An improved KNN algorithm based on ensemble methods and correlation

Youness Manzali
Applied Physics, Computer Science and Statistics Laboratory
Faculty of Sciences Dhar El Mahraz
Sidi Mohammed Ben Abdellah University
Fez, Morocco
younes.manzali@usmba.ac.ma

Khalidou Abdoulaye Barry
Applied Physics, Computer Science and Statistics Laboratory
Faculty of Sciences Dhar El Mahraz
Sidi Mohammed Ben Abdellah University
Fez, Morocco
khalidou.barry@usmba.ac.ma

Mohamed EL Far
Applied Physics, Computer Science and Statistics Laboratory
Faculty of Sciences Dhar El Mahraz
Sidi Mohammed Ben Abdellah University
Fez, Morocco
mohamed.elfar@usmba.ac.ma

2023 7th IEEE Congress on Information Science and Technology (CiSt) | 978-1-6654-6133-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/CiSt56084.2023.10409922

Abstract—**K-Nearest Neighbors is a widely used algorithm due to its simplicity and efficacity. However, KNN suffers from many drawbacks, such as it does not work well with datasets with a high number of features. Also, not all the features contribute to the classification process. To resolve these issues, we present an improved KNN algorithm which uses KNN as a base learner in an ensemble method and correlation for selecting the features subsets. The experiment results show that the proposed algorithm performed better than other machine learning algorithms.**

Keywords—KNN algorithm, ensemble method of KNN, subsampling, bagging.

I. Introduction

K-Nearest Neighbors (KNN) is a straightforward yet effective machine-learning technique for classification and regression issues. It works on the assumption that similar data points belong to the same class or have similar output values.

The method searches the training set for the k nearest neighbours of a given data point [1], where k is a preset hyperparameter. The neighbours are found using a similarity metric, usually Euclidean distance or another distance metric [2]. In the case of classification issues, after the nearest neighbours are discovered, the method assigns a label to the new data point based on the most popular class label among its k nearest neighbours. In regression problems, the output value is calculated by taking the average or weighted average of the k nearest neighbours' output values.

The KNN method is basic and straightforward, making it an excellent starting point for machine learning learners. It makes no assumptions about the underlying data distribution and can handle numerical and categorical data. Also, it can handle multi-class cases [3]. Despite these advantages, KNN has some disadvantages, including:

- It is computationally intensive and requires the entire training dataset to be held in memory, which can be difficult for large datasets.

- It can be sensitive to unnecessary or redundant features.

- It can be affected by the distance metric used to estimate the similarity between instances.

KNN is a helpful technique for some tasks, but it may not be the ideal solution for large data sets or when computational efficiency is important. Other KNN variants use feature selection, such as ReliefF-KNN and Forward Selection-KNN, to select the most important features for classification. These methods select features before applying KNN to the smaller feature space.

Ensemble methods aim to improve a model's predictive power by combining multiple models' outputs. KNN ensemble methods apply this concept to KNN models by creating an ensemble of KNN classifiers or regressors. Each KNN model is trained on a subset of the training data or with different parameter settings, resulting in a diverse set of models.

The ensemble is typically constructed by aggregating the predictions of individual KNN models. The most common approach for classification tasks is majority voting, where the class with the highest number of votes from the ensemble members is selected as the final prediction. In regression tasks, the ensemble predictions can be averaged or combined using other aggregation techniques.

KNN ensemble methods can be computationally expensive, particularly when dealing with large datasets or high-dimensional feature spaces. Developing efficient algorithms or optimization strategies to improve scalability is an ongoing research area. This could include techniques such as data reduction, instance selection, or parallelization of ensemble construction.

For these reasons, we propose an improved KNN ensemble method which uses a feature subselection technique based on correlation. The new algorithm creates an ensemble of KNN models and then aggregates their predictions to obtain the final predictions. This technique may first reduce the space memory the algorithm allows since it uses just a subset of features at each iteration. Second, it could improve the performance because the ensemble prediction performs better than a single estimator. The rest of the paper is organized as follows: Section 2 presents the related works. Section 3 describes in detail the proposed method. Section 4 presents the experiment results and the corresponding analysis. Section 5 concludes the paper and proposes future directions.

978-1-6654-6133-7/23/$31.00 ©2023 IEEE

II. RELATED WORKS

KNN may be used as a basic classifier in ensemble methods like bagging and boosting to increase the algorithm's performance. For example, KNN is used to classify data subsets chosen randomly from the feature space in the random subspace method. The use of the k-Nearest Neighbor (k-NN) algorithm as a base learner in ensemble methods dates back to at least the year 2000 when Pedro Domingos and Geoff Hulten [4] presented a new ensemble method for k-Nearest Neighbor (k-NN) classification. The authors describe a technique for using multiple k-NN classifiers in an ensemble, where each classifier is trained on a randomly selected subset of the training data. The predictions of these classifiers are then combined using a weighted voting scheme to produce the final classification decision. The authors demonstrate that their approach can improve the accuracy of k-NN classification on several datasets, and they compare their method with other ensemble methods such as bagging and boosting.

Similarly, Bojan Cestnik [5] proposes a new approach for improving the classification of imbalanced datasets using k-Nearest Neighbor (k-NN) and bagging. The author describes a technique for creating an ensemble of k-NN classifiers, where each classifier is trained on a resampled subset of the imbalanced dataset. The resampling is performed by either oversampling the minority class or undersampling the majority class. The author demonstrates that their approach can improve the accuracy of k-NN classification on several imbalanced datasets, and they compare their method with other approaches such as cost-sensitive learning and boosting.

Ajibade et al. [6] conducted a study where they utilized ensemble methods with various base learners, including Naive Bayes (NB), Decision Tree (ID3), Support Vector Machines (SVM), and K-Nearest Neighbor (KNN), to predict student academic performance. The researchers confirmed that ensemble methods improved performance compared to individual base learners alone.

Similarly, Ajibade et al. [7] propose a new performance prediction model based on data mining methods, specifically utilizing the behavioural features of students. The model incorporates sequential feature selection and evaluates its performance using classifiers such as Support Vector Machine (SVM), K-Nearest Neighbour (KNN), and Decision Tree (DT). Ensemble methods, including Bagging, Boosting, and Random Forest, enhance classifier performance. The results demonstrate a strong correlation between student behaviour and academic performance.

Mahfouz et al. propose [8] an ensemble classifier, EKNN, for automated cancer diagnosis using microarray-based approaches. EKNN addresses challenges such as high dimensionality, noise, sample imbalance, and small dataset sizes. The classifier combines traditional kNN with four novel classification models, leveraging density and connectivity measures. EKNN demonstrates improved accuracy compared to traditional KNN and other existing ensemble methods on various datasets.

Mehanovic et al. [9] emphasize the significance of timely heart disease detection and propose using the artificial neural network, k-nearest neighbour, and support vector machine algorithms to develop a prediction model. They conduct several experiments with each algorithm and explore the effectiveness of ensemble learning techniques in their study.

Swarna et al. propose [10] an ensemble Random Subspace (RS) classifier for the detection of High Impedance Fault (HIF) in photovoltaic connected power networks. The classifier utilizes feature extraction through discrete wavelet transform and trains the RS ensemble with base classifiers, including K-nearest neighbour, Logistic regression, and Random tree. The results show that the RS ensemble classifier outperforms the individual base classifiers in terms of accuracy and success rate for discriminating HIF.

In their work, Zhang et al. [11] introduce a novel approach called the Weighted Heterogeneous Distance Metric (WHDM). This metric and evidence theory forms the foundation for developing a progressive kNN classifier. Building upon this classifier, the authors propose a new algorithm called Reduced Random Subspace-based Bagging (RRSB). RRSB leverages techniques such as the random subspace method, attribute reduction, and Bagging to construct an ensemble classifier. This algorithm effectively enhances the diversity of component classifiers while preserving their accuracy.

Wijayanto and Sartono [12] aimed to evaluate the performance of KNN (K-Nearest Neighbors) and ensemble KNN methods. Despite its simplicity, the KNN method offers several advantages compared to other approaches. For instance, it can effectively generalize from relatively small training sets. However, an important aspect of this method is selecting the number of k-nearest neighbours. To address this, the authors employed an ensemble technique that enhances the accuracy of predictions within the KNN method, eliminating the need to search for an optimal value of k. The results indicated that the Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE) of predictions tend to decrease as the number of k-nearest neighbours increases. Overall, the KNN ensemble method performed superior to the standalone KNN method.

Xiao [13] introduces a novel ensemble learning approach for traffic incident detection. The method utilizes individual SVM (Support Vector Machine) and KNN (K-Nearest Neighbors) models as a first step, followed by an ensemble learning strategy to combine them and enhance the final output. The proposed method demonstrates superior performance through experimentation compared to other methods analyzed. Additionally, the ensemble learning strategy significantly improves the robustness of the individual models, making them more reliable in various scenarios.

Grabowski [14] proposed a new approach to k-Nearest Neighbor (k-NN) classification. An ensemble of k-NN classifiers is used, each trained on a random partition of the training set and assigned its k value. This is a departure from traditional ensemble schemes. Despite using an ensemble of classifiers, the classification speed in this new algorithm is similar to that of the original k-NN algorithm.

Choi et al. [15] presented a new ensemble method for the k-Nearest Neighbor (k-NN) classification of imbalanced medical datasets. The authors propose a technique for creating an ensemble of k-NN classifiers, where each classifier is trained on a different resampled subset of the imbalanced dataset. The resampling is performed by either oversampling the minority class or undersampling the majority class. The authors use a genetic algorithm to select the best combination of classifiers from the ensemble to make the final classification decision. Gul et al. [17] proposed an ensemble of a subset of kNN classifiers, ESkNN, for classification tasks in two steps. Firstly, they choose classifiers based on their performance using out-of-sample accuracy. The selected classifiers are then combined sequentially, starting from the best model and assessed for collective performance on a validation data set.

III. PROPOSED METHOD

A. *Ensemble methods*

An ensemble method of kNN (k-Nearest Neighbors) is a technique that combines multiple kNN classifiers to achieve better performance than a single kNN classifier.

The general idea is to divide the training data into subsets and train a kNN classifier on each subset. Each classifier has its k parameter value, the number of nearest neighbours considered for classification. These kNN classifiers are then combined in some way to produce a final prediction.

B. *Proposed method*

Similarly, as the general idea of an ensemble method, we propose an ensemble method that uses KNN as a base learner. The following flowchart describes the process of the proposed algorithm:

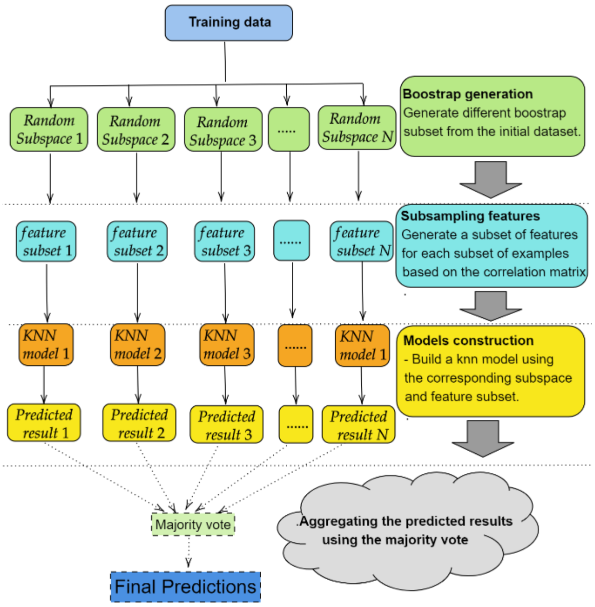


Figure 1: Flowchart of the proposed method.

First, we generate N bootstrap examples from the initial data set with replacement. For each subset of examples, we compute the correlation between all the features and the target attribute, and we maintain just the features with a positive correlation with the target attribute. Next, we use the selected features to select the k-nearest neighbours from the unlabeled test example, i.e., just the subset of selected features is used in the process of the Euclidian distance calculation. Finally, we aggregate the model's predictions using the majority vote.

1) *Example of correlation calculation between features*

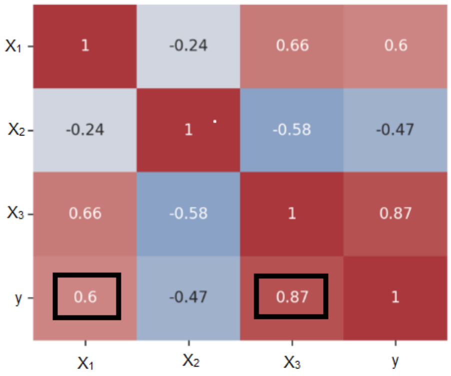


Figure 2:Example of a correlation matrix

Taking the correlation matrix mentioned above (Fig.2) as an example, our primary interest lies in understanding the correlations between various attributes and the output attribute y. Upon analyzing the correlation matrix, it becomes evident that X1 and X3 exhibit a positive correlation, unlike X2. Consequently, we will select X1 and X3 for this dataset and employ them in calculating the Euclidean distance to determine the k nearest neighbours.

The exact process of the proposed method is described in the following algorithm:

Proposed method algorithm

Input:

- train_data: The training dataset with labelled instances

- test_instance: The instance for which we want to make a prediction

- k: The number of nearest neighbours to consider

- num_classifiers: The number of KNN classifiers in the ensemble

Output:

- predicted_class: The predicted class label for the test_instance

```
Procedure KNN_Ensemble(train_data, test_instance, k, num_classifiers):
    predictions ← array of size num_classifiers
    for i ← 1 to num_classifiers do
        d_i ← randomly sample train_data with replacement
        features ← select features that positively
                   correlate with the target attribute according to d_i.
        classifier_knn <- KNN(train_data, k, features)
        preds[i]←classifier_knn.predict(test_instance)
    final_prediction <- Majority_Vote(preds)
    return final_prediction
```

Here are the main steps of the proposed method:

- **Step 1:** Generate N subsets selected randomly with replacement from the initial dataset using the bootstrap method.
- **Step 2:** Generate the correlation matrix on each subset of the training set. And eliminate features whose correlation with target attributes are negative.
- **Step 3:** Train multiple k-NN classifiers on each subset of the training set. Each classifier may be trained on a different subset of features.
- **Step 4:** Combine the predictions of the k-NN classifiers to make a final prediction using the majority vote.
- **Step 5:** tune the hyperparameters of the k-NN classifiers or the ensemble method itself using the cross-validation technique.
- **Step 6:** Use the trained ensemble classifier to make predictions on new unseen data.

C. *Advantages and drawbacks of the proposed method*

The proposed method can bring several benefits, such as:

- **Improved accuracy:** feature weighting can increase KNN accuracy by emphasizing the most important characteristics. This is especially beneficial when working with datasets containing multiple characteristics, some of which may be less informative than others.
- **Reduction in Dimensionality:** By using only a subset of features, the ensemble method can reduce the dimensionality of the dataset. This can be particularly beneficial when dealing with high-dimensional data, as it can help mitigate the "curse of dimensionality" and improve computational efficiency.
- **Increased efficiency:** by lowering the number of irrelevant characteristics utilized in the distance computation, feature weighting can further increase the efficiency of KNN. KNN can save needless calculations and lower the algorithm's computational cost by assigning a weight of zero to non-informative features.
- **Flexibility:** the number of selected attributes varies according to the subset of examples, which increases the diversity of the created models and, consequently, their accuracy.

Despite its usefulness, the proposed method (PM) has some limitations when compared to other methods, such as:

- **Information Loss:** Using only a subset of features can lead to information loss, especially if important features are excluded. This may result in less accurate predictions, particularly when missing relevant features.
- **Optimal Subset Selection:** Determining the best subset of features can be challenging. The ensemble method must have a robust mechanism to identify the most relevant features to include, which can require additional computation and careful tuning.

D. *Algorithm complexity*

1) The complexity of a KNN ensemble method (using all features):

- Suppose the original dataset has n data points, and the full feature set has d dimensions (features).
- In the classic KNN ensemble method (KNNEM) algorithm, where all features are used to calculate the Euclidean distance, the time complexity for computing the distance for a single data point is O(d).
- Therefore, the time complexity for predicting a single data point using classic KNNEM is O(d).
- The overall time complexity for predicting all data points in the dataset is O(n * d), where n is the number of data points.

2) The complexity of the proposed method

- The ensemble method selects a subset of features of size k (where k << d), and the complexity of computing the Euclidean distance for a single data point reduces from O(d) to O(k), where k is the number of selected features.
- Therefore, the time complexity for predicting a single data point using this ensemble method is O(k) instead of the traditional O(d) in the standard KNNEM algorithm.
- The overall time complexity for predicting all data points in the dataset remains O(n * k), where n is the number of data points.

3) Comparison.

The proposed method has a reduced time complexity of O(k) per prediction compared to the classic KNNEM's O(d) per prediction when k is much smaller than d.

If the selected subset size, k, is significantly smaller than the total number of features, d, the ensemble method can be computationally more efficient than the classic KNNEM. The reduction in time complexity becomes more pronounced as the number of features in the original dataset increases.

IV. Experiment Results

To evaluate the practicality of the proposed method, we conducted a comparative analysis against several other algorithms. Specifically, we compared the performance of the proposed method with that of the K-nearest neighbours (KNN) algorithm [1], the Approximate Nearest Neighbor (ANN) algorithm [9], and the Ensemble Method of KNN (EMKNN).

A. *Datasets*

To assess the effectiveness of the proposed method, we conducted experiments using twenty real-world datasets publicly available from the UCI repository [16] and Kaggle. The details of these datasets are provided in Table 1, and they all involved binary classification tasks. We recorded each dataset's size and the number of nominal and numerical attributes.

TABLE I. Data set details.

Id	***Dataset***	***Size***	**Attributes**	
			Num	***Cat***
1	Heart	303	6	7

Id	*Dataset*	*Size*	Attributes	
			Num	*Cat*
2	Bands	540	24	15
3	Hepatitis	129	6	13
4	Titanic	1307	2	24
5	Spect	267	44	6
6	Sonar	208	60	0
7	Wdbc	569	30	0
8	Hill_Valey	606	100	0
9	Magic04	19020	10	0
10	Eighthr	2533	73	0
11	Clean	6598	168	0
12	Eye	14980	14	0
13	Ionosphere	351	14	0
14	Credit cards	30000	24	0
15	Pima	768	8	0

B. *Performance metrics*

To evaluate the performance of the proposed method and the other compared algorithms, we used three performance measures, which are detailed as follows:

- **Accuracy:** it is the ratio of the number of correctly classified instances to the total number of instances. This metric is widely used to evaluate the performance of classifiers, and it is calculated using the following formula:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$

- **Precision** measures the proportion of correctly predicted positive instances out of all positive predictions. This metric is particularly important for minority classes, as their precision determines accuracy. Precision is calculated using the following formula:

$$Precision = \frac{TP}{(TP + FP)}$$

TP denotes the true positive predictions, and FP denotes the false positive ones.

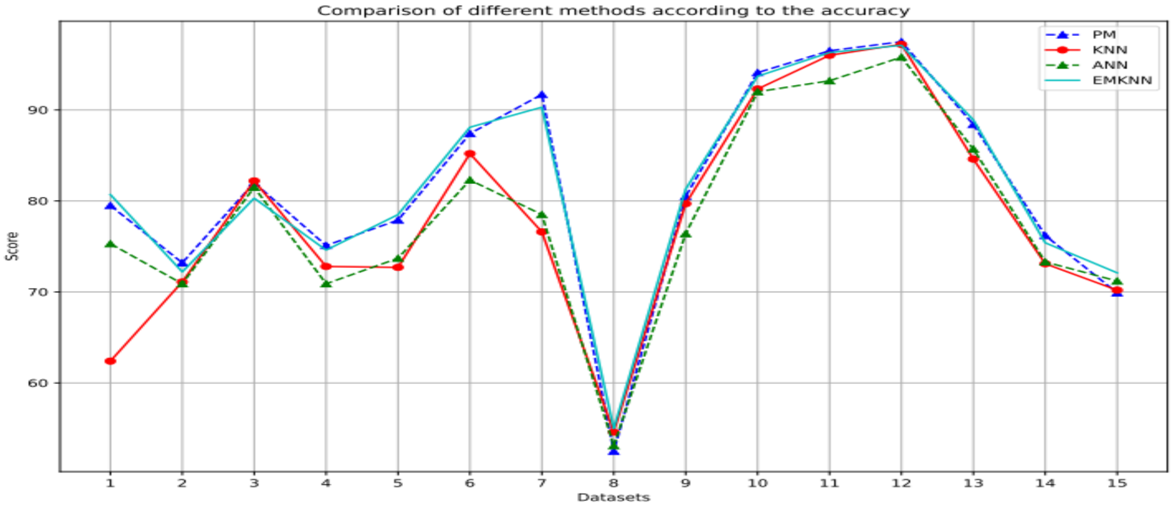
- **Recall:** it measures the proportion of correctly predicted positive instances out of all positive ones. This metric provides an idea of the coverage of the positive class. The Recall is calculated using the following formula:

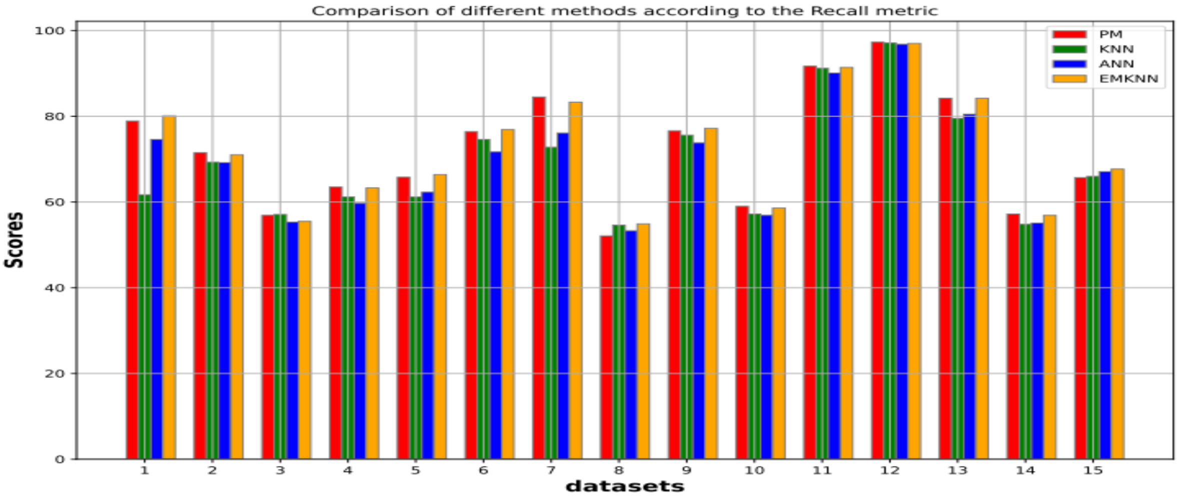
$$Recall = \frac{TP}{(TP + FN)}$$

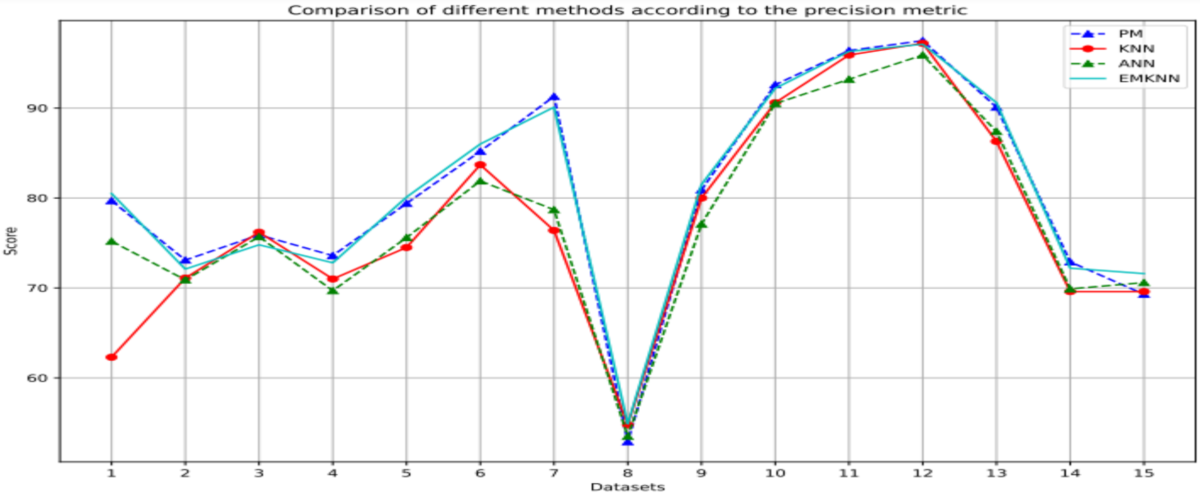
Where FN represents the false negative predictions.

C. *Results of the experiments*

To achieve the results, each dataset undergoes five cross-validations. Five models are generated from each training dataset in our experiments, and their performance is evaluated against the corresponding test datasets. The reported performances in the tables below represent the average of the five folds. Table 2 displays the accuracy results of the proposed method compared to other algorithms. In contrast, Table 3 showcases the precision scores achieved, and Table 4 shows the performance of the evaluated methods in terms of Recall. The highest scores achieved for each dataset are indicated in bold.







D. *Analysis of the results*

Table 2 shows that the proposed method outperforms competitors on seven out of fifteen datasets, with only small differences in performance for the remaining datasets. As a result, the proposed method and EMKNN have the best average accuracy across all datasets. In Table 3, the proposed method demonstrates good performance in most datasets, with consistently the highest precision in 'Bands', 'wdbc', and 'credit cards'. Looking at Table 4, the proposed method has the best recall performance on most datasets, particularly 'Bands', 'wdbc', and 'eighthr'. Overall, the proposed method performs better than most other methods in terms of the three performance metrics, except for EMKNN, where the proposed method's performance is very close but with a slight advantage over EMKNN.

CONCLUSION AND FUTURE WORKS

This paper introduced a novel ensemble method based on the K-nearest neighbours (KNN) algorithm, which involves building multiple KNN models using different bootstrap examples and feature selection techniques based on correlation. Our experimental evaluation demonstrated that this method is efficient and accurate compared to the classical KNN algorithm, the Approximate Nearest Neighbor (ANN) algorithm, and the Ensemble Method of KNN (EMKNN) regarding the three performance metrics.

The proposed method efficiently utilizes only a subset of features at each estimator, yet it outperforms other algorithms. This can be attributed to PM's ability to eliminate irrelevant attributes, leading to substantial improvements in occupied memory, execution time, and overall performance.

As part of our future work, we plan to investigate this feature selection technique's application in other ensemble methods. Additionally, we intend to explore this technique as a feature weighting technique and test its effectiveness when applied to different machine learning algorithms.

REFERENCES

[1] E. Fix and J. L. Hodges Jr., "A method of classifying patterns," in Proceedings of the 1962 ACM National Conference, 1962, pp. 81-83.

[2] H. A. Abu Alfeilat, A. B. Hassanat, O. Lasassmeh, A. S. Tarawneh, M. B. Alhasanat, H. S. Eyal Salman and V. S. Prasath, "Effects of Distance Measure Choice on k-Nearest Neighbor Classifier Performance: A Review," in Big Data, vol. 7, no. 4, pp. 221-248, Dec. 2019, doi: 10.1089/big.2019.0049.

[3] S. Kanwal, S. Shahzad and S. H. Zaidi, "A Review of Advantages and Drawbacks of k-Nearest Neighbors Based Classification," in IEEE Access, vol. 9, pp. 75387-75399, 2021, doi: 10.1109/ACCESS.2021.3081491.

[4] Domingos, P., & Hulten, G. Improving k-NN with ensemble decisions. In Proceedings of the 17th International Conference on Machine Learning (ICML),2000.pp. 175-182.

[5] B. Cestnik, "Bagging k-NN for Imbalanced Data Classification," in Proceedings of the 14th European Conference on Machine Learning (ECML), 2003, pp. 106-117.

[6] Ajibade, S. S. M., Dayupay, J., Ngo-Hoang, D. L., Oyebode, O. J., & Sasan, J. M. (2022). Utilization of Ensemble Techniques for Prediction of the Academic Performance of Students. Journal of Optoelectronics Laser, 41(6), 48-54.

[7] Ajibade, S. S. M., Bahiah Binti Ahmad, N., & Mariyam Shamsuddin, S. (2019, August). Educational data mining: enhancement of student performance model using ensemble methods. In IOP Conference Series: Materials Science and Engineering (Vol. 551, No. 1, p. 012061). IOP Publishing.

[8] Mahfouz, M. A., Shoukry, A., & Ismail, M. A. (2021). EKNN: Ensemble classifier incorporating connectivity and density into kNN with application to cancer diagnosis. Artificial Intelligence in Medicine, 111, 101985.

[9] Mehanović, D., Mašetić, Z., & Kečo, D. (2020). Prediction of heart diseases using majority voting ensemble method. In CMBEBIH 2019: Proceedings of the International Conference on Medical and Biological Engineering, 16–18 May 2019, Banja Luka, Bosnia and Herzegovina (pp. 491-498). Springer International Publishing.

[10] Swarna, K. S. V., Vinayagam, A., Ananth, M. B. J., Kumar, P. V., Veerasamy, V., & Radhakrishnan, P. (2022). A KNN based random subspace ensemble classifier for detection and discrimination of high impedance fault in PV integrated power network. Measurement, 187, 110333.

[11] Zhang, Y., Cao, G., Wang, B., & Li, X. (2019). A novel ensemble method for k-nearest neighbor. Pattern Recognition, 85, 13-25.

[12] Sinta, D., Wijayanto, H., & Sartono, B. J. A. M. S. (2014). Ensemble K-Nearest neighbors method to predict rice price in Indonesia. Appl. Math. Sci, 8(160), 7993-8005.

[13] Xiao, J. (2019). SVM and KNN ensemble learning for traffic incident detection. Physica A: Statistical Mechanics and its Applications, 517, 29-35.

[14] Grabowski, S. (2002, February). Voting over multiple k-nn classifiers. In Modern problems of radio engineering, telecommunications and computer science (IEEE Cat. No. 02EX542) (pp. 223-225). IEEE.

[15] Y. H. Choi, S. U. Shin and E. Y. Kang, "kNN-Based Ensemble with Genetic Algorithm for Imbalanced Medical Data Classification," in Journal of Medical Systems, vol. 39, no. 6, p. 67, 2015, doi: 10.1007/s10916-015-0267-9.

[16] M. Lichman et al., "Uci machine learning repository," 2013

[17] Gul, A., Perperoglou, A., Khan, Z., Mahmoud, O., Miftahuddin, M., Adler, W., & Lausen, B. (2018). Ensemble of a subset of k NN classifiers. Advances in data analysis and classification, 12, 827-840.