ASSIGNMENT REPORT

DATA ANALYSIS WITH PYTHON

SYED ARSALAN HABIB

SEAT NO : B17101108

11-07-20

OVERVIEW

This assignment consisted of working with the iris dataset, and using a popular data science library 'Pandas' to load the dataset from a csv file and manipulate the data to obtain insightful statistics about the data and then using machine learning to build a classifier for predicting the classes.

CODE ARCHITECTURE

I divided the code into two scripts namely, 'iris-python-script-1' and 'iris-python-script-2'.

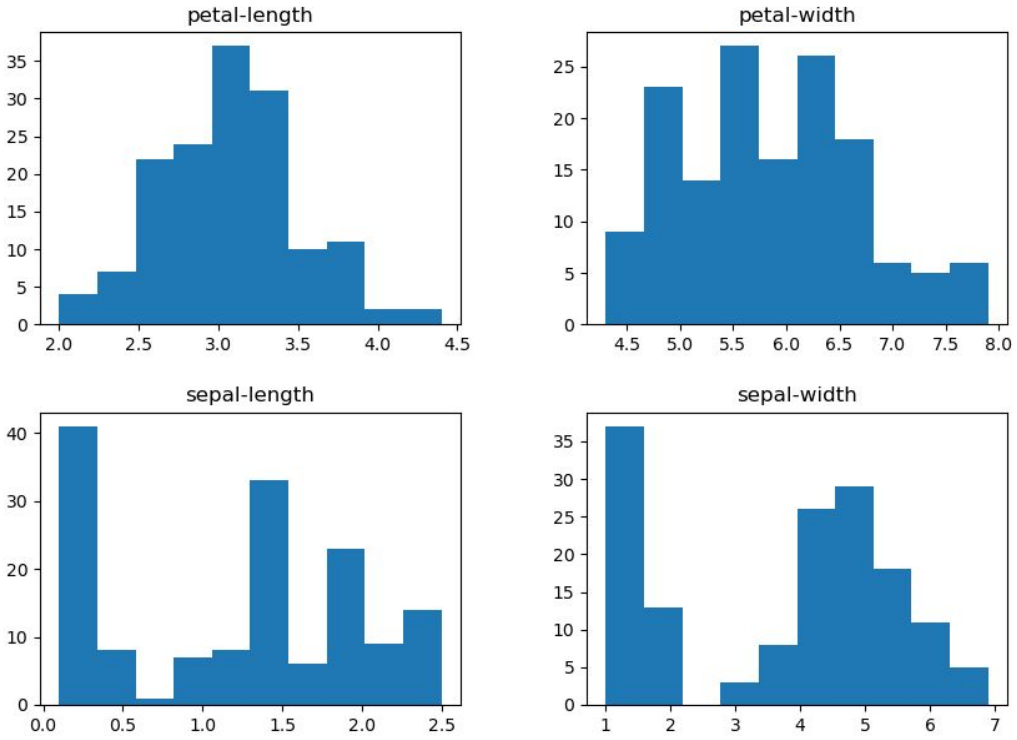
Starting with the first script,

- Firstly we import the required libraries 'Pandas' and 'numpy', then we load the csv file as a pandas dataframe object into the variable df with the read_csv method.
- Since the dataset did not have column headers, I added column names to the columns accordingly.
- AsI had to correct some of the values of the data and the indices of the incorrect values were given using 1-indexing, I changed the dataframe to use 1-indexing to avoid any confusion.
- I then checked the shape, data-types of the dataset.
- Added two new columns Petal.Ratio and Sepal.Ratio which were the ratio of their width to length values accordingly.

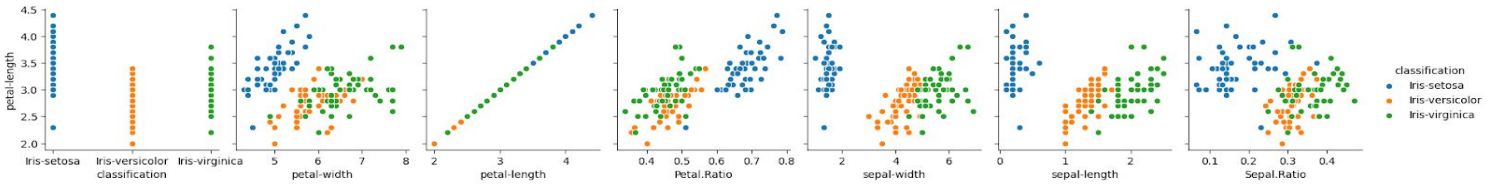
- Re-ordered the columns for better understanding and readability.
- In the last line, I saved the new dataframe as a csv file named iris_corrected.csv .

For the Second Script,

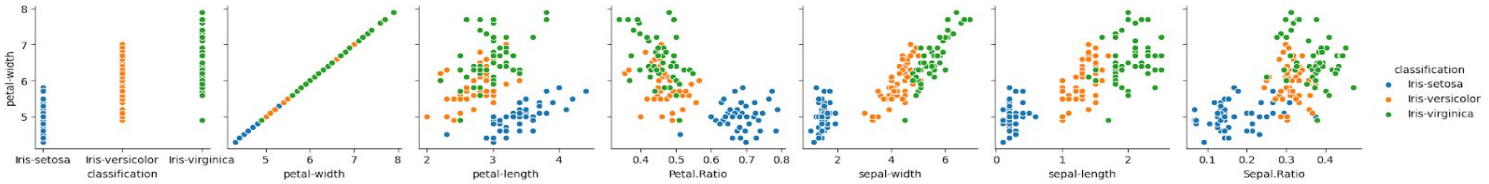
- Imported the previous two libraries with the addition of seaborn and sklearn. Seaborn is a data visualization library and sklearn is a machine learning library.
- Loaded the new iris_corrected.csv as a pandas dataframe.
- Grouped the data by classification and used pandas aggregate operations to obtain different statistics including mean, median and standard deviation.
- Visualized the data in different forms,
    - As histograms



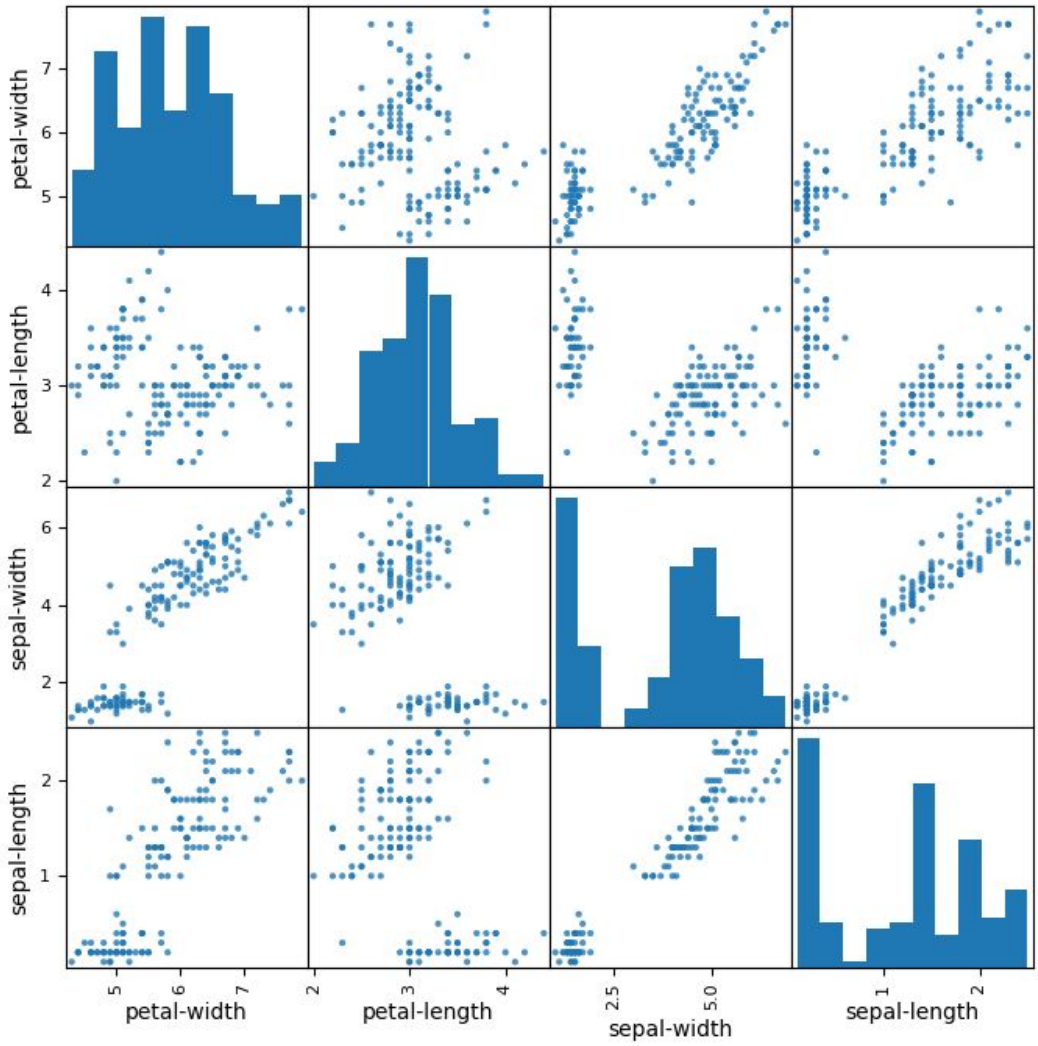
    - As scatter-plots between petal-width and all other features



    - As scatter-plots between petal-length and all other features



- As a scatter-matrix plotting all the possible combinations.



- The machine learning algorithm I chose was the decision tree algorithm.
- First, we separate the target variables(Classes of the flowers) from the other variables.
- Then we separate the data into training data (80%) and testing data (20%).
- We initialize the decision tree classifier and pass in the parameters
  - The criteria for making splits is the entropy value.
  - The max_depth=4 limits the tree from exceeding a depth of 4 levels
- The fit method then trains the model with the training data passed in as a parameter.
- We then run the classifier on the testing data and compare the model predicted answers with the actual answers and calculate the accuracy of the model.
- In the last line we use the classifier to predict the class of the given flower based on

the measurements provided.

RESULTS

- Some of the code output images are shared below:

	petal-width	petal-length	sepal-width	sepal-length	classification
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3.0	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5.0	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5.0	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa

```
print(df.head())
```

```
   petal-width  petal-length  sepal-width  sepal-length classification  \
1          5.1           3.5          1.4           0.2    Iris-setosa
2          4.9           3.0          1.4           0.2    Iris-setosa
3          4.7           3.2          1.3           0.2    Iris-setosa
4          4.6           3.1          1.5           0.2    Iris-setosa
5          5.0           3.6          1.4           0.2    Iris-setosa

   Petal.Ratio  Sepal.Ratio
1     0.686275     0.142857
2     0.612245     0.142857
3     0.680851     0.153846
4     0.673913     0.133333
5     0.720000     0.142857
```

	Petal.Ratio					Sepal.Ratio				
	mean	median	min	max	std	mean	median	min	max	std
classification										
Iris-setosa	0.684248	0.683502	0.511111	0.788462	0.051871	0.167868	0.142857	0.066667	0.375000	0.065789
Iris-versicolor	0.467680	0.462687	0.354839	0.566667	0.046829	0.311106	0.308608	0.243902	0.375000	0.029213
Iris-virginica	0.453396	0.460928	0.337662	0.548387	0.047015	0.366739	0.375000	0.250000	0.470588	0.050232

```
Accuracy of the classifier is 96.66666666666667

The given flower belongs to the class of ['Iris-setosa']
```

CHALLENGES AND DESIGN DECISIONS

There were no major challenges while working on the assignment as it was quite straightforward.
I used the seaborn library for visualizing the scatter plots as the library provides a good feature that allows us to group the data for better understanding.

REFERENCES

- https://youtu.be/vmEHCJofslg (A good video for introduction to Pandas)
- https://stackoverflow.com/questions/34091877/how-to-add-header-row-to-a-pandas-dataframe
- https://youtu.be/RmajweUFKvM (A very good video to help understand how the Decision tree algorithm works as well as how to implement it using the scikit-learn library)
- The Data Analysis presentation shared in the classroom.