CREDIT CARD DEFAULT PREDICTION

Low Level Design (LLD)

Chinta Krishna Mourya.

06/03/2023

Contents

1. Introduction

1.1. What is Low-Level design document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Food Recommendation System. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

1.2. Scope

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software

architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

2. Project

2.1 Project Introduction

In the current society, banks often face the risk of credit card defaults, which can result in significant financial losses and penalties. When a customer defaults on their credit card payments, the bank loses the potential interest income and may have to write off the debt as a loss. Additionally, the bank may incur legal and administrative expenses associated with debt collection. To mitigate these risks, banks can develop machine learning techniques to predict customer defaults based on demographic and behavioral data, such as age, gender, payment history, and transaction patterns. By leveraging these predictive models, banks can take proactive measures to prevent defaults and minimize their financial losses.

2.2 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 faces 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.

2.3 Dataset Description

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)

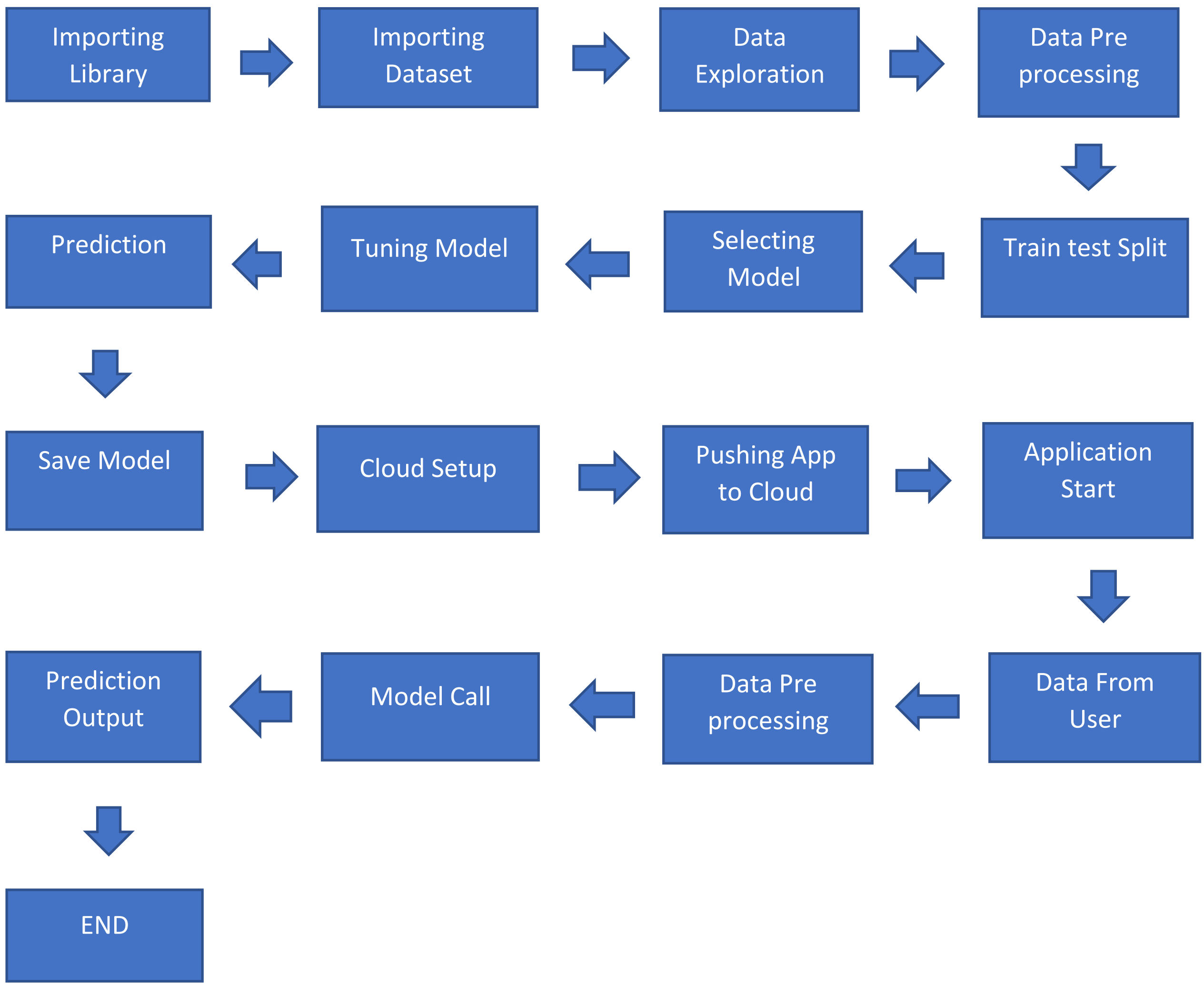
PAY_AMT4: Amount of previous payment in June, 2005 (NT dollar)

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

Data Description

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. Collected from Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science. Dataset consists of 30000 rows and 25 columns.

Data Exploration

This step includes the understanding of the data and the insights from the data which helps us to find the relations among the features and the distribution of the data.

Data Pre-processing

It includes cleaning the dataset like manipulating the columns to our comfort and removing duplicates, missing values, balancing the dataset and scaling the data to get more accurate results.

Train test Split

Splitting the dataset in the ratio 75:25, where 75% is used for training and 25% for testing the model.

Selecting Model

This step involves trying the different models and selecting the model that is performing better.

Tuning Model

The selected model is tuned with different hyperparameters using Grid Search to select the best parameters. Now the selected is tuned with these parameters and trains it.

Prediction

Now, the model is called to predict the Test split and measures the accuracy etc. The accuracy for this model is 86%.

Save Model

This step involves saving the model, so that it can be called when user input is given.

Pushing App to cloud

Selecting the cloud platform AWS and deployment to the cloud by configuring the necessary dependencies, libraries, and environment variables.

Data From User

The user input is now goes to pre processing step like scaling the data to suit the model and goes for prediction.

Prediction Output

Model will predict by user input and gives the output.