**Ministry of Education and Science of the Republic of Kazakhstan**

**M. Narikbayev KAZGUU University**

«Approved for Defense»

Supervisor

E.Toqbolat

«05» May 2021

**MASTER'S THESIS (PROJECT)**

# «Peer-to-peer lending loan default prediction: machine learning classification algorithms applied to Lending Club data, investors' perspective»

**program 7M04124 - «Finance»**

**Written by**

M. Syzdykov

**Supervisor**

E.Toqbolat

**Nur-Sultan, 2021**

**M. Narikbayev KAZGUU University**

# PEER-TO-PEER LENDING LOAN DEFAULT PREDICTION: MACHINE LEARNING CLASSIFICATION ALGORITHMS APPLIED TO LENDING CLUB DATA, INVESTORS' PERSPECTIVE

Marat Syzdykov

May, 2020

«Approved»

Supervisor's E. Toqbolat

Supervisor's Signature Signed

«05» May 2021

**Nur-Sultan, 2021**

## Abstract

Recent years have witnessed an emergence of online social lending market, also known as peer-to-peer, or P2P lending. Borrowers and lenders are allowed to interact through P2P lending platforms online without a presence of a strong intermediary such as conventional banks. Nevertheless, as P2P platforms promote wider financial inclusion, the market is also characterized by the issue of higher levels of information asymmetry than that faced by traditional banks. For said reason, this thesis studies how well can the individual investors deal with information asymmetry by the means of machine learning default prediction modelling data provided by Lending Club P2P platform. To that purpose, we first examine the findings of related literature. We then choose Random Forest and XGBoost machine learning classification algorithms for experimental part of our study, with Logistic Regression classifier as performance benchmark. Our study emphasizes the use of appropriate performance metrics in presence of class imbalance, but also fair and transparent interpretation of the classification results. Next, we conduct a thorough and transparent data preparation. In the experimental results, the performance of the chosen classifiers is compared between themselves, with no significant difference between them to justify their ranking. Additionally, the results of premier classifiers of six related works are showcased, and the similarity of these results generally coincides with those of our research. However, unlike the related literature, our study further introduces the thresholding technique for the prediction results, which is illustrated to be capable of reducing the number of misclassified loan defaults, providing the opportunity for higher and more stable portfolio returns for the individual investors. Although we demonstrate how machine learning classification algorithms combined with thresholding technique can provide reasonable results for the investors, the observable consistency of the prediction results across the field suggest that the type of data provided by Lending Club may be insufficient to build machine learning models of high predictive power. Thus, we underline the need for wider use of alternative data in P2P lending market. However, this notion raises a number of questions for further research regarding alternative data regulations, privacy, and ethics in P2P lending.

## Contents

# List of Tables

## List of Illustrations

# Introduction

Online peer-to-peer (commonly referred to as P2P) lending platforms, also known as alternative lending, crowdlending, marketplace lending, or debt-based crowdfunding, directly connect potential borrowers and lenders, be they individuals or legal entities on either side. The first online P2P lending company Zopa was established in the United Kingdom in 2005, shortly after, Prosper, Lending Club, and others followed suit in 2006. Through the years, P2P lending market grew to an estimated USD 67.93 billion in 2019 (Khan et al. 2020).

However, contrary to the image of a young industry that the P2P lending can project, as Cummins et al. (2019) correctly summarize, “P2P lending and collective financing are not new ideas in themselves”, and have extensive history. Obviously, individuals have lent money to each other before. For instance, Dermineur (2019) explores the peer-to-peer lending in pre-industrial France of XVIII century and in part showcases it’s sizeable role in the country’s economy at the time.

Everett (2014) argues that contemporary form of social lending, powered by technology, is in major part owes its origins to the English “friendly societies” of XVIII and XIX centuries Britain, that were prominent enough to be the subject of the Friendly Societies Act of 1793 by the Parliament of Great Britain; members of those societies were entitled to open deposits and borrow debt, and also to receive help in case of certain adverse events.

Other studies also show that lending practices of similar spirit were taking place in Ireland’s lending societies of XVIII–XIX centuries (Hollis & Sweetman, 1997), and in German credit cooperatives of XIX century (Stark, 2015).

Thus, it can be argued that before the later uncontested dominance of conventional banks in the matters of lending, peer-to-peer practices were prominent parts of economies throughout different societies of the world; the contemporary technological advancements made the creation of online P2P platforms possible and have essentially reinvigorated a once prominent form of economic relations in new ways.

During its relatively short lifespan, online P2P lending industry experienced exponential growth for a number of reasons. The rise of the industry can in major part be attributed to the coinciding financial crisis of the late 2000s (Kirby & Worner, 2014; Mateescu, 2015). The financial distress at the time restrained banks in their ability to fund SMEs and individuals, which created a corresponding credit deficit partially filled by P2P lending platforms. As Kirby and Worner also outline, the loss of trust in banks by the general public, which can be supported by several researches' findings (Bennett & Kottasz, 2012; Carbo-Valverde et al., 2013; Gillespie & Hurley, 2013; Roth, 2009), too, played into the rise of alternative investments such as fintech lending. Moreover, P2P lending offers higher return rates than such conventional investments as savings accounts and government bonds.

Needless to say, the Covid-19 pandemic did adversely influence the P2P lending market in 2020, and more than 75% of CEOs in P2P lending, who participated in a survey done by AltFi (2020), attested to the ensuing damage upon the business. However, Swaper P2P lending platform reports that, after the initial performance declines in March and April, the European P2P lending industry started showing signs of recovery in May (Swaper, 2021); furthermore, despite the adverse economic circumstances, there were platforms like Swaper that even managed to grow their alternative lending business.

As of beginning of 2021 there are 251 online P2P lending platforms registered by P2PMarketData (2021), with their regional distribution depicted in Figure 1. Despite the recent pandemic complications, overall, the P2P lending market in general is persevering and even showing optimistic signs of growth so far in first quarter of 2021 as evidenced in the statistics provided by P2PMarketData (2021) for 54 European platforms with publicly available funding data, and illustrated in Figure 2.

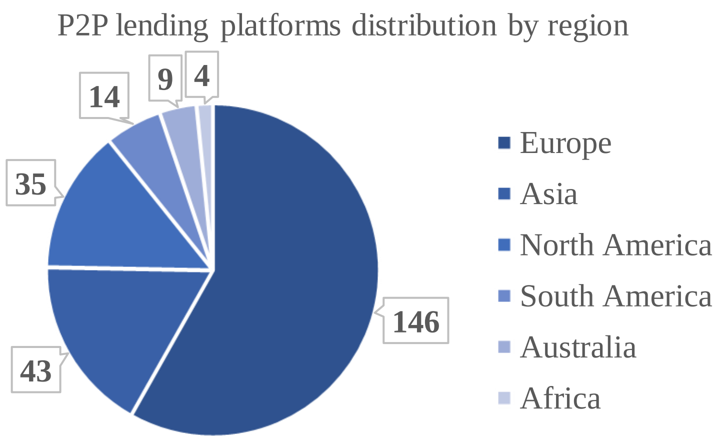


*Figure 1.* Quantity of P2P lending platforms across the world's regions.

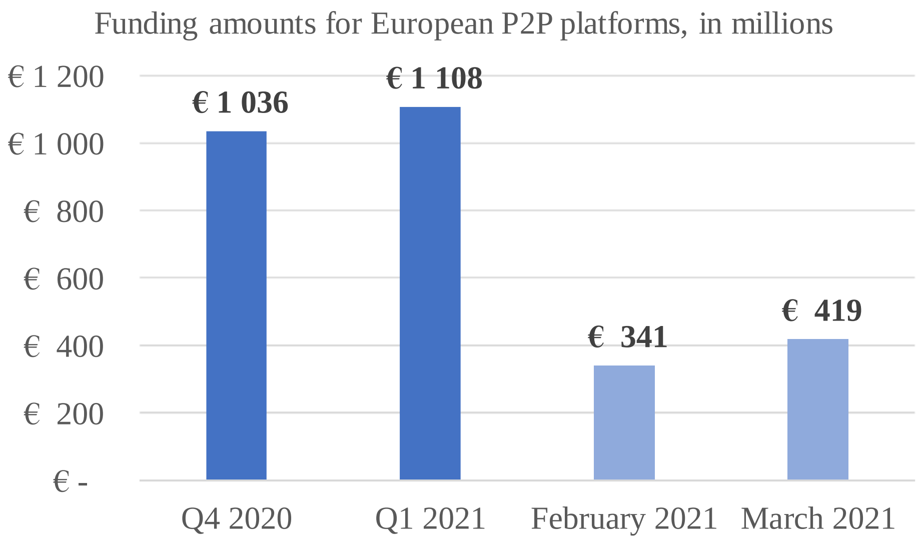


*Figure 2.* Comparative increase in European P2P loan funding for the last reported periods.

When it comes to the business models and structure of P2P lending platforms, as Mateescu (2015, p. 1) has put it, "Peer-to-peer lending started out as a relatively simple system for facilitating loans between individuals online, but has since grown into a complex ecosystem of technologies, institutions, and auxiliary startups". Although, different P2P lending platforms implement their own unique processes, systems, and features, there are standard similarities that can be traced across the sector (Claessens et al., 2018).

Crucial part of those common features is loan application by a potential borrower and the platform credit scoring model classifying the request. The exact requirements for the information potential borrowers provide about themselves, when applying for a loan, vary across platforms, but, in general, are expected to disclose such details as their income, age, residence, etc. by which the individual proprietary models of P2P lending platforms are evaluated, assigned a grade from A (the safest) to G in the case of Lending Club, for example, and judged in combination with their FICO score[1].

Nevertheless, P2P lending platforms face a problem of affirmation asymmetry, a more severe one compared to banks (Giudici & Misheva, 2017), which lead to credit scoring mistakes and adverse selection issues. Serrano-Cinca et al. (2015) describes how P2P lending platforms struggle to get borrowers' profiles of the same quality as traditional banks, who have access to their credit history, or know them in person for the least. Giudici and Misheva argue that, because of most P2P operators not being loan originators (i.e., not directly owning the counterparty default risk) it further worsens the issue at hand as the investors face the risk instead and the financial incentive is lesser for such operators compared to banks as a result. Moreover, the results of the research by Giudici and Misheva highlight how, despite being statistically significant, Lending Club's grading lacks predictive power. Therefore, in regards to risks facing investors in P2P, the default risk and the quality of its evaluation can be seen as of the highest importance.

When it comes to the approaches to assessing the default risk, however, to the best of our knowledge, most P2P lending platforms use models based on rating (grading). However, while large financial entities such as banks or P2P lending platforms can assume relatively large risks compared to individual investors, as Guo et al. (2016) state, for the latter the rating models implemented by such institutions are ill suited. Guo et al. also correctly mention that, due to the possibility to partially invest in any individual loan in P2P lending platforms, diversification has been made possible for the respective investors.

[1] A FICO score is a credit score created by the Fair Isaac Corporation (FICO). Lenders use borrowers' FICO scores along with other details on borrowers' credit reports to assess credit risk and determine whether to extend credit. FICO scores take into account data in five areas to determine creditworthiness: payment history, current level of indebtedness, types of credit used, length of credit history, and new credit accounts.

Nonetheless, taking into account the sheer difference in regards to the magnitude of funds available for investment between financial entities and private lenders, it is logical to infer that the P2P individual investors have direct interest in devising their own default prediction models in hopes of attaining higher predictive power to improve the quality of their investment decision process.

This study revises the literature on the subject of loan default prediction in P2P lending, explores how individual investors can pursue such a goal by means of general predictive models built on Random Forest, Extreme Gradient Boosting (XGBoost) machine learning classification algorithms on the loans data provided by Lending Club. The performance of the algorithms is then assessed, with metrics suitable to the nature of the classification task at hand, and compared against each other and the benchmark performance of Logistic Regression classification model. Additionally, it is important to note that to conduct the experiment in question this study mainly relied on R software (for specifics of software implemented refer to Appendix A).

The remainder of this thesis is structured in the ensuing order. Literature review on the topic of classification techniques is displayed in the following second section. Next, in third section, the methodology elaborates on the methods behind the classification algorithms and performance metrics, as well as the imbalanced classification issue and the logic behind the choice of the algorithms in question. After that, the data analysis and preparation is performed in fourth section. Lastly, experimental results are provided in section five, followed up by conclusion in section six.

# Literature Review

This section is going to explore studies concerning machine learning classification algorithms and their use for credit scoring in general, and how such algorithms are represented in the research field of peer-to-peer lending for loan default prediction, with main focus on literature utilizing Lending Club loan records. For a better understanding of the review of related works readers may refer to performance metrics subsection of methodology.

To begin with, multiple industries including financial sector have been increasingly incorporating machine learning based solutions (Emerson et al., 2019). Specifically, machine learning also finds numerous applications in the investment process, as Emerson et al. demonstrate. To provide a general definition of machine learning as per Dixon et al. (2020, p. 8), it is a vast subject "covering various classes of algorithms for pattern recognition and decision-making". For the matter of loan default predictive modelling in question, we are naturally focused on the type of the machine learning algorithms known as the supervised learning.

The supervised machine learning classification, specifically, involves training on data with already labeled classes (i.e., present output values), as defined by Sathya and Abraham (2013), in contrast with unsupervised learning, which trains on data only with input values, finding its structure. Supervised algorithms essentially learn by example, in presence of correct actual values to check for the predictions' quality, hence, the name "supervised". Such algorithms normally involve training and testing phases, meaning that the algorithm is first fitted to the data and then tested to see how well it generalizes (i.e., how well it performs on unseen data). During the training stage, supervised machine learning algorithms find patterns in input features that correlate with dependent output attribute to consequently make either classification or regression predictions.

In regards to the respective literature, as Teply and Polena (2020) argue, there is an abundant variety of research works exploring the relative performance of distinct classification approaches for default prediction, or credit scoring, in general. The situation is different, however, for the availability of such studies in the field of P2P lending, and namely works concerning data from Lending Club platform.

On the given matter, there have not been conducted numerous studies of scope, nor have they been exhaustive, to the best of our knowledge.

Among the literature studying default prediction of P2P loans on Lending Club data, studies experimenting with machine learning classifiers on reasonably large datasets, to the best of our knowledge, had been conducted by Teply and Polena (2020), Boiko Ferreira et al. (2017), Zanin (2020), Niu et al. (2020), Malekipirbazari and Aksakalli (2015), Moscato et al. (2021), Song et al. (2020), and Namvar et al. (2018). These literature pieces provide evaluation and comparison of diverse combinations of classification approaches for the default prediction of peer-to-peer lending loans based on the publicly available Lending Club historic loans data. Review of the experiments and findings of the respective research works will be conducted bellow.

Research conducted by Teply and Polena (2020) claims to be the first one to propose default risk classifiers techniques rankings for the ten models selected by them. Teply and Polena obtained their dataset of 212,280 observations from Lending Club records from the years 2009 through 2013, and implemented 10 following classification algorithms: Logistic Regression, Linear Discriminant Analysis, Support Vector Machine, Artificial Neural Network, K-Nearest Neighbor, Naïve Bayes, Bayesian Net-Work, Classification and Regression Tree, and Random Forest. Nevertheless, Teply and Polena did not elaborate on and did not address the skewed nature of the respective target class data, which throughout the history of Lending Club records shows significant proportional prevalence of negative class, or paid off, loans. Accordingly, the metrics that were used for assessing the discriminatory power of the mentioned classifiers (e.g., Accuracy, ROC-AUC, Kolmogorov-Smirnov statistic, Brier score, etc.) were likely not objectively representative of the imbalanced classification, as it is explained further below in subsection 5 of Methodology section 3. For context, however, it is worth stating that the higher ROC-AUC score of 0.6979 was obtained by Logistic Regression classifier out of the applied algorithms, as well as Logistic Regression being stated as the best overall according to metrics average ranking among the 10 models.

Another work conducted on the subject is by Boiko Ferreira et al. (2017), and it compares ensemble machine learning algorithms (i.e., AdaBoost, Bagging, and Random Forest), cost-sensitive applications of three classifiers (i.e., Decision Tree, Gaussian Naïve Bayes, and Logistic Regression), and combinations of the three mentioned algorithms with sampling techniques (i.e., SMOTE, SMOTE Borderline2, and Random Under-sampling) according to their measurements of sensitivity, specificity, and ROC-AUC. The final data set for this study consisted of 578,331 loans extracted from origination period between 2007 and 2016, and had negative class prevalence of approximately 4:1; however, it still contained 133 input features, which, even despite one-hot encoding of the categorical variables, is a large number of features not necessarily suitable for the methods in question. Overall, their results had seemingly ambiguous interpretation as multiple classifier's performance results have been extremely skewed towards sensitivity (i.e., these models simply classified almost every instance of the test set as the positive class case), while the best overall results were shown by the logistic regression in combination with sampling techniques showing ROC-AUC scores in the 0.64 – 0.66 range. As it can be seen, Boiko Ferreira et al. recognize the class imbalance and apply prediction techniques suitable to the task, nonetheless, metrics that are appropriate for imbalanced target class data, aside from sensitivity and specificity, were not implemented by the study in focus.

Zanin (2020) proposes a comparison of predictive power of single classifiers, such as Generalized Additive Model, Naïve Bayes, Random Forest, and Extreme Gradient Boosting in combination with sampling techniques (i.e., over-sampling, under-sampling, over- and under-sampling, and ROSE sampling method from "ROSE" R package), and also the aggregated models combining probability predictions from the previously stated classification approaches by means of  regularized logistic regression (i.e., Lasso, Ridge, and Elastic-Net regressions). The data chosen were only 36-months loans of the 2010-2015 time period, totaling 612,745 observations, which were further split into training, validation, and test samples. In the provided results, author identified the ensemble lasso regression as the best among the selected classifiers with scores of 0.3274 F-1 and 0.6288 G-mean. However, F-1 score is not well-suited for imbalanced classification evaluation, as discussed further in Performance Metrics subsection of Methodology section of our study, and all of the models proposed by Zanin had the Sensitivity score ranging from 0.48 to 0.59, meaning that correct identification of positive class cases was near random guessing level. Moreover, the recorded improvements in the respective performance metric scores of aggregated models over single algorithm counterparts, although being statistically significant, were not of substantial size.

Niu et al. (2020) introduce original scoring modelling approach called resampling ensemble model based on data distribution (REMDD), performance of which is evaluated and compared to datasets from three different P2P lending platforms and then compared to single algorithm models (i.e., logistic regression, decision tree, Random Forest, XGBoost) without prior sampling and decision tree ensembles combined with sampling techniques (i.e., random over-sampling, random under-sampling, SMOTE, under-bagging, a combination of SMOTE and bagging with differentiating sampling rates (SBD) by Sun et al. (2018), and clustering based under-sampling (SBC) by Yen and Lee (2009) ).

The proposed REMDD credit scoring approach itself is comprised of two major stages. Firstly, it resamples the training set with, as the authors claim, original under-sampling method based on majority class distribution (UMCDD), which essentially creates multitude of balanced training folds based on K-means clustering of majority class instances followed up with bagging procedure of both classes. The second step includes training base classifiers, the decision tree ensembles, on the created training samples, validation, and selection of base learners respective to their ROC-AUC score. After that, the resulting REMDD is applied to classify the testing sample instances.

The data sources for Niu et al. (2020) were Lending Club 2015 records (292,655 total observations with imbalance ratio of 3.96), popular Chinese P2P platform PaiPaiDai 2015-2017 records (118,767 total records with imbalance ratio of 11.37), and Prosper records (28,399 total records with imbalance ratio of 2.3). The assessment measurements selected by the authors (i.e., sensitivity, specificity, ROC-AUC, and G-mean) concluded that REMDD had the best overall performance compared to other chosen classifiers. However, the REMDD predictions on Lending Club and Prosper data performed noticeably worse than those on PaiPaiDai records, with sensitivity and recall scores both approximately of 0.7 compared to their

PaiPaiDai platform counterpart's sensitivity and recall both around 0.9 values respectively; thus, implying the possibility of higher quality feature sets provided by the Chinese P2P lending platform.

Malekipirbazari and Aksakalli (2015) conducted a relatively smaller predictive performance comparison of Random Forest classifier against alternative scoring models based on algorithms such as k-Nearest Neighbors, Support Vector Machine, and Logistic Regression on different feature set variations. The data set extracted by the researches was from Lending Club 2014 period and contained approximately 350,000 total loan records on 23 selected features. The results, showed by the authors, claim Random Forest to be the superior classifier among the other selected algorithms according to the performance measures on Accuracy, ROC-AUC, Root mean squared error (RMSE), True Positive and False Negatives rates. However, their interpretation does not take into attention how the Sensitivity, or the True Positive rate for "bad" loans, stays below 0.4 value for all presented variations of Random Forest classifier. Thus, Malekipirbazari and Aksakalli likely do not provide a fair representation of their results, nor they recognize class imbalance issue and the need for the application of relevant performance measurements.

Moscato et al. (2021) proposed a benchmark design for machine learning application for credit scoring in P2P lending. For this cause, Moscato et al. chose to compare models based on Logistic Regression, Random Forests, and Multilayer Perceptron (an Artificial Neural Network algorithm variant) in combination with various sampling techniques for imbalanced data, and evaluated these approaches in terms of their explainability. The data for their experimental analysis was extracted from Lending Club data records of the time period of years 2016-2017, containing 877,956 total loans. Among the implemented classification approaches, Moscato et al. concluded Random Forest in combination with random under-sampling to be the best classifier according to the overall performance of selected metrics (i.e., Accuracy, ROC-AUC, Sensitivity, Specificity, false positive rate, and geometric mean), namely with 0.717 ROC-AUC, Sensitivity of 0.63, and 0.68 Specificity. The author also examined their results in terms of their explainability, and concluded part of the approaches to be feasible for the proposed benchmarking.

Song et al. (2020) present original classification method designated as “distance-to-model and adaptive clustering-based multi-view ensemble” (DM-ACME), which uses a blend of multi-view learning, adaptive cluster-sampling to generate an ensemble of gradient boosting decision trees learners. The proposed approach is then compared to an array of base classifiers combined with sampling methods (i.e., Gradient Boosting Decision Tree, Random Forest, AdaBoost, Decision Tree, Logistic Regression, Multilayer Perceptron algorithms with under- and over-sampling). The data source was Lending Club records from 2014, resulting in 70,860 total selected instances. According to the performance measurements of this study (i.e., Accuracy, ROC-AUC, Sensitivity, Specificity, and G-mean), the authors deemed DM-ACME as effective for the purpose of default prediction. However, said DM-ACME classifier had sensitivity of only 0.4607 with specificity of 0.7678; and while DM-ACME had the highest ROC-AUC score of 0.6697 among the comparison group, Random Forest produced sensitivity score of 0.6623 with 0.5791 specificity. Discussed method’s sensitivity of 0.4607, signifying that said classifier can recognize less than half of test sample defaulted loans.

Namvar et al. (2018) focuses on comparing the predictive performance of implementing a variety of sampling techniques for class imbalance (i.e., random over- and under-sampling, instance-hardness threshold, SMOTE, SMOTE with Tomek links, SMOTE with edited nearest neighbors, and ADASYN) in combination with Logistic Regression, Linear Discriminant Analysis, and Random Forest classification algorithms. Their dataset contained approximately 636,000 initial entries from Lending Club records of the years 2016 and 2016. The metrics used were Accuracy, ROC-AUC, Sensitivity, Specificity, FP-rate, and G-mean. Their experimental results found combination of Random Forest and random under-sampling to be the best performing classification model with Sensitivity of 0.717, Specificity of 0.582, and 0.69 ROC-AUC. Namvar et al. results reinforce the instances of other researches above implementing popular techniques such as SMOTE, which did not necessarily guarantee increased prediction performance results.

On the other hand, there were other literature pieces in this specific research field, that can be deemed as providing questionable results such as Hou (2020), Al-qerem et al. (2019), and Arora and Kaur

(2020). Hou presented classifiers with nearly perfect results, which in fact were obtained due to feature leakage, as they included attributes regarding recovered principal and recovered interest of loans, the information that was clearly was recorded after the loan origination date. Al-qerem et al. provided classifiers of extremely high predictive power, but did not disclose their selected features used for the experiment; such results could not be regarded as credible, due to the potential of feature leakage presence. Arora and Kaur provided only Accuracy and ROC-AUC metrics for the evaluation of the classification models constituting their experiment, making these results hardly interpretable due to the imbalanced nature of lending data.

To summarize, there may not exist single best performing credit scoring approach as studies by Abdou and Pointon (2011) and Ala'raj and Abbod (2016) argue, however, our goal is not to identify the best single classifier for loan default prediction, but to see how the investors can fair with the information asymmetry issue present and how powerful can be the default prediction classifiers they can construct by the means of machine learning algorithms on the provided Lending Club data.

So far, in the illustrated related research highlights, we have seen that not every study on the topic recognizes and addresses the imbalanced nature of the data, and the multiple researches do not provide enough performance measurements appropriate for the presence of the skewed target class. We made sure to review only those studies, that were conducted in transparent manner with the use of objectively selected data. Finally, there have not been sighted cases of outstanding prediction results among the novel and already existing classification approaches for credit scoring demonstrated by the credible literature.

# Methodology

This section is going to be organized the following way. First subsection addresses the class imbalance problem. The selection of classification algorithms is showcased in the second subsection. The classification algorithms chosen are discussed in the following three subsections. Lastly, the metrics necessary to evaluate the performance of the models based on the given algorithms are described in the closing subsection of methodology. Moreover, a simplified summary is presented in the last subsection.

## Addressing the Imbalanced Classification Issue

To begin with, it is necessary to outline the fact that the classification in question is characterized as imbalanced binary classification. Imbalanced classification problem arises when dealing with imbalanced data set, where the distribution of observations among the two outcome classes is significantly unequal (Fernández et al., 2018). As will be showcased below in Data Analysis section, the class imbalance in the selected data set is approximately of proportion 1:5, with loans designated as “default” being the minority class and “paid off” loans constituting the majority.

Observed class inequality is intrinsic property of historic data sets for loan default prediction. Nevertheless, imbalanced data set should not be treated with standard classification procedures. For the most part, machine learning classification algorithms are usually designed with assumption of equal distribution of observations among the target classes data (Elrahman & Abraham, 2013). However, such traditional approaches tend to provide poor class prediction results (Lemnaru & Potolea, 2012).

The problem of class imbalance in different fields of expertise, as well as machine learning in general, was researched by numerous studies (Krawczyk, 2016). Nonetheless, as Krawczyk states, despite the issue of imbalanced classification being extensively researched for decades, there are no definitive, universal solutions to the problem, as distinct classification cases respond differently to any given imbalance handling technique due to data and classification task specifics. However, some general and popular approaches for addressing classification on imbalanced data can be outlined as can be seen through the example of works such as Elrahman and Abraham (2013) and Kotsiantis et al. (2005). These methods can be grouped into the following categories.

Sampling based methods resample the original data set to achieve equal proportions among target classes. As showcased in He and Ma (2013), the respective studies in the domain regard random over- and under-sampling, data generating synthetic sampling techniques, sampling methods based on clustering, and combinations of some of those sampling approaches.

Cost-sensitive classification algorithms take into account misclassification costs, penalties associated with incorrect predictions, which can be calculated in the framework of a cost matrix. As defined by Elkan (2001), cost matrix assigns true positives/negatives, false positives/negatives their respective costs, although true predictions are typically represented by zero cost. In the case of binary imbalanced classification problems, the failure to correctly classify the minority class instances, or Type I error, normally leads to substantially more severe consequences than misclassifying the majority class examples, Type II error. The goal of cost-sensitive algorithms is to minimize the sum of these costs, the total cost (Ling & Sheng, 2010). Cost-insensitive learning, on the contrary, does not take the given costs into consideration.

Other notable techniques to mention include ensemble-based methods, which combine multiple classifiers to improve the overall prediction results, and recognition-based methods, also known as one-class learners.

## Choice of Classification Algorithms

To gain insight into how the task of default prediction in P2P lending on Lending Club loans data can be approached with the means of machine learning, we need to choose what machine learning algorithms to employ. To make such choice one can rely on the evidence about popularity and effectiveness of certain machine learning algorithms provided by Kaggle. Kaggle is a popular online community of data professionals and practitioners, with over 5 million registered users worldwide.

Kaggle regularly conducts competitions involving machine learning attracting thousands of participating teams and individuals.

According to the information provided by the CEO of Kaggle, Anthony Goldbloom (2016), the most popular machine learning models employed by the winners of Kaggle competitions since 2015 were those that incorporated Neural Network, Gradient Boosting Machine, and Random Forest algorithms; such preferences seemingly remain relevant among the Kaggle users to the present day, according to the report published by Kaggle (Kaggle, 2021). The results for the survey in the mentioned report regarding the most frequently used machine learning algorithms identified the Linear or Logistic Regression as the most popular choice, with Decision Trees or Random Forests, Gradient Boosting Machines, and Neural Networks algorithms filling in the rankings in the respective order. That said Neural Networks models are known for being comparatively time consuming, difficult to interpret and susceptible to overfitting (Akinsola, 2017; Bhavsar & Ganatra, 2012).

For these reasons, Logistic Regression, Random Forest, and Gradient Boosting Machine variant, XGBoost, classification algorithms will be used for experiment framework of this study, with Logistic Regression classifier serving as the performance benchmark, as Logistic Regression is regarded as a standard for credit scoring models as stated by Lessmann et al. (2015).

### Logistic Regression

Logistic regression is a popular parametric classification algorithm when dealing with the prediction of dichotomous dependent variables. Essentially, Logistic regression classifies instances through fitting S-shaped curve, the sigmoid function, which assigns them probabilities 0 through 1.

In the framework of this thesis, we are going to implement backwards Stepwise Logistic Regression approach, in combination with 5-fold cross validation, to efficiently find significant set of features on already pre-processed training data with filtered set of relevant variables.

## Decision Trees and Random Forest

In order to understand the Random Forest algorithm, it is necessary to describe the Decision Trees, its building blocks. According to a detailed account by Kirasich et al. (2018), Decision trees are a tree-like structured algorithm where the initial/top node is designated as the “root”, starting from which, a sequence of recursive splitting/branching of successive decision nodes ultimately leads to reaching the terminal, or “leaf” nodes which represent the prediction result. The decision tree algorithm is a top-down “greedy” technique which divides the dataset into lesser subsets. Greedy algorithms received their name due to preferring simpler solutions over the often more complex, optimal ones. At each node the splitting decision is based on a test about the data.

At each decision node, the data is split into two branches depending on a single feature values and this process is repeated until the leaf nodes are reached, which is used to make the final prediction. To determine which feature to split on at each node, different criteria, such as reduction of variance for example, are applied to rank the usefulness of variables in segregating the class labels. Accordingly, the root node at the top of the tree corresponds to the best predictor variable.

Tree-based models can be trained on large datasets with both quantitative and qualitative attributes. Moreover, decision trees are immune to redundant features which may prompt overfitting in other algorithms. They also have very few tunable parameters and are insensitive to outliers and missing values. Nevertheless, trees are prone to overfitting, even despite the pruning, which dismisses redundant parts of the decision tree. Overfitting, one of the central issues of supervised machine learning (Hawkins 2004), occurs when a model learns training data too well, negatively impacting its performance on unseen data. The noise, random fluctuations, of the training data is learned as concepts, which, however, fail to apply to new data, hindering the model’s generalization capability.

Random Forest is an ensemble machine learning algorithm proposed by Breiman (2001), that solves the mentioned issue of overfitting in decision trees by incorporating multitude of unpruned trees. Random Forests achieve this by decorrelating its trees by the means of bagging (bootstrap aggregating) which is a resampling technique with replacement that reduces variance, and randomly sampling a specified number of features at each decision node (Kirasich et al., 2018). Ultimately, Random Forest uses a divergent collection of trees to average (regression) or compute majority votes (classification) in the terminal leaf nodes to make significantly more accurate predictions than the single decision trees algorithms.

### Gradient Boosting Machines, XGBoost

XGBoost is a variant of Gradient Boosting Machines algorithm introduced by Friedman (2001); hence it is necessary to describe gradient boosting algorithm in general before reviewing the aforementioned classifier that will be used in the experimental part of this study. According to a concise account by Ayyadevara (2018), gradient boosting can be conveniently understood through a comparison to random forests.

In the previous subsection, it has been deliberated upon how random forest is a bagging (bootstrap aggregating) algorithm that makes a prediction based on a collection of outputs of multiple trees. Bagging algorithms generally are composed of manifold parallel independent base learner algorithms (e.g., decision trees) built on bootstrap samples to aggregate their average prediction. In contrast, gradient boosting algorithms are characterized as a sequential and additive structure of learners (Natekin & Knoll, 2013); after the initial estimates, each consecutive learner minimizes the loss function (Wald, 1992) of the predecessor by means of gradient descent (Ruder, 2017), thus boosting (improving) the prediction results.

Essentially, loss function, also known as cost function, discussed by Wald estimates the quality of the model – the lower its value, the better the model performs. Gradient descent, as described by Ruder, is a method of minimizing an objective function by updating the parameters of the function towards negative gradient of the function (i.e., to achieve lower output values); another way to generally define the gradient in gradient boosting is as the loss function's derivative that describes its slope. Thus, the gradient is utilized to find the direction towards which to modify the parameters of the model to maximally reduce the error in the following training round by decreasing the loss function or “descending the gradient”.

Introduced by Chen and Guestrin (2016), Extreme Gradient Boosting, or XGBoost, can be defined as an improvement upon original GBM framework; it was developed in such a way to simultaneously enhance both performance results and computational efficiency compared to original gradient boosting. Extreme gradient boosting possesses numerous such improvements developed by Chen and Guestrin, most important of them are discussed below.

More efficient split point search. While regular GBM computes every single split’s potential loss to extend the next branch, XGBoost implements Weighted Quantile Sketch for determining estimated best split point. Essentially, the given technique creates a histogram for each feature, and histogram bins boundaries are considered candidates for the best split point search. Additionally, the Weighted Quantile Sketch assigns weights to the data points respective to the “confidence” of their given predictions and the histograms are constructed in a way as to allocate identical total weight to each bin, in contrast to the same quantity of points in the base quantile sketch framework. Consequently, higher number of candidate points results in a more detailed search for the low performing areas of the model. In turn, it leads to quicker investigation of hyperparameter settings, which are numerous for XGBoost.

Pruning of decision trees. The stopping criterion for branching of trees in regular gradient boosting is “greedy” as it relies on the reduced error at the point of split. XGBoost instead depends on maximum depth parameter specified, and performs backwards pruning. The described ‘depth-first’ approach, also known as level-wise approach, significantly increases computational productivity.

In XGBoost the decision trees may have distinct quantities of leaf nodes, which are shrunk proportionally to the data points contained.

Gradient descent approach is replaced with Newton Boosting based on Newton-Raphson method of approximations (Akram & Ann, 2015) which offers a more direct route to the loss function minima.

Sparsity-aware XGBoost algorithm accepts sparse data inputs by imputing best missing value according to training loss and processes diverse kinds of data sparsity patterns efficiently.

Parallelization is implemented during the tree building process in XGBoost when determining optimal split points at features level (e.g., while one core searches for the optimum split point and its respective error for attribute A, second core performs the same task for variable B, and the point corresponding to the lowest loss value is chosen).

XGBoost applies both L1, or LASSO, and L2, or Ridge, regularization techniques (Hastie, 2020; Mairal & Yu, 2012) to more complex learners in order to mitigate overfitting.

The additional parameters for randomization, which randomly subsample training data by columns and rows according to specified values, can simultaneously decorrelate the base learners and speed up computational process.

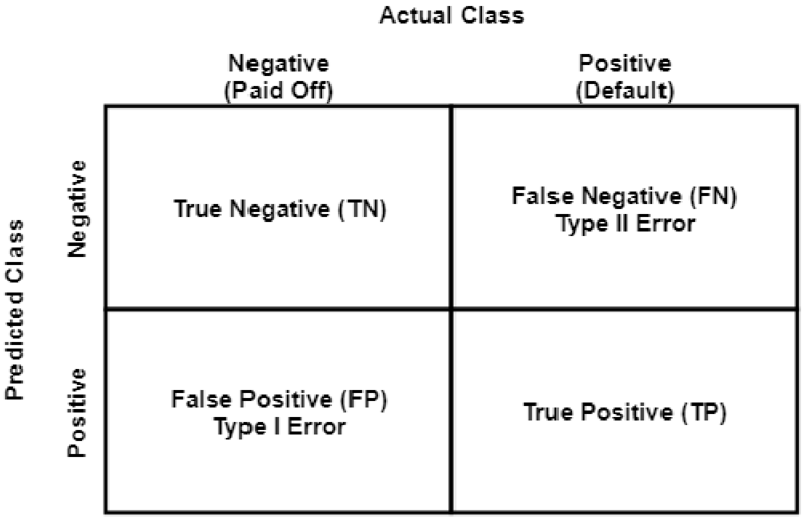
Moreover, apart from algorithm optimizations, XGBoost's design also provides efficient operation of hardware resources. The framework achieves cache awareness by allocation of internal buffers in each thread to store gradient statistics. Additional augmentations such as 'out-of-core' computing utilize available disk space when dealing with big data sets which do not fit into memory.

In order to tune the hyperparameters of the XGBoost model, we are going to apply 30 iteration random grid search with 5-fold cross validation.

## Performance Metrics

Based on the models' performance evaluation results, the decision is made which classifier to accept for further usage. Thus, choosing the right criteria, the suitable performance metrics, is a key step towards making a justified decision.

Before elaborating on representative classification quality measurement techniques further below, it is worth outlining the concepts behind some of the metrics underlying the discussion as illustrated by He and Garcia (2009). It should also be taken into consideration that, in the classification framework of this research, minority class (defaulted loans) are identified as positive class, while the majority class (paid off loans) instances are recognized as negative class examples. In Figure 3, True Positives and True Negatives correspond to the correctly classified instances of defaulted loans and paid off loans respectively, while False Positives and False Negatives identify the misclassified respective examples of actual paid off loans and defaulted loans.



| | | Actual Class | |
|---|---|---|---|
| | | Negative<br>(Paid Off) | Positive<br>(Default) |
| Predicted Class | Negative | True Negative (TN) | False Negative (FN)<br>Type II Error |
| | Positive | False Positive (FP)<br>Type I Error | True Positive (TP) |

*Figure 3.* Confusion matrix.

Accuracy metric in binary classification is defined as the share of correct predictions (sum of True Positives and True Negatives) out of all the prediction instances, or can be defined as Percentage of Correctly Classified cases (PCC) for short. Error Rate, the opposite of Accuracy, represents the percentage of incorrect predictions. Specificity, or True Negative Rate, measures the share of properly identified negatives. Sensitivity, also known as Recall, and True Positive Rate, is the proportion of correctly classified positive class instances out of all actual positives. Precision, or Positive Prediction value, stands for proportion of correctly labeled positives out of all instances labeled positive by the classifier. The Receiver Operating Characteristic curve is depicted by plotting True Positive Rate (Recall) against the False Negative Rate; Area Under this Curve (ROC-AUC) is used as an estimate of classifier's discriminatory capability.

In case of imbalanced binary classification, popular metrics for the assessment of model's predictive power such as Accuracy and ROC-AUC can be misleading (Saito & Rehmsmeier, 2015). Accuracy scores alone hold little value, if there is a substantially large majority class, for example, simply labeling every instance as negative, in the data set with 80% actual negatives, would give an 80% accuracy.

As for ROC-AUC, Saito and Rehmsmeier (2015) recommends relying on addition of the Precision-Recall curve instead, and subsequent Area Under Curve score (PR-AUC), since the interpretation of solely ROC curve can be misleading when dealing with class imbalance (J. Davis & Goadrich, 2006; Fawcett, 2006).

Additionally, performance metrics such as Matthews Correlation Coefficient (MCC) (Matthews, 1975) and $F_2$ measure can be relevant in binary classification. Chicco and Jurman (2020) demonstrates how MCC is more reliable score in assessing binary classifications than Accuracy or F-1 score, being that it is equally directly proportionate to desirable results in all of the four confusion matrix categories. While the F-measure (F-1) is known as harmonic mean of Precision and Recall, it's variation, the $F_2$ score, incorporates weighting parameter beta that places twice the importance on Precision (Sasaki, 2007), used for cases when identifying positive observations is of higher priority. G-mean, or geometric mean, measure is similar to F-1 score and will be omitted from selection for the same reason.

Nonetheless, we will still include Accuracy (PCC) and Area Under Curve of Receiver Operating Characteristic (ROC-AUC) for contextual reference, noting that these measures should not be relied upon in isolation for often being misleading in the context of imbalanced predictions. Consequently, for the purposes of our experimental analysis the following performance measurements will be additionally displayed: PCC, ROC-AUC, Sensitivity (Recall), Specificity, Precision, PR-AUC, MCC, and $F_2$ score. For objective interpretation the emphasis should be put on Sensitivity (Recall) and Specificity recall for the percentage of correctly identified instances among the positive and negative classes respectively, as well as the rest of the chosen measures relevant to imbalanced binary classification nature of the classification task. For concise description of the metrics ultimately selected see Table 1 below.

**Table 1**

*Selected performance metrics for classification assessment*

| Metric | Formula | Description |
|---|---|---|
| Accuracy (PCC) | $\frac{TP + TN}{TP + TN + FP + FN}$ | The percentage of correctly classified units out of total classifications. |
| Sensitivity (Recall) | $\frac{TP}{TP + FN}$ | The proportion of actual positives that are correctly classified. |
| Specificity | $\frac{TN}{TN + FP}$ | The proportion of actual negatives that are correctly classified. |
| Precision | $\frac{TP}{TP + FP}$ | The percentage of actual positives among the instances classified as positives. |
| ROC-AUC | Calculation of area under the curve | ROC curve plots the tradeoff between Sensitivity and Specificity. The higher the area, the more overall discriminatory power a classifier has. |
| PR-AUC | Calculation of area under the curve | PR curve plots the tradeoff between Precision and Recall. The higher the area, the more discriminatory power focused on positive class a classifier has. |
| MCC | $\frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$ | Defines correlation between the predicted classes and ground truth. It is informative due being directly proportional to the performance of all four confusion matrix categories. |
| $F_2$ measure | $\frac{(5 \times Precision \times Recall)}{(4 \times Precision + Recall)}$ | Combines precision and recall with double the emphasis on recall. |

*Note.* The formulas use elements of the confusion matrix illustrated in Figure 3.

## Methodology of the Research in a Nutshell

To summarize, the methodology section of our study addresses the mentioned class imbalance issue, the selection of the machine learning classification algorithms for the experiment, then describes the inner workings of the selected classifiers, and, lastly, elaborates upon the appropriate performance measurement approaches in the case of binary imbalanced classification.

The classification at hand is considered imbalanced, as there is about five paid off loans for each loan default in our dataset, and the nature of lending business in general is so that the majority of potentially non-performing loans are declined at the application stage, spurring the given imbalance. We generally review the techniques for handling imbalanced classification such as cost-sensitive learning, sampling-based methods, and other approaches. Subsequently, we decide to make use of straightforward technique of random under-sampling, which randomly removes the quantity of the majority class observations exceeding that of a minority class, balancing the class distribution evenly.

In regards to choosing the algorithms for the experimental part of this study, we justify our decision according to two main points: their popularity, as indication of their reliability, and computational efficiency, as we conduct the experiment by the means of average consumer laptop. To this end, we refer to the data provided by Kaggle, arguably the largest online community of data professionals and practitioners; according to this data, we choose Random Forest and XGBoost classification algorithms, along with Logistic Regression classifier to serve as the performance benchmark. However, we dismiss other popular algorithm, the Neural Networks, mainly for its comparatively high computational requirements, and difficult interpretation.

Concerning the chosen algorithms, we concentrate on providing a general understanding of Random Forest and XGBoost classification algorithms for the audience. Since both are commonly based on decision trees classification algorithms, we begin by describing decision trees classifiers. Note that here we provide only general logic behind the algorithms in question, for more detailed explanation see the respective subsections of methodology.

Decision trees algorithm is a non-parametric supervised learning method utilized for both classification and regression. Non-parametric characteristic of it means it does not make assumptions about the form of a function in question (e.g., data distribution), while supervised designation refers to models dealing with the data that contains records for both the explanatory and the predicted variable, also identified as input and output data respectively.

These trees "learn" from data to approximate sine-like curve with a set of if-then-else decision rules. The larger or, more specifically, deeper the tree, the more complex decision rules and fitter the model. Model fit refers to how well the given model predicts the given data, or how many instances it labels/predicts correctly in case of classification.

Decision tree classifier is built in a tree-like or a top-down flow chart structure of decision nodes connected by branches. Precisely, said algorithm breaks the data down in incrementally smaller subsets at each node based on specific questions about the data point value for a certain variable; the answers to those questions are the resulting branches that lead to the consequent nodes where the process is repeated with new such questions until the terminal (leaf) node is reached. The final result is a decision tree with decision nodes, starting from a root node, and leaf nodes which represent the classification decision. Decision trees can intake both categorical and numerical data.

The top-down "growing" of a decision tree from root node to leaf nodes depends on partitioning the data into subsets that contain more homogenous instances, or instances with similar values. Entropy generally calculates the homogeneity of such samples; entropy of zero corresponds to a completely homogenous sample, contrastingly, the entropy of one represents a sample that is equally divided. Thus, at each step the data is split on different variables, the entropy for the resulting corresponding branches is calculated, then proportionally added to get the total entropy for the split on a given variable. The resulting entropy is subtracted from the entropy before the split. The produced value of a decrease in entropy is called the Information Gain. Consequently, the predictor with the largest information gain is chosen as the decision node at each step.

However, the decision trees tend to describe the training data too well, or overfit, and consequently fail to produce high quality predictions on the unseen data, or to generalize; even despite the tree pruning, or the reduction of the number of nodes, meant to reduce overfitting. Although the smaller size decision trees do generalize better, they are prone to either overfit or perform poorly in the presence of high-dimensional data.

Random Forests machine learning algorithm on the other hand addresses the overfitting issue of decision trees by building a multitude of decorrelated unpruned decision trees in parallel and calculating the average of their resulting decision outputs to produce a classification probability for each data instance. The given decision trees are decorrelated in part due to bootstrapping, which a sampling technique with replacement. Sampling with replacement essentially means that when randomly selecting one instance from the original data pool to include in a sample we do not remove it from the original pool, meaning that it can appear again in the created sample, or that it does not affect the probability of other data points being included to the sample, making these probabilities have zero correlation. Moreover, Random Forest "mtry" parameter sets the number of randomly sampled variables at each decision node calculation. The total amount of decision trees constituting a Random Forest usually numbers in hundreds.

The resulting Random Forest predictions tend generally provide higher quality classifications than single decision tree classifiers. Random Forest is designated as bagging, or bootstrap aggregating type algorithm, because, as we saw, it aggregates the predictions of a number of learners built on bootstrap samples.

Another tree-based algorithm selected for the experimental part of this study is the XGBoost, or Extreme Gradient Boosting. In contrast to bagging-based structure of the Random Forests, the XGBoost is regarded as a gradient boosting type algorithm.

Gradient boosting is an iterative learning approach that trains the base learners, decision trees, in succession, with each consecutive model trained to predict the residuals, or prediction errors, of the preceding learner. Essentially, descending the gradient, or slope, of the loss function the local minima of that loss function is found, which provides the least amount of model prediction errors. The gradient is descended by adjusting the value of model coefficients accordingly; the learning rate parameter value dictates how large of a step is made in descending direction. This way, in gradient boosting, the first decision tree's residuals are used to construct a new improved tree learner to predict them, then the results of the two learners are aggregated leading to smaller overall error rate, the aggregated model's error is improved upon by the consequent decision tree after that at each iteration, and the process continues until there are no gains or iterations limit is reached

XGBoost is a powerful improvement upon a gradient boosting framework, that was designed to dramatically increase the latter by numerous optimizations on algorithm, software, and hardware levels as described in detail in the corresponding subsection of Methodology. In short, XGBoost has a multitude of parameters aside from already mentioned learning rate to control for overfitting, such as maximum depth of trees, regularization and randomization parameters, etc. XGBoost performs the same amount of computations at exceedingly faster rate than the basic gradient boosting machines.

After the methodological description of machine learning algorithms in question, we elaborated upon the appropriate performance metrics for the unbalanced binary classification case. Taking into account that most machine learning algorithms are built in a way to maximize the overall prediction accuracy and the popular metrics correspond to that fact, the performance measurements that emphasize the higher relative importance of minority class predictions, or identifying default on loans, should be used instead in our case.

The focus is placed mainly on Sensitivity and Specificity, but also other complementary metrics such as Precision, Area Under Precision Recall Curve, Matthews Correlation coefficient and $F_2$ score. Moreover, two popular metrics, Accuracy and Area Under Receiver Operating Characteristic Curve, are pointed out to be useful in context of the above-mentioned metrics, but misleading on their own due to skew towards the majority class predictions. Sensitivity corresponds to the rate of correctly identified minority class, or default instances, out of all the presented minority class; while Specificity performance the same measurement for the majority class cases. These two metrics alone provide simple, yet sufficient information about the classifiers' performances to make an informed judgement, while the rest of the chosen measurement techniques provide additional evaluation and comparison grounds.

## Data Analysis and Preparation

The given section discusses the choice of the source of data, and data pre-processing, including feature filtering and transformation, training-test data split, outlier detection, resampling for imbalanced data, and feature sets' selection for the respective models.

A seemingly well-known fact in the domain of machine learning, the performance of machine learning algorithms, classifiers included, depends on quality of the source data (Kotsiantis et al., 2007). Thus, we first need to responsibly select a data set to conduct our calculations. In terms of the size of the publicly available loan data amongst online P2P lending platforms, Lending Club,  which had been one of the largest market players (P2PMarketData.com, 2020)[2], is probably still the most extensive array of loan records for the market, to the best of our knowledge. However, the loans issue prior to year 2017 should most likely not be considered for analysis due to the following circumstances. During the year 2016, there was a scandal surrounding a prior history of Lending Club machinations and misrepresentation regarding the quality of its loans' portfolio, which led to eventual resignation of the man behind it, it's founder and CEO, Renaud Laplanche (Popper, 2018; Chafkin & Buhayar, 2016).

Moreover, Lending Club underwent positive underwriting policy changes that reportedly improved it's loans' performance in the first half of that same year (Wu, 2016). The combination of prior points leads to loans originated only during the period from the beginning of year 2017 and later being eligible for the analysis. The loans data further discussed was obtained from the Lending Club website through a registered potential borrower account, where it was publicly available for its registered users.

Therefore, the peer-to-peer loans with the origination year of 2017 are the most dated at our disposal. In terms of the payback period, Lending Club have been offering to accommodate only 36 and 60-motnhs loans throughout its history. Correspondingly, despite the 60-months loans still approaching its maturity date, the 36-months loans originated in 2017 are past the respective due dates of their original

[2] Lending Club has effectively retired its note issuing of P2P notes due to restructuring to become "full-spectrum fintech marketplace bank" and offer new products to its clients as reported on Lending Club's website. (https://help.lendingclub.com/hc/en-us/articles/360050574891-Important-Updates-to-the-LendingClub-Notes-Platform)

payback period at the time of this research. Upon analyzing the data, we see that it has 150 features for every origination year and 443579 entries in 2017 (see Appendix B for descriptions of all 150 variables provided by Lending Club data records).

Firstly, looking at the “loan_status” column it can be seen that among the 443579 loans funded during that year 312486 were “Fully Paid” off, 34 are labeled as “Default”, and 70147 were “Charged Off”, while the rest are either “In Grace Period” or “Late (16-30 days)”/ “Late (31-120 days)”. Nevertheless, the definitions for “Default” and “Charged Off” loans provided on the Lending Club official website are difficult to distinguish from each other, since the charged off loans are removed from investor’s account balance after 120 days of delinquency, but “default” loans having no specified timeframe are not removed from the balance [3]; however, loans with delinquency of less than 120 days are designated as “Late” as mentioned above. Therefore, this study will designate both “Charged Off” and “Default” status loans as loan defaults for convenience. Meanwhile, loans categorized as current, late, and in grace period should be discarded from the data set. The resulting variable “loan status” was then label encoded, with “1” representing default loans and “0” designating the fully paid ones.

The next step is to reduce the high-dimensional data to a set of relevant features. Since the goal of our experiment is to construct and discuss predictive models capable of generalizing on unseen data of real prospective loans, the features containing the information about the loan’s state recorded after its date of origin should be thoroughly removed first (e.g., features regarding amounts of received loan payments, hardship plan attributes, etc.). Then, columns carrying no relevant informational value (e.g., id’s, URLs, title, etc.), redundant features, variables consisting of large proportions of missing instances, and attributes displaying zero or near-zero variance are detached too (see Appendix C for the information on all the features that were dismissed out of the original data).

[3] The mentioned ambiguous definitions are stated on the respective information page of the Lending Club website (https://help.lendingclub.com/hc/en-us/articles/216127747)

Additionally, the following features were transformed. Categorical attribute about employment length of the borrower, “emp_length”, was transformed into numerical of year values 1 through 10, with year 1 values incorporating those of employment experience with one years and below, while value of 10 years represented the former category of borrowers with ten years of working experience and above. Features representing the date of earliest credit line, “earliest_cr_line”, opening and loan issue date, “issue_d”, were combined into feature “cr_hist_months”, denominating the total time period of credit history in months prior to loan origination. Categorical variable “grade”, designating grades “A” through “G” in descending order of quality assigned by Lending Club, was transformed into numerical one with values 1 through 7 correspondingly. Features “fico_range_high” and “fico_range_low”, representing upper and lower boundary ranges the borrower’s FICO score at loan origination date, were combined into one single variable “fico_mean” which stands for their average.

The resulting dataset contained the total of 326,546 observations, with 268,905 paid off loans and 57,641 loan defaults resulting in a class imbalance of 4.67 to 1. The data was then randomly split into training and test sets by 80:20 ratio respectively, resulting in 261,237 observations for the training sample and 65,309 instances for the test data.

After that, from “solitude” R package, we implemented robust and computationally effective anomaly detection algorithm, Isolation Forest by Liu et al. (2012), on training data, which essentially assigns scores for interpreting which instances can be considered outliers; by applying 1.5 times interquartile range value subtraction and addition to first and third quartiles of these scores respectively, and removing observations lying outside the defined boundaries, 13,947 outliers were identified and removed. These outliers were dismissed in order to avoid biases during additional information gain-based feature selection and consequential training processes. This feature set of 56 input variables was used for training of XGBoost classifier (see Appendix C for the list of dismissed and retained attributes.).

To address the notable class imbalance, we decided to resort to straightforward approach of random under-sampling to evenly rebalance the data by randomly removing a number of majority class cases exceeding the quantity of instances in minority class. The down-sampled training dataset contained 83,542 total observations with equal 41,771 instances of both classes.

Under-sampling procedure was conducted not only to avoid the classifiers training favoring prevalent negative class instances, but also in order to allow for balanced feature evaluation with information gain scores without the bias towards features describing mostly the majority class. Thus, the input variables of the resulting training sample were further analyzed with Information Gain method using "FSelector" R package in order to identify the strongest predictors (see Appendix D for the information gain scores and the resulting feature reduction to 28 total variables for the data used during the training of logistic regression and random forest classifiers). Information gain is an entropy-based attribute assessment method, popular in the domain of machine learning (Lei, 2012)

## Experimental Results

This section is organized in the following way. Firstly, the results of the selected classifiers are presented, examined, and assessed in contrast to each other. In the second part, the given results are discussed in the context of the literature.

### Results Overview

In this study, we evaluate how well popular classifiers such as XGBoost, Random Forest, and Logistic Regression perform in the task of default predictive modelling for peer-to-peer loans of Lending Club. In order to achieve said goal, six relevant performance metrics were chosen. To address class imbalance, random under-sampling was implemented for every classifier involved. For concise depiction of the performance measurements refer back to Table 1.

The resulting comparison across the selected metrics[4] as shown in Table 2 demonstrates that the three selected classifiers possess no sizeable differences in predictive power, so we considered ranking these classifiers unreasonable in such circumstances. Nonetheless, in regards to computational efficiency, as XGBoost demonstrated the ability to process larger amounts of data notably faster.

**Table 2**

*Classification results*

| Classifier | Predictive performance measurements | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | PCC | ROC-AUC | Sensitivity | Specificity | Precision | PR-AUC | MCC | $F_2$ |
| LR | 0,6673 | 0,6610 | 0,6512 | 0,6707 | 0,2973 | 0,2697 | 0,2519 | 0,3336 |
| RF | 0,6462 | 0,6573 | 0,6744 | 0,6402 | 0,2862 | 0,2632 | 0,2432 | 0,3234 |
| XGBoost | 0,6392 | 0,6518 | 0,6712 | 0,6324 | 0,2809 | 0,2588 | 0,2343 | 0,3179 |

[4] Refer to Figure 3 and Table 1 in Data Analysis and Preparation section for a concise account of the performance metrics logic

Overall, in terms of sensitivity and specificity, the presented classifiers did not notably diverge from the type of performance demonstrated by related works as can be seen in Table 3.

**Table 3**

*Results from the experiments of related studies' premier classifiers*

| Study | Classifier | Predictive performance measurements | | | |
|---|---|---|---|---|---|
| | | PCC | ROC-AUC | Sensitivity | Specificity |
| Boiko Ferreira et al. (2017) | LR+SM | – | 0.6500 | 0.6900 | 0.6200 |
| Zanin (2020) | Lasso ensemble +ROUS | 0.6702 | 0.6834 | 0.5769 | – |
| Niu et al. (2020) | REMDD | – | 0.7002 | 0.6715 | 0.7299 |
| Moscato et al. (2021) | RF+RUS | 0.6400 | 0.7170 | 0.6300 | 0.6800 |
| Song et al. (2020) | RF+RUS | 0.5912 | 0.6207 | 0.6623 | 0.5791 |
| Namvar et al. (2018) | RF+RUS | 0.6920 | 0.6900 | 0.7170 | 0.5820 |

*Note.* This table demonstrates the respective researches best performing classifiers as discussed in more detail in Literature Review part of this study. LR+SM = logistic regression with SMOTE resampling. Lasso ensemble + ROUS = Lasso regression regularized ensemble model with random over- and under-sampling. REMDD = resampling ensemble model based on data distribution. RF+RUS = random forest with random under-sampling.

However, depending on investor's preferences, Provost (2000) argues that the thresholding may need to be applied on the given models in order to increase sensitivity, at the cost of reducing specificity, to identify a higher proportion of actual default loans. Table 4 shows the resulting changes in performance of the three classifiers after setting the classification threshold from standard 0.5 value to a lower mark equal to 0.4 value, meaning that every test sample instance assigned positive class probability greater than 40% by a given classifier will be labeled as loan "default" for prediction purposes.

**Table 4**

*Classification results with threshold 0.4*

| Classifier | Predictive performance measurements | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | PCC | AU-ROC | Sensitivity | Specificity | Precision | AU-PRC | MCC | $F_2$ |
| LR | 0,5277 | 0,6436 | 0,8226 | 0,4646 | 0,2474 | 0,2400 | 0,2222 | 0,2876 |
| RF | 0,5081 | 0,6395 | 0,8424 | 0,4366 | 0,2423 | 0,2363 | 0,2182 | 0,2826 |
| XGBoost | 0,3676 | 0,5907 | 0,9353 | 0,2461 | 0,2098 | 0,2086 | 0,1685 | 0,2483 |

While such approach may reduce the general performance indicators of a model across the board, it does allow for a greater sensitivity performance, or the stronger identification of default loans. Among other metric indicators, specificity obviously does drop, but given the nature of credit scoring, the investors are most likely to face the same class imbalance in P2P lending specifically. Moreover, the given thresholding approach does improve the ratio of true negatives and false positives, increasing the prevalence of the former in the resulting pool of predicted negative class instances, as shown in Table 5.

**Table 5**

*Comparison of the distribution of negative class predictions before and after thresholding*

| Classifier | True Negatives (TN) | False Negatives (FN) | TN/FN ratio |
|---|---|---|---|
| LR | 35984 | 4053 | 8.89 |
| RF | 34441 | 3748 | 9.19 |
| XGBoost | 34022 | 3785 | 8.99 |
| LR, threshold 0.4 | 24995 | 2042 | 12.24 |
| RF, threshold 0.4 | 23486 | 1814 | 12.95 |
| XGBoost, threshold 0.4 | 13242 | 745 | 17.78 |

## Results Discussion

To begin with, it is worth pointing out that given higher computational means, beyond that of an average commercial laptop, the above tree-based algorithms are likely to provide predictions of higher quality due to the ability to run increased number of distinct hyperparameter iterations for mitigating overfitting issues in shorter amounts of time. Furthermore, it can also be seen how popular performance measurements such as area under ROC curve should be evaluated in combination with other metrics relevant to imbalanced default prediction tasks, such as the ones applied above.

The provided results seem generally coinciding with those of revised related literature, not achieving particularly outstanding loan default prediction results. Nevertheless, unlike any other related study to the best of our knowledge, we have shown that investors should apply thresholding to their classification results in accordance with their preferences, portfolio needs and risk tolerance, as said technique does reduce the proportion of unidentified defaults among the predicted to be paid off loans as already showcased in Table 5. Therefore, such approach towards default prediction can be seen as reasonable to make further investment decisions upon the resulting classification, although the initial results are not particularly outstanding.

Nonetheless, the absence of high performing predictive models across the respective research field’s findings likely speaks towards the mentioned problem of information asymmetry in the industry, and also insufficiency of conventional data provided by Lending Club for individual investors to build scoring models of high predictive power. The reason for such informational asymmetry being P2P providing opportunity for and attracting candidate borrowers without credit history, or specific loan needs that are likely not to be covered by traditional banks. As a result, conventional data does not guarantee high reliability in regards to the credit scoring of these types of borrower candidates.

Furthermore, while traditional logistic regression, which is a linear model, requires a selection of a limited number of predictor variables that describe the dependent feature as also seen in our experiment, state-of-the-art machine learning algorithms such as XGBoost can handle high dimensionality of the input data effectively and are able to capture non-linear relationships present, while also requiring significantly less time for computations. Accordingly, Machine learning is well suited for implementation alternative, or unconventional data, for such data is less structured and more abundant in features (Aggarwal, 2020).

Alternative data itself can be an ambiguous concept, but generally "alternative" designation signifies the data source being distinct from the conventional descriptive features used by traditional banks for credit scoring, however, according to account by Aggarwal, it can be distinguished in two types: non-credit financial data and non-credit non-financial data. The former is represented by such examples as rental and mobile phone payment data, while the latter includes instances of not only education and employment experience, but also social media activity, online behavioral data, etc. While the Lending Club loan records have features such as home ownership status, employment length, they lack attributes sourced from, for example, social media, online behaviors, which could contribute a positive difference to the prediction results.

Machine learning classifiers are likely to provide greater results in credit scoring, including peer-to-peer loans, if provided access to such data in addition to traditional features, as illustrated in the study by Óskarsdóttir et al. (2019), where the authors showcase how including alternative data of mobile phone data, namely by constructing call networks through call records, in addition to traditional data features can enhance loan default prediction performance of machine learning classifiers.

To summarize, the proposed thresholding classification approach for making default predictions in P2P lending investment can reasonably be used by the individual investors. Furthermore, alternative data in combination with machine learning algorithms is likely to allow for higher quality predictive modelling by individual investors, resulting in more reliable investment decisions, but also to allow for a greater financial inclusion of larger number of loan applicants (K. T. Davis & Murphy, 2016; Wyman, 2017). However, in accordance with Aggarwal (2020) and Óskarsdóttir et al. (2019), we stress that the call for alternative data use leads to a series of discussions about access to what kinds of alternative data can be considered ethical, where are the boundaries of individual privacy, and how the official regulatory bodies throughput the world should address such issues.

# Conclusion

This thesis explores the research field of default prediction in peer-to-peer lending on the data provided by Lending Club, and tackles the question of how effectively individual investors can implement machine learning on the respective data for credit scoring purposes. After a thorough examination of existing related literature, we follow up with conducting our own experiment with attentive and transparent data pre-processing and exploring classification effectiveness of popular machine learning algorithms, namely Random Forest and XGBoost, for the classification task in question. The ensuing results of these two classifiers are then compared between themselves and to that of lending industry benchmark, the Logistic Regression.

Our study recognizes the imbalanced nature of loan default records and addresses it through the means of random under-sampling applied to the training data, and carefully chosen evaluation metrics suitable for interpretation of imbalanced classification results and eventual comparison of distinct classifiers, placing main focus on standard measurements of Sensitivity and Specificity, in combination with Area Under Precision Recall Curve, Matthews Correlation Coefficient, $F_2$ score. We also add Accuracy and Area Under Curve of Receiver Operating Characteristic, but emphasize that these metrics should not be regarded in isolation due to often being misleading in imbalanced classification cases.

We justify our specific choice of machine learning classifiers by their recorded popularity among data professionals and practitioners according to research by Kaggle, but also by their computational efficiency when dealing with large data sets. After developing a detailed description of said algorithms' logical structure and characteristics, this thesis moves on to evaluating the results of the three classifiers in question, comparing them in between themselves and with the findings of comparable credible related literature discovered previously in Literature Review.

The provided findings of this study are generally consistent in terms of the predictive power with those of related literature. However, in contrast to the existing researches, our study highlights the ability and need for implementing thresholding technique on classifiers predicted default probabilities in order to improve the utility of the consequent results and better accommodate the portfolio needs of individual investors while also providing the opportunity for a higher and more reliable portfolio returns. The illustrated threshold lowering allows for a sizeable reduction of actual loan defaults inside the pool of predicted to be paid off loans, which in turn has positive implications for further investment decisions by the investors.

Furthermore, the highlighted general consistency of the experimental findings across the specific research field in question, or the absence of outstanding predictive modelling results, challenges the notion of data publicly provided by P2P lending platforms such as Lending Club being sufficient for the investors and industry as a whole to succeed. This calls for extension of alternative data use in P2P lending markets to enhance the credit scoring results, consequently increasing the benefits for platforms, investors, and loan applicants alike in perspective of the potential for higher return rates for the market.

As machine learning is known to excel in the context of alternative data, it could significantly benefit not only investors, but also P2P lending organizations themselves. However, a wide implementation of alternative data raises additional questions for further research on legal regulation and ethics regarding the use, collection, and availability of alternative data for the online P2P lending market as a whole.

## References

Abdou, H. A., & Pointon, J. (2011). Credit scoring, statistical techniques and evaluation criteria: a review of the literature. Intelligent systems in accounting, finance and management, 18(2-3), 59-88. https://doi.org/10.1002/isaf.325

Aggarwal, N. (2021). The norms of algorithmic credit scoring. The Cambridge Law Journal, 80(1), 42-73. https://doi.org/10.1017/S0008197321000015

Akinsola, J. E. T. (2017). Supervised Machine Learning Algorithms: Classification and Comparison. *International Journal of Computer Trends and Technology (IJCTT)*, *48*, 128–138. https://doi.org/10.14445/22312803/IJCTT-V48P126

Akram, S., & Ann, Q. ul. (2015). *Newton Raphson method*. *6*(7), 1748–1752.

Ala'raj, M., & Abbod, M. F. (2016). Classifiers consensus system approach for credit scoring. *Knowledge-Based Systems*, *104*, 89–105. https://doi.org/10.1016/j.knosys.2016.04.013

Al-qerem, A., Al-Naymat, G., & Alhasan, M. (2019). Loan Default Prediction Model Improvement through Comprehensive Preprocessing and Features Selection. *2019 International Arab Conference on Information Technology (ACIT)*, 235–240. https://doi.org/10.1109/ACIT47987.2019.8991084

AltFi (2020). *Alternative Lending State of the Market Annual Report 2020*. Retrieved from https://www.altfi.com/downloads/alternative-lending-state-of-the-market-report-2020.pdf

Arora, N., & Kaur, P. D. (2020). A Bolasso based consistent feature selection enabled random forest classification algorithm: An application to credit risk assessment. *Applied Soft Computing*, *86*, 105936. https://doi.org/10.1016/j.asoc.2019.105936

Ayyadevara, V. K. (2018). Gradient Boosting Machine. In V. K. Ayyadevara, *Pro Machine Learning Algorithms* (pp. 117–134). Apress. https://doi.org/10.1007/978-1-4842-3564-5_6

Bennett, R., & Kottasz, R. (2012). Public attitudes towards the UK banking industry following the global financial crisis. *International Journal of Bank Marketing*, *30*(2), 128–147. https://doi.org/10.1108/02652321211210877

Bhavsar, H., & Ganatra, A. (2012). A Comparative Study of Training Algorithms for Supervised Machine Learning. *International Journal of Soft Computing and Engineering (IJSCE)*, *2*.

Boiko Ferreira, L. E., Barddal, J. P., Gomes, H. M., & Enembreck, F. (2017). Improving Credit Risk Prediction in Online Peer-to-Peer (P2P) Lending Using Imbalanced Learning Techniques. *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, 175–181. https://doi.org/10.1109/ICTAI.2017.00037

Carbo-Valverde, S., Maqui Lopez, E., & Rodriguez-Fernandez, F. (2013, August). Trust in Banks: Evidence from the Spanish Financial Crisis. In 26th Australasian Finance and Banking Conference. https://doi.org/10.2139/ssrn.2310273

Chafkin, M., & Buhayar, N. (2016, August 18). How Lending Club's Biggest Fanboy Uncovered Shady Loans. *Bloomberg.Com*. https://www.bloomberg.com/news/features/2016-08-18/how-lending-club-s-biggest-fanboy-uncovered-shady-loans

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics, 21*(1), 6. https://doi.org/10.1186/s12864-019-6413-7

Claessens, S., Frost, J., Turner, G., & Zhu, F. (2018). Fintech credit markets around the world: size, drivers and policy issues. *BIS Quarterly Review September*.

Cummins, M., Lynn, T., Mac an Bhaird, C., & Rosati, P. (2019). Addressing Information Asymmetries in Online Peer-to-Peer Lending. In T. Lynn, J. G. Mooney, P. Rosati, & M. Cummins (Eds.), *Disrupting Finance* (pp. 15–31). Springer International Publishing. https://doi.org/10.1007/978-3-030-02330-0_2

Davis, J., & Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. *Proceedings of the 23rd International Conference on Machine Learning - ICML '06*, 233–240. https://doi.org/10.1145/1143844.1143874

Davis, K. (2016). Peer-to-peer lending: structures, risks and regulation. *JASSA*, (3), 37-44.

Dixon, M. F., Halperin, I., & Bilokon, P. (2020). *Machine Learning in Finance: From Theory to Practice*. Springer International Publishing. https://doi.org/10.1007/978-3-030-41068-1

Elkan, C. (2001). *The Foundations of Cost-Sensitive Learning*. 6.

Elrahman, S. M. A., & Abraham, A. (2013). A Review of Class Imbalance Problem. *Journal of Network and Innovative Computing, 1*, 332–340.

Emerson, S., Kennedy, R., O'Shea, L., & O'Brien, J. (2019, May). Trends and applications of machine learning in quantitative finance. In *8th international conference on economics and finance research (ICEFR 2019)*.

P2PMarketData. (2021). *EU Peer-to-Peer Lending & Equity Funding Volumes*. Retrieved from https://p2pmarketdata.com/p2p-lending-funding-volume-eu/

Everett, C. R. (2014). Origins and Development of Credit-Based Crowdfunding. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.2442897

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., & Herrera, F. (2018). *Learning from Imbalanced Data Sets*. Springer International Publishing. https://doi.org/10.1007/978-3-319-98074-4

Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics, 29*(5), 1189–1232.

Gillespie, N., & Hurley, R. (2013). Trust and the global financial crisis. In R. Bachmann and A. Zaheer (Eds.), *Handbook of advances in trust research* (pp. 177-203). Edward Elgar Publishing. https://doi.org/10.4337/9780857931382.00019

Giudici, P., & Misheva, B. H. (2017, April). *Scoring models for P2P lending platforms: An evaluation of predictive performance*. Paper presented at Statistics and data science: new challenges, new generations.

Goldbloom, A. (2016). *What algorithms are most successful on Kaggle?* [Kaggle.com]. Retrieved from https://www.kaggle.com/antgoldbloom/what-algorithms-are-most-successful-on-kaggle

Guo, Y., Zhou, W., Luo, C., Liu, C., & Xiong, H. (2016). Instance-based credit risk assessment for investment decisions in P2P lending. *European Journal of Operational Research*, *249*(2), 417–426. https://doi.org/10.1016/j.ejor.2015.05.050

Hastie, T. (2020). Ridge Regularization: An Essential Concept in Data Science. *Technometrics*, *62*(4), 426–433. https://doi.org/10.1080/00401706.2020.1791959

He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263–1284. https://doi.org/10.1109/TKDE.2008.239

He, H., & Ma, Y. (Eds.). (2013). *Imbalanced learning: Foundations, algorithms, and applications*. John Wiley & Sons, Inc.

Hollis, A., & Sweetman, A. (1997). Microcredit in Pre-Famine Ireland. In *Economic History* (No. 9704002; Economic History). University Library of Munich, Germany. https://ideas.repec.org/p/wpa/wuwpeh/9704002.html

Hou, X. (2020). P2P Borrower Default Identification and Prediction Based on RFE-Multiple Classification Models. *Open Journal of Business and Management*, *08*(02), 866–880. https://doi.org/10.4236/ojbm.2020.82053

Kaggle. (2021). *State of Data Science and Machine Learning 2020*. Retrieved from https://www.kaggle.com/kaggle-survey-2020

Khan, S., Goswami, A., & Kumar, V. (2020). *Peer to Peer Lending Market by Business Model (Alternate Marketplace Lending and Traditional Lending), Type (Consumer Lending and Business Lending), and End User (Consumer Credit Loans, Small Business Loans, Student Loans, and Real Estate*

*Loans): Global Opportunity Analysis and Industry Forecast, 2020-2027*. Retrieved from Allied Market Research website: https://www.alliedmarketresearch.com/peer-to-peer-lending-market

Kirasich, K., Smith, T., & Sadler, B. (2018). *Random Forest vs Logistic Regression: Binary Classification for Heterogeneous Datasets*. *1*(3), 25.

Kirby, E., & Worner, S. (2014). *Crowd-funding: An infant industry growing fast*. IOSCO Research Department.

Kotsiantis, S., Kanellopoulos, D., & Pintelas, P. (2005). Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, *30*, 25–36.

Krawczyk, B. (2016). Learning from imbalanced data: Open challenges and future directions. *Progress in Artificial Intelligence*, *5*(4), 221–232. https://doi.org/10.1007/s13748-016-0094-0

Lei, S. (2012). A Feature Selection Method Based on Information Gain and Genetic Algorithm. *2012 International Conference on Computer Science and Electronics Engineering*, 355–358. https://doi.org/10.1109/ICCSEE.2012.97

Lemnaru, C., & Potolea, R. (2012). Imbalanced Classification Problems: Systematic Study, Issues and Best Practices. In R. Zhang, J. Zhang, Z. Zhang, J. Filipe, & J. Cordeiro (Eds.), *Enterprise Information Systems* (Vol. 102, pp. 35–50). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-29958-2_3

Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, *247*(1), 124–136. https://doi.org/10.1016/j.ejor.2015.05.030

Ling, C. X., & Sheng, V. S. (2010). Cost-Sensitive Learning. In C. Sammut & G. I. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 231–235). Springer US. https://doi.org/10.1007/978-0-387-30164-8_181

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2012). Isolation-Based Anomaly Detection. *ACM Transactions on Knowledge Discovery from Data*, *6*(1), 1–39. https://doi.org/10.1145/2133360.2133363

Mairal, J., & Yu, B. (2012). Complexity Analysis of the Lasso Regularization Path. *ArXiv:1205.0079 [Cs, Math, Stat]*. http://arxiv.org/abs/1205.0079

Malekipirbazari, M., & Aksakalli, V. (2015). Risk assessment in social lending via random forests. *Expert Systems with Applications*, *42*(10), 4621–4631. https://doi.org/10.1016/j.eswa.2015.02.001

Mateescu, A. (2015). *Peer-to-Peer lending*. *2*.

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure, 405*(2), 442–451. https://doi.org/10.1016/0005-2795(75)90109-9

Moscato, V., Picariello, A., & Sperlí, G. (2021). A benchmark of machine learning approaches for credit score prediction. *Expert Systems with Applications*, *165*, 113986. https://doi.org/10.1016/j.eswa.2020.113986

Namvar, A., Siami, M., Rabhi, F., & Naderpour, M. (2018). Credit risk prediction in an imbalanced social lending environment. *International Journal of Computational Intelligence Systems*, *11(1)*, 925-935. Retrieved from http://arxiv.org/abs/1805.00801

Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, *7*. https://doi.org/10.3389/fnbot.2013.00021

Niu, K., Zhang, Z., Liu, Y., & Li, R. (2020). Resampling ensemble model based on data distribution for imbalanced credit risk evaluation in P2P lending. *Information Sciences*, *536*, 120–134. https://doi.org/10.1016/j.ins.2020.05.040

Óskarsdóttir, M., Bravo, C., Sarraute, C., Vanthienen, J., & Baesens, B. (2019). The value of big data for credit scoring: Enhancing financial inclusion using mobile phone data and social network analytics. *Applied Soft Computing*, *74*, 26–39. https://doi.org/10.1016/j.asoc.2018.10.004

P2PMarketData. (2021). *P2P Lending Platforms of the World*. Retrieved from https://p2pmarketdata.com/p2p-lending-platforms-of-the-world/

P2PMarketData. (2021). *P2PMarketData—Top 90 P2P Platforms in P2P Lending & Equity*. Retrieved from P2PMarketData. https://p2pmarketdata.com/

Popper, N. (2018, September 28). LendingClub Founder, Ousted in 2016, Settles Fraud Charges. *The New York Times*. https://www.nytimes.com/2018/09/28/technology/lendingclub-renaud-laplanche-fraud.html

Provost, F. (2000). *Machine Learning from Imbalanced Data Sets 101*. 3.

Roth, F. (2009). The effect of the financial crisis on systemic trust. *Intereconomics*, *44*(4), 203–208. https://doi.org/10.1007/s10272-009-0296-9

Ruder, S. (2017). An overview of gradient descent optimization algorithms. Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv, arXiv-1609.* http://arxiv.org/abs/1609.04747

Kotsiantis, S., Kanellopoulos, D., & Pintelas, P. (2007). *Data Preprocessing For Supervised Leaning*. https://doi.org/10.5281/ZENODO.1082415

Saito, T., & Rehmsmeier, M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLOS ONE*, *10*(3), e0118432. https://doi.org/10.1371/journal.pone.0118432

Sasaki, Y. (2007). *The truth of the F-measure*. 5.

Sathya, R., & Abraham, A. (2013). Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligence*, *2*. https://doi.org/10.14569/IJARAI.2013.020206

Serrano-Cinca, C., Gutiérrez-Nieto, B., & López-Palacios, L. (2015). Determinants of Default in P2P Lending. *PLOS ONE*, *10*(10), e0139427. https://doi.org/10.1371/journal.pone.0139427

Song, Y., Wang, Y., Ye, X., Wang, D., Yin, Y., & Wang, Y. (2020). Multi-view ensemble learning based on distance-to-model and adaptive clustering for imbalanced credit risk assessment in P2P lending. *Information Sciences*, *525*, 182–204. https://doi.org/10.1016/j.ins.2020.03.027

Stark, M. (2015). Networks of Lenders and Borrowers: A Rural Credit Market in the Nineteenth Century. *Debtors, Creditors, and Their Networks: Social Dimensions of Monetary Dependence from the Seventeenth to the Twentieth Century*, 99–118.

Sun, J., Lang, J., Fujita, H., & Li, H. (2018). Imbalanced enterprise credit evaluation with DTE-SBD: Decision tree ensemble based on SMOTE and bagging with differentiated sampling rates. *Information Sciences*, *425*, 76–91. https://doi.org/10.1016/j.ins.2017.10.017

Swaper. (2021). 19 P2P Investing Statistics You Need to Know for 2021. Retrieved from https://swaper.com/blog/p2p-investing-statistics/

Teply, P., & Polena, M. (2020). Best classification algorithms in peer-to-peer lending. *The North American Journal of Economics and Finance*, *51*, 100904. https://doi.org/10.1016/j.najef.2019.01.001

Wald, A. (1992). Statistical Decision Functions. In S. Kotz & N. L. Johnson (Eds.), *Breakthroughs in Statistics* (pp. 342–357). Springer New York. https://doi.org/10.1007/978-1-4612-0919-5_22

Wu, J. (2016, May 26). Changes in Lending Club Underwriting—Looking Beneath the Headlines. *MonJa*. https://www.monjaco.com/blog/changes-in-lending-club-underwriting-looking-beneath-the-headlines/

Wyman, O. (2017). *Accelerating Financial Inclusion in South-East Asia with Digital Finance* (Cambodia, Indonesia, Myanmar, Philippines). Asian Development Bank. https://www.adb.org/publications/financial-inclusion-south-east-asia-digital-finance

Yen, S.-J., & Lee, Y.-S. (2009). Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, *36*(3), 5718–5727. https://doi.org/10.1016/j.eswa.2008.06.108

Zanin, L. (2020). Combining multiple probability predictions in the presence of class imbalance to discriminate between potential bad and good borrowers in the peer-to-peer lending market. *Journal of Behavioral and Experimental Finance*, *25*, 100272. https://doi.org/10.1016/j.jbef.2020.100272

## Appendix A.

### The Details of the Software Used Throughout This Study

To run the experiment in question we implemented Modern CSV software for minor adjustments for the csv-files containing Lending Club records, so that it could later be used in R software.

The 4.0.4 version of R was used for the rest of the experimental framework, including data analysis and preparation, classifiers training and testing. The names of the R packages used aside from the base ones were the following: dplyr, caret, reprex, tidyverse, PRROC, pROC. solitude, randomForest, Matrix, xgboost, mltools, MLmetrics.

The original dataset, data pre-processing details, as well as the R code used for the conducted experiment can be accessed through the following link: https://github.com/maratsyzdykov/P2PlendingMLdefaultprediction.

## Appendix B.

## Loan Attributes' Names and Descriptions Provided by Lending Club

| | |
|---|---|
| acc_now_delinq | The number of accounts on which the borrower is now delinquent. |
| acc_open_past_24mths | Number of trades opened in past 24 months. |
| addr_state | The state provided by the borrower in the loan application |
| all_util | Balance to credit limit on all trades |
| annual_inc | The self-reported annual income provided by the borrower during registration. |
| annual_inc_joint | The combined self-reported annual income provided by the co-borrowers during registration |
| application_type | Indicates whether the loan is an individual application or a joint application with two co-borrowers |
| avg_cur_bal | Average current balance of all accounts |
| bc_open_to_buy | Total open to buy on revolving bankcards. |
| bc_util | Ratio of total current balance to high credit/credit limit for all bankcard accounts. |
| chargeoff_within_12_mths | Number of charge-offs within 12 months |
| collection_recovery_fee | post charge off collection fee |
| collections_12_mths_ex_med | Number of collections in 12 months excluding medical collections |
| delinq_2yrs | The number of 30+ days past-due incidences of delinquency in the borrower's credit file for the past 2 years |
| delinq_amnt | The past-due amount owed for the accounts on which the borrower is now delinquent. |
| desc | Loan description provided by the borrower |
| dti | A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income. |
| dti_joint | A ratio calculated using the co-borrowers' total monthly payments on the total debt obligations, excluding mortgages and the requested LC loan, divided by the co-borrowers' combined self-reported monthly income |
| earliest_cr_line | The month the borrower's earliest reported credit line was opened |
| emp_length | Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years. |
| emp_title | The job title supplied by the Borrower when applying for the loan. * |
| fico_range_high | The upper boundary range the borrower's FICO at loan origination belongs to. |
| fico_range_low | The lower boundary range the borrower's FICO at loan origination belongs to. |

| | |
|---|---|
| funded_amnt | The total amount committed to that loan at that point in time. |
| funded_amnt_inv | The total amount committed by investors for that loan at that point in time. |
| grade | LC assigned loan grade |
| home_ownership | The home ownership status provided by the borrower during registration or obtained from the credit report. Our values are: RENT, OWN, MORTGAGE, OTHER |
| id | A unique LC assigned ID for the loan listing. |
| il_util | Ratio of total current balance to high credit/credit limit on all install acct |
| initial_list_status | The initial listing status of the loan. Possible values are – W, F |
| inq_fi | Number of personal finance inquiries |
| inq_last_12m | Number of credit inquiries in past 12 months |
| inq_last_6mths | The number of inquiries in past 6 months (excluding auto and mortgage inquiries) |
| installment | The monthly payment owed by the borrower if the loan originates. |
| int_rate | Interest Rate on the loan |
| issue_d | The month which the loan was funded |
| last_credit_pull_d | The most recent month LC pulled credit for this loan |
| last_fico_range_high | The upper boundary range the borrower’s last FICO pulled belongs to. |
| last_fico_range_low | The lower boundary range the borrower’s last FICO pulled belongs to. |
| last_pymnt_amnt | Last total payment amount received |
| last_pymnt_d | Last month payment was received |
| loan_amnt | The listed amount of the loan applied for by the borrower. If at some point in time, the credit department reduces the loan amount, then it will be reflected in this value. |
| loan_status | Current status of the loan |
| max_bal_bc | Maximum current balance owed on all revolving accounts |
| member_id | A unique LC assigned Id for the borrower member. |
| mo_sin_old_il_acct | Months since oldest bank installment account opened |
| mo_sin_old_rev_tl_op | Months since oldest revolving account opened |
| mo_sin_rcnt_rev_tl_op | Months since most recent revolving account opened |
| mo_sin_rcnt_tl | Months since most recent account opened |
| mort_acc | Number of mortgage accounts. |
| mths_since_last_delinq | The number of months since the borrower's last delinquency. |
| mths_since_last_major_derog | Months since most recent 90-day or worse rating |
| mths_since_last_record | The number of months since the last public record. |
| mths_since_rcnt_il | Months since most recent installment accounts opened |
| mths_since_recent_bc | Months since most recent bankcard account opened. |
| mths_since_recent_bc_dlq | Months since most recent bankcard delinquency |
| mths_since_recent_inq | Months since most recent inquiry. |

| | |
|---|---|
| mths_since_recent_revol_delinq | Months since most recent revolving delinquency. |
| next_pymnt_d | Next scheduled payment date |
| num_accts_ever_120_pd | Number of accounts ever 120 or more days past due |
| num_actv_bc_tl | Number of currently active bankcard accounts |
| num_actv_rev_tl | Number of currently active revolving trades |
| num_bc_sats | Number of satisfactory bankcard accounts |
| num_bc_tl | Number of bankcard accounts |
| num_il_tl | Number of installment accounts |
| num_op_rev_tl | Number of open revolving accounts |
| num_rev_accts | Number of revolving accounts |
| num_rev_tl_bal_gt_0 | Number of revolving trades with balance >0 |
| num_sats | Number of satisfactory accounts |
| num_tl_120dpd_2m | Number of accounts currently 120 days past due (updated in past 2 months) |
| num_tl_30dpd | Number of accounts currently 30 days past due (updated in past 2 months) |
| num_tl_90g_dpd_24m | Number of accounts 90 or more days past due in last 24 months |
| num_tl_op_past_12m | Number of accounts opened in past 12 months |
| open_acc | The number of open credit lines in the borrower's credit file. |
| open_acc_6m | Number of open trades in last 6 months |
| open_il_12m | Number of installment accounts opened in past 12 months |
| open_il_24m | Number of installment accounts opened in past 24 months |
| open_act_il | Number of currently active installment trades |
| open_rv_12m | Number of revolving trades opened in past 12 months |
| open_rv_24m | Number of revolving trades opened in past 24 months |
| out_prncp | Remaining outstanding principal for total amount funded |
| out_prncp_inv | Remaining outstanding principal for portion of total amount funded by investors |
| pct_tl_nvr_dlq | Percent of trades never delinquent |
| percent_bc_gt_75 | Percentage of all bankcard accounts > 75% of limit. |
| policy_code | publicly available policy_code=1<br>new products not publicly available policy_code=2 |
| pub_rec | Number of derogatory public records |
| pub_rec_bankruptcies | Number of public record bankruptcies |
| purpose | A category provided by the borrower for the loan request. |
| pymnt_plan | Indicates if a payment plan has been put in place for the loan |
| recoveries | post charge off gross recovery |
| revol_bal | Total credit revolving balance |
| revol_util | Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit. |
| sub_grade | LC assigned loan subgrade |
| tax_liens | Number of tax liens |

| | |
|---|---|
| term | The number of payments on the loan. Values are in months and can be either 36 or 60. |
| title | The loan title provided by the borrower |
| tot_coll_amt | Total collection amounts ever owed |
| tot_cur_bal | Total current balance of all accounts |
| tot_hi_cred_lim | Total high credit/credit limit |
| total_acc | The total number of credit lines currently in the borrower's credit file |
| total_bal_ex_mort | Total credit balance excluding mortgage |
| total_bal_il | Total current balance of all installment accounts |
| total_bc_limit | Total bankcard high credit/credit limit |
| total_cu_tl | Number of finance trades |
| total_il_high_credit_limit | Total installment high credit/credit limit |
| total_pymnt | Payments received to date for total amount funded |
| total_pymnt_inv | Payments received to date for portion of total amount funded by investors |
| total_rec_int | Interest received to date |
| total_rec_late_fee | Late fees received to date |
| total_rec_prncp | Principal received to date |
| total_rev_hi_lim | Total revolving high credit/credit limit |
| url | URL for the LC page with listing data. |
| verification_status | Indicates if income was verified by LC, not verified, or if the income source was verified |
| verified_status_joint | Indicates if the co-borrowers' joint income was verified by LC, not verified, or if the income source was verified |
| zip_code | The first 3 numbers of the zip code provided by the borrower in the loan application. |
| revol_bal_joint | Sum of revolving credit balance of the co-borrowers, net of duplicate balances |
| sec_app_fico_range_low | FICO range (high) for the secondary applicant |
| sec_app_fico_range_high | FICO range (low) for the secondary applicant |
| sec_app_earliest_cr_line | Earliest credit line at time of application for the secondary applicant |
| sec_app_inq_last_6mths | Credit inquiries in the last 6 months at time of application for the secondary applicant |
| sec_app_mort_acc | Number of mortgage accounts at time of application for the secondary applicant |
| sec_app_open_acc | Number of open trades at time of application for the secondary applicant |
| sec_app_revol_util | Ratio of total current balance to high credit/credit limit for all revolving accounts |
| sec_app_open_act_il | Number of currently active installment trades at time of application for the secondary applicant |
| sec_app_num_rev_accts | Number of revolving accounts at time of application for the secondary applicant |
| sec_app_chargeoff_within_12_mths | Number of charge-offs within last 12 months at time of application for the secondary applicant |
| sec_app_collections_12_mths_ex_med | Number of collections within last 12 months excluding medical collections at time of application for the secondary applicant |

| | |
|---|---|
| sec_app_mths_since_last_major_derog | Months since most recent 90-day or worse rating at time of application for the secondary applicant |
| hardship_flag | Flags whether or not the borrower is on a hardship plan |
| hardship_type | Describes the hardship plan offering |
| hardship_reason | Describes the reason the hardship plan was offered |
| hardship_status | Describes if the hardship plan is active, pending, canceled, completed, or broken |
| deferral_term | Amount of months that the borrower is expected to pay less than the contractual monthly payment amount due to a hardship plan |
| hardship_amount | The interest payment that the borrower has committed to make each month while they are on a hardship plan |
| hardship_start_date | The start date of the hardship plan period |
| hardship_end_date | The end date of the hardship plan period |
| payment_plan_start_date | The day the first hardship plan payment is due. For example, if a borrower has a hardship plan period of 3 months, the start date is the start of the three-month period in which the borrower is allowed to make interest-only payments. |
| hardship_length | The number of months the borrower will make smaller payments than normally obligated due to a hardship plan |
| hardship_dpd | Account days past due as of the hardship plan start date |
| hardship_loan_status | Loan Status as of the hardship plan start date |
| orig_projected_additional_accrued_interest | The original projected additional interest amount that will accrue for the given hardship payment plan as of the Hardship Start Date. This field will be null if the borrower has broken their hardship payment plan. |
| hardship_payoff_balance_amount | The payoff balance amount as of the hardship plan start date |
| hardship_last_payment_amount | The last payment amount as of the hardship plan start date |
| disbursement_method | The method by which the borrower receives their loan. Possible values are: CASH, DIRECT_PAY |
| debt_settlement_flag | Flags whether or not the borrower, who has charged-off, is working with a debt-settlement company. |
| debt_settlement_flag_date | The most recent date that the Debt_Settlement_Flag has been set |
| settlement_status | The status of the borrower’s settlement plan. Possible values are: COMPLETE, ACTIVE, BROKEN, CANCELLED, DENIED, DRAFT |
| settlement_date | The date that the borrower agrees to the settlement plan |
| settlement_amount | The loan amount that the borrower has agreed to settle for |
| settlement_percentage | The settlement amount as a percentage of the payoff balance amount on the loan |

# Appendix C.

## Information About Features Dismissed

Features designated by “drop” selection decision were filtered out, while the blank cell in the respective column signifies that feature were not dismissed at that specific stag of the process.

| Features | Selection decision | Reason of dismissal |
|---|---|---|
| acc_now_delinq | drop | near zero variance |
| acc_open_past_24mths | | |
| addr_state | drop | deemed non-informative |
| all_util | | |
| annual_inc | | |
| annual_inc_joint | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| application_type | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| avg_cur_bal | | |
| bc_open_to_buy | drop | missing values |
| bc_util | drop | missing values |
| chargeoff_within_12_mths | drop | near zero variance |
| collection_recovery_fee | drop | future information |
| collections_12_mths_ex_med | drop | future information |
| cr_hist_mths* | | |
| debt_settlement_flag | drop | future information |
| debt_settlement_flag_date | drop | future information |
| deferral_term | drop | future information |
| delinq_2yrs | | |
| delinq_amnt | drop | near zero variance |
| desc | drop | description feature was not filled for the years 2017 and onwards |
| dti | | |
| dti_joint | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| earliest_cr_line | drop | transformed |
| emp_length | | |
| emp_title | drop | deemed non-informative |
| fico_mean* | | |
| fico_range_high | drop | transformed |
| fico_range_low | drop | transformed |
| funded_amnt | drop | deemed non-informative |
| funded_amnt_inv | drop | deemed non-informative |
| grade | | |
| hardship_amount | drop | future information |
| hardship_dpd | drop | future information |

| | | |
|---|---|---|
| hardship_end_date | drop | future information |
| hardship_flag | drop | future information |
| hardship_last_payment_amount | drop | future information |
| hardship_length | drop | future information |
| hardship_loan_status | drop | future information |
| hardship_payoff_balance_amount | drop | future information |
| hardship_reason | drop | future information |
| hardship_start_date | drop | future information |
| hardship_status | drop | future information |
| hardship_type | drop | future information |
| home_ownership | | |
| id | drop | deemed non-informative |
| il_util | drop | missing values |
| initial_list_status | drop | deemed non-informative |
| inq_fi | | |
| inq_last_12m | | |
| inq_last_6mths | | |
| installment | | |
| int_rate | | |
| issue_d | drop | transformed |
| last_credit_pull_d | drop | deemed non-informative |
| last_fico_range_high | drop | future information |
| last_fico_range_low | drop | future information |
| last_pymnt_amnt | drop | future information |
| last_pymnt_d | drop | future information |
| loan_amnt | | |
| loan_status | | |
| max_bal_bc | drop | near zero variance |
| member_id | drop | deemed non-informative |
| mo_sin_old_il_acct | drop | |
| mo_sin_old_rev_tl_op | | |
| mo_sin_rcnt_rev_tl_op | | |
| mo_sin_rcnt_tl | | |
| mort_acc | | |
| mths_since_last_delinq | drop | missing values |
| mths_since_last_major_derog | drop | missing values |
| mths_since_last_record | drop | missing values |
| mths_since_rcnt_il | drop | missing values |
| mths_since_recent_bc | drop | missing values |
| mths_since_recent_bc_dlq | drop | missing values |
| mths_since_recent_inq | drop | missing values |
| mths_since_recent_revol_delinq | drop | missing values |
| next_pymnt_d | drop | future information |
| num_accts_ever_120_pd | | |
| num_actv_bc_tl | | |
| num_actv_rev_tl | | |
| num_bc_sats | | |
| num_bc_tl | | |
| num_il_tl | | |
| num_op_rev_tl | | |

| | | |
|---|---|---|
| num_rev_accts | | |
| num_rev_tl_bal_gt_0 | | |
| num_sats | | |
| num_tl_120dpd_2m | drop | missing values |
| num_tl_30dpd | drop | near zero variance |
| num_tl_90g_dpd_24m | drop | near zero variance |
| num_tl_op_past_12m | | |
| open_acc | | |
| open_acc_6m | | |
| open_act_il | | |
| open_il_12m | | |
| open_il_24m | | |
| open_rv_12m | | |
| open_rv_24m | | |
| orig_projected_additional_accrued_interest | drop | future information |
| out_prncp | drop | future information |
| out_prncp_inv | drop | future information |
| payment_plan_start_date | drop | future information |
| pct_tl_nvr_dlq | drop | near zero variance |
| percent_bc_gt_75 | drop | future information |
| policy_code | drop | deemed non-informative |
| pub_rec | | |
| pub_rec_bankruptcies | | |
| purpose | | |
| pymnt_plan | drop | future information |
| recoveries | drop | future information |
| revol_bal | | |
| revol_bal_joint | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| revol_util | | |
| sec_app_chargeoff_within_12_mths | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_collections_12_mths_ex_med | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_earliest_cr_line | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_fico_range_high | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_fico_range_low | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_inq_last_6mths | drop | joint applications were disregarded due to high rates of missing values in valuable features |

| | | |
|---|---|---|
| sec_app_mort_acc | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_mths_since_last_major_derog | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_num_rev_accts | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_open_acc | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_open_act_il | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| sec_app_revol_util | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| settlement_amount | drop | future information |
| settlement_date | drop | future information |
| settlement_percentage | drop | future information |
| settlement_status | drop | future information |
| settlement_term | drop | future information |
| sub_grade | drop | inconsistent levels' default rates; feature "grade" was selected instead |
| tax_liens | drop | deemed non-informative |
| term | | |
| title | drop | deemed non-informative |
| tot_coll_amt | drop | near zero variance |
| tot_cur_bal | | |
| tot_hi_cred_lim | | |
| total_acc | | |
| total_bal_ex_mort | | |
| total_bal_il | | |
| total_bc_limit | | |
| total_cu_tl | | |
| total_il_high_credit_limit | | |
| total_pymnt | drop | future information |
| total_pymnt_inv | drop | future information |
| total_rec_int | drop | future information |
| total_rec_late_fee | drop | future information |
| total_rec_prncp | drop | future information |
| total_rev_hi_lim | | |
| url | drop | deemed non-informative |
| verification_status | | |
| verification_status_joint | drop | joint applications were disregarded due to high rates of missing values in valuable features |
| zip_code | drop | deemed non-informative |

## Appendix D.

### Information Gain Scores and Resulting Feature Selection

Feature with score value below 0,003 were dismissed.

| **Features selected** | **Information gain importance score** | **Features dismissed** | **Information gain importance score** |
|---|---|---|---|
| int_rate | 0,05771 | mo_sin_rcnt_tl | 0,00260 |
| grade | 0,05323 | num_actv_bc_tl | 0,00230 |
| term | 0,02474 | open_acc_6m | 0,00225 |
| fico_mean* | 0,01597 | purpose | 0,00200 |
| installment | 0,01544 | annual_inc | 0,00177 |
| loan_amnt | 0,01031 | cr_hist_mths* | 0,00168 |
| verification_status | 0,00750 | inq_fi | 0,00145 |
| open_rv_24m | 0,00650 | revol_bal | 0,00106 |
| tot_hi_cred_lim | 0,00631 | total_bal_ex_mort | 0,00081 |
| avg_cur_bal | 0,00535 | pub_rec | 0,00078 |
| tot_cur_bal | 0,00535 | total_acc | 0,00068 |
| acc_open_past_24mths | 0,00527 | num_op_rev_tl | 0,00066 |
| mort_acc | 0,00521 | open_il_12m | 0,00064 |
| total_bc_limit | 0,00480 | pub_rec_bankruptcies | 0,00061 |
| revol_util | 0,00457 | num_bc_sats | 0,00060 |
| total_rev_hi_lim | 0,00434 | open_il_24m | 0,00056 |
| all_util | 0,00415 | total_il_high_credit_limit | 0,00046 |
| num_actv_rev_tl | 0,00400 | open_act_il | 0,00045 |
| inq_last_6mths | 0,00392 | delinq_2yrs | 0,00044 |
| home_ownership | 0,00381 | total_bal_il | 0,00030 |
| open_rv_12m | 0,00355 | emp_length | 0,00028 |
| mo_sin_rcnt_rev_tl_op | 0,00345 | num_il_tl | 0,00027 |
| num_rev_tl_bal_gt_0 | 0,00342 | open_acc | 0,00019 |
| inq_last_12m | 0,00341 | num_sats | 0,00018 |
| num_tl_op_past_12m | 0,00326 | num_accts_ever_120_pd | 0,00000 |
| dti | 0,00317 | num_bc_tl | 0,00000 |
| mo_sin_old_rev_tl_op | 0,00307 | num_rev_accts | 0,00000 |
| | | total_cu_tl | 0,00000 |