Credit Card Default Prediction

Revision Number: 1.0

Last date of revision: 06/08/2022

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 06/08/2022 | 1 | Initial HLD — V1.0 | Vignesh R |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[Document Version Control. 2](#_TOC_250021)

[Abstract. 4](#_TOC_250020)

1. [Introduction 5](#_TOC_250019)
   1. [Why this High-Level Design Document?. 5](#_TOC_250018)
   2. [Scope. 5](#_TOC_250017)
   3. [Definitions 5](#_TOC_250016)
2. [General Description. 6](#_TOC_250015)
   1. [Product Perspective 6](#_TOC_250014)
   2. [Problem statement 6](#_TOC_250013)
   3. [PROPOSED SOLUTION 6](#_TOC_250012)
   4. [Technical Requirements. 6](#_TOC_250010)
   5. [Data Requirements 6](#_TOC_250009)
   6. [Tools used. 7](#_TOC_250008)
      1. Hardware Requirements. 7
   7. [Constraints 8](#_TOC_250007)
   8. [Assumptions. 8](#_TOC_250006)
3. [Design Details 9](#_TOC_250005)
   1. [Process Flow. 9](#_TOC_250004)
      1. [Model Training and Evaluation 10](#_TOC_250003)
   2. [Event log 10](#_TOC_250001)
   3. [Error Handling 10](#_TOC_250000)
   4. Performance. 10
   5. Reusability. 11
   6. Application Compatibility 11
   7. Resource Utilization 11
   8. Deployment. 11
4. Conclusion 11

## Abstract

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.

1. **Introduction**

###### Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

* + - Present all of the design aspects and define them in detail
    - Describe the user interface being implemented
    - Describe the hardware and software interfaces
    - Describe the performance requirements
    - Include design features and the architecture of the project
    - List and describe the non-functional attributes like: o Security
      * Reliability
      * Maintainability
      * Portability
      * Reusability
      * Application compatibility
      * Resource utilization
      * Serviceability

##### Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

* 1. **Definitions**

Term

*Database*

*IDE*

*Description*

Collection of all the information monitored by this system

Integrated Development Environment

### General Description

#### Product Perspective

The credit card default system is a machine learning-based classification model which will help us to defaulters among the credit card owners.

* 1. Problem statement

The goal is to predict the probability of credit default based on credit card owner's

characteristics and payment history.

* 1. PROPOSED SOLUTION

The solution proposed here is a credit card default prediction system for the above mentioned use case. This is a complete pipeline solution which involves steps like Data Ingestion, Data Validation, Data Transformation, Model Training, Model Evaluation, Model Push along with the prediction module.

#### Technical Requirements

This document addresses the requirements for predicting the credit card defaulters. The technical requirements states that the below items are required.

1. IDE for python environment (VS code)
2. Python 3.7
3. Packages mentioned in requirements.txt
   1. **Data Requirements**

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.

There are 25 variables:

- 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)

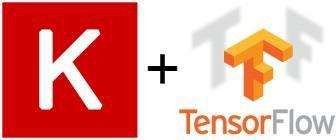
- PAY\_AMT5: Amount of previous payment in May, 2005 (NT dollar)

- PAY\_AMT6: Amount of previous payment in April, 2005 (NT dollar)

- default.payment.next.month: Default payment (1=yes, 0=no)

* 1. Tools used

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



Logo

Description automatically generatedIcon

Description automatically generated A picture containing text, clipart

Description automatically generated

* + - VS code is used as IDE.
    - For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
    - Heroku is used for deployment of the model.
    - Front end development is done using HTML/CSS
    - Python Flask is used for backend development.
    - GitHub is used as version control system.
    1. Hardware Requirements
       - PC

#### Constraints

The Credit card default prediction system must be user friendly, as automated as possible and users should not be required to know any of the workings.

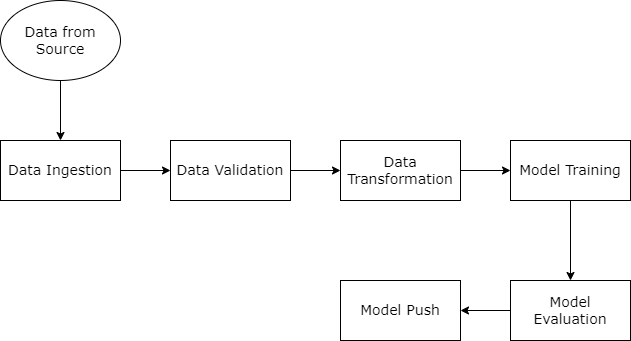
#### Assumptions

The main objective of the project is to implement the use cases as previously mentioned (2.2 Problem Statement) for new dataset that comes through

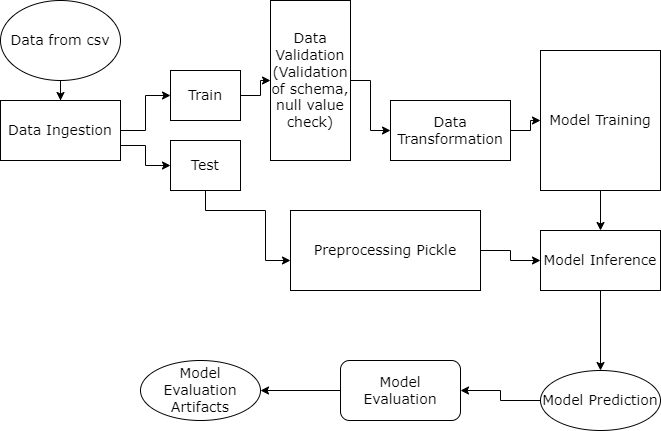
## Design Details

##### Process Flow

Proposed methodology



##### Model Training and Evaluation



* 1. Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

1. The System identifies at what step logging required
2. The System should be able to log each and every system flow.
3. Developer can choose logging method. You can choose database logging/ File logging as well.
4. System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.
   1. **Error Handling**

Should errors be encountered, an explanation will be displayed as to what went wrong? An error will be defined as anything that falls outside the normal and intended usage.

## Performance

The credit card default prediction system is a complete pipeline system where users can do prediction on the credit card default based on the inputs with around 90% accuracy with good precision and recall values. Also, model retraining is very important to improve the performance so it can be done using the tool anytime it is required.

#### Reusability

The code written and the components used should have the ability to be reused with no

problems.

#### Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

#### Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

#### Deployment

Icon

Description automatically generated

## Conclusion

The credit card default prediction system is a complete pipeline system where users can do prediction on the credit card default based on the inputs with around 90% accuracy with good precision and recall values