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 $\nabla a^{[L]}(x)$.

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

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

Let $\sigma'(z)$ be the derivative of $\sigma(z)$.

By Chain rule, $\frac{\partial a^{[l]}}{\partial a^{[l-1]}} = \mathrm{diag}(\sigma'(W^{[l]} a^{[l-1]} + b^{[l]}))\, W^{[l]}$

Thus $\nabla_x a^{[L]}(x)^T = \frac{\partial a^{[L]}}{\partial a^{[1]}} = \prod_{l=L}^{2} \mathrm{diag}(\sigma'(W^{[l]} a^{[l-1]} + b^{[l]}))\, W^{[l]}$

$\Rightarrow \nabla_x a^{[L]}(x) = W^{[L]T} \mathrm{diag}(\sigma'(W^{[L]} a^{[L-1]} + b^{[L]})) \cdots W^{[2]T} \mathrm{diag}(\sigma'(W^{[2]} a^{[1]} + b^{[2]})) \in \mathbb{R}^{n_1}$.