NEURAL NETWORKS AND DEEP LEARNING CST 395 CS 5TH SEMESTER HONORS COURSE- Dr Binu V P, 9847390760

CS 5th Semester Honors course for the Computer Science at KTU- Dr Binu V P

Differentiation of the Sigmoid activation and cross-entropy loss function

August 20, 2022

A step-by-step differentiation of the Sigmoid activation and cross-entropy loss function is discussed here.
The understanding of derivatives of these two functions is essential in the area of machine learning when performing back-propagation during model training.

Derivative of Sigmoid Function

Sigmoid/ Logistic function is defined as:
$g(x) = \frac{1}{1+e^{-x}} \in (0, 1)$

For any value of x, the Sigmoid function $g(x)$ falls in the range $(0, 1)$. As the value of x decreases, g(x) approaches 0, whereas as x grows bigger, $g(x)$tends to 1. Examples,

$g(-5.5) = 0.0040$
$g(6.5) = 0.9984$
$g(0.4) = 0.5986$

$$\frac{dg}{dx} = \frac{d}{dx}\left(\frac{1}{1+e^{-x}}\right) = \frac{vu^{'} - uv^{'}}{v^2}$$
$$= \frac{(0)(1-e^{-x}) - 1(-e^{-x})}{(1-e^{-x})^2}$$
$$= \frac{e^{-x}}{(1-e^{-x})^2}$$

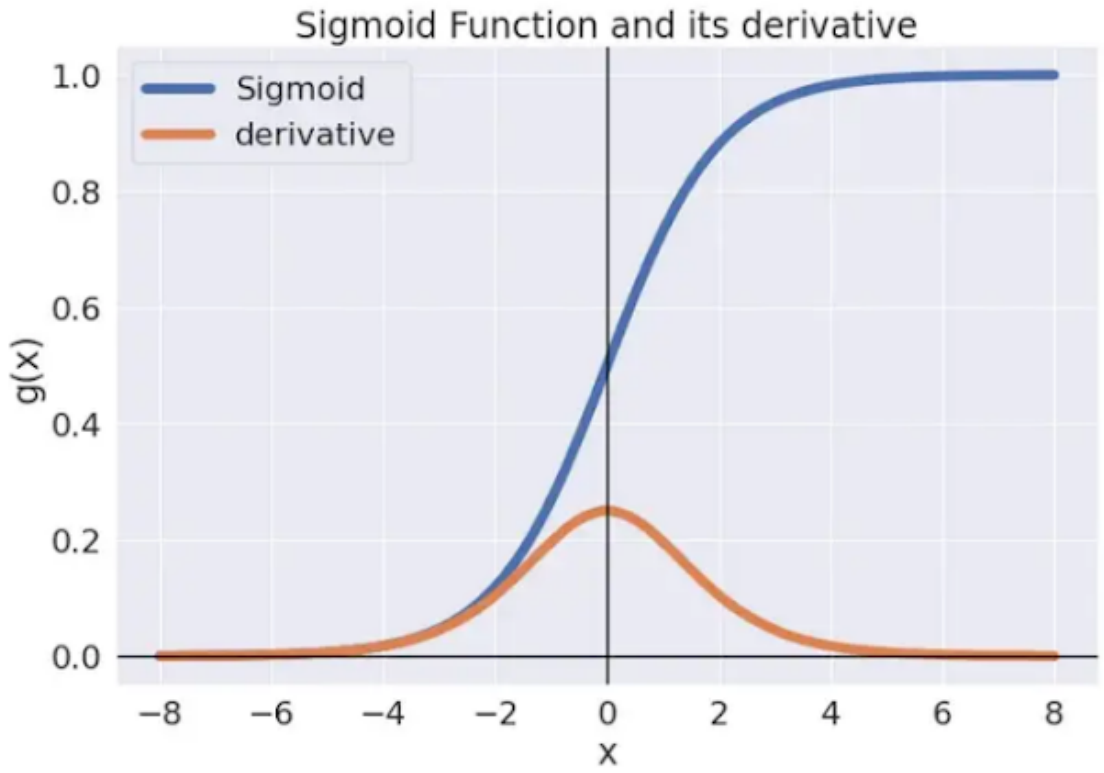
It is noted that we can further simplify the derivative and write in term of g(x)

$$\frac{dg}{dx} = \frac{d}{dx}\left(\frac{1}{1+e^{-x}}\right) = \frac{e^{-x}}{(1+e^{-x})^2}$$
$$= \frac{1}{1+e^{-x}} \cdot \frac{e^{-x}}{1+e^{-x}}$$
$$= \frac{1}{1+e^{-x}} \cdot \frac{1+e^{-x}-1}{1+e^{-x}}$$
$$= \frac{1}{1+e^{-x}} \cdot \frac{\cancel{1+e^{-x}}}{\cancel{1+e^{-x}}} - \frac{1}{1+e^{-x}}$$
$$= \frac{1}{1+e^{-x}} \cdot 1 - \frac{1}{1+e^{-x}}$$
$$\therefore \frac{dg}{dx} = g(x)(1-g(x))$$

Why do we use this version of the derivative?

In the forward propagation step, you compute the sigmoid function $(g(x))$ and have its value handy. While computing the derivative in the backpropagation step, all you have to do is plug in the value of $g(x)$ in the formula derived above.

Here is the plot of sigmoid function and its derivative



Derivative of Cross-Entropy Function

Cross-Entropy loss function is a very important cost function used for classification problems.The concept of cross-entropy traces back into the field of Information Theory where Claude Shannon introduced the concept of entropy in 1948. Before diving into Cross-Entropy cost function, let us introduce entropy .

Entropy

Entropy of a random variable X is the level of uncertainty inherent in the variables possible outcome.

For $p(x)$ — probability distribution and a random variable X, entropy is defined as follows

$$H(X) = \begin{cases} -\int_x p(x) \log p(x), & \text{if } X \text{ is continous} \\ -\sum_x p(x) \log p(x), & \text{if } X \text{ is discrete} \end{cases}$$

Cross-Entropy Loss Function is also called **logarithmic loss, log loss** or **logistic loss.**

Each predicted class probability is compared to the actual class desired output 0 or 1 and a score/loss is calculated that penalizes the probability based on how far it is from the actual expected value. The penalty is logarithmic in nature yielding a large score for large differences close to 1 and small score for small differences tending to 0.

Cross-entropy loss is used when adjusting model weights during training. The aim is to minimize the loss, i.e, the smaller the loss the better the model. A perfect model has a cross-entropy loss of 0.

Cross-entropy is defined as

$E_g = -\sum_{i=1}^{n} t_i ln(p_i)$
where t_i is the truth value and p_i is the probability of the i^{th} class.

For classification with two classes, we have binary cross-entropy loss which is defined as follows

$$E_b = -[t_n ln(\bar{y}) + (1-t) ln(1-\hat{y})]$$

The truth label, t, on the binary loss is a known value, whereas $\hat{y}$ is a variable. This means that the function will be differentiated with respect to $\hat{y}$ and treat t as a constant. Let's go ahead and work on the derivative now.

$$\begin{aligned}\frac{\partial}{\partial \hat{y}}(t \ln \hat{y}) &= t\frac{\partial}{\partial \hat{y}}(\ln \hat{y}) + \ln(\hat{y})\frac{\partial}{\partial \hat{y}}(t) \\ &= \frac{t}{\hat{y}} + \ln \hat{y}(0) \\ &= \frac{t}{\hat{y}}\end{aligned}$$

$$\begin{aligned}\frac{\partial}{\partial \hat{y}}((1-t)\ln(1-\hat{y})) &= (1-t)\frac{\partial}{\partial \hat{y}}(\ln(1-\hat{y})) + \ln((1-\hat{y}))\frac{\partial}{\partial \hat{y}}(1-t) \\ &= -\frac{1-t}{1-\hat{y}} + \ln(1-\hat{y})(0) \\ &= -\frac{1-t}{1-\hat{y}}\end{aligned}$$

And therefore, the derivative of the binary cross-entropy loss function becomes

$$\therefore \frac{\partial E_b}{\partial \hat{y}} = -\left(\frac{t}{\hat{y}} - \frac{1-t}{1-\hat{y}}\right)$$
$$= \left(\frac{t}{\hat{y}} + \frac{1-t}{1-\hat{y}}\right)$$

Partial Derivative of cost function for linear regression

First, here are the elements in the hypothesis:

- x is an mxn matrix containing one row for each training record and one column for each feature
- θ is a 1xn matrix containing the weights for the hypothesis
- y is an mx1 matrix containing the response values

So n represents the number of features in the training set and m represents the number of rows in the training set.

The hypothesis is:

$$h_\theta = \theta^T x$$

This is the cost function in its simplest form (the superscript i's refer to a particular feature):

$$J(\theta) = \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

This simply gives the total of the squared errors. To get the mean of the squared errors:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

We can also multiply this by $\frac{1}{2}$ for a purely arbitrary reason, to make the derivative easier to calculate, as we'll see below. Note that this is *not* now the mean of the squared errors, but rather one-half of the mean, but this is not important, because we will only be comparing various calculations of one-half the mean against each other.

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

Now we can take the partial derivative for any one feature in x.

$$\frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 = ?$$

First, the $\frac{1}{2m}$ is outside of the $\sum$, so it acts as a constant that will not change. So we need to take the partial derivative of everything inside the $\sum$. We have a quantity that is squared, so the first step is to apply the chain rule, which effectively means:

$$\frac{\partial}{\partial_{\text{stuff}}} (\text{stuff})^2 = 2(\text{stuff}) \times \frac{\partial}{\partial_{\text{stuff}}} \text{stuff}$$

So:

$$\frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 = \frac{1}{2m} \sum_{i=1}^{m} 2(h_\theta(x^{(i)}) - y^{(i)}) \times \frac{\partial}{\partial \theta_j} h_\theta(x^{(i)})$$

We know from above that the function h_θ is:

$$h_\theta = \theta^T x$$

So:

$$\frac{\partial}{\partial \theta_j} h_\theta = \frac{\partial}{\partial \theta_j} \theta_j^T x_j$$

Here, x_j is a constant, and the derivative of θ_j with respect to θ_j is 1, so we are left with:

$$\frac{\partial}{\partial \theta_j} h_\theta = x_j$$

Plug this back into the above equation and we have:

$$\frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 = \frac{1}{2m} \sum_{i=1}^{m} 2(h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)}$$

As the final step, the 2 in the denominator outside of the $\sum$ and the 2 inside the $\sum$ cancel each other out, leaving:

$$\frac{\partial}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)}$$

Partial Derivative of Cost Function for Logistic Regression

Cost function

The cost function for logistic regression is:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))$$

Minimized cost function

This is what we know . to be the partial derivative of the cost function. Can we find it?

$$\frac{\partial}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

Hypothesis (sigmoid) function

$$h_\theta(x^{(i)}) = \frac{1}{1 + e^{-\theta^T x}}$$

Rules for logarithmic expressions

$$\log\left(\frac{x}{y}\right) = \log(x) - \log(y)$$

$$\log(e^a) = a$$

Finding the partial derivatives for each j in θ

1. Simplify the cost function

$$J(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\left(y^{(i)}(\log h_\theta(x^{(i)})) + (1-y^{(i)})\log(1-h_\theta(x^{(i)}))\right)\right]$$

Replace $h_\theta(x^{(i)})$ with sigmoid

$$= -\frac{1}{m}\left[\sum_{i=1}^{m}\left(y^{(i)}\log(\frac{1}{1+e^{-\theta^T x^{(i)}}}) + (1-y^{(i)})\log(1-\frac{1}{1+e^{-\theta^T x^{(i)}}})\right)\right]$$

Convert right term to single rational expression

$$= -\frac{1}{m}\left[\sum_{i=1}^{m}\left(y^{(i)}\log(\frac{1}{1+e^{-\theta^T x^{(i)}}}) + (1-y^{(i)})\log(\frac{e^{-\theta^T x^{(i)}}}{1+e^{-\theta^T x^{(i)}}})\right)\right]$$

Apply $\log(\frac{a}{b}) = \log(a) - \log(b)$ on left term

$$= -\frac{1}{m}\left[\sum_{i=1}^{m}\left(y^{(i)}(\log(1) - \log(1+e^{-\theta^T x^{(i)}})) + (1-y^{(i)})\log(\frac{e^{-\theta^T x^{(i)}}}{1+e^{-\theta^T x^{(i)}}})\right)\right]$$

$$= -\frac{1}{m}\left[\sum_{i=1}^{m}\left(-y^{(i)}\log(1+e^{-\theta^T x^{(i)}}) + (1-y^{(i)})\log(\frac{e^{-\theta^T x^{(i)}}}{1+e^{-\theta^T x^{(i)}}})\right)\right]$$

Apply $\log(\frac{a}{b}) = \log(a) - \log(b)$ to right term

$$= -\frac{1}{m}\left[\sum_{i=1}^{m}\left(-y^{(i)}\log(1+e^{-\theta^T x^{(i)}}) + (1-y^{(i)})\log(e^{-\theta^T x^{(i)}}) - (1-y^{(i)})(\log(1+e^{-\theta^T x^{(i)}}))\right)\right]$$

Apply $\log(e^a) = a$ to right term

$$= -\frac{1}{m}\left[\sum_{i=1}^{m}\left(-y^{(i)}\log(1+e^{-\theta^T x^{(i)}}) + (1-y^{(i)})(-\theta^T x^{(i)}) - (1-y^{(i)})(\log(1+e^{-\theta^T x^{(i)}}))\right)\right]$$

Move minus sign inside $\sum$

$$= \frac{1}{m}\left[\sum_{i=1}^{m}\left(y^{(i)}\log(1+e^{-\theta^T x^{(i)}}) + (1-y^{(i)})(\theta^T x^{(i)}) + (1-y^{(i)})(\log(1+e^{-\theta^T x^{(i)}}))\right)\right]$$

Combine first and third terms

$$= \frac{1}{m}\left[\sum_{i=1}^{m}\left(\log(1+e^{-\theta^T x^{(i)}}) + (1-y^{(i)})(\theta^T x^{(i)})\right)\right]$$

2. Take the partial derivative

$$\frac{\partial}{\partial \theta_j} J(\theta) = \frac{1}{m}\left[\sum_{i=1}^{m}\left(\frac{e^{-\theta^T x^{(i)}}(-x_j^{(i)})}{1+e^{-\theta^T x^{(i)}}} + (1-y^{(i)})x_j^{(i)}\right)\right]$$

Now factor out $x^{(i)}{}_j$

$$= \frac{1}{m}\left[\sum_{i=1}^{m}\left(\frac{-e^{-\theta^T x^{(i)}}}{1+e^{-\theta^T x^{(i)}}} + 1 - y^{(i)}\right)x_j^{(i)}\right]$$

Combine first two terms

$$= \frac{1}{m}\left[\sum_{i=1}^{m}\left(\frac{1}{1+e^{-\theta^T x^{(i)}}} - y^{(i)}\right)x_j^{(i)}\right]$$

Substitute $h_\theta(x^{(i)})$ for sigmoid function

$$= \frac{1}{m}\left[\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)}\right]$$

Popular posts from this blog

NEURAL NETWORKS AND DEEP LEARNING CST 395 CS 5TH SEMESTER HONORS COURSE NOTES - Dr Binu V P, 9847390760

October 03, 2022

About Me Syllabus Question Paper Dec 2022 Module 1 (Basics of Machine Learning) Overview of Machine Learning Machine Learning Algorithm Linear Regression Capacity, Overfitting and Underfitting Regularization Hyperparameters and Validatioı ...

READ MORE

...

Machine Learning Algorithm

October 01, 2022

A machine learning algorithm is an algorithm that is able to learn from data. But what do we mean by learning? Mitchell (1997) provides the definition "A computer program is said to learn from experience E with respect to some class of tasks T and ...

READ MORE

Syllabus

October 01, 2022

Syllabus Module - 1 (Basics of Machine Learning) Machine Learning basics - Learning algorithms - Supervised, Unsupervised, Reinforcement, overfitting, Underfitting, Hyper parameters and Validation sets, Estimators -Bias and Variance. Challenges ...

READ MORE