Low Level Design (LLD)

Credit risk assessment using South German credit data

December 31, 2021

Abstract

Normally, most of the bank's wealth is obtained from providing credit loans so that a marketing bank must be able to reduce the risk of non-performing credit loans. The risk of providing loans can be minimized by studying patterns from existing lending data.
Hence creating a solution, which will utilize past experience to assist credit officers, can lead to less risk exposure of banks.

Revision number: 1.0

Last date of revision: December 31, 2021

Contents

Chapter 1

Introduction

1.1 Purpose of LLD

Return to table of contents

This document is aimed to provide a detailed description of credit risk assessment solution. It will explain the aim and features of the solution, interfaces of the solution, what the solution will do, and constraints under which it must operate. This document is for both stakeholders and developers of the solution.

The goal of the project is to get credit default probability and assist a credit officer to make a decision.

1.2 Scope

Return to table of contents

The solution will be a web application. It will take an applicant's characteristics as input and return default probability in line with a possible decision to make.

1.3 Constraints

Return to table of contents

The dataset contains only 1000 observations.

Chapter 2

Technical specifications

2.1 Dataset

Return to table of contents

The data contains 1000 observations for South German bank credit market. Specifically 700 good and 300 bad credits with 20 predictor variables (features). It covers a period from 1973 to 1975. The data is hosted here.

2.2 Assessing credit risk

Return to table of contents

1. The system displays input required.
2. User provides the input values.
3. The system assesses credit risk and displays it.

2.3 Logging

Return to table of contents

The logging is applied in model building process almost after each step, but logging results will not be accessible for end users (credit officers). Moreover logging is made accessible via console and a file for developers. Logging is mandatory as it allows to find debug issues more easily.

2.4 Database

Return to table of contents

The database is hosted on Astra DB. The system does not store any request into the database.

2.5 Deployment

Return to table of contents

2.6 Technology stack

Return to table of contents

Front end	HTML/CSS
Back end	Flask
Database	Cassandra
Deployment	Heroku

Chapter 3

Proposed solution

3.1 Description

Return to table of contents

As there are only 1000 observations in the dataset, it will be reasonable to use simple models. Otherwise we can encounter overfitting. So we will use 3 simple models:

1. logistic regression where all categorical features are onehot-encoded
2. logistic regression which takes as inputs only categorical features transformed using the concept of weight of evidence
3. simple decision tree.

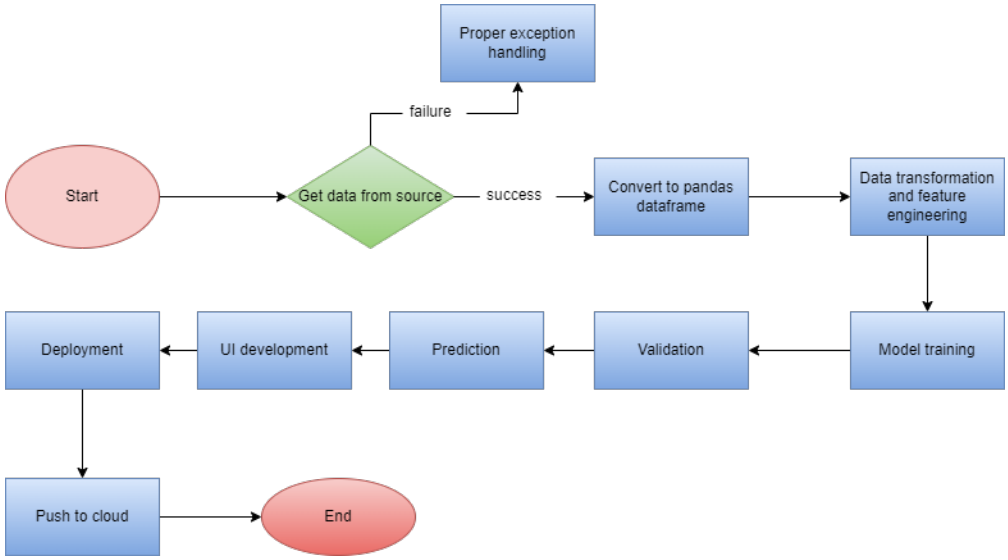
After estimating all the models we average their outputs to get final output (probability of default).

Chapter 4

Model training/validation workflow

4.1 Diagram

Return to table of contents

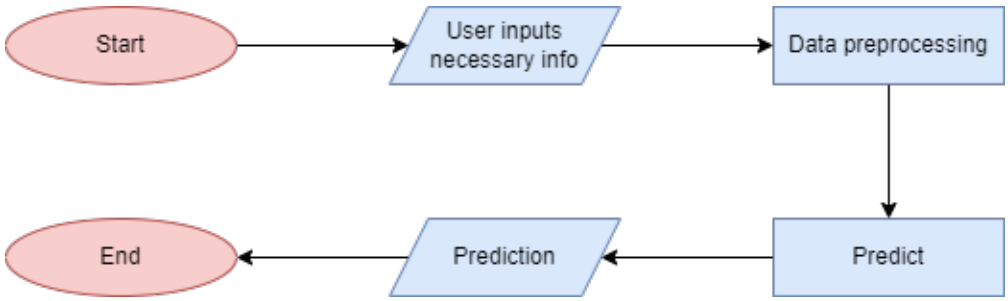


Chapter 5

User I/O workflow

5.1 Diagram

Return to table of contents



Chapter 6

Test cases

6.1 Description

Return to table of contents

Testing has been conducted only for a function and a class related to weight of evidence transformation.