**Vysoká škola ekonomická v Praze**

Fakulta financí a účetnictví

katedra bankovnictví a pojišťovnictví

Finanční inženýrství

**DIPLOMOVá práce**

2024 Stevan Vujčić

**Vysoká škola ekonomická v Praze**

Fakulta financí a účetnictví

katedra bankovnictví a pojišťovnictví

Finanční inženýrství

**Komparativní analýza modelů pro predikci pravděpodobnosti defaultu**

Autor diplomové práce: Stevan Vujčić

Vedoucí diplomové práce: Milan Fičura

Rok obhajoby: 2024

Čestné prohlášení

Prohlašuji, že jsem diplomovou práci na téma „Komparativní analýza modelů pro predikci pravděpodobnosti defaultu“ vypracoval samostatně a veškerou použitou literaturu a další prameny jsem řádně označil a uvedl v přiloženém seznamu.

V Praze dne doplnit datum

.................................................................................

Stevan Vujčić (výše podepsat)

Abstrakt

(stručný obsah práce s uvedením hlavních výsledků v maximálním rozsahu 15 řádků)

Klíčová slova

Abstract

Abstract in English.

Key words

Key words in English

Table of Contents

[Introduction 6](#_Toc154915406)

[1 Theoretical Framework 7](#_Toc154915407)

[1.1 Institutional Environment 7](#_Toc154915408)

[1.1.1 EBA Note on the Use of Machine Learning for IRB models 7](#_Toc154915409)

[1.2 Literature Review 8](#_Toc154915410)

[2 Overview of Modeling Approaches and Model Testing 9](#_Toc154915411)

[2.1 Estimation Algorithms 9](#_Toc154915412)

[2.1.1 Logistic regression 9](#_Toc154915413)

[2.1.2 Neural Network 11](#_Toc154915414)

[2.1.3 Support Vector Machines 11](#_Toc154915415)

[2.1.4 Decision Tree 11](#_Toc154915416)

[2.1.5 Random Forest 11](#_Toc154915417)

[2.1.6 Gradient Boosting 11](#_Toc154915418)

[2.1.7 KNN 11](#_Toc154915419)

[2.2 Preprocessing Techniques 11](#_Toc154915420)

[2.2.1 Oversampling 11](#_Toc154915421)

[2.2.2 Data Transformations 12](#_Toc154915422)

[2.2.3 Forward feature selection 12](#_Toc154915423)

[2.2.4 Performance Testing 12](#_Toc154915424)

[3 Application on Czech Mortgages Portfolio 13](#_Toc154915425)

[3.1 Modeling 13](#_Toc154915426)

[3.1.1 Preprocessing: data cleaning 14](#_Toc154915427)

[3.1.2 EDA 14](#_Toc154915428)

[3.1.3 Preprocessing: transformations and feature selection 15](#_Toc154915429)

[3.1.4 Logistic regression 19](#_Toc154915430)

[3.1.5 Neural Network 20](#_Toc154915431)

[3.2 Comparison of Results 23](#_Toc154915432)

[Final Remarks 24](#_Toc154915433)

[References 25](#_Toc154915434)

[Graphics and Tables 26](#_Toc154915435)

[Appendices 27](#_Toc154915436)

# Introduction

In this thesis, a comparison of 2 probability of default (PD) estimation methods is outlined. The traditional logistic regression approach is contrasted against Artificial Neural Networks (ANN).

Write about: credit risk environment, conceptual advantages vs disadvantages, datasets, results, more transparent modeling procedure than in articles.

Abbreviations list:

EBA – European Banking Authority

ML – Machine Learning

IRB – Internal Ratings-based

GDPR – General Data Protection Regulation

AI – Artificial Intelligence

PD – Probability of Default

LGD – Loss Given Default

EaD – Exposure at Default

LTV – Loan-to-Value

# Theoretical Framework

In the following subsections, a systematical approach to finding the relevant literature published so far is outlined. Then, relevant findings are summarized.

## Institutional Environment

Text.

### EBA Note on the Use of Machine Learning for IRB models

In August 2023, EBA published EBA/REP/2923/28 – “Follow-up report from the consultation on the discussion paper on machine learning for IRB models”. The document addresses the question of how ML techniques fit into the IRB framework. It also takes a broader look from the standpoint of GDPR regulation and the AI act. As from the context of the document, ML techniques here refer to non-traditional modeling approaches (e.g. artificial neural networks). The main findings/conclusions of the report are summarized in the next paragraphs.

EBA reports that the use of ML models is still at a developmental stage in the industry. When used, ML techniques are employed by financial institutions for PD modeling. Conversely, they are not used or are used to a lesser extent when other parameters such as EaD or LGD are being modeled. Furthermore, it is reported that financial institutions use ML in the pre-model estimation phase of development. For example, inputting missing data or feature selection. Also ML techniques were reported to be used as benchmarks for traditional models. Financial institutions do not necessarily refrain from the use of traditional approaches. However, once ML is employed to set the performance bar, this feedback is useful input into making a conclusion about whether the traditional model itself should be better.

A concern is raised about the interpretability of these estimation techniques as compared to a traditional model such as the logistic regression. It often cannot be with straightforwardness and ease concluded how the individual model features contributed to the model from the standpoint of economic interpretation.

It is claimed that financial institutions are struggling to deal with the tendency of ML models to overfit the data. Therefore, financial institutions should, according to EBA, invest special attention to in-sample and out-of-sample comparisons when using ML.

Financial institutions are reported to not always have mature data infrastructures and procedures that handle the complexity that comes with the decision to use these techniques. This especially plays a role in the case of older data which was not collected not stored in such detail.

The EBA comments that financial institutions should understand the fact that the determination of capital requirements is being harmonized across the continent. To interpret this point, this means that the even larger diversity of modeling solutions that ML would inherently bring at least conceptually goes against this effort.

Finally, the EBA concludes that the financial institutions globally agree with the recommendations of the underlying discussion paper. Consequently, the EBA sets that the discussion paper should be used as the basis for ML modeling in risk applications.

## Literature Review

The scope of the relevant literature collection is defined as any literature that compares different estimation techniques in the field of probability of default modeling. This is aimed at finding relevant (1) research papers published in academic journals, (2) publications from financial institutions as well as (3) other insightful sources that are presented in other formats.

Table 1. Literature search results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Search engine | Criteria | # of results | # of referenced results | Query link |
| Web of Science | 1. keywords: machine learning credit scoring; 2. filter: “4.61.1820 Credit Scoring” 3. filter: from November 2013 to November 2023 | 315 |  | https://www.webofscience.com/wos/woscc/summary/adb5bbe9-768c-4a8d-967c-332fa6b5e22b-b17545fd/sort-group-background-citingcount/1 |
| Scopus | probability of default alternative methods | 132 |  |  |

# Overview of Modeling Approaches and Model Testing

In this section, the basics of estimation techniques for the probability of default classification problem and the performance tests are commented.

## Estimation Algorithms

In this subsection, the estimation techniques which are used in the application section are reviewed.

### Logistic regression

The logistic regression is used as the industry standard due to its sound performance and good interpretability of its results. The logistic function is used to confine the possible model output values between 0 and 1:

|  |  |
| --- | --- |
|  | # |

where z is subject to model specification and in this application decomposes into:

|  |  |
| --- | --- |
|  | # |

where ***X*** is the matrix of explanatory variables and is the vector of coefficients. The sigmoid curve is depicted below.

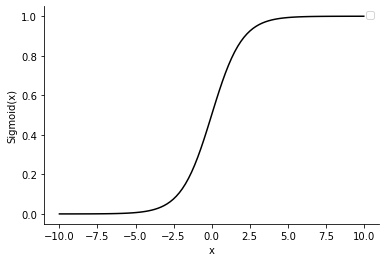


Figure 1: sigmoid function, source: author

To solve the equation defined above, the maximum likelihood estimator is used. In its log-transformed form:

|  |  |
| --- | --- |
|  | # |

where *L* is the likelihood function. The log-likelihood function is then maximized with respect to *θ*.

Another possibility is to use the probit instead of the logit model. There, the normal distribution is used as the link function. Generally these two approaches yield very similar results. The logistic distribution however has heavier tails when compared to the normal distribution (Witzany YYYY). The fitted value obtained from this model is interpreted as a probability. Other formulations of the above are the good-bad odds and the log good-bad odds:

|  |  |
| --- | --- |
|  | # |

and

|  |  |
| --- | --- |
|  | # |

The former gives the ratio between good and bad probabilities and the changes in the explanatory variable values are interpreted as multiplicative impacts. The latter is the additive impact formulation on the log good-bad odds.

Interpreting the impact of the individual coefficients is not as straightforward as is the case with, for example, OLS. This is due to the use of the sigmoid function, which is nonlinear. Therefore, the ***X*** matrix values which get mapped to the tails of the distribution get relatively similar values – close to 0 or 1, depending on the side. On the other hand, the ***X*** matrix values that are in the middle can mutually appear as more similar than in the previous case, but their assigned probability (sigmoid) curve can be significantly different. In econometrics, average marginal and partial marginal effects are used to arrive at interpretations. Nevertheless, interpreting the model coefficients to such degree can be regarded out of scope of this work since the primary goal is to focus on how the model fits.

### Neural Network

Text.

### Support Vector Machines

Text.

### Decision Tree

Text.

### Random Forest

Text.

### Gradient Boosting

Text.

### KNN

Text.

## Preprocessing Techniques

### Oversampling

In cases when a classification problem is subject to a large disbalance between its classes, the estimated models might not be able to discriminate between classes. In credit risk datasets, the number of bad observations is always lower than the number of good observations. Consequently, a classifier that cannot handle unbalanced data would classify all observations into the larger class. Such a model is obviously useless.

Oversampling techniques are introduced to solve this issue. These techniques address the class imbalance problem by synthetically creating data. Popular are variations of the Synthetic Minority Over-sampling Technique (SMOTE) – BorderlineSMOTE, KMeansSMOTE, SVMSMOTE.

The basic SMOTE algorithm utilizes the idea of the KNN solution. A difference between, for example, 2 observations of the underrepresented sample is computed. Then, the difference is multiplied with a random number between 0 and 1. This exercise is repeated for a number of times that is sufficient to balance-out the classes in the dataset. More specifically, a new observation of 2 nearest neighbors is generated by the following formulae:

|  |  |
| --- | --- |
|  | # |

where attains values between 0 and 1, and , are actual observations from the minority class.

Another popularly used approach is ADASYN. It is conceptually the same as SMOTE. It practically extends the SMOTE algorithm by adding a random term to the generated observations. Therefore, the generated observation is no longer a strict linear combination of the neighbors.

|  |  |
| --- | --- |
|  | # |

where is the random term.

### Data Transformations

Binning, WoE.

### Forward feature selection

Text.

### Performance Testing

Confusion matrix, ROC & AUC, KS Distance, Somers D

# Application on Czech Mortgages Portfolio

The modeling dataset concerns a part of the Czech mortgages portfolio of a major bank in the country. The available dataset has observations for which at least 6 months elapsed since the moment of contract signing. Therefore, the approach to scoring will be the behavioral one. Given a large number of columns in the dataset – 325, an automated variable selection based on statistical metrics is performed. Only after shortlisting the explanatory variables, a more granular analysis of the features is performed. The rest of the section is organized as follows:

The dataset has 200,000 observations, out of which 726 facilities were marked as defaulted. This amounts to an overall default rate of 0.36%.

Table 2. Overview of the modeling dataset

|  |  |  |  |
| --- | --- | --- | --- |
| cohort date | # observations | # defaults | default rate (%) |
| 201101 | 15,182 | 103 | 0.68% |
| 201201 | 16,230 | 82 | 0.51% |
| 201301 | 18,936 | 104 | 0.55% |
| 201401 | 20,153 | 134 | 0.66% |
| 201501 | 21,923 | 73 | 0.33% |
| 201601 | 23,727 | 84 | 0.35% |
| 201701 | 25,930 | 70 | 0.27% |
| 201801 | 28,406 | 36 | 0.13% |
| 201901 | 29,513 | 40 | 0.14% |

Source: CSOB.

Without suggesting the possible drivers of the development of the portfolio as outlined in the table above, it can be observed that the default rate showed a downward trend in both absolute and relative terms. The analyzed mortgage portfolio doubled in size in less than 10 years, both in terms of the number of facilities as well as the outstanding amount which increased from 1,600bn to 3,700bn CZK over the period.

## Modeling

In this section, the preprocessing, exploratory data analysis and modeling steps are outlined.

### Preprocessing: data cleaning

The focus of the data preprocessing procedure is on getting the data into an analyzable shape from the technical point of view. The longlist of all available variables along with their descriptions is included in Appendix X. The treatment of missing values is performed depending on the characteristics of a variable. For example, a nan value in the case of a delinquency flag column implies that no delinquency took place. On the other hand, practically nothing can be reliably assumed about a missing outstanding amount. The rates of missing entries per column in the dataset as well as their treatment is also included in Appendix X. Globally, the following decisions are made:

1. Missing entries of variables of string type such as topographical names, academic titles etc. are not treated. Their categorical derivatives are however subject to decision-making. For example, the worst category can be assigned to missing cells.
2. The missing integer entries of variables that code the delinquency status are replaced with 0 values. This is done based on the implication that no delinquency happened.
3. Other integer variables are not treated for missing entries.
4. Count variables, such as number of months are not treated for missing entries.
5. Missing entries of flag variables are resolved by inputting 0 values.

A more refined and specific treatment of some variables that exhibit missing amounts might be taken into consideration upon their inclusion into a shortlist. The scope of explanatory variables that enter a shortlist depends on univariate analyses. Since each estimation techniques as explained in Section X are preceded with different univariate analyses, each feature selection (shortlist) might also look different. Aside from the missing values, the data quality is sufficiently good to proceed with further analyses.

### EDA

Text.

### Preprocessing: transformations and feature selection

In this subsection, the transformative and feature selection data preprocessing steps are outlined.

The data is split into training and testing samples in the 70:30 ratio. For replicability of the results, the seed is set to 130816. Two types of datasets are going to be tried in each model estimation:

1. A preprocessed dataset without oversampling, and
2. A preprocessed dataset with oversampling.

The intention of using these two versions of datasets is to capture how well individual estimation techniques perform in the case when the data is class-imbalanced and when it is not.

#### Oversampling

The chosen oversampling technique is SMOT. This solution is chosen over the more refined ADASYNC oversampling technique due to the fact that the available software solution for SMOT in Python is able to elegantly handle categorical variables. The Python library used in this application is imblearn, the module over\_sampling has the function SMOTENC (the “ENC” part of its name representing the ability to encode categorical variables). Five nearest neighbors are linearly combined in order to arrive at each new observation. The choice of 5 neighbors is rather arbitrary as it is the default setting in the SMOTENC implementation.

#### Binning and WoE

Now, the outlier treatment, binning and the WoE transformation is computed by one function. The Python library scorecardpy has a complete infrastructure for probability of default modeling in credit risk applications. In this instance, the function woebin is used. It performs binning and subsequently computes the WoE values. Also, outliers are taken out when bins are being defined. The relatively straightforward interquartile range method is used to address these. Then, the bins are applied back to the whole dataset (including the outliers) using the function woebin\_ply.

Since the longlist of variables is fairly large, it would not be economical to plot and inspect the binning and the resulting WoE transformation for all explanatory variables. Therefore, only several variables are shown and their economic interpretation is outlined. In the Czech mortgages dataset the retail behavioral score of clients is collected. This refers to the behavior of clients on their retail non-mortgage products that they have at the bank. The higher the score, the lower the default rate should be. The 2 plots below show that this simple economic expectation is fulfilled – the WoE values drop monotonically with lower score bins.

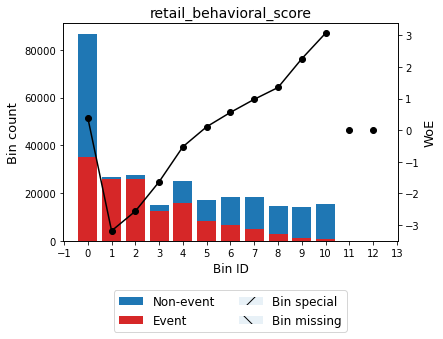


Figure 2: binning and WoE transformation for retail behavioral score, source: author

The WoE values across bins drop monotonically going from left to right in the case of the number of days in delinquency. This is also expected from sanity check standpoint. The longer the client struggles to meet their obligations, the higher the chance that the financial issues they encounter are substantial. This concretely is a 6M average of the count of days in delinquency. In the dataset, other averages are available for this as well as the simple count of days in delinquency as at the snapshot moment. From the plot, it can also be observed that most clients did not ever become delinquent.

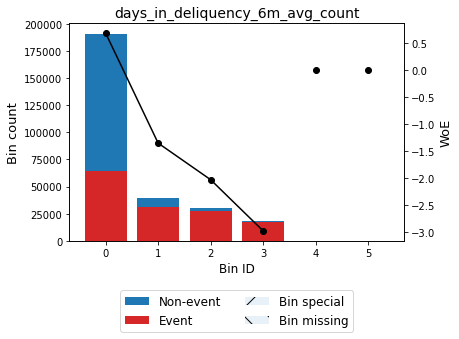


Figure 3: binning and WoE transformation for 6M-averaged count of days in deliquency, source: author

The low loan-to-value ratio (LTV) indicates clients that had substantial share of their own funds to participate in the purchase of the real estate asset, relative to its value. A sharp drop in the WoE follows to clients that had less cash available at the moment of purchase. Finally, the WoE-values increase that follows after that group could be explained only in interaction with other metrics. For example, these clients might have had higher retail scores, disposable collaterals, guarantors or other reasons for which they were granted with a high LTV in the first place. Ultimately, it appears that these clients did not struggle to manage the large loans as much as the group in the middle of the plot. It can be concluded that the LTV alone is not a metric that can explain the future performance of a client.

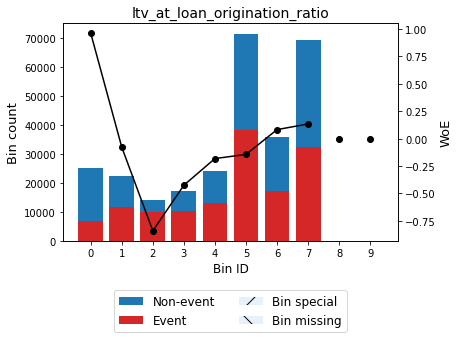


Figure 4: binning and WoE for loan-to-value at origination ratio, source: author

#### Univariate Analysis

Now, the univariate analysis can either be performed in the same way for all subsequent multivariate models. The second option is to perform them in such way that matches the method of the final model, if possible. For example the multivariate logistic regression to be preceded with a battery of univariate logistic regressions. In some cases this would not be possible, for example estimating a univariate neural network would be a questionable and time-consuming exercise. Finally, one can consider skipping the feature selection step based on the univariate models. In that case, some features would be excluded based on the multicollinearity assessment. Ultimately, the final set of features would be arrived at by limiting the maximum number of explanatory variables in the estimated model. The explanatory variables that would stay in the model are the ones that perform the best.

In this work, it is chosen to perform the univariate feature selection based on traditional credit risk practices. The aim of this feature selection procedure is not to arrive at the final set of explanatory variables. Rather, only features that clearly underperform are excluded at this instance. Two criteria are used and both have to be satisfied – the IV value has to be above 0.02 and the gini coefficient from a univariate logistic regression has to be above 0.1. The univariate feature exclusion procedures are ran separately on both the initial and SMOT-extended datasets.

#### Multicollinearity Removal

Finally, the multicollinearity assessment on the remaining variables is performed using the SelectNonCollinear function from the collinearity library. The maximum tolerated correlation between 2 explanatory variables is 0.5 (threshold set as per Witzany YYYY). When there is a group of variables with correlation above the specified threshold, the one that has the strongest ANOVA-F statistic with respect to the target variable is chosen.

### Logistic regression

In this subsection, the logistic regression model is applied. The number of remaining explanatory variables is rather high and they should be reduced to some 7-15 as per Witzany (yyyy). To fulfill this optimization task, the forward selection algorithm is used. The LogisticRegression function from the linear\_model module of the sklearn library is combined with the SequentialFeatureSelector function from the feature\_selection module of the mlxtend library. Finally, the explanatory variables used in the construction of the logistic regression model are listed in the table below.

Table 3. Variable selection using the forward selection algorithm

|  |  |
| --- | --- |
| variable\_name | IV |
| age |  |
| brki\_installment\_amt |  |
| collateral\_required\_amt |  |
| days\_in\_deliquency\_6m\_max\_to\_next\_installment\_ratio |  |
| debt\_summary\_2qs\_max\_amt |  |
| education\_categorical |  |
| interest\_paid\_to\_next\_installment\_6m\_max\_ratio |  |
| penalty\_interest\_paid\_mtd\_amt |  |
| product\_type |  |
| retail\_behavioral\_score |  |

Source: author

The AUC is similar and indicates good model performance on both the training and the test samples. The D-value of the Kolmogorov-Smirnov statistics are also similar across the train and test samples.

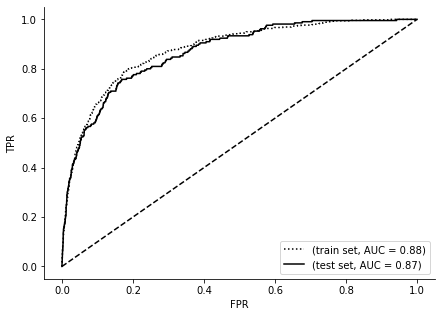


Figure 5: ROC cuve for train and test sets, logistic regression, source: author

Add confusion matrix.

### Neural Network

The sklearn library implemented in Python is the programming basis of this computation. There, a multilayer perceptron classifier solution is available. The most troublesome part of a neural network estimation is narrowing down to the best one given the vast options (number of hidden layers, activation function, solver algorithm, learning rate, regularization term). In order to tackle the search for the most appropriate neural network, multiple combinations are tried out and the best fitting neural network is selected. All combinations of the settings listed in the table below are considered. The same list of variables as in the logistic regression section is used to estimate this model.

Table 4: combinations for optimal neural network search

| parameter | combination |
| --- | --- |
| Hidden layer sizes *(the number of integers in the bracket implies the number of layers, n integers means n layers)* | [7] |
| [7, 7] |
| [10, 5] |
| [5, 5] |
| Activation function | Logistic |
| Tanh |
| Relu |
| Solver | Sgd |
| Adam |
| Alpha | 0.001 |
| Learning rate | Constant |
| Invscaling |
| Adaptive |

Source: author

The structure of the resulting neural network is summarized below.

Table 5: best performing neural network

|  |  |
| --- | --- |
| parameter | combination |
| Hidden layer sizes | [7] |
| Activation function | Logistic |
| Solver | Adam |
| Alpha | 0.001 |
| Learning rate | Constant |

Source: author

It’s performance on the train and test samples is mutually consistent.

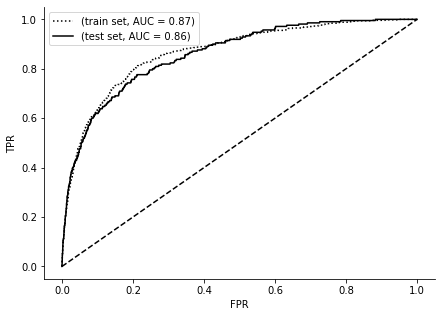


Figure 6: ROC curve for train and tests sets, artificial neural network, source: author

Next, a neural network using all explanatory variables is estimated. A larger number of neurons was introduced into the structure in order to accommodate 44 features. The optimal neural network in this case has a single hidden layer of 22 neurons, the activation function is sigmoid, it has a constant learning rate and it is solved by the adam algorithm. As can be seen from the performance assessment below, this neural network has a marginally better ROC curve for the train and test sets (reference figure). Therefore, it can be concluded that using the brute force of the large number of variables rather brings complications in the area of model management and implementation.

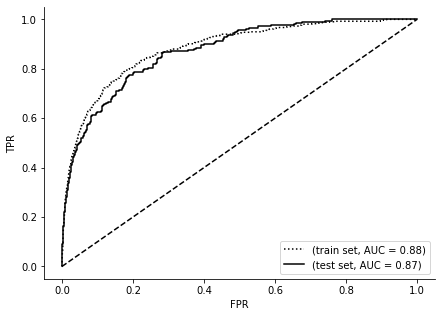


Figure 7: ROC curve for train and test sets, artificial neural network with all available variables, source: author

## Comparison of Results

# Final Remarks

# References

# Graphics and Tables

Graphics

**No table of figures entries found.**

Tables

**No table of figures entries found.**

# Appendices

Appendix A –

Appendix B –

Appendix A – Title

Appendix B – Title