Data Science for Business 2
2019

Lending Club Portfolio Construction

Sirong Huang, 361150
Rainer Klein, 399779
Rohit Saluja, 723028

Contents

Executive summary

Lending Club is the world's largest online peer to peer lending portfolio, and it publishes almost all of its loan data online. Our objective was to take that data, understand and preprocess it and then find an optimal portfolio that maximizes the return on investment. This involved predicting the probability of default using two ML algorithms namely Logistic Regression and Random Forest, calculating the expected return and finally passing this data through Gurobi, along with portfolio diversification constraints to find the optimal portfolio.

We started with data preprocessing and exploratory data analysis to find the get rid of the missing values in the data and the features that have less predictive significance. Then feature engineering was done to add new categorical variables followed by one-hot encoding of few features. After this, some time was spent on Data Visualization to visually understand the relationship between various features of the dataset. Since, the dataset was heavily imbalanced we tried various methods to fix this and finally settled on the Class Weight method. Logistic Regression and Random Forests were used to predict the probability of default for each loan and then calculate the estimated return. Finally, Gurobi was used to find a portfolio that can maximize the expected return.

We recommend investing according to the model developed by logistic regression. Our calculations indicated a realized return of 21.29% on the test set (compared to realized return of 10.87% for the whole Lending Club investable set).

Introduction

MOTIVATION

The primary motivation for us as inspiring machine learning engineers is to solve a real time problem using machine learning. Using Machine Learning in the field of finance is really picking up because it is helping in making the investment predictions much more accurate and reducing the risk factor drastically. Each member of our team is quite intrigued by the application of machine learning in Finance and through this project we have learnt a lot about various aspects of applying ML in Finance ranging from dealing with financial data to building an optimized and diverse portfolio to minimize risks and maximize returns.

PROBLEM DESCRIPTION

Lending Club is the world's largest online peer to peer lending portfolio. We are working with lending club's loan portfolio data to train a Machine Learning model to predict the probability of default for each loan and then eventually find an optimal portfolio to maximize the return on investment.

OBJECTIVES AND EXPECTED RESULTS

- Understanding the lending club data and preprocessing it.
- Using ML models to predict the probability of default for each loan and expected returns for the test dataset.
- Find the optimal investment portfolio that maximizes the expected return and minimize the risk.

Data

EXPLORATORY DATA ANALYSIS AND PREPROCESSING

The training and testing datasets consist of 20 predictive features and roughly 100,000 loan records respectively. Various preprocessing steps were conducted to transform the raw data into suitable formats for default probability predictive modeling.

TARGET VARIABLE

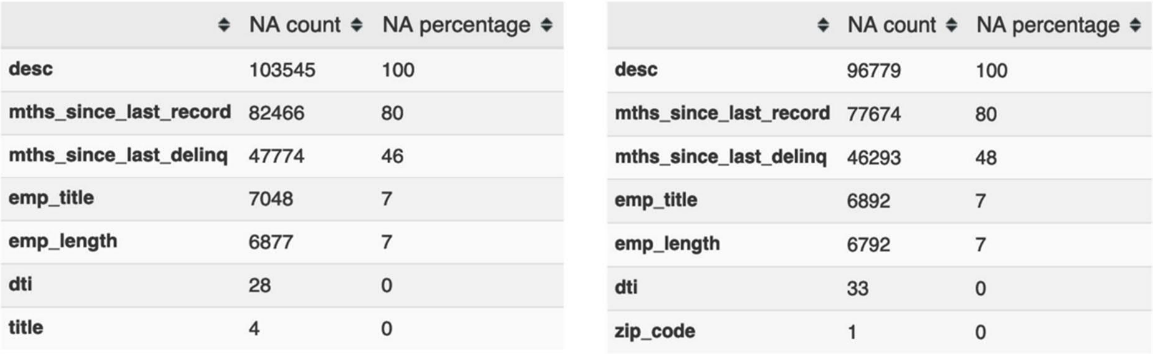
Firstly, the target variable is defined as such: "grace period", "late", "charged off", and "default" are categorized into the "default" loan category, whereas "current" and "fully paid" are defined as "not in default".

MISSING VALUES

Both training and testing datasets have missing values in the variables shown in Table 1. These missing values were either removed or filled with sensible values chosen in accordance with the meaning of the variables:

- "desc" variable has 100% missing values, so it was simply removed.
- "mths_since_last_record" means the number of months since last loan record. Missing values most likely means it's the first loan for the borrower in the platform. Therefore, the NAs are replaced with zero.
- "mths_since_last_delinq" refers to the number of months since last loan default. Missing value most likely means there is no loan default (within this dataset). Therefore, its NAs are replaced with a number (1000) much larger than the maximum value (195) of this variable, in order to maintain the logical order of the numerical values.
- "emp_title" and "emp_length" describes two employment status of the loan applicatns. NA in this case can mean that the borrower is unemployed at the moment. Therefore, their NAs are filled with empty strings and zeros respectively.
- "dti" is a ratio calculated using the borrower’s total monthly debt payments on the total debt obligations. Its NA values are replaced with its mean.

- "zip_code" contains only 1 missing value. It's replaced with the most common value from the same state.
- "title" is the loan title related to the application. The NAs are replaced with empty string.



	NA count	NA percentage
desc	103545	100
mths_since_last_record	82466	80
mths_since_last_delinq	47774	46
emp_title	7048	7
emp_length	6877	7
dti	28	0
title	4	0

	NA count	NA percentage
desc	96779	100
mths_since_last_record	77674	80
mths_since_last_delinq	46293	48
emp_title	6892	7
emp_length	6792	7
dti	33	0
zip_code	1	0

Table 1 Missing value counts for training and testing dataset

FEATURE ENGINEERING

In order to fully capture the explanatory power of the variables, some new variables were generated from existing variables.

"management_role" is a new binary variable generated from "emp_title".

There are 38, 367 unique employment titles in the datasets. It's impractical to convert all these titles into one hot encoded binary variables. Very fine-grained job title and industry differences are probably not crucial to the prediction of loan default. Management and senior roles, however, is a logical indicator of higher salary and more job stability. As can be seen from the word cloud in **Figure 1**, management and senior roles constitute a significant portion of the borrowers, around 27%. Therefore, we use "management_role" to represent the most important information in the variable "emp_title” and remove the original "emp_title" variable from future modelling process.

"region" is a new categorical variable generated from "states".

There are 49 states in the datasets. Some states may contain crucial information on the economy and income level of their residences. The "region" variable summarizes and groups the regional difference on a larger geographical scale. The 49 states are grouped 10 categories such as "east", "west", "mid-west", "south west", "north east" etc. based on North American's economical and geographical differences.

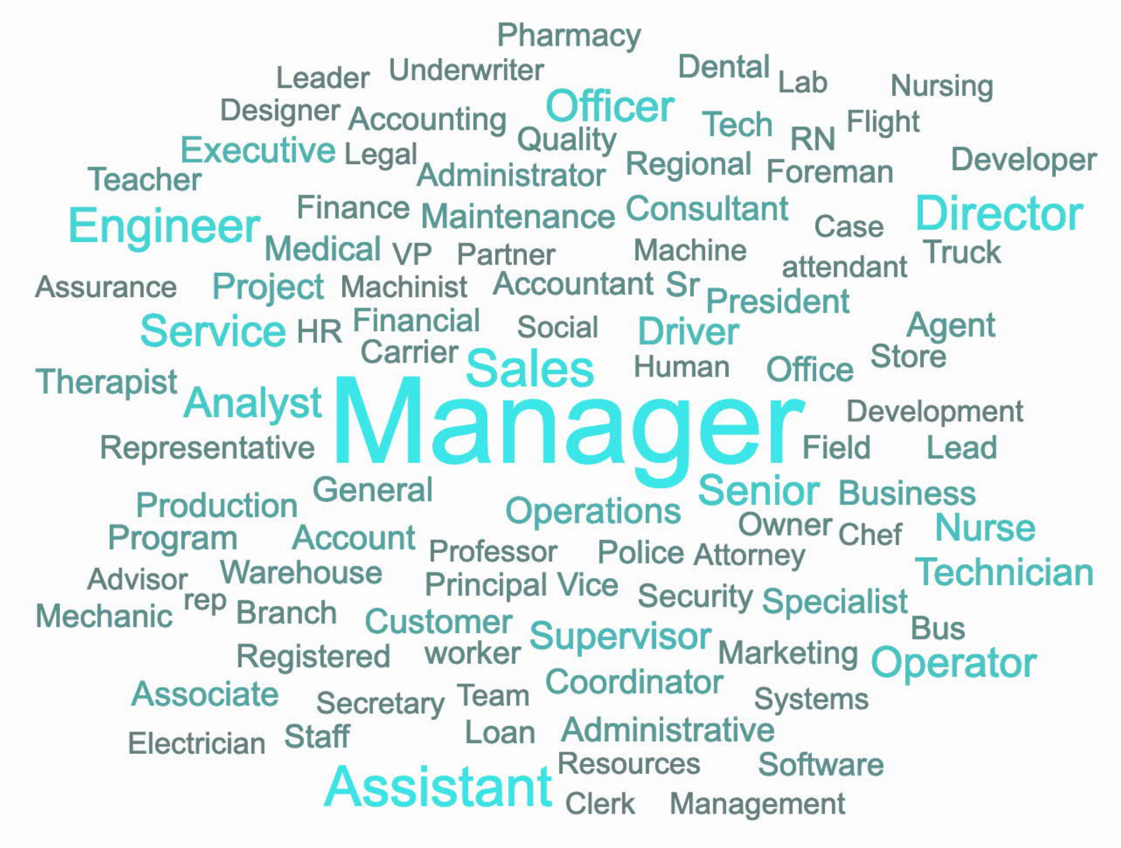


Figure 1 Word cloud of borrowers' employment titles

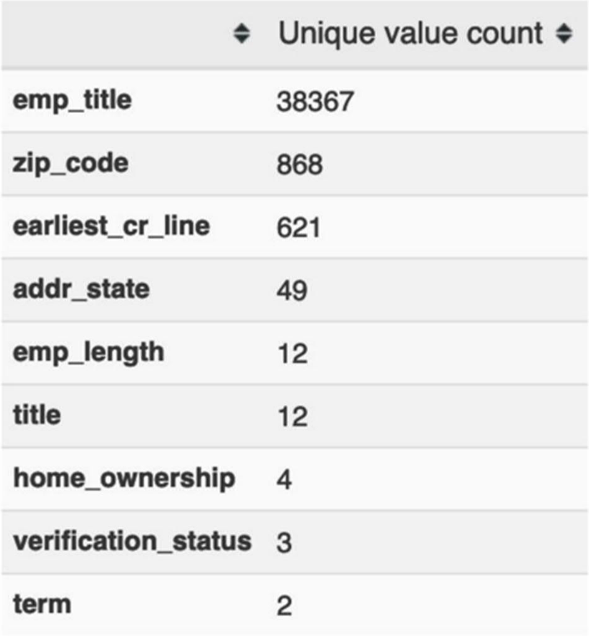
CATEGORICAL VARIABLE TRANSFORMATION

There are 9 categorical variables in the raw datasets, and 2 generated categorical variables.

"emp_title" and "zip_code" contains too many unique values, which may cause overfitting even for such large number of records in the training set. Therefore, they are removed for the modeling process.

"earliest_cr_line" means the month of the earliest credit record of the borrower. The data type is string, but it's actually a numerical variable in nature. This variable is first transformed into date format, and then the number of days between the first credit record and a recent date (1.4.2019) was calculated and used as the final values.

The rest of the categorical variables contain reasonable amount of unique values and are simply converted into binary variables using one hot encoding technique.



⬍	Unique value count ⬍
emp_title	38367
zip_code	868
earliest_cr_line	621
addr_state	49
emp_length	12
title	12
home_ownership	4
verification_status	3
term	2

Table 2 Unique value count for categorical (non-numerical) variables

DATA VISUALIZATION

Most numerical variables have left-skewed long-tail distribution. However, those outliers are very important potential indicator of loan default. Therefore, we decide not to scale or transform those values, in order to preserve their information.

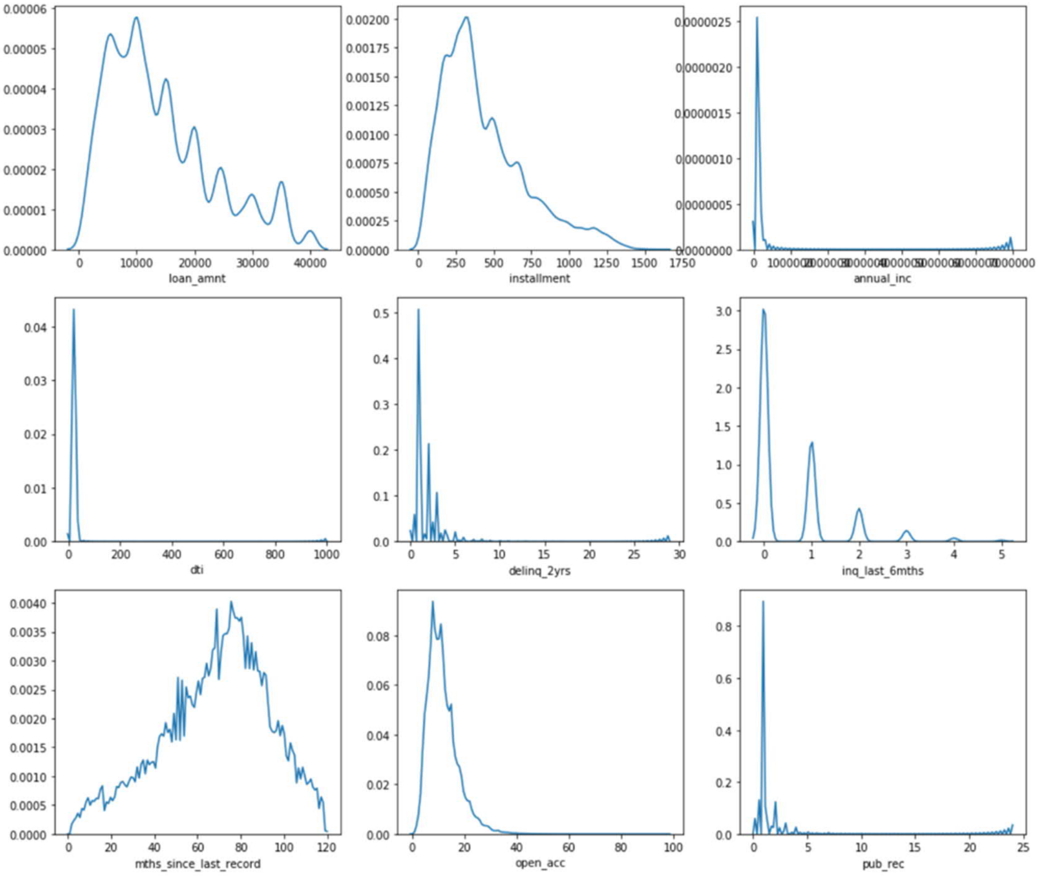


Figure 3 Unique value count for categorical (non-numerical) variables

As can be seen from the correlation heatmap in Figure 3, most features are not highly correlated. There are a few exceptions. "loan_amt" and "installment" were very strongly correlated, as well as "pub_rec" and "mths_since_last_record". These strong correlations will be taken into consideration in the model discussion section.

The target variable doesn't appear to be strongly related to any variables.

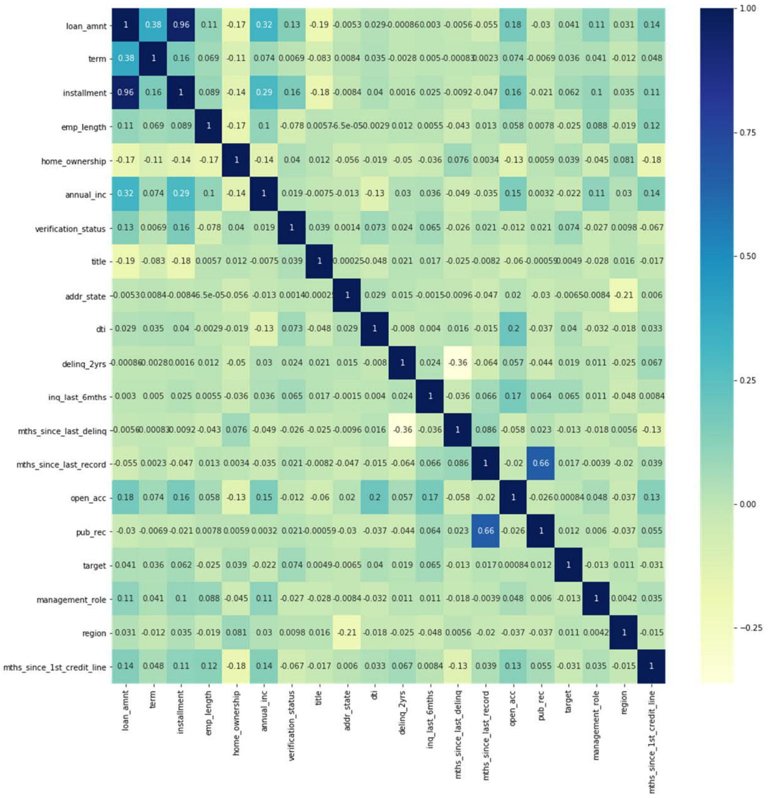


Figure 2 Correlation between predictive variables

FINAL DATASETS FOR MODELING

After feature engineering, the original 20 variables are transformed into 90 variables. The resulting shape for training and testing datasets are: (103546, 91) and (96779, 91) respectively.

CLASS IMBALANCE AND PERFORMANCE MEASUREMENT

From the graph below we can see that the target variable is extremely unbalanced, with only around 8.3% default cases. This may cause our classifiers to be overly biased towards "not in default" predictions and harm their capability to learn the default cases properly. In extreme cases, the classifiers may learn to only classify everything as "not in default" which still manages achieve almost 92% accuracy. Therefore, accuracy in this case is not a good measurement of model performance. In model selection, we used ROC AUC score for performance measurement instead.

We attempted to tackle this problem through different methods, such as upsampling the default class, downsampling the not in default class, and setting class weights in classifier settings. Downsampling and setting class weights achieved similar improvements on both ROC AUC and F1 score, and both outperformed upsampling method. Since there are very limited "not in default" cases, up sampling may not be able to generate pseudo samples with good enough quality, and down sampling may reuse the same "not in default" cases too many times that the model simply memorized those samples' labels. Therefore, we decided to use class weights as the final method to deal with the class imbalance issue. The optimal class weights were found through grid search with ROC AUC score and 5-fold cross validation.

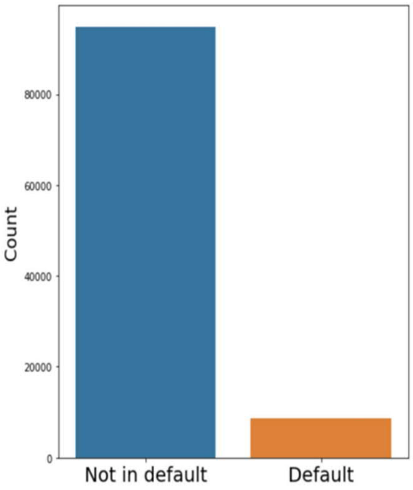


Figure 3 Target Variable Categorization

Predictive model and results

We chose Logistic Regression and Random Forests models to predict whether a loan would default or not. The performance of the models was compared using the confusion matrix and accuracy score. Finally, the probability of default for each loan was taken from the models to calculate the Expected Relative Loss and Expected Return. The results of the models have been discussed below.

LOGISTIC REGRESSION

The first algorithm we worked with to predict the probability of default was logistic regression. It was observed during the preprocessing phase that the data is heavily imbalanced and therefore we were getting a very high accuracy with our model. To tackle the imbalanced nature of the dataset we decide to use an inbuild method in logistic regression called class_weight. Using the class_weight we can incorporate the weights of the class in the objective function of the logistic regression and make it aware of the imbalanced nature of the data. We experimented with different class_weights and compared the performance of our models using AUC and accuracy score. We chose the Logistic regression model with the best AUC score. Our best logistic regression model gave us an AUC score of 0.67 and an accuracy of 70 %. Given below are the AUC and accuracy scores we got for the various Logistic Regression models. Feature importance chart can be found from the Appendix.

Table 4: AUC and Accuracy score comparison for Logistic Regression models with different class weights

AUC Score	Accuracy Score
0.633	0.89
0.670	0.703
0.578	0.914

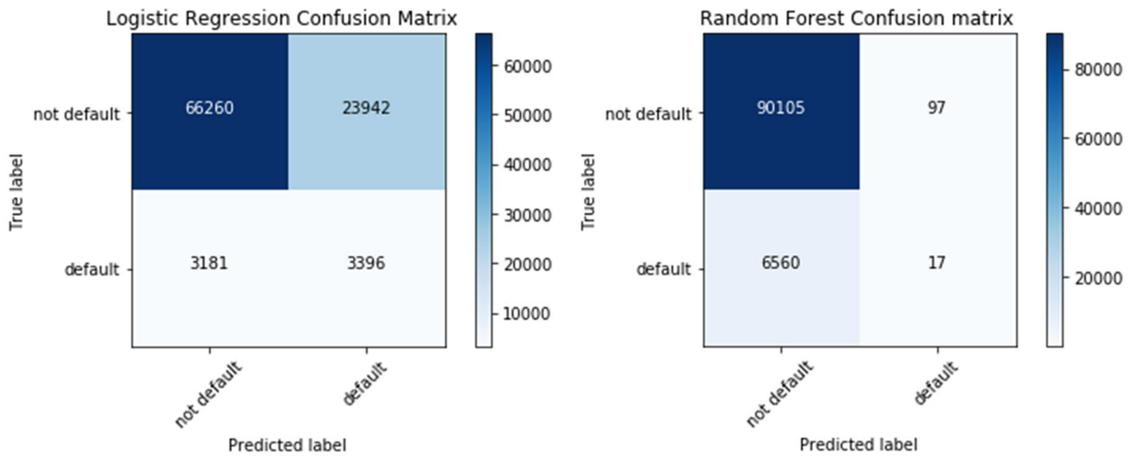


Figure 3: Confusion matrices

RANDOM FOREST

The second model we used was Random Forest. We also built a model with balanced random forest from imblearn library to tackle the imbalanced nature of the dataset but the results we got with the balanced random forest during portfolio optimization were unreasonable and hence we decided to stick with the normal random forest classifier. The AUC score we got with random was 0.55 and an accuracy score of 0.93. Given below is the confusion matrix and the feature importance chart for decision trees.

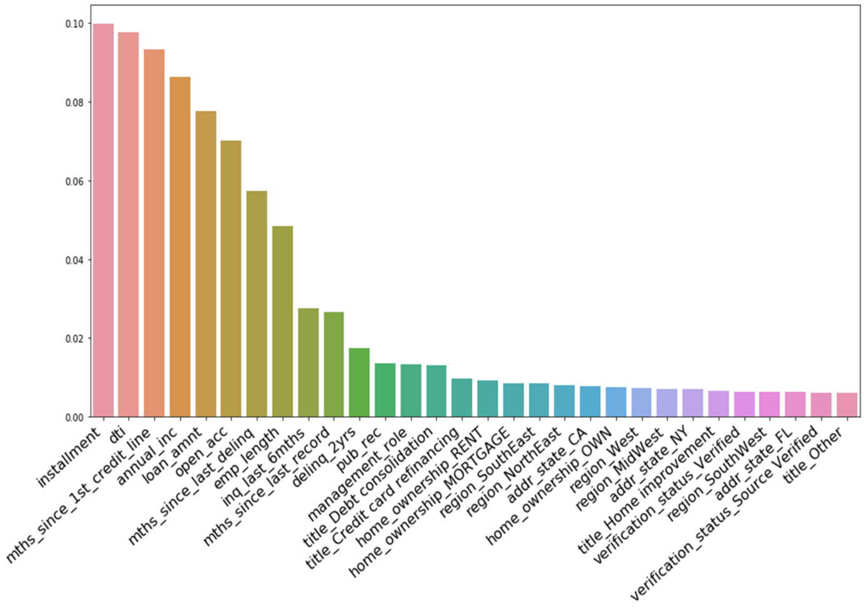


Figure 4: Feature Importance from Random Forest

Prescriptive model and results

OPTIMIZATION MODEL

We used the two probability of default (PD) vectors obtained in the previous task for the loan data in the LC_TestData.csv. With default probabilities for each loan, we calculated their expected return, defined as

$$E(R) = int_{rate} - PD * LGD, where$$

int_{rate} is given in the Lending Club dataset and

LGD (loss given default) is estimated to be 40% for all loans.

With this vector of expected returns, we turned to Gurobi Optimizer to build the optimal portfolio given the following objective function:

$$\max ER_{Portfolio} = \max(\boldsymbol{E}(\boldsymbol{R}) \cdot \boldsymbol{inv}), where$$

$\boldsymbol{E}(\boldsymbol{R})$ is the vector of return estimates and

$\boldsymbol{inv}$ is the vector of investments in loans

Both vectors are of the size 96779 x 1 and their dot product represents the sum of their element-wise products – the estimated return of the portfolio $ER_{Portfolio}$.

The constraints are presented below:

1. Total loan portfolio size $5 000 00

$$\sum_{i=1}^{96779} inv_i = 5\,000\,000$$

2. Max investment in any loan is the loan amount

$$inv_i \leq loan\_amnt_i$$

3. Max $500 000 invested in any US state

$$\sum_{i \in i_{state}} inv_i \leq 500\,000$$

Constrains nr 1 and 2 were simple to implement. To successfully implement constraint nr 3 we tested some 10 different solutions posted online. In total the model had 96781 rows, 96779 columns and 206923 nonzeros.

REALIZED RETURN CALCULATION

Past mean return is the best unbiased estimate for the mean future return (Diderich, 2009). We started with the ER formula described on the previous page, substituting default probability with “target” – a binary variable indicating default. When optimizing our portfolio to maximize the estimated return, we reached realized returns of around 30% calculated according to the following formula:

$$R(R) = int_{rate} - target * LGD,$$

int_{rate} is given in the Lending Club dataset,

$target$ is 1 if the loan went into default and 0 if it did not,

LGD (loss given default) is estimated to be 40% for all loans.

When inspecting the results loan-by-loan we noticed that there are some interesting results. Namely the portfolio consisted of many high-interest loans. The real returns will depend highly on the amount of payments the loan applicants will make before going into default. For example, if we give out a loan promising 30% interest and we get no payments at all, R(R) = 30%-1*40%=-10%, while the actual return is -40%. The formula above assumes that 1-year worth of interest is gathered from each loan and the default happens 1 year from the moment the loan is given.

Our experience with corporate bonds (where LGD values hover around 60%) led us to believe that LGD value of 40% might be low. (Guangyou;Zhang;& Luo, 2018) study of Lending Club loans indicated a mean LGD of 62% with standard deviation of 22%. We found that practices to vary somewhat but the common standard is that interest and fees should be included in the LGD calculation (European Banking Authority, 2016). For the purposes result comparability, we kept the LGD value at 40% and continue to use the realized return formula presented above.

RESULTS

The Gurobi Optimizer was run for our optimization problem with the two probability vectors (logistic and random forest). The optimal portfolios were assessed based on their realized return. The estimated and realized returns are presented in Table 5.

Table 5: Return comparison

	Logistic	Random forest
Lending Club whole-universe estimated return	10.25%	9.94%
Lending Club whole-universe realized return	10.87%	10.87%
Our portfolio estimated return	28.86%	30.17%
Our portfolio realized return	21.29%	20.95%

Logistic Regression

An overview of the portfolio constituents from the logistic model is presented in Table 6 and Figure 5.

Table 6: Optimal portfolio constituents according to logistic regression

	PD	ERL	loan_amnt	int_rate	ER	target	inv	RR
Min	0 %	0 %	1000	5 %	-29 %	0	0	-35 %
Median	9 %	3 %	12000	13 %	9 %	0	0	11 %
Mean	10 %	4 %	14858	13 %	10 %	0	52	10 %
Max	89 %	36 %	40000	31 %	31 %	1	40000	31 %
Count							238	

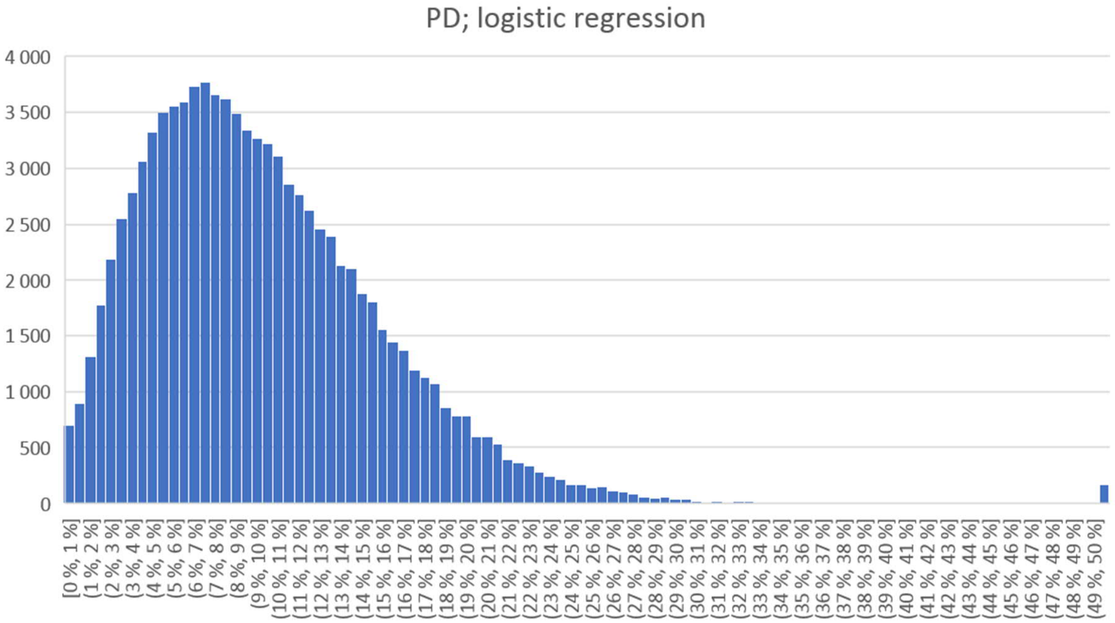


Figure 5: Histogram of default probabilities assigned to loans by logistic regression

Random Forest

An overview of the portfolio constituents from the random forest model is presented in Table 7 and Figure 6.

Table 7: Optimal portfolio constituents according to random forest

	PD	ERL	loan_amnt	int_rate	ER	target	inv	RR
Min	0 %	0 %	1000	5 %	-21 %	0	0	-35 %
Median	10 %	4 %	12000	13 %	9 %	0	0	11 %
Mean	9 %	4 %	14858	13 %	10 %	0	52	11 %
Max	90 %	36 %	40000	31 %	31 %	1	36000	31 %
Count							286	

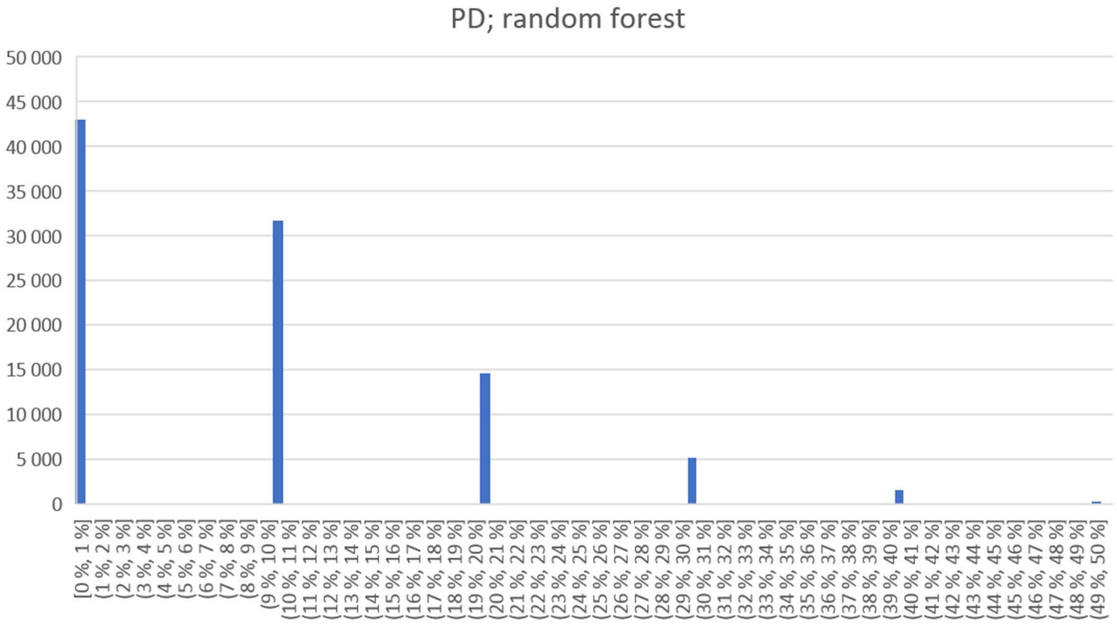


Figure 6: Histogram of default probabilities assigned to loans by random forest

The portfolio constructed with the logistic PD model seems to perform slightly better (+0.3% realized return over the random forest model). Overall the results of the two models are very similar. While the logistic regression provided continuous probability values, the random forest results contain only 5 discrete values. The diversification value is nearly the same with the number of investments chosen being very close at 238 vs 286. The investments by state differ as can be seen from the Appendix.

Discussion and managerial implications

RESULTS

We created two default probability estimation models – logistic regression and random forest. These models allow us to estimate the default probabilities of Lending Club loans. We tested the models by creating two portfolios that maximize the whole-portfolio return estimate. Realized returns on the test set were 21.29% for the logistic regression model and 20.95% for the random forest model (compared to 10.87% for the Lending Club whole-universe realized return).

LIMITATIONS

In the preprocessing part, we imputed missing values based on average and common values, which may introduce some errors into the data. Similarly, the manually-engineered features may introduce bias into the dataset. Furthermore, with 90 variables expanded from 20 original variables, the chance of overfitting also increased.

For the modeling part, logistic regression is built on two basic assumptions that the data observations are independently distributed, and that the predictive variables also should not be highly correlated. The first assumption depends on data collection schema, which is beyond our control in this case. As we see from the correlation heatmap in Figure 3, there are 2 pairs of highly correlated variables. In the modelling process, since we use regularization terms, the negative effect of this violation is mitigated.

First of all, random forest model is time-consuming to train and test, which may come into disadvantage in some loan evaluation scenarios when the portfolio has to be built in a timely fashion. Due to the best split mechanism of random forest, it tends to favor common occurrences. However, for default prediction, which is closer to outlier detection in nature, random forest tends to ignore those rare occurrences and outputs zero default probabilities. This leads to over-optimistic, risk-seeking portfolio decisions. Last but not least, random forest can also overfit to noisy data.

Whereas portfolio optimization is concerned, it should be noted that these return numbers are based on historical results and offer no guarantees for future returns. It was assumed that 40% of loan value can be retrieved in case of default. This number may fluctuate highly depending on location and state of the economy. The bankruptcies were estimated to take place 1 year after the loan is given. It is likely that the mean time to default will differ for different-term loans (24 vs 36 months). The correlation of LGD and mean time to default should be studied over full business cycles to determine whether adverse effects in these factors are likely to combine. No transaction costs were considered. It was also assumed that all loans are available to our investor (no preferred investors or other limitations).

BUSINESS OPPORTUNITIES

Our work provides a good basis for consumer loan investment analysis in US. These findings could be used directly to invest in the US consumer loans market (via Lending Club or other means). Alternatively, the methods presented here could be used as a basis for further work in developing default probability models (to be applied for example in bank-side mortgage rate assessment). Due to the growing popularity of peer-to-peer finance, these results should be further tested on other less-known platforms that share their data, such as Bondora. There is likely less competition and the realized return of a well-optimized portfolio might be higher. Finally, portfolio optimization could be sold as a service to the many peer-to-peer lending enthusiasts.

References

Diderich, C. (2009). Positive Alpha Generation: Designing Sound Investment Processes.

European Banking Authority. (2016). Guidelines on PD estimation, LGD estimation and the treatment of defaulted exposures. Retrieved from https://eba.europa.eu/documents/10180/1659311/Consultation+Paper+on+Guidelines+on+PD+LGD+estimation+and+treatment+of+defaulted+assets+%28EBA-CP-2016-21%29.pdf

Guangyou, Z., Zhang, Y., & Luo, S. (2018). P2P Network Lending, Loss Given Default and Credit Risks. Sustainability.

Appendix

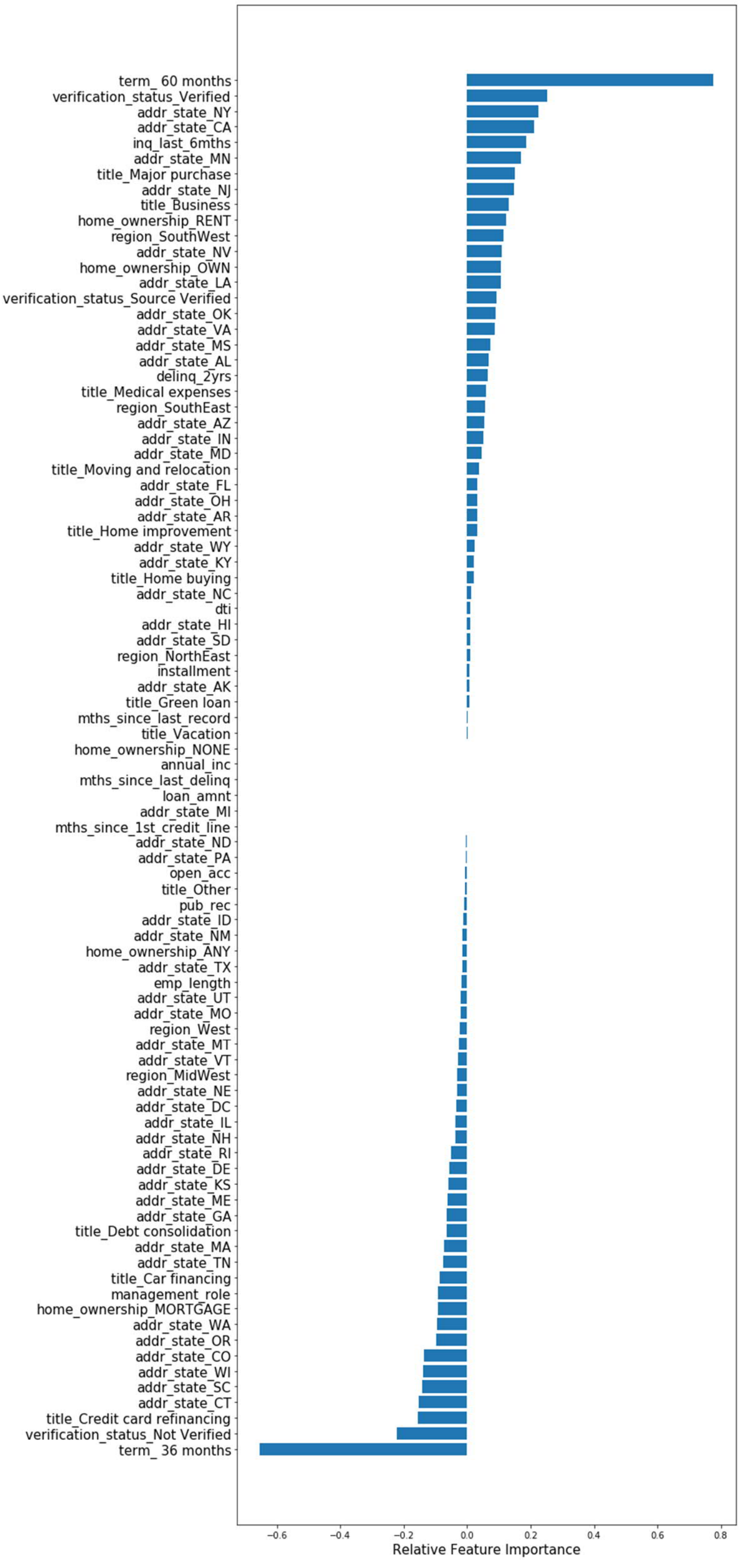


Figure 7: Relative Feature Importance for Logistic Regression

Table 8: Investment by state, logistic regression

State	Investment	State	Investment
AL	0	NE	30 325
AZ	0	NV	94 700
AR	16 000	NH	0
CA	240 525	NJ	0
CO	500 000	NM	0
CT	176 000	NY	197 500
DE	85 950	NC	39 200
FL	0	ND	48 800
GA	0	OH	395 025
HI	252 400	OK	180 275
ID	183 325	OR	91 775
IL	0	PA	57 000
IN	25 000	RI	208 150
IA	384 700	SC	40 000
KS	21 925	SD	42 200
KY	51 000	TN	0
LA	16 000	TX	48 000
ME	120 325	UT	434 150
MD	156 675	VT	0
MA	187 050	VA	211 800
MI	30 675	WA	0
MN	177 425	WV	144 000
MS	44 000	WI	0
MO	40 925	WY	11 200
MT	16 000	Total	5 000 000

Table 9: Investments by state; random forest

State	Investment
AL	0
AZ	57 000
AR	61 600
CA	104 225
CO	500 000
CT	115 225
DE	147 950
FL	14 625
GA	28 000
HI	392 775
ID	102 550
IL	16 800
IN	18 375
IA	258 225
KS	56 700
KY	0
LA	19 025
ME	114 400
MD	69 800
MA	208 125
MI	34 425
MN	257 775
MS	37 700
MO	67 600
MT	0
NE	0
NV	90 000
NH	21 175
NJ	16 000
NM	9 500
NY	182 750
NC	0
ND	52 775
OH	467 625
OK	199 150
OR	12 000
PA	33 200
RI	205 625
SC	14 175
SD	49 650
TN	0
TX	48 000
UT	401 200
VT	74 500
VA	147 000
WA	16 000
WV	207 975
WI	68 800
WY	0
Total	5 000 000