MIS637 - Data Analytics and Machine Learning
Assignment 5
Komal Wavhal (20034443)

Question:

Chapter 8, page 162, problems 6 (2004 edition)

Suppose that we have the following data:
a b c d e f g h i j
(2,0) (1,2) (2,2) (3,2) (2,3) (3,3) (2,4) (3,4) (4,4) (3,5)

Identify the cluster by applying the k-means algorithm, with k = 2. Try using initial cluster centers as far apart as possible.

Solution: **k-means clustering algorithm** in detail on your dataset with **k = 2**, starting with **initial centers as far apart as possible**.

We have 10 points labeled a through j

a: (2,0)

b: (1,2)

c: (2,2)

d: (3,2)

e: (2,3)

f: (3,3)

g: (2,4)

h: (3,4)

i: (4,4)

j: (3,5)

Choose Initial Cluster Centers

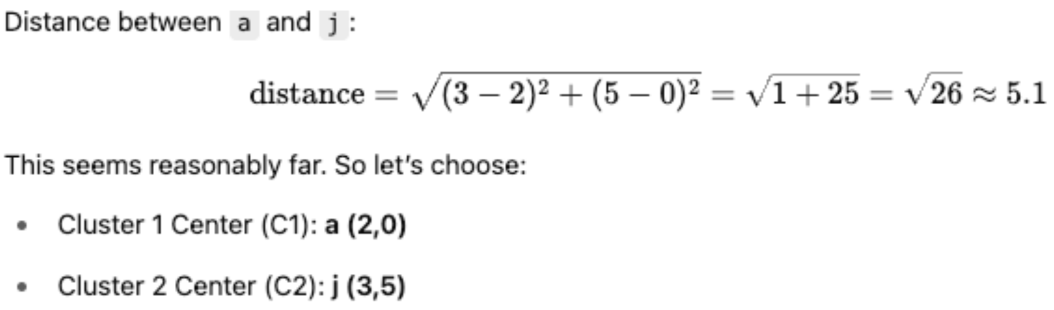
We need to select two initial cluster centers, preferably far apart.

Let's calculate Euclidean distances between points to find the two most distant ones.

Some obvious candidates by observation:

- **Point a = (2,0) is the lowest point.**

- **Point j = (3,5) is the highest in y-coordinate.**



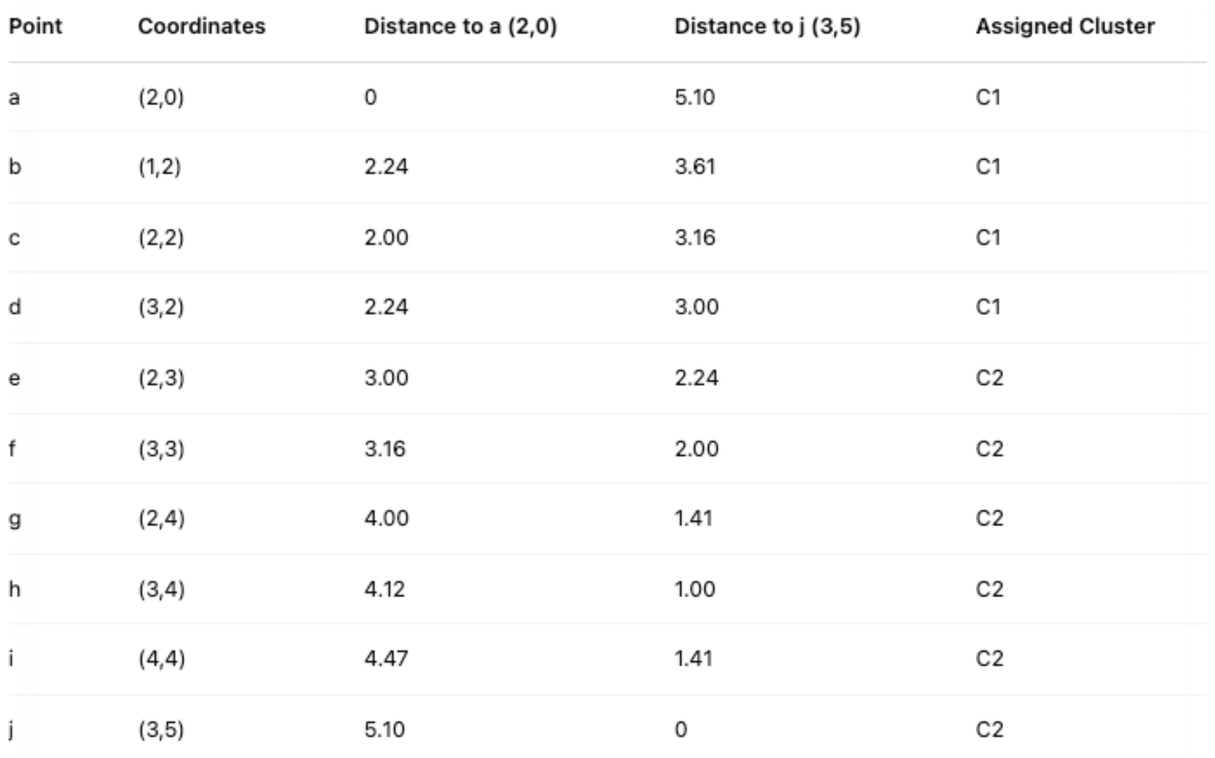
Distance between `a` and `j`:

$$\text{distance} = \sqrt{(3-2)^2 + (5-0)^2} = \sqrt{1+25} = \sqrt{26} \approx 5.1$$

This seems reasonably far. So let's choose:

- Cluster 1 Center (C1): **a (2,0)**
- Cluster 2 Center (C2): **j (3,5)**

Assign Points to the Nearest Cluster Center: We'll compute the Euclidean distance of each point to the two centers and assign it to the nearest one.



Point	Coordinates	Distance to a (2,0)	Distance to j (3,5)	Assigned Cluster
a	(2,0)	0	5.10	C1
b	(1,2)	2.24	3.61	C1
c	(2,2)	2.00	3.16	C1
d	(3,2)	2.24	3.00	C1
e	(2,3)	3.00	2.24	C2
f	(3,3)	3.16	2.00	C2
g	(2,4)	4.00	1.41	C2
h	(3,4)	4.12	1.00	C2
i	(4,4)	4.47	1.41	C2
j	(3,5)	5.10	0	C2

Cluster 1: a, b, c, d
Cluster 2: e, f, g, h, i, j

Recalculate Cluster Centers: Now that we've grouped the points, let's compute the centroid (mean of x and y) of each cluster.

Cluster 1: Points a, b, c, d

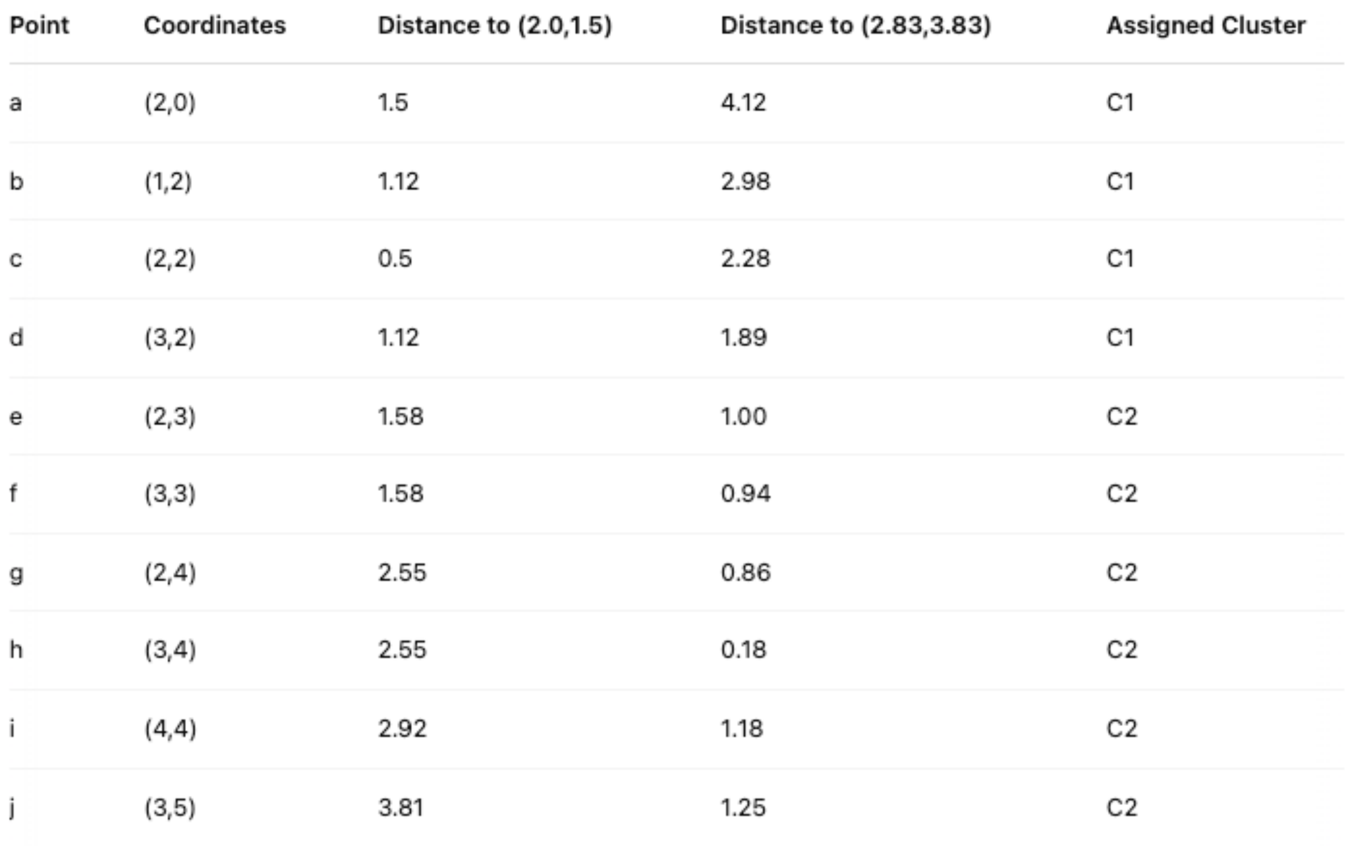
- **Coordinates: (2,0), (1,2), (2,2), (3,2)**
- **Mean x: (2 + 1 + 2 + 3)/4 = 8/4 = 2.0**
- **Mean y: (0 + 2 + 2 + 2)/4 = 6/4 = 1.5**
  New C1: (2.0, 1.5)

Cluster 2: Points e, f, g, h, i, j

- **Coordinates: (2,3), (3,3), (2,4), (3,4), (4,4), (3,5)**
- **Mean x: (2 + 3 + 2 + 3 + 4 + 3)/6 = 17/6 ≈ 2.83**
- **Mean y: (3 + 3 + 4 + 4 + 4 + 5)/6 = 23/6 ≈ 3.83**
  New C2: (2.83, 3.83)

Reassign Points to the Nearest Center (Using Updated Centers)

Now repeat the assignment step.



Point	Coordinates	Distance to (2.0,1.5)	Distance to (2.83,3.83)	Assigned Cluster
a	(2,0)	1.5	4.12	C1
b	(1,2)	1.12	2.98	C1
c	(2,2)	0.5	2.28	C1
d	(3,2)	1.12	1.89	C1
e	(2,3)	1.58	1.00	C2
f	(3,3)	1.58	0.94	C2
g	(2,4)	2.55	0.86	C2
h	(3,4)	2.55	0.18	C2
i	(4,4)	2.92	1.18	C2
j	(3,5)	3.81	1.25	C2

No changes in assignment — clusters have stabilized!

Final Clusters

Cluster 1:

- **Points: a, b, c, d**
- **Center: (2.0, 1.5)**

Cluster 2:

- **Points: e, f, g, h, i, j**
- **Center: (2.83, 3.83)**

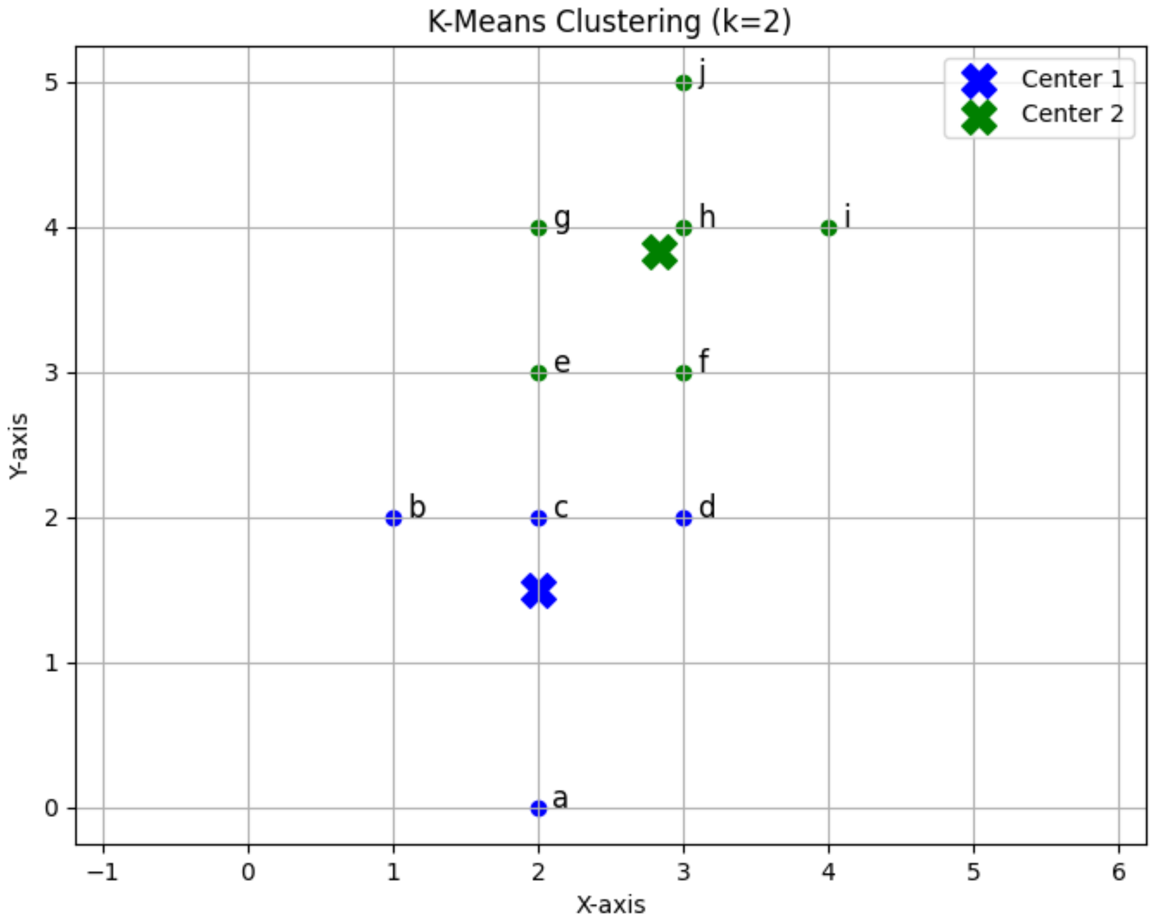
1. **We applied k-means with k = 2 and initialized using two farthest points: a and j.**

2. **After 2 iterations, the clusters stabilized.**
3. **The algorithm separated points into a lower cluster (Cluster 1) and an upper cluster (Cluster 2).**

- **Below Python script will generate a scatter plot showing:**
  A. **Points in Cluster 1 (blue)**
  B. **Points in Cluster 2 (green)**
  C. **Cluster centers as large X markers**

```
import matplotlib.pyplot as plt
# Define the data points
points = {
   'a': (2, 0), 'b': (1, 2), 'c': (2, 2), 'd': (3, 2),
   'e': (2, 3), 'f': (3, 3), 'g': (2, 4), 'h': (3, 4),
   'i': (4, 4), 'j': (3, 5)
}
# Define clusters after convergence
cluster_1 = ['a', 'b', 'c', 'd']
cluster_2 = ['e', 'f', 'g', 'h', 'i', 'j']


# Cluster centers
center_1 = (2.0, 1.5)
center_2 = (2.83, 3.83)


# Plotting
plt.figure(figsize=(8, 6))
```

```
# Plot points for Cluster 1
for label in cluster_1:
    x, y = points[label]
    plt.scatter(x, y, color='blue')
    plt.text(x + 0.1, y, label, fontsize=12)

# Plot points for Cluster 2
for label in cluster_2:
    x, y = points[label]
    plt.scatter(x, y, color='green')
    plt.text(x + 0.1, y, label, fontsize=12)

# Plot centers
plt.scatter(*center_1, color='blue', marker='X', s=200, label='Center 1')
plt.scatter(*center_2, color='green', marker='X', s=200, label='Center 2')

plt.title("K-Means Clustering (k=2)")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.grid(True)
plt.legend()
plt.axis('equal')
plt.show()
```


K-Means Clustering (k=2)
Center 1
Center 2
j
g
h
i
e
f
b
c
d
a
Y-axis
X-axis
5
4
3
2
1
0
−1
0
1
2
3
4
5
6