First and last name: Sofya Aksenyuk | Index: 150284 | Score: / 20p.

1. [3p] Given the representation of 7 users (U1-U7) in terms of 2 documents (D1-D2), use the **3-means algorithm** (K-means with K=3) to group these users into clusters. One has already drawn the three centroids (C1-C3; e.g., C1 is a centroid of group G1) in the 1st iteration. Also, a similarity matrix between the users and the centroids has been computed. Present the groups (G1-G3) obtained after the 1st iteraion (e.g., G1: U2, U5). Compute the value of the J measure after the first iteration. Compute the representation of centroids (C1-C3) used in the 2nd iteration based on the obtained clustering.

Representation of 7 users (objects) in terms of 2 documents (features):

	U1	U2	U3	U4	U5	U6	U7
D1	0.7	0.3	0.6	0.2	0.8	0.6	0.2
D2	0.2	0.7	0.7	0.6	0.3	0.8	0.5

Similarity matrix between the centroids and the users:

	U1	U2	U3	U4	U5	U6	U7
C1	0.83	0.95	1.0	0.92	0.87	0.99	0.94
C2	0.61	0.99	0.94	0.99	0.67	0.96	1.0
C3	1.0	0.63	0.83	0.56	0.99	0.79	0.61

Clustering in the 1st iterat.

for each cluster, list the users

G1: U3, U6
G2: U2 , U4 , U7
G3: U1, U5

Centroids in the 2nd iteration:

	C1	C2	C3
D1	0,6	0,23	0,75
D2	0,75	0,6	0,25

J measure after the 1st iteration

J = (1+0,99) + (0,99+0.99+1) + (1+0.99) = 6,96

2. [2.5p] Given the classification and similarities of 10 training documents (D1-D10) with document Y, determine the class of Y using **6-NN (6-Nearest Neighbour)** classifier incorporating the majority rule voting or the weighted voting. List the documents that you consider to determine the class of Y. Compute the scores for classes A, B, and C for both variants (unweighted and weighted) of the classifier.

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	Y
Class	B	C	B	A	B	A	A	B	C	C	?
Similiraity with Y	0.7	0.8	0.9	0.65	1.0	0.6	0.75	0.5	0.8	0.9	-

Documets considered to classify Y in 6-NN:						
6-NN (majority rule)	Class A:	2	Class B:	3	Class C:	1
Answer: Y is assigned to class:		B				
6-NN (weighted voting)	Class A:	0,65+0,6=1.25	Class B:	0.7+0.9+1=2.6	Class C:	0.8
Answer: Y is assigned to class:		B				

3. [1.5p] Given the **confusion matrix** and the **cost matrix** for the classification problem involving five classes C1-C5 (original classes in rows; predicted classes in columns) and 100 documents, compute the classification accuracy, recall for class C4, and misclassification cost

Confusion matrix

	C1	C2	C3	C4	C5
C1	10	0	2	0	0
C2	2	16	0	8	0
C3	2	4	14	0	0
C4	2	0	4	10	0
C5	4	0	0	2	20

Cost matrix

	C1	C2	C3	C4	C5
C1	0	3	4	1	2
C2	1	0	1	1	2
C3	1	1	0	1	3
C4	1	0	5	0	1
C5	1	1	1	1	0

Classification accuracy = (10+16+14+10+20)/100 = 70/100 Recall for C4 = 10/(10+2+4) = 10/16

Misclassification cost = 2*4 + 1*2 + 8*1 + 1*2 + 4*1 + 2*1 + 4*5 +4*1 + 2*1 = 8+2+8+2+4+2+20+4+2 = 52

4. [2p] Given two chromosomes X = [1 5 3 7 2 4 6] and Y = [4 7 5 3 2 6 1] representing permutations, present a) a pair of chromosomes obtained after applying **order 1 crossover** to X and Y with arbitrary parts inherited by the respective children between 2nd and 4th genes, b) a pair of chromosome obtained after applying **cycle crossover** to X and Y.

a) order 1 crossover (parents)

X	1	5	3	7	2	4	6
Y	4	7	5	3	2	6	1

a) offspring

4	5	3	7	2	6	1
1	7	5	3	2	4	6

1st (↑) and 2nd (↓) child

b) cycle crossover (parents)

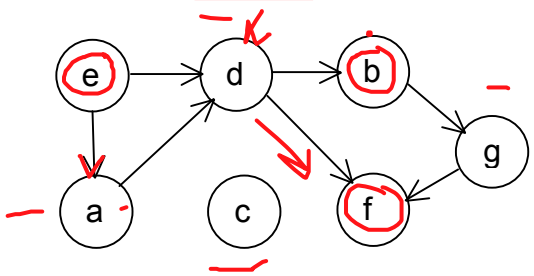
X	1	5	3	7	2	4	6
Y	4	7	5	3	2	6	1

b) offspring

1	7	5	3	2	4	6
4	5	3	7	2	6	1

1st (↑) and 2nd (↓) child

5. [2p] Given the outranking graph concerning seven alternatives a-g, find its **kernel** (you do not need to present the steps of the formal algorithm involving the analysis of predecessors) and indicate the relations holding for the two showed pairs of alternatives from among P (preference), I (indifference), and ? (incomparability). An incorrect answer cancels out the points for one correct answer.



Kernel =	{e, b, f, c}	
Indicate relations	a ?..... b	d P..... f

a S c and c S a

6. [3.5p] Given the information table referring to condition attributes A and B and a decision attribute C (class C1 or C2), compute *Ent*(C), *Ent*(C,A), *Ent*(C,B), *InformationGain*(C,A) and *InformationGain*(C,B) in the first iteration. Fill in the entire decision tree obtained with **the ID3 algorithm** incorporating information gain for splitting in each node. You do not need to present any computations for the second tree level (if needed) - since there are only two attributes, it is natural that in the second iteration you will use the attribute that has not been used in the first iteration. Remember that: $\log_2 1 = 0$, $\log_2(1/2) = -1$, $\log_2(1/3) = -1.585$, $\log_2(2/3) = -0.585$, $0 \cdot \log_2 0 = 0$; $1/2\log_2(1/2) + 1/2\log_2(1/2) = -1$; $1/3\log_2(1/3) + 2/3\log_2(2/3) = -0.918$; $3/8*0.918 = 0.344$.

	A	B	C
1	U	T	C1
2	U	T	C1
3	P	W	C1
4	P	W	C1
5	P	T	C2
6	R	T	C2
7	R	W	C2
8	R	W	C2

Ent(C) = - 4/8*log2 4/8 - 4/8*log2 4/8 = 0.5 + 0.5 = 1

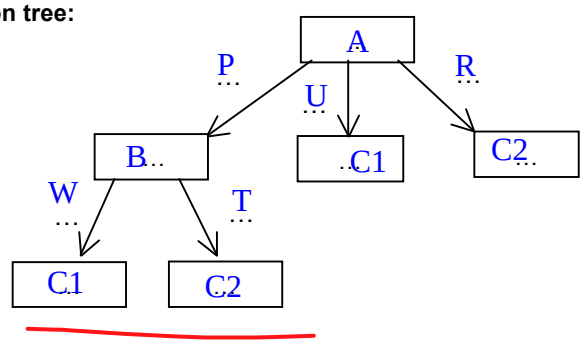
Ent(C,A) = 4/8*(-2/4*log2 2/4 - 2/4*log2 2/4 - 0/4*log2 0/4) + 4/8*(-0/4*log2 0/4 - 1/4*log2 1/4 - - 3/4*log2 3/4) = 0.5*(0.5 + 0.5 + 0) + 0.5*(0 + 0.5 + 0.31125) = 0.5 + 0.4 = 0.9

Ent(C,B) = 4/8*(-2/4*log2 2/4 - 2/4*log2 2/4) + 4/8*(-2/4*log2 2/4 - 2/4*log2 2/4) = 1

InformationGain(C,A) = *Ent*(C) - *Ent*(C,A) = 1 - 0.9 = 0.1

InformationGain(C,B) = *Ent*(C) - *Ent*(C,B) = 1- 1 = 0

Decision tree:



7. [2.5p] Given the 4x4 matrix of inputs (see below), **convolve** it with the 3x3 filter (**the bias value is given below the 3x3 matrix**) with a stride of 1, then apply **ReLU** on the matrix obtained after the convolution, and finally apply **average (AVE) pooling** with a filter of **size 2x2** and a stride of 2 on the matrix obtained after ReLU. You need to determine the size of matrices obtained after each operation on your own (the 4x5 matrices given below serve only for filling the results, but you should use only as many as cells as you need).

input (4x4)

0	-2	0	-1
0	2	1	-2
-2	1	0	0
1	0	2	1

filter (3x3)

1	0	0
0	-1	0
0	0	1

bias = +2

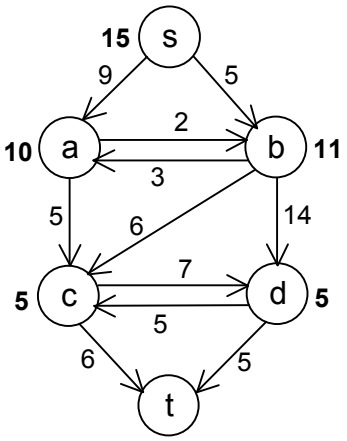
after convolution

-2	-3	2	-1	0

after ReLU

after AVE pooling

8. [3p] Find the shortest path from **s** to **t** in the graph presented below using the **A*** (A star) algorithm (the distances between the nodes are given next to the edges and the heuristc distances ***h*** from node **t** are given in bold next to the nodes). Fill in the table showing the process of finding the shortest pass. Indicate the shortest path and provide its length. To denote the visited node in the first column, use the "X" symbol. You do not need to cross out the distances or previous nodes that are updated in the next iterations, but they need to be visible in the table, so do not delete them either.



Visit?	Node	Shortest distance from **s**	Heuristic distance to **t**	Total distance **f** = g + h	Previous node
X	s	0	15	15	
	a	~~∞~~ 9	10	19	s
X	b	~~∞~~ 5	11	16	s
X	c	~~∞~~ 11	5	16	b
	d	~~∞~~ 18	5	23	c
X	t	~~∞~~ 17	0	17	c

Answer: The shortest path from **s** to **t** is: s - b - c - t

The length of the shortest path is: 48