**Program: General/Intelligent Systems/Cybersecurity**

**Level: Third** **Term: Fall 2023/2024**

**Course Code: 02-24-01203** **Course Title: Data Science Tools &Software**

**Total points: 5** **Professor name: Dr. Mohamed Abd El-Hafeez**

# Assignment #4

Q1) Given the following dissimilarity matrix between three objects

$$P(X)\begin{bmatrix} 0 & 1 & 4 \\ 1 & 0 & 2 \\ 4 & 2 & 0 \end{bmatrix}$$

a) Sketch a dendrogram using single linkage strategy?

b) Sketch a dendrogram using complete linkage strategy?

c) Write down the corresponding Python steps of parts (a) include a proper cut-off points?

d) Using the first and second object as initial two centers for clusters $C_1$ and $C_2$ respectively. What is the hard membership of the third object in $C_1$ and $C_2$

e) What is the center of the new formed cluster in part (d) if $x_1$=[1 3], $x_2$=[2 5] and $x_3$=[3 7]

Q2) Hierarchical, Medoid-based and Density-based clustering

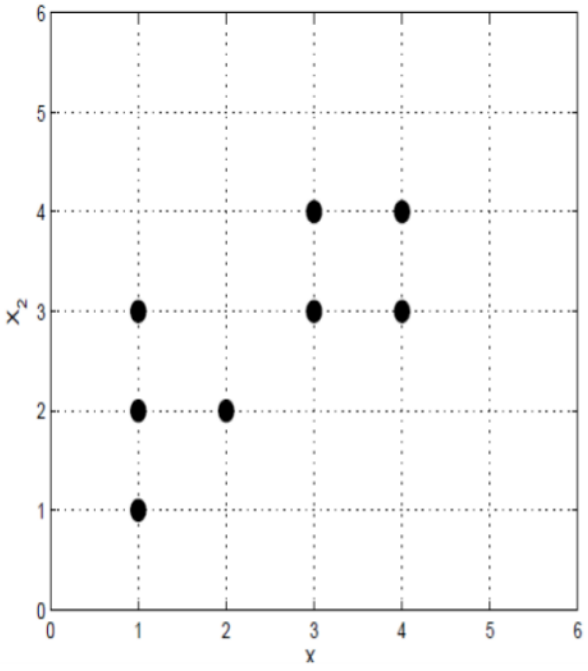
a) Starting with the two cluster medoids: m1= (1, 1) and m2= (3, 3). What is the cost of replacing m1 with (2,2)?

b) Cluster above data using complete linkage strategy and sketch the corresponding dendrogram

c) Sketch k-dist graph for $k$=3 and estimate suitable value for Epson.

Apply DBSCAN on the given dataset starting from (4,4) using $k$=1 and Epson=$\sqrt{2}$

d) Estimate cutting edges for the dendrogram in part b to produce the same clusters produced by DBSCAN in part d



Q3) Given the following dissimilarity matrix between five objects

$$P(X) = \begin{bmatrix} 0 & 1 & 5 & 6.4 & 7.4 \\ 1 & 0 & 4.2 & 5.7 & 6.7 \\ 5 & 4.2 & 0 & 1.4 & 2.5 \\ 6.4 & 5.7 & 1.4 & 0 & 1.1 \\ 7.4 & 6.7 & 2.5 & 1.1 & 0 \end{bmatrix}$$

a) Draw a dendrogram using single linkage strategy?

b) Select a proper cut off point and output the corresponding clustering?

c) Draw a dendogram using complete linkage strategy?

d) Write down the corresponding Python steps of parts a,b,c?

e) comment on the resulting dendrograms in a and c?

f) What is the complexity of the algorithm in terms of number of objects?

Q4) Given Dataset X of size n×d

a) The size of similarity matrix between elements of X is -----

b) The size of membership matrix of X in k clusters is -----

c) The size of Linkage matrix output by matlab is------

d) The number of internal nodes of dendrogram is ------

e) in order to determine the pair of clusters that is going to be merged at the *r + 1 level,* -------- pairs of clusters have to be considered.

f) The complexity of heirarchical clustering is ---------

Q5) Select (True/False) to answer the following questions:

1. Number of clusters is a required parameter for hierarchical algorithms [ ]
2. Cancer Diagnosis is a clustering application [ ]
3. Image segmentation is a classification problem [ ]
4. Noise is a random error or variance in a measured variable [ ]

Q6) Compute Accuracy, Sensitivity and Specificity of the following classification results?

| ID | Income | Marital Status | Refund | Actual Class | Predicted Class |
|---|---|---|---|---|---|
| 1 | 95K | Married | Yes | -1 | 1 |
| 2 | 120K | Single | Yes | -1 | -1 |
| 3 | 140K | Single | No | 1 | 1 |
| 4 | 80K | Married | No | -1 | -1 |
| 5 | 160K | Divorced | Yes | -1 | 1 |
| 6 | 100K | Married | Yes | -1 | -1 |
| 7 | 90K | Single | Yes | -1 | 1 |
| 8 | 75K | Divorced | No | -1 | -1 |
| 9 | 170K | Divorced | No | 1 | 1 |
| 10 | 125K | Single | No | 1 | -1 |

Q7) Consider the following k-nearest table of data points. What are the accuracy, specificity precision, recall, f-measure and sensitivity of k-nearest neighbor on them assuming the actual labels of x1,x3, and x5 is 1 and x2,x4 and x6 is -1 using leave-one-out cross validation method (k=1)?

| Object xi | Nearest Neighbors |
|---|---|
| x1 | x3,x4,x6 |
| x2 | x4,x6,x2 |
| x3 | x5,x2,x3 |
| x4 | x6,x3,x1 |
| x5 | x4,x2,x1 |
| x6 | x4,x1,x2 |

Q8) Assume we have the following two models and the following test data to be classified. Compute accuracy, sensitivity and specificity of M1 and M2

M1: if a>60 and b >60 then class = 1 otherwise class=-1

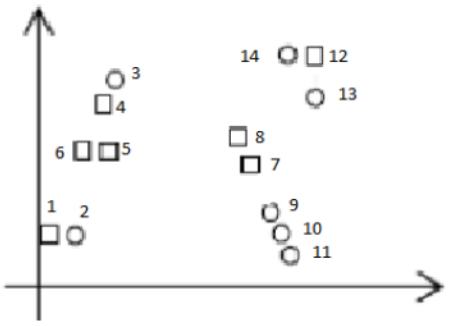
M2: if a>55 and c >65 then class = 1 otherwise class= -1

| a | b | c | class |
|---|---|---|---|
| 65 | 57 | 54 | 1 |
| 45 | 65 | 48 | -1 |
| 70 | 46 | 62 | -1 |
| 48 | 91 | 87 | 1 |
| 61 | 33 | 38 | 1 |
| 66 | 59 | 76 | -1 |
| 58 | 84 | 53 | 1 |

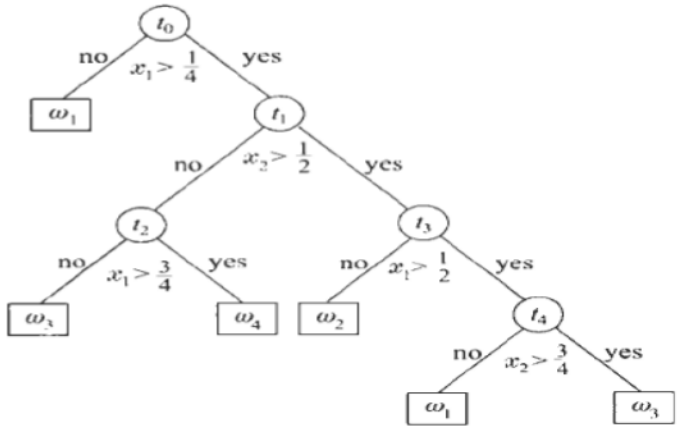
Q9) Predict the house price index if age = 38 and loan $55,000 using kNN regressor with k=3

| Age | Loan | House Price Index | Distance |
|---|---|---|---|
| 25 | $40,000 | 135 | 102000 |
| 35 | $60,000 | 256 | 82000 |
| 45 | $80,000 | 231 | 62000 |
| 20 | $20,000 | 267 | 122000 |
| 35 | $120,000 | 139 | 22000 |
| 52 | $18,000 | 150 | 124000 |
| 23 | $95,000 | 127 | 47000 |
| 40 | $62,000 | 216 | 80000 |
| 60 | $100,000 | 139 | 42000 |
| 48 | $220,000 | 250 | 78000 |
| 33 | $150,000 | 264 | 8000 |

Q10) Consider the data points at the right. Using visual inspection, compute Accuracy, Sensitivity and Specificity when kNN with $k$ =1 is applied to this dataset using leave-one-out cross validation. Note: create a table contains id, actual and predicted labels where circle represents negative sample and square represents positive sample before computing the metrics.



Q11) Regarding the following Decision Tree, generate induction rules for all classes $w_1$-$w_4$. What is the class that will be assigned for[$x_1$ $x_2$] equals [0.4, 0.6]?




Q12) Given the following training set where the column with caption 'Cheat' is the class label, compute the information gain of Refund and Marital Status and select the best of them to start with and complete your decision tree by choosing the unselected attribute followed by the income attribute as your next choices to build your model. Classify an unseen sample with the following attribute values:- income=60k, single, and refund=no.

| Income | Marital Status | Refund | Cheat |
|---|---|---|---|
| 95K | Married | Yes | NO |
| 120K | Single | Yes | NO |
| 140K | Single | No | YES |
| 80K | Married | No | NO |
| 160K | Divorced | Yes | NO |
| 100K | Married | Yes | NO |
| 90K | Single | Yes | NO |
| 75K | Divorced | No | NO |
| 170K | Divorced | No | YES |
| 125K | Single | No | YES |

Q13) Given the following training set, classify
X=(age<=30 , income =medium, student=yes, credit_rating=fair)
using NB classifier.

| age | income | student | credit_rating | buys_computer |
|---|---|---|---|---|
| >40 | medium | no | fair | yes |
| >40 | low | yes | fair | yes |
| >40 | low | yes | excellent | no |
| <=30 | medium | no | fair | no |
| <=30 | low | yes | fair | yes |
| >40 | medium | yes | fair | yes |

Q13) add label and information to nodes and edges in the following example and show the output

```
>>> g.add_edges_from([(1 ,2) ,(1 ,3)])
>>> g.add_node('a')
>>> g.number_of_nodes() # also g.order()
4
>>> g.number_of_edges() # also g.size()
2
>>> g.nodes()
[1, 2, 3, 'a']
>>> g.edges()
[(1, 2), (1, 3)]
>>> g.neighbors(1)
[2, 3]
>>> g.degree(1)
2
```

**Q14) reverse the direction of edges and rerun the following example to show the output**

```
>>> dg = nx.DiGraph()
>>> dg.add_weighted_edges_from([(1,4,0.5), (3,1,0.75)])
>>> dg.out_degree(1,weighted=True)
0.5
>>> dg.degree(1,weighted=True)
1.25
>>> dg.successors(1)
[4]
>>> dg.predecessors(1)
[3]
```

**Q14) show the output of each of the following code**

```
# small famous graphs
>>> petersen=nx.petersen_graph()
>>> tutte=nx.tutte_graph()
>>> maze=nx.sedgewick_maze_graph()
>>> tet=nx.tetrahedral_graph()

# classic graphs
>>> K_5=nx.complete_graph(5)
>>> K_3_5=nx.complete_bipartite_graph(3,5)
>>> barbell=nx.barbell_graph(10,10)
>>> lollipop=nx.lollipop_graph(10,20)

# random graphs
>>> er=nx.erdos_renyi_graph(100,0.15)
>>> ws=nx.watts_strogatz_graph(30,3,0.1)
>>> ba=nx.barabasi_albert_graph(100,5)
>>> red=nx.random_lobster(100,0.9,0.9)
```

**Q15) replace the dataset in the following code by any of your choice and print nodes, edges count and average degree**

```
hartford = nx.read_edgelist('hartford.txt',
          create_using=nx.DiGraph(),nodetype=int)
```

**Q16) add new node e with edge to c and a having weight 0.9 and 1.3 respectively and recompute the shortest path between a and d and the minimum spanning tree.**

```
>>> import networkx as nx
>>> g = nx.Graph()
>>> g.add_edge('a','b',weight=0.1)
>>> g.add_edge('b','c',weight=1.5)
>>> g.add_edge('a','c',weight=1.0)
>>> g.add_edge('c','d',weight=2.2)
>>> print nx.shortest_path(g,'b','d')
 ['b', 'c', 'd']
>>> print nx.shortest_path(g,'b','d',weighted=True)
 ['b', 'a', 'c', 'd']
```