Predicting Peer-to-Peer Microloan Defaults

Jack Magill
Princeton University
jmagill

Mike Hallee
Princeton University
mdhallee

Abstract

In this paper, we analyze a dataset of approximately 1 million loans made by the peer-to-peer lending company Lending Club between 2007 and 2015. Utilizing 75 financial and demographic features, we begin by building binary classifiers for loan default based on the information that would have been available at the loan application. We found Gaussian Naive Bayes to be the most effective, fitting our goal of having a classifier with high recall at the cost of precision, which most closely resembles the risk averse nature of most small investors.

Our classifier correctly predicted 37% of loan defaults using only data available prior to loan issuance. In addition to simply being a binary classifier, Naive Bayes classifiers allow for class probabilities to be examined, allowing us to see the 'default probability' of every single loan, and compare it to the grades given to it by Lending Club. We find that our classifier closely matches the actual grades.

We also examine our classifier's performance in detecting loan default in low-grade, high-risk loans, finding that we could find a strategy with a lower expected default rate than using Lending Club's default tiers.

1 Motivation

The Lending Club is the largest of several online peer-to-peer lending platforms. In peer-to-peer lending, potential borrowers apply to the Lending Club (or another platform) with their financial and demographic information, and if approved, their loan is added to the holdings of the platform. Loans typically range between $5,000 and $25,000 and are used for purposes such as large purchases, debt consolidation, home-improvement, weddings, etc.. Investors on the platform then chose which borrowers loans– broken up into 'notes" that represent a fraction of the loan amount– that they want to purchase in exchange for fixed payments with interest. Importantly, many borrowers utilize these platforms because their credit scores are too low to qualify for loans from traditional financial institutions [5]. This underscores the importance of reliably identifying the risk factors for default.

Why is there value to our investigation? Investors in the Lending Club (and presumably other platforms) can chose which notes they invest in, either by a specific mix of loan grades, or more interestingly: by manually selecting individual notes to add to their portfolio based on their specific borrower information [2] . The Lending Club uses their own methodology to assign grades to notes ranging from lowest risk, A1 with 6%-symbol interest rates, to the highest risk, E5 which can exceed 30%-symbol interest rates [2]. Crucially, if utilizing alternative methodologies can identify specific notes that are expected to have lower risk than the Lending Club has graded them, then an investor can build a portfolio that maximizes returns (interest rate) while minimizing risk and, by extension, earning above the average return.

2 Prior Work

Professors Michael Shaw and James Gentry of the University of Illinois at Urbana-Champaign published a well-cited early paper in 1988 on the use of inductive learning to predict the probability

of loan defaults [4]. Their approach was centered around commercial banking of companies rather than individuals. They used a "similarity-based" model that learned classes of borrowers such as "A firm whose asset exceeds $1 million, debt is less than $250,000, and whose annual growth rate is more than 10%" and made decisions based on a borrowers closest class. The achieved an impressive 86.2%-symbol accuracy [4] of predicting loan defaults in this simplistic method, suggesting the high potential of developing latent loan classes. Most notably, the authors concluded that frequently the qualitative information is of greater value in the lending decision than the financial statement analysis. [4] They also underscored our idea that developing improved models increased competitive advantage and investment payouts.

A more recent 2016 paper published in the IEEE International Conference on Knowledge Engineering and Applications analyzed an earlier version of the dataset we will be working with. The researchers focused on three tree-based classifier models: Decision Tree, Random Forest, and Bagging [5]. They achieved similar results with all three classifiers: approximately 96%-symbol precision at the expense of accuracy in the low eighties. To access their findings, we notice they failed to explore probabilistic models like Naive Bayes or other methods like Support Vector Machine, which we will explore[5]. And more importantly, they were not motivated with identifying opportunities for strategic investment. This means they did not prioritize precisely eliminating the likely defaults from the high-risk, high-interest-rate category of loans, which would provide the highest potential for maximized investment returns. We will focus on extending classification to incorporate the maximized expected value of identifying non-defaulting loans from the high-risk, high-interest-rate categories.

3 Dataset Overview

Our dataset consists of metadata for approximately 2.3 million loans issued by Lending Club (LC) between 2007 and 2018. Whilst available directly from LC, the data was found nicely packaged in a Kaggle repository published by Wendy Kan [1]. The raw file consists of 146 variables, which can be grouped into four main 'groups'.

Borrower Information

This category consists of the personal information of the borrower, including income, ZIP code/state, occupation, and purpose of loan. In addition, there are a large number of numerical values that would be found on a credit report, such as number of credit cards, utilization rates, history of delinquent accounts, etc.

Loan Information

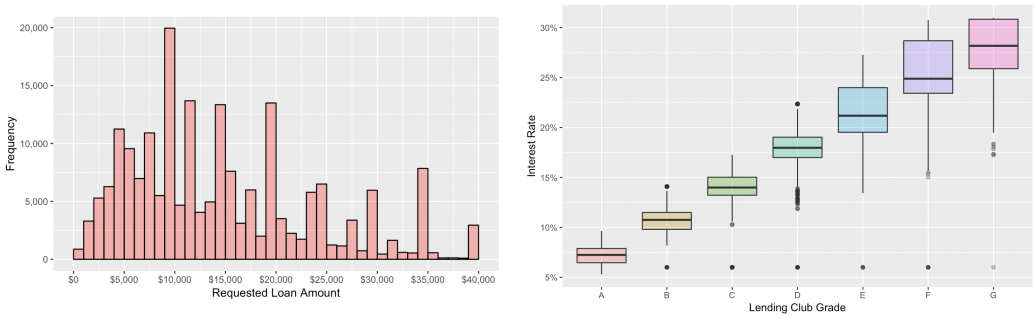
This category consists of the information of the loan itself as a financial instrument, such as total amount, interest rate, monthly payment, and term.

Lending Club Information

This consists of the grade and sub-grade that lending club assigns to the loan. They use a system that assigns each loan a grade between A1 (lowest risk, lowest return) to E5 (highest risk, highest return), while some older loans include F and G ratings. To give context, A grade loans average 4.83% return whilst E grade loans average 6.37% return [2].

Outcome Information

The prior three groups consist of information that is known at the time of loan issuance. The outcome information is updated during the repayment period of the loan. It includes current status (closely define later), current amount of repayment, number of late payments, and any settlement or hardship conditions on the loan.



(a) Loan Amount Distribution

(b) Interest Rate by LC Grade

Figure 1: Summary of Loan Information

4 Preprocessing and Imputation

4.1 Creating Binary Class Variable

We want to reduce the problem of default prediction to a binary classification problem. The data field 'Loan Status' consists of more than just a 'Default' and 'Normal' status, and includes late payment, grace period, paid off, and charged off (where no more payments are expected). Therefore, we need to group these different statuses into two groups.

The initial approach was to simply try to identify the 'Default' loans; however, they only represent $< .1\%$ of all loans, and having such a massive class size mismatch would be problematic, therefore we expand our definition as describe in Table 1, giving us a class balance of approximately 13% to 87%.

Status	Occurrences	Class Label
Default	31	1
Charged Off	261,655	
Late 31-120 Days	21,897	
Late 16-30 Days	3,737	
Grace Period	8,952	0
Current	919,695	
Fully Paid	1,041,952	

Table 1: Definition of Binary Outcome

4.2 Dropping Data

Our goal is to be able to try and predict future loan defaults *at the time of issuance*, and because of this there are several components of the dataset that need to be removed before passing it to a classifier.

The first is Lending Club's proprietary grade and sub-grades, as our goal is to do something similar. We also need to remove any of the variables that are continually updated as the loan is being repaid, such as any settlement or hardship plans as well as the current rate of repayment. None of this information is available to an investor when loan funds are initially being raised, so we don't want to consider them as predictive factors (given that they are related to repayment they should be really effective at predicting default).

4.3 Missing Data and Imputation

We decided to remove any variable where less than 75% of the data was non-null. This resulted in some borrower fields being removed, such as member and loan ID (which were all missing), and around 20% of the credit report numerical data.

For the remaining numerical variables, and missing data was replaced with the column median. In addition, several numerical variables consisted of negative values when it logically shouldn't (income for example). We also replaced these occurrences the median.

4.4 Categorical Variables

Whilst most of the data is numerical, there are several categorical variables, such as 'home ownership' and 'loan purpose'. For these, we used one-hot encoding with an added variable for 'missing'.

4.5 Final Dataset

Once considering the removal of missing/unnecessary fields and the addition of more variables due to the encoding of categorical data, we were left with 2.3 million total observations of 99 variables.

5 Classification Methods

In Section 4.1 we defined our version of the 'Default' variable that turns loan status, a categorical variable, into a binary variable. Thus, we are seeking to solve a supervised binary classification problem, with the goal of being able to predict which loans are likely to default or have late payments at some point over their lifespan.

5.1 List of Classifiers

In order to build our classifiers, we used the implementations published by scikit-learn [3].

1. *Logistic Regression Classifier* (LR)
2. *Bernoulli Naive Bayes* (BNB)
3. *Gaussian Naive Bayes* (GNB)
4. *Logistic Loss SGD* (LGD)
5. *Perceptron Loss LGD* (PGD)
6. *AdaBoost Classifier* (AB)
7. *Decision Tree Classifier* (DT)

There were two classifiers that we originally included: (1) Support Vector Machine and (2) Random Forest classifier. However, once expanding our training dataset from a small subset of the data (approximately 5% or 100,000 samples) to the entire set (2.3 million samples), they became prohibitively long to compute, so we discarded them from our final set.

In order to evaluate each classifier, we split our dataset, allowing 80% for training and the remaining 20% for testing.

5.2 Tuning Hyperparameters

In order to select the best parameters for each of the classifiers, we decided to use a grid search method as implemented via the scikit-learn *GridSearchCV* library [3]. Whilst doing so requires us to train our model multiple times using the different combinations of parameters, it allows us to find a model that could perform better than the default settings.

The parameters we search over included the coefficients for the regularizing terms of loss functions (alpha/C) as well as depth and feature count for the AdaBoost and DecisionTree classifiers.

Grid searching requires a 'scoring' function for each parameter combination, which defaults to using the mean accuracy over testing data. However, for the reason explained in Section 6, we instead chose to use recall (true positive rate) in order to select the best combination.

6 Classifier Evaluation

6.1 Classification Metrics

We decided to consider several evaluation metrics for classifiers, all of which are functions of the four true outcomes (True/False + Postive/Negative).

$$\text{Precision: } \frac{TP}{TP+FP} \qquad \text{Recall: } \frac{TP}{TP+FN} \tag{1}$$

$$\text{Specificity: } \frac{TN}{TN+FP} \qquad \text{F1 Score: } 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2}$$

We also want to consider accuracy, but due to the class imbalance, we choose to use weighted accuracy which takes the mean of the accuracy predicting each class weighted by their prevalence in the data set.

6.2 Importance of Recall

Unlike our first assignment (sentiment classification), loan default prediction is a problem in which we value false negative and false positives differently. A false positive would be predicting a loan to default when in fact it would fully pay out. A false negative would be predicting a loan to fully pay out when in fact it would default.

Individual investors on this platform only need to select a tiny fraction of available loans, they also are more likely to be risk averse than truly risk neutral or risk seeking. Because of this, we believe that the negative cost of a false negative is substantially higher than that of a false positive. Therefore, we want to minimize the number of false negatives, which maximizes the true positive rate, or recall. Therefore, when picking the 'best' of our classifiers, we want to find a really high rate of recall.

7 Results

We trained and tested our classifiers using a random 80/20 train/test split (using the same split for each classifier). Several metrics were calculated for each classifier, and reported in Table 2. ROC and PR represent the areas-under-the-curve (AUC) of the Receiver Operator Characteristic and Precision-Recall curves respectively. The classifiers are sorted by recall and as we can see, the Gaussian Naive Bayes outperforms the competition by a significant margin. Even when grid searching over parameters and selecting the one with best recall, 4 of the 7 classifiers have a recall of less than 10%, including a baseline logistic regression which failed to predict a single default.

Classifier	ROC	PR	Precision	Recall	F1
GNB	0.650	0.217	0.234	0.367	0.285
LGD	0.572	0.155	0.166	0.276	0.208
DT	0.539	0.142	0.191	0.210	0.200
BNB	0.650	0.219	0.365	0.046	0.082
PGD	0.561	0.151	0.184	0.025	0.045
AB	0.721	0.272	0.520	0.007	0.014
LR	0.612	0.176	0.000	0.000	0.000

Table 2: Results

We also chose to examine the ROC and Precision-Recall curves for the classifiers, which can be seen in Figure 2. We can see that while the AdaBoost classifier has a substantially higher AUC for both curves (usually an indication of a good, robust, classifier), given our unequal view of false positives and negatives, paired with a class prevalence imbalance, it doesn't perform well under our setting.

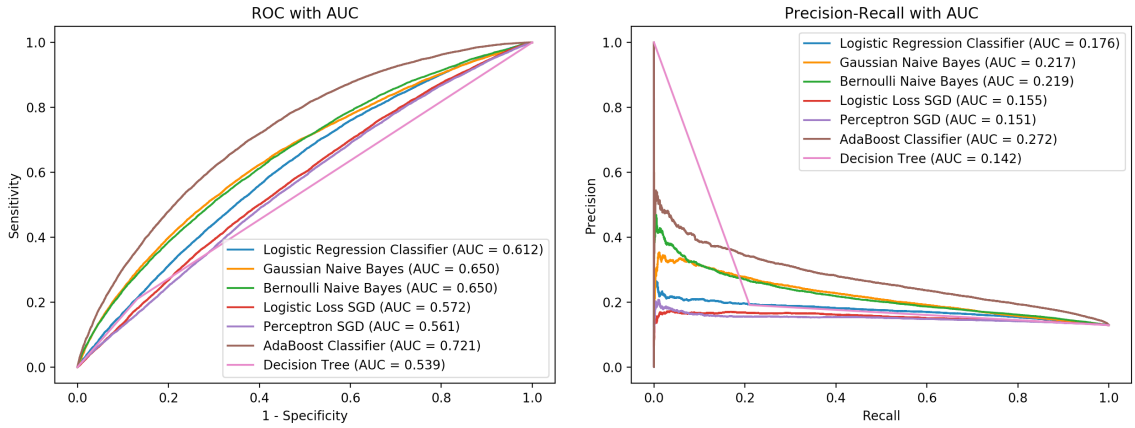


Figure 2: ROC and PR Curves

8 Applying Our Results

Although our actual classification problem is predicting loan default as a binary outcome, in most financial applications, default is considered a probabilistic measure used to define the risk of an investment or asset. Thankfully, our most effective classifier, Gaussian Naive Bayes, is capable of generating probabilistic predictions for each label. We can use this measure is a proxy for the probability of default.

8.1 Comparison to Lending Club Grades

Our original dataset included Lending Club's 'grade' of each loan (we removed this as it was already basically doing what we wanted to do), with each grade representing a higher risk loan. Figure 3 shows all 2.3 million loans in the data set, grouped by the grade given to them by Lending Club. Each is a box plot showing the distribution of the default probability given by our Naive Bayes model.

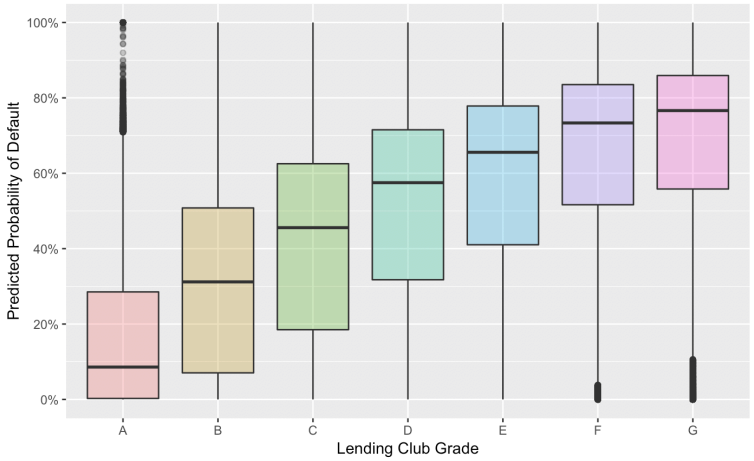


Figure 3: Predicted Probability of Default Grouped by LC Grade

While each grade has values across the entire range of default probabilities (0 to 1), but the median and 25th and 75th percentiles both show a monotonically increase risk of default as the grade worsens. This suggests that Lending Club's risk assessment is similar to ours (at least in a relative manner).

8.2 Actual Default Rates

We can also look at other features for each grade on loan. Table 3 shows the the actual occurrence of defaults on each grade of loan (note that our definition of default includes all late loans). We can see that for every grade of loan our model tends to over-predict defaults by a factor of between 2x and 3x.

Grade	# Loans	% Total	Actual Default Rate	Predicted Default Rate	Median Default Prob.
A	433,027	19.1%	3.54%	16.15%	8.59%
B	663,557	29.4%	8.55%	31.02%	31.11%
C	650,053	28.8%	14.12%	41.11%	45.56%
D	324,424	14.4%	20.17%	50.27%	57.50%
E	135,639	6.0%	28.08%	56.62%	65.56%
F	41,800	1.8%	36.33%	63.21%	73.34%
G	12,168	0.5%	39.86%	65.63%	76.62%

Table 3: Actual Loan Outcomes vs. Predictions

This is not surprising given that we emphasized recall as the most important metric in our model selection. Whilst Lending Club is motivated to create a system of grades with steadily increasing rates of risk (the consistent growth in actual default rate shows that they did this well), our goal is to over-predict default if necessary in order to give us the best ability to select a small set of loans that will be very unlikely to default.

8.3 Predicting High-Risk Loans

One of the most useful applications of having a superior risk model would be to take advantage of the high returns of low-graded loan (D,E,F,G) whilst taking a below-expected level of risk. When we filter out the original loan dataset to only include these low-grade loans, we are left with 514,031 observations.

Of these, 123,588 (24.1% of total) defaulted, and our model correctly identified 87,099 of them. This is a recall of 70.5%. Of the loans we predicted to *not default*, 185,872 in total, only 36,489 of them actually defaulted. This is a false negative rate of just 19.6%.

Therefore, consider a high risk investment strategy were you invest in all of the low-grade loans (or randomly), there is an expectation of 24.1% to default at some point in their lifespan. If you were to instead invest (either all or randomly) in loans that our classifier didn't predict to default, there is an expectation of just 19.6% default, almost 5% less than if you were to invest using Lending Club's grade system.

8.4 Driving Variables

One of the measures available to investors when making decisions is the interest rate of the loan, and as a measure of the return they will receive, is important in their decision making for investing. However, one of our worries was that because Lending Club heavily considers their estimation of credit-worthiness and default risk when assigning interest rates, our model would heavily rely on it as an estimator.

Figure 4 is a 2D histogram (with 1% bin widths on both axes), with each bin colored using a log scale. Our default prediction is on the x-axis, whilst the interest rate of the loan is on the y-axis.

By following the center of the brightest band across the graph, we see that the default probability isn't extremely correlated with the interest rate in the areas between 5% and 15% interest rate and 0% and 50% default probability. However, once interest rates rise above 15%, the most dense narrows and default probabilities are high. The lack of an extreme relationship is good, as it shows that our model's predictive power isn't heavily reliant upon another composite metric reflecting a 3rd party's expectation of default risk.

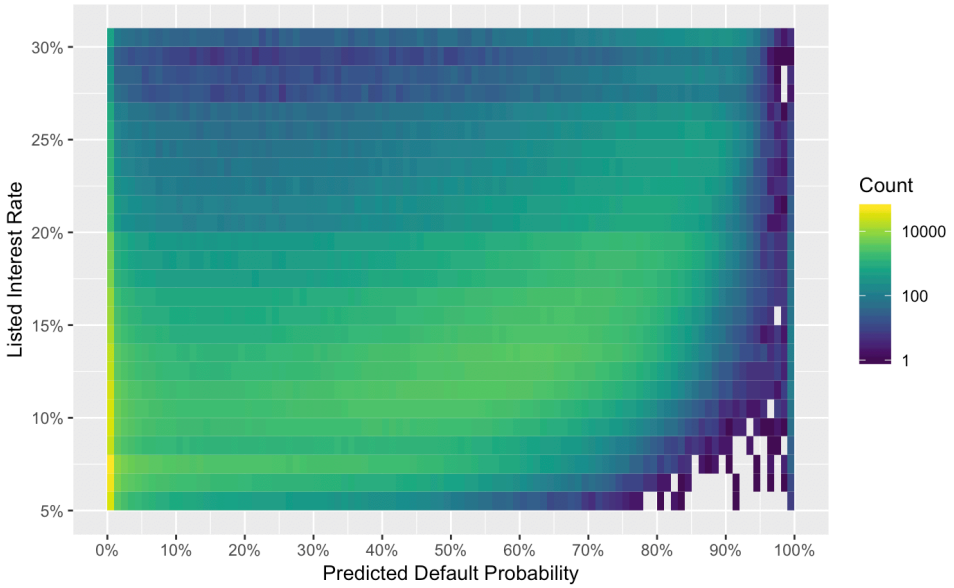


Figure 4: Predicted Probability vs. Listed Interest Rate

9 Discussion

This precursory analysis creates a solid platform to develop some sort of investment framework. The investigations we performed in Chapter 8 rely upon building a binary classifier that allows for probabilistic class estimates (and thus probabilities of default). Section 8.3 shows that just a simple random selection method (when paired with our classifier's distinctions) could outperform a simple investment in Lending Club's grades. Any style of investment portfolio (combination of expected return and the variance of that return) could be generated based on these estimates, whether it be in 20%+ high risk loans are extremely low risk 5% loans.

Lending Club provides tiers of loans based on perceived risk, and has the interest rates adjusted accordingly. If an investor was able to discriminate between all loans of a single grade, they could achieve above average performance and excess returns.

If we were to continue working on this project, I think that more effort could be put towards examining the 50+ numeric credit-related variables, and if missing data should be imputed in a different manner for each. Similarly, given that unlike the default implementations of most classifiers (which use accuracy as a proxy for a 'score'), we could try and make our own scoring function with custom penalties for false positives and false negatives that depend upon the risk averseness of an investor. Having a really robust scoring function would allow us to grid search over a larger range of parameters and more finely tune our model.

References

[1] Wendy Kan. Lending club loan data. `https://www.kaggle.com/wendykan/lending-club-loan-data`, 2019.

[2] Lending Club. Note interest rates and fees. `https://www.lendingclub.com/investing/investor-education/interest-rates-and-fees`. Accessed: 2019-04-23.

[3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[4] Michael J. Shaw and James A. Gentry. Using an expert system with inductive learning to evaluate business loans. *Financial Management*, 17(3):45–56, 1988.

[5] Vinod Kumar L, Natarajan S, Keerthana S, Chinmayi K M, and Lakshmi N. Credit risk analysis in peer-to-peer lending system. In *2016 IEEE International Conference on Knowledge Engineering and Applications (ICKEA)*, pages 193–196, Sep. 2016.