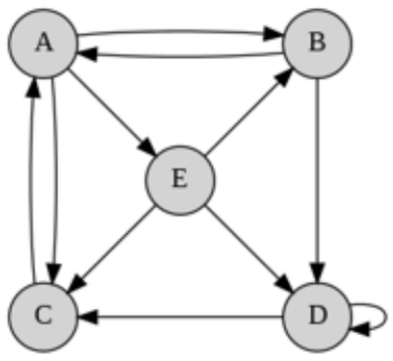
DAMG 60 Written Assignment 3

Panagiotis Paltsokas
std163861

TOPIC 2 — PAGERANK

a) (5 points) Compute the transition probability matrix **M** of going from each node to any other adjacent node, such that the probabilities sum up to one in a column-wise way. Fill in the placeholders (?) with the appropriate values in the transition probability **M** matrix shown below.



Transition probability matrix M

	A	B	C	D	E
A	0	1/2	1	0	0
B	1/3	0	0	0	1/3
C	1/3	0	0	1/2	1/3
D	0	1/2	0	1/2	1/3
E	1/3	0	0	0	0
COL SUM	1	1	1	1	1

(+)

b) (4 points) Assuming that all nodes (A – E) share the same initial probabilities and $\beta = 0.85$, calculate by hand the values of the column vector v' after the **2nd Iteration** when the PageRank algorithm with teleportation is executed. Fill in the placeholders (?) in the vector v' below with the correct values. During all calculations truncate the intermediate and final results to 3 decimal digits.

$\beta = 0.85 \rightarrow 85\%$ follow links, 15% teleports

Initial probability vector $v_0' = [1/5, 1/5, 1/5, 1/5, 1/5]^T$

$$v' = \beta \cdot M \cdot v + (1-\beta)\cdot \frac{e}{5} = 0.85 \cdot Mv + \frac{0.15}{5} = 0.85 \cdot Mv + 0.03 \cdot e$$

Iteration 1

Transition probability matrix M

	A	B	C	D	E
A	0	1/2	1	0	0
B	1/3	0	0	0	1/3
C	1/3	0	0	1/2	1/3
D	0	1/2	0	1/2	1/3
E	1/3	0	0	0	0

$M \cdot V_0' = \begin{bmatrix} \cdot \end{bmatrix}_{5\times5} \times \begin{bmatrix} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{bmatrix}_{5\times1} = \begin{bmatrix} 0.3 \\ 0.1\overline{3} \\ 0.23\overline{3} \\ 0.26\overline{6} \\ 0.06\overline{6} \end{bmatrix}_{5\times1}$

$V_1' = M \cdot V_0 + \begin{bmatrix} 0.03 \\ 0.03 \\ 0.03 \\ 0.03 \\ 0.03 \end{bmatrix} = \begin{bmatrix} 0.285 \\ 0.143 \\ 0.228 \\ 0.256 \\ 0.086 \end{bmatrix}$ ← truncated

Iteration 2

Transition probability matrix M

	A	B	C	D	E
A	0	1/2	1	0	0
B	1/3	0	0	0	1/3
C	1/3	0	0	1/2	1/3
D	0	1/2	0	1/2	1/3
E	1/3	0	0	0	0

$M \cdot V_1' = \begin{bmatrix} \cdot \end{bmatrix}_{5\times5} \times \begin{bmatrix} 0.285 \\ 0.143 \\ 0.228 \\ 0.256 \\ 0.086 \end{bmatrix}_{5\times1} = \begin{bmatrix} 0.2995 \\ 0.123\overline{6} \\ 0.251\overline{6} \\ 0.228\overline{76} \\ 0.095 \end{bmatrix}_{5\times1}$

$V_2' = 0.85 \cdot MV_1' + \begin{bmatrix} 0.03 \\ 0.03 \\ 0.03 \\ 0.03 \\ 0.03 \end{bmatrix} = \begin{bmatrix} 0.284 \\ 0.135 \\ 0.243 \\ 0.223 \\ 0.770 \end{bmatrix}$ ← truncated

c) (4 points) Compute the topic-sensitive PageRank for the initial graph, assuming the teleport set is $S = \{C, D\}$ and $\beta = 0.8$. All nodes (A – E) share the same initial probabilities. Calculate by hand the values of the column vector v' after the **2nd iteration** and fill in the placeholders (?) with the correct values. During all calculations truncate the intermediate and final results to 3 decimal digits. (do not round up or down the numbers)

$\beta = 0.8 \rightarrow 80\%$ follows links

20% teleports

$$v' = 0.8 \cdot M \cdot v + 0.2 \cdot \begin{bmatrix} 0 \\ 0 \\ 1/2 \\ 1/2 \\ 0 \end{bmatrix} = 0.8 \cdot M \cdot v + \begin{bmatrix} 0 \\ 0 \\ 0.1 \\ 0.1 \\ 0 \end{bmatrix}$$

Topic-Sensitive Pagerank

teleportation restricted to C and D

NEW initial probability vector $v_0 = [0, 0, 1/2, 1/2, 0]^T$

Iteration 1

Transition probability matrix M

	A	B	C	D	E
A	0	1/2	1	0	0
B	1/3	0	0	0	1/3
C	1/3	0	0	1/2	1/3
D	0	1/2	0	1/2	1/3
E	1/3	0	0	0	0

$\begin{bmatrix} \cdot \end{bmatrix}_{5\times5} \times \begin{bmatrix} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{bmatrix}_{5\times1} = \begin{bmatrix} 0.3 \\ 0.13\overline{3} \\ 0.23\overline{3} \\ 0.26\overline{6} \\ 0.06\overline{6} \end{bmatrix}_{5\times1}$

$V_1' = 0.8 \cdot \begin{bmatrix} 0.3 \\ 0.133 \\ 0.233 \\ 0.266 \\ 0.066 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.1 \\ 0.1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.24 \\ 0.106 \\ 0.286 \\ 0.313 \\ 0.053 \end{bmatrix}$

Iteration 2

Transition probability matrix M

	A	B	C	D	E
A	0	1/2	1	0	0
B	1/3	0	0	0	1/3
C	1/3	0	0	1/2	1/3
D	0	1/2	0	1/2	1/3
E	1/3	0	0	0	0

$\begin{bmatrix} \cdot \end{bmatrix}_{5\times5} \times \begin{bmatrix} 0.24 \\ 0.106 \\ 0.286 \\ 0.312 \\ 0.052 \end{bmatrix}_{5\times1} = \begin{bmatrix} 0.339 \\ 0.09\overline{76} \\ 0.25\overline{416} \\ 0.22\overline{716} \\ 0.08 \end{bmatrix}_{5\times1}$

$V_2' = 0.8 \cdot \begin{bmatrix} 0.339 \\ 0.097 \\ 0.253 \\ 0.226 \\ 0.08 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.1 \\ 0.1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.271 \\ 0.078 \\ 0.303 \\ 0.281 \\ 0.064 \end{bmatrix}$

d) (2 points) Which node presents the highest rank in each implementation?

Answer:

PageRank with teleportation: Node A presents the highest rank (0.284)

Topic Sensitive PageRank: Node C presents the highest rank (0.303)

PAGERANK with teleportation

$V_2' = \begin{bmatrix} 0.284 \\ 0.135 \\ 0.243 \\ 0.223 \\ 0.770 \end{bmatrix}$ ← node A

TOPIC-SENSITIVE PAGERANK

$V_2' = \begin{bmatrix} 0.271 \\ 0.077 \\ 0.303 \\ 0.281 \\ 0.064 \end{bmatrix}$ ← node C

Topic 2 PageRank

(b)

[1]:
```
import numpy as np
import pandas as pd
```

[2]:
```
def truncate(matrix, decimals=3):
    return np.floor(matrix * 10**decimals) / 10**decimals
```

[3]:
```
m1= np.array([0, 1/3, 1/3, 0 , 1/3])
m2 = np.array([1/2,0,0,1/2,0])
m3 = np.array([1,0,0,0,0])
m4 = np.array([0,0,1/2,1/2,0])
m5 = np.array([0,1/3,1/3,1/3,0])
```

[4]:
```
v0=np.array([0.2, 0.2, 0.2, 0.2, 0.2])
```

[5]:
```
M = np.column_stack([m1, m2, m3, m4, m5])
M
```

[5]:
```
array([[0.        , 0.5       , 1.        , 0.        , 0.        ],
       [0.33333333, 0.        , 0.        , 0.        , 0.33333333],
       [0.33333333, 0.        , 0.        , 0.5       , 0.33333333],
       [0.        , 0.5       , 0.        , 0.5       , 0.33333333],
       [0.33333333, 0.        , 0.        , 0.        , 0.        ]])
```

[6]:
```
Mv0=M @ v0
Mv0
```

[6]:
```
array([0.3       , 0.13333333, 0.23333333, 0.26666667, 0.06666667])
```

[7]:
```
v1=(0.85*Mv0)+np.array([0.03,0.03,0.03,0.03,0.03])
v1_truncated=truncate(v1)
v1_truncated
```

[7]:
```
array([0.285, 0.143, 0.228, 0.256, 0.086])
```

[8]:
```
Mv1 = M @ v1_truncated
Mv1
```

[8]:
```
array([0.2995    , 0.12366667, 0.25166667, 0.22816667, 0.095     ])
```

[9]:
```
v2=0.85*Mv1+np.array([0.03,0.03,0.03,0.03,0.03])
v2_truncated = truncate(v2)
v2_truncated
```

[9]:
```
array([0.284, 0.135, 0.243, 0.223, 0.11 ])
```

(c)

[10]:
```
t = 0.2*np.array([0, 0, 1/2, 1/2, 0])
t
```

[10]:
```
array([0. , 0. , 0.1, 0.1, 0. ])
```

[11]:
```
b=0.8
```

[12]:
```
Mv0 = M @ v0
Mv0
```

[12]:
```
array([0.3       , 0.13333333, 0.23333333, 0.26666667, 0.06666667])
```

[13]:
```
v1=b*Mv0+t
v1_truncated = truncate(v1)
v1_truncated
```

[13]:
```
array([0.24 , 0.106, 0.286, 0.313, 0.053])
```

[14]:
```
Mv1 = M @ v1_truncated
Mv1
```

[14]:
```
array([0.339     , 0.09766667, 0.25416667, 0.22716667, 0.08      ])
```

[15]:
```
v2 = b*Mv1+t
v2
```

[15]:
```
array([0.2712    , 0.07813333, 0.30333333, 0.28173333, 0.064     ])
```

[16]:
```
truncate(v2)
```

[16]:
```
array([0.271, 0.078, 0.303, 0.281, 0.063])
```

[17]:
```
truncate(np.array([0.2712, 0.07813333, 0.30333333, 0.28173333, 0.064]))
```

[17]:
```
array([0.271, 0.078, 0.303, 0.281, 0.064])
```

[18]:
```
truncate(v2)
```

[18]:
```
array([0.271, 0.078, 0.303, 0.281, 0.063])
```

[19]:
```
print(v2)
print(v2[4])
```

```
[0.2712     0.07813333 0.30333333 0.28173333 0.064     ]
0.06399999999999999
```

Interestingly enough, the 5th element in the v2 array, is not 0.064, but 0.0639999999999999. Therefore, when we truncate, it should actually be 0.063.

TOPIC 3

Topic 3: PCY Algorithm

(15 total points) In the following table, a collection of 15 transactions is given. Each transaction consists of a set of items generated from a total of nine (9) distinct items, enumerated from 1 to 9. The minimum support threshold is set to 6.

Transaction	Items
T1	{1, 2, 5}
T2	{2, 3, 6}
T3	{3, 4, 5, 7}
T4	{1, 3, 5}
T5	{2, 4, 7, 8}

Transaction	Items
T6	{1, 5, 6, 9}
T7	{2, 3, 4}
T8	{2, 4, 5, 7}
T9	{3, 5, 7, 8}
T10	{2, 4, 8}

Transaction	Items
T11	{1, 2, 6, 9}
T12	{1, 3, 7}
T13	{2, 5, 8}
T14	{3, 4, 9}
T15	{4, 6, 7, 9}

a) **(2 Points)** Compute the support for each item and each pair of items and fill in the placeholders (?) with the correct values in the tables below.

Answer: count how many times they appear

Item	1	2	3	4	5	6	7	8	9
Support	5	8	7	7	7	4	6	4	4

$\Sigma = 52$

count how many times both items (i,j) appear together

Support of pairs (i, j)		j							
		2	3	4	5	6	7	8	9
i	1	2	2	0	3	2	1	0	2
	2		2	4	3	2	2	3	1
	3			3	3	1	3	1	1
	4				2	1	4	2	2
	5					1	3	2	1
	6						1	0	3
	7							2	1
	8								0

b) **(6 Points)** Using a hash table consisting of nine (9) buckets, with each pair $\{i, j\}$ hashed to bucket $(i \times j)\ mod\ 9$, calculate the total support of every bucket. In the table below, fill in the placeholders (?) with the correct values.

Answer:

Bucket	0	1	2	3	4	5	6	7	8
Support	?	?	6	?	?	7	?	4	?

generally $n \bmod 9 = 4 \quad \forall m \in \{m \in \mathbb{N}^* / m < 9\}$

$h(i,j) = (i \times j) \bmod 9$

Pair	Support	Bucket
(1,2)	2	(1·2) mod 9 = 2 mod 9 = 2
(1,3)	2	3 mod 9 = 3
(1,4)	0	4 mod 9 = 4
(1,5)	3	5
(1,6)	2	6
(1,7)	1	7
(1,8)	0	8
(1,9)	2	0

Pair	Support	Bucket
(2,3)	2	6 mod 9 = 6
(2,4)	4	8
(2,5)	3	1
(2,6)	2	3
(2,7)	2	5
(2,8)	3	7
(2,9)	1	0

Pair	Support	Bucket
(3,4)	3	12 mod 9 = 3
(3,5)	3	6
(3,6)	1	0
(3,7)	3	3
(3,8)	1	6
(3,9)	1	0
(4,5)	2	2
(4,6)	1	6

Pair	Support	Bucket
(4,7)	4	1
(4,8)	2	5
(4,9)	2	0
(5,6)	1	3
(5,7)	3	8
(5,8)	2	4
(5,9)	1	0
(6,7)	1	6

Pair	Support	Bucket
(6,8)	0	3
(6,9)	3	0
(7,8)	2	2
(7,9)	1	0
(8,9)	0	0

Bucket	Total Support
0	12
1	7
2	6
3	11
4	2
5	7
6	10
7	4
8	7

Answer:

Bucket	0	1	2	3	4	5	6	7	8
Support	12	7	6	11	2	7	10	4	7

c) **(3 points)** Which buckets are frequent? Replace the placeholders (**?**) with the buckets that are frequent.

Answer:

Frequent buckets are: 0, ?, ?, ?, ?, ?, ?.

minimum support threshold = 6 therefore, frequent buckets: ✓ 0, ✓ 1, ✓ 2, ✓ 3, ~~4~~, ✓ 5, ✓ 6, ~~7~~, ✓ 8

(d) (4 points) Which pairs are finally counted in the 2nd pass of PCY? Replace the placeholders ({?, ?}) with the correct pairs.

Answer:

Candidate pairs are: {2, 3}, {2,4}, {?, ?}, {?, ?}, {?, ?}, {?, ?}, {?, ?}, {?, ?}, {?, ?}, {?, ?}, {?, ?}

2nd pass candidates are only pairs that: a) have frequent items
b) are hashed into frequent buckets

Table #1

Item	1	2	3	4	5	6	7	8	9
Support	5	8	7	7	7	4	6	4	4

We rule out infrequent items.

We only consider pair with 2, 3, 4, 5, and 7

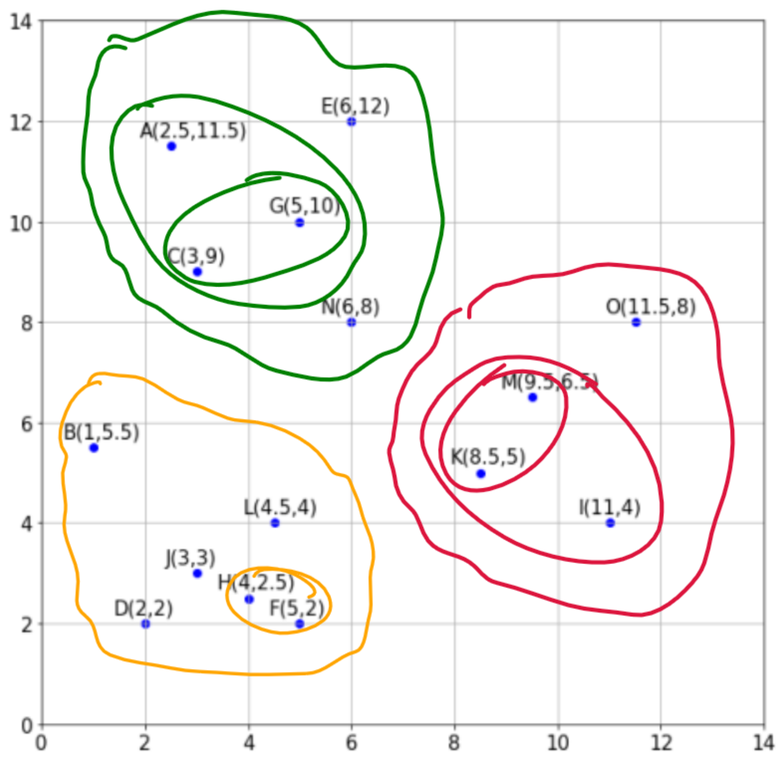
Pair	Bucket	frequent?
(2,3)	6	✓
(2,4)	8	✓
(2,5)	1	✓
(2,7)	5	✓
(3,4)	3	✓
(3,5)	6	✓
(3,7)	3	✓
(4,5)	2	✓
(4,7)	1	✓
(5,7)	8	✓

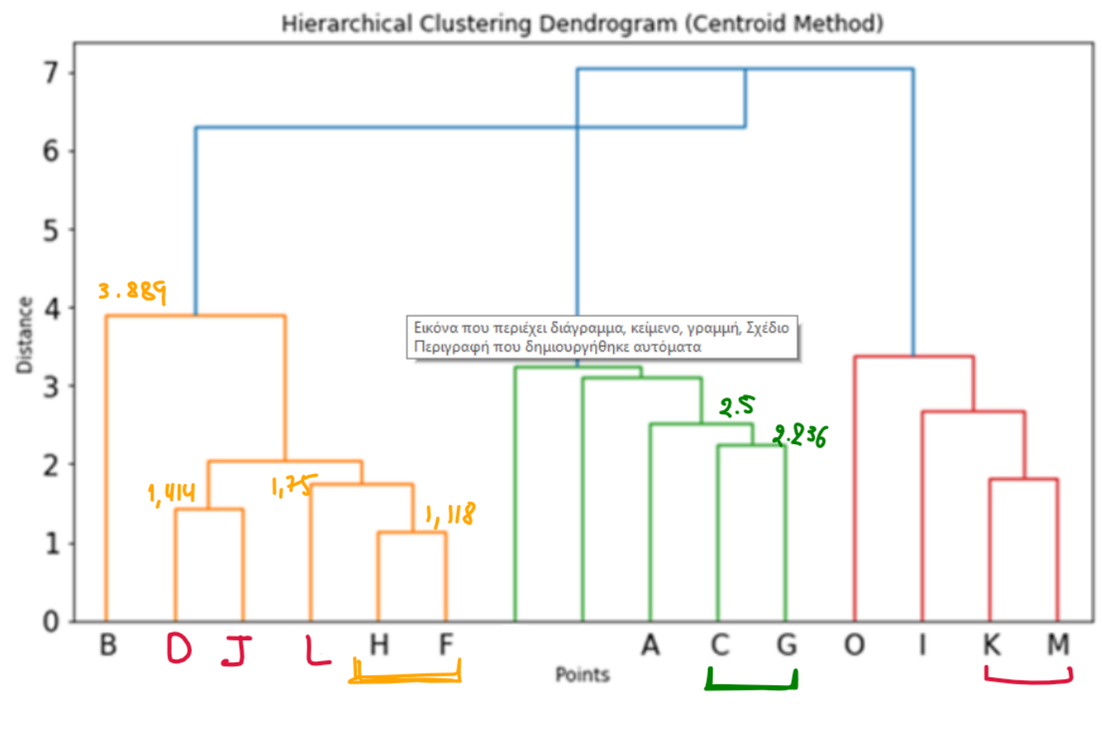
All there pairs are finally counted in the 2nd pass of PCY.

TOPIC 4 CURE ALGORITHM

Topic 4: CURE Algorithm

(15 total points) In this topic, a set of points is clustered using the CURE algorithm. Consider the following 15 points in a two-dimensional Euclidean space which will be clustered using CURE:





(a) (**5 points**) Perform hierarchical clustering of the given points, assuming clusters are represented by their centroid and at each step the clusters with the closest centroids are merged. Assume that after hierarchically clustering the points, three clusters are created. Fill in the missing points, indicated by the numerated placeholders **1?**, **2?** etc in the following dendrogram with the correct values, to specify the three formed clusters.

New centroid $(H,F) \rightarrow C_{HF}\left(\frac{4+5}{2}, \frac{2.5+2}{2}\right)$

$C_{HF}(4.5, 2.25)$

$d(L, C_{HF}) = \sqrt{(4.5-4.5)^2+(4-2.25)^2} = \sqrt{0+(1,75)^2} = 1,75$

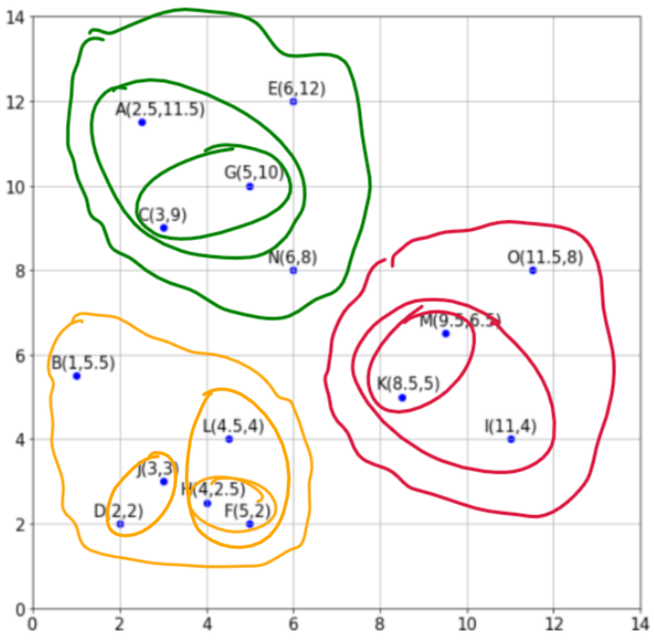
$d(H,F) = \sqrt{(5-4)^2+(2-2,5)^2} = \sqrt{1+0,25} = \sqrt{1,25} \simeq 1,118$

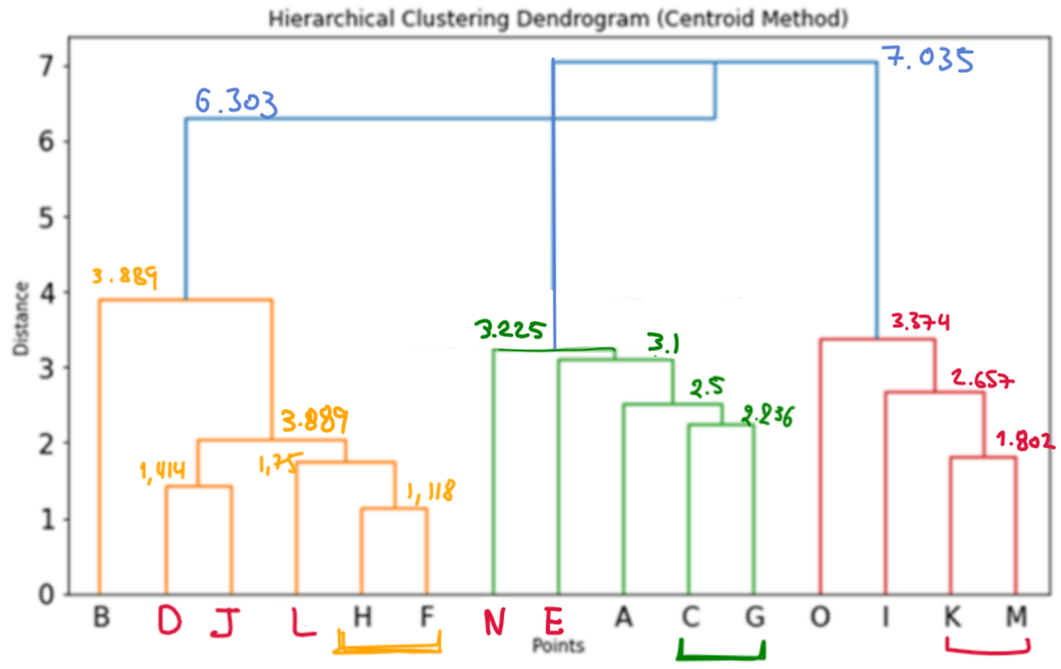
$d(D,J) = \sqrt{(3-2)^2+(3-2)^2} = \sqrt{2} \simeq 1,414$

$d(J,L) = \sqrt{(4,5-3)^2+(4-3)^2} = \sqrt{2,25+1} = \sqrt{3,25} \simeq 1,80$

$d(D,L) > d(J,L) > d(D,J)$

Therefore 1 → D 2 → J 3 → L





New Centroid (A, C, G) → $C_{ACG}\left(\frac{2.5+5+3}{3}, \frac{11.5+9+10}{3}\right)$

$C_{ACG}(3.5, 10.2)$

Therefore 4 → N

5 → E

$d(N, C_{ACG}) = \sqrt{(6-3.5)^2+(8-10.2)^2} = \sqrt{6.25+4.84} = \sqrt{11.09} \simeq 3.3$

$d(E, C_{ACG}) = \sqrt{6.25+(12-10.2)^2} = \sqrt{6.25+(1.8)^2} = \sqrt{9.49} \simeq 3$

$d(E, C_{ACG}) < d(N, C_{ACG})$

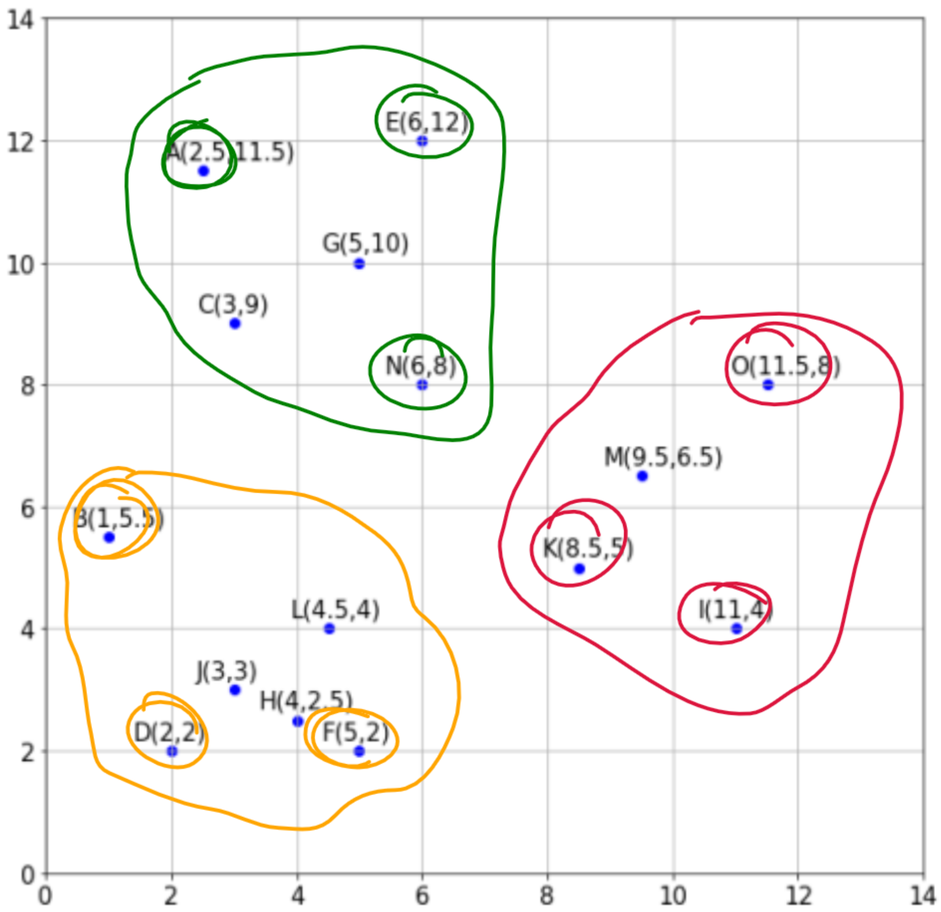
$d(C, G) = 2.24$

b) (**4 points**) Having formed three clusters, select three points from the first cluster (cluster C1) and the three points from the second cluster (cluster C2) that will be the representative points for each cluster. Fill in the placeholders (**?**) to specify the representative points. These points should be chosen to be as far from one another as possible. For cluster C1 use F as the initial random point; for cluster C2 use A as the initial random point. Assume that for cluster C3 the selected representative points are {I, K, O} (required for the next question).

Answer:

representative points for cluster C1: {F, B, D}

representative points for cluster C2: {A, N, E}



start F

$d(F,B) = 5.315$ ← dmax

$d(F,D) = 3$

$d(F,J) = 2.236$

$d(F,H) = 7.118$

$d(F,L) = 2.061$

then B

$d(B,D) = 3.64$ #3

$d(B,J) = \sout{3.201}$ too close to both

$d(B,H) = 4.243$ #1

$d(B,L) = 3.807$ #2

* $\frac{d(B,H)}{d(F,H)} = 3,795$ + $\frac{d(B,L)}{d(F,L)} = 1,847$ * $\frac{d(B,D)}{d(F,D)} = 1,213$

D is almost as far away from F as it is from B

start A

$d(A,C) = 2.549$

$d(A,G) = 2.915$

$d(A,E) = 3.536$

$d(A,N) = 4.95$ ← dmax

then N

$d(N,C) = 3.162$

$d(N,G) = \sout{2.236}$ too close to both

$d(N,E) = 4$ ← here the choice is obvious.

E is further away from A and N than any other point

(c) (**3 points**) Move each of the representative points of cluster C3 only 20% closer to the centroid of the cluster it belongs to. Give the new positions of the representative points for cluster C3 and fill in the placeholders (**?**). During all calculations truncate the intermediate and final results to 3 decimal digits (do not round up or down the numbers). The cluster centroid is computed over the selected representative points.

Answer:

Representative points for cluster C3: I(**?**, **?**), K(**?**, **?**), O(**?**, **?**)

Centroid C_3

$$x = \frac{11 + 8.5 + 11.5}{3} = \frac{31}{3} = 10.333$$

$$y = \frac{4 + 5 + 8}{3} = \frac{17}{3} = 5.666$$

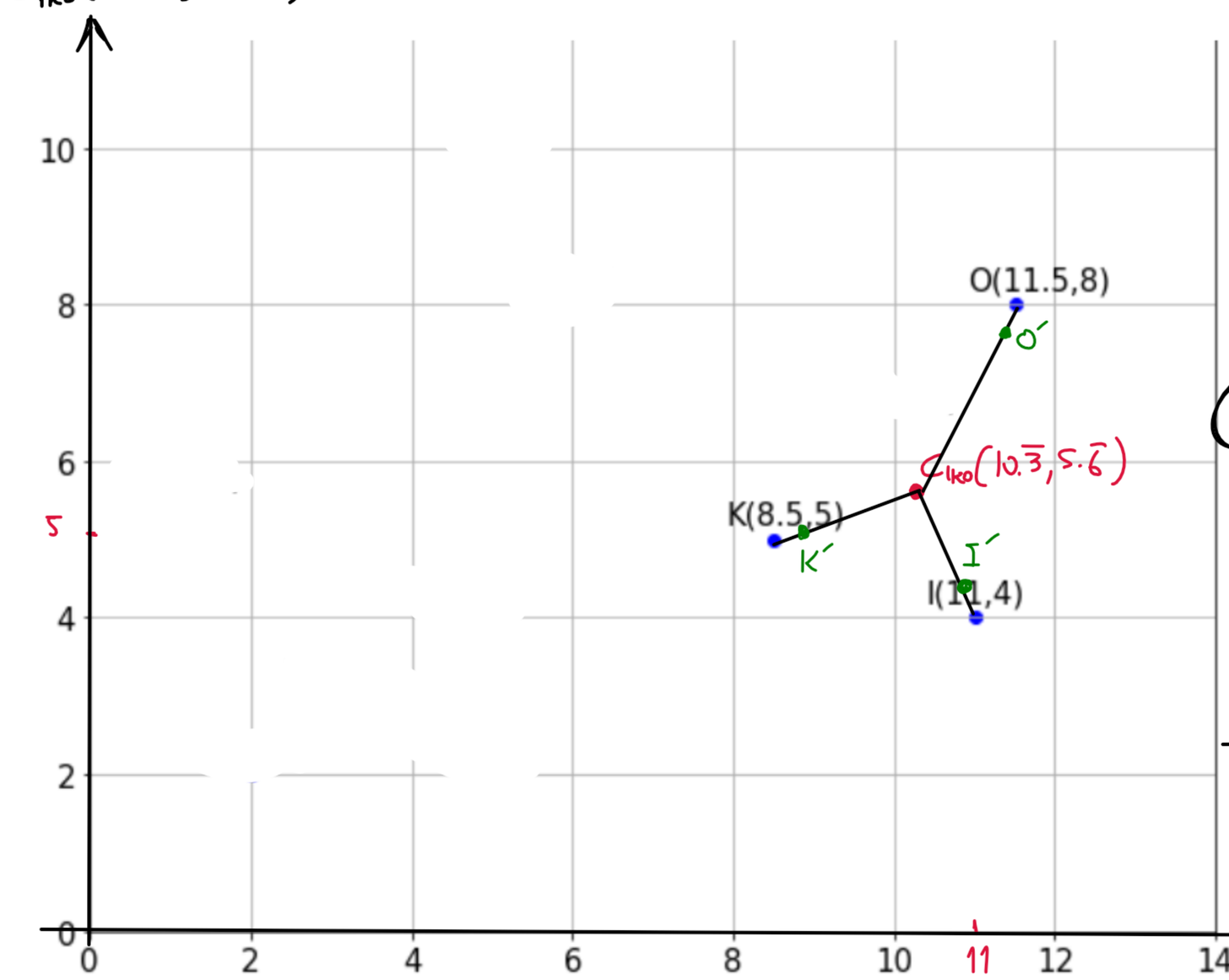
$C_{1ko}(10.333, 5.666)$

I(11, 4)

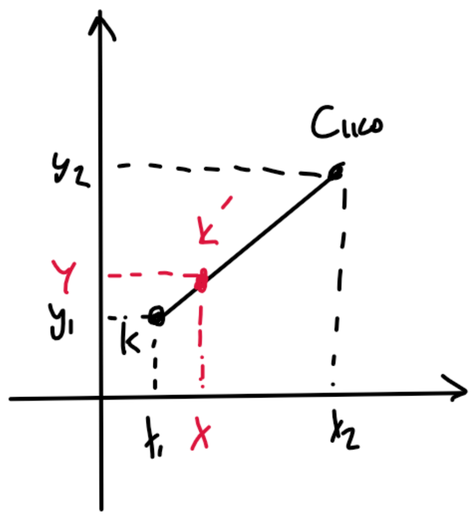
K(8.5, 5)

O(11.5, 8)

$C_{1ko}(10.333, 5.666)$



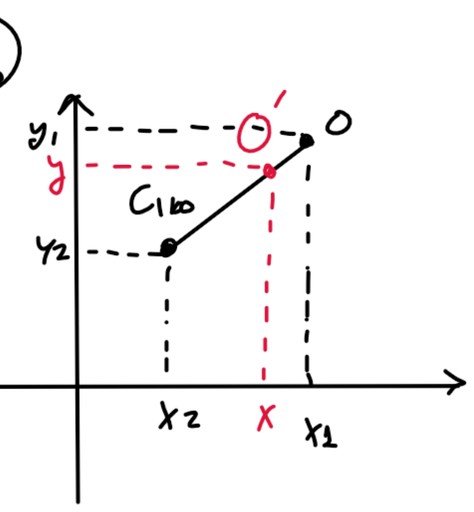
K



$$x = x_1 + 0.2(x_2 - x_1)$$

$$y = y_1 + 0.2(y_2 - y_1)$$

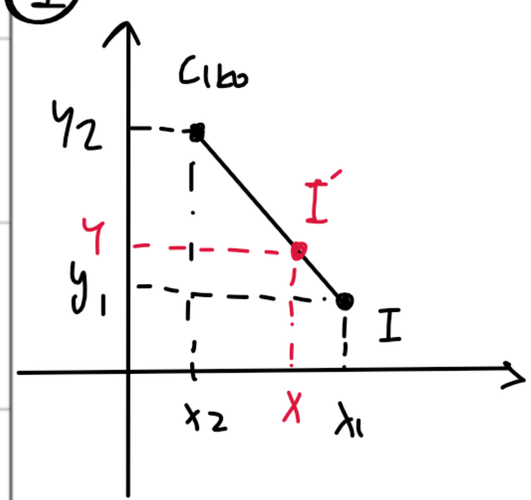
O



$$x = x_1 + 0.2(x_2 - x_1)$$

$$y = y_1 + 0.2(y_2 - y_1)$$

I



$$x = x_1 + 0.2(x_2 - x_1)$$

$$y = y_1 + 0.2(y_2 - y_1)$$

K´

$$x_{K'} = x_K + 0.2(x_C - x_K) = 8.5 + 0.2 \cdot (10.333 - 8.5) = 8.866$$

$$y_{K'} = y_K + 0.2 \cdot (y_C - y_K) = 5 + 0.2(5.666 - 5) = 5.133$$

O´

$$x_{O'} = x_O + 0.2(x_C - x_O) = 11.266$$

$$y_{O'} = y_O + 0.2(y_C - y_O) = 7.533$$

I´

$$x_{I'} = x_I + 0.2(x_C - x_I) = 10.866$$

$$y_{I'} = y_I + 0.2(y_C - y_I) = 4.333$$

K´(8.866, 5.133) O´(11.266, 7.533) I´(10.866, 4.333)

(d) (3 points) Calculate the minimum possible value for threshold distance d that results in 3 clusters (that is, no pair of clusters are merged), and fill in the placeholder (?). Truncate the distances between representative points to 3 decimal digits (do not round up or down the numbers). Use three decimal digits precision for d.

Answer:

$d = ?$

$C_1 : \{F, B, D\}$

$C_2 : \{A, N, E\}$

$C_3 : \{I, K, O\}$

- Compute the centroids of C_1 and C_2

C_1	C_2
F(5,2)	A(2.5,11.5)
B(1,5.5)	N(6,8)
D(2,2)	E(6,12)

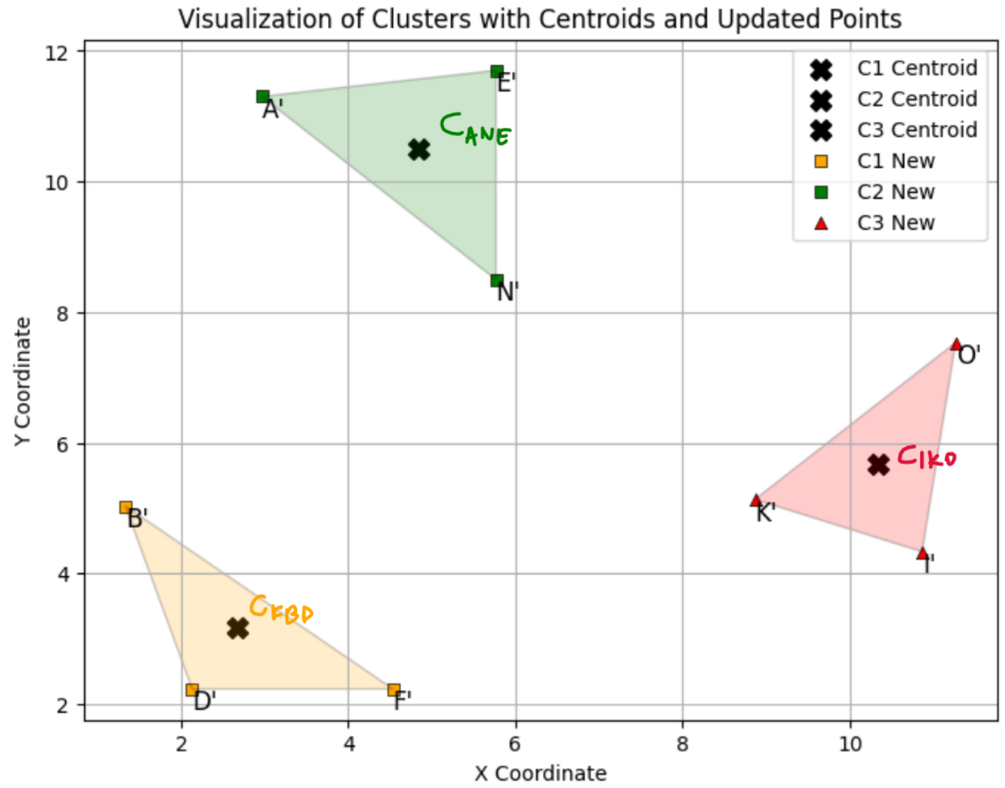
$C_{FBD} = \left(\frac{5+1+2}{3}, \frac{2+5.5+2}{3}\right) = (2.666, 3.166)$

$C_{ANE} = (4.833, 10.5)$

- Applying $X_{NEW} = X_{OLD} + 0.2 \cdot (X_{centroid} - X_{OLD})$ we calculate the coordinates of the new points.
  $Y_{NEW} = Y_{OLD} + 0.2 (Y_{centroid} - Y_{OLD})$

C_1: $F'(4.533, 2.233)$, $B'(1.373, 5.033)$, $D'(2.133, 2.233)$

C_2: $A'(2.966, 11.300)$, $N'(5.766, 8.5)$, $E'(5.766, 11.700)$



Calculate pairwise distances between C_1 and C_2 (I will use F, B, D, A etc instead of F', B', D', A', etc)

$d(F,A) = 9.201$, $d(F,N) = 6.387$, $d(F,E) = 9.546$

$d(B,A) = 6.476$, $d(B,N) = 5.627$, $d(B,E) = 8.006$

$d(D,A) = 9.105$, $d(D,N) = 7.243$, $d(D,E) = 10.14$

minimum distance $d_{min}(C_1, C_2) = 5.627$

Calculate pairwise distances between C_1 and C_3

$d(F,I) = 6.672$, $d(F,K) = 5.213$, $d(F,O) = 8.568$

$d(B,I) = 9.558$, $d(B,K) = 7.533$, $d(B,O) = 10.242$

$d(D,I) = 8.981$, $d(D,K) = 7.33$, $d(D,O) = 10.559$

minimum distance $d_{min}(C_1, C_3) = 5.213$

Calculate pairwise distances between C_2 and C_3

$d(A,I) = 10.533$, $d(A,K) = 8.534$, $d(A,O) = 9.114$

$d(N,I) = 6.585$, $d(N,K) = 4.576$, $d(N,O) = 5.584$

$d(E,I) = 8.96$, $d(E,K) = 7.261$, $d(E,O) = 6.9$

minimum distance $d_{min}(C_2, C_3) = 4.576$

To prevent merging, the smallest distance between any two clusters, must be greater than 4.576