1. Read Deep Learning: An Introduction for Applied Mathematicians. Consider a network as defined in (3.1) and (3.2). Assume that $n_L = 1$, find an algorithm to calculate $\nabla a^{[L]}(x)$.

(3.1) $a^{[1]} = x \in \mathbb{R}^{n_1}$,

(3.2) $a^{[l]} = \sigma(W^{[l]} a^{[l-1]} + b^{[l]}) \in \mathbb{R}^{n_l}$ for $l = 2, 3, \dots, L$.

Define $z^{[l]} = W^{[l]} a^{[l-1]} + b^{[l]}$, $a^{[l]} = \sigma(z^{[l]})$, (Similar to the lecture)

Then, by chain rule, we have

$$\frac{\partial a^{[l]}}{\partial a^{[l-1]}} = \frac{\partial a^{[l]}}{\partial z^{[l]}} \frac{\partial z^{[l]}}{\partial a^{[l-1]}} = \sigma'(z^{[l]}) \cdot W^{[l]}$$

by (3.2), we know that $\sigma'(z^{[l]}) = \begin{pmatrix} \sigma'(z_1^{[l]}) & & 0 \\ & \ddots & \\ 0 & & \sigma'(z_{n_l}^{[l]}) \end{pmatrix}$

$$= \operatorname{diag}(\sigma'(z^{[l]})).$$

Define $D^{[l]} := \operatorname{diag}(\sigma'(z^{[l]}))$.

Then we have

$$\frac{\partial a^{[L]}}{\partial a^{[1]}} = \frac{\partial a^{[L]}}{\partial a^{[L-1]}} \cdot \dots \cdot \frac{\partial a^{[2]}}{\partial a^{[1]}}$$

$$= (\sigma'(z^{[L]}) \cdot W^{[L]}) \cdot \dots \cdot (\sigma'(z^{[2]}) \cdot W^{[2]})$$

Define $g_z^{[\ell]} := \frac{\partial a^{[L]}}{\partial z^{[\ell]}} \in \mathbb{R}^{1 \times n_\ell}$

Since $n_L = 1$ by hypothesis, we can do the algorithm below:

Forward:

① Let $a^{[1]} = x \in \mathbb{R}^{n_x}$.

② Let $z^{[\ell]} = W^{[\ell]} a^{[\ell-1]} + b^{[\ell]}$ for $\ell \in \{2, 3, \cdots, L\}$

$a^{[\ell]} = \sigma(z^{[\ell]})$ where $W^{[\ell]} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$

Backward:

① Let $g^{[L]} = 1$ $(\because n_L = 1)$

② $g^{[\ell-1]} = (g^{[\ell]} \odot \sigma'(z^{[\ell]T})) W^{[\ell]}$ for $\ell \in \{L, \cdots, 2\}$

Then we will get $\nabla_x a^{[L]}(x) = g^{[1]} \in \mathbb{R}^{1 \times n_1}$. $\square$

2. There are unanswered questions during the lecture, and there are likely more questions we haven't covered. Take a moment to think about them and write them down here.

此週課程推導了奇數和偶數 neurons 要怎麼運算，雖然大概理解了原理，但想知道實際在訓練運算時，會訓練多少時間，或是多久會收斂。