HIGH LEVEL DESIGN(HLD)

CREDIT CARD DEFAULT PREDICTION

Ineuron Internship

Dhruv Bakshi

23/03/2023

Contents

Abstract 3

1. Introduction 4

1.1. Why this High-Level Design document? 4

1.2. Scope 4

2. Problem Statement 4

3. Dataset information 5

4. Software and Account Requirement 6

5. Tools used 6

6. Design Details 7

6.1. Code flow in pipeline stages 7

6.2. Pipeline Stages 7

6.3. Pipeline Process 8

6.4. Local application deployment process 9

6.5. Logging 9

6.6. Exception Handling 9

7. Conclusion 10

8. Future Scope 10

Abstract

In these uncertain financial times managing risk is becoming utmost priority of financial institutions. Banks and credit card companies face the impact of users defaulting on their payments now and then. Some users may default on payment once or twice due to some genuine reasons (like health emergency, loss of job etc.) and repay later. Some others may default payment continuously. Hence there is need to predict whether the user will default on payments based on his/her previous financial history and demographic data. Machine learning model tries to provide a solution to this problem.

1. Introduction

1.1. Why this High-Level Design document?

The goal of HLD or a High-Level Design (HLD) Document is to add the necessary detail to the Credit card default prediction project description to represent a suitable model for coding. This document is also intended to help the programmer prior to coding and can be used as a reference manual for how the modules interact at a high level.

1.2. Scope

High-level design (HLD) 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. 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.

3. Dataset information

Dataset URL : https://www.kaggle.com/uciml/defaultof-credit-card-clients-dataset

The dataset has 30,000 rows and 25 columns. There are 25 columns details are:

- 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)

4. Software and Account Requirement

Vscode is used as source code editor. Gitcli and Github acoount is used for code versioning. DVC is used for data versioning. Kaggle account is used for generating api token and further downloading dataset. Dagshub account is used for generating remote mlflow tokens.

5. Tools used

Python programing language is used for writing codes. Jupyter notebook is used for writing raw codes, EDA and experimentations. Pandas, numpy, sklearn are major libraries used in the complete code.

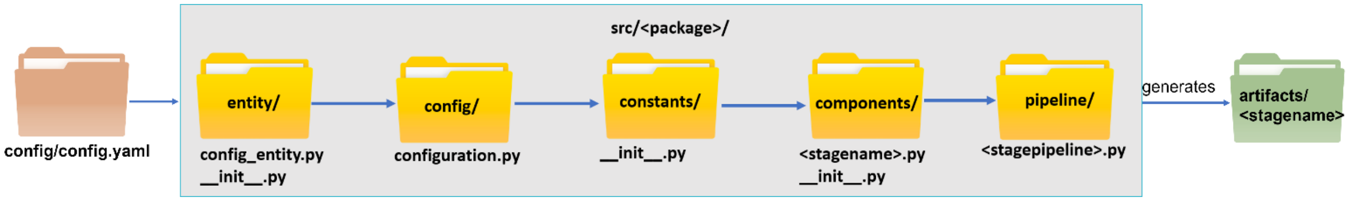
For report generation evidently and pandas profiling are used which generated data drift report and eda reports (Test_profile_report, Train_profile_report) repectively.

For experimentation on machine learning model mlfow is used.

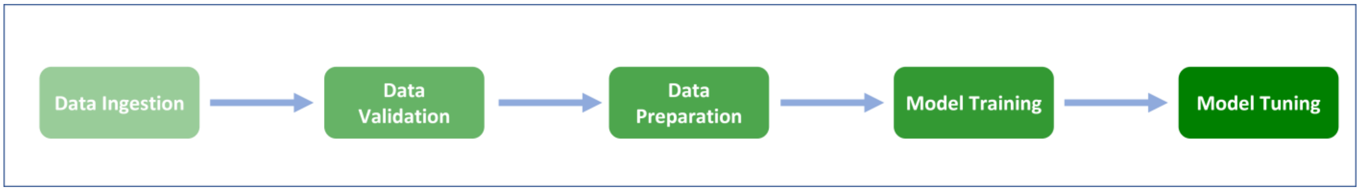
Flask is used for application development.

6. Design Details

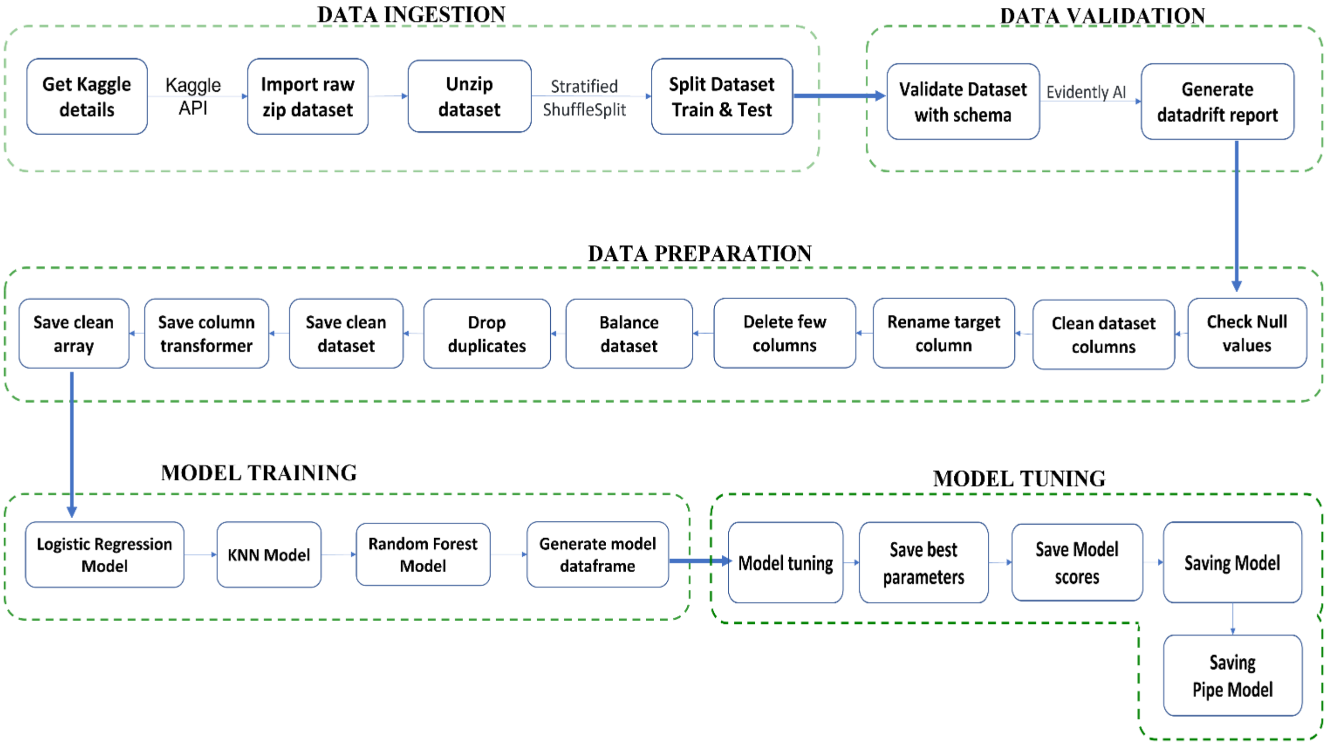
6.1. Code flow in pipeline stages



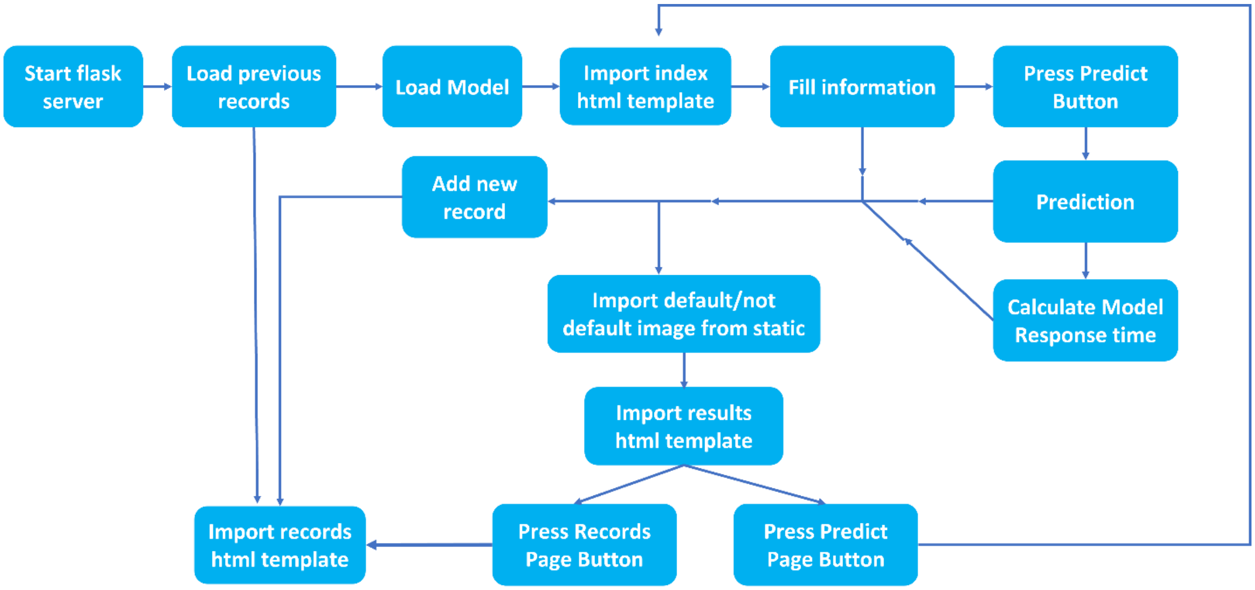
6.2. Pipeline Stages



6.3. Pipeline Process



6.4. Local application deployment process


Start flask server
Load previous records
Load Model
Import index html template
Fill information
Press Predict Button
Add new record
Prediction
Import default/not default image from static
Calculate Model Response time
Import results html template
Import records html template
Press Records Page Button
Press Predict Page Button

6.5. Logging

Using logging module of python, a logger is created. Various log statements with logger.info are saved in folder named logs in the form of text files named running_logs.

6.6. Exception Handling

Code is written with try and except statement to catch and handle exceptions

7. Conclusion

Based on the problem statement a flask application is developed. The application has very simple user interface and can be easily operated by anyone. Usage of application does not require any technical knowledge. The application could be used by banks or credit card companies to predict whether user will default or not next month. This information could be used to take necessary actions by authorities against the default users. The application also creates a new record every time it is used for prediction and saves it alongwith timestamp and response time. This could be used for future reference.

8. Future Scope

Few ideas that could be executed in future are:

- Data and the application could be hosted on cloud.
- Application could also be dockerized.
- The user entered records could be further used to train model.
- More ML models could be experimented with to predict result.
- Two or more models could be used in combination to predict result.