Peer to Peer Loan Default Prediction

This project involves building a machine learning model to predict loan defaults using the Prosper P2P lending dataset. The goal is to explore the data, engineer relevant features, and build a robust classification model that can distinguish between loans that will be fully paid and those that will default.

The analysis follows a real-world validation process where the model is trained on an older set of loans and then tested on a more recent, unseen set of loans.

Project Objective

The primary objectives of this project are:

To perform a thorough exploratory data analysis (EDA) on the Prosper loan dataset.

To clean and prepare the data for modeling, including feature engineering.

To build a binary classification model to predict the likelihood of a loan default.

To evaluate the model's performance on both in-sample (training) and out-of-sample (testing) data.

To identify the most important variables that influence loan default.

Dataset

The data used for this project is the Prosper Loan Dataset. For this analysis, the scope is limited to 3-year loans only. The dataset is split into two periods:

In-Sample Data: An earlier period used for training and building the model.

Out-of-Sample Data: A more recent period used for testing the final model's performance.

Models & Algorithms

This project can be implemented using a variety of classification algorithms. The primary model is chosen from the following list, implemented using Python's scikit-learn and statsmodels libraries:

Logistic Regression

Decision Tree (CART)

Random Forests

Gradient Tree Boosting

Support Vector Machine (SVM)

Neural Networks

k-Nearest Neighbors (k-NN)

Project Workflow

Data Cleaning: The initial dataset is cleaned and prepared. This includes handling missing values, correcting data types, and filtering for the required 3-year loan term.

Feature Engineering: New features are created from existing variables to better capture information that might predict default.

Model Training: A chosen classification model is trained on the in-sample data.

In-Sample Evaluation: The model's performance is assessed on the same data it was trained on to establish a baseline.

Out-of-Sample Testing: The final, trained model is used to make predictions on the new, unseen data. This is the true test of the model's predictive power and its ability to generalize.

Analysis & Interpretation: The results are analyzed to determine the model's effectiveness and to identify the key variables that are most predictive of default.

How to Run

To run this analysis, you will need a Python environment with standard data science libraries installed.

- **Prerequisites:**

Python 3.x

Jupyter Notebook or JupyterLab

Libraries: pandas, numpy, scikit-learn, statsmodels, matplotlib, seaborn

- **Installation:**

pip install pandas numpy scikit-learn statsmodels matplotlib seaborn

- **Execution:**

Place the Prosper dataset file in the project's data directory.

Launch Jupyter and open the .ipynb notebook file.

Run the cells sequentially from top to bottom to execute the entire data analysis and modeling pipeline.