Graded Assignment
Shallow Neural Networks

Assignment details

Due

February 2, 11:59 PM +07Feb 2, 11:59 PM +07

Attempts

3 left (3 attempts every 8 hours)

Submitted

January 31, 10:36 PM +07Jan 31, 10:36 PM +07

Time limit

50 minutes per attempt50 min per attempt

Submissions

1 left (1 total within the time limit)

Retry

Your grade

To pass you need at least 80%. We keep your highest score.

88.57%

View submission

See feedback

Like Dislike Report an issue

Back

Shallow Neural Networks

Graded Assignment • 50 min

English

DueFeb 2, 11:59 PM +07

Your grade: 88.57%

Next item

Your latest: 88.57%•
Your highest: 88.57%•
To pass you need at least 80%. We keep your highest score.

1.

Question 1

Which of the following are true? (Check all that apply.)

☑ X is a matrix in which each column is one training example.

✓ Correct

☐ $a_4^{[2]}$ is the activation output of the 2^{nd} layer for the 4^{th} training example

☐ X is a matrix in which each row is one training example.

☑ $a_4^{[2]}$ is the activation output by the 4^{th} neuron of the 2^{nd} layer

✓ Correct

☐ $a^{[2]}$ denotes the activation vector of the 2^{nd} layer.

☐ $a^{[2](12)}$ denotes activation vector of the 12^{th} layer on the 2^{nd} training example.

☑ $a^{[2](12)}$ denotes the activation vector of the 2^{nd} layer for the 12^{th} training example.

✓ Correct
You didn't select all the correct answers
0.8571428571428571 / 1 point

2.

Question 2

The sigmoid function is only mentioned as an activation function for historical reasons. The tanh is always preferred without exceptions in all the layers of a Neural Network. True/False?

False

True

Correct

Yes. Although the tanh almost always works better than the sigmoid function when used in hidden layers, thus is always proffered as activation function, the exception is for the output layer in classification problems.

1 / 1 point

3.

Question 3

Which of these is a correct vectorized implementation of forward propagation for layer l, where $1 \leq l \leq L$?

- $Z^{[l]} = W^{[l]}A^{[l-1]} + b^{[l]}$
- $A^{[l]} = g^{[l]}(Z^{[l]})$

- $Z^{[l]} = W^{[l]}A^{[l]} + b^{[l]}$
- $A^{[l+1]} = g^{[l+1]}(Z^{[l]})$

- $Z^{[l]} = W^{[l]}A^{[l]} + b^{[l]}$
- $A^{[l+1]} = g^{[l]}(Z^{[l]})$

- $Z^{[l]} = W^{[l-1]}A^{[l]} + b^{[l-1]}$
- $A^{[l]} = g^{[l]}(Z^{[l]})$

Correct

1 / 1 point

4.

Question 4

When building a binary classifier for recognizing cats (y=1) vs raccoons (y=0). Is better to use the sigmoid function as activation function for the hidden layers. True/False

◉ False

○ True

⊘ Correct

Yes. Using tanh almost always works better than the sigmoid function for hidden layers.

1 / 1 point

5.

Question 5

Consider the following code:

#+begin_src python

x = np.random.rand(3, 2)

y = np.sum(x, axis=0, keepdims=True)

#+end_src

What will be y.shape?

○ (1, 2)

◉ (3, 1)

○ (3,)

○ (2,)

⊗

Incorrect

No. By choosing the axis=0 the sum is computed over each column of the array, thus the resulting array is a row vector with 2 entries. Since the option keepdims=True is used the first dimension is kept.

0 / 1 point

6.

Question 6

Suppose you have built a neural network. You decide to initialize the weights and biases to be zero. Which of the following statements is true?

Each neuron in the first hidden layer will perform the same computation in the first iteration. But after one iteration of gradient descent they will learn to compute different things because we have “broken symmetry”.

Each neuron in the first hidden layer will perform the same computation. So even after multiple iterations of gradient descent, each neuron in the layer will be computing the same thing as other neurons.

The first hidden layer’s neurons will perform different computations from each other even in the first iteration; their parameters will thus keep evolving in their own way.

Each neuron in the first hidden layer will compute the same thing, but neurons in different layers will compute different things, thus we have accomplished “symmetry breaking” as described in the lecture.

Correct

1 / 1 point

7.

Question 7

Using linear activation functions in the hidden layers of a multilayer neural network is equivalent to using a single layer. True/False?

True

False

Correct

Yes. When the identity or linear activation function $g(c) = c$ is used the output of composition of layers is equivalent to the computations made by a single layer.

1 / 1 point

8.

Question 8

You have built a network using the tanh activation for all the hidden units. You initialize the weights to relatively large values, using np.random.randn(..,..)*1000. What will happen?

This will cause the inputs of the tanh to also be very large, thus causing gradients to be close to zero. The optimization algorithm will thus become slow.

This will cause the inputs of the tanh to also be very large, thus causing gradients to also become large. You therefore have to set α to a very small value to prevent divergence; this will slow down learning.

So long as you initialize the weights randomly gradient descent is not affected by whether the weights are large or small.

This will cause the inputs of the tanh to also be very large, causing the units to be "highly activated" and thus speed up learning compared to if the weights had to start from small values.

Correct

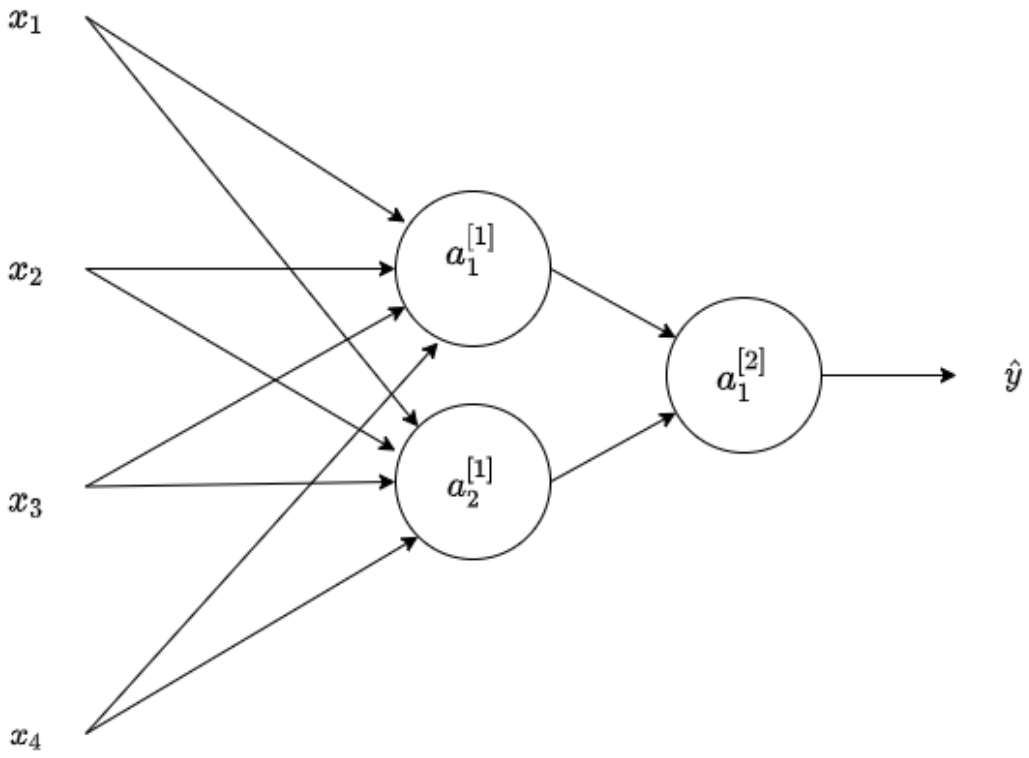
Yes. tanh becomes flat for large values; this leads its gradient to be close to zero. This slows down the optimization algorithm.

1 / 1 point

9.

Question 9

Consider the following 1 hidden layer neural network:



Which of the following statements are True? (Check all that apply).

☑ $b^{[1]}$ will have shape (2, 1).

Correct

Yes. $b^{[k]}$ is a column vector and has the same number of rows as neurons in the k-th layer.

☑ $W^{[2]}$ will have shape (1, 2)

Correct

Yes. The number of rows in $W^{[k]}$ is the number of neurons in the k-th layer and the number of columns is the number of inputs of the layer.

☐ $b^{[1]}$ will have shape (4, 2)

☐ $W^{[2]}$ will have shape (2, 1)

$W^{[1]}$ will have shape (4, 2).

$W^{[1]}$ will have shape (2, 4).

Correct

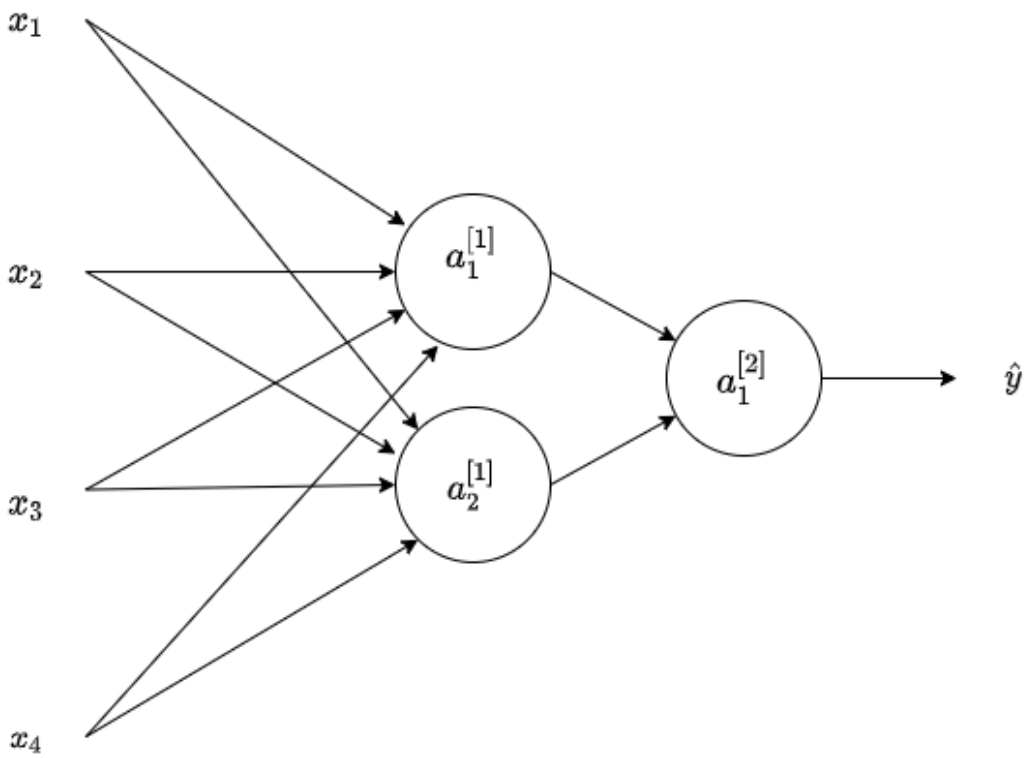
Yes. The number of rows in $W^{[k]}$ is the number of neurons in the k-th layer and the number of columns is the number of inputs of the layer.

1 / 1 point

10.

Question 10

Consider the following 1 hidden layer neural network:



What are the dimensions of $Z^{[1]}$ and $A^{[1]}$?

$Z^{[1]}$ and $A^{[1]}$ are (2, m)

$Z^{[1]}$ and $A^{[1]}$ are (4, 1)

$Z^{[1]}$ and $A^{[1]}$ are (4, m)

$Z^{[1]}$ and $A^{[1]}$ are (2, 1)

Correct

Yes. The $Z^{[1]}$ and $A^{[1]}$ are calculated over a batch of training examples. The number of columns in $Z^{[1]}$ and $A^{[1]}$ is equal to the number of examples in the batch, m. And the number of rows in $Z^{[1]}$ and $A^{[1]}$ is equal to the number of neurons in the first layer.

1 / 1 point