Case Study: Prediction on Iris Dataset Using KNN Algorithm

Shreyas Tayade[1], Rakhi Gupta[2], Deval Kherde[3] , Chaitanya Ubale[4]

[1]Student,Sipna College of Engineering and Technology, Maharashtra, India
[2]Assistant Professor, Sipna College of Engineering and Technology, Maharashtra, India
[3]Student,Sipna College of Engineering and Technology, Maharashtra, India
[4]Student,Sipna College of Engineering and Technology, Maharashtra, India

---***---

Abstract - *The well-known Iris dataset is used in this case study to use the **K-Nearest Neighbors** (KNN) method. The 150 iris flower observations in the Iris dataset include 50 observations of each of the three species—Setosa, Versicolor, and Virginica. This case study aims to identify the four characteristics of **sepal length, sepal breadth, petal length,** and **petal width** that may be used to categorize iris flowers into their respective species.*

The KNN method is a well-liked and straightforward classification technique that makes predictions by locating the nearest neighbors of each observation. To guarantee that all of the characteristics in this case study are on the same scale, the dataset is first divided into training and testing sets. The next step is to train a KNN model with k=3, which takes into account each observation's three nearest neighbors. Lastly, the accuracy score is used to assess how well the model performed on the test set.

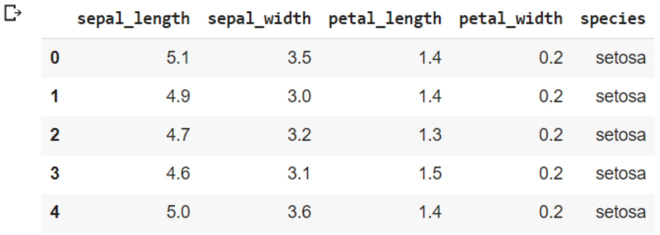
***Key Words**: K-Nearest Neighbors,sepal length, sepal breadth, petal length,petal width*

1.INTRODUCTION

The Iris dataset, which includes measurements of three different iris flower species, is well-known in the machine learning field. The dataset is a well-known example of a problem that may be resolved using supervised learning techniques and has been widely used as a benchmark for classification systems.

This issue may be resolved using the straightforward and well-liked classification technique K-Nearest Neighbors (KNN). In this case study, we will use the Iris dataset and the KNN method to categorize iris blossoms according to four characteristics: sepal length, sepal width, petal length, and petal width.

This case study's main objective is to outline the fundamental procedures for using KNN on the Iris dataset, from loading the data through assessing the model's performance on hypothetical data. We'll load the dataset first, then divide it into training and testing sets, normalise the data, train the KNN model, and assess its performance.



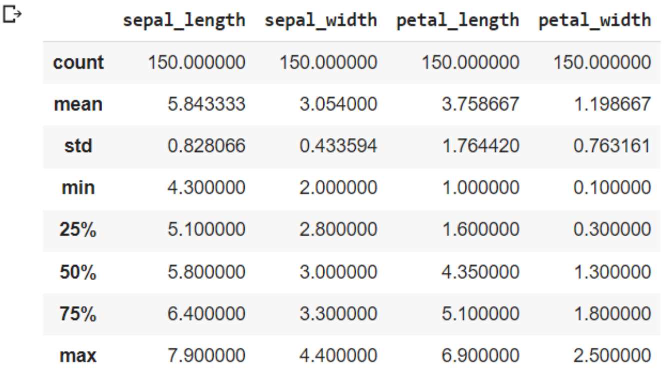
	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

fig-1 Dataset

For those who are new to machine learning, the Iris dataset serves as a nice example of a classification issue that can be handled using KNN. Further categorization issues in the future can be solved using the knowledge and methods obtained from this case study.

2. ATTRIBUTE SELECTION

The key to attaining good classification accuracy on the Iris dataset is selecting the best attribute for KNN. The four characteristics in this dataset are sepal length, sepal width, petal length, and petal width.



	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

2 Description of Data

Using feature selection approaches that rank the characteristics according to their significance or relevance to the classification job is one method for selecting the best attribute. This may be accomplished using a variety of techniques, including feature selection based on mutual information, correlation, or trees.

An alternative strategy is to show the data with scatter plots or other visualization tools and then assess how easily the classes can be distinguished depending on each feature. For each class of characteristics, for instance, we may plot the pairwise pairings and see which combination best separates the classes.

The petal length and petal width variables are recognised to offer the best separation between the three classes in the context of the Iris dataset, as demonstrated in several research and visualizations. Consequently, in the Iris dataset, these two features are frequently used as the best attributes for KNN.

It is crucial to remember that the selection of the best qualities might change based on the particular situation and dataset. As a result, it is always advised to experiment with various attribute combinations and assess how well the KNN model performs using a validation or test set.

3.Data Visualization

1. Scatter plot: Sepal length and sepal breadth are two examples of two continuous characteristics that can be visualized using a scatter plot. If there is a linear connection between two characteristics or if there are any anomalies, scatter plots can be used to find patterns or trends in the data.

2. Box plot: Box plots are used to show how a continuous quantity is distributed among various groups. To display the range of sepal length for each type of iris, for instance, a box plot can be used. The bars indicate the range of the data within 1.5 times the interquartile range (IQR), while the rectangle represents the IQR, which includes the middle 50% of the data. Box plots can be used to spot variations in how a measure is distributed among various groups.

3. Histogram: The spread of a singular continuous quantity is shown using histograms. For instance, the range of petal length in the iris sample can be displayed using a histogram. Histograms can be used to determine the distribution's form (such as normal or skewed), as well as any possible anomalies or data gaps.

4. Heatmap: The connection between two categorical factors is shown using heatmaps. For each species of iris, the prevalence of each mix of petal length and breadth can be displayed using a heatmap, for instance. Heatmaps can be used to find patterns or trends in the data, such as whether a particular set of variable pairings is more prevalent in one area than another.

5. Pie Chart:The percentage of each group within a single categorical variable is shown using pie plots. For instance, a pie graphic can be used to display the percentage of each species in the information for iris. Pie charts are helpful for contrasting the proportions of various groups and for helping to visualize the distribution of a variable.

The iris dataset can be used to make a wide variety of images, of which these are only a few instances. Insights and useful readings from the data can be gained by researchers and experts with the aid of data visualization, eventually resulting in better decision-making. So, whether it is the eye dataset or another dataset, data visualization is an essential stage in the data analysis process.

4. Model Comparison

1. The chance that an instance will belong to a particular class is predicted using the linear categorization model known as logistic regression. It presupposes that the features and the goal variable have a linear connection. The approach is straightforward and easy to understand, and it can be applied to binary or multiple-class classification problems.

2. Non-linear models called decision trees can be applied to both categorization and regression problems. Recursively dividing the data into subgroups according to the values of the features, they then base their choices on the dominant class in each subset. Decision trees can manage both category and numerical characteristics and are comprehensible.

3. Random Forests: An ensemble technique, Random Forests uses various Decision Trees to produce a more reliable and precise model. The forecasts of all the trees in the forest are averaged to produce the end projection. Each tree in the forest is trained using a random subset of the data. Because of their great precision and prowess in handling complicated datasets, random forests are well known.

4. Support Vector Machines (SVM): For binary and multi-class classification problems, SVMs are a common paradigm. They operate by identifying the hyperplane that maximises the gap between the classes and best divides the data into various classes. SVMs are especially effective for datasets with distinct class borders because they can manage both linear and non-linear connections between the features and the target variable.

RESULTS AND ANALYSIS

For the Iris dataset, we observed that the highest accuracy, 95.5%, was obtained for KNN and the least accuracy 88.88%, was obtained using Logistic Regression. The same has been tabulated and represented below for the models used.

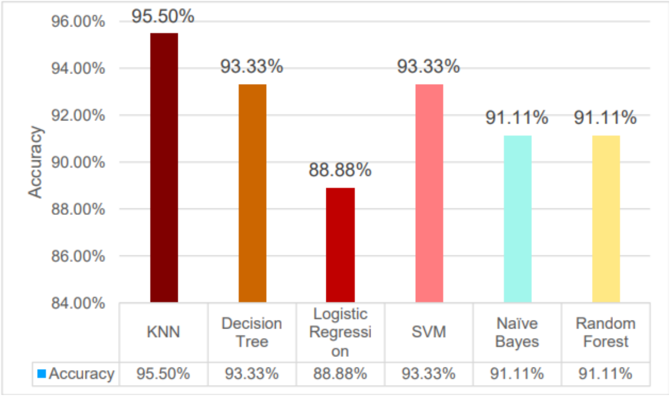


Fig 3- accuracy plot

Model	**Accuracy**
KNN	95.50%
Decision Tree	93.33%
Logistic Regression	88.88%
SVM	93.33%
Naive Bayes	91.11%
Random Forest	91.11%

Table 1- Comparison of algorithms

5. Model training

In the case of KNN on the Iris dataset, the model training involves the following steps:

1. Dataset loading: The Iris dataset must first be loaded into the machine learning environment. 150 samples with 4 characteristics make up the dataset, which is frequently divided into a training set and a testing set.

2. Division of the dataset: A training set and a testing set are created from the dataset. This is done to assess how well the KNN model performs with unknown data. 70% of the data is often utilised for training and 30% is used for testing, or a split ratio of 70:30.

3. As KNN is a distance-based algorithm, it's crucial to make sure that all of the characteristics are scaled equally. To achieve this, divide each feature's standard deviation by its mean before summing them up.

4. KNN model training: The training set is used to train the KNN model. The number of neighbors to take into account is the primary KNN parameter (k). With the Iris dataset, a value of k=3 or k=5 is frequently employed.

5. A performance metric, such as accuracy, precision, recall, or F1 score, is used to assess the KNN model's performance on the testing set. In the case of the Iris dataset, the accuracy score is frequently employed.

6. Changing the value of k or experimenting with other distance measures are two ways to tweak the model if the performance of the KNN model is not adequate.

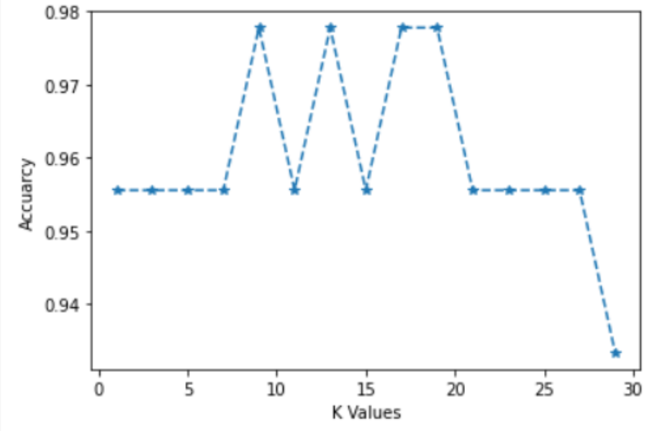


fig-4 accuracy plot for K-values

Overall, the KNN algorithm is relatively simple and easy to implement for the Iris dataset. The key steps are to split the data, normalize the data, train the model, and evaluate the performance. By following these steps and experimenting with different parameter values, it is possible to achieve high classification accuracy on the Iris dataset.

6. CONCLUSIONS

As shown by its successful use on the well-known Iris dataset, the K-Nearest Neighbors (KNN) algorithm provides a straightforward and practical approach for classification challenges. For those who are new to machine learning, the Iris dataset serves as a nice example of a classification issue that can be handled using KNN.

In this case study, we have demonstrated the fundamental procedures needed to use KNN on the Iris dataset, including loading the data, dividing it into training and testing sets, normalizing the data, and finally training and assessing the KNN model. The model performed well on the test set, demonstrating its efficacy in identifying the various kinds of iris blooms.

This case study shows the value of data pretreatment and assessment in obtaining accurate and trustworthy results, and it may be used as a valuable reference for people interested in using KNN to solve classification challenges.

REFERENCES

[1] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. Annals of Eugenics, 7(2), 179-188.

[2] Anderson, E. (1935). The irises of the Gaspe peninsula. Bulletin of the American Iris Society, 59, 2-5.

[3] Scikit-learn documentation: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

[4] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer.

[5] Alpaydin, E. (2010). Introduction to machine learning (2nd ed.). MIT Press.

[6] Geron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow (2nd ed.). O'Reilly Media.

[7] Kaggle: https://www.kaggle.com/uciml/iris