Neural Networks Basics

Congratulations! You passed!

Grade received 100% | Latest Submission Grade 100% | To pass 80% or higher

Go to next item

1. In logistic regression given the input $\mathbf{x}$, and parameters $w \in \mathbb{R}^{n_x}, b \in \mathbb{R}$, how do we generate the output $\hat{y}$? **1 / 1 point**

- (•) $\sigma(W\mathbf{x} + b)$.
- ( ) $\tanh(W\mathbf{x} + b)$
- ( ) $W\mathbf{x} + b$
- ( ) $\sigma(W\mathbf{x})$

Expand

Correct
Right, in logistic regression we use a linear function $W\mathbf{x} + b$ followed by the sigmoid function σ, to get an output y, referred to as ŷ, such that $0 < \hat{y} < 1$.

2. Suppose that $\hat{y} = 0.9$ and $y = 1$. What is the value of the "Logistic Loss"? Choose the best option. **1 / 1 point**

- ( ) 0.005
- ( ) $\mathcal{L}(\hat{y}, y) = -(\hat{y} \log y + (1 - \hat{y}) \log(1 - y))$
- ( ) $+\infty$
- (•) 0.105

Correct

Yes. Since $\mathcal{L}(\hat{y}, y) = -\big(y \log \hat{y} + (1-y) \log(1-\hat{y})\big)$, for the given values we get $\mathcal{L}(\hat{y}, y) = -\big(1 \log 0.9 + 0 \log 0.1\big)$

3. Suppose img is a (32,32,3) array, representing a 32x32 image with 3 color channels red, green and blue. How do you reshape this into a column vector x? **1 / 1 point**

- x = img.reshape((3,32*32))
- x = img.reshape((1,32*32,3))
- x = img.reshape((32*32,3))
- x = img.reshape((32*32*3,1))

Expand

Correct

4. Consider the following random arrays a and b, and c: **1 / 1 point**

$a = np.random.randn(3,3) \# a.shape = (3,3)$

$b = np.random.randn(2,1) \# b.shape = (2,1)$

$c = a + b$

What will be the shape of c?

- The computation cannot happen because it is not possible to broadcast more than one dimension
- c.shape = (2, 3, 3)
- c.shape = (2, 1)
- c.shape = (3,3)

Expand

Correct

Yes. It is not possible to broadcast together a and b. In this case there is no way to generate copies of one of the arrays to match the size of the other.

5. Consider the two following random arrays a and b: **1 / 1 point**

$a = np.random.randn(4, 3)$ # $a.shape = (4, 3)$

$b = np.random.randn(1, 3)$ # $b.shape = (1, 3)$

$c = a * b$

What will be the shape of c?

- (●) c.shape = (4, 3)
- ( ) The computation cannot happen because it is not possible to broadcast more than one dimension.
- ( ) c.shape = (1, 3)
- ( ) The computation cannot happen because the sizes don't match.

Expand

Correct
Yes. Broadcasting is invoked, so row b is multiplied element-wise with each row of a to create c.

6. Suppose you have n_x input features per example. Recall that $X = [x^{(1)}x^{(2)}...x^{(m)}]$. What is the dimension of X? **1 / 1 point**

- (●) (n_x, m)
- ( ) $(m, 1)$
- ( ) (m, n_x)
- ( ) $(1, m)$

Correct

7. Recall that $np.dot(a, b)$ performs a matrix multiplication on a and b, whereas $a * b$ performs an element-wise multiplication. **1 / 1 point**

Consider the two following random arrays a and b:

$a = np.random.randn(12288, 150)$

$\#\, a.shape = (12288, 150)$

$b = np.random.randn(150, 45)$

$\#\, b.shape = (150, 45)$

$c = np.dot(a, b)$

What is the shape of c?

- c.shape = (12288, 150)
- The computation cannot happen because the sizes don't match. It's going to be "Error"!
- c.shape = (150,150)
- c.shape = (12288, 45)

Expand

Correct

Correct, remember that a np.dot(a, b) has shape (number of rows of a, number of columns of b). The sizes match because: "number of columns of a = 150 = number of rows of b"

8. Consider the following code snippet: **1 / 1 point**

$a.shape = (3, 4)$

$b.shape = (4, 1)$

for i in range(3):

for j in range(4):

c[i][j] = a[i][j] + b[j]

How do you vectorize this?

- c = a.T + b
- c = a + b
- c = a.T + b.T
- c = a + b.T

Expand

Correct

9. Consider the following arrays: **1 / 1 point**

$a = np.array([[1, 1], [1, -1]])$

$b = np.array([[2], [3]])$

$c = a + b$

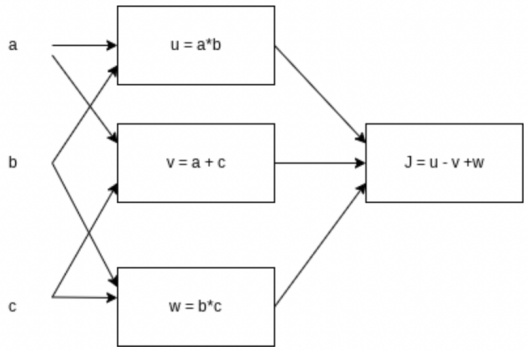
Which of the following arrays is stored in c?

- $\begin{pmatrix} 3 & 3 \\ 3 & 1 \\ 4 & 4 \\ 5 & 2 \end{pmatrix}$
- The computation cannot happen because the sizes don't match. It's going to be an "Error"!
- $\begin{matrix} 3 & 3 \\ 4 & 2 \end{matrix}$
- $\begin{matrix} 3 & 4 \\ 3 & 2 \end{matrix}$

Expand

Correct
Yes. The array b is a column vector. This is copied two times and added to the array a to construct the array c.

10. Consider the following computational graph. **1 / 1 point**



What is the output of J?

- $(a+c),(b-1)$
- $ab+bc+ac$
- $(c-1),(a+c)$
- $(a-1),(b+c)$

Expand

Correct
Yes.
$J = u - v + w = ab - (a + c) + bc = ab - a + bc - c = a\,(b - 1) + c\,(b - 1) = (a + c)\,(b - 1)$