IJEIT
International Journal of Emerging Information Technology
ISSN: XXXX-XXXX / Volume 1 Issue 1 September - October 2025 / Page No: 22-31
Paper Id: IJEIT-V1I1P103

Original Article

Iris Flower Classification Using Machine Learning

M. Narasimha Rao[1], G. Srikar[2], Md. Zubair Ahmed[3], T. Sai Yashwanth[4]

Department of SCOPE, VITAP University.

[1]*Corresponding Author : narasimha.22bce9812@vitapstudent.ac.in*

Received: 14 June 2025 Revised: 26 July 2025 Accepted: 07 September 2025 Published: 25 September 2025

Abstract *- Iris flower group has 3 different species. 1. Setosa, 2. Versicolor, 3. Virginica. It is very difficult to classify the species by just looking at them and measuring their sepal and petals. Iris flower classification is useful for botany people to automate the process of Iris flower classification, so the work can be done easily, it is a foundational dataset for the learners to know and gain knowledge about classification of categories. This is very helpful for the research purposes, apart from this we can use this usecase as a benchmark for other classification problems in the real world. We explored the use of different algorithms and the advantages, disadvantages of each algorithm. Overall this study gives a thorough review of the work we have done.*

Keywords *- Botany, Setosa, Versicolor, Virginica.*

1. Introduction

Iris is a flowering plant which has about 310 accepted Species. These flowers are grown in dry climates, from bulbs. They have long erect stems. There are 3 main species of iris. They are Setosa, Versicolor and Virginica.

Fig. 1 Iris Versicolor. Source: [1]

This is one of the 3 Different main species of Iris. Iris- versicolor. In this study we are going to use different machine learning algorithms to learn and predict. Then we are going to select the model which gives the maximum accuracy. In this study we are going to show the work we had done on the dataset and the process we followed to achieve the output and different models performance.

2. Literature Review

Botany and machine learning research have both delved deeply into the classification of iris flower species. A number of studies have focused on incorporating classification models into online applications for greater accessibility. Researchers have investigated numerous methodologies and techniques to accurately categorize flowers of iris based on their petal and sepal properties and how they contribute towards the classification. Some significant contributions in these fields are highlighted in this literature review.

2.1. Fisher's Dataset

Fisher's Iris dataset, written by Ronald Fisher in 1936, is one of the foundational studies in the taxonomy of iris flower species . This collection contains 150 examples of iris flowers, each with measurements of the sepal length, sepal width, petal length, and petal width, in addition to the associated species designations. Using this dataset, Fisher created the linear discriminant analysis (LDA) and showed how well it could differentiate between different Iris flower species. [2]

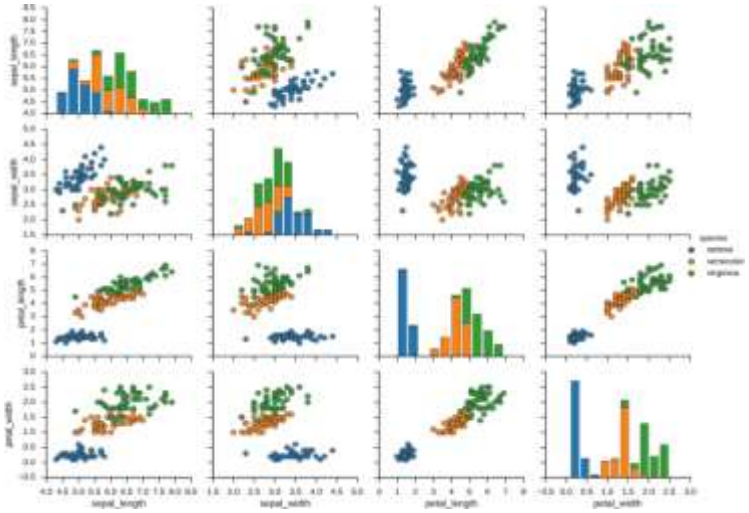

Fig. 2 Fisher's Iris Dataset. Source: [2]

2.2. KNN, Clustering Research

Knn model is K - Nearest Neighbours algorithm. This algo- rithm classifies the species by finding the euclidean distance of the given point to all the points in the dataset and uses voting to select the class which the given data point belongs to. And hence gives the result. Next they also used unsupervised learning to make clusters using K means clustering. It provided different benefits to classify.

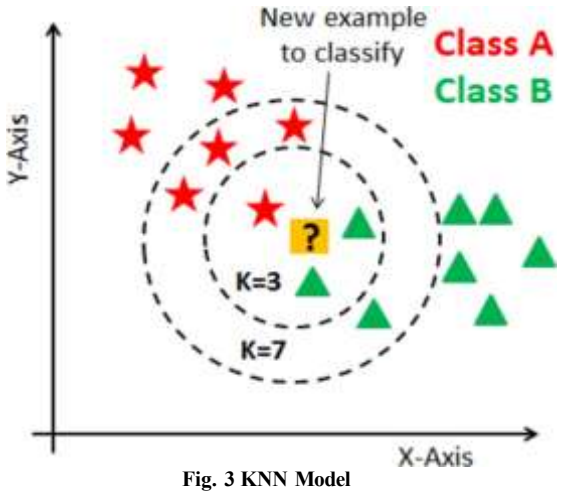

Fig. 3 KNN Model

2.3. SVM, ANN AND RF

Gonti Suchitra has developed a machine learning model using 3 Different models that are SVM, ANN and RF [4]. The model takes the data set and is trained by 3 models and based on the accuracy the best model is saved and used for classification.

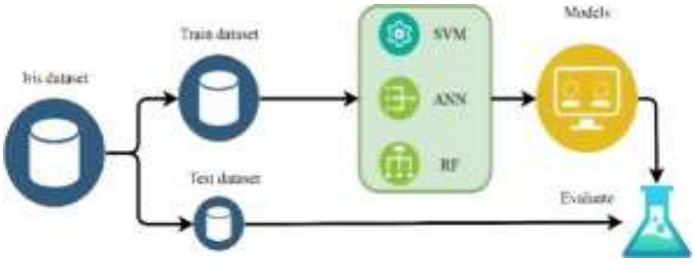

Fig. 4 Model. Source: [4]

This is the methodology she used. It might be more accurate if some more models are tested.

2.4. Multi Model

An Iris classification model is made by using DT, RF, NB, LR, KNN and Linear SVC classifiers. They chose the Linear SVC Model which gave the highest Accuracy. [7]

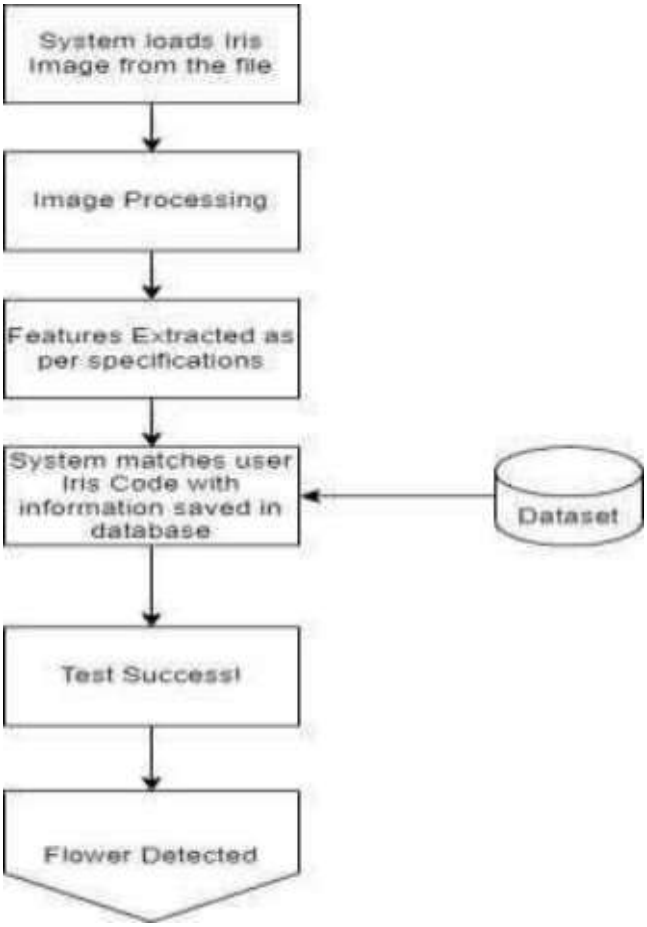

Fig. 5 Work Flow. Source: [7]

2.5. KNN and LR Model

A Model is developed where the developers used KNN and LR models to classify the Iris flowers. They have used a dataset with 150 samples and splitted them into train and test data and trained KNN and LR models with that data. In their model KNN with K=5 gave the highest accuracy for them. [8]

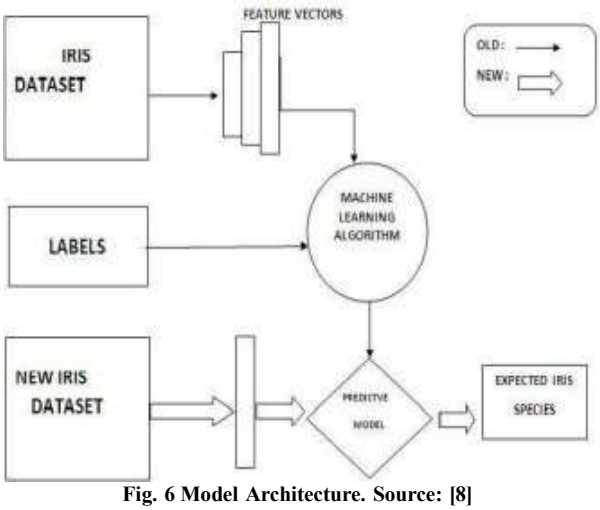

Fig. 6 Model Architecture. Source: [8]

3. Methodology

3.1. Dataset

We have used kaggle's iris dataset made by Manimala. The data contains petal length, petal width, sepal length and sepal width. There are 150 samples of data.

The first 5 samples of the data :

	sepal_length	sepal_width	petal_length	petal_width	Target
0	4.9	3.0	1.4	0.2	Iris-setosa
1	4.7	3.2	1.3	0.2	Iris-setosa
2	4.6	3.1	1.5	0.2	Iris-setosa
3	5.0	3.6	1.4	0.2	Iris-setosa
4	5.4	3.9	1.7	0.4	Iris-setosa

Fig. 7 Dataset

3.2. Data Pre-Processing

We have generated a correlation matrix for all the features and we have generated a heatmap representing the correlation among all possible combinations of features, it is also calles as Multivariate analysis. We found all features are correlated so we kept all features. Sepal width is least correlated with other features but it is somewhat correlated with sepal length. In the Heat map, the lighter the color both the features are more correlated, darker the color they are less correlated. Let us consider with the target variable,

- Target is much correlated with petal length
- Correlation with the target is in the order of petal length, sepal length, petal width, sepal width.

Next we found some outliers in the sepal width feature so we have removed those outlier points. We also plotted the pair plot for all the features in the data set with separating the colors of the points with respect to the target column. The plot shows that, the data points of particular categories gather at one place in feature space, this helps us to classify the dataset easily using the alogithms like SVM, KNN.

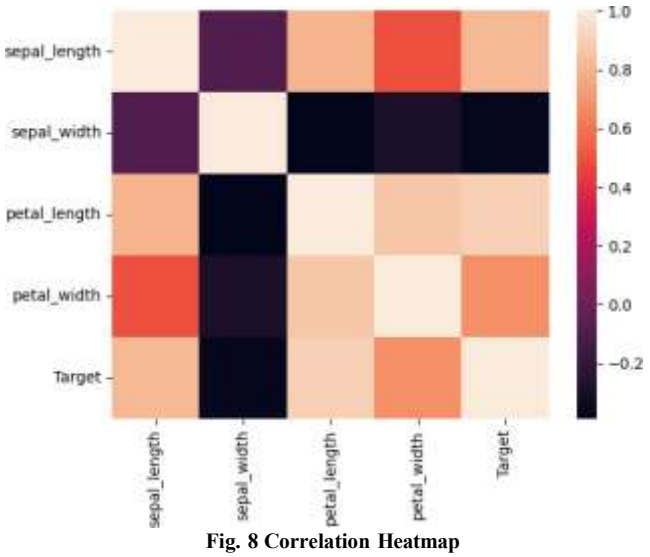

Fig. 8 Correlation Heatmap

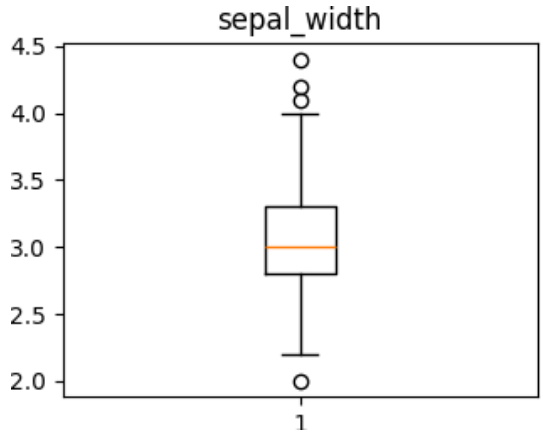

Fig. 9 Box Plot

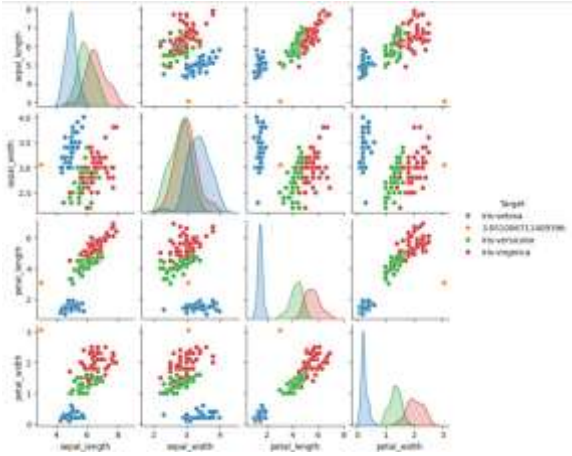

Fig. 10 Pair plot

4. Model Selection

In this part we trained different models with the dataset, we had split the dataset into train(70 percent) and test(30 percent) sets using ”train-test-split” method in sklearn library.

4.1. Naive Bayes Classifier

Naive Bayes is a supervised Machine Learning algorithm, mostly used in classification. For the new data point to be clas- sified, it calculates the probability using contingency tables for each column with respect to the target column. Based on the highest probability value for the category in the Target column. It uses the formula, Bayes theorem $P(yX)=P(X)P(Xy)P(y)$ where, y is class variable and X is a dependent feature (of size n) where: X = (x1,x2,x3,. ,xn).

We trained a model with our dataset, using Naive Bayes, on testing the model using test-set we got ”Accuracy of 80 percent.” And we have drawn confusion matrix to visualize the model performance.

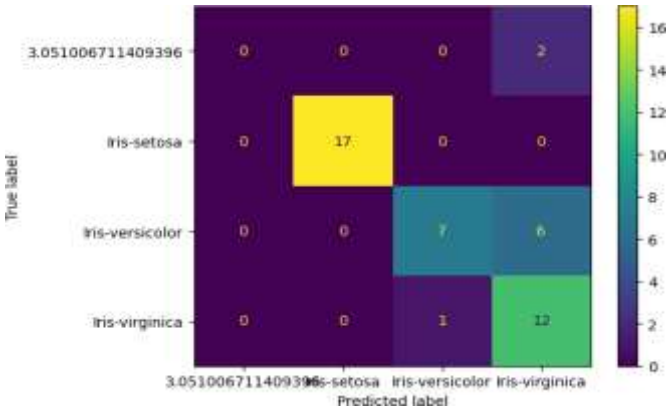

Fig. 11 NB Confusion Matrix

4.2. Decision Tree Classifier

Decision tree is a Machine Learning and supervised al- gorithm, used for both regression and classification. This algorithm constructs a tree, it has a hierarchical structure with root node, branches, leaf nodes, internal nodes. Given the dataset, it chooses the root node using different techniques (Gini index , Information gain/entropy, etc..) and continues to split the dataset into subsets until it reaches the stopping criteria or leaf nodes.

Gini index formula, high value corresponds high significance,

$$\text{Gini} = 1 - \sum (p_i)^2$$

Entropy formula, high entropy corresponds high significance,

$$\text{Entropy} = -\sum p_i \log_2(p_i)$$

When there is a new data point to classify, it traverses the entire tree and predict the output. Our trained model got ”86.67 Accuracy” and here is the confusion matrix visualization to know how the data is classified.

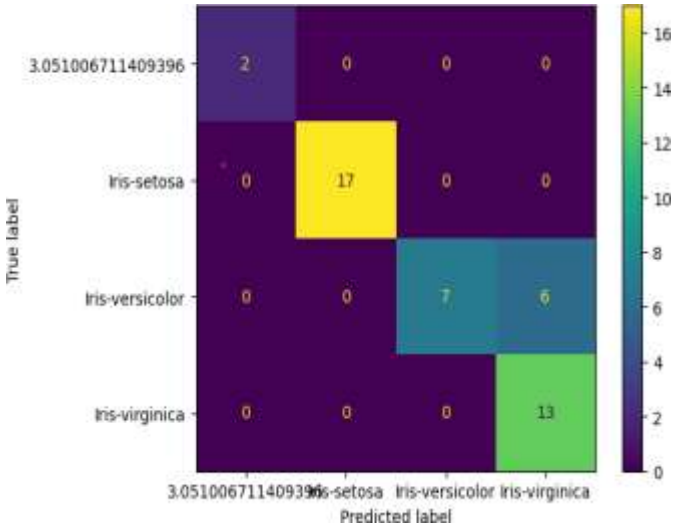

Fig. 12 DT Confusion Matrix

4.3. K Nearest Neighbors / K Means Classifier

KNN is a simple, supervised Machine Learning algorithm, used for both classification and regression tasks. It classified with the idea that similar data points are close to each other. For the new data point to be classified it identifies K number of closest points to the given point in the entire dataset, now based on voting it classifies the point into its particular class. Distance metrics used are,

Euclidean Distance:

$$d(p, q) = \sqrt{\sum (q_i - p_i)^2}$$

Manhattan Distance

$$d(p, q) = \sum |q_i - p_i|$$

We trained a model using KNN, on testing the model with test-set we got ”Accuracy of 88.88 percent.” And we have drawn confusion matrix to visualize the model.

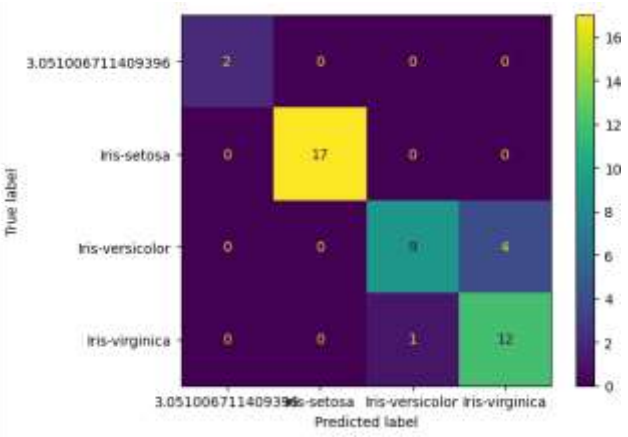

Fig. 13 KNN Confusion Matrix

4.4. Support Vector Machine(SVM)

SVM is a Machine Learning, supervised Algorithm primar- ily used for classification, and can also be used in regression. SVM finds the Hyperplane that separates the data points in different classes in a dataset. If there are multiple hyperplanes which separates the datapoints, the optimal one which maxi- mizes the margin is chosen and the datapoints which are on or near to hyperplane are called support vectors.

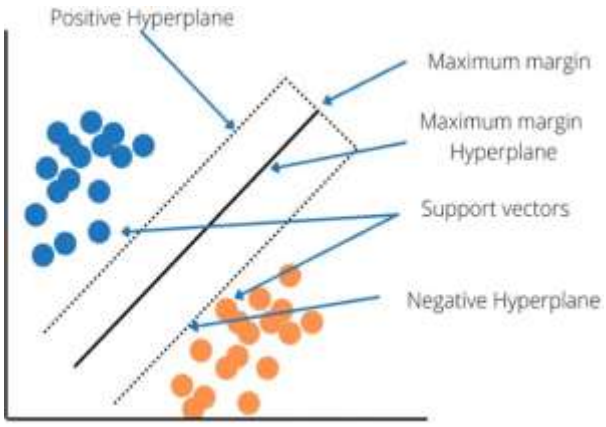

Fig. 14 Implementation Diagram. Source: [9]

Next we trained a model using SVM classifier, and predicted the outputs on test set, we got the ”accuracy of 91.11percent” and here is the visualization(Fig. 14) of the models confusion matrix

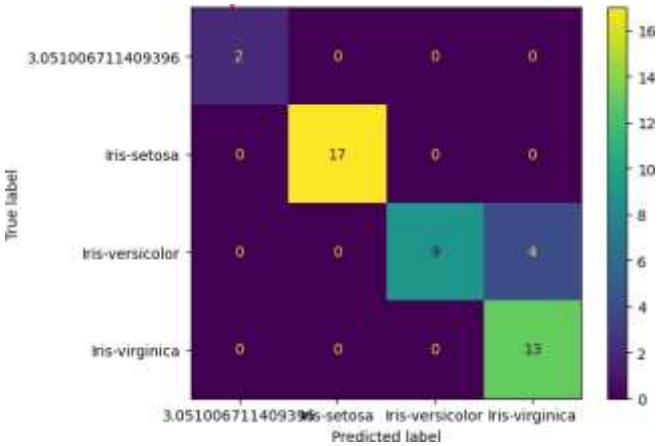

Fig. 15 SVM Confusion Matrix

4.5. Random Forest Classifier

Random Forest is a ensemble learning algorithm, particu- larly under bagging techniques, used for classification tasks. The entire data set is converted into subsets randomly and train multiple decision trees, which improves accuracy and overfitting. The algorithm now combines all the results of all trees and predicts the output.

We trained a model using Random Forest classifier and we got ”accuracy of 86.67 percent.” Below is the visualization of the confusion matrix for the random forest model we trained.

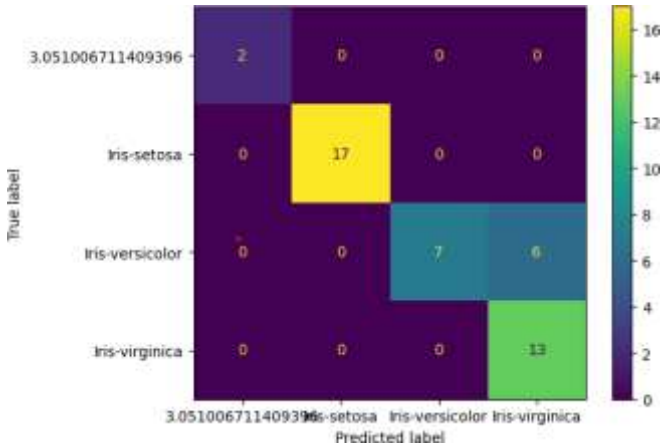

Fig. 16 Random Forest Confusion Matrix

4.6. XG Boost

XGBoost also called as Extreme Gradient Boosting, is an ensemble learning algorithm based on gradient boosting, used for both classification and regression. XGBoost builds decision trees sequentially by correcting the errors in the previous trees. XGboost optimises the loss fuction, for regression it optimises Mean Sqaured Error and for classification tasks it optimises the log-loss.

The accuracy we got for this algorithm is 84.44 percent. The confusion matrix visualization shows how xgb model classified the datapoints.

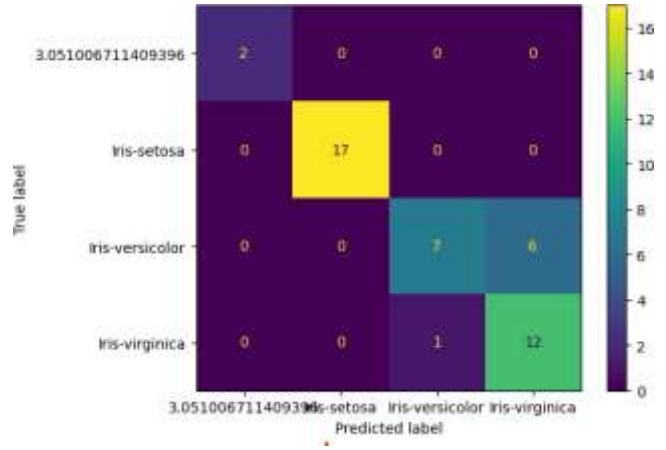

Fig. 17 XGB Confusion Matrix

5. Result

5.1. Finalizing The Model

Now, we have drawn comparison graphs for model accu- racies, and finally, we got the highest accuracy for the SVM Model, so we selected and saved the SVM Model.

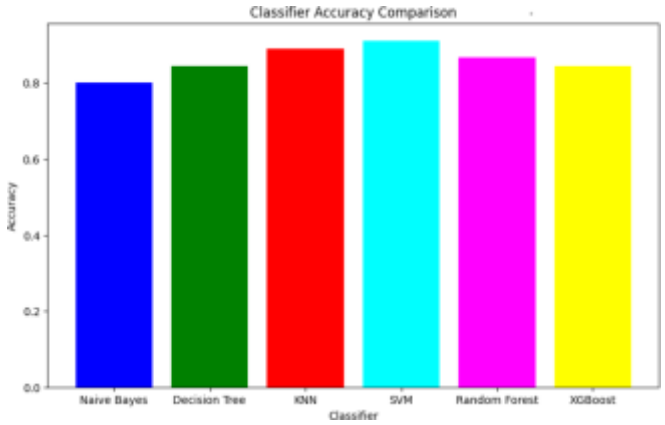

Fig. 18 Comparison Graph

5.2. Block/Architecture Diagram

- We first collected dataset
- Preprocessed the data by Removed some outliers, there are no duplicates and null values.
- We divided the dataset into Train set and Test set.
- Then trained the models Naive Bayes, Decision Tree, K-Nearest Neighbors, Support Vector Machine(SVM), Random Forest and XGB Classifiers with train set.
- Next tested the models with Test set choosing Accuracy as a metric to compare.
- Finally came to a conclusion. The Architecture diagram is shown in fig 18.

Fig. 19 Architecture Diagram

6. Conclusion

In this project, we used multiple classification algorithms to classify Iris flower into three categories Setosa, Versicolor, Virginica based on the features of the flower such as Sepal length and width, Petal length and width. Here is the Accuracy of each classification model we trained.

Based on the above results, SVM performs better than other models for classifying the Iris data with highest accuracy.

Classifier	Accuracy
Naive Bayes	0.8
Decision Tree	0.8667
KNN	0.8889
SVM	0.9111
Random Forest	0.8667
XGB	0.8444

However, KNN, Random Forest has good accuracy we can choose the model based on our requirements. Since SVM performs best we had choosen this model. Futher improve- ments can be made using Hyperparameter Tuning or adding additional features to improve accuracy.

List of Acronyms

Abbreviation	Full Form
ML	Machine Learning
SVM	Support Vector Machine
KNN	K Nearest Neighbors
DT	Decision Tree
NB	Naive Bayes
RF	Random Forest
XGB	Extreme Gradient Boosting

References

[1] Wikipedia contributors, ”Iris versicolor,” *Wikipedia*. Available: https:// en.wikipedia.org/wiki/Iris versicolor.

[2] ResearchGate, ”A scatter matrix for Fisher’s iris dataset,” Available: https://www.researchgate.net/figure/A-scatter-matrix-for-Fishers-iris-data-set-with-histograms-for-each-variable-on-thefig5 338774679.

[3] T. Deepak, ”Introduction to K-Nearest Neighbors (KNN) Algorithm,” *Plain English*. Available: https://ai.plainenglish.io/ introduction-to-k-nearest-neighbors-knn-algorithm-e8617a448fa8.

[4] G. Sucharitha, ”Iris classification using Machine Learning,” *IJRPR*. Available: https://ijrpr.com/uploads/V4ISSUE12/IJRPR20661.pdf.

[5] IRJMETS, ”Iris Flower Classification,” *IRJMETS*, vol. 7, July 2024. Available: https://www.irjmets.com/uploadedfiles/paper/issue 7 july 2024/60416/final/fin irjmets1721672252.pdf.

[6] IJHSSM, ”A Detailed Review on Iris Flower Classification using Machine Learning integrated with Flask,” *IJHSSM*. Available: https://ijhssm.org/issue dcp/A%20Detailed%20Review%20on%20Iris% 20Flower%20Classification%20using%20Machine%20Learning% 20integrated%20with%20Flask.pdf.

[7] G. Ahuja, M. Aggarwal, J. Tyagi, O. Mehra, ”Identification of Different Species of Iris Flower Using Machine Learning Algorithms,” *IRJET*, vol. 10, no. 1, 2024. Available: https://www.irjet.net/archives/V10/i1/ IRJET-V10I175.pdf.

[8] T. Srinivasarao, ”Iris Flower Classification Us- ing Machine Learning,” *ResearchGate*. Available: https://www.researchgate.net/profile/Tumma-Srinivasarao/ publication/359064863 Iris Flower Classification Using Machine Learning/links/622632fb3c53d31ba4af19e1/ Iris-Flower-Classification-Using-Machine-Learning.pdf.

[9] Hands-on.Cloud, ”SVM-Python-Tutorial,” Available: https://hands-on. cloud/svm-python-tutorial/.

[10] Medium, ”Introduction to K-Nearest Neighbors (KNN) Algorithm,” Available: https://ai.plainenglish.io/ introduction-to-k-nearest-neighbors-knn-algorithm-e8617a448fa8.

[11] J. S. Park, ”A Comprehensive Overview of Fisher’s Iris Dataset,” *Machine Learning Mastery*, Available: https://machinelearningmastery. com/a-comprehensive-overview-of-fishers-iris-dataset/.

[12] H. Smith, ”Using Iris Dataset for Classification,” *Towards Data Science*, Available: https://towardsdatascience.com/ using-iris-dataset-for-classification-6563950c4a31.

[13] M. Mu¨ller, ”Understanding the Iris Dataset in Data Science,” *Data Science Central*, Available: https://www.datasciencecentral.com/profiles/ blogs/understanding-the-iris-dataset-in-data-science.

[14] K. P. James, ”K-Nearest Neighbors Algorithm on Iris Dataset,” *Ana- lytics Vidhya*, Available: https://www.analyticsvidhya.com/blog/2021/09/ k-nearest-neighbors-algorithm-on-iris-dataset/.

[15] A. Gupta, ”Iris Dataset and its Applications in Machine Learning,” *Medium*, Available: https://medium.com/analytics-vidhya/ iris-dataset-and-its-applications-in-machine-learning-7b3b4edcba61.

[16] R. Singh, ”How to Use the Iris Dataset for Classification Problems,” *GeeksforGeeks*, Available: https://www.geeksforgeeks.org/ how-to-use-the-iris-dataset-for-classification-problems/.

[17] S. H. Lee, ”A Beginner’s Guide to Classifying Iris Flow- ers Using Python,” *Real Python*, Available: https://realpython.com/ beginner-guide-to-classifying-iris-flowers-using-python/.

[18] S. J. Anderson, ”Support Vector Machines and Iris Dataset,” *Kaggle*, Available: https://www.kaggle.com/learn/support-vector-machines/.

[19] A. Sharma, ”Machine Learning Algorithms on Iris Dataset,” *DataFlair*, Available: https://data-flair.training/blogs/ machine-learning-algorithms-on-iris-dataset/.

[20] D. Kumar, ”Exploring the Iris Dataset for Machine Learning,” *Ana- lytics Vidhya*, Available: https://www.analyticsvidhya.com/blog/2020/11/ exploring-the-iris-dataset-for-machine-learning/.