Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 60(4): 1-12 (2023)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
https://doi.org/10.53560/PPASA(60-4)831

Pakistan Academy of Sciences

Review Article

A Supervised Machine Learning Algorithms: Applications, Challenges, and Recommendations

Aqib Ali[1,2*], and Wali Khan Mashwani[3]

[1]Department of Computer Science, Concordia College Bahawalpur, Pakistan
[2]Key Laboratory Measurement and Control of CSE Ministry of Education, School of Automation, Southeast University, Nanjing 210002, China
[3]Institute of Numerical Sciences, Kohat University of Science & Technology, Kohat 26000, Pakistan

Abstract: Machine Learning (ML) is an advanced technology that empowers systems to acquire knowledge autonomously, eliminating the need for explicit programming. The fundamental objective of the machine learning paradigm is to equip computers with the ability to learn independently without human intervention. In the literature, categorization in data mining has received a lot of traction, with applications ranging from health to astronomy and finance to textual classification. The three learning methodologies in machine learning are supervised, unsupervised, and semi-supervised. Humans must give the appropriate input and output and offer feedback on the prediction accuracy throughout the training phase for supervised algorithms. Unsupervised learning methods differ from supervised learning methods because they do not require any training. However, supervised learning methods are more accessible to implement than unsupervised learning methods. This study looks at supervised learning algorithms commonly employed in data classification. The strategies are evaluated based on their objective, methodology, benefits, and drawbacks. It is anticipated that readers will be able to understand the supervised machine learning techniques for data classification.

Keywords: Machine Learning, Supervised Learning, Classification, Supervised Algorithms.

1. INTRODUCTION

Machine learning (ML) is a broad term that encompasses computer science, statistics, probability, artificial intelligence, psychology, neuroscience, and various other fields [1, 2]. Problems may be solved using ML by simply creating a model that is a good representation of a given dataset. ML has enhanced the study of statistics by establishing fundamental computational statistical theories of learning processes [3]. It has also elevated the subject of statistics into a broad discipline by teaching computers to emulate the human brain. ML allows a system to learn and improve without being explicitly designed. ML algorithms are advantageous when explicitly coding for fast-speed performance is impractical. *Sorting numbers* is a significant task that involves providing a set of numbers as input and obtaining a sorted list as the output. We know input values and the appropriate algorithm to achieve the desired outcome [4]. However, specific tasks, like email filtering to differentiate between legitimate and spam messages, can be perplexing. Although we comprehend the required input and anticipated true or false output, the precise instructions guiding the programmer in executing these tasks are yet to be determined [5]. In unusual instances when there is no precise technique to accomplish success, we rely on data and direct the computer to evaluate and make intelligent sense of the data. Consider computer software that learns to identify and forecast cancer based on medical research reports from a patient. Your performance will improve as you experience interpreting medical research papers from a more comprehensive patient group [6]. Correct

Received: July 2023; Accepted: December 2023
*Corresponding Author: Aqib Ali <aqibcsit@gmail.com>

predictions and cancer case detections will be counted and evaluated by an experienced oncologist to determine your performance. ML is used in various fields, including robotics, natural language processing, computer games, pattern recognition, etc. [7]. ML methods involve training models on a dataset to learn patterns and subsequently applying these models to a separate test dataset for tasks such as classification or prediction. Figure 1 depicts the process of supervised ML algorithms.

1.1 ML Algorithm Categories

Supervised learning algorithms, supporting vector machines, decision trees, probabilistic summaries, algebraic functions, and other methods can be used to describe classifiers. Classification is one of the most researched models and possibly relevant in practice, alongside regression and probability estimation. Advances in this category have enormous potential advantages since the approach greatly influences other fields, both inside data mining and its applications [8].

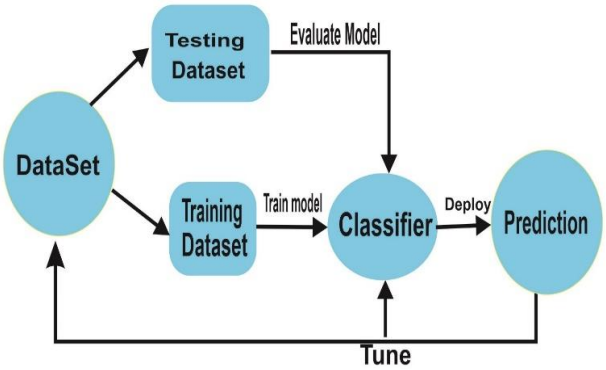

Fig. 1. Supervised Learning Workflow

Unsupervised learning algorithms [9], on the other hand, have no categories. The primary goal of unsupervised learning is to build grading labels automatically. These algorithms seek patterns in the data to see whether they can be categorized and grouped. These groupings represent the whole family of clustered ML approaches, referred to as clusters. The machine does not know how the collections are organized in this unsupervised categorization (cluster analysis). We have a better chance of surprising ourselves if we use cluster analysis. As a result, cluster analysis is a promising method for delving into the correlations between various variables. Figure 2 shows the categories of ML algorithms.

This article depicts many types of supervised ML algorithms and how they may be used to make more efficient judgments and complete tasks more efficiently. This article will demonstrate how different algorithms provide the machine with a varied learning experience and how the machine adopts other things from the environment before making a choice and doing specialized jobs.

2. SUPERVISED LEARNING

In a fundamental machine learning model, the learning process comprises two distinct phases: training and testing. During the training phase, the training data samples serve as inputs for the learning algorithm or learner, allowing it to acquire knowledge and construct the learning model [10]. Subsequently, in the testing phase, the learning model utilizes an execution engine to generate

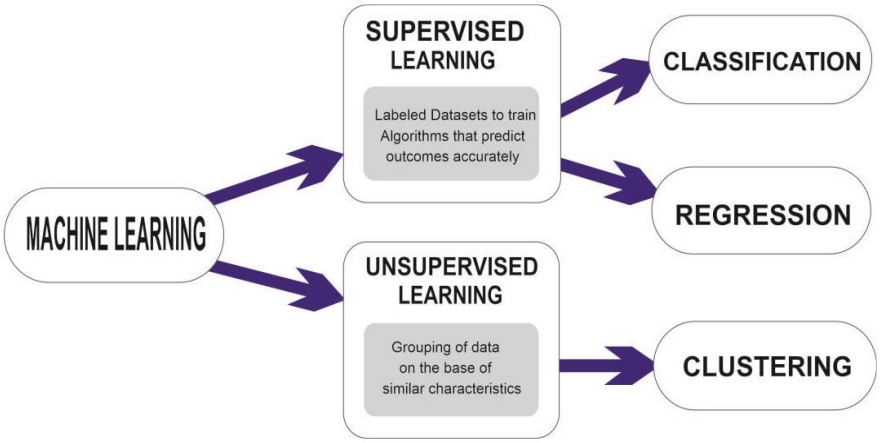

Fig. 2. Machine Learning Algorithm Categories

test or production data predictions. The output of the learning model consists of labeled data, which provides the final forecasts or categorized information. Supervised learning is the most commonly employed approach in classification problems, as it aims to create a classification system based on the constructed machine learning model.

In supervised learning, the input probabilities, such as inputs with known predicted results, are typically unspecified [11]. This process yields a dataset comprising labeled data and corresponding features. The primary objective is to develop an estimator that can determine an item's label based on a set of characteristics. The learning algorithm is supplied with inputs consisting of feature sets and their respective desired outputs. The algorithm learns from errors by comparing its actual output to the correct outputs and adjusts the model accordingly. However, if input values are missing, no reliable conclusions can be drawn about the outputs. Supervised learning is widely employed in training neural networks and decision trees, which rely on labeled data for default categorization [12]. This learning approach is also utilized in programs that predict probable occurrences of functions based on historical data. Its practical applications are vast, including tasks such as inferring the species of an iris flower based on measurements. As mentioned earlier, supervised learning tasks can be categorized into classification and regression. While regression deals with continuous labels, classification involves discrete labels [13].

The approach described in Figure 3 distinguishes between the observed data X, which serves as the training data, and the structured data presented to the model during the training phase. The objective of the supervised learning algorithm is to create a prediction model through this process. Once trained, the fitted model will attempt to predict the most likely labels for a new set of samples X in the test set. Based on the nature of the objective and the type of labels involved, supervised learning can be categorized as follows:

- **Classification:** Classification is termed as such when the objective is to predict the value of y from a predetermined set of classes. The prediction model is trained to assign the most appropriate class label to each input sample.
- **Regression:** The challenge lies in predicting the value y when it consists of floating-point values. The prediction model is designed to estimate continuous numerical values rather than discrete classes.

These two categories, classification, and regression,

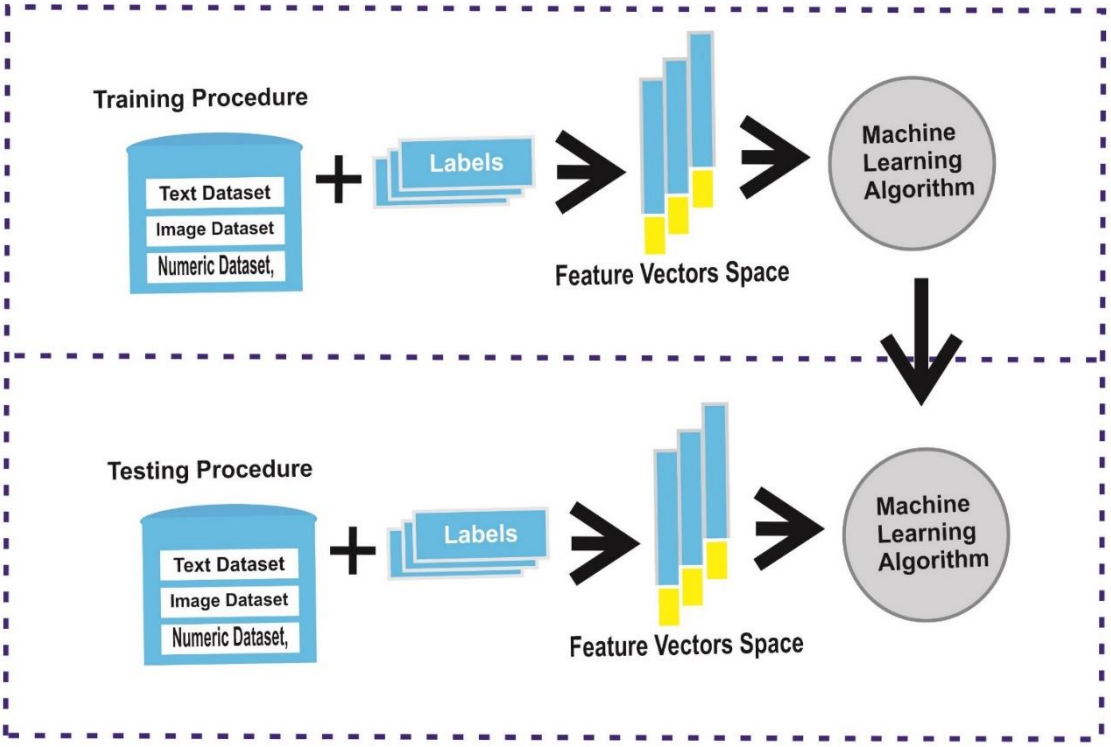

Fig. 3. Supervised Learning Model

capture the primary objectives of supervised learning, depending on the predicted label type.

3. REVIEW ON ML SUPERVISED ALGORITHM

We undertake a thorough analysis of the literature on supervised learning methodologies and algorithms [14-16], performance measures used in supervised learning, and the benefits and drawbacks of eight supervised learning research in this paper, as sown in Figure 4. This research will help academics go in a new direction by identifying new research areas and filling a research vacuum in supervised learning.

3.1 Bayesian Network

The non-dependence on functions of the BN classifier is a crucial property. Another factor to consider is that all operations are interdependent. Returns the BN model as a directed acyclic graph with random variables and conditional dependencies as nodes and edges. It is assumed to be a complete model for the variable and its connection. As a result, a total joint probability distribution (JPD) is defined for all variables in a model, as shown in Figure 5 [17].

3.1.1 Advantages

In Bayesian learning, a priori probability distribution is initially selected and subsequently updated to generate a posteriori distribution. As additional observations become available, the a posteriori distribution can be utilized as a new priori distribution for further updates. The Bayesian network can deal with incomplete datasets. The strategy can prevent data from being over-fitted. It is not essential to eliminate the data's inconsistencies [18].

3.1.2 Disadvantage

The following are some of the drawbacks of Bayesian learning: preselection is a challenging task. The anterior distribution might have an important influence on the posterior distribution. If the preceding option is wrong, the forecasts will be erroneous. It has the potential to be computationally demanding. Text extraction is not suggested since the BN classifier is computationally demanding [19].

3.1.3 Applications

Bayesian learning may be utilized in various areas, including medical diagnosis and disaster victim identification.

3.2 Naive Bayes

Naïve Bayes (NB) classification [20] is a supervised learning method that shares similarities with the

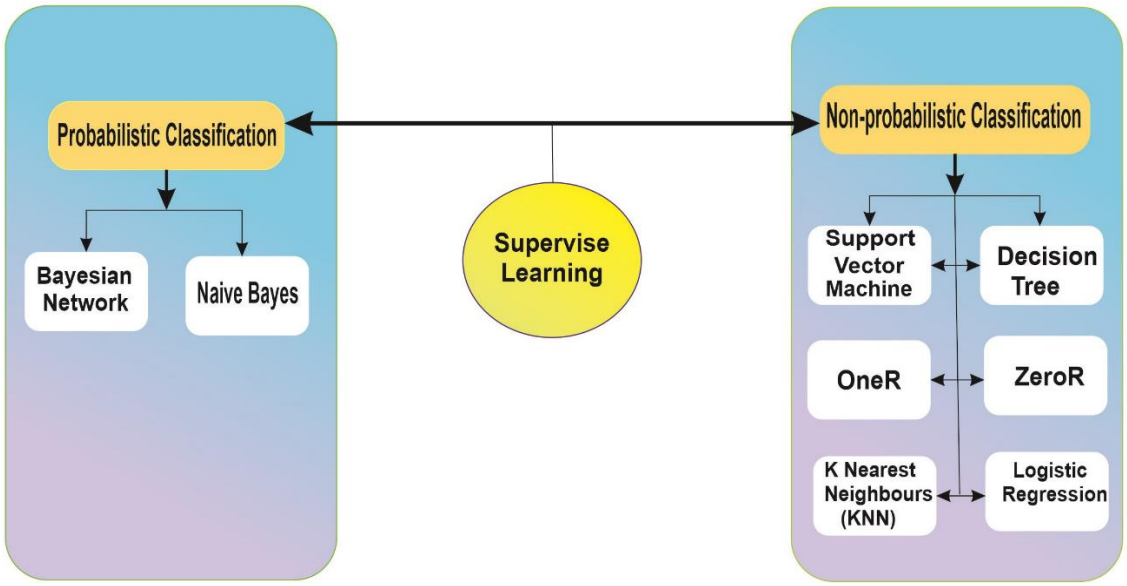

Fig. 4. Supervised Learning Algorithm Review

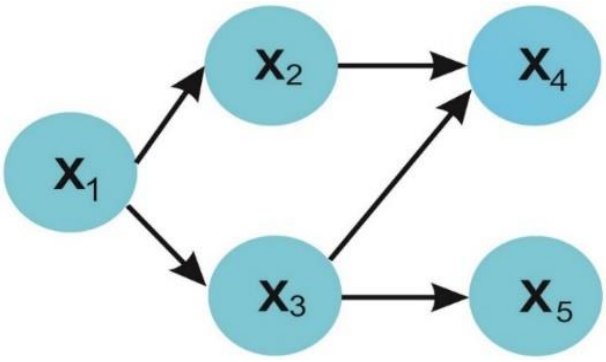

Fig. 5. Visualization and Workflow of a Bayesian Network

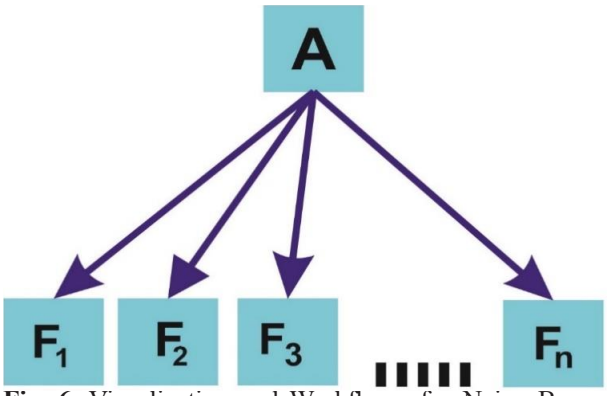

Fig. 6. Visualization and Workflow of a Naive Bayes Algorithm

statistical approach to classification. It is based on a fundamental probabilistic model, which allows for representing uncertainty in the model by computing outcome probabilities. The primary objective of Bayesian classification is to address prediction tasks, and it incorporates observed data while employing effective learning techniques. One of the advantages of Bayesian classification is that it can facilitate the understanding and evaluation of learning algorithms [21]. The graphical representation and primary usage of the NB classifier can be seen in Figure 6. By calculating explicit probabilities for the assumptions and utilizing incoming data to refine the model, the NB classification approach aims to make accurate predictions.

In the context of a two-valued generic probability distribution, without sacrificing generality, we can derive the following equation using Bayes' rule:

$$P(y_1, y_2) = P(y_1 \mid y_2)P(y_2) \quad (1)$$

If another variable of class c is present, we get the following equation:

$$P\,(y_1 \mid y_2 \mid c) \;=\; P\,(y_1 \mid y_2, c)\,(y_2 \mid c)\,P \quad (2)$$

When we extend the scenario to include two variables and assume conditional independence for a collection of variables Y1, . . ., Yn that depends on another variable c, we obtain the following equation:

$$P(y \mid c) = \prod_{i=i}^{N} P(y_i \mid c) \quad (3)$$

3.2.1 Advantages

Naive Bayes (NB) offers several benefits, including its straightforward implementation, efficient performance, and ability to achieve reliable results with smaller training datasets. The growth of predictors and data points remains linear, scaling appropriately with their numbers. NB is versatile, accommodating binary and multi-class classification tasks while generating probabilistic predictions. Furthermore, it handles many data types, encompassing continuous and discrete variables [22].

3.2.2 Disadvantages

The drawbacks of Naive Bayes include its inherent simplicity, which makes it prone to being outperformed by more complex and accurately fitted models. Directly using Naive Bayes becomes challenging when dealing with features like continuous variables (e.g., time). Although approximations like creating "cubes" for continuous variables are possible, they are not entirely accurate. Moreover, the absence of an online Naive Bayes variant necessitates saving all data for retraining, which can be impractical. The scalability of Naive Bayes is limited, especially when the number of classes is substantial, exceeding 100,000. In such cases, resizing becomes problematic. Even for predictions, alternatives like Support Vector Machines (SVM) or basic logistic regression tend to consume more runtime RAM. Additionally, computations can be time-consuming, particularly for models with many variables. [23].

3.2.3 Applications

Naive Bayes may be used in seed classifications, tumor classification, and stock exchange predictions.

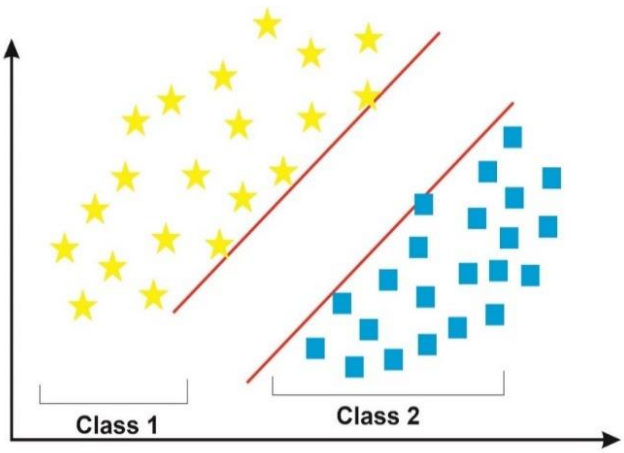

Fig. 7. Visualization and Workflow of a Support Vector Machine

3.3 Support Vector Machine (SVM)

The Supporting Vector Machines (SVM) is a sort of algorithm that can solve classification and regression issues. The hyper-plane, which serves as the decision boundary, must be defined in this technique. When a collection of components from distinct classes has to be separated, judgments must be made. Whether or not items can be divided linearly, complicated mathematical functions known as kernels are necessary to separate objects from distinct classes, as shown in Figure 7. SVM is used to properly categorize things using examples from the training dataset [24].

3.3.1 Advantages

SVM offer several advantages. They are adept at handling both structured and semi-structured data, particularly when a suitable kernel function can be determined. It is scalable with massive datasets and does not get trapped in local optimality.

3.3.2 Disadvantage

The following are some of SVM's drawbacks: with a vast data collection, its performance suffers because of the bigger training time. The proper kernel function will not be easy to find. The SVM does not perform well on noisy datasets. SVM does not offer probability estimates. The final SVM model is hard to understand.

3.3.3 Applications

The final SVM model is challenging to comprehend. Cancer diagnosis, credit card fraud finding, handwriting recognition, face recognition, and text categorization are all support vector machine applications.

3.4 Decision Tree

A decision tree serves as a recursive partitioning of the instance space, functioning as a classifier [25]. It comprises various nodes, with the root node being

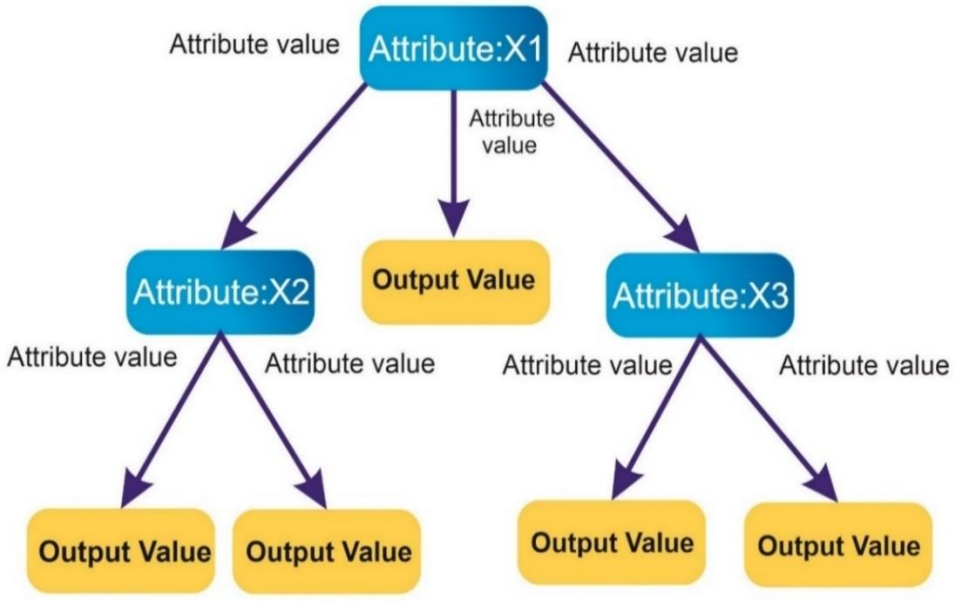

Fig. 8. Visualization and Workflow of a Decision Tree Algorithm

the primary node and the remaining nodes forming the tree structure. Internal or test nodes possess outgoing branches, while leaves represent terminal nodes. In a decision tree, each test node divides the instance space into multiple subspaces based on discrete functions of the input values. Typically, a single attribute is considered in each test, leading to the division of the instance space according to attribute values. For numerical properties, ranges are used as criteria. Each leaf node is assigned to the class with the highest objective value, and the likelihood of a specific target property value can be represented through a probability vector. Based on the test results, instances are categorized along the path from the root to the leaf node. The practical utility of decision trees is demonstrated in Figure 8, where each node is labeled with the checked attribute, and the corresponding attribute values are marked on the branches. By employing this classifier, analysts can predict the responses of elite prospects and analyze the behavioral characteristics of the entire prospect population [26].

Mathematically, decision trees can be understood as collections of hyperplanes orthogonal to the axes, particularly in the case of numeric attributes. Decision-makers often prefer less complex decision trees as they are perceived to be more comprehensive.

3.4.1 Advantages

Here are some advantages of using a Decision Tree:

- Suitable for Classification and Regression: Decision Trees are well-suited for classification and regression problems. They can effectively handle tasks that require predicting categorical outcomes or estimating numerical values.
- Interpretability: Decision Trees offer a straightforward and intuitive interpretation. The tree structure allows for clear visibility into the decision-making process, making it easier to understand and explain the reasoning behind predictions.
- Handling Categorical and Quantitative Values: Decision Trees can handle categorical and quantitative input variables. They can accommodate various data types without extensive data preprocessing or feature engineering.
- Efficient Performance: Decision Trees often demonstrate good performance due to the efficiency of the tree traversal technique. Navigating through the tree to make predictions is computationally efficient, enabling quick decision-making for new instances.

Additionally, when it comes to addressing the overfitting problem, Decision Trees can benefit from ensemble modeling techniques. Random Forest, in particular, is a popular solution that utilizes multiple Decision Trees to mitigate overfitting issues [27].

3.4.2 Disadvantages

The decision tree has the drawbacks of being unbalanced, hard to regulate the scope of the tree, prone to sampling mistakes, and providing a locally optimum answer rather than an overall optimal result [28].

3.4.3 Applications

Decision trees may be utilized in various applications, including forecasting future library book usage.

3.5 OneR

"A One-R rule" is a ranking method that generates a unique rule for each of the predictors in the data before selecting the "unique rule" with the fewest overall error. It's a straightforward and accurate algorithm. Create a frequency table for each predictor against their objectives to develop a rule for the estimator. OneR is virtually as accurate as state-of-the-art classification algorithms and is simple to understand [29].

- The One-R algorithm counts the frequency of each target value first (class).
- Choose the category more frequently.
- The predictor is assigned to that class by the rule.
- The total inaccuracy of the regulations for each estimator is then calculated.
- Choose the estimate with the minor gross error.

3.5.1 Advantages

High performance on non-linear problems, not affected by outliers, and not overfitting sensitive. Effective, unaffected by outliers, work on non-linear problems and takes a probabilistic approach [30].

3.5.2 Disadvantage

The assessment tables (Gain, Lift, K-S, and ROC) are not applicable because OneR does not provide a score or probability [30].

2.5.3 Applications

OneR, known for its simplicity and interpretability, is versatile in educational contexts, teaching fundamental ML concepts and serving as a foundational model for complex classification tasks.

3.6 ZeroR

It is one of the most basic classification approaches since it is goal-oriented and ignores all predictors. In basic terms, the "zero rule" categorization technique asks you to predict which class has the most members. The "zero rule" can be used to analyze the performance of a benchmark as the baseline for specific different ranking techniques, albeit it is not predictive. The Zero-R technique is straightforward: construct a table of frequencies and select the most frequent value [31].

3.6.1 Advantages

ZeroR is the most straightforward goal-based classification algorithm that ignores all predictors. The majority category is anticipated by the ZeroR classifier (class) even though ZeroR has no predictive power [31].

3.6.2 Disadvantage

The assessment tables (Gain, Lift, K-S, and ROC) are not applicable because ZeroR does not provide a score or probability.

3.6.3 Applications

It helps to establish a benchmark's performance for other ranking systems.

3.7 K Nearest Neighbor

The K Nearest Neighbor (KNN) technique is used to classify data. The method attempts to classify the supplied sample data point as a classification problem utilizing a database with data points sorted into multiple groups, as shown in Figure 9. Because KNN assumes no underlying data distribution, it is called non-parametric [32].

3.7.1 Advantages

Here are several benefits of the KNN algorithm: it offers a straightforward approach that is easy to implement. Constructing the model is cost-effective. KNN is a highly flexible classification system that performs well with multimodal classes. It can handle various class labels within the records. While its error rate can be double that of Bayesian error, KNN is often considered the most effective method. In predicting protein function based on the expression profile, KNN has been found to outperform SVM. [33].

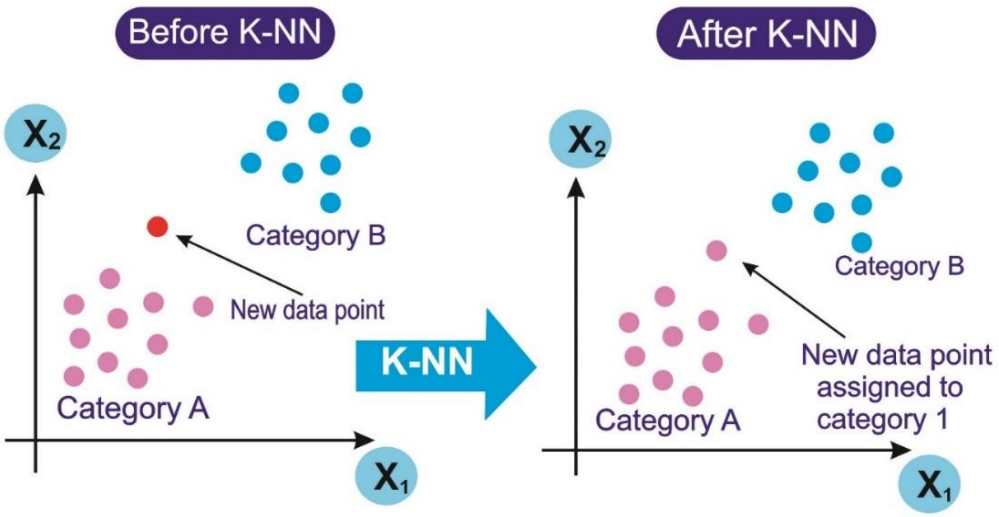

Fig. 9. Visualization and Workflow of a K Nearest Neighbor

3.7.2 Disadvantage

The following are some of KNN's drawbacks: the cost of classifying unknown records is relatively high. Calculation of the nearest k-neighbor distance is required. The method becomes more computationally costly as the training set size grows. Accuracy will suffer as a result of noisy or irrelevant attributes. It is worth noting that KNN can be computationally expensive, particularly for large datasets. Despite these drawbacks, KNN is a method that preserves all training data and makes no assumptions about it. This characteristic allows it to handle enormous datasets, albeit with the trade-off of potentially costly computations.

3.7.3 Applications

The precision of the areas will be reduced as the data becomes more dimensional. KNN has applications in recommendation systems, medical analysis of different diseases with similar symptoms, credit rating based on resemblance of traits, and handwriting recognition [33].

3.8 Logistic Regression

It's a tried-and-true method of resolving binary classification issues. The likelihood that a result has just two values is predicted using logistic regression. The logistic function, an S-shaped curve that takes any number with an accurate weight and transfers it to a value between 0 and 1, but never precisely between 0 and 1, is at the heart of logistic regression. [34].

The logistic regression utilizes data to fit a logistic function and calculate the probability that an event will occur [35]. Logistic regression uses several predictive variables, some of which may be numerical or categorical, like many other forms of regression analysis. As stated by the logistic regression hypothesis:

$$g(y) = g(\theta^T y) \tag{4}$$

where g is a sigmoid function defined as follows:

$$g(x) = \frac{1}{1+e^{-x}} \tag{5}$$

As shown in Figure 10, the sigmoid function has

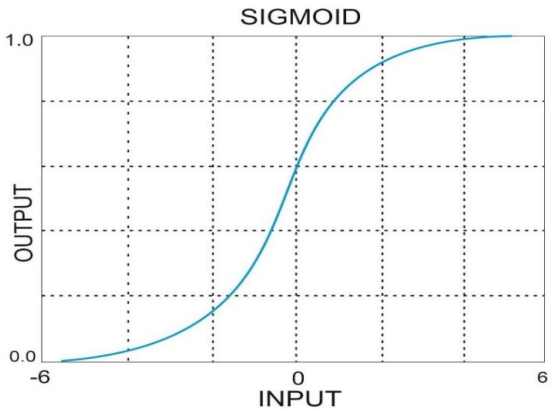

Fig. 10. The logistic function as a visual depiction

specific features that result in the range [0.1] values. The logistic regression cost function is as follows:

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left[-z^{(i)}\log\left(g_\theta\left(y^{(i)}\right)\right) - \left(1-z^{(i)}\right)\log\left(1-g_\theta\left(y^{(i)}\right)\right)\right] \tag{6}$$

3.8.1 Advantages

Implementation ease, computational efficiency, training efficiency, and regularization ease are only a few of the benefits of logistic regression. There is no need to scale the input functions. This strategy is mainly utilized in the industry to address large-scale difficulties. Because the logistic regression output is a probability score, using it to solve a business problem necessitates the creation of unique performance indicators to obtain a breakpoint that can be used to prioritize the objective. Furthermore, modest data noise and multicollinearity have little effect on logistic regression [36].

3.8.2 Disadvantage

The following are some of the drawbacks of logistic regression: because its decision surface is linear and vulnerable to over-fitting, it cannot address nonlinear issues. It will not operate successfully until all independent variables are known.

3.8.3 Applications

Practical uses of Logistic Regression include predicting the risk of acquiring the disease and diagnosing cancer.

4. COMPARATIVE ANALYSIS

We present a comparison of the most popular

Table 1: A Comparison of Eigth Commonly Used Supervised Classification Algorithms

	Naive Bayes	Decision Tree	Logistic Regression	Support Vector Machine	K Nearest Neighbor	Bayesian Network	OneR	ZeroR
Accuracy in General	Satisfactory	Good	Good	Superb	Good	Excellent	Good	Good
Speed of Learning	Superb	Excellent	Satisfactory	Satisfactory	Superb	Superb	Excellent	Satisfactory
Speed of Classification	Superb	Superb	Good	Superb	Satisfactory	Superb	Superb	Good
Tolerance to Missing Values	Superb	Excellent	Superb	Good	Satisfactory	Excellent	Good	Good
Tolerance to Irrelevant Attributes	Good	Excellent	Good	Superb	Good	Good	Excellent	Excellent
Tolerance to Redundant Attributes	Satisfactory	Good	Excellent	Excellent	Good	Good	Satisfactory	Satisfactory
Tolerance to Highly Interdependent Attributes	Satisfactory	Good	Excellent	Excellent	Satisfactory	Good	Good	Satisfactory
Tolerance to Noise	Excellent	Good	Satisfactory	Good	Satisfactory	Good	Satisfactory	Good
Dealing with Danger of Overfitting	Excellent	Good	Superb	Good	Excellent	Superb	Satisfactory	Satisfactory
Attempts for Incremental Learning	Superb	Good	Superb	Good	Superb	Good	Good	Satisfactory
Transparency of Knowledge/ Classification	Superb	Superb	Good	Satisfactory	Good	Superb	Satisfactory	Good
Support Multiclassi-fication	Naturally Extended	Superb	Good	Binary Classifier	Superb	Excellent	Good	Satisfactory

supervised classification algorithms. Several strategies have been created, some of which have been addressed in earlier sections. Based on available facts and theoretical studies, Table 1 compares various regularly used supervised algorithms. This comparison demonstrates that no single learning algorithm beats the others.

5. CONCLUSIONS

This article aimed to look at some of the most often used supervised machine learning methods for regression, classification, and clustering. The benefits and drawbacks of various algorithms were highlighted, as in terms of performance, learning speed, and other factors, several algorithms are compared (where possible). In addition, examples of how these algorithms may be used in the real world were addressed. Machine learning approach such as supervised learning was considered. It is hoped that by recognizing the various machine learning algorithm possibilities, subsequently, in the precise troubleshooting situation, selecting the appropriate machine learning algorithm, the readers would be able to make an educated decision.

6. CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

7. REFERENCES

1. B. Dou, Z. Zhu, E. Merkurjev, L. Ke, L. Chen, J. Jiang, and G.W. Wei. Machine learning methods for small data challenges in molecular science. *Chemical Reviews* 123(13): 8736-8780 (2023).
2. A. Ali, S. Naeem, S. Anam, and M.M. Ahmed. A Supervised Machine Learning Algorithms: Applications, Challenges, and Recommendations. *International Journal of Computing and Digital Systems* 13(1): 1-10 (2023).
3. M. Zubair. Machine Learning Based Biomedical Image Analysis and Feature Extraction Methods. *Journal of Applied and Emerging Sciences* 13(1): 31-39 (2023).
4. J.G. Greener, S. M. Kandathil, L. Moffat, and D.T. Jones. A guide to machine learning for biologists. *Nature Reviews Molecular Cell Biology* 23(1): 40-55 (2022).
5. N. Burkart, and M.F. Huber. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research* 70: 245-317 (2021).
6. J. Ranjan, and C. Foropon. Big data analytics in building the competitive intelligence of organizations. *International Journal of Information Management* 56: 102231 (2021).
7. X. Xiang and S. Foo. Recent Advances in Deep Reinforcement Learning Applications for Solving Partially Observable Markov Decision Processes (POMDP) Problems: Part 1—Fundamentals and Applications in Games, Robotics and Natural Language Processing. *Machine Learning and Knowledge Extraction* 3(3): 554-581 (2021).
8. T. Jiang, J.L. Gradus, and A.J. Rosellini. Supervised machine learning: a brief primer. *Behavior Therapy* 51(5): 675-687 (2020).
9. A. Roohi, K. Faust, U. Djuric, and P. Diamandis. Unsupervised machine learning in pathology: the next frontier. *Surgical Pathology Clinics* 13(2): 349-358 (2020).
10. Y. Liu, S. Arunachalam, and K. Temme. A rigorous and robust quantum speed-up in supervised machine learning. *Nature Physics* 17(9): 1013-1017 (2021).
11. I.H. Sarker. Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science* 2(3): 1-21 (2021).
12. S. Laghmati, B. Cherradi, A. Tmiri, O. Daanouni, and S. Hamida. Classification of patients with breast cancer using neighbourhood component analysis and supervised machine learning techniques. *3rd International Conference on Advanced Communication Technologies and Networking (CommNet)* pp. 1-6, IEEE (September 2020).
13. L.J. Muhammad, E.A. Algehyne, and S.S. Usman. Predictive supervised machine learning models for diabetes mellitus. *SN Computer Science* 1(5): 1-10 (2020).
14. A. Rawson, and M. Brito. A survey of the opportunities and challenges of supervised machine learning in maritime risk analysis. *Transport Reviews* 43(1): 108-130 (2023).
15. J.A. Pruneski, A. Pareek, K.N. Kunze, K.R. Martin, J. Karlsson, J.F. Oeding, and R.J. Williams III. Supervised machine learning and associated algorithms: applications in orthopedic surgery. *Knee Surgery, Sports Traumatology, Arthroscopy* 31(4): 1196-1202 (2023).
16. A. Dinesh, S.A. Selvasofia, K.S. Datcheen, and D.R. Varshan. Machine learning for strength evaluation of concrete structures–Critical review. *Materials Today: Proceedings* (2023).
17. S. Gil-Begue, C. Bielza, and P. Larrañaga. Multi-dimensional Bayesian network classifiers: A survey. *Artificial Intelligence Review* 54(1): 519-559 (2021).
18. G.A. Ruz, P.A. Henríquez, and A. Mascareño. Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers. *Future Generation Computer Systems* 106: 92-104 (2020).
19. P. Awasthi, N. Frank, and M. Mohri. On the existence of the adversarial bayes classifier. *Advances in Neural Information Processing Systems* 34: 2978-2990 (2021).
20. Z. Xue, J. Wei, and W. Guo. A real-time Naive Bayes classifier accelerator on FPGA. *IEEE Access* 8: 40755-40766 (2020).
21. K.G. Reddy, and P.S. Thilagam. Naïve Bayes classifier to mitigate the DDoS attacks severity in ad-hoc networks. *International Journal of Communication Networks and Information Security* 12(2): 221-226 (2020).
22. I.E. Tiffani. Optimization of naïve bayes classifier by implemented unigram, bigram, trigram for sentiment analysis of hotel review. *Journal of Soft Computing Exploration* 1(1): 1-7 (2020).
23. A. Kurani, P. Doshi, A. Vakharia, and M. Shah. A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting. *Annals of Data Science* 10(1): 183-208 (2023).
24. M. Tanveer, T. Rajani, R. Rastogi, Y.H. Shao, and M.A. Ganaie. Comprehensive review on twin support vector machines. *Annals of Operations Research*: pp. 1-46 (2022).
25. P. Priyanka, and D. Kumar. Decision tree classifier: A detailed survey. *International Journal of Information and Decision Sciences* 12(3): 246-269 (2020).
26. S.H. Yoo, H. Geng, T.L. Chiu, S.K. Yu, D.C. Cho, J. Heo, and H. Lee. Deep learning-based decision-tree classifier for COVID-19 diagnosis from chest X-ray imaging. *Frontiers in Medicine* 7: 427 (2020).
27. A. Ali, S. Naeem, S. Anam, and M.M. Ahmed. Current State of Artificial Intelligence (AI) in Oncology: A Review. *Current Trends in OMICS* **3**(1): 01-17 (2023).
28. O. Mehrpour, C. Hoyte, F. Goss, F.M. Shirazi, and S. Nakhaee. Decision tree algorithm can determine the outcome of repeated supratherapeutic ingestion

(RSTI) exposure to acetaminophen: review of 4500 national poison data system cases. *Drug and Chemical Toxicology* 46(4): 692-698 (2023).
29. Y.T. Chang, and N.H. Fan. A novel approach to market segmentation selection using artificial intelligence techniques. *The Journal of Supercomputing* 79(2): 1235-1262 (2023).
30. M. Farahani, S.V. Razavi-Termeh, A. Sadeghi-Niaraki, and S.M. Choi. A Hybridization of Spatial Modeling and Deep Learning for People's Visual Perception of Urban Landscapes. *Sustainability* 15(13): 10403 (2023).
31. M.U. Abdulazeez, W. Khan, and K.A. Abdullah. Predicting child occupant crash injury severity in the United Arab Emirates using machine learning models for imbalanced dataset. *IATSS Research* 47(2): 134-159 (2023).
32. P. Cunningham, and S.J. Delany. k-Nearest neighbour classifiers-A Tutorial. *ACM Computing Surveys (CSUR)* 54(6): 1-25 (2021).
33. C. Li, S. Ding, X. Xu, H. Hou, and L. Ding. Fast density peaks clustering algorithm based on improved mutual K-nearest-neighbor and sub-cluster merging. *Information Sciences* 647: 119470 (2023).
34. Q. Li, S. Zhao, S. Zhao, and J. Wen. Logistic Regression Matching Pursuit algorithm for text classification. *Knowledge-Based Systems* 277: 110761 (2023).
35. G. Troiano, L. Nibali, H. Petsos, P. Eickholz, M.H. Saleh, P. Santamaria, and A. Ravidà. Development and international validation of logistic regression and machine-learning models for the prediction of 10-year molar loss. *Journal of Clinical Periodontology* 50(3): 348-357 (2023).
36. C. Hercus, and A.R. Hudaib. Delirium misdiagnosis risk in psychiatry: a machine learning-logistic regression predictive algorithm. *BMC Health Services Research* 20(1): 1-7 (2020).