

Fonction de coût simplifiée pour la régression logistique

As a reminder, here is the loss function that we had defined in the previous video for logistic regression. Because we're still working on a binary classification problem, y is either zero or one. Because y is either zero or one and cannot take on any value other than zero or one, we'll be able to come up with a simpler way to write this loss function. You can write the loss function as follows.

Given a prediction f of x and the target label y , the loss equals negative y times log of f minus 1 minus y times log of 1 minus f . It turns out this equation, which we just wrote in one line, is completely equivalent to this more complex formula up here. Let's see why this is the case. Remember, y can only take on the values of either one or zero.

Simplified loss function

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = \begin{cases} -\log(f_{\bar{w}, b}(\vec{x}^{(i)})) & \text{if } y^{(i)} = 1 \\ -\log(1 - f_{\bar{w}, b}(\vec{x}^{(i)})) & \text{if } y^{(i)} = 0 \end{cases}$$

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = -y^{(i)} \log(f_{\bar{w}, b}(\vec{x}^{(i)})) - (1 - y^{(i)}) \log(1 - f_{\bar{w}, b}(\vec{x}^{(i)}))$$

In the first case, let's say y equals 1. This first y over here is one and this 1 minus y is 1 minus 1, which is therefore equal to 0. So the loss becomes negative 1 times log of f of x minus 0 times a bunch of stuff. That becomes zero and goes away. When y is equal to 1, the loss is indeed the first term on top, negative log of f of x .

Simplified loss function

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = \begin{cases} -\log(f_{\bar{w}, b}(\vec{x}^{(i)})) & \text{if } y^{(i)} = 1 \\ -\log(1 - f_{\bar{w}, b}(\vec{x}^{(i)})) & \text{if } y^{(i)} = 0 \end{cases}$$

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = -y^{(i)} \log(f_{\bar{w}, b}(\vec{x}^{(i)})) - (1 - y^{(i)}) \log(1 - f_{\bar{w}, b}(\vec{x}^{(i)}))$$

$$\text{if } y^{(i)} = 1: \quad \frac{1}{\cancel{\log(f_{\bar{w}, b}(\vec{x}^{(i)}))}} \quad \frac{(1-1)}{\cancel{\log(1 - f_{\bar{w}, b}(\vec{x}^{(i)}))}}$$

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = \underbrace{-1}_{\text{if } y^{(i)} = 1} \underbrace{\log(f_{\bar{w}, b}(\vec{x}^{(i)}))}_{\text{if } y^{(i)} = 0}$$

Let's look at the second case, when y is equal to 0. In this case, this y here is equal to 0, so this first term goes away, and the second term is 1 minus 0 times that logarithmic term. The loss becomes this negative 1 times log of 1 minus f of x . That's just equal to this second term up here. In the case of y equals 0, we also get back the original loss function as defined above.

What you see is that whether y is one or zero, this single expression here is equivalent to the more complex expression up here, which is why this gives us a simpler way to write the loss with just one equation without separating out these two cases, like we did on top.

Using this simplified loss function, let's go back and write out the cost function for logistic regression.

Simplified loss function

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = \begin{cases} -\log(f_{\bar{w}, b}(\vec{x}^{(i)})) & \text{if } y^{(i)} = 1 \\ -\log(1 - f_{\bar{w}, b}(\vec{x}^{(i)})) & \text{if } y^{(i)} = 0 \end{cases}$$

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = -y^{(i)} \log(f_{\bar{w}, b}(\vec{x}^{(i)})) - (1 - y^{(i)}) \log(1 - f_{\bar{w}, b}(\vec{x}^{(i)}))$$

if $y^{(i)} = 1$:

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = -\log(f_{\bar{w}, b}(\vec{x}))$$

if $y^{(i)} = 0$:

$$L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = - (1 - \log(1 - f_{\bar{w}, b}(\vec{x})))$$

Here again is the simplified loss function. Recall that the cost J is just the average loss, average across the entire training set of m examples. So it's 1 over n times the sum of the loss from i equals 1 to m . If you plug in the definition for the simplified loss from above, then it looks like this, 1 over m times the sum of this term above. If you bring the negative signs and move them outside, then you end up with this expression over here, and this is the cost function.

The cost function that pretty much everyone uses to train logistic regression. You might be wondering, why do we choose this particular function when there could be tons of other costs functions we could have chosen? Although we won't have time to go into great detail on this in this class, I'd just like to mention that this particular cost function is derived from statistics using a statistical principle called maximum likelihood estimation, which is an idea from statistics on how to efficiently find parameters for different models.

This cost function has the nice property that it is convex. But don't worry about learning the details of maximum likelihood. It's just a deeper rationale and justification behind this particular cost function.

Simplified cost function

$$\text{loss} \quad L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)}) = -y^{(i)} \log(f_{\bar{w}, b}(\vec{x}^{(i)})) - (1 - y^{(i)}) \log(1 - f_{\bar{w}, b}(\vec{x}^{(i)}))$$

$$\text{cost} \quad J(\bar{w}, b) = \frac{1}{m} \sum_{i=1}^m [L(f_{\bar{w}, b}(\vec{x}^{(i)}), y^{(i)})]$$

convex
(single global minimum)

$$= -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log(f_{\bar{w}, b}(\vec{x}^{(i)})) + (1 - y^{(i)}) \log(1 - f_{\bar{w}, b}(\vec{x}^{(i)}))]$$

maximum likelihood
(don't worry about it!)