Analyzing Loan Application Trends in New York State: Machine Learning Insights into Who Can Get a Loan

Yihao(Rainer) Luo, Yiqing (Selina) Xin, Yiwen (Wendy) Sun

ORIE 5741 Project Team Sweet Potatoes

05/09/2023

Cornell Engineering

Outline

1. Project Background & Problem
2. Dataset Introduction
3. Data Cleaning Process
4. Three Machine Learning Models
5. Summary & Next Steps

Background & Problem

- General Background:
  - financial institutions face credit risk: possibility of loss if borrowers fail to repay loans or meet contractual obligations.
    $\longrightarrow$ problems: interruption of cash flow & increase cost of loan collection
  - normally, bank require applicants' information to decide whether to lend a loan.

- Project Goal / Decision Problem:
  - identify applicant characteristics favored by lenders and that have higher probability of application acceptance to help applicants to determine whether the application will be accepted.

Cornell Engineering

Dataset & Variables

- Dataset intro:
    - FFIEC Home Mortgage Disclosure Act (HMDA)
    - state: New York, year: 2021
    - 787,436 observations
    - 99 variables of information about borrowers, demographics, loan details, etc.

99 variables $\xrightarrow{\text{select essential variables}}$

- 1 response variable: denial_reason_1
- 23 predictive variables → 6 categories:
    1. demographics (e.g. race, sex)
    2. borrower basics (e.g. income)
    3. property nature (e.g. house values)
    4. the purpose of the loan (loan reasons)
    5. contract details (e.g. interest rates)
    6. others (e.g. bank id)

Data Cleaning

- Feature Engineering:
  - response variable → 1: approved; -1: denied
  - categorical predictive variables → one-hot encoding

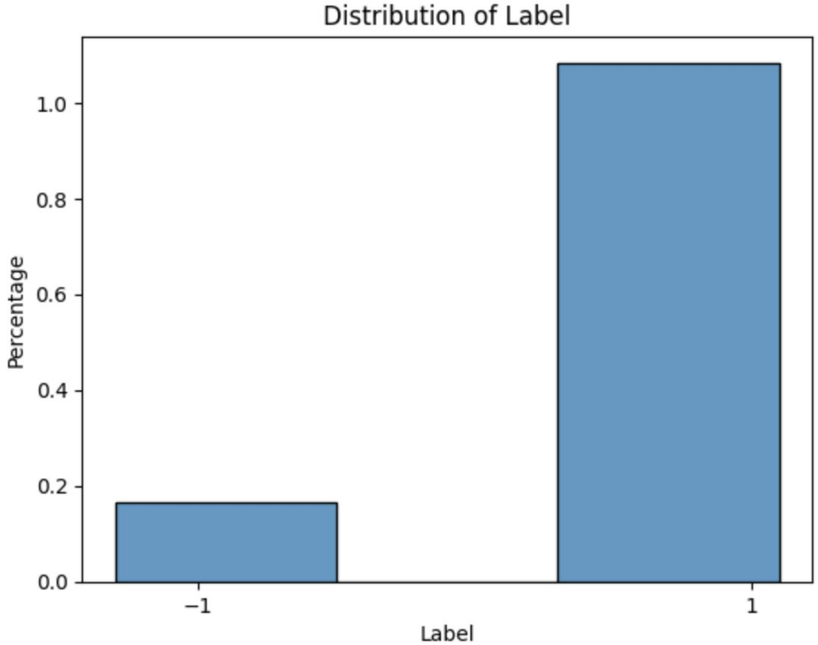
- Missing Values for Numerical Predictive Variables
  - few missing observations → remove rows with missing observations
  - many missing observations → use average to fill missing values
  - Missing values for certain variables occur when the corresponding applications are not accepted, meaning that information is only collected for accepted applications. (e.g. loan term, interest rate)

↓

772458 observations, 178 predictors

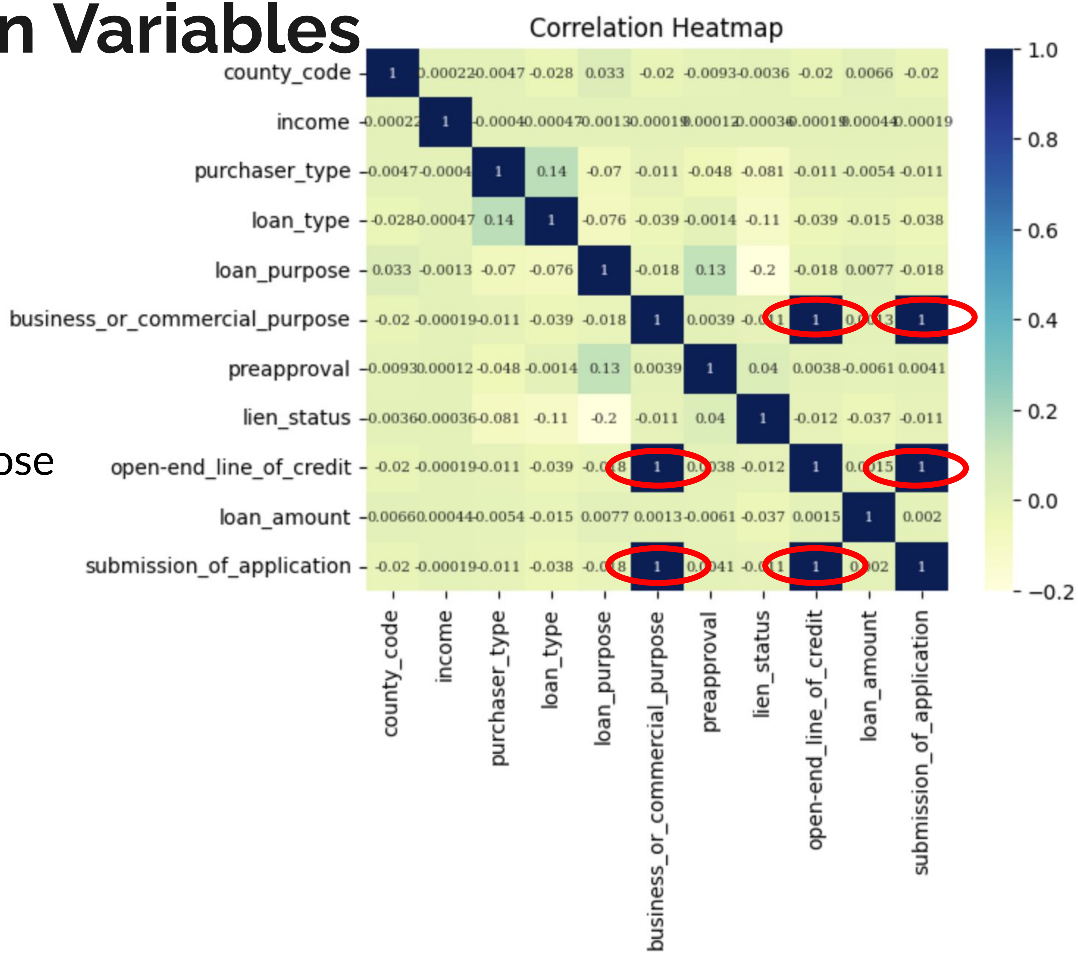
Exploratory Data Analysis

- Percentage of acceptance and rejection
  - acceptance 87%
  - rejection 13%



Correlation Between Variables

- remove variables that have strong correlations
  - open-end line of credit
  - submission of application
  - business_or_commercial_purpose



Machine Learning Models

1. Logistic Regression
2. Decision Tree
3. Random Forest Tree

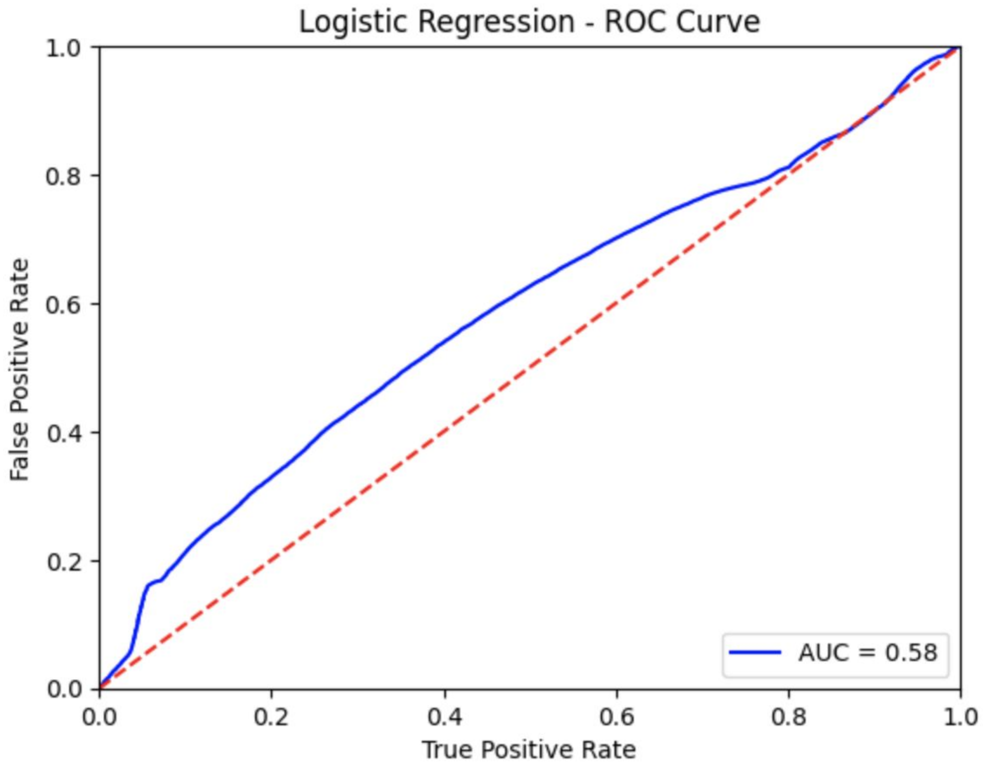
- Train/Test split → train: 80%, test: 20%

Logistic Regression Method

- Model probability of certain events or outcomes by using sigmoid function to map real-valued numbers to a value between 0 and 1
- Pros:
    - classification problem (response = -1 or 1)
    - easy to implement and interpret; model will output a probability estimate for each observation
- Cons:
    - Assumes linear relationship between predictor and log odds of variables, but may not always be true
    - may not perform well if the dataset is very imbalance

Logistic Regression

- Classification Model
- Results
  - precision = 0.83
  - recall = 0.99
  - AUC = 0.58
- Low AUC → poor ability to distinguish between positive and negative classes



Tree Based Algorithms

- Robust to any kind of Inputs (i.e. Numerical / Categorical) <- Important when client information comes in different forms

- Fast in Making predictions, making it suitable for online learning (i.e. Loan Pre-approval) within the Financial Industry

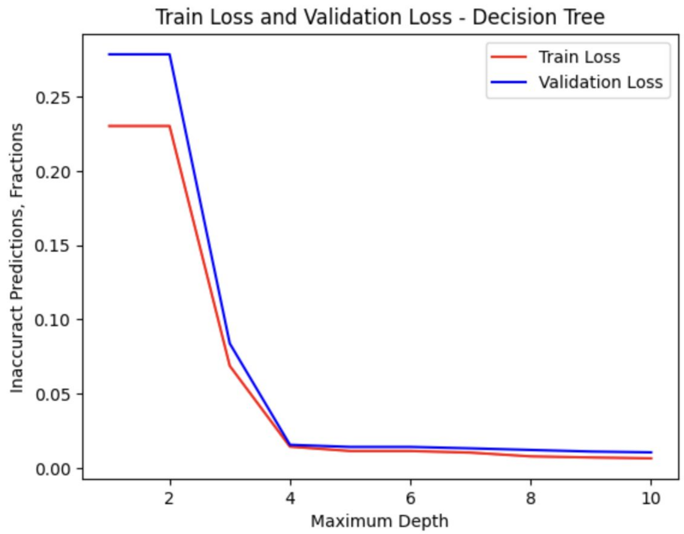
- Ability to detect feature importance, so Data Scientists could understand the causality.

- Boosting and Bagging will further empower the model

Decision Tree

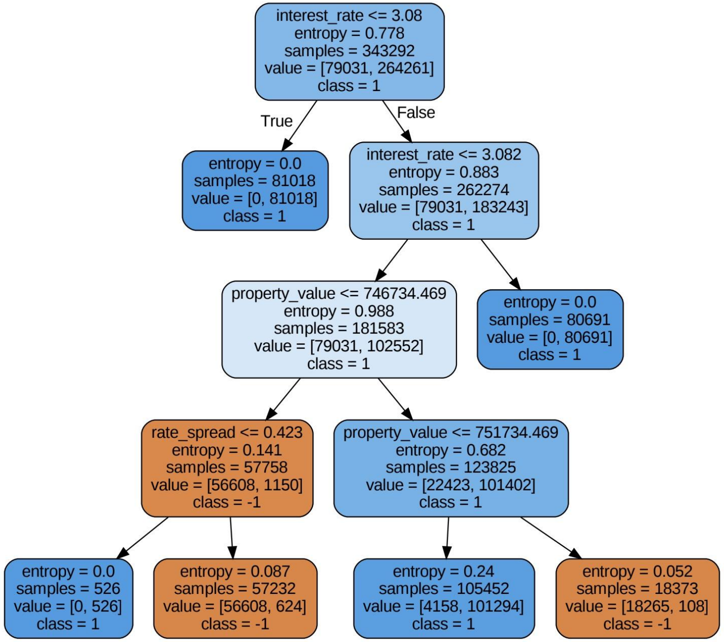
- Tree based algorithm that seek to minimize impurity (Entropy) at each node

- Tree Depth is selected basing on validation loss



Decision Tree

- Tree Visualization is exhibited at the RHS
- A small tree (max depth = 4) ensures a 98% validation accuracy
- Conclusion: Only a few features are important (by Mean Decrease Impurity)
    - Property_value (61.4%)
    - Interest_rate (37.2%)
    - Rate_spread (1.4%)

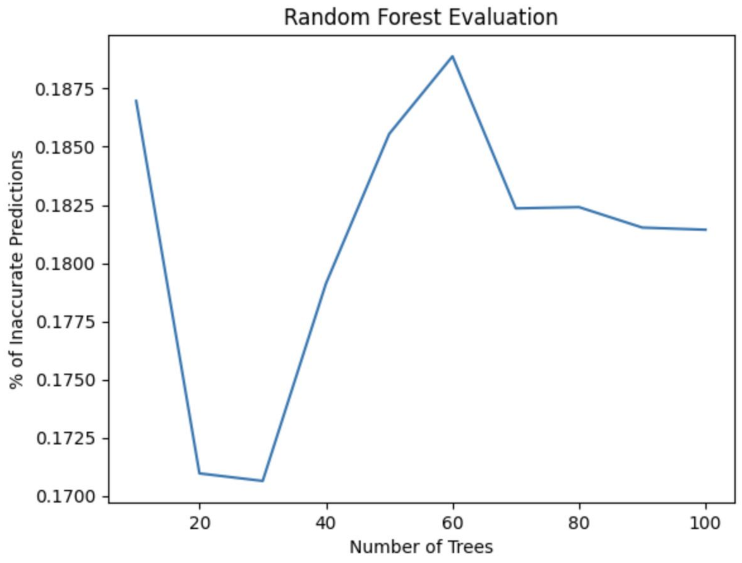


Random Forest

- We have noticed each individual tree is a strong learner. So we fit a Random Forest to:
    - Increase Generalizability
    - Identify other Important Features

- Number of Trees is selected on training loss, as it is an unbiased estimation of the Validation loss.

- Boosting algorithms are not considered



Random Forest

- Random Forest (30 Trees) ‘forces’ the model to pay attention to other features.
- As a result, accuracy decreases to 82%
- We get interpretable results on other deterministic factors of loan approval

```
Score: 0.8208755228784824
```

	feature	importance
3	interest_rate	0.224368
23	debt_to_income_ratio->60%	0.181814
4	rate_spread	0.180363
0	property_value	0.102915
1	loan_to_value_ratio	0.062061
7	debt_to_income_ratio-50%-60%	0.047234
6	income	0.043117
140	open-end_line_of_credit-1	0.016489
141	open-end_line_of_credit-2	0.016173
54	applicant_age-8888	0.011249

Summary & Next Step

- **Logistic Regression**
  - high precision and recall
  - but low AUC (0.58) suggests the model performs no better than random chance
- **Decision Tree & Random Forest**
  - Decision Tree with max depth of 4 achieves high accuracy (98%)
  - Random Forest with 30 trees improves the model's attention to other features with accuracy of 82% , which provides insight into other factors of loan approval.
- **Next Step**
  - Deploy the random forest model on a web platform or integrate into an existing loan application system to provide instant feedback to applicants on their application's acceptance probability.

Thank You