LP1 Assignment DA R1

Summary statistics, data visualization and boxplot for the features on the Iris dataset

Date - 9th August, 2020.

Assignment Number - DA R1

Title

Summary statistics, data visualization and boxplot for the features on the Iris dataset

Problem Definition

Download the Iris flower dataset or any other dataset into a DataFrame. (eg https://archive. ics.uci.edu/ml/datasets/Iris) Use Python and Perform following:

- How many features are there and what are their types (e.g., numeric, nominal)?
- Compute and display summary statistics for each feature available in the dataset. (eg. minimum value, maximum value, mean, range, standard deviation, variance and percentiles
- Data Visualization-Create a histogram for each feature in the dataset to illustrate the feature distributions. Plot each histogram.
- Create a boxplot for each feature in the dataset. All of the boxplots should be combined into a single plot. Compare distributions and identify outliers.

Learning Objectives

- Learn to use dataset, dataframes, features of dataset in an application
- Learn to compute summary statistics for the features.
- Learn to use visualization techniques.

Learning Outcomes

I will be able to compute statistics on the features of the dataset, use histograms and boxplot on the features of the dataset.

Software Packages and Hardware Apparatus Used

- Operating System : 64-bit Ubuntu 18.04
- Programming Language : Python 3
- Jupyter Notebook Environment : Google Colaboratory
- Python Libraries : Numpy, Pandas, MatPlotLib

Related Mathematics

Mathematical Model

Let S be the system set:

S = {s; e;X; Y; Fme; Ff;DD;NDD; Fc; Sc}

where Dataset is loaded into the dataframe

s=start state

Iris Dataset

e=end state

Summary statistics for each feature is computed.

X=set of inputs

X = {X1}

Where X1 = IRIS Dataset

- 5 Features
  - 4 Numerical Feature
  - 1 Nominal Feature
- Data Count - 150

Y=set of outputs

1) Number of features
2) Types of features
3) Minimum value for each feature in the dataset
4) Maximum value for each feature in the dataset
5) Mean for each feature in the dataset
6) Range for each feature in the dataset
7) Standard deviation for each feature in the dataset
8) Variance for each feature in the dataset
9) Percentiles for each feature in the dataset

10) Histogram for each feature in the dataset
11) Boxplot for each feature in the dataset

Fme is the main function

It calls friend functions

Ff is the set of friend functions

Ff = {f1,f2,f3,f4,f5,f6}

where

f1 = function to load dataset into dataframe

f2 = function to to get number of features

f3 = function to get feature type

f4 = function to get minimum,maximum,mean,range,standard deviation,variance and percentile for each feature

f5 = function to draw histogram for each feature

f6 = function to draw boxplot for each feature

DD= Deterministic Data

IRIS dataset

- 5 Features
  - 4 Numerical Feature
  - 1 Nominal Feature
- Data Count - 150

NDD=Non-deterministic data

No non deterministic data

Fc =failure case:

No failure case identified for this application

Concepts related Theory

Data analysis is a process of inspecting, cleansing, transforming, and modelling data with the goal of discovering useful information, informing conclusions, and supporting decision-making.Data analysis has multiple facets and approaches, encompassing diverse techniques under a variety of names, while being used in different business, science, and social science domains.

A data set (or dataset) is a collection of data. Most commonly a data set corresponds to the contents of a single database table, or a single statistical data matrix, where every column of the table represents a particular variable, and each row corresponds to a given member of the data set in question.

Mean, standard deviation, regression, sample size determination and hypothesis testing are the fundamental data analytics methods.

Mean

The sum of all the data entries divided by the number of entries.

$$\text{Population Mean: } \mu = \frac{\Sigma x}{N}$$

$$\text{Sample Mean: } \overline{x} = \frac{\Sigma x}{n}$$

Range

The difference between the maximum and minimum data entries in the set.

Range = (Max. data entry) – (Min. data entry)

Standard Deviation

The standard deviation measures variability and consistency of the sample or population. In most real-world applications, consistency is a great advantage.

$$\text{Population Standard Deviation} = \sigma = \sqrt{\frac{\Sigma(x-\mu)^2}{N}}$$

$$\text{Sample Standard Deviation } = s = \sqrt{\frac{\Sigma(x-\overline{x})^2}{n-1}}$$

Variance

Variance is the average squared deviation from the mean.

Percentile

Let p be any integer between 0 and 100. The pth percentile of the data set is the data value at which p percent of the value in the data set is less than or equal to this value.

Steps for Execution

1. Download Iris Dataset
2. Open Google Colaboratory
3. Upload Iris Dataset to Google Colaboratory
4. Import Python Packages like Numpy, Pandas, MatPlotLib

5. Get features from the Dataset into Pandas Dataframe
6. Give Feature Names to Pandas Dataframe
7. Get Feature Count and Type of Feature
8. Compute Statistics in the Problem Definition
9. Generate Histograms and Boxplots for features of the dataset

Useful Python Functions

Pandas Functions

read_csv

Read Data from Iris Dataset downloaded and uploaded to Google Colab

shape

Get (number of samples,number of features)

iteritems

Iterate through features/columns of the dataset

describe

Get Useful Statistics on the dataset like mean, max value, min value, standard deviation and percentiles

var

Get Variance of features of the data set

hist

Plot Histogram of features of the dataset

max

Get the max value from all the features

min

Get the min value from all the features

plot

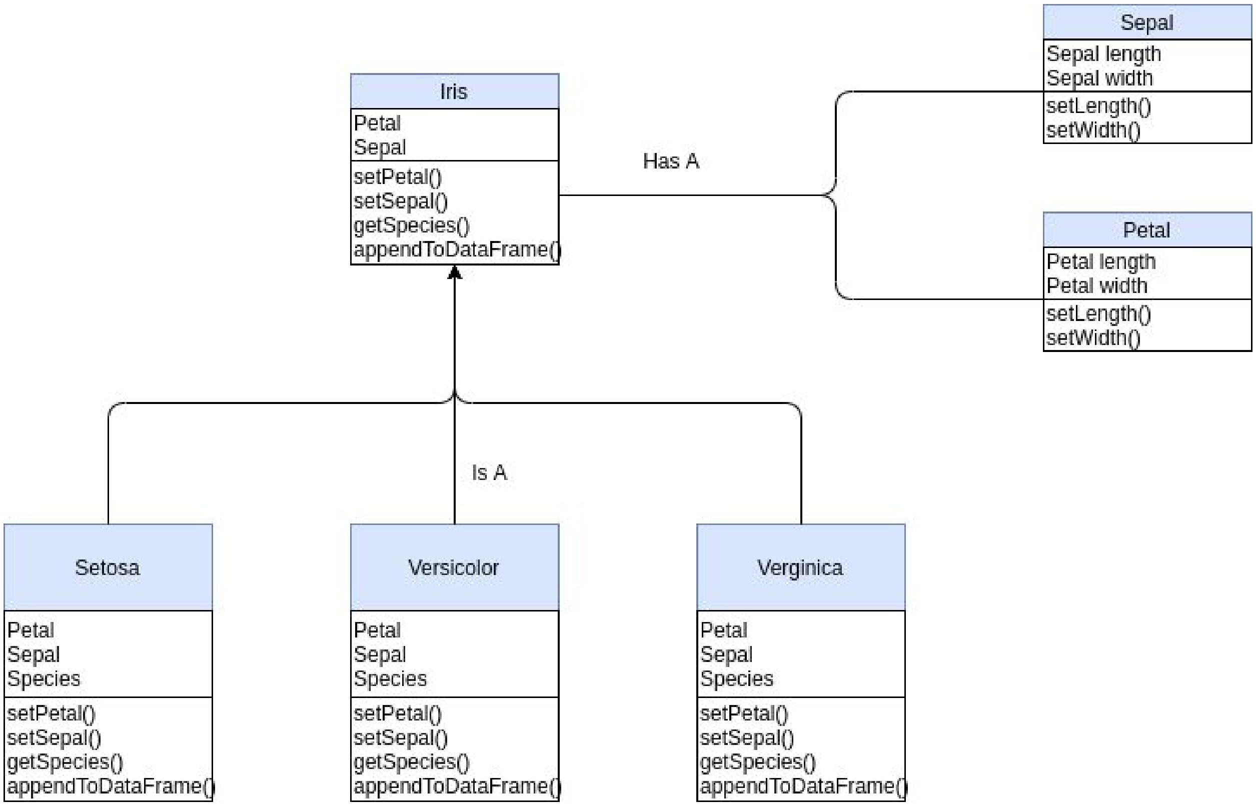
Plot types of graph including box plot

Matplot Library Functions

show

Show the plotted graph using Matplot Library

Flowchart Design



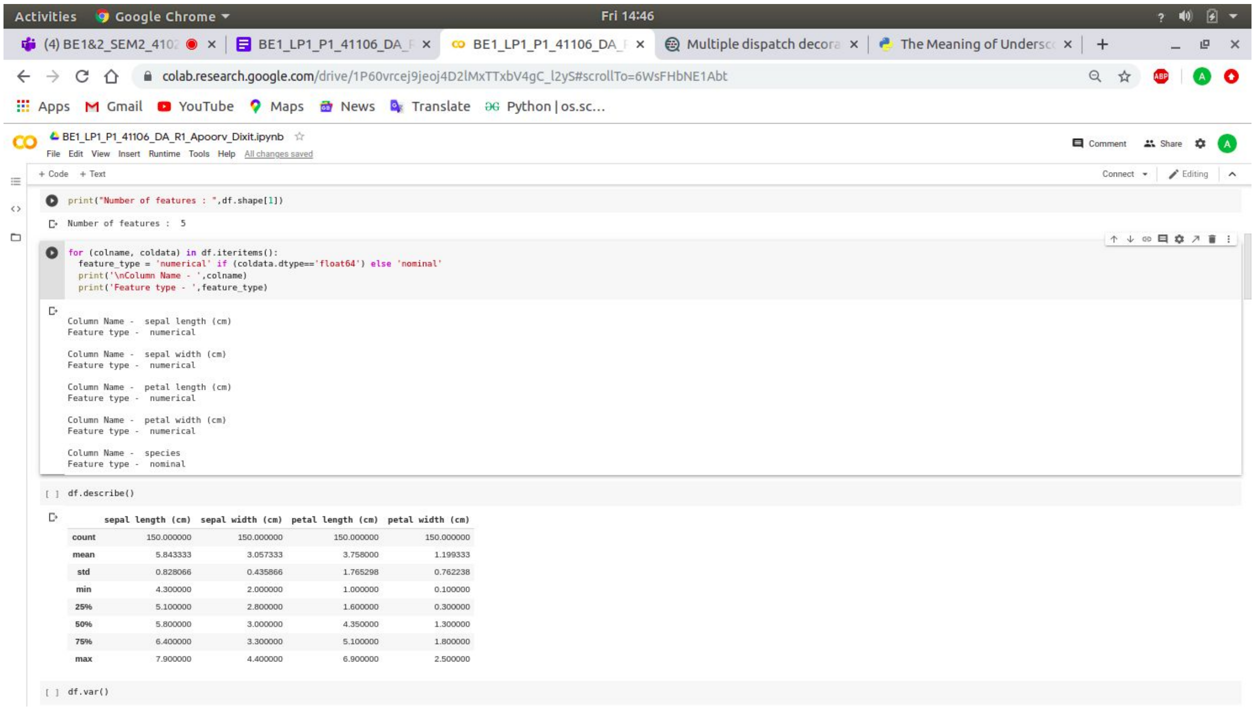
Footnotes

- As per the Python naming conventions, Use one leading underscore only for non-public methods, instance variables and Classes. In this assignment, class _Petal and _Sepal are treated as Non Public Classes. Private instance variables include _species of Classes Setosa, Virginica and Versicolor; and _length and _width of classes _Petal and _Sepal.

- Python does not support Method Overloading by default, To emulate method Overloading in Python, Programmer can do the following things (https://codippa.com/how-to-perform-method-overloading-in-python/)
    - Use Default Parameters
    - Modify program according to different number of parameters
- Python does not have the concept of Private Instance of Variables. To emulate the same, programmers use the concept of name mangling. In name mangling, a double underscore prefix is placed before variable names. It causes the Python interpreter to rewrite the attribute name in order to avoid naming conflict.

Output Screenshots

Number of Features, Feature Type and Statistics



```
print("Number of features : ",df.shape[1])
```

```
Number of features :  5
```

```
for (colname, coldata) in df.iteritems():
  feature_type = 'numerical' if (coldata.dtype=='float64') else 'nominal'
  print('\nColumn Name - ',colname)
  print('Feature type - ',feature_type)
```

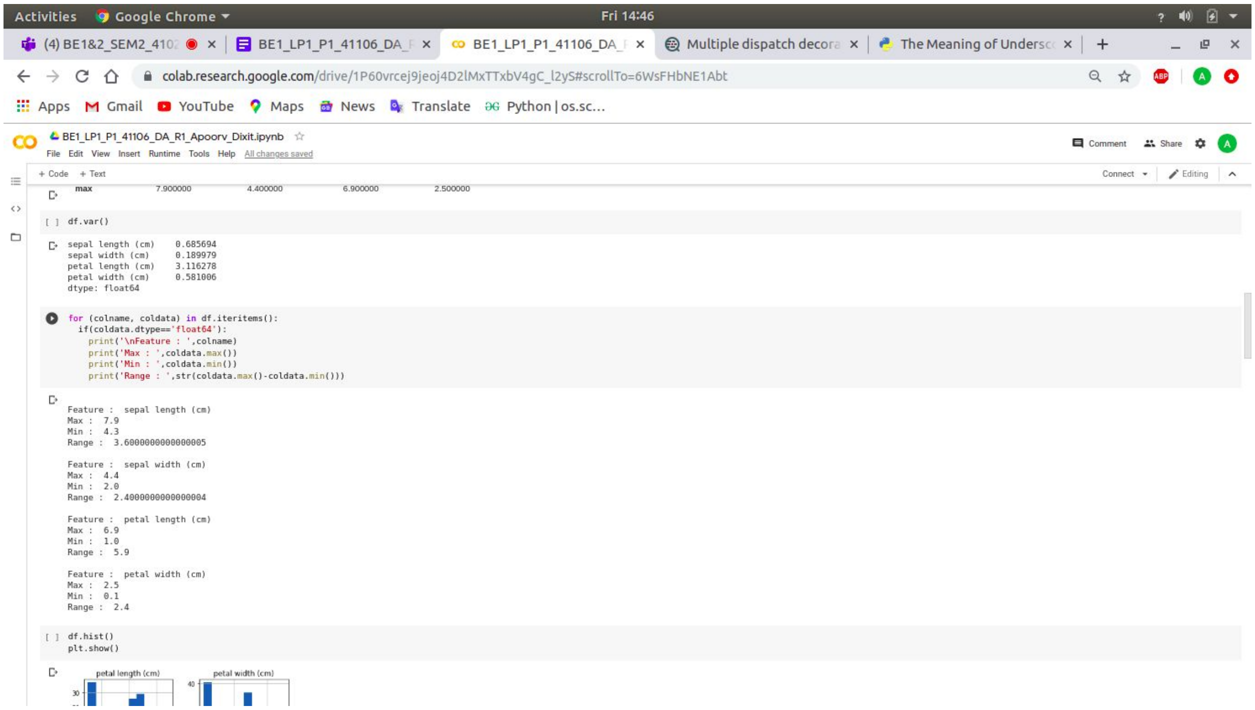
```

Column Name -  sepal length (cm)
Feature type -  numerical

Column Name -  sepal width (cm)
Feature type -  numerical

Column Name -  petal length (cm)
Feature type -  numerical

Column Name -  petal width (cm)
Feature type -  numerical

Column Name -  species
Feature type -  nominal
```

```
df.describe()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

```
df.var()
```

Variance and Range



```
max         7.900000         4.400000         6.900000         2.500000
```

```
df.var()
```

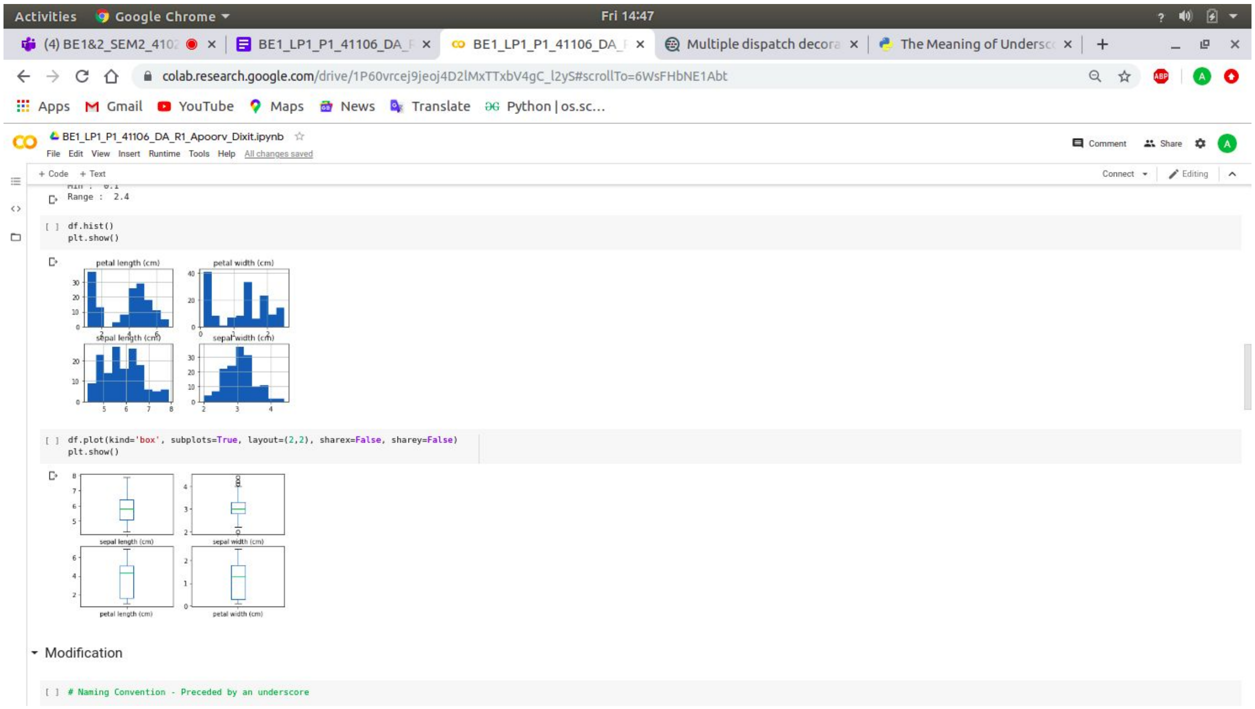
```
sepal length (cm)    0.685694
sepal width (cm)     0.189979
petal length (cm)    3.116278
petal width (cm)     0.581006
dtype: float64
```

```
for (colname, coldata) in df.iteritems():
  if(coldata.dtype=='float64'):
    print('\nFeature : ',colname)
    print('Max : ',coldata.max())
    print('Min : ',coldata.min())
    print('Range : ',str(coldata.max()-coldata.min()))
```

```
Feature :  sepal length (cm)
Max :  7.9
Min :  4.3
Range :  3.6000000000000005

Feature :  sepal width (cm)
Max :  4.4
Min :  2.0
Range :  2.4000000000000004

Feature :  petal length (cm)
Max :  6.9
Min :  1.0
Range :  5.9

Feature :  petal width (cm)
Max :  2.5
Min :  0.1
Range :  2.4
```

```
df.hist()
plt.show()
```

Histogram Plots and Box Plots



```
Range :  2.4
```

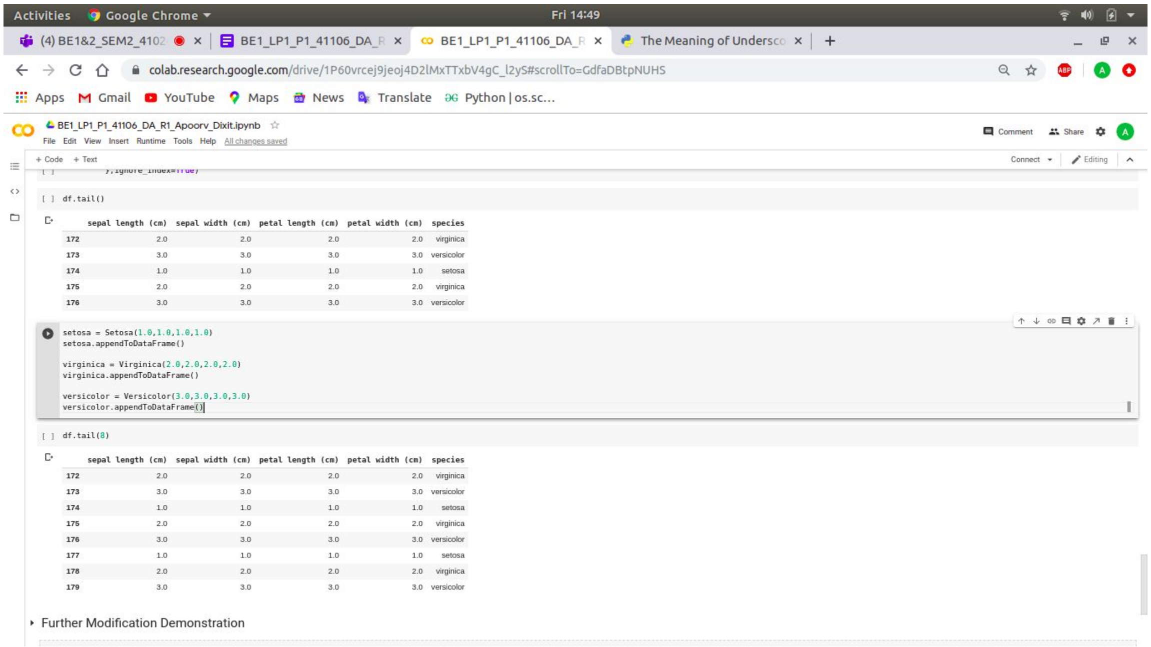
```
df.hist()
plt.show()
```

```
df.plot(kind='box', subplots=True, layout=(2,2), sharex=False, sharey=False)
plt.show()
```

Modification

```
# Naming Convention - Preceded by an underscore
```

Modification - AppendToDataFrame() function demonstration of Setosa, Virginica and Versicolor classes


Activities
Google Chrome
Fri 14:49
(4) BE1&2_SEM2_4102
BE1_LP1_P1_41106_DA_R
BE1_LP1_P1_41106_DA_R
The Meaning of Undersco
colab.research.google.com/drive/1P60vrcej9jeoj4D2lMxTTxbV4gC_l2yS#scrollTo=GdfaDBtpNUHS
Apps
Gmail
YouTube
Maps
News
Translate
Python | os.sc...
BE1_LP1_P1_41106_DA_R1_Apoorv_Dixit.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Connect
Editing
Further Modification Demonstration

```
df.tail()
```

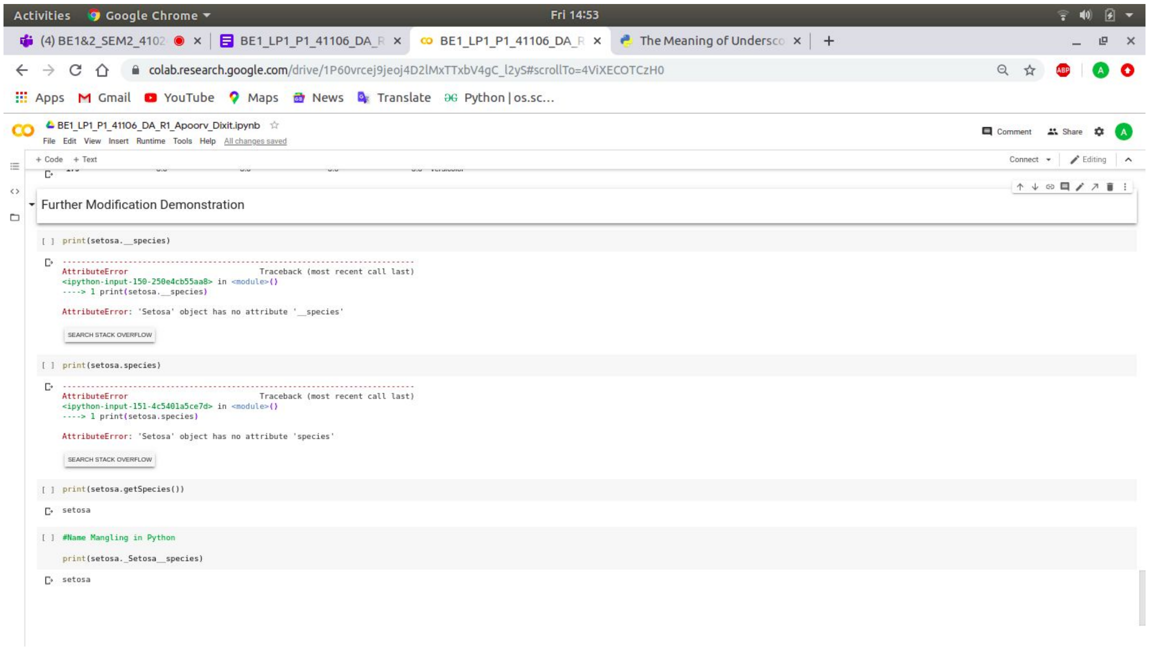
	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species
172	2.0	2.0	2.0	2.0	virginica
173	3.0	3.0	3.0	3.0	versicolor
174	1.0	1.0	1.0	1.0	setosa
175	2.0	2.0	2.0	2.0	virginica
176	3.0	3.0	3.0	3.0	versicolor

```
setosa = Setosa(1.0,1.0,1.0,1.0)
setosa.appendToDataFrame()

virginica = Virginica(2.0,2.0,2.0,2.0)
virginica.appendToDataFrame()

versicolor = Versicolor(3.0,3.0,3.0,3.0)
versicolor.appendToDataFrame()
```

```
df.tail(8)
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species
172	2.0	2.0	2.0	2.0	virginica
173	3.0	3.0	3.0	3.0	versicolor
174	1.0	1.0	1.0	1.0	setosa
175	2.0	2.0	2.0	2.0	virginica
176	3.0	3.0	3.0	3.0	versicolor
177	1.0	1.0	1.0	1.0	setosa
178	2.0	2.0	2.0	2.0	virginica
179	3.0	3.0	3.0	3.0	versicolor

Modification - Demonstrating Name Mangling in Python



```
print(setosa.__species)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-150-250e4cb55aa8> in <module>()
----> 1 print(setosa.__species)

AttributeError: 'Setosa' object has no attribute '__species'
```

```
print(setosa.species)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-151-4c5401a5ce7d> in <module>()
----> 1 print(setosa.species)

AttributeError: 'Setosa' object has no attribute 'species'
```

```
print(setosa.getSpecies())
```

```
setosa
```

```
#Name Mangling in Python

print(setosa._Setosa__species)
```

```
setosa
```

Conclusion

We have successfully computed statistics on the features of the Iris dataset, and used histograms and boxplot on the features of the dataset.