Assignment 2.1

Tuesday, February 20, 2024 8:49 PM

YiJie (Jackie) Zhu, 20832931

1.

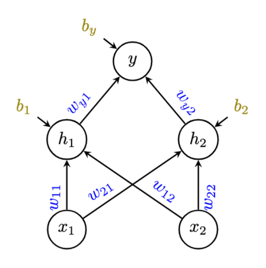


Figure 1: Neural Network

$h_1 = f(w_{11}x_1 + w_{12}x_2 + b_1)$
$= w_{11}x_1 + w_{12}x_2 + b_1$

$h_2 = f(w_{21}x_1 + w_{22}x_2 + b_2)$
$= w_{21}x_1 + w_{22}x_2 + b_2$

$y = f(w_{y1}h_1 + w_{y2}h_2 + b_y)$
$= w_{y1}(w_{11}x_1 + w_{12}x_2 + b_1) + w_{y2}(w_{21}x_1 + w_{22}x_2 + b_2) + b_y$

2.

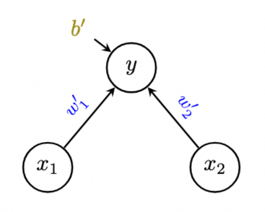


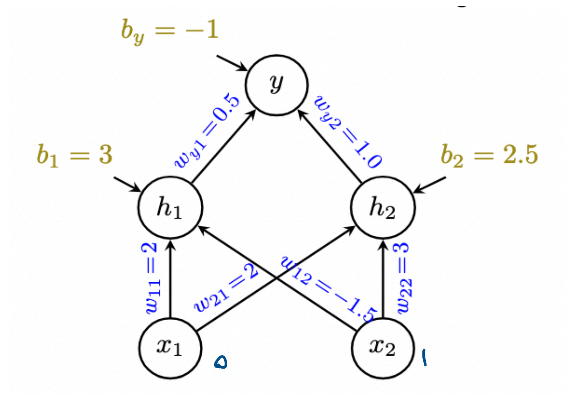
Figure 2: Collapsed Neural Network

Taking equation for y from 1

$y = w_{y1}(w_{11}x_1 + w_{12}x_2 + b_1) + w_{y2}(w_{21}x_1 + w_{22}x_2 + b_2) + b_y$
$y = w_{y1}w_{11}x_1 + w_{y1}w_{12}x_2 + w_{y1}b_1 + w_{y2}w_{21}x_1 + w_{y2}w_{22}x_2 + w_{y2}b_2 + b_y$
$= (w_{y1}w_{11} + w_{y2}w_{21})x_1 + (w_{y1}w_{12} + w_{y2}w_{22})x_2 + w_{y1}b_1 + w_{y2}b_2 + b_y$
$= w'_1 x_1 + w'_2 x_2 + b'$

$\therefore w_1' = w_{y1}w_{11} + w_{y2}w_{21}$
$w_2' = w_{y1}w_{12} + w_{y2}w_{22}$
$b' = w_{y1}b_1 + w_{y2}b_2 + b_y$

3.



let z_a be the output of a before the activation function

$h_1 = \sigma(w_{11}x_1 + w_{12}x_2 + b_1)$
$= \sigma(2(0) - 1.5(1) + 3)$
$= \sigma(1.5)$
$= 0.8176$

$h_2 = \sigma(w_{21}x_1 + w_{22}x_2 + b_2)$
$= \sigma(2(0) + 3(1) + 2.5)$
$= \sigma(5.5)$
$= 0.9959$

$h_y = \sigma(w_{y1}h_1 + w_{y2}h_2 + b_y)$
$= \sigma(0.5(0.8176) + 1(0.9959) - 1)$
$= \sigma(0.4047)$
$= 0.5998$

4. $\mathcal{L} = \frac{1}{2}(t - y)^2 = \frac{1}{2}(1 - 0.5998)^2 = 0.08008$

let z_a be the output of a before the activation function

Forward pass

$\mathcal{L} = \frac{1}{2}(t - y)^2$
$y = \sigma(z_y)$
$z_y = w_{y1}h_1 + w_{y2}h_2 + w_{by}b_y$

$h_1 = \sigma(z_{h_1})$
$h_2 = \sigma(z_{h_2})$
$z_{h_1} = w_{11}x_1 + w_{12}x_2 + w_{b_1}b_1$
$z_{h_2} = w_{21}x_1 + w_{22}x_2 + w_{b_2}b_2$

$w_{by}, w_{b_1}, w_{b_2} = 1$

Backward pass

$\bar{\mathcal{L}} = 1$
$\bar{y} = \frac{\partial \mathcal{L}}{\partial y} = -(t - y)$
$\bar{z}_y = \frac{\partial \mathcal{L}}{\partial z_y} = \bar{y}(\sigma'(z_y))$
$\bar{w}_{y1} = \frac{\partial \mathcal{L}}{\partial w_{y1}} = \bar{z}_y h_1$
$\bar{w}_{y2} = \bar{z}_y h_2$
$\bar{w}_{by} = \bar{z}_y b_y$

$\bar{h}_1 = \bar{z}_y w_{y1}$
$\bar{h}_2 = \bar{z}_y w_{y2}$
$\bar{z}_{h_1} = \bar{h}_1 \sigma'(z_{h_1})$
$\bar{z}_{h_2} = \bar{h}_2 \sigma'(z_{h_2})$
$\bar{w}_{11} = \bar{z}_{h_1}x_1$ $\bar{w}_{21} = \bar{z}_{h_2}x_1$
$\bar{w}_{12} = \bar{z}_{h_1}x_2$ $\bar{w}_{22} = \bar{z}_{h_2}x_2$
$\bar{w}_{b_1} = \bar{z}_{h_1}b_1$ $\bar{w}_{b_2} = \bar{z}_{h_2}b_2$

$\sigma' = \sigma(1 - \sigma)$

$\bar{y} = -(1 - 0.5998)$
$= -0.4002$
$\bar{z}_y = \bar{y}(\sigma(z_y)(1 - \sigma(z_y)))$
$= -0.4002(0.5998(1 - 0.5998))$
$= -0.09606$
$\bar{h}_1 = \bar{z}_y w_{y1}$
$= -0.04803$
$\bar{h}_2 = \bar{z}_y w_{y2}$
$= -0.09606$
$\bar{z}_{h_1} = \bar{h}_1(\sigma(z_{h_1})(1 - \sigma(z_{h_1})))$
$= -0.04803(0.8176(1 - 0.8176))$
$= 0.00716$
$\bar{z}_{h_2} = -0.09606(0.9959(1 - 0.9959))$
$= 0.000392$

$\bar{w}_{y1} = -0.09606(0.8176)$
$= -0.07854$
$\bar{w}_{y2} = -0.09567$
$\bar{w}_{by} = 0.09606$

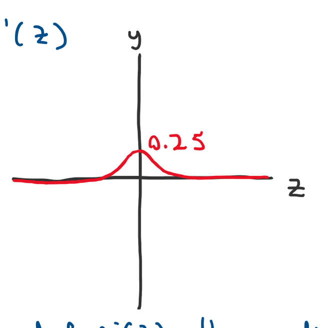
$\bar{w}_{11} = 0$
$\bar{w}_{12} = 0.00716$
$\bar{w}_{b_1} = 0.02148$
$\bar{w}_{21} = 0$
$\bar{w}_{22} = 0.000392$
$\bar{w}_{b_2} = 0.00098$

Gradient Descent

$w_{y1}' = w_{y1} - 0.1(-0.0784)$
$= 2.00784$
$w_{y2}' = 1.009567$
$w_{by}' = 0.9904$
$w_{11}' = 2$
$w_{12}' = -1.500716$
$w_{b_1}' = 2.998$
$w_{21}' = 2$
$w_{22}' = 2.9999$
$w_{b_2}' = 2.4999$

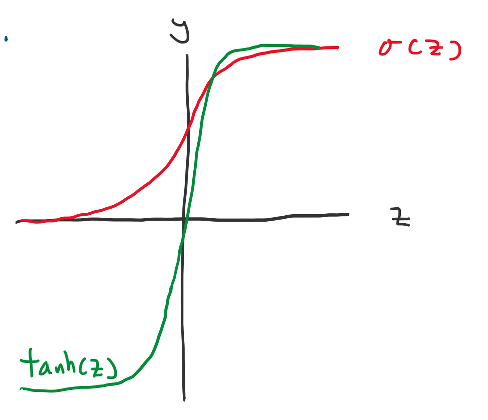
5. $\sigma(z) = \frac{1}{1 + e^{-z}} \Rightarrow \sigma'(z) = \frac{0 \cdot (1 + e^{-z}) - 1(-e^{-z})}{(1 + e^{-z})^2}$
$= \frac{e^{-z}}{(1 + e^{-z})^2}$ or $(\sigma(z)(1 - \sigma(z)))$

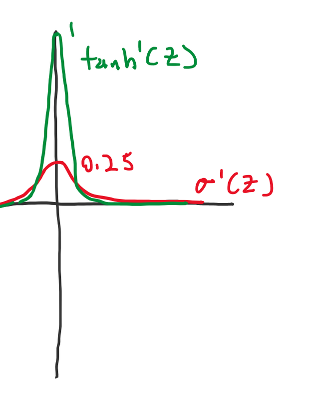
plot of $\sigma'(z)$



At the tail end of $\sigma'(z)$, the gradient is essentially 0. Since we are using this gradient to adjust the weights at each node, if the input to the sigmoid function deviates too much from 0, it's gradient will be extremely small and the adjusted weights would not change much. This is seen in question 4, where the inputs to the sigmoid function is 1.5, 5.5, 0.4047 for hidden and output layers respectively. Using the formula of $\sigma'(z)$, their gradient would be 0.15, 0.0041, 0.24, all very small when combined with the learning rate. Thus, the weights from question 4 did not change much.

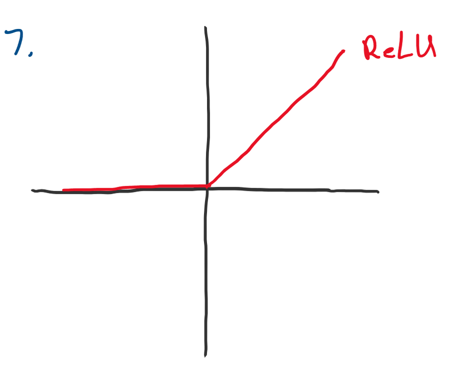
6.





The shape of tanh(z) and it's derivative are very similar to $\sigma(z)$ and $\sigma'(z)$. Both functions are smooth and differentiable. However, tanh starts at -1 as opposed to 0. Furthermore, the derivative of tanh peaks at a value of 1 at $z = 0$ as opposed to 0.25. However, tanh is still susceptible to the vanishing gradient problem as gradients further away from $z = 0$ are still small. Thus, at hidden nodes where $z = 1.5$ and 5.5, tanh would not be a good alternative for sigmoid
You would want to use tanh over sigmoid for output node when the output range is between (-1, 1) instead of (0, 1).

7.



$ReLU' = step(z)$

ReLU does seem to be a more suitable activation function to overcome the vanishing gradient problem since the gradient is a step function now. For all positive inputs, the gradient is 1, thus accelerating the learning of the weights compared to sigmoid and prevents the vanishing gradient case.

Forward pass

$h_1 = ReLU(1.5)$
$= 1.5$

$h_2 = ReLU(5.5)$
$= 5.5$

$y = \sigma(0.5(1.5) + 1(5.5) - 1)$
$= \sigma(5.25)$
$= 0.9948$

Back propagation

$\bar{\mathcal{L}} = 1$
$\bar{y} = \frac{\partial \mathcal{L}}{\partial z_y} = -(t - y)$
$\bar{z}_y = \frac{\partial \mathcal{L}}{\partial y} = \bar{y}(\sigma'(z_y))$
$\bar{w}_{y1} = \frac{\partial \mathcal{L}}{\partial w_{y1}} = \bar{z}_y h_1$
$\bar{w}_{y2} = \bar{z}_y h_2$
$\bar{w}_{by} = \bar{z}_y b_y$

$\bar{h}_1 = \bar{z}_y w_{y1}$
$\bar{h}_2 = \bar{z}_y w_{y2}$
$\bar{z}_{h_1} = \bar{h}_1 ReLU'(z_{h_1})$
$\bar{z}_{h_2} = \bar{h}_2 ReLU'(z_{h_2})$
$\bar{w}_{11} = \bar{z}_{h_1}x_1$ $\bar{w}_{21} = \bar{z}_{h_2}x_1$
$\bar{w}_{12} = \bar{z}_{h_1}x_2$ $\bar{w}_{22} = \bar{z}_{h_2}x_2$
$\bar{w}_{b_1} = \bar{z}_{h_1}b_1$ $\bar{w}_{b_2} = \bar{z}_{h_2}b_2$

$\bar{y} = -0.0052$
$\bar{z}_y = -0.0052(\sigma(y)(1 - \sigma(y)))$ 0.9948
$= -0.0000269$
$\bar{w}_{y1} = -0.00004035$
$\bar{w}_{y2} = -0.000148$
$\bar{w}_{by} = 0.0000269$

$\bar{w}_{11} = (-0.00001345)(0)$
$= 0$
$\bar{w}_{12} = -0.00001345$
$\bar{w}_{b_1} = -0.00004035$
$\bar{w}_{21} = 0$
$\bar{w}_{22} = -0.0000269$
$\bar{w}_{b_2} = -0.00006725$

$\bar{h}_1 = (-0.0000269)(0.5)$
$= -0.00001345$
$\bar{h}_2 = -0.0000269$
$\bar{z}_{h_1} = (-0.00001345)(1)$ $z_{h_1} = 1.5$, $step(1.5) = 1$
$= -0.00001345$
$\bar{z}_{h_2} = -0.0000269$

Gradient descent:

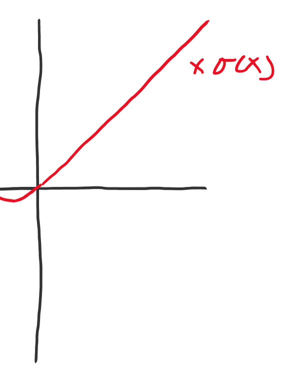
$w_{y1}' = 0.5 - (0.1)(-0.00004035)$
$= 0.500004035$
$w_{y2}' = 1.0000148$
$w_{by}' = 0.99999731$

$w_{21}' = 2$
$w_{22}' = 3.00000269$
$w_{b_2}' = 1.000007$

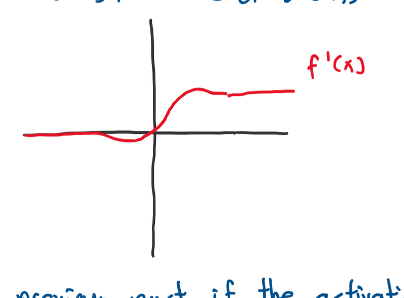
$w_{11}' = 2$
$w_{12}' = -1.4\overline{99}$
$w_{b_1}' = 1.000004$

The updated weights are essentially the same as the original weights. They changed even less than part 4 where sigmoid function was used in the hidden nodes. This is because: 1. Due to the ReLU function used in the hidden nodes, their outputs are large, and 2. The output node is still using the sigmoid activation function. Since h_1 and h_2 are large, z_y is large, which makes $\sigma'(z_y)$ extremely small. This small derivative impacts all the values in backpropagation, making any changes in the weights small, despite the derivative of ReLU in the hidden node being 1.

8.



Derivative:
$f(x) = x\sigma(x)$
$= \sigma(x) + x\sigma(x)(1 - \sigma(x))$



It would combat the problem from the previous part if the activation function of the output node is SILU instead of Sigmoid. It helps since the gradient of SILU for large positive numbers is 1 instead of near 0. This would eliminate the vanishing gradient problem from the output node and prevent it from impacting the adjustment of weights in the hidden nodes.