Prediction of Iris Species using 4 Classification Algorithms from Machine Learning

*Note: with dataset from UCI Machine Learning Repository

Ratheeshwaraa Machine Learning Intern
AI Technologies and Systems
ai-techsystems.com
ratheeshwaraa007@gmail.com

***Abstract*— Machine learning is the scientific study of algorithms and statistic model where the machines are used to perform specific task without being explicitly programmed. It can predict any object or type which is manually done by human. In order to predict something, the computer need to be trained with training data. Here we have taken Iris Species dataset from UCI Machine Learning Repository and going to predict the Iris Species by training the input dataset. The prediction can be done here with 4 classification algorithms such as Logistic Regression, K- Nearest Neighbors (KNN), Random Forest, Decision Tree. Then we after building the machine learning model it can able to predict the Iris Species correctly. And we are going to compare the results of various algorithms.**

Keywords*— *Decision Tree, Logistic Regression, K-Nearest Neighbors (KNN), Random Forest, Iris Data, Machine Learning.

I. Introduction

Iris is a flower and its originated from Greek. The flower is named as Iris in Greece because the flower has a shape of human eye in nature. There are various species of Iris Flower based on their color shape and petal size. Iris Flower data or Fisher's Iris data set is a multivariate data was introduced by British Statistician and biologist named Ronald Fisher. The dataset contains 50 samples from each of three species of Iris. We have four features for this dataset. With this dataset as input our model is going to be developed with various classification algorithms.

And this model development has six stages to be processed before its prediction they are:

1. Preparation of data.
2. Feature Engineering of dataset.
3. Visualization of dataset.
4. Training and testing of dataset.
5. Choosing the classification algorithm, to build the model.
6. Evaluate the Model.

We have 3 classes of Iris Species from 50 samples of each out of 150 Samples of data. This data is spitted into two parts:

1. Training data.
2. Testing data.

And the training data as input to the model and Testing data as output is Evaluated with its result.

II. Pre Requistes

To implement our machine learning model, we need the following as the pre requisites to run our model successfully they are:

1. CPU core: i3 (minimum)
2. RAM: 4GB (minimum)
3. Jupyter Notebook
4. Necessary Packages

A. Preparation of data

The data is taken from UCI Machine Learning Repository and it is imported to the Jupyter Notebook environment as a data frame using Python packages such as NumPy and Pandas. Using Pandas, the dataset can be viewed as a tabular format.

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa

Fig. 1.

From the above Iris dataset, we can able to see the dataset with 4 features such as SepalLengthCm, SepalWidthCm, PetalLengthCm, PetalWidth are the independent variable and the Species is considered as target variable (dependent variable) where the actual prediction is going to be happening.

B. Feature Engineering of dataset

Feature Engineering is the process of creating new features for the data by adding or removing existing features to generate features that make machine learning algorithms works better. Here we have removed the feature id.

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa

Fig. 2.

C. *Visualization of dataset.*

The data visualization is the process of visualizing the dataset in graphical form. Here we have visualized the dataset using the package called as matplotlib and seaborn. The visualization is done for the dataset with three classes of data they are Iris Setosa, Iris Vercicolor and Iris Verginicolor. Here each color represents separate class of Iris Flower based on their features visualization is done using box plot.

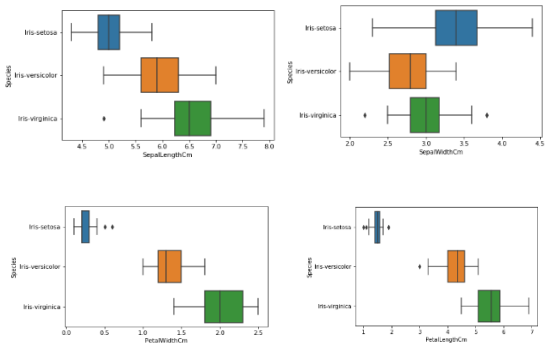

Fig. 4.

D. *Training and testing of dataset.*

The dataset is spitted with training and testing dataset. Here the independent variable is separated from the dependent variable so that it can be used for test dataset to predict the result.

```
array([[5.1, 3.5, 1.4, 0.2],
       [4.9, 3. , 1.4, 0.2],
       [4.7, 3.2, 1.3, 0.2],
       [4.6, 3.1, 1.5, 0.2],
       [5. , 3.6, 1.4, 0.2],
       [5.4, 3.9, 1.7, 0.4],
       [4.6, 3.4, 1.4, 0.3],
       [5. , 3.4, 1.5, 0.2],
```

```
array(['Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
```

Fig. 5.

E. *Choosing the Classification Algorithm*

We build the model using the classification algorithms from the Sci-kit learn package. Choosing the algorithm for developing the model is essential to get the prediction. The classification part using algorithm is all done after the data preprocessing, visualization, training and testing of dataset.

a. Logistic Regression Classifier.

So, as a start we have picked Logistic Regression Classifier as the algorithm to get the Iris Species prediction using target value which is trained with input. Logistic Regression is a classification algorithm which describe data to explain relation with one dependent variable and more than one independent variable. And it's found that Logistic Regression is predicting the Species of Iris Flower with accuracy of 97.32%. Which is 1% improved from the Author [1].

```
logreg = LogisticRegression()
logreg.fit(X_train, y_train)
Y_pred = logreg.predict(X_test)
acc_log = round(logreg.score(X_train, y_train) * 100, 2)
print("Accuracy using Logistic Regression:", acc_log)
```

```
Accuracy using Logistic Regression: 97.32
```

Fig. 6.

b. K- Nearest Neighbor Classifier

The k- Nearest Neighbor algorithm is a non parametric method used for both classification and regression problem. An object classified by its neighbor is being assigned to the class which is most to its k- nearest neighbors. Secondly we have choosed K- Nearest Neigbor Classifier as the algorithm to predict the Iris Flower Species. And it's observed that the prediction from this algorithm is as same as the accuracy from the Logistic Regression. Which is 0.6% improved from the Author[2].

```
knn = KNeighborsClassifier(n_neighbors = 3)
knn.fit(X_train, y_train)
Y_pred = knn.predict(X_test)
acc_knn = round(knn.score(X_train, y_train) * 100, 2)
print("Accuracy using K-Nearest Neigbhor:", acc_knn)
```

```
Accuracy using K-Nearest Neigbhor: 97.32
```

Fig. 7.

c. Random Forest Classifier

The Random Forest or random decision forests are the method used in Classification and regression problem. Here it operates by constructing multitude of decision trees at training time and outputting target class as individual tree. Thirdly we have selected Random forest as our algorithm to predict the the Iris Flower Species. It's observed that using Random Forest Classifier the accuracy is found as 100%. Which pretty not acceptable because Random Forest has an habit of overfitting the training data set and tries to give result almost less prediction error.

```
from sklearn.ensemble import RandomForestClassifier
random_forest = RandomForestClassifier(n_estimators=100)
random_forest.fit(X_train, y_train)
Y_pred = random_forest.predict(X_test)
random_forest.score(X_train, y_train)
acc_random_forest = round(random_forest.score(X_train, y_train) * 100, 2)
print("Accuracy using Random Forests:", acc_random_forest)
```

```
Accuracy using Random Forests: 100.0
```

Fig. 8.

And here we choose estimator value as 100 in random forest object to predict the result.

d. Decision Tree Classifier

The Decision Tree Classifier uses a decision tree to make decision about the target variable. Here target variable can take discrete set of values called classification trees in classification problem. And last but not least we have selected Decision Tree as the algorithm to predict the target variable from the Iris Flower Dataset. And it is examined that both Decision tree and Random Forests are giving the prediction as 100%. Even in Decision tree algorithm we didn't get any prediction error. So it has also overfitted the training data.

```
from sklearn.tree import DecisionTreeClassifier
decision_tree = DecisionTreeClassifier()
decision_tree.fit(X_train, y_train)
Y_pred = decision_tree.predict(X_test)
acc_decision_tree = round(decision_tree.score(X_train, y_train)*100, 2)
print("Accuracy using Decision Tree is:", acc_decision_tree)
```

```
Accuracy using Decision Tree is: 100.0
```

Fig. 9.

And here we have used 4 classification algorithm on Iris flower dataset to predict the result.

F. To Evaluate the Model.

Now we are in the final stage that is to evaluate the model by choosing any one or more classification algorithm which we have used. And the evaluation is done on target variable Species. Here we have used K-NN and Decision tree as the algorithm to evaluate our machine learning model.

a. Using K- Nearest Neighbor Classifier.

Here, after training the dataset and by building the model to examine the Iris Species Class we have given sample values as input to predict the type of Iris- Species. It is observed that our model is giving better result and prediction using K-Nearest Neighbor.

```
knn = KNeighborsClassifier(n_neighbors=12)
knn.fit(x, y)
#making prediction for sample_values for observation
knn.predict([[6, 3, 4, 2]])
```

```
array(['Iris-versicolor'], dtype=object)
```

Fig. 10.

b. Using Decision Tree Classifier.

And we have made an attempt to evaluate the model with another classification algorithm that is 'Decision Tree'. Its is found that prediction of Iris Flower Species is good in Decision Tree.

```
In [52]: decision_tree.predict([[4,5,6,7]])
Out[52]: array(['Iris-virginica'], dtype=object)
```

Fig. 11.

G. Comparing the Accuracy for 4 Classifiers.

After all the six process our model is ready to predict the Iris Flower Species. Here we have made a Comparison table for the 4-classification algorithm which we have used to predict the Iris Flower species is given below.

Out[65]:

	Model	Score
2	Random Forest	100.00
3	Decision Tree	100.00
0	KNN	97.32
1	Logistic Regression	97.32

Fig. 12. (Comparison Table of 4 Classifier Algorithm for Iris Flower Prediction).

H. Conclusion

In this paper we have implemented a machine learning model which can predict the Iris Flower Species correctly. We have also compared the results of 4 classification algorithm. And finally, we are concluding that Decision tree and Random forests are giving more accuracy than others.

REFERENCES

[1] Shashidhar T Halakatti1, Shambling T Halakatti2, "Identification of Iris Flower Species Using Machine Learning", International Journal on Computer Science and Engineering (IJCSE), volume 5, Issue 8, August 2017

[2] https://towardsdatascience.com/python-for-data-science-from-scratch-part-iii-7755f6defcc3

[3] https://www.kaggle.com/jchen2186/machine-learning-with-iris-dataset

[4] Iris Flower Dataset from UCI Machine learning Repository.