CREDIT CARD DEFAULT PREDICTION

High Level Design (HLD)

Chinta Krishna Mourya

06/03/2023

Contents

Abstract

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.

1. **Introduction**
   1. **Why this high Level Document?**

The purpose of this High level Design (HLD) Document is to add 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 at the high level.

* 1. **Scope**

The HLD document 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 | Description |
| Credit Card Default | credit card holder fails to pay the required amount on their credit card by the due date |
| IDE | Integrated development Environment |
| AWS | Amazon Web Services |

**2. Project Introduction**

* 1. **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.

* 1. **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\_0,1,2,3,4,5,6**: Repayment status in September,August, July, June, May, April , 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)

**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)

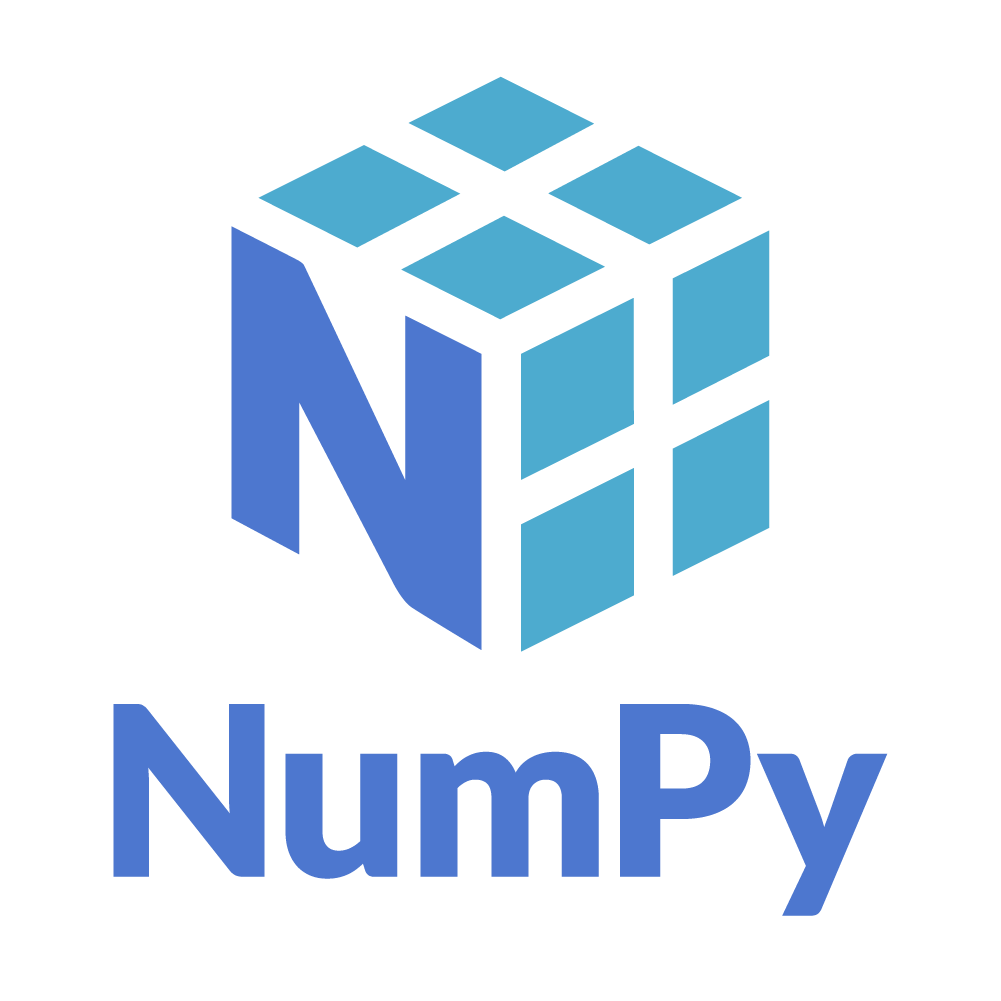
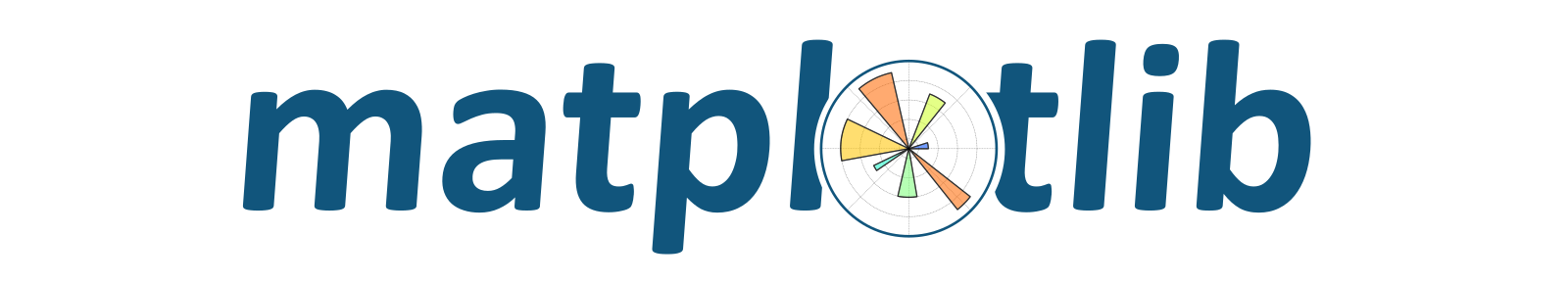
**PAY\_AMT5**: Amount of previous payment in May, 2005 (NT dollar)

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

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

3**. Tools Used**

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

4. Design Details

4.1. Process Flow

Data Pre processing

Data Exploration

Importing Dataset

Tuning Model

Prediction

Selecting Model

Train test Split

4.2. Deployment Process

5. Conclusion