**Project Proposal**

**Home Credit Default Risk**

**FP\_GroupN\_11**

Kumud Sharma([kumsharm@iu.edu](mailto:kumsharm@iu.edu))

Kamna Chaudhary ([kamchau@iu.edu](mailto:kamchau@iu.edu))

Bhavya Mistry([brmistry@iu.edu](mailto:brmistry@iu.edu))

Jaydeep Patel([jp157@iu.edu](mailto:jp157@iu.edu))



**Abstract**

In this project, we're aiming to forecast whether a particular client would pay back the loan they obtained. The "application train.csv" dataset will be the main one we use, although other information from related subsets of datasets will also be helpful. We want to perform extensive feature engineering and data preparation, and we might decide to lower the application dataset's current 121 feature count. We're going to release a couple various models so we can evaluate them. These models include logistic regression with lasso, ridge, and no regularization, support vector machines with a linear kernel and radial, decision trees or random forests, and neural networks with PyTorch with various network topologies. We may also employ K-nearest neighbors if we can significantly reduce the number of features. The F1 score measure will be applied to adjust the models' hyperparameters. To determine the model that performs the best, we will then compare the F1 scores of the tweaked models.

**Data Description**

The primary training segment of the data is organized in a CSV file named "application train.csv," and the corresponding test data are in "application test.csv." Only the target column for whether the client repaid the loan separates the training data set's columns from those in the test set. The application dataset has 121 feature columns that comprise both categorical variables, such gender, and numerical features, like income.

A number of additional datasets build on one another and correlate with the application dataset. Any credit history row for the customer prior to the application date that corresponds to a loan in the application dataset is present in the "bureau.csv" dataset. Data for the prior credits stated in the bureau dataset are included in the "bureau balance.csv" dataset for each month in history. These supporting datasets interact with one another and the application dataset in this manner. Categorical values as well as positive and negative numerical values are present in these auxiliary datasets. There are a total of six subsidiary datasets, but since the test set is dependent on the application set, that is the one we will be focusing on.

The graphic below illustrates the connections between the various datasets offered in the issue:

**1. application\_{train|test}.csv**

This is the primary dataset, which consists of training and test samples. This set uses the SK ID CURR property to connect to other sets and contains information on the loan applicant's personal characteristics.

**2. previous \_application.csv**

This dataset contains details regarding prior Home Credit loans made by the applicant. It includes characteristics including loan status, down payment, and loan type.

**3. instalments\_payments.csv**

Information on loan repayment from the past is included in this dataset.

**4. credit\_card\_balance.csv**

Information on Home Credit credit card transactions is included in this dataset.

**5. POS\_CASH\_balance.csv**

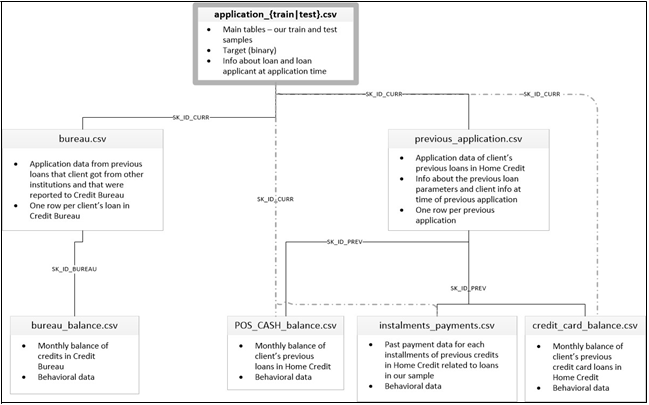
The items in this dataset describe the individual's past credit history at Home Credit, which includes personal loans and consumer credit.

**6. bureau.csv**

This dataset contains details about a person's prior credit history at other financial institutions that were reported to a credit agency.

**7. Previous\_application.csv**

This dataset includes the credit bureau's monthly credit balance.



**Machine Learning Algorithms and Metrics**

**Models**:

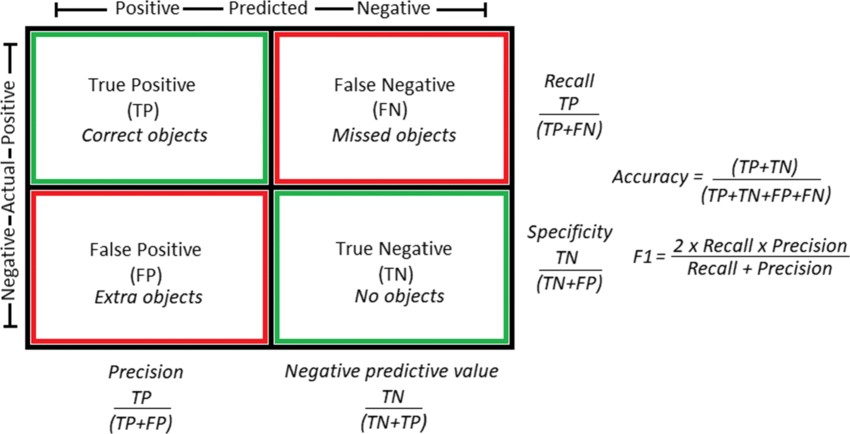
We are aiming to anticipate the client's ability to repay the loan, as explained in the abstract and data description module, and this ability is decided by a single binary output variable called TARGET (0|1). Under supervised learning, this is a binary classification issue because the goal has only two discrete possibilities that could occur. We will compare various categorization models using the chosen metrics to find the model that best fits the situation.

We will use Logistic Regression (with Lasso, Ridge, and No Regularization) as our basic model because this is a binary classification problem. Based on the Logistic Regression model, we will evaluate the effectiveness of different models. The likelihood of various classes or clusters occurring in the dataset will be predicted using a Naive Bayes classifier. We will use the Decision Tree Classifier model by developing rules based on the application train.csv dataset, notably on the income and credit columns such as amt income total, amt credit, etc. We will also investigate Support Vector Machines (SVM), Random Forest Classifier, K-Nearest Neighbors, and XGBoost Classifier. Later on, we'll concentrate on dimensionality reduction and hyperparameter tweaking, setting the stage for a thrilling algorithmic race.

**Metrics**:

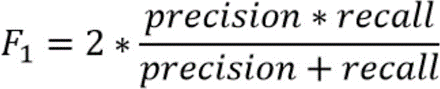
1. **Confusion Matrix:**

A confusion matrix is used to evaluate the effectiveness of a classification model, and the resulting matrix reveals the degree of classification accuracy and potential prediction mistakes for each record. It is used to gauge AUC-ROC curve, recall, accuracy, and precision.



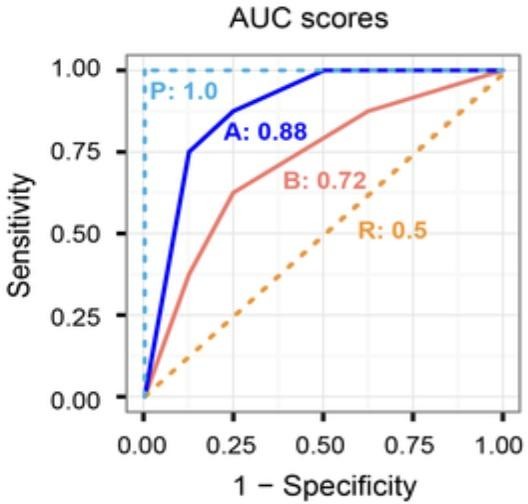
**2. F-1 Score:**

It will be used to compare the effectiveness of two classifiers based on their precision and recall values. It will be crucial in deciding which model fits the issue more effectively.

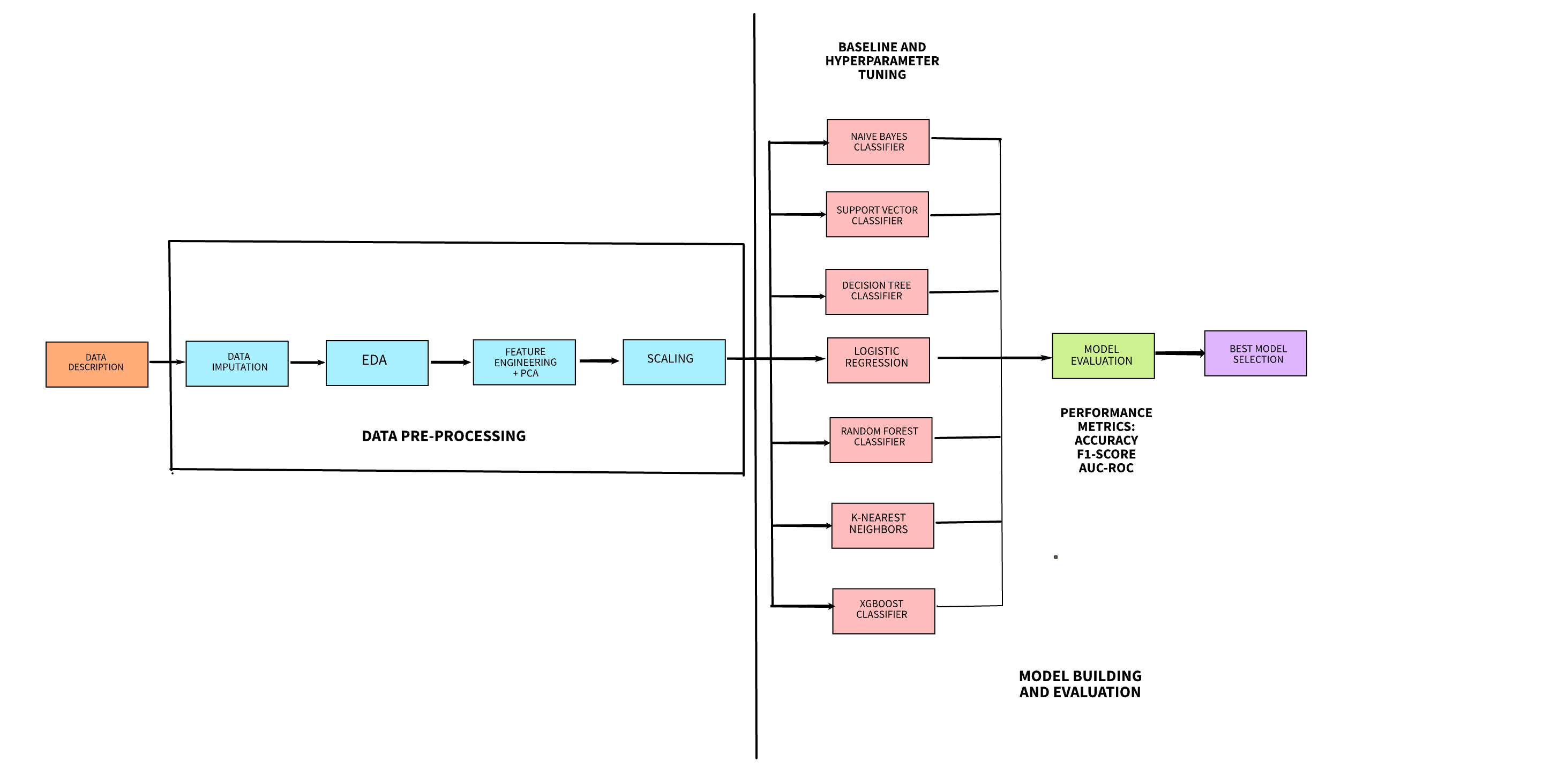


**3.AUC-ROC Curve:**

When the actual outcome is positive, the ROC curve will assist us in calculating the likelihood of correctly predicting the positive class. Its performance is distilled into a single value using the AUC curve.



**Machine Learning Pipeline**

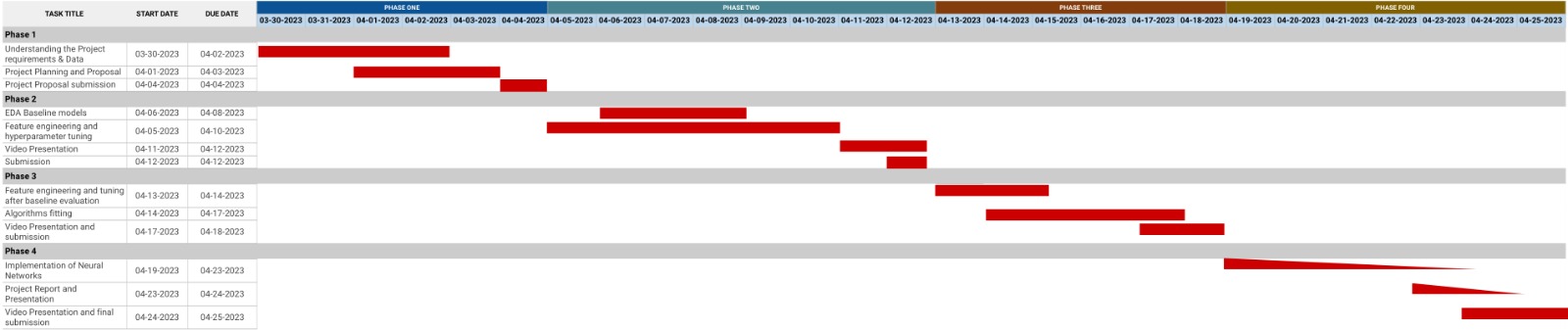


The dataset is briefly described at the outset of our project. It is crucial to gain an understanding of how the dataset is structured because we are working with numerous files here. We will identify the numerous problems with the current data, such as missing values, duplicate values, and inconsistent data types, and address each problem individually. We will use a straightforward imputer to substitute the missing values for the continuous variables' median values and the categorical features' modal values. An exploratory data analysis of the application train and application test data will then come after this. The features that are crucial for our goal feature can be identified based on the conclusions gained.

As each dataset contains more than 120 columns, we may apply principal component analysis to reduce the features that are less important based on their variance. All of the aforementioned machine learning models will then undergo baseline training and evaluation, which will allow us to assess how well each model performs in relation to the aforementioned performance indicators. Due to the dataset's extreme imbalance and the fact that accuracy cannot be trusted with such datasets, F1-score and ROC-AUC will be given higher priority. As ensemble models can choose a subset of features during training, they can perform better in this scenario than models like XGBoost and Random Forest. In order to prevent data loss, a sklearn pipeline will also be used. By running a successful race and evaluating which models perform best, we will choose the best ones and fine-tune them. A second model made out of a PyTorch deep neural network will also be put into practice and compared to our improved models. We will choose our best model after comparing the results.

**Timeline**

The proposed timeline for the Home Credit Default Risk Project:



**Phase Leader Plan**

|  |  |  |
| --- | --- | --- |
| **Final Project Phase** | **Phase Leader** | **Phase Plan** |
| Phase 0 | Team | Team Formation |
| Phase 1: Project Proposal | Kumud Sharma | There will be a project proposal created outlining all of the project's components. Each team member is given a task to do. The dataset is investigated, and potential machine learning techniques that could be used in the project are described. A base pipeline for the project is chosen, and appropriate evaluation metrics are determined. |
| Phase 2: EDA + Baseline | Bhavya Mistry | With the chosen machine learning methods, this phase focuses on baseline model training and evaluation as well as exploratory data analysis, data imputation, and baseline model evaluation. The EDA and baseline performances of several models will be used to derive conclusions, and then judgments will be taken regarding hyperparameter tuning and feature selection. |
| Phase 3: Feature Engineering +  Hyperparameter Tuning | Kamna Chaudhary | Here, we'll concentrate on the feature of choosing the most suitable features based on several strategies, including correlation, developing new features, and dimensionality reduction, as well as our comprehension of the data from earlier phases. To help determine the ideal set of parameters for each machine learning model, we will also begin experimenting with the different parameters of the models. This will pave the stage for an exciting race between the various algorithms. |
| Phase 4: Final Submission | Jaydeep Patel | Deep neural networks will be used in this step, and we will compare their performance to the existing fine-tuned models from our previous phase. We will choose the best model and submit it in our final submission based on our numerous performance indicators. |

**Credit Assignment Plan(Phase 1):**

|  |  |
| --- | --- |
| **Team Member** | **Task** |
| Kumud Sharma | EDA and Data Preprocessing |
| Kamna Chaudhary | Feature Engineering and Baseline Modeling |
| Bhavya Mistry | Feature Selection, Dimensionality Reduction, Model Evaluation |
| Jaydeep Patel | Hyperparameter Tuning, Best model selection |