Springboard Data Science Career Track

Capstone 1

Lending Club Data - Predicting Return on Investment for Investors

Derek Samsom

November 2018

Introduction

Background

Peer-to-peer lending is a an online service that matches borrowers and lenders. Online lending typically has lower overhead costs than traditional financial institutions, allowing for better interest rates for borrowers, and a greater return for investors than with other investment products..

Problem Statement:

While peer-to-peer lending is attractive due to the higher returns compared to other financial investment products, there is still risk a financial loss to the lender if a loan defaults and is not paid back in full. Riskier loans have a higher interest to offset the higher risk of default. If a lender wants to maximize their potential profit, they have to accept a higher risk loan.

One way to try to minimize the loss due to default is to try try to predict whether a loan will default or not (classification). While predicting if a loan will default or not is valuable, it doesn't take everything into account. For example, there could be a situation where a high-interest loan defaults after being nearly paid off, that actually pays back the lender more than a low-interest rate loan that is paid in full. Therefore, this project will take a different approach and predict the return on investment (the amount repaid less the original loan amount, divided by the original loan amount.

My client is an investor that wants to maximize their return on their lending, and minimize the risk of losses due to default. When an investor obtains a list of available loans they can invest in, I would like to provide a prediction model that will generate a prediction of the return on investment, allowing the investor to choose loans with a higher ROI than would be chosen without a prediction model.

Approach

Data

The data for this project comes from https://www.lendingclub.com/info/download-data.action. The data is available for download in csv format. Lending Club loans are for either 3 or 5 year terms. Therefore, the data used was for years 2012-2013 since this is the latest data at this time that contains completed 5-year loans. There are 188,183 loan observations, and 145 columns in the original data set.

Data Dictionary

Variable	**Description**
bc_open_to_buy	Total open to buy on revolving bankcards.
application_type	Indicates whether the loan is an individual application or a joint application with two co-borrowers
disbursement_method	The method by which the borrower receives their loan. Possible values are: CASH, DIRECT_PAY

dti	A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income.
earliest_cr_line	The month the borrower's earliest reported credit line was opened
emp_length	Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years.
funded_amnt	The total amount committed to that loan at that point in time.
grade	LC assigned loan grade
home_ownership	The home ownership status provided by the borrower during registration or obtained from the credit report. Our values are: RENT, OWN, MORTGAGE, OTHER
installment	The monthly payment owed by the borrower if the loan originates.
int_rate	Interest Rate on the loan
issue_d	The month which the loan was funded
last_credit_pull_d	The most recent month LC pulled credit for this loan
mths_since_last_delinq	The number of months since the borrower's last delinquency.
mths_since_last_major_derog	Months since most recent 90-day or worse rating
mths_since_last_record	The number of months since the last public record.
mths_since_recent_bc_dlq	Months since most recent bankcard delinquency
mths_since_recent_revol_delinq	Months since most recent revolving delinquency.
num_actv_rev_tl	Number of currently active revolving trades
num_rev_tl_bal_gt_0	Number of revolving trades with balance >0
num_sats	Number of satisfactory accounts
open_acc	The number of open credit lines in the borrower's credit file.
policy_code	publicly available policy_code=1 new products not publicly available policy_code=2
purpose	A category provided by the borrower for the loan request.
revol_util	Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
title	The loan title provided by the borrower
tot_cur_bal	Total current balance of all accounts
tot_hi_cred_lim	Total high credit/credit limit
total_pymnt	Payments received to date for total amount funded
zip_code	The first 3 numbers of the zip code provided by the borrower in the loan application.

Data Cleaning and Wrangling

The first steps were remove the unnecessary data. Only the variables that contain information known before the loan is funded will be useful for predicting the ROI, so any column that contains information that is collected after the loan is funded will be removed, except for total_pymnt, which will be used to calculate ROI, the dependent variable. Loans that are still within their repayment period were removed, since the total_pymnt is not yet known for these.

Next, I took a look at the missing values. There were several variables where all observations were null, which were dropped.

There were 20 columns that had the same number of missing observations. I found that there was no data collected for these columns in the first 6 months of the 2-years of data. I removed these observations from the data, leaving 1.5 years of complete data, rather than 2-years with a lot of imputed data.

There were a few variables that represent months since a delinquency occurred, such as months since there was a delinquency on revolving credit. In these cases, I suspect that missing values represent loans where the applicant hasn't had the delinquency, which is meaningful but is something that can't be represented numerically like the number of months. Therefore, I converted these to categorical variables, with categories ‘none’, ‘1-2 years’, etc.

The remaining missing data was replaced with the median.

There were 2 date variables the represented the date of the last credit pull and the date of the loan applicant’s earliest credit history. These variables were converted numeric variables that showed the length of time since the events, rather than the date of the event.

I also checked for columns that only had one unique value, and removed them since they won’t add anything to the model.

Finally, I checked to make sure the remaining variables were all of the right type. Object variables were converted to categorical. Two variables, ‘interest_rate’ and ‘revol_util’,   were percentages that were misread as objects, so I removed the ‘%’ symbol, and converted them to numerical.

Data Storytelling and Inferential Statistics

ROI and Loan Grade

I originally intended to use the amount paid back on the loan as the dependent variable, however, this created a challenge: whether the amount repaid is good or bad can’t be determined without considering the original loan amount. Looking at the amount repaid by grade highlights this.

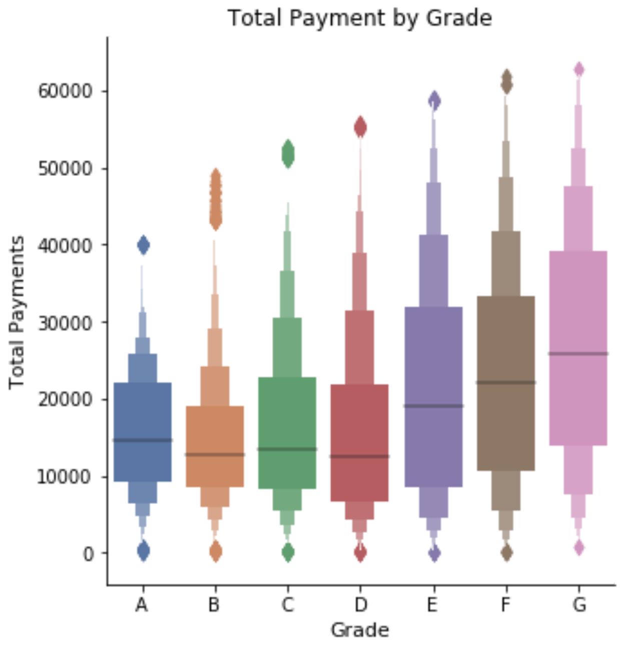

Figure 1: Total Payments by Loan Grade

Figure 1 shows that G-grade loans pay back the most, but that doesn't mean they are better-performing loans. The only way to determine how well they perform is to also look at how much was originally borrowed. Since G is the lowest grade, I expect that the only thing that could account for the high repayment amount is a high initial loan amount.

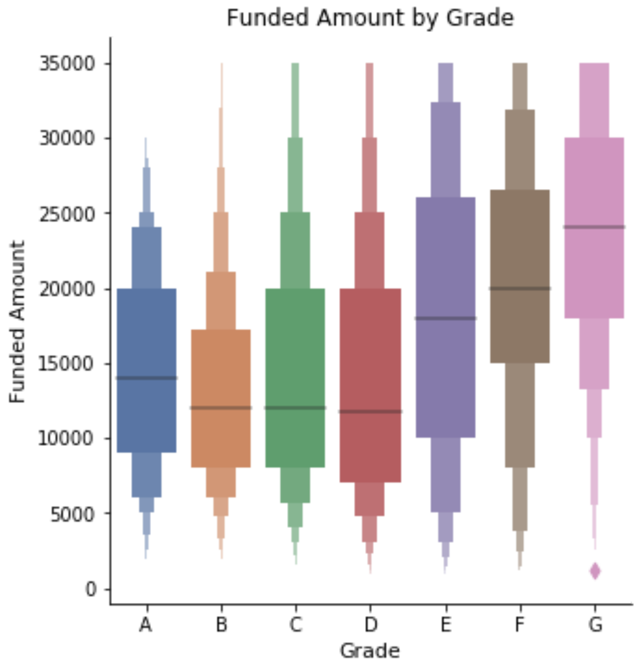

Figure 2: Funded Amount by Loan Grade

Looking at the funded amount by grade in Figure 2 confirms that the G-grade loans indeed have the highest median loan amount of about $25,000. This suggests that the loan amount requested affects the assigned grade, rather than a predetermined grade affecting how much an applicant is allowed to borrow.

The solution to the problems associated with looking at just the amount repaid as the dependent variable is to use ROI instead, which which is the ratio of the profit compared to the amount borrowed, and removes the effect of loan size.

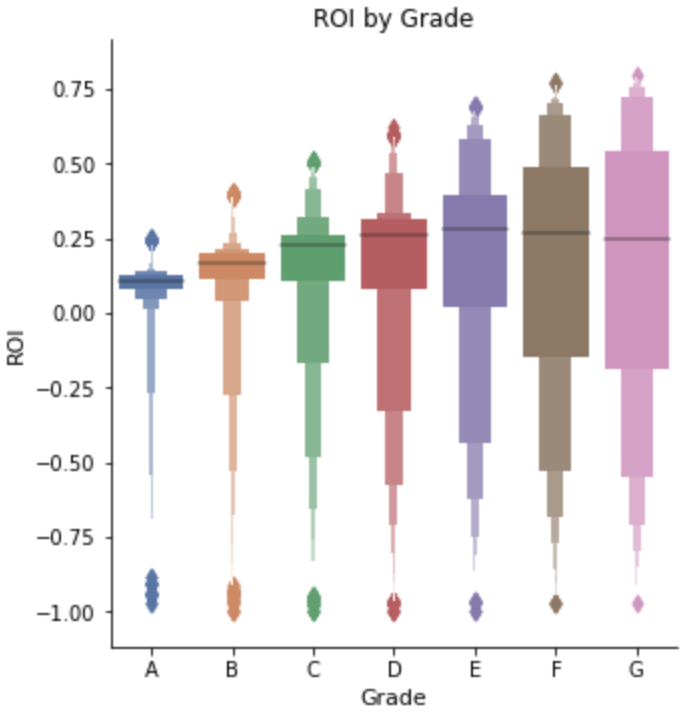

Figure 3: ROI by Loan Grade

Looking at the ROI by grade in Figure 3 is more useful than just the amount repaid by grade. Since ROI removes the effects of original loan size it does a better job of showing how well the loans are performing.

The least-risky A-grade loans have the lowest median ROI of around 10%. Grades E, F, and G have higher average ROI, but the higher risk is shown by the bigger spread of ROI values, with a higher proportion of negative ROI values.

Grade E loans have the highest mean ROI of 16.0% and highest median ROI of 28.4%, meaning that on average, E loans will provide investors the highest returns. This demonstrates that the higher interest rates on the lower-graded loans more than makes up for the losses due to defaults through grade E, on average. Past grade E, the defaults start taking away more of the additional profits made from the higher interest rates, which suggests that these are getting into the interest rate territory that fewer borrowers are willing to pay.

There are some things to consider regarding the use of ROI rather than amount repaid as the dependent variable. When predicting the amount repaid, the original loan amount is a strong predictor. When predicting ROI, the original loan amount will no longer be nearly as important, if at all, and the model may not score as highly yet still be as useful.

Second, it highlights the fact that there is already an attempt to even out the ROI by the assigning higher interest rates to riskier loans, making the differences in ROI across a predictor variable more subtle. In the case of loan grade, however, it is easy to see that there are still differences in the ROI across the range of values, giving the models something to work with. I can perform statistical tests to see if the differences in ROI are significant.

Loan Purpose

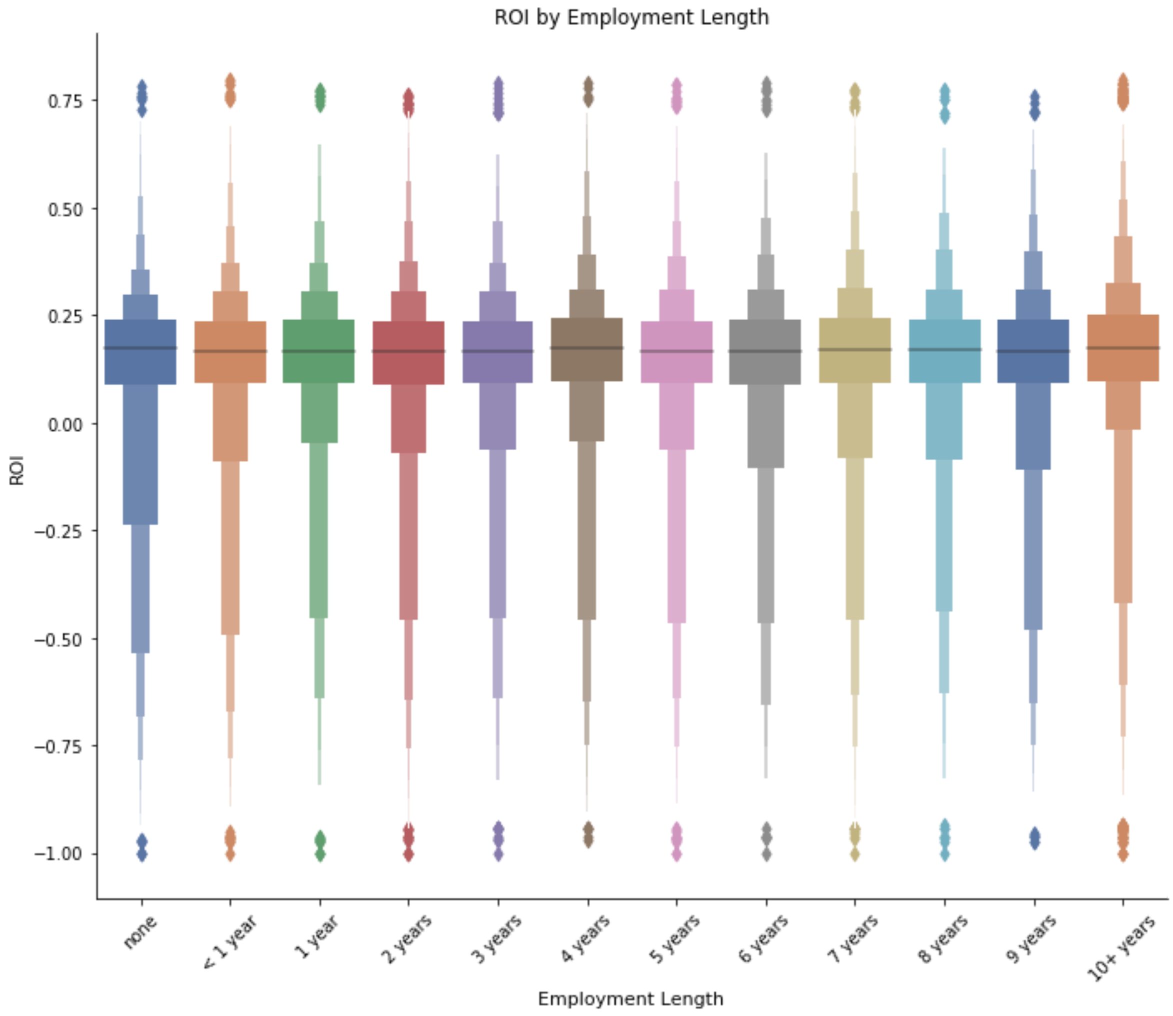

Figure 4: ROI by Loan purpose

The ROI by purpose seen in Figure 4 can be broadly put into two groups. Loans for cars, credit cards, debt consolidation, home improvement, and house have lower ROIs, but are less risky as they have lower standard deviations.

Loans for medical, moving, other, renewable energy, vacation, and weddings tend to have higher ROIs than the first group, bit also more risk of default with higher standard deviations.

Small business loans are the one category that is hard to put into these two groups, as it has the lower median ROI of the first group, and the higher risk associated with the second group. It also has both the lowest mean ROI and the highest standard deviation, making it, on average, the worst loan purpose to invest in.

I set up a hypothesis test to determine whether the ROI between the first and second groups is statistically significant:

Null hypothesis: There is not a difference in mean ROI between group 1 and group 2 borrowers.
Alternate Hypothesis: There is a difference in mean ROI between group 1 and group 2 borrowers.

Test Statistic: Difference of mean ROI.
Significance level: 0.01

The observed difference in mean ROI was 0.6 %. I took 10,000 bootstrap permutation samples, and calculated the test statistic. None of the 10,000 permutation samples produced a difference of means as great as 0.6%, resulting in a p-value of 0.0. Therefore, the null hypothesis was rejected.

ROI by DTI

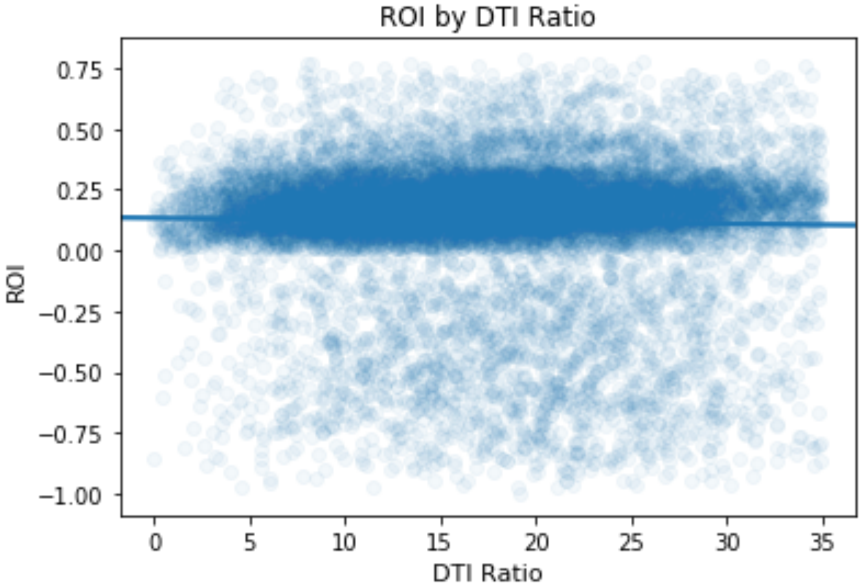

Figure 5: ROI by Debt-to-Income(DTI) Ratio

Looking at ROI by DTI (Figure 5), , there is a very slight negative correlation that is hard to detect visually.

I created two groups, a low-DTI group with DTI below 10, and a high-DTI group with DTI above 25, and set up a hypothesis test to determine if the mean ROI between the groups is statistically significant:

Null hypothesis: There is not a difference in mean ROI between low-DTI borrowers (DTI < 10) and high-DTI borrowers (DTI >= 25).

Alternate Hypothesis: There is a difference in mean ROI between low-DTI borrowers(DTI < 10) and high-DTI borrowers (DTI >= 25).

Test Statistic: Difference of mean ROI.

Significance level: 0.01

The observed difference in mean ROI is 0.6 %. I took 10,000 bootstrap permutation samples, and calculated the test statistic for each. Only one of the 10,000 permutation samples produced a difference of means as great as 0.6%, resulting in a p-value of 0.0001, which is below the 0.01 significance level. Therefore, the null hypothesis was rejected.

Despite the small differences seen in mean ROI across the range of DTI and the loan purpose groups, the differences are still statistically significant. Despite the tendency of ROI to be somewhat evened out across the ranges of predictor variables by increasing the interest rates to offset loans that are more likely to underpay, there are still significant differences that the models should be able to pick up on.

Machine Learning Base Models

The following three models will be used to predict the loan ROI.

- Linear Regression
- Random Forest
- Gradient Boost

The data was split into train, validation, and test sets, as I used the validation set for hyperparameter tuning rather than cross-validation due to the size of the data set and computational limitations.

First I will run the linear regression, random forest, and gradient boosting models with the defaults and no tuning to get some baseline scores.

I will evaluate performance based on the following metrics:

- Validation Scores (R-squared)
- Root Mean Squared Error (RMSE)
- Residuals, 90th percentile
- Residual plot and histogram.

Linear Regression Baseline Model

Train Score:	0.0488
Validation Score:	0.0473
RMSE:	0.2556
Residuals, 90th Percentile:	-0.7594, 0.2355

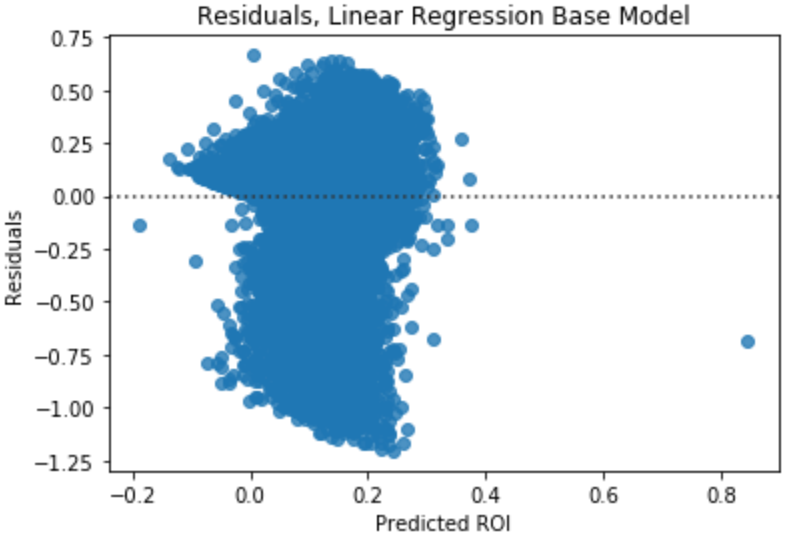

Figure 6: Residual Plot, Linear Regression base Model

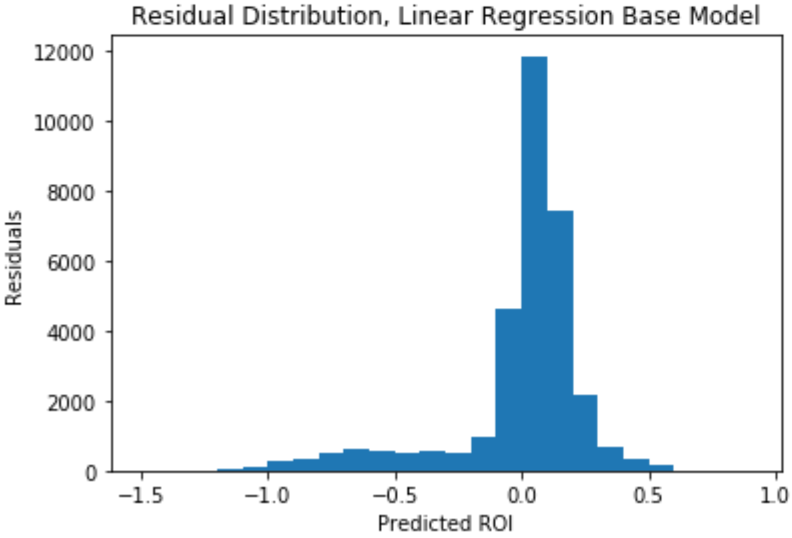

Figure 7: Residual Distribution, Linear Regression Base Model

The baseline linear regression model has a low train score of 0.0488 on the train set and 0.0473 on the validation set. The small difference between the scores indicates minimal overfitting, and it is likely that the model is underfitting based on the low scores on both sets.

The residuals are centered above zero, and the negative residuals have a heavy tail, which shows that there are cases where the model is over-predicting ROI by a large margin.

Random Forest Baseline Model

OOB Score:	0.2091
Validation Score:	0.2265
RMSE:	0.2303
Residuals, 90th Percentile:	-0.645, 0.2529

The base random forest model has an OOB score of 0.2091 and a validation score of 0.2265. The validation score is quite a bit higher than the linear regression score of 0.0473.

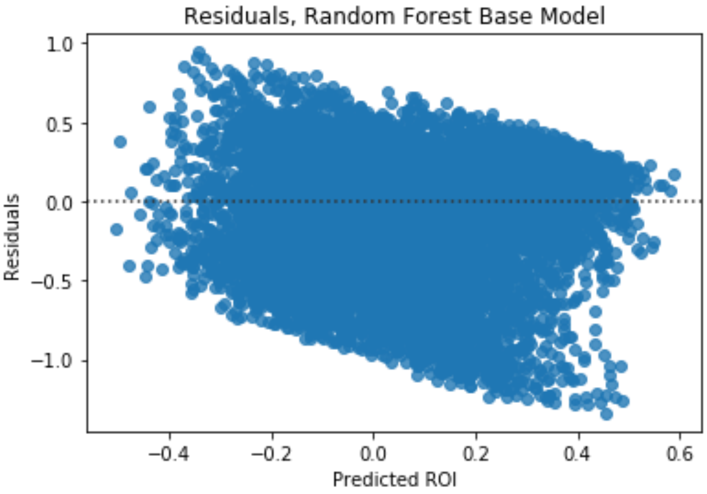

Figure 8: Residuals, Random Forest Base Model

Blank

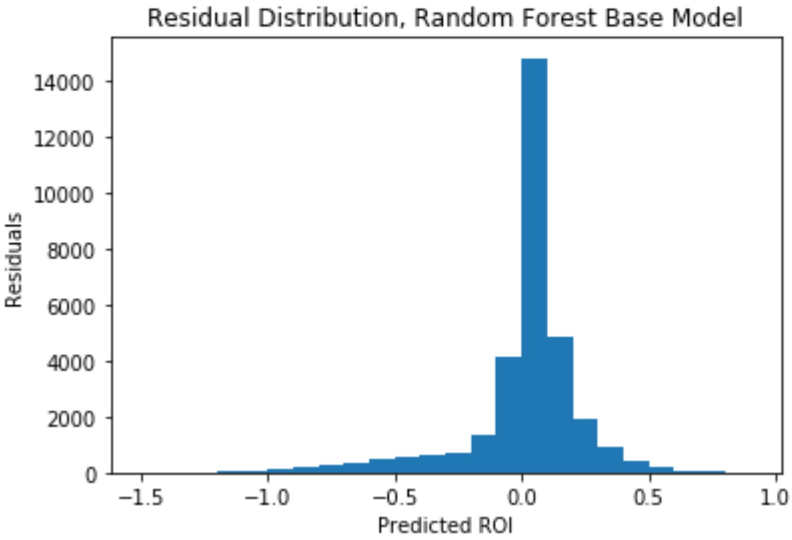

Figure 9: Residual Distribution, Random Forest Base Model

The residuals distribution in Figure 9 for the random forest base model shows that the residuals mostly fall between -0.10 and 0.20, and a large percentage between 0 and .1, but there is still a heavy tail on the negative side resulting in a 90th percentile range of -0.645 - 0.253.

Gradient Boost Baseline Model

Train Score:	0.2007
Validation Score:	0.1941
RMSE:	0.2351
Residuals, 90th percentile:	-0.6596, 0.2422

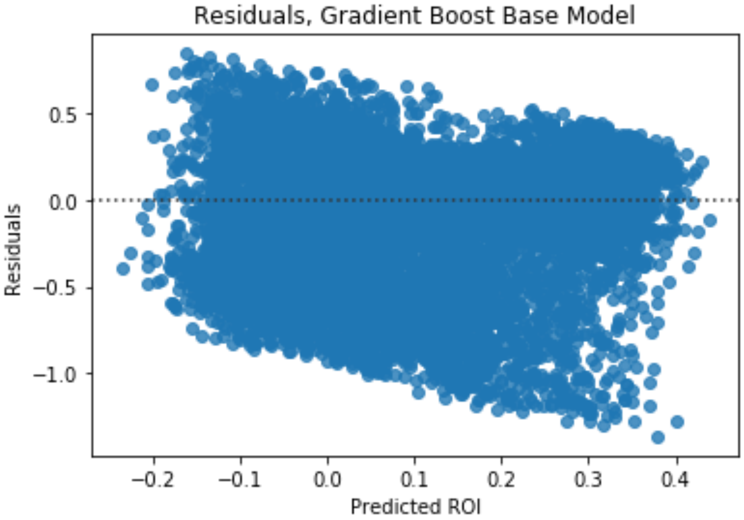

Figure 10: Residuals, Gradient Boost Base Model

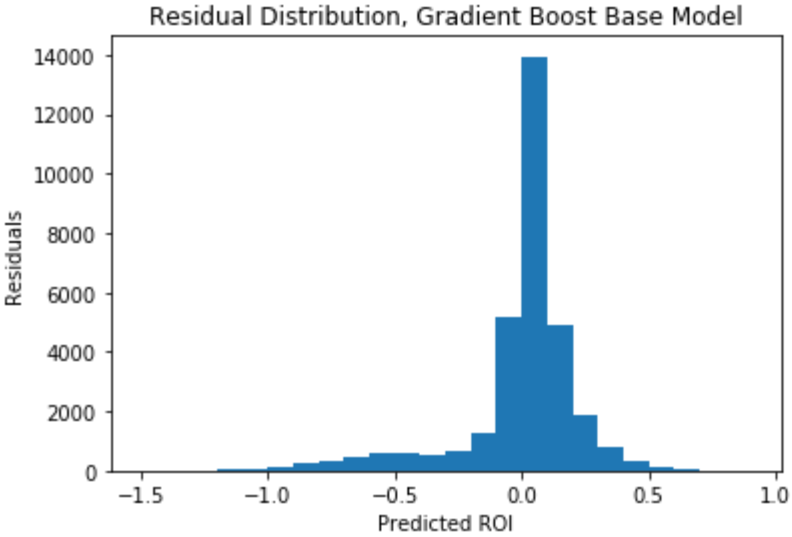

Figure 11: Residual Distribution, Gradient boost Base Model

The gradient boost baseline scored 0.1941 on the validation set, which not quite as high as the 0.2265 scored on the baseline random forest. The residuals are centered slightly above zero, with the negative residuals having a heavy tail as with the other models.

Table 1: Base Model Summary

Baseline Model	Validation Score	RMSE	Residuals, 90th Percentile
Linear Regression	0.0473	0.2556	-0.7594, 0.2355
Random Forest	0.2265	0.2303	-0.645, 0.2529
Gradient Boost	0.1941	0.2351	-0.6596, 0.2422

Table 1 summarized the scores for the 3 baseline models. The baseline random forest and gradient boost models are outperforming the baseline linear regression model, with much higher validation scores and lower RMSE.

Further Model Tuning

LInear Regression

I applied lasso and ridge regularization to the linear model to see if there would be any improvements. I tested each model over a range of alpha values, and determined that a lasso model with an alpha value of 0.0001 offered the most improvement in validation score and RMSE, however the improvement was minimal.

Baseline Linear Regression

Train Score:	0.04884
Validation Score:	0.04731
RMSE:	0.25560
Residuals, 90th percentile:	-0.7594, 0.2355

Lasso Regression

Train Score:	0.04808
Validation Score:	0.04756
RMSE:	0.25556
Residuals, 90th percentile:	-0.7592, 0.2359

The improvements of the ridge model over the baseline model were quite minimal, offering just a slight improvement over the baseline validation score and RMSE. The lower overall scores than the other models and the similarity between the training and test scores suggest that the linear regression model is underfitting rather than overfitting. Since lasso and ridge regularization are meant to address overfitting, it isn’t a surprise that they did not offer much improvement. Further research and variable transformation is needed to address the underfitting.

Random Forest

I first checked to see if dropping some of the least important variables would change the model. Dropping variables with feature a feature importance of less than 0.0015 resulted in dropping 70 of the 146 variables, with a minimal reduction in the scoring metrics. Beyond the 0.0015 threshold, further reduction of dimensions resulted in a larger negative impact on the scores.

With the reduced dimension data set, I tuned the max_features and n_estimators hyperparameters by plotting the out-of-box error rates across the range of n_estimators for three values of max_features: square root, log2, and None(all features).

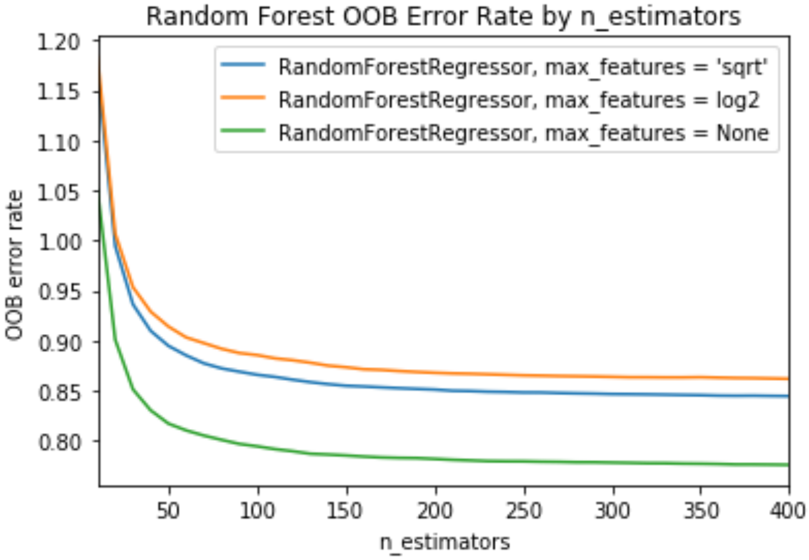

Figure 12: Random Forest OOB Error Rate by n_estimators

Figure 12 shows the out-of-box error rate is lowest using max_features = None, which uses all features. The reduction in error rate starts to level off past around 200. I chose Max_features = None and n_esimators = 400 for the hyperparameters.

Baseline Random Forest

OOB Score:	0.2091
Validation Score:	0.2265
RMSE:	0.2303
Residuals, 90th percentile:	-0.645, 0.2529

Tuned Random Forest

OOB Score:	0.2259
Validation Score:	0.2321
RMSE:	0.2295
Residuals, 90th percentile:	-0.6437, 0.2552

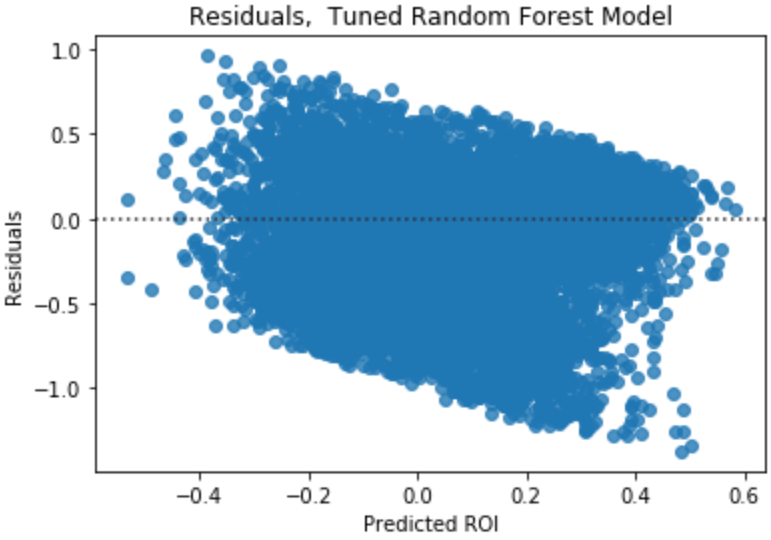

Figure 13: Residuals, Tuned Random Forest Model

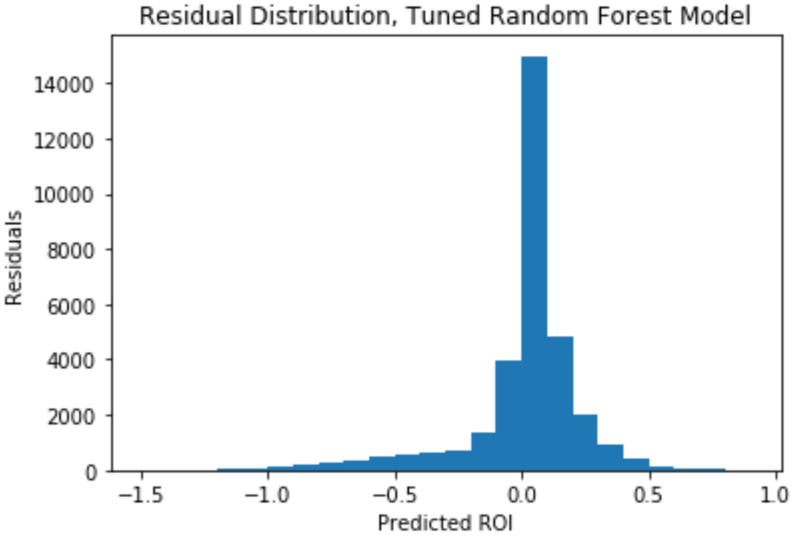

Figure 14: Residuals Distribution, Reduced Dimension and Tuned Random Forest

The dimension reduction and tuning of the random forest resulted in modest improvements on the validation score from 0.2265 to 0.2321, and reduced the RMSE from 0.2303 to 0.2295. Looking at the distribution of residuals in Figure 14, the residuals are still centered just above zero and there is a long tail of negative residuals, but it is reduced in comparison to the base model.

Gradient Boost

To tune the gradient boost, I did a grid search over the following hyperparameters, all while increasing the number of estimators to 1000:

Learning_Rate:	0.1, 0.05, 0.02
Min_Samples_Leaf:	1, 3
Max_Features	1, 0.5, 0.2

The best results were achieved with learning_rate 0.1, max_features 0.5, and min_samples_leaf 1.

Baseline Gradient Boost

Train Score:	0.2007
Validation Score:	0.1941
RMSE:	0.2351
Residuals, 90th percentile:	-0.6496, 0.2422

Tuned Gradient Boost

Train Score:	0.3079
Validation Score:	0.2456
RMSE:	0.2275
Residuals, 90th percentile:	-0.6221, 0.2407

Tuning the gradient boost increased the validation score from 0.1941 to 0.2456, and reduced the RMSE from 0.2351 to 0.2275.

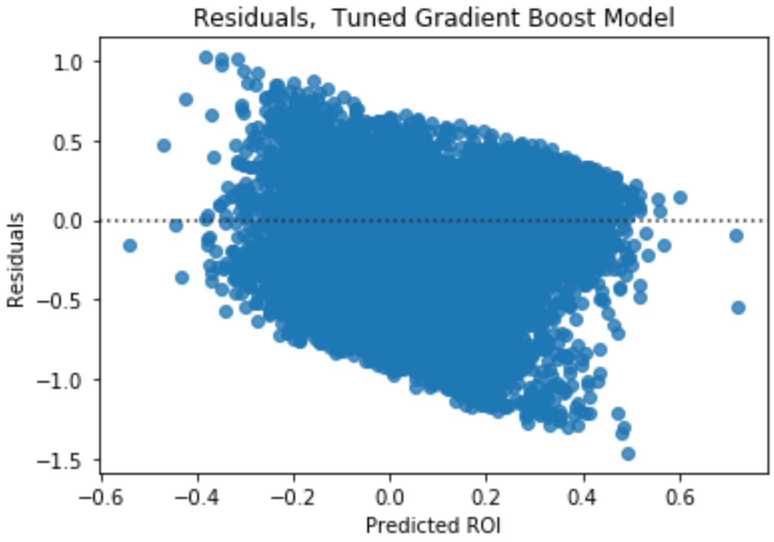

Figure 15: Residuals, Tuned Gradient Boost

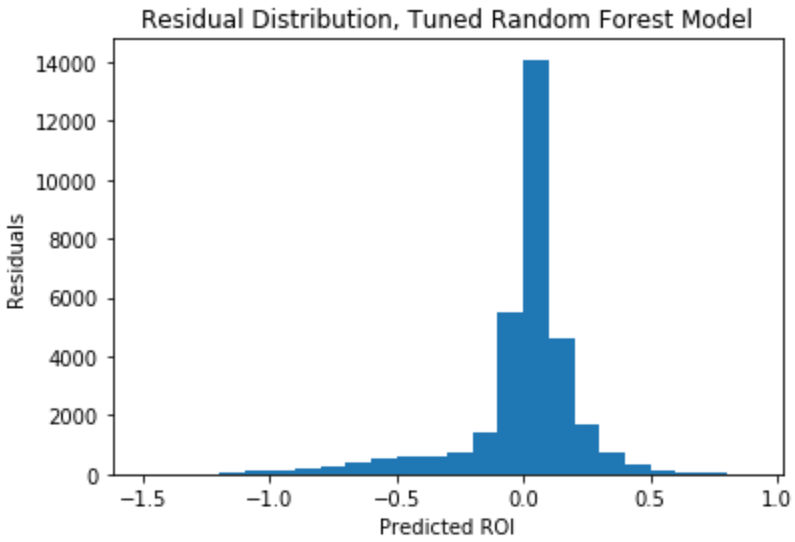

Figure 16: Residuals Distribution, Tuned Gradient Boost

The training and test residuals are still centered slightly above zero, but appear to be slightly closer to zero than the baseline model. The heavy tail in the negative residuals is still present, indicating there are still loans where the model is predicting a higher than actual ROI.

Next I will look at the scores for the baseline models compared to the tuned models.

Table 2: Summary of Baseline Model Scores

Baseline Model	Validation Score	RMSE	Residuals, 90th Percentile
Linear Regression	0.04731	0.2556	-0.7594, 0.2355
Random Forest	0.2265	0.2303	-0.645, 0.2529
Gradient Boost	0.1941	0.2351	-0.6596, 0.2422

Table 3: Summary of Tuned Model Scores

Tuned Model	Validation Score	RMSE	Residuals, 90th Percentile
Linear Regression	0.04746	0.25556	-0.7592, 0.2359
Random Forest	0.2321	0.2295	-0.6437, 0.2552
Gradient Boost	0.2456	0.2275	-0.6221, 0.2407

After tuning, the gradient boost looks like the best model to use. It has both the highest validation score and the lowest RMSE.

Findings

Analysis of Results

To best estimate how this model will perform on unseen data, I performed one final fit on all available data except the test data (the train and validation data).
The final results are a test R-squared value of 0.2545 and a RMSE of 0.2223.

The overall score is not that high, but the real test is to see how the model can help an investor choose higher ROI loans to invest in, and lower the risk of unprofitable loans. I will do this using two scenarios. For each scenario, I will look at the distribution of ROIs for the chosen loans with and without using the model, and I will also look at the percentage of loans with a negative ROI , which will give an idea of the risk of losing money.

In the first scenario, I will simulate the results that could happen if an investor had no model at all and randomly chooses loans, and compare that to what might be expected if the investor used the model, and only invested in loans that have a predicted ROI above the 95th percentile. I am picking the top 5 percent because not all loans are available at any given time so it isn't realistic to only select the few with the highest predicted ROI, and it will give me a large enough sample of loans to get an idea of the distribution of ROIs for those loans.

First I'll look at what can be expected when an investor chooses a loan at random.

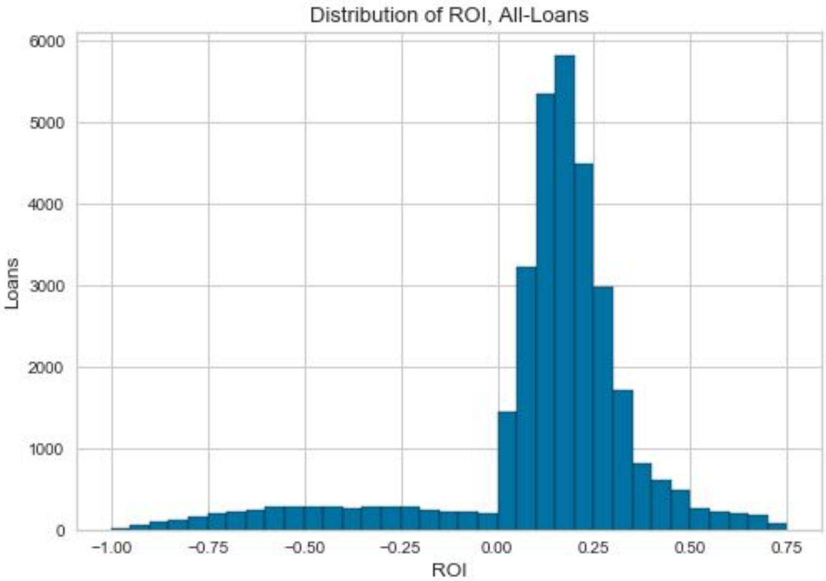

Figure 17: ROI Distribution, All Loans

Figure 17 shows the distribution of ROI for all loans. Choosing loans at random, the mean ROI is 12.2%, with most loans falling between 0 and 35% ROI. There is a long tail of loans with a negative ROI, and loans that pay greater than 35% are uncommon.

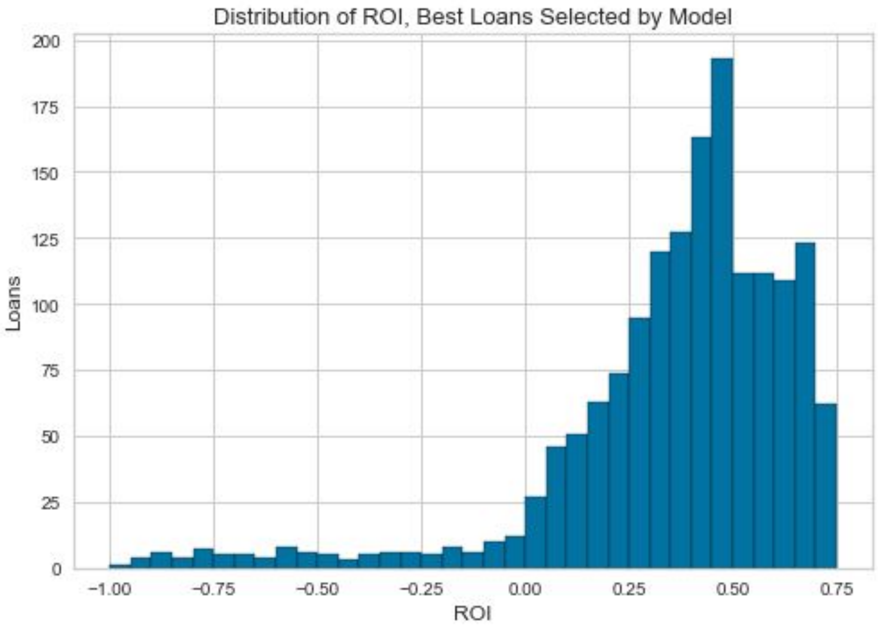

Figure 18: ROI Distribution, Top 5%Predicted ROI

Choosing the models that are only in the top 5 percentile of the loan predictions results in a much better distribution of actual ROI, seen in Figure 18. By only choosing from the top ROI predictions, the mean ROI increases from 12.2% to 36.8%. There are many more loans with an ROI above 35% in the loans selected from the model compared to the loans selected at random. There is still a long tail of negative-ROI loans, meaning that there is still a risk of losing money, but it is lower than choosing loans at random. In the total population of loans, 13.25% had a negative ROI, but it is reduced to 7.21% among the top loans chosen by the model.

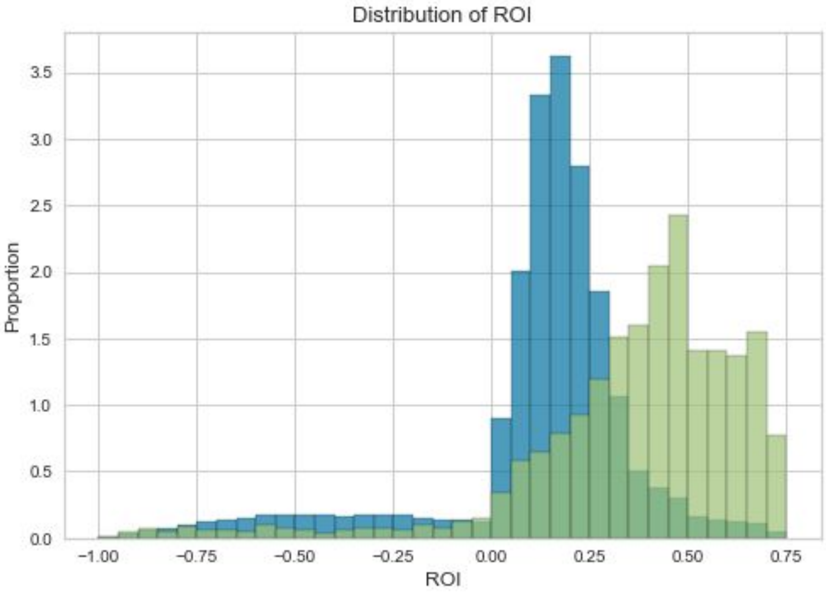

Figure 19: ROI Distribution, All Loans Compared to Top 5% Predicted ROI

Figure 19 shows both distributions overlapping, where it is easy to see the improvement in ROI by selecting the from the highest predictions (green) compared to selecting loans at random (blue). The model lowers risk by choosing fewer of the negative-ROI loans, and the proportion of loans above 30% ROI is much greater among the loans chosen by the model.

For the second scenario, I will assume most investors don't just select loans at random. While I can't run a scenario for all investment strategies, I can create a scenario from the earlier discussion of loan grade and purpose. E-grade loans have the highest average ROI, but are also riskier than higher grade loans. Also, I will exclude small business loans, since it was discovered that they have both the lowest average ROI and higher risk of default than the other loan purposes. In this scenario, I will compare the results of using the model to select the best loans, with the results of an investor who has a strategy to improve their mean ROI by selecting E grade loans, that are not for small businesses.

Again, I'll look at the distribution of ROIs as well as the average ROI and percentage of loans with a negative ROI.

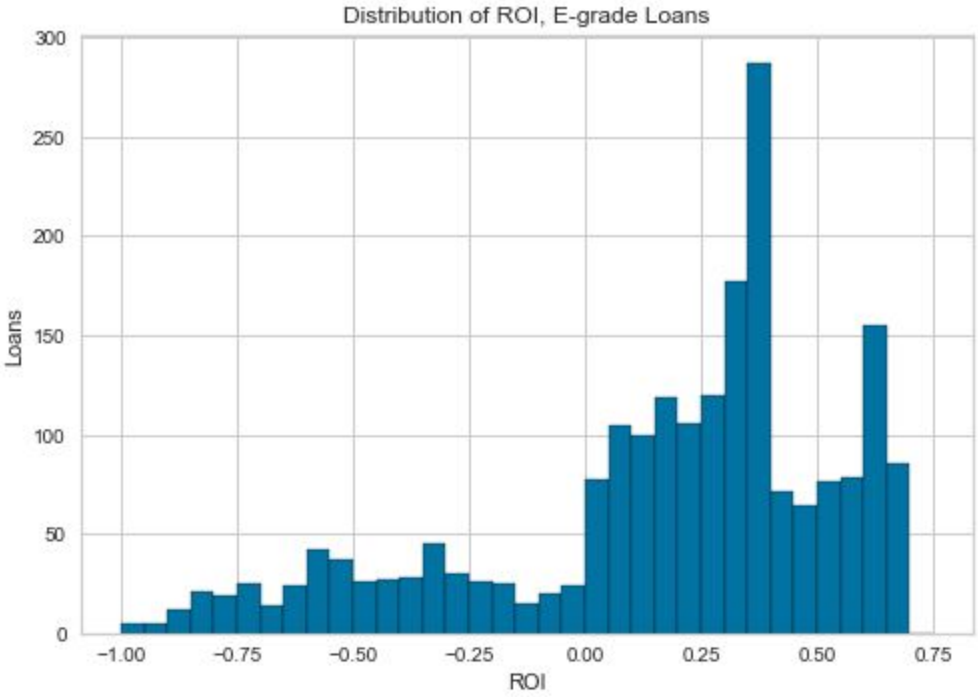

Figure 20: ROI Distribution, E-Grade Loans Excluding Small Business Loans

E-Grade loans excluding small business loans have a mean ROI of 17.32% compared to 12.2% for all loans. In Figure 20, you can see there is a larger proportion above 35% ROI than choosing from all loans at random, but there is also a higher proportion of loans with a negative ROI. In this scenario, investors are still better off selecting from the model's top predictions that choosing the e-grade loans that are not for small business, but by a smaller margin. The following plot overlaps the egrade loans(blue) with the loans from the top predictions(green).

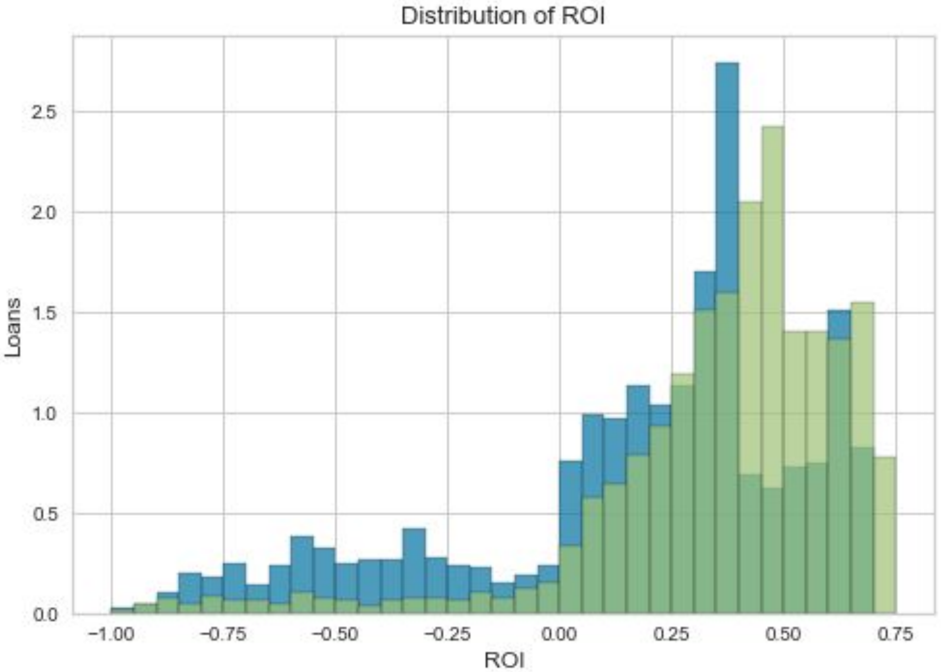

Figure 21: ROI Distribution, E-Grade Loans Excluding Small Business Loans

There is much more overlap in the second scenario, seen in Figure 21, but there are still some important differences for investors. The loans chosen by the model have a much lower risk than choosing E-grade loans excluding small business loans, as shown by the much smaller proportion of loans with negative ROI. Another important thing for investors is the higher proportion of the loans chosen by the model with an ROI above 40%.

Further Research

There are many areas of further research that could offer more help to loan investors. First, the linear regression is performing poorly compared to the other models, likely due to high bias/underfitting. This could be the result of making false assumptions of linearity of the data. Further research could be done to check the variables for linearity and to see if the model could be improved by transforming some variables.

Another area of further research is to look at scenarios for different investment strategies. For example, risk-averse investors are more likely to invest in A and B grade loans because they are willing to trade a lower average ROI for lower risk. It would be interesting to see how much the model would help these investors, and whether it would allow for a higher average ROI without increasing the low risk.

Research could also be done based on the length of loan, which is either 36 or 60 months. For this project they were grouped together, with the length as a predictor variable, however it would be interesting to model them separately and see the results. It would also allow for more recent data to be used in the case of 36 month loans.

Client Recommendations

Using the machine learning model can help peer-to-peer loan investors both increase their ROI, and decrease the risk of losing money on loans. The benefit of the model is unique to each investor's strategy for choosing loans to invest in.

When choosing loans at random, using the model is highly beneficial. The ROI is projected to increase from 12.2% to 36.8%, more than tripling the returns. For $100,000 invested, average earnings are expected to increase from $12,200 to $36,600.

For investors willing to invest in higher-risk, higher return E-Grade loans, using the model is expected to increase the ROI from 17.32% to 36.8%, while substantially reducing the risk that any given loan will have a negative return, dropping fromm 22% of loans down to 7.2% of loans.

For other investment strategies, I recommend calculating the summary statistics of the data to see the average ROI and percent of loans with a negative ROI that are expected using that strategy, and compare the results to using the model.

One final word of caution: Since the length of the loans is up to 60 months, the data used to build the models had to be at least 60 months old, since the ROI would otherwise not be known. I expect the accuracy of the models to erode over time due to changes over the last 5-7 years, such as economic changes or changes to the way the data is collected and categorized. The model would certainly have been helpful for choosing loans to invest in from 2012 to 2013, but there is uncertainty in how helpful the model will be for choosing loans that originate today. Still, many of the same general principles that were true 5-7 years ago will be true today, and the model should still offer a higher ROI and less risk for an investor.