Theory Questions for Assignment one

Q1.Explain backward and forward propagation :

Forward and Backward prop are easily the most important elements in any neural network

Forward propagation is the process through which inputs pass successively from one layer to another till the out put has been reached.

Backward propagation is a process through which all the wieghts and biases get updated so as to minimize loss.

Q2.Vectorization :

Every input in the initial layer is processed as a matrix of size $(1, 4)$.

$$x = [x_1\ x_2\ x_3\ x_4]$$

The weight matrix for every neuron in the hidden layer would look like this :

$$W_i^1 = [w_1^1\ w_2^1\ w_3^1\ w_4^1]^T$$

where W_i^j denotes weights of i^{th}neuron in j^{th} layer.

the neuron returns

$$a_i^1 = \sigma(x.W_i^1)\ \forall i \in \{0, 1, 2, 3\}$$

these a_i^1 's are sent into the neuron in output layer which works in the same way.

BackProp :

the gradient is calculated of the loss wrt weights (for a single input).

the Loss function is :

$$L(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$$

the gradient so calculated is a column vector of size $(4, 1)$,and each element $(1, i)$ in it is given by :

$$\frac{\partial L}{\partial w_i^1} = \frac{\partial L}{\partial y}.\frac{\partial y}{\partial w_i^2}.\frac{\partial L}{\partial a_i^1}.\frac{\partial a_i^1}{\partial w_i^1}$$

and the weights are then updated as :

$$W^1 := W^1 - \alpha \nabla_W L$$

Q3. List activation functions and their derivatives

- **Sigmoid :**

$$\sigma(x) = \frac{1}{1+e^x}$$
$$\sigma'(x) = \sigma(x)(1-\sigma(x))$$

- **ReLU :**

$$relu(x) = \begin{cases} x & if x = 0 \\ 0 & otherwise \end{cases}$$

$$relu'(x) = \begin{cases} 1 & if x \geq 0 \\ 0 & otherwise \end{cases}$$

- **Leaky ReLU :**

$$lrelu(x) = \begin{cases} x & if x \geq 0 \\ 0.01x & otherwise \end{cases}$$

$$lrelu'(x) = \begin{cases} 1 & if x \geq 0 \\ 0.01 & if otherwise \end{cases}$$

- **tanh :**

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

$$\tanh'(x) = 1 - \tanh^2(x)$$

-

softmax for k classes :

$$p_i = \frac{e^{a_i}}{\sum_{k=1}^{N} e_k^a}$$

with its derivative as :

$$\frac{\partial p_i}{\partial a_j} = \begin{cases} p_i(1 - p_j) & if\ i = j \\ -p_j.p_i & if\ i \neq j \end{cases}$$