Yuxuan Wang, December 2020, solution for final project Q2.

2 Neural network

2.1 Derive the gradient

The architecture of the neural network is:

$$\begin{aligned} W^{[1]}x + b^{[1]} &= z^{[1]}, \quad a^{[1]} = g\left(z^{[1]}\right) \\ W^{[2]}a^{[1]} + b^{[2]} &= z^{[2]}, \quad a^{[2]} = g\left(z^{[2]}\right) \\ W^{[3]}a^{[2]} + b^{[3]} &= z^{[3]}, \quad a^{[3]} = g\left(z^{[3]}\right) \end{aligned}$$

where we choose the activation function as:

$$g(z) = \frac{1}{1+e^{-z}}$$

The gradient of the activation function is:

$$\frac{\partial g}{\partial z} = \frac{e^{-z}}{(1+e^{-z})^2}$$

If we only consider only one instance, the loss function is defined by:

$$\mathcal{L} = -\left(y \log a^{[3]} + (1-y)\log(1-a^{[3]})\right)$$

Consider the architecture of the neural network:

$$\begin{aligned} a^{[3]} &= g\left(z^{[3]}\right) \\ z^{[3]} &= W^{[3]}a^{[2]} + b^{[3]} \end{aligned}$$

When $l = 3$, using the chain rule, the gradient $\frac{\partial \mathcal{L}}{\partial W^{[3]}}$ and $\frac{\partial \mathcal{L}}{\partial b^{[3]}}$ are:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial W^{[3]}} &= \frac{\partial \mathcal{L}}{\partial a^{[3]}} \frac{\partial a^{[3]}}{\partial z^{[3]}} \frac{\partial z^{[3]}}{\partial W^{[3]}} \\ &= -\left(\frac{y}{a^{[3]}} + \frac{y-1}{1-a^{[3]}}\right) \frac{e^{-z^{[3]}}}{(1+e^{-z^{[3]}})^2} a^{[2]} \\ \frac{\partial \mathcal{L}}{\partial b^{[3]}} &= \frac{\partial \mathcal{L}}{\partial a^{[3]}} \frac{\partial a^{[3]}}{\partial z^{[3]}} \frac{\partial z^{[3]}}{\partial b^{[3]}} \\ &= -\left(\frac{y}{a^{[3]}} + \frac{y-1}{1-a^{[3]}}\right) \frac{e^{-z^{[3]}}}{(1+e^{-z^{[3]}})^2} \end{aligned} \tag{1}$$

Similarly, when $l = 2$, using the chain rule, the gradient $\frac{\partial \mathcal{L}}{\partial W^{[2]}}$ and $\frac{\partial \mathcal{L}}{\partial b^{[2]}}$ are:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial W^{[2]}} &= \frac{\partial \mathcal{L}}{\partial a^{[3]}} \frac{\partial a^{[3]}}{\partial z^{[3]}} \frac{\partial z^{[3]}}{\partial a^{[2]}} \frac{\partial a^{[2]}}{\partial z^{[2]}} \frac{\partial z^{[2]}}{\partial W^{[2]}} \\ &= -\left(\frac{y}{a^{[3]}} + \frac{y-1}{1-a^{[3]}}\right) \frac{e^{-z^{[3]}}}{(1+e^{-z^{[3]}})^2} W^{[3]} \odot \frac{e^{-z^{[2]}}}{(1+e^{-z^{[2]}})^2} a^{[1]} \\ \frac{\partial \mathcal{L}}{\partial b^{[2]}} &= \frac{\partial \mathcal{L}}{\partial a^{[3]}} \frac{\partial y}{\partial a^{[3]}} \frac{\partial a^{[3]}}{\partial z^{[3]}} \frac{\partial z^{[3]}}{\partial a^{[2]}} \frac{\partial a^{[2]}}{\partial z^{[2]}} \frac{\partial z^{[2]}}{\partial b^{[2]}} \\ &= -\left(\frac{y}{a^{[3]}} + \frac{y-1}{1-a^{[3]}}\right) \frac{e^{-z^{[3]}}}{(1+e^{-z^{[3]}})^2} W^{[3]} \odot \frac{e^{-z^{[2]}}}{(1+e^{-z^{[2]}})^2} \end{aligned} \tag{2}$$

When $l = 1$, also using the chain rule, the gradient $\frac{\partial \mathcal{L}}{\partial W^{[1]}}$ and $\frac{\partial \mathcal{L}}{\partial b^{[1]}}$ are:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial W^{[1]}} &= \frac{\partial \mathcal{L}}{\partial z^{[2]}} \frac{\partial z^{[2]}}{\partial a^{[1]}} \frac{\partial a^{[1]}}{\partial z^{[1]}} \frac{\partial z^{[1]}}{\partial W^{[1]}} \\ &= -\left(\frac{y}{a^{[3]}} + \frac{y-1}{1-a^{[3]}}\right) \frac{e^{-z^{[3]}}}{(1+e^{-z^{[3]}})^2} W^{[3]} \odot \frac{e^{-z^{[2]}}}{(1+e^{-z^{[2]}})^2} \odot W^{[2]} \odot \frac{e^{-z^{[1]}}}{(1+e^{-z^{[1]}})^2} x \\ \frac{\partial \mathcal{L}}{\partial b^{[1]}} &= \frac{\partial \mathcal{L}}{\partial z^{[2]}} \frac{\partial z^{[2]}}{\partial a^{[1]}} \frac{\partial a^{[1]}}{\partial z^{[1]}} \frac{\partial z^{[1]}}{\partial b^{[1]}} \\ &= -\left(\frac{y}{a^{[3]}} + \frac{y-1}{1-a^{[3]}}\right) \frac{e^{-z^{[3]}}}{(1+e^{-z^{[3]}})^2} W^{[3]} \odot \frac{e^{-z^{[2]}}}{(1+e^{-z^{[2]}})^2} \odot W^{[2]} \odot \frac{e^{-z^{[1]}}}{(1+e^{-z^{[1]}})^2} \end{aligned} \tag{3}$$

It is worth noticing that $\odot$ represents the element-wise multiplication rather than matrix multiplication. For simpler notification, the transpose of W has been omitted due to the element-wise multiplication.

2.2 Implement the stochastic gradient descent to train the neural network.

In Figure 1, we plot the normalized loss and accurancy of the training process using stochastic gradient descent algorithm. The neural network converges quite fast due to the easy architecture, only after 5 epoch, it is almost converges.

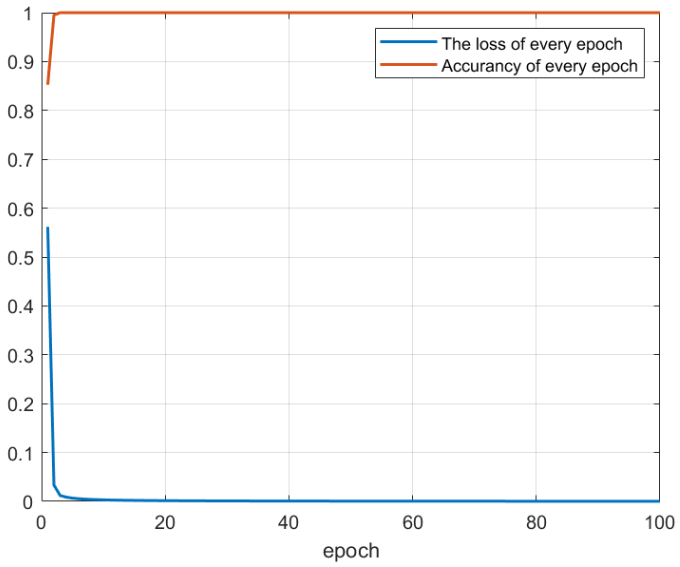

Figure 1: The loss of traing

The train error and test error are calculated by:

$$error = \frac{N_{\text{wrongly classification}}}{N_{\text{total}}}$$

Using the first 400 instances as training data and the remaining data as the test set, the training and training error and accurancy are reported inn Table 1. The test error is 0, showing the good performances of the neural network.

Table 1: Train and test error

training error	test error	training accurancy	test accurancy
0	0	1	1