Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Leveraging Machine Learning to Predict Credit Card Customer Segmentation

Ridha Maya Faza Lubis[1], Jen-Peng Huang[2]

Abstract

Explained in this paper is how data mining provides a way to work on distributed Machine Learning (ML) systems, which are already often used in data mining operations. This paper examines eight strategies applied in cases of Taiwanese customer default. The eight methods. Eight distinct classifications are evaluated for prediction accuracy: Random Forest, Naïve Bayesian Classifier, K-Nearest Neighbour, Support Vector Machine, Neural Net, Decision Tree, Logistic Regression, and Deep Learning. Utilizing this method raises the possibility of many consumer loans and is one way to evaluate risk management outcomes, such as the exact probability of credit card loan default. Large financial losses for the borrower could result from default due to the method's overall effectiveness and efficiency. 30,000 Taiwanese clients with twenty-five qualities, all of whom have full payment histories, are the subject of this study's payment data analysis. Four approaches (weighting, SMOTE, Imbalance, and Downsampling) were used to balance the data in this study. We shall contrast four approaches and outline eight distinct approaches in this study.

Keywords: *Default of Credit Card Payment, Machine Learning, Debt, Balance Data, Credit History Data, Taiwan Banks.*

Introduction

Credit card payment is a commonly utilized method for settling shopping expenses. An advantage of having a card as a client is that it guarantees payment for the expenses incurred by the client while purchasing services and items [1]. Numerous banks provide credit card payment services to their consumers, typically offering exclusive promotions and discounts for credit card transactions [2]. [3] The bank will get benefits and increase its customer base by providing promotional discounts to credit card holders. Providing incentives to credit card holders can capture the interest of young individuals in Taiwan who are the intended customers [4]. Historical data indicates that the low income of young credit card holders led to a rise in unpaid payments, increasing credit card debt. This can lead to issues in Taiwan, such as the rising prevalence of suicides and other illicit activities undertaken to settle credit card debts. The problem resulting from several clients encountering payment failure can lead to a decrease in consumer confidence. Recent data indicates that credit card issuing banks are experiencing a crisis as the accumulation of loans continues to rise [5]. Hence, our study, based on extensive research and analysis of multiple prior studies on payment failure prediction, will serve as a valuable resource for forecasting credit card payment defaults in the future. [6] The researcher's study examines instances of payment failures among credit card users in Taiwan. The study also evaluates the accuracy of probability predictions using six data mining techniques: K-Nearest Neighbor classifiers (KNN), Logistic Regression (LR), Discriminant Analysis (DA), Naïve Bayes classifier (NB), Artificial Neural Networks (ANNs), and Classification Trees (CTs). This research examines six mining engineering approaches and highlights subtle variations among the six artificial neural network methods. The findings demonstrate that the artificial neural network achieves more precise classification compared to the other five methods. The artificial neural network demonstrated superior performance in accurately forecasting the chance of default, as evidenced by its high R2 value of 0.9647 (near to 1), low regression intercept of 0.0145 (almost to 0), and strong regression coefficient of 0.9971 (close to 1). The predictive default probability supplied by an Artificial Neural Network (ANN) is the sole representation of probability that may be utilized. From a risk control standpoint, determining the likelihood of default is more significant than categorizing clients into binary outcomes of hazardous and non-risky. Hence, it is advisable to employ artificial neural networks instead of alternative data mining techniques, like logistic regression, to tackle these client scores. [7] The paper utilizes seven methods, specifically: K-Nearest

[1] Department of Business and Management, Southern Taiwan University of Science and Technology, No. 1 號, Nantai St, Yongkang District, Tainan City, 71005, Email: db01g208@stust.edu.tw, Email: ridhamayafazalubis@gmail.com.
[2] Department of Information Management, Southern Taiwan University of Science and Technology, No. 1 號, Nantai St, Yongkang District, Tainan City, 71005, Email: jehuang@stust.edu.tw.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Neighbor classifiers (KNN), Logistic Regression (LR), Naïve Bayes classifier (NB), Random Forest (RF), Support Vector Clustering (SVC), and Linear Support Vector Clustering (SVC). The analysis examines payment failure data from 30,000 clients in Taiwan, including twenty-three features. The findings indicate that only a small number of the factors employed are sufficient for analyzing the characteristics of default in lending decisions.

The results offer valuable feedback to credit assessors, lending institutions, and business analysts for a comprehensive study. In addition, they emphasize the significance of employing cautious lending methods to gain a deeper understanding of the behavior of credit card borrowers, along with certain accounting, historical, and demographic attributes. The majority of customers in developed countries consistently maintain personal credit through the use of credit cards. This study aims to identify the essential traits that enable cardholders to make reasonable decisions to optimize their satisfaction. Nevertheless, certain credit card clients continue to demonstrate a tendency to misuse their credit cards and occasionally fall victim to manipulation by credit institutions. The primary significance of this work is in the incorporation of crucial client elements, such as financial data, outstanding payments, and other operational attributes, which highlight the need to assess their reliability. A variety of machine learning algorithms were utilized to analyze the credit portfolio from April to September 2005. This portfolio consisted of consumer credit card data, and the purpose was to evaluate the creditworthiness of these clients. The precision of the generated model varies between 70% and 82.6%. Thus, its precision might be deemed satisfactory and consequently employed by financial institutions or credit card businesses to classify prospective clients according to their financial stability throughout the approval procedure, while utilizing less data instead of dealing with numerous customers. Financial, population-related, and credit-related data.

Specifically, there were 30,000 instances of credit card cases, with 23,364 cases showing no signs of default (meaning they were paid on time or with a minor delay), and a total of 6,636 cases were classified as default conditions based on customer circumstances in September 2005. [8] In his comparative study of other algorithms, specifically Random Forest (RF), AdaBoost, and the bagging algorithm, the bagging algorithm yielded a result of 0.72. [9] The research focuses on loan data collected between 2012 and 2014 from the Korea Student Aid Foundation. The objective of this study is to create a prediction model for clients who are unable to make credit card payments. The research employs both a logistic regression model and the Cox proportional hazard model to develop a risk prediction model. [10] This study explores the utilization of three methodologies, specifically decision trees, neural networks, and logistic regression, to address and resolve payment issues. Additionally, it investigates the effectiveness of combining these techniques with ensemble models.

Literature Review

Since 1997, there has been a significant amount of study conducted on the topic of credit card payment failures. Credit cards play a crucial role in supporting banking operations, particularly by offering discounts and the convenience of deferred payment. Based on prior research findings, numerous machine-learning techniques are employed to identify instances of payment failures. In 1997, a researcher [11] was undertaking research on credit cards in the United Kingdom (UK) using formal statistical methods to classify class divisions into 'good' or 'poor' categories. This tool can evaluate the credit expansion of consumers who default on their payments. [12] In his study, the researcher employed conventional statistical techniques, specifically logistic regression and discriminant analysis, to assess the creditworthiness of clients. Additionally, machine learning methods such as neural networks, decision trees, and support vector machines were successfully employed to classify clients as either eligible borrowers or potential loan defaulters. Among the three methods employed to assess decision trees and evaluate their capabilities, they are readily comprehensible. This study employs eight methodologies that have been previously established. However, this research distinguishes itself by utilizing a combination of these methods, namely Decision Tree (DT), Naïve Bayes Classifier (NB), Logistic Regression (LR), K-Nearest Neighbor Classifier (KNN), Random Forest (RF), Support Vector Machine (SVM), Neural Net (NN), and Deep Learning (D.L.). There are four approaches for balancing data: downsampling, imbalance, Synthetic Minority Oversampling Technique (SMOTE), and weighting.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Downsampling

Downsampling is a signal processing technique that includes reducing the sample rate of a signal. This can be achieved for several goals, like as reducing the amount of data to be processed or transmitted, or decreasing the resolution of a signal. Downsampling in digital signal processing is the process of selecting and removing samples from a signal by keeping only every Nth sample and discarding the rest. For example, if you have a signal that has been sampled at a frequency of 1000 Hz and you reduce the sampling rate by a factor of 2, you would keep every other sample, thereby effectively reducing the sampling rate to 500 Hz [13].

Aliasing is a phenomenon that can happen during downsampling, in which the high-frequency components of a signal are mistakenly portrayed as lower frequencies. In order to avoid aliasing, it is common practice to first apply a low-pass filter to the signal before downsampling. This filter removes high-frequency components that could cause aliasing when the signal is downsampled [14].

Downsampling is a frequently used approach in image processing to decrease the resolution of an image, hence reducing its size and potentially saving storage space or improving processing speed. Downsampling is a useful technique for reducing the size or resolution of signals or images while maintaining crucial data, as long as potential aliasing effects are considered [15].

Imbalances

An "imbalance" refers to the lack of equilibrium or proportionality in a system or situation [16]. It suggests that the components inside the system are spread unevenly or not in the correct proportions [17]. This term is relevant in various circumstances, encompassing [18]:

Physical balance, in the context of the physical world, pertains to the condition of stability or equilibrium where an object or system is inclined or leaning excessively in one direction.

In the realm of chemistry, the phrase "chemical balance" denotes a scenario characterized by an asymmetrical dispersion of substances or an inequitable ratio of reactants and products inside a chemical reaction.

Emotional Balance: In the realm of psychology, an imbalance can be defined as an inequitable allocation of emotions or an undesirable predominance of negative emotions to positive ones.

Economic equilibrium refers to a state in economics when there is a balance in the distribution of wealth, absence of trade deficits, and equal growth throughout various areas or industries.

Social balance, within the realm of sociology, refers to inequalities in power, privilege, or access to resources within a specific society.

Work-Life Balance: Imbalance in the context of personal growth may suggest an inequitable distribution of time and energy between work, personal life, and other endeavors.

Identifying and addressing inequalities is often crucial for maintaining stability, coherence, and efficiency in different systems, such as physical, chemical, social, or human systems.

Synthetic Minority Oversampling Technique (SMOTE)

SMOTE stands for Synthetic Minority Over-sampling Technique. Oversampling is a technique used in machine learning to address the problem of class imbalance. In many real-world datasets, especially in binary classification problems, there is frequently a significant imbalance between the representation of the minority class (with fewer occurrences) and the majority class (with more instances) [19]. Class imbalance

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

can lead to the creation of biased models that have inferior performance in predicting the minority class [20].

SMOTE functions by generating synthetic instances of the minority class to balance the class distribution. This is accomplished by creating new instances of the minority class that closely match the existing ones. Consequently, the dataset is enlarged, providing a greater amount of precise training data for the minority class. This improves the effectiveness of machine learning models, particularly in scenarios with imbalanced class distributions [21].

Weighting

Weighting refers to the process of assigning different levels of importance or influence to individual objects inside a system or dataset. This approach is widely used in the domains of statistics, data analysis, and machine learning [22].
Weighting, in the context of statistical analysis, refers to the act of allocating higher priority to certain data points or observations based on their level of significance or reliability. In the domain of survey research, weighting is commonly used to adjust the results to improve the representativeness of the target population. This is accomplished by giving higher priority to underrepresented groups [23].
Weighting is a machine-learning technique that enables the prioritization of some samples or features over others during the training process. This can result in improved performance of the model on particular tasks or subsets of data [24]. Weighting allows for a more sophisticated and precise analysis or modeling by acknowledging and including the relative importance of multiple factors [25].

Methodology

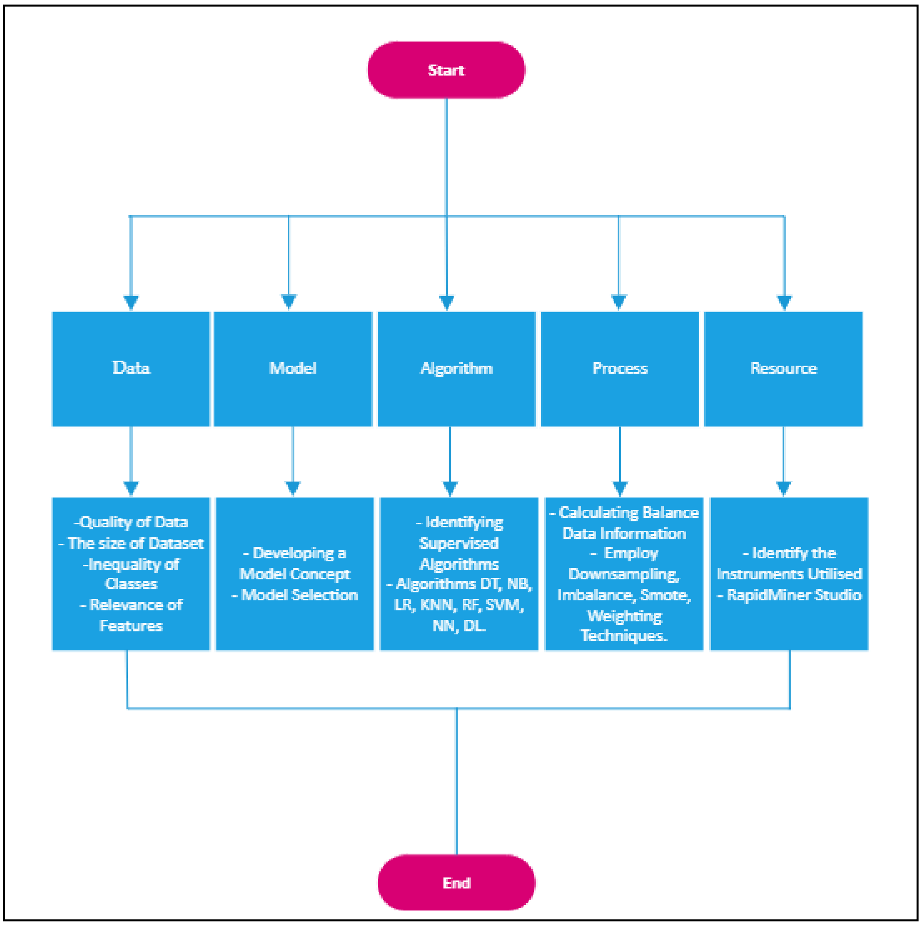


Figure 1. Purpose Methodology

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Data

The dataset consists of 30,000 data points, with 24 variables as indicated in table 1. Data quality selection is employed to identify and address issues such as unclean, noisy, or erroneous transaction data. The purpose of this process is to determine the presence of fraud and assess the informativeness of the data in detecting fraudulent activities. The period. The data gathering methodology employs preprocessing techniques to cleanse data and mitigate noise, as well as employ data augmentation or synthetic data synthesis techniques.

Model

The choice of a model that is either excessively intricate or overly simplistic, and the failure to select a model that aligns with the specific attributes of the data. Overfitting and underfitting refer to situations when a model is excessively tailored to the training data or inadequately adapted to the validation data, respectively. The model selection methodology involves employing Machine Learning techniques and utilizing cross-validation to ensure the model's generalisability and performance.

Algorithm

The efficiency of an algorithm determines its performance, with an inefficient method being one that takes a significant amount of time to compute. The method's lack of flexibility to changes in transaction patterns necessitates the implementation of a more efficient algorithm that employs an approximation approach to expedite computation. Additionally, it is crucial to design or adopt a more robust and flexible monetary algorithm. This work employs eight supervised algorithms, specifically the Decision Tree (DT) algorithm, Naïve Bayesian Classifier (NB), Logistic Regression (LR), K-Nearest Neighbor Classifier (KNN), Random Forest (RF), Support Vector Machine (SVM), Neural Net (NN), and Deep Learning (DL).

Process

The research process is hindered by ineffective or inconsistent data pipelines, inadequate process automation, and a lack of automation in the training and detection phase. Creating streamlined and automated data pipelines by utilizing technologies like AutoML approaches to automate the process of selecting the most suitable model and fine-tuning its hyperparameters.

Resources

The resources utilized in this study encompass limited computer resources, such as GPU or CPU, together with the time-constrained process of running data for model training. The approach employed for resource investigation involves identifying solutions that facilitate Machine Learning applications using cloud computing or platforms that offer access to extensive computer resources. Utilizing parallelization or distributed computing methods to expedite data processing.

DownSampling

Downsampling, as discussed in this paper, refers to the act of reducing the spatial resolution of an image while retaining its two-dimensional representation. It is a basic image operation employed to decrease the storage or transmission demands of images by reducing the number of pixels while preserving the overall structure and appearance. The study assesses various downsampling approaches, such as binomial filters and biorthogonal wavelet filters, to identify the most efficient methods for reducing image size while minimizing data loss and preserving image quality [41]. Downsampling, as used in this research paper, is the technique used to decrease the resolution of a 2D input image while retaining important information. The main goal of downsampling is to reduce the storage size of images while preserving as much detail as possible, to obtain high-quality images without introducing undesirable artifacts. The performance evaluation of downsampling techniques is carried out using precise metrics to quantify their effectiveness,

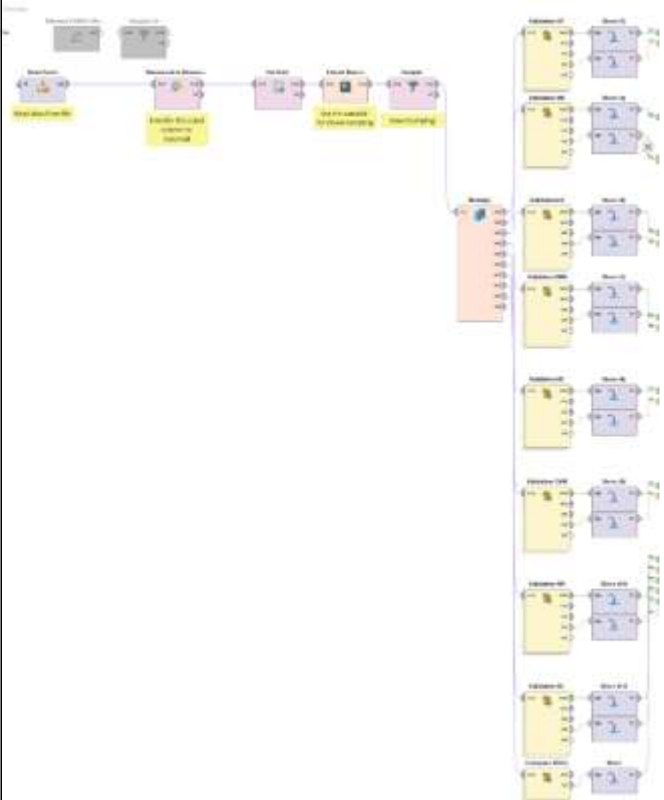
Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

strengths, and limitations. This evaluation is based on rigorous testing conducted on meticulously chosen image datasets [42].

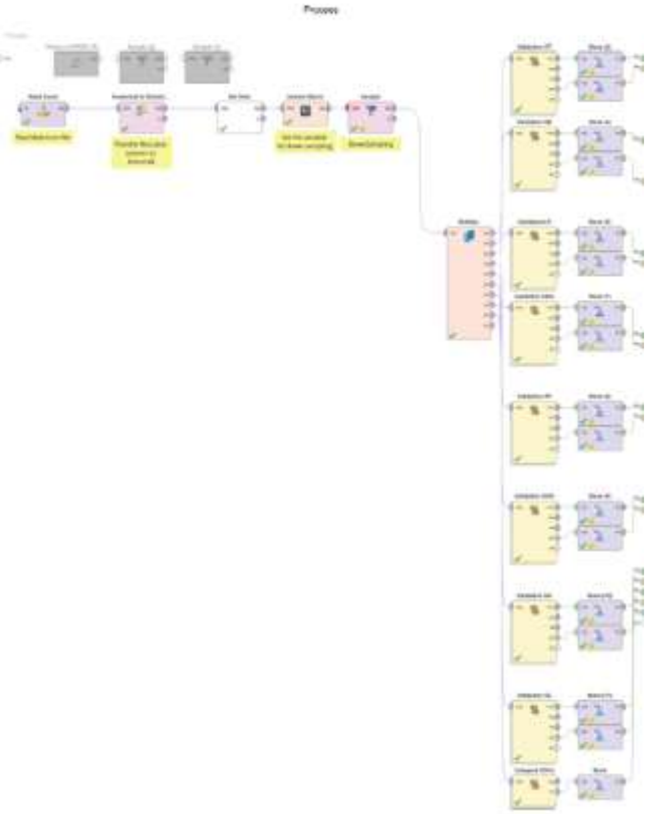
The process of downsampling in RapidMiner Studio, as shown in Figure 2.10, involves inputting the dataset from UCI Machine Learning (https://archive.ics.uci.edu/) to extract the data inputted in the form of an Excel file utilized in the study. The numerical to a binomial operator is utilized to interpret association rules. Its purpose is to convert numeric data in transaction data into binomial data with the values "true" and "false". Converting numerical data to binomial involves two main steps: utilizing the "Generate Attributes" operator and employing the "Discretise by Binning" operator. The "Nominal to numerical" and "Numerical to Nominal" operators convert data types between nominal and numerical. Following this, the "Filter Examples" operator is employed. The last stage involves the usage of the "Set Role" operator, also known as the Role Operator, which assigns roles to attributes in the dataset. The attribute's role dictates its utilization in the analysis and modeling process. The set role function includes determining the target or label, assigning the predictor attribute, designating the ID attribute, specifying the weight attribute, and setting the special attribute.

The Extract Macro function is employed to retrieve values from data or process outcomes and store them as macros. In RapidMiner, the term "macro" refers to a variable that serves the purpose of storing information and is utilized at different phases of the analytical process. Additional functionalities of the extracted macro include: storing values for future utilization, using values in parameters, adapting operations based on situations, and automating and repeating procedures.
The sample operator is employed to extract a subset of the dataset. This operator proves highly advantageous in multiple scenarios, including reducing the dataset's size for preliminary analysis, generating training and testing datasets, and conducting cross-validation. The primary function of the sample operator is to diminish the dataset's size and facilitate the creation of training and testing datasets through random or stratified sampling. Furthermore, the Multiply function is employed to replicate a preexisting dataset. The operator has great utility in many scenarios, particularly when there is a need to simultaneously execute multiple actions on a single dataset. Other functionalities of the multiplication operator include dataset replication, parallel processing, testing and validation, and conducting experiments with multiple models.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471



Before Process Down Sampling



After Process the sampling

Figure 2. Process Down Sampling

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Result of Algorithm Naïve Bayes

The study employed the Naïve Bayes Algorithm in the RapidMiner Studio tool to analyze the Simple Distribution model's findings for the label attribute "default payment next month." The class returns a bogus value of 0.500. There are 23 distributions and the class True has a probability of 0.500. There are 23 distributions.

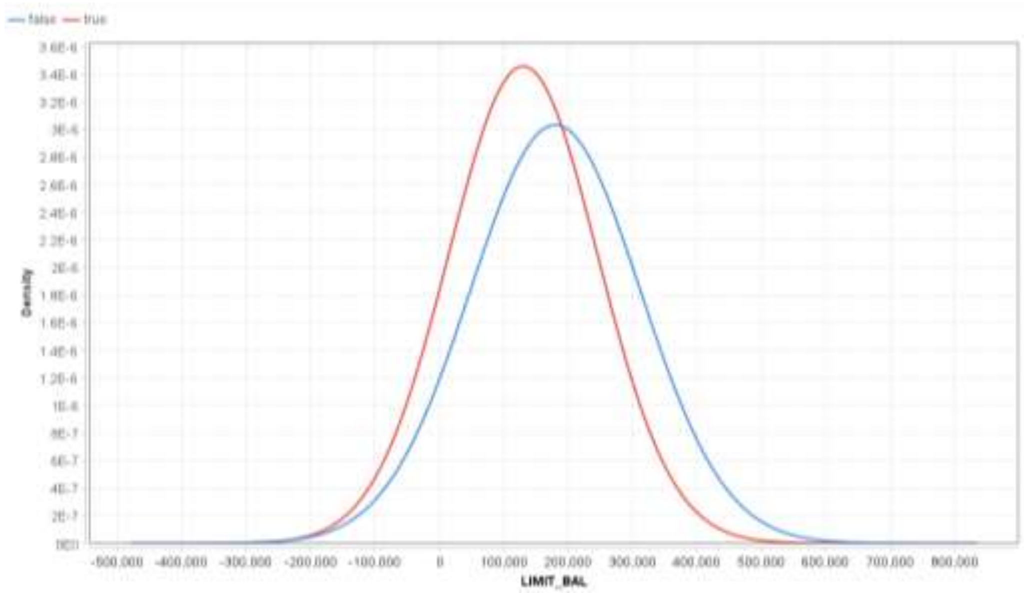


Figure 3. Simple Chart Algorithm Naïve Baye

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Distribution Table Algorithm Naïve Bayes

Table 1. Simple Distribution Table Naïve Bayes

ATTRIBUTE	PARAMETER	FALSE	TRUE
LIMIT_BAL	MEAN	180202.883	130109.656
LIMIT_BAL	STANDARD DEVIATION	131546.55	115378.541
SEX	MEAN	1.615	1.567
SEX	STANDARD DEVIATION	0.487	0.496
EDUCATION	MEAN	1.837	1.895
EDUCATION	STANDARD DEVIATION	0.81	0.728
MARRIAGE	MEAN	1.565	1.528
MARRIAGE	STANDARD DEVIATION	0.517	0.525
AGE	MEAN	35.381	35.726
AGE	STANDARD DEVIATION	8.974	9.693
PAY_0	MEAN	-0.233	0.668
PAY_0	STANDARD DEVIATION	0.943	1.383
PAY_2	MEAN	-0.309	0.458
PAY_2	STANDARD DEVIATION	1.03	1.502
PAY_3	MEAN	-0.325	0.362
PAY_3	STANDARD DEVIATION	1.036	1.499
PAY_4	MEAN	-0.359	0.255
PAY_4	STANDARD DEVIATION	1	1.509
PAY_5	MEAN	-0.389	0.168
PAY_5	STANDARD DEVIATION	0.97	1.483
PAY_6	MEAN	-0.396	0.112
PAY_6	STANDARD DEVIATION	1.001	1.486
BILL_AMT1	MEAN	52105.48	48509.162
BILL_AMT1	STANDARD DEVIATION	73571.544	73782.067
BILL_AMT2	MEAN	49757.148	47283.618
BILL_AMT2	STANDARD DEVIATION	71490.467	71651.03
BILL_AMT3	MEAN	48046.701	45181.599
BILL_AMT3	STANDARD DEVIATION	72008.462	68516.976
BILL_AMT4	MEAN	43693.142	42036.951
BILL_AMT4	STANDARD DEVIATION	64663.717	64351.076
BILL_AMT5	MEAN	41134.085	39540.19
BILL_AMT5	STANDARD DEVIATION	61859.143	61424.696
BILL_AMT6	MEAN	39811.737	38271.436
BILL_AMT6	STANDARD DEVIATION	60623.749	59579.674
PAY_AMT1	MEAN	6243.008	3397.044
PAY_AMT1	STANDARD DEVIATION	17786.109	9544.252
PAY_AMT2	MEAN	7202.575	3388.65
PAY_AMT2	STANDARD DEVIATION	31899.599	11737.986
PAY_AMT3	MEAN	5593.488	3367.352
PAY_AMT3	STANDARD DEVIATION	15605.804	12959.624
PAY_AMT4	MEAN	5519.515	3155.627
PAY_AMT4	STANDARD DEVIATION	16433.054	11191.973
PAY_AMT5	MEAN	5524.257	3219.14
PAY_AMT5	STANDARD DEVIATION	17343.929	11944.731
PAY_AMT6	MEAN	5645.396	3441.482
PAY_AMT6	STANDARD DEVIATION	17805.57	13464.006

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Performance Vector Decision Tree (DT)

The following are the performance metrics of the Decision Tree (DT): Accuracy, Precision, Recall, AUC (Optimistic), AUC, AUC (Pessimistic).

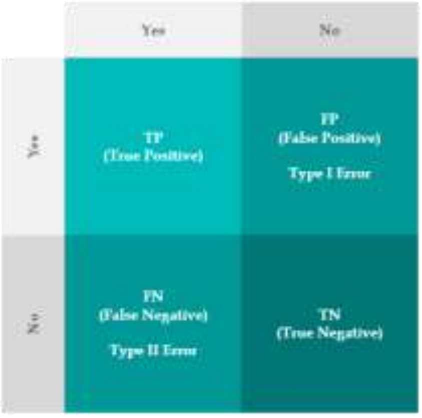


Figure 4. Illustration Figure Confusion Matrix

Accuracy Decision Tree

Table View Plot View

accuracy: 66.87% +/- 2.13% (micro average: 66.87%)

	true false	true true	class precision
pred. false	6163	3924	61.10%
pred. true	473	2712	85.15%
class recall	92.87%	40.87%	

Figure 5. Result Accuracy Confusion Matrix Decision Tree

Formula Confusion Matrix

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$

$$Class\ Precision = \frac{TP}{TP+FP}\ or\ \frac{TN}{TN+FN} \quad (2)$$

$$Class\ Recall = \frac{TP}{TP+FN}\ or\ \frac{TN}{TN+FP} \quad (3)$$

Where:

TP: True Positive

TN: True Negative

TP: True Positive

FP: False Positive

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

TN: True Negative

FN: False Negative

Results Confusion Matrix Decision Tree

$$\text{Accuracy} = \frac{6.163+2.712}{6.163+3.924+2.712+473} = \frac{8.875}{13.272} = 0.6687 \text{ x } 100\% = 66.87\%$$

$$\text{Class Precision} = \frac{6.163}{6.163+3.924} = \frac{6.163}{10.087} = 0.6109 \text{ x } 100\% = 61.09\%$$

$$\text{or } \frac{2.712}{473+2.712} = \frac{2.712}{3.185} = 0.5524 \text{ x } 100\% = 85.1491\%$$

$$\text{Class Recall} = \frac{6.163}{6.163+473} = \frac{6.163}{6.636} = 0.9287 \text{ x } 100\% = 92.872\%$$

$$\text{or } \frac{2.712}{2.712+3.924} = \frac{2.712}{6.636} = 0.4086 \text{ x } 100\% = 40.867\%$$

Precision Decision Tree

The precision values of the Decision Tree differ from the accuracy numbers, although being calculated using the same formula.

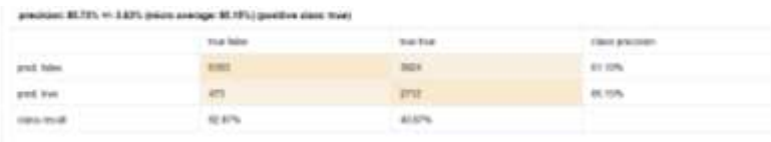


Figure 6. Result Precision Confusion Matrix Decision Tree

Recall Decision Tree

The following are the recall values for the Decision Tree, which differ from the accuracy values using the same formula.

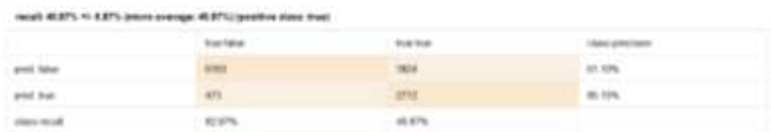


Figure 7. Result Recall Confusion Matrix Decision Tree

Performance Vector Decision Tree (DT)

The values for the performance of the vector decision tree, as mentioned before, are as follows: an accuracy of 66.67% +/- 2.13% (micro average: 66.87%). The available metrics include confusion matrix values, precision, recall, optimistic AUC values, and pessimistic AUC.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

PerformanceVector

```
PerformanceVector:
accuracy: 66.87% +/- 2.13% (micro average: 66.87%)
ConfusionMatrix:
True:   false   true
false:  6163    3924
true:   473     2712
precision: 85.73% +/- 3.63% (micro average: 85.15%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  6163    3924
true:   473     2712
recall: 40.87% +/- 6.87% (micro average: 40.87%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  6163    3924
true:   473     2712
AUC (optimistic): 0.953 +/- 0.017 (micro average: 0.953) (positive class: true)
AUC: 0.671 +/- 0.024 (micro average: 0.671) (positive class: true)
AUC (pessimistic): 0.390 +/- 0.062 (micro average: 0.390) (positive class: true)
```

Figure 8. Performance Vector Decision Tree

Graph Tree Decision Tree (DT)

The graph below illustrates a tree diagram that starts with the September payment, which is determined by the background education being greater than 5,500. The condition "education ≤ 5,500" is associated with the explanation provided in the tree diagram graph.

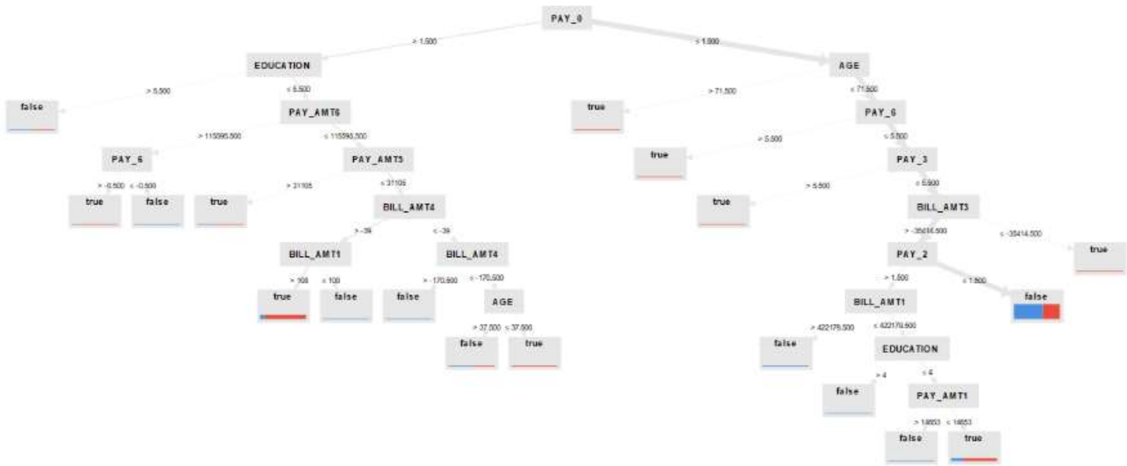


Figure 9. Tree Diagram Decision Tree

Description Tree Diagram Decision Tree

Explanation of the tree diagram illustrating the variables associated with the initial payment commencing in September 2005.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

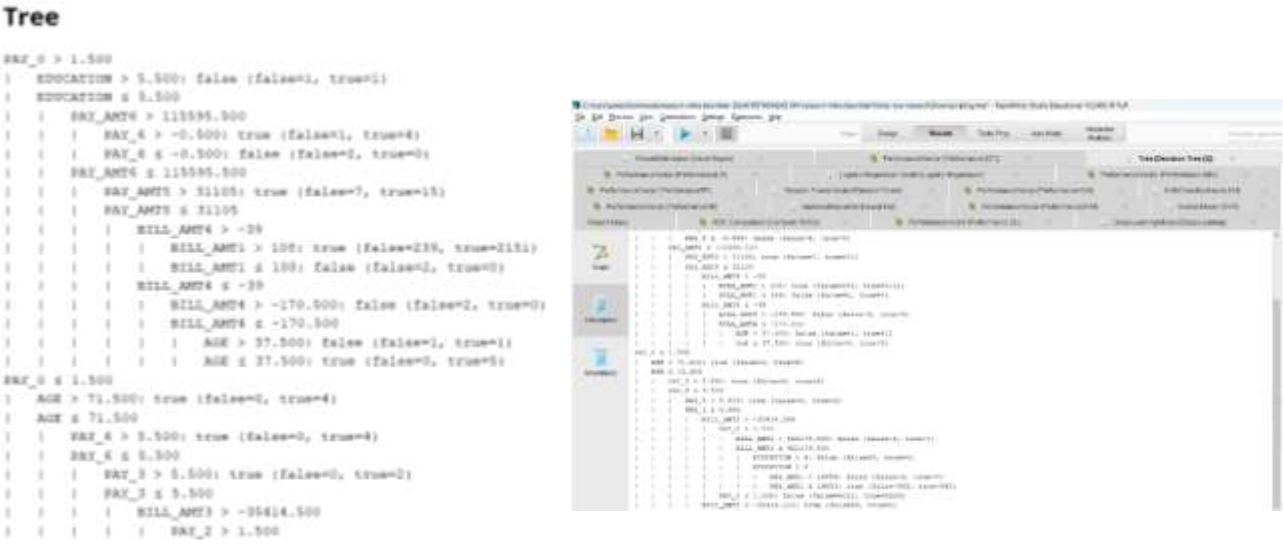


Figure 10. Description Tree Diagram Decision Tree

Accuracy Performance Vector Logistic Regression (LR)

The confusion matrix of the logistic regression algorithm can be evaluated using the following metrics: accuracy, precision, and recall.

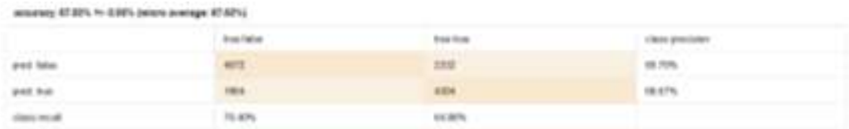


Figure 11. Accuracy Performance Vector Logistic Regression

Results Confusion Matrix Logistic Regression

$$\text{Accuracy} = \frac{4.672+4.304}{4.672+2.332+4.304+1.964} = \frac{8.976}{13.272} = 0.6756 \text{ x } 100\% = 67.56\%$$

$$\text{Class Precision} = \frac{4.672}{4.672+2.332} = \frac{4.672}{7.004} = 0.6670 \text{ x } 100\% = 66.70\%$$

$$\text{or } \frac{4.304}{4.304+1.964} = \frac{4.304}{6.268} = 0.6866 \text{ x } 100\% = 68.66\%$$

$$\text{Class Recall} = \frac{4.672}{4.672+1.964} = \frac{4.672}{6.636} = 0.7040\text{x } 100\% = 70.40\%$$

$$\text{or } \frac{4.304}{4.304+2.332} = \frac{4.304}{6.636} = 0.6485\text{x } 100\% = 64.85\%$$

Precision Performance Vector Logistic Regression (LR)

The logistic regression algorithm's precision calculation has been previously explained. Here are the precision values obtained using the RapidMiner Studio tool.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

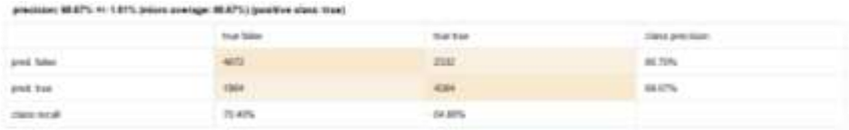


Figure 12. Precision Performance Vector Logistic Regression

Recall Performance Vector Logistic Regression (LR)

The recall calculation in the logistic regression approach has been explained previously. Here are the recall outcomes obtained using the RapidMiner Studio tool.

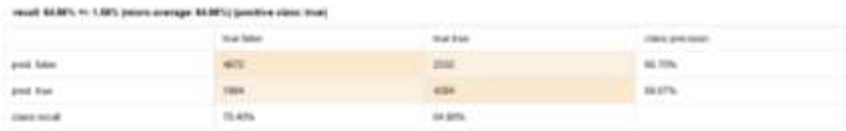


Figure 13. Recall Performance Vector Logistic Regression

Performance Vector Logistic Regression

The following description provides an overview of the performance of vector logistic regression, which achieved an accuracy level of 67.63% +/- 0.93% (micro average: 67.63%). Additionally, it includes the process of determining the confusion matrix value using the aforementioned formula.

Figure 14. Performance Vector Logistic Regression

Logistic Regression Model

Presented below is a regression table with characteristics, coefficients, standard coefficients, standard error, z-values, and p-values.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Attribute	Coefficient	Std. Coefficient	Std. Error	z-Value	p-Value
LIMIT_BAL	-0.000	-0.113	0.000	-4.634	0.000
SEX	-0.152	-0.051	0.039	[illegible]	[illegible]
EDUCATION	-0.080	-0.074	0.035	-3.645	0.000
MARRIAGE	-0.179	-0.083	0.040	-4.440	0.000
AGE	0.007	0.066	0.002	3.002	0.032
PAY_0	0.546	0.604	0.022	24.601	0
PAY_2	0.082	0.111	0.025	3.231	0.001
PAY_3	0.075	0.094	0.028	2.473	0.013
PAY_4	0.029	0.036	0.031	0.918	0.358
PAY_5	0.023	0.058	0.034	0.088	0.401
PAY_6	-0.035	-0.045	0.028	-1.248	0.212
BILL_AMT1	-0.000	-0.385	0.000	-4.230	0.000
BILL_AMT2	0.000	0.171	0.000	1.458	0.145
BILL_AMT3	0.000	0.108	0.000	0.940	0.347
BILL_AMT4	0.000	0.041	0.000	0.393	0.694
BILL_AMT5	0.000	0.029	0.000	0.258	0.796
BILL_AMT6	-0.000	-0.051	0.000	-0.610	0.542
PAY_AMT1	-0.000	-0.174	0.000	-5.082	0.000
PAY_AMT2	-0.000	-0.234	0.000	-4.793	0.000
PAY_AMT3	-0.000	-0.025	0.000	-0.929	0.353
PAY_AMT4	-0.000	-0.056	0.000	-1.961	0.040
PAY_AMT5	-0.000	-0.048	0.000	-1.688	0.091
PAY_AMT6	-0.000	-0.027	0.000	-1.146	0.252
Intercept	0.019	0.087	0.101	4.855	0.000

Figure 15. Logistic Regression Model

Accuracy Performance Vector Naïve Bayes (NB)

The accuracy value derived from Naïve Bayes modeling can be characterized as follows:

accuracy: 60.93% +/- [illegible] (micro average: 60.93%)

	true false	true true	class precision
pred. false	2534	1083	70.05%
pred. true	4102	5553	57.51%
class recall	38.18%	83.68%	

Figure 16. Accuracy Performance Vector Naïve Bayes

Results Confusion Matrix Naïve Bayes

Accuracy $= \frac{2.534+5.553}{2.534+1.083+5.553+4.102} = \frac{8.087}{13.272} = 0.6093 \text{ x } 100\% = 60.93\ \%$

Class Precision $= \frac{2.534}{2.534+1.083} = \frac{2.534}{3.617} = 0.7005 \text{ x } 100\% = 70.05\%$

or $\frac{5.553}{5.553+4.102} = \frac{5.553}{6.636} = 0.8367 \text{ x } 100\% = 83.67\%$

Class Recall $= \frac{2.534}{2.534+4.102} = \frac{2.534}{6.636} = 0.3818 \text{ x } 100\% = 38.18\%$

or $\frac{5.553}{5.553+1.083} = \frac{5.553}{6.636} = 0.8367 \text{ x } 100\% = 83.67\%$

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Precision Performance Vector Naïve Bayes (NB)

The precision value derived from the naïve Bayes algorithm is given by the formula shown above:

precision: 57.54% +/- 1.28% (micro average: 57.51%) (positive class: true)

	true false	true true	class precision
pred. false	2534	1083	70.06%
pred. true	4102	5553	57.51%
class recall	38.19%	83.68%	

Figure 17. Precicison Performance Vector Naïve Bayes

Recall Performance Vector Naïve Bayes (NB)

The formula above describes the recall value produced using Naïve Bayes:

recall: 83.68% +/- 1.24% (micro average: 83.68%) (positive class: true)

	true false	true true	class precision
pred. false	2534	1083	70.06%
pred. true	4102	5553	57.51%
class recall	38.19%	83.68%	

Figure 18. Recall Performance Vector Naïve Bayes

Performance Vector Performance Naïve Bayes (NB)

The vector performance results are provided here, including accuracy, precision, and recall numbers:

PerformanceVector

```
PerformanceVector:
accuracy: 60.93% +/- 1.55% (micro average: 60.93%)
ConfusionMatrix:
True:   false   true
false:  2534    1083
true:   4102    5553
precision: 57.54% +/- 1.20% (micro average: 57.51%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  2534    1083
true:   4102    5553
recall: 83.68% +/- 1.24% (micro average: 83.68%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  2534    1083
true:   4102    5553
AUC (optimistic): 0.740 +/- 0.013 (micro average: 0.740) (positive class: true)
AUC: 0.740 +/- 0.013 (micro average: 0.740) (positive class: true)
AUC (pessimistic): 0.740 +/- 0.013 (micro average: 0.740) (positive class: true)
```

Figure 19. Performance Vector Performance Naïve Bayes

Accuracy Performance Vector Random Forest (RF)

The accuracy value derived from Random Forest (RF) modeling can be characterized as follows:

accuracy: 69.22% +/- 1.18% (micro average: 69.22%)

	true false	true true	class precision
pred. false	5957	3406	63.62%
pred. true	679	3230	82.63%
class recall	89.77%	48.67%	

Figure 20. Accuracy Performance Vector Performance Random Forest

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Results Confusion Matrix Random Forest

$$\text{Accuracy} = \frac{5.957+3.230}{5.957+3.406+3.230+679} = \frac{9.187}{13.272} = 0.6922 \text{ x } 100\% = 69.22\%$$

$$\text{Class Precision} = \frac{5.957}{5.957+3.406} = \frac{5.957}{9.363} = 0.6362 \text{ x } 100\% = 63.62\%$$

$$\text{or } \frac{3.230}{3.230+679} = \frac{3.230}{3.909} = 0.8262 \text{ x } 100\% = 82.62\%$$

$$\text{Class Recall} = \frac{5.957}{5.957+679} = \frac{5.957}{6.636} = 0.8976 \text{ x } 100\% = 89.76\%$$

$$\text{or } \frac{3.230}{3.230+3.406} = \frac{3.230}{6.636} = 0.4867 \text{ x } 100\% = 48.67\ \%$$

Precision Performance Vector Random Forest (RF)

The precision value derived from Random Forest (RF) modeling can be characterized as follows:

precision: 82.68% +/- 1.82% (micro average: 82.62%) (positive class: true)

	true false	true true	class precision
pred. false	5957	3406	63.62%
pred. true	679	3230	82.62%
class recall	89.77%	48.67%	

Figure 21. Precision Performance Vector Performance Random Forest

Recall Performance Vector Random Forest (RF)

The recall value produced from Random Forest modeling can be defined as:

recall: 48.67% +/- 2.11% (micro average: 48.67%) (positive class: true)

	true false	true true	class precision
pred. false	5957	3406	63.62%
pred. true	679	3230	82.62%
class recall	89.77%	48.67%	

Figure 22. Recall Performance Vector Performance Random Forest

Performance Vector Random Forest

The performance metrics of the vector random forest model are as follows: accuracy, precision, and recall values.

PerformanceVector

```
PerformanceVector:
accuracy: 69.22% +/- 1.18% (micro average: 69.22%)
ConfusionMatrix:
True:   false   true
false:  5957    3406
true:   679     3230
precision: 82.65% +/- 1.82% (micro average: 82.63%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  5957    3406
true:   679     3230
recall: 48.67% +/- 2.11% (micro average: 48.67%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  5957    3406
true:   679     3230
AUC (optimistic): 0.811 +/- 0.019 (micro average: 0.811) (positive class: true)
AUC: 0.745 +/- 0.010 (micro average: 0.745) (positive class: true)
AUC (pessimistic): 0.680 +/- 0.030 (micro average: 0.680) (positive class: true)
```

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Figure 23. Performance Vector Random Forest

Graph Model Random Forest

The graphic model of the random forest method is depicted as follows, commencing with the payments in September 2005 and subsequently incorporating the education and age data.

Figure 24. Graph Model Random Forest

Description Random Forest Model (Random Forest)

This text describes the tree graph used in the Random Forest (RF) algorithm, specifically focussing on the concepts of True Positive, False Positive, True Negative, and False Negative.

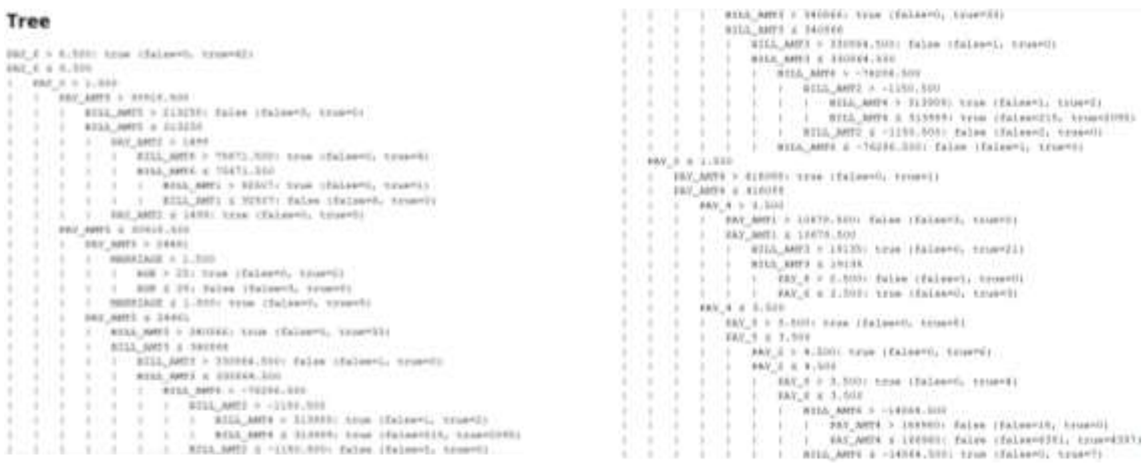
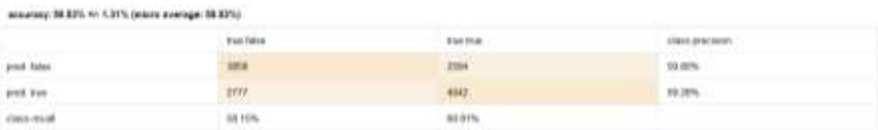


Figure 25. Description Model Random Forest

Accuracy Performance Vector K-Nearest Neighbor Classifier (KNN)

The accuracy value derived from K-NN modeling can be characterized as follows:



Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Figure 26. Accuracy Performance Vector K-Nearest Neighbor Classifier

Results Confusion Matrix K-Nearest Neighbor Classifier

Accuracy $= \frac{3.859 + 4.042}{3.859 + 2.594 + 4.042 + 2.777} = \frac{7.901}{13.272} = 0.5953 \text{ x } 100\% = 59.53\%$

Class Precision $= \frac{3.859}{3.859 + 2.954} = \frac{3.859}{6.813} = 0.5664 \text{ x } 100\% = 56.64\%$

or $\frac{4.042}{4.042 + 2.777} = \frac{4.042}{6.819} = 0.5927 \text{ x } 100\% = 59.27\ \%$

Class Recall $= \frac{3.859}{3.859 + 2.777} = \frac{3.859}{6.636} = 0.5815 \text{ x } 100\% = 58.15\%$

or $\frac{4.042}{4.042 + 2.594} = \frac{4.042}{6.636} = 0.6091 \text{ x } 100\% = 60.91\%$

Precision Performance Vector K-Nearest Neighbor Classifier (KNN)

The precision value derived from K-NN modeling can be characterized as follows:

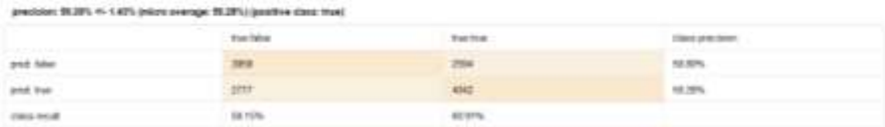


Figure 27. Precision Performance Vector K-Nearest Neighbor Classifier

Recall Performance Vector K-Nearest Neighbor Classifier (KNN)

The recall value derived from K-NN modelling can be defined as:

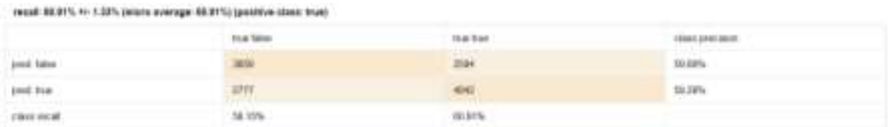


Figure 28. Recall Performance Vector K-Nearest Neighbor Classifier

K-Nearest Neighbor Classification

The K-Nearest Neighbor classification algorithm can be defined as follows:

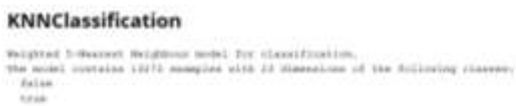


Figure 29. K-Nearest Neighbor Classifier Classification

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Accuracy Performance Vector Neural Net (NN)

The accuracy value derived from Neural Net (NN) modeling can be defined as follows:

accuracy: 70.00% +/- [illegible] (micro average: 70.00%)

	true false	true true	class precision
pred. false	5259	2604	66.88%
pred. true	1377	4032	74.54%
class recall	79.25%	60.76%	

Figure 30. Accuracy Performance Vector Neural Net

Results Confusion Matrix Neural Net

$$\text{Accuracy} = \frac{5.259 + 4.032}{5.259 + 2.604 + 4.032 + 1.377} = \frac{9.291}{13.271} = 0.70000 \text{ x } 100\% = 70.00\%$$

$$\text{Class Precision} = \frac{5.259}{5.259 + 2.604} = \frac{5.259}{7.863} = 0.6688 \text{ x } 100\% = 66.88\ \%$$

$$\text{or } \frac{4.032}{4.032 + 1.377} = \frac{4.032}{5.409} = 0.7454 \text{ x } 100\% = 74.54\ \%$$

$$\text{Class Recall} = \frac{5.259}{5.259 + 1.377} = \frac{5.259}{6.636} = 0.7924 \text{ x } 100\% = 79.24\%$$

$$\text{or } \frac{4.032}{4.032 + 2.604} = \frac{4.032}{6.636} = 0.6075 \text{ x } 100\% = 60.75\%$$

Precision Performance Vector Neural Net (NN)

The precision value derived from neural network modeling can be characterized as follows:

precision: [illegible] (micro average: 74.54%) (positive class: true)

	true false	true true	class precision
pred. false	5259	2604	66.88%
pred. true	1377	4032	74.54%
class recall	79.25%	60.76%	

Figure 31. Precision Performance Vector Neural Net

Recall Performance Vector Neural Net (NN)

The recall value derived from neural network modeling can be defined as follows:

recall: 60.77% +/- 8.04% (micro average: 60.76%) (positive class: true)

	true false	true true	class precision
pred. false	5259	2604	66.88%
pred. true	1377	4032	74.54%
class recall	79.25%	60.76%	

Figure 32. Recall Performance Vector Neural Net

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Performance Vector Neural Net

The vector neural net performance is characterized by accuracy, precision, and recall values, which are outlined below:

PerformanceVector

```
PerformanceVector:
accuracy: 70.00% +/- 2.01% (micro average: 70.00%)
ConfusionMatrix:
True:   false   true
false:  5259    2606
true:   1377    4032
precision: 75.55% +/- 3.57% (micro average: 74.34%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  5259    2606
true:   1377    4032
recall: 60.77% +/- 5.04% (micro average: 60.76%) (positive class: true)
ConfusionMatrix:
True:   false   true
false:  5259    2606
true:   1377    4032
AUC (optimistic): 0.767 +/- 0.017 (micro average: 0.767) (positive class: true)
AUC: 0.767 +/- 0.017 (micro average: 0.767) (positive class: true)
AUC (pessimistic): 0.767 +/- 0.017 (micro average: 0.767) (positive class: true)
```

Figure 33. Performance Vector Neural Net

Improved Neural Net

The graph below illustrates the interconnections between the input, hidden 1, and output layers.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

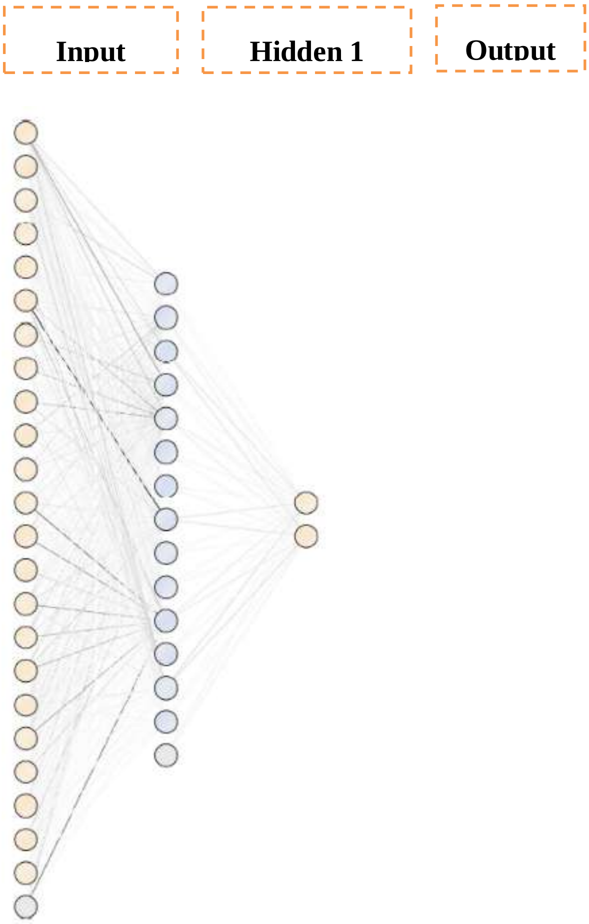


Figure 34. Performance Vector Neural Net

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Improved Neural Net

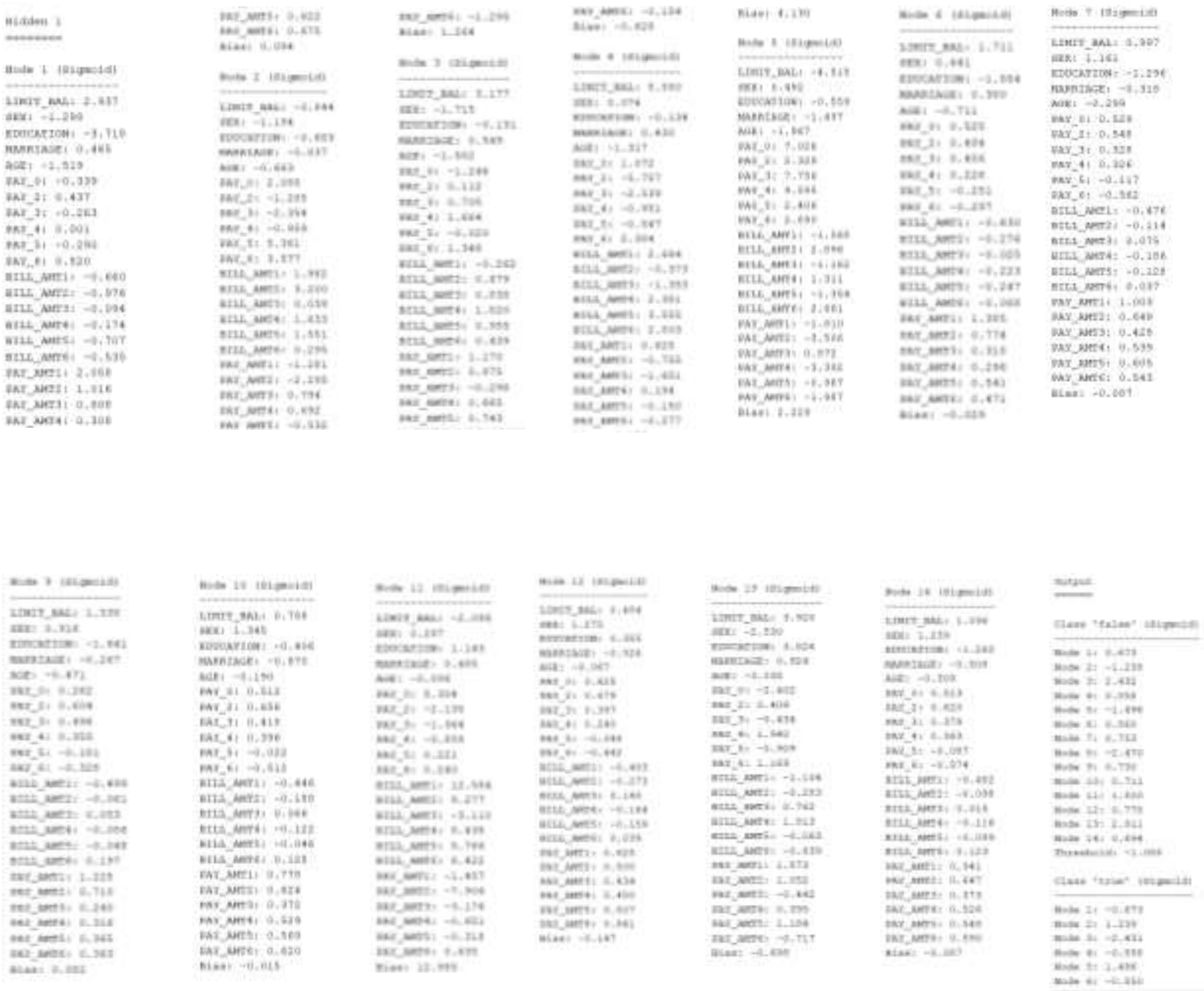


Figure 35. Improved Neural Net

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Performance Vector Support Vector Machine (SVM)

The accuracy value derived from Support Vector Machine (SVM) modeling can be characterized as follows:

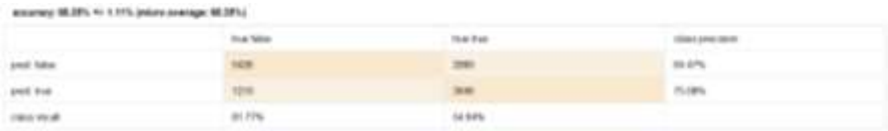


Figure 36. Accuracy Performance Vector Neural Net

Results accuracy Support Vector Machine (SVM)

$$\text{Accuracy} = \frac{5.426 + 3.646}{5.426 + 2.990 + 3.646 + 1.210} = \frac{9.072}{13.272} = 0.6835 \text{ x } 100\% = 68.35\%$$

$$\text{Class Precision} = \frac{5.426}{5.426 + 2.990} = \frac{5.426}{8.416} = 0.6447 \text{ x } 100\% = 64.47\%$$

$$\text{or } \frac{3.646}{3.646 + 1.210} = \frac{3.646}{4.856} = 0.7508 \text{ x } 100\% = 75.08\%$$

$$\text{Class Recall} = \frac{5.426}{5.426 + 1.210} = \frac{5.426}{6.636} = 0.8176 \text{ x } 100\% = 81{,}76\%$$

$$\text{or } \frac{3.646}{3.646 + 2.990} = \frac{3.646}{6.636} = 0.5494 \text{ x } 100\% = 54.94\%$$

Precision Performance Support Vector Machine (SVM)

The precision value derived from Support Vector Machine (SVM) modeling can be characterized as follows:

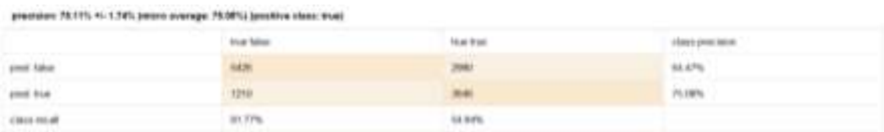


Figure 37. Precision Performance Support Vector Machine

Recall Performance Support Vector Machine (SVM)

The recall value derived from Support Vector Machine (SVM) modeling can be defined as follows:

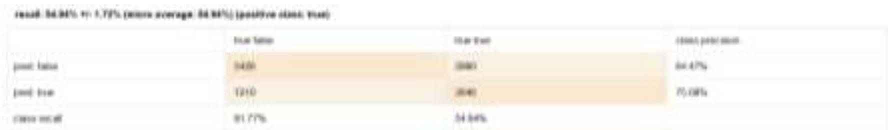


Figure 38. Recall Performance Support Vector Machine

Kernel Model Support Vector Machine (SVM)

The kernel model of the Support Vector Machine (SVM) derived from modeling can be characterized by the following parameters: the total number of Support Vectors is 13272, and the bias (offset) is -0.180.

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Kernel Model

```
Total number of Support Vectors: 13272
Bias (offset): -0.180

w[LIMIT_BAL] = -0.226
w[SEX] = -0.074
w[EDUCATION] = -0.097
w[MARRIAGE] = -0.088
w[AGE] = 0.077
w[PAY_0] = 0.693
w[PAY_2] = 0.148
w[PAY_3] = 0.061
w[PAY_4] = 0.017
w[PAY_5] = 0.028
w[PAY_6] = -0.046
w[BILL_AMT1] = -0.370
w[BILL_AMT2] = 0.088
w[BILL_AMT3] = 0.045
w[BILL_AMT4] = 0.164
w[BILL_AMT5] = 0.039
w[BILL_AMT6] = -0.080
w[PAY_AMT1] = -0.085
w[PAY_AMT2] = -0.148
w[PAY_AMT3] = 0.002
w[PAY_AMT4] = -0.010
w[PAY_AMT5] = -0.006
w[PAY_AMT6] = -0.001
```

Figure 39. Kernel Model Support Vector Machine (SVM)

Weight Table Support Vector Machine (SVM)

The Weight Table value derived from Support Vector Machine (SVM) modeling can be characterized as follows:

Table 2. Kernel Model Support Vector Machine (SVM)

ATRRIBUTE	WEIGHT
LIMIT_BAL	-0.226
SEX	-0.074
EDUCATION	-0.097
MARRIAGE	-0.088
AGE	0.077
PAY_0	0.693
PAY_2	0.148
PAY_3	0.061
PAY_4	0.017
PAY_5	0.028
PAY_6	-0.046
BILL_AMT1	-0.370
BILL_AMT2	0.088
BILL_AMT3	0.045
BILL_AMT4	0.164
BILL_AMT5	0.039
BILL_AMT6	-0.080
PAY_AMT1	-0.085
PAY_AMT2	-0.148
PAY_AMT3	0.002
PAY_AMT4	-0.010
PAY_AMT5	-0.006
PAY_AMT6	-0.001

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Weight Visualizations Support Vector Machine (SVM)

The Weight Visualisations value derived from Support Vector Machine (SVM) modelling can be represented in a graph as follows:

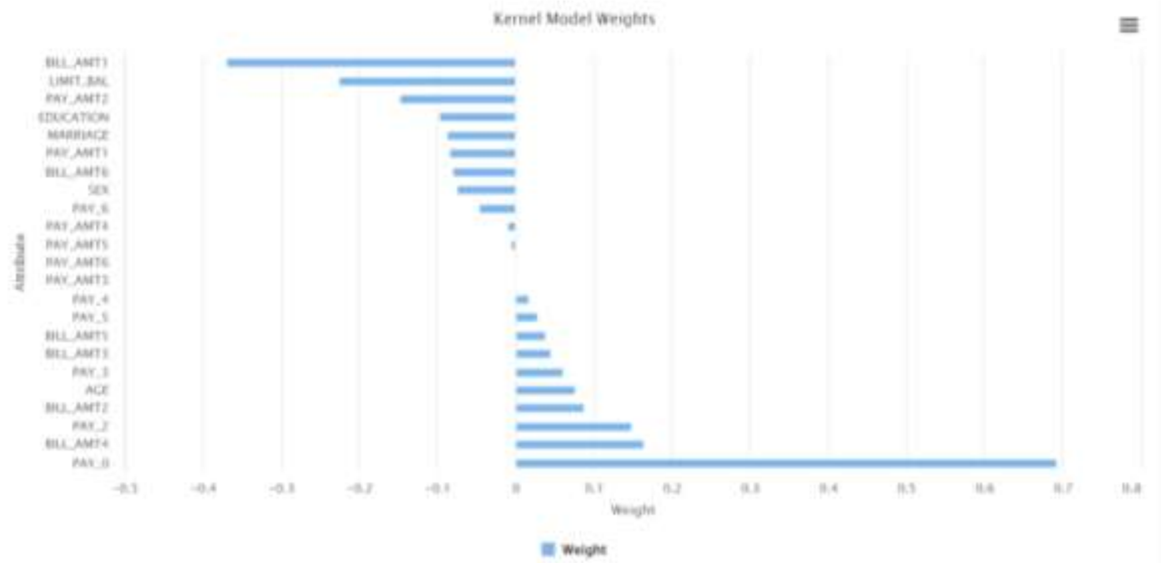


Figure 40. Weight Visualizations Support Vector Machine (SVM)

Support Vector Visualizations Support Vector Machine (SVM)

The Weight Visualization value derived by Support Vector Machine (SVM) modeling can be represented in a graph as follows:

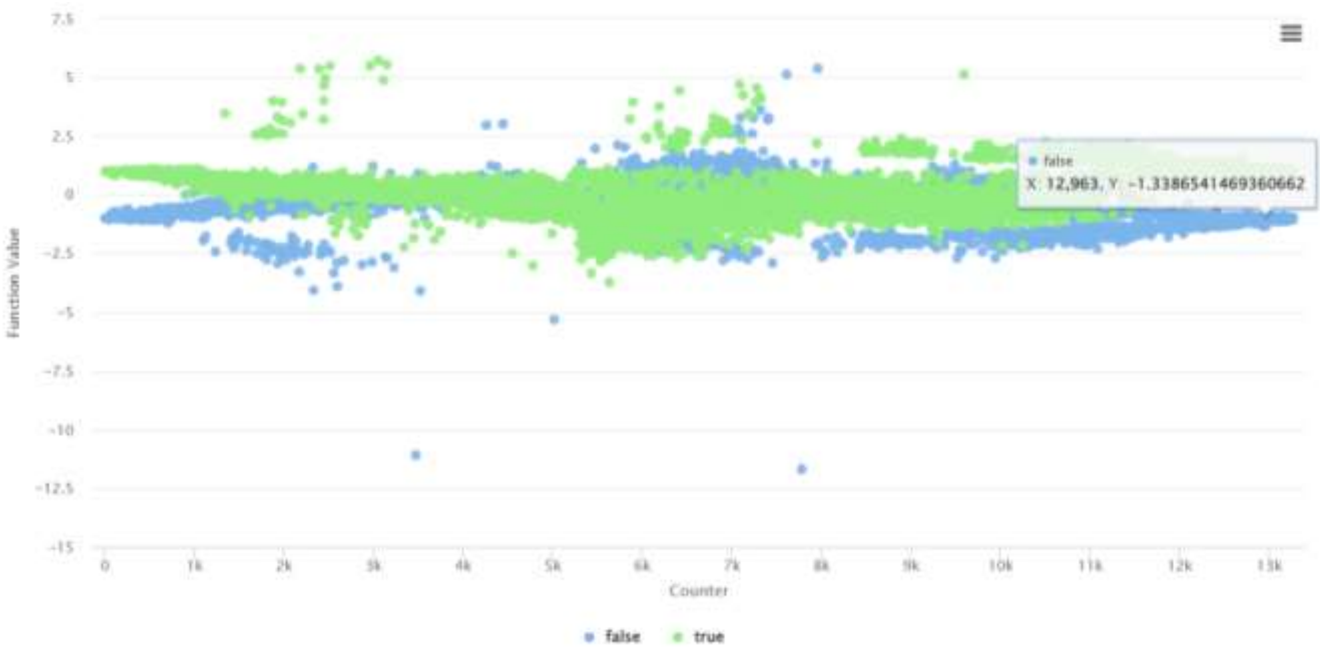


Figure 41. Weight Visualizations Support Vector Machine (SVM)

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

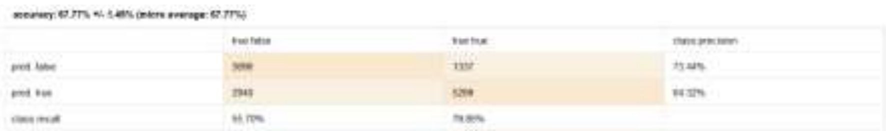
Receiver Operating Characteristic (ROC) Curve Downsampling

The ROC graph presented below illustrates the accuracy calculation of each algorithm. Based on the research findings, the Downsampling algorithm with the highest performance is Decision Tree (DT), followed by Random Forest (RF) in second place. Deep Learning (DL) ranks third, Neural Net (NN) fourth, Naive Bayes (NB) fifth, Logistic Regression (LR) sixth, Support Vector Machine (SVM) seventh, and K-Nearest Neighbor Classifier (K-NN) last.

Figure 42. Receiver Operating Characteristic (ROC) Curve Downsampling

Performance Vector Deep learning (DL)

The accuracy value derived from Deep Learning (DL) modeling can be defined as:



accuracy: 67.77% +/- 1.48% (micro average: 67.77%)

	true false	true true	class precision
pred. false	3696	1337	73.44%
pred. true	2940	5299	64.32%
class recall	55.70%	79.86%	

Figure 43. Accuracy Performance Vector Deep Learning (DL)

Results Accuracy Deep Learning (DL)

$$\text{Accuracy} = \frac{3.696 + 5.299}{3.696 + 1.337 + 5.299 + 2.940} = \frac{8.995}{13.272} = 0.677\text{x } 100\% = 67.77\%$$

$$\text{Class Precision} = \frac{3.696}{3.696 + 1.337} = \frac{3.696}{5.033} = 0.7343 \text{ x } 100\% = 73.43\%$$

$$\text{or } \frac{5.299}{5.299 + 2.940} = \frac{5.299}{8.239} = 0.6431 \text{ x } 100\% = 64.31\%$$

$$\text{Class Recall} = \frac{3.696}{3.696 + 2.940} = \frac{3.696}{6.636} = 0.5569 \text{ x } 100\% = 55.69\%$$

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

$$\text{or } \frac{5.299}{5.299 + 1.337} = \frac{5.299}{6.636} = 0.7985 \text{ x } 100\% = 79.85\%$$

Precision Performance Deep Learning (DL)

The precision value derived from Deep Learning (DL) modelling can be characterised as follows:

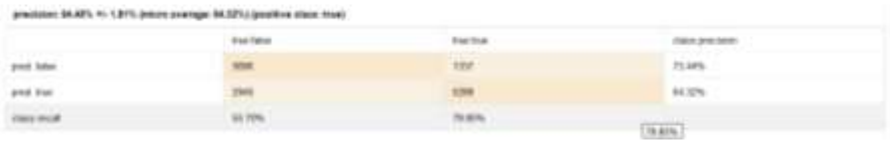


Figure 44. Precision Performance Vector Deep Learning (DL)

Recall Performance Deep Learning (DL)

The recall value derived from Deep Learning (DL) modeling can be defined as:

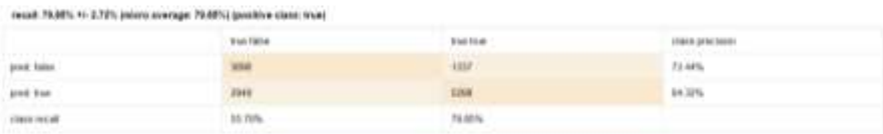


Figure 45. Recall Performance Vector Deep Learning (DL)

Performance Vector Deep Learning (DL)

The Deep Learning vector performance is presented here, including accuracy, precision, and recall values.

Figure 46. Performance Vector Deep Learning (DL)

Imbalance

This paper discusses the concept of class distribution imbalance in a dataset, where one class (majority class) greatly surpasses another class (minority class). This imbalance is commonly observed in real-world applications, where the minority class, usually the positive class, is much smaller in proportion compared to the majority class. This poses challenges in achieving accurate classification [43].

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

This paper discusses the concept of imbalance, which refers to situations where datasets have significantly unequal class distributions. Imbalance is a common problem in various fields, including telecommunication management, bioinformatics, fraud detection, and medical diagnosis. It poses a significant challenge in data mining and pattern recognition, as it can hinder the learning process for machine learning algorithms [44].

This paper discusses the concept of skewed class distribution in classification tasks, when one class is much more prevalent than the other class in the dataset [45].

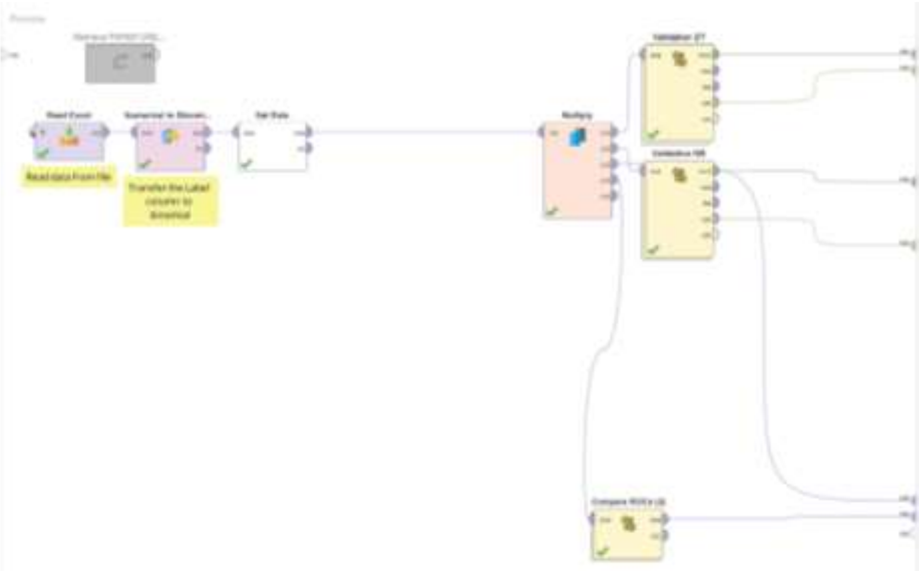


Figure 47. Process Imbalance Using RapidMiner Studio

Tree Graph Decision Tree (DT) Imbalance

The tree graph is characterized by the interdependence of initial payments, as exemplified by the Decision Tree (DT) graph provided below:

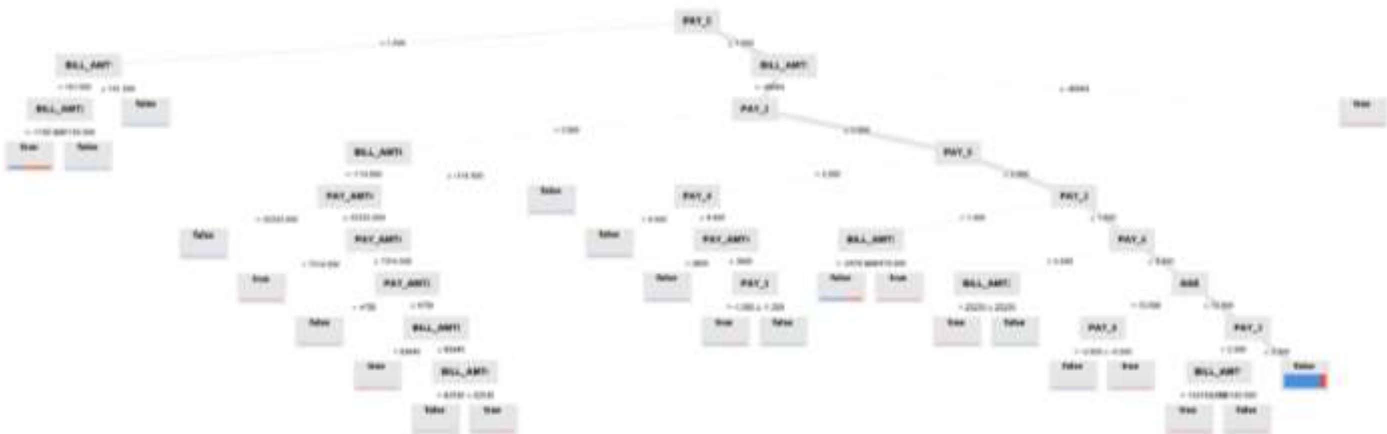


Figure 48. Tree Graph Decision Tree (DT)

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Performance Vector Decision Tree (DT)

The accuracy value derived from Decision Tree (DT) modeling can be characterized as follows:

[illegible]

	true false	true true	class precision
pred. false	22298	4362	83.64%
pred. true	1066	2274	68.08%
class recall	95.44%	34.27%	

Figure 49. Accuracy Performance Vector Decision Tree (DT)

Results Accuracy Decision Tree (DT)

$$\text{Accuracy} = \frac{22.298 + 2.274}{22.298 + 4.362 + 2.274 + 1.066} = \frac{24.572}{30.000} = 0.8190 \text{ x } 100\% = 81.90\%$$

$$\text{Class Precision} = \frac{22.298}{22.298 + 4.362} = \frac{22.298}{26.660} = 0.8363 \text{ x } 100\% = 83.63\ \%$$

$$\text{or } \frac{2.274}{2.274 + 1.066} = \frac{2.274}{3.340} = 0.6808 \text{ x } 100\% = 68.08\%$$

$$\text{Class Recall} = \frac{22.298}{22.298 + 1.066} = \frac{22.298}{23.364} = 0.9543 \text{ x } 100\% = 95.43\%$$

$$\text{or } \frac{2.274}{2.274 + 4.362} = \frac{2.274}{6.636} = 0.3426 \text{ x } 100\% = 34.26\%$$

Precision Performance Decision Tree (DT)

The precision value derived from Decision Tree (DT) modeling can be characterized as follows:

[illegible]

	true false	true true	class precision
pred. false	22298	4362	83.64%
pred. true	1066	2274	68.08%
class recall	95.44%	34.27%	

Figure 50. Precision Performance Vector Decision Tree (DT)

Recall Performance Decision Tree (DT)

The recall value derived from Decision Tree (DT) modeling can be defined as follows:

[illegible]

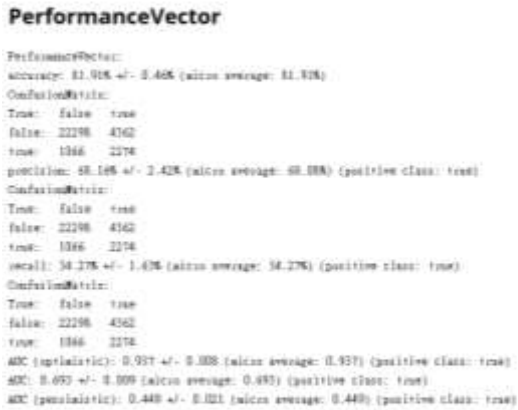
	true false	true true	class precision
pred. false	22298	4362	83.64%
pred. true	1066	2274	68.08%
class recall	95.44%	34.27%	

Figure 51. Recall Performance Vector Decision Tree (DT)

Journal of Ecohumanism
2024
Volume: 3, No: 7, pp. 3386 – 3418
ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471

Performance Vector Decision Tree (DT)

The performance metrics of the Decision Tree model, including accuracy, precision, and recall values, are outlined below:



PerformanceVector

PerformanceVector:
accuracy: 81.91% +/- 0.46% (micro average: 81.91%)
ConfusionMatrix:
True: false true
false: 22298 4362
true: 1366 2274
precision: 68.16% +/- 2.42% (micro average: 68.08%) (positive class: true)
ConfusionMatrix:
True: false true
false: 22298 4362
true: 1366 2274
recall: 34.27% +/- 1.43% (micro average: 34.27%) (positive class: true)
ConfusionMatrix:
True: false true
false: 22298 4362
true: 1366 2274
AUC (optimistic): 0.937 +/- 0.008 (micro average: 0.937) (positive class: true)
AUC: 0.693 +/- 0.009 (micro average: 0.693) (positive class: true)
AUC (pessimistic): 0.448 +/- 0.021 (micro average: 0.448) (positive class: true)

Figure 52. Performance Vector Decision Tree (DT)

Simple Charts Distributions Naive Bayes (NB)

The graph illustrates the distribution of the Naive Bayes algorithm for imbalance, showcasing the true and false values in the following diagram.

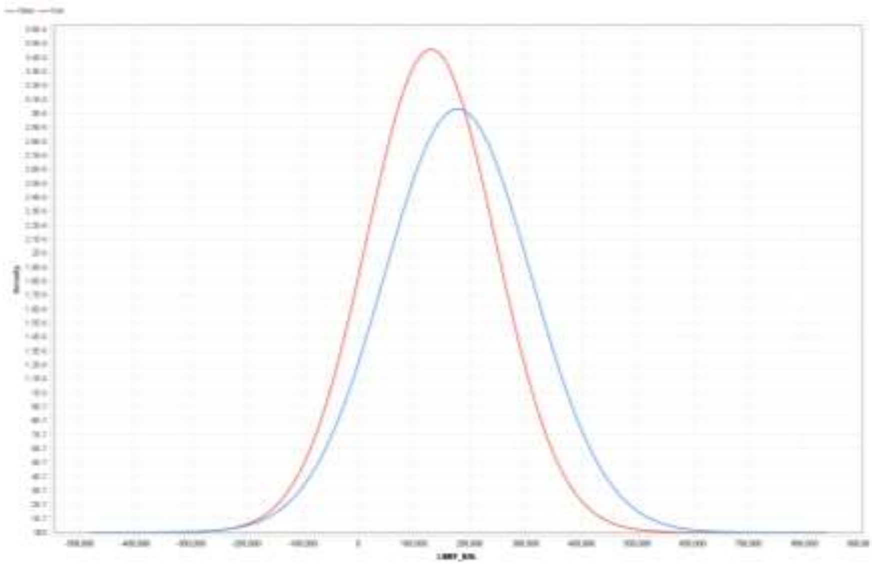


Figure 53. Simple Charts Distribution Decision Tree (DT)

Conclusion

This research employs four machine learning algorithms, as previously mentioned [6][51]: Imbalance Technique, Downsampling Technique, Weighting Technique, SMOTE Technique. The study used eight algorithms K-Nearest Neighbor (KNN), Logistic Regression (LR), Naïve Bayesian Classifier (NB), Random Forest (RF), Decision Tree (DT), Neural Net (NN), Deep learning (DL), Support Vector Machine (SVM).

Imbalance technique for Decision Tree (DT) algorithm with accuracy level value of 81.91% with AUC value of 0.937 and Naive Bayes Classifier (NB) value with accuracy level value of 71.43% and AUC value of 0.737. Downsampling technique for Decision Tree (DT) algorithm with accuracy level value of 66.87% with AUC value of 0.953, Logistic Regression (LR) algorithm with accuracy level value of 67.63% with AUC value of 0.729, Naive Bayes Classifier (NB) algorithm with accuracy level value of 60.93% with AUC value of 0.740, Random Forest (RF) algorithm with accuracy level value of 69.22% with AUC value of 0.811, K-Nearest Neighbor (KNN) algorithm with accuracy level value of 59.53% with AUC value of 0.627, Neural Net (NN) algorithm with accuracy level value of 70.00% with AUC value of 0.767, Support Vector Machine (SVM) algorithm with accuracy level value of 68.35% with AUC value of 0.727, Deep Learning (DL) algorithm with accuracy level value of 67.77% with AUC value of 0.777. Weighting technique for the Naive Bayes Classifier (NB) algorithm with an accuracy level value of 60.56% with an AUC value of 0.736, the Decision Tree (DT) algorithm with an accuracy level value of 64.31% with an AUC value of 0.956.

The SMOTE technique for the Decision Tree (DT) algorithm with an accuracy level value of 68.15% with an AUC value of 0.945, the Naive Bayes Classifier (NB) algorithm with an accuracy level value of 58.42% with an AUC value of 0.740.

References

R. Liu, "Machine Learning Approaches to Predict Default of Credit Card Clients," Mod. Econ., vol. 09, no. 11, pp. 1828–1838, 2018, doi: 10.4236/me.2018.911115.
Y. Ma, "Prediction of Default Probability of Credit-Card Bills," Open J. Bus. Manag., vol. 08, no. 01, pp. 231–244, 2020, doi: 10.4236/ojbm.2020.81014.
A. Bindal and S. Chaurasia, "Predictive risk analysis for loan repayment of credit card clients," 2018 3rd IEEE Int. Conf. Recent Trends Electron. Inf. Commun. Technol. RTEICT 2018 - Proc., pp. 2508–2513, 2018, doi: 10.1109/RTEICT42901.2018.9012366.
A. Lawi and F. Aziz, "Classification of credit card default clients using LS-SVM ensemble," Proc. 3rd Int. Conf. Informatics Comput. ICIC 2018, pp. 1–4, 2018, doi: 10.1109/IAC.2018.8780427.
A. Subasi and S. Cankurt, "Prediction of default payment of credit card clients using Data Mining Techniques," Proc. 5th Int. Eng. Conf. IEC 2019, pp. 115–120, 2019, doi: 10.1109/IEC47844.2019.8950597.
I. C. Yeh and C. hui Lien, "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients," Expert Syst. Appl., vol. 36, no. 2 PART 1, pp. 2473–2480, 2009, doi: 10.1016/j.eswa.2007.12.020.
N. Sariannidis, S. Papadakis, A. Garefalakis, C. Lemonakis, and T. Kyriaki-Argyro, "Default avoidance on credit card portfolios using accounting, demographical and exploratory factors: decision making based on machine learning (ML) techniques," Ann. Oper. Res., vol. 294, no. 1–2, pp. 715–739, 2020, doi: 10.1007/s10479-019-03188-0.
A. Bacova and F. Babic, "Predictive Analytics for Default of Credit Card Clients," SAMI 2021 - IEEE 19th World Symp. Appl. Mach. Intell. Informatics, Proc., pp. 329–333, 2021, doi: 10.1109/SAMI50585.2021.9378671.
J. T. Han, J. S. Choi, M. J. Kim, and J. Jeong, "Developing a Risk Group Predictive Model for Korean Students Falling into Bad Debt," Asian Econ. J., vol. 32, no. 1, pp. 3–14, 2018, doi: 10.1111/asej.12139.
J. Zurada, "Data Mining Techniques in Predicting Default Rates on Customer Loans," Databases Inf. Syst. II, pp. 285–296, 2002, doi: 10.1007/978-94-015-9978-8_22.
D. J. Hand and W. E. Henley, "Statistical classification methods in consumer credit scoring: A review," J. R. Stat. Soc. Ser. A Stat. Soc., vol. 160, no. 3, pp. 523–541, 1997, doi: 10.1111/j.1467-985X.1997.00078.x.
A. C. Of, T. German, and D. Set, "Rates And Extracting Interpretable Rules In The Credit Scoring Task :," vol. 12, no. 3, pp. 45–54, 2008.
J. G. Proakis and D. G. Monolakis, "Digital signal processing: principles, algorithms, and applications," Pentice Hall, pp. 1–42, 1996.
"Multirate digital signal processing," Signal Processing, vol. 7, no. 2. pp. 199–200, 1984, doi: 10.1016/0165-1684(84)90077-x.
A. Oppenheim and R. Schafer, "Discrete-Time Processing- Second Edition." 1999.
Y. Wang, W. Gan, J. Yang, W. Wu, and J. Yan, "Dynamic curriculum learning for imbalanced data classification," Proc. IEEE Int. Conf. Comput. Vis., vol. 2019-Octob, no. 2, pp. 5016–5025, 2019, doi: 10.1109/ICCV.2019.00512.

ISSN: 2752-6798 (Print) | ISSN 2752-6801 (Online)
https://ecohumanism.co.uk/joe/ecohumanism
DOI: https://doi.org/10.62754/joe.v3i7.4471
T. Ryan Hoens and N. V. Chawla, "Imbalanced datasets: From sampling to classifiers," Imbalanced Learn. Found. Algorithms, Appl., pp. 43–59, 2013, doi: 10.1002/9781118646106.ch3.
[H. Kang, T. Vu, and C. D. Yoo, "Learning Imbalanced Datasets With Maximum Margin Loss," Proc. - Int. Conf. Image Process. ICIP, vol. 2021-Septe, no. NeurIPS, pp. 1269–1273, 2021, doi: 10.1109/ICIP42928.2021.9506389.
K. B. Syariah and G. Ilmu, "No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析 Title," no. september 2016, pp. 1–6.
N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," J. Artif. Intell. Res., vol. 16, no. June 2002, pp. 321–357, 2002, doi: 10.1613/jair.953.
N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "snopes.com: Two-Striped Telamonia Spider," J. Artif. Intell. Res., vol. 16, no. Sept. 28, pp. 321–357, 2002, [Online]. Available: https://arxiv.org/pdf/1106.1813.pdf%0Ahttp://www.snopes.com/horrors/insects/telamonia.asp.
F. Provost and T. Fawcett, "Data Science for Business," no. August 2013, p. 387, 2013.
T. D. Adugna, A. Ramu, and A. Haldorai, A Review of Pattern Recognition and Machine Learning, vol. 4, no. 1. 2024.
M. Borovcnik, H.-J. Bentz, and R. Kapadia, A Probabilistic Perspective. 1991.
M. Learning and A. P. Perspective, "Machine learning : a probabilistic perspective : A probabilistic perspective - DTU Findit," 2012, [Online]. Available: https://research.google/pubs/pub38136/%0Ahttps://books.google.co.uk/books?hl=en&lr=&id=RC43AgAAQBAJ&oi=fnd&pg=PR7&dq=Machine+Learning+-+A+Probabilistic+Perspective+(2012)&ots=umlueCPx-7&sig=Fez88u5ceM3dJjO9DGl3dugeR6Y#v=onepage&q=Machine Learning - A Prob.
S. Cycles, "Chapter 9 Chapter 9," Cycle, vol. 1897, no. Figure 1, pp. 44–45, 1989, doi: 10.1007/0-387-25465-X.
M. Ramoni and P. Sebastiani, "Robust Bayes classifiers," Artif. Intell., vol. 125, no. 1–2, pp. 209–226, 2001, doi: 10.1016/S0004-3702(00)00085-0.
S. Tschiatschek, K. Paul, and F. Pernkopf, "Integer Bayesian network classifiers," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8726 LNAI, no. PART 3, pp. 209–224, 2014, doi: 10.1007/978-3-662-44845-8_14.
J. Mendling, L. Sánchez-González, F. García, and M. La Rosa, "Thresholds for error probability measures of business process models," J. Syst. Softw., vol. 85, no. 5, pp. 1188–1197, 2012, doi: 10.1016/j.jss.2012.01.017.
G. Heinze and M. Schemper, "A solution to the problem of separation in logistic regression," Stat. Med., vol. 21, no. 16, pp. 2409–2419, 2002, doi: 10.1002/sim.1047.
G. James, DanielaWitten, T. Hastie, and R. Tibshirani, Springer Texts in Statistics An Introduction to Statistical Learning wth application in R. 2013.
Second Edition. .
Z. Jin, J. Shang, Q. Zhu, C. Ling, W. Xie, and B. Qiang, "RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 12343 LNCS, pp. 503–515, 2020, doi: 10.1007/978-3-030-62008-0_35.
M. Doumpos, C. Zopounidis, and V. Golfinopoulou, "Additive support vector machines for pattern classification," IEEE Trans. Syst. Man, Cybern. Part B Cybern., vol. 37, no. 3, pp. 540–550, 2007, doi: 10.1109/TSMCB.2006.887427.
H. Kinsley and D. Kukieła, "Neural Networks from Scratch in Python," p. 658, 2020, [Online]. Available: https://www.youtube.com/playlist?list=PLQVvvaa0QuDcjD5BAw2DxE6OF2tius3V3%0Ahttps://nnfs.io/.
J. M. Czum, "Dive Into Deep Learning," J. Am. Coll. Radiol., vol. 17, no. 5, pp. 637–638, 2020, doi: 10.1016/j.jacr.2020.02.005.
A. I. A. Without, "AI Applications Without a PhD for Coders with."
B. Gavranović, "Fundamental Components of Deep Learning: A category-theoretic approach," 2024, [Online]. Available: http://arxiv.org/abs/2403.13001.
H. Wang, R. Czerminski, and A. C. Jamieson, "Neural Networks and Deep Learning," Mach. Age Cust. Insight, pp. 91–101, 2021, doi: 10.1108/978-1-83909-694-520211010.
J. Kwaku et al., "A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions," Decis. Anal. J., vol. 6, no. November 2022, p. 100163, 2023, doi: 10.1016/j.dajour.2023.100163.
D. Filters, "A Study of Image Upsampling and," 2019.
I. Downsampling, U. Methods, and A. Youssef, "Analysis and Comparison of Various," vol. 1, no. November, p. 1, 2002.
C. X. Ling and V. S. Sheng, "Cost-Sensitive Learning and the Class Imbalance Problem," Encycl. Mach. Learn., pp. 231–235, 2008, [Online]. Available: http://www.springer.com/computer/ai/book/978-0-387-30768-8%5Cnhttp://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.164.4418&rep=rep1&type=pdf.
S. Kiyohara, T. Miyata, and T. Mizoguchi, "Prediction of grain boundary structure and energy by machine learning," vol. 18, pp. 1–5, 2015, [Online]. Available: http://arxiv.org/abs/1512.03502.
A. Holden, B. Sammler, B. Powers, and S. Schmidt, Geometric Properties. 2020.
J. Cui et al., "Region Rebalance for Long-Tailed Semantic Segmentation," 2022, [Online]. Available: http://arxiv.org/abs/2204.01969.
D. W. Hoffman, No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析 Title. .
A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, Data Level Preprocessing Methods. 2018.
H. He and E. A. Garcia, "Learning from imbalanced data," IEEE Trans. Knowl. Data Eng., vol. 21, no. 9, pp. 1263–1284, 2009, doi: 10.1109/TKDE.2008.239.
R. Blagus and L. Lusa, "SMOTE for high-dimensional class-imbalanced data," BMC Bioinformatics, vol. 14, 2013, doi: 10.1186/1471-2105-14-106.
T. M. Alam et al., "An Investigation of Credit Card Default Prediction in the Imbalanced Datasets," vol. 8, pp. 201173–201198, 2020, doi: 10.1109/ACCESS.2020.3033784.