CHAPTER-1

INTRODUCTION

1.1 Outline

The primary scope this project is an in-depth study into the Exploratory data analysis and the various statistical and plotting tools to represent and understand the data set before modeling and machine learning is done. Our main objective is to use the iris dataset and classify a given iris flower into one of the three categories (setosa, virginica, versicolor). This is shown by plotting graphs to extract the essential features required before starting further analysis.

1.2 Scope of Work

The project intends to critically revisit and access the following topics: Data and its types(Here an IRIS data set is taken and explained) , EDA (what and why is it done) , Analysis Process (Statistical measures: Measures of Central Tendency, Measures of Dispersion) and Graphical Representation (Scatter Plots and Pair Plots) to show which are the most useful features among sepal length, sepal width, petal length and petal width to identify various flower types.

CHAPTER-2

REVIEW OF LITERATURE

2.1 Description of Survey

EDA was originally developed by John Tukey, an American mathematician, in the 1970s. It's often thought of as more of a philosophical approach to data analysis than a statistical method. While performing exploratory data analysis, researchers begin to make sense of the data that they have access to so that they can figure out what questions to ask, how to frame these questions, and how to approach survey respondents so that they can uncover any insights that they feel might be missing.

Researchers and data analysts use EDA to understand and summarize the contents of a dataset, typically with a specific question in mind, or to prepare for more advanced statistical modeling in future stages of data analysis. EDA relies on data visualizations that enable researchers to identify and define patterns and characteristics in the dataset that they otherwise would not have known to look for.

EDA entails the examination of patterns, trends, outliers, and unexpected results in existing survey data, and using visual and quantitative methods to highlight the narrative that the data is telling.

Researchers that conduct exploratory data analysis are able to:

- Identify mistakes that have been made during data collection, and areas where data might be missing.
- Map out the underlying structure of the data.
- Identify the most influential variables in the dataset.
- List and highlight anomalies and outliers.
- Test previously proposed hypotheses.
- Establish a parsimonious model.
- Estimate parameters, determine confidence intervals, and define margins of error.

CHAPTER-3

ARCHITECTURE

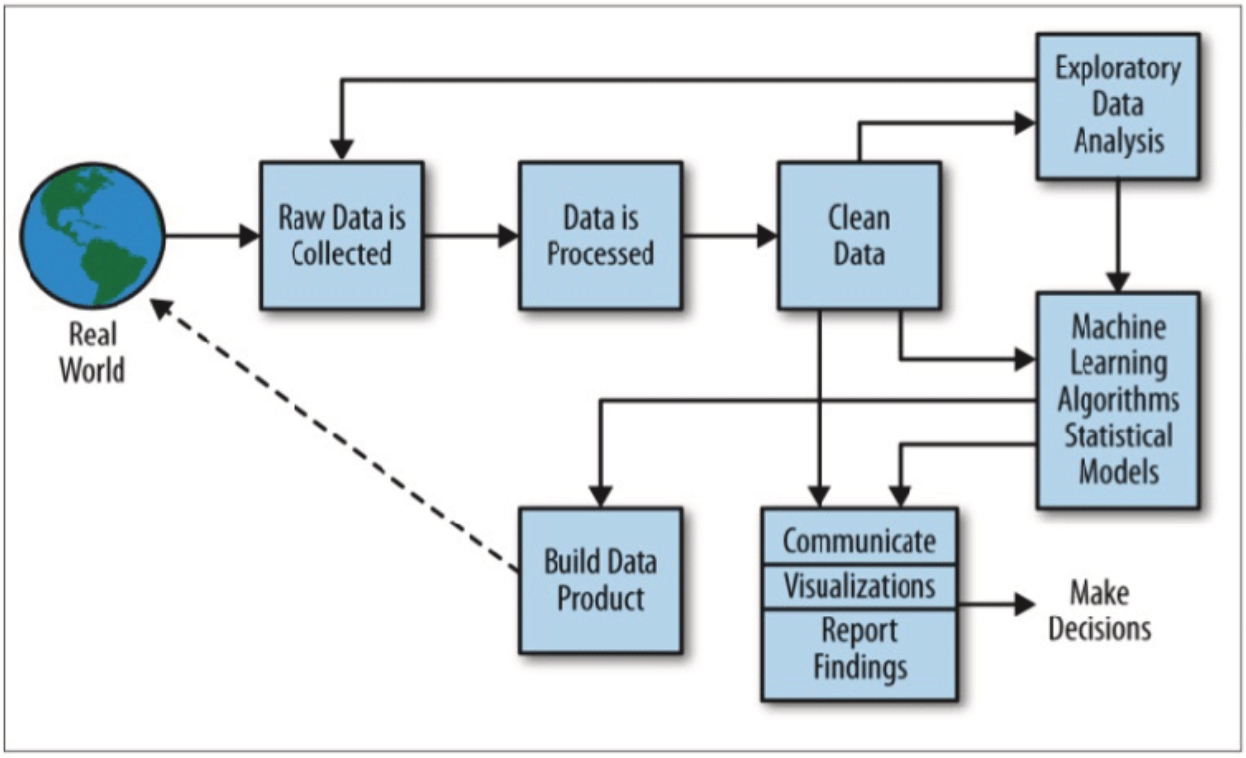

Fig.3 Architecture

This diagram shows the data science process.

- Data is collected from sensors in the environment, represented by the globe.
- Data is "cleaned" or otherwise processed to produce a data set (typically a data table) usable for processing.
- Exploratory data analysis and statistical modeling may then be performed.
- A "data product" is a program such as retailers use to suggest new purchases based on purchase history. It can also create data and feed it back into the environment.

CHAPTER-4

THEORETICAL IMPLEMENTATION

4.1 Measures of Central Tendency

A measure of central tendency is a single value that attempts to describe a set of data by identifying the central position within that set of data. As such, measures of central tendency are sometimes called measures of central location. They are also classed as summary statistics. The mean (often called the average) is most likely the measure of central tendency that you are most familiar with, but there are others, such as the median and the mode.

The mean, median and mode are all valid measures of central tendency, but under different conditions, some measures of central tendency become more appropriate to use than others. In the following sections, we will look at the mean, mode and median, and learn how to calculate them and under what conditions they are most appropriate to be used.

4.1.1 Mean

The mean (or average) is the most popular and well known measure of central tendency. It can be used with both discrete and continuous data, although its use is most often with continuous data (see our Types of Variable guide for data types). The mean is equal to the sum of all the values in the data set divided by the number of values in the data set. So, if we have n values in a data set and they have values x_1, x_2, ..., x_n, the sample mean, usually denoted by $\bar{x}$ (pronounced x bar), is:

$$\bar{x} = \frac{(x_1 + x_2 + \cdots + x_n)}{n}$$

This formula is usually written in a slightly different manner using the Greek capitol letter, $\sum$, pronounced "sigma", which means "sum of...":

$$\bar{x} = \frac{\sum x}{n}$$

4.1.2 Median

The median is the middle score for a set of data that has been arranged in order of magnitude. The median is less affected by outliers and skewed data. In order to calculate the median, suppose we have the data below:

65	55	89	56	35	14	56	55	87	45	92

We first need to rearrange that data into order of magnitude (smallest first):

14	35	45	55	55	**56**	56	65	87	89	92

Our median mark is the middle mark - in this case, 56 (highlighted in bold). It is the middle mark because there are 5 scores before it and 5 scores after it. This works fine when you have an odd number of scores, but what happens when you have an even number of scores? What if you had only 10 scores? Well, you simply have to take the middle two scores and average the result. So, if we look at the example below:

65	55	89	56	35	14	56	55	87	45

We again rearrange that data into order of magnitude (smallest first):

14	35	45	55	**55**	**56**	56	65	87	89

Only now we have to take the 5th and 6th score in our data set and average them to get a median of 55.5.

4.1.3 Mode

The mode is the most frequent score in our data set. On a histogram it represents the highest bar in a bar chart or histogram. You can, therefore, sometimes consider the mode as being the most popular option.

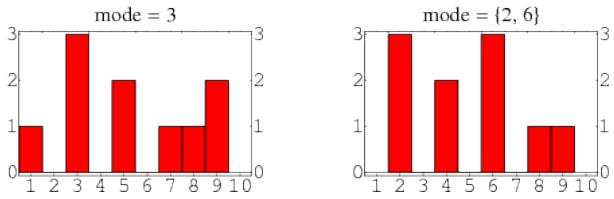

Fig.4.1.3 Mode

4.2 Measures of Dispersion

Measure of dispersion shows the scatterings of the data. It tells the variation of the data from one another and gives a clear idea about the distribution of the data. The measure of dispersion shows the homogeneity or the heterogeneity of the distribution of the observations. The main idea about the measure of dispersion is to get to know how the data are spread. It shows how much the data vary from their average value.

Characteristics of Measures of Dispersion

- A measure of dispersion should be rigidly defined
- It must be easy to calculate and understand
- Not affected much by the fluctuations of observations
- Based on all observations

Classification of Measures of Dispersion

The measure of dispersion is categorized as:

(i) An absolute measure of dispersion:

The measures which express the scattering of observation in terms of distances i.e., range, quartile deviation.The measure which expresses the variations in terms of the average of deviations of observations like mean deviation and standard deviation.

(ii) A relative measure of dispersion:

We use a relative measure of dispersion for comparing distributions of two or more data set and for unit free comparison. They are the coefficient of range, the coefficient of mean deviation, the coefficient of quartile deviation, the coefficient of variation, and the coefficient of standard deviation.

4.2.1 Range

A range is the most common and easily understandable measure of dispersion. It is the difference between two extreme observations of the data set. If X_{max} and X_{min} are the two extreme observations then

$$\text{Range} = X_{max} - X_{min}$$

4.2.2 Quartile Deviation

The quartiles divide a data set into quarters. The first quartile, (Q_1) is the middle number between the smallest number and the median of the data. The second quartile, (Q_2) is the median of the data set. The third quartile, (Q_3) is the middle number between the median and the largest number.

Quartile deviation or semi-inter-quartile deviation is

$$Q = ½ \times (Q_3 - Q1)$$

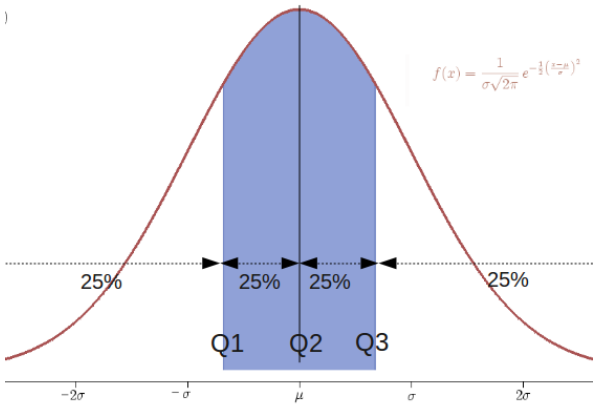

Fig.4.2.2 Quartile Deviation

4.2.3 Standard Deviation

A standard deviation is the positive square root of the arithmetic mean of the squares of the deviations of the given values from their arithmetic mean. It is denoted by a Greek letter sigma, σ. It is also referred to as root mean square deviation. The standard deviation is given as

$$\sigma = [(\Sigma_i (y_i - \bar{y})/n]^{½} = [(\Sigma_i y_i^2/n) - \bar{y}^2]^{½}$$

For a grouped frequency distribution, it is $\sigma = [(\Sigma_i f_i (y_i - \bar{y})/N]^{½} = [(\Sigma_i f_i y_i^2/n) - \bar{y}^2]^{½}$

The square of the standard deviation is the **variance**. It is also a measure of dispersion.

$$\sigma^2 = [(\Sigma_i (y_i - \bar{y}) / n]^{½} = [(\Sigma_i y_i^2/n) - \bar{y}^2]$$

4.3 Graphical Representation

Portraying data graphically certainly contributes toward a clearer and more penetrative understanding of data and also makes sophisticated statistical data analyses more marketable. This realization has emerged from many years of experience in teaching students, in research, and especially from engaging in statistical consulting work in a variety of subject fields. Consequently, we were somewhat surprised to discover that a comprehensive, yet simple presentation of graphical exploratory techniques for the data analyst was not available.

4.3.1 1-D Plot and 2-D Plot

1-D plot is a plot where we plot the graph with sepal length/sepal width/petal length/petal width of all the three flowers on X-axis, but there is no clarity as everything overlaps on itself. Plot(X,Y) creates a 2-D line plot of the data in Y versus the corresponding values in X.

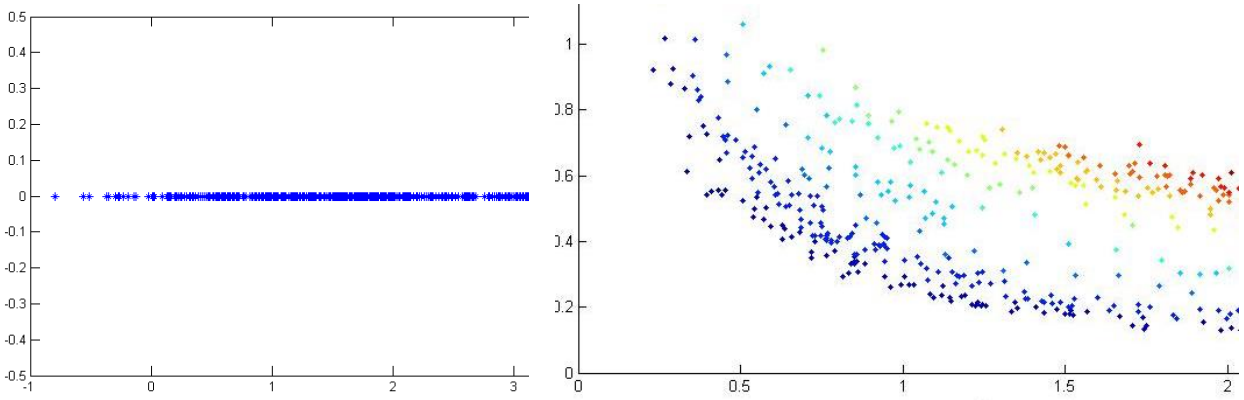

Fig.4.3.1 1-D Plot & 2-D Plot

4.3.2 Pair Plot

Plot pairwise relationships in a dataset.By default, this function will create a grid of Axes such that each variable in data will by shared in the y-axis across a single row and in the x-axis across a single column. The diagonal Axes are treated differently, drawing a plot to show the univariate distribution of the data for the variable in that column.It is also possible to show a subset of variables or plot different variables on the rows and columns.This is a high-level interface for **PairGrid** that is intended to make it easy to draw a few common styles. You should use **PairGrid** directly if you need more flexibility.

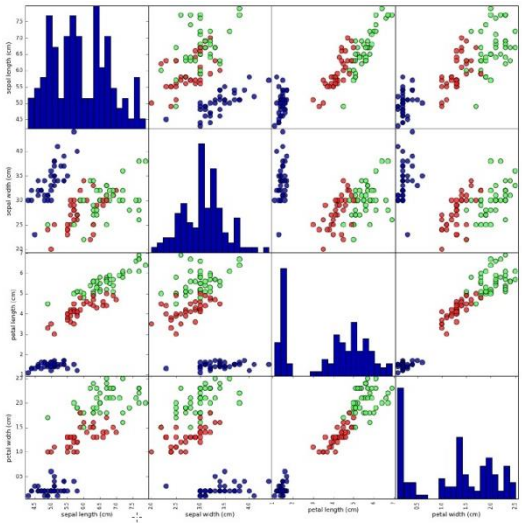

Fig.4.3.2 Pair plot

4.3.3 Box Plot

A box plot (or box-and-whisker plot) shows the distribution of quantitative data in a way that facilitates comparisons between variables or across levels of a categorical variable. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution, except for points that are determined to be “outliers” using a method that is a function of the inter-quartile range.

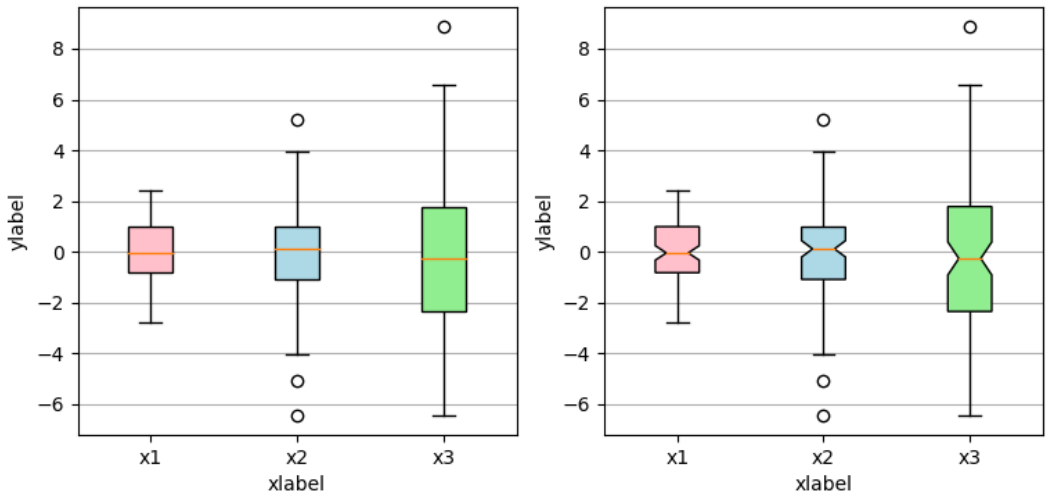

Fig.4.3.3 Box plot

4.3.4 Violin Plot

A violin plot is a method of plotting numeric data. It is similar to a box plot, with the addition of a rotated kernel density plot on each side. Violin plots are similar to box plots, except that they also show the probability density of the data at different values, usually smoothed by a kernel density estimator

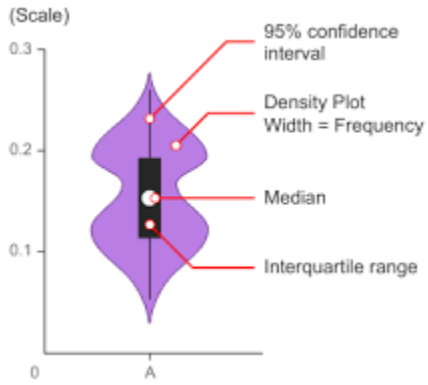

Fig.4.3.4 Violin plot

CHAPTER-5

PRACTICAL IMPLEMENTATION

5.1 Dataset (Case Study)

The **Iris flower data set** or **Fisher's Iris data set** is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper The use of multiple measurements in taxonomic problems as an example of analysis. It is sometimes called **Anderson's Iris data set** because Edgar Anderson collected the data to quantify the morphologic variation of Iris flowers of three related species. Two of the three species were collected in the Gaspé Peninsula "all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus".

The data set consists of 50 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris versicolor). Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters. Based on the combination of these four features, Fisher developed a linear discriminant model to distinguish the species from each other.

Fig.5.1 Iris Flowers

The dataset contains a set of 150 records under five attributes - petal length, petal width, sepal length, sepal width and species.

5.1.1 Dataset Table

Dataset Order ▲	Sepal length ◆	Sepal width ◆	Petal length ◆	Petal width ◆	Species ◆
1	5.1	3.5	1.4	0.2	*I. setosa*
2	4.9	3.0	1.4	0.2	*I. setosa*
3	4.7	3.2	1.3	0.2	*I. setosa*
4	4.6	3.1	1.5	0.2	*I. setosa*
5	5.0	3.6	1.4	0.3	*I. setosa*
51	7.0	3.2	4.7	1.4	*I. versicolor*
52	6.4	3.2	4.5	1.5	*I. versicolor*
53	6.9	3.1	4.9	1.5	*I. versicolor*
54	5.5	2.3	4.0	1.3	*I. versicolor*
55	6.5	2.8	4.6	1.5	*I. versicolor*
56	5.7	2.8	4.5	1.3	*I. versicolor*
101	6.3	3.3	6.0	2.5	*I. virginica*
102	5.8	2.7	5.1	1.9	*I. virginica*
103	7.1	3.0	5.9	2.1	*I. virginica*
104	6.3	2.9	5.6	1.8	*I. virginica*
105	6.5	3.0	5.8	2.2	*I. virginica*
106	7.6	3.0	6.6	2.1	*I. virginica*

Fig.5.1.1 Dataset Table

5.2 Implementation of Analysis Process

Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	*I. setosa*
4.9	3.0	1.4	0.2	*I. setosa*
4.7	3.2	1.3	0.2	*I. setosa*
4.6	3.1	1.5	0.2	*I. setosa*
5.0	3.6	1.4	0.3	*I. setosa*
5.4	3.9	1.7	0.4	*I. setosa*

Data-point/vector/Observation
Feature/Variable/Input-variable/Independent-variable
Label/Dependent-variable/Output-varible/Class/Class-label/Response label

Fig.5.2.1 Schema of the Dataset

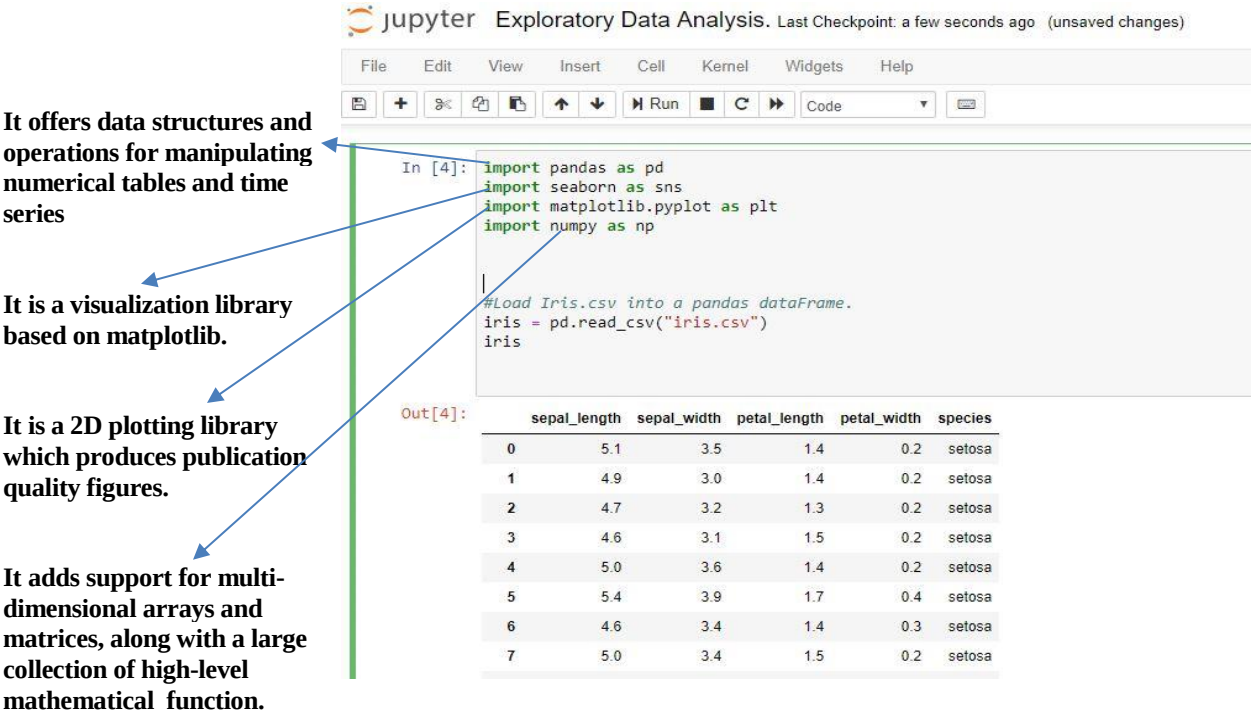

Fig.5.2.2 Analysis Process

Q) how many data-points and features?

```
In [2]: print (iris.shape)

        (150, 5)
```

Q) What are the column names in our dataset?

```
In [3]: print (iris.columns)

Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',
       'species'],
      dtype='object')
```

Q) How many flowers for each species are present?

```
iris["species"].value_counts()

Out[5]: virginica     50
        versicolor    50
        setosa        50
        Name: species, dtype: int64
```

Iris is a balanced dataset as the number of data points for every class is 50

Fig.5.2.3 Measures of Central Tendency

```
In [12]: import numpy as np
         print("Means:")
         print(np.mean(iris_setosa["petal_length"]))
         #Mean with an outlier.
         print(np.mean(np.append(iris_setosa["petal_length"],50)));
         print(np.mean(iris_virginica["petal_length"]))
         print(np.mean(iris_versicolor["petal_length"]))
```

```
Means:
1.464
2.4156862745098038
5.552
4.26
```

Mean [1,2,3,4,5] = 3

without outlier [1,2,3,4,50] = 12

```
print("\nMedians:")
print(np.median(iris_setosa["petal_length"]))
#Median with an outlier
print(np.median(np.append(iris_setosa["petal_length"],50)));
print(np.median(iris_virginica["petal_length"]))
print(np.median(iris_versicolor["petal_length"]))
```

```
Medians:
1.5
1.5
5.55
4.35
```

Median [1,2,3,4,5] = 3

without outlier [1,2,3,4,50] = 3

Fig.5.2.4 Mean and Median

```
print("\nQuantiles:")
print(np.percentile(iris_setosa["petal_length"],np.arange(0, 100, 25)))
print(np.percentile(iris_virginica["petal_length"],np.arange(0, 100, 25)))
print(np.percentile(iris_versicolor["petal_length"], np.arange(0, 100, 25)))

print("\n90th Percentiles:")
print(np.percentile(iris_setosa["petal_length"],90))
print(np.percentile(iris_virginica["petal_length"],90))
print(np.percentile(iris_versicolor["petal_length"], 90))

from statsmodels import robust
print ("\nMedian Absolute Deviation")
print(robust.mad(iris_setosa["petal_length"]))
print(robust.mad(iris_virginica["petal_length"]))
print(robust.mad(iris_versicolor["petal_length"]))
```

```
Quantiles:
[ 1.     1.4    1.5    1.575]
[ 4.5    5.1    5.55   5.875]
[ 3.    4.    4.35  4.6 ]

90th Percentiles:
1.7
6.31
4.8

Median Absolute Deviation
0.148260221851
0.667170998328
0.518910776477
```

```
print("\nStd-dev:");
print(np.std(iris_setosa["petal_length"]))
print(np.std(iris_virginica["petal_length"]))
print(np.std(iris_versicolor["petal_length"]))
```

```
Std-dev:
0.17176728442867115
0.5463478745268441
0.4651881339845204
```

Fig.5.2.5 Measures of Dispersion

1-D scatter plot 2-D scatter plot 3-D scatter plot

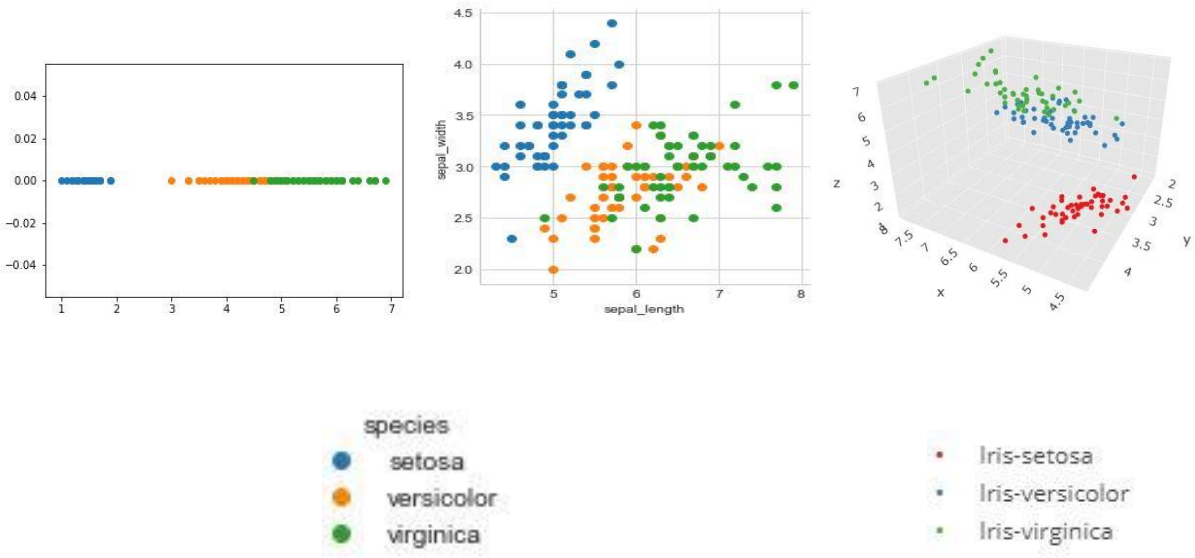

Fig.5.2.6 Graphical Representation (1-d,2-d,3-d)

Pair Plots

If we have more than 3 dimensions or features in our dataset we do not have the capability to visualize. One solution to this problem is pair plots. As the name suggests we actually do pairs of features and plot them all.

For example, let's say we have four features Name, Place, Animal and Thing in our dataset. In that case, we will have 4C2 plots i.e. 6 unique plots. The pairs in this case will be i. (Name, Place); ii. (Name, Animal); iii. (Name, Thing); iv. (Place, Animal); v. (Place, Thing) and vi. (Animal, Thing).

Fig.5.2.7 Pair Plots

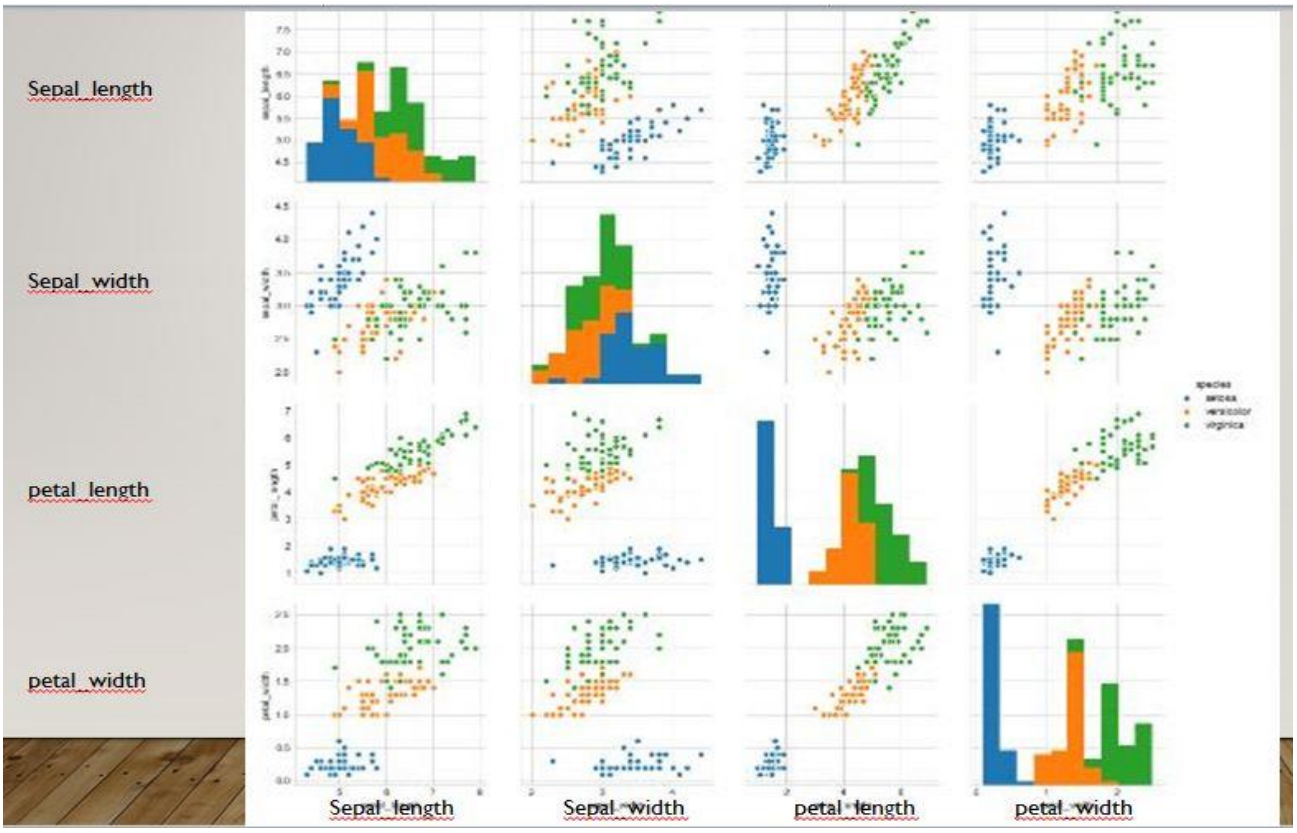

Fig.5.2.8 Pair plot representation

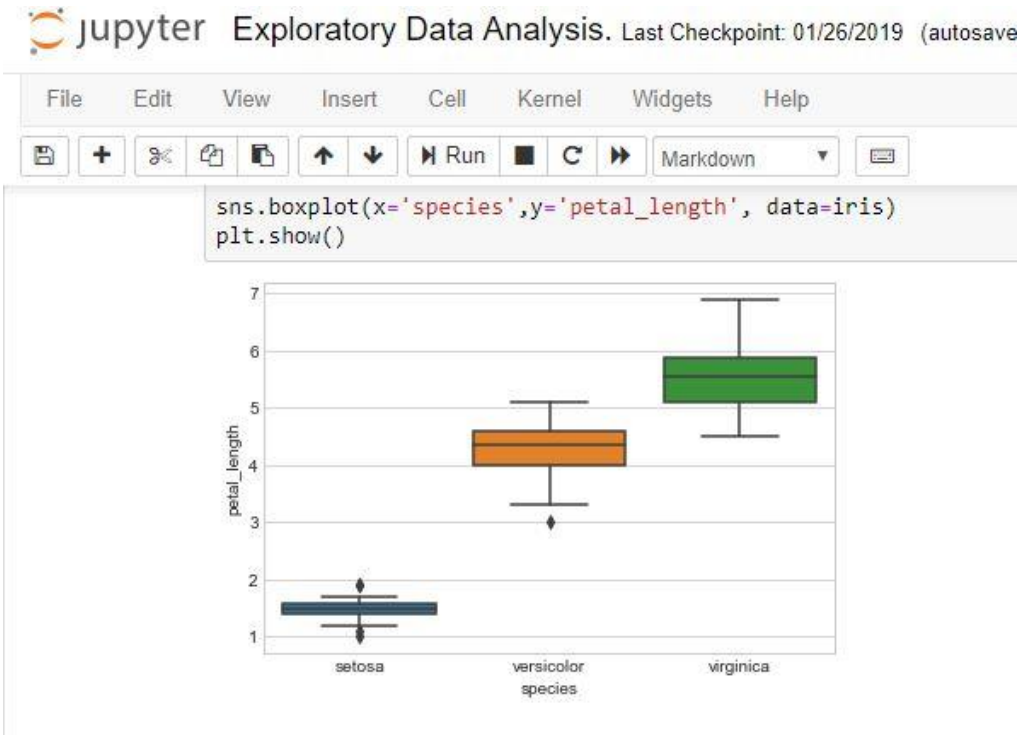

Fig.5.2.9 Box Plot

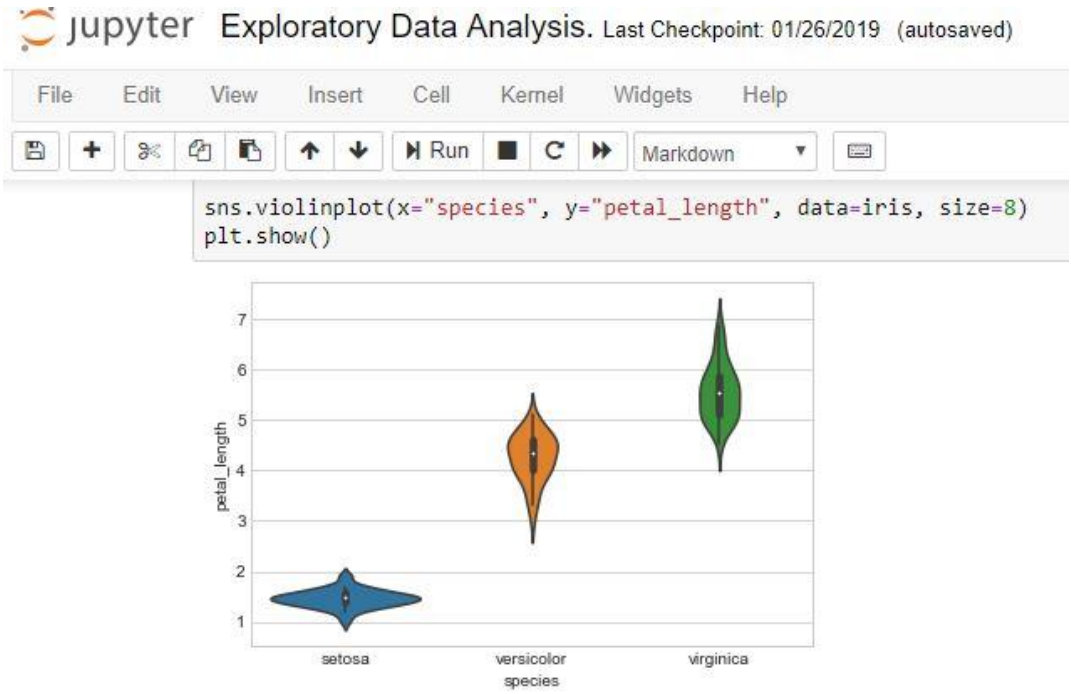

Fig.5.2.10 Violin Plot

5.3 Tools

5.3.1 Anaconda

Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing, that aims to simplify package management and deployment. Package versions are managed by the package management system conda.

5.3.2 Jupyter Notebook (IDE)

The Jupyter Notebook is an incredibly powerful tool for interactively developing and presenting data science projects. A notebook integrates code and its output into a single document that combines visualizations, narrative text, mathematical equations, and other rich media. The intuitive workflow promotes iterative and rapid development, making notebooks an increasingly popular choice at the heart of contemporary data science, analysis, and increasingly science at large.

5.4 Technologies

5.4.1 Python

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales.

CHAPTER-6

CONCLUSION & FUTURE ENHANCEMENTS

6.1 Conclusion

From Fig. 5.2.8 we can infer that

- Petal_length and Petal_width are the most useful features to identify various flower types.
- While Setosa can be easily identified (linearly separable), Virginica and Versicolor have some overlap (almost linearly separable).
- We can find "lines" and "if-else" conditions to build a simple model to classify the flower types.

6.2 Future Enhancements

Similarly this analysis process can be applied to any other dataset. Eg. If we have a customer purchasing dataset ,we can analyze the purchasing history and recommended purchases that customer might enjoy, in a Wine Dataset where we check the pH value of the wines and come to a conclusion of how much acidity is present in the wine etc. Feature extraction can then be done according to our requirements.

CHAPTER-7

REFERENCES

- https://en.wikipedia.org/wiki/Iris_flower_data_set
- https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15
- https://apps.dtic.mil/dtic/tr/fulltext/u2/a266775.pdf
- https://en.wikipedia.org/wiki/Exploratory_data_analysis
- https://www.anaconda.com/distribution/Python 3.7 version
- https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html
- https://seaborn.pydata.org/
- https://pandas.pydata.org/pandas-docs/stable/
- https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.html
- https://matplotlib.org/tutorials/index.html
- https://www.kaggle.com/lalitharajesh/iris-dataset-exploratory-data-analysis
- https://scikit-learn.org/