EAI Endorsed Transactions
on Internet of Things

Research Article EAI.EU

Machine Learning Classifiers for Credit Risk Analysis

Sudiksha[1], Preethi Nanjundan [2, *] Jossy P George[3]

[1,2,3]Department of Data Science, Christ University, Pune Lavasa Campus, India

Abstract

The modern world is a place of global commerce. Since globalization became popular, entrepreneurs of small and medium enterprises to large ones have looked up to banks, which have existed in various forms since antiquity, as their pillars of support. The risk of granting loans in various forms has significantly increased as a consequence of this, the businesses face financing difficulties. Credit Risk Analysis is a major aspect of approving the loan application that is done by analyzing different types of data. The goal is to minimize the risk of approving the loan for the Individuals or businesses who might not pay back on time. This research paper addresses this challenge by applying various machine learning classifiers to the German credit risk dataset. By evaluating and comparing the accuracy of these models to identify the most effective classifier for credit risk analysis. Furthermore, it proposes a contributory approach that combines the strengths of multiple classifiers to enhance the decision-making process for loan approvals. By leveraging ensemble learning techniques, such as the Voting Ensemble model, the aim is to improve the accuracy and reliability of credit risk analysis. Additionally, it explores tailored feature engineering techniques that focus on selecting and engineering informative features specific to credit risk analysis.

Keywords: Credit risk, Machine learning, Naive Bayes classifier, Decision tree, K-Nearest Neighbour

Received on 17 December 2023, accepted on 05 March 2024, published on 12 March 2024

doi: 10.4108/eetiot.5376

*Corresponding author. Email: preethinanjundan@christuniversity.in

1. Introduction

Many financial incidents around the world have proved the major need for credit risk analysis. Credit can be defined as a contract-based agreement of granting money, credit card, or any other form of money between a lender and a borrower, which needs to be paid back to the lender, in most of cases with some amount of interest rate. Credit Risk is the potential risk involved in lending a loan to an individual or a firm, that may not pay back the loan. Credit Risk that showcases the potential loss and loan defaults contribute to the majority of the loss faced by the banking Industry.[1] Commercial banks often tend to take low risk in providing credit loans to small and medium businesses due to their inability to withstand macroeconomic factors such as a recession. Over the period of time, the banking industry has understood that in order to have healthy and fair competition it is necessary to have customer retention and fraud prevention as well as detection strategies [2]. Machine Learning Classification Algorithms can help analyze the data and provide insights on important decisions, whether to approve loans for a particular candidate or not, thus reducing the chances of huge losses.

2. Related work

Financial organizations must perform credit risk analysis in order to manage their loan portfolios efficiently and evaluate the creditworthiness of prospective borrowers. Machine learning classifiers have been increasingly common as a technique for credit risk analysis in recent years due to their capacity to process massive volumes of data and provide precise predictions. The use of machine learning classifiers for German credit risk analysis is examined in this literature review. Support vector machines (SVMs) were employed by [3] in one of the initial research projects in this area to categorize credit applications as either good or bad. SVMs performed better than conventional linear classifiers, according to their findings, suggesting that nonlinear approaches would be more appropriate for credit risk assessments.

Since then, numerous research has been carried out in Germany utilizing different machine learning classifiers to analyze credit risk. For instance, it [4] predicted default risk for German small and medium-sized firms using logistic regression, decision trees, and random forests (SMEs). According to their findings, random forests outperformed the other classifiers and had the highest predictive power. Similar to this,[5] predicted credit default risk for German borrowers using decision trees, random forests, and gradient boosting machines. Their findings suggested that ensemble approaches may be more useful for credit risk analysis as random forests and gradient-boosting machines have stronger predictive power than decision trees.

In a separate study,[6] assessed the effectiveness of various machine learning classifiers for analyzing German credit risk, including Support Vector Machines, Logistic regression, Decision trees, Random forests, and Neural networks. The maximum accuracy as well as recall was obtained in random forests, suggesting that it would be the most effective classifier for this task. Overall, the research points to machine learning classifiers as useful tools for analyzing German credit risk, particularly non-linear techniques like SVMs and ensemble techniques like random forests and gradient boosting machines. However, the particular dataset and issue at hand may influence the classifier that is used.

3. Methodology

3.1 Dataset Used

In much of the Research Work for Credit Risk Analysis, German Credit Risk data is used to compare and understand the performance of various machine learning algorithms. This Dataset is obtained from the UCI machine repository, online(http://archive.ics.uci.edu/ml/). There are two labels in the dataset which are “good” and “bad” showcasing the worthiness of Borrowers. The dataset was split in 80:20, train and test ratio. Feature Selection was done by calculating the correlation between all the independent variables and independent variables.

Table 1. Summary of Dataset Used

Dataset Name	Features	Instances	Labels
German	20	1000	2

Credit risk analysis can be approached in various ways, some of the machine learning approaches are as follows:

3.2 K Nearest Neighbour

A supervised machine learning approach known as K Nearest Neighbor (KNN) is utilized in classification and regression analysis. This non-parametric technique bases its predictions on how closely fresh data points are spaced from previous data points in the training set.

Finding the k no. of closest data points to a new data point in the feature space and using their average or majority class as the forecast for the new data point is the fundamental tenet of KNN. A hyperparameter, k, needs to be selected based on the data and the current situation. Usually, k is calculated by taking under the root of the total number of observations. There are various ways to calculate the nearest distance point such as Manhattan Distance, Euclidean Distance which is mostly used and calculated by the formula Euclidean distance.

$$d = \sqrt{[\,(x22 - x11)^2 + (y22 - y11)^2]} \quad (1)$$

3.3 Decision Tree

A supervised machine learning algorithm for classification tasks is a decision tree classifier. Each internal node serves as a feature or attribute, each branch serves as a decision rule, and each leaf node serves as a class label in this tree-like model. The optimal characteristic for dividing the data into subgroups is first chosen via the decision tree algorithm. It selects the attribute that optimizes the division of classes into separate subsets. The decision tree method is capable of handling non-linear correlations between features as well as categorical and continuous features. The decision tree method chooses the optimal feature to split on at each node based on information gain and impurity index. The reduction in entropy or impurity brought about by splitting the data on a certain feature is measured as the gain of Information. It is employed to reveal the feature that offers the greatest degree of class separation in the generated subsets.

On the other side, the impurity index is a measurement of the homogeneity or impurity of the class labels at a specific node.

$$IG(s,a) = H(s) - \sum (|s(v)| / |s|) * H(s(v)) \quad (2)$$

$$\text{Entropy:}H(s) =-\sum p(i) * \log2(p(i)) \quad (3)$$

$$\text{Gini Index:}G(s) = 1 - \sum p(i)\hat{}2 \quad (4)$$

3.4 Random Forest Classifier

A well-liked machine learning approach for classification tasks is a random forest classifier. Several decision trees are combined in this ensemble learning technique to produce predictions. A random subset of the features and various subsets of the training data are used to construct a huge number of decision trees in a random forest classifier. To avoid overfitting and guarantee that the model generalizes effectively to new data, each tree is trained on a subset of the characteristics and data. The random forest classifier aggregates all of the decision trees' predictions during prediction to produce a final prediction. The majority vote of all the decision trees serves as the foundation for the final projection, where y is the predicted target variable, x is the input features, and f1(x), f2(x), ..., fk(x) are the predictions of k decision trees

$$y = f1(x) + f2(x) + ... + fk(x) \quad (5)$$

3.5 Naive Bayes Classifier

It is a probability-based model that estimates the likelihood that a given sample will belong to a specific class using Bayes' theorem. The word "naive" in naive Bayes refers to the basic premise that each feature utilized to produce a prediction is independent of every other feature. In other words, whether one feature is present or not has no bearing on whether any other trait is present or not. Based on the values of each feature, the algorithm determines the likelihood that a sample belongs to each class. The predicted class is then determined to be the one with the highest probability. Naive Bayes classifiers have a number of benefits, including simplicity, effectiveness, and the capacity to handle sizable datasets with numerous features. They are extensively used in many different industries, such as sentiment analysis, spam filtering, natural language processing, and image classification. where P(y|x1, x2, ..., xn) is the probability of class y given the input features x1, x2, ..., xn. P(y) is the prior probability of class y. P(xi|y) is the conditional probability of feature xi given class y. P(x1, x2, ..., xn) is the marginal probability of the input features.

$$P(y|x1, x2, ..., xn) = P(y) * P(x1|y) * P(x2|y) * ... * P(xn|y) / P(x1, x2, ..., xn) \quad (6)$$

3.6 Support Vector Classifier

Support Vector Classifier, a well-liked machine learning technique used for classification problems, goes by the abbreviation SVC. It is a kind of supervised learning algorithm that may be applied to multi-class and binary classification. The SVC algorithm operates by identifying the hyperplane that best divides the various input data classes. A decision boundary called the hyperplane maximizes the distance between nearby data points of various types. Finding the hyperplane with the biggest margin is the goal of the SVC method since it is more likely to generalize to novel, untested data. SVC can employ a method known as the kernel trick to transform the input data into a higher-dimensional space where a linear hyperplane can be utilized to separate the classes when a linear hyperplane is unable to do so effectively. SVC has a number of benefits, including its capacity for managing high-dimensional data and its resistance to outliers. It is frequently used in many different industries, including as bioinformatics, picture recognition, and text categorization. The design function in a Support Vector Classifier (SVC) can be written as:

$$f(x) = sign(w^T x + b] \quad (7)$$

where x represents input data, w is associated with weight vector, b is bias term, and sign() is the sign function that returns +1 or -1 depending on the sign of the argument.

3.7 Multi-Layer Perceptron

A Multi-Layer Perceptron (MLP) classifier is a subtype of artificial neural network that is majorly used in supervised machine learning for classification tasks. In an MLP classifier, the input layer consists of a set of input features, and the output layer consists of a set of output nodes, each corresponding to a specific class label. The intermediate layers, known as hidden layers, consist of one or more layers of nodes, each of which performs a non-linear transformation of the input data.[7] During training, the MLP classifier modifies the weights between the network's nodes to learn how to categorize the input data. A set of labeled training data is often fed into the network during the training process, and then the weights are adjusted to reduce the difference between the predicted outputs and actual values. This procedure is repeated until the network reaches an acceptable level of accuracy.

By feeding the input features into the input layer and then passing them through the network to produce a predicted class label, the MLP classifier may be used to categorize new data once it has been trained,where y is the predicted target variable, x is the input features, w1, and w2 are the weights of the neurons in the first and second layers, b1 and b2 are the biases of the neurons, and f is the activation function.

$$y = f(w2f(w1x + b1) + b2) \quad (8)$$

3.8 Ensemble The Classifiers

A group of base classifiers that have undergone independent training compose an ensemble of classifiers. Base classifiers decide whether to ensemble a classifier. Voting is used to reach a consensus on how new and unusual cases should be classified. To ensemble the classifier, the basic classifiers are joined in a way that makes the composite classifier perform better than the single classifier alone.

3.1.1 Bagging

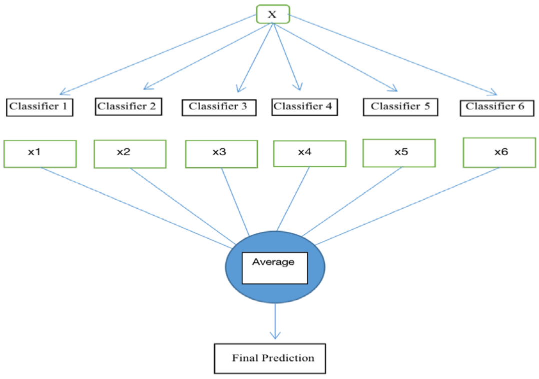
An ensemble technique called bagging, or bootstrap aggregating, entails creating numerous subsets of the training data and training a different model on each group. It was introduced in 1996 [8]. Combining all of the models' projections yields the ultimate conclusion. When a model's variance is high and it tends to overfit the data, bagging may be used.

3.1.2 Boosting

Boosting is an assembly approach in which several weak models are combined to form a strong model.[9] Each weak model in boosting is trained on a portion of the training data and concentrates on the samples that the preceding model incorrectly identified. The predictions of all the weak models are combined to get the final prediction.

3.1.3 Voting

On the other hand, voting is an assembling strategy that combines the outcomes of various models that were trained on the same dataset. Each model is given a vote during voting, and the classification that receives the most votes ultimately determines the prediction.



4. Architect Design

The architecture design showcases how input data can be passed through various classifiers, in this design six, each classifier will produce an output by using the Voting method, which chooses the highest frequency of a class and gives the final prediction.

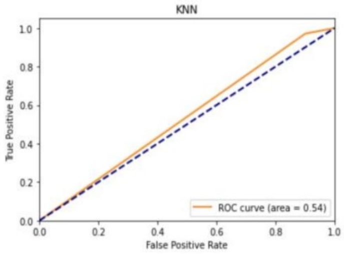
It is recommended to use odd numbers of classifiers in ensemble learning to avoid any case of an even score.

5. Result and Discussions

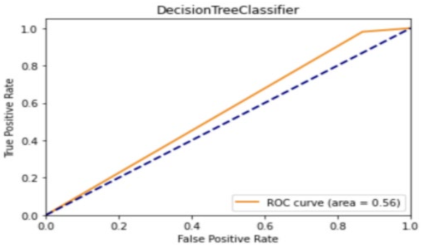
Comparison of the different types of accuracy scores of different algorithms using the German Credit Risk dataset.

Classifier	Accuracy	Precision	Recall	F1 Score
1. KNN	0.61	0.68	0.8	0.57
2. Decision Tree	1.0	1.0	1.0	1.0
3. Random Forest Classifier	1.0	1.0	1.0	1.0
4. Naive Bayes Classifier	1.0	1.0	1.0	1.0
5. Support Vector Classifier	0.71	0.70	1.0	1.0
6. Multi-Layer Perceptron	0.71	0.70	0.88	0.57

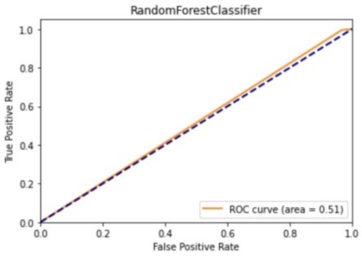
5.1 Graphical Analysis



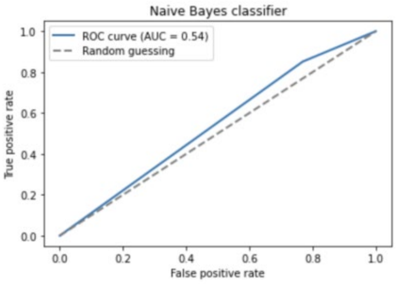
(a)



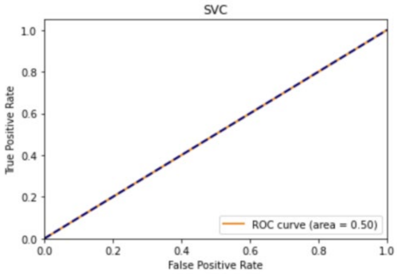
(b)



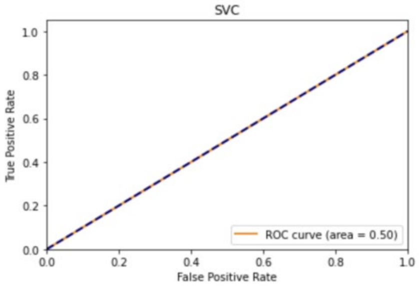
(c)



(d)



(e)



(f)

The figures above represent ROC curves, where (a) represents KNN, (b) represents Decision Tree,(c) shows Random Forest,(d) for Naive Bayes,(e) for the SVC, and (f) for MLP. Decision Tree has the high ROC of 0.56 followed by KNN and Naive Bayes.

Decision Tree, Random Forest Classifier, and Naive Bayes Classifier have the highest accuracy of 100% which showcases that these algorithms are best suited for such data as well as an F1 Score of 1, followed by Support vector classifier and multi-layer perceptron out of all the approaches. The above-given architect design shows how all the classifiers can be used using the Voting Ensemble modeling to get the best results and decrease the risk of approving a loan of a probable default application.

6. Conclusion

The research paper examined the use of machine learning (ML) algorithms for credit risk analysis and found that Decision Tree, Random Forest Classifier, and Naive Bayes Classifier were the most accurate algorithms. The study demonstrates the potential of ML to improve credit risk analysis and decision-making, which is critical for financial institutions. The results highlight the importance of selecting the appropriate algorithm for a specific task and dataset, as the performance of different algorithms can vary significantly. Different datasets and problem domains might require different algorithms or combinations of algorithms for optimal performance. Overall, this research contributes to the growing body of literature on the application of ML in finance and provides insights for practitioners and researchers seeking to develop more accurate and efficient credit risk models. Further research could explore the use of other ML techniques and datasets to confirm and extend these findings. Credit risk analysis involves various external factors that can influence loan repayments, such as economic conditions, industry-specific factors, and changes in regulations. These factors may not have been explicitly considered or incorporated into the analysis, which could limit the comprehensive understanding and assessment of credit risk.

References

[1] Rob Gerritsen, “Loan Risks: A Data Mining Case Study

[2] Frawley, W. J., Piatetsky-Shapiro, G., and Matheus, C. J. (1992). Knowledge discovery in databases: An overview. AI Magazine, 13(3):57.

[3] Schölkopf, B., Smola, A. J., Williamson, R. C., & Bartlett, P. L. (1999). New support vector algorithms. Neural Computation, 12(5), 1207-1245.

[4] Khandani, A. E., Kim, A. J., & Lo, A. W. (2010). Consumer credit-risk models via machine-learning algorithms. Journal of Banking & Finance, 34(11), 2767-2787.

[5] Nguyen, T. T., Tran, H. T. M., & Pham, H. T. (2018). Credit risk prediction using machine learning techniques: A comparison of three methods. Expert Systems with Applications, 114, 125-135.

[6] Zhao, J., Yan, Y., Li, Q., Li, S., & Li, Y. (2019). A comparative study of machine learning algorithms for credit risk assessment. Journal of Ambient Intelligence and Humanized Computing, 10(1), 275-284.

[7] Chen, M.CHuang, S.H, ‘Credit scoring and rejected instances reassigning through evolutionary computation techniques’, Expert Systems with Applications, Vol. 24(4), pp. 433–441, 2003.

[8] Dietterich, T.G., ‘Experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization’, Machine Learning, Vol. 40, pp.139–157, 2000.

[9] Pandey, T.N., Jagadev, A.K., Choudhury, D. and Dehuri, S., ‘Machine learning-based classifiers ensemble for credit risk assessment’, Int. J. Electronic Finance, Vol. 7(3/4), pp.227–249, 2013.