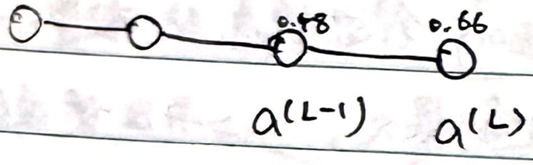
决定激活值的加权和中的权重，像是上一层神经元连接的强弱
偏置表明神经元是否更容易激活。

让代价函数最小. Gradient descent 梯度下降法



$a^{(L)}$ 不是指数，而是用来标记我们在哪一层

① 1.00 desired output

$\downarrow$

y

训练样本的代价

$C_0 = (a^{(L)} - y)^2$

(cost)

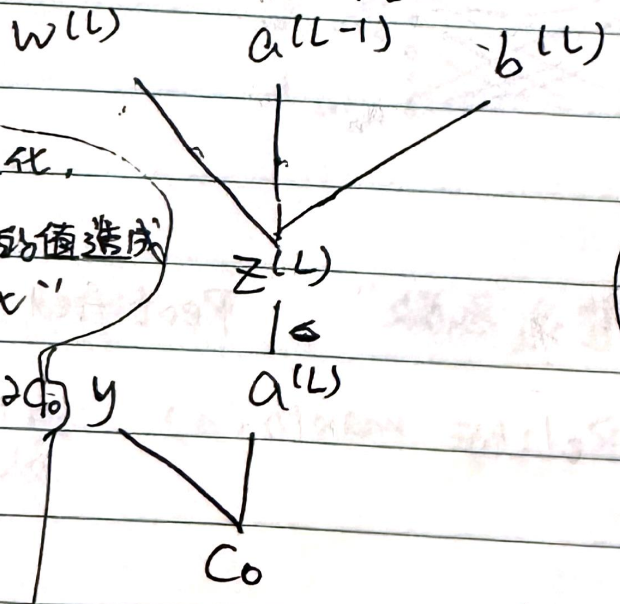
② $z^{(L)} = w^{(L)} \cdot a^{(L-1)} + b^{(L)}$

$a^{(L)} = \sigma(z^{(L)})$

$\partial w^{(L)}$ 即 $w^{(L)}$ 的小变化，

∂C_0 即"$w^{(L)}$ 改变对C的值造成的变化"

$\partial w^{(L)} \to \partial z^{(L)} \to \partial a^{(L)} \to \partial C_0$



weights

$\frac{\partial C_0}{\partial w^{(L)}}$

chain rule

$$\frac{\partial C_0}{\partial w^{(L)}} = \frac{\partial z^{(L)}}{\partial w^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C_0}{\partial a^{(L)}}$$

$$= a^{(L-1)} \sigma'(z^{(L)}) 2(a^{(L)} - y)$$

$$\frac{\partial C_0}{\partial a^{(L)}} = 2(a^{(L)} - y)$$

导数大小与网格最终输出减目标结果的差成正比

(网络输出差别很大，则 w 稍变一点代价改变也很大)

$$\frac{\partial a^{(L)}}{\partial z^{(L)}} = \sigma'(z^{(L)})$$

$$a^{(L)} = \sigma(z^{(L)})$$

$$\boxed{\frac{\partial z^{(L)}}{\partial w^{(L)}} = a^{(L-1)}}$$

权重改变量 ∂w 对最后一层影响取决于前一层神经元

$$\frac{\partial C}{\partial w^{(L)}} = \overbrace{\frac{1}{n}\sum_{k=0}^{n-1}}^{\text{Average of all training examples}} \frac{\partial C_k}{\partial w^{(L)}}$$

Derivative of full cost function

$$\nabla C = \begin{bmatrix} \frac{\partial C}{\partial w^{(1)}} \\ \frac{\partial C}{\partial b^{(1)}} \\ \vdots \\ \frac{\partial C}{\partial w^{(L)}} \\ \frac{\partial C}{\partial b^{(L)}} \end{bmatrix}$$

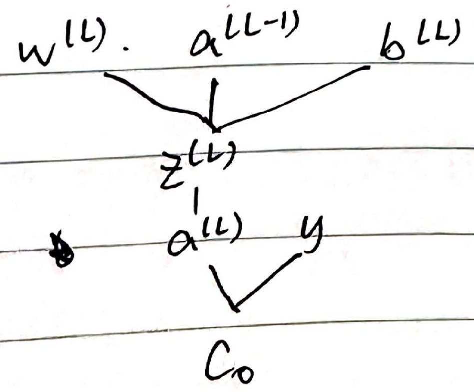
一个分量

bias

对于 $\frac{\partial C_0}{\partial b^{(L)}}$ 仅需将式子中 $\partial w^{(L)}$ 替换即可.

$$\frac{\partial C_0}{\partial b^{(L)}} = \frac{\partial z^{(L)}}{\partial b^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C_0}{\partial a^{(L)}} = \sigma'(z^{(L)})\, 2(a^{(L)} - y)$$

$$\frac{\partial z^{(L)}}{\partial b^{(L)}} = 1$$

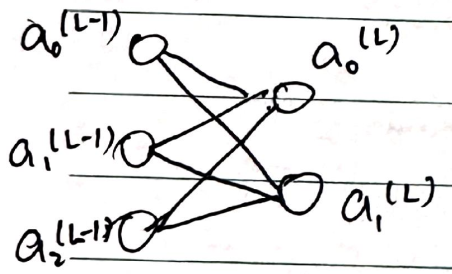


previous layer $\frac{\partial C_0}{\partial a^{(L-1)}}$ 代价函数对上一层激活值的敏感度

$$\frac{\partial C_0}{\partial a^{(L-1)}} = \frac{\partial z^{(L)}}{\partial a^{(L-1)}}\frac{\partial a^{(L)}}{\partial z^{(L)}}\frac{\partial C_0}{\partial a^{(L)}} = w^{(L)}\cdot\sigma'(z^{(L)})\cdot 2(a^{(L)}-y)$$

$$\frac{\partial z^{(L)}}{\partial a^{(L-1)}} = w^{(L)}$$

so 反向应用链式法则，从 $a^{(L-1)}$ 入手 来计算代价函数对之前的权重和偏置敏感度



desired output

$\circ \leftarrow y_0$

$\circ \leftarrow y_1$

$$C_0 = \sum_{j=0}^{n_L-1}(a_j^{(L)}-y_j)^2$$

$a_k^{(L-1)}$ $a_j^{(L)}$

k index the layer (L-1) j index the layer (L)

$w_{jk}^{(L)}$ is connecting this k-th neuron to the j-th neuron

$$z_j^{(L)} = w_{j0}^{(L)}a_0^{(L-1)} + w_{j1}^{(L)}a_1^{(L-1)} + w_{j2}^{(L)}a_2^{(L-1)} + b_j^{(L)}$$

$$\frac{\partial C_0}{\partial w_{jk}^{(L)}} = \frac{\partial z_j^{(L)}}{\partial w_{jk}^{(L)}}\frac{\partial a_j^{(L)}}{\partial z_j^{(L)}}\frac{\partial C_0}{\partial a_j^{(L)}}$$ (no change)

$$\frac{\partial C_0}{\partial a_k^{(L-1)}} = \underbrace{\sum_{j=0}^{n_L-1}\frac{\partial z_j^{(L)}}{\partial a_k^{(L-1)}}\frac{\partial a_j^{(L)}}{\partial z_j^{(L)}}\frac{\partial C_0}{\partial a_j^{(L)}}}_{\text{sum over layer } L}$$

$$
\left.
\begin{array}{l}
\dfrac{\partial C}{\partial w_{jk}^{(L)}} = a_k^{(L-1)} \, \sigma'(z_j^{(L)}) \boxed{\dfrac{\partial C}{\partial a_j^{(L)}}} \\[2ex]
\qquad \displaystyle\sum_{j=0}^{n_{L+1}-1} w_{jk}^{(L+1)} \, \sigma'(z_j^{(L+1)}) \dfrac{\partial C}{\partial a_j^{(L+1)}} \\[2ex]
\qquad\qquad \text{or} \\[1ex]
\qquad\qquad 2(a_j^{(L)} - y_j)
\end{array}
\right\} \nabla C
$$

$$
w_{ij} = w_{ij} - \alpha \frac{\partial C}{\partial w_{jk}^{(L)}}
$$

α 为"学习率"

一反方向增大减小权重