TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Andrii Khrystian 166805IVSM

SUBPRIME CONSUMER CREDIT SCORING

Master's thesis

Supervisor: PhD Sven Nõmm

MSc Karl Märka

Tallinn 2018

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Andrii Khrystian 166805IVSM

KÕRGE RISKITASEMEGA TARBIJA KREDIIDIHINDAMINE

Magistritöö

Juhendaja: PhD Sven Nõmm

MSc Karl Märka

Tallinn 2018

Author's declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Andrii Khrystian

07.05.2018

Abstract

The purpose of this work is to validate different theoretical approaches from the fields of machine learning and credit scoring theory, implement them in a real business environment, assess the feasibility of a machine learning approach in the described business environment and provide a practical workflow for an automated credit scoring system implementation.

The main body of work provides an overview of the specifics and challenges of credit scoring and an empirical evaluation of the built model using common quality of model metrics.

The results of implementation show quite a significant improvement in the quality of application assessment due to the implementation of machine learning methods, resulting in monetary savings.

This thesis is written in English and is 41 pages long, including 6 chapters, 11 figures and 5 tables.

Annotatsioon

Kõrge riskitasemega tarbija krediidihindamine

Selle töö eesmärk on valideerida erinevaid teoreetilisi lähenemisviise masinõppe ja krediidihindamise teoreetilistes valdkondades, rakendada neid reaalses ettevõtluskeskkonnas, hinnata masinõppelise meetodi teostatavust kirjeldatud ettevõtluskeskkonnas ja pakkuda välja praktiline töövoog krediidivõime automatiseeritud hindamise rakendamiseks.

Töö põhiosa annab ülevaate krediidihindamise eripäradest ja väljakutsetest ning empiirilise hinnangu loodud mudelile, kasutades levinud mudelikvaliteedi hindamismõõdustikku.

Mudeli rakendamise tulemused näitavad taotluste hindamise kvaliteedi märkimisväärset paranemist masinõppe meetodite kasutamise tagajärjel, tulemuseks rahaline kokkuhoid.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 41 leheküljel, 6 peatükki, 11 joonist, 5 tabelit.

List of abbreviations and terms

AUC	Area under the curve
NN	Neural Network
SVM	Support Vector Machine
LGD	Loss given default
ROC	Receiver operating characteristic

Table of Contents

List of figures

List of tables

1 Introduction

Subprime consumer lending is inevitably associated with credit risk – this may present itself in the form of fraud, loan defaults, loan write-offs, court expenses, low customer retention rates etc. The profitability of the companies active in this field is directly dependent on how effective they are in predicting and managing these risks. Additional complexity is added by the fuzzy definitions of 'risk' and 'profitability' which are dependent on the time scale on which these characteristics are observed and measured.

At the moment, credit lending decision process in Creditstar Group is done manually in some of the target markets. Which means that decision process is based on a subjective decision of loan officer, the accuracy of which depends on the experience of the person and his/her assessment of credit application. The main goal of this thesis is to build machine learning model/ensemble of models for credit risk scoring, which will allow performing an almost instantaneous evaluation of applications while ultimately being more accurate than manual assessment process. The built model will be operating in the Swedish market.

The purpose of this experimental work is to validate different theoretical approaches from the fields of machine learning and credit scoring theory, implement them in a real business environment, assess the feasibility of a machine learning approach in the described business environment and provide a practical workflow for an automated credit scoring system implementation.

The result will be evaluated in multiple ways. Default "quality of the model" metrics (accuracy, precision, recall, area under curve - AUC, etc.) will be used to evaluate model on test dataset (archived data of the previously granted loans) and on a live/production environment dataset (obtained by scoring the so-called "through the door population"). The ultimate criteria for evaluation of whether achieved result is successful or not will be the difference in monetary losses due to incorrect assessment of the application.

The expected outcome of the implementation of the mentioned model is a reduction of the losses due to incorrect prediction no less than 10%

2 Related work

Following chapter will review the related literature on topics of credit scoring, approaches to credit scoring, specifics of dataset and features, bias.

2.1 Credit scoring and approaches

Credit scoring in its modern understanding was first introduced in 1941 with work by Durand [1]. He used a data-driven method of discriminant analysis to differentiate between good and bad borrowers. Since then storing loan application history for credit risk analysis became an integrant part of a credit lending process.

The idea of credit evaluation is to compare the features or the characteristics of a loan application with historical data of previously granted/refused loans. If application's characteristics are similar to those that have been accepted, and have consequently defaulted, the application will usually be rejected. If the application's features are similar to those that have not defaulted, the application will usually be approved. There exist two methods for credit evaluation – judgmental evaluation and credit scoring (statistical scoring).

Historically, credit lending decisions were based on the judgments made by human experts, using common guiding principles and professional experience. This type of approach is called judgmental evaluation. In their works, Sullivan [2] and Bailey [3] argue that success of decision process in judgmental evaluation depends on the experience and the common sense of the credit analyst. As a result, judgmental techniques are associated with subjectivity, inconsistency and individual bias motivating decisions. In terms of advantages over statistical scoring processes, Chandler and Coffman [4] concluded:

"… it seems that, on the whole, the empirical evaluation process has no serious deficiencies not also shared by judgmental evaluation. It also appears that empirical evaluation of creditworthiness has certain advantages that do not exist with judgmental evaluation. On the other hand, judgmental evaluation may have an advantage in dealing with individual cases that are truly exceptions from past experience."

As was mentioned above, another approach to credit risk assessment is called statistical scoring. According to Rosenberg and Gleit [5], since the early 1990s majority of consumer lending decision in the US have been made using automated scoring systems, which implies that statistical methods have been preferred to judgmental techniques for almost three decades.

Traditionally, the most widely used techniques for building scorecards were discriminant analysis, linear regression, logistic regression and multivariate adaptive regression splines. However, the problem with using statistical techniques is that some assumptions, such as the multivariate normality for independent variables are frequently violated which makes them theoretically invalid for finite samples. [6]

With the emergence of information technologies and constantly increasing computational power, simpler statistical models got replaced by machine learning techniques: decision trees, artificial neural networks (NN), support vector machines (SVM), fuzzy logic, genetic programming, hybrid methods, ensemble methods, etc.

In contrast to the statistical models, the machine learning methods do not assume any specific prior knowledge, but automatically extract information from past observations. The drawback, for which certain machine learning approaches such as NN or SVM are often being criticized, is the lack of understanding about underlying principles for a made decision. Such methods are often being compared to a “black box” which given a specific input, will provide the prediction.

Numerous papers have confirmed superiority of machine learning techniques compared to traditional statistical approaches, although it’s unlikely there exists a single classifier achieving best results in the whole application domain. Taking this fact into consideration, classifier ensembles have emerged to exploit the different behavior of individual classifiers and reduce prediction errors. In his work Finlay [7] compared several different multi-model classifier architectures. Marqués, García and Sánchez [8] explored the behavior of base classifiers in the ensemble. In their later work, they analyzed the performance of two-level ensemble models [9]. Durga Devi and Manicka Chezian [10] performed a relative evaluation of ensemble methods for the credit risk

scoring purposes. Mentioned works demonstrated that classifier ensembles generally perform better than single classifier in most credit scoring problems.

However, an ensemble of classifiers is efficient only if these have a minimum of errors in common [11]. In other words, the individual classifiers have to make decisions as diverse as possible. For example, if multiple models fail on the same set of inputs, they effectively function as a single model, and there is not much effect in having an ensemble. The higher diversity in the set of error case, the higher the effectiveness of an ensemble.

2.2 Dataset and features

To build a scoring model, or "scorecard", historical data on the performance of previously made loans and borrowers characteristics are required. A high-quality model should give high percentage of high scores to "good" customers and high percentage of low scores to "bad" customer. To build such a model, high-quality variables should be selected. Here the problem of defining what "high" quality variable means in terms of credit risk scoring arises. Ang, Chua and Bowling [12] investigated the profiles of late-paying consumer loan borrowers using variables such as gross amount of loan, age, gender, marital status, number of dependents, years lived at residence, monthly net income, monthly net income of spouse, own or rent residence, other monthly income, total monthly payments on all debts, type of bank accounts, number of credit references listed, years on job, total family monthly income per month, debt to income ratio, total number of payments on the loan, and annual percentage interest on the loan. Koh, Tan and Goh [13] used age, annual income, gender, marital status, number of children, number of other credit cards held and whether the applicant has an outstanding mortgage loan to construct a credit scoring model to predict credit risk of credit card applicants as bad loss, bad profit, and good risk.

Nature of data is heterogeneous in terms of categorization and include data related to a financial situation of the applicant, personal data, employment, etc. Vojtek and Kocenda [14] provided a table of indicators that are typically important in retail credit scoring models. They classify the indicators into 4 categories: demographic, financial, employment and behavioral.

The process of variable selection varies from study to study based on the nature of the data, and on what cultural or economic variables may affect the quality of the model and be appropriate in a particular market. These variables can differ from one country to another.

A number of variables incorporated into the model, while obviously depending on the nature of data, varies from simple models based on 3 variables [15] to a model which made use of 85 variables by Dvir et al. [16]. However, a higher number of variables does not guarantee in any way better performance comparing to a model with fewer variables. Nevertheless, datasets with a high number of variables give more opportunity to engineer new variables and perform feature selection based on their statistical significance, where fewer variables datasets would limit such possibilities. However, from the customer viewpoint, it will result in really lengthy questionnaires, which might scare them off and make them go elsewhere, although this can be avoided if additional information is received from third-party information providers.

Another problem relevant to the credit scoring is that in general only those who are accepted for credit will be followed up to find out whether they do turn out to be a good or bad risk, so that the design sample will be a biased sample from the overall population of applicants. Attempts to tackle this, using what information there is on the rejected applicants (namely their values on the characteristics, but not their true classes) are called reject inference. This distortion of the distribution of applicants clearly has implications for the accuracy and general applicability of any new score-card that is constructed. [17]

Advantages of statistical credit scoring based on machine learning methods over judgmental techniques include, but are not limited to the following list:

- Statistical credit scoring requires less information to make a decision because credit scoring models have been optimized to include only those variables which are significantly correlated with repayment performance, whereas judgemental decisions have no statistical significance and thus no variable reduction.
- Credit scoring models attempt to correct the bias that would result from considering the repayment histories of only accepted applications and not all applications.

- Better performance in terms of cost, effort, accuracy, speed.
- A credit scoring model includes a large number of a customer's characteristics simultaneously, including their interactions, while a loan analyst's mind cannot arguably do this, for the task is too challenging and complex.

While not fully presenting all possible advantages of using credit scoring in credit lending process, it obviously explains why nowadays almost all financial institutions and companies make use of machine learning techniques.

This thesis work will be focused on implementation of machine learning models which will try to solve/negate challenges specific to this sphere of machine learning application, having the unique ability to use real-world data and ability to evaluate the performance of models on real customers.

3 Challenges and specifics

As was mentioned before, credit scoring tries to deal with very peculiar challenges specific to its sphere, which will be discussed in this chapter.

3.1 Dataset size and bias

Importance of dataset size can't be underestimated in credit scoring since it seems to be the most influential factor in building "successful" scoring model (followed by the quality and number of features, and the details of implementation and used algorithms). The problem of small dataset size is quite obvious if you consider that every entry of dataset, in reality, is a customer looking to take a loan at a subprime credit lending company. While large credit-granting organizations are obviously much less affected by this problem since they have records of tens of thousands or even hundreds of thousands of customers, smaller companies are much more vulnerable to this problem.

To date, best practice in sampling credit applicants has been established based largely on expert opinion, which generally recommends that small samples of 1500 instances each of both "goods" and "bads" are sufficient and that the heavily biased datasets observed should be balanced by undersampling the majority class [18].

The empirical evaluation carried out internally at Creditstar Group seems to differ in this opinion. According to Figure 1, the dependency of the area under the curve (AUC) on dataset size seems to be having a logarithmic relationship with dataset size. Significant growth of AUC stops at around 4000 entries.

The dataset used in this work has around 2500 entries in total. Split between new and repeat customers is 65% / 35%. Within those folds, respectively 47% / 53% and 69% / 31% are the split between "good" and "bad" customers, which puts us well under the "sufficient" mark mentioned earlier both in terms of dataset size and bias within the subset (for repeat customer subset). Suggested solution for bias neglection would leave us with the even smaller dataset. Augmentation of the dataset with entries from historical data of already operating models from different markets is also impossible due to the

nature of variables used in those models which is the initial reason for the creation of separate model specific to the market.

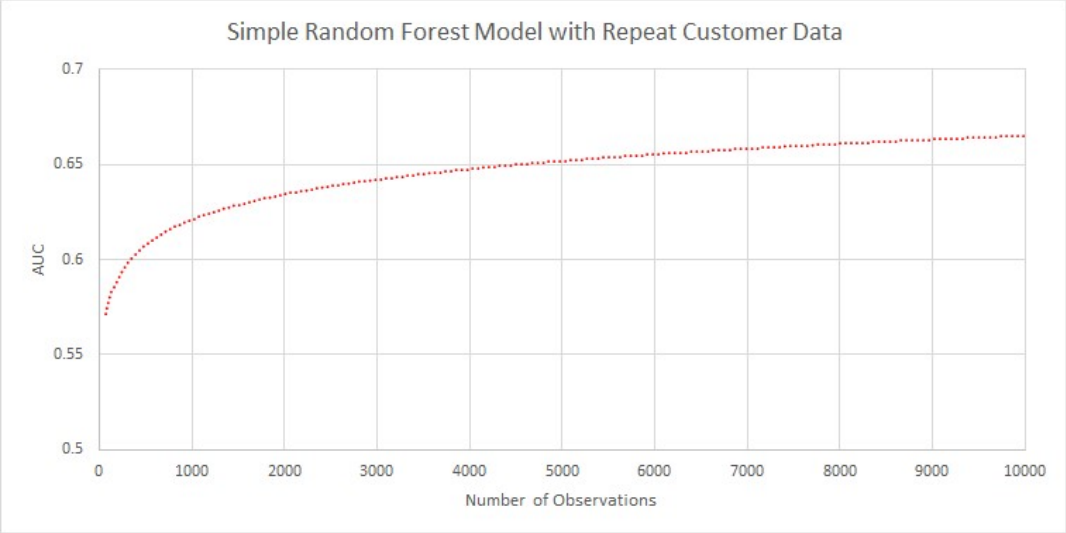

Figure 1. Dependency of AUC on dataset size. The curve represents the average AUC across 10 randomly sampled folds of given size from the total dataset from a larger target market.

3.2 Features and their importance

Features (the number and quality of these) is the second most influential factor in the model building process. As was mentioned, nature and number of features varies from dataset to dataset and highly depends on the availability of financial, personal and other data in different markets, which in turn is regulated by the informational visibility of a person and regulatory limitations existing in operating market.

Data about applicants mostly comes from two sources. The first one is applicants themselves since before getting a loan, most companies gather information through questionnaires and documents provided by a customer. Features gathered through this process forms a small but relatively significant part of all available features.

The remaining, bigger source of information about customers is third-party information providers. Those companies usually provide all kinds of information about applicant coming from different sources – public registries, banks, government instances, social networks, digital footprint etc. While bringing a lot of obvious advantages, it also has its

drawbacks. Those companies usually operate only within one country/market, meaning one set of variables is available in one country and another set of variables in another country, which are most of the times not intertransformable leading to very limited, if at all possible, reusability of model or parts of it. The topic of feature generation for credit scoring from third-party sources containing natural language and other difficult-to-process data is discussed more thoroughly in other concurrent research from Creditstar Group. [19]

In terms of feature importance, there seems to be a common theme, which is - there are 3 features which have relatively high importance while all the remaining features have low importance, however they shouldn't be ignored. The three rather obvious features are loan amount, loan duration and previous exposure of the customer ("repeat" vs "new" customer). Influence of those variables over the outcome is clear – bigger loan sum and longer loan durations lead to higher chances of going to loan collection. While being a repeat customer - taking multiple loans and paying them back usually minimizes the risk of fraud which in turn lowers the chances of consequent loans to default.

Remaining features, while not having as big of an impact, add up. It is noted that even an insignificant increase in prediction accuracy can result in quite significant savings for the company. (Details on costs of misclassification will be discussed in the following chapters).

Usually, repeat and new customers are scored separately. The reason behind it is that repeat customer features play a big role in the decision process. As time passes the repeat customers would comprise an increasingly large proportion of the dataset and having a much lower (53% vs 31%) a priori payment default probability would bias the model against new customers, limiting the company's growth ambitions. As an exception, repeat and new customers can be scored together when sizes of separated datasets are really small, and increase in accuracy of prediction due to bigger dataset size outweighs the decrease due to bias towards repeat customers.

3.3 Model structure

The following section describes the structure of the model in the order resembling the data flow (which can be seen in Figure 2): starting from data input to data transformation/preprocessing to data cleaning/imputation, model training/scoring, and output.

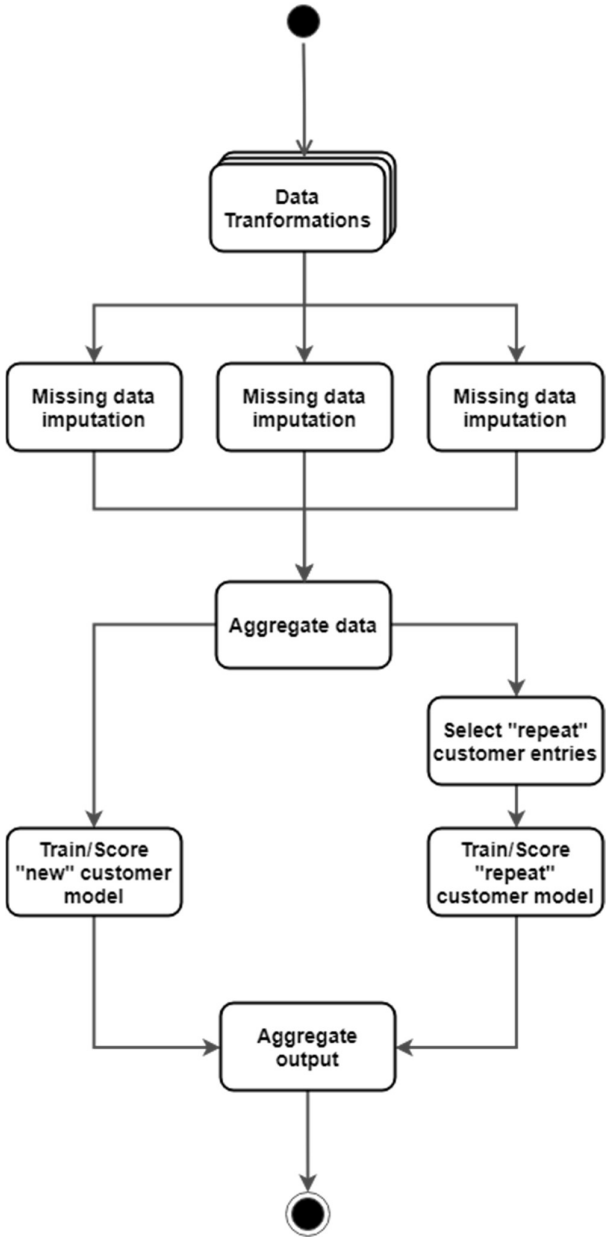

Figure 2 Execution flow of prediction web service.

3.3.1 Data input. Preprocessing and transformation

The built model is served as cloud-based web-service communication with which is performed through REST call – service-consumer does a POST request to a predictor-service in the Microsoft Azure cloud and receives a response with the defined output. Entry to be scored is passed as the body of the request in the form of JSON consisting of model variables and responses from third-party services, which will be transformed into actual variables.

Data transformation step consist of numerous Python scripts, the aim of which mostly falls into two categories: extraction/transformation of data into usable format (for example transform response of third parties which comes in different formats e.g. XML or JSON arrays of objects, type casting between appropriate types, etc.) and data "expansion". In this context, expansion means construction of new features based on the value of the ones that have been passed using some simple algorithm (simple mathematical operation between existing features, a transformation of categorical variables into indicator variables, etc.). Importance of these operations shouldn't be underestimated. It allows to significantly reduces the complexity of client-side implementation, which results in lower cost of implementation from a business perspective while decreasing the complexity of maintenance and required computational resources.

3.3.2 Missing data imputation

Next step is missing data imputation. Quite a few entries in the test data don't contain all of the required information – most commonly data provided by third parties are missing. To impute missing entries multiple techniques are utilized. Firstly, the dataset is split by features into multiple uneven chunks. Chunks were created based on the expert opinion and empirical methods. Used methods vary from really simple ones – replacing with specific substitution values, median, mode; to quite sophisticated statistical methods – MICE (Multivariate Imputation using Chained Equations), where each variable with missing data is modeled conditionally using the other variables in the data before filling in the missing values. All imputation methods introduce some error or bias, but multiple

imputation better simulates the process generating the data and the probability distribution of the data.

The resulting dataset doesn't contain missing data and can be passed further to perform remaining data manipulation before training model.

3.3.3 Model training

The first step of this phase for new customer scoring model consists of removing variables which don't have any importance for the model, specifically, zero variance features. An obvious example of such feature would be the number of loans previously taken from the company. Since customers are new, values would always be the same – 0. Repeat customer scoring model doesn't have such features, so this step is skipped.

Next step is dataset partitioning for cross-validation. Dataset was partitioned into 5 folds. The division into a higher number of folds would result in really small folds, which would nullify the aimed effect. A small number of folds would result in bigger size folds, and quite significantly reduce the size of training dataset, which is really bad since the dataset is small in the first place.

Models are trained against partitioned dataset using exhaustive model hyperparameter grid, the best model is then evaluated against test dataset. The criteria for selecting the best model is AUC. The reason behind selecting AUC as main characteristic instead of Gini coefficient (which is a very popular metric in economical calculation) is that receiver operating characteristic (ROC) allows to check predictive performance at any specific cutoff point, while Gini coefficient is integral over all cutoff points thus providing single value which can be used to compared different models. The former is exactly what is required in the current situation. Another advantage of AUC, which can be considered subjective, is that its interpretation much more intuitive.

3.3.4 Output generation

This step is specific to the production environment and responsible for output generated by web service as a response to the initial request. The response contains application score

S_f which is obtained by subtracting value predicted by the model S_o from 1 and multiplying the result by 100.

$$S_f = (1 - S_o) * 100$$

Resulting value lies in range [0; 100]. The reason behind representing score in such way has to do with user friendliness and specifics of interpretation of values. The original score provided by the model indicates the likeliness of loan going to loan collection; can take a value in the range [0; 1] where 1 means definite "bad" outcome from the point of view of the company and 0 means opposite – loan has a really low risk of going over the defined duration. This information while clearly describing the outcomes, feels counter-intuitive from the point of view of the loan officer. From their point of view, the low score should indicate "bad" application and high score – the "good" ones. Another small change which improves user experience is scaled up score values. The reason behind it lies in specifics of perception between really small rational numbers and same values scaled up to be in the range between 1 to 100.

Another output value is a hexadecimal color code associated with a score, which is intended to provide a color indication regarding "goodness" of application. The color code is used in the frontend tool which presents information regarding application to loan officers. The implications of this are quite simple and have to do with color coding information perception, e.g. red gamma colors will be intuitively associated with "bad" applications and green gamma with "good" ones.

"Suggestion" is another output value. It is a textual representation of the suggested actions regarding the application. It consists of a limited set of predefined values: accept, reject, auto-reject. Accept/reject decision is made based on whether the score is higher or lower than threshold (cutoff) value. "Auto-decline" is the boolean value indicating immediate rejection due to policy rules mandated by the regulator – in this case the Swedish Financial Inspection "*Finansinspektionen*". By auto-declining such applications, time spent by loan officers on analysis of application that effectively would be later considered as failed is greatly decreased.

The output also contains the version identifier and date of the current model.

Example of user interface loan officers interact with, containing application score and decision suggestion, can be seen in Figure 3.

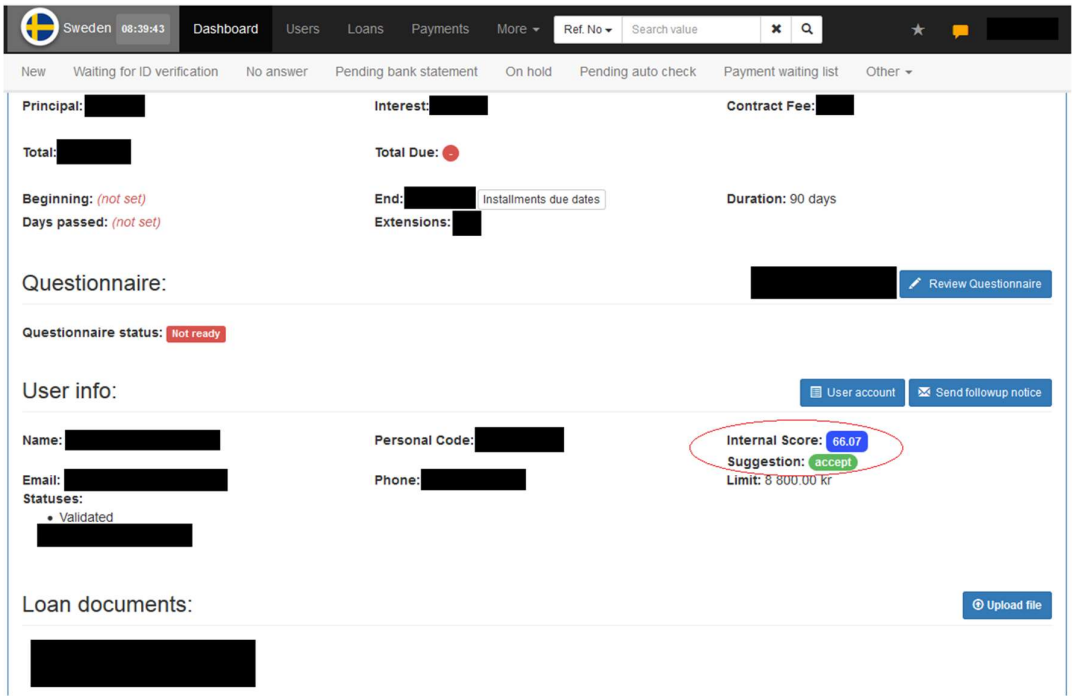

Figure 3. Loan officer user interface.

3.3.5 Alternative models

One of the significant advantages of web service and overall environment, in which machine learning solution was built and operates, is support of A/B testing. It allows for addition and removal of new models to the production environment, therefore enabling the ability to quickly add and evaluate alternative approaches in a matter of minutes.

3.4 Overview of models

Web-service predictor contains multiple models which utilize different approaches throughout every step of the earlier mentioned flow. This subchapter will provide a general overview of the new and repeat customer scoring models which can be considered final at the moment of writing the thesis.

Initial steps of data transformation, as was mentioned before, consist of numerous Python scripts the content of which itself is not essential, while the step itself is. Preprocessed data is cleaned and missing data imputation using common methods (replace with mean, median, constant, etc.) is performed, the result of which is passed to count-based featurization – one of the key components of model performance.

Learning with counts (another name for count-based featurization) is an efficient way to create a compact set of dataset features that are based on counts of the values. The concept behind it is simple, yet really powerful in certain cases. For example, let's assume that address region is one of the available features which has big number N of possible values. To utilize those values one option would be the introduction of N new features. However, that would result in longer model training, higher model complexity and more importantly, given the limited size of the dataset, higher chances of overfitting, which is obviously undesirable. Another option would be to use learning with counts, instead of introducing N new features, one can observe the counts and proportions of class labels for each address region. Count-based learning is attractive for many reasons: fewer features, which requires fewer parameters. Fewer parameters make for faster learning, faster prediction, smaller predictors, and less potential to overfit.

The dataset with new transformed features used to train models. As for algorithm used, “new customers” utilizes AdaBoosted decision trees, while “repeat customers” model uses a neural network. The decision behind utilizing these specific algorithms was made based on the empirical evaluation of different algorithms on the same dataset utilizing exhaustive hyperparameter grid. List of compared algorithms (excluding already mentioned) consists of decision forest, decision jungle, logistic regression, support vector machine. AdaBoosted decision trees consistently outperform other algorithms, followed by decision jungle and decision forest for “new customers” model. As for “repeat customers” models, neural network and other tree-based algorithms have performed on a comparable level, however, the neural network seems to generalize slightly better. Comparison of different algorithms on new and repeat customers training datasets can be seen in Table 1 and Table 2.

Table 1. Comparison of different algorithms on new customers training dataset (5-fold cross-validation).

Algorithm	**AUC**	**Accuracy**	**Precision**	**Recall**	**F1 score**
AdaBoosted Decision Trees	0.732	0.679	0.624	0.599	0.611
Neural Network	0.721	0.672	0.623	0.564	0.592
Decision Forest	0.722	0.669	0.625	0.535	0.577
Decision Jungle	0.726	0.671	0.626	0.545	0.583
Logistic Regression	0.721	0.667	0.621	0.540	0.578

Table 2. Comparison of different algorithms on repeat customers training dataset (5-fold cross-validation).

Algorithm	**AUC**	**Accuracy**	**Precision**	**Recall**	**F1 score**
AdaBoosted Decision Trees	0.684	0.654	0.426	0.254	0.318
Neural Network	0.668	0.691	0.537	0.216	0.308
Decision Forest	0.694	0.693	0.569	0.140	0.224
Decision Jungle	0.680	0.681	0.496	0.284	0.361
Logistic Regression	0.687	0.693	0.530	0.297	0.380

3.5 Model output calibration

One of the important parts of the model building process that often gets overlooked is probability calibration. A well-calibrated model is the one which produces probabilistic forecasts that correspond with observed probabilities. For example, let's consider one hundred loans in a score band where the probability to default is predicted to be ten percent. If the model is well-calibrated, the actual number of eventually defaulting loans in this band should be close to ten. [20]

Probability calibration is important for multiple reasons: regulatory requirements (e.g. Basel Accord), essential step of model evaluation in financial terms (through calculating expected gains/losses), etc.

Calibration assumes that the relationship between the raw score, which a classification model produces, and the true probability distribution is monotonic. Therefore, calibration consists of estimating a monotonic function to map raw scores to (calibrated) probability distributions.

One of the greatest advantages of employing probability calibration is that it matches the biased outputs of predictive models to real probabilities. Some algorithms such as random forest are especially prone to produce uncalibrated probability estimates.

Original paper [20] proposes multiple algorithms which can be used for the purposes of probability calibration (rescaling algorithm, Platt scaling, isotonic regression, generalized additive models (GAMs), etc.). Implementation created during this work utilizes logistic regression. The decision behind algorithm selection is mostly based on the simplicity of implementation in the specific development environment, while not significantly sacrificing the quality of the calibration. As a potential improvement, GAMs should be given a closer look.

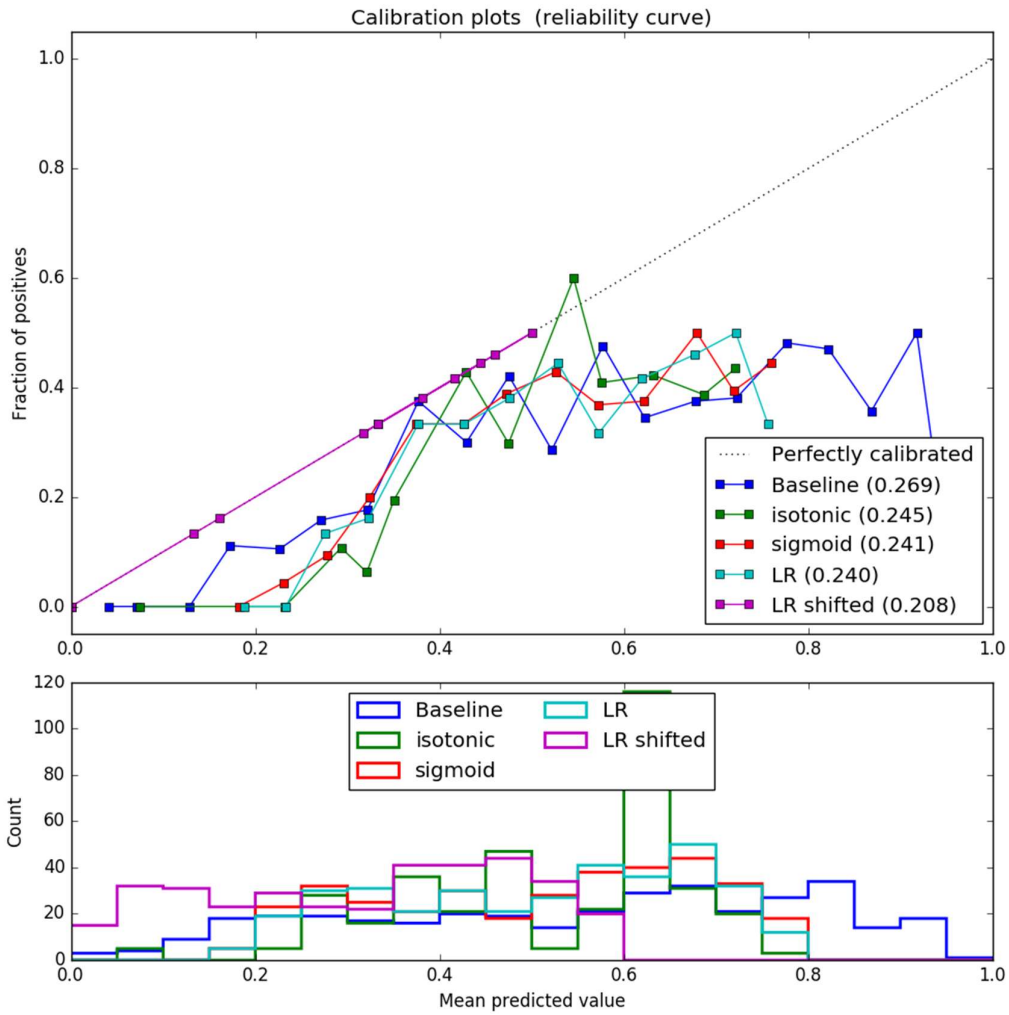

Figure 4. Calibration curves for different calibration methods used in new customer scoring model.

Figure 4 and Figure 5 show calibration curves for new and repeat customer scoring models respectively, utilizing different calibration algorithms. In the graph "Baseline" stands for the uncalibrated output of the classifier, "isotonic" and "sigmoid" for respective function fit on baseline output. "LR" stands for logistic regression model trained using scored probabilities produced by the original model, and true labels. "LR shifted" is the extension of the previous method which shifts all scores produced by logistic regression by the difference of mean predicted score value per score bin and fraction of real positives in the respective bin.

Quality of calibration is also measured by calculating Brier score. The calculated values can be seen in parentheses near respective calibration method names in the graph.

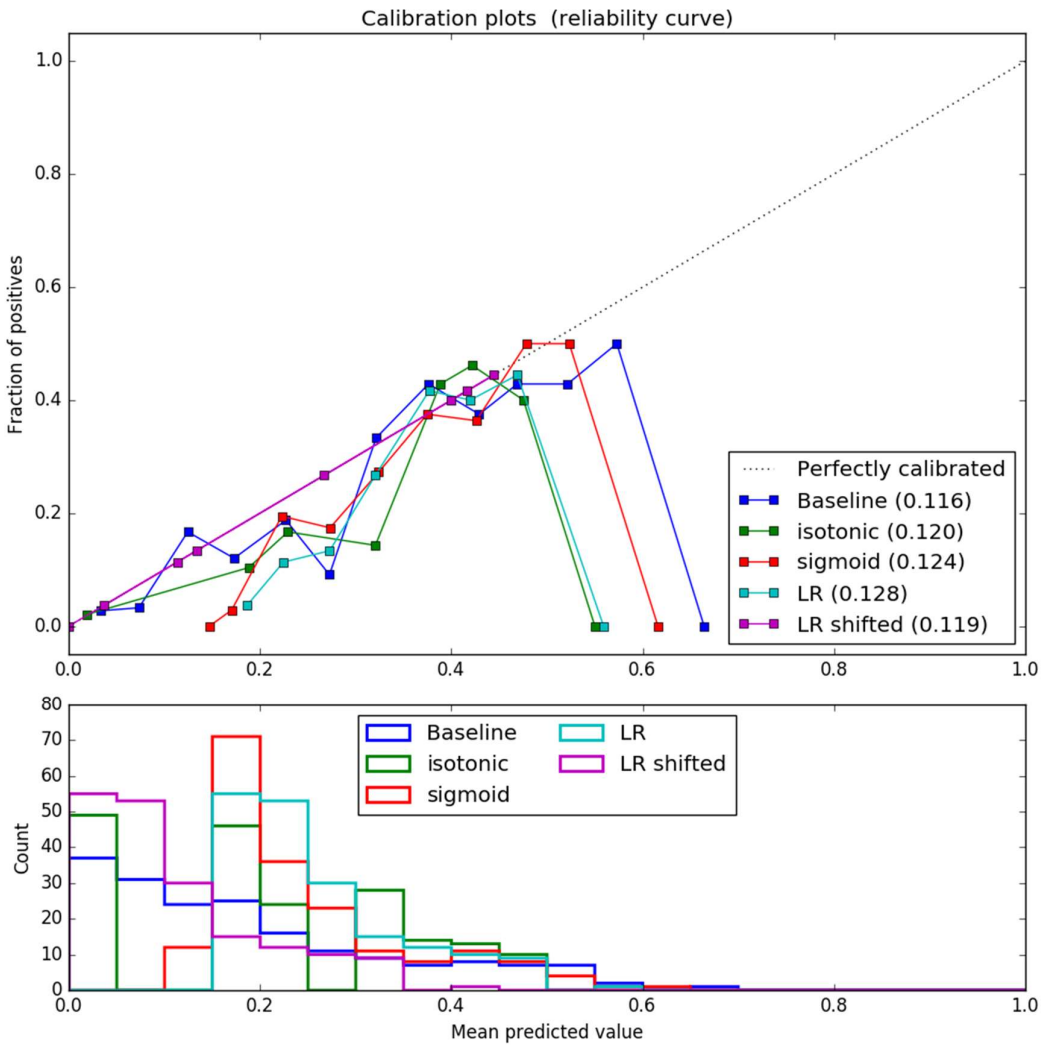

Figure 5. Calibration curves for different calibration methods used in repeat customer scoring model.

Quite a significant improvement in the quality of calibration for "new customers" model between baseline output of classifier (Brier score of 0.269) and shifted output of LR (Brier score of 0.208) can be spotted. As for the quality of calibration for "repeat customer" model, it seems to become worse (baseline – Brier score of 0.116, LR shifted – Brier score of 0.119). While being seemingly worse, calibrated output actually deals with the problem of the model being overly pessimistic in regard to repeat customers, which can be seen as a drop in a fraction of positives for mean predicted values past 0.5 in Figure 5. This drop indicates the fact that model unjustifiably assigns a higher probability of a customer being marked as problematic starting from probability > 0.6. Given that most of the revenue of any company comes from repeat customers, companies try to retain

such customers. In our case, models shouldn't reject repeat customers without being certain. Model calibrated using "LR shifted" method avoids this problem, therefore complying with imposed business requirements.

3.6 Optimal cutoff point definition

Another important step in the model building process is optimal cutoff point definition. The idea behind the process is to define optimal cutoff point for accept/reject application decision. The definition of "optimal" in this case is based on business policies and acceptable level of risk.

To define an optimal cutoff point, default and acceptance rate should be calculated. Default rate describes the ratio of misclassified entries (loan application accepted while in reality, it went to loan collection) to all accepted entries. Figure 6 and Figure 7 depict these relations for new and repeat customer scoring models respectively. The ratio seems to be having a linear relationship for "new customers" model while capping at the default rate of just above 0.4. For "repeat customers" model character of the curve seems to resemble logarithmic function with two small spikes at the beginning of the curve, capping just below the value of 0.3 for default rate.

Acceptance rate describes the ratio of accepted applications to all created applications. Figure 8 depicts the overall acceptance rate for new and repeat customers combined. The shape of the curve resembles logarithmic function, effectively capping at the cutoff point of 0.8.

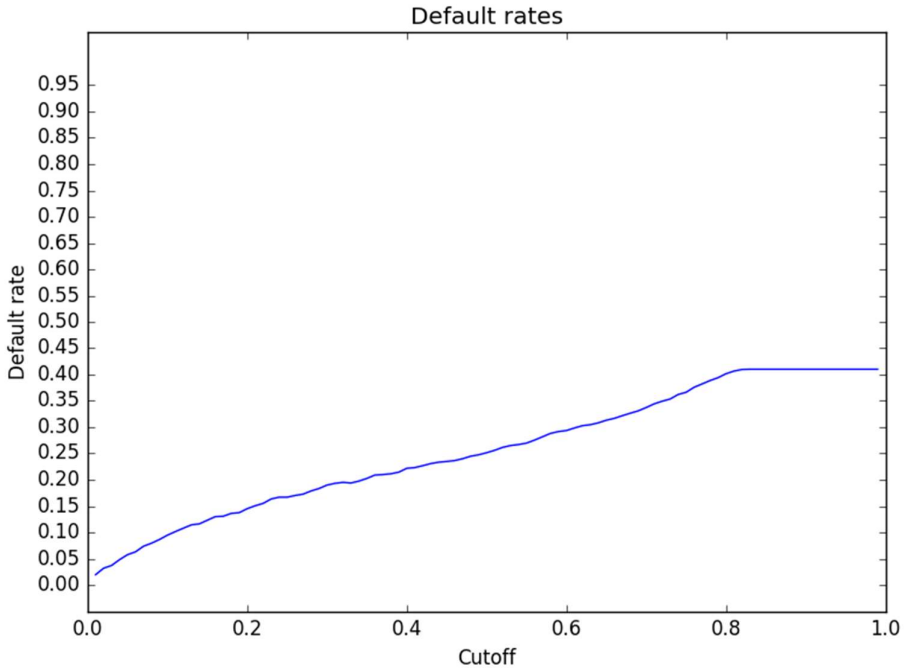

Figure 6. Dependency of default rate of "new customer" model on the cutoff point.

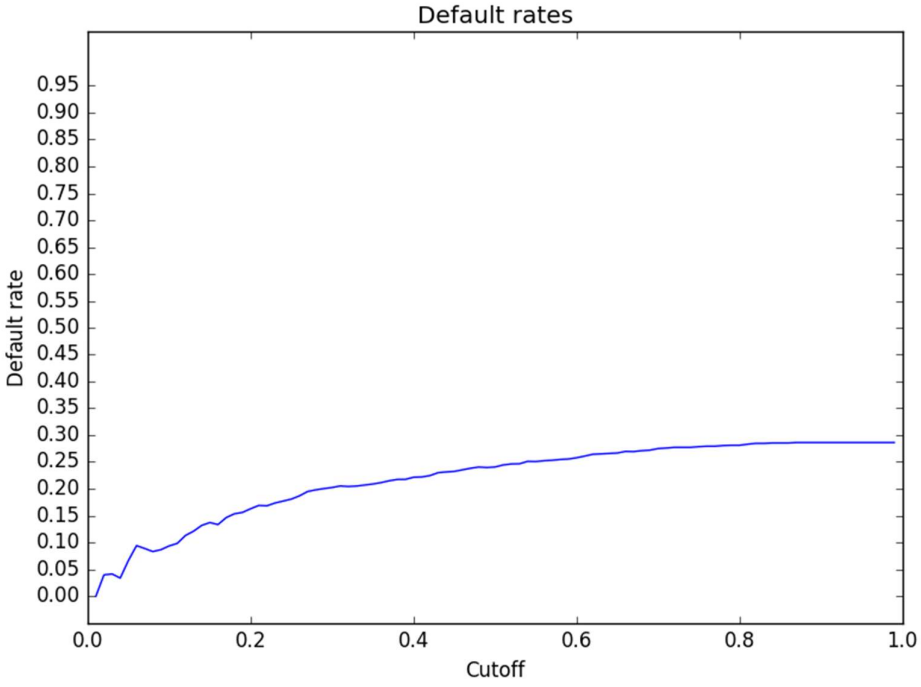

Figure 7. Dependency of default rate of "repeat customer" model on the cutoff point.

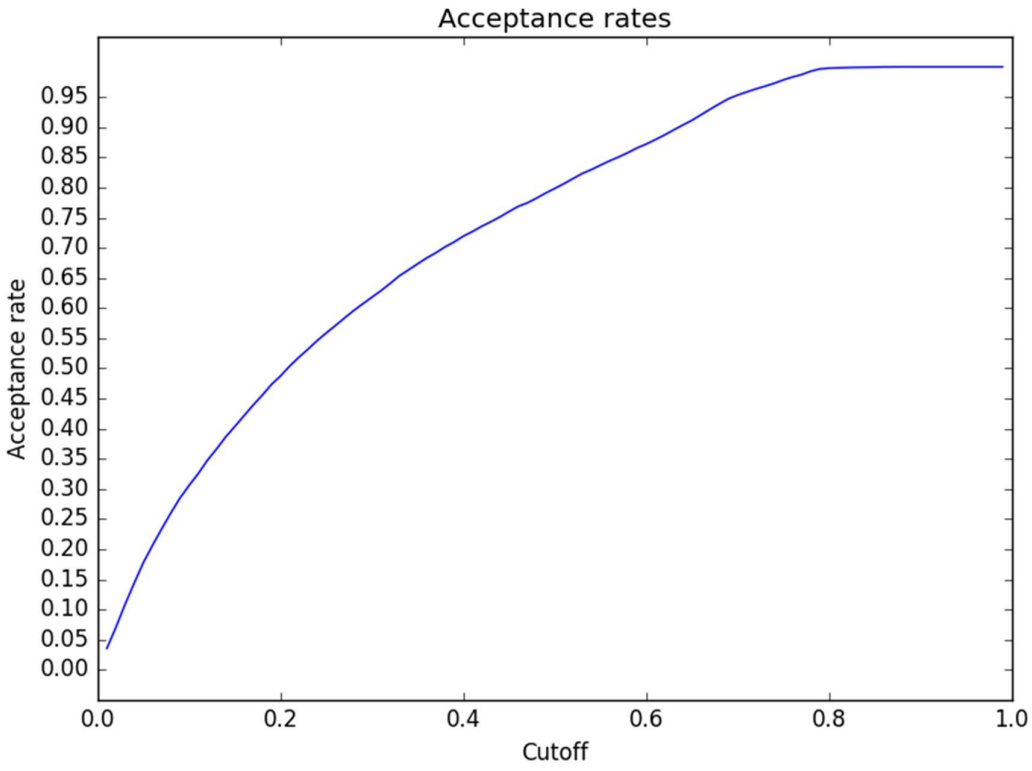

Figure 8. Dependency of overall acceptance rate on the cutoff point.

Given acceptance and default rates, one can determine the optimal cutoff point where limitations imposed by business regarding acceptable default rate and acceptance rate can be met. The cutoff optimization problem in credit scoring is more thoroughly discussed in other concurrent research conducted at Creditstar Group. [21]

4 Statistical and business model evaluation

The following section provides an overview of statistical metrics with respective values used to evaluate the performance of new and repeat customer models on training and test data and overall predictor service performance in the production environment. All metrics are calculated using default probability cutoff point of 0.5. Test and training datasets metrics are evaluated on calibrated model output. Production environment metrics are evaluated on uncalibrated output.

4.1 Training and test data

Figure 9 presents ROC curves of "new customer" model on training and test data. Table 3 contains a list of metrics and their respective values evaluated on training and test data.

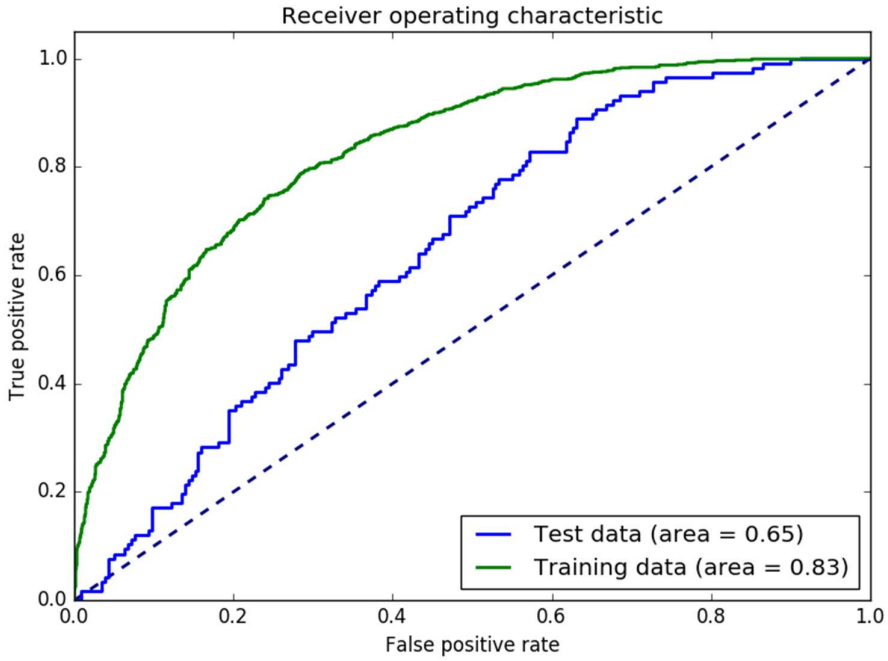

Figure 9. ROC curve for training and test data of "new customers" model.

From Figure 9, the quite significant difference in values of AUC between the train and test dataset can be spotted. There are two reasons behind this. One of them has to do with specifics of entries contained in training and test datasets. Training datasets consist of both new and repeat customers entries, while test dataset consists exclusively of unseen

new customers entries, which in turn has to do with earlier mentioned performance boost due to increased dataset size.

The second reason is related to possible overfitting.

Table 3. Statistical metrics of “new customers” model.

Metric	Training dataset	Test dataset
Accuracy	0.7484	0.5859
Precision	0.75	0.65
Recall	0.75	0.59
F1 score	0.74	0.60
AUC	0.83	0.65

Figure 10 and Table 4 contain the same set of metrics re-evaluated for “repeat customers” model.

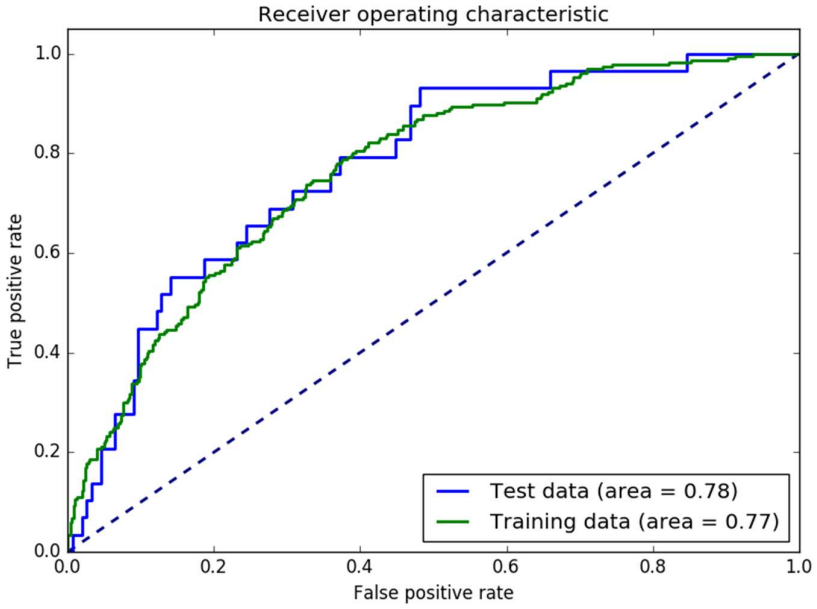

Figure 10. ROC curve for training and test data of “repeat customers” model.

From Figure 10 one can notice that “repeat customers” model seems to generalize really well since the difference between test and training dataset is minimal. Repeat customer scoring model also performs better in general terms.

Table 4. Statistical metrics of “repeat customers” model.

Metric	Training dataset	Test dataset
Accuracy	0.7183	0.8270
Precision	0.70	0.83
Recall	0.72	0.83
F1 score	0.67	0.83
AUC	0.77	0.78

4.2 Production data

Table 5 and Figure 11 contain earlier presented set of metrics re-evaluated on production data consisting of two datasets – 10 and 30 days overdue loans scored by the production model. The reason for evaluating the production model using 10 and 30 days overdue loans is the time it takes to observe the outcome of the loan. The 65+ days overdue outcome becomes observable only after the duration of the loan + 65 days has passed. With payment schedules of 30 days, this amounts to 95 days since the issue date. There wasn’t a sufficient number of observations available at the time of this writing.

Table 5. Statistical metrics of overall prediction quality in the production environment.

Metric	Overdue, 10 days	Overdue, 30 days
Accuracy	0.5971	0.6349
Precision	0.59	0.64
Recall	0.60	0.63

Metric	Overdue, 10 days	Overdue, 30 days
F1 score	0.59	0.64
AUC	0.65	0.65

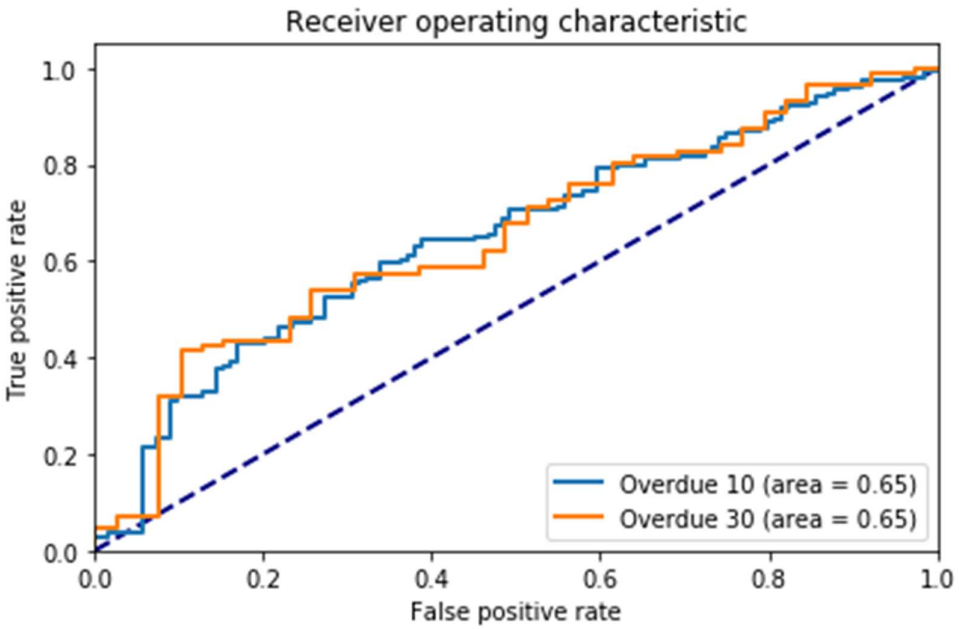

Figure 11. ROC curve of overall performance in the production environment.

The unavoidable problem related to the evaluation of model performance in production is bias, and this evaluation is obviously no exception. Since only selected few out of all population get to the point where they can actually be evaluated, evaluation dataset is obviously biased. This problem is also known as a selection bias and it's the single biggest challenge of credit scoring and similar forms of statistical analysis. Because of it, a true evaluation of performance on live population is almost impossible. A true evaluation would require issuing loans for all incoming applications which is unreasonably risky from a business perspective.

To slightly reduce the impact of the bias on the evaluation process, built model accepts loan which otherwise would be considered as "bad" with random decreasing probability. E.g. application that would've got rejected, gets accepted with a random small probability which provides the ability to check whether the model has correctly marked the

application as truly "bad" or it was wrong, and the application should've been accepted in the first place.

4.3 Business model evaluation

As a metric for evaluating model from the business perspective, monetary savings due to correct loan default prediction is evaluated. All other monetary savings due to automation of evaluation, etc. are not taken into consideration but from industry practice can be assumed to almost equal the profit increase directly related to prediction accuracy in some markets.

Monetary savings due to model integration calculated based on the difference of default rate before and after implementation, given the same acceptance rate. Savings amount is calculated on a monthly basis.

$$Savings = (DR_{Manual} - DR_{Model}) * Monthly\ Volume * Cost\ of\ loan\ default$$

DR stands for a default rate of respective methods given the same acceptance rate. *Monthly volume* – the monetary cost of all given out loans. *Cost of loan default* – coefficient indicating the relative cost of the loan in relation to the loan amount. For the Swedish market, the loss given default (LGD) is around 75% of the loan amount. The average monthly loan volume in the Swedish market is 800 000 SEK.

$$Savings = (0.2788 - 0.0637) * 800{,}000 * 0.75 = 129{,}060\ SEK\ \approx 12{,}151.64\ EUR$$

The value of the default rate with a model in place can be considered slightly optimistic, due to the problem of selection bias mentioned in the previous subchapter. Regardless, an obvious improvement from a business perspective can be seen. A significant difference in default rate allows increasing acceptance rate, therefore increasing potential revenue by allowing acceptance of new customers. Estimations do not take into account natural growth of customer base due to other different reasons, which would allow saving even more on monetary losses.

5 Discussion

The goal of the work is to validate existing body of knowledge related to credit scoring and machine learning by implementing the theoretical concepts in real life business environment. Empirical evaluation of the built models confirms with the existing conclusion made throughout related works.

Credit scoring has its own peculiarities and challenges. Most of those challenges can't be avoided in the real business environment due to unacceptable monetary losses related to the possible solutions. The biggest challenge is the selection bias. It prevents true evaluation both in a production environment and during development phase leading to overly optimistic/pessimistic results. As a result, a lot of uncertainty added to business policy planning.

Developed predictor fulfills business requirement imposed during initial planning, providing automated application scoring and simplifying work of loan officers.

The system can potentially be improved both in quality of prediction and usability by experimenting/implementing following suggestions:

- Automatic calculation of the maximal loan sum applicant can apply for according to the regulations
- Implementation of an ensemble of models for both new and repeat customer scoring models
- Dynamic adjustment of cutoff score
- Addition of some features regarding financial information of applicants.

6 Summary

The main goal of this thesis was to build machine learning model/web-service for credit risk scoring, which would fulfill following requirements:

- instantaneous evaluation of applications
- ultimately, be more accurate than manual assessment process
- monetary losses due to incorrect assessment of application decrease no less than by 10 percent.

Built model fulfills mentioned requirements, and reduces default rate, based on optimistic evaluation, by 21.511 percent which roughly equals to 12,151.64 EUR in monetary savings per month, excluding savings related to natural growth of customer base or savings due to automation, which from industry practice can be assumed to almost equal the profit increase directly related to prediction accuracy.

References

[1] D. Durand, Risk elements in consumer instalment financing, New York: National Bureau of Economics, 1941.

[2] A. C. Sullivan, Consumer Finance, in Altman, E.I. Financial Handbook, New York: John Wiley & Sons, 1981.

[3] M. Bailey, Credit Quality: Underwriting, Scoring, Fraud Prevention and Collections., Bristol: White Box Publishing, 2004.

[4] G. G. Chandler and J. Y. Coffman, "A comparative analysis of empirical versus judgemental credit evaluation," *The Journal of Retail Banking,* vol. 1, no. 2, pp. 15-26, 1979.

[5] E. Rosenberg and A. Gleit, "Quantitative Methods in Credit Management: A Survey," *Operations Research,* vol. 42, no. 4, pp. 589-613, 1994.

[6] Z. Huang, H. Chen, C.-J. Hsu, W.-H. Chen and S. Wu, "Credit rating analysis with support vector machines and neural networks: a market comparative study," *Decision Support Systems,* vol. 37, no. 4, pp. 543-558, 2004.

[7] S. Finlay, "Multiple classifier architectures and their application to credit risk assessment," *European Journal of Operational Research,* vol. 210, no. 2, pp. 368-378, 2011.

[8] A. I. Marqués, C. García and J. S. Sánchez, "Exploring the behaviour of base classifiers in credit scoring ensembles," *Expert Systems with Applications,* vol. 39, no. 11, pp. 10244-10250, 2012.

[9] A. I. Marqués, V. García and J. S. Sánchez, "Two-level classifier ensembles for credit risk assessment," *Expert Systems with Applications,* vol. 39, no. 12, pp. 10916-10922, 2012.

[10] C. R. Durga Devi and R. Manicka Chezian, "A relative evaluation of the performance of ensemble learning in credit scoring," in *IEEE International Conference on Advances in Computer Applications*, Coimbatore, 2016.

[11] S. Bian and W. Wang, "On diversity and accuracy of homogeneous and heterogeneous ensembles," *International Journal of Hybrid Intelligent Systems,* vol. 4, no. 2, pp. 103-128, 2007.

[12] J. S. Ang, J. H. Chua and C. H. Bowling, “The Profiles of Late-Paying Consumer Loan Borrowers: An Exploratory Study: Note,” *Journal of Money, Credit and Banking,* vol. 11, no. 2, pp. 222-226, 1979.

[13] H. C. Koh, W. C. Tan and C. P. Goh, “Credit Scoring Using Data Mining Techniques,” *Singapore Management Review,* vol. 26, no. 2, pp. 25-47, 2004.

[14] M. Vojtek and E. Kocenda, “Credit-Scoring Methods,” *Czech Journal of Economics and Finance,* vol. 56, no. 3-4, pp. 152-167, 2006.

[15] D. Fletcher and E. Goss, “Forecasting with neural networks: an application using bankruptcy data,” *Infromation & Management,* vol. 24, no. 3, pp. 159-167, 1993.

[16] D. Dvir, A. Ben-David, A. Sadeh and A. Shenhar, “Critical managerial factors affecting defense projects success: A comparison between neural network and regression analysis,” *ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE,* vol. 19, no. 5, pp. 535-543, 2006.

[17] D. J. Hand and W. E. Henley, “Statistical Classification Methods in Consumer Credit Scoring: A Review,” *Journal of the Royal Statistical Society. Series A (Statistics in Society),* vol. 160, no. 3, pp. 523-541, 1997.

[18] S. F. Cron and S. Finlay, “Instance sampling in credit scoring: An empirical study of sample size and balancing,” *International Journal of Forecasting,* vol. 28, no. 1, pp. 224-238, 2012.

[19] L. Salia, *Improving consumer credit scoring model accuracy with advanced feature engineering techniques,* Tallinn: School of Engineering, Tallinn University of Technology, 2018.

[20] A. Bequé, K. Coussement, R. Gayler and S. Lessmann, “Approaches for credit scorecard calibration: An empirical analysis,” *Knowledge-Based Systems,* vol. 134, pp. 213-227, 2017.

[21] M. Herasymovych, *Optimizing Acceptance Thresholds in Credit Scoring using Reinforcement Learning,* Tartu: School of Economics and Business Administration, University of Tartu, 2018.

[22] H. A. Abdou and J. Pointon, “CREDIT SCORING, STATISTICAL TECHNIQUES AND EVALUATION CRITERIA: A REVIEW OF THE LITERATURE,” *Intelligent Systems in Accounting, Finance and Management,* vol. 18, no. 2-3, p. 59–88, 2011.

[23] F. Louzada, A. Ara and G. B. Fernandes, “Classification methods applied to credit scoring: Systematic review and overall comparison,” *Surveys in Operations Research and Management Science,* vol. 21, no. 2, pp. 117-134, 2016.

[24] J. Abellán and J. G. Castellano, “A comparative study on base classifiers in ensemble methods for credit scoring,” *Expert Systems with Applications,* vol. 73, pp. 1-10, 2017.

[25] F. Pedregosa, et al., “Scikit-learn: Machine Learning in Python,” *The Journal of Machine Learning Research,* vol. 12, pp. 2825-2830, 2011.

[26] J. Crook, D. B. Edelman and L. Thomas, “Recent developments in consumer credit risk assessment,” *European Journal of Operation Research,* vol. 187, pp. 1447-1465, 2007.