Fintech Project: A Robotic Investor for Lending Club

Xiaofei Zheng [*]

Data Application Lab, Data Scientist Bootcamp

1 Definition

1.1 Project Background

Lending Club (a Peer to Peer lending marketplace) contains hundreds of loan projects, which makes it difficult for investors to choose a profitable one. In this FinTech project, we will use the data science knowledge to design a product as an intelligent investment advisor, helping investors identify the values of different projects in Lending Club, to determine the optimal projects to invest in. When the new loan project comes into the platform, our product would automatically analyze the project's parameters and screen out the best investment projects. A simple web page is also designed to realize the interaction between our product and users. Throught the web user interface, our product realizes the project evaluation and best investment project screening on Lending Club.

1.2 Problem Statement

The centerpiece of model's entire operation is using machine learning's algorithm to choose which loan to invest in. There are many ways to evaluate the performance of a loan, for example, the return on investment (ROI), or the risk level of the investment. Regression models can be applied to predict ROI based on the historical data and classification algorithms are able to predict whether the status of a loan will be charged-off or default. With a reliable prediction, it becomes possible for investors to pick up the good loans to invest. In this project, we will predict the status of the loans. That is, it is handled as a binary classification problem. We are more interested in predicting the probability of charged-off or default. The algorithms used are AdaBoost and XGBoost, which are common examples of boosting method. We have seen in many Kaggle competitions that XGBoost outperforms with so many advantages, for example, its nature of handling of data with heterogeneous features, strong predictive power, robustness to outliers in output space, ability to figure out important features and so on. After training an optimal model, we will save it using pickle and develop a web interface using Flask.

[*] xfzhengnankai@gmail.com

1.3 Metrics

For classification problems, accuracy is an important metric to evaluate the performance of models. However, for our problem, it is not enough. On one hand, if a "bad" loan is predicted to be a "good" loan, the investors may loss a lot of money and it will affect the credibility of our product. On the other hand, if we are too conservative to predict a loan to be a good one, then this will reduce the amount of transitions on our platform. So considering the two aspects together, metrics like AUC and F-β score are more appropriate to be applied to evaluate the performance of our models.

For readers who are interested in the mathematical definitions of those metrics, we show them here. Accuracy is defined to be the percentage of the correct predictions among all predictions. Precision is the percentage of correct prediction among all cases which are predicted to be positive (charged-off). Recall is the percentage of correct prediction among all cases whose true labels are positive (charged-off). We also use the F1 score (also F-score or F-measure) to evaluate the performance. F score considers both the precision and the recall. The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst at 0:

$$F_1 = 2 \times \frac{1}{\frac{1}{precision} + \frac{1}{recall}}.$$

2 Modeling Process

In the following sections, we will demonstrate the modeling process. This process is put into 5 separate experiments, with each containing one of the following steps as show in Figure 1 .

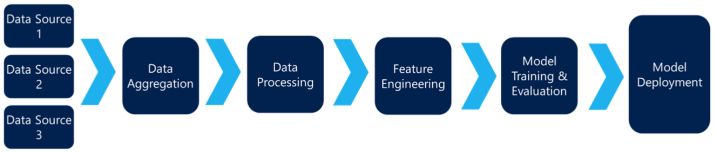


Figure 1: Modeling Process

2.1 Data Sources

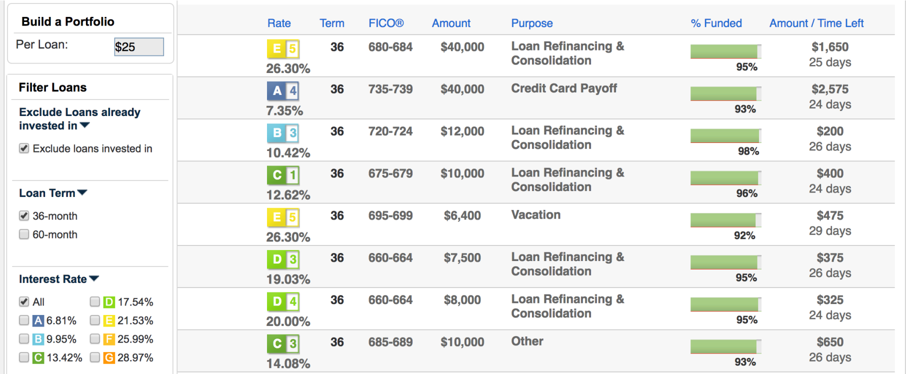
The datasets used in this project are from the official website of Lending Club[1]. Lending Club is the trailblazer in peer-to-peer lending, it has evolved into America's largest online marketplace that allows borrowers to apply for personal loans, auto refinancing, business loans, and elective medical

[1] Click me!

procedures. We downloaded the data of loans launched in 2014 (37646 kb) through this link and obtained the Json form of the current data via API (A personal account has to be created).

The data in 2014 contain complete loan data for all loans issued through the year 2014, including the current loan status (Current, Late, Fully Paid, etc.) and latest payment information. There are 128 features and 235629 rows. The definitions for all the data attributes included in the Historical data file can be found in the Data Dictionary. The reason why we choose the data in 2014 is that there are two types of loans: term 36 months and term 60 months. For this project, we only build the model based on the loans with term 36 months. Most of the loans issued in 2014 are to be mature for the moment, and we have more information of them.

The current data is about the loans that currently are to be funded. There are 105 features. The details of all the features can be found here. We downloaded the current dataset on 7/13/2017. Lending Club provides two ways of investing: manually investing and automatically investing. In Figure 2 and Figure 3 we show the interfaces of the two ways.



Build a Portfolio

Per Loan: $25

Filter Loans

Exclude Loans already invested in

☑ Exclude loans invested in

Loan Term

☑ 36-month
☐ 60-month

Interest Rate

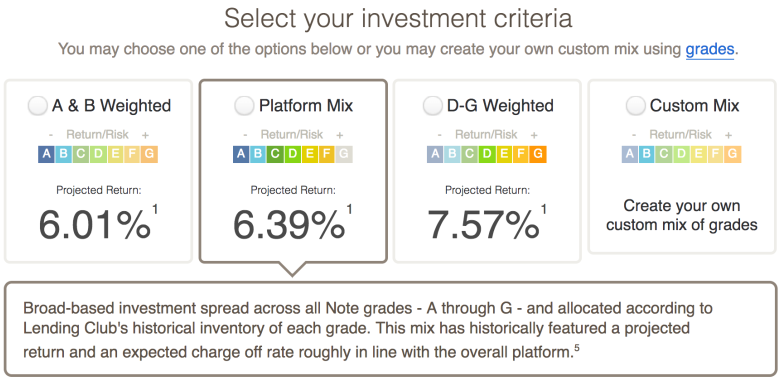
☑ All ☐ D 17.54%
☐ A 6.81% ☐ E 21.53%
☐ B 9.95% ☐ F 25.99%
☐ C 13.42% ☐ G 28.97%

Rate	Term	FICO®	Amount	Purpose	% Funded	Amount / Time Left
E 5 26.30%	36	680-684	$40,000	Loan Refinancing & Consolidation	95%	$1,650 25 days
A 4 7.35%	36	735-739	$40,000	Credit Card Payoff	93%	$2,575 24 days
B 3 10.42%	36	720-724	$12,000	Loan Refinancing & Consolidation	98%	$200 26 days
C 1 12.62%	36	675-679	$10,000	Loan Refinancing & Consolidation	96%	$400 24 days
E 5 26.30%	36	695-699	$6,400	Vacation	92%	$475 29 days
D 3 19.03%	36	660-664	$7,500	Loan Refinancing & Consolidation	95%	$375 26 days
D 4 20.00%	36	660-664	$8,000	Loan Refinancing & Consolidation	95%	$325 24 days
C 3 14.08%	36	685-689	$10,000	Other	93%	$650 26 days

Figure 2: Current Loans

2.2 Data preprocessing

Our goal is to make prediction for the current loans based on the model trained using the historical dataset. So it is necessary that the current dataset and the historical dataset have the same set of features. From the names and the meanings of the features, it turns out that there are 93 common features in both datasets. Although the features "issued" and "loan status" are not available in the current dataset, they are useful in the later process, we will keep them. Feature "issued" indicates the month in which the loans were published, so we will use it to split the training and testing datasets. Feature "loan status", as the name suggests, is the status of the loans when they are mature, which acts as our prediction target. Among the 95 features, there are 18 features which miss all the values through all observations, so we drop all of the 18 features. There are 59 numerical features and 18 categorical features remained. Figure 4 illustrates the information of the features.


Select your investment criteria

You may choose one of the options below or you may create your own custom mix using grades.

○ A & B Weighted
- Return/Risk +
A B C D E F G
Projected Return:
6.01%[1]

○ Platform Mix
- Return/Risk +
A B C D E F G
Projected Return:
6.39%[1]

○ D-G Weighted
- Return/Risk +
A B C D E F G
Projected Return:
7.57%[1]

○ Custom Mix
- Return/Risk +
A B C D E F G
Create your own custom mix of grades

Broad-based investment spread across all Note grades - A through G - and allocated according to Lending Club's historical inventory of each grade. This mix has historically featured a projected return and an expected charge off rate roughly in line with the overall platform.[5]

Figure 3: Automated Investing

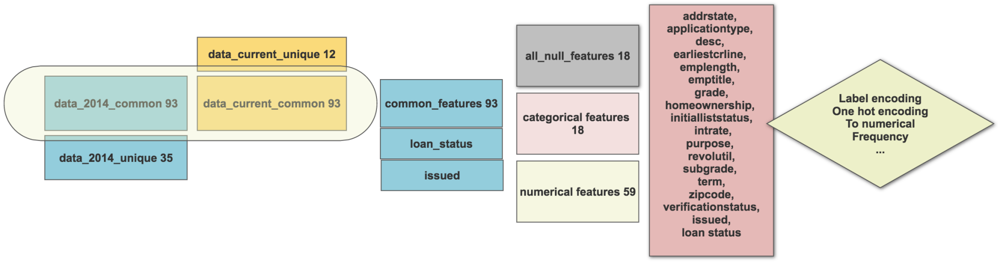
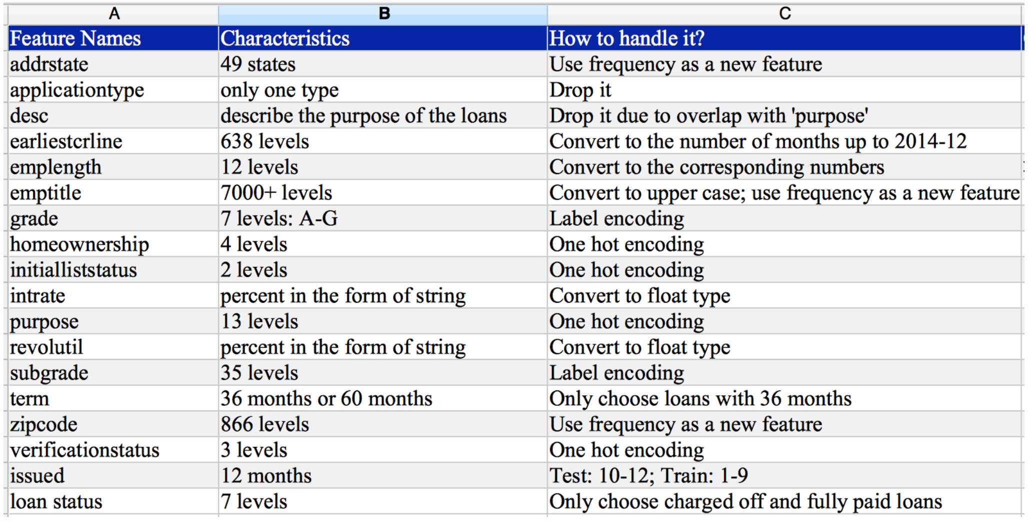


Figure 4: Features

2.2.1 Categorical features

The 18 categorical features are handled in the way showed in Figure 5. We make some further remarks here. For employment length, there are 12 levels. For levels like "n years" where n is from 1 to 9, we convert them to n. For level "<1 year", it is represented by 0 and for level ">10 years", it is converted to be 10. There are also some rows with "NA", it is very likely that they don't have jobs. This increases the probability that the loan will be charged off. So we set the employment length for such rows to be -9999 to distinguish from the other levels. For the feature "zipcode", there are 866 levels. One way is to use the frequency of the 866 levels to represent the levels, as what we did in this project. Another possible way to deal with zipcode is to clustering the areas according to the loan status. It should work well if we can take care of the risk of information leakage. Also for the feature "desc", natural language processing can be considered to extract more information. But since in the current dataset, all rows miss the value of "desc", we just drop this feature.



A	B	C
Feature Names	Characteristics	How to handle it?
addrstate	49 states	Use frequency as a new feature
applicationtype	only one type	Drop it
desc	describe the purpose of the loans	Drop it due to overlap with 'purpose'
earliestcrline	638 levels	Convert to the number of months up to 2014-12
emplength	12 levels	Convert to the corresponding numbers
emptitle	7000+ levels	Convert to upper case; use frequency as a new feature
grade	7 levels: A-G	Label encoding
homeownership	4 levels	One hot encoding
initialliststatus	2 levels	One hot encoding
intrate	percent in the form of string	Convert to float type
purpose	13 levels	One hot encoding
revolutil	percent in the form of string	Convert to float type
subgrade	35 levels	Label encoding
term	36 months or 60 months	Only choose loans with 36 months
zipcode	866 levels	Use frequency as a new feature
verificationstatus	3 levels	One hot encoding
issued	12 months	Test: 10-12; Train: 1-9
loan status	7 levels	Only choose charged off and fully paid loans

Figure 5: Categorical features

2.2.2 Numerical features

There are some numerical features having missing values. But since the XGBoost has the nature of dealing with the missing values, we won't fill the missing values or drop them. The details of the missing values are shown in Figure 6.

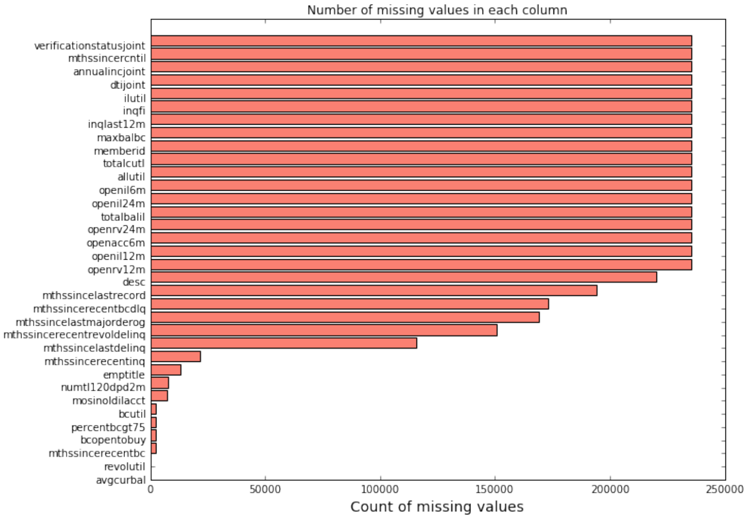


Figure 6: Features with missing values in data of 2014

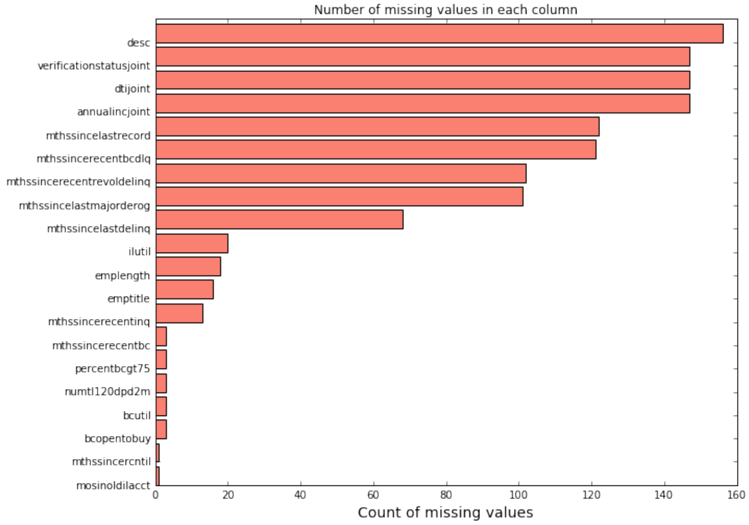


Figure 7: Features with missing values in the current dataset

Some features have outliers, which will be considered in the future work.

2.3 Data Exploration and visualization

In this section, we show some interesting discoveries when we explore the data.

In Figure 8, one can find that nearly 60% of loans are fully paid when we downloaded the data of 2014. There are about 15% loans are charged off. The distribution of the loans changes as time goes on as shown in Figure 9. We marked the increasing trend using grey.

As for the purpose of the loans, one can find in Figure 8 that more than 70% of them are used for debt consolidation and credit cards. Common sense tells us that people who borrow money to consolidate debt or to pay their credit cards are less likely to leave their loans unpaid.

We also explore the relationship between some categorical features with the loan status. For example, in Figure 10, the distribution of the loan status across the features "grade" and "home-ownership" are showed. One can find that loans with grade B and C are more than the other levels. As grade decreases, the percent of charged-off loans increases. People with mortgage are more likely to borrow money from Lending Club. But for people renting, the percent of charged-off is higher than that of the mortgage class.

From Figure 11, we can see that the distributions of loan amount show similar patterns for different loan status. Figure 12 shows that the distribution of the total loan amount in different states. California, Texas and New York rank the first three places. Here we have not considered

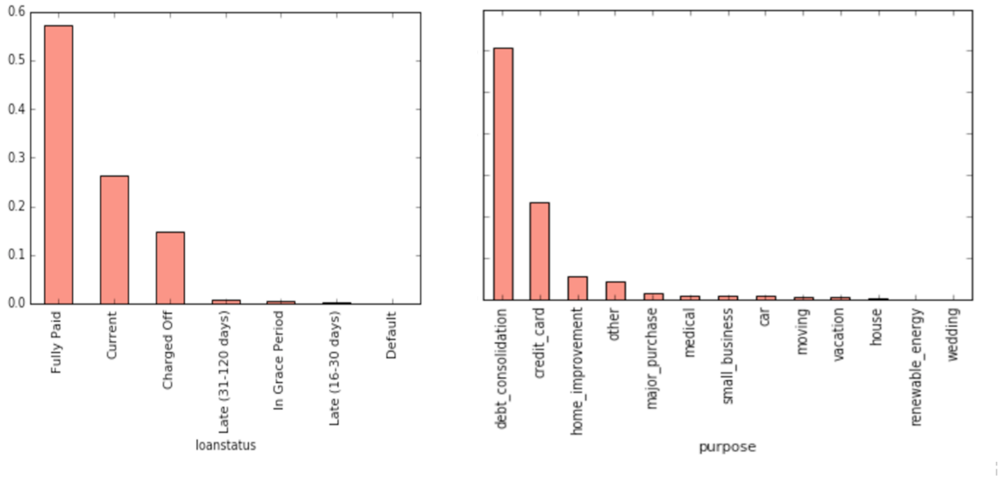
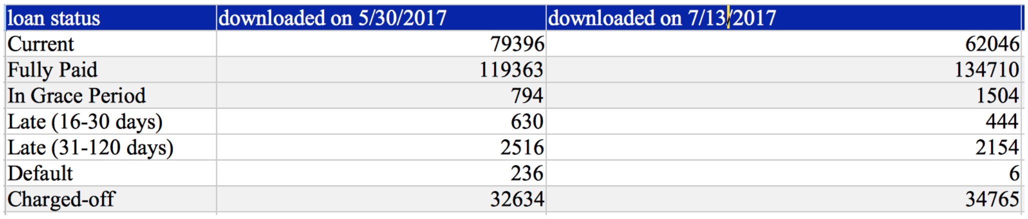


Figure 8: Distributions of loan status and purpose



loan status	downloaded on 5/30/2017	downloaded on 7/13/2017
Current	79396	62046
Fully Paid	119363	134710
In Grace Period	794	1504
Late (16-30 days)	630	444
Late (31-120 days)	2516	2154
Default	236	6
Charged-off	32634	34765

Figure 9: Trend of loan status

the fact that 100 dollars' real values are different across the 49 states.

The interests of the loan from different grades are shown in Figure 13.

We are also interested in the relationship between the amount of loans and the annual incomes. From Figure 14, one can see there is a linear relationship when the income is under \$50000. The highest amount of loans is \$35000. It may be related to the policy of Lending Club. The scatter plot is very dense when the income is under \$200000. They are the main target customers of Lending Club.

There are still a lot of information waiting to be explored. We will not go further for the moment.

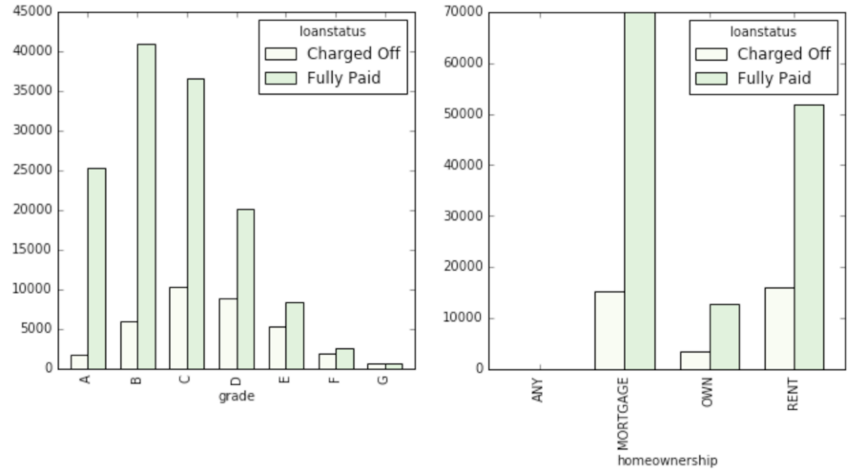


Figure 10: The relationship between grade, homeownership and the loan status

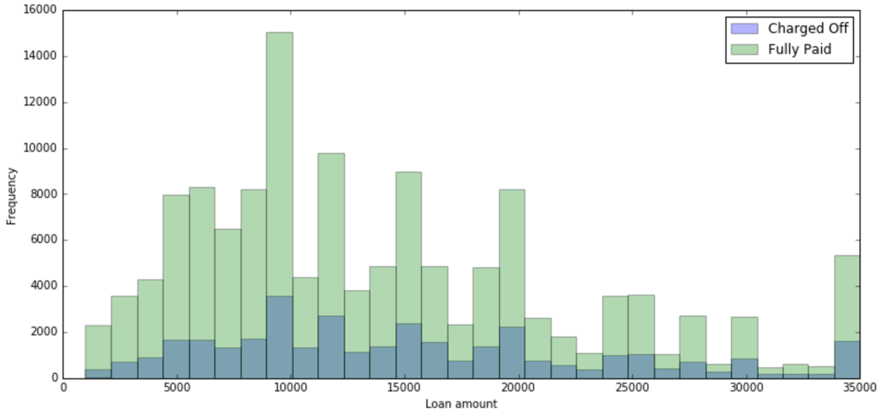


Figure 11: The loan amount distribution and the loan status

2.4 Benchmark

As we discussed before, accuracy is not the only measure to evaluate the performance of the model. So when we train our model, not only the accuracy should be greater than 0.1599 (the percent of the loans charged off among all loans charged off and fully paid is 15.99%), but also the AUC should be larger than 0.5. We will also build an AdaBoosting Tree model and set it to be the benchmark.

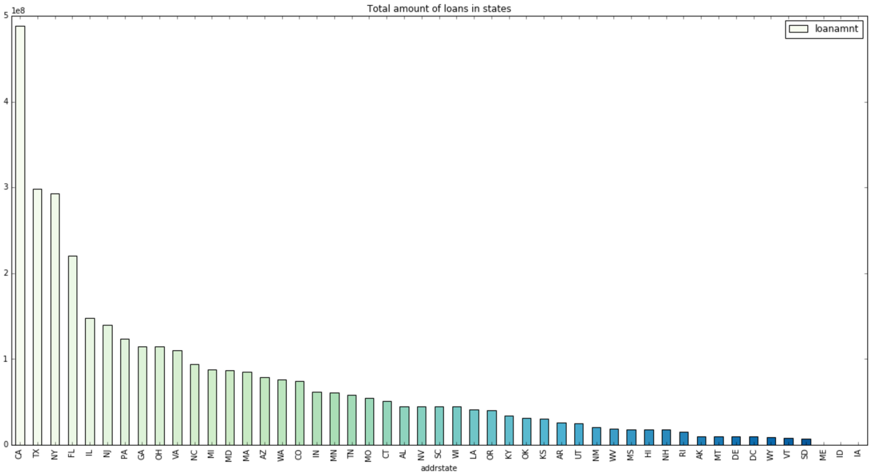


Figure 12: The total loan amount distribution in different states

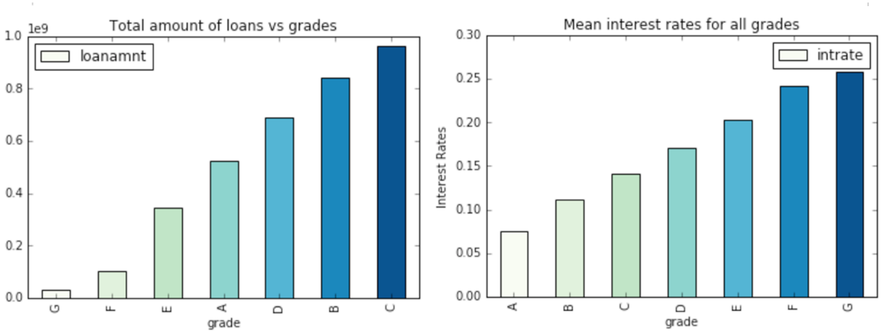


Figure 13: The total amount of loans and mean interest rate vs grades

3 Methodology

3.1 Training and testing datasets

We use the data of loans issued in the first 9 months as the training dataset. It will be used to train the model via cross validation. The data in the last 3 months will be used as the testing dataset. For the current dataset, since we have no labels, there is no way to use it to evaluate the models. It will only be used to make prediction.

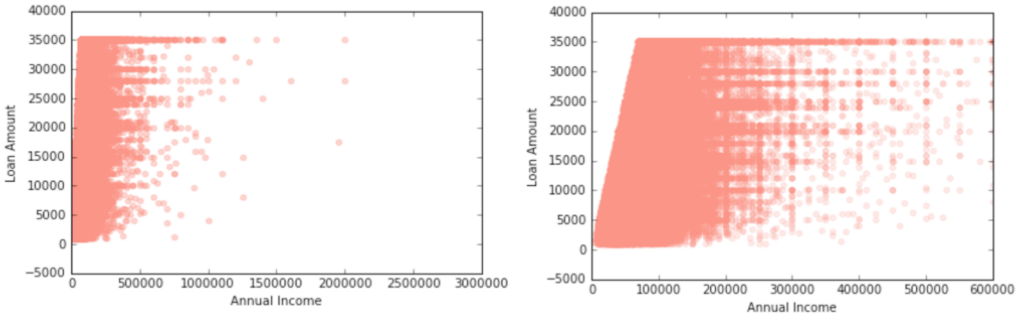


Figure 14: The amount of loans vs incomes

3.2 Models

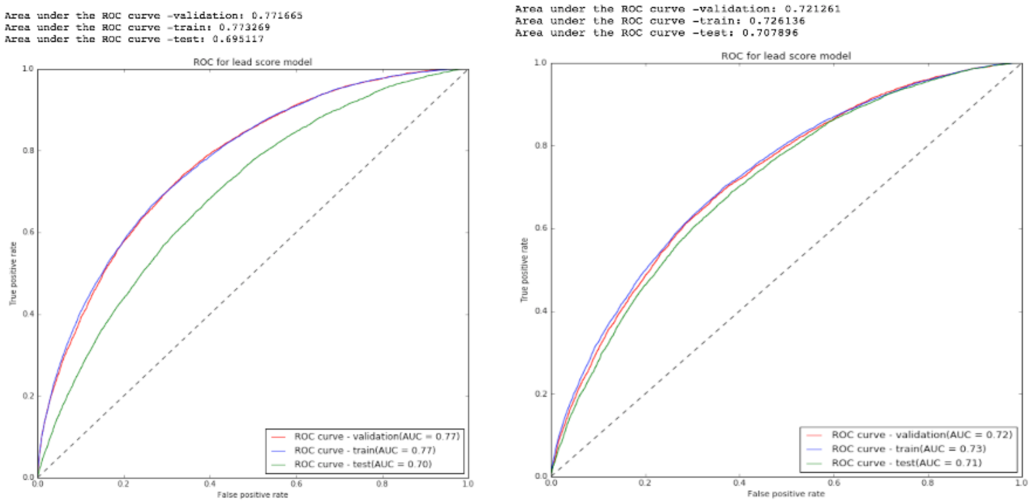
We first choose four features "dti", "annualinc", "totalilhighcreditlimit", "revolutil" to train an Adaboost tree model. Unlike XGBoost algorithm, which can handle the missing values by itself, Adaboost Tree in sklearn can not deal with missing values. So we use the imputer in sklearn to fill the missing values by the most frequency strategy. Without tuning parameters, the accuracy is 0.8528 and the AUC score is 0.395. So it is far from being satisfying. We choose XGBoost models by tuning various parameters manually and automatically.

The following parameters are tuned in sequence: max depth, min child weight, colsample bytree, subsample and gamma. It turns out that the best max depth is 3, the best min child weight is 1, the best colsample bytree is 1, the best subsample is 0.2 and the best gamma is 0.2. With the best parameters obtained manually, the AUC score is 0.69 on the test dataset. With the best parameters from manually tuning in mind, we choose finer grids and use Bayesian Optimization to tune the parameters. The best parameters vary and they are as follows: subsample: 1.0, max depth: 2, eta: 0.1, gamma: 2.0, min child weight: 35, colsample bytree: 0.126.

3.3 Results

In Figure 15, we show the results of the two models. The left one is the XGBoost model with the parameters tuned manually; and the right one is the model with the parameters tuned automatically. The second one is more stable with less variability on the training and testing datasets. AWS is used to shorten the training time.

One advantage of tree models is that they can tell the feature importance. In Figure 16, we can get the most important features corresponding the two models. The following features "dti", "intrate", "annualinc", "avgcurbal", "bcutil", "earliestcrline month" and "mosinoldrevtlop" are important in both models.



```
Area under the ROC curve -validation: 0.771665
Area under the ROC curve -train: 0.773269
Area under the ROC curve -test: 0.695117
```

```
Area under the ROC curve -validation: 0.721261
Area under the ROC curve -train: 0.726136
Area under the ROC curve -test: 0.707896
```

Figure 15: The performance of the XGBoost models

Now let's have a closer look at those important features to understand why they have strong predictive power. The feature "dti" is 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. So it is not a surprise that it is important. The feature "intrate" is the interest rate on the loan, which is positive related to the grade of the loans. From this, we can see that the grades given by Lending Club is relatively reliable. Feature "annualinc" is the self-reported annual income provided by the borrower during registration. It indicates the borrowers' ability of returning the money they borrowed. Feature "avgcurbal" is the average current balance of all accounts. Feature "bcutil" is the ratio of total current balance to high credit/credit limit for all bankcard accounts. They show the financial situations that the borrowers are in. Feature "earliestcrline month" is the month the borrowers' earliest reported credit line was opened up to 2014-12. Feature "mosinoldrevtlop" is Months since oldest revolving account opened. They describe borrowers' historical information. They are also related to the ages and employment lengths of borrowers. So it makes sense that those features are important when considering whether a loan will be charged off or fully paid.

4 Conclusion

4.1 A recap

After performing a exploration data analysis, we trained XGBoosting trees to prediction the status of loans. The best AUC score on the testing dataset is 0.71, which is a big improvement

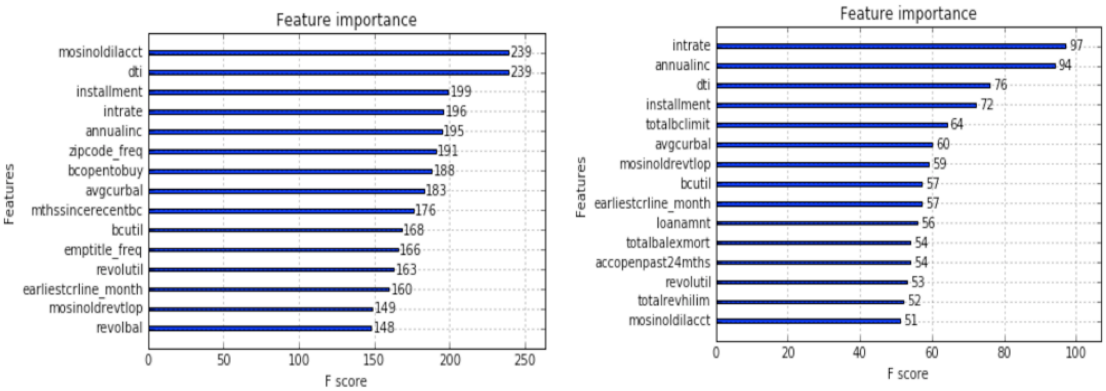


Figure 16: The important features in the XGBoost models

compared to the coarse Adboost model. However, there are still a lot of techniques and insights which can be applied to improve the performance of the model.

4.2 Improvement

From the point of feature engineering, we may work more on categorical features. Firstly, features zipcode, employment title have many levels. This pulls down the performance level of the model. So one possible way is to cluster similar levels. Secondly, there are levels which rarely occur and they have minimal chance of making a real impact on the model prediction. So one may combine such rare levels into one. Thirdly, it is possible that one level is dominant and there is little variance. In this case, this feature may be useless.

From the point of the models, we may stack the XGBoost models from manually tuning and from automatically tuning. As many Kaggle competitions show, this is an effective way to improve the performance of the models. We may also introduce other models different from ensemble trees, and stack them with XGBoost models. The philosophy is that different algorithms captures different aspects of the problems, so when they cooperate, the result may be amazing.

4.3 Reflection

4.3.1 Investment strategy and the default rate

With the probability of default or charged off, how can we choose our investment strategy? LendGuardian working on both Lending Club and Prosper, statistically identifies the loans that are better investments. The article Algorithm Investing Part 1: What is a Secondary Credit Model? gives a good explanation. The essence is in Figure 17. For readers' connivence, I quote it here. "For

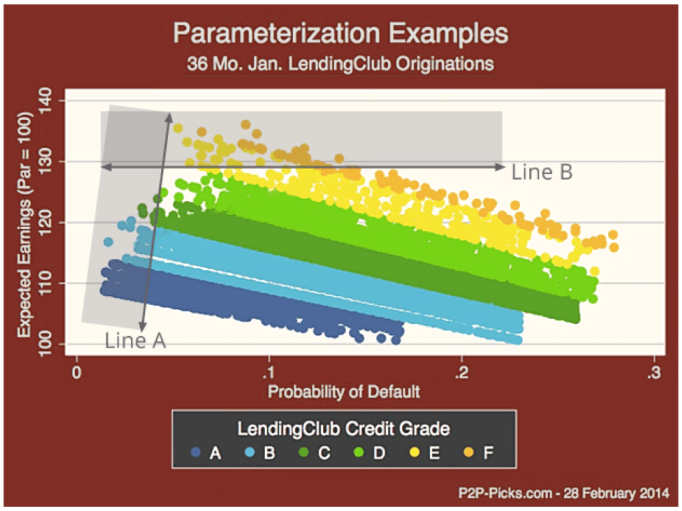


Figure 17: Investment strategy

example, in the above graphic, Bryce has created a scatter plot of loans on Lending Club's platform from January 2014. The vertical axis is ROI, meaning, the mathematical return that each loan is likely to give an investor, calculated by the LendGuardian statistical algorithm. The horizontal axis is the default rate, with the algorithm again having calculated how likely each loan is to default on its payments.

The dark lines are buy-lines. Line A would be the buy-line for an investor who is trying to minimize their risk (while still earning a great return). Someone in retirement might choose to invest above Line A. In contrast, the Line B buy-line is what I personally would use. Line B solely seeks to maximize returns (ignoring risk altogether). Most investors would probably want to tilt that line a bit for a safer investment."

4.3.2 Loans' age and Payment

Knowing the default rate, it is not sufficient for us to decide which loans to invest in. Gauging the ongoing performance of a portfolio requires investors to be able to predict how much a given loan will return before reaching maturity. For example, the status of the loans issued in 2014 are dynamic. In this project, we only download the data in July, 2017. With monitoring the status of the loans more frequently, one can do a better job in the prediction. Luckily, LendingRobot has done a study on this. Readers are strongly suggested to read the article Predicting the Number of Payments in Peer Lending. In short, the risk of default increases sharply over time, especially after a few months, then decreases once a loan has passed half-maturity. The idea is shown in Figure 18, where the curve shows the probability of default over time for loans that will default.

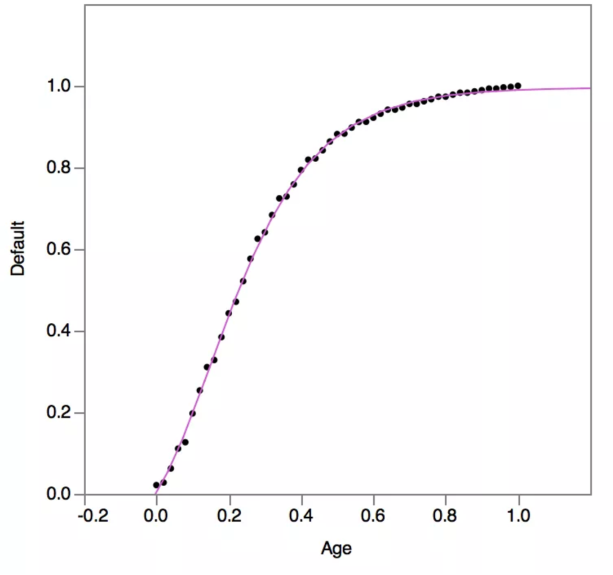


Figure 18: Age vs Default Rate