Home Loans Default Prediction Milestone Part One

Xinyu Zhao

07/18/2022

Problem Definition

- **The context:**
- Non-Performing Loan (NPA) eats up a significant chunk of a bank's profits. We need to build a machine model that can automatically check a customer's creditworthiness with high efficiency and low bias.
- **The objectives:**
- We aim to simplify the decision-making process for home equity lines of credit. Our model predicts risky clients and recommends features to consider when approving a loan.
- **The problem formulation:**
- We will build an empirically derived classification model. The model will be based on the existing loan underwriting data, using predictive modeling techniques, and can justify any adverse behavior (rejections).
- **The key questions:**
- Should we approve clients based on the information they provided?
- What kinds of clients are likely to default on their loans?
- What are essential features to consider while approving a loan?

Data Exploration - Variables

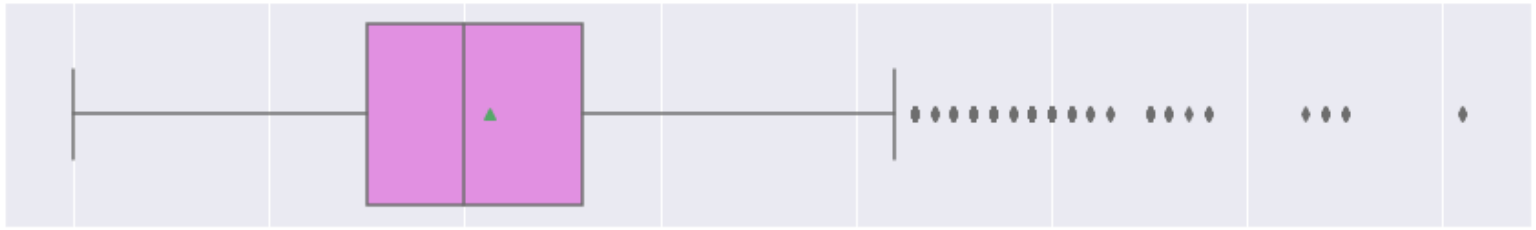
5960 Cases → 12 variables → 20% Default Cases

- **Target Variable – "BAD"- Binary:** 1 = Client defaulted on loan, 0 = loan repaid
- **LOAN:** Amount of loan approved.
- **MORTDUE:** Amount due on the existing mortgage.
- **VALUE:** Current value of the property.
- **REASON:** Reason for the loan request. (HomeImp = home improvement, DebtCon= debt consolidation)
- **JOB:** The type of job that loan applicant has
- **YOJ:** Years at present job.
- **DEROG:** Number of major derogatory reports (which indicates a serious delinquency).
- **DELINQ:** Number of delinquent credit lines (fail to make the minimum payments 30 to 60 days past the due date).
- **CLAGE:** Age of the oldest credit line in months.
- **NINQ:** Number of recent credit inquiries.
- **CLNO:** Number of existing credit lines.
- **DEBTINC:** Debt-to-income ratio (monthly debt payments divided by gross monthly income -- Measurement Ratio)

Data Exploration – Cleaning

- **Clean Up Missing Value** – Fill missing categorical value with mode and numerical value with median
- **Remove Outliers** – Remove outliers from each columns (outliers in the box plot need to be removed)

- **No missing value and no outliers -> Data is ready to be modeled**



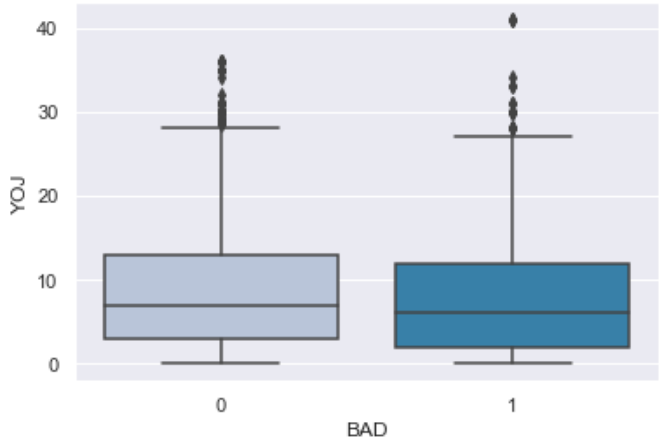

```
0   BAD       5960 non-null   category
1   LOAN      5960 non-null   int64
2   MORTDUE   5442 non-null   float64
3   VALUE     5848 non-null   float64
4   REASON    5708 non-null   category
5   JOB       5681 non-null   category
6   YOJ       5445 non-null   float64
7   DEROG     5252 non-null   float64
8   DELINQ    5380 non-null   float64
9   CLAGE     5652 non-null   float64
10  NINQ      5450 non-null   float64
11  CLNO      5738 non-null   float64
12  DEBTINC   4693 non-null   float64
```

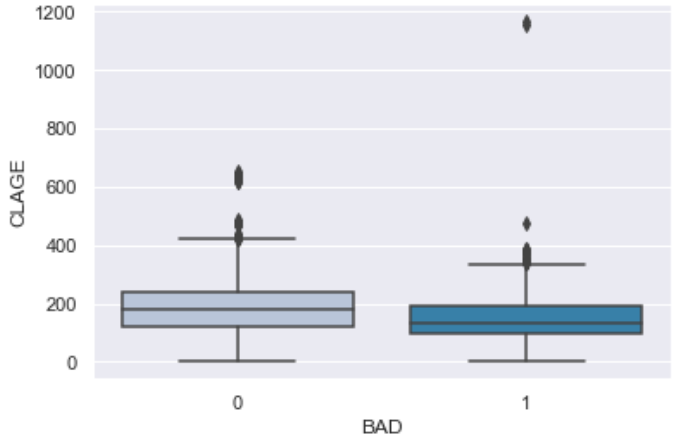
```
% of missing values in the each column
BAD          0.000
LOAN         0.000
MORTDUE      8.691
VALUE        1.879
REASON       4.228
JOB          4.681
YOJ          8.641
DEROG       11.879
DELINQ       9.732
CLAGE        5.168
NINQ         8.557
CLNO         3.725
DEBTINC     21.258
```

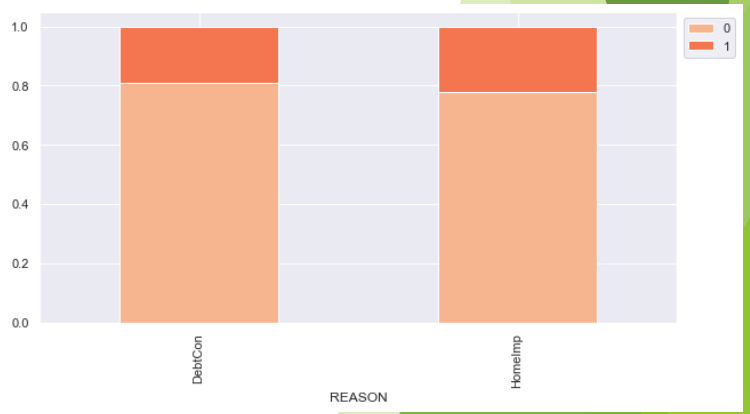
Data Exploration – Preliminary Findings

Some key findings in the data is worth noted:

- People are more likely to default if they have:
  - Lower years of working experience
  - Lower age of credit history
  - Higher Debt to income ratio
  - Higher number of existing credit lines
  - High amount of delinquency credit lines or derogatory reports
  - Require Loan for home improvement







Proposed Approach

- **Potential techniques**
- Popular classification Models: Decision Tree, Random Forest, logistic regression, Support Vector Machine, KNN
- Train-test split & k-fold cross-validation & hyperparameters tunning
- Dimension reduction: PCA and TSNE. Since we only have 10 columns, it is not necessary to use PCA.

- **Overall solution design**

1. Split the dataset to 20% Test and 80% Train. We can also use k-fold to split it into multiple train-test sets
2. Train models, check the evaluation matrix on training and testing data, and determine target measurement matrix
3. Tune the models using the grid-search method and find the best working model
4. Deploy the model and explore the most important features
5. Evaluate features to check their rationale and recommend our model and features to the management team.

- **Measures of success**
- Confusion matrix, Jaccard score, precision, recall, and f1-score
- Investigate the important features to ensure they are reasonable.