Predicting Default Risk of Lending Club Loans

Haiyang Sun, Hyder Ali

ABSTRACT

The Peer-to-Peer (P2P) lending industry has grown significantly in the past 10 years. Although it provides an alternative source of income other than bank deposit and corporate loans to investors, default risk due to asymmetric information has always been a top concern among P2P investors. In this study, we applied various machine learning algorithms to predict default risk of borrowers on Lending Club platform. Among all models, random forest achieved the highest recall score (0.906), which will effectively help investors avoid unnecessary losses from bad loans. We also identified top features with the highest predictive power. Feature impacts were measured and interpreted to generate insights and facilitate investment decisions. Finally, such predictive models could be potentially integrated into portfolio optimization algorithms to further improve portfolio returns.

1. INTRODUCTION

As online Peer-to-Peer (P2P) lending emerged in recent years, online micro lending platforms such as Lending Club have opened up a new channel for investors to generate steady investment income. As one of the largest P2P lending companies in US, Lending Club attracts millions of borrowers and investors and total loans issued have reached $35.9 billion by Q1 2018[1]. However, asymmetric information exists between borrowers and investors: Investors do not know the credibility of borrowers. Thus, losses may be incurred on bad loans. Although various information about borrowers are provided, such as loan purposes, annual income, delinquent records etc., investors generally lack technical knowledge to analyse default risk based on vast amounts of user information. In this study, we applied machine learning techniques to help investors identify loans with high default risk and avoid unnecessary investment loss. Such models can also be potentially integrated into portfolio optimization algorithms to maximize return given different risk preferences.

2. DATA PREPARATION

2.1 Dataset Description

Lending Club data for all 3-year loans in 2014 were extracted [2]. In total, there were 161,284 observations with 120 fields. Irrelevant fields such as loan IDs and URL were removed. By 2018, all 3-year loans have completed with a final status. To label the dataset, we classified any loans *defaulted* or *charged off* as negative samples (y=0) while *fully paid* loans were classified as positive samples (y=1).

2.2 Feature Constructions

Standard data pre-processing techniques were applied to clean up the dataset, such as encoding categorical variables, imputing missing values, removing outliers etc. Details can be found in Appendix I. Beyond the existing data set, we also built new features based on business intuitions, in the hope that they will bring in additional predictive power. A few examples are listed below:

- *tot_bal_2_inc*: Total Current Balance/Annual Income. Total current balance does not provide a full picture of the borrowers' debt burden. How heavy the burden is, relative to annual income remains to be explored. Hence, this ratio measures the proportion of total current balance compared to annual income.
- *revol_bal_2_inc*: Revolving Balance/Annual Income. Similarly, this ratio measures the proportion of revolving loan balance compared to annual income.
- *satisfaction rate*: #Satisfactory Accounts/#Total Accounts. This ratio measures the proportion of total accounts which have satisfactory status.

Additionally, to get a perspective on how socioeconomic characteristics of each state might affect borrowers' default rate, we added two more features, namely, racial composition of each state [3] and poverty rate in each state [4] to our dataset. Since we have the data on which state the borrower belongs to, we were able to map these features to the respective loan details.

2.3 Down Sampling

Since positive samples only accounted for 14% of all observations, classes are highly imbalanced. As conventional machine learning algorithms are biased towards the majority class, down sampling needs to be performed to reduce bias. We first split the dataset into training and testing set according to 80:20 ratio. Down sampling was subsequently performed on the training set. Figure 2.1 demonstrates the overall flow. "Fully Paid" samples are equally partitioned into 6 portions, and each portion will then be merged with "Charged Off" samples to form a training subset. On each training subset, we use grid search with K-Fold cross validation to identify the best set of parameters. Using the optimal hyperparameters, we will train a model on each training subset. Their prediction results will be ensembled to produce a final predicted label.

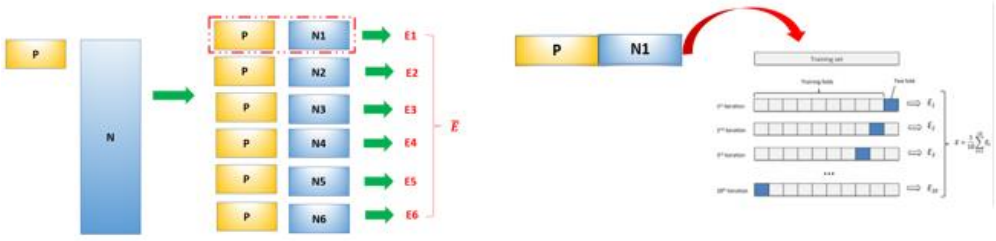

Figure 2.1 Down-sampling on the training set

3. FEATURE SELECTION

Boruta was a novel feature selection algorithm which finds all relevant features to the decision variable [5]. It is a wrapper algorithm around Random Forest Classification Trees which adds randomness to the system and collects results from the ensemble of randomized samples. In contrast to conventional feature selection algorithms, it considers all features which are either strongly or weakly relevant to the decision variable.

3.1 Boruta Algorithm

In Boruta algorithm, a "shadow attribute" is created for each variable, which was obtained by shuffling values of the original attribute across objects. Subsequently it trains a Random Forest Classifier on the extended dataset and measures the importance of each attribute by mean decrease in accuracy [5]. On the extended system, mean decrease in model accuracy (i.e. mean increase in out-of-bag error) is calculated by permuting values in each feature. For an unimportant feature, no matter its values are randomly shuffled or not, it will not affect model accuracy, since by no means it has no predictive power on the target variable. In contrast, after values of an important feature are shuffled, model accuracy will drastically decline. Z score is obtained for each feature by dividing the average loss by its standard deviation. For each attribute, a two-sided test of equality will be performed to compare its Z score with the *maximum Z score among shadow attributes* (MZSA). In this way, we are able to identify attributes with importance significantly higher or lower than MZSA. The above procedure will be repeated until the importance is assigned for all attributes.

3.2 Testing Result

After 12 rounds of iterations, 65 features were selected from Boruta Algorithm. Top 5 and Bottom 5 features are listed in Table 3.1. *Last_fico_range_high, last_fico_range_low, interest rate, dti* and *revol_bal_2_inc* have significantly higher importance scores than the rest. *Num_tl_30dpd, nonzero_delin1_amnt, purpose_house, delinq_amnt* and *num_tl_120dpd_2m* perform the worst, thus lacking predictive power.

Table 3.1 Top 5 and Bottom 5 Features from Boruta Algorithm

No.	Feature	Mean Imp	Median Imp	Min Imp	Max Imp	Importance Level
1	last_fico_range_high	34.06	33.96	30.57	38.99	Highest
2	last_fico_range_low	33.85	33.53	31.89	37.10	
3	int_rate	13.50	13.50	10.43	15.56	
4	dti	13.26	13.35	10.82	15.02	
5	revol_bal_2_inc	12.79	13.27	10.19	14.42	
…….						
95	num_tl_30dpd	-0.09	-0.21	-1.57	1.92	Lowest
96	nonzero_delin1_amnt	-0.25	0	-1.94	0.79	
97	purpose_house	-0.29	0	-2.29	1.02	
98	delinq_amnt	-0.35	-0.46	-2.54	1.60	
99	Num_tl_120dpd_2m	-0.36	-0.69	-1.73	1.14	

3.3 Feature Explanation

Further explanation is provided for the top features, which are listed below:

- **FICO:** FICO (Fair Isaac Corporation Score) is a type of credit score constructed by the Fair Isaac Corporation. Various factors are considered, such as payment history, current level of indebtedness, types of credit used, length of credit history and new credit accounts [6]. It is an important measure to assess borrowers' creditworthiness. With a typical range between 300 and 850, scores above 650 generally indicates good credit history.
- **Interest Rate:** As Lending Club offers loans of various grades, higher interest rates are assigned to borrowers with lower grades. Therefore, higher interest rate implies higher loan default risk.
- **Debt-to-Income Ratio (DTI):** DTI is a measure of an individual's ability to manage monthly payment and repayment of debts. As it is computed by dividing total recurring monthly debt by gross monthly income, typically a higher DTI increases the likelihood of default.

Furthermore, it is also worthwhile to notice that *revol_bal_2_inc*, a self-created feature based on domain knowledge, ranked one of the highest by the Boruta algorithm. This reinforces the importance of feature engineering during the data preparation stage.

4. MODEL INTERPRETATION

4.1 Results

Logistic Regression, Support Vector Machine, Random Forest and Gradient Boosting predictive models were applied to predict the default risk of Lending Club loans. The hyperparameters for the respective algorithms were selected based on Grid Search with K-Fold cross validation.

Due to the heavily imbalanced nature of Lending Club dataset, a high accuracy score does not reflect the true performance of the model. Thus, Precision and Recall were computed to measure the testing result. Since False Negative will potentially lead to investment loss (i.e. negative ROI), Recall score = TP / (TP + **FN**) will be particularly important to measure model performance. False Positive in Precision score represents opportunity cost of not investing in healthy loans. This is less concerning, because given vast amount of loans on the platform, there is no shortage of alternative loans which yield similar or better return.

Table 4.1 shows that all models yielded very similar AUC results (0.874-0.881), though XGBoost achieves the highest AUC among all four models. Interestingly, there exists a trade-off between precision and recall score: Logistic Regression has the highest precision score 0.519 but lowest recall score 0.862 among all models. On the opposite side, Random Forest has the highest recall score 0.906 but lowest precision score 0.452. Even though SVM and XGBoost did not achieve top performance in precision and recall, similar trade-off can also be observed. Such trade-off patterns will be visualized in Section 4.2 with further elaborations.

Table 4.1 Predictive Model Classification Results

	Logistic Regression	**Kernel SVM**	**Random Forest**	**XGBoost**
Optimal Parameters from Grid Search	C = 10 Penalty = 'L1'	Kernel = "rbf" C = 50 Gamma = 0.015	n estimators = 400 Max depth = 3 Criterion = 'entropy'	n estimators = 100 Learning rate = 0.01 Max depth = 3 Min child weight = 100
AUC	0.874	0.880	0.874	0.881
Recall	0.862	0.892	0.906	0.905
Precision	0.519	0.490	0.452	0.460

4.2 Model Comparison

Visualizing higher-dimensional data can be tricky. Fortunately, in our dataset, only a few features have high predictive power. In order to visualize the classifiers, two dimensions were constructed: debt-to-income ratio and average of four FICO scores. Figure 4.1 shows the contour plot of Logistic Regression and Random Forest. The scatter points represent a random sample of test dataset. It is obvious that when decision boundary moves to the right, a higher percentage of red scatter points (positive samples) fall under the red shade (predicted as 'positive'). Meanwhile, under the red shade (predicted as 'positive'), a lower percentage of scatter points are now red (true positive samples). Therefore, recall score increases and precision score decreases. Compared with Logistic Regression, Random Forest's decision boundary lies more on the right side. Hence, with the aid of visualization techniques, trade-off between precision and recall scores can be substantiated.

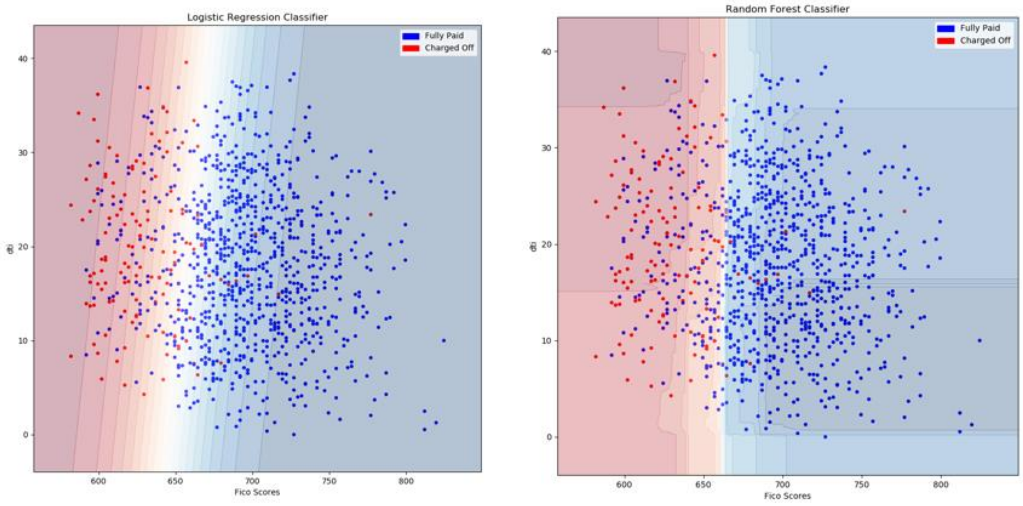

Figure 4.1 Model Comparison between (a) Logistic Regression Classifier and (b) Random Forest Classifier

Likewise, such trade-off also exists between True Positive Rate (TPR) and False Positive Rate (FPR). As the decision boundary moves further to the right, both TPR and FPR increase. In terms of AUC, XGBoost performs the best among all other models. This is evident from its decision boundary, whereby it does not lie too much to the left or right. Thus, XGBoost achieves a good balance between TPR and FPR with the highest AUC score.

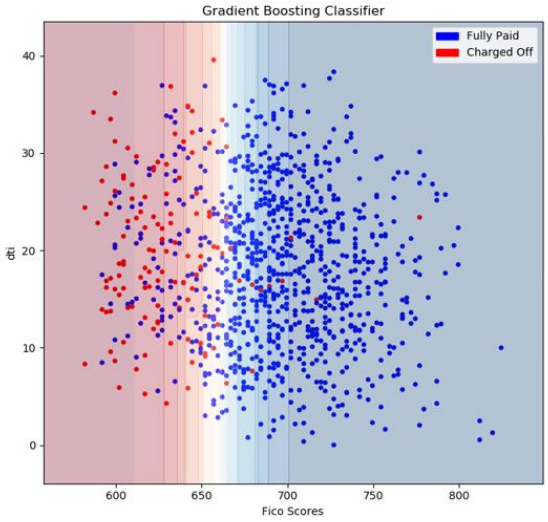

Figure 4.2 Gradient Boosting Classifier Visualization

5. FEATURE INTERPRETATION

5.1 Feature's Predictive Power

In Random Forest Classifier, *feature_importances_* measures the relative importance of each feature with respect to the predictability of the target variable. More often a feature is used as the splitting variable in a tree, more important it is. In addition, number of times each feature is used as the splitting variable is weighted by tree depth. Features used at the top of the tree have a stronger influence on prediction decision, since a larger fraction of samples were split by those features [7]. By averaging feature importance score of each tree, we can reduce the variance of the estimate and use it as a valid metric to evaluate features' predictive power.

Table 5.1 Top 10 Features with highest random forest importance score

Rank	Feature Name	Score	Rank	Feature Name	Score
1	Last Fico Range High	0.437	6	Fico Range Low	0.006
2	Last Fico Range Low	0.427	7	Fico Range High	0.006
3	Interest Rate	0.030	8	Debt-to-income	0.006
4	Grade	0.022	9	Annual Income	0.006
5	Sub Grade	0.022	10	# Accounts opened in the past 24 months	0.005

The top 10 features are listed in Table 5.1 *Last_Fico_Range_High* and *Last_Fico_Range_Low* have significantly higher importance scores than the rest. However, such quantitative measures are not intuitive and lack interpretability. Why did certain features (e.g. *Last_Fico_Range_High*, score=0.437) outperform the rest (e.g. *#active bankcard accounts*, score=0.00017)? Are we able to visualize each feature's predictive power to figure out the reasons behind? In section 5.1, we will follow a qualitative approach to interpret feature importance through visualization.

We aim to visualize predictive powers of the following three sets of features:

- Strong Features: Features whose random forest importance scores are in the Top 10 list
- Medium Features: Features whose random forest importance scores are in the mid-range
- Weak Features: Features whose random forest importance scores are the Bottom 10 List

In each feature set, we construct two dimensions to train the model. Using Strong Features as an example, we summarize the steps as below:

- Select the Top 10 features with the highest random forest importance score.

- Calculate pairwise correlation and combine features with high correlation into one group. If a feature does not have high correlation with all rest features, it can form a group itself.
- For each group, calculate the average importance score. Select two groups with the highest random forest importance scores.
- Collapse each group into one dimension by computing the mean of all features inside that group

Table 5.2 summarizes the two dimensions in each feature set. We will train a random forest classifier model on the two dimensions from each feature set. Subsequently, we will visualize features' predictive power via contour plot.

Table 5.2 Summary of feature groups for strong, medium and weak feature sets

Feature Set	Dimension	Description
Strong Features	1	Average (Last_Fico_Range_High, Last_Fico_Range_Low, Fico_Range_High, Fico_Range_Low)
	2	Debt-to-income ratio
Medium Features	1	Average (# Revolving trade with balance>0, # Open accounts, # Bankcard accounts)
	2	Annual Income
Weak Features	1	Average (# Active bankcard accounts, # Satisfactory bankcard accounts)
	2	Total credit balance excluding mortgage

5.1.1 Strong Features

Figure 5.1 (a) shows the contour plot for strong feature set (dti vs fico scores). Most 'Charged-Off' samples (i.e. red points) are located in the left panel. Hence, a vertical-line decision boundary is easy to separate the two classes. As Figure 5.1 (b) shows, the two-class distribution plots of *Last_Fico_Range_High* are highly separable with a small overlap region. 'Fully-Paid' group generally has much higher scores compared with its counterpart. Hence separability contributes to high predictive power. We also plotted the distribution of predicted probability of all mesh grid points. Extreme probability values are observed in Figure 5.1 (c): Most probability values fall close to either 0 or 1. Therefore, by passing samples to a good model which is built on strong features, we are able to get predicted loan status with high confidence.

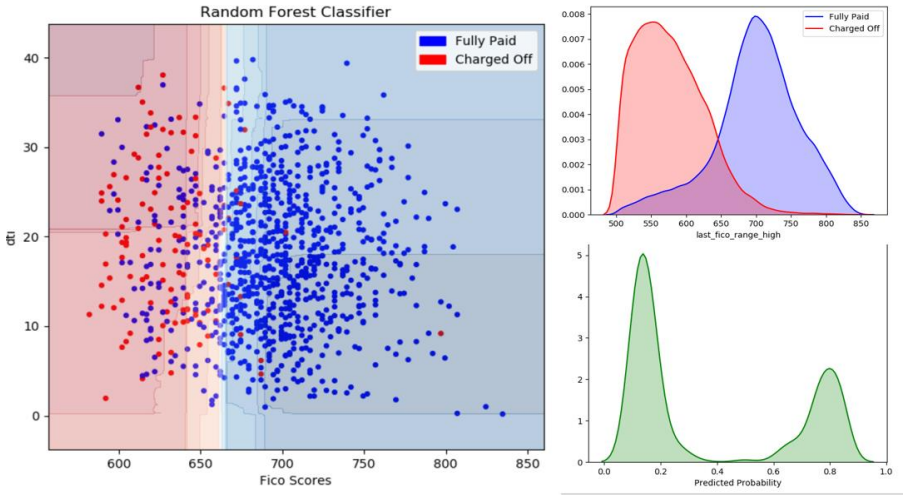

Figure 5.1 Visualization of Strong Features: (a) Left- contour plot (b) Upper Right- Two-class distribution of Last_Fico_Range_High (c) Lower-Right- Predicted mesh_proba distribution

5.1.2 Medium Features

Figure 5.2 (a) shows that samples from the two classes (i.e. blue points vs red points) are not separable using the selected dimensions. An ideal decision boundary similar to Figure 5.1 (a) cannot be found. The two-class distribution plots of annual income have a large overlap region (Figure 5.2 (b)). Hence the degree of separability

is much lower than the strong feature set. In *mesh_proba* distribution plot, extreme probability values are no longer found (Figure 5.2 (c)). Most predicted probability values fall in the range between 0.4 and 0.6. Thus, if features have lower predictive power, the model will yield predicted results with less confidence.

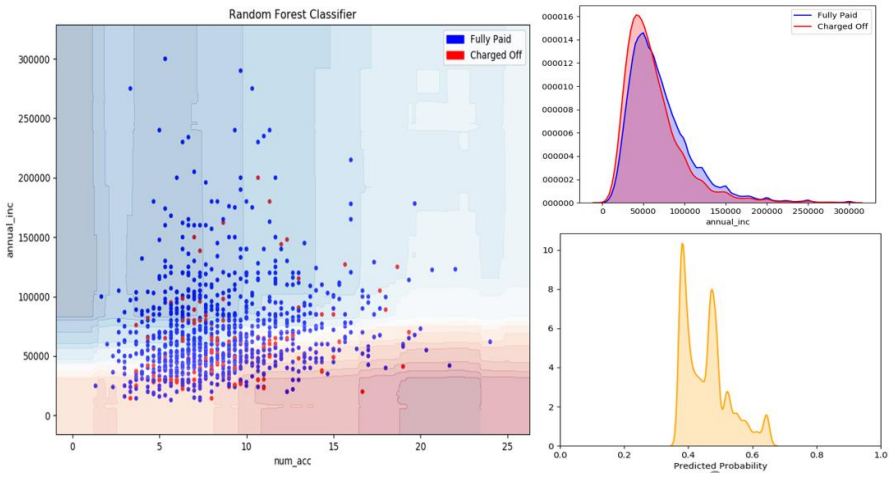

Figure 5.2 Visualization of Medium Features: (a) Left- contour plot (b) Upper Right- Two-class distribution of Annual Income (c) Lower-Right- Predicted mesh_proba distribution

5.1.3 Weak Features

In the contour plot for weak feature set, samples from the two classes are entirely non-separable (Figure 5.3 (a)). The two-class distribution plots of *total_balance_except_mortgage* almost perfectly overlap (Figure 5.3 (b)). Such extremely low separability is consistent with its low predictive power (i.e. importance score in the Bottom 10 List). One interesting finding is that most predicted probability values concentrate around 0.5 (Figure 5.3 (c)). Intuitively, when models are trained using "garbage" features, the model ends up with random guessing. In other words, *garbage in, garbage out*. Most areas in the contour plot has white colour, which corresponds to 0.5 predicted probability, i.e. random guessing.

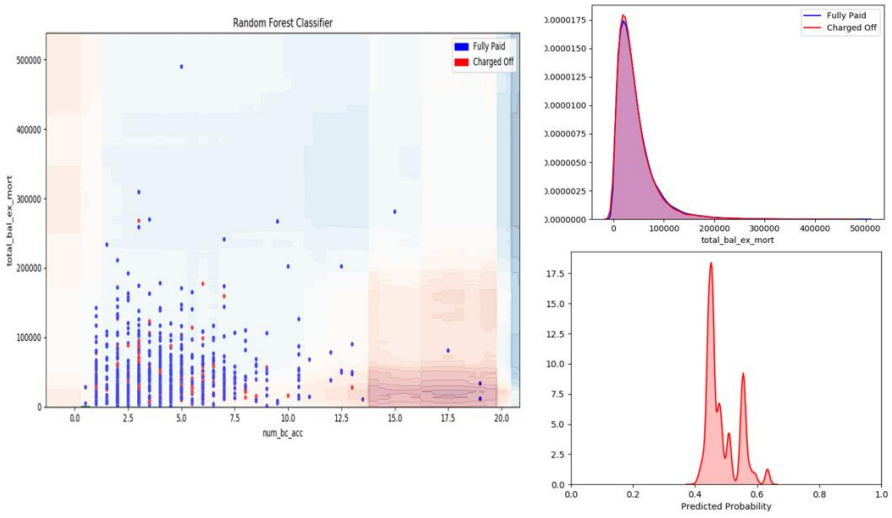

Figure 5.3 Visualization of Weak Features: (a) Left- contour plot (b) Upper Right- Two-class distribution of Total Balance Except Mortgage (c) Lower-Right- Predicted mesh_proba distribution

In short, features with high predictive power can generally well separate samples between the two classes. When models are trained on strong features, it tends to output predicted probability values close to either 0 or 1. This reduces uncertainty and generates more insights to guide investment decisions.

5.2 Feature Impacts

So far, we have measured features' predictive power through visualization. However, many questions remain to be answered: Does each feature impact the target variable in a positive or negative way? Are we able to quantify such impacts as well? Measuring feature impacts does not necessarily rely on machine learning models. Through Exploratory Data Analysis (EDA), insights can also be generated. In section 5.2, we aim to use both EDA and machine learning models to evaluate features' impacts on default probability, with a combination of qualitative and quantitative approaches. The general framework is elaborated below:

- Exploratory Data Analysis (EDA)
    - Qualitative Approach: Boxplot:

      For each continuous variable, we visualize the boxplots of the two classes. If the inter-quartile range of the default class is higher than the non-default class, the feature generally has a positive relationship with default probability.
    - Quantitative Approach: Two-sample t-test:

      If the p-value of the t statistic is smaller than the chosen level, we will reject the null hypothesis and conclude that the feature's sample mean between the two classes are significantly different from 0.
- Machine Learning Models
    - Qualitative Approach: Interactive scatterplot:

      We train a logistic regression model using training data. A scatterplot between predicted default probability and fico score will be drawn. Given the chosen feature, if the sample has higher feature value, the sample point will be assigned with darker colour. By holding fico score constant, we can visualize the feature's relationship with default probability. More will be demonstrated using examples below.
    - Quantitative Approach: Model coefficient:

      In logistic regression, log odds can be expressed as a linear combination of covariates (i.e. $\beta^T x$). Hence the odds ratio can be expressed as $e^{\beta^T x}$. When the i'th feature increases one unit, the new odds ratio becomes $e^{\beta^T x + \beta_i}$. Hence it can be derived that $\%\Delta \text{ odds} = e^{\beta_i} - 1$.

We use Last_Fico_Range_High and debt-to-income ratio as illustration. All other features can be measured in a similar approach.

- Last_Fico_Range_High
    - In Figure 5.4 (a), we observe that the inter-quartile range of *Last_Fico_Range_High* between the two classes are completely separable. The first quartile of the fully-paid class is even higher than the third quartile of the default class. Given that the first quartile of the fully-paid class has a value of 669, when *Last_Fico_Range_High* falls below 669, it is highly likely the loan will default. Investors need to be cautious about accepting such borrowers' request.
    - T-statistics = -309.93 (p-value = 0.0). Two sample means are significantly different at 1% level. Given the negative value, we know that there may be a negative relationship between Last_Fico_Range_High and default likelihood.
    - The scatterplot between default probability and Last_Fico_Range_High shows a negative relationship (in a sigmoid shape). When fico score increases from a low value, default probability first sharply declines, then decreases at a much slower pace (Figure 5.4 (b)).
    - Logistic regression coefficient = -0.0169. Since fico score is usually in an increment of 10, when Last_Fico_Range_High increases 10 units, odds ratio will decrease by 15.6%.

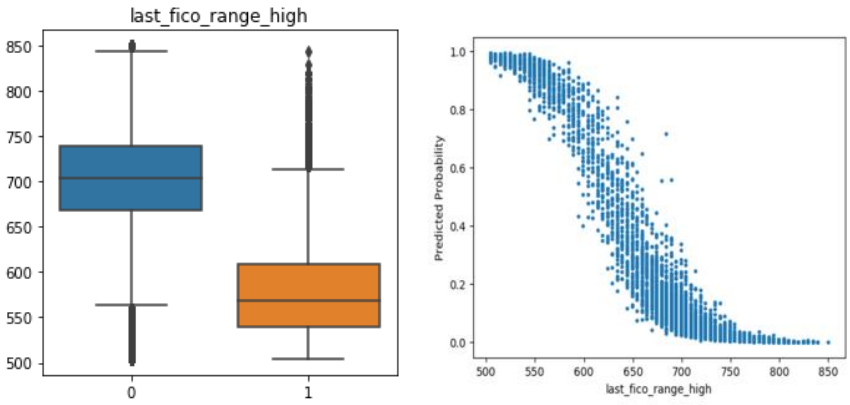

Figure 5.4 Visualization of Last_Fico_Range_High (a) Left- boxplot (b) Right-Scatterplot between default probability and feature value

- Debt-to-income ratio
    - In Figure 5.5 (a), we observe that the default class has a higher inter-quartile range in terms of dti compared with the fully-paid class. The two inter-quartile range boxes are less separable compared with Last_Fico_Range_High due to lower predictive power.
    - T-statistics = 32.33 (p-value = 1.58 x 10^{-24}). Two sample means are significantly different at 1% level. Given the positive value, we know that there may be a positive relationship between dti and default likelihood.
    - In Figure 5.5 (b), higher dti values are associated with darker colour. When we hold fico score constant and look vertically, we observe that most darker blue points are in the upper region and lighter blue points are in the lower region. Therefore, holding fico score constant, we see that higher debt-to-income ratio translate into higher default probability.
    - Logistic regression coefficient = 0.02214. When dti increases 1 percentage point, odds ratio will increase by 2.2%

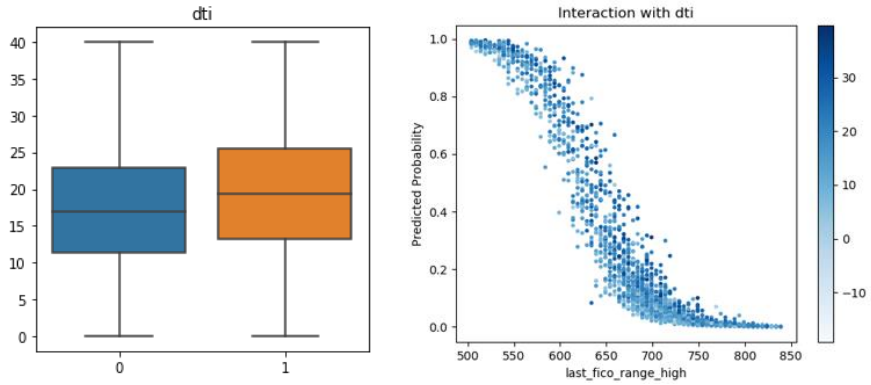

Figure 5.5 Visualization of dti (a) Left- boxplot (b) Right-Interactive scatterplot for dti

We will not repeat the analytical procedure above for all features. Instead, we will quantify the impact of some features with high predictive power:

- When interest rate increases 1%, odds ratio will increase by 5%. Such positive impact on default probability comes from two sources: (1) Lending Club sets higher interest rate for those borrowers with lower credit grades. Hence interest rate is associate with creditworthiness. (2) Higher interest rate leads to higher interest payment. With heavier repayment burden, default probability will increase.
- Whenever a borrower opens a new account, odds ratio will increase by 1.27%. Similarly, in the interactive scatterplot (omitted), most dark colour points appear in the upper region if we hold fico score constant. Hence, when fixing fico score, higher number of accounts generally translate into a higher probability of default.
- If a feature's two-class inter-quartile ranges in boxplot are separable (analogous to Last_Fico_Range_High), we can efficiently generate similar insights for investors: When interest rate is higher than 12.5%, or grade is

lower than C, or debt-to-income ratio is higher than 23%, investors are advised to be cautious about accepting such loans. The default probability will generally be very high.

6. CONCLUSION

In this study, we used different machine learning algorithms to predict Lending Club loans' status and default probability. Gradient Boosting yields the highest AUC, which is 0.881. Overall, all models have high recall scores (0.86 – 0.91). This will effectively help investors prevent investment loss by identifying loans which are prune to default. Precision score is much lower (0.46 – 0.52). Investors may forgo good investment probabilities. Fortunately, opportunity cost is limited. Given the large transaction volume on Lending Club, there is always no shortage of alternative loans with similar or even better returns.

We also tried to measure features' predictive power and impact on the target variable through visualization. Measuring feature impacts does not necessarily rely on machine learning models. In fact, quick insights can be generated more efficiently through boxplot visualization. If any interesting patterns are found, we can further quantify such impacts through machine learning algorithms and model coefficient interpretation.

In the future, we will incorporate this loan prediction model into some existing portfolio optimization algorithms. By eliminating loans with high default risk, we can further improve portfolio return for investors.

REFERENCES

[1] Lendingclub.com, 'Lending Club Statistics – Total Loan Issuance', 2018 [Online]. Available: https://www.lendingclub.com/info/statistics.action [Accessed: 4 Aug, 2018].

[2] Lendingclub.com, 'Lending Club Statistics – Download Loan Data', 2018 [Online]. Available: https://www.lendingclub.com/info/download-data.action [Accessed: 4 Aug, 2018].

[3] Census.gov, 'U.S. Census Bureau, Current Population Survey, 2014 to 2017 Annual Social and Economic Supplements', 2018 [Online]. Available: https://www2.census.gov/programs-surveys/demo/tables/p60/259/statepov.xls [Accessed: 4 Aug, 2018].

[4] Governing.com, 'State Minority Population Data', 2018 [Online]. Available: http://www.governing.com/gov-data/census/state-minority-population-data-estimates.html [Accessed: 4 Aug, 2018]

[5] M. B. Kursa and W. R. Rudnicki, "Feature Selection with the Boruta Package", *Journal of Statistical Software* vol. 36 Sept. 2010

[6] S. Gatti, and F. Querci. "The sensitivity of the loss given default rate to systematic risk: new empirical evidence on bank loans", *Journal of Financial Services Research* vol. 34 2008

[7] scikit-learn.org, "Ensemble Methods", 2017 [Online]. Available: http://scikit-learn.org/stable/modules/ensemble.html#random-forest-feature-importance [Accessed: 4 Aug, 2018]

APPENDIX I

The entire data cleaning process is further elaborated with more details below. Note that under each bullet point, a few representative examples are provided, i.e. examples are not exhaustive.

- Encode categorical variables.
    - Grade: {'A'=0, 'B'=1, 'C'=2, 'D'=3, 'E'=4, ' F'=5, 'G'=6}
    - Verification Status: {'Not Verified'=1,'Verified'=0,'Source Verified'=0}
    - Loan Purpose: One-Hot encoding
- Delete payment-related features because such obvious information is not supposed to be revealed for loan default prediction.
    - Sample of features deleted: Total received principal amount, total received interest payment, last payment amount and total received late fees.
- Handle missing values
    - Remove features with a large percentage of missing values. 15 features were removed in this way.
    - Fill missing values for columns such as 'Number of months since most recent bankcard account opened' with the median value.
    - Fill 'Accounts ever 120 or more days past due' with 0, since more than 90% observations have zero value.
- Remove outliers
    - Annual Income: remove observations > $300k
    - Revolving Balance: remove observations > $100k
    - Total Current Balance: remove observations > $700k