CREDIT CARD DEFAULT PREDICTION

LOW LEVEL DESIGN

BY
ANSHUL MEHRA

SEPTEMBER 9, 2023
PHYSICS WALLAH

1. INTRODUCTION

There are times when even a seemingly manageable debt, such as credit cards, goes out of control. Loss of job, medical crisis or business failure are some of the reasons that can impact your finances. In fact, credit card debts are usually the first to get out of hand in such situations due to hefty finance charges (compounded on daily balances) and other penalties. A lot of us would be able to relate to this scenario. We may have missed credit card payments once or twice because of forgotten due dates or cash flow issues. But what happens when this continues for months? How to predict if a customer will be defaulter in next months? To reduce the risk of Banks, this 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

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. The goal is to predict the probability of credit default based on credit card owner's characteristics and payment history.

3. DATASET INFORMATION

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)

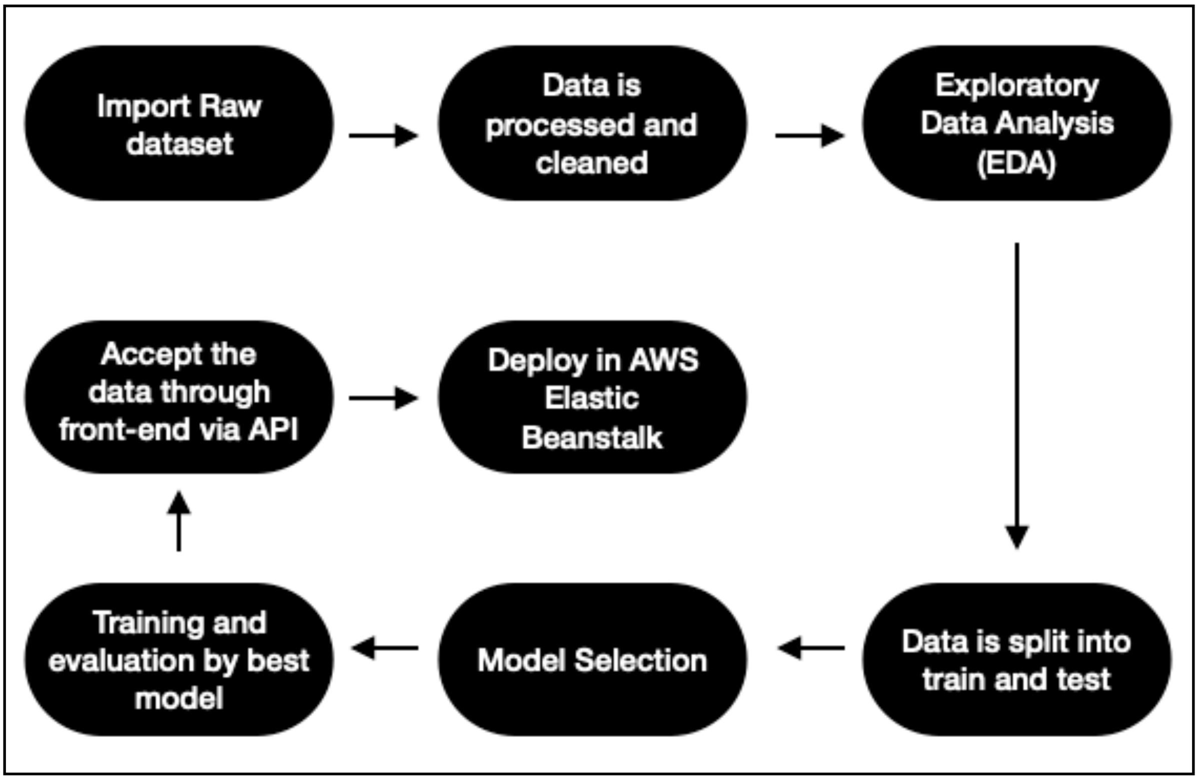
PAY_AMT3: Amount of previous payment in July, 2005 (NT dollar)

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)



Architecture of the Project

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 Pre-processing.

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

4.3.Data Analysis

Here we handled the null values, changed the column names, plotted multiple graphs in seaborn, matplotlib and other visualization library for proper understanding of the data and the distribution of information in the same. As there were no null values in the data, we proceeded with the visualization and analysis.

For each specific feature we analysed the data using visualization, and jotted down the important key points which can impact the final predictions.

4.4. Feature Engineering

Merging 2 or mode columns to get indepth knowledge and information regarding the data.

4.5.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 2% was used to predict the same.

4.6 Selecting Model

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

4.7. Prediction

The Accuracy of Gradient Boosting Classifier was **81.2%.**

4.8 Save Model

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

4.9 Deploy in AWS Elastic BeanStalk

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

Here are the images of the same: