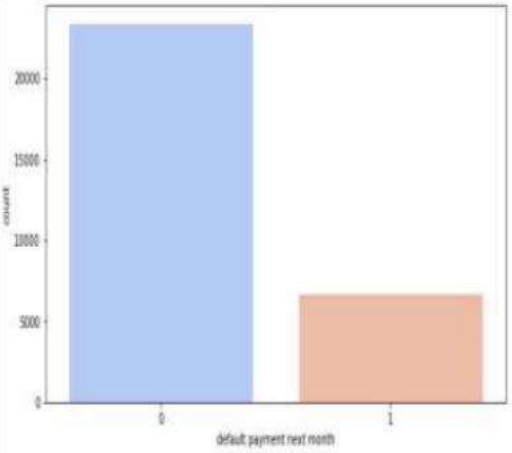
7.2 Describe:

It computes a summary of statistics pertaining to the DataFrame columns. This function gives the mean, std and IQR values. And, function excludes the character columns and gives a summary about numeric columns.

7.3 Analysis on different variables:

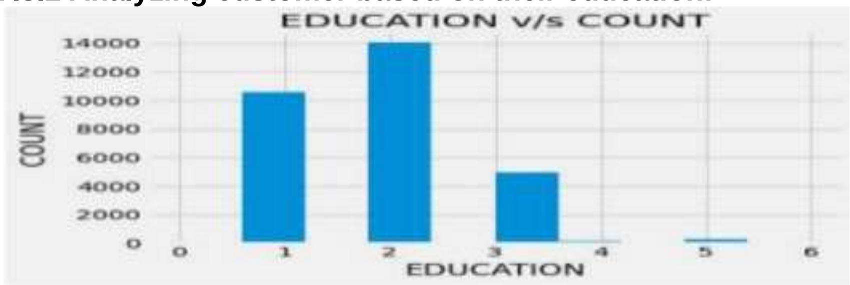
7.3.1 Analyzing the default pay:

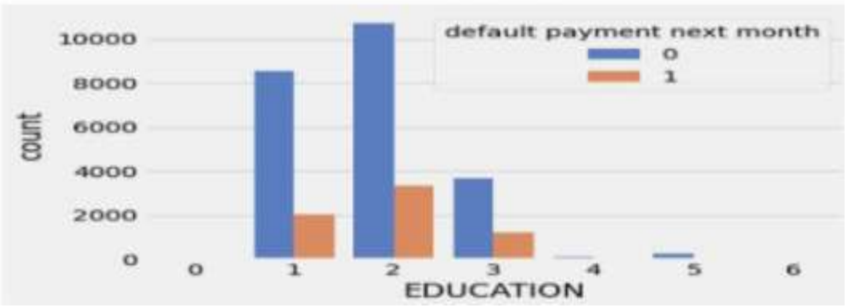




From the above graph on x-axis 0 indicates as not a default payment and 1 indicates the default payment. From this we can say that for more customers there are no default payments for next month. score of the customer: Using this variable will help the model predict defaults more effectively.

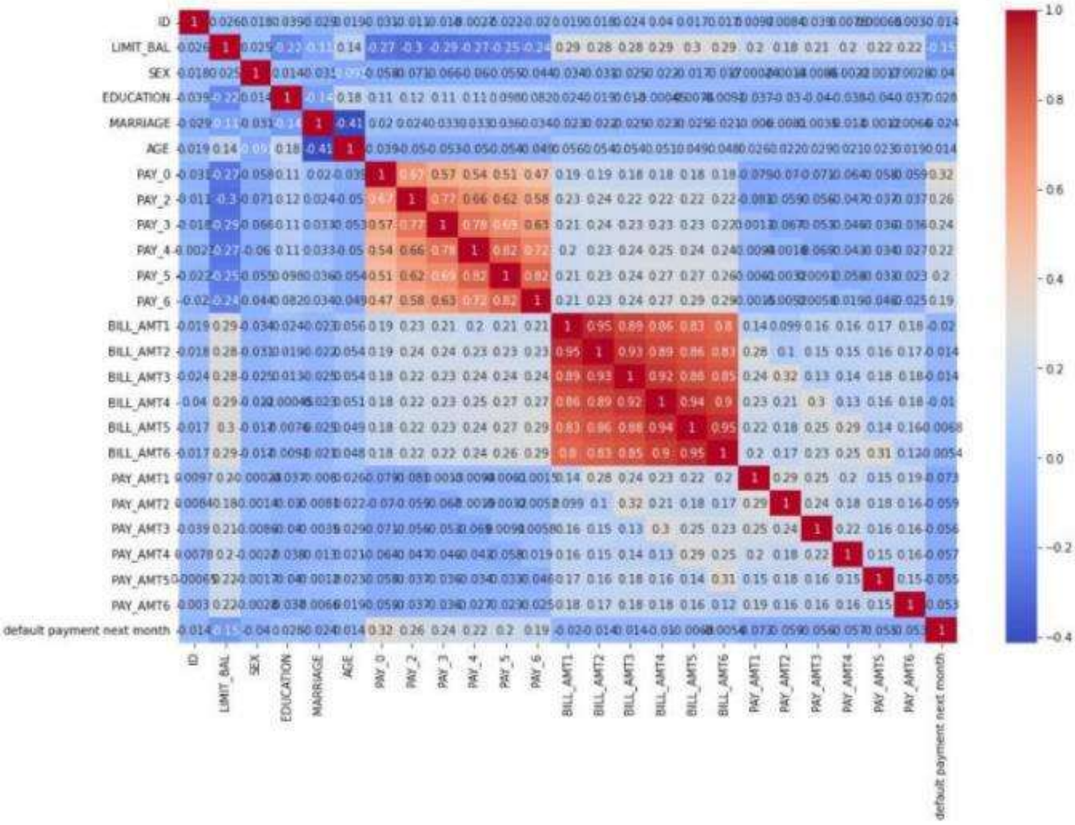
7.3.2 Analyzing customer based on their education:





Education (1 = graduate school; 2 = university; 3 = high school; 4 = others) It might be useful to see whether the education level of the customer is in any way related to his/her probability of default. The distribution of defaults based on education level will be an interesting chart to look at. From the above we can say that most of the people are university educated followed by graduated school. More number of credit holders are university students followed by Graduates and then High school students.

7.4 Correlation:



8. Training the models:

- **Logistic Regression**
- **Support Vector Classification**
- **Decision Tree Classification**
- **Random Forest Classification**
- **K Nearest Neighbors Classification**
- **Naïve Bayes**

The dataset is now ready to fit a model. The training set is fed into the algorithm to learn how to predict values. Testing data is given as input after Model Building a target variable to predict.

For all models based on the below algorithms, 20-fold cross validation can be used. Essentially cross validation provides an indication of how well a model is generalizing to the unseen results.

Testing with Linear and Tree-based models. Linear based Models like Multi–Linear, Ridge, and Lasso

Tree-based Models like Decision Tree, Random Forest, Radiant Boosting.

8.1 Logistic Regression:

Logistic regression is a statistical model used for binary classification. It predicts the probability of an event belonging to one of two classes. It uses the logistic function (sigmoid function) to transform the input variables into a value between 0 and 1, representing the estimated probability. The model provides interpretable coefficients for each input variable, indicating their influence on the predicted probability. Logistic regression is trained using maximum likelihood estimation and can be evaluated using metrics such as accuracy, precision, recall, and the ROC curve.

8.2 Support Vector Classification:

The Support Vector Classifier (SVC), also known as Support Vector Machine (SVM), is a classification algorithm. It finds an optimal hyperplane that separates data points of different classes with the maximum margin. SVC can handle nonlinear data by using the kernel trick, and it aims to find a hyperplane with the largest margin between support vectors. It is trained by solving an optimization problem, and it can classify new data points based on their position relative to the hyperplane.

8.3 Decision Tree Classification:

A decision tree is a supervised machine learning algorithm that uses a tree-like structure to make predictions by partitioning the feature space based on input variables and their values. It consists of nodes representing decisions and branches representing possible outcomes, with leaf nodes containing the final predictions or target values.

8.4 Random Forest Classification:

Random Forest is an ensemble machine learning algorithm that combines multiple decision trees to make predictions. It works by creating a collection of decision trees, each trained on different subsets of the data and features. The final prediction is obtained by aggregating the predictions of all the individual trees. Random Forest improves upon the performance of a single decision tree by reducing overfitting and providing more robust predictions.

8.5 K Nearest Neighbors Classification:

K Nearest Neighbors (KNN) Classification is a simple machine learning algorithm used for classification tasks. It classifies a new data point based on the majority vote of its K nearest neighbors in the feature space. The algorithm measures the distance between the new point and the existing labeled data points, and assigns the class label that is most common among the K nearest neighbors. KNN is a non-parametric algorithm, meaning it doesn't make any assumptions about the underlying data distribution. It is easy to understand and implement, but can be sensitive to the choice of K and the distance metric used for measuring proximity.

8.6 Naïve Bayes:

Naive Bayes is a simple and probabilistic machine learning algorithm used for classification tasks. It is based on the Bayes' theorem, which describes the probability of an event based on prior knowledge. In Naive Bayes, each feature is assumed to be independent of others, hence the term "naive".

The algorithm calculates the probability of a data point belonging to a particular class by multiplying the conditional probabilities of each feature given that class. It then assigns the class with the highest probability as the predicted class for the data point. Naive Bayes is computationally efficient and performs well even with a small amount of training data. It is particularly useful for text classification and spam filtering tasks. However, the assumption of feature independence may not hold in all cases, which can affect its accuracy.

9 Database:

We use Cassandra DB to retrieve, insert, delete, and update the customer input database. Flask is a micro web framework written in python. Here we use the flask to create an API that allows sending data and receives a prediction as a response. Flask API’s act as an interface between the user, ML- Model and DB. Users interact with the deployed model on HEROKU and the result will be displayed.

8. FAQs & Answers

Q1) What's the source?

The dataset of this project work has been published in Kaggle Data Source: https://github.com/prachi24rai/Credit_card_default_prediction

Q2) What was the type of the data?

Total 23 features given in the dataset 15 are numerical and 8 (including the target variable) are categorical features.

Q3) Describe the overall flow of this project?

Pls. refer Page 6 Figure.1 also refer architecture design Document.

Q4) After the File validation what do you do with incompatible files or files which didn't pass the validation?

We concatenated the training and test data CSV file as one, used for the prediction.

Q5) How Logs managed?

In production we used different logs to monitor model training log, prediction log etc.

Q6) What techniques were used for data preprocessing?

Removing unwanted attributes.
Visualising relation of independent variables with each other and output variables.
Checking and changing Distribution of continuous values.
Removing outliers.
Cleaning data and imputing if null values are present.
Converting categorical data into numerical values.
Scaling the data.
Handling class imbalance, to resolve we can use Near miss Under sampling.

Q7) How did the prediction come about?

The testing files are shared by the client. We pass this data to a saved model, then we get the prediction, which is displayed in the prediction page and also those data are stored in Cassandra Database. Random Forest has given the highest –Model Accuracy: 82.22%