Comparing Supervised Classification Methods for Financial Domain Problems

Victor Ulisses Pugliese[a], Celso Massaki Hirata[b] and Renato Duarte Costa[c]
Instituto Tecnológico de Aeronáutica, Praça Marechal Eduardo Gomes, 50, São José dos Campos, Brazil

Keywords: Ranking, Machine Learning, XGBoost, Nonparametric Statistic, Optimization Hyperparameter.

Abstract: Classification is key to the success of the financial business. Classification is used to analyze risk, the occurrence of fraud, and credit-granting problems. The supervised classification methods help the analyzes by 'learning' patterns in data to predict an associated class. The most common methods include Naive Bayes, Logistic Regression, K-Nearest Neighbors, Decision Tree, Random Forest, Gradient Boosting, XGBoost, and Multilayer Perceptron. We conduct a comparative study to identify which methods perform best on problems of analyzing risk, the occurrence of fraud, and credit-granting. Our motivation is to identify if there is a method that outperforms systematically others for the aforementioned problems. We also consider the application of Optuna, which is a next-generation Hyperparameter optimization framework on methods to achieve better results. We applied the non-parametric Friedman test to infer hypotheses and we performed Nemeyni as a posthoc test to validate the results obtained on five datasets in Finance Domain. We adopted the performance metrics F1 Score and AUROC. We achieved better results in applying Optuna in most of the evaluations, and XGBoost was the best method. We conclude that XGBoost is the recommended machine learning classification method to overcome when proposing new methods for problems of analyzing risk, fraud, and credit.

1 INTRODUCTION

Business success and failure have been extensively studied. Most of the studies try to identify the various determinants that can affect business existence (Yu et al., 2014). Businesses operations are conducted based on how companies make financial decisions and depend on models to support the decisions. Inadequate models can lead to business failure (Damodaran, 1996).

In most of the studies, decisions are based on the prediction of classification about problems such as granting credit, credit card fraud detection, and bankruptcy risk and are commonly treated as binary classification problems (Yu et al., 2014)(Lin et al., 2011).

In this paper, we conduct a comparative study to identify which supervised classification methods perform best on problems of analyzing risk, the occurrence of fraud, and credit granting. The motivation is to identify a winning method that has the best performance for all the aforementioned problems.

In order to achieve the goal, we selected nine predictive methods. To contextualize our work, we made a survey of the related work. Then, we conducted an evaluation comparing the methods using two groups of datasets. The first group is associated to finance domain. The other is related to health care and ionosphere. Finally, we present the main findigs and conclude the paper.

2 BACKGROUND

This section briefly describes the nine methods for financial prediction. They are Naive Bayes, Logistic Regression, Support Vector Classifier, k-Nearest Neighbors, Decision Tree, Random Forest, Gradient Boosting, XGBoost, and Multilayer Perceptron.

Naïve Bayes (NB) classifier is based on applying Bayes's theorem with strong (naïve) independence assumptions (Rish et al., 2001):

$$p(X|Y) = \prod_{i=1}^{n} p(X_i|Y) \quad (1)$$

[a] https://orcid.org/0000-0001-8033-6679
[b] https://orcid.org/0000-0002-9746-7605
[c] https://orcid.org/0000-0002-8378-5485

Pugliese, V., Hirata, C. and Costa, R.
Comparing Supervised Classification Methods for Financial Domain Problems.
DOI: 10.5220/0009426204400451
In *Proceedings of the 22nd International Conference on Enterprise Information Systems (ICEIS 2020) - Volume 1*, pages 440-451
ISBN: 978-989-758-423-7

where p is a probability, $X(X_1,\ldots,X_n)$ is a feature vector and Y is a class. The theorem establishes that the class Y given the feature X, the posterior probability, $p(Y|X)$, can be calculated by the class prior probability, $p(Y)$, multiplied by the observed feature probability, $p(X|Y)$, or likelihood, divided by the total feature probabilit, $p(X)$, which is constant for all classes (Pearl et al., 2016).

$$p(Y|X) = \frac{p(Y) * p(X|Y)}{p(X)} \tag{2}$$

Although the independence between features is a condition not fully sustained in most cases, the Naïve Bayes has proved its strength in practical situations with comparable performance to Neural Network and Decision Tree classifiers (Islam et al., 2007).

Logistic Regression (LR) is a classification method used to predict the probability of a categorical dependent variable assigning observations to a discrete set of classes (yes or no, success or failure). Unlike linear regression which outputs continuous number values, logistic regression transforms its output using the logistic sigmoid function (Equation 3) to return a probability value, which can then be mapped to discrete classes. The logistic sigmoid function maps any real value into another value between 0 and 1. A decision threshold classifies values into classes 0 or 1.

$$S(z) = \frac{1}{1+e^{-z}} \tag{3}$$

The Logistic Regression is binary if the dependent variable is a binary variable (pass or fail), multinomial if the dependent variable is categorical as type of animal or flower, and ordinal for ordered classes like Low, Medium or High. Ng and Jordan (Ng and Jordan, 2002) present a comparison between Naïve Bayes and Logistic Regression classifier algorithms.

K-Nearest Neighbor (kNN) is a non-parametric method for classification and regression tasks. It is one of the most fundamental and simplest methods, being the first choice method for classification when there is little or no prior knowledge about the distribution of the data (Peterson, 2009). Examples are classified based on the class of their nearest neighbors. It is usually used to identify more than one neighbor, where k is a referee for determining classes number. This method uses metrics that must conform to the following four criteria (where d(x, y) refers to the distance between two objects x and y) (Cunningham and Delany, 2007):

- d(x, y) is greater or equal to zero; non-negativity
- d(x, y) is equal to zero only if x=y; identity
- d(x, y) is equal to d(y, x); symmetry
- d(x, z) is less or equal to d(x, y) + d(y, z); triangle inequality

Support Vector Classifier (SVC) is a statistical learning method that is suitable for binary classification (Zareapoor and Shamsolmoali, 2015). The objective of the Support Vector Classifier is to find a hyperplane in n-dimensional space, where n is the number of features, that distinctly classifies the data (Suykens and Vandewalle, 1999).

Decision Tree (DT) is a flow chart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node holds a class label (Lavanya and Rani, 2011). Decision tree classifiers are commonly used in credit card, automobile insurance, and corporate fraud problems.

Random Forest (RandFC) is proposed as an additional layer of randomness bagging tree (Breiman, 2001) (Liaw et al., 2002). The Random Forest collects data and searches a random selection of features for the best division on each node, regardless of previous trees. In the end, a simple majority vote is made for prediction. Random Forest performs very well compared to many other classifiers, including discriminant analysis, support vector classifier and neural networks, being robust against overfitting (Liaw et al., 2002).

Gradient Boosting (GradB) is based on a different constructive strategy of ensemble set like Random Forest. The Boosting's main idea is to add new models to the ensemble sequentially (Natekin and Knoll, 2013). Boosting fits the "weak" tree classifiers to different observation weights in a dataset (Ridgeway, 1999). In the end, a weighted vote is made for prediction (Liaw et al., 2002).

XGBoost (XGB) is a scalable machine learning system for optimized tree boosting. The method is available as an open source package. Its impact has been widely recognized in a number of machine learning and data mining challenges (Chen and Guestrin, 2016). XGBoost became known after winning the Higgs Challenge, available at https://www.kaggle.com/c/higgs-boson/overview. XGBoost has several features such as parallel computation with OpenMP. It is generally over 10 times faster than Gradient Boosting. XGBoost takes several types of input data. It supports customized objective function and evaluation function. It has better performance on several different datasets.

Multilayer Perceptron (NN) is a feed-forward artificial neural network model for supervised learning, composed by a series of layers of nodes or neurons with full interconnection between adjacent layer nodes. The feature vector X is presented to the in-

put layer. Its nodes output values are fully connected to the next layer neurons through weighted synapses. The connections repeat until the output layer, responsible to present the results of the network. The learning of NN is made by the back-propagation algorithm. The training is done layer by layer, adjusting the synaptic weights from the last to the first layer, to minimize the error. Accuracy metrics such as minimum square error is used. The algorithm repeats the training process several times. Each iteration is called epoch. On each epoch, the configuration that presents the best results is used as the seed for the next interaction, until some criterion as accuracy or number of iterations is reached.

To measure the performance of the predictive methods employed in this study, we use the metrics: *F1 Score, Precision, Recall and AUROC*.

F1 Score is the harmonic mean of *Precision* and *Recall*. *Precision* is the number of correct positive results divided by the number of positive results predicted. *Recall* is the number of correct positive results divided by the number of all samples that should have been identified as positive. F1 score reaches its best value at 1 (perfect precision and recall) and worst at 0 (Equation 4).

$$F1\,Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

AUROC (Area under the Receiver Operating Characteristic) is a usual metrics for the goodness of a predictor in a binary classification task.

To evaluate the methods with the datasets, we employ some tests. The *Friedman* Test is a nonparametric equivalent of repeated measures analysis of variance (ANOVA) (Demšar, 2006). The purpose of the test is to determine if one can conclude from a sample of results that there is a difference between the treatment effect (García et al., 2010).

The *Nemenyi* Test is a post-hoc test of Friedman applied when all possible pairwise comparisons need to be performed. It assumes that the value of the significance level α is adjusted in a single step by dividing it merely by the number of comparisons performed.

Hyperparameter optimization is one of the essential steps in training Machine Learning models. With many parameters to optimize, long training time and multiple folds to limit information leak, it is a cumbersome endeavor. There are a few methods of dealing with the issue: grid search, random search, and Bayesian methods. Optuna is an implementation of the latter one

Optuna is a next-generation Hyperparameter Optimization Framework (Akiba et al., 2019). It has the following features: define-by-run API that allows users to construct the parameter search space dynamically; efficient implementation of both searching and pruning strategies; and easy-to-setup, versatile architecture

3 RELATED WORK

There are two systematic literature reviews (Bouazza et al., 2018) (Sinayobye et al., 2018) that describe the works on data mining techniques applied in financial frauds, healthcare insurance frauds, and automobile insurance frauds.

Moro et al. (Moro et al., 2014) propose a data mining technique approach for the selection of bank marketing clients. They compare four models: Logistic Regression, Decision Tree, Neural Networks, Support Vector Machines, using the performance metrics AUROC and LIFT. For both metrics, the best results were obtained by Neural Network. Moro et al. do not use bagging or boosting tree.

Zareapoor and Shamsolmoali (Zareapoor and Shamsolmoali, 2015) apply five predictive methods: Naive Bayes, k-Nearest Neighbors, Support Vector Classifier, Decision Tree, and Bagging Tree to credit card's dataset. They report that Bagging Tree shows better results than others. Zareapoor and Shamsolmoali do not use Boosting Tree, Neural Network and Logistic Regression as we do. Their survey does not have a nonparametric test.

Wang et al. (Wang et al., 2011) explore credit scoring with three bank credit datasets: Australian, German, and Chinese. They made a comparative assessment of performance of three ensemble methods, Bagging, Boosting, and Stacking based on four base learners, Logistic Regression, Decision Tree, Neural Network and Support Vector Machine. They found that Bagging performs better than Boosting across all credit datasets. Wang et al do not use k-Nearest Neighbors and XGBoost.

4 EVALUATION OF THE METHODS WITH DATASETS OF THE FINANCIAL AREA

We used five datasets of the financial domain in the evaluations. They are briefly described as follows.

The *Bank Marketing* dataset is about direct marketing campaigns (phone calls) of a Portuguese banking institution. It contains personal information and banking transaction data of clients. The classification goal is to predict if a client will subscribe to a

term deposit. The dataset is multivariate with 41188 instances (4640 subscription), 21 attributes (5 real, 5 integer and 11 object), no missing values and it is available at https://archive.ics.uci.edu/ml/datasets/Bank+Marketing (Moro et al., 2014).

The *Default of Credit Card Clients* dataset contains information of default payments, demographic factors, credit data, history of payment, and bill statements of credit card clients in Taiwan from April to September 2005. The classification goal is to predict if the clients is credible. The dataset is multivariate with 30000 instances (6636 creditation), 24 integer attributes, no missing values, and it is available at https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients (Bache and Lichman, 2013).

The *Kaggle Credit Card* dataset is a modified version of Default of Credit Card Clients, with data in the same period. Both datasets have the same classification goal: predict if the client is credible. However, Kaggle Credit Card has more features, almost 31 only numerical attributes. and a lower number of positive credible client instances. The dataset has 284807 instances (492 positive credible client instances). The dataset is highly unbalanced and the positive class accounts for 0.172% of all instances. It is available at https://www.kaggle.com/uciml/default-of-credit-card-clients-Dataset (Dal Pozzolo et al., 2015)

The *Statlog German Credit* dataset contains categorical and symbolic attributes. It contains credit history, purpose, personal client data, nationality, and other information. The goal is to classify clients using a set of attributes as good or bad for credit risk. We used an alternative dataset provided by Strathclyde University. The file was edited and several indicator variables were added to make it suitable for algorithms that cannot cope with categorical variables. Several attributes that are ordered categorically (such as attribute 17) were coded as integer. The dataset is multivariate with 1,000 instances (300 instances are classified as Bad), 24 integer attributes, no missing values and it is available at https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data) (Hofmann, 1994).

The *Statlog Australian Credit Approval* dataset is used for analysis of credit card operations. All attribute names and values were anonymized to protect data privacy. The dataset is multivariate with 690 instances (307 instances are labeled as 1), 14 attributes (3 real and 11 integer), no missing values and it is available at http://archive.ics.uci.edu/ml/datasets/statlog+(australian+credit+approval) (Quinlan, 1987).

For each dataset, we preprocessed the attributes, sampled the data, and divided the data into 90% for training and 10% for testing. After splitting the dataset, we employed cross-validation with ten Stratified k-folds, fifteen seeds (55, 67, 200, 245, 256, 302, 327, 336, 385, 407, 423, 456, 489, 515, 537), and nine predictive methods. Firstly, the methods used the scikit-learn default hyperparameters. The *F1 Score and AUROC* metrics were measured. Tests were performed on the measured metrics to rank statistic differences over methods. Finally, we employed Optuna to optimize the hyperparameters and used the classification methods again.

The main scikit-learn default hyperparameters used to test the different methods are::

- GaussianNB: priors='None', and var_smoothing='1e-09'.
- Logistic Regression: C=1.0, fit_intercept=True, intercept_scaling=1, max_iter=100, penalty='l2', random_state=None, solver='warn', and tol=0.0001.
- kNN: algorithm='auto', leaf_size=30, metric='minkowski', n_neighbors=5, p=2, and weights='uniform'.
- SVC: C=1.0, cache_size=200, decision_function_shape='ovr', degree=3, kernel='rbf', shrinking=True, and tol=0.001.
- Decision Tree: criterion='gini', min_samples_split=2, and splitter='best'.
- Random Forest: bootstrap=True, criterion='gini', min_samples_leaf=1, min_samples_split=2, and n_estimators='warn'.
- Gradient Boosting: criterion='friedman_mse', learning_rate=0.1, loss='deviance', max_depth=3, min_samples_leaf=1, min_samples_split=2, n_estimators=100, subsample=1.0, tol=0.0001, and validation_fraction=0.1.
- XGBoost: base_score=0.5, booster='gbtree', learning_rate=0.1, max_depth=3, and n_estimators=100.
- Multilayer Perceptron: activation='relu', hidden_layer_sizes=(100,), learning_rate='constant', max_iter=200, solver='adam', and tol=0.0001.

We have used Optuna to optimize the hyperparameters in the methods, running one study with 100 iterations, using the following ranges:

- GaussianNB: none.
- Logistic Regression: C range: 1e-10 to 1e10.
- kNN: N_neighbors range: 1 to 100; Distances range: 1 to 10.

- SVC: C range: 1e-10 to 1e10; Kernel options: linear, rbf, poly; Gamma range: 0.1 to 100; Degree range: 1 to 6.
- Decision Tree: Max_depth range: 2 to 32; Min_samples_split range: 2 to 100; Min_samples_leaf range: 1 to 100.
- Random Forest: Same hyperparameters used in Decision Tree; N_estimators range: 100 to 1000.
- Gradient Boosting: Same hyperparameters used in Random Forest; Learning_rate range: 0.01 to 1.
- XGBoost: Booster options: gbtree, gblinear, dart; Lambda range: 1e-8 to 1.0; Alpha range: 1e-8 to 1.0. Testing booster as gbtree or dart, than Max_depth range: 1 to 9; eta range: 1e-8 to 1.0; Gamma range: 1e-8 to 1.0; Grow_policy options: depthwise or lossguide. As Dart Booster, we could test too, Sample_type options: uniform or weighted; Normalize_type options: tree or forest; Rate_drop range: 1e-8 to 1.0; Skip_drop range: 1e-8 to 1.0.
- Multilayer Perceptron: Hidden Layer Sizes options: (100,), (50,50,50), (50,100,50); Activation options: identity, logistic, tanh, relu; Solver options: sgd or adam; Alpha range: 0.0001 to 5; Learning_rate options: constant or adaptive.

5 RESULTS WITH THE DATASETS OF THE FINANCE DOMAIN

In this section, we present the results of the classification methods for the five datasets in the finance domain.

For the *Bank Marketing* dataset, we transformed categorical data with One-Hot-Encoding. Afterwards, we applied undersampling to balance the dataset. Undersampling is an algorithm to deal with class-imbalance problems. It uses only a subset of the majority class for efficiency (Liu et al., 2008), and we employed the methods. The results are shown in Table 1.

As it can be observed, the lowest values of F1 Score and AUROC were obtained by Naive Bayes with 65.47% and 71.59%, respectively. The best results were achieved with GradientBoosting with 88.87% of F1 Score and 88.41% of AUROC, followed by XGBoost (88.76% of F1 and 88.23% of AUROC). With respect to standard deviations for F1 Score and AUROC, Gradient Boosting resulted in 0.08% and 0% respectively and XGBoost resulted in 0% for both.

Table 1: Cross-validation for Bank Marketing Dataset.

Classifiers	F1	std	AUROC	std
DT	83.19	0.18	83.19	0.11
RandFC	86.43	0.20	86.47	0.21
GradB	88.87	0.08	88.41	0.00
XGB	88.76	0.00	88.23	0.00
LR	87.01	0.00	86.85	0.00
SVC	85.62	0.00	84.76	0.00
kNN	85.48	0.00	85.20	0.00
NN	81.97	2.26	80.95	1.83
NB	65.47	0.00	71.59	0.00

Figures 1 and 2 illustrate the Critical Difference Diagram constructed using Nemenyi Test for F1 Score and AUROC in the Bank Marketing dataset.

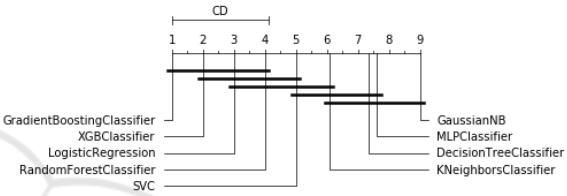

Figure 1: Critical difference diagram over F1 measure of Bank Marketing Dataset.

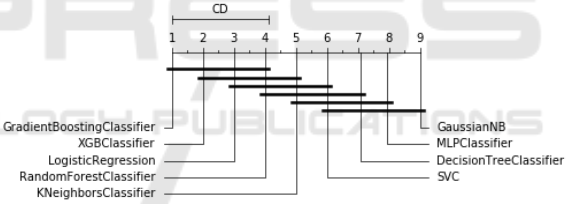

Figure 2: Critical difference diagram over AUROC measure of Bank Marketing Dataset.

As it can be seen in Bank Marketing dataset, Gradient Boosting, XGBoost, Logistic Regression, and Random Forest are the best, but no statistically significant difference could be observed among them. Thus, the methods can be used, with similar efficiency, to classify clients for a term deposit.

We employed Optuna over the methods in the Bank Marketing dataset (Table 2). The best result was achieved with XGBoost with 89.56% of F1 Score and 89.11% of AUROC.

We obtained the best results using XGBoost with Optuna for the Bank Marketing dataset, with the following setting parameters: 'booster' = 'dart', 'lambda' = 4.763778055855053e-06, 'alpha' = 0.0056726686023193555, 'max_depth' = 5, 'eta' = 1.9313322604903697e-07, 'gamma' = 1.567431491678084e-08, 'grow_policy' = 'lossguide', 'sample_type' = 'uniform', 'normalize_type'

= ’forest’, ’rate_drop’ = 0.003875800179107411, ’skip_drop’ = 1.4617070871276763e-08.

Table 2: Cross-validation with Optuna in Bank Marketing Dataset.

Classifiers	F1	std	AUROC	std
DT	84.54	0.00	84.31	0.00
RandFC	84.62	0.20	84.07	0.11
GradB	88.60	0.00	88.11	0.00
XGB	89.56	0.00	89.11	0.00
LR	87.02	0.00	86.85	0.00
SVC	87.08	0.00	86.88	0.00
kNN	85.80	0.00	85.45	0.00
NN	87.25	0.00	86.59	0.00
NB	65.47	0.00	71.59	0.00

For the *Default of Credit Card Clients dataset*, we applied undersampling to balance the dataset. Afterwards, we employed the methods. The results are shown in Table 3

Table 3: Cross-validation for Default of Credit Card Clients Dataset.

Classifiers	F1	std	AUROC	std
DT	62.77	0.19	62.64	0.17
RandFC	65.15	0.35	67.99	0.26
GradB	68.43	0.09	70.84	0.07
XGB	68.43	0.00	70.85	0.00
LR	65.19	0.00	62.54	0.00
SVC	8.49	0.00	51.51	0.00
kNN	59.23	0.00	58.87	0.00
NN	59.14	3.12	58.85	1.19
NB	67.38	0.00	54.16	0.00

As it can be observed, the lowest values of F1 Score and AUROC were obtained by Support Vector Classifier with 8.49% and 51.51%, respectively. The best results were achieved with XGBoost with 68.43% of F1 Score and 70.85% of AUROC, followed by Gradient Boosting with 68.43% of F1 and 70.84% of AUROC. With respect to standard deviations for F1 Score and AUROC, Gradient Boosting and XGBoost resulted in almost 0%.

Figures 3 and 4 show the Critical Difference Diagram constructed using Nemenyi Test for F1 Score and AUROC in the Default of Credit Card Clients dataset.

As it can be seen, Gradient Boosting, XGBoost, and Naive Bayes are the best, but no statistically significant difference could be observed among them. Thus, the above methods can be used with similar efficiency for classifying a credible client.

We employed Optuna over the methods in the Default of Credit Card Clients dataset (Table 4). The best results were achieved by GradB with 68.88% of F1 Score and 70.91% of AUROC, with standard deviation of 0.00%. XGBoost obtained very similar results.

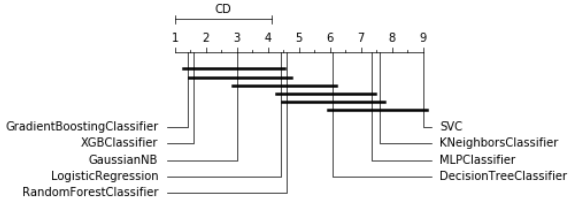

Figure 3: Critical difference diagram over F1 measure of Default of Credit Card Clients Dataset.

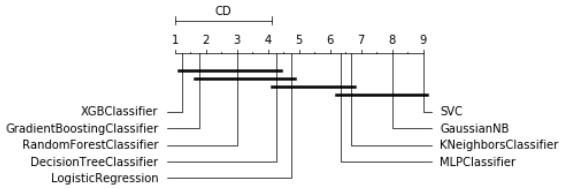

Figure 4: Critical difference diagram over AUROC measure of Default of Credit Card Clients Dataset.

Table 4: Cross-validation with Optuna for Default of Credit Card Clients Dataset.

Classifiers	F1	std	AUROC	std
DT	61.81	0.00	68.31	0.00
RandFC	66.19	0.17	69.50	0.13
GradB	68.88	0.00	70.91	0.00
XGB	68.61	0.00	70.52	0.00
LR	65.26	0.00	61.75	0.00
SVC	61.01	0.00	60.56	0.00
kNN	64.70	0.00	61.28	0.00
NN	58.01	5.37	58.74	0.67
NB	67.38	0.00	54.21	0.00

We obtained the best results using GradB with Optuna for the Default of Credit Card Clients dataset, with the following setting parameters: ’learning_rate’: 0.06551574044228455, ’n_estimators’: 355.41370517846616, ’max_depth’: 4.935444994782639, ’min_samples_split’: 12.868275268442062, ’min_samples_leaf’: 5.444818807968713.

For the *Kaggle Credit Card dataset*, we applied undersampling to balance the dataset. Afterwards, we employed the methods. The results are shown in Table 5.

As it can be observed, the lowest values of F1 Score and AUROC were obtained by Naive Bayes (89.88% and 90.68%), and Decision Tree (90.18% and 90.31%). The best results were achieved by XG-

Table 5: Cross-validation for Kaggle Credit Card Dataset.

Classifiers	F1	std	AUROC	std
DT	90.18	0.60	90.31	0.36
RandFC	92.62	0.47	92.94	0.37
GradB	93.50	0.07	93.76	0.06
XGB	94.07	0.00	94.29	0.00
LR	92.98	0.00	93.26	0.00
SVC	92.28	0.00	92.59	0.00
kNN	92.51	0.00	92.89	0.00
NN	93.40	0.25	93.68	0.20
NB	89.88	0.00	90.68	0.00

Boost with 94.07% of F1 Score and 94.29% of AUROC, and Gradient Boosting (93.50% and 93.76%). When it comes to standard deviation for F1 Score and AUROC, Gradient Boosting and XGBoost resulted in 0%.

Figures 5 and 6 bring the Critical Difference Diagram constructed using Nemenyi Test for F1 Score and AUROC in the Kaggle Credit Card dataset.

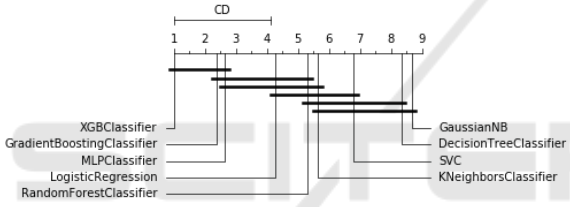

Figure 5: Critical difference diagram over F1 measure of Kaggle Credit Card Dataset.

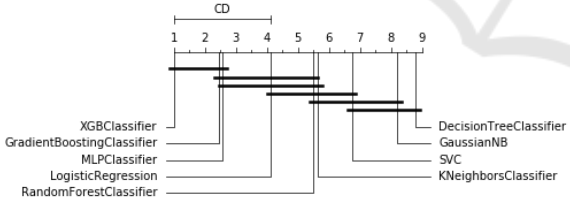

Figure 6: Critical difference diagram over AUROC measure of Kaggle Credit Card Dataset.

As it can be seen in Kaggle Credit Card dataset, XGBoost, Gradient Boosting, Multilayer Perceptron, and Logistic Regression obtained the best results, but no statistically significant difference could be observed among them. Thus, the above methods can be used with similar efficiency for classifying who is a creditable client or not.

We employed Optuna over the methods in the Kaggle Credit Card dataset (Table 6). XGBoost keeps the best results after applying Optuna as well for this dataset.

For Statlog German Credit dataset, we applied SMOTE algorithm to balance the dataset. In SMOTE,

Table 6: Cross-validation with Optuna in Kaggle Credit Card Dataset.

Classifiers	F1	std	AUROC	std
DT	91.03	0.00	91.36	0.00
RandFC	91.57	0.00	93.88	0.00
GradB	93.66	0.00	92.13	0.00
XGB	94.05	0.00	94.28	0.00
LR	92.39	0.00	92.70	0.00
SVC	92.28	0.00	92.59	0.00
kNN	92.96	0.00	93.26	0.00
NN	93.37	0.00	93.56	0.00
NB	89.88	0.00	90.68	0.00

the minority class is oversampled by duplicating samples. Depending on the oversampling required, numbers of nearest neighbors are randomly chosen (Bhagat and Patil, 2015). Afterwards, we employed the predictive methods. The results are shown in Table 7.

Table 7: Cross-validation for Statlog German Credit.

Classifiers	F1	std	AUROC	std
DT	72.57	0.00	72.17	1.41
RandFC	77.23	0.25	78.17	1.33
GradB	81.39	0.00	81.31	0.52
XGB	82.06	0.00	82.03	0.00
LR	79.46	0.00	79.33	0.00
SVC	80.95	0.00	48.57	0.00
kNN	78.92	0.00	81.31	0.00
NN	81.83	1.43	67.90	1.32
NB	72.06	0.00	74.24	0.00

As it can be observed, the lowest values of F1 Score and AUROC were obtained by Naive Bayes with 72.06% and 74.24%, respectively. The best results were achieved by XGBoost with 82.06% of F1 Score and 82.03% of AUROC. When it comes to standard deviation for F1 Score and AUROC, XGBoost resulted in 0% .

Figures 7 and 8 show the Critical Difference Diagram constructed using Nemenyi Test for F1 Score and AUROC in the Statlog German Credit dataset.

As it can be seen in the figures, Support Vector Classifier, Multilayer Perceptron, Gradient Boosting, and XGBoost obtained the best results, but no statistically significant difference could be observed among them. Thus, the aforementioned methods can be employed with similar efficiency for classifying who is credible client.

We employed Optuna over the methods in the Statlog German Credit dataset (Table 8). The best results were achieved by XGBoost with 84.93% of F1 Score and 70.95% of AUROC.

We obtained the best results using XGBoost with Optuna for the Statlog German Credit dataset,

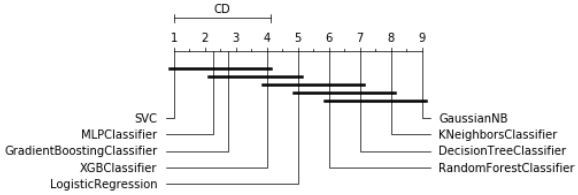

Figure 7: Critical difference diagram over F1 measure of Statlog German Credit Dataset.

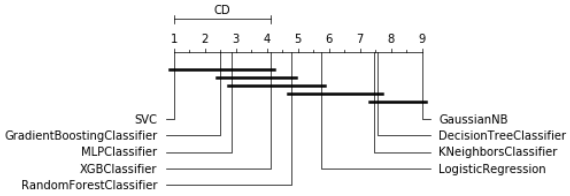

Figure 8: Critical difference diagram over AUROC measure of Statlog German Credit Dataset.

Table 8: Cross Validation with Optuna for Statlog German Credit.

Classifiers	F1	std	AUROC	std
DT	81.37	0.00	60.42	0.00
RandFC	85.31	0.57	64.01	1.32
GradB	80.55	0.00	64.76	0.00
XGB	84.93	0.00	70.95	0.00
LR	81.42	0.00	69.04	0.00
SVC	75.55	0.00	63.09	0.00
kNN	65.00	0.00	59.52	0.00
NN	83.00	1.05	61.57	1.87
NB	72.06	0.00	74.24	0.00

with the following setting parameters: 'booster' = 'gbtree', 'lambda' = 0.0005393794046856518, 'alpha' = 4.896353471497812e-07, 'max_depth' = 6, 'eta' = 3.48108440454574e-07, 'gamma' = 0.004501584677856371, 'grow_policy' = 'loss-guide'.

For the Statlog Australian Credit Approval dataset, we just employed the methods without pre-processing the data. The results are shown in Table 9.

As it can be observed, the worst values of F1 Score and AUROC were obtained by Support Vector Classifier with 6.04% and 50.01%, respectively. The best results were achieved by XGBoost with 84.78% of F1 Score and 86.23% of AUROC, followed by Gradient Boosting (84.53% and 85.97%). When it comes to standard deviation for F1 Score and AUROC, XGBoost resulted in 0%.

Figures 9 and 10 show the Critical Difference Diagram constructed using Nemenyi Test among F1 Scores and AUROCs of Statlog Australian Credit dataset.

Table 9: Cross-validation for Statlog Australian Credit Dataset.

Classifiers	F1	std	AUROC	std
DT	79.09	0.50	80.62	0.31
RandFC	83.63	0.96	85.54	0.78
GradB	84.53	0.00	85.97	0.00
XGB	84.78	0.00	86.23	0.00
LR	84.15	0.00	85.52	0.00
SVC	6.04	0.00	50.01	0.00
kNN	59.79	0.00	66.53	0.00
NN	72.20	2.41	73.42	2.32
NB	74.89	0.00	78.74	0.00

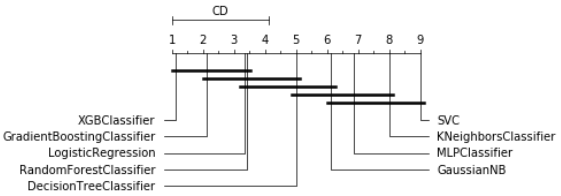

Figure 9: Critical difference diagram over F1 measure of Statlog Australian Credit Dataset.

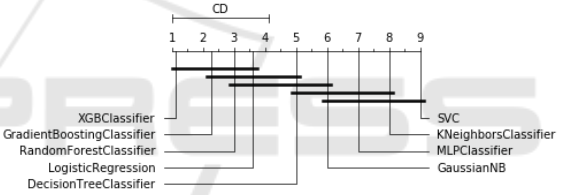

Figure 10: Critical difference diagram over AUROC measure of Statlog Australian Credit Dataset.

As it can be seen in the figures, XGBoost, Gradient Boosting, Logistic Regression, and Random Forest were considered the best, but no statistically significant difference can be observed among them. Thus, the methods can be used with similar efficiency for classifying who is a credible client.

We employed Optuna over the methods in the Statlog Australian Credit dataset (Table 10). The best result was achieved by Gradient Boosting with 85.34% of F1 Score and 86.75% of AUROC. Again, XGBoost obtained similar results.

We obtained the best results using Gradient Boosting with Optuna for the Statlog Australian Credit dataset, with the following setting parameters: 'learning_rate': 0.19027288485989355, 'n_estimators': 214.4696898054894, 'max_depth': 5.367595574055688, 'min_samples_split': 70.98506021007175, 'min_samples_leaf': 1.4109947261432878.

Table 10: Cross-validation with Optuna for Statlog Australian Credit Dataset.

Classifiers	F1	std	AUROC	std
DT	84.67	0.00	85.69	0.00
RandFC	84.26	0.32	85.94	0.27
GradB	85.34	0.36	86.75	0.20
XGB	84.78	0.00	86.23	0.00
LR	84.26	0.00	85.59	0.00
SVC	78.57	0.00	81.68	0.00
kNN	59.80	0.00	65.07	0.00
NN	79.53	0.97	81.96	0.72
NB	74.89	0.00	78.74	0.00

6 EVALUATIONS OF THE METHODS IN OTHER DOMAINS

In this section, we show the results of the methods in domains other than Finance. We employed three other datasets to verify the performance of XGBoost in healthcare and ionosphere domains.

The *Heart Disease* dataset contains information on patient's heart exams, and the complete dataset has 76 attributes. Typically, published experiences refer to the use of a subset with no missing values and 14 numerical attributes, such as client personal data and cardiac test results. We used the dataset from Cleveland database because it is the only one that has been used by Machine Learning researchers. The purpose of using the dataset is to classify who has or does not have a heart disease. It is available at https://archive.ics.uci.edu/ml/datasets/Heart+Disease (Dua and Graff, 2017).

For the Heart Disease dataset, we just employed the predictive methods without preprocessing the data. The results are shown in Table 11.

Table 11: Cross-validation for Heart Disease Dataset.

Classifiers	F1	std	AUROC	std
DT	78.20	0.81	74.79	1.02
RandFC	83.00	1.74	80.47	1.36
GradB	80.24	0.19	76.52	0.21
XGB	81.57	0.00	79.07	0.00
LR	84.27	0.00	80.73	0.00
SVC	71.39	0.00	50.00	0.00
kNN	65.38	0.00	59.39	0.00
NN	83.09	1.55	79.47	1.72
NB	84.03	0.00	81.55	0.00

As it can be observed, the worst value of F1 Score was obtained by kNN with 65.38%. With SVC, we obtained the worst value of AUROC with 50.00%. The best results were achieved by Logistic Regression with 84.27% for F1 Score and 80.73% for AUROC, followed by Naive Bayes (84.03% and 81.55%). When it comes to standard deviation for F1 Score and AUROC, both methods resulted in 0%.

Figures 11 and 12 show the Critical Difference Diagram constructed using Nemenyi Test between F1 Score and AUROC of Heart Disease dataset.

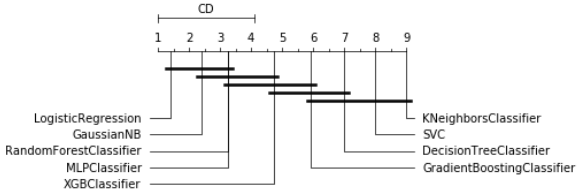

Figure 11: Critical difference diagram over F1 measure of Heart Disease Dataset.

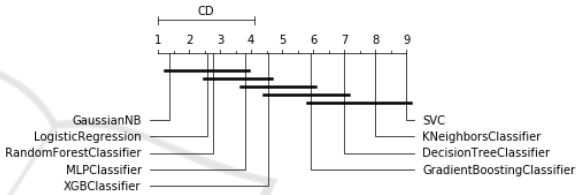

Figure 12: Critical difference diagram over AUROC measure of Heart Disease Dataset.

As it can be seen in Heart Disease dataset, Logistic Regression, Naive Bayes, Multilayer Perceptron, and Random Forest were considered the best, but no statistically significant difference is observed among them. Thus, the above methods can be used with similar efficiency for classifying who has heart disease.

We employed Optuna over the methods in the Heart Disease dataset. With Random Forest, we obtained 86.14% for F1 Score and 82.56% for AUROC. These results are better than those obtained without Optuna.

The *Ionosphere* dataset consists of a phased array of 16 high-frequency antennas with a total transmitted power in the order of 6.4 kilowatts. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those pass through the ionosphere. The purpose of using the dataset is to classify what is returned from radar.

The dataset is multivariate with 351 instances (224 instances are "Good"), 34 attributes (32 real and 2 integer), no missing values and it is available at https://archive.ics.uci.edu/ml/datasets/ionosphere (Dua and Graff, 2017).

For the Ionosphere dataset, we just employed the predictive methods without preprocessing the data.

The results are shown in Table 12.

Table 12: Cross-validation for Ionosphere Dataset.

Classifiers	F1	std	AUROC	std
DT	90.96	0.60	86.27	1.18
RandFC	93.77	0.63	91.11	1.13
GradB	93.80	0.27	89.32	0.36
XGB	93.14	0.00	88.50	0.00
LR	89.14	0.00	79.86	0.00
SVC	94.83	0.00	90.63	0.00
kNN	88.49	0.00	77.41	0.00
NN	94.16	0.37	89.95	0.57
NB	85.19	0.00	82.72	0.00

As it can be observed, the lowest value of F1 Score was obtained by Naive Bayes with 85.19%. With kNN, we obtained the lowest value of AUROC with 77.41%. The best results were achieved by Support Vector Classifier with 94.83% of F1 Score and 90.63% of AUROC, followed by Multilayer Perceptron (94.48% and 89.95%). When it comes to standard deviation for F1 Score and AUROC, SVC method resulted in 0% for both metrics, and Multilayer resulted in 0.37% and 0.57% respectively.

Figures 13 and 14 bring the Critical Difference Diagram constructed using Nemenyi Test between F1 Scores and AUROCs of Ionosphere dataset.

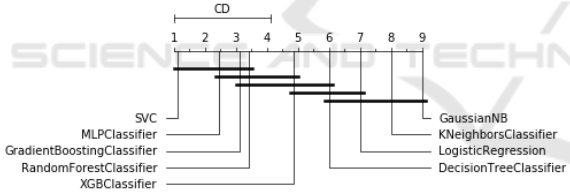

Figure 13: Critical difference diagram over F1 measure of Ionosphere Dataset.

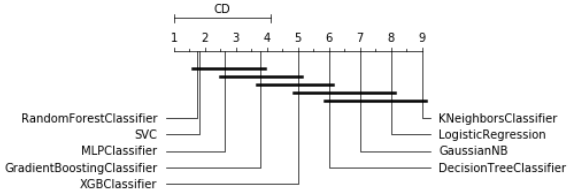

Figure 14: Critical difference diagram over AUROC measure of Ionosphere Dataset.

As it can be seen for Ionosphere dataset, Support Vector Classifier, Random Forest, Multilayer Perceptron, and Gradient Boosting methods are considered the best, but no statistically significant difference can be observed among them. Thus, the above methods can be used with similar efficiency to classify what is returned from radar.

We employed Optuna over methods in the Ionosphere dataset. XGBoost obtained 95.01% of F1 Score and 91.50% of AUROC. These results are better than those obtained with Support Vector Classifier without Optuna.

The *Blood Transfusion Service Center* dataset is intended to evaluate the RFMTC marketing model. To build the model, 748 donors were selected from the donor database. The donor dataset includes the following information: months since last donation, total number of donations, volume of blood donated, months since first donation (Yeh et al., 2009). The purpose of using the dataset is to classify who can donate blood.

For the Transfusion dataset, we applied the SMOTE algorithm, and employed the methods. The results are shown in Table 13.

As it can be observed, the worst values of F1 Score and AUROC were obtained by Multilayer Perceptron with 69.24% and 69.26% respectively. The best results were achieved with kNN with 75.89% of F1 Score and 74.23% of AUROC, followed by Random Forest (75.16% and 75.99%). When it comes to standard deviations for F1 Score and AUROC, kNN resulted in 0% for F1 Score and 0% for AUROC, and Random Forest resulted in 0.97% and 0.57% respectively.

Table 13: Cross-validation for Blood Transfusion Service Center Dataset.

Classifiers	F1	std	AUROC	std
DT	72.93	0.24	74.08	0.40
RandFC	75.16	0.97	75.99	0.57
GradB	73.06	0.10	72.76	0.08
XGB	74.00	0.00	74.32	0.00
LR	71.11	0.00	69.23	0.00
SVC	71.18	0.00	69.71	0.00
kNN	75.89	0.00	74.23	0.00
NN	69.24	0.46	69.26	0.33
NB	71.23	0.00	67.69	0.00

Figures 15 and 16 show the Critical Difference Diagram constructed using Nemenyi Test between F1 Scores and AUROCs of Transfusion Blood dataset.

As it can be seen, kNN, Random Forest, XGBoost, and Gradient Boosting obtained the best results, but no statistically significant difference could be observed among them. Therefore the aforementioned methods can be used with similar efficiency for classifying if a person can donate blood.

kNN with Optuna obtained 76.82% of F1 Score and 76.15% of AUROC, followed by XGBoost (75.86% and 75.96%), these results are better results than obtained with default hyperparameters.

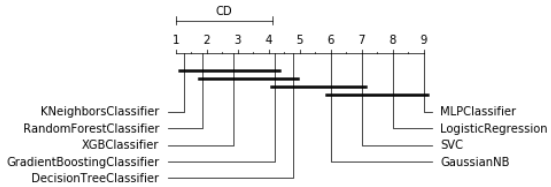

Figure 15: Critical difference diagram over F1 measure of Transfusion Blood Dataset.

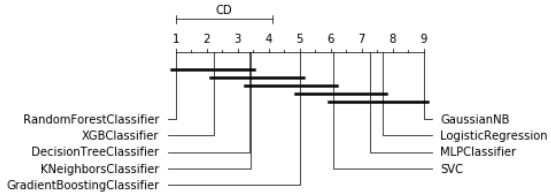

Figure 16: Critical difference diagram over AUROC measure of Transfusion Blood Dataset.

7 CONCLUDING REMARKS

This study has investigated supervised classification methods for finance problems with focus on risk, fraud and credit analysis. Nine supervised predictive methods were employed in five financial datasets. All of them are public. The methods were evaluated using the classification performance metrics *F1 Score* and *AUROC*. The nonparametric Friedman Test was used to infer hypotheses and the Nemeyni Test to validate it with Critical Difference Diagram. In the finance domain, we obtained the best results with the decision tree family of classification methods. XGBoost regularly showed good results in the evaluations.

We experimented the methods in other problem domains such as health care and ionosphere, where XGBoost also obtained good results, but not systematically better than Logistic Regression, Naive Bayes, Multilayer Perceptron, Random Forest, Support Vector Classifier, and Gradient Boosting.

When we applied Optuna in both domains, we achieve better results in all evaluations, and XGBoost was the best method again for the finance domain. So, we believe that one of the reasons is the setting of hyperparameters required in the dataset. However, this overfitting can be misleading, perhaps in production systems, we have to consider the concept drift for retraining the method.

Nielsen (Nielsen, 2016) explains that there are some reasons for the good performance of XGBoost. XGBoost can be seen as a Newton's method of numerical optimization, using a higher-order approximation at each iteration, being capable of learning "better" tree structures. Second, XGBoost provides clever penalization of individual trees, turning it to be more adaptive than other Boosting methods, because it determines the appropriate number of terminal nodes, which might vary among trees. Finally, XGBoost is a highly adaptive method, which carefully takes the bias-variance trade-off into account in nearly every aspect of the learning process.

Other non-tree methods, such as Naive Bayes, Support Vector Classifier and k-Nearest Neighbors algorithms have shown performance worse than the decision tree classification methods. The analysis indicates that the non-tree methods are not the recommended ones for the finance problems we investigated.

Based on the conducted evaluations, we conclude that XBGoost is the recommended machine learning classification method to be overcome when proposing new methods for problems of analyzing risk, fraud, and credit.

REFERENCES

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631. ACM.

Bache, K. and Lichman, M. (2013). Uci machine learning repository [http://archive. ics. uci. edu/ml]. irvine, ca: University of california. *School of information and computer science*, 28.

Bhagat, R. C. and Patil, S. S. (2015). Enhanced smote algorithm for classification of imbalanced big-data using random forest. In *2015 IEEE International Advance Computing Conference (IACC)*, pages 403–408. IEEE.

Bouazza, I., Ameur, F., et al. (2018). Datamining for fraud detecting, state of the art. In *International Conference on Advanced Intelligent Systems for Sustainable Development*, pages 205–219. Springer.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM.

Cunningham, P. and Delany, S. J. (2007). k-nearest neighbour classifiers. *Multiple Classifier Systems*, 34(8):1–17.

Dal Pozzolo, A., Caelen, O., Johnson, R. A., and Bontempi, G. (2015). Calibrating probability with undersampling for unbalanced classification. In *2015 IEEE Symposium Series on Computational Intelligence*, pages 159–166. IEEE.

Damodaran, A. (1996). *Corporate finance*. Wiley.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

García, S., Fernández, A., Luengo, J., and Herrera, F. (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences*, 180(10):2044–2064.

Hofmann, H. (1994). Statlog (german credit data) data set. *UCI Repository of Machine Learning Databases*.

Islam, M. J., Wu, Q. J., Ahmadi, M., and Sid-Ahmed, M. A. (2007). Investigating the performance of naive-bayes classifiers and k-nearest neighbor classifiers. In *2007 International Conference on Convergence Information Technology (ICCIT 2007)*, pages 1541–1546. IEEE.

Lavanya, D. and Rani, K. U. (2011). Performance evaluation of decision tree classifiers on medical datasets. *International Journal of Computer Applications*, 26(4):1–4.

Liaw, A., Wiener, M., et al. (2002). Classification and regression by randomforest. *R news*, 2(3):18–22.

Lin, W.-Y., Hu, Y.-H., and Tsai, C.-F. (2011). Machine learning in financial crisis prediction: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):421–436.

Liu, X.-Y., Wu, J., and Zhou, Z.-H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550.

Moro, S., Cortez, P., and Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31.

Natekin, A. and Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7:21.

Ng, A. Y. and Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in neural information processing systems*, pages 841–848.

Nielsen, D. (2016). Tree boosting with xgboost-why does xgboost win "every" machine learning competition? Master's thesis, NTNU.

Pearl, J., Glymour, M., and Jewell, N. P. (2016). *Causal inference in statistics: A primer*. John Wiley & Sons.

Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, 4(2):1883.

Quinlan, J. R. (1987). Simplifying decision trees. *International journal of man-machine studies*, 27(3):221–234.

Ridgeway, G. (1999). The state of boosting. *Computing Science and Statistics*, pages 172–181.

Rish, I. et al. (2001). An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46.

Sinayobye, J. O., Kiwanuka, F., and Kyanda, S. K. (2018). A state-of-the-art review of machine learning techniques for fraud detection research. In *2018 IEEE/ACM Symposium on Software Engineering in Africa (SEiA)*, pages 11–19. IEEE.

Suykens, J. A. and Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300.

Wang, G., Hao, J., Ma, J., and Jiang, H. (2011). A comparative assessment of ensemble learning for credit scoring. *Expert systems with applications*, 38(1):223–230.

Yeh, I.-C., Yang, K.-J., and Ting, T.-M. (2009). Knowledge discovery on rfm model using bernoulli sequence. *Expert Systems with Applications*, 36(3):5866–5871.

Yu, Q., Miche, Y., Séverin, E., and Lendasse, A. (2014). Bankruptcy prediction using extreme learning machine and financial expertise. *Neurocomputing*, 128:296–302.

Zareapoor, M. and Shamsolmoali, P. (2015). Application of credit card fraud detection: Based on bagging ensemble classifier. *Procedia computer science*, 48(2015):679–685.