COMP 472: Artificial Intelligence

k-means Clustering

Solutions

Python code associated with question 2 is available on Moodle.

Question 1 Consider the following data set with two attributes **a1** and **a2**.

	a1	**a2**
Data1	1.0	1.0
Data2	1.5	2.0
Data3	3.0	4.0
Data4	5.0	7.0
Data5	3.5	5.0
Data6	4.5	5.0
Data7	3.5	4.5

(a) Assume that we initialize the clusters using Data1 and Data4 as initial centroids. Using the Euclidean distance, in which cluster will each individuals be initially assigned? Do not perform the entire clustering - only do an initial assignment of points.

	Individuals	Centroid
Cluster 1	1	(1.0, 1.0)
Cluster 2	4	(5.0, 7.0)

For Data2:

Distance to Centroid 1: $\sqrt{(1.5-1.0)^2+(2.0-1.0)^2} \approx 1.1$

Distance to Centroid 2: $\sqrt{(1.5-5.0)^2+(2.0-7.0)^2} \approx 6.1$

	Distance to Centroid 1	Distance to Centroid 2
Data1	0.0	7.2
Data2	1.1	6.1
Data3	3.6	3.6
Data4	7.2	0.0
Data5	4.7	2.5
Data6	5.3	2.1
Data7	4.3	2.9

(b) Recalculate the centroids based on the current partition, reassign the individuals based on the new centroids. Which individuals (if any) changed clusters as a result?

For cluster 1:

$$\frac{1.0 + 1.5 + 3.0}{3} = 1.83$$

$$\frac{1.0 + 2.0 + 4.0}{3} \approx 2.33$$

	Individuals	Centroid
Cluster 1	1, 2, 3	(1.83, 2.33)
Cluster 2	4, 5, 6, 7	(4.13, 5.38)

	Distance to Centroid 1	Distance to Centroid 2
Data1	1.6	5.4
Data2	0.5	4.3
Data3	2.0	1.8
Data4	5.6	1.9
Data5	3.1	0.7
Data6	3.8	0.5
Data7	2.7	1.1

Data3 changed from cluster 1 to cluster 2. All other data points remained in the same cluster.

Question 2 Consider the following data set of points with two attributes **x** and **y**.

	x	y
Data1	0.0	1.0
Data2	1.0	0.0
Data3	-1.0	0.0
Data4	0.0	-1.0
Data5	0.5	0.5
Data6	-0.5	-0.5
Data7	-0.5	0.5
Data8	0.5	-0.5
Data9	4.0	4.0
Data10	-4.0	-4.0
Data11	-4.0	4.0
Data12	4.0	-4.0
Data13	4.0	0.0
Data14	-4.0	0.0
Data15	0.0	4.0
Data16	0.0	-4.0

(a) Apply k-means on the data set above given the 2 configurations of initial centroids indicated in the table below.

	Initial Centroid 1	Initial Centroid 2
Configuration 1	(0.0, 0.0)	(4.0, 4.0)
Configuration 2	(-5.0, 0.0)	(2.0, 0.0)

You need to apply k-means separately for each configuration and report the two clusters you have found for each setup.

Let's start with configuration 1:

Config. 1	Distance to Centroid 1	Distance to Centroid 2	Cluster
Data1	1.0	5.0	1
Data2	1.0	5.0	1
Data3	1.0	6.4	1
Data4	1.0	6.4	1
Data5	0.7	4.9	1
Data6	0.7	6.3	1
Data7	0.7	5.7	1
Data8	0.7	5.7	1
Data9	5.6	0.0	2
Data10	5.6	11.3	1
Data11	5.6	8.0	1
Data12	5.6	8.0	1
Data13	4.0	4.0	1
Data14	4.0	8.9	1
Data15	4.0	4.0	1
Data16	4.0	8.9	1

Now that we have assigned each data to their closest centroid, we have to re-compute the new clusters centroid and repeat the process until none of the data instances change cluster.

For cluster 1:

$$x_1 = \frac{0.0 + 1.0 - 1.0 + 0.0 + 0.5 - 0.5 + 0.5 - 0.5 - 4.0 + 4.0 - 4.0 + 4.0 - 4.0 + 0.0 + 0.0}{15} \approx -0.266$$

$$y_1 = \frac{0.0 + 1.0 - 1.0 + 0.0 + 0.5 - 0.5 + 0.5 - 0.5 - 4.0 + 4.0 - 4.0 + 0.0 + 0.0 + 4.0 - 4.0}{15} \approx -0.266$$

For cluster 2:

$$x_1 = \frac{4.0}{1} \approx 4.0$$
$$y_1 = \frac{4.0}{1} \approx 4.0$$

	Individuals	Centroid
Cluster 1	1, 2, ..., 8, 10, 11, 12, 13, 14, 15, 16	(-0.266, -0.266)
Cluster 2	9	(4.0, 4.0)

Let's calculate the distances to the new centroids for the second time.

Config. 1	Distance to Centroid 1	Distance to Centroid 2	Cluster
Data1	1.29	5.0	1
Data2	1.29	5.0	1
Data3	0.78	6.4	1
Data4	0.78	6.4	1
Data5	1.08	4.9	1
Data6	0.80	6.3	1
Data7	0.80	5.7	1
Data8	0.32	5.7	1
Data9	6.03	0.00	2
Data10	5.27	11.3	1
Data11	5.66	8.0	1
Data12	5.66	8.0	1
Data13	4.27	4.0	2
Data14	3.74	8.9	1
Data15	4.27	4.0	2
Data16	3.74	8.9	1

Let's calculate the centroids again:
For cluster 1:

$$x_1 = \frac{0.0 + 1.0 - 1.0 + 0.0 + 0.5 - 0.5 + 0.5 - 0.5 - 4.0 - 4.0 + 4.0 - 4.0 + 0.0}{13} \approx -0.615$$

$$y_1 = \frac{0.0 + 1.0 - 1.0 + 0.0 + 0.5 - 0.5 + 0.5 - 0.5 - 4.0 - 4.0 + 4.0 - 4.0 + 0.0}{13} \approx -0.615$$

For cluster 2:

$$x_1 = \frac{4.0 + 4.0}{3} \approx 2.667$$

$$y_1 = \frac{4.0 + 4.0}{3} \approx 2.667$$

Let's calculate the distances to the new centroids for the third time. But since one of the centroids is the same we can just copy the distance values for that centroid.

Config. 1	Distance to Centroid 1	Distance to Centroid 2	Cluster
Data1	1.72	3.14	1
Data2	1.72	3.14	1
Data3	0.72	3.66	1
Data4	0.72	3.66	1
Data5	1.57	3.06	1
Data6	0.02	4.48	1
Data7	1.11	3.84	1
Data8	1.11	3.84	1
Data9	6.52	1.88	2
Data10	4.78	9.42	1
Data11	5.72	6.79	1
Data12	5.72	6.79	1
Data13	4.65	2.98	2
Data14	3.44	7.18	1
Data15	4.65	2.98	2
Data16	3.44	7.18	1

Since none of the instances changed cluster, we can stop here and report the clusters we have found.

Let's apply k-means on our data set again using the second configuration:

Config. 2	Distance to Centroid 1	Distance to Centroid 2	Cluster
Data1	5.1	2.2	2
Data2	6.0	1.0	2
Data3	4.0	3.0	2
Data4	5.1	2.2	2
Data5	5.5	1.5	2
Data6	4.5	2.5	2
Data7	4.5	2.5	2
Data8	5.5	1.5	2
Data9	9.8	4.4	2
Data10	4.1	7.2	1
Data11	4.1	7.2	1
Data12	9.8	4.4	2
Data13	9.0	2.0	2
Data14	1.0	6.0	1
Data15	6.4	4.4	2
Data16	6.4	4.4	2

Now, we have to find the clusters centroid and repeat the process until none of the data instances change cluster.

For cluster 1:

$$x_1 = \frac{-4.0 - 4.0 - 4.0}{3} = -4.0$$

$$y_1 = \frac{-4.0 + 4.0 + 0.0}{3} = 0.0$$

For cluster 2:

$$x_1 = \frac{12.0}{13} \approx 0.92$$

$$y_1 = \frac{0.0 + 4.0}{13} = 0.0$$

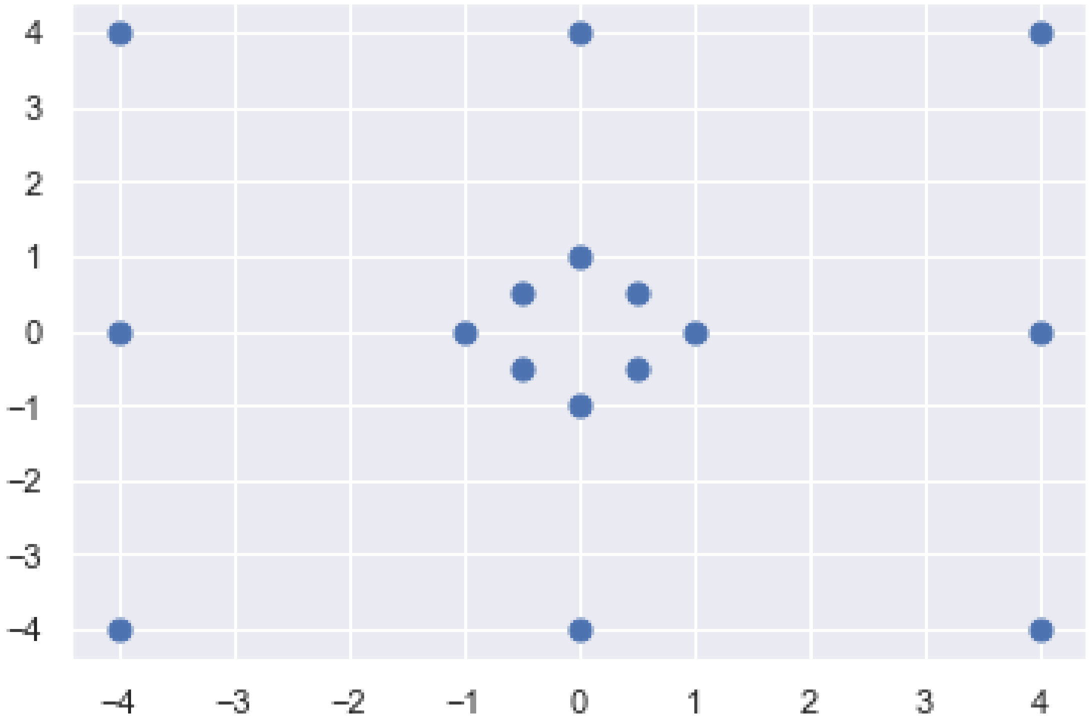
	Individuals	Centroid
Cluster 1	10, 11, 14	(-4.0, 0.0)
Cluster 2	1, 2, ..., 8, 9, 12, 13, 15, 16	(0.92, 0.0)

Let's calculate the distances to our new cluster centroids.

Config. 2	Distance to Centroid 1	Distance to Centroid 2	Cluster
Data1	4.1	1.3	2
Data2	5.0	0.1	2
Data3	3.0	1.9	2
Data4	4.1	1.3	2
Data5	4.5	0.6	2
Data6	3.5	1.5	2
Data7	3.5	1.5	2
Data8	4.5	0.6	2
Data9	8.9	5.1	2
Data10	4.0	6.3	1
Data11	4.0	6.3	1
Data12	8.9	5.1	2
Data13	8.0	3.1	2
Data14	0.0	4.9	1
Data15	5.6	4.1	2
Data16	5.6	4.1	2

None of the data instances changed cluster, so the algorithm will stop here and final centroids are the same as in the previous step.

(b) Plot the data and analyze your results from **part a**. Do the clusters you have found seem reasonable?
The figure below shows the distribution of the data. As the figure shows, the data cannot be naturally separated into two clear clusters and so k-means failed to find representative clusters. k-means is limited to linear cluster boundaries which means that it will fail to find more complicated boundaries.



Question 2-b Visualization of the data set

(c) Having **part a** in mind, were you surprised that the two runs of k-means with the 2 different initial configurations gave different resulting clusters? *No, k-means is very sensitive to the initial choice of centroids and usually, if you run k-means with two random sets of centroids you won't get the same results.*