Credit Card Defaulter Prediction

High Level Design (HLD)

Tanuja S.Dhope

29Th July 2023

INEURON

1. INTRODUCTION

Credit risk is very important in the banking industry. Banks' primary businesses include lending, credit card, investment, mortgage, and other services. Credit cards have been one of the most successful financial services offered by banks in recent years. However, as the number of credit card users grows, banks face an increasing credit card default rate. In fact, credit card debts are usually the first to get out of hand in such situations due to costly finance charges (compounded on daily balances) and other penalties. We may have missed credit card payments once or twice due to missing due dates or cash flow concerns. But what happens when this goes on for months? How do you predict if a customer will be deficient in the next months? As a result, data analytics can provide answers for dealing with the current issue and managing credit risks. In this project, machine learning based model has been developed to predict customer defaulter based on demographic data like gender, age, marital status and behavioral data like last payments, past transactions etc.

2. PROBLEM STATEMENT

In recent years, credit cards have been one of the most successful financial services given by banks. Banks, on the other hand, risk an increasing credit card default rate as the number of credit card users climbs. The goal is to predict the probability of credit default based on the credit card owner's characteristics and payment history.

Information About Dataset

ID: ID of each client

LIMIT_BAL: Amount of given credit in NT dollars (includes individual and family/supplementary = credit)

SEX: Gender (1=male, 2=female)

EDUCATION: (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)

MARRIAGE: Marital status (1=married, 2=single, 3=others)

AGE: Age in years

PAY_0: Repayment status in September, 2005 (-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)

PAY_2: Repayment status in August, 2005 (scale same as above)

PAY_3: Repayment status in July, 2005 (scale same as above)

PAY_4: Repayment status in June, 2005 (scale same as above)

PAY_5: Repayment status in May, 2005 (scale same as above)

PAY_6: Repayment status in April, 2005 (scale same as above)

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 payment (1=yes, 0=no)

1. DATA REQUIREMENTS

The required data must satisfy the above said columns.

2. TOOLS USED

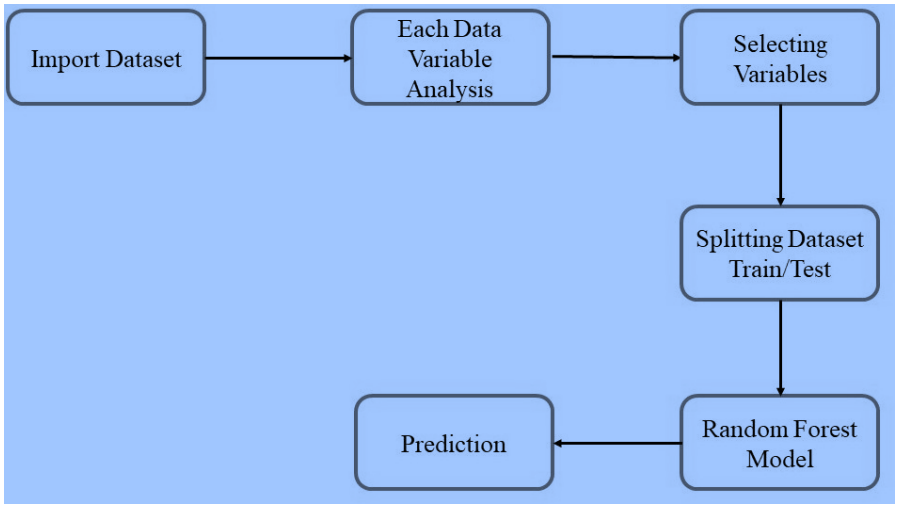
Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, Matplotlib, Seaborn are used to build the whole model.

- VSCODE is used as IDE
- For visualization of plots matplotlib,Seaborn are used
- Front end development is done using HTML and using Flask application.

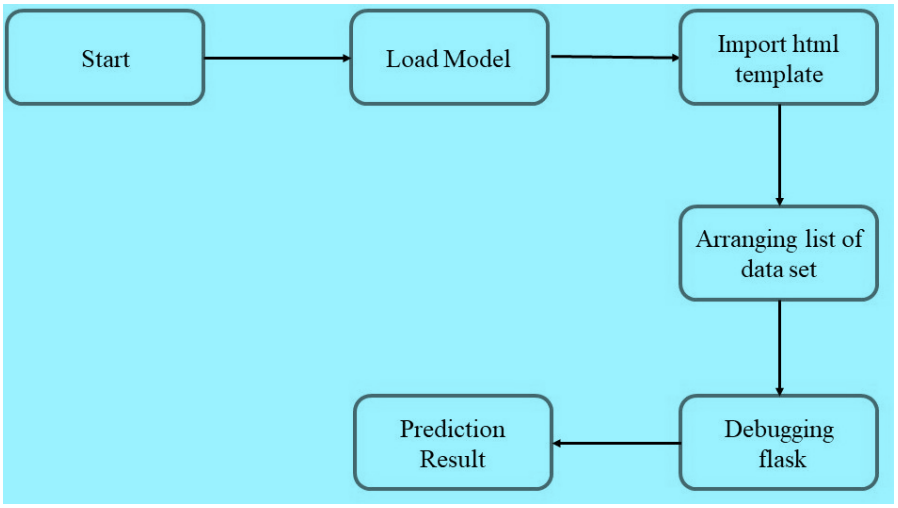
- GITHUB is used as version control system
- Scikit-learn is used for various machine learning models and parameters

3. DESIGN DETAILS

3.1. Process flow



3.2. Deployment process



6.3 Event Log:

The system is logging every event so that user can understand wht process is running internally.example:

```
[ 2023-06-25 23:26:29,224 ] 24 root - INFO - Data Ingestion methods Starts
[ 2023-06-25 23:26:29,786 ] 29 root - INFO - Dataset read as pandas Dataframe
[ 2023-06-25 23:26:30,649 ] 33 root - INFO - Train test split
[ 2023-06-25 23:26:33,129 ] 39 root - INFO - Ingestion of Data is completed
[ 2023-06-25 23:26:33,208 ] 54 root - INFO - Read train and test data completed
[ 2023-06-25 23:26:33,254 ] 57 root - INFO - Obtaining preprocessing object
[ 2023-06-25 23:26:33,254 ] 26 root - INFO - Data Transformation initiated
[ 2023-06-25 23:26:33,254 ] 31 root - INFO - Pipeline Initiated
[ 2023-06-25 23:26:33,254 ] 40 root - INFO - Pipeline Completed
[ 2023-06-25 23:26:33,379 ] 82 root - INFO - Applying preprocessing object on training and testing datasets.
[ 2023-06-25 23:26:33,382 ] 93 root - INFO - Preprocessor pickle file saved
[ 2023-06-25 23:26:33,384 ] 26 root - INFO - Splitting Dependent and Independent variables from train and test data
[ 2023-06-25 23:26:37,431 ] 40 root - INFO - Model Report : {'RandomForestClassifier': 0.8189333333333333}
[ 2023-06-25 23:26:37,431 ] 53 root - INFO - Model Name : RandomForestClassifier , Accuracy Score : 0.8189333333333333
```

6.4 Error Handling

An explanation is displayed for handling errors as to what went wrong?

7.PERFORMANCE

The prediction accuracy is 81% which clearly beneficial for predicting credit card defaulter and helpful to finance domain.

7.1 Reusability

The code written and its components are reusable.

7.2 Application Compatibility

The different components for this project used Python as an interface between them.Each component perform its own task to perform and Python ensures proper transfer of information.

7.3. Resource utilization

Whan any task is performed,it used the processing power available until the function is finished.

7.4 Deployment

Deployment is done on local host using Flask

8.CONCLUSION

Because the project is written in flask, it is open to everyone. The UI is designed to be user-friendly, so the user does not need to know much about any tools and only needs the information for results. Based on the model's predictions, the aforementioned design method will assist banks and loan lenders in predicting whether clients will fail on credit card payments or not, allowing the bank or related departments to take appropriate action.

Credit Card Defaulter Prediction

Low Level Design (LLD)

Tanuja S.Dhope

29Th July 2023

INEURON

1. INTRODUCTION

Credit risk is very important in the banking industry. Banks' primary businesses include lending, credit card, investment, mortgage, and other services. Credit cards have been one of the most successful financial services offered by banks in recent years. However, as the number of credit card users grows, banks face an increasing credit card default rate. In fact, credit card debts are usually the first to get out of hand in such situations due to costly finance charges (compounded on daily balances) and other penalties. We may have missed credit card payments once or twice due to missing due dates or cash flow concerns. But what happens when this goes on for months? How do you predict if a customer will be deficient in the next months? As a result, data analytics can provide answers for dealing with the current issue and managing credit risks. In this project, machine learning based model has been developed to predict customer defaulter based on demographic data like gender, age, marital status and behavioral data like last payments, past transactions etc.

2. PROBLEM STATEMENT

In recent years, credit cards have been one of the most successful financial services given by banks. Banks, on the other hand, risk an increasing credit card default rate as the number of credit card users climbs. The goal is to predict the probability of credit default based on the credit card owner's characteristics and payment history.

Information About Dataset

ID: ID of each client

LIMIT_BAL: Amount of given credit in NT dollars (includes individual and family/supplementary = credit)

SEX: Gender (1=male, 2=female)

EDUCATION: (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)

MARRIAGE: Marital status (1=married, 2=single, 3=others)

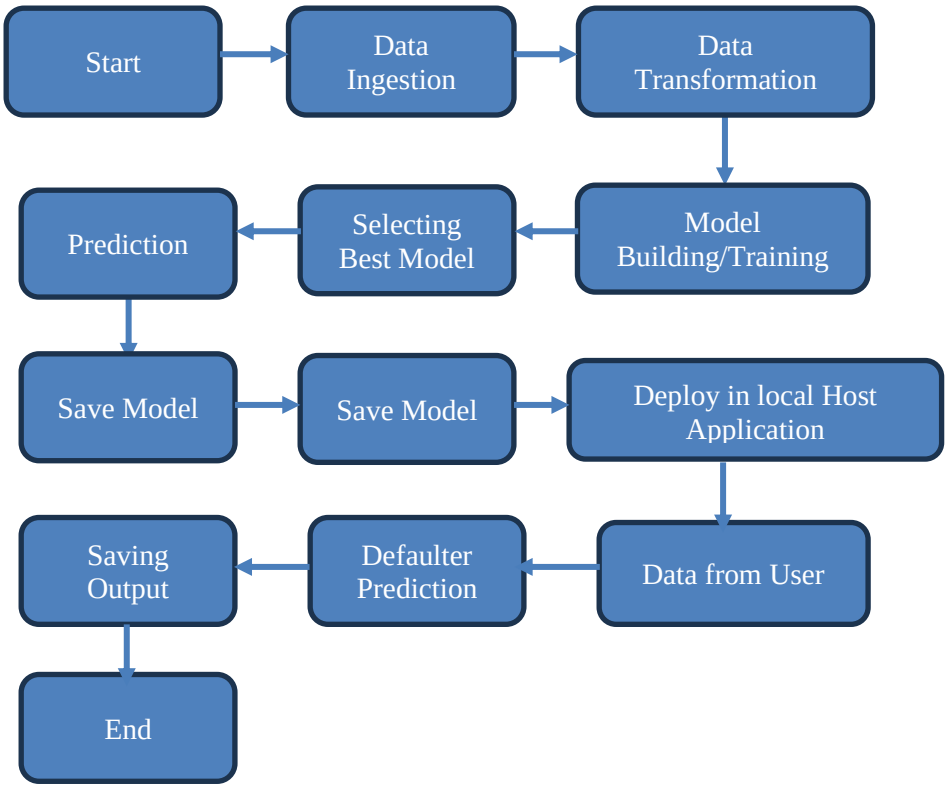
AGE: Age in years

PAY_0: Repayment status in September, 2005 (-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)

PAY_2: Repayment status in August, 2005 (scale same as above)
PAY_3: Repayment status in July, 2005 (scale same as above)
PAY_4: Repayment status in June, 2005 (scale same as above)
PAY_5: Repayment status in May, 2005 (scale same as above)

PAY_6: Repayment status in April, 2005 (scale same as above)
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 payment (1=yes, 0=no)

3.Architecture



4.Architecture Description

4.1 Data Description:

The dataset was taken from Kaggle (URL: https://www.kaggle.com/uciml/default- of-credit-card-clients-dataset), 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.

4.2 Data Ingestion

This included importing of important libraries such as seaborn, matplotlib, pandas ,scikit-learn etc. We imported the same dataset mentioned above from Kaggle.

4.3 Data Pre-processing and Transformation

In data pre-processing the checking and handling the null values, changes in the column names has been done. The multiple graphs in seaborn, matplotlib have been done and other visualization library for proper understanding of the data and the distribution of information in the same and finding the correlation among the independent variable,has been done. As there were no null values in the data, I proceeded with the visualization and analysis.For each specific feature the data visualization has been carried out , and the important key points which can impact the final predictions ,jotted down.The observations noted in data analysis has been utilized for feature engineering viz the information's which has multicollinearity , were dropped which unnecessary increases the training .Thus transforming raw data into features that are suitable for machine learning models.

Train/Test Split

This library was imported from Sklearn to divide the final dataset into the ratio of 75-25%, where 75% of the data was used to train the model and the latter 25% was used to predict the same.

4.6 Model Building

We tried and tested multiple models such as XGBoost, Random Forest,Decision Tree, ADABoost for the model and came up with the model with the best performance, i.e the Random Forest Classifier.

4.7 Selecting Best Model

The Random Forest Classifier(RFC) with hyper tuning of parameters has been done to select the best RFC model.

4.8 Prediction

The selected best model of RFC provides accuracy **81.7%** and the F1 score was **47.3%.**

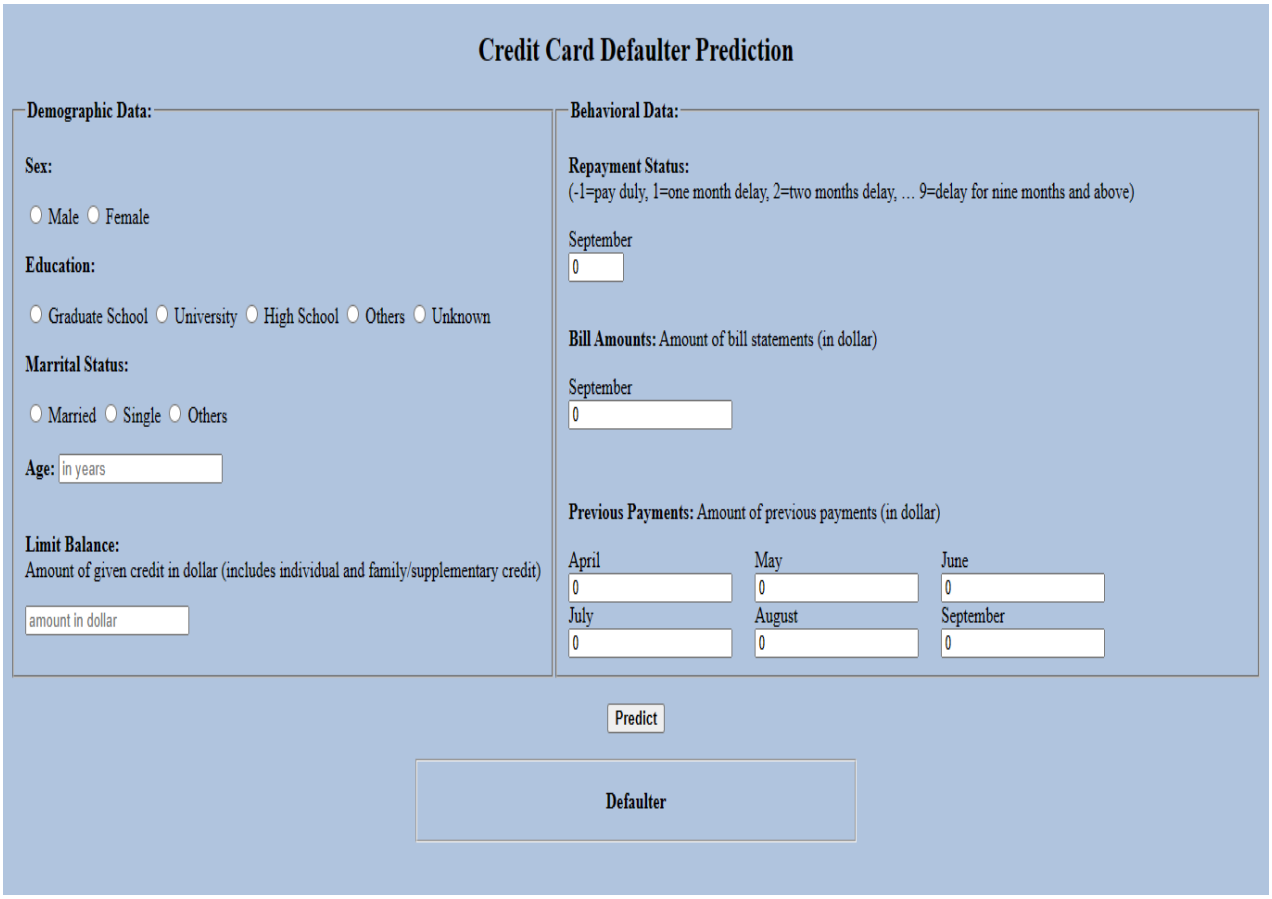
4.9 Save Model

Model was saved using the pickle library which saves the file in a binary mode.

4.10 Deploy in Local Host

We created a HTML template and deployed the model through Flask.

Here is the image of the same:



4.11 Data from User

User entry from user had been done.

4.12 Predication

The data from user has been verified and fed to prediction pipline .

4.13 Saving Output

The predicted output whether customer is Defaulter or Not Defaulter has been saved and displayed on the form.