

ELEC 4727/5727**Computer Vision: Introduction to Machine Learning****KNN Analysis**

[Part A] K-NN (K Nearest Neighbor Classification)- Perform 3-NN analysis for the following 2-dimensional training values

Data	X	Y	Classification
A	1	1	Good
B	3	7	Bad
C	3	3	Good
D	2	1	Good
E	3	2	Bad

Use the query attributes of the new point F: [X= 3, Y= 4]

Use Euclidean Distance as a measure and MAJORITY to designate the classification as: Good, Bad, No Classification. Clearly fill out the following table showing the 3 nearest neighbors (give data rank 1=min, 2=second min, ..., 5=farthest point)

Data	Distance to Point F	Nearest Neighbor Rank
A	$\sqrt{(1-3)^2 + (1-4)^2} = 3.61$	5
B	$\sqrt{(3-3)^2 + (7-4)^2} = 3$	3
C	$\sqrt{(3-3)^2 + (3-4)^2} = 1$	1
D	$\sqrt{(2-3)^2 + (1-4)^2} = 3.16$	4
E	$\sqrt{(3-3)^2 + (2-4)^2} = 2$	2

Classification	Good or Bad
K=1	Good
K=3	Bad

KMeans Clustering

[Part A] Perform iterative K-Means (K=2) Clustering for the following 2-dimensional attributes.

Point	X	Y
A	2	1
B	3	1
C	5	2
D	6	0

Your final answer should designate the coordinates for 2 centroids and designate the cluster that each data point resides. You must show your work to get credit and your answer should be entered into the final table:

Iteration 0:

Centroids:

	Point	X-coordinate	Y-coordinate
Centroid 1	A	2	2
Centroid 2	B	3	1

Distance Table: Points to Centroids

You do not need to use sqrt, just the magnitude of the sum of squared differences of x and y positions

Point	Centroid 1	Centroid 2
A	$ (2^2 - 2^2) + (1^2 - 2^2) = 3$	$ (2^2 - 3^2) + (1^2 - 1^2) = 5$
B	$ (3^2 - 2^2) + (1^2 - 2^2) = 2$	$ (3^2 - 3^2) + (1^2 - 1^2) = 0$
C	$ (5^2 - 2^2) + (2^2 - 2^2) = 21$	$ (5^2 - 3^2) + (2^2 - 1^2) = 19$
D	$ (6^2 - 2^2) + (0^2 - 2^2) = 28$	$ (6^2 - 3^2) + (0^2 - 1^2) = 26$

Group Assignment based on Distance Table

I based my grouping off of whether points A,B,C, or D were closer to centroid 1 or centroid 2.

Point	Assign Group 1 or 2 based on closest to the centroids
A	Centroid 1
B	Centroid 2
C	Centroid 2
D	Centroid 2

Derive the new centroids based on group assignment

Centroids:

Because Centroid 1 only has 1 member, the new centroid becomes that point A(2,1).

Calculation for Centroid 2 = $\left(\frac{3+5+6}{3}, \frac{1+2+0}{3}\right)$

	X-coordinate	Y-coordinate
Centroid 1	2	1
Centroid 2	4.67	1

Distance Table: Points to Centroids

You do not need to use sqrt, just the magnitude of the sum of squared differences of x and y positions

Point	Centroid 1	Centroid 2
A	$ (2^2 - 2^2) + (1^2 - 1^2) = 0$	$ (2^2 - 4.67^2) + (1^2 - 1^2) = 17.81$
B	$ (3^2 - 2^2) + (1^2 - 1^2) = 5$	$ (3^2 - 4.67^2) + (1^2 - 1^2) = 12.81$
C	$ (5^2 - 2^2) + (2^2 - 1^2) = 24$	$ (5^2 - 4.67^2) + (2^2 - 1^2) = 6.19$
D	$ (6^2 - 2^2) + (0^2 - 1^2) = 31$	$ (6^2 - 4.67^2) + (0^2 - 1^2) = 13.19$

Group Assignment based on Distance Table

I based my grouping off of whether points A,B,C, or D were closer to centroid 1 or centroid 2.

Point	Assign Group 1 or 2 based on closest to the centroids
A	Centroid 1
B	Centroid 1
C	Centroid 2
D	Centroid 2

KNN (Graduate students)

Write a Python function for processing an Numpy array of (x,y,class) into the KNN

Example:

D =
np.array([[3,13,0],[2.75,5,0],[3.5,1,0],[3.75,9,0],[1,9,1],[1.5,5,1],[5.5,1,1],[5.25,5,1],[6.5,9,1],[6,13,1]])

KMeans Clustering (Graduate students)

Write a Python function for processing an Numpy array of (x,y,class) into the Kmeans clustering algorithm

Example:

D =
np.array([[3,13,0],[2.75,5,0],[3.5,1,0],[3.75,9,0],[1,9,1],[1.5,5,1],[5.5,1,1],[5.25,5,1],[6.5,9,1],[6,13,1]])