Your grade: 90%

Your latest: **90%** • Your highest: **90%** • To pass you need at least 80%. We keep your highest score.

Next item →

1. In logistic regression given $\mathbf{x}$ and parameters $w \in \mathbb{R}^{n_x}, b \in \mathbb{R}$. Which of the following best expresses what we want $\hat{y}$ to tell us? 1 / 1 point

- $\sigma(W\,\mathbf{x} + b)$
- $P(y = 1|\mathbf{x})$
- $\sigma(W\,\mathbf{x})$
- $P(y = \hat{y}|\mathbf{x})$

Correct
Yes. We want the output $\hat{y}$ to tell us the probability that $y = 1$ given $\mathbf{x}$.

2. Suppose that $\hat{y} = 0.5$ and $y = 0$. What is the value of the "Logistic Loss"? Choose the best option. 1 / 1 point

- 0.693
- $\mathcal{L}(\hat{y}, y) = -\left(y\,\log\hat{y} + (1-y)\;\log(1-\hat{y})\right)$
- $+\infty$
- 0.5

Correct
Yes. Given the values of $\hat{y}$ and y we get $\mathcal{L}(0.5, 0) = -\left(0\,\log 0.5 + 1\,\log(0.5)\right) \approx 0.693$.

3. Consider the Numpy array x: 1 / 1 point

$x = np.array([[[1], [2]], [[3], [4]]])$

What is the shape of x?

- (4,)
- (2, 2)
- (1, 2, 2)
- (2,2,1)

Correct
Yes. This array has two rows and in each row it has 2 arrays of 1x1.

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

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

$b = np.random.randn(1, 4)\ \#\ b.shape = (1, 4)$

$c = a + b$

What will be the shape of c?

- The computation cannot happen because it is not possible to broadcast more than one dimension.
- c.shape = (3, 4)
- c.shape = (3, 1)
- c.shape = (1, 4)

Correct
Yes. Broadcasting is used, so row b is copied 3 times so it can be summed to each row of a.

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

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

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

$c = a * b$

What will be the shape of c?

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

Correct
Yes. Broadcasting allows row a to be multiplied element-wise with each row of b to from 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

- ○ (m, n_x)
- ○ $(1, m)$
- ◉ (n_x, m)
- ○ $(m, 1)$

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?

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

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: 0 / 1 point

$a.shape = (4, 3)$

$b.shape = (4, 1)$

for i in range(3):

for j in range(4):

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

How do you vectorize this?

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

Incorrect
No. Notice that b is a column vector; but we are using it to fill the row i of c.

9. Consider the code snippet: 1 / 1 point

$a.shape = (3, 3)$

$b.shape = (3, 3)$

$c = a * *2 + b.T * *2$

Which of the following gives an equivalent output for *c*?

- The computation cannot happen because the sizes don't match. It's going to be an "Error"!
- ✓

```
for i in range(3):
    for j in range(3):
        c[i][j] = a[i][j]**2 + b[j][i]**2
```

-

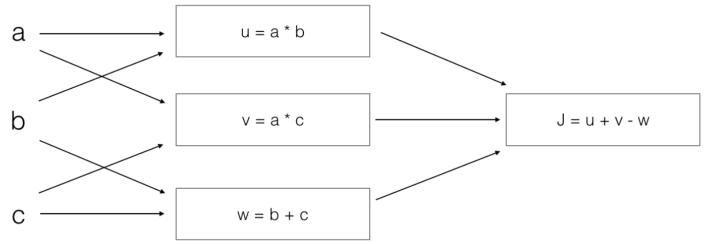
```
for i in range(3):
    for j in range(3):
        c[i][j] = a[i][j]**2 + b[i][j]**2
```

-

```
for i in range(3):
    c[i] = a[i]**2 + b[i]**2
```

Correct
Yes. This code squares each entry of a and adds it to the transpose of b square.

10. Consider the following computation graph. 1 / 1 point



What is the output J?

- ✓ $J = (a - 1) * (b + c)$
- $J = (c - 1) * (b + a)$
- $J = a * b + b * c + a * c$
- $J = (b - 1) * (c + a)$

Correct
Yes.
$J = u + v - w = a * b + a * c - (b + c) = a * (b + c) - (b + c) = (a - 1) * (b + c)$.