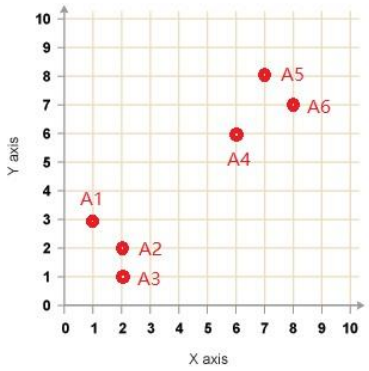
The K-means algorithm

Data Processing Using Python by Dazhuang@NJU

The K-means algorithm is a typical distance-based cluster analysis algorithm which uses distance as the evaluation measure. The smaller the distance of two object is, the more similar the two objects are.
In the process, a data cluster is made up by objects close to each other. Therefore, the algorithm is aimed to calculate compact and standalone clusters.
Suppose to divide objects into k clusters. The algorithm will be:
(1) Randomly pick up k objects to be the centroid of each cluster. Initially each cluster will be represented by the only object.
(2) For each object which is not one of the centroids, assign it to the cluster of which the centroid is the nearest.
(3) Re-calculate the centroid of each cluster.
(4) Repeat step (2) and (3) until the new centroid and old centroid of each cluster are identical or closer than a preset threshold. At the point, the algorithm ends.

As an example, if we randomly pick up a few data, the algorithm will be:



Given six points - A1, A2, … A6:

	X	Y
A1	1	3
A2	2	2
A3	2	1
A4	6	6
A5	7	8
A6	8	7

If we want to have two cluster, the procedure will be:
(1) Assume A1 and A2 as initial centroid:
(2) Calculate the Euclidean distance of each non-centroids (A3-A6) and each centroids (A1-A2) with the formular $d = sqrt((x1-x2)^2+(y1-y2)^2)$

	A1	A2
A3	2.24	1
A4	5.83	5.66
A5	6.4	6.4
A6	8.06	7.81

(3) Based on the table above, A3,A4 and A6 are all closer to A2 while A5 is evenly close to both A1 and A2. Let's just assign A5 to the cluster where A2 is. Then we will have two new clusters with all objects included:

Cluster 1： A1

Cluster 2： A2， A3， A4， A5， A6

(4) Calculate new centroids

The new centroid of cluster 1： A1

The new centroid C_temp of cluster 2 is the average of all dimenons.

((A2.x+A3.x+A4.x+A5.x+A6.x)/5 , (A2.y+A3.y+A4.y+A5.y+A6.y)/5)=(5, 4.8)

	A1	C_temp
A2	1.41	4.1
A3	2.24	4.84
A4	5.83	1.56
A5	6.4	3.77
A6	8.06	3.72

(5) Base on the distances to the new centroids, objects are divided into two new clusters.

Cluster 1： A1， A2， A3

Cluster 2： A4， A5， A6

New centroid 1 "C_temp1"： ((A1.x+A2.x+A3.x)/3 , (A1.y+A2.y+A3.y)/3)=(1.67, 2)

New centroid 2 "C_temp2"： ((A4.x+A5.x+A6.x)/3 , (A4.y+A5.y+A6.y)/3)=(7, 7)

	C_temp1	C_temp2
A1	1.2	7.21
A2	0.33	7.07
A3	1.05	7.81
A4	5.89	1.41
A5	6.66	1
A6	6.71	1

(6) bingo ヽ(◑‿◐)ノ

The two new clusters are having the same objects as the previous two clusters respectively and the processing ends here.

Cluster 1： A1， A2， A3

Cluster 2： A4， A5， A6

Note

(1) The selection of k in K-means algorithm is subject to human experience.

(2) Euclidean distance or cosine similarity is usually selected to calculate distance. Euclidean distance is based on position coordinates which mainly

represent the difference of individual values. Cosine similarity represents more directional difference than value difference. The more the value of cos θ is close to 1, the more similar the two objects are similar. Cosine similarity can offset the difference caused by various measure systems.

(3) Initialization of centroid chosen from the space is very vital and sometimes it will get stuck in local optima.