Investing in Peer2Peer Lending

Section 1
Team Gamma 1

Cameron Santos, Trey Tanjuatco, Noah Merenbloom,
Sam Gomez Olvera, Justin Schmitt, Daisy Zhou

LendingClub

Agenda

1. Big Question
2. Cleaning and Imputing the data
3. Accepted/Rejected Model
4. Loan Default Model
5. Investment Financial
6. Recommendations and Implications

Big Question: Where should Dr. D invest his money?

Goal: Maximize $10,000,000 investment with respect to key factors

Key Factors:

- Loan defaulting
- Return/value of loan
- Time of loan
- Possibility of delay in payment

How:

→ Identify predictive features for loan approval and status
→ Evaluate the financial outcomes for default
→ Determine how to value a loan based on desired risk

LendingClub

Find common feature variables and impute null and NaN values in accepted data

Compared the feature variables between the accepted and rejected data and found 9 features that will match between the two datasets

ACCEPTED DATA

1. 'Loan_amnt' → Drop values with missing loan amount
2. 'Issue_d' → Drop values with no date
3. 'Purpose' → Impute missing values with 'Other'
4. 'Fico_range_low'
5. 'Fico_range_high'
6. 'Dti' → Impute missing value with mean DTI
7. 'Addr_state'
8. 'Emp_length'
9. 'Policy_code'

- ❑ Feature engineer a binary 'accepted' score
- ❑ Calculate a risk score by averaging the high and low FICO score

LendingClub

Finish cleaning and imputing rejected data and combine accepted and rejected dataframes of common features

Compared the feature variables between the accepted and rejected data and found **9 features** that will match between the two datasets

ACCEPTED DATA
+
REJECTED DATA

1. 'Loan_amnt'
2. 'Issue_d'
3. Purpose
4. 'Fico_range_low'
5. 'Fico_range_high'
6. 'Dti'
7. 'Addr_state'
8. 'Emp_length'
9. 'Policy_code'

Combined data frame of ACCEPTED and REJECTED data has 9 common feature variables plus **'Accepted?'** and **'Risk_Score'** engineered variables

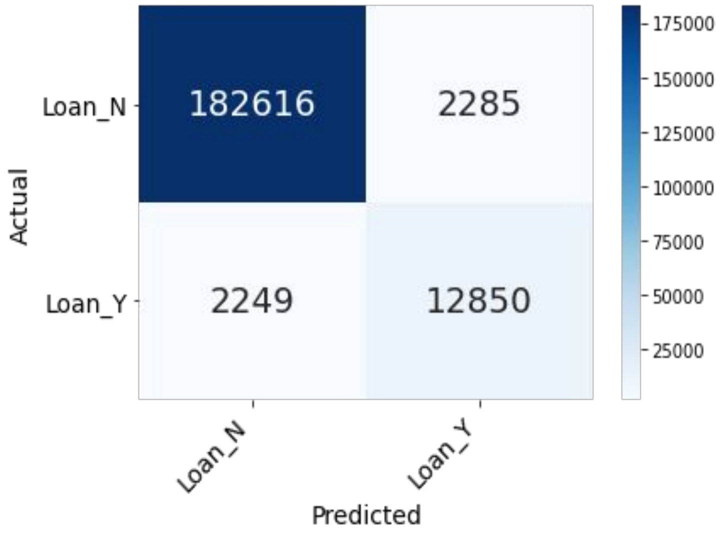
LendingClub

jupyter

Accepted/Rejected Model for prediction

Results: This model has an accuracy of 0.9773 in predicting whether a loan is accepted/rejected. It scores highly for recall and precision.

- ❑ Read in all concatenated data from drive
- ❑ Take a random sample and clean data, dfSample
- ❑ One-hot encode data
- ❑ Utilize train_test_split
- ❑ Implement **Random Forest**
- ❑ Evaluate model accuracy via Confusion Matrix

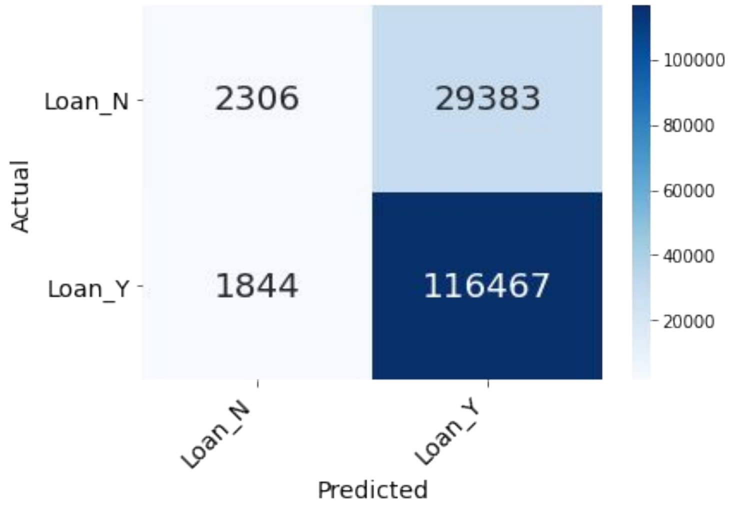


LendingClub

Based off accepted/rejected model, we created a model for predicting loan defaults

Results: This model has an accuracy of 0.79182 in predicting whether a loan will default through logistic regression

- Take smaller sample of accepted data to Colab crashed / more manageable
- Scale features (Standard Scaler)
- One-hot encode data
- Utilize train__test__split
- Implement **logistic regression**
- Evaluate model accuracy via Confusion Matrix



LendingClub

Calculating ROI based on the logistic regression model

$$\underline{\textbf{Simplified ROI of a loan}} = \left[\frac{\text{Monthly installment} \times \text{Term Length (36 or 60 months)}}{\text{Total loan amount}} \right] - 1$$

ROI of a loan considering the risk of default = Simplified ROI of a loan x Chance of Default from Logistic Regression

LendingClub

Assuming an initial investment of $10M, Dr. D can average over $1.01M in profit (over 3 years) from investing in Lending Club

Investing in Lending Club

ROI: 10.16% three-year rate

Implications:

- ❑ Higher level of risk (A-F)
- ❑ Higher level of return
- ❑ ROI is normalized to 3 month horizon
- ❑ Return on $10M =
  - ❑ ~$1,016,000

VS

Investing in Treasury Bonds

ROI: 3.44%* three-year rate

Implications:

- ❑ Stable return
- ❑ Guaranteed by government
- ❑ Return on $10M =
  - ❑ ~$344,000

*ROI of three-year treasury rate based upon historic value, not current rate

LendingClub