Student: Luminita Done

*Student ID: G00411226*

*Lecturer: Andrew Beatty*

Pands PROJECT

**Fisher’s Iris data set**

Contents

[Summary 2](#_Toc101192812)

[Imports 2](#_Toc101192813)

[Libraries 2](#_Toc101192814)

[Pandas 2](#_Toc101192815)

[Numpy 2](#_Toc101192816)

[Matplotlib 2](#_Toc101192817)

[Seaborn 2](#_Toc101192818)

[Plots 3](#_Toc101192819)

[How do we visualise data? 3](#_Toc101192820)

[References 3](#_Toc101192821)

# Summary

R.A. Fisher published the first report on the iris data set in the Annals of Human Genetics in 1936. It is a collection of 50 samples collected by the author on each of three Irises species: setosa, versicolor, and virginica. Four attributes of 50 flowers from each of the plants were measured: sepal length, sepal width, petal length, and petal width. According to the author, the lengths and widths of the petal and sepal are features that can be utilized to determine which species they belong to using a linear discriminant model. The linear discriminant model, a statistical, machine learning, and pattern recognition technique used to distinguish between two or more objects, classes, or events, was created by Fischer himself. (Wikipedia, n.d.)

Fischer recorded the findings for the three species in a table containing each of the four measurements, followed by tables of observed means, sums of squares, and other statistics to show how each species could be distinguished from the others. Fischer uses sums of squares and products of deviations from each mean to produce a linear function that best distinguishes between the two species. The ANOVA test is a strong statistical method for determining correlations (such as differences) between variables by comparing the means of the variables. (statisticssolutions.com, n.d.)

# Imports

# Libraries

## Pandas

## Numpy

A multi-dimensional array and matrix data structures are included in NumPy. It can execute a variety of mathematical operations on arrays, including trigonometric, statistical, and algebraic algorithms. As a result, there are a lot of mathematical, algebraic, and transformation functions in the library. (medium.com, n.d.)

## Matplotlib

## Seaborn

Based on matplotlib, Seaborn is a Python data visualization package. It provides a high-level interface for creating visually appealing and useful statistics visuals.

You can read the introductory notes or the paper to get a quick overview of the library's concepts. To learn how to install the package and get started with it, go to the installation page. You can look through the example gallery to see what you can achieve with seaborn, and then read the tutorial or API reference to learn how. (seaborn.pydata, n.d.)

Seaborn is a Matplotlib-based Python data visualization package.

The following syntax is the most frequent way to import Seaborn into your Python environment:

import seaborn as sns

Python is told to introduce the Seaborn library into your present environment via the import seaborn portion of the code.

The as sns section of the code instructs Python to assign Seaborn the alias sns. You can now use Seaborn functions by using sns.function name instead of seaborn.function name.

After Seaborn is imported, we may utilize the built-in methods to visualize data rapidly. (statology.org, n.d.)

#### FaceGrid

Using numerous panels, the FacetGrid class allows you to see the distribution of one variable as well as the relationship between multiple variables within subsets of your collection.

A FacetGrid can have up to three dimensions, including row, col, and hue. Consider the hue variable as a third dimension along a depth axis, with different levels represented with distinct colours.

The FacetGrid object accepts a data frame as well as the names of the variables that will make up the grid's row, column, and hue dimensions. The variables should be categorical, and data from each level of the variable will be used to create a facet along that axis. (tutorialspoint, n.d.)

# Plots

# Below we have a series of ways to create a plot with violin plot and boxplot for the other type of analysis I will create

## Violin Plot

#Violin plot example, we plot violin plot of our iris data:

#sns.violinplot(x="type",y="petal\_length",data=iris)

#plt.show()

## Box Plot

#BoxPlot plot display for petal lenght:

#sns.boxplot(x="type",y="petal\_length",data=iris)

#plt.show()

#BoxPlot plot display for petal width:

#sns.boxplot(x="type",y="petal\_width",data=iris)

#plt.show()

#BoxPlot plot display for petal lenght:

#sns.boxplot(x="type",y="sepal\_lenght",data=iris)

#plt.show()

#BoxPlot plot display for petal lenght:

#sns.boxplot(x="type",y="sepal\_width",data=iris)

#plt.show()

### How do we visualise data?

# Scatter Plot

Images display

%pylab inline

import matplotlib.pyplot as plt

import matplotlib.image as mpimg

img = mpimg.imread('your\_image.png')

imgplot = plt.imshow(img)

plt.show()

# References

medium.com, n.d. *medium.com.* [Online]   
Available at: https://medium.com/fintechexplained/why-should-we-use-numpy-c14a4fb03ee9  
[Accessed 19 04 2022].

seaborn.pydata, n.d. *seaborn.pydata.* [Online]   
Available at: https://seaborn.pydata.org/  
[Accessed 18 04 2022].

statisticssolutions.com, n.d. *statisticssolutions.com.* [Online]   
Available at: https://www.statisticssolutions.com/anova-analysis-of-variance/  
[Accessed 18 04 2022].

statology.org, n.d. *statology.* [Online]   
Available at: https://www.statology.org/import-seaborn-as-sns/  
[Accessed 19 04 2022].

tutorialspoint, n.d. *tutorialspoint.* [Online]   
Available at: https://www.tutorialspoint.com/seaborn/seaborn\_facet\_grid.htm  
[Accessed 18 04 2022].

Wikipedia, n.d. *Wikipedia.* [Online]   
Available at: https://en.wikipedia.org/wiki/Linear\_discriminant\_analysis  
[Accessed 18 04 2022].