Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 215 (2022) 422–431

Procedia
Computer Science

www.elsevier.com/locate/procedia

4th International Conference on Innovative Data Communication Technology and Application

A Comparative Analysis of Machine Learning Algorithms for Classification Purpose

Vraj Sheth[a], Urvashi Tripathi[a], Ankit Sharma[a]*

[a]*Electronics and Instrumentation Engineering Department, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India*

Abstract

A few of the popular data-mining techniques are clustering, classification, and association. The classification process simplifies the process of identifying and accessing data. Classification of data is crucial for risk management, compliance, and data security. Classifying data facilitates its search-ability and traceability by categorising the information. Each data mining model has a distinct level of information. The success of a model is solely determined by the datasets being used, as there is no such thing as an excellent or a poor model. As a part of this study, we examine how accurate different classification algorithms are on diverse datasets. On five different datasets, four classification models are compared: Decision tree, SVM, Naive Bayesian, and K-nearest neighbor. The Naive Bayesian algorithm is proven to be the most effective among other algorithms.

Keywords: Naive Bayes; K-Nearest Neighbour; Decision Tree; Support Vector Machine;

1. Introduction

Data is one of most significant, long term performance defining assets of an organisation. The process of predicting outcomes by analysing the anomalies, patterns, and correlations in huge data sets is called Data mining. Classification is termed for a predictive modelling procedure that predicts a class label based on input data. As part of the modelling process, classifiers require a training dataset that contains many inputs and outputs from which to learn. We can compute the best course of approach to map samples of raw data to the predefined class labels using the training data. As a result, the training dataset must be representative of the issue and include a substantial number of sources for each class label.

There is an ensemble of classification approaches for creating a classifier model. Many studies have been conducted to evaluate various algorithms in order to determine which one is the best. The literature study reveals that there is no one solution, as evidenced by the fact that various investigations provide diverse outcomes. One fundamental reason

* Corresponding author. Tel.:+91-787-432-7003.
E-mail address: ankit.sharma@nirmauni.ac.in

10.1016/j.procs.2022.12.044

for this might be that the outcome of these algorithms is influenced by a variety of parameters such as dataset size, application, feature selection, and so on.

For proposed work, we have regarded some of the most prominent classification methods, including Naive Bayesian, K-nearest neighbour, SVM, and Decision Trees over different datasets to obtain a comprehensive understanding of the algorithms ’performance and choosing the most optimum one.

Section 2 outlines previous projects, and Section 3 describes the various classification algorithms. Section 4 discusses the metrics evaluation, and Section 5 covers the datasets adopted for the study. The assessment of the prediction models is covered in Section 6. The results are summarised in Section 9.

2. Related Work

In this domain, a great deal of research has been done using various methodologies. This section seeks to provide a brief analysis of the current classification model research initiatives. The section presents a survey of the literature in the domain of classifier implementation. The first input in this area comes from Matthew Anyanwua (2009) [1] who performed a comparative analysis of three decision tree algorithms namely C4.5, CART and ID3 to internal assessment data to predict students’performance in an examination. The performance is analysed based on accuracy and time taken to derive the tree. C4.5 is proven to be the best of all for small-scale datasets. SPRINT and SLIQ decision tree algorithms are suitable for larger datasets. M. J. Muzammil (2013) [2] proposed a novel approach to compare different classifiers adapted for Statistical IDS whose performance is evaluated over WEKA. The classification algorithms employed were Naïve Bayesian, C4.5 Decision Tree, Decision Table, ZeroR, and OneR. To evaluate the model’s Performance, several performance metrics were used, including True Positive, True Negative, False Positive, False Negative, Model Building Time, and Margin Curve. Anuradha (2015) [3] proposed a novel approach to implementing classification techniques namely the C4.5 (J48) Bayesian classifiers, decision tree technique, the KNN algorithm, and two rule learner algorithms, JRip and OneR to predict and analyse students’ performance in examinations using data mining. The dataset is retrieved from the college database and a structured questionnaire. Bayesian classifiers such as Nave Bayes and BayesNet are proven to perform the best with high accuracy greater than 70% followed by JRip classifier and J48 classifiers. The JRip results in the highest accuracy for the Distinction. R. Muhamedyev (2015) [4] Implemented the learning curves experiment to evaluate which learning rate occurs during machine learning training using Feedforward Artificial Neural Network (ANN), Naïve Bayes and k-Nearest-Neighbors (k-NN). For error detection, accuracy, weighted mean precision and weighted mean recall were used. The information comes from 30 boreholes in the Inkai uranium deposit. All other algorithms are outperformed by the ANN algorithm. Amit Tate (2016) [5] Presented and compared different classification models for disease predictions namely Naive Bayes (NB), Support Vector Machine (SVM) Random Forest (RF). The dataset is classified using Weka. The performances are compared by estimating accuracy, training time, precision, recall. The results are comparable but the random forest algorithm outperforms the other models. Rafet Duriqi (2016) [6] proposed a novel approach to evaluate different classification algorithms such as Random Forest, Naive Bayes, and K * on three different datasets using the WEKA tool. The data is obtained from the UCI Machine Learning repository. The results suggest that the dataset, particularly the quantity of attributes in the dataset, has an impact on a classifier's performance. Wesley Becari (2016) [7] compares the categorisation methods used by the iCub platform's humanoid hand tactile sensors in considerable detail. Support Vector Machines (SVM), k-Nearest Neighbors Classifiers (kNN), and Decision Trees were used as classifiers. The classification with two fingers performed well. With an accuracy of 97.4%, the Gaussian SVM kernel is proven to be the best as it resulted in the highest percentage of correct answers. P. Srikanth (2016) [8] proposed a novel approach to predict Diabetes Disease using Data Mining Techniques of Classification Algorithms namely the Decision Tree Algorithm, Bayes Algorithm, and Rule-based Algorithm. The Classification Algorithm with the applied classification methods results in high accuracy. Archanaa R (2017) [9] Presented a detailed comparative analysis of distinct machine learning algorithms including Bayes classifiers, Rule learning, Decision trees, and Ensemble classifiers. The dataset is acquired from the University of Queen Mary repository. The Ensemble classifiers show the best results with a classification accuracy of over 99%. The Decorate algorithm belonging to the ensemble classifiers performs the best. Saeed M. Alqahtani (2017) [10] Proposed a comparative analysis of four kinds Vraj Sheth Et al. / Procedia Computer Science 00 (2019) 000–000 3 of classification techniques namely Decision tree (DT), OneR, Naive Bayes (NB), and K-nearest neighbor (KNN). The ISCX dataset is employed and Accuracy, Sensitivity, Precision, F-measure and Specificity were estimated to evaluate the best classifier algorithm. All other algorithms are outperformed by the decision tree classifier. Preeti Nair (2017) [11] Put forward a novel approach for a detailed analysis of various classification algorithms such as Decision tree (DT), OneR, Naive Bayes (NB), and K-nearest neighbor (KNN) using binary classification problems. The datasets are derived from the UCI data repository. The model is evaluated with a confusion matrix and Accuracy, Precision, Recall, and Specificity are calculated to assess each model's performance. The Naive Bayesian classification model is observed to achieve a greater number of highest values as per the performance metrics. S. Sharma (2018) [12] Put forward a detailed analysis of different multi-label classification models namely BR, CC, PS, LS, and Random Forest using the MEKA tool. The dataset (multi-label) employed for the study is retrieved from the engineering students' database of a private university. Random forest outperforms all the models with an accuracy of 96%. Muhammad Alghobiri (2018) [13] Implemented a comprehensive approach to compare and analyse several classification algorithms including Decision tree (DT), Naive Bayes (NB) and Support vector machines (SVM). Ten different datasets have been considered for the assessment. The evaluation metrics employed are Accuracy, Precision, and F-Measure. SVM is proven to be the best. Siddhi Velankar (2018) [14] Proposed a comparative analysis of Bayesian regression (BR), Generalised Linear Model (GLM), and Random Forest (RF) along with a combination of five mean normalisation techniques to find out the best possible combination for bitcoin price forecasting. The database is collected from Quandl and CoinmarketCap to retrieve bitcoin values for a five-year frame. Karunya Rathan (2019) [15] proposed a comparative analysis of LR and Decision Trees (DT) for bitcoin price forecasting. Data for the proposed approach was downloaded from quandl.com. LR outperformed Decision Trees by having an accuracy of 97.59%. A. Demir (2019) [16] presented various machine learning algorithms, including artificial neural networks (ANN), long-short term memory (LSTM), decision trees, Naive Bayes (NB), the nearest neighbour algorithm, and support vector machines (SVM). The dataset was retrieved from KAGGLE. The implementation resulted in LSTM outperforming all the other algorithms with an accuracy of 97.2%. Dr. Pasumpon pandian (2019) [17] presented the review on the methods of the big-data-analytics and the machine-learning in the analytics of high voluminous data to extract the valuable and information. Dr. T. Vijaya kumar (2019) [18] proposed the Caps Net based classification system that can be trained using a smaller number of datasets to detect the type of cancerous tumors in brain. Reaz Chowdhury (2020) [19] proposed a unique approach for projecting the closing price of cryptocurrency. Gradient boosted trees (GBT), k-Nearest Neighbor (K-NN), Neural net (NN), Ensemble learning approach, and other machine learning algorithms have all been conducted to a thorough comparison. Coinmarketcap.com provided the data for this analysis. The three approaches outperformed the state-of-the-art models, with an accuracy of 92.4 percent and an RMSE of 0.2 percent for the ensemble learning method. N. N. Qomariyah (2020) [20] used the WEKA data mining tool to implement a pairwise comparison. This is used to build a decision tree for learning user preferences. The performance of the DT's models, J48, ID3, RandomForest, and RandomTree, A 10-fold cross-validation and hold-out technique is used to evaluate it. J48 performed best when the data was split into 65 percent of the training and 35 percent of the test sets. I. S. Balabanova (2020) [21] present a comparative analysis of the indicators in the development of models based on machine learning approaches for detecting RMS noise levels of tones with different frequencies Decision trees, k-NN, and Nave Bayes are implemented with k-NN having accuracy in the range of 89.800% to 91.050%. Decision tree and k-NN perform well with great efficiency. Mayukh Sammadar (2021) [22] carried out a well-framed comparative analysis of many machine learning algorithms with neural network algorithms taken as convolutional neural network (CNN), artificial neural network (ANN) and recurrent neural network (RNN) and supervised learning algorithms like Random Forest (RF) and k-nearest neighbors (k-NN). The dataset is sourced from Kaggle and scaled afterward using MinMaxScaler. CNN outperforms all the other models with higher accuracy and the least loss. RNN also performs finely. The non-deep learning algorithm is observed to be less accurate.

3. Classification Algorithms

This section offers a concise explanation of all of the classification algorithms used in the proposed work.

3.1. Naive Bayesian

The Bayes' Theorem is used to generate naive Bayes classifiers, which are a group of classification methods. It consists of a number of algorithms which all work on the same principle: each pair of features to be categorised is independent. NB (Naive Bayes) uses the Bayes rule as follows:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)} \tag{1}$$

Y is a class variable and X is an n-dimensional dependent feature vector.

$$X = (x_1, x_2, x_3, \ldots, x_n)$$

The class variable(y) has only two outcomes in our case: yes or no. The classification may be multivariate in some circumstances. As a result, we must find the class Y with the highest probability.

$$y = p(y) \prod_{i=1}^{n} p(X|y) \tag{2}$$

The precision of the projected output values is used to define the procedure's error. If the goal values are categorical, the error is presented as an error rate. The error rate is the proportion of times the prediction was wrong. The Bayes error rate is the lowest error rate that any classifier of a random outcome can provide.

Naive Bayes is easy to set up, produces good results, scales proportionally with the number of predictors and data points, requires less training data, manages discrete and continuous data, can tackle binary and multi-class classification problems, and makes stochastic recommendations. Data can be processed in a continuous or discontinuous manner. It is unaffected by non-essential features. Naive Bayes assumes conditional independence, means the relationship between all input features are independent.

The following are the drawbacks of naive Bayes: Naive Bayes models are too simplistic, models that have been properly trained and optimised often outperform them. If one of the features is required to be a "continuous variable" (such as time), it is complicated to implement Naive Bayes effectively. Even if "buckets" for "continuous variables" can be created, they are not 100% accurate. Because there is no genuine online option for Naive Bayes, all data must be saved in order to retrain the model. When the number of attributes is really high, such as > 100K, it will not scale. In comparison to SVM or simple logistic regression, it requires higher runtime memory for prediction. It consumes much time to compute, especially for models with a lot of variables.

3.2. Decision Tree

A Decision Tree is a supervised learning technique that can be used to perform classification and regression tasks, while it is most typically employed for classification.

A decision tree has a root node, branch nodes, and leaf nodes, similar to a tree, with each node representing a characteristic or attribute, each branch representing a decision or rule, and each leaf representing a result. To split the features, decision tree algorithms are used. At each node, the splitting is tested to see if it is the most suited for the respective classes. A decision tree is a graphical layout that allows you to get all of the various answers for a decision based on the current situation. It only focuses on one question, and the tree is split into subtrees based on the answer.

The following are some of the benefits of using a Decision Tree: It is effective for both regression and classification problems, with ease of interpretation, the ability to fill incomplete data in attributes with the most likely value and handling categorical and quantitative values. It also has a superior productivity due to the efficiency of the tree traversal algorithm. Over-fitting is a problem that Decision Tree may experience, and the answer is Random Forest, which is based on an ensemble modelling technique.

The following are the downsides of using a Decision Tree: it is being unstable, difficult to manage tree size, prone to errors in sampling, and providing a locally optimal answer rather than a globally ideal solution.

3.3. K-Nearest Neighbour

K-nearest neighbours (KNN) are supervised machine learning algorithms that can be utilised to solve both classification and regression problems. With the K-NN model, fresh data can be quickly sorted into well-defined categories. To estimate the values of any new data points, the KNN algorithm makes use of "feature similarity." It evaluates the distances between a query and each example in the data, picks the K examples that are closest to the query, and then selects the label with the highest frequency (in the case of classification) or averages the labels (in the case of regression).

KNN analyses a given test tuple with comparable training tuples in process of learning. An n- dimensional pattern space is used to hold all of the training tuples. A k-nearest-neighbor classifier examines the pattern space for the k training tuples that are nearest to the unidentified tuple when given one. These k training tuples are the unknown tuple's k "nearest neighbours." [2].

Advantages of KNN algorithm are the following: It is a simple technique that may be implemented quickly. It is inexpensive to construct the model. It's a very adaptable categorisation technique that's ideal for Multi-modal classes. There are several class labels on the records. The mistake rate is twice as high as the Bayes error rate. It is sometimes the most effective way. When it came to predicting protein, function based on expression profiles, KNN outperformed SVM.

Disadvantages of KNN are the following: It is relatively costly to classify unknown records. It requires calculating the distance between k-nearest neighbours. The algorithm becomes more computationally costly as the size of the training set grows. Accuracy will degrade as a result of noisy or irrelevant features.

3.4. Support Vector Machine

In Supervised Learning, Support Vector Machines (SVMs) are widely used for dealing with classification and regression problems. The purpose of SVM is to find the optimal line or decision boundary for classifying ndimensional space into sections so that successive data points may be classified conveniently. These boundaries are known as hyperplanes. SVM can handle unstructured, semi structured and structured data. Kernel functions eases the complexities in data type.

This algorithm is divided into two categories: linear data and non-linear data. Mathematical programming and kernel functions are the two main implementations of SVM technology. In a high-dimensional space, the hyperplane divides data points of distinct kinds [4].

SVM has a number of limitations, including the following: Because of the longer training time, it performs poorly when working with large data sets. The correct kernel function will be tough to locate. When a dataset is noisy, SVM does not perform well. Probability calculations are not provided by SVM. It's difficult to interpret the final SVM model.

4. Datasets

The different datasets used for the classification and testing of algorithms are split into sets of test and training models, with 70% as test and 30% as training datasets. The machine learning model is fitted using the train dataset and the test Dataset is used to assess how well a machine learning model fits the data.

1. Placement Dataset

The dataset consists of information including gender, SSC percentage, board of education HSC percentage, specialisation, degree info, work experience, employability test percentage, and salary. This dataset has 15 attributes and 215 entries."Salary" and "ssc_p" are two relevant features for predicting the status of placement for a student, whereas 'workex' and 'specialisation' are two important features for predicting status.

2. Heart Disease Dataset

The dataset consists of information including age, sex, chest pain, restBP, chol, MaxHR, Exang, Oldpeak, Slope, FBS, RestECG, Ca, and thal. This dataset has 15 attributes and 303 entries. This dataset is used to classify whether a person has a certain heart disease or not. It uses attributes such as chest pain and thal, which are later differentiated between sex and thalamic value, which helps in prediction.

3. Wine Quality Dataset

The dataset contains several numerical values that represent information about various wines, such as fixed acidity, residual sugar, pH, sulphates, alcohol chlorides, free sulphur dioxide, volatile acidity, total sulphur dioxide, citric acid, and density. This dataset is also quite large, with over 1000 rows and no null entries. The numerical values in each variable differ substantially, maybe due to the units. If a model is susceptible to these differences, the dataset may need to be standardised. Wines low in both citric acid and alcohol are of poor quality, whereas good quality wine is generally high in both.

4. Glass Quality Dataset

The dataset consists of the refractive index and measures of various elements in the glass, such as ID number, RI: refractive index, Mg: Magnesium, Na: Sodium, Al: Aluminium Si stands for Silicon, K stands for Potassium, Ca stands for Calcium, Ba stands for Barium, and Fe stands for Iron of glass. This dataset has 10 attributes and 214 entries. Using the values of elements present in each glass type, the glass with different element compositions can be predicted.

5. Classification of Jobs

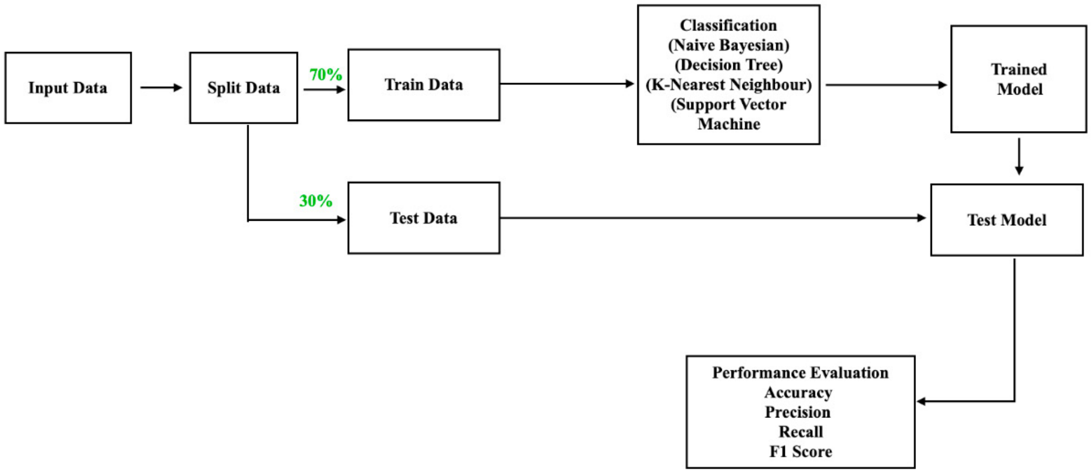
Dataset The dataset mainly has information about a job profile, including ID, JobFamily, JobFamilyDescription, JobClass, JobClassDescription, PayGrade, EducationLevel, Experience, OrgImpact, problem-solving, supervision, contact level, financial budget, PG This dataset has 14 attributes and 66 entries.

5. Analysing the classifiers

Quantitative metrics and qualitative metrics are the two fundamental approaches used for data analysis. In Quantitative Metrics, numerical data constitute the foundation of quantitative information. Ratios, percentages, averages, currency values, and other straightforward expressions of quantitative measures are frequently used.

Whereas, in Qualitative metrics, non-numerical facts constitute the foundation of qualitative information. In a dataset, this inaccuracy is referred to as noise. Any important information's prediction can be greatly impacted by noisy data. Train-Test split and Cross-Validation was used in proposed work.

For proposed work, we chose five data sets at random from Kaggle and ran each dataset through each classifier, comparing the results to see which classifier provided the most accurate value in the majority of cases. For each dataset, a confusion matrix is produced. We calculated and reported each performance measure in the tables using the confusion matrix generated by each classifier. Flowchart for the ML classifier is shown in Figure 1.



.

Fig.1: Flowchart for the Classifier

Table.1, Table.2, Table.3 and Table.4 depicts the classification accuracy, precision, recall and F1 score respectively for each algorithm used on the datasets. The values for each metric evaluation vary as the datasets used are random and have different classes and categories, which generates variation in the output of the algorithm.

Fig.2, Fig.3, Fig.4 and Fig.5 depicts the classification accuracy, precision, recall and F1 score respectively for each algorithm used on the datasets. The proportion of real positive examples among those that the model classified as positive is known as precision. Recall, commonly referred to as sensitivity, is the percentage of positive examples among all the positive ones. The F1-score is a method of integrating the model's recall and precision and is the harmonic mean of the two.All of the classifiers are operating admirably, and the accuracy scores are high. There are relatively few variations between the results obtained. However, when we compared these figures to see which one was the best, we concluded that the Naive Bayes and SVM model produced the best results with the highest accuracy values in two datasets each. Even though k-NN's accuracy values are closer to the high values it does not give the maximum value for any dataset while Decision Tree delivers the highest value for one case. As a result of this observation, we can infer that Naive Bayes and SVM outperform Decision Tree and k-NN

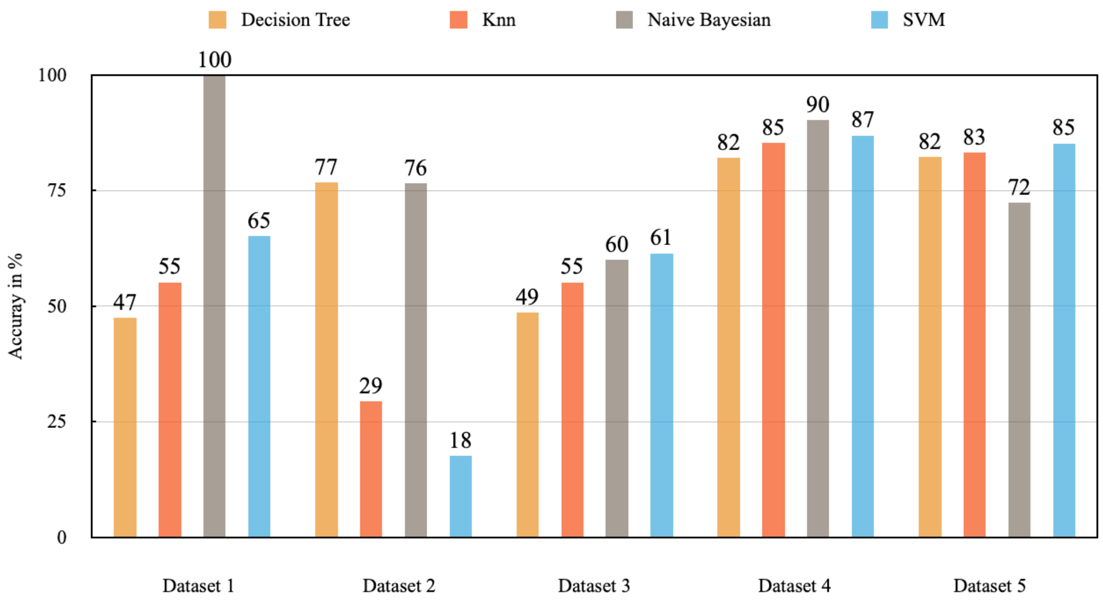
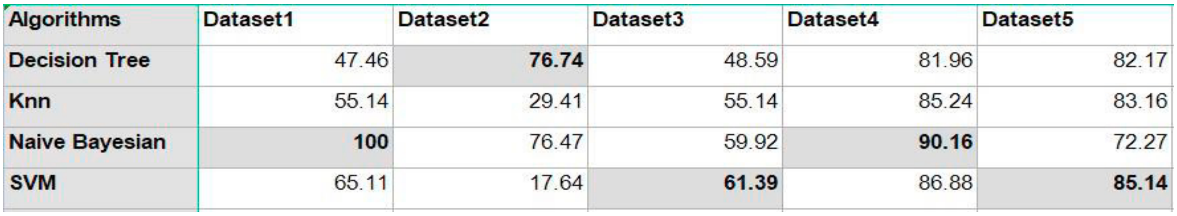


Fig.2: Accuracy (in percentage) of the algorithms applied on each dataset

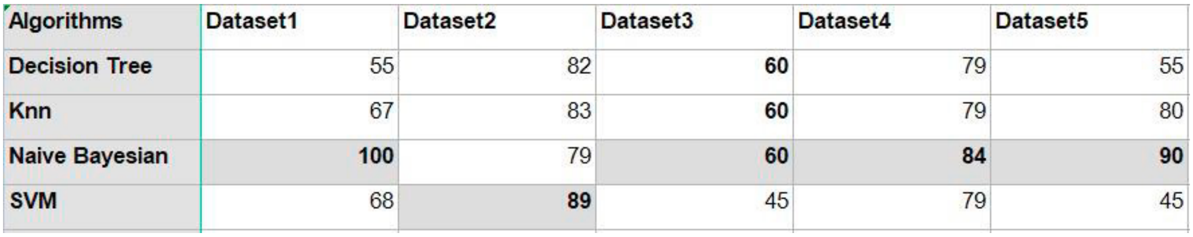
The result shows that the k-NN algorithm has the highest precision values in percentage form, with values that are extremely close to those of Naive Bayes. SVM and Decision Tree perform similarly, however neither produces the highest precision numbers in any circumstance.

Table1: Accuracy (in percentage) of the algorithms applied on each dataset



Algorithms	**Dataset1**	**Dataset2**	**Dataset3**	**Dataset4**	**Dataset5**
Decision Tree	47.46	**76.74**	48.59	81.96	82.17
Knn	55.14	29.41	55.14	85.24	83.16
Naive Bayesian	**100**	76.47	59.92	**90.16**	72.27
SVM	65.11	17.64	**61.39**	86.88	**85.14**

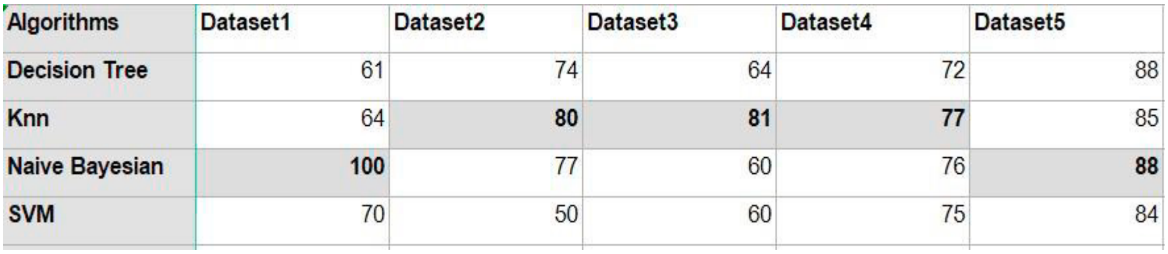
Table 2: Precision (in percentage) of the algorithms applied on each dataset



Algorithms	**Dataset1**	**Dataset2**	**Dataset3**	**Dataset4**	**Dataset5**
Decision Tree	55	82	**60**	79	55
Knn	67	83	**60**	79	80
Naive Bayesian	**100**	79	**60**	**84**	**90**
SVM	68	**89**	45	79	45

The Result shows that the Naive Bayes method performs the best, with the highest recall percentage values, followed by the SVM algorithm, which delivers the highest value for one dataset. With the lowest values for recall percentage, the decision tree works badly.

Table 3: Recall (in percentage) of the algorithms applied on each dataset



Algorithms	Dataset1	Dataset2	Dataset3	Dataset4	Dataset5
Decision Tree	61	74	64	72	88
Knn	64	**80**	**81**	**77**	85
Naive Bayesian	**100**	77	60	76	**88**
SVM	70	50	60	75	84

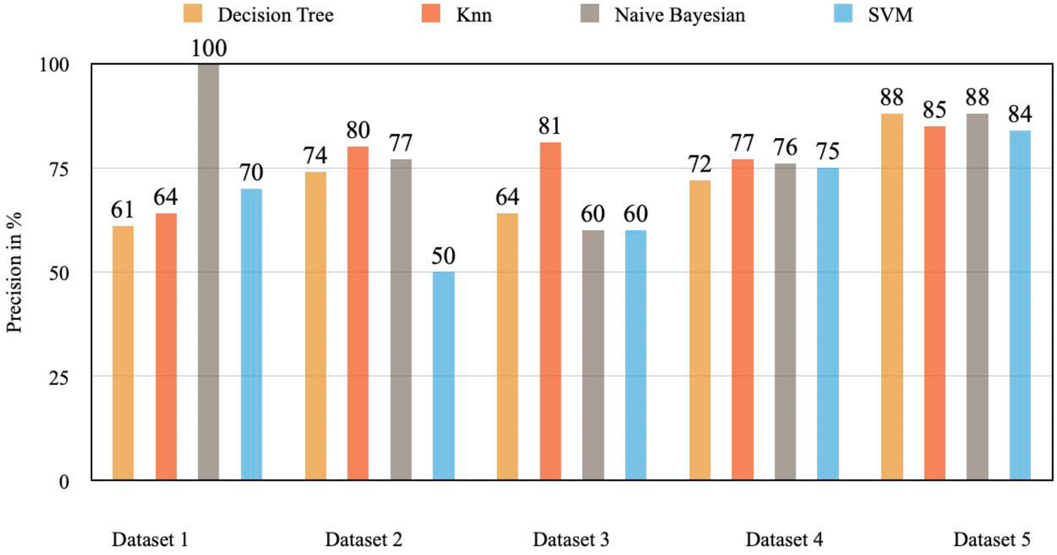
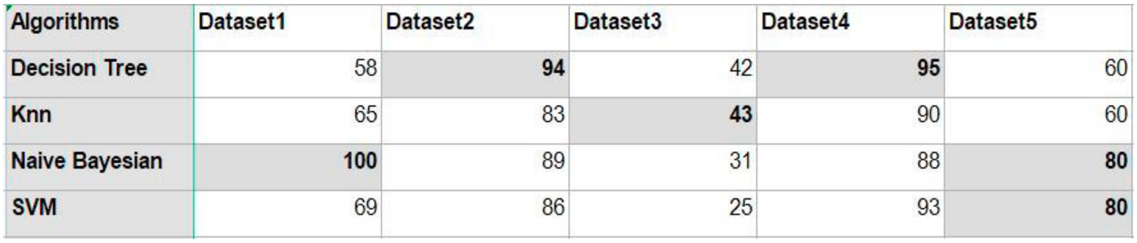


Fig. 3: Precision (in percentage) of the algorithms applied on each dataset

Table 4: F1 Score (in percentage) of the algorithms applied on each dataset



Algorithms	Dataset1	Dataset2	Dataset3	Dataset4	Dataset5
Decision Tree	58	**94**	42	**95**	60
Knn	65	83	**43**	90	60
Naive Bayesian	**100**	89	31	88	**80**
SVM	69	86	25	93	**80**

We discovered that Naive Bayes and Decision Tree did the best in terms of F1 score, with SVM coming in second, while k-NN performed the worst, with the lowest F1 score value.

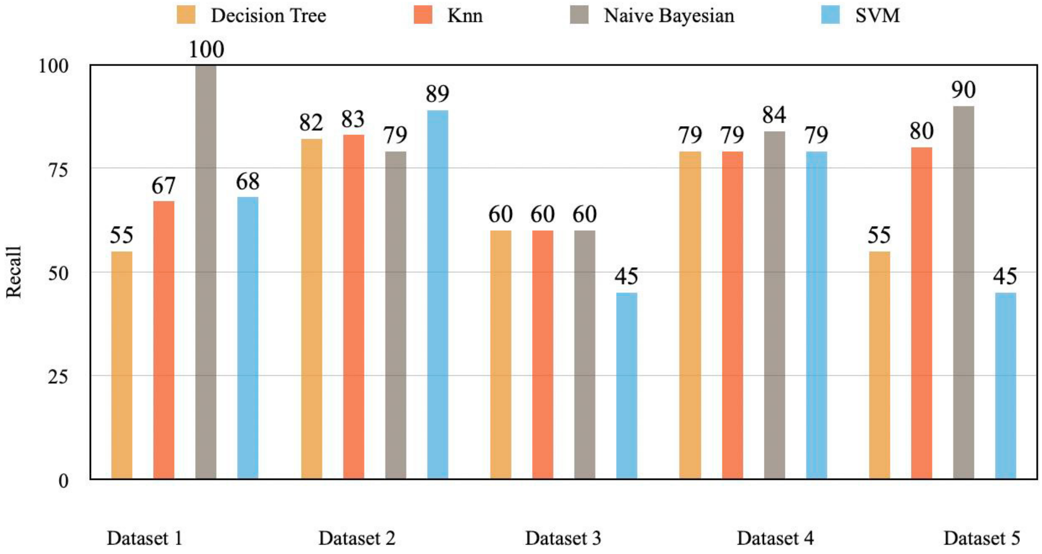


Fig. 4: Recall (in percentage) of the algorithms applied on each dataset

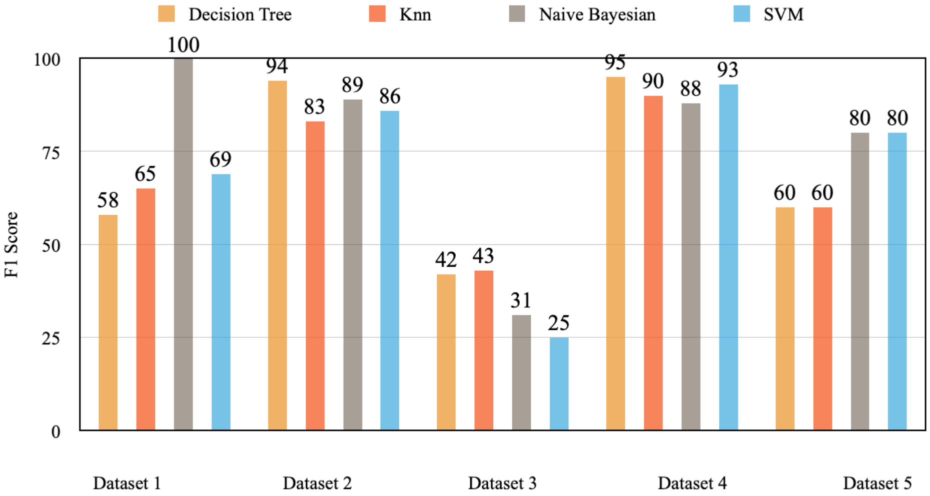


Fig. 5: F1 Score (in percentage) of the algorithms applied on each dataset

6. Conclusion

The prediction of classes is handled by a classification algorithm in this paper. There are different classification models available which are based on a variety of logic and methodologies. We compiled several datasets and compared the accuracy, recall, precision and F1 score of four most commonly used classifiers, namely decision trees, k-NN, SVM, and Naive Bayes. Each model performs differently depending on the size and characteristics of the data sets. There were only a few minor differences in performance measurements between the algorithms. A table including each performance measure against each dataset and each algorithm was created to determine which algorithm was the most effective overall. To gain a better understanding of the scores, we created a graphical representation of it in percentage form. After analysing the data, we ascertained that the Naive Bayesian classification model surpasses the others in terms of accuracy, recall, precision, and F1 score. The second-best classifier is SVM, which is preceded by K-Nearest Neighbor and Decision

The primary goal of this research has been to choose the best classifier from the most popular techniques. However, other models may be considered in the future work for comparison and selection. Various noise reduction strategies could be applied to enhance the results of this study, besides the one mentioned. Another aspect that would be intriguing to employ in any further research is usage of various measures to compare the performance of the algorithms.

References

[1] Anyanwu, Matthew & Shiva, S. (2009). Comparative Analysis of Serial Decision Tree Classification Algorithms. International Journal of Computer Science and Security. 3(3).

[2] M. J. Muzammil, S. Qazi and T. Ali, "Comparative analysis of classification algorithms performance for a statistical-based intrusion detection system," 2013 3rd IEEE International Conference on Computer, Control and Communication (IC4), 2013, pp. 1-6, DOI: 10.1109/IC4.2013.6653738.

[3] Anuradha, C & T, Velmurugan. (2015). A Comparative Analysis on the Evaluation of Classification Algorithms in the Prediction of Students Performance. Indian Journal of Science and technology. 8. 974-6846. 10.17485/ijst/2015/v8i15/74555.

[4] R. Muhamedyev, K. Yakunin, S. Iskakov, S. Sainova, A. Abdilmanova and Y. Kuchin, "Comparative analysis of classification algorithms," 2015 9th International Conference on Application of Information and Communication Technologies (AICT), 2015, pp. 96-101, DOI: 10.1109/ICAICT.2015.7338525.

[5] Amit Tate, Bajrangsingh Rajpurohit, Jayanand Pawar, Ujwala Gavhane,Gopal B. Deshmukh."Comparative Analysis of Classification Algorithms Used for Disease Prediction in Data Mining" Vol. 2 - Issue 6 (Nov - Dec 2016), International Journal of Engineering and Techniques (IJET), ISSN: 2395 - 1303, www.ijetjournal.org

[6] R. Duriqi, V. Raca and B. Cico, "Comparative analysis of classification algorithms on three different datasets using WEKA," 2016 5th Mediterranean Conference on Embedded Computing (MECO), 2016, pp. 335-338, DOI: 10.1109/MECO.2016.7525775.

[7] W. Becari, L. Ruiz, B. G. P. Evaristo and F. J. Ramirez-Fernandez, "Comparative analysis of classification algorithms on tactile sensors," 2016 IEEE International Symposium on Consumer Electronics (ISCE), 2016, pp. 1-2, DOI: 10.1109/ISCE.2016.7797324.

[8] P. Srikanth and D. Deverapalli, "A Critical Study of Classification Algorithms Using Diabetes Diagnosis," 2016 IEEE 6th International Conference on Advanced Computing (IACC), 2016, pp. 245- 249, DOI: 10.1109/IACC.2016.54.

[9] R. Archanaa, V. Athulya, T. Rajasundari and M. V. K. Kiran, "A comparative performance analysis on network traffic classification using supervised learning algorithms," 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), 2017, pp. 1-5, DOI: 10.1109/ICACCS.2017.8014634.

[10] S. M. Alqahtani and R. John, "A comparative analysis of different classification techniques for cloud intrusion detection systems' alerts and fuzzy classifiers," 2017 Computing Conference, 2017, pp. 406-415, DOI: 10.1109/SAI.2017.8252132.

[11] Journal of Basic and Applied Engineering Research p-ISSN: 2350-0077; e-ISSN: 2350-0255; Volume 4, Issue 3; April-June, 2017, pp. 180-184 © Krishi Sanskriti Publications http://www.krishisanskriti.org/Publication.html

[12] S. Sharma and D. Mehrotra, "Comparative Analysis of Multi-label Classification Algorithms," 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), 2018, pp. 35-38, DOI: 10.1109/ICSCCC.2018.8703285.

[13] Alghobiri, M. (2018). A Comparative Analysis of Classification Algorithms on Diverse Datasets. Engineering, Technology & Applied Science Research. 8. 2790-2795. 10.48084/etasr.1952.

[14] S. Velankar, S. Valecha, and S. Maji, "Bitcoin price prediction using machine learning," 2018 20th International Conference on Advanced Communication Technology (ICACT), 2018, pp. 144-147, DOI: 10.23919/ICACT.2018.8323676.

[15] K. Rathan, S. V. Sai and T. S. Manikanta, "Crypto-Currency price prediction using Decision Tree and Regression techniques," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 190-194, DOI: 10.1109/ICOEI.2019.8862585.

[16] A. Demir, B. N. Akılotu, Z. Kadiroğlu and A. Şengür, "Bitcoin Price Prediction Using Machine Learning Methods," 2019 1st International Informatics and Software Engineering Conference (UBMYK), 2019, pp. 1-4, DOI: 10.1109/UBMYK48245.2019.8965445.

[17] pandian, Dr. " REVIEW OF MACHINE LEARNING TECHNIQUES FOR VOLUMINOUS INFORMATION MANAGEMENT". Journal of Soft Computing Paradigm. 2019. 103-112. DOI: 10.36548/jscp.2019.2.005.

[18] kumar, Dr. (2019). "CLASSIFICATION OF BRAIN CANCER TYPE USING MACHINE LEARNING". Journal of Artificial Intelligence and Capsule Networks. 2019. DOI:10.36548/jaicn.2019.2.006.

[19] Reaz Chowdhury, M. Arifur Rahman, M. Sohel Rahman, M.R.C. Mahdy, an approach to predict and forecast the price of constituents and index of cryptocurrency using machine learning, Physica A: Statistical Mechanics and its Applications, Volume 551, 2020, 124569, ISSN 0378-4371, https://DOI.org/10.1016/j.physa.2020.124569.

[20] N. N. Qomariyah, E. Heriyanni, A. N. Fajar and D. Kazakov, "Comparative Analysis of Decision Tree Algorithm for Learning Ordinal Data Expressed as Pairwise Comparisons," 2020 8th International Conference on Information and Communication Technology (ICoICT), 2020, pp. 1- 4, DOI: 10.1109/ICoICT49345.2020.9166341.

[21] I. S. Balabanova, V. I. Markova, S. S. Kostadinova and G. I. Georgiev, "Comparative Analysis between Machine Learning Methods in Tones Classification," 2020 28th National Conference with International Participation (TELECOM), 2020, pp. 45-48, DOI: 10.1109/TELECOM50385.2020.9299535.

[22] M. Samaddar, R. Roy, S. De and R. Karmakar, "A Comparative Study of Different Machine Learning Algorithms on Bitcoin Value Prediction," 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), 2021, pp. 1-7, DOI: 10.1109/ICAECT49130.2021.9392629.