National University of Computer and Emerging Sciences
Lahore Campus

Artificial Intelligence (CS/DS 2002)
Date: 2024

Course Instructors
Dr. Asma Ahmad
Dr. Hajra Waheed
Dr. Mirza Mubasher Baig
Dr. Sadaf Jabeen
Ms. Bushra Rashid
Ms. Maham Naeem
Ms. Saba Tariq
Mr. Saif Ul Islam

FINAL
Total Time: 3 Hours
Total Marks: 85
Total Questions: 06

Semester: SP-2024
Campus: Lahore

_______________________________ ___________________ _______________________
Student Name Roll No Section Signature

ATTEMPT ALL QUESTIONS ON THE ANSWER SHEET

Question No 1. [Short Questions] [10 x 3 Points]

i. It is a well-known fact that a local search algorithm can converge to a local optima. Assume, you have a black-box implementation of a local search algorithm but you want to minimize the probability of finding a locally optimal solution. How can use the black-box implementation (i.e. the implementation you cannot change) to find a better (or perhaps the best) solution by repeatedly using it.

Multiple Restarts: Run the algorithm multiple times with different random starting points **AND PICK THE BEST SOLUTION.**

ii. Given two arbitrary admissible heuristics, $h1$ and $h2$, which composite heuristics are also admissible? Justify.

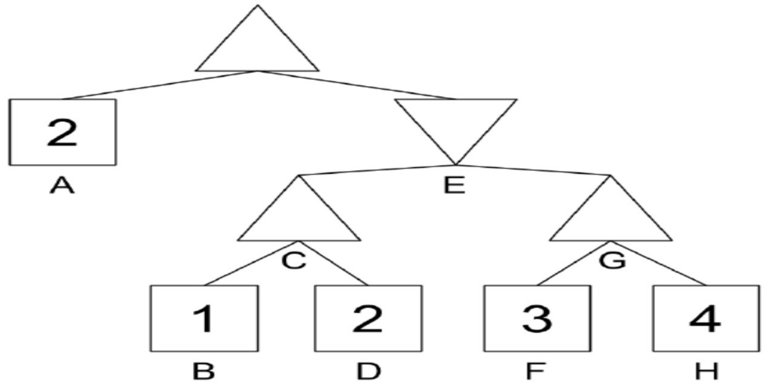
a) Max (h_1, h_2)	**b)** ($h1 + h2$)/2	**c)** Min ($h1$, $h2$)?	**d)** h1*h2

Since both h1 and h2 are admissible, the maximum value will also be less than or equal to the actual cost Therefore, **Max(h1, h2)** is also admissible **SO IS MIN**.
The average will also be less than or equal to the actual cost. Therefore, **(h1 + h2) / 2** is also admissible.

iii. In an A* implementation, you have a choice of using one of the heuristics given in the previous part. Which one of the heuristics will you use? Justify your choice

PICK MAX BECAUSE ADMISSIBLE HEURISTIC WITH LARGE VALUE IS PREFERRED

For the next three questions consider the following short game tree being used by minimax to decide a move.



iv. Which terminal nodes will never be explored as a results of pruning if the minimax explores the nodes from left to right i.e. node A is explored first.
NODE F & H

v. Now assume that you are allowed to change value of one of the leaf node. Suggest a new value of one of the selected node such that no pruning takes place if the tree is processed from left to right
MAKE THE VALUE OF EITHER B OR D LARGER THAN 2

vi. Argue that the following statement is true for local search.

Hill climbing with random restarts is guaranteed to find the global optimum if it runs long enough on a finite state space.

BECAUSE IT IS AT LEAST AS GOD AS RANDOM SEARCH SO IS COMPLETE

vii. What would go wrong during gradient based learning if all weights of a multi-layer neural network are initialized to 0 (ZERO)? Think about backpropagation step to answer this question.

NO LEARNING IN HIDDEN LAYERS AS WEIGHTS ARE MULTIPLIED WITH GRADIENTS DURING BACK PROPAGATION

All neurons in a layer compute identically due to identical weights, hindering them from learning diverse features.

viii. What would be the size of output feature-map if a 7x7 filter is convolved with a 56x56 image with a stride of 2 followed by a 2x2 max pooling with a stride of 2?

Output feature-map: 25x25
Max pooling: 12

28 X 28 AND 14 ARE ALSO ACCEPTABLE

ix. Calculate the size (i.e. number of learnable weights) of a feed-forward neural network that has a 28x28 image as input and has 20 neurons in a single hidden layer and 10 neurons in the output layer.

Hidden layer: 28x28x20+20 = 15700
Output layer: 20x10+10 = 210
Total weights = 15910

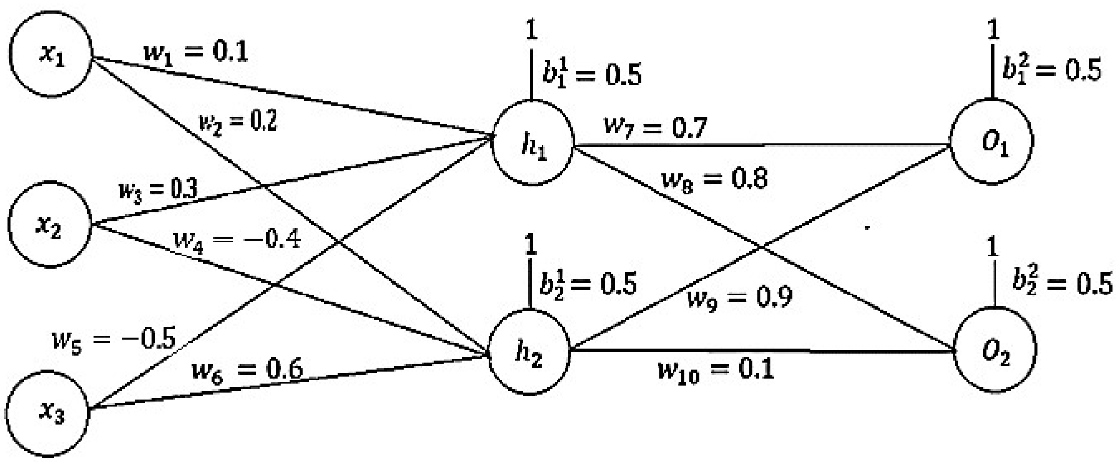
x. Repeat the previous part for the same 28x28 input image and for a CNN with a convolution layer consisting of 10 filters of size 3x3 applied with a stride of 2 followed by a 2x2 max pooling layer and a fully connected layer consisting of 20 neurons in a single hidden layer and 10 neurons in the output layer. Assume that the convolution is applied with zero padding enabled.

Convolutional Layer weights= **3x3x10+10** = 100
Convolution output = 14x14
Max Pool output = 7x7
Hidden layer weights = 7x7x20+20 = 1000
Output layer weights = 20x10+10 = 210
Total = 100+1000+210 = 1310

Question No 2 [Learning a NN Using Back Propagation] **(Marks: 15)**

Calculate new weights and bias values by using Back propagation Algorithm. Multi-layer feed-forward neural network is given below. The input and target values for this problem are x1=10, x2=40, x3=50 and target at O1=0 and O2=1. Initialization of weights is already given below. Calculate the updated weights using Back Propagation Algorithm where network uses ReLU as activation function in the hidden layer and sigmoid activation at the output layer. Assume the learning rate of 0.1.

The marks will be awarded for the following
i) Correct forward pass **(5 Points)** ii) Correct Gradient terms calculated at the output layer **(2 Points)**
iii) Correct propagation of gradients from output to hidden layers **(4 Points)** and iv) correctly updating the weights **(4 Points)**



Neural Network

Forward pass

$x_1 = 10, \; x_2 = 40, \; x_3 = 50$

$z_1 = w_1 x_1 + w_3 x_2 + w_5 x_3 + b_1^1$

$z_1 = 0.1 \times 10 + 0.3 \times 40 + (-0.5 \times 50) + 0.5$

$z_1 = -11.5$

$h_1 = 0$

$z_2 = w_2 x_1 + w_4 x_2 + w_6 x_3 + b_2^1$

$z_2 = 0.2 \times 10 + (-0.4 \times 40) + (0.6 \times 50) + 0.5$

$z_2 = 16.5$

$h_2 = 16.5$

$y_1 = w_7 h_1 + w_9 h_2 + b_1^2$

$y_1 = (0.7 \times 0) + (0.9 \times 16.5) + 0.5$

$y_1 = 15.35$

$O_1 = \frac{1}{1+e^{-O_1}} = 0.999$

$y_2 = w_8 h_1 + w_{10} h_2 + b_2^2$

$= (0.8 \times 0) + (0.1 \times 16.5) + 0.5$

$y_2 = 2.15$

$O_2 = 0.895$

Back propagation

$d_1 = 0,\ d_2 = 1,\ O_1 = 0.99,\ O_2 = 0.89$

$\Delta w_{jk} = -c\left[-(d_k - O_k)\cdot f'(act_k)\cdot x_j\right]$

$\Delta w_{jk} = c\left[(d_k - O_k)O_k(1-O_k)x_j\right]$

$\Delta w_7 = 0.1\left[(0-0.99)0.99(1-0.99)\times 0\right] = 0$

$\boxed{w_7 = 0.7}$

$\Delta w_8 = 0$

$\boxed{w_8 = 0.8}$

$\Delta w_9 = c\left[(d_1 - O_1)O_1(1-O_1)h_2\right]$

$= 0.1\left[(0-0.99)0.99(1-0.99)16.5\right]$

$\Delta w_9 = -0.016$

$w_9 = 0.9 - 0.016$

$\boxed{w_9 = 0.883}$

$\Delta w_{10} = c\left[(d_2 - O_2)O_2(1-O_2)h_2\right]$

$= 0.1\left[(1-0.89)(0.89)(1-0.89)16.5\right]$

$= 0.017$

$w_{10} = 0.1 + 0.017$

$\boxed{w_{10} = 0.117}$

$\Delta b_1^2 = c\left[(d_1 - O_1)O_1(1-O_1).1\right]$

$= -9.801\times 10^{-4}$

$b_1^2 = 0.5 - 9.8\times 10^{-4} = 0.499$

$\boxed{b_1^2 = 0.499}$

$\Delta b_2^2 = c\left[(d_2 - O_2)O_2(1-O_2).1\right]$

$= 1.076\times 10^{-3}$

$\boxed{b_2^2 = 0.501}$

Hidden Layer weights

$\Delta w_{ij} = -c\left(\partial Errr/\partial w_{ij}\right)$

$= -c\left[-\sum_k \{(d_k - O_k) f'(act_k) w_{jk}\} f'(act_j)\, x_i\right]$

$\Delta w_1 = c\left[\{(d_1-O_1)O_1(1-O_1)w_7 + (d_2-O_2)O_2(1-O_2)w_8\}\, 0\times h_1\right]$

$\Delta w_1 = 0$

$\boxed{w_1 = 0.1}$

$\Delta w_3 = 0,\ \Delta w_5 = 0,\ \Delta b_1^1 = 0$

$\boxed{w_3 = 0.3},\ \boxed{w_5 = 0.5},\ \boxed{b_1^1 = 0.5}$

$\Delta w_2 = c\left[\{(d_1-O_1)O_1(1-O_1)w_9 + (d_2-O_2)O_2(1-O_2)w_{10}\} \times 1 \times x_1\right]$

$= \cancel{-8.829\times 10}$

$= 0.1\left[-8.82\times 10^{-3} + 1.0769\times 10^{-3}\right]\times 10$

$= -7.743\times 10^{-3}$

$\boxed{w_2 = 0.192}$

$\Delta w_4 = -0.030$

$\boxed{w_4 = -0.430}$

$\Delta w_6 = -0.038$

$\boxed{w_6 = 0.561}$

$\Delta b_1^1 =$

$\Delta b_2^1 = -7.743\times 10^{-4}$

$\boxed{b_2^1 = 0.499}$

Question No 3 [CNN] [5 + 2 + 3 Points]

i. Using the given 6x6 input image and a 3x3 filter with a stride of 1 and a bias of 1, compute the output feature map obtained after applying ReLU activation. Assume that zero-padding is not applied during the convolution operation.

Image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

Filter

1	0	-1
1	0	-1
1	0	-1

-4	-3	1	9
-9	-1	3	4
1	-1	-3	-6
-2	-1	-2	-15

After RELU

0	0	1	9
0	0	3	4
1	0	0	0
0	0	0	0

ii. Apply a 2x2 max pooling filter on the resultant feature map with stride=2 and show the output matrix.

0 9
1 0

iii. Flatten the resultant output from the MaxPool and pass it to a output layer with a single neuron. Take any weights of your choice for the output neuron and compute the final output assuming that the sigmoid activation is used at the output layer. Don't forget to include the bias term at the output layer.

Question No 4 [Measuring Performance of Learning Algorithms] [4 + 6 Points]

Imagine you have a dataset containing samples of flowers, where each flower can be classified into one of three categories: roses, daisies, or tulips. You've trained a neural network on this dataset and evaluated it on test data. After evaluation, you've obtained the following results:

a) Out of the 50 roses in the test set, the model correctly predicted 40 as roses, but mistakenly classified 5 as daisies and 5 as tulips.

b) Out of the 50 daisies in the test set, the model correctly predicted 45 as daisies, but mistakenly classified 3 as roses and 2 as tulips.

c) Out of the 50 tulips in the test set, the model correctly predicted 48 as tulips, but mistakenly classified 1 as roses and 1 as daisies.

i. Show a confusion matrix for this model using the information of performance given above

ii. Using the confusion matrix created in previous part compute accuracy, average precision, average recall

Confusion Matrix

Predicted

		Roses	Diases	Tulips
Actual	R	40	5	5
	D	3	45	2
	T	1	1	48

Accuracy := $(40+45+48)/150 = 88.67\%$

Average Precision := (Precision R + Precision D + Presion T) / 3

$$= \left(\frac{40}{40+3+1} + \frac{45}{45+5+1} + \frac{48}{48+5+2}\right) / 3$$

$$= 2.664/3 = 0.888$$

Average Recall := (Recall R + Recall D + Recall T) / 3

$$= \left((40/50) + (45/50) + (48/50)\right)$$

$$= 0.887$$

Question No 5 [Clustering] [4 + 4 + 2 Points]

For the following points use two iterations of K-means clustering to partition the given points in 2 clusters.

Points: $A_1 = (1, 10)$, $A_2 = (2, 7)$, $A_3 = (8, 5)$, $A_4 = (4, 8)$, $A_5 = (8, 6)$, $A_6 = (7, 5)$, $A_7 = (2, 3)$, $A_8 = (5, 10)$

Assume that the following points are the Initial centres: C1= (2, 10), C2= (4, 8)

i. Make a table as follows and fill it with the correct values.

Points	Distance from C1	Distance from C2	Cluster
A1	1	3.605551275	C1
A2	3	2.236067977	C2
A3	7.810249676	5	C2
A4	2.828427125	0	C2
A5	7.211102551	4.472135955	C2
A6	7.071067812	4.242640687	C2
A7	7	5.385164807	C2
A8	3	2.236067977	C2

Updated C1:(1,10)
C2:(5.142857, 6.2857)
Iteration 2:

Points	Distance from C1	Distance from C2	Cluster
A1	0	5.564097826	C1
A2	3.16227766	3.223003973	C1
A3	8.602325267	3.133101756	C2
A4	3.605551275	2.060315173	C2
A5	8.062257748	2.871393148	C2
A6	7.810249676	2.258769712	C2
A7	7.071067812	4.546808398	C2
A8	4	3.717032232	C2

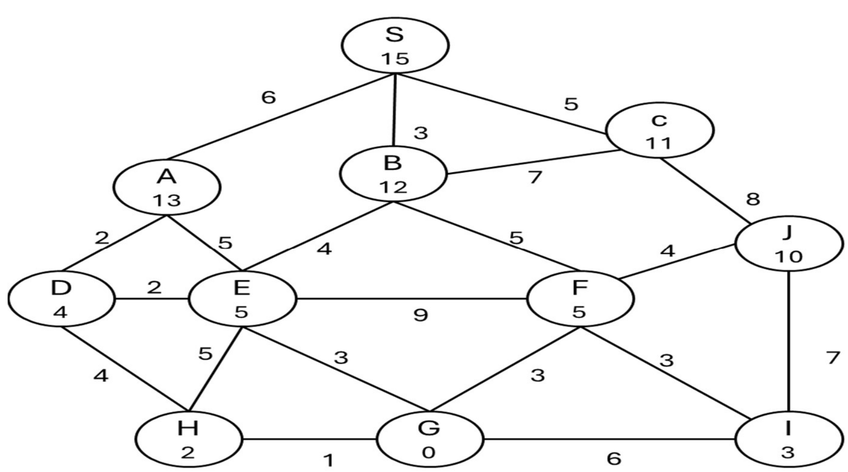
Updated C1:(1.5,8.5)
C2:(5.667,6.1667)

i. Determine the new clusters using the table computed in part I

ii. Calculate the updated means for the the new clusters.

Question No 6 [A* Search] [5 + 3 + 2 Points]

Consider the state space graph shown below. A is the start state and G is the goal state. The transition costs are next to the edges, and the heuristic estimates are given inside the node. Each edge can be traversed in both directions



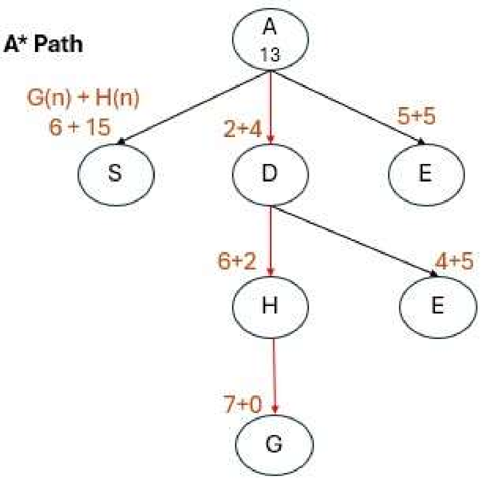
S

Consider A* graph search on the graph above. Assume that ties are broken alphabetically (So a partial plan S->X->A would be expanded before S->X->B in case of a tie.

i. What is the order of states expanded and the path found by A* graph search algorithm when used on this problem?
ii. Specify the nodes in the fringe/frontier/Queue when the goal is found.
iii. Is the heuristic function admissible? Justify your answer.

Solution: A is the start state and G is the goal state.

(i) A* Path



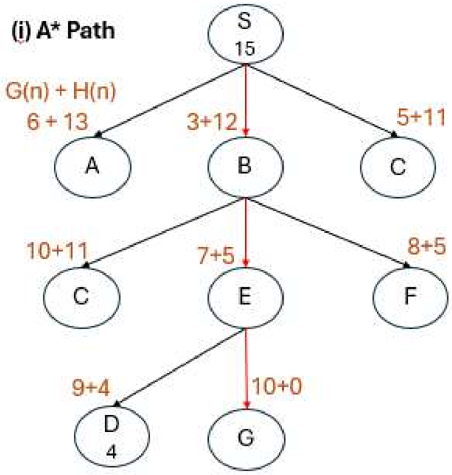
(ii) Status of the Frontier

A	13				
S	21	D	6	E	10
H	8	S	21	E	9
G	7	S	21	E	9

(iii) Not Admissible: The function is not admissible as it overestimates the value of reaching goal state as compared to the actual value. For example, the actual cost of reaching from A to G is 7. However, the heuristics offers the value 13.

Solution 2: S is the start state and G is the goal state.

(i) A* Path



(ii) Status of the Frontier

S	15

A	19	B	15	C	16

E	12	F	13	A	19	C	16

G	10	D	13	F	13	A	19	C	16

(iii) Not Admissible: The function is not admissible as it overestimates the value of reaching goal state as compared to the actual value. For example, the actual cost of reaching from A to G is 7. However, the heuristics offers value 13.