**Jonathan Houston**

**Omu075**

**1. [35] Trace the k-means algorithm to find three clusters in the given dataset. Assume that the Euclidean distance is used and the initially randomly selected centroids are (8, 4), (5, 8), and (1, 2), respectively.**

**(a) Find the cluster for each data point with current centroids**

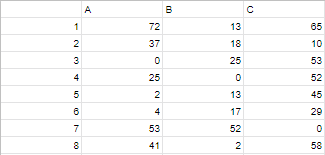
Use the centroid to find the clusters

A = (8, 4)

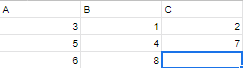
B = (5, 8)

C = (1, 2)

Find the distance between all the points (Euclidean distance)



Cluster them according to smallest distance and centroids



**(b) Find the new centroids (mean)**

A: (7 , 3.7)

X: (8 + 7 + 6) / 3 = 7

Y: (4 + 5 + 4) / 3 = 4.3

B: (3.7, 9)

X: (2 + 5 + 4) / 3 = 3.7

Y: (10 + 8 + 9) / 3 = 9

C: (1.5 , 3.5)

X: (2 + 1) / 2 = 1.5

Y: (5 + 2) / 2 = 3.5

**(c) Repeat (a) and (b)**

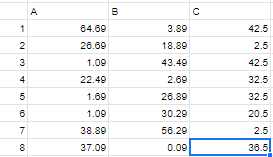
Use the centroid to find the clusters

A = (7 , 3.7)

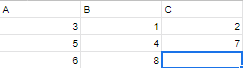
B = (3.7, 9)

C = (1.5 , 3.5)

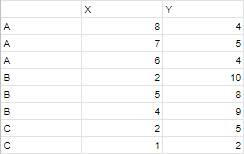
Find the distance between all the points (Euclidean distance)



Cluster them according to smallest distance and centroids



**(d) Show the final three clusters.**



**2. [35] Trace the PAM algorithm to find three clusters in the given dataset. Assume that the**

**Euclidean distance is used and the randomly selected initial medoids are (2, 5), (5, 8), and**

**(1, 2), respectively.**

**(Used as reference: https://en.wikipedia.org/wiki/K-medoids)**

1. **Trace the execution of the first iteration of the main loop.**

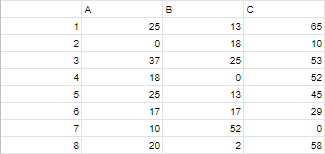
Use the medoids:

A: (2, 5)

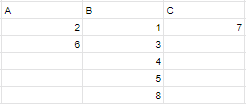
B: (5, 8),

C: (1, 2)

Calculate the distance between each point and medoid (Euclidean)



**(b) Show the final three clusters.**



Cost for clusters:

A: 0 + 17 = **17**

B: 13 + 25 + 0 + 13 + 2 = **53**

C: 0 = **0**

**3. [30] Write a Python statements in a notebook to use scikit-learn cluster.KMeans to cluster the data in hmwk08.csv. You should load the data into a DataFrame. Try to cluster the data using different subset of the columns, and different number of clusters. For an additional 5 points bonus, plot the clusters.**

Answer can be found in ‘Data Mining HW8 – Part 3.ipynb’