MSDS 6306_Live Session Homework 13--Angela Horacek

## 1. CREATE A LIST NAMED my_list IN PYTHON WITH THE FOLLOWING DATA POINTS:

```
In [1]: my_list = [45.4, 44.2, 36.8, 35.1, 39.0, 60.0, 47.4, 41.1, 45.8, 35.6]
```

1.a. Print the 5th element in the list.

```
In [2]: my_list[4]
```

```
Out[2]: 39.0
```

1.b. Append 55.2 to my_list

```
In [3]: var1 = 55.2
        my_list.append(var1)
        print (my_list)
```

```
[45.4, 44.2, 36.8, 35.1, 39.0, 60.0, 47.4, 41.1, 45.8, 35.6, 55.2]
```

1.c. Remove the 6th element in the list.

```
In [4]: del my_list [5]
        print (my_list)
```

```
[45.4, 44.2, 36.8, 35.1, 39.0, 47.4, 41.1, 45.8, 35.6, 55.2]
```

1.d Iterate over the list to print data points greater than 45.

```
In [5]: for val in my_list:
            if val > 45:
                print (val)
```

```
45.4
47.4
45.8
55.2
```

## 2. INTRODUCTION TO NUMPY

2.a. Import the numpy library using the following command import numpy.

```
In [6]: import numpy
```

2.b. Declare numpy array with the same data points as in my_list using numpy.array()

In [7]:
```
array1 = numpy.array([45.4, 44.2, 36.8, 35.1, 39.0, 60.0, 47.4, 41.1, 45.8, 35.6])
print (array1)
```

```
[ 45.4  44.2  36.8  35.1  39.   60.   47.4  41.1  45.8  35.6]
```

2.c. Compute the mean and standard deviation using numpy.mean() and numpy.std() of the above array.

In [8]:
```
mean1 = numpy.mean(array1)
print ("Mean is", mean1)
sd1 = numpy.std(array1)
print ("Standard deviation is", sd1)
```

```
Mean is 43.04
Standard deviation is 7.06118970146
```

2.d. Use logical referencing to get only those values that are less than 45.

In [9]:
```
array1[numpy.where(array1<45)]
```

Out[9]:
```
array([ 44.2,  36.8,  35.1,  39. ,  41.1,  35.6])
```

2.e. Compute the max and min or the array using numpy.max() and numpy.min().

In [10]:
```
max1 = numpy.max(array1)
print ("Max is", max1)
min1 = numpy.min(array1)
print ("Min is", min1)
```

```
Max is 60.0
Min is 35.1
```

## 3. INTRODUCTION TO PANDAS

3.a. Import the pandas library-import pandas.

In [11]:
```
import pandas
```

3.b. Read the IRIS dataset into iris using pandas.read_csv(). Data file-

In [12]:
```
iris = pandas.read_csv("C:/Users/Angela/Anaconda3/Lib/site-packages/blaze/examples/
data/iris.csv", index_col=False)
```

3.c. Using iris.head(), display the head of the dataset. Note: I had to add an "id" column into the original file to show that I could drop the id column in 3.d.

In [13]:
```
iris.head()
```

Out[13]:

| | id | sepal_length | sepal_width | petal_length | petal_width | species |
|---|---|---|---|---|---|---|
| **0** | 1 | 5.1 | 3.5 | 1.4 | 0.2 | Iris-setosa |
| **1** | 2 | 4.9 | 3.0 | 1.4 | 0.2 | Iris-setosa |
| **2** | 3 | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa |
| **3** | 4 | 4.6 | 3.1 | 1.5 | 0.2 | Iris-setosa |
| **4** | 5 | 5.0 | 3.6 | 1.4 | 0.2 | Iris-setosa |

3.d. Use DataFrame.drop() to drop the id column.

In [16]:
```
iris.drop(iris.columns[[0]], axis=1)
```

Out[16]:

| | sepal_length | sepal_width | petal_length | petal_width | species |
|---|---|---|---|---|---|
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | Iris-setosa |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | Iris-setosa |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | Iris-setosa |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | Iris-setosa |
| **5** | 5.4 | 3.9 | 1.7 | 0.4 | Iris-setosa |
| **6** | 4.6 | 3.4 | 1.4 | 0.3 | Iris-setosa |
| **7** | 5.0 | 3.4 | 1.5 | 0.2 | Iris-setosa |
| **8** | 4.4 | 2.9 | 1.4 | 0.2 | Iris-setosa |
| **9** | 4.9 | 3.1 | 1.5 | 0.1 | Iris-setosa |
| **10** | 5.4 | 3.7 | 1.5 | 0.2 | Iris-setosa |
| **11** | 4.8 | 3.4 | 1.6 | 0.2 | Iris-setosa |
| **12** | 4.8 | 3.0 | 1.4 | 0.1 | Iris-setosa |
| **13** | 4.3 | 3.0 | 1.1 | 0.1 | Iris-setosa |
| **14** | 5.8 | 4.0 | 1.2 | 0.2 | Iris-setosa |
| **15** | 5.7 | 4.4 | 1.5 | 0.4 | Iris-setosa |
| **16** | 5.4 | 3.9 | 1.3 | 0.4 | Iris-setosa |
| **17** | 5.1 | 3.5 | 1.4 | 0.3 | Iris-setosa |
| **18** | 5.7 | 3.8 | 1.7 | 0.3 | Iris-setosa |
| **19** | 5.1 | 3.8 | 1.5 | 0.3 | Iris-setosa |
| **20** | 5.4 | 3.4 | 1.7 | 0.2 | Iris-setosa |
| **21** | 5.1 | 3.7 | 1.5 | 0.4 | Iris-setosa |
| **22** | 4.6 | 3.6 | 1.0 | 0.2 | Iris-setosa |
| **23** | 5.1 | 3.3 | 1.7 | 0.5 | Iris-setosa |
| **24** | 4.8 | 3.4 | 1.9 | 0.2 | Iris-setosa |
| **25** | 5.0 | 3.0 | 1.6 | 0.2 | Iris-setosa |
| **26** | 5.0 | 3.4 | 1.6 | 0.4 | Iris-setosa |
| **27** | 5.2 | 3.5 | 1.5 | 0.2 | Iris-setosa |
| **28** | 5.2 | 3.4 | 1.4 | 0.2 | Iris-setosa |
| **29** | 4.7 | 3.2 | 1.6 | 0.2 | Iris-setosa |
| **...** | ... | ... | ... | ... | ... |
| **120** | 6.9 | 3.2 | 5.7 | 2.3 | Iris-virginica |
| **121** | 5.6 | 2.8 | 4.9 | 2.0 | Iris-virginica |
| **122** | 7.7 | 2.8 | 6.7 | 2.0 | Iris-virginica |
| **123** | 6.3 | 2.7 | 4.9 | 1.8 | Iris-virginica |
| **124** | 6.7 | 3.3 | 5.7 | 2.1 | Iris-virginica |
| **125** | 7.2 | 3.2 | 6.0 | 1.8 | Iris-virginica |

In [ ]:
```
iris.drop(iris.columns[[0]], axis=1)
```

3.e. Subset dataframe to create a new data frame that includes only the measurements for the setosa species.

In [ ]:
```
setosa_df = iris[iris.species == "Iris-setosa"]
setosa_df
```

3.f. Use DataFrame.describe() to get the summary statistics.

In [ ]:
```
iris.describe()
```

4.g. Use DataFrame.grouby() to create grouped data frames by Species and compute summary statistics using DataFrame.describe()

In [ ]:
```
species_group = iris.groupby(['species']).describe()
species_group
```

3.h. Use DataFrame.boxplot() to plot boxplots by Species.

In [ ]:
```
%matplotlib inline
species_group.boxplot(return_type='dict')
```

3.i. Plot a scatter matrix plot using the seaborn library. Use the following to load and plot. a) import seaborn as sns (note: need to install first "conda install seaborn b) Seaborn.pairplot(dataframe, by='column_name')

In [ ]:
```
import seaborn as sns
```

In [ ]:
```
sns.pairplot(iris, hue='species')
```