Reg. No. : | | | | | | | | | | | | |

Question Paper Code: U2912

M.E. DEGREE EXAMINATION, MAY 2024

Elective

Computer Science and Engineering

21PCS512 - DEEP LEARNING TECHNIQUES

(Regulations 2021)

Duration: Three hours | Maximum: 100 Marks

Answer ALL Questions

PART A - (5 x 20 = 100 Marks)

1. (a) Identify the clustering algorithms which are more appropriate for the following examples : 8 samples to convert into them into 3 clusters: A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9). Assume the initial seeds are A1, A4, and A7. CO3-App (20)

Or

(b) You are a robot in a lumber yard, and must learn to discriminate Oak wood from Pine wood. Apply Navie Bayes algorithm to classify the sample data. You are given the following (noisy) examples: CO3-App (20)

Example	Density	Grain	Hardness	Class
Example #1	Light	Small	Hard	Oak
Example #2	Heavy	Large	Hard	Oak
Example #3	Heavy	Small	Soft	Oak
Example #4	Heavy	Small	Soft	Oak
Example #5	Light	Large	Hard	Pine
Example #6	Light	Small	Soft	Pine
Example #7	Heavy	Large	Soft	Pine
Example #8	Light	Large	Hard	Pine

Consider a new example :

(Density=Heavy ^ Grain=Small ^ Hardness=Hard). Analyze these class probabilities as the product of α and common fractions from above.

2. (a) You want to train a neural network to drive a car. Your training data consists of grayscale 64 ×64 pixel images. The training labels include the human driver's steering wheel angle in degrees and the human driver's speed in miles per hour. Your neural network consists of an input layer with 64 × 64 = 4,096 units, a hidden layer with 2,048 units, and an output layer with 2 units (one for steering angle, one for speed). You use the ReLU activation function for the hidden units and no activation function for the outputs (or inputs). CO3- Ana (20)

a. Calculate the number of parameters (weights) in this network. You can leave your answer as an expression. Be sure to account for the bias terms.

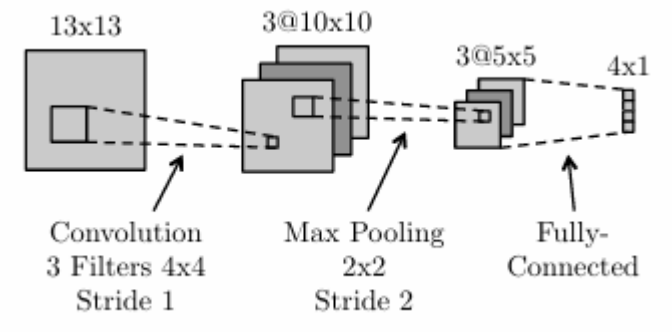
b. Write $\partial J/\partial W$ as an outer product of two vectors. $\partial J/\partial W$ is a matrix with the same dimensions as W; it's just like a gradient, except that W and $\partial J/\partial W$ are matrices rather than vectors.

c. You train your network with the cost function $J = 1 / 2 |y - z|^2$. Use the following notation. x is a training image (input) vector with a 1 component appended to the end, y is a training label (input) vector, and z is the output vector. All vectors are column vectors. $r(\gamma) = \max\{0, \gamma\}$ is the ReLU activation function, $r\ 0\ (\gamma)$ is its derivative (1 if $\gamma > 0$, 0 otherwise), and r(v) is $r(\cdot)$ applied component-wise to a vector. g is the vector of hidden unit values before the ReLU activation functions are applied, and h = r(g) is the vector of hidden unit values after they are applied (but we append a 1 component to the end of h). V is the weight matrix mapping the input layer to the hidden layer; g = V x. W is the weight matrix mapping the hidden layer to the output layer; z = Wh. Derive $\partial J/\partial W_{ij}$.

d) Derive $\partial J/\partial W_{ij}$.

Or

(b) Below is a diagram of a small convolutional neural network that converts a 13x13 image into 4 output values. The network has the following layers/operations from input to output: convolution with 3 filters, max pooling, ReLu, and finally a fully-connected layer. For this network we will not be using any bias/offset parameters (b). CO3- Ana (20)



Answer the following questions about this network.

a. How many weights in the convolutional layer do we need to learn?
b. How many ReLu operations are performed on the forward pass?
c. How many weights do we need to learn for the entire network?
d. A fully-connected neural network with the same size layers as the above network (13x13→ 3x10x10 → 3x5x5 → 4x1) can represent any classifier that the above convolutional network can represent. Yes or no Justify your answer.

What is the disadvantage of a fully-connected neural network compared to a convolutional neural network with the same size layers?

3. (a) Consider yourself to be one of the participants in the show. You choose one out of the three doors. Before opening your chosen door, Monty opens another door behind which would be one of the goats. Now you are left with two doors, behind one could be the car, and behind the other would be the other goat. CO2-App (20)

Monty then gives you the option to either switch your answer to the other unopened door or stick to the original one.

Is it in your favor to switch your answer to the other door? Well,

probability says it is!

Let's see how:

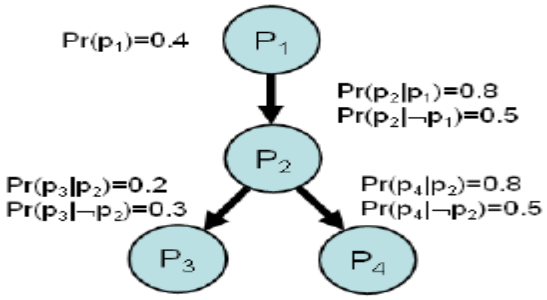
Initially, there are three unopen doors in front of you. The probability of the car being behind any of these doors is 1/3.

- Let's say you decide to pick door #1 as the probability is the same (1/3) for each of these doors. In other words, the probability that the car is behind door #1 is 1/3, and the probability that it will be behind either door #2 or door #3 is 2/3.
- Monty is aware of the prize behind each door. He chooses to open door #3 and reveal a goat. He then asks you if you would like to either switch to door #2 or stick with door #1.

To solve the problem, let's switch to Python and apply the Monte Carlo simulation.

Or

(b) Given the network below, calculate Pr (¬p3), Pr (p2|¬p3) using exact inference in Bayesian networks (Variable elimination algorithm). CO2-App (20)



4.	(a)	Apply Deep Boltzmann machine training procedure used to classify the MNIST dataset.	CO2- App	(20)
		Or		
	(b)	Design a network for Real time intra pulse recognition of radar using RBM	CO2- App	(20)
5.	(a)	Apply deep learning model in Computer Vision.	CO2-App	(20)
		Or		
	(b)	Apply deep learning model in Natural Language Processing.	CO2-App	(20)