SSRG International Journal of Computer Science and Engineering
ISSN: 2348–8387 / https://doi.org/10.14445/23488387/IJCSE-V12I6P102

Volume 12 Issue 6, 10-15, June 2025
© 2025 Seventh Sense Research Group®

Original Article

Enhancing Flower Type Classification Using K-Nearest Neighbors (KNN)

Prit Senjaliya[1], Khushi Prajapati[2], Avani Vagadiya[3]

[1,2]Department of Information Technology, Gandhinagar University, Gandhinagar, Gujarat, India.
[3]Computer Engineering Department, Gandhinagar University, Gandhinagar, Gujarat, India.

[1]Corresponding Author : pritsenjaliya1162@gmail.com

Received: 29 April 2025 Revised: 01 June 2025 Accepted: 13 June 2025 Published: 30 June 2025

Abstract - *This research uses a custom-generated dataset to implement the K-Nearest Neighbors (KNN) algorithm for flower classification. The dataset that has been taken for this research consists of 1,000 samples with features including petal and sepal dimensions and tilt angle. Data preprocessing techniques have been employed to improve and clean the manually generated dataset and optimize the performance of the classification model. A 10-fold cross-validation approach has been applied to evaluate the accuracy of classification algorithms and analyze the impact of different hyperparameters on the performance of the KNN algorithm. This study highlights the importance of the feature selection method, data normalization, and k-value optimization in achieving higher accuracy in the classification process. Furthermore, this research focuses on the challenges encountered during model training, such as class imbalance and overfitting. This study also provides perceptions of enhancing KNN's efficiency in multi-class classification tasks and establishes its potential for real-world applications in automated flower identification and similar pattern recognition problems.*

Keywords - *Data Preprocessing, Flower Classification, Hyperparameter Tuning, K-Nearest Neighbors, Machine Learning, Multi-Class Classification.*

1. Introduction

Classification technique, which is used in data mining and machine learning, is adapted to categorize the data into labelled classes. Machine learning is a widely used area that is applied to train the model to perform supervised and unsupervised learning. [1] Machine learning (ML) classification algorithms are substantial tools for handling a variety of real-world problems, like image recognition, pattern recognition, sentiment analysis, spam or fraud detection, medical diagnosis, and automated decision-making. These techniques are further divided into two types: supervised learning, which includes classification and regression techniques, and unsupervised learning, which includes clustering and association rule mining techniques. This research concentrates on the requirement and adaptability of classification algorithms and their effectiveness in solving real-world complex problems.

[2, 11] Amongst some of the classification algorithms like Random Forest, Support Vector Machine (SVM) and K-Nearest Neighbors (KNN), the K-Nearest Neighbors (KNN) algorithm looms as a simple and important instance-based learning technique that classifies the data to labeled classes/instances depending on their proximity in the dataset. [3] KNN is especially suited for tasks that involve image recognition, pattern recognition and classification due to its non-parametric nature. However, the performance of the KNN algorithm is highly dependent on various factors, including the selection of appropriate features, the choice of distance metrics, and the determination of the optimal Value of k, which defines the total number of neighbors considered during classification.

1.1. Research Gap and Overview

Whilst machine learning is used in many areas like image recognition, pattern recognition and medical diagnosis, little research has focused on training the model for classifying the various flowers. One of the reasons could be that few large and labeled flower datasets are available, and some flowers may often look very similar, making the classification task more difficult. Also, most earlier studies in plant science have mainly focused on leaves or crops instead of flowers. This research helps fill that gap by showing how the KNN (K-Nearest Neighbors) algorithm, with the right setup and data processing, can work well for classifying flowers. This research also includes the comparison of the accuracy of classifying the custom flower dataset achieved with the KNN algorithm and Random

Forest algorithm. It can be helpful in building mobile apps for identifying various flowers, helping farmers to choose the right plants, and even supporting studies based on environmental science. This research aims to make flower classification easier, faster, and more accurate using a simple and reliable machine-learning classification method.

Flower classification with precise results holds considerable Value in botany, agriculture, and environmental conservation fields. Due to the wide variety of morphological characteristics among flower species, manual classification methods are often inefficient, subjective, and susceptible to human errors. These limitations direct towards the need for an automated and data-driven classification approach that can treat large datasets with high accuracy and consistency.

In this study, the effectiveness of the KNN algorithm was evaluated in classifying different flower types using a custom dataset consisting of 1,000 samples with various flower attributes. Data preprocessing techniques such as data cleaning, data normalization and feature selection have been employed to improve the performance of the classification process. Additionally, a survey of how different hyperparameters and distance measures influence the model accuracy has been done. The goal is to enhance the performance of KNN in multi-class classification tasks and demonstrate its applicability in real-world scenarios, such as automated flower species identification and precision agriculture.

1.2. Uniqueness

The thing that makes this research different is that a custom dataset is used with unique features like the tilt angle of flowers, along with petal and sepal measurements. Most existing studies of flower classification only use basic features like length and width, but by adding some new features like tilt angle, this study has introduced a new way to help the model better recognize flower types. In this research, the Value of k is also carefully tested, multiple distance measures are tried, and 10-fold cross-validation is performed to make sure the results are accurate and reliable. Although the use of a simple method like KNN is still getting good results, it shows that even basic algorithms can work well if the data is prepared and used properly.

1.3. Literature Review

Multiple studies have applied and researched the enhancement of the K-Nearest Neighbors (KNN) algorithm and some other classification algorithms in several domains, providing valuable insights for its adaptation to flower classification mechanisms.

[4] Intersection Similarity-Based KNN Models proposed a KNN model based on intersection similarity and proved that with the application of feature extraction and weighting techniques, classification performance can be improved significantly. Their result implies that these methods can also benefit flower classification through better precision and recall.

[5] KNN was employed to establish the air quality level, and it was concluded that the algorithm's efficiency relies heavily on the quality of data preprocessing and the fine-tuning of model hyperparameters. Such conclusions strengthen the argument for the application of KNN in cases where more feature importance needs to be analyzed, like in predicting the type of flower.

[6] Genetic Algorithms were integrated with KNN so that the Value of k could be flexibly adjusted, improving accuracy in image classification. This approach facilitates more advanced studies aimed at improving the classification of flowers using flexible optimization techniques.

[7] Improved KNN for Talent Classification uses a modified KNN model for talent classification that integrates adaptive weighting and rough set theory. Their approach provided better results in complex pattern recognition, providing a solution for multi-class problems such as flower detection.

[8] KNN in Spam Filtering focused on using KNN in spam detection, focusing on feature selection and data resampling to manage an imbalanced dataset. These approaches are applicable to flower datasets where some classes are dominant, and others are not.

[9] KNN for Stock Market Prediction examines the KNN algorithm for projecting stock market prices based on short-term fluctuations. The research revealed that distance metric selection and tuning the k parameter improves prediction results and implements the finding that KNN needs to adjust properly for optimal classification outcomes.

2. Method

2.1. Dataset Description

The dataset that has been utilized in this study is made up of 1,000 samples, each representing a distinct flower type. Each entry of the dataset includes five key features: Sepal length (cm), Sepal width (cm), Petal length (cm), Petal width (cm) and Tilt angle (numeric Value).

To ensure a high-quality model performance, some data preprocessing techniques were applied to the dataset. This included normalizing all numerical attributes to a proportionate and comparable scale and encoding methods to convert any categorical information into a machine-readable format. The dataset was also carefully examined for inconsistencies such as missing values, outliers, and noise, all of which were addressed to enhance the quality of the input dataset.

The dataset was handpicked to wrap various flower types by providing a varied and balanced sample set. This diversity makes it well-suited for evaluating the effectiveness of the KNN algorithm in multi-class classification scenarios. The proper preprocessing process ensured that the model could learn meaningful patterns and relationships among the features, leading to more accurate and precise classification outcomes.

2.2. Methodology

The study follows these key steps:

1. Dataset Preprocessing: Missing values were handled, categorical features were encoded, and numerical attributes were normalized.
2. Model Selection: The KNN algorithm was implemented with varying values of k to determine optimal performance.
3. Evaluation Metrics: Model effectiveness was assessed by accuracy, precision, recall, and F1-score.
4. Cross-Validation: A 10-fold cross-validation technique was employed to evaluate the robustness of the model.

2.3. Units

In this research:

1. Flower dimensions like a petal and sepal lengths are measured in centimetres.
2. Classification accuracy is calculated in a percentage (%).

2.4. Equations

2.4.1. Euclidean Distance

[12] Euclidean distance is represented as a straight line, which shows the distance between two points in a multi-dimensional space. It is derived from the Pythagorean theorem and is the most commonly used distance metric in machine learning and statistics.

$$d\ (i,j) = \sqrt{\sum_{m=1}^{n} (X_{im} - X_{jm})^2} \tag{1}$$

2.4.2. Manhattan Distance

[22] Manhattan distance, also known as Taxicab Distance or City Block Distance, measures the distance between two points by adding the absolute differences of their coordinates. It represents how a taxi would navigate through a grid-like city.

$$d\ (i,j) = \sum_{m=1}^{n} [X_{im} - X_{jm}] \tag{2}$$

2.5. Figures and Tables

The image below represents a bar graph that includes the accuracy achieved in KNN versus k value.

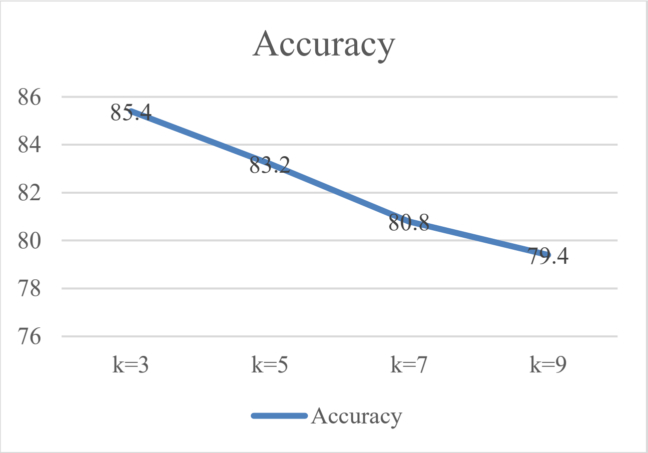
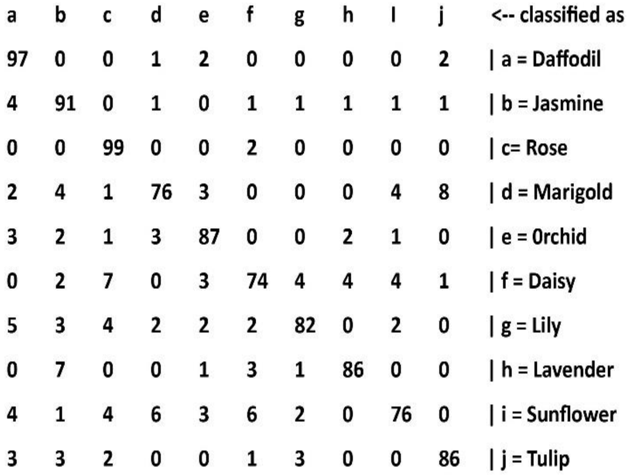


Fig. 1 Accuracy v/s K-value graph

2.6. Confusion Matrix

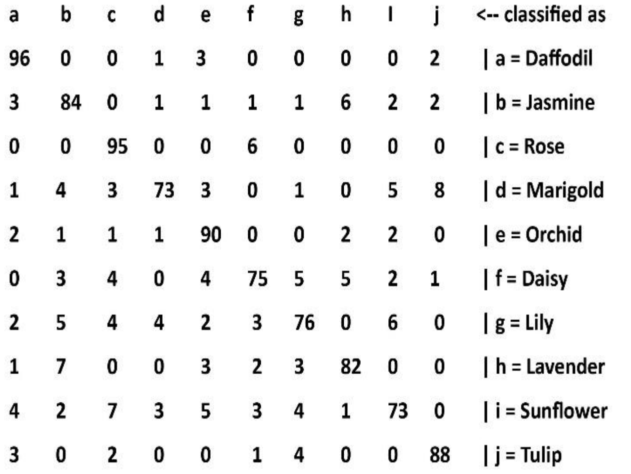
2.6.1. Value of K=3:



a	b	c	d	e	f	g	h	I	j	<-- classified as
97	0	0	1	2	0	0	0	0	2	| a = Daffodil
4	91	0	1	0	1	1	1	1	1	| b = Jasmine
0	0	99	0	0	2	0	0	0	0	| c= Rose
2	4	1	76	3	0	0	0	4	8	| d = Marigold
3	2	1	3	87	0	0	2	1	0	| e = 0rchid
0	2	7	0	3	74	4	4	4	1	| f = Daisy
5	3	4	2	2	2	82	0	2	0	| g = Lily
0	7	0	0	1	3	1	86	0	0	| h = Lavender
4	1	4	6	3	6	2	0	76	0	| i = Sunflower
3	3	2	0	0	1	3	0	0	86	| j = Tulip

Fig. 2 Confusion matrix for k-value 3

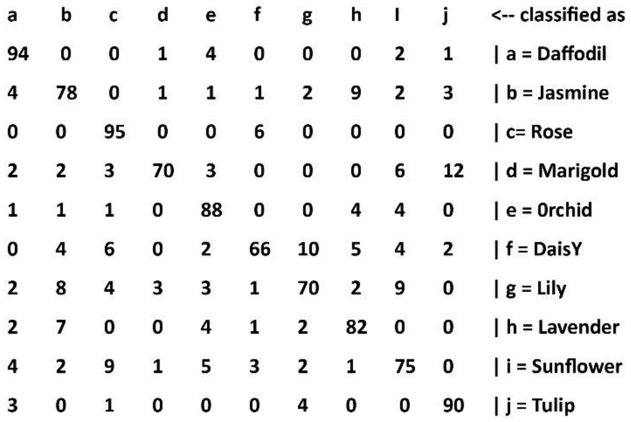
2.6.2 Value of K=5:



a	b	c	d	e	f	g	h	I	j	<-- classified as
96	0	0	1	3	0	0	0	0	2	| a = Daffodil
3	84	0	1	1	1	1	6	2	2	| b = Jasmine
0	0	95	0	0	6	0	0	0	0	| c = Rose
1	4	3	73	3	0	1	0	5	8	| d = Marigold
2	1	1	1	90	0	0	2	2	0	| e = Orchid
0	3	4	0	4	75	5	5	2	1	| f = Daisy
2	5	4	4	2	3	76	0	6	0	| g = Lily
1	7	0	0	3	2	3	82	0	0	| h = Lavender
4	2	7	3	5	3	4	1	73	0	| i = Sunflower
3	0	2	0	0	1	4	0	0	88	| j = Tulip

Fig. 3 Confusion matrix for k-value 5

2.6.3. Value of K=7:



```
a   b   c   d   e   f   g   h   I   j   <-- classified as
94  0   0   1   4   0   0   0   2   1   | a = Daffodil
4   78  0   1   1   1   2   9   2   3   | b = Jasmine
0   0   95  0   0   6   0   0   0   0   | c= Rose
2   2   3   70  3   0   0   0   6   12  | d = Marigold
1   1   1   0   88  0   0   4   4   0   | e = 0rchid
0   4   6   0   2   66  10  5   4   2   | f = DaisY
2   8   4   3   3   1   70  2   9   0   | g = Lily
2   7   0   0   4   1   2   82  0   0   | h = Lavender
4   2   9   1   5   3   2   1   75  0   | i = Sunflower
3   0   1   0   0   0   4   0   0   90  | j = Tulip
```

Fig. 4 Confusion matrix for k-value 7

Table 1. KNN accuracy for different values of k

k value	Accuracy
3	85.4%
5	83.2%
7	80.8%

Table 2. Performance Matrix for different values of k

k value	Kappa Statistic	Mean Absolute Error	RMSE	Relative Absolute Error (%)	Root Relative Squared Error (%)
3	0.8378	0.0402	0.1372	22.3337	45.7414
5	0.8133	0.0547	0.1571	30.3624	52.3695
7	0.7867	0.0625	0.1669	34.7134	55.6412

3. Results and Discussion

Initial experiments with the raw dataset produced suboptimal classification results, primarily due to unscaled features and data noise. However, after applying proper preprocessing—such as normalization, noise reduction, and redundant feature elimination—the KNN model demonstrated significantly improved performance.

3.1. Key Observations

3.1.1. Optimal k-Value Selection

While a lower value of k = 3 achieved the highest accuracy of 85.4%, it also showed signs of overfitting. In contrast, k = 5 and k = 7 provided a more balanced performance by reducing model variance, though with slightly lower accuracy.

3.1.2. Impact of Feature Selection

The exclusion of irrelevant or redundant features helped improve the classifier's ability to distinguish between similar flower types, leading to more robust results.

3.1.3. Effect of Normalization

Proper scaling of feature values significantly enhanced the effectiveness of distance-based calculations, which is critical for KNN. This contributed to better clustering of data points in multi-dimensional space.

3.2. Performance of KNN in Flower Classification

Table 3. Actual vs. Predicted Instances

Flower Type	Before Applying KNN	After Applying KNN
Daffodil	102	91
Jasmine	101	76
Rose	101	90
Marigold	98	64
Orchid	99	74
Daisy	99	64
Lily	102	64
Lavender	98	70
Sunflower	102	64
Tulip	98	75

3.3. Performance Comparison of KNN and Random Forest Algorithm

Although there are multiple machine learning classification algorithms, the effectiveness of KNN and Random Forest algorithm in classifying the flower types has been compared in this study. Both the opted algorithms are powerful and essential algorithms for performing classification and regression. [10, 11] Both algorithms follow a supervised learning approach. The only difference between these two algorithms is that the KNN algorithm works on the k-nearest Value, which points to the number of neighbor instances, whereas the Random Forest algorithm combines multiple decision trees in which features are represented as nodes. On the basis of various aspects, both the algorithms are compared. By all means of higher accuracy and lower errors, the K-nearest neighbor (KNN) algorithm is more suitable for pattern recognition that has been taken on a custom dataset for classifying the data into labelled instances.

Table 4. KNN v/s Random Forest

	KNN	Random Forest
Kappa Statistic	0.83	0.55
Mean Absolute Error	0.04	0.13
Root mean Square Error	0.13	0.25
Relative Absolute Error (%)	22.33	75.06
Root Relative Squared Error (%)	45.74	82.53

These results reveal that the KNN model performs particularly well for certain flower types (e.g., Daffodil and Rose), while others present more classification challenges—possibly due to feature overlap or insufficient differentiation in morphological traits.

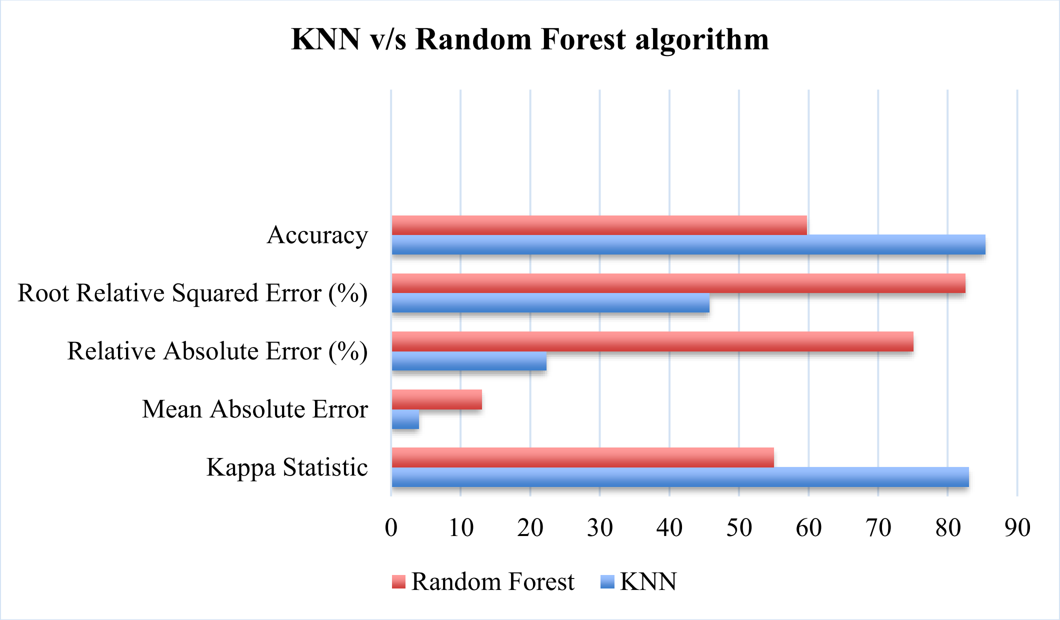


Fig. 5 Comparison of KNN with Random Forest algorithm

In summary, the results affirm the suitability of the KNN algorithm for flower classification tasks, especially when supported by appropriate preprocessing and parameter tuning. While performance is promising, some limitations related to class overlap and imbalance suggest avenues for future improvements using ensemble methods or hybrid models.

4. Conclusion

This research embodies that the K-Nearest Neighbors (KNN) algorithm can be satisfactorily used for flower classification when proper data preprocessing techniques and model tuning are provided. By working with a custom dataset and applying data mining and machine learning techniques such as feature selection, normalization, and k-value optimization, the classifier's general performance was significantly improved. Using 10-fold cross-validation accomplished a more reliable and precise accuracy evaluation and highlighted how different parameters affect the model's outcomes. Challenges like class imbalance and overfitting were addressed through observant preprocessing and training strategies, further promoting the model's strength. The study proves that with the right adjustments, KNN can deliver a strong tool for handling multi-class classification problems in real-world scenarios, including automated flower recognition and related applications.

References

[1] Amer F.A.H. Alnuaimi, and Tasnim H.K. Albaldawi, “An Overview of Machine Learning Classification Techniques,” *BIO Web of Conferences*, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Shovan Chowdhury, and Marco P. Schoen, “Research Paper Classification using Supervised Machine Learning Techniques,” *2020 Intermountain Engineering, Technology and Computing*, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Jingwen Sun, Weixing Du, and Niancai Shi, “A Survey of kNN Algorithm,” *Information Engineering and Applied Computing*, vol. 1, no. 1, 2018. [CrossRef] [Publisher Link]

[4] Wei Lv et al., “Research and Application of Intersection Similarity Algorithm Based on KNN Classification Model,” *2021 International Conference on Artificial Intelligence, Big Data and Algorithms*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Yang Gong, and Pan Zhang, “Research and Realization of Air Quality Grade Prediction Based on KNN,” *2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[6] Yaling Zhu, Jundi Wang, and Xiangwei Li, “Research on GA-KNN Image Classification Algorithm,” *2022 4th International Conference on Artificial Intelligence and Advanced Manufacturing*, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Lisha Yao, and Tiancheng Cao, “Research on Talent Classification Based on Improved KNN Algorithm,” *2022 International Symposium on Control Engineering and Robotics*, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Loredana Firte, Camelia Lemnaru, and Rodica Potolea, “Spam Detection Filter using KNN Algorithm and Resampling,” *Proceedings of the 2010 IEEE 6th International Conference on Intelligent Computer Communication and Processing*, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[9] Junhao Tan, “Stock Index Forecasting Model Based on Short-term Volatility Trend and KNN Algorithm,” *2022 8th Annual International Conference on Network and Information Systems for Computers*, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] F.Y. Osisanwo et al., “Supervised Machine Learning Algorithms: Classification and Comparison,” *International Journal of Computer Trends and Technology*, vol. 48, no. 2, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[11] Uduak Idio Akpan, and Andrew Starkey, “Review of Classification Algorithms with Changing Inter-class Distances,” *Machine Learning with Applications*, vol. 4, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] Li-Yu Hu et al., “The Distance Function Effect on k-nearest Neighbor Classification for Medical Datasets,” *Springer Plus*, vol. 5, 2016. [CrossRef] [Google Scholar] [Publisher Link]