Improving Risk Detection for Peer-to-Peer Lending

Xuyin Zhu
Supervisor: Dr. Shi Zhou

MSc in Web Science & Big Data Analytics
Department of Computer Science
University College London

September 2016

This report is submitted as part requirement for the MSc in Web Science & Big Data Analytics at University College London. It is substantially the result of my own work except where explicitly indicated in the text.

Abstract

This project aims to improve risk detection for peer-to-peer lending by machine learning algorithms and network analysis. Online peer-to-peer lending industry has experienced a fast growth and drawn more and more attention in the past few years. However, a high default rate of around 20% still exists despite all the efforts made by peer-to-peer lending platforms.

In this project, machine learning algorithms combined with lending network analysis are employed to detect the loans that are likely to default. Instead of reviewing each borrower individually, which is the most common way used by financial institutions, this project uses machine learning algorithms to extract the pattern from historical data and use it to make predictions on new loan requests. The result turns out that the default rate can be significantly improved by as much as 60% by using even simple machine learning algorithms.

Furthermore, after adding network topological features as extra inputs to machine learning models, the performance is further improved. The result suggests that the graphical properties of the lending network provide implicit information which can be useful for risk detection.

Acknowledgements

I would like to show my sincere gratitude to my supervisor, Dr. Shi Zhou, for his supervision, advice and encouragement throughout this project. I really appreciate the time he spent on meetings and discussions with me, from which I have learnt a lot to keep improving my work.

I also would like to thank my friends for their supports during the last years, giving me the courage to face all the difficulties in study and daily life.

Contents

1. Introduction

Online peer-to-peer lending marketplace is a financial platform which provides unsecured microloans by matching individuals who want to borrow and invest money. It has experienced fast development in the past few years. However, a high default rate in the marketplace makes the lenders suffer a great loss.

Machine learning is a subfield evolved from computational learning theory and pattern recognition in artificial intelligence area. It has been widely used to automatically produce models that can quickly analyze more complex data and deliver more accurate results on a large scale.

Complex network theory is a part of network work theory, the study of representing relations between discrete objects by graphs. Complex network specifically refers to the graphs with non-trivial topological features, which often occur in networks of real systems. It is a relative young study but has been applied in various areas like computer science, sociology and biology.

In this project, machine learning algorithms combined with lending network analysis are employed to detect the loans that are likely to default. Firstly, popular machine learning algorithms are used to built different classification models for risk detection. The models are trained and tested on real data and their performances are evaluated by different methods. After that, the network topological features are used as extra inputs to classification models to explore the influence of network factor on models' performance.

The result shows that over 60% of defaulted loans can be detected by machine learning algorithms and performance can be improved at most by 4% after adding network features to the models.

2. Background

2.1 Peer-to-peer lending

First emerging in 2005, online peer-to-peer lending marketplace is a financial platform which provides unsecured microloans by matching individuals who want to borrow and invest money. In this kind of lending marketplace, borrowers can submit loan requests, which are also called lists, while lenders can choose lists to invest. Once the money required by a list is fully funded within a certain period, the list will officially become a loan. Normally, a loan is funded by dozens of lenders since it is a wiser strategy for lenders to spread the risk across different loans. Compared to traditional financial institutions, online peer-to-peer lending offers lenders with more investment channels with higher returns and borrowers can get loans at respectively lower interest rates. Since all the operations are entirely online without the participation of any intermediary, service fee is less and the whole procedure is more time efficient.

At present, there are more than 30 online peer-to-peer lending platforms in more than 10 countries (Xu, Qiu, & Lin, 2011) such as Zopa in UK, Prosper and LendingClub in US, Grouplend in Canada, SocietyOne in Australia and PPDai in China. Due to its benefits and flexibility, peer-to-peer lending industry has experienced a fast growth and been turned out to be a great success in the past few years. LendingClub, the largest peer-to-peer lender in US, has issued 1.6 million loans for over $20 billion in total as of 2016, followed by Prosper, which has issued $6 billion in total. In a report from PWC, peer-to-peer lending industry has been described as experiencing 'tremendous growth' and the loans issued by it are predicted to reach $150 billion by 2025 (PWC, 2015). In another economic research report, the industry is expected to obtain a market share of 10% for retail lending worldwide (Egham, 2008). Harvard Business Review even has made the prediction that every major commercial bank will start its own peer-to-peer lending business in the next decade (Harvard Business, 2009).

The investment on peer-to-peer lending seems to be attractive and rewarding, however, it presents a significant risk. Despite all the efforts claimed by those peer-to-peer lending companies that have been put into detecting risky borrowers, the default rate in the marketplace is still high. An economic research has reported that the default rate on customer loans of all commercial banks in Q1 2014 is 2.32% (Fred, 2016). While the default rate of LendingClub at the same period of time is as high as 13.02% (LendingClub, 2016).

One explanation for the high default rate may be the inherent risk of loans made via online platforms to strangers without any collateral (Yum, Lee, & Chae, 2012). Lower interest rates and less restrictions attract a large number of borrowers with bad credits to stay in the market and seek for loans. However, that's not the whole story. Unlike commercial banks, which make profits from interests of loans, peer-to-peer lending companies generate revenue by charging borrowers and investors service fee. And they don't take any responsibility for the defaulted loans. If a loan defaults, the lender has to bear all the losses by himself. Based on this fact, a reasonable suspicion can be made that those peer-to-peer lending companies may relax the requirements for each borrower on purpose since they don't want to loss any potential customers.

2.2 Risk assessment based on credit score

Normally, when financial institutions or online lending platforms receive a loan application from a borrower, they will run a review process to decide whether to offer the loan or not. Many factors would be considered during the review such as the credit history, employment status and current income of the borrower. Among them, borrower's credit score is the most fundamental and determined characteristic that lenders will concentrate on.

Credit score is calculated using the information in a borrower's credit report, which is collected and kept by credit reference agencies. Once the borrower submits a loan request, the lender will be granted the permission to review his credit report from credit reference agencies. Instead of reading through all the details in the credit report, credit score is used as a numerical representation of borrower's credit situation that can help the lender make his decision more efficiently.

There are various algorithms for calculating the credit score. The one that is most widely used by the majority of commercial banks and credit grantors is called FICO score. It is made up of five components as follow:

FICO score	
1. Payment history:	Whether the borrower pays his loans on time
2. Amounts owed:	The amount of money the borrower owes
3. Length of credit history:	The age of the oldest account and the average account age
4. Credit mix:	The variety of accounts
5. New credit:	Recently opened accounts

Table 2.1 Components of FICO score

The following figure shows the percentage of each component that will be considered when calculating the FICO score based on their importance:

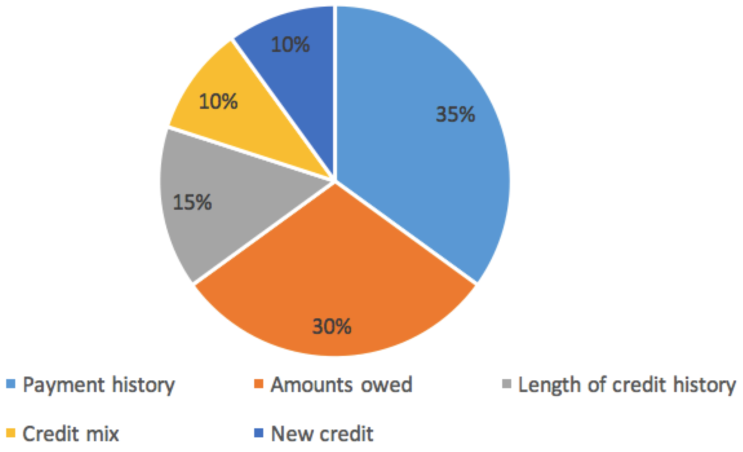


Figure 2.1 Proportion of five components

The range of a FICO score is from 300 to 850. Borrowers with higher FICO scores usually have more chances to get their loans funded. According to the five components above, to obtain a good FICO score, a borrower should have more on time payments, less amount of owed money, a relatively longer credit history, a stronger mix of credit accounts and less recently opened accounts.

2.3 Machine learning

Machine learning is a subfield evolved from computational learning theory and pattern recognition in artificial intelligence area. It was defined as “Field of study that gives the computer the ability to learn without being explicitly programmed” by Arthur Samuel in 1959 (Simon, 2013). It explores to construct algorithms which can learn from data and make predictions (Kohavi, & Provost, 1998). These algorithms are used to build models from example inputs, which are also called training data, and make predictions driven by data instead of following specific program instructions.

Machine learning can be divided into three main tasks, which are supervised learning, unsupervised learning and reinforcement learning. In terms of supervised learning, the learning algorithm is given the example inputs as well as their corresponding outputs, which are also known as labels. While in the case of unsupervised learning, labels are not provided to the algorithm. Reinforcement learning refers to the interaction between the computer program and a dynamic environment where the program has to perform a certain task like learning to play a game by playing against an opponent.

Although many machine learning algorithms have already been proposed for decades, it is a recent development for them to be widely applied in many practical issues. The resurging interest in machine learning is due to the growing volumes of available data, powerful computational processing ability and affordable data storage. All these factors make it possible to apply machine learning algorithms to automatically produce models that can quickly analyze more complex data and deliver more accurate results on a large scale.

The value of machine learning has been widely recognized by industries that work with a large amount of data. For example, in marketing and sales, websites recommend you products by analyzing your purchasing history. This ability of machine learning to capture, analyze and use data to personalize the shopping experience of each customer has become a trend in retail industry. In oil and gas industry, machine learning is used to analyze minerals in the ground, stream oil distribution and predict refinery sensor failure and the use cases are still expanding. In health care industry, with the development of wearable devices and sensors, more available data can be used by machine learning to assess patients' health condition in real time. It also has been employed by some medical experts to analyze the spread pattern of a disease and identify trend that can improve the treatment. In fact, the application of machine learning has covered every aspect of human life.

2.4 Network analysis

2.4.1 Complex network theory

Complex network theory is a part of network work theory, the study of representing the relations between discrete objects by graphs (Alhajj, & Rokne, 2014). Complex network specifically refers to the graphs with non-trivial topological features, which often occur in networks of real systems. The study of complex network is relatively young, mainly inspired by the empirical study of real-world networks such as biological networks and social networks.

Scale-free network and small-world network are the two most well known and studied complex networks. A scale-free network is defined to be a network with a power-law degree distribution, which implies the distribution has no characteristic scale. The power-law distribution exists in various areas including physics, biology and social activities. For example, the size of craters on the moon, the frequencies of words in most languages, the number of flights of different routes, they all follow a power-law degree distribution. A small-world network refers to a network where most nodes are able to be reached by other nodes by only a few steps. A well know example of small-world network is the collaboration network of actors.

The study of complex network has attracted researchers from many areas like computer science, mathematics, sociology and biology (Albert, & Motter, 2016). It has been applied to many practical issues such as developing vaccination strategies to control the disease, identifying bottlenecks in city traffic, designing scalable communication networks and analyzing metabolic and genetic regulatory networks.

2.4.2 Network properties

2.4.2.1 Node degree

In network theory, the most basic attribute of a node is called degree, which is the number of edges linked to the node. In a directed graph, the node degree can be divided into in-degree and out-degree. In-degree refers to the number of edges coming into the node and out-degree refers to the number of edges coming out from the node.

When considering the whole network, degree distribution is used to describe the probability distribution of degrees over the network. Degree distribution is the most important property that characterizes the structure of the network. It is represented by the probability of a randomly chosen node having degree *k*:

$$P(k) = \frac{N_k}{N}$$

where N_k is the number of nodes with degree *k*, *N* is the total number of nodes in the network.

For example, random graph, which is the simplest network model, has a binomial degree distribution that each of *n* nodes is connected with probability *p*:

$$P(k) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

In a random network with binomial distribution, most nodes have similar degrees. This is very different from real-world networks. In most real-world networks, a small number of nodes will have very large degrees and the rest of them have relatively small degrees. The degree distribution of this kind of networks is a power law:

$$P(k) \sim k^{-\gamma}$$

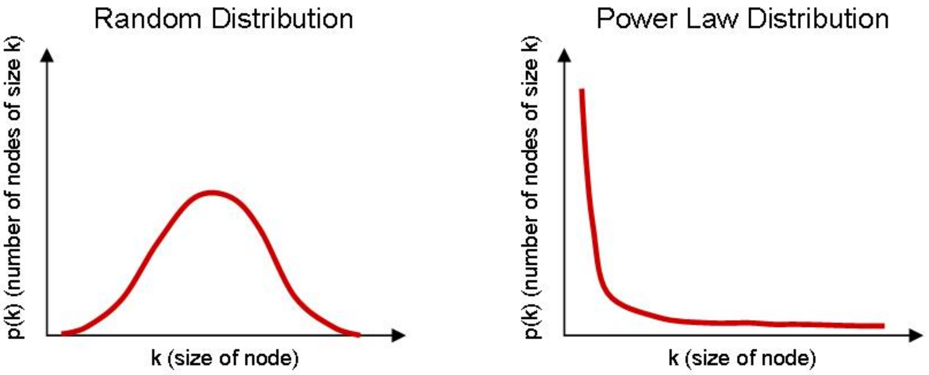


Figure 2.2 Binomial and power-law distribution

2.4.2.2 Centrality

In graph theory, centrality is a measure that determines the important nodes in the network. It gives each node a real value to provide a ranking of nodes in terms of their importance. The word ‘importance’ has a wide range of meanings. For example, it may mean the influential persons in a social network, the key infrastructures in the Internet, or the spreaders of a disease. These different meanings lead to different definitions of centrality.

There are two categorizations that have been proposed to characterize centrality, which are by network flow and by walk structure. A network can be described as a series of paths. The characterization by network flow is generated based on the type of the path which is encoded by centrality. By contrast, the characterization by walk structure considers how the centrality is constructed. This can be divided into two classes, which are radial and medial. Radial centrality calculates the number of walks that start from or end at the given node such as degree centrality. Medial centrality calculates the number of walks that pass through the given node such as betweenness centrality. In addition, the calculation is able to capture either the volume of walks or the length of walks. The volume of walks refers to the total number of walks of a certain type, for example, walks of length one. Degree centrality and betweenness centrality fall into this category. While the length of walks refers to the total distance from the given node to any other node in the graph. Closeness centrality is calculated based on the length of walks.

Two common centrality measures used in analyzing the structure of the network are betweenness centrality and closeness centrality. Betweenness centrality quantifies how many times a node is passed through by the shortest path between two other nodes. It can be represented as follow:

$$C_B(i) = \sum_{j<k} \frac{g_{jk}(i)}{g_{jk}}$$

where g_{jk} is the number of shortest paths between node *j* and node *k*, $g_{jk}(i)$ is the number of shortest paths between node *j* and node *k* passing through node *i*. The formula above can be further normalized by the number of pairs of nodes in the network excluding node *i*:

$$C'_B(i) = \frac{C_B(i)}{(n-1)(n-2)/2}$$

Closeness centrality can be defined as the inverse of farness, which refers to the sum of a node's shortest paths to all other nodes in the graph. It is represented as follow:

$$C_C(i) = \frac{1}{\sum_{j=1}^{N} d(i,j)}$$

According to the formula of closeness centrality, the shorter a node's distance to all other nodes, the more central it is in the graph. This property of the node can be viewed as its efficiency in spreading information over the network. Note that the concept of distance is considered as a more meaningful centrality measure in directed graphs than in undirected graphs. Normalized closeness centrality is represented as follow:

$$C'_C(i) = C_C(i)(N-1)$$

3. Literature survey

3.1 Risk detection based on machine learning

In financial industry, many models have been built by machine learning algorithms to detect risky borrowers. The common idea of them is to set a threshold on the probability of a loan being defaulted and reject applications that are below the threshold. These models are usually called credit scoring models. The most common methods used for credit scoring are discriminant analysis and logistic regression. They are widely used in practical issues and have shown good performance. Apart from them, many other competitive methods have been proposed in the past years, including non-parametric statistical algorithms such as k-nearest neighbors and soft computing algorithms such as neural network, support vector machines, survival models and genetic programming.

Neural network is an especially useful alternative when there are complicated non-linear relationships between different features. Research group led by Malhotra has applied the adaptive neuro-fuzzy inference system and back propagation network on the loans of credit unions in US to built credit scoring models (Malhotra, 2002). David West has compared the performance of five neural network models on detecting risky borrowers: multi-layer perceptron, mixture-of-experts, radial basis function, learning vector quantization and fuzzy adaptive resonance (West, 2000).

Support vector machine is another popular machine learning algorithm in credit scoring. Research group led by Gestel has proposed least squares support vector machine classifiers within the Bayesian evidence framework to analyze the creditworthiness of borrowers (Gestel, et al., 2006). Yang has applied an incremental kernel method to construct a novel and practical adaptive credit scoring system, which can be adjusted according to an update procedure and can converge to the optimal solution without information loss (Yang, 2007).

Furthermore, research group led by Ong has employed genetic programming to automatically determine the adequate discriminant functions and the valid attributes to provide better performance in credit scoring (Ong, Huang, &Tzeng, 2005). Another research group led by Noh has applied survival analysis on the information from credit card center in South Korea, which uses time-dependent variables when building the credit scoring model to improve performance (Noh, Roh, & Han, 2005).

3.2 Lending behaviors based on network analysis

Some researchers study the problem from a different aspect. They focus on people's behaviors and consider the relationships among borrowers and lenders as a social network. Social factors have been analyzed to learn their influences on lenders' behaviors to help them make better investment decisions.

Research group led by Dawei Shen has modeled the behaviors of lenders by preferential attachment and fragmentation. The result shows the existence of significant herding effect when lenders make their decisions, which indicates the lenders in peer-to-peer lending marketplace tend to make irrational investment decisions by following the herd instead of considering the risks and returns (Shen, Krumme, & Lippman, 2010).

Another research led by Katherine Krumme also has observed the herd effect in peer-to-peer lending marketplace. Furthermore, they have proved that financial indicators are correlated with social factors such as descriptive profile texts and the participation in groups (Krumme, & Herrero-Lopez, 2009).

Some other studies focus on the distinction between structural aspects and relational aspects of the social network and their influences on the lending outcomes. And they have drawn the conclusion that structural aspects have limited and even no significance on predicting lending outcomes while relational aspects are turned out to be strong predictors. More verifiable and stronger relational measures indicate a higher likelihood of a loan being funded and a lower risk of it being defaulted (Lin, Prabhala, & Viswanathan, 2013).

Considering the fact that most previous studies on the relationship between social capital and peer-to-peer lending performance are limited to the western context. Yun Xu and his team has conducted a compared study of the role played by social capital in peer-to-peer lending in U.S and China (Xu, Qiu, & Lin, 2011). After comparing the different communities and different cultures, they have found that the influence of social capital is not equally important. More specifically, social capital is more influential for a loan to get funded in Chinese market while it affects more on the interest rates in U.S market.

4. Project plan

4.1 Problem statement

Attracted by the lower interest rates and higher returns, more and more people are seeking for loans or investing their money in peer-to-peer lending nowadays. However, despite the fast development and increasing transaction volume of the industry, actions that have been taken to detect risky borrowers in order to reduce default rate are very limited.

The high default rate is actually within the expectation due to the differences between online peer-to-peer lending and commercial bank lending. Firstly, online lending has less constraints on a borrower's qualification to request for loans. As long as a borrower has a valid id and a bank account with fine credit records, he is qualified to submit loan applications. Secondly, all the loans in online peer-to-peer marketplace are unsecured, which means lenders don't have any insurance of their money. Lastly, the platforms don't take any responsibility for defaulted loans, all the losses have to be beard by lenders themselves.

Despite all the risky elements mentioned above, it is surprising that almost all the online peer-to-peer lending companies still use traditional credit score to determine whether to approve a loan request or not, which is apparently not sufficient in this scenario. As the weak group in the marketplace, lenders should be provided with more powerful risk detection tools to protect their rights.

4.2 Objectives

For all the borrowers who are presently active on peer-to-peer lending platforms, they are already considered as "trustworthy" by the platforms. However, there is still a large number of risky borrowers among them. The objective of this project is to provide lenders and managers of the platforms a better way to detect them.

The risk detection problem can be simply regarded as a binary classification task. For a loan application from a borrower, the main task is to make a prediction that the loan will complete or default. From the prediction, the decision can be made that whether to accept or reject the loan application.

In this project, some popular machine learning algorithms will be used to build different classification models for risk detection. The models will be trained and tested on real data to evaluate the performance. After that, the network topological features will be used as extra inputs to classification models to explore the influence of network factor on models' performance.

Although machine learning has been used in a variety of practical issues in the past few years, the lending industry, no matter peer-to-peer lending or commercial bank lending still stick to the traditional ways for risk assessment. There is every reason to believe that machine learning algorithms will greatly improve the result of risk detection compared to traditional methods, which is what this project aims to prove. Therefore, the objective of the project is to use as simple and common ways as possible to prove the point of view instead of pursuing the best performance by fancy and complicated algorithms.

4.3 Data

4.3.1 Overview

The data used in this project is from the loan transactions occurred on Prosper, the second biggest online peer-to-peer lending platform established at the end of 2005 in U.S.

From 2006 to 2008, Prosper adopted a variable interest rate model, which used an online Dutch auction system to determine the interest rate. Under this model, a borrower can post a listing with the amount he needs and the maximum interest rate he can afford. While an investor has access to all the listings and he can select any of them to bid with the interest rate he is willing to offer. If a listing is fully funded within a certain period, it will be materialized into a loan and the investor with lowest bid will be chosen by the system. With the variable rate model, interest rates are completely determined by the borrowers and lenders through auctions. However, it led to a significant default rate before 2009 since a large number of desperate borrowers with low credits offered high interest rates that they couldn't afford in order to attract investors. From July 2009, Prosper has adopted a new business model that all the interest rates are preset solely by the platform with stricter restrictions. Only borrowers with a FICO credit score higher than 640 have the qualification to apply for loans. A formula will be applied on each loan to assign an interest rate based on the borrower's credit report and financial situation. An investor only has the choice to decide whether to invest or not in a listing with preset interest rate. After applying this new model, the default rate has significantly decreased.

There are two separate datasets used in this project, which are the loan data and the lending network data. The original loan dataset includes all the loans on Prosper from November 2005 to March 2014. While the original lending network dataset includes all the loans on Prosper from November 2005 to September 2011. Considering the significant influence made by the change, only data from July 2009 to September 2011 is selected to make sure all the loans are issued under the new interest rate model.

4.3.2 Loan data

The loan data of Prosper is open to the public, various versions can be obtained from online resources. The loan dataset used in the project is from this website (Joash, 2015). The original dataset consists of loans originated from November 2005 to March 2014. Due to the reason mentioned above, only data from July 2009 to September 2011 is selected for the project, including 14,629 loan records in total.

Each loan of a certain borrower has 81 features, which can be divided into financial information, personal characteristics and the current situation of the loan. Financial information includes credit rating, borrower rate, estimated loss, bankcard utilization, etc. Personal characteristics include member key, employment status, loan purpose, whether the borrower is a home owner, etc. The current situation of the loan includes all the information related to this loan by the time the data is collected, including monthly payment, loan months since origination, any delinquency, etc. Furthermore, each loan has a feature called “status” to indicate whether the loan is completed, defaulted, in current or past due.

4.3.3 Network data

The network dataset is provided by a research group (Redmond, & Cunningham, 2013), consisting of all the loans from November 2005 to September 2011 represented in a graph. Again, only data from July 2009 to September 2011 is selected for the project, including 32,154 nodes and 1,399,202 edges in total.

The network is an unweighted directed graph. Each node represents a member of Prosper, which can be either a borrower or a lender. Each directed edge represents a loan relationship between two nodes, which points to a borrower from a lender. It should be noticed that in the Prosper marketplace, a loan is typically funded by many different lenders. Therefore, normally more than two nodes are involved in a single loan. Apart from source node and target node, each edge has 4 additional attributes to describe the loan, which are the timestamp when the loan was issued, borrower rate, borrower’s credit rating and loan amount.

4.3.4 Combination of two datasets

The two dataset are obtained from different research sources. There is no direct connection between them. Fortunately, the lending network data is provided along with some crucial information like originated time, loan amount, credit rating and borrower rate which can help to identify each loan and connect them with the records in the loan dataset.

First, all completed loans are selected from the loan dataset. Only considering completed loans here is because the final status of an undergoing loan is unknown, which means it doesn't have the label for models to train. Then for each completed loan in the loan dataset, the loan with same timestamp, borrower rate, loan amount and credit rating of the borrower in the lending network is considered as one same loan. However, the situation can happen that on the same day, two borrowers with same credit rating borrow the same amount of money with the same interest rate. These loans that can't be identified are removed from the dataset.

After the processing mentioned above, a subset of loan data with 9853 loans is used as the final dataset in this project.

4.4 Methodology

4.4.1 Machine leaning

There are three machine learning algorithms used here to build classification models, which are linear discriminant analysis, logistic regression and multi-layer perceptron. They are popular and common methods used for classification tasks and have achieved good performance in practice. Unlike most traditional ways for risk detection, which are based on reviewing each borrower separately, machine learning explores the pattern from historical data. Therefore, it can analysis the problem from the perspective of the whole population instead of each individual.

4.4.1.1 Linear discriminant analysis

Linear discriminant analysis is generalized from Fisher's linear discriminant, which was first proposed by Ronald Fisher in 1936. It is widely used for dimensionality reduction in statistics, machine learning and pattern recognition. Another major application of linear discriminant analysis is to characterize objects into two or multiple classes by finding a linear combination of continuous independent variables.

When implementing linear discriminant analysis, there are some statistical assumptions that need to be satisfied. Firstly, the data has to be independent and represents a sample from a normal distribution. Secondly, the covariance matrix of variables is required to be homogeneous across groups. Therefore, linear discriminant analysis is actually a special case with stronger assumptions, which should have restricted its application. However, it has been reported to be more robust even when the assumptions are violated (Sánchez, & Sarabia, 1995) (Marcucci, 1997).

Derived from simple probabilistic models, linear discriminant analysis models conditional distribution of the data for each class k as $P(X|y=k)$. Then Bayes' rule is applied to make the prediction, where X is a set of observations (also can be called as variables, features or attributes) and y is the predicted class.

$$P(y=k|X)=\frac{P(X|y=k)P(y=k)}{P(X)}=\frac{P(X|y=k)P(y=k)}{\sum_l P(X|y=l)\cdot P(y=l)}$$

The class k that can maximize the conditional probability is selected as the prediction result.

More specifically, linear discriminant analysis models $P(X|y)$ as a multivariate Gaussian distribution with density. It assumes that the conditional probability density functions are both normally distributed with mean parameters μ and covariance parameters Σ.

$$P(X|y=k)=\frac{1}{(2\pi)^n|\Sigma_k|^{1/2}}exp\left(-\frac{1}{2}(X-\mu_k)^t\sum\nolimits_k^{-1}(X-\mu_k)\right)$$

The model can be used as a classifier by estimating the class priors $P(y=k)$ (by the proportion of instances of class k), the covariance matrices (either by a regularized estimator or by the empirical sample class covariance matrices) and the class means μ_k (by the empirical sample class means) from the training data.

4.4.1.2 Logistic regression

Logistic regression is one of the most widely used statistical tools for classification problems. By estimating the probabilities with a logistic function, it measures the relationship between independent variables and a categorical dependent variable. One big advantage of logistic regression compared to other linear models is that it suits various kinds of distribution like normal, Poisson and Gamble distribution.

In a binary classification task, given an example x, the prediction $y \in \{0,1\}$, so the hypotheses $h_\theta(x)$ is applied to predict the probability that the example belongs to class 1 versus the probability that it belongs to class 0:

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

where:

$$\theta^T x = \theta_0 + \theta_1 x_1 + \theta_2 x_2 +, \dots, + \theta_n x_n = \theta_0 + \sum_{j=1}^{n} \theta_j x_j$$

and:

$$g(z) = \frac{1}{1 + e^{-z}}$$

is the logistic function, which is also called sigmoid function.

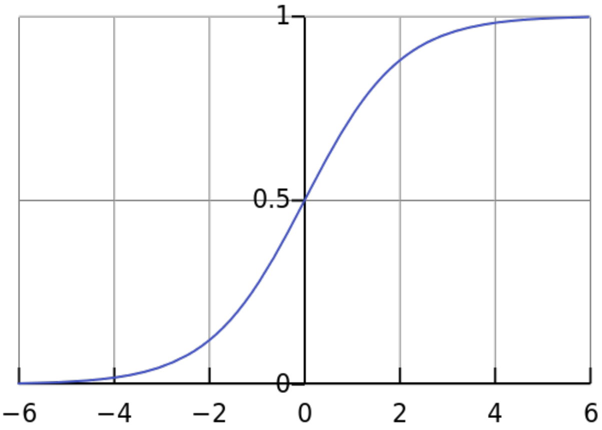


Figure 4.1 Sigmoid function

The sigmoid function $g(z)$ tends to 1 as $z \rightarrow \infty$ and to 0 as $z \rightarrow -\infty$. Therefore, the values of $g(z)$ and $h(x)$ can be bounded between 0 and 1. Other functions also can be used as the logistic function as long as they increase smoothly from 0 to 1. However, sigmoid function has a useful property of its derivative:

$$g'(z) = \frac{d}{dz}\frac{1}{1+e^{-z}} = g(z)(1-g(z))$$

The probabilistic assumptions of the model can be endowed as follow:

$$P(y=1|x;\theta) = h_\theta(x),\ P(y=0|x;\theta) = 1-h_\theta(x)$$

Which can be described more compactly as:

$$p(y|x;\theta) = \left(h_\theta(x)\right)^y\left(1-h_\theta(x)\right)^{1-y}$$

Assuming that the m training examples are independently generated, the likelihood of the parameters can be written as:

$$L(\theta) = p(\vec{y}|X;\theta) = \prod_{i=1}^{m} p\left(y^{(i)}|x^{(i)};\theta\right) = \prod_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)\right)^{y^{(i)}}\left(1-h_\theta\left(x^{(i)}\right)\right)^{1-y^{(i)}}$$

Log likelihood is used since it is easier to be maximized:

$$\ell(\theta) = logL(\theta) = \sum_{i=1}^{m} y^{(i)}logh\left(x^i\right) + \left(1-y^{(i)}\right)log\left(1-h\left(x^i\right)\right)$$

Gradient ascent is used to maximized the likelihood. Consider one training example (x, y) and take derivatives to derive the stochastic gradient ascent:

$$\frac{\partial}{\partial\theta_j}\ell(\theta) = \left(y-h_\theta(x)\right)x_j$$

And the updates of the parameters can be obtained as follow based on the fact that $g'(z) = g(z)(1-g(z))$:

$$\theta_j := \theta_j + \alpha\left(y^{(i)} - h_\theta\left(x^{(i)}\right)\right)x_j{}^{(i)}$$

4.4.1.3 Multi-layer perceptron

The concept of artificial neural network was first inspired by humans' central nervous system. It was developed to imitate the neurophysiology of the human brain. It is a class of statistical models that contain sets of adaptive weights and particularly capable of finding the complicated non-linear relationship in the data by tuning the weights using different learning algorithms.

In an artificial neural network, the basic elements are called neurons, which are connected together to form the architecture of the network. The typical architecture of an artificial neural network can be represented as a three-layer system, which is constituted by input layer, hidden layer (one or more) and output layer. The input layer is used to receive and process the input data vector. The hidden layers receive the output from the previous layer and calculate the weights by the activation function such as sigmoid function, Heaviside step function and hyperbolic tan function. The output layer receives the result produced by the final hidden layer and outputs the target value. Following figure shows a fully connected three-layer neural network:

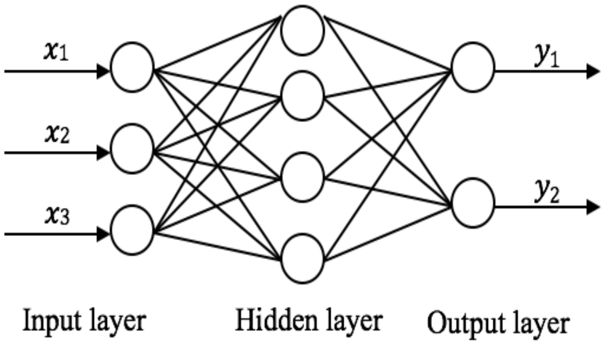


Figure 4.2 Three-layer neural network

A multi-layer perceptron is a feed-forward neural network that is made up of multiple layers of nodes, where each layer is fully connected with each other. It can be considered as a modification of linear perceptron that is able to distinguish non-linearly separable data. Each node in the network is called a perceptron with a nonlinear activation function. Each perceptron calculates a single value from multiple inputs by applying a linear combination on the input weights, which can be mathematically represented as follow:

$$y = \varphi\left(\sum_{i=1}^{n} \omega_i x_i + b\right) = \varphi(w^T x + b)$$

where x is the input vector, w is the weights, b is the bias and φ is the activation function. Following figure shows the signal flow of a single perceptron:

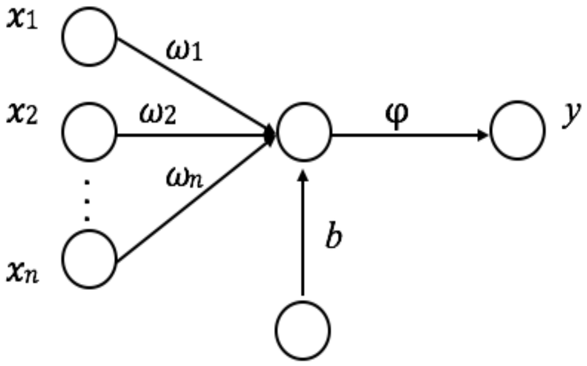


Figure 4.3 Signal flow of a single perceptron

A multi-layer perceptron with one hidden layer and a linear output layer can be similarly represented as follow:

$$y = B\varphi(Ax + a) + b$$

where x is the input vector, A is the weight matrix of the first layer, a is the bias vector of the first layer. While B is the weight matrix and b is the bias vector of the second layer respectively.

In fact, the power of a multi-layer perceptron with only one hidden layer is surprisingly large (Hornik, Stinchcombe, & White, 1989). It can approximate any continuous function $f: R^n \to R^m$ when provided with sufficient hidden units.

4.4.2 Network analysis

From the literature survey, it can be learned that all the machine learning methods for risk detection only use the most common and basic information like financial history and personal characteristics as inputs to the model. However, since the lending network is available, network features associated with each borrower may provide implicit information that is useful in improving the performance of models.

In this project, a borrower's role in the lending network is analyzed by network topological properties like in and out degree, betweenness centrality and closeness centrality. Then these properties will be added as extra inputs to train the classification models.

4.5 Tasks

The project can be divided into two main tasks. In the first task, the loan data will be used as training set to train the models built by three different machine learning algorithms, which are linear discriminant analysis, logistic regression and multi-layer perceptron. And the performance of each model will be evaluated by various evaluation methods. In the second task, the properties extracted from the lending network will be used as extra inputs to train the new models with network features. The same evaluation methods will be applied on the performance of new models to find out if network properties have a positive or negative influence on risk detection.

5. Task1: Risk detection based on loan-related features

5.1 Data

5.1.1 Data properties

The loan data consists of all the completed loans originated from July 2009 to September 2011. Among these 9853 loans, 8005 are completed and 1848 are defaulted. The default rate of the dataset is around 18.76%. In the Prosper marketplace, each borrower is assigned a credit rating from AA to HR based on his FICO score. AA represents an excellent credit score while HR represents a high risk. If calculate the default rate based on each credit rating, a distribution as follow can be observed:

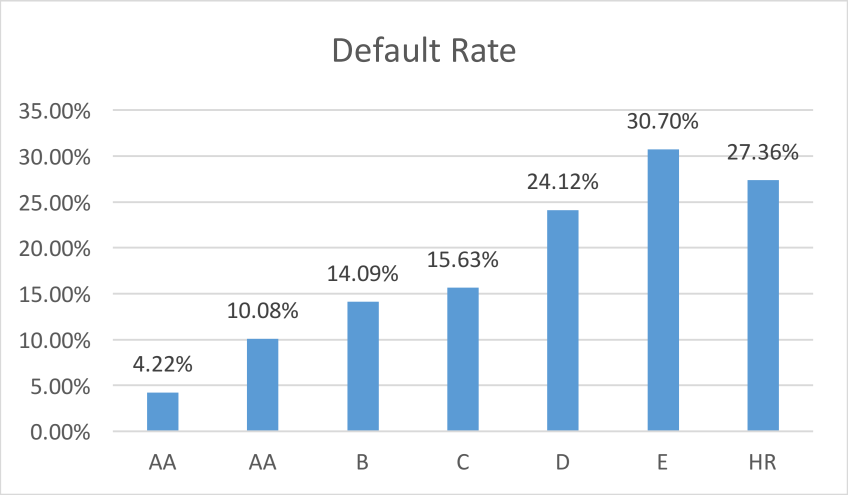


Figure 5.1 Default rate distribution

5.1.2 Data processing

5.1.2.1 Categorical variables

The loan data is made up of many categorical features such as occupation, income range and employment status. Actually, most real-word data contains both categorical and numerical variables. However, some classification models like multi-layer perceptron and k-nearest neighbors require the data to be purely numerical. Therefore, the first step of data processing is to convert all the categorical variables into numerical forms. For each categorical variable, according to the number of its possible values, it is simply encoded with the value from 0 to the number minus 1.

5.1.2.2 Missing values

There are some missing values existing in the loan data which are encoded as blanks. Missing values are incompatible with the estimators of some models which assume all values should have meanings. A simple strategy to deal with this problem is to discard the all rows that contain missing values. However, this may lead to the loss of valuable data even though it is incomplete. What’s more, when the dataset is not big enough, dropping too much data may cause a substantial loss of the statistical power. Therefore, imputing the missing values is a better strategy. Three most commonly used imputing strategies are using the median, mean or most frequent value in the column where the missing values are located. In this project, the mean value is used to imputing the missing values.

5.1.2.3 Feature scaling

In some machine learning algorithms, feature scaling is a common requirement for the objective functions to work as expected since the value range of the raw data varies widely. For instance, most classification algorithms use Euclidean distance to calculate the distance of two objects. If a particular feature has a wide range, the distance will be significantly influenced by it. Thus, it is important to transform the data to the same scale before applying any algorithm.

There are two popular methods of feature scaling, normalization and standardization. Data normalization refers to rescale the feature to lie between a minimum and a maximum value, which are usually zero and one. Data standardization means shifting the distribution of the feature so that it has a mean value of zero and a standard deviation of one.

In this project, normalization is used for feature scaling as follow:

$$x_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

where x_{min} and x_{max} is the value range of the feature.

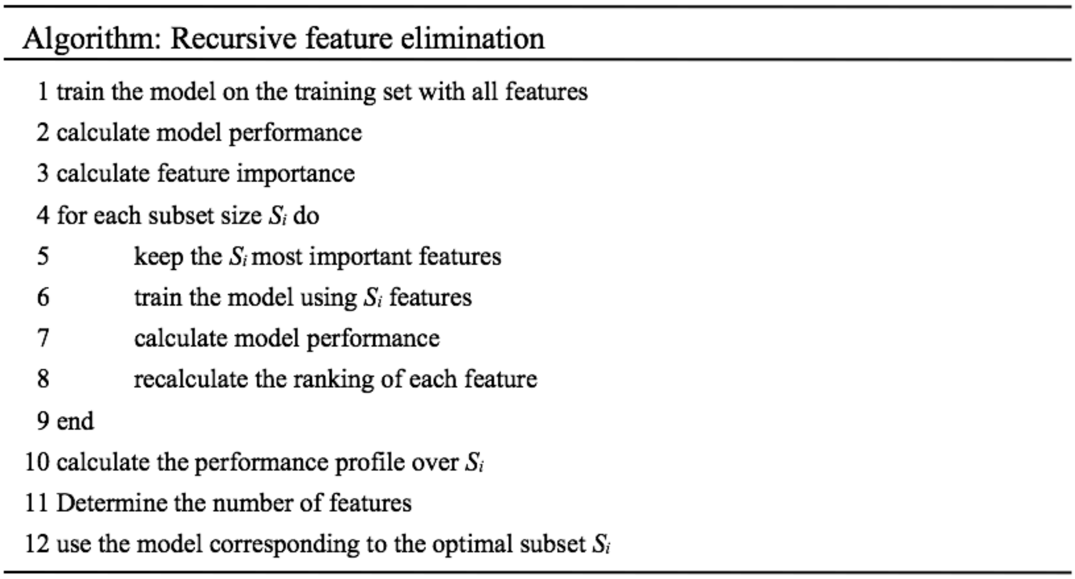
5.2 Feature selection

5.2.1 Recursive feature elimination

Feature selection is the process of selecting relevant features for machine learning algorithms. The major purpose of feature selection is to enhance accuracy of the model, shorten learning time and reduce overfitting. It is especially useful when the dataset contains a large number of redundant or irrelevant features.

Feature selection is a combination of search technique and evaluation measure. Search technique is used to find possible subsets of features and evaluation measure is used to score different subsets. In this project, recursive feature elimination algorithm is used for feature selection.

The idea of recursive feature elimination is recursively removing the features with lower weights which are assigned by the external estimator such as coefficients of a linear model. First, the algorithm fits the model to all features. Each feature is ranked by its importance to the model. Assume *S* is the number of features to retain ($S_1>S_2\cdots$), at each iteration of the procedure, only top S_i features are retained while others are removed from the training set. Then, the model is refit and the performances of the features retained from last iteration are reassessed. This selection procedure is repeated until a certain number of features is reached.



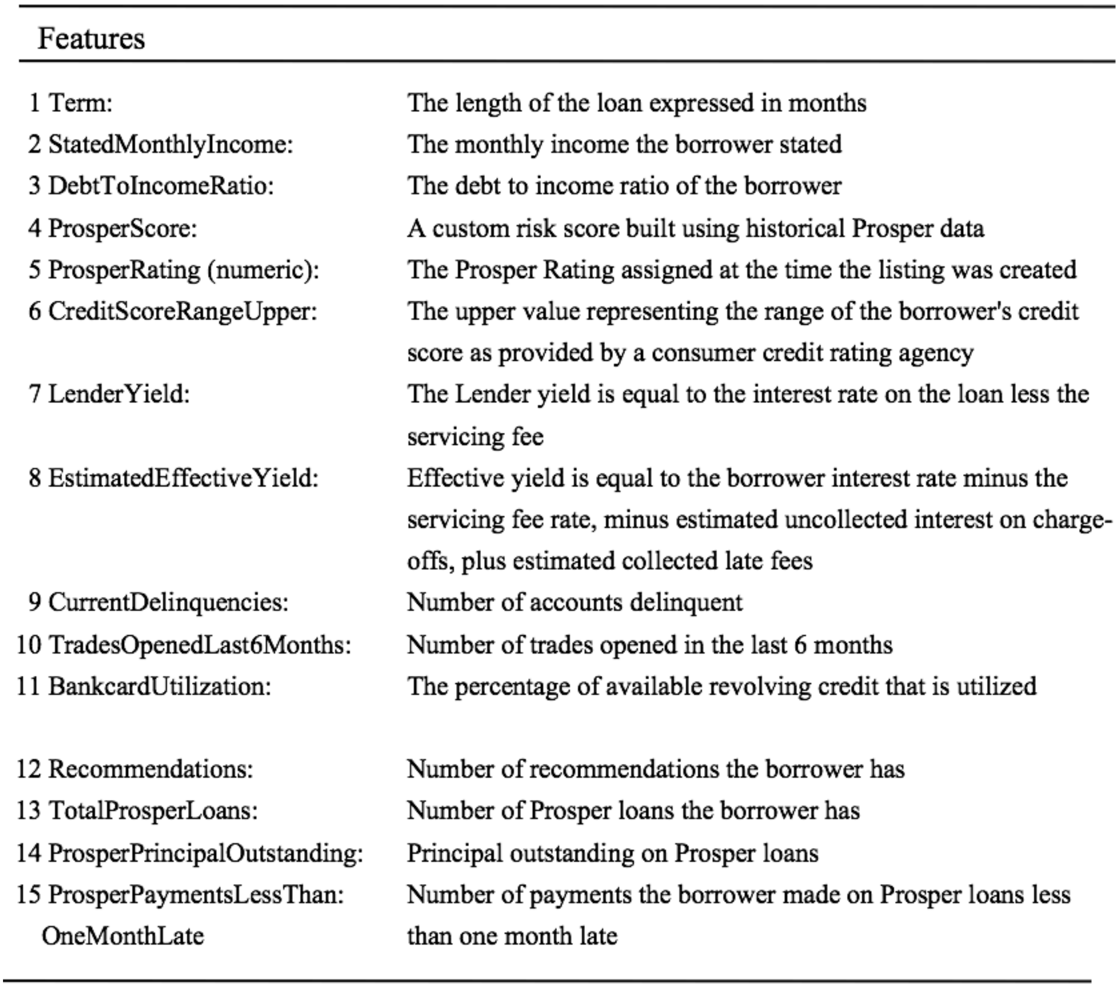
```
Algorithm: Recursive feature elimination
train the model on the training set with all features
calculate model performance
calculate feature importance
for each subset size $S_i$ do
    keep the $S_i$ most important features
    train the model using $S_i$ features
    calculate model performance
    recalculate the ranking of each feature
end
calculate the performance profile over $S_i$
Determine the number of features
use the model corresponding to the optimal subset $S_i$
```

Table 5.1 Procedure of recursive feature elimination

5.2.2 Selected feature set

There are 81 features associated with each loan. Before applying the recursive feature elimination algorithm, a preliminary selection can be done. First, features with distinct value for each loan are removed such as member key and the date when a loan was originated. Then, features used to describe loan status after it has been funded are removed such as monthly payment, loan months since origination, etc. Since such features can only be observed after a loan starts.

After eliminating all useless features described above, there are 48 features left that can be used as the initial feature set of the recursive feature elimination algorithm. Since the intention of this paper is trying to use simple models to prove the assumption, a smaller size of feature set is preferred in order to reduce the complication. Therefore, on the premise that all important information can be kept, the required feature number is set as 15. The final 15 features obtained from recursive feature elimination algorithm are as follow.



Features	
1 Term:	The length of the loan expressed in months
2 StatedMonthlyIncome:	The monthly income the borrower stated
3 DebtToIncomeRatio:	The debt to income ratio of the borrower
4 ProsperScore:	A custom risk score built using historical Prosper data
5 ProsperRating (numeric):	The Prosper Rating assigned at the time the listing was created
6 CreditScoreRangeUpper:	The upper value representing the range of the borrower's credit score as provided by a consumer credit rating agency
7 LenderYield:	The Lender yield is equal to the interest rate on the loan less the servicing fee
8 EstimatedEffectiveYield:	Effective yield is equal to the borrower interest rate minus the servicing fee rate, minus estimated uncollected interest on charge-offs, plus estimated collected late fees
9 CurrentDelinquencies:	Number of accounts delinquent
10 TradesOpenedLast6Months:	Number of trades opened in the last 6 months
11 BankcardUtilization:	The percentage of available revolving credit that is utilized
12 Recommendations:	Number of recommendations the borrower has
13 TotalProsperLoans:	Number of Prosper loans the borrower has
14 ProsperPrincipalOutstanding:	Principal outstanding on Prosper loans
15 ProsperPaymentsLessThan: OneMonthLate	Number of payments the borrower made on Prosper loans less than one month late

Table 5.2 Feature set

Apart from 15 features above, each loan is assigned with a label indicating the class it belongs to. For all completed loans, the value of the label is 1, while for all defaulted loans, the value of the label is 0.

5.3 Class weight balance

Until now, all the data processing and feature selection has been done. If use the processed data as the input of the three model introduced above, a perfect result of more than 80% accuracy can be easily obtained. However, if look into the prediction result, a frustrating fact can be found that almost all the samples in the test set are labeled as the majority class happening in the training set. This problem is caused by imbalanced dataset, where the classes are not represented equally.

In fact, the problem of imbalanced data is expected in practice of various areas such as medical diagnosis, fraud detection and oil spillage detection, where the number of one class greatly surpasses the number of other classes. In this project, majority loans are ‘completed’ and only less than 20% of loans are ‘defaulted’. Therefore, the classifier cleverly decides to label almost every sample as ‘completed’ and still can achieve a good performance. A simple way to deal with this problem is assigning different weights to different classes when training classification models.

$$class\_weight_i = \frac{n_{samples}}{n_{classes} n_{samples\_class_i}}$$

where $n_{samples}$ is the number of all samples, $n_{classes}$ is the number of classes, $n_{samples\_class_i}$ is the number of samples with class i. The formula above adjusts the weight of each class inversely proportional to class frequencies in the input data.

5.4 Experiment

5.4.1 Tool and setting

In the first task, three machine learning algorithms, linear discriminant analysis, logistic regression and multi-layer perceptron need to be implemented. Here, the models are built in python using PyCharm, an integrated development environment used for programming in python.

The linear discriminant algorithm is implemented by scikit-learn, which is a machine learning library for programming in Python. The function LinearDiscriminantAnalysis of class discriminant_analysis is employed by the project to build the classifier with a linear decision boundary by fitting a Gaussian density to each class and using Bayes' rules.

The logistic regression algorithm is also implemented by scikit-learn using the function LogisticRegression of class linear_model, which uses liblinear library, newton-cg and lbfgs solvers to implement regularized logistic regression.

Multi-layer perceptron is implemented by Keras, which is a highly modular neural networks library that runs on top of TensorFlow and Theano. The multi-layer perceptron model has three layers. The input layer and hidden layer have 80 and 60 hidden units respectively, using rectifier as activation function. The output layer uses sigmoid activation function to scale the result into range zero to one.

5.4.2 Input and output

For all three models, the inputs are processed 9853 loans with 15 selected features. And the expected outputs are 9853 continuous values from zero to one, each representing the probability of the loan to be defaulted.

5.5 Result

5.5.1 Cross validation

When assessing the result of a prediction model, the learning and testing process should not be run on the same dataset. Otherwise, the model would just repeat the true label of each sample and use it as the predictive label. This situation is called overfitting, which will lead to the consequence that the model has a perfect accurate score but fails to make any useful prediction on yet-unseen data.

To avoid this problem, cross validation is used here on the dataset before applying the models. By cross validation, the dataset is partitioned into two complementary subsets. Part of the available data is held out as the test set for evaluation, while the learning process is performed on the training set. Here, 80% of the data is used to train the models and the left 20% of the data is used for testing.

5.5.2 Result on test data

After splitting the data into training set and testing set, the testing data contains 1971 loans, among which 1612 are completed while 359 are defaulted. The output is in the form of probability. To make it easier to compare the prediction with the true condition, if the probability is above threshold 0.5 then the loan is considered as 'completed', otherwise defaulted. The predictions on the testing data of each model is shown as follow:

Linear discriminant analysis			
	True completed	True defaulted	Sum
Predicted completed	1369	238	1607
Predicted defaulted	243	121	364
Sum	1612	359	1971

Table 5.3 Result of linear discriminant analysis

Logistic regression			
	True completed	True defaulted	Sum
Predicted completed	1042	112	1154
Predicted defaulted	570	247	817
Sum	1612	359	1971

Table 5.4 Result of logistic regression

Multi-layer perceptron			
	True completed	True defaulted	Sum
Predicted completed	1413	226	1639
Predicted defaulted	199	133	332
Sum	1612	359	1971

Table 5.5 Result of multi-layer perceptron

However, since the predicted label is determined by the setting of threshold. The tables above can only provide a rough overview of the results of three models compared to true condition. More detailed analysis will be discussed in the part of evaluation.

5.6 Evaluation

5.6.1 Accuracy

Accuracy is a basic statistical measure of how accurate a binary classification model can identify an entity. It can be simply obtained by calculating the proportion of correctly predicted results among the number of all samples that have been predicted.

$$accuracy(y,\hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(\hat{y}_i = y_i)$$

However, in binary classification, the output prediction of each sample is a probability instead of a class label. The classier would simply set a threshold value, usually 0.5 to determine the class of the sample. For example, if the output probability is above 0.5, then the sample belongs to the positive class, and if the probability is below 0.5 then the sample belongs to the negative class. Therefore, the accuracy highly depends on the probability distribution of the prediction result which is determined by the nature of the data. The accuracies of linear discriminant analysis, logistic regression and multi-layer perceptron under different thresholds are shown as follow:

Model			Threshold		
	0.40	0.45	0.50	0.55	0.60
Linear discriminant analysis	76.92%	76.36%	75.60%	75.09%	74.23%
Logistic regression	73.82%	70.27%	65.40%	60.88%	55.00%
Multi-layer perceptron	80.16%	79.76%	77.42%	73.52%	67.02%

Table 5.6 Accuracy under different thresholds

From the table above, it can be learned that the overall accuracy varies from different cut points. Although by choosing the best cut point of each model, multi-layer perceptron seems to show a best performance, it is hard to compare the three models under a consistent condition. In addition, accuracy is sensitive to the size of different classes. When there is a skewed class distribution in the dataset, a high accuracy may only imply the overfitting to a certain class. Therefore, only using accuracy is not enough. A better evaluation measure should be employed to assess the performance of models.

5.6.2 Receiver operating characteristic curve

Consider a binary classification problem, the prediction of an instance can be labeled as either positive or negative. Therefore, the prediction outcomes of a classifier can be divided into four cases. If the predicted label and the true label are both positive, then it is a true positive. If the predicted label is positive while the true label is negative, then it is a false positive. If the predicted label and the true label are both negative, then it is a true negative. At last, if the predicted label is negative while the true label is positive, it is a false negative. These four cases can be represented by the following confusion matrix:

		Predicted condition	
	Total population	Positive	Negative
True condition	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Table 5.7 Confusion matrix

Receiver operating characteristic curve plots true positive rate (TPR) and false positive rate (FPR) in a graph to show the the performance of a binary classification model at various thresholds. In binary classification, the prediction outcome of an example is made on a continuous variable. Given a threshold, the instance is labeled as positive if the continuous variable is above the threshold, and negative otherwise. The continuous variable *X* follows a probability density $f_1(x)$ if the true label of the instance is positive, and $f_0(x)$ otherwise. Thus, the true positive rate and the false rate can be represented as follow:

$$\mathrm{TPR}(T) = \int_T^\infty f_1(x)dx$$

$$\mathrm{FPR}(T) = \int_T^\infty f_0(x)dx$$

Where *T* is the threshold. ROC curve plots $\mathrm{FPR}(T)$ as *x* axis and $\mathrm{TPR}(T)$ as *y* axis with threshold *T* as the varying parameter.

Below is a ROC curve by random guessing, which is represented as a diagonal line from coordinate (0,0) to (1,1). It can be used as a baseline for assessing the performance of a model. Any prediction above the line is considered a good result, otherwise it implies a poor performance even worse than random.

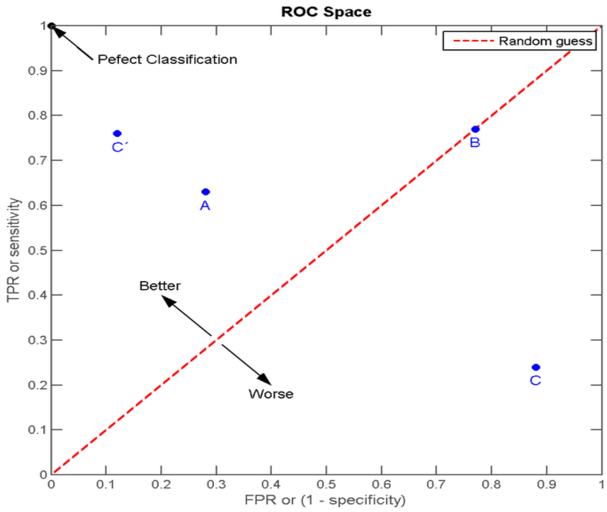


Figure 5.2 ROC of random guess

It reasonable to use a single value to summarize the ROC curve, which is AUC, the area of the convex shape below the ROC curve. AUC can be given by:

$$A = \int_{\infty}^{-\infty} TPR(T)FPR'(T)dT = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(T' > T)f_1(T')f_0(T)dT'dT = P(X_1 > X_0)$$

The figures below show the ROC curves of linear discriminant analysis, logistic regression and multi-layer perceptron respectively:

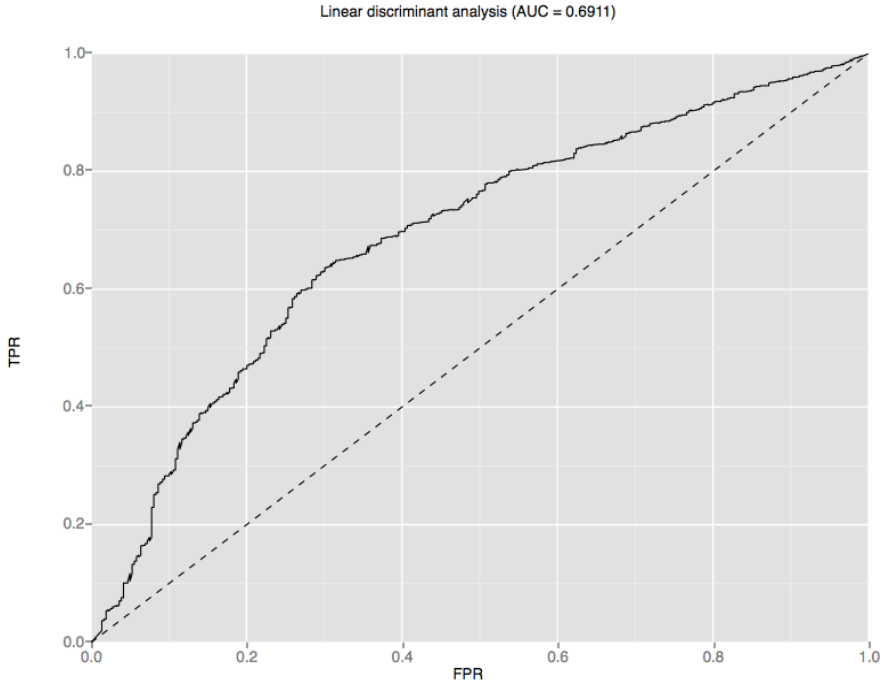


Figure 5.3 ROC of linear discriminant analysis

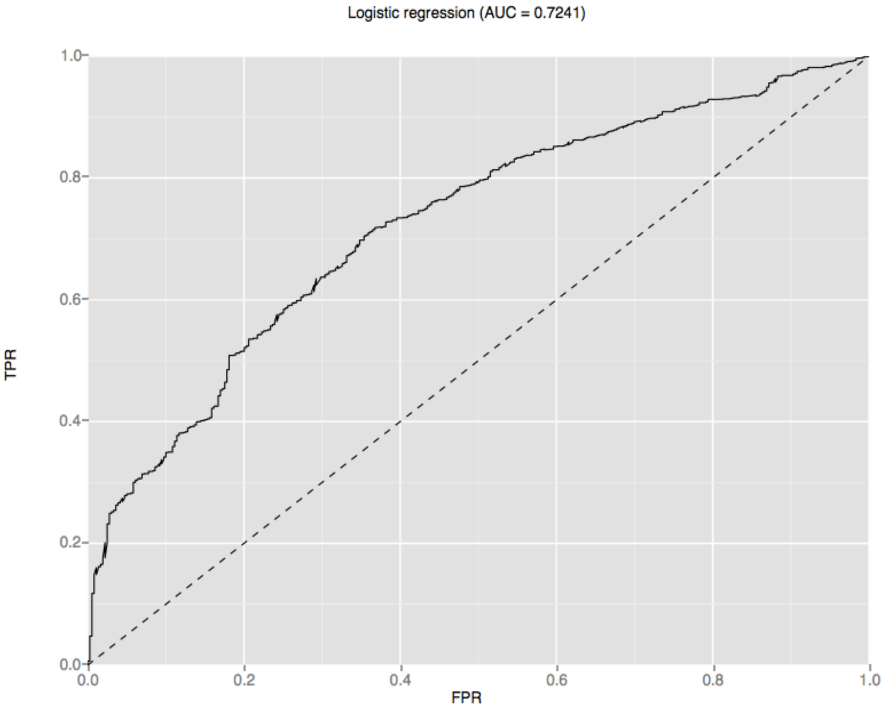


Figure 5.4 ROC of logistic regression

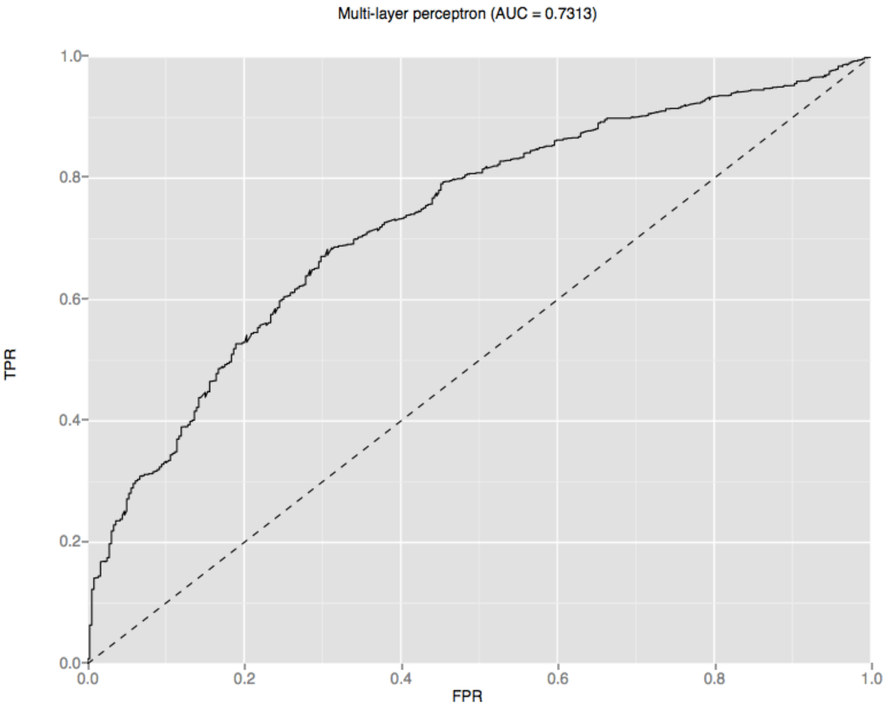


Figure 5.5 ROC of multi-layer perceptron

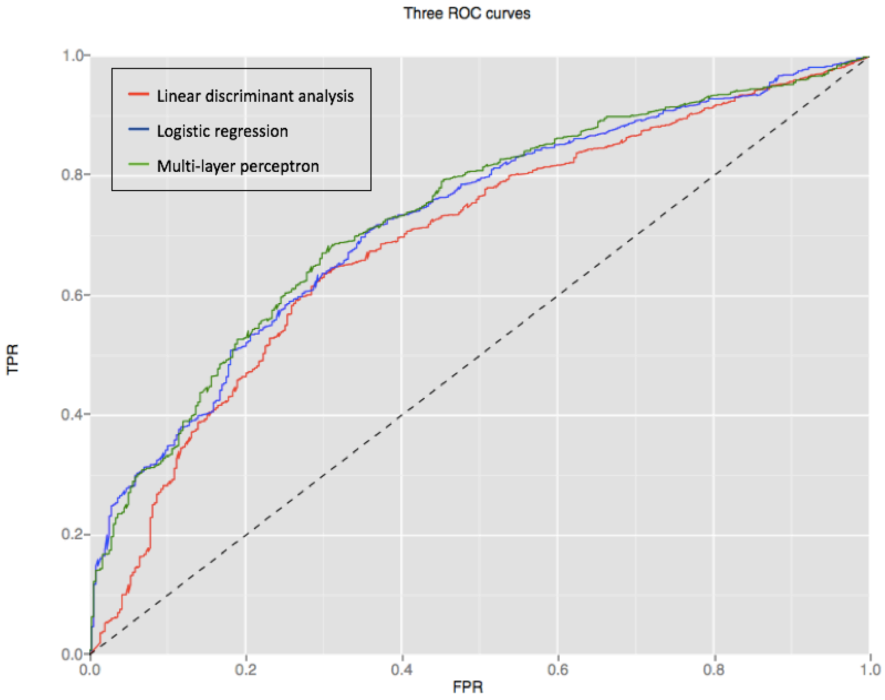


Figure 5.6 ROC of three models

The table below shows the AUC values of linear discriminant analysis, logistic regression and multi-layer perceptron respectively:

Model	AUC
Linear discriminant analysis	69.11%
Logistic regression	72.41%
Multi-layer perceptron	73.13%

Table 5.8 AUC of three models

From the ROC and AUC analysis above, it can be learned that multi-layer perceptron and logistic regression show better performance than linear discriminant analysis. However, the difference between multi-layer perceptron and logistic regression is not very obvious.

5.6.3 Precision and recall of defaulted class

The evaluation made above is from the perspective of the whole performance of a model. In practice, the ability of the classier to detect defaulted loan is more essential.

Take the dataset used in this project as an example, all 9853 loans were approved by the Prosper, which means the borrowers of the loans were considered ‘good’ and were believed that they would pay the loan on time. However, according to the result of loan payment, there are around 20% of borrowers that failed to pay their loan. The objective of the classier is to find as many risky borrowers as possible from these 20%. While how many ‘good’ borrowers that the classier can correctly identify is less important. Therefore, the performance of detecting risky borrowers is evaluated separately by the precision and recall of defaulted class.

For the defaulted class, precision is the probability that a detected risky loan is truly risky while recall is the probability that a truly risky loan is detected. Based on the confusion matrix introduced above, they can be represented respectively as follow:

$$Precision = \frac{TN}{TN + FN}$$

$$Recall = \frac{TN}{TN + FP}$$

The precision and recall of defaulted class is shown as follow:

Model	Precision	Recall
Linear discriminant analysis	31.14%	59.33%
Logistic regression	33.39%	60.45%
Multi-layer perceptron	33.23%	59.33%

Table 5.9 Precision and recall of three models

From the table above it can be learned that, in terms of the ability to detect risky loans, logistic regression and multi-layer perceptron still outperform linear discriminant analysis. The precision and recall of logistic regression is even slightly higher than multi-layer perceptron.

6. Task 2: Risk detection based on network-related features

6.1 Data

The lending network data consists of the loans originated from July 2009 to September 2011. All the loans no matter they are completed or uncompleted are considered when building the network and all related calculations are based on the complete network. However, the properties of 9853 completed loans introduced above will be especially analyzed and used in task 2.

The network includes 32154 nodes and 1399202 edges. A node can represent a lender or a borrower. And if there is a loan relationship between them, they will be connected by an edge pointing from lender to borrower. A loan can be funded by several lenders, and each lender has to pay at least 25$ on the loan. A typical graphic representation of a loan is as follow:

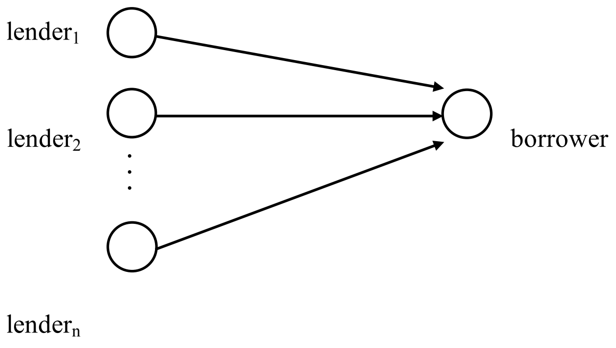


Figure 6.1 Graphic representation of a loan

The average degree of the network is around 87. It should be noticed that the degree doesn't equal to the number of loans that an individual is involved in. As a borrower, he can have a large in-degree for one loan, for he is funded by many investors. The degree distribution is shown as following:

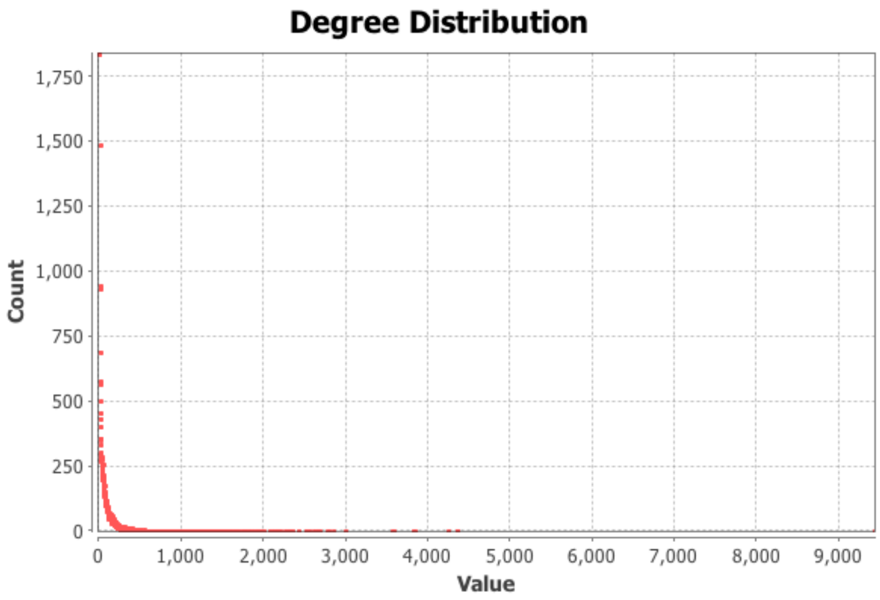


Figure 6.2 Degree distribution

The degree distribution of the lending network is a power law, which means a small number of people have been involved in many loans while most people only have one or two loans whether as a lender or a borrower. Let's look into the distribution of in-degree and out-degree respectively:

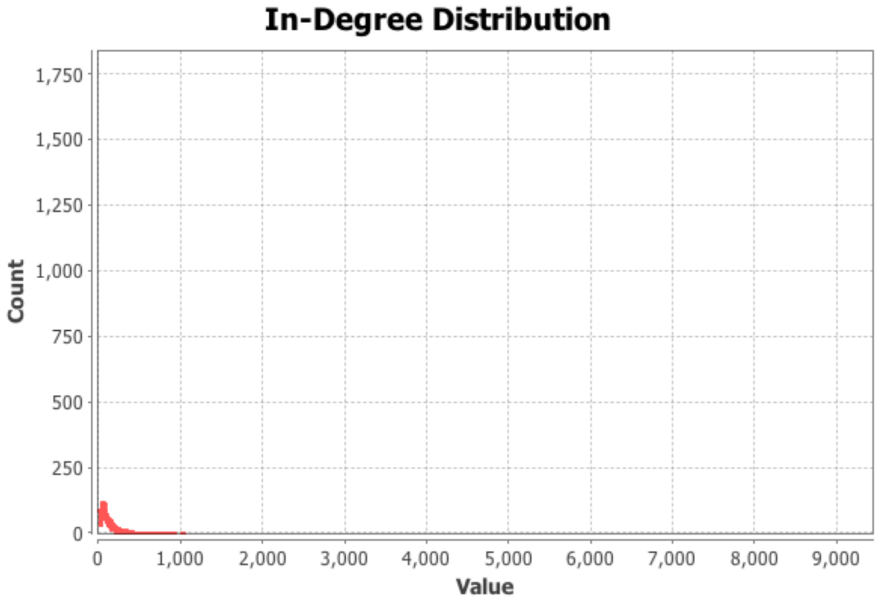


Figure 6.3 In-degree distribution

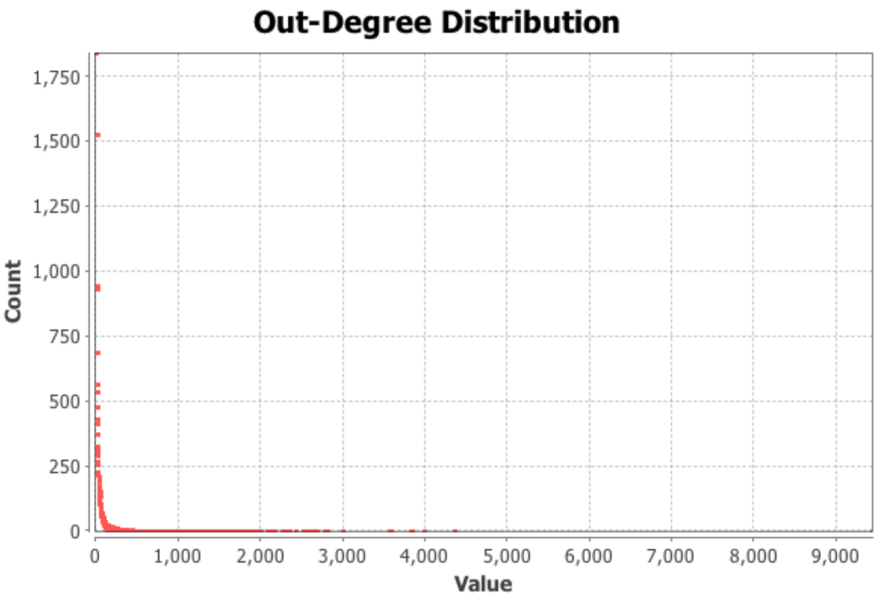


Figure 6.3 Out-degree distribution

The in-degree distribution doesn't follow a power law. In fact, most people have an in-degree around 100. This is because most borrowers only have one loan and the number of investors of a loan is usually around 100. The out-degree distribution follows a power law, implying that a small number of people lend their money to many other borrowers. These group of people can be considered as purely investors in the market.

6.2 Feature selection

There are four network properties are selected as additional features of machine learning algorithms, which are in-degree, out-degree, betweenness centrality and closeness centrality.

They are chosen because they are the most common properties and describe a node from different perspective. In addition, it is an intuitive thought that if an individual is a borrower and a lender at the same time, which means he has both in-degree and out-degree, then it is less likely for him to default. His true aim is not to borrower money but to make profit from interest rate difference by borrowing money at a lower interest and lending it at a higher interest. Same for betweenness centrality and closeness centrality. If a person is in the center of the network, it means the cash flows through him. He acts more like an intermediary in the network, so the chance of him to be a risky borrower is relatively low.

However, the direct relation between these four features and risk detection still remains unclear. So the objective of task 2 is to prove the hypothesis that network properties have a positive effect on risk detection.

6.3 Experiment

6.3.1 Tool and setting

Task 2 involves two parts of lending work analysis. First part is the visualization of network structure and properties, which is implemented by Gephi, an interactive platform to visualize and explore networks, complex systems and dynamic graphs. The second part is to calculate the four properties introduced above for each borrower. This part is implemented by Networkx in python programming language, which is a Python package to create and study the structure of complex networks.

6.3.2 Input and output

The input is the whole network, where each instance is represented by source id, target id, timestamp, amount, borrower rate and credit. The output is the in-degree, out-degree, betweenness centrality and closeness centrality of each node. And the nodes which represent the borrower of 9853 loans will be selected along with their network properties, which will be used as network features of the models.

6.4 Evaluation

6.4.1 Receiver operating characteristic curve

After adding lending network features, the ROC curves of three models are as follow:

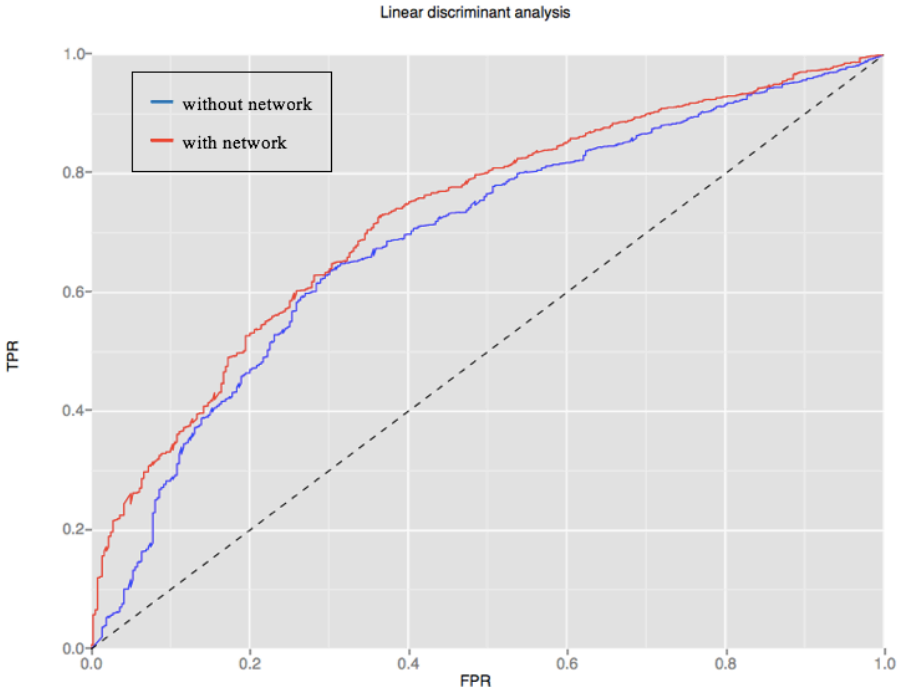


Figure 6.5 ROC of linear discriminant analysis

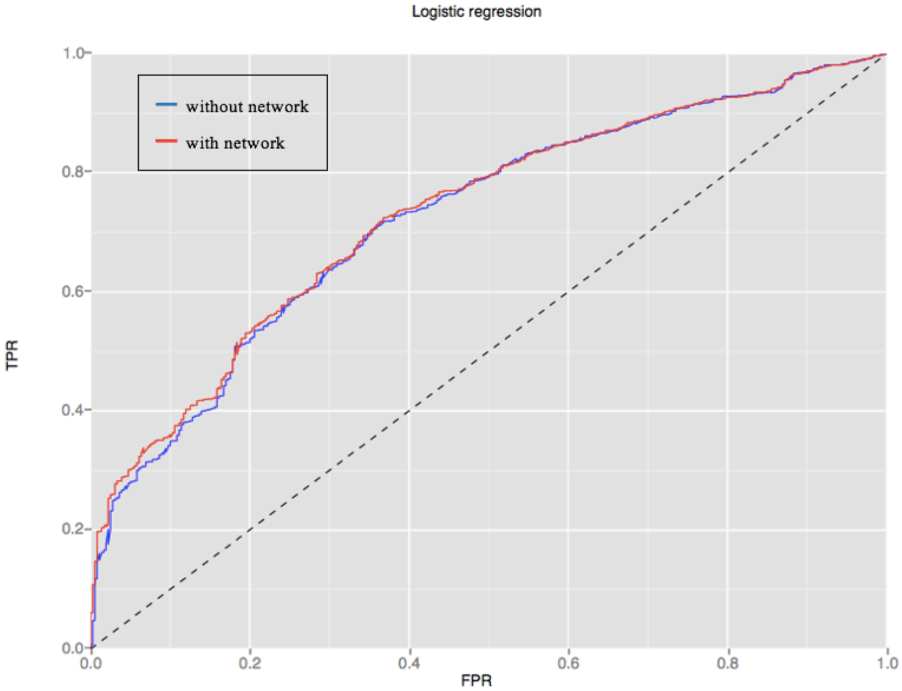


Figure 6.6 ROC of logistic regression

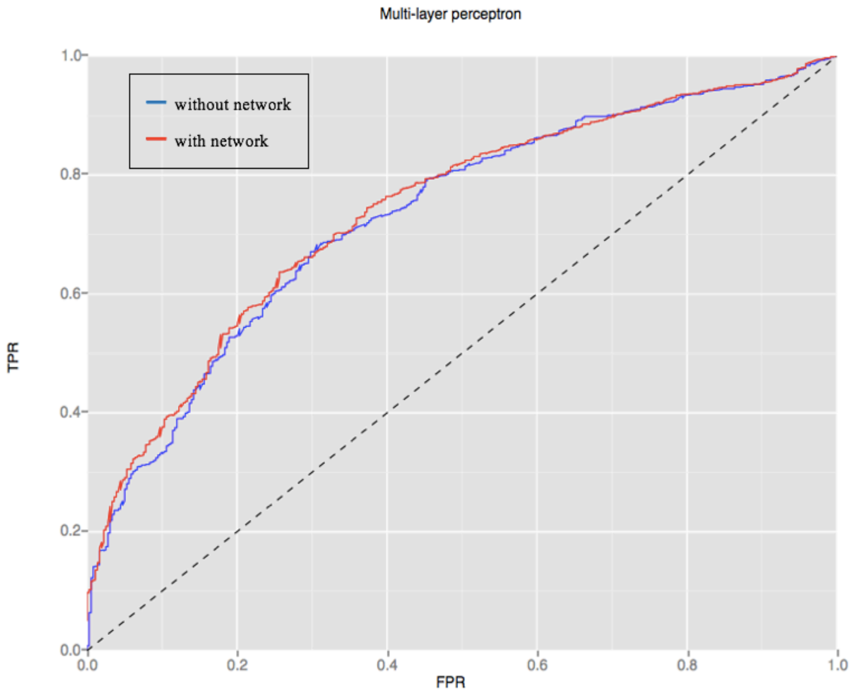


Figure 6.7 ROC of multi-layer perceptron

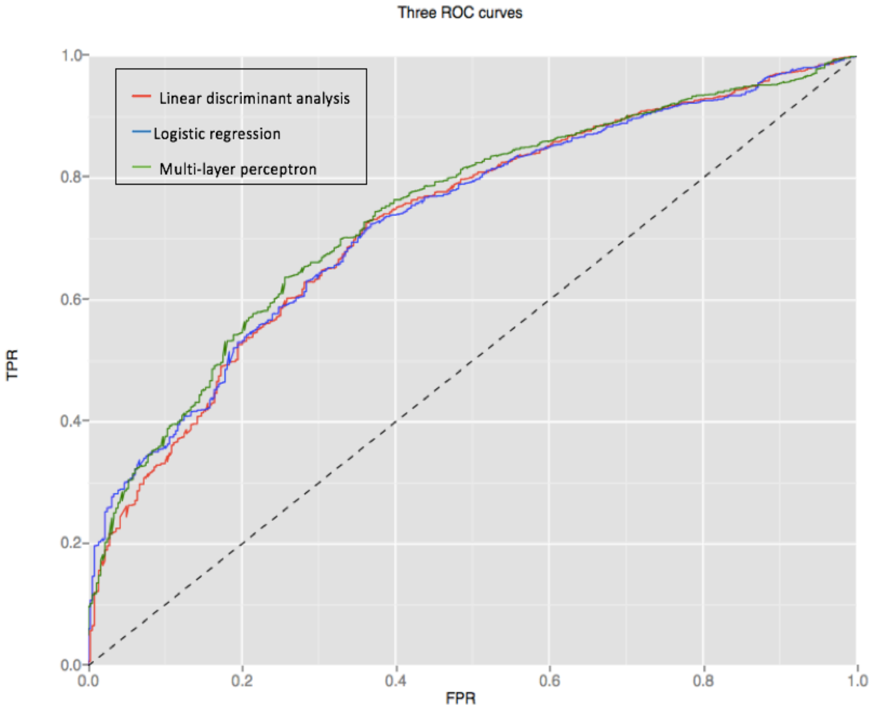


Figure 6.8 ROC of three models

The AUC valus of three models are as follow:

Model	AUC without network	AUC with network
Linear discriminant analysis	69.11%	72.71%
Logistic regression	72.41%	73.02%
Multi-layer perceptron	73.13%	74.06%

Table 6.1 AUC of three models

From the ROC and AUC it can be learned that the performances of three models have been improved by 1%~3% after adding network properties as extra features.

6.4.2 Precision and recall of defaulted class

The precision and recall of defaulted class is shown as follow:

Model	Precision_before	Precision_after	Recall_before	Recall_after
Linear discriminant analysis	31.14%	33.58%	59.33%	63.79%
Logistic regression	33.39%	33.48%	60.45%	63.23%
Multi-layer perceptron	33.23%	33.69%	59.33%	62.12%

Table 6.2 Precision and recall of three models

Network features have improved performances of all three models. However, they seem to have more positive effects on the recall while the improvement of precision is limited.

7. Discussion

In task 1, three machine learning algorithms have been used to build classification models for risk detection. They all have shown good performances. Among the borrowers who have been considered trustworthy by the platform, the models can correctly detect over 60% of borrowers whose loans are actually defaulted and 30% of the predictions of defaulted loans are correct. Let's assume there are 1000 qualified borrowers who want to apply for a loan and all of them they would have been approved by the platform. However, 200 borrowers of them would not pay their loans eventually. Now the platform is provided by the models that tell them 400 of these 1000 borrowers are risky. Furthermore, among these 400 borrowers, 120 will default, which account for 60% of bad borrowers. This is a great progress compared to the original strategy.

The key to machine learning's success is that it considers all the historical records of all borrowers and explores the pattern hidden behind the data. Therefore, the prediction is not made by only analyzing the target borrower, but based on the behavior of the whole group of borrowers who have similar characteristics with the target.

In task 2, the network features have been proved to have positive effect for risk detection. The AUC of the models increase by 1%~3% after adding in-degree, out-degree, betweenness centrality and closeness centrality as extra inputs to the models. The improvement may not be that significant. But considering the fact that the analysis of the network in this project is very basic, it has reason to believe that there is plenty room for the improvement to be enhanced if more specific algorithm of network analysis can be developed.

The positive effect of network properties indicates the implicit information involved in the lending network which is helpful to detect risky borrowers by analyzing their behaviors in lending activities. Such information is already possessed by peer-to-peer lending companies but the importance of it seems to be overlooked.

The objective of the project is to prove two points of view. First, machine learning can help to detect risky borrowers in peer-to-peer lending marketplace. Second, properties extracted from lending network can further improve the performance of machine learning algorithms. Thus, the methods chosen in the project are the most commonly used ones. More creative algorithms which are specific to this kind of problems can be designed to achieve better performance. This can be included in the future work.

In peer-to-peer lending marketplace, lenders are always the weaker side since their rights are not guaranteed by the platform. Although all the data of some peer-to-peer lending platforms is public, most lenders have no ability and no tool to take good use of it. The program of this project can be packed as an application and provided to the lenders to help them make investment decisions. Apart from the application in peer-to-peer lending industry, the method of using machine learning algorithms combined with network analysis can also be introduced to traditional financial institutions like commercial banks. So they can run double check by using the method to prevent loss from risky customers.

8. Conclusion

The project has successfully achieved the objectives set by me and my supervisor at the beginning. The final result even turns to be better than our expectation. From literature survey, data collection to coding and reporting, it is a good practice to apply what I have learned in the past year systemically to resolve some real issues. As a Msc project, the meaning of it is more than just implementing and evaluating some algorithms, but showing my ability to explore and solve problems independently. It’s a valuable experience for my further study and research work in the future.

References

Albert, R. & Motter, A. (2016). Networks in Motion. Physics Today, 64(3), 43-48.

Alhajj, R., & Rokne, J. (2014). Encyclopedia of social network analysis and mining. Springer Publishing Company, Incorporated.

Egham. Gartner Says Social Banking Platforms Threaten Traditional Banks for Control of Financial Relationships. Gartnercom. 2008. Available at: http://www.gartner.com /newsroom/id/597907.

Fred. Delinquency Rate on Consumer Loans, All Commercial Banks. Fred Economic Research. 2016. Available at: https://fred.stlouisfed.org/series/DRCLACBS.

Funahashi, K. (1989). On the approximate realization of continuous mappings by neural networks. Neural Networks, 2(3), 183-192.

Gestel, T., Baesens, B., Suykens, J., Van den Poel, D., Baestaens, D., & Willekens, M. (2006). Bayesian kernel based classification for financial distress detection. European Journal of Operational Research, 172(3), 979-1003.

Harvard Business. Breakthrough Ideas for 2009. Harvard Business Review. 2009. Available at: https://hbr.org/2009/02/breakthrough-ideas-for-2009.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. Neural Networks, 2(5), 359-366.

Joash. Prosper Loan Data. Github. 2015. Available at: https://github.com/

Kohavi, R., & Provost, F. (1998). Glossary of terms. Machine Learning, 30(2-3), 271-274.

Krumme, K. A., & Herrero, S. (2009, August). Lending behavior and community structure in an online peer-to-peer economic network. InComputational Science and Engineering, 2009. CSE'09. International Conference on (Vol. 4, pp. 613-618). IEEE.

LendingClub. Lending Club Statistics - Lending Club. Lendingclubcom. 2016. Available at: https://www.lendingclub.com/info/demand-and-credit-profile.action.

Lin, M., Prabhala, N. R., & Viswanathan, S. (2013). Judging borrowers by the company they keep: Friendship networks and information asymmetry in online peer-to-peer lending. Management Science, 59(1), 17-35.

Malhotra, R. & Malhotra, D. (2002). Differentiating between good credits and bad credits using neuro-fuzzy systems. European Journal of Operational Research, 136(1), 190-211.

Marcucci, M. (1997). Applied Multivariate Techniques. Technometrics, 39(1), 100-101.

Noh, H., Roh, T., & Han, I. (2005). Prognostic personal credit risk model considering censored information. Expert Systems with Applications, 28(4), 753-762.

Ong, C., Huang, J., & Tzeng, G. (2005). Building credit scoring models using genetic programming. Expert Systems with Applications, 29(1), 41-47.

PWC. How peer-to-peer lending platforms are transforming the consumer lending industry. ConsumerFinance. 2015. Available at: https://www.pwc.com/us/en/consumer-finance/publications/assets/peer-to-peer-lending.pdf.

Redmond, U. & Cunningham, P. (2013). A temporal network analysis reveals the unprofitability of arbitrage in The Prosper Marketplace. Expert Systems with Applications, 40(9), 3715-3721.

Sánchez, M. & Sarabia, L. (1995). Efficiency of multi-layered feed-forward neural networks on classification in relation to linear discriminant analysis, quadratic discriminant analysis and regularized discriminant analysis. Chemo metrics and Intelligent Laboratory Systems, 28(2), 287-303.

Shen, D., Krumme, C., & Lippman, A. (2010, August). Follow the profit or the herd? Exploring social effects in peer-to-peer lending. In Social Computing (SocialCom), 2010 IEEE Second International Conference on (pp. 137-144). IEEE.

Simon, P. (2013). Too Big to Ignore: The Business Case for Big Data (Vol. 72). John Wiley & Sons.

West, D. (2000). Neural network credit scoring models. Computers & Operations Research, 27(11-12), 1131-1152.

Xu, Y., Qiu, J., & Lin, Z. (2011, November). How Does Social Capital Influence Online P2P Lending? A Cross-Country Analysis. In Management of e-Commerce and e-Government (ICMeCG), 2011 Fifth International Conference on (pp. 238-245). IEEE.

Yang, Y. (2007). Adaptive credit scoring with kernel learning methods. European Journal of Operational Research, 183(3), 1521-1536.

Yum, H., Lee, B., & Chae, M. (2012). From the wisdom of crowds to my own judgment in microfinance through online peer-to-peer lending platforms. Electronic Commerce Research and Applications, 11(5), 469-483.