

# 1 Gradient Descent

$$dC(\omega) = \lim_{\varepsilon \rightarrow 0} \frac{C(\omega + \varepsilon) - C(\omega)}{\varepsilon} \quad (1)$$

(2)

## 1.1 Derivative Of One Parameter Function

Within the *Twice* example we described a model with one parameter -  $w$

The formula had a form like this:

$$f(x) = x \cdot w \quad (3)$$

Function  $C$  which takes one parameter  $w$  is defined as:

$$C(w) = \frac{1}{n} \sum_{i=1}^n (x_i \cdot w - y_i)^2 \quad (4)$$

Let's compute the derivative  $C'$  of our function:

$$C'(w) = (C)' \quad (5)$$

$$= \left( \frac{1}{n} \sum_{i=1}^n (x_i \cdot w - y_i)^2 \right)' = \quad (6)$$

$$= \left( \frac{1}{n} \sum_{i=1}^n (x_i \cdot w - y_i)^2 \right)' = \quad (7)$$

$$= \frac{1}{n} \left( \sum_{i=1}^n (x_i \cdot w - y_i)^2 \right)' = \quad (8)$$

$$= \frac{1}{n} \sum_{i=1}^n ((x_i \cdot w - y_i)^2)' = \quad (9)$$

$$= \frac{1}{n} \sum_{i=1}^n (2 \cdot (x_i \cdot w - y_i)(x_i \cdot w - y_i)') = \quad (10)$$

$$= \frac{1}{n} \sum_{i=1}^n (2 \cdot (x_i \cdot w - y_i) \cdot x_i) \quad (11)$$

The final form of our derivative:

$$C'(w) = \frac{1}{n} \sum_{i=1}^n (2 \cdot (x_i \cdot w - y_i) \cdot x_i) \quad (12)$$

## 1.2 One Neuron Model With 2 Inputs

One neuron model is defined as:

$$z = \sigma(x \cdot w_1 + y \cdot w_2 + b) \quad (13)$$

$x_1$  ... input parameter

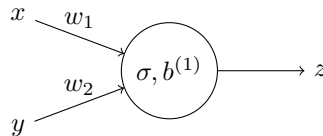
$x_2$  ... input parameter

$w_1$  ... weight parameter

$w_2$  ... weight parameter

$b$  ... bias parameter

$\sigma$  ... sigmoid activation function



### 1.2.1 Cost

Let's recall the Sigmoid activation function

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (14)$$

$$\sigma(x)' = \sigma(x) \cdot (1 - \sigma(x)) \quad (15)$$

Let's define the cost function  $C$  for our model

$$a_i = \sigma(x_i \cdot w_1 + y_i \cdot w_2 + b) \quad (16)$$

$$C(x) = \frac{1}{n} \sum_{i=1}^n (a_i - z_i)^2 \quad (17)$$

Let's compute the derivative  $C'$  for our function

We have to modify TWO parameters  $w, b$

For this we will use PARTIAL DERIVATIVES this means that first we compute a derivative in respect to  $w_1, w_2$  and then we compute another derivative in respect to  $b$

1. Partial Derivative in respect to  $w_1$

$$a_i = \sigma(x_i \cdot w_1 + y_i \cdot w_2 + b) = \quad (18)$$

$$\partial_{w_1} a_i = \partial_{w_1} (\sigma(x_i \cdot w_1 + y_i \cdot w_2 + b)) = \quad (19)$$

$$= a_i(1 - a_i) \partial_{w_1} (x_i \cdot w_1 + y_i \cdot w_2 + b) = \quad (20)$$

$$\partial_{w_1} a_i = a_i(1 - a_i) \cdot x_i \quad (21)$$

$$(22)$$

$$\partial_{w_1} C = \partial_{w_1} \left( \frac{1}{n} \sum_{i=1}^n (a_i - z_i)^2 \right) = \quad (23)$$

$$= \frac{1}{n} \sum_{i=1}^n \partial_{w_1} ((a_i - z_i)^2) = \quad (24)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \partial_{w_1} (a_i - z_i) = \quad (25)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \partial_{w_1} a_i = \quad (26)$$

$$\partial_{w_1} C = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \cdot a_i(1 - a_i) \cdot x_i \quad (27)$$

$$(28)$$

## 2. Partial Derivative in respect to $w_2$

$$a_i = \sigma(x_i \cdot w_1 + y_i \cdot w_2 + b) = \quad (29)$$

$$\partial_{w_2} a_i = \partial_{w_2} (\sigma(x_i \cdot w_1 + y_i \cdot w_2 + b)) = \quad (30)$$

$$= a_i(1 - a_i) \partial_{w_2} (x_i \cdot w_1 + y_i \cdot w_2 + b) = \quad (31)$$

$$\partial_{w_2} a_i = a_i(1 - a_i) \cdot y_i \quad (32)$$

$$(33)$$

$$\partial_{w_2} C = \partial_{w_2} \left( \frac{1}{n} \sum_{i=1}^n (a_i - z_i)^2 \right) = \quad (34)$$

$$= \frac{1}{n} \sum_{i=1}^n \partial_{w_2} ((a_i - z_i)^2) = \quad (35)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \partial_{w_2} (a_i - z_i) = \quad (36)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \partial_{w_2} a_i = \quad (37)$$

$$\partial_{w_2} C = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \cdot a_i(1 - a_i) \cdot y_i \quad (38)$$

$$(39)$$

### 3. Partial Derivative in respect to $b$

$$a_i = \sigma(x_i \cdot w_1 + y_i \cdot w_2 + b) \quad (40)$$

$$\partial_b a_i = \partial_b (\sigma(x_i \cdot w_1 + y_i \cdot w_2 + b)) = \quad (41)$$

$$= a_i(1 - a_i) \partial_b (x_i \cdot w_1 + y_i \cdot w_2 + b) = \quad (42)$$

$$= a_i(1 - a_i) \cdot 1 = \quad (43)$$

$$\partial_b a_i = a_i(1 - a_i) \quad (44)$$

$$(45)$$

$$\partial_b C = \partial_b \left( \frac{1}{n} \sum_{i=1}^n (a_i - z_i)^2 \right) = \quad (46)$$

$$= \frac{1}{n} \sum_{i=1}^n \partial_b ((a_i - z_i)^2) = \quad (47)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \partial_b (a_i - z_i) = \quad (48)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \partial_b a_i = \quad (49)$$

$$\partial_b C = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \cdot a_i(1 - a_i) \quad (50)$$

$$(51)$$

To summarize the partial derivatives are:

$$a_i = \sigma(x_i \cdot w_1 + y_i \cdot w_2 + b) \quad (52)$$

$$\partial_{w_1} C = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \cdot a_i(1 - a_i) \cdot x_i \quad (53)$$

$$\partial_{w_2} C = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \cdot a_i(1 - a_i) \cdot y_i \quad (54)$$

$$\partial_b C = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i - z_i) \cdot a_i(1 - a_i) \quad (55)$$

$$(56)$$

### 1.2.2 Execution Time Comparison

Let's compare computation time difference between **Finite Difference** and **Gradient Descent**

My machine is Lenovo Legion Slim 5:

- All computations are run on the CPU
- CPU: AMD Rayzen 7 7840HS (16) 5.137Ghz

The test:

- Neural network will try to learn the proper configuration for simulating NAND gate
- Comparison of training the model using the *Finite Difference* method and *Gradient Descent*
- 8.000.000 iterations(epochs) of training will be run (overkill I know)

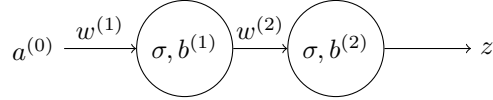
#### RESULTS:

**Finite Difference** :  $\approx 1.556$  *seconds*

**Gradient Descent** :  $\approx 0.473$  *seconds*

Let's not forget that NAND gate simulation is preatty much trivial and both methods of computation would have approximatly the same time when not doing as much iterations(epochs) of training

### 1.3 Two Neuron Model And 1 Input



Let's define the mathematical model

$$a^{(1)} = \sigma(x \cdot w^{(1)} + b^{(1)}) \quad (57)$$

$$a^{(2)} = \sigma(a^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (58)$$

$$(59)$$

$$a^{(i)} \dots \text{activation of the i-th layer} \quad (60)$$

$$(61)$$

$$z = a^{(2)} \quad (62)$$

$$z = \sigma(a^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (63)$$

$$(64)$$

### 1.4 Cost

#### 1.4.1 Cost Of The 2nd Layer

Let's define the cost function  $C^{(2)}$  for the second layer of our model:

$$a_i^{(1)} = \sigma(x_i \cdot w^{(1)} + b^{(1)}) \quad (65)$$

$$a_i^{(2)} = \sigma(a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (66)$$

$$(67)$$

$$a_i^{(l)} \dots \text{activation of the i-th sample of the l-th layer} \quad (68)$$

$$(69)$$

$$C^{(2)}(x) = \frac{1}{n} \sum_{i=1}^n (a_i^{(2)} - z_i)^2 \quad (70)$$

Before we start computing derivatives let's think about them

- Firstly we have to compute partial derivatives for  $w^{(2)}$  and  $b^{(2)}$  which should not be hard because we already covered similar calculations in the past
- When we try to compute partial derivatives inner  $w^{(1)}$  and  $b^{(1)}$  we notice that the parameters  $w^{(1)}$  and  $b^{(1)}$  are deeply nested inside  $a^{(1)}$  which can present a challenge when trying to compute partial derivatives
- Introduce a separate cost functions for each individual layer
  - Alexey Kutepov

- For each of these *specialized* layers you compute the cost only for the variables that are nearly accessible
- Let's treat the *previous activation* as a variable of the cost function and let's differentiate it
- The result of the differentiation of the cost function is actually an *difference(error)* that we can use for the computation of the *Difference(error)* of the inner layer
- We continue to compute these *differances(errors)* for all layers all the way back to the *input layer* => This is where the idea of **back-propagation** comes to play

Let's compute the derivative  $C^{(2)'} of our function$

Partial derivative in regards to  $w^{(2)}$

$$a_i^{(2)} = \sigma(a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (71)$$

$$\partial_{w^{(2)}} a_i^{(2)} = \partial_{w^{(2)}} \sigma(a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (72)$$

$$= a_i^{(1)} (1 - a_i^{(1)}) \cdot \partial_{w^{(2)}} (a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (73)$$

$$\partial_{w^{(2)}} a_i^{(2)} = a_i^{(1)} (1 - a_i^{(1)}) \cdot a_i^{(1)} \quad (74)$$

$$(75)$$

$$(76)$$

$$\partial_{w^{(2)}} C^{(2)} = \partial_{w^{(2)}} \left( \frac{1}{n} \sum_{i=1}^n (a_i^{(2)} - z_i)^2 \right) \quad (77)$$

$$= \frac{1}{n} \sum_{i=1}^n \partial_{w^{(2)}} (a_i^{(2)} - z_i)^2 \quad (78)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot \partial_{w^{(2)}} (a_i^{(2)} - z_i) \quad (79)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot \partial_{w^{(2)}} a_i^{(2)} \quad (80)$$

$$\partial_{w^{(2)}} C^{(2)} = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot a_i^{(1)} (1 - a_i^{(1)}) \cdot a_i^{(1)} \quad (81)$$

$$(82)$$



Partial derivative in regards to  $b^{(2)}$

$$a_i^{(2)} = \sigma(a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (83)$$

$$\partial_{b^{(2)}} a_i^{(2)} = \partial_{b^{(2)}} \sigma(a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (84)$$

$$= a_i^{(1)} (1 - a_i^{(1)}) \cdot \partial_{b^{(2)}} (a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (85)$$

$$\partial_{b^{(2)}} a_i^{(2)} = a_i^{(1)} (1 - a_i^{(1)}) \quad (86)$$

$$(87)$$

$$(88)$$

$$\partial_{b^{(2)}} C^{(2)} = \partial_{b^{(2)}} \left( \frac{1}{n} \sum_{i=1}^n (a_i^{(2)} - z_i)^2 \right) \quad (89)$$

$$= \frac{1}{n} \sum_{i=1}^n \partial_{b^{(2)}} (a_i^{(2)} - z_i)^2 \quad (90)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot \partial_{b^{(2)}} (a_i^{(2)} - z_i) \quad (91)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot \partial_{b^{(2)}} a_i^{(2)} \quad (92)$$

$$\partial_{b^{(2)}} C^{(2)} = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot a_i^{(1)} (1 - a_i^{(1)}) \quad (93)$$

$$(94)$$

Partial derivative in regards to  $a_i^{(1)}$

$$a_i^{(2)} = \sigma(a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (95)$$

$$\partial_{a_i^{(1)}} a_i^{(2)} = \partial_{a_i^{(1)}} \sigma(a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (96)$$

$$= a_i^{(1)}(1 - a_i^{(1)}) \cdot \partial_{a_i^{(1)}} (a_i^{(1)} \cdot w^{(2)} + b^{(2)}) \quad (97)$$

$$\partial_{a_i^{(1)}} a_i^{(2)} = a_i^{(1)}(1 - a_i^{(1)}) \cdot w^{(2)} \quad (98)$$

$$(99)$$

$$(100)$$

$$\partial_{a_i^{(1)}} C^{(2)} = \partial_{a_i^{(1)}} \left( \frac{1}{n} \sum_{i=1}^n (a_i^{(2)} - z_i)^2 \right) \quad (101)$$

$$= \frac{1}{n} \sum_{i=1}^n \partial_{a_i^{(1)}} (a_i^{(2)} - z_i)^2 \quad (102)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot \partial_{a_i^{(1)}} (a_i^{(2)} - z_i) \quad (103)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot \partial_{a_i^{(1)}} a_i^{(2)} \quad (104)$$

$$\partial_{a_i^{(1)}} C^{(2)} = \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(2)} - z_i) \cdot a_i^{(1)}(1 - a_i^{(1)}) \cdot w^{(2)} \quad (105)$$

$$(106)$$

### 1.4.2 Cost Of The 1st layer

Here is an example of **back-propagation** in the works:

- We know that the expected value for the 2nd layer is  $z_i$
- Let's use  $e_i$  to denote the expected value for i-th sample of the 1st layer
- $e_i$  is the difference between the activation of the first layer and the derivative of the cost function of the second layer

$$e_i = a_i^{(1)} - \partial_{a_i^{(1)}} C^{(2)}$$

- Now we can define the cost function of the 1st layer as such

$$C^{(1)} = \frac{1}{n} \sum_{i=1}^n (a_i^{(1)} - e_i)^2$$

Now that we can easily access the weights and bias of the first layer let's compute the corresponding partial derivatives

$$a_i^{(1)} = \sigma(x_i \cdot w^{(1)} + b^{(1)}) \quad (107)$$

$$(108)$$

$$\partial_{w^{(1)}} a_i^{(1)} = a_i^{(1)}(1 - a_i^{(1)}) \cdot a_i^{(0)} \quad (109)$$

$$\partial_{b^{(1)}} a_i^{(1)} = a_i^{(1)}(1 - a_i^{(1)}) \quad (110)$$

We know that the activation of 0th layer -  $a_i^{(0)}$  represents the  $i$ -th input value -  $x_i$

$$\boxed{\partial_{w^{(1)}} a_i^{(1)} = a_i^{(1)}(1 - a_i^{(1)}) \cdot x_i} \quad \boxed{\partial_{b^{(1)}} a_i^{(1)} = a_i^{(1)}(1 - a_i^{(1)})} \quad (111)$$

$$\partial_{w^{(1)}} C^{(1)} = \partial_{w^{(1)}} \left( \frac{1}{n} \sum_{i=1}^n (a_i^{(1)} - e_i)^2 \right) = \quad (112)$$

$$= \frac{1}{n} \sum_{i=1}^n \partial_{w^{(1)}} (a_i^{(1)} - e_i)^2 = \quad (113)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(1)} - e_i) \cdot \partial_{w^{(1)}} (a_i^{(1)} - e_i) = \quad (114)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(1)} - e_i) \cdot \partial_{w^{(1)}} a_i^{(1)} = \quad (115)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(1)} - (a_i^{(1)} - \partial_{a_i^{(1)}} C^{(2)})) \cdot \partial_{w^{(1)}} a_i^{(1)} = \quad (116)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot (a_i^{(1)} - a_i^{(1)} + \partial_{a_i^{(1)}} C^{(2)}) \cdot \partial_{w^{(1)}} a_i^{(1)} = \quad (117)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot \partial_{a_i^{(1)}} C^{(2)} \cdot \partial_{w^{(1)}} a_i^{(1)} = \quad (118)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot \partial_{a_i^{(1)}} C^{(2)} \cdot a_i^{(1)}(1 - a_i^{(1)}) \cdot x_i = \quad (119)$$

$$= \frac{1}{n} \sum_{i=1}^n 2 \cdot \partial_{a_i^{(1)}} C^{(2)} \cdot a_i^{(1)}(1 - a_i^{(1)}) \cdot x_i \quad (120)$$

$$(121)$$

$$(122)$$

$$\partial_{b^{(1)}} C^{(1)} = \frac{1}{n} \sum_{i=1}^n 2 \cdot \partial_{a_i^{(1)}} C^{(2)} \cdot a_i^{(1)}(1 - a_i^{(1)}) \quad (123)$$

## 1.5 Arbitrary Neural Model With 1 Input

Let's say that our neural model has  $m$  layers

### 1.5.1 Feed-Forward

TODO: Describe

To calculate the activation of the  $l$ -th layer for the  $i$ -th sample you must multiply the activation of the  $l - 1$  layer with the weight of the  $l$ -th layer then add the bias of the  $l$ -th layer and apply an activation function to the result.

*As an example the sigmoid- $\sigma$  function is used.*

Let's also consider the activation of the 0th layer  $a_i^{(0)}$  to represent the  $i$ -th sample of the input data -  $x_i$

$$a_i^{(l)} = \sigma(a_i^{(l-1)} \cdot w^{(l)} + b^{(l)}) \quad (124)$$

Partial derivatives

$$\partial_{w^{(l)}} a_i^{(l)} = a_i^{(l)} \cdot (1 - a_i^{(l)}) \cdot a_i^{(l-1)} \quad (125)$$

$$\partial_{b^{(l)}} a_i^{(l)} = a_i^{(l)} \cdot (1 - a_i^{(l)}) \quad (126)$$

$$\partial_{a_i^{(l-1)}} a_i^{(l)} = a_i^{(l)} \cdot (1 - a_i^{(l)}) \cdot w^{(l)} \quad (127)$$

### 1.5.2 Back-Propagation

TODO: Describe

Let's denote this difference  $a_i^{(m)} - z_i$  as a partial derivative  $\partial_{a_i^{(m+1)}}$

$$C^{(l)} = \frac{1}{n} \sum_{i=1}^n (\partial_{a_i^{(l)}} C^{(l+1)})^2 \quad (128)$$

$$(129)$$

$$\partial_{w^{(l)}} C^{(l)} = \frac{1}{n} \sum_{i=1}^n 2 \cdot (\partial_{a_i^{(l)}} C^{(l+1)}) \cdot a_i^{(l)} \cdot (1 - a_i^{(l)}) \cdot a_i^{(l-1)} \quad (130)$$

$$\partial_{b^{(l)}} C^{(l)} = \frac{1}{n} \sum_{i=1}^n 2 \cdot (\partial_{a_i^{(l)}} C^{(l+1)}) \cdot a_i^{(l)} \cdot (1 - a_i^{(l)}) \quad (131)$$

$$\partial_{a_i^{(l-1)}} C^{(l)} = \frac{1}{n} \sum_{i=1}^n 2 \cdot (\partial_{a_i^{(l)}} C^{(l+1)}) \cdot a_i^{(l)} \cdot (1 - a_i^{(l)}) \cdot w^{(l)} \quad (132)$$

## 1.6 Combining Feed-Forward and Back-Propagation

Let's describe the way we will use concepts described in the previous two sections together.

Let's say we have  $k$  samples of training data

1. For each sample we will **forward** the sample through the neural network until we reach the output. - **Feed-Forward**

*Note: Neural Network output is the  $a_i^{(m+1)}$  activation*

2. Starting from the  $m + 1$  activation let's move backwards through the neural network propagating the differences of layer activations - **Back-Propagation**