European Management Studies
Vol. 21, No. 3, p. 46–70, e-ISSN: 2300-8792
https://doi.org/10.7172/2956-7602.101.3

Credit Risk Modeling Using Interpreted XGBoost

Marcin Hernes
Wroclaw University of Economics and Business, Poland
https://orcid.org/0000-0002-3832-8154

Jędrzej Adaszyński
Wroclaw University of Economics and Business, Poland
https://orcid.org/ 0009-0006-1731-1210

Piotr Tutak
Wroclaw University of Economics and Business, Poland
https://orcid.org/0000-0002-0976-4037

Submitted: 13.04.2023 | Accepted: 03.08.2023

Abstract

Purpose: The aim of the paper is to develop a credit risk assessment model usingb the XGBoost classifier supported by interpretation issues.
Design/methodology/approach: The risk modeling is based on Extreme Gradient Boosting (XGBoost) in the research. It is a method used for regression and classification problems. It is based on a sequence of decision trees using a gradient-based optimization method of the loss function to minimize the errors of weak estimators. We use also methods for performing local and global interpretability: ceteris paribus charts, SHAP and feature importance approach.
Findings: Based on the research results, it can be concluded that XGBoost achieved higher values of performance metrics than logistic regression, except sensitivity. It means that XGBoost indicated a smaller percentage of all bad client. Results of local interpretability enable a conclusion that in the case of the client in question, the credit decision is positively influenced by credit scores from external suppliers, while it is negatively influenced by minimal external scoring and short seniority. The number of years in the car and higher education are also positive. Such information helps to justify a negative credit decision. Results of global interpretability enable a conclusion that higher values of the traits associated with the z-scores are accompanied by negative Shapley values, which can be interpreted as a negative effect on the explanatory variable.
Research limitations/implications: XGBoost, A ceteris paribus plot, SHAP, and feature importance methods can be used to develop a credit risk assessment model including machine learning interpretability. The main limitation of research is to compare the results of XGBoost only to the logistic regression results. Future research should focus on comparing the results of XGBoost to other machine learning methods, including neural networks.

Correspondence address: Wroclaw University of Economics and Business, Komandorska 118-120, 53-345 Wrocław, Poland; e-mails: marcin.hernes@ue.wroc.pl; 180667@student.ue.wroc.pl; piotr.tutak@ue.wroc.pl.

Suggested Citation: Hernes, M., Adaszyński, J., & Tutak, P. (2023). Credit Risk Modeling Using Interpreted XGBoost. *European Management Studies*, *21*(3), 46–70. https://doi.org/10.7172/2956-7602.101.3.

Originality/value: One of the key processes in a bank is the credit decision process, which is the evaluation of a client's repayment risk. In the consumer finance sector, the processes are usually largely automated, and increasingly the latest machine learning methods based on neural networks and ensemble learning methods are being used for the purpose. Although machine learning models allow for achieving higher accuracy of credit risk assessment compared to traditional statistical methods, the main problem is the low interpretability of machine learning models. The models often perform as the "black box". However, the interpretation of the results of risk assessment models is very important due to the need to explain to the client the reasons for assessing their credit risk.

Keywords: credit risk, risk modeling, XGBoost, machine learning interpretability, explainable artificial intelligence.

JEL: C63, C88, D81

Modelowanie ryzyka kredytowego z wykorzystaniem interpretowalnego algorytmu XGBOOST

Streszczenie

Cel: celem niniejszych badań jest opracowanie modelu oceny ryzyka kredytowego z wykorzystaniem klasyfikatora XGBoost z uwzględnieniem interpretowalności tego modelu.
Metodologia: w niniejszych badaniach w celu modelowania ryzyka wykorzystano metodę Extreme Gradient Boosting (XGBoost). Jest to metoda stosowana do problemów regresji i klasyfikacji. Opiera się na sekwencji drzew decyzyjnych wykorzystujących gradientową metodę optymalizacji funkcji straty w celu minimalizacji błędów słabych estymatorów. Wykorzystano również metody umożliwiające dokonanie lokalnych i globalnych interpretacji: wykresy ceteris paribus, SHAP i badanie ważności cech.
Wyniki: na podstawie wyników badań można stwierdzić, że XGBoost osiągnął wyższe wartości metryk efektywności niż regresja logistyczna, z wyjątkiem wartości metryki czułości, Oznacza to, że XGBoost wskazał mniejszy odsetek wszystkich złych klientów. Wyniki interpretacji lokalnej pozwalają stwierdzić, że w przypadku klienta na decyzję kredytową pozytywnie wpływają oceny punktowe od zewnętrznych dostawców, liczba lat samochodu oraz wykształcenie wyższe, natomiast negatywnie wpływają niska zewnętrzna ocena scoringowa oraz krótki staż pracy. Taka informacja pozwala na uargumentowanie negatywnej decyzji kredytowej. Wyniki interpretacji globalnej pozwalają wnioskować, że wyższym wartościom cech związanych ze wskaźnikami towarzyszą ujemne wartości Shapleya, co można interpretować jako negatywny efekt wpływu na zmienną objaśniającą.
Ograniczenia/implikacje badawcze: metody XGBoost, A ceteris paribus plot, SHAP i feature importance mogą być wykorzystane do opracowania modelu oceny ryzyka kredytowego z uwzględnieniem interpretowalności uczenia maszynowego. Głównym ograniczeniem badań jest porównanie wyników XGBoost jedynie z wynikami regresji logistycznej. Przyszłe badania powinny skupić się na porównaniu wyników XGBoost z innymi metodami uczenia maszynowego, w tym z sieciami neuronowymi.
Oryginalność/wartość: jednym z kluczowych procesów realizowanych w bankach, jest proces podejmowania decyzji dotyczących udzielenia kredytów, czyli ocena ryzyka spłaty zobowiązania przez klienta. W sektorze finansów konsumenckich procesy te są zwykle w dużym stopniu zautomatyzowane, a coraz częściej wykorzystuje się w tym celu najnowsze metody uczenia maszynowego oparte na sieciach neuronowych i metodach uczenia zespołowego. Choć modele uczenia maszynowego pozwalają na osiągnięcie wyższej dokładności oceny ryzyka kredytowego w porównaniu z tradycyjnymi metodami statystycznymi, to głównym problemem jest niska interpretowalność modeli uczenia maszynowego. Modele te często występują jako „black box". Interpretacja wyników modeli oceny ryzyka jest jednak bardzo ważna ze względu na konieczność wyjaśnienia klientowi powodów oceny jego ryzyka kredytowego.

Słowa kluczowe: ryzyko kredytowe, modelowanie ryzyka, XGBoost, interpretowalność uczenia maszynowego, wyjaśnialna sztuczna inteligencja.

1. Introduction

Banks, due to the nature of their business, collect large volumes of data on client and their financial products. The data can be used for statistical modeling and the generation of machine learning algorithms that can help predict future events based on historical data, thereby improving decision-making processes.

The credit decision process, which is the evaluation of a client's ability to repay a debt is the key process in a bank, from an operational perspective. In the consumer finance sector, the processes are usually largely automated, and the latest machine learning methods based on neural networks and ensemble learning methods are being increasingly used for the purpose. The algorithms are often referred to as black boxes, meaning that the method of operation of such algorithms is complex and often unintuitive. In the context of credit risk models, a deeper understanding of the algorithm allows one to deepen business knowledge, prevent errors, but also respond to regulatory requirements.

Although machine learning models allow to achieve higher accuracy of credit risk assessment, as compared to traditional statistical methods (Addo et al., 2018), the main problem is the low interpretability of machine learning models. The models often perform as the "black box". However, the interpretation of the results of risk assessment models is very important due to the need to explain to the client the reasons for assessing their credit risk.

The aim of the paper is to develop a credit risk assessment model using the XGBoost classifier supported by interpretation issues. We use the XGBoost classifier (Li et al., 2021) because it allows risk modeling in relation both to a large and a small sample of data. Most other machine learning models (for example neural networks) require a large sample of data. Both local and global interpretability has been analyzed.

The rest of the paper is divided as follows: the background and methods description are presented after the introduction part of the paper. Next the results of research related to developing and assessing the XGBoost classifier are presented. The last part of the paper presents the analysis of local and global interpretability of the developed XGBoost, discussion, and conclusion.

2. Background

Financial organizations analyze clients in terms of their ability to repay the credit. The aim of the process is both accurate in the forecast and effective, i.e. optimal use of the resources of the organization dealing with lending activities (Kuziak & Piontek, 2022). The problem of credit risk assessment also concerns information asymmetry between the lender and

https://doi.org/10.7172/2956-7602.101.3

the borrower (Bazarbash, 2019). Reducing this asymmetry takes place both through access to information about the client and the use of appropriate statistical methods that will allow assessing the probability with which the client will repay the liability (Siddiqi, 2017). Application scoring is the basic type of credit risk modeling (Louzada et al., 2016). It is used in the process of granting new financing agreements, including credits and loans. The scoring assesses the applicant's risk of default based on the client's data, information about the product for which the client has applied, and data provided by the credit bureau. The result of the model is used to decide whether to grant credit. Scoring is only a part of the entire application process, it also covers other elements such as legal analysis and verification of completeness and correctness of data. With the current degree of automatization, this is a very important part of the contracting process, especially in the consumer finance sector. In practice, scoring is not usually the only method of credit evaluation. Some applications are analyzed through the so-called manual process, in which an analyst has to decide whether to accept or reject the application (Louzada et al., 2016) and makes the decision.

Fraud scoring is another type of the scoring model used in the credit decision process. The purpose of the scoring is to analyze applications for possible defrauding. The result of the scoring can be used to select applications that should be more closely scrutinized for the risk of possible extortion. Fraud scoring is particularly important in the context of the digitalization of the sector, but also new methods of cybercrime (Zhou et al., 2018).

Scoring methods are also applicable in the analysis of credits or loans and advances already made to clients. According to the guidelines of the European Banking Authority (EBA), banks should screen assets for significant increases in credit risk. For this purpose, banks use behavioral scorings (Goel & Rastogi, 2023). Their score is calculated based on the client's repayment history, but also other data (sociodemographic or financial conditions). The result of behavioral scoring is indirect, by reclassifying exposures between phases, used to create allowances for expected losses. As a result, the result of such scoring affects the costs and ultimately the bank's profit. Banks use the results of the behavioral model, by identifying clients with good payment discipline, in cross-upselling, for example, by offering credit card limit increases to clients with high behavioral scores (Björkegren & Grissen, 2022). Different statistical methods have been used for credit risk modeling, such as the multiple discriminant analysis (Mvula Chijoriga, 2011), Z-score Altman, E. I. (2018), and logistics regression (Falconieri et al., 2020). Credit risk modeling is also performed by artificial intelligence methods, such, as artificial neural networks (Akhtar et al., 2019), genetic algorithms (Metawa et al., 2019), support vector machines (Harris, 2013), random forest, rough set theory (Yeh et al., 2017) or XGBoost (Givari et al., 2022) or clustering methods (Kou et al., 2014). The results of the

existing approach are very distributed. Accuracy of prediction is from 64% to 93% and depends mainly on the characteristics of the analyzed data set.

Interpretability of models based on artificial intelligence is an important problem. Two main types of interpretability are indicated in related research.

The first type is local interpretability. It is a state in which the estimation result for a particular case (a single observation) can be explained in the context of the variables used in the model. Local interpretability can be particularly useful in situations where the decision is incomprehensible to the user of the algorithm or where the decision has resulted in an incorrect decision. For this reason, local interpretability is particularly important in areas such as medicine or finance (Botari et al., 2022).

Global interpretability is the second type of interpretability of machine learning models. It involves understanding how the model makes decisions based on a holistic view of its features and each of its learned components, such as weights, parameters, and structures. The model's global interpretability helps understand the distribution of the target outcome based on the variables (Molnar et al., 2020). Examples of dynamically developed interpretability methods include LIME (Di Cicco, 2019), SHAP (Silva et al., 2022) or integrated gradients (Sundararajan et al., 2017) which use mathematical theorems of the game theory or local regression models to build explanations.

Based on the existing research in the field of credit risk modeling it can be concluded, that they take into account the issues of interpretability of machine learning results to a small extent. Therefore the research question of this study is formulated as follows: How to develop a credit risk assessment model including machine learning interpretability? The main contribution of the research concerns the development of a credit risk assessment model using the XGBoost algorithm, taking into account local and global interpretability.

3. Methods

The risk modeling is based on Extreme Gradient Boosting (XGBoost) in the research. It is a method used for regression and classification problems. It is based on a sequence of decision trees using a gradient-based optimization method of the loss function to minimize the errors of weak estimators. It is an open-source library with implementations available in many programming languages (including C++, Python, and R). The algorithm was described by (Chen & Guestin, 2016). The XGBoost implementation is particularly well-known for its popularity on platforms running machine learning and artificial intelligence competitions (Nielsen, 2016). XGBoost uses the Classification and Regression Tree (CART) algorithm by default. CART is a binary tree, i.e. each non leaf node has two sub-nodes. Branches in the tree can be determined by entropy values

https://doi.org/10.7172/2956-7602.101.3

or the Gini impurity measure (Gini impurity) (Chen & Guestin, 2016). Based on the value, the best splitting point is chosen by selecting the value that will best separate the classes occurring in the set. In the case of numeric variables, all possible splitting values are analyzed, while with categorical variables the splitting point for a given variable is fixed. The tree can be deep enough to ideally classify into two classes, they can lead to a model overfitting effect. The number of nodes can be adjusted, i.e. the macro number of nodes can be set (Chen & Guestin, 2016). In the boosting process, successive classifiers are generated sequentially. If the estimator is based on decision trees, the first classifier is generated as in the case of a random forest, but the next classifier takes into account the classification quality of the previous classifier. Cases of incorrect predictions are marked with higher weights to improve classification. The final model result is based on the weighted prediction of individual estimators. A single classifier is a decision tree. The classifier learns sequentially by adding more trees, taking into account the classification results of previous trees based on probabilities. In subsequent iterations, boosting also uses bootstrapping, but as incorrect classifications of previous classifiers have increased weights, they are more likely to be correctly clustered. The result of an algorithm based on boosting can be expressed as (Chen & Guestin, 2016):

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i)$$

Where :
K is the number of classifiers used,
x are the explanatory variables,
f is the classifier belonging to the used set of classifiers.

In the XGBoost learning process, it is possible to select parameters that will affect the final classifier. Adjusting the parameters can help improve the quality of the classifier, and on the other hand, they help achieve the desired trade-off between load and variance. To reduce model overfitting, methods based on classifier regularization have been used. Regularization involves tuning the weights given in successive iterations. The parameters responsible for regularization include:

- *learning_rate* – a parameter that regulates the learning process in successive iterations of the boosting process. The value adjusts the weights selected in successive iterations. The default value is 0.3. Higher values can lead to overlearning of the classifier.
- *reg_alpha* – L1 regularization parameter; default value 0.
- *reg_lambda* – L2 regularization parameter; default value 1.
  For L1 and L2 parameters, higher values limit the degree of overfitting.

In addition, individual decision trees can also be adjusted, by pruning during the generation of the classifier or after the entire process. The parameters responsible for the shape of the trees are:

- *max_depth* – the value of the parameter determines the maximum depth of individual trees, i.e. the maximum length of the branches. The default value used is 6. Too large a value can lead to over-learning of the model and slows down the learning process.
- *gamma (min_split_loss)* – the value of the parameter determines the minimum spike in the value of the loss function at which the split will be preserved. This step is performed after the tree is generated.

By default, subsequent trees within an iteration are generated as based on the entire learning sample, but you can limit the number of both observations and variables. The parameters responsible for limiting the sample and the pool of variables include:

- *subsample* – denotes the portion of the learning sample that will be used to generate the next tree. The default value is 1, which means that on subsequent iterations the classifiers learn on the entire sample. Smaller values make the algorithm more conservative.
- *colsample_bytree* – specifies the fraction of variables from the X matrix that will be selected to be taken into account when generating the next tree. The default value is 1, at which all variables can be used.

Additional parameters include:

- *n_estimators* – indicates the number of trees generated by the boosting process. With a value of 1, the classifier is a single decision tree, so the boosting method is not applied. By default, the value of 100 is applied.
- *scale_pos_weight* – parameter used in the binary classification process with unbalanced classes in the learning sample; the default value used is 1. According to the documentation, the suggested value is [negative class count]/[positive class count].

In addition to the parameters set for the XGBoost classifier class object, the learning process can also be adjusted by changing the parameters in the *fit*() method. Through the *eval_set* argument, a dataset can be indicated that is not used for model estimation but is used to analyze the predictive power of data outside the learning sample. In subsequent iterations, the predictive power of the model on the specified sample is analyzed on an ongoing basis according to the specified *eval_metric*. For binary classification models, available metrics include the area under the ROC curve (AUC). If the parameters are used, the number of iterations that are subject to analysis for the value of the metric is also provided. If the quality of the classifier does not improve by the given number of subsequent iterations, the learning process stops and the one with the highest value of the metric is used as the final estimator. When using a validation set, the parameter *n_estmiators* represents only the maximum number of estimators used (Chen & Guestin, 2016).

https://doi.org/10.7172/2956-7602.101.3

In the research, we use also methods for performing local and global interpretability: ceteris paribus charts, SHAP and feature importance approach.

Ceteris paribus charts are also known as "what-if plots". For a given case, the effect of a change in a given variable on the estimate of the explanatory variable is analyzed under the assumption of no change in the other independent variables used. In the case of a classification problem such as a credit risk assessment – the effect of a change on the logarithm of the odds quotient (log-odds) or probability is evaluated. Such an analysis helps to understand what the algorithm's decision would look like with a change in a given variable (Kuźba et al., 2019).

The SHAP (Shapley Additive Explanations) library enables an interactive analysis of a predictive model. It is a model-independent method (model agnostic approach), which means that the interpretation does not analyze the structure of the model's performance, but only the impact of individual variables on the final result. The logic behind the library is based on Shapley's values used in game theory. The values are used to analyze the influence of each player on the outcome of a team game (Silva et al., 2022).

The feature importance approach relies on the indication statistical contribution of each feature (variable) to the underlying model when making decisions. We use such techniques as the frequency of use of variables, coverage, and gain. (Du et al., 2019).

4. Results

4.1. Tools and Data Source

Data preparation and estimation and analysis were developed in the Google Colaboratory environment. It is a free integrated development environment (from IDE) that allows code execution in Python language within the cloud. Google Colaboratory is based on Jupyter Notebooks. Python language version 3.7.13 and libraries were used for the analysis: Numpy 1.21.6, Pandas 1.3.5, Sklearn 1.0.2, XGBoost 0.90, Scipy 1.7.3, Seaborn 0.11.2, Matplotlib 3.2.2, Shap 0.40.0, Optbinning 0.14.1, Statsmodels 0.10.2.

The analysis was conducted on a Home Credit dataset made available as part of a competition held on the Kaggle.com platform. Home Credit is a loan company founded in 1997 in the Czech Republic, it offers its services in nine countries. The dataset was prepared in CSV format.

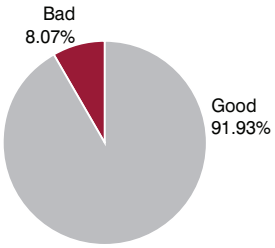
The dataset contains information about the client (age, gender, education) and the product (type of loan, day, and time of application). The dataset also contains information about the client's debt history from internal data sources, as well as from credit bureaus. The information from credit bureaus also includes information on external scoring results. The data

was largely unchanged from the source data, but some changes were applied to data that could help identify individuals or credit bureaus. Information on dates of birth and employment was replaced by the number of days relative to the date of application. Information about the building in which the applicant lives and credit scores from credit bureaus has been normalized.

4.2. Data Processing

The dataset contains 307,511 rows and 122 columns. The unique key for each client is the SK_ID_CURR column. The explained variable is located in the TARGET column. Its definition has not been defined, but by the column designation, it has been based on days of delay in repayment. Figure 1 shows that bad contracts account for about 8% of the total set of data.

Figure 1
Pie chart showing the share of classes present in the dataset



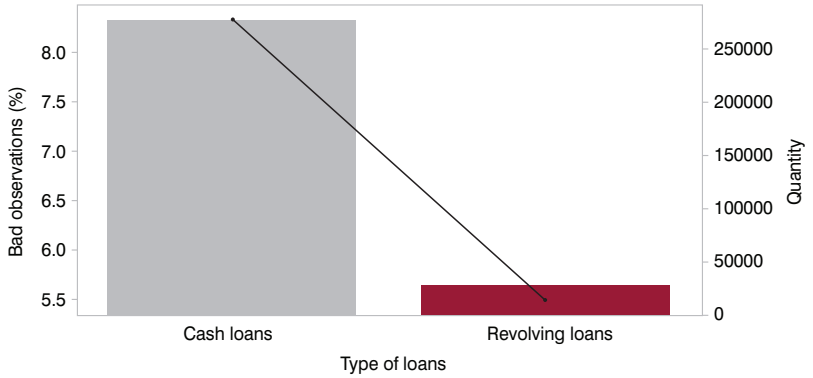
Among the variables that can be used in the modeling process, there are 105 numerical variables and 15 categoric variables. Many of the variables are characterized by a significant proportion of missing values. Some of them may be due to client characteristics, but also to the way the application is filled out. No information provided by the client during the application or no information from credit information bureaus can possess the degree of risk, so variables having missing values will be used in the estimation process.

Among the variables, there is information on the type of loan, it determines the specifics of taking funds. Among the loans, there are both cash loans and revolving loans. As can be seen in Figure 2, the degree of risk varies by type – revolving loans, which constitute a minority in the entire set, are characterized by a lower share of bad observations.

Information about the loan taken by the client also includes the amount of the loan, the amount of the installment, and the value of the financed purchase for a specific loan purpose. Among the information about clients' financial situation, there are variables specifying income, type of income, occupation, and variables specifying whether the client owns their property and their car, along with the age of the car owned.

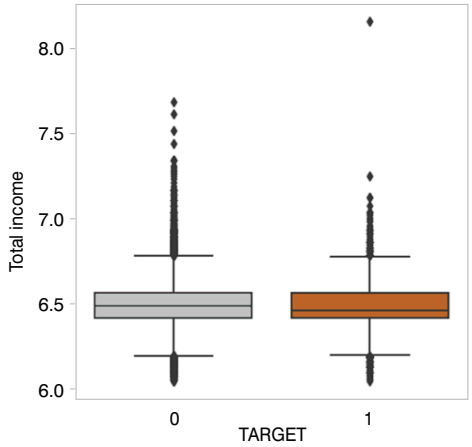
https://doi.org/10.7172/2956-7602.101.3

Figure 2
Graph of the share of "bad" cases by type of loan



Income in the set is characterized by a very wide range of values – the median is about 147 thousand, while the highest value is 117 million. To reduce the impact of outliers on the estimation results, a logarithmic transformation will be applied. In the case of the value *0* (no income shown), the missing value will be shown within the logarithmized variable. Figure 3 shows the newly created variable by the class occurring in the set.

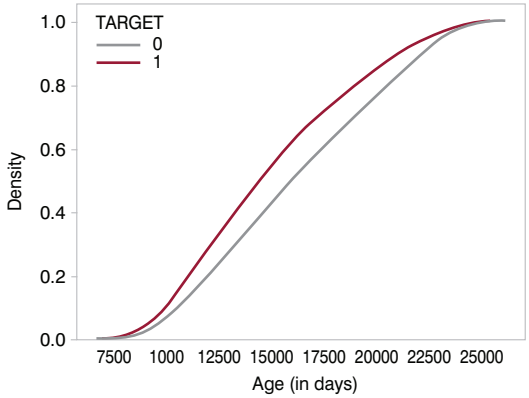
Figure 3
Box plot of the variable Total income with class division



As can be observed, the median logarithm is lower for clients with repayment problems, but the differences in the distributions are not significant.

The socio-demographic variables are age (in days), education, gender, marital status, and the number of children. The variables can have a major impact on a client's ability to repay a debt, as an example, Figure 4 shows graphs of cumulative density distributions of each class against the Age (in days) variable.

Figure 4
Graph of cumulative distributions of the Age (in days) variable by class



Since the curve for bad contracts (TARGET = 1) is above the curve for good contracts, this may suggest that young clients pose a higher risk to the lender.

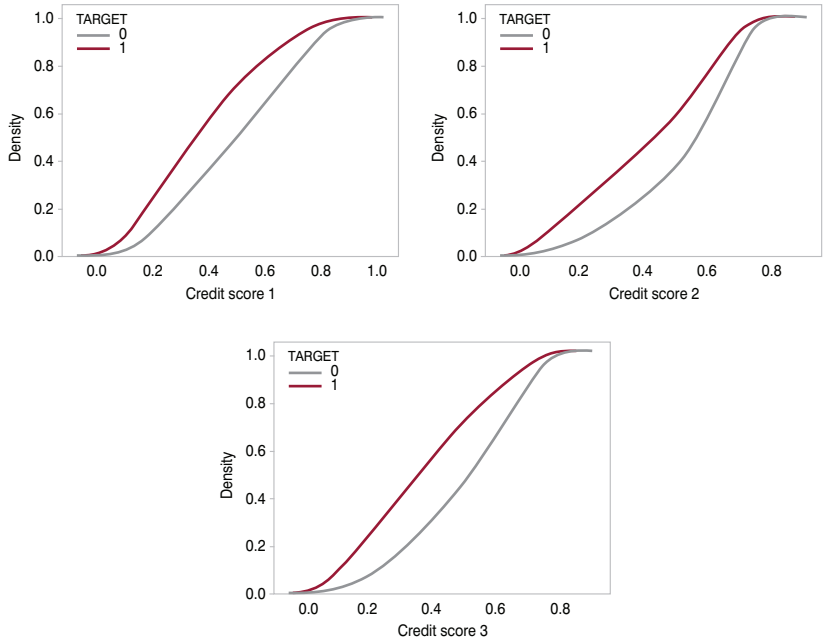
Based on the source data, additional variables were created that could potentially improve classification performance:

- the ratio of total credit to earnings earned by the client,
- the value of the financed purchase divided by the value of the loan, the value referred to in banking as LTV (Loan to Value),
- the difference between the income and the installment amount,
- the number of days of employment divided by the number of days since the client was born,
- income divided by the number of children,
- whether the value of the financed purchase field is empty; a binary variable,
- whether the age of the car owned field is empty; a binary variable (value of 1 for clients who declared having a car, but its age is unknown).

The dataset also contains information on delays and entries into default status of more than 30 and 60 days in repayments among family and people related to the client (four variables in total). The dataset also includes credit scores from three credit information of institutions. The data was normalized, i.e. moved to the interval from 0 to 1. In Figure 5, it can be seen that the variables have high predictive power.

https://doi.org/10.7172/2956-7602.101.3

Figure 5
Plots of the cumulative distributions of the Credit score 1, Credit score 2, and Credit score_3 variables by class



Based on them, variables will be added to the collection:

- the average of the scores,
- the maximum value from the point grades,
- the minimum value from the point grades,
- the difference between the maximum and minimum point grades.

The set also contains information about the number of client checks in credit reference bureaus in the last hour, and day (not including the last hour) and analogously for the week, month, quarter, and year. The remainder of the collection consists of variables about the building in which the client lives and binary variables about the individual documents the client was asked for at the time of application. The document data has been anonymized, so we don't know which documents it refers to.

The collection was divided into a training sample, representing 70% of the total collection, and a validation sample, which will use the remaining 30% of the collection. The samples were drawn, but the same proportion of bad contracts was kept in each sample.

4.3. Risk Modeling Using XGBoost

Numeric variables have been retained in their original form, while categorical variables have been transformed into binary (dummy) variables. Each categorical variable will be stored as k columns, where k is the number of categories within the variable. The XGBoost classifier, when creating nodes in successive trees, always uses the variable that maximizes the ability to separate observations from the classes present in the set, so no prior selection of variables for the mode list is required. The first step will be to select the appropriate parameters to carry out the learning process. For this purpose, cross-validation will be used. The training sample will be divided into four subsamples. For each set of parameters, the learning process will be carried out four times. In each iteration, one of the samples will be used as a validation sample, and the others will be used as learning samples.

Each iteration is evaluated in terms of the classifier quality. The metric used will be the AUC ROC, the area under the graph of the classifier quality assessment curve (Receiver Operating Characteristic). The score is then averaged over all four iterations. The iterative analysis will be carried out using the GridSearchCV class provided within the Scikit Learn library. In each iteration, the learning process will be stopped if for the next 10 iterations, the score on the validation subset does not improve. Table 10 shows the parameters along with the values considered in the search for the best combination of parameters.

Table 1
The set of searchable parameters of the XGBoost algorithm

Parameter	Searched values
objective	"binary:logistic"
max_depth	2, 3, 4, 5
learning_rate	0.05, 0.1, 0.2, 0.3
scale_pos_weight	11.387
reg_alpha	0, 5, 10, 15
subsample	0.6, 0.9, 1
colsample_bytree	0.6, 0.8, 1
n_estimators	300

The *scale_pos_weight* value was set as the ratio of good cases to bad cases, while as the case of this problem analyzed in this work involves binary classification, the objective parameter was set as "binary: logistic". In addition, the learning process used the *early_stopping_rounds* parameter, it

https://doi.org/10.7172/2956-7602.101.3

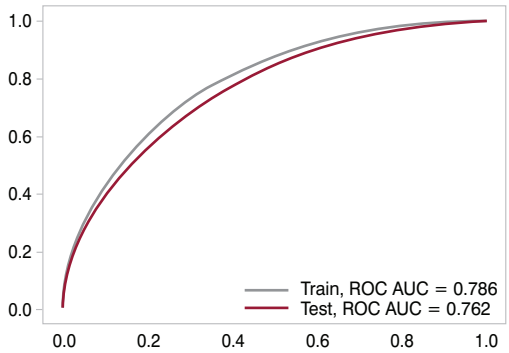
took the value of 10, meaning that the process ended if, for 10 consecutive iterations, the ROC value of the AUC on the cross-validation test sample did not improve. The highest average value of the area under the ROC curve was achieved for the values shown in Table 2.

Table 2
The selected set of parameters in the XGBoost algorithm

Parameter	Selected values of parameters
objective	"binary:logistic"
max_depth	3
learning_rate	0.1
scale_pos_weight	11.387
reg_alpha	10
subsample	0.9
colsample_bytree	0.8
n_estimators	500

The set of parameters was then used to learn the model on the entire training sample. In the process of learning the model, the same completion assumptions were used as in the cross-validation (*early_stopping_rounds* = 10) with the maximum number of iterations set as 500. Finally, the highest AUC ROC value on the test set was observed for 343 iterations. The ROC curve of the model is presented in Figure 6.

Figure 6
ROC curves for the XGBoost classifier based on the training and test sample



The area under the ROC curve is slightly larger, but no overfitting was observed. Results of the quality of the classifier are presented in table 3.

Table 3
Metrics of predictive power for the XGBoost classifier

	Training set	**Testing set**
ROC AUC	78.6%	76.2%
Gini	57.2%	52.4%
KS Statistics	42.8%	38.7%

4.4. Evaluation of the Developed Method

Results will be compared to the logistic regression method using the WOE transformation. Confusion matrices will be used to compare the effectiveness of the models. In the case analyzed, we were dealing with an unbalanced data set. The search class accounted for about 8% of the total dataset. Details of the counts of each class are presented in Table 4.

Table 4
Sample sizes by the explanatory variable

	Training set	%	Testing set	%
Target = 1 (bad client)	17 377	8.07%	7 448	8.07%
Target = 0 (good client)	197 880	91.93%	84 806	91.93%

When using classification algorithms, the selection of a probability cut-off point above which observations will be labeled as a wanted class (Target = 1) is an important element. In the case of the XGBoost algorithm, the difference in the size of each class was taken into account using the scale_pos_weight parameter, so the cut-off point will be a probability of 50%. In the case of logistic regression, the cutoff point will be set as 8.07% – observations for which the probability of occurrence of a class denoting a bad client is higher will be marked as bad. Thus, the share of each class in the predictions will be the same as in the training set. The confusion matrix of the XGBoost classifier is shown in Table 5, while the confusion matrix for the logistic regression model is shown in Table 6.

https://doi.org/10.7172/2956-7602.101.3

Table 5
The confusion matrix of the XGBoost classifier

		Prediction	
		Good client	**Bad client**
Real state	**Good client**	59 875	24 931
	Bad client	2 401	5 047

Table 6
Confusion matrix of the logistic regression model

		Prediction	
		Good client	**Bad client**
Real state	**Good client**	58 127	26 679
	Bad client	2 350	5 098

Based on the confusion matrix, metrics were determined that describe the performance of the final classification. They are presented in Table 7.

Table 7
Comparison of metrics of prediction efficiency for the XGBoost algorithm and logistic regression

Metrics	XGBoost	Logistic regression
Accuracy	70,4%	68,5%
Sensitivity	67,8%	68,4%
Precision	16,8%	16,0%
Specificity	70,6%	68,5%
F1-score	27,0%	26,0%

The metrics were counted under the assumption that the class searched for is the default case, i.e. a bad client. As can be observed, XGBoost achieved higher values of performance metrics than logistic regression, except sensitivity. It means, that XGBoost indicated a smaller percentage of all bad clients. The F1-score metric, which places the same emphasis on misidentifying a good client as well as a bad client, indicates that XGBoost is a slightly better classifier for the case under study. For metrics that are independent of cutoff points, and based only on the probability values of the class sought, XGBoost performs better. A comparison of the values of statistics determining the quality of prediction is presented in table 8.

Table 8
Comparison of metrics of the predictive power of the XGBoost classifier and logistic regression

Metric	XGBoost		Logistic regression	
	Training set	Testing set	Training set	Testing set
ROC AUC	78.6%	76.2%	74.7%	74.8%
Gini	57.2%	52.4%	49.3%	49.6%
KS Statistics	42.8%	38.7%	36.6%	37.1%

The difference in the value of the Gini metric is 2.8% on the test set. The improved predictive ability of the model used to evaluate loan applications is a tangible benefit to the bank. Lower portfolio loss is the lower cost of risk, which directly translates into profit generated by the organization (Goel & Rastogi, 2023). However, the use of such complex classification methods is associated with the loss of the benefits of logistic regression design, i.e. a very easy interpretation of results. Performance of complex algorithms, such as XGBoost, can be interpreted using the methods presented later in the article.

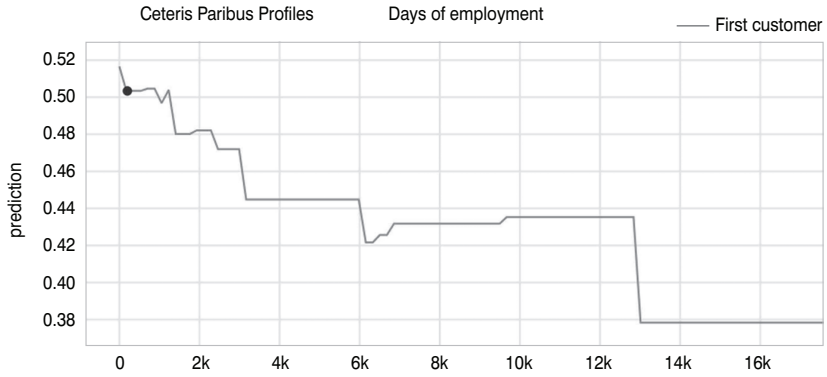
5. Interpretability of the Developed Model

5.1. Local Interpretability

In the case of the credit risk problem, local interpretability methods allow one to understand why, according to the algorithm, a credit application was rejected or accepted. A wrong decision in the case of a bank affects efficiency of the organization. Denying credit to a good client is a loss of potential profit for the bank while granting a loan to a person who cannot repay the debt is a higher cost of risk. Similar to analysts making credit decisions in a manual process, where the ability to justify decisions is required, there is increasing talk of the need for similar feedback in the case of decision-making algorithms. Figure 24 shows the dependence of the explained variable on the number of days of employment variable for one of the clients in the test sample. Assuming that the applications of customer clients for whom the probability of problems in repayment is higher than 50% are rejected, the client would not currently receive a positive credit decision. However, as can be seen in the graph shown in Figure 7, if the period of employment had been longer (instead of 503 days, the client would have been employed in his current job for 1050 days), the credit decision would have been positive.

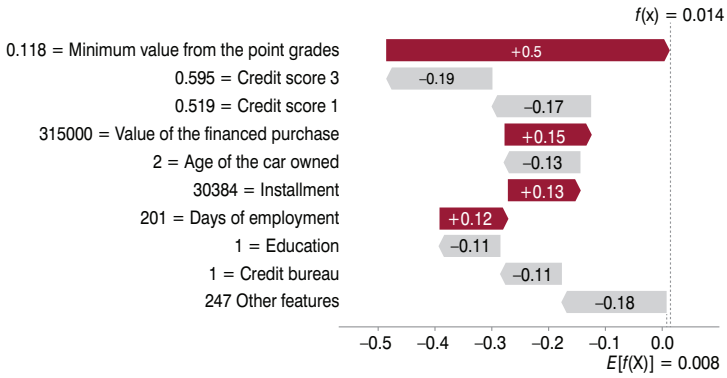
https://doi.org/10.7172/2956-7602.101.3

Figure 7
A ceteris paribus plot of the number of days of employment variable for one of the clients



As can be observed, the relationship is not monotonic at each od-cut, but on average, as seniority increases, the probability of problems in repayment decreases. No monotonicity, which would be preserved in the estimation of the model by the logistic regression method, is due to the process of the algorithm generation. The SHAP method allows the final result to be broken down into individual variables. An analysis of such a chart allows conclusions to be drawn about the relevance of individual variables (in the case of a particular client) and the direction of their influence on the final decision. In the case of the client analyzed in the previous example, the influence of individual variables is presented in Figure 8.

Figure 8
Diagram of the influence of the explanatory variables on the outcome in the case of one of the clients

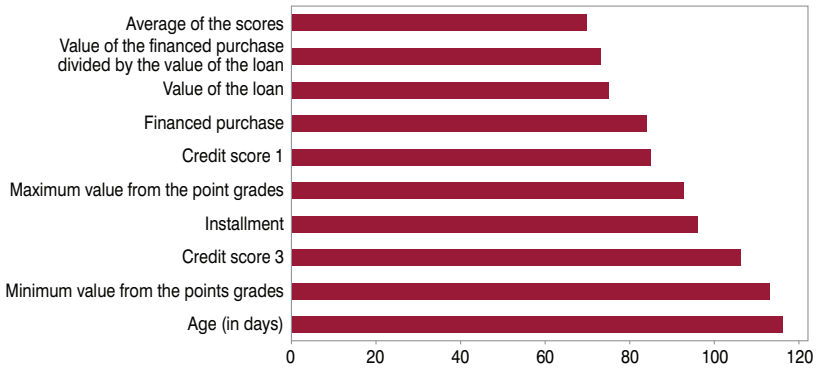


In the chart shown, the X-axis represents the natural logarithm of the chances of occurrence of the sought-after class (positive values indicate probabilities above 50%). In the case of the client in question, the credit decision is positively influenced by credit scores from external suppliers (numbers 1 and 3), while it is negatively influenced by minimal external scoring (resulting from a low score according to supplier number 2) and short seniority. The number of years in the car and higher education are also positive. Such information helps to justify a negative credit decision.

5.2. Global Interpretability

The first way to examine the performance of the XGBoost model is to analyze the feature importance of individual variables. This is a method implemented in the XGBoost library. Significance of variables (feature importance) can be measured using various techniques. The first is the frequency of use of variables across all nodes in all decision trees occurring in the model. Figure 9 shows the ten features based on which divisions are most often created within decision trees in the classifier presented in the previous section.

Figure 9
Variable relevance chart by frequency of variable use

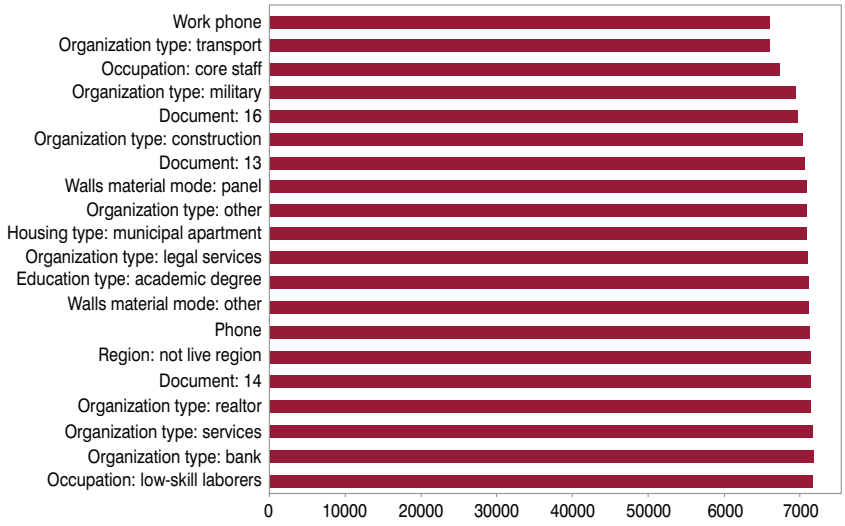


As can be observed, the most frequently used variables were those denoting the age of the client (in days), variables based on external scoring values, and variables denoting the value of the installment and the financed good. Another way of determining the relevance of individual variables is coverage, which means the total number of observations that have been separated within the tree by a given variable. Figure 10 shows a bar chart with the top 20 variables according to the metric.

This metric does not add significant information about the relevance of individual variables, as they are large variables created by discretizing variables.

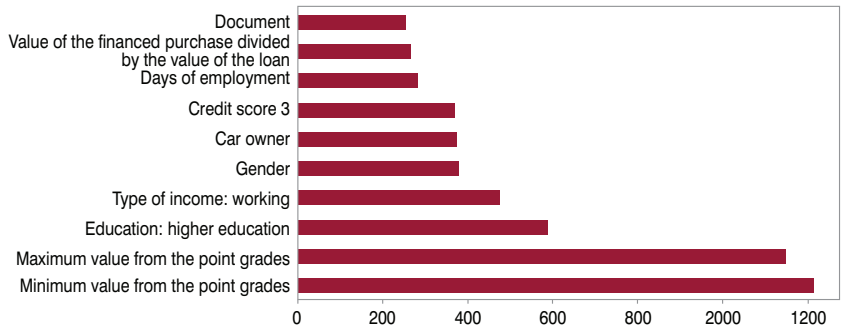
https://doi.org/10.7172/2956-7602.101.3

Figure 10
Significance chart of variables according to the coverage metric



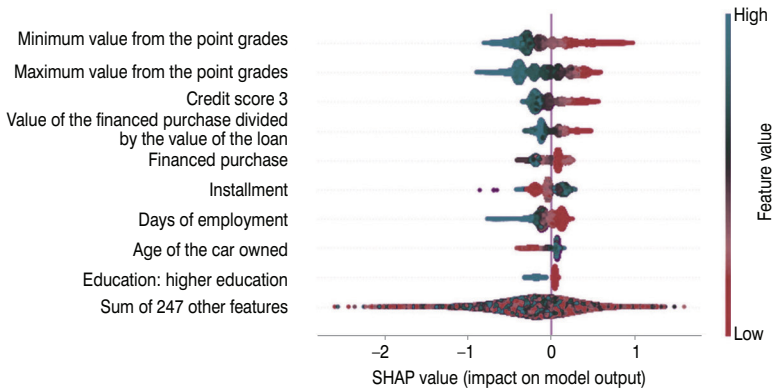
The last metric that can be used to analyze the relevance of variables is gain, the improvement that the use of a given variable brings to the final classification result. The Gain value is also the basis for determining the optimal nodes within each tree in the XGBoost algorithm. The ten most important features in the model based on the measure are shown in Figure 11. The features most important from the information "gain" perspective are the variables based on external scoring and the variables of the source of income, education, and gender. As can be observed, the choice of metrics has a large impact on the final order of significance, so all available options should be used for a complementary analysis.

Figure 11
Significance chart of variables by the Gain metric



Significance of individual variables alone is important for understanding the model, but it is not possible to tell from it the direction of the variable's effect on the explanatory variable. Beeswarm charts available within the SHAP library can help in such an analysis. An example of such a chart is shown in Figure 12.

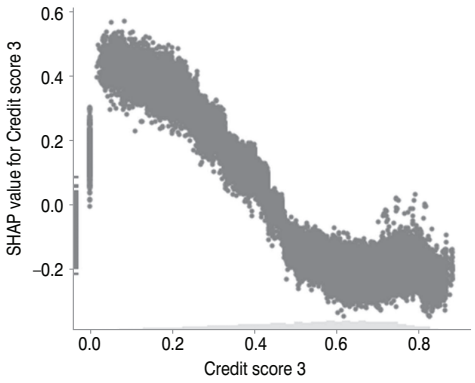
Figure 12
Beeswarm chart for the XGBoost classifier



As can be observed, higher values of the traits associated with the z-scores are accompanied by negative Shapley values, which can be interpreted as a negative effect on the explanatory variable. Since the explanatory variable takes the value of 1 in the case of a bad client, such a re-location is consistent with intuition. The situation is reversed for the age of the car owned variable describing the age of the car of the client applying for the loan. In the case of the variable, owners of older cars are assessed by the algorithm as potentially riskier. Analysis of the presented graph may indicate inconsistencies in the algorithm or the operation of the algorithm based on inconsistent intuition and business knowledge. The relationship between the value of a variable and its impact on prediction can also be shown in a dot plot. Figure 13 shows such a relationship for the variable credit score 3. According to intuition – lower scoring from an external credit information provider on average lowers the chance of a positive decision by the algorithm.

https://doi.org/10.7172/2956-7602.101.3

Figure 13
Shapley score plot for the credit score 3 variable



6. Discussion

Based on the results of the study, it can be concluded that the XGBoost algorithm achieves better credit risk forecasting results than logistic regression. The value of the Gini metric was almost three percentage points higher for the XGBoost classifier, proving its superiority. The difference may seem insignificant, but its impact on profitability can be significant from a bank's perspective. However, the effectiveness of each algorithm depends on the information value contained in the data set, which is influenced by the characteristics of the market in which the bank operates and the bank's ability to acquire the data, so it is necessary to compare different algorithms for each data set.

The interpretability of the XGBoost algorithm has been also analyzed. The analysis of the relevance of individual variables in the model was carried out by analyzing the branching that occurs in successive trees generated in the learning process. The method allows for a deeper understanding of the algorithm and an increased business knowledge of the client portfolio. Methods based on Shapley values, based on decision trees for algorithms, are optimized which allowed to reduce their computation time, allowing understanding the direction of the influence of individual variables on the sought class. Ceteris paribus charts can be used in explaining individual credit decisions and determining the conditions under which the decision would be different. Importantly, the methods can also be used with other algorithms, such as support vector-based models and neural networks. With regulatory changes potentially requiring banks in the future to more deeply understand the models used in their decision-making processes, the methods can respond to new requirements.

7. Conclusion

Due to the optimization of iterative processes, the developed credit risk assessment model using the XGBoost classifier supported by interpretation issues is a potentially interesting alternative to the standard methodology of creating a scoring card. Due to the multitude of parameters set before the learning process, the XGBoost classifier allows you to easily select parameters so that the predictive ability is at a high level. These parameters also make it possible to control the phenomenon of model overfitting, which, due to incorrect decisions, can cause large losses for the bank. The selection of parameters in the work was made using cross-validation for each possible combination of parameters, but there are other methods of such analysis. To use an approach based on Bayesian reasoning is an interesting alternative. The selection of parameters can also be carried out through optimization using genetic algorithms (Alibrahim & Ludwig, 2021).

In the case of many banks, the use of advanced decision-making algorithms in credit decision processes will also require adaptation of the IT infrastructure, which may also entail additional investment expenditures. However, as was presented in the paper, the algorithms can have a real impact on the effectiveness of the decision-making process. Summarizing the answer to the research question, it can be stated that XGBoost, A ceteris paribus plot, SHAP, and feature importance methods can be used to develop a credit risk assessment model including machine learning interpretability.

The main limitation of research is to compare the results of XGBoost only to the logistic regression results. Future research should focus on comparing the results of XGBoost to other machine learning methods, including neural networks.

Declaration of Conflicting Interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and publication of this article.

References

Addo, P.M., Guegan, D., & Hassani, B. (2018). Credit risk analysis using machine and deep learning models. *Risks*, *6*(2), 38. https://doi.org/10.3390/risks6020038

Akhtar, M., Kraemer, M.U., & Gardner, L.M. (2019). A dynamic neural network model for predicting the risk of Zika in real-time. *BMC Medicine*, *17*(1), 1–16. https://doi.org/10.1186/s12916-019-1389-3

Alibrahim, H., & Ludwig, S.A. (2021). Hyperparameter optimization: Comparing genetic algorithm against grid search and bayesian optimization. *2021 IEEE Congress on Evolutionary Computation (CEC)*, 1551–1559, IEEE. https://doi.org/10.1109/CEC45853.2021.9504761

Altman, E.I. (2018). A fifty-year retrospective on credit risk models, the Altman Z-score family of models and their applications to financial markets and managerial strategies. *Journal of Credit Risk*, *14*(4). https://doi.org/10.21314/JCR.2018.243

Bazarbash, M. (2019). Fintech in financial inclusion: machine learning applications in assessing credit risk. *IMF Working Paper*, *19*(109).

https://doi.org/10.5089/9781498314428.001
Björkegren, D., & Grissen, D. (2020). Behavior revealed in mobile phone usage predicts credit repayment. *The World Bank Economic Review*, *34*(3), 618–634. https://doi.org/10.1093/wber/lhz006
Bluhm, C., Overbeck, L., & Wagner, C. (2016). *Introduction to credit risk modeling*. Chapman and Hall/CRC. https://doi.org/10.1201/9781584889939
Botari, T., Izbicki, R., & de Carvalho, A.C.P.L.F. (2020). Local interpretation methods to machine learning using the domain of the feature space. In P. Cellier, K. Driessens (Eds), *Machine Learning and Knowledge Discovery in Databases*. ECML PKDD 2019. Communications in Computer and Information Science (vol 1167, p. 241–252). Springer International Publishing. https://doi.org/10.1007/978-3-030-43823-4_21
Chen, T., & Guestrin, C. (2016, August). *Xgboost: A scalable tree boosting system*. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). https://doi.org/10.1145/2939672.2939785
Di Cicco, V., Firmani, D., Koudas, N., Merialdo, P., & Srivastava, D. (2019). *Interpreting deep learning models for entity resolution: an experience report using LIME*. In Proceedings of the Second International Workshop on Exploiting Artificial Intelligence Techniques for Data Management (pp. 1–4). https://doi.org/10.1145/3329859.3329878
Du, M., Liu, N., & Hu, X. (2019). Techniques for interpretable machine learning. *Communications of the ACM*, *63*(1), 68–77. https://doi.org/10.1145/3359786
Duffie, D., & Singleton, K.J. (2012). *Credit risk. In Credit Risk*. Princeton University Press. https://doi.org/10.2307/j.ctv30pnvpg.17
Falconieri, N., Van Calster, B., Timmerman, D., & Wynants, L. (2020). Developing risk models for multicenter data using standard logistic regression produced suboptimal predictions: a simulation study. *Biometrical Journal*, *62*(4), 932–944. https://doi.org/10.1002/bimj.201900075
Givari, M.R., Sulaeman, M.R., & Umaidah, Y. (2022). Perbandingan Algoritma SVM, Random Forest Dan XGBoost Untuk Penentuan Persetujuan Pengajuan Kredit. *Nuansa Informatika*, *16*(1), 141–149. https://doi.org/10.25134/nuansa.v16i1.5406
Goel, A., & Rastogi, S. (2023). Credit scoring of small and medium enterprises: a behavioural approach. *Journal of Entrepreneurship in Emerging Economies*, *15*(1), 46–69. https://doi.org/10.1108/JEEE-03-2021-0093
Harris, T. (2013). Quantitative credit risk assessment using support vector machines: Broad versus Narrow default definitions. *Expert Systems with Applications*, *40*(11), 4404–4413. https://doi.org/10.1016/j.eswa.2013.01.044
Kou, G., Peng, Y., & Wang, G. (2014). Evaluation of clustering algorithms for financial risk analysis using MCDM methods. *Information Sciences*, 275, 1–12. https://doi.org/10.1016/j.ins.2014.02.137
Kuźba, M., Baranowska, E., & Biecek, P. (2019). pyCeterisParibus: explaining machine learning models with ceteris paribus profiles in Python. *Journal of Open Source Software*, *4*(37), 1389. https://doi.org/10.21105/joss.01389
Kuziak, K., & Piontek, K. (2022). Assessment of the Systemic Risk in the German Banking Industry. In T. Klein, S. Loßagk, M. Straßberger, Th. Walther (Eds.), Modern Finance and Risk Management: Festschrift in Honour of Hermann Locarek-Junge (pp. 313–332). World Scientific Publishing Company. https://doi.org/10.1142/9781800611917_0014
Li, J., Liu, H., Yang, Z., & Han, L. (2021). A credit risk model with small sample data based on G-XGBoost. *Applied Artificial Intelligence*, *35*(15), 1550–1566. https://doi.org/10.1080/08839514.2021.1987707
Louzada, F., Ara, A., & Fernandes, G.B. (2016). Classification methods applied to credit scoring: Systematic review and overall comparison. *Surveys in Operations Research and Management Science*, 21(2), 117–134. https://doi.org/10.1016/j.sorms.2016.10.001

Metawa, N., Hassan, M.K., & Elhoseny, M. (2017). Genetic algorithm based model for optimizing bank lending decisions. *Expert Systems with Applications*, 80, 75–82. https://doi.org/10.1016/j.eswa.2017.03.021

Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C.A., Casalicchio, G., Grosse-Wentrup, M., & Bischl, B. (2022). General pitfalls of model-agnostic interpretation methods for machine learning models. In A. Holzinger, R. Goebel, R. Fong, T. Moon, K.R. Müller, W. Samek, W. (Eds.), xxAI – Beyond Explainable AI. xxAI 2020. Lecture Notes in Computer Science(), vol 13200. Springer, Cham. https://doi.org/10.1007/978-3-031-04083-2_4

Mvula Chijoriga, M. (2011). Application of multiple discriminant analysis (MDA) as a credit scoring and risk assessment model. *International Journal of Emerging Markets*, *6*(2), 132–147. https://doi.org/10.1108/17468801111119498

Nielsen, D. (2016). *Tree Boosting with XGBoost. Tree boosting with XGBoost*. Norwegian University of Science and Technology.

Siddiqi, N. (2017). *Intelligent credit scoring: Building and implementing better credit risk scorecards*. John Wiley & Sons. https://doi.org/10.1002/9781119282396

Silva, S.J., Keller, C.A., & Hardin, J. (2022). Using an Explainable Machine Learning Approach to Characterize Earth System Model Errors: Application of SHAP Analysis to Modeling Lightning Flash Occurrence. *Journal of Advances in Modeling Earth Systems*, *14*(4), e2021MS002881. https://doi.org/10.1029/2021MS002881

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. Proceedings of the 34th International Conference on Machine Learning. *Proceedings of Machine Learning Research*, *70*, 3319–3328. https://proceedings.mlr.press/v70/sundararajan17a.html.

Yeh, C.C., Lin, F., & Hsu, C.Y. (2012). A hybrid KMV model, random forests and rough set theory approach for credit rating. *Knowledge-Based Systems*, *33*, 166–172. https://doi.org/10.1016/j.knosys.2012.04.004

Zhou, X., Cheng, S., Zhu, M., Guo, C., Zhou, S., Xu, P., Xue, Z., & Zhang, W. (2018). A state of the art survey of data mining-based fraud detection and credit scoring. *MATEC Web of Conferences*, *189*(3), 03002. https://doi.org/10.1051/matecconf/201818903002