HOME CREDIT DEFAULT RISK

Abstract:

Many people struggle to get loans due to insufficient or non-existent credit histories. Unfortunately, this population is often taken advantage of by untrustworthy lenders. Home credit strives to broaden financial intrusion for the unbanked population by providing a positive and safe borrowing experience. In order to make sure this underserved population has a positive loan experience, Home Credit makes use of a variety of alternative data--including telco and transactional information--to predict their clients' repayment abilities. There are multiple datasets that will be taken from Kaggle that are mentioned in the below section. We perform exploratory data analysis and feature engineering on the given datasets and train the data using multiple classification algorithms. Based on some specific metrics, we decide which model is the best among all. The output of the model will be either 0 or 1 where 0 corresponds to saying that the customer will repay the loan and 1 meaning that there's some risk to the customer repaying to the lender.

Data Description:

Application_train/application_test: This dataset contains information about each loan application at Home Credit. Each loan is depicted by a row and is identified by the feature SK_ID_CURR. The training application comes with target indicating 0 which means the loan was repaid and 1 the loan was not repaid.
Bureau: Data concerning client's previous credits from other financial institutions. Each previous credit has its own row in bureau, but one loan can have multiple previous credits.
Bureau_balance: Monthly data about the previous credits in bureau. Each row is one month of a previous credit, and a single previous credit can have multiple rows, one for each month of the credit length
Previous_Application: Previous applications for loans at Home Credit of clients who have loans in the application data. Each current loan in the application data can have multiple previous loans. Each previous application has one row and is identified by the feature SK_ID_PREV.
POS_CASH_BALANCE: Monthly data about previous point of sale or cash loans client have had with Home Credit. Each row is one month of a previous point of sale or cash loan, and a single previous loan can have many rows.
Credit_Card_Balance: Monthly data about previous credit cards clients have had with Home Credit. Each row is one month of a credit card balance, and a single credit card can have many rows.
Installments_payment: Payment history for previous loans at Home Credit. There is one row for every made payment and one row for every missed payment.

Machine Learning Algorithm:

We plan on implementing Logistic Regression, Decision Trees, Random Forest, Support Vector Machines, K- Nearest Neighbors, Neural Networks (Recurrent Neural Networks and Multilayer perceptron) as our model classifiers.
The output of the models would be either "Risk Free" and "Defaulter". The output of the classification model would be 0 or 1 where 0 corresponds to "Risk Free" and 1 corresponds to "Defaulter". Logistic Regression is a process of modeling the probability of a discrete outcome given an input variable.

Decision Trees are non-parametric supervised learning method used for classification and regression and a combination of decision trees is a Random Forest. Support Vector Machines is a linear model for classification and regression problems. It can solve linear and non-linear problems and work well for many practical problems. The idea of SVM is simple: The algorithm creates a line or a hyperplane which separates the data into classes. KNN is a supervised machine learning algorithm. The algorithm can be used to solve both classification and regression problem statements. The number of nearest neighbors to a new unknown variable that must be predicted or classified is denoted by the symbol 'K'. A recurrent neural network is a class of artificial neural networks where connections between nodes form a directed or undirected graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior.
A Multilayer Perceptron has input and output layers, and one or more hidden layers with many neurons stacked together. And while in the Perceptron the neuron must have an activation function that imposes a threshold, like ReLU or sigmoid, neurons in a Multilayer Perceptron can use any arbitrary activation function.

Metrics used:

The metrics used for measuring the accuracy and performance of our models are ROC, F1 Score, MSE, KS Score, RMSE.
F1 score: The F1-score combines the precision and recall of a classifier into a single metric by taking their harmonic mean. It is primarily used to compare the performance of two classifiers. The equation is as below: 2 * (Precision * Recall)/ (Precision + Recall)

MSE: Mean Square Error measures the amount of error in statistical models. The equation is as below: $1/n \sum_{i=1}^{n} (Yi - \check{Y}\imath)$ ^2

RMSE: It is a frequently used measure of the difference between values predicted by a model, or an estimator and the value observed. $\sqrt{\sum_{i=1}^{N} (xi - \check{x}\imath)^2 / N}$

KS Score: The Kolmogorov- Smirnov statistics quantifies a distance between the empirical distribution function of the sample and the cumulative distribution function of the reference distribution, or between the empirical distribution functions of two samples.

ROC AUC: ROC curve is a graph showing the performance of a classification model at all classification thresholds.

Block diagram (Gantt Diagram):

Assignments **April 1** **April 5** **April 12** **April 19** **April 26**

Phase 0

- Choose project and create abstract
- Decide on data and ML algorithms
- Choose metrics and data pipelines
- Discuss and divide project task among team

Phase 1

- Perform EDA on data and metrics creation
- Build the baseline pipeline and perform required analysis
- Create report with proper structure, style, and content
- Describe decision making process and analysis

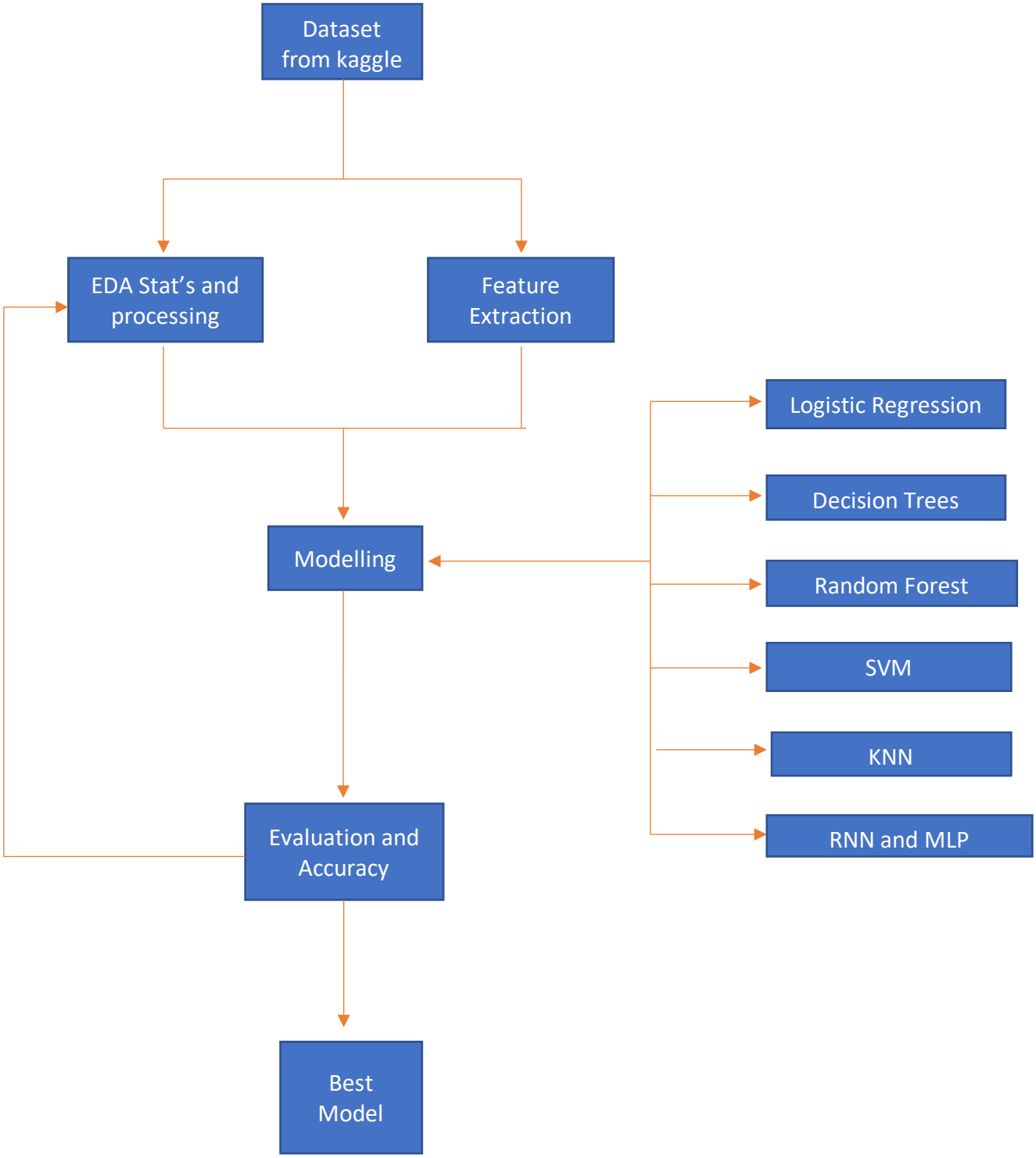
Phase 2

- Work on feature engineering and hyperparameter tuning
- Feature selection and analysis of feature importance
- Use multiple models to choose best accuracy one
- Start with building MLP model in Pytorch

Phase 3

- Add deep learning model to the phase 2
- Build MLP in PyTorch, using tensorboard to visual
- Build PyTorch model for classification and regression
- Build Multi-headed load default system and Using OOP API in PyTorch with CXE + MSE
- Prepare for final submission of the project

Pipeline Steps:



The first step would be gathering datasets from Kaggle. Then, we do exploratory analysis and perform feature engineering. After this, we use this data and train using multiple classification models like Logistic Regression, Decision Trees, Random Forest, Support Vector Machines, K-Nearest Neighbors, and Neural Networks. We check the output of each model using evaluation metrics like F1-score, MSE, RMSE, KS Score, ROC AUC and decide on which algorithm, we're getting the best accuracy. It is important we do this carefully as we need to make sure the worthy people are sanctioned loans and unworthy ones are rejected.

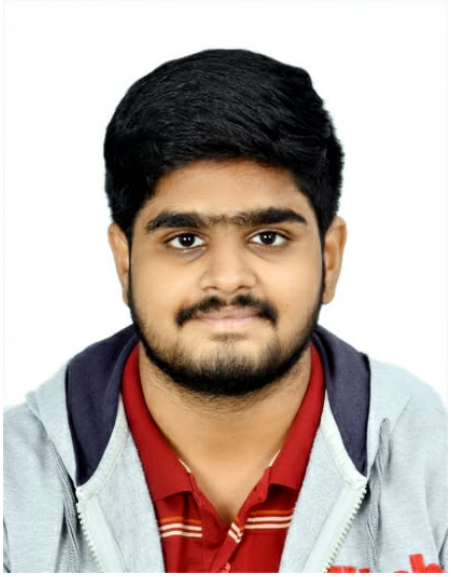
Team members details:

Yashvanth Guntupalli (yakguntu@iu.edu)

Srilekha Malraju (msrilekh@iu.edu)

Revanth Sai Chowdary Rayala (rerayala@iu.edu)

Vinay Chandra Makineni (vimakin@iu.edu)



Project members contribution:

Revanth will be working on exploratory data analysis and statistics and removing inconsistent samples. Visualization is used to analyze the data to figure out the balance maintained, other loans taken, installments and EMI and other features.

Vinay will be working on creating metrics ROC, F1 Score, MSE, KS Score, RMSE. He is also responsible for decision making trees and random forests to decide feature importance. Responsible for implementation of K-nearest neighbours algorithm for classification of features.

Yashwanth will be doing SVM model. Will also be responsible for feature engineering and hyperparameter tuning, and analysis of features. Build data pipeline and choose the best model using metrics and decide the best scored model.

Srilekha will be responsible for building MLP and RNN model in PyTorch for classification. Application of tensorboard to visual training data. Responsible to work on logistic regression model.
Team is responsible to contribute in others work also if required and cross verify each other's work.