**Fraction\_xy.py**

Line 1 to 10 installing necessary libraries for the project

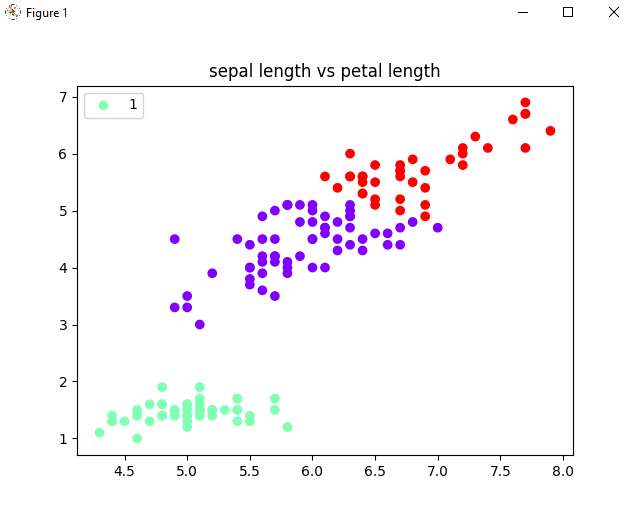
Line 12 and 13 loaded the iris dataset from Sklearn datasets

Line 16 adding the species column onto our datasets

Selected the relevant features from the dataset the sepal width(cm), sepal length (cm), petal width (cm), petal length (cm). code used for this is in line 23

Declared the number of clusters in line 27, chose number of clusters(k) as 3 because it is the best for this type of data as shown in elbow method in the qerror.py file

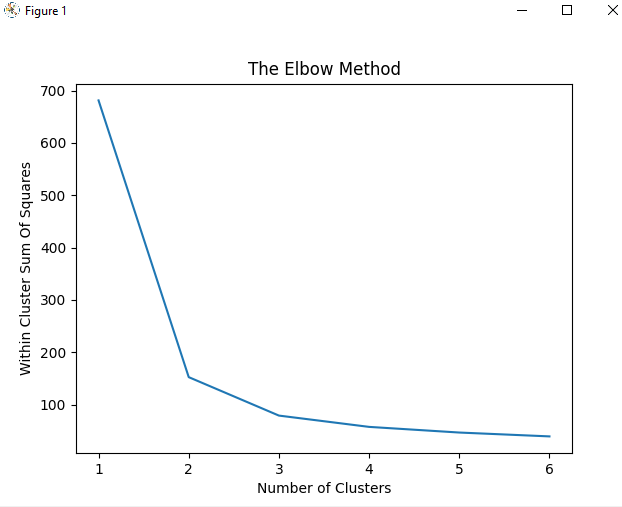
Using matplotlib the results are shown basing on the three clusters. For instance, creating a scatter plot for sepal length and petal length we get this type of plot



**2. Qerror.py**

The first 24 lines similar to the fraction\_xy.py file.

Introduced the WCSS (Within Cluster Sum of Squares) to get the squared distance between each centroid and point within cluster. This is helpful in the elbow technique where an elbow shaped diagram is created which shows the optimal number of clusters that can be formed from the dataset. This is shown by the elbow flattening out on cluster 3. The kmeans.inertia\_ value shows the quantization error value for our model



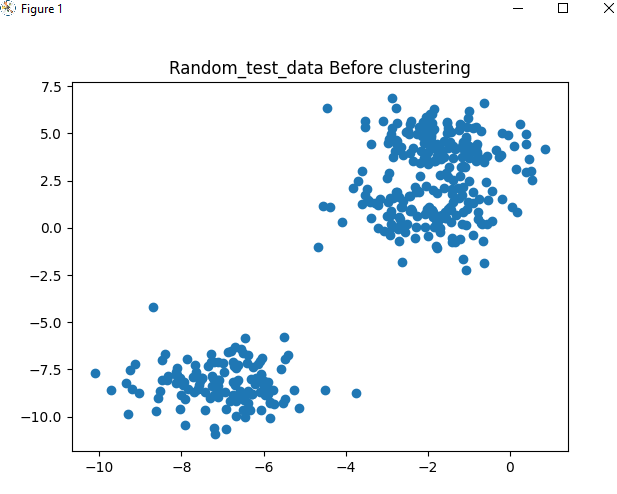
**3. Iloyd.py and kmeanspp.py**

Created a random dataset of arrays using the random generator function for our program. Make sure you type in as the file name appears. The k, r and the data file all are given from the command line. For our data file, we will use “data1.txt” given from the command line. Remember this data file is a randomly generated dataset from the code above in line 10 in the iloyd.py file. Data1.txt dataset is used for both programs.

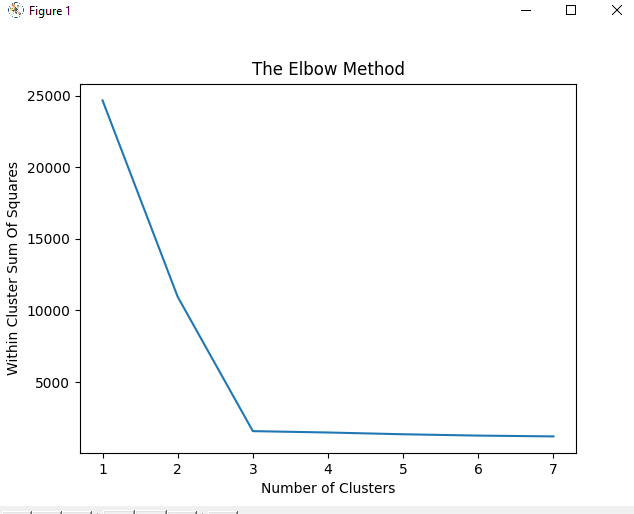
For our output we have the quantization value and the 0…k-1 values(clusters). Graphs for results from iloyd.py and kmeanspp.py are in plot.py file

**4. plot.py**

Used the same code as the one in iloyd.py and kmeanspp.py programs. This file is meant to give chart results from the two programs above. One of the chart is given below.



Using the elbow method shown below, three clusters are the best for this dataset with seven iterations.



N/B still working on improving kmeans++ and the experimentalkmeanspp.py