Flower Species Classification basesd on Iris Dataset

Suryash Chakravarty, Hooman Esteki & Bright Arafat Bello

Table of contents

Github URL: https://github.com/hoomanesteki/iris-ml-predictor

Summary

In this analysis we developed a classification model by utilizing the famous Iris dataset Fisher (1936). The features of the iris flowers: sepal length, sepal width, petal length, and petal width were the basis on which a Decision Tree Classifier was used for prediction. In order to check its performance, the model was first trained on one part of the dataset and then validated on another part (test set).

The outcome of our model was quite impressive, as it reached a very high accuracy (86.67%) on the test set.

The significance of this analysis is mainly associated with the Iris dataset which is considered to be one of the best datasets for introducing basic supervised learning concepts. It is easy but meaningful to see how numerical features can be used to separate different classes.

On the other hand, one can't ignore the limitations of this work as well. The size of the dataset (150 samples) is relatively small which could affect the generalization of our results over the whole population. Furthermore, only one model (DecisionTreeClassifier) was evaluated with very slight tuning; hence, if cross-validated model selection or advanced algorithms were used, better performance might be attained.

Introduction

For this analysis, the Iris dataset was chosen, a well-known dataset in both machine learning and statistics. Iris flowers are the subjects of the dataset, which contains 150 samples. Each flower is represented by four attributes: sepal length, sepal width, petal length, and petal width. The species of the iris flower is the target variable, which can be one of three species: Iris setosa, Iris versicolor, or Iris virginica.

The columns of the dataset are:

`sepal length`, `sepal width`, `petal length`, `petal width`, `class`

The main task of the present analysis is to create a classification model that predicts the species of an iris flower solely based on its features with high accuracy. A DecisionTreeClassifier model will be applied to make this prediction and we will give a summary of the results obtained from this model, including its accuracy on the test data.

Revealing the relationships among the features in this dataset has a bearing on the data characteristics since the Iris dataset is widely used to show the basic ideas of classification tasks. Furthermore, it helps to visualize how the differences in feature distributions influence the model's discriminative power between classes.

Additionally, the small dataset size and the overlapping feature distributions, particularly between the classes versicolor and virginica, limit the model's performance. Consequently, these limitations should be taken into account when interpreting the results.

Methods and Results

First, we obtained our data from the UC Irving machine learning repository Fisher (1936) and loaded it into our environment for analysis under the alias `iris`.

Our preliminary EDA revealed 150 non null values in each column, and a total of 150 rows. Each class had exactly 50 instances. The absence of null values is good, very good.

Using the pointblank package, we performed some validation on our data setting up rigorous rubrics to ensure we worked only with the data that passes these tests. This is the dataset we will use for the final stage of model building.

Before we get into the fun stuff, we split our data into training and test samples using the famous sklearn library in python Pedregosa et al. (2011), to ensure total separation of the data for training and testing. This ensures that there is no leakage of data between the two samples, and our models integrity is maintained. We also converted our target variable to numeric instead of character, assigning labels of 0,1 and 2 to 'setosa','versicolor' and 'virginica' respectively.

Enough raw text, lets see what the spread looks like for each class. These plots were generated using the amazing seaborn plotting library. Waskom (2021)

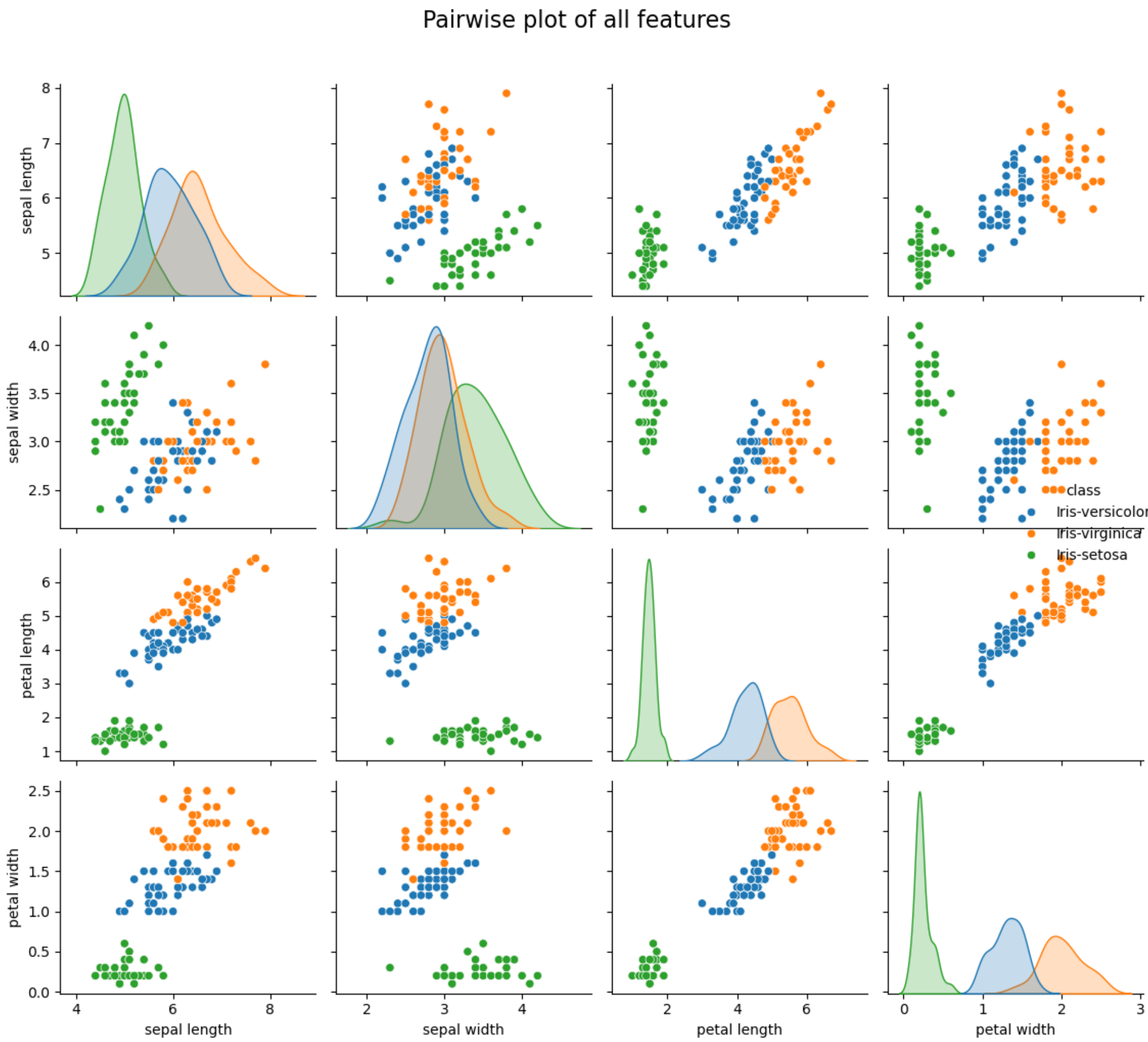

Figure 1: Distribution of each flower class

We can see from Figure 1 above, that `setosa` has the smallest petal width and length while `virginica` has the largest.

Is there a correlation between our features? Lets see.

We see a strong correaltion between petal length and sepal length in Figure 2. There is also a strong correlation between `PetalWidthCm` and `PetalLengthCm`. This implies that the wider a petal is, the longer it also could be.

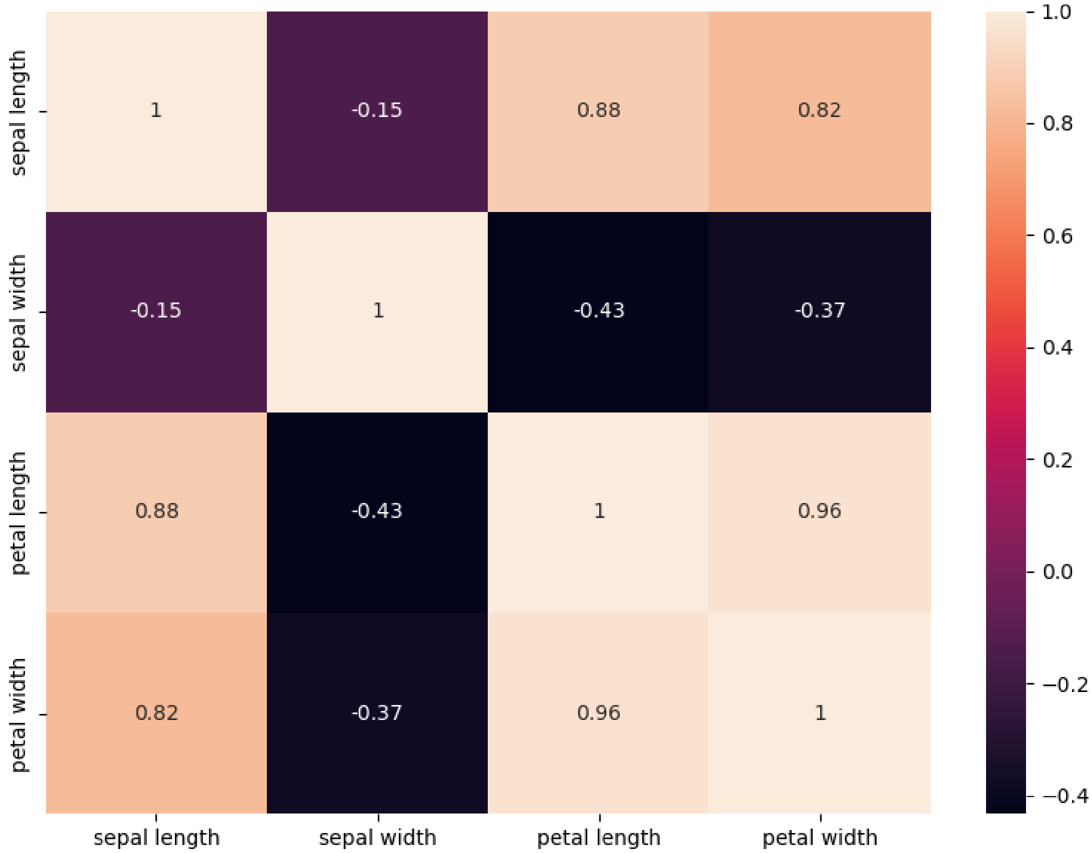

Figure 2: Correlation heatmap

Lets investigate what the distribution of Petal Length looks like for each of our species

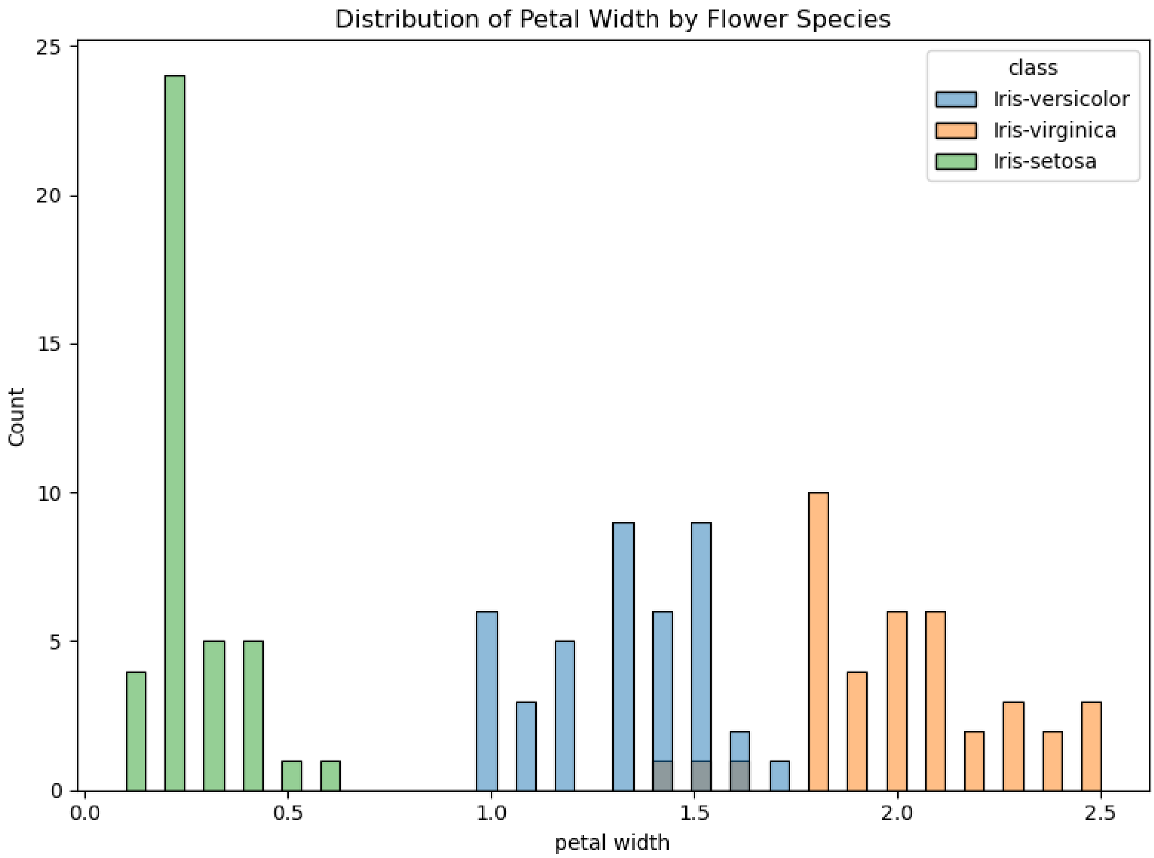

Figure 3: Petal length distribution

From Figure 3, we can see that `Setosa` flower has the smallest petal size while `Virginica` is the largest.

Now that the the EDA is done and we have a sense of our data, lets fit a Decision Tree Classifier to our data.

We start with a `DummyClassifier` model, to serve as a baseline to compare our model against.

The dummy classifier achieved an accuracy of 0.33 on the test set, which is expected since it randomly predicts one of the three classes.

With that in mind, lets fit a Decision Tree Classifier to our data.

Table 1

	accuracy	precision_weighted	recall_weighted	f1_weighted
0	0.866667	0.866667	0.866667	0.866667

We see that the decision tree classifier achieves an accuracy of approximately 86.67% on the test set, which is a significant improvement over the dummy classifier. This indicates that the decision tree model is able to effectively capture patterns in the data to make accurate predictions about the species of iris flowers based on their features.

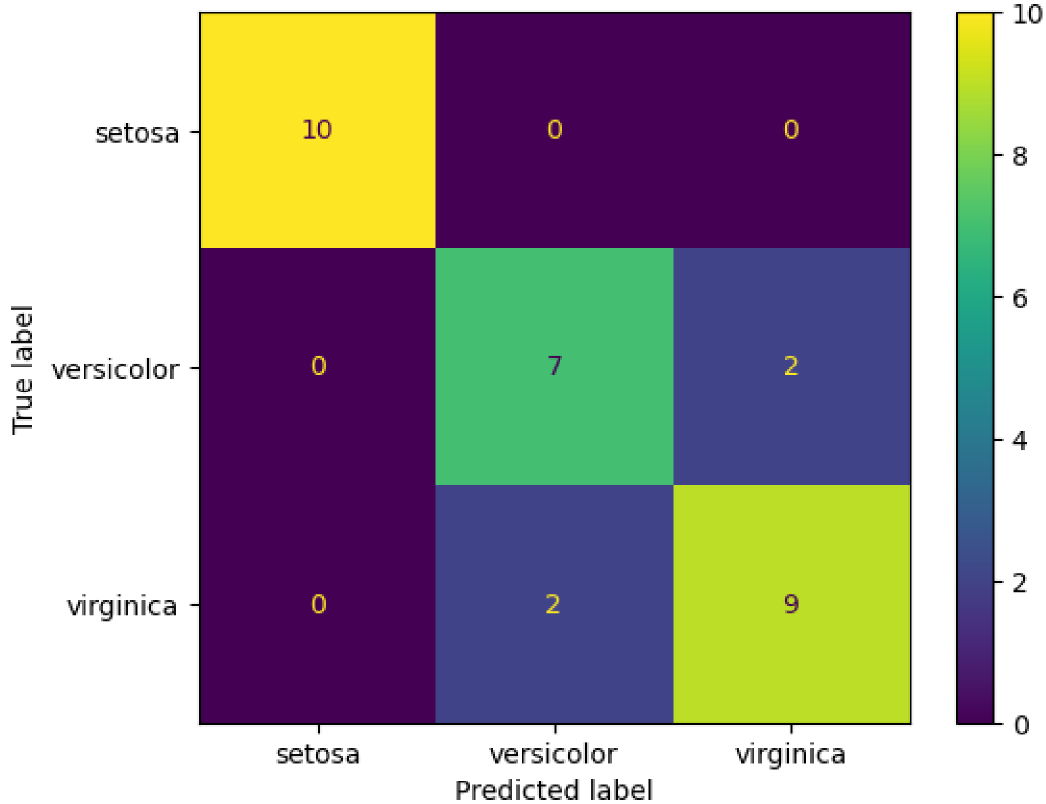

Figure 4: Confusion matrix

Discussion

We observe that the model predicts `Iris setosa` perfectly, while there are some misclassifications between `Iris versicolor` and `Iris virginica`. This is likely due to the fact that these two species have more similar feature values compared to `Iris setosa`, which is distinctly different in terms of petal length and width.

This model will be able to accurately predict the species of iris flowers based on their features with a high degree of accuracy. Further improvements could be made by tuning the hyperparameters of the decision tree or exploring other classification algorithms.

Future work could include testing this model on different flower species datasets to evaluate its generalizability and robustness. Future improvements could also involve exploring other models, such as Random Forests or logistic regression, to potentially enhance predictive performance.

References

1. UCI Machine Learning Repository: Iris Data Set. https://archive.ics.uci.edu/ml/datasets/iris
2. Milestone 1 of DSCI 522.
3. Scikit-learn documentation: https://scikit-learn.org/stable/
4. Seaborn documentation: https://seaborn.pydata.org/
5. DSCI 571 course materials.

Fisher, R. A. 1936. "Iris." UCI Machine Learning Repository.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. 2011. "Scikit-Learn: Machine Learning in Python." *Journal of Machine Learning Research* 12: 2825–30.

Waskom, Michael L. 2021. "Seaborn: Statistical Data Visualization." *Journal of Open Source Software* 6 (60): 3021. https://doi.org/10.21105/joss.03021.