Unsupervised learning

Task 1 Suppose we have 10 college football teams X1 to X10. We want to cluster them into 2 groups. For each football team, we have two features: One is # wins in Season 2016, and the other is # wins in Season 2017.

Team	# wins in Season 2016 (x-axis)	# wins in Season 2017 (y-axis)
X1	3	5
X2	3	4
X3	2	8
X4	2	3
X5	6	2
X6	6	4
X7	7	3
X8	7	4
X9	8	5
X10	7	6

(1) Initialize with two centroids, (4, 6) and (5, 4). Use Manhattan distance as the distance metric. First, perform one iteration of the K-means algorithm and report the coordinates of the resulting centroids. Second, please use K-Means to find two clusters.

Ans)

Manhattan Distance = $|x_2 - x_1| + |y_2 - y_1|$

Centroid is calculated by average of datapoints belonging to same clusters.

C1 = (4,6)

C2 = (5,4)

Iteration 1:

Teams	Distances		Cluster
	C1 (4,6)	C2 (5,4)	
X1(3,5)	2	3	C1
X2(3,4)	3	2	C2
X3(2,8)	4	7	C1
X4(2,3)	5	4	C2
X5(6,2)	6	3	C2
X6(6,4)	4	1	C2
X7(7,3)	6	3	C2
X8(7,4)	5	2	C2
X9(8,5)	5	4	C2
X10(7,6)	3	4	C1

Centroids after Iteration 1:

C1 = (4,6.33)

C2 = (5.57,3.57)

Iteration 2:

Teams	Distances		Cluster
	C1 (4,6.33)	C2 (5.57,3.57)	
X1(3,5)	2.33	4	C1
X2(3,4)	3.33	3	C2
X3(2,8)	3.67	8	C1
X4(2,3)	5.33	4.14	C2
X5(6,2)	6.33	2	C2
X6(6,4)	4.33	0.86	C2
X7(7,3)	6.33	2	C2
X8(7,4)	5.33	1.86	C2
X9(8,5)	5.33	3.86	C2
X10(7,6)	3.33	3.86	C1

Centroids after iteration 2:

C1 = (4,6.33)

C2 = (5.57,3.57)

We can observe that the Centroids after Iteration 1 and Iteration 2 are same. So don't need to do iteration again. Two Clusters are as follows:

Cluster 1 -> X1(3,5), X3(2,8), X10(7,6)

Cluster 2 -> X2(3,4), X4(2,3), X5(6,2), X6(6,4), X7(7,3), X8(7,4), X9(8,5)

(2) Initialize with two centroids, (4, 6) and (5, 4). Use Euclidean distance as the distance metric. First, perform one iteration of the K-means algorithm and report the coordinates of the resulting centroids. Second, please use K-Means to find two clusters.

Ans)

Euclidean Distance = $((x_2 - x_1)^2 + (y_2 - y_1)^2)^{\frac{1}{2}}$

Centroid is calculated by average of datapoints belonging to same clusters.

C1 = (4,6)

C2 = (5,4)

Iteration 1:

Teams	Distances		Cluster
	C1 (4,6)	C2 (5,4)	
X1(3,5)	1.4142	2.2361	C1
X2(3,4)	2.2361	2.0000	C2
X3(2,8)	2.8284	5.0000	C1
X4(2,3)	3.6056	3.1623	C2
X5(6,2)	4.4721	2.2361	C2
X6(6,4)	2.8284	1.0000	C2
X7(7,3)	4.2426	2.2361	C2
X8(7,4)	3.6056	2.0000	C2
X9(8,5)	4.1231	3.1623	C2
X10(7,6)	3	2.828427	C2

Centroids after iteration 1:

C1 = (2.5,6.5)

C2 = (5.75,3.875)

Iteration 2:

Teams	Distances		Cluster
	C1 (2.5,6.5)	C2 (5.75,3.875)	
X1(3,5)	1.58	2.97	C1
X2(3,4)	2.55	2.75	C1
X3(2,8)	1.58	5.57	C1
X4(2,3)	3.54	3.85	C1
X5(6,2)	5.70	1.89	C2
X6(6,4)	4.30	0.28	C2
X7(7,3)	5.70	1.53	C2
X8(7,4)	5.15	1.26	C2
X9(8,5)	5.70	2.52	C2
X10(7,6)	4.53	2.47	C2

Centroids after iteration 2:

C1 = (2.5,5)

C2 = (6.83,4)

Iteration 3:

Teams	Distances		Cluster
	C1 (2.5,5)	C2 (6.83,4)	
X1(3,5)	0.50	3.96	C1
X2(3,4)	1.12	3.83	C1
X3(2,8)	3.04	6.27	C1
X4(2,3)	2.06	4.93	C1
X5(6,2)	4.61	2.17	C2
X6(6,4)	3.64	0.83	C2
X7(7,3)	4.92	1.01	C2
X8(7,4)	4.61	0.17	C2
X9(8,5)	5.50	1.54	C2
X10(7,6)	4.61	2.01	C2

Centroids after iteration 3:

C1 = (2.5,5)

C2 = (6.83,4)

We can observe that the Centroids after Iteration 2 and Iteration 3 are same. So don't need to do iteration again. Two Clusters are as follows:

Cluster 1 -> X1(3,5), X3(2,8), X2(3,4), X4(2,3)

Cluster 2 -> X5(6,2), X6(6,4), X7(7,3), X8(7,4), X9(8,5), X10(7,6)

(3) Initialize with two centroids, (3, 3) and (8, 3). Use Manhattan distance as the distance metric. First, perform one iteration of the K-means algorithm and report the coordinates of the resulting centroids. Second, please use K-Means to find two clusters.

Ans)

Manhattan Distance = $|x_2 - x_1| + |y_2 - y_1|$

Centroid is calculated by average of datapoints belonging to same clusters.

C1 = (3,3)

C2 = (8,3)

Iteration 1:

Teams	Distances		Cluster
	C1 (3,3)	C2 (8,3)	
X1(3,5)	2	7	C1
X2(3,4)	1	6	C1
X3(2,8)	6	11	C1
X4(2,3)	1	6	C1
X5(6,2)	4	3	C2
X6(6,4)	4	3	C2
X7(7,3)	4	1	C2
X8(7,4)	5	2	C2
X9(8,5)	7	2	C2
X10(7,6)	7	4	C2

Centroids after Iteration 1:

C1 = (2.5,5)

C2 = (6.83,4)

Iteration 2:

Teams	Distances		Cluster
	C1 (2.5,5)	C2 (6.83,4)	
X1(3,5)	0.5	4.83	C1
X2(3,4)	1.5	3.83	C1
X3(2,8)	3.5	8.83	C1
X4(2,3)	2.5	5.83	C1
X5(6,2)	6.5	2.83	C2
X6(6,4)	4.5	0.83	C2
X7(7,3)	6.5	1.17	C2
X8(7,4)	5.5	0.17	C2
X9(8,5)	5.5	2.17	C2
X10(7,6)	5.5	2.17	C2

Centroids after iteration 2:

C1 = (2.5,5)

C2 = (6.83,4)

We can observe that the Centroids after Iteration 1 and Iteration 2 are same. So don't need to do iteration again. Two Clusters are as follows:

Cluster 1 -> X1(3,5), X2(3,4), X3(2,8), X4(2,3)

Cluster 2 -> X5(6,2), X6(6,4), X7(7,3), X8(7,4), X9(8,5), X10(7,6)

(4) Initialize with two centroids, (3, 2) and (4, 8). Use Manhattan distance as the distance metric. First, perform one iteration of the K-means algorithm and report the coordinates of the resulting centroids. Second, please use K-Means to find two clusters.

Ans)

Manhattan Distance = $|x_2 - x_1| + |y_2 - y_1|$

Centroid is calculated by average of datapoints belonging to same clusters.

C1 = (3,2)

C2 = (4,8)

Iteration 1:

Teams	Distances		Cluster
	C1 (3,2)	C2 (4,8)	
X1(3,5)	3	4	C1
X2(3,4)	2	5	C1
X3(2,8)	7	2	C2
X4(2,3)	2	7	C1
X5(6,2)	3	8	C1
X6(6,4)	5	6	C1
X7(7,3)	5	8	C1
X8(7,4)	6	7	C1
X9(8,5)	8	7	C2
X10(7,6)	8	5	C2

Centroids after Iteration 1:

C1 = (4.86,3.57)

C2 = (5.67,6.33)

Iteration 2:

Teams	Distances		Cluster
	C1 (4.86,3.57)	C2 (5.67,6.33)	
X1(3,5)	3.29	4	C1
X2(3,4)	2.29	5	C1
X3(2,8)	7.29	5.34	C2
X4(2,3)	3.43	7	C1
X5(6,2)	2.71	4.66	C1
X6(6,4)	1.57	2.66	C1
X7(7,3)	2.71	4.66	C1
X8(7,4)	2.57	3.66	C1
X9(8,5)	4.57	3.66	C2
X10(7,6)	4.57	1.66	C2

Centroids after iteration 2:

C1 = (4.86,3.57)

C2 = (5.67,6.33)

We can observe that the Centroids after Iteration 1 and Iteration 2 are same. So don't need to do iteration again. Two Clusters are as follows:

Cluster 1 -> X1(3,5), X2(3,4), X4(2,3), X5(6,2), X6(6,4), X7(7,3), X8(7,4)

Cluster 2 -> X3(2,8), X9(8,5), X10(7,6)

Task 2 K-Means Clustering with Real World Dataset

First, download a simulated dataset: hw4_kmeans_data.zip from Modules->Datsets. Then, implement the K-means algorithm **from scratch**. K-means algorithm computes the distance of a given data point pair. Replace the distance computation function with Euclidean distance, 1-Cosine similarity, and 1 – the **Generalized** Jarcard similarity (refer to: https://www.itl.nist.gov/div898/software/dataplot/refman2/auxillar/jaccard.htm).

Q1: Run K-means clustering with Euclidean, Cosine and Jarcard similarity. Specify K= the number of categorical values of y (the number of classifications). Compare the SSEs of Euclidean-K-means, Cosine-K-means, Jaccard-K-means. Which method is better?

Ans)

All the SSE are almost same. So it is difficult to comment.

Q2: Compare the accuracies of Euclidean-K-means Cosine-K-means, Jarcard-K-means. First, label each cluster using the majority vote label of the data points in that cluster. Later, compute the predictive accuracy of Euclidean-K-means, Cosine-K-means, Jarcard-K-means. Which metric is better?

Ans) Overall accuracy of all 3 of them is good. Accuracy of Jaccard is Better.

Q3: Set up the same stop criteria: "when there is no change in centroid position OR when the SSE value increases in the next iteration OR when the maximum preset value (e.g., 500, you can set the preset value by yourself) of iteration is complete", for Euclidean-K-means, Cosine-Kmeans, Jarcard-K-means. Which method requires more iterations and times to converge?

Ans) Euclidean distance requires more iterations to converge.

Q4: Compare the SSEs of Euclidean-K-means Cosine-K-means, Jarcard-K-means with respect to the following three terminating conditions:

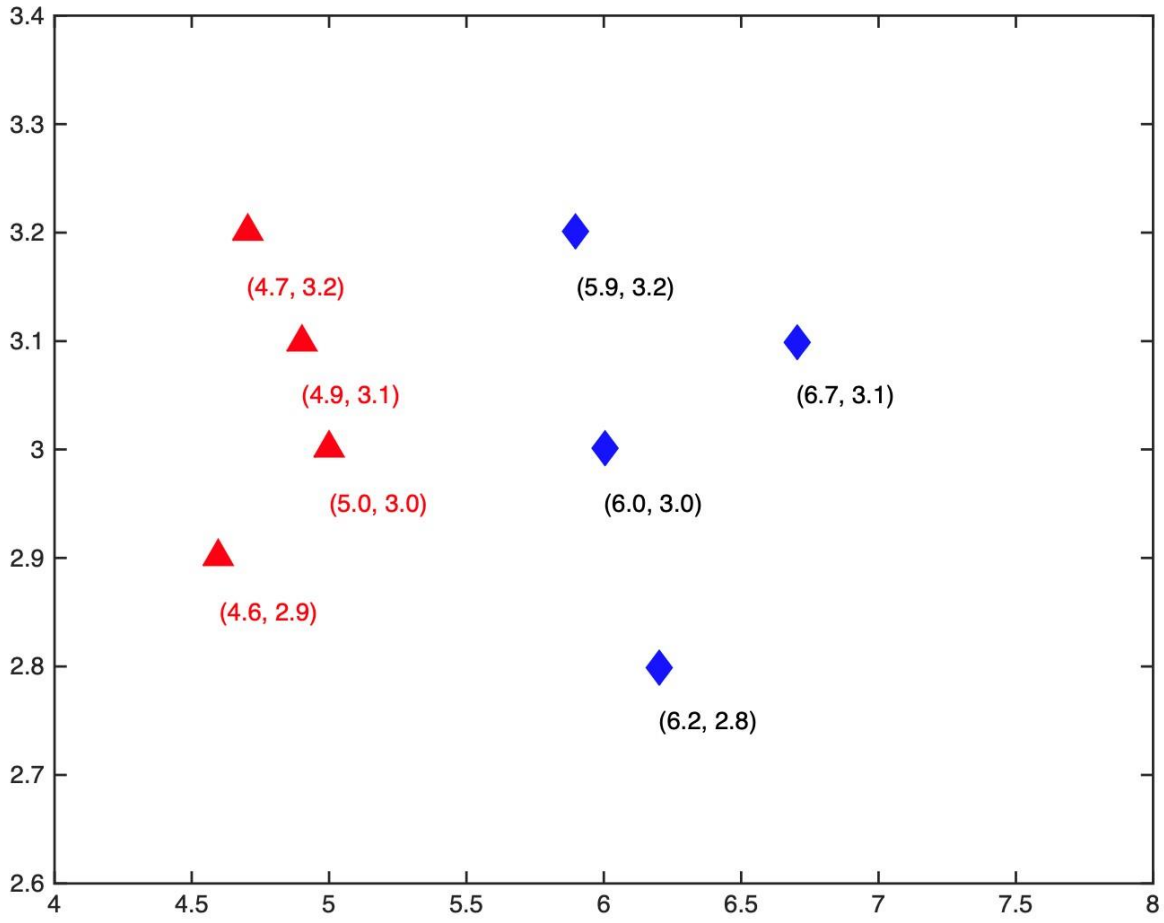
- when there is no change in centroid position
- when the SSE value increases in the next iteration
- when the maximum preset value (e.g., 100) of iteration is complete

Q5: What are your summary observations or takeaways based on your algorithmic analysis?

Ans)

1) The accuracies are different but there is not a lot difference between the three.
2) The SSE of Euclidean, Jaccard and Cosine does not have a lot of difference. But one of them performs well.

Task 3, There are two clusters A (red) and B (blue), each has four members and plotted in Figure. The coordinates of each member are labeled in the figure. Compute the distance between two clusters using Euclidean distance.



A. What is the distance between the two farthest members? (round to four decimal places here, and next 2 problems);

B. What is the distance between the two closest members?

C. What is the average distance between all pairs?

D, Discuss which distance (A, B, C) is more robust to noises in this case?

Ans)

Euclidean Distance = $((x_2 - x_1)^2 + (y_2 - y_1)^2)^{\frac{1}{2}}$

Euclidean Distance between all points of 2 clusters				
	A(5.9,3.2)	**B(6.7,3.1)**	**C(6,3)**	**D(6.2,2.8)**
E(4.7,3.2)	1.2000	2.0025	1.3153	1.5524
F(4.9,3.1)	1.0050	1.8000	1.1045	1.3342
G(5,3)	0.9220	1.7029	1.0000	1.2166
H(4.6,2.9)	1.3342	2.1095	1.4036	1.6031

A) Distance between farthest members:
B(6.7,3.1) & H(4.6,2.9) = 2.1095

B) Distance between two Closest members:
A(5.9,3.2) & G(5,3) = 0.9220

C) Average distance between all pairs:
= 1.4129

D) The Average distance between all pairs is more robust as it won't be changed a lot, even it there were outliers in the data.

Additional Questions:

- Approximately how many hours did you spend on this assignment?

Ans) I spent around 25-30 hours for this assignment.

- Which aspects of this assignment did you find most challenging? Were there any significant stumbling blocks?

Ans)

- Which aspects of this assignment did you like? Is there anything you would have changed?

Ans)

Please submit a PDF report. In your report, please answer each question with your explanations, plots, results in brief. DO NOT paste your code or snapshot into the PDF. At the end of your PDF, please include

a website address (e.g., Github, Dropbox, OneDrive, GoogleDrive) that can allow the TA to read your code.