HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

MSC BIG DATA TECHNOLOGY

5003 - BIG DATA COMPUTING

Credit Default Risk - Project Proposal

Group 30

Hui Ho Yin, Jeffrey (Student ID: 20745101)

Wong Tsz Ho (Student ID: 20725187)

Chiu King Yuen, Anthony (Student ID: 20737245)

Wong Chin Hang, Eric (Student ID: 20123808)

18 October 2020

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Problem Definition

A loan is developed when a party lends money to another party, it can be further classified into secured or unsecured loan, depend on whether the borrowers have any collaterals when the loan is borrowed. In this project, we will focus on loans provided by a financial institution.

When a borrower lends, he or she is required to repay the debt by installment or full payment according to the repayment schedule. However, the borrower may not make a payment or makes partial payments for the loan at a repayment day. Excessive delay in repayments maybe classified as credit default by the issuer. The issuer will then incur a loss when the borrower defaults.

Credit default risk is the risk to the issuer that the borrower cannot repay the loan on their debt obligation. To mitigate the impact of default risk, the issuer often employs statistical models to explore the repayment habits of its customers. In practice, the issuer usually classifies customers into groups and then predict each group's chances of default and chances of late payment.

Expected outcome

With the rise of machine learning, loan issuers find a new way to improve their business model to achieve better risk control. Our target in this project is to build a machine learning model that predicts the chances of default and chances of late repayment for borrower given a set of information about the borrower, such as gender, monthly balance, repayment history for the previously disbursed loans. We will be exploring the dataset, cleaning the data, analysing the data, modelling our predictions, and visualizing our result in a reader-friendly way.

From a modelling point of view, we will be focusing on the following relationships:

- Finding out clusters between low and high repayment rate borrowers
- Predicting the probabilities of not able to repay on schedule base on features input
- Finding the correlation of features with respect to repayment rate
- Finding the correlation between features

We expect to conclude the relationship between features and chances of defaults or late repayment, and thus helping financial institutions to decide the interest rate when issuing loans to borrowers.

Dataset description

From Kaggle, we have found multiple datasets related to credit default risk. The dataset we have chosen to use in this project is from a relational data base, which contains 7 tables which can be further categorized into credit records in other financial institutions, or credit records in the targeted financial institution. There are more than 350,000 loan records and more than 300 features, for example, default statue, gender, car ownership, number of children, outstanding balance, receivables etc… There are various types of data including but not limited to: continuous data, discrete data, ordinal data, binomial data.

We will be using all of the relational table, but not all columns within. CSV files can be downloaded from this URL: Kaggle Dataset. The total size of the dataset is about 2.5 GB.

The dataset descriptions are provided in Table 1 below. As there are more than 350,000 rows and 300 features, there are missing data that would require data preprocessing and data cleansing. In addition to data preprocessing and cleansing, we would also apply other feature engineering methodology whenever necessary in encountering outliers or noises.

Table 1. Dataset and description.

Dataset	Description	Size
HomeCredit_columns_description.csv	• A summary of other csv file	36KB
application_train.csv application_test.csv	• Main table • 308,000 borrowers with 121 columns of features	183MB
bureau.csv bureau_balance.csv	• Credit record in Credit Bureau • 308,000 borrowers with 18 columns of features	520MB
previous_application.csv	• Application record of client in Home Credit • 1,670,000 loan entries with 36 columns of features	386MB
POS_CASH_balance.csv	• Snapshot of monthly balance • 10,000,000 monthly balance entries with 7 columns of features	374MB
installments_payments.csv	• Repayment history for the previously disbursed credits in Home Credit • 13,600,000 payment record entries with 7 columns of features	689MB
credit_card_balance.csv	• Snapshot of monthly balance • 3,840,000 monthly balance entries with 22 columns of features	404MB

Technology

1) Data storage with Apache HBase or Hive

As the system is an analytical system. The loadings are mainly read operations from feature engineering, model training and analytical reporting. HBase is the one of the ideal data storages that supports fast read operations. Also, it is built on top of Hadoop and with Apache HBase Connector it supports querying HBase via Spark SQL and the DataFrame APIs.

2) Task Management with Apache Airflow

Airflow comes with a web UI for monitoring tasks execution status, execution histories and error traces. With this we can trigger/schedule different tasks (feature engineering, model training etc.) with the web UI. It triggers the actual computations on the Spark clusters with Airflow Spark operator.

3) Data streaming with Apache Kafka + Spark Streaming

Kafka can be used for event streaming to handle real time data updates to update the analytical database. It can be used for generating predictions for newly arrived data and notifying the predictions to other applications.

Spark Streaming then reads data from Kafka and put newly arrived data into batches for bulk database insert and model prediction.

4) Data Processing and Feature Engineering with Apache Spark SQL

Aggregations are slow to compute, there are many aggregations features on the loan applicant level in this dataset. Computing these features with Spark cluster can be faster.

5) Machine Learning with Apache Spark MLlib

Tree based models are usually useful for relational dataset with both numerical and categorical data. There is a GradientBoostedTreesModel in MLlib. Neural network based solution will also be explored.

6) Deployment with K8s (Azure AKS), Docker, Microservices

To enhance the performance of the system and to provide more agile development, different functions will be divides into different small Microservices. Different Microservices can use different programming language which allow us to make use of our familiar programming language during development. This speed up the whole development process. Different Microservices then will be packed into standardized units for development, shipment and deployment. We might use Docker as our container technology enabler. To further manage different docker containers, container orchestration tool may come handy for automating deployment, scaling, and management of containerized applications. Kubernetes is one of the open source container orchestration platforms. We might build our own Kubernetes from scratch or we might adopt AKS (Azure Kubernetes Service) for better experience with integrating different cloud services.

7) Collaborative Development with Gitpod + CI/CD

Teamwork is necessary nowadays. Gitpod provide us a web-based development tools which integrate GitHub repository and enable collaborative coding. Code is then be pushed on GitHub and we can make use of DevOps tools like GitHub Action to automate software workflow. We might manage to use Azure Pipelines to gain the better experience on managing our development and deployment process with azure services.

8) Data visualization with Plotly Dash

Visualization is a key part for data scientist to interact with the audience. We will launch web application because web app is cross-platform and universal. Users only require modern browser to run the application. There are a lot of different web app framework in the market, React, Vue etc. We choose Dash from Plotly as our web app framework. Dash is a data visualization specific framework which we can visualize data in only few lines of code. We might design a user interface for user to enter the attributes to predict the corresponding values bases on our model. We would pick tensorflow.js to interface our model.