Week-12, Practice

Common Data

Statement

Common data for questions (1) to (4):

Consider a network that has the following architecture for a multi-class classification problem:

```
[15, 20, 30, 40, 15, 5]
```

The first layer is the input and last layer is the output. The network shall be denoted by h, input vector by $\boldsymbol{x}$ and the output produced by the network by $\hat{\boldsymbol{y}}$.

Question-1

Statement

How many hidden layers does the network have?

Answer

4

Solution

The first element in the list corresponds to the input layer and the last element to the output layer.

Question-2

Statement

How many parameters (weights + biases) does the network have?

Answer

2885

Solution

For the first hidden layer, the number of weights is given by $15 \times 20 = 300$. Since there are 20 neurons in the first hidden layer, there are 20 terms for the bias. Total number of parameters contributed by the first hidden layer is therefore $300 + 20 = 320$. This process has to be repeated for every hidden layer. Finally, since there is a weight matrix and a bias vector associated with the output layer, that also has to be taken into consideration.

Question-3

Statement

What is the value of the following expression? $\mathbf{1}_5$ is a vector of ones.

$$\hat{\boldsymbol{y}}^T \mathbf{1}_5$$

Answer

1

Solution

Probabilities should sum to 1.

Question-4

Statement

What is the shape of the weight matrix at layer 2? Note that zero-indexing is used for the layers.

Options

(a)

15×20

(b)

20×30

(c)

30×40

(d)

40×15

Answer

(b)

Solution

Self-explanatory

Question-5

Statement

Consider a network for multi-class classification that has 10 classes. A data-scientist accidentally replaces all the weights at the final layer with ones and the biases with zeros. He is not aware of this accident and does a forward pass. Which of the following could be the output produced by the network for some random input vector?

Options

(a)

$[1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$

(b)

$[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

(c)

$[0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]$

(d)

$[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]$

Answer

(c)

Solution

If the weights are ones and biases are zero, then all the neurons at the output layer will compute the same value because of the symmetric arrangement. Performing a softmax on top of this will result in equal probabilities for all classes. This is just a uniform distribution over the k classes. In this case, $k = 10$, so each class will have a probability of 0.1.

Question-6

Statement

The following is the activation vector output by some hidden layer in a neural network when some input vector is given to it.

$[1.2, 0.5, 1, 0.8, 0, 0.5]$

Which of the following could be the activation function used in this layer?

Options

(a)

Softmax

(b)

Sigmoid

(c)

ReLU

(d)

Tanh

Answer

(c)

All the values are positive. So, it could either be sigmoid, softmax or relu. But both sigmoid and softmax push values between 0 and 1. So, the only alternative is relu.

Question-7

Statement

Can you use a neural network for a regression problem that has multiple outputs?

Options

(a)

Yes

(b)

No

Answer

(a)

Solution

Yes, it can be used. Two changes would have to be made:

- Have as many neurons in the output layer as there are outputs
- Modify the loss function, say sum of the squared errors for each of the outputs

Question-8

Statement

How many neurons will the final layer have if there are five targets in a multi-output regression problem?

Options

(a)

5

(b)

This question doesn't make sense as a neural network cannot be used as a model in a multi-output regression problem.

Answer

(a)

Solution

Self explanatory

Question-9

Statement

Assume for a moment that you have a neural network for a classification problem with no hidden layers: only input and output layers, with Softmax activation function at the output layer. This neural network is closest to which of the following learning models?

Options

(a)

SVM

(b)

Least squares classification

(c)

Perceptron

(d)

Softmax regression

Answer

(d)

Solution

With just an input and an output layer, we have:

$$\hat{\boldsymbol{Y}} = \text{softmax}(\boldsymbol{XW} + \boldsymbol{b})$$

This is nothing but softmax regression.

Question-10

Statement

If $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ are two vectors, is the following statement true or false? Here, h represents a neural network.

$$h(\boldsymbol{x_1} + \boldsymbol{x_2}) = h(\boldsymbol{x_1}) + h(\boldsymbol{x_1})$$

Options

(a)

True

(b)

False

Answer

(b)

Solution

Neural networks learn non-linear relationships. The equation in this question is linear and hence won't hold true.