Assignment 2

Question 2

2) Derive the update rule, and show how to train a 2 layer (1 hidden, 1 output) neural network with backward propagation for regression using MSE loss. Assume you are using sigmoid activation for hidden layer. Explain how this is different from update rule for network trained for binary classification using log loss.

→ Answer:

let X is the input features $[n \times f]$
y is the output features (targets)

Since it is a two layer network.

first layer → w_1, bias $= b_1$
second layer → w_2, bias $= b_2$

for hidden layer

$$z_1 = w_1 x + b_1$$
$$a_1 = g(z_1)$$

for output layer, $z_2 = w_2 a_1 + b_2$

$$a_2 = g(z_2)$$

The output of neural network is

$$\bar{y} = a_{\cancel{3}2}$$

The mean square error is given by

$$L = (a_2 - y)^2$$

→ To train the model

i) Provide random values for weights w_1, w_2, w_3 & biases b_1, b_2, b_3

ii) Update weights using gradient descent

$$w_i = w_i - \alpha \frac{dL}{dw_i}$$

$$b_i = b_i - \alpha \frac{dL}{db_i}$$

iii) Repeat until convergence

finding $\frac{dL}{dw_2}$

$\Rightarrow \frac{d(a_2 - y)^2}{dw_2}$

$$= \frac{d(a_2-y)^2}{d(a_2-y)} \times \frac{d(a_2-y)}{dw_2}$$

$$= 2(a_2-y) \times \frac{d(a_2)}{dz_2} \times \frac{dz_2}{dw_2}$$

$$\Rightarrow 2(a_2-y) \cdot g'(z_2) \cdot a_1$$

$$\therefore \frac{dL}{dw_2} = a_1^T \, 2(a_2-y) \, g'(z_2) \quad - \text{①}$$

$$\therefore \frac{dL}{db_2} = 2(a_2-y)$$

finding $\frac{dL}{dw_1}$

Use chain rule

$$\frac{dL}{dw_1} = \frac{dL}{da_2} \cdot \frac{da_2}{dz_2} \cdot \frac{dz_2}{da_1} \cdot \frac{da_1}{dz_1} \cdot \frac{dz_1}{dw_1} \quad - \text{(II)}$$

We know $\frac{dL}{dw_2} = a_1^T 2(a_2-y) \, g'(z_2)$.

Use chain rule on $\frac{dL}{dw_2}$

$$\frac{dL}{dw_2} = \frac{dL}{da_2} \cdot \frac{da_2}{dz_2} \cdot \frac{dz_2}{dw_2}$$

Substituting values from $\textcircled{I}$ on $\textcircled{II}$

$$\frac{\partial L}{\partial w_1} = 2(a_2-y)\, g'(z_2)\, \frac{\partial z_2}{\partial a_1} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial w_1}$$

$$= 2(a_2-y)\, g'(z_2) . \frac{\partial z_2}{\partial a_1} . \frac{\partial a_1}{\partial z_1} \, \frac{\partial z_1}{\partial w_1} \quad - \textcircled{III}$$

We know $z_2 = w_2 a_1 + b_2$

Then $\frac{\partial z_2}{\partial a_1} = w_2 \quad - \textcircled{IV}$

Again

$a_1 = g(z_1)$

Then $\frac{\partial a_1}{\partial z_1} = g'(z_1) \quad - \textcircled{V}$

Also

$z_1 = w_1 x + b_1$

Then

$\frac{\partial z_1}{\partial w_1} = x \quad - \textcircled{VI}$

Substituting $\textcircled{IV}$, $\textcircled{V}$, $\textcircled{VI}$ in eqⁿ $\textcircled{III}$

$$\frac{\partial L}{\partial w_1} = 2(a_2-y)\, g'(z_2) \cdot w_2 \cdot g'(z_1) \cdot x$$

$$\therefore \frac{\partial L}{\partial w_1} = x^T\, 2(a_2-y) \cdot g'(z_2) \,.\, w_2^T g'(z_1)$$

$$\frac{\partial L}{\partial b_1} = 2(a_2-y) \cdot g'(z_2)\, w_2^T g'(z_1)$$

→ How this is different from update rule for binary classification using log loss

the process is very similar. the differences is observed in the gradients for log loss

$$\frac{\partial L}{\partial w_2} = (a_2 - y) \cdot a_1^T$$

$$\frac{\partial L}{\partial b_2} = (a_2 - y)$$

Also,

$$\frac{\partial L}{\partial w_1} = w_2^T (a_2 - y) g'(z_1) \cdot x^T$$

$$\frac{\partial L}{\partial b_1} = (a_2 - y) w_2 \, g'(z_1)$$

- the log loss function is used when performing a binary classification problem

- the main difference between the two operations is the cost function. As cost function changes, so does the gradients.