Loan Default Prediction in LendingClub

A Master's Project
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
Mathematics

---

by
Shirong Zhao
Aug 2020

Accepted by:
Dr. Xin Liu, Committee Chair
Dr. Yuyuan Ouyang
Dr. Andrew Brown

Abstract

This project aims to analyze the credit risk of the loans in the peer-to-peer platform LendingClub. Various machine learning techniques are applied to predict the probability that a requested loan will be charged off over 2007–2017Q1. If only considering the accuracy rate, we would pick the logistic regression classifier as our primary model because it achieves a relative high accuracy rate in both the training and test data, and is cheaper in computation and easier to interpret. Moreover, it is found that the most important features for predicting credit risk are interest rate, term, annual income, FICO score, debt-to-income ratio, total credit revolving balance, and loan amount. However, if we instead focus on the area under the receiver operating characteristic curve (AUCROC), the XGBoost performs the best in both the training and test data.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

As an alternative tool to the traditional consumer loans made by the banks, peer-to-peer lending platforms are becoming much more popular in recent years. Through platforms such as LendingClub, Upstart, and so on, individual investors now have access to lending money directly to individual borrowers without banks as intermediaries.

The process goes as follows. Borrowers first submit loan applications to the platforms that will perform an evaluation of the applications and then decide whether to list the loans on their websites (The websites are mainly designed for investors). Individual investors can search and browse the loan listings on the platform and select loans to invest based on the information of the borrowers, such as the amount of loan, loan grade, interest rate, loan purpose, and so on. Investors make money from the interest on the invested loans and the platform makes money by charging borrowers an origination fee and investors a service fee.

These loans are not completely safe as they involve substantial risk of default. Data from LendingClub show that there are 18.01% of loans made from 2007 to 2017Q1 already charged off. Due to asymmetric information, it requires more effort from platforms and investors to identify and determine "good" borrowers from a pool of unknown users than traditional banks. This is very different from banks since banks in some sense "know" much more about their borrowers. That is why the loans on these platforms are much riskier than those made by traditional banks and hence the interest rate on the loans from the peer-to-peer platforms is also much higher. Therefore, investors usually try to diversify their portfolio by investing only a small amount in each loan by exploring imperfect correlations among the loans.

It is crucial for the investors to decide whether to invest in the loans. On the other hand, it is important to keep peer-to-peer lending healthy so that small businesses can grow healthily with these loans. This motivates us to build currently popular machine learning techniques to predict whether a loan will default or not. Specifically, we want to quantify the default risk of the loans from LendingClub since the information about the loans from LendingClub is available in public.

There are several literature using machine learning techniques to predict the probability of default using loan data information from LendingClub. Li and Han (2015) build machine learning models capable of predicting loan defaults on LendingClub's 2012–2015 data set having $745,529$ credit records. The classifiers used are logistic regression, neutral network and random forest. For each model, they achieve weighted average of 0.89 for both precision and recall. However, they use $1,097$ features and do not randomly split the data. Chang, oong Kim and Kondo (2015) carry out logistic regression, naive Bayes, and support vector machine machine learning techniques over the data from 2007 to 2015. They find that Naive Bayes with Gaussian performs the best with default prediction (80.1% sensitivity). Vinod, Natarajan, Keerthana, Chinmayi and Lakshmi (2016) apply decision tree, random forest, and bagging methods over the data from 2013 to 2015 having $656,724$ credit records. They find that random forest is better in identifying the default, while decision tree is more powerful in finding the good credits.

This project applies ten different models to the loan data set from LendingClub over 2007–2017Q1. For more details about those models, please refer to Book "The Elements of Statistical Learnin" (Hastie, Tibshirani and Friedman (2008)). We contribute to the literature by using the largest ever data set from LendingClub and also applying as many machine learning models as possible. We find that if only considering the accuracy rate, we would pick the logistic regression classifier as our primary model because it achieves a relative high accuracy rate in both the training and test data, and is cheaper in computation and easier to interpret. Moreover, it is found that the most important features for predicting credit risk are interest rate, term, annual income, FICO score, debt-to-income ratio, total credit revolving balance, and loan amount. However, if we instead focus on the area under the receiver operating characteristic curve (AUCROC), the XGBoost performs the best in both the training and test data.

In the next section, the data and features are covered. Our machine learning classification methods are briefly discussed in Section 3. Classification results are presented in Section 4. Conclusions and discussions are given in Section 5.

Chapter 2

Data and Features

2.1 Data Overview

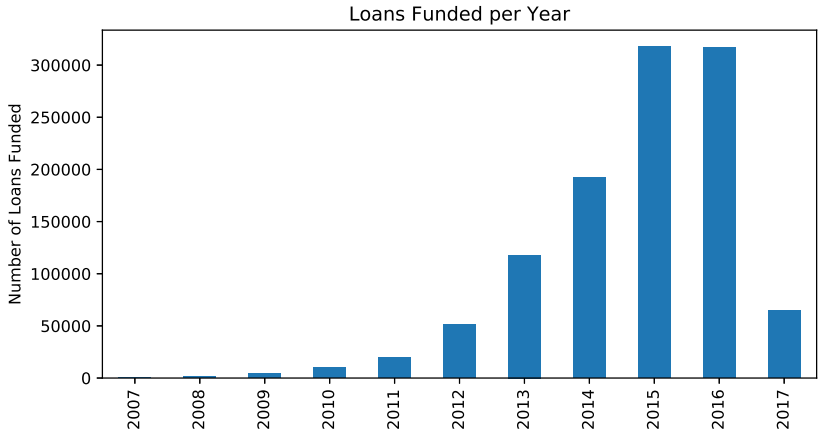
We use public data set published by LendingClub.[1] Those loans were first initiated starting from 2007. Since the loans in LendingClub are either in 36-month or 60-month terms, we focus on the loans issued over 2007–2017Q1 so that the statuses of most of the loans (especially 36-month loans) are known by now. Initially, we have a data set of size $1,418,645$ and 150 features. After filtering out the loans whose statuses are not yet final, such as "Current", "Late", and "In Grace Period", we have a data set of size $1,339,388$ with $1,089,887$ fully paid loans and $249,501$ charged-off loans. We treat "Charged Off" loans as positive labels and "Paid Off" loans as negative labels.

2.1.1 Number of Loans

Figure 2.1 shows the total number of loans funded by each year. As we can see from this table, the total number of funded loans continuously increases over time (Except in 2017, in which only the data in the first quarter are included).

[1] For details, see https://www.lendingclub.com/info/statistics.action.

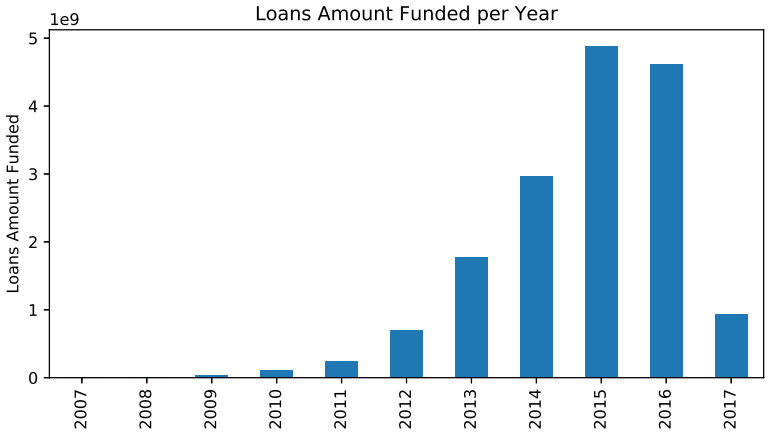
Figure 2.1: Number of Loans By Year



2.1.2 Loan Amount

Figure 2.2 shows the total dollar amount of loans funded by each year. As we can see from this table, the total funded dollar amount continuously increases from 2007 to 2015 and then decreases a little bit from 2015 to 2016.

Figure 2.2: Loan Amount By Year



2.1.3 Feature Processing

First, we check the features with missing values. If one feature has more than 50% missing values, then this feature is dropped to have a cleaner data set. Using this criterion, 58 features are dropped. All the remaining features have less than 13% missing values. Second, since the goal of

this project is to predict whether a loan will be paid off before lending money, we need to remove the features that were unavailable before lending money, such as "acc_open_past_24mths"(number of trades opened in past 24 months) and "last_fico_range_high"(the upper boundary range the borrower's last FICO pulled belongs to). At the same time, we also remove the irrelevant features, such as "url"(URL for the LC page with listing data), and redundant features, such as "funded_amnt", which is the same as "loan_amnt". In total, another 58 features are dropped. Now we have 34 remaining features. The features that we use are listed in Table 2.1. We then check the remaining features one by one, especially their relationship with "Charged Off".

Table 2.1: Variables Definitions

*	*
Features	**Definitions**
addr_state	The state provided by the borrower in the loan application
annual_inc	The self-reported annual income provided by the borrower during registration.
application_type	Indicates whether the loan is an individual application or a joint application with two co-borrowers
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.
emp_title	The job title supplied by the Borrower when applying for the loan.*
fico_range_high	The upper boundary range the borrower's FICO at loan origination belongs to.
fico_range_low	The lower boundary range the borrower's FICO at loan origination belongs to.
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
id	A unique LC assigned ID for the loan listing.
initial_list_status	The initial listing status of the loan. Possible values are – W, F
*	This goes at the bottom *

Table 2.1: (Continued)

*	*
Variables	**Definitions**
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
loan_amnt	The listed amount of the loan applied for by the borrower. If at some point in time, the credit department reduces the loan amount, then it will be reflected in this value.
loan_status	Current status of the loan
mo_sin_old_il_acct	Months since oldest bank installment account opened
mo_sin_old_rev_tl_op	Months since oldest revolving account opened
mort_acc	Number of mortgage 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
pub_rec	Number of derogatory public records
pub_rec_bankruptcies	Number of public record bankruptcies
purpose	A category provided by the borrower for the loan request.
revol_bal	Total credit revolving balance
revol_util	Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
sub_grade	LC assigned loan subgrade
term	The number of payments on the loan. Values are in months and can be either 36 or 60.
title	The loan title provided by the borrower
total_acc	The total number of credit lines currently in the borrower's credit file
verification_status	Indicates if income was verified by LC, not verified, or if the income source was verified
*	This goes at the bottom *

We also remove the outliers. If one observation's value for one feature is smaller than $Q1 - 3 \times IQR$ or larger than $Q3 + 3 \times IQR$, then this observation is defined as an outlier, where $Q1$ is the first quartile, $Q3$ is the third quartile, and IQR is the interquartile range (i.e., $IQR = Q3 - Q1$). After the outliers are removed, we have a data set of size $1,096,824$ with 18.01% "Charged Off" and 81.99% "Paid Off".

For features with missing values, they are imputed with the median value for each feature. Categorical features, such as "subgrade" and "state", are replaced with their one-hot representations. Normalization is then performed at the end on all features so they have zero mean and one standard deviation.

2.1.4 Label Definition

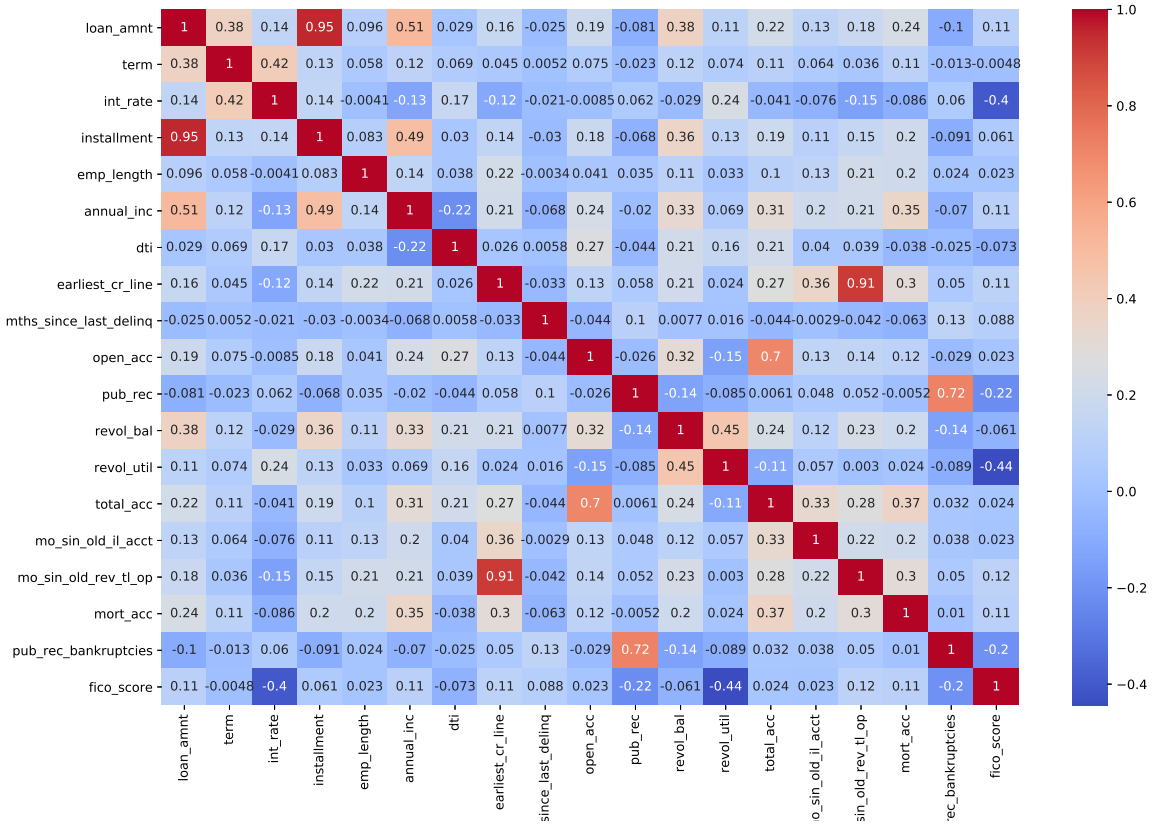
For the target variable, "Charged Off" is assigned label 1 and "Fully Paid" is assigned label 0.

2.2 Exploratory Data Analysis

2.2.1 Feature Correlations

We then check the correlations among the selected numerical features. Figure 2.3 shows the correlation heat map. Notice that some features are highly correlated, such as installment and loan_amnt, pub_rec and pub_rec_bankruptcies, total_acc and open_acc, mo_sin_old_rev_tl_op and earliest_cr_line. We then remove installment, mo_sin_old_rev_tl_op,total_acc, pub_rec_bankruptcies from our features to avoid collinearity.

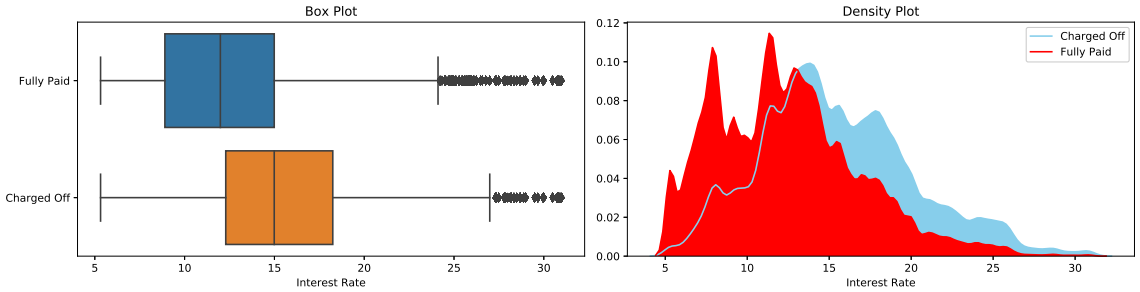
Figure 2.3: Feature Correlation Heatmap



2.2.2 Interest Rate

Figure 2.4 shows the interest rate by the loan status. As we can see from this table, charged-off loans have a higher mean interest rate than paid off loans. Moreover, the distribution of interest rate for the charged-off loans lies on the right side of the distribution for the paid off loans.

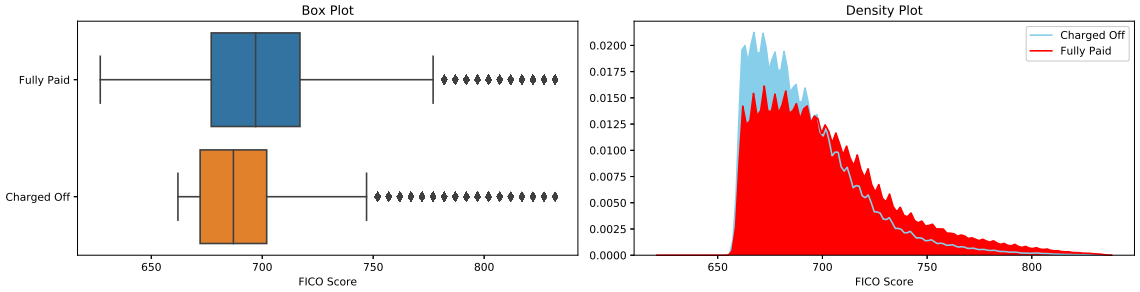
Figure 2.4: Interest Rate By Loan Status



2.2.3 FICO Score

Figure 2.5 shows the FICO score by the loan status. It is clear that the borrowers that cannot pay the loans tend to have a lower mean FICO score than the borrowers who paid.

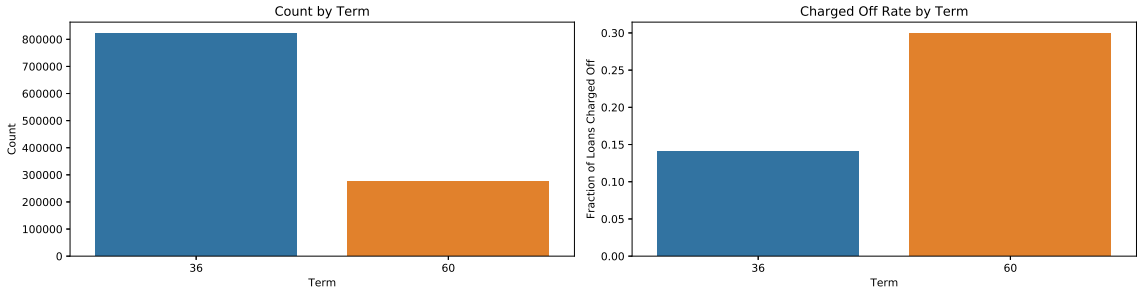
Figure 2.5: FICO Score By Loan Status



2.2.4 Term

Figure 2.6 shows the term by the loan status. The counts for loans with 36 months term are larger than that for loans with 60 months term. However, loans with 60 months have a higher charged off rate.

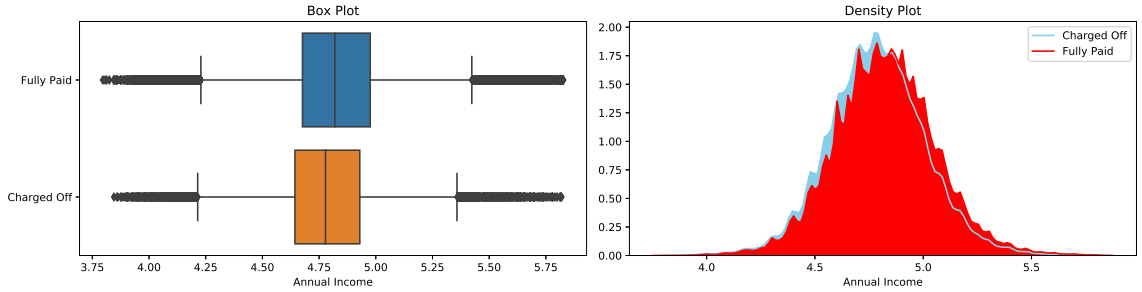
Figure 2.6: Term By Loan Status



2.2.5 Annual Income

Figure 2.7 shows the annual income by the loan status. The borrowers who finally default on the loans tend to have lower annual income than those who finally pay off the loans.

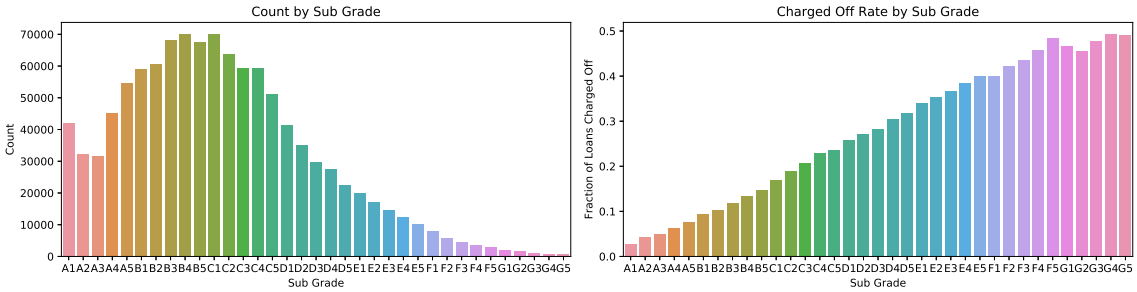
Figure 2.7: Annual Income By Loan Status



2.2.6 Sub Grade

Figure 2.8 shows the sub grade by the loan status. It shows that the loans with better sub grades tend to have lower charged off rate and smaller counts. The loans with middle sub grades have the highest counts.

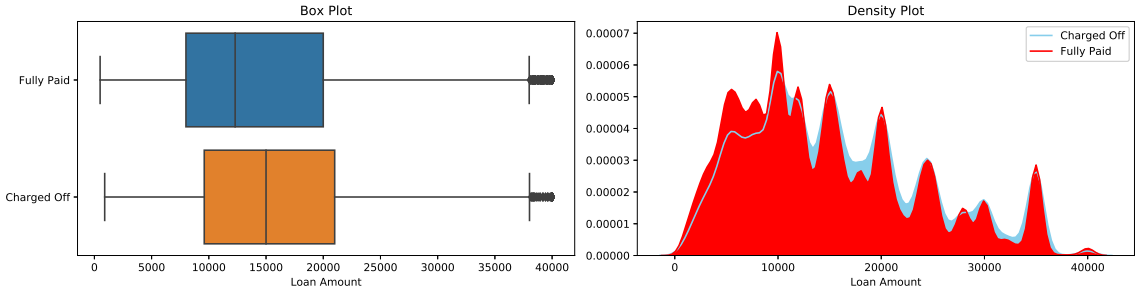
Figure 2.8: Sub Grade By Loan Status



2.2.7 Loan Amount

Figure 2.9 shows the loan amount by the loan status. It shows that the charged off loans tend to have higher dollar amount than the fully paid off loans.

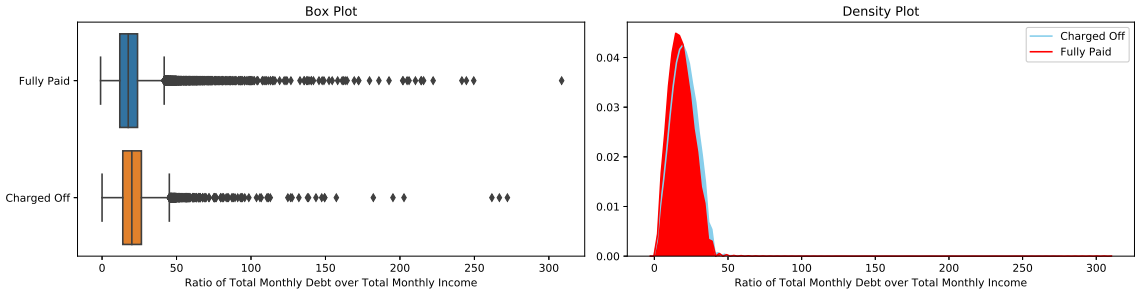
Figure 2.9: Loan Amount By Loan Status



2.2.8 DTI

Figure 2.10 shows the DTI (Ratio of Total Monthly Debt over Total Monthly Income) by the loan status. It shows that the charged off loans tend to have higher dti than the fully paid off loans.

Figure 2.10: DTI By Loan Status



2.3 Training/Test Split

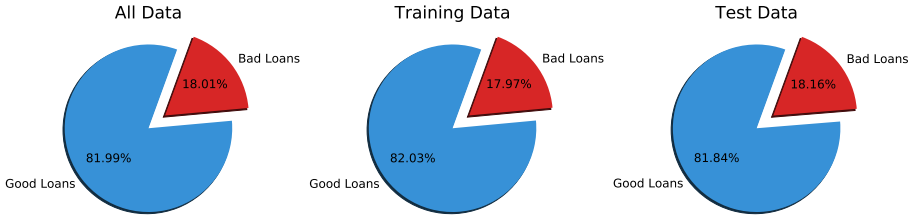
Most previous literature randomly split the data into training and test data over the whole sample period. However, the goal of the project is to predict whether a loan will be fully paid off in the future, therefore, we need to use historical data to predict the statuses of the future loans, and hence we take the first 80% data in our sample period as the training data and the remaining 20% data as the test data. After splitting, our test data mainly consist of $234,165$ loans made between $06-01-2016$ and $03-01-2017$. All the loans made before $06-01-2016$ are considered as training data.

Since we split the data by time, our results cannot be compared to the previous literature's results that randomly split the data.

2.3.1 Loan Status

Figure 2.11 shows the percentages of charged off and paid off for the whole, training, and test data sets. As we can see from this table, the test data set has a slightly lower default rate than the training data set.

Figure 2.11: Loan Status Distribution



Chapter 3

Classification Methods

3.1 Classification Problem Overview

Our goal is to predict whether a loan belongs to either "Paid Off" or "Charged Off". In the following section, we will implement several machine learning techniques including Logistic Regression, Support Vector Machine, K Nearest Neighbor Classification (KNN), Naive Bayes Classification, Decision Tree, Random Forest, AdaBoost, Gradient Boosting and XGBoost. In machine learning, performance measurement is an essential task. For metrics to evaluate the performance of these techniques, we mainly use accuracy and the area under the receive operating characteristic curve (AUCROC). We also report confusion matrix whose rows represent true lables and columns represent predicted label. The confusion matrix has the following form as Table 3.1.

Table 3.1: Confusion Matrix

	Predicted Paid Off	Predicted Charged Off
True Paid Off	TN	FP
True Charged Off	FN	TP

The Lending Club data set is an imbalanced data set in which there are 82.2% negative

examples. The accuracy will depend on the overall default rate of the test data set. Hence we also report the following metrics:

$$
\begin{aligned}
&\text{Precision} = \frac{TP}{TP+FP} \\
&\text{Recall/Sensitivity/True Positive Rate (TPR)} = \frac{TP}{TP+FN} \\
&\text{F1-score} = \frac{2TP}{2TP+FP+FN} \\
&\text{Specificity} = \frac{TN}{TN+FP} \\
&\text{False Positive Rate (FPR)} = \frac{FP}{TN+FP} \\
&\text{Accuracy} = \frac{TN+TP}{N} \\
&\text{support} = \text{the number of true instances for each label} \\
&\text{Weighted-avg metric} = \text{metric values weighted by the support}
\end{aligned}
\tag{3.1}
$$

AUCROC is one of the most important evaluation metrics for checking the performance of classification models at various thresholds settings. ROC is a probability curve and AUC represents degree or measure of separability. AUCROC tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at distinguishing between charged off loans and fully paid loans. The ROC curve is plotted with TPR against the FPR where TPR is on y-axis and FPR is on the x-axis.

In the following, we briefly give the overview of the models applied in this project. For more details about those models, please refer to Book "The Elements of Statistical Learnin" (Hastie et al. (2008)).

3.2 Models Overview

Logistic Regression

The logistic function maps a linear combination of features into the binary outputs through Sigmoid function. That is, $P(y_i = 1) = \frac{1}{1+\exp(-\theta^T x)}$. To derive the optimal parameters, the model minimizes the negative log likelihood function as following

$$
-\sum_{i=1}^{n} (y_i \log P(y_i = 1) + (1 - y_i) \log(1 - P(y_i = 1))) \tag{3.2}
$$

However, when there are too many features, sometimes it is better to use regularization methods to increase bias a little bit and decrease the variance, and hence reduce the mean squared error. The logistic regression with L_2 regularization has the following form

$$-\sum_{i=1}^{n}(y_i \log P(y_i = 1) + (1 - y_i) \log(1 - P(y_i = 1))) + \lambda\|\theta\|_2^2 \tag{3.3}$$

The logistic regression with L_1 regularization has the following form

$$-\sum_{i=1}^{n}(y_i \log P(y_i = 1) + (1 - y_i) \log(1 - P(y_i = 1))) + \lambda|\theta| \tag{3.4}$$

where $\lambda > 0$ is hyper-parameter and need to be tuned using cross validation methods.

We first define a metric (either accuracy or AUCROC) that we mainly focus on, and then using cross validation to tune the hyperparameter and select either L_1 or L_2 in search for the highest value of this metric.

Support Vector Machine

Since the data is not likely linearly separable, we use the hinge loss function.

$$\max(0, 1 - y_i(w^T x_i - b)) \tag{3.5}$$

We then wish to minimize

$$\frac{1}{n}\sum_{i=1}^{n}\max(0, 1 - y_i(w^T x_i - b)) + \lambda\|w\|_2^2 \tag{3.6}$$

Where λ is a tuning parameter that controls the tradeoff between loss and ridge penalty.

However, we could also consider the LASSO penalty, in this case, we wish to minimize

$$\frac{1}{n}\sum_{i=1}^{n}\max(0, 1 - y_i(w^T x_i - b)) + \lambda\|w\|_1 \tag{3.7}$$

K Nearest Neighbors

K nearest neighbors (KNN) is a simple algorithm that stores all available cases and classifies new cases based on the distance functions. We use linear discriminant analysis to reduce the dimensions before training our data using KNN. We have two hyperparameters. One is the number of reduced dimensions ranging from 2 to 6. Another one is the number of parameters that could

take values of $5, 25, 50, 125$. As before, we use cross validation to tune the hyperparameters.

Naive Bayes Classifier

Naive Bayes Classifiers are based on applying Bayes' theorem with naive independence assumptions between the features. We ran Naive Bayes using Gaussian probability distributions. That is, given a class C_k, $k = 1, 2$, the probability distribution is

$$p(x = v|C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp - \frac{(v - \mu_k)^2}{2\sigma_k^2} \tag{3.8}$$

As before, we use cross validation to tune the hyperparameters and select either L_1 or L_2 in search for the highest value of this metric.

Decision Tree

It uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). There are also hyperparameters that we need to use cross validation to tune, such as the maximum depth of the tree and minimum samples split at the nodes.

Random Forest

Random Forest classifier is one of the tree ensemble methods that operate by constructing a multitude decisions based on randomly splitting a subset of features, and then combine the outputs of multiple weak tree classifiers to have a strong classifier since it will reduce the variable at the cost of high bias.

Neural Network

For neurons in each layer, the $j - th$ output in layer i is computed as

$$a_j^i = g(W_j^{i^T} x + b_j^i) \tag{3.9}$$

The loss function for the final output of the network is using cross entropy (log loss). That is, the loss functions is

$$-\sum_{i=1}^{n} (y_i \log \hat{y}_i + (1 - y_i)(1 - \log \hat{y}_i)) \tag{3.10}$$

AdaBoost

Adaboost is short for adaptive boosting. The output of the weak learners is combined into

a weighted sum that represents the final output of the boosted classifier. It is adaptive because subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers. The base estimator here is decision tree. The hyperparameter in AdaBoost is the maximum number of estimators at which boosting is terminated.

Gradient Boosting

Like AdaBoost method, gradient boosting combines weak learners into a single strong learner in an iterative fashion. The major difference between AdaBoost and Gradient Boosting Algorithm is how the two algorithms identify the shortcomings of weak learners (for example, decision trees). While the AdaBoost model identifies the shortcomings by using high weight data points, gradient boosting performs the same by using gradients in the loss function.

XGBoost

XGBoost is an ensemble learning method. XGBoost lies in its scalability, which drives fast learning through parallel and distributed computing and offers efficient memory usage.

Chapter 4

Classification Results

4.1 Accuracy Metric

If the accuracy is chosen as our primary metric, after training the data, we get the following results shown in Table 4.1 for training data and Table 4.2 for test data. The highest accuracy score for training data is 0.9956 when using Random Forest. However, it clearly faces over-fitting problem as Table 4.2 shows the accuracy score for test data is just 0.8178. Table 4.1 and Table 4.2 also show that Naive Bayes Classifier performs the worst among all the models both in training and test data. However, it seems that all the remaining models perform almost the same. Logistic Regression, K Nearest Neighbors, Gradient Boosting, and XGBoost have the highest accuracy score both in training and test data. And hence if we only care about accuracy score, we would pick Logistic Regression, K Nearest Neighbors, Gradient Boosting, and XGBoost as our best models. Among these three models, the Logistic Regression is chosen as our primary model, since its computation is cheaper, and the coefficients of logistic regression are easier to interpret. Therefore, in the following, more results are shown only for Logistic Regression.

Table 4.1: Accuracy Rate of Training Data Set for Each Model

Model	Accuracy Rate
Logistic Regression	0.8215
Support Vector Machine	0.8203
K Nearest Neighbors	0.8216
Naive Bayes Classifier	0.7399
Decision Tree	0.8203
Random Forest	0.9956
Neutral Network	0.8217
AdaBoost	0.8203
Gradient Boosting	0.8216
XGBoost	0.8215

Table 4.2: Accuracy Rate of Test Data Set for Each Model

Model	Accuracy Rate
Logistic Regression	0.8211
Support Vector Machine	0.8184
K Nearest Neighbors	0.8196
Naive Bayes Classifier	0.7427
Decision Tree	0.8184
Random Forest	0.8178
Neutral Network	0.8202
AdaBoost	0.8184
Gradient Boosting	0.8211
XGBoost	0.8205

The estimated first 10 largest absolute coefficients values are reported in Table 4.3. As we can see that higher values of term, interest rate, dti, and the loan amount will lead to higher default probability. However, higher values of FICO score and annual income will lead to lower default probability. These relations are also shown earlier in Section 2 and in our expectation.

Table 4.3: Coefficients for the Logistic Regression

Features	Coefficients
term	0.234
int_rate	0.230
fico_score	-0.178
dti	0.128
annual_inc	-0.119
loan_amnt	0.093
revol_bal	-0.091
home_ownership_RENT	0.083
sub_grade_A2	-0.075
open_acc	0.071

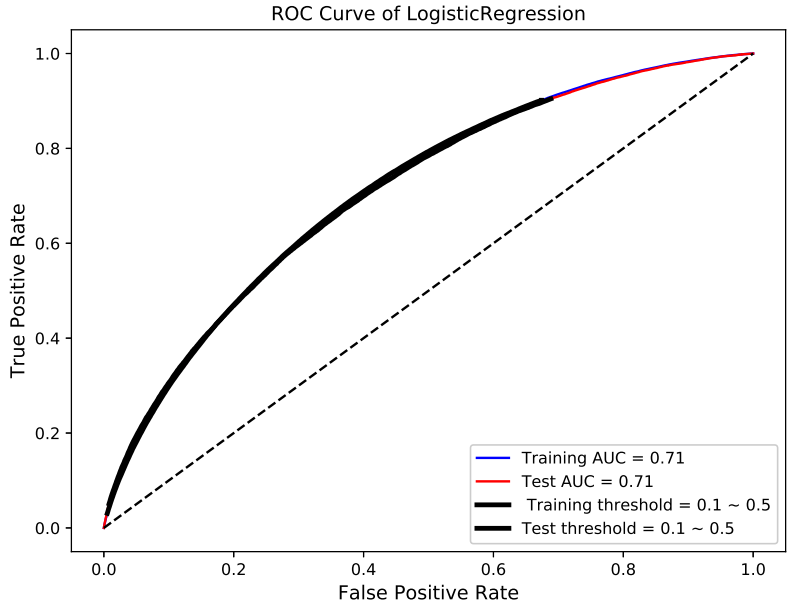
We then use Recursive Feature Elimination (RFE) to pick the 10 most important features. The result is shown in Table 4.4. The result here is similar to that in Table 4.3. The most important features are still interest rate, term, FICO score, dti, mort_acc, annual income, and loan amount. Combining Table 4.3 and 4.4, we conclude that interest rate, term, annual income, FICO score, dti, and loan amount are the most important features in determining whether a loan will be charged off or not.

Table 4.4: Feature Importance for the Logistic Regression

Features	Rank
int_rate	1
term	2
fico_score	3
dti	4
mort_acc	5
annual_inc	6
loan_amnt	7
revol_bal	8
open_acc	9
home_ownership_RENT	10

The ROC Curve of the Logistic Regression is shown in Figure 4.1. Surprisingly, the AUCROC score for the test data is the same as the training data.

Figure 4.1: AUCROC Curve for Logistic Regression



Here we report the confusion matrix and classification report for both training and test data.

Table 4.5: Confusion Matrix of Training Data Set for Logistic Regression

	Predicted Paid Off	Predicted Charged Off
True Paid Off	703882	3750
True Default	150262	4765

Table 4.6: Confusion Matrix of Test Data Set for Logistic Regression

	Predicted Paid Off	Predicted Charged Off
True Paid Off	190112	1527
True Default	40356	2170

Table 4.7: Classification Report of Training Data Set for Logistic Regression

Class	Precision	Recall	f1-score	support
Paid Off	0.824	0.995	0.901	707632
Default	0.56	0.031	0.058	155027
Weighted Avg	0.777	0.821	0.75	862659

Table 4.8: Classification Report of Test Data Set for Logistic Regression

Class	Precision	Recall	f1-score	support
Paid Off	0.825	0.992	0.901	191639
Default	0.587	0.051	0.094	42526
Weighted Avg	0.782	0.821	0.754	234165

4.2 AUCROC Metric

If the AUCROC score is chosen as our primary metric, after training the data, we get the following results shown in Table 4.9 for training data and Table 4.10 for test data. Notice that for

some models, we cannot calculate the AUCROC score, and hence we only report the results for eight models. The highest AUCROC score for training data is 0.7271 and for test data is 0.7153 when both using XGBoost. And hence if we just consider the AUCROC score, we would pick XGBoost as our best model.

Table 4.9: AUCROC Score of Training Data Set for Each Model

Model	Aucroc Score
Logistic Regression	0.7127
K Nearest Neighbors	0.7211
Decision Tree	0.726
Random Forest	0.7149
Neutral Network	0.7177
AdaBoost	0.713
Gradient Boosting	0.7109
XGBoost	0.7271

Table 4.10: AUCROC Score of Test Data Set for Each Model

Model	Aucroc Score
Logistic Regression	0.7101
K Nearest Neighbors	0.6995
Decision Tree	0.6711
Random Forest	0.7074
Neutral Network	0.7124
AdaBoost	0.7105
Gradient Boosting	0.709
XGBoost	0.7153

Chapter 5

Conclusions and Discussions

Various machine learning techniques are applied to predict the probability that a requested loan on LendingClub will be charged off over 2007–2017Q1 when almost all the loans have the final statuses. If only considering the accuracy rate, we would pick the logistic regression classifier as our primary model because it achieves a relative high accuracy rate in both the training and test data, and is cheaper in computation and easier to interpret. It achieves 0.8215 for training data and 0.8211 for test data. Moreover, it is found that the most important features for predicting credit risk are interest rate, term, annual income, FICO score, debt-to-income ratio, total credit revolving balance, and loan amount. However, if we instead focus on the area under the receiver operating characteristic curve (AUCROC), the XGBoost performs the best in both the training and test data. Hence our conclusion is that, depending on the focus, researchers may choose different models.

Bibliography

Chang, S., S. D. oong Kim, and G. Kondo (2015), Predicting default risk of lending club loans .
URL: *https://www.semanticscholar.org/paper/Predicting-Default-Risk-of-Lending-Club-Loans-S-Chang-Kim/6f64741e33b82e2dea0fe1179678e14bae05555b*

Hastie, T., R. Tibshirani, and J. Friedman (2008), *The Elements of Statistical Learning*, Springer Series in Statistics, New York, NY, USA: Springer New York Inc.

Li, P. and G. Han (2015), Lendingclub loan default and profitability prediction.
URL: *http://cs229.stanford.edu/proj2018/report/69.pdf*

Vinod, K. L., S. Natarajan, S. Keerthana, K. M. Chinmayi, and N. Lakshmi (2016), Credit Risk Analysis in Peer-to-Peer Lending System, in *2016 IEEE International Conference on Knowledge Engineering and Applications (ICKEA)*, pp. 193–196.