Q1: Consider a neural network with three output neurons for a multiclass classification task. The network produces raw scores [2.0, 1.0, 0.5] as its output for a given input. Calculate the probabilities for each class using the softmax activation function. Explain how these probabilities can be interpreted in the context of classification.

Solution:

The softmax function is defined as:

$$P(y_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

where zi are the raw scores (logits) for each class, and e is the base of the natural logarithm.

Given the raw scores for the output neurons:

Z = [2.0,1.0,0.5]

1. Calculate the Exponentials of Each Score

Compute e^zi for each score:

$$e^{2.0} \approx 7.389$$

$$e^{1.0} \approx 2.718$$

$$e^{0.5} \approx 1.649$$

2. Sum of Exponentials

Calculate the sum of these exponentials:

$$\text{Sum} = e^{2.0} + e^{1.0} + e^{0.5} \approx 7.389 + 2.718 + 1.649 \approx 11.756$$

3. Compute Probabilities

Apply the softmax formula for each class:

$$P(y_1) = \frac{e^{2.0}}{\text{Sum}} \approx \frac{7.389}{11.756} \approx 0.629$$

$$P(y_2) = \frac{e^{1.0}}{\text{Sum}} \approx \frac{2.718}{11.756} \approx 0.231$$

$$P(y_3) = \frac{e^{0.5}}{\text{Sum}} \approx \frac{1.649}{11.756} \approx 0.140$$

4. Interpretation of Probabilities

The probabilities obtained from the softmax function can be interpreted as follows:

Class Probabilities: The softmax function converts the raw scores into probabilities that sum to 1. These probabilities represent the network's confidence in each class.

- $P(y1) \approx 0.629$: The network is about 62.9% confident that the input belongs to Class 1.
- $P(y2) \approx 0.231$: The network is about 23.1% confident that the input belongs to Class 2.
- $P(y3) \approx 0.140$: The network is about 14.0% confident that the input belongs to Class 3.

- **Classification Decision**: In a multiclass classification task, the class with the highest probability is usually chosen as the predicted class. Here, Class 1 has the highest probability, so the network's prediction for the input is Class 1.
- **Relative Confidence**: The probabilities also provide a measure of the relative confidence of the network in its predictions. For example, the network is significantly more confident about Class 1 compared to Classes 2 and 3.

2) **Sigmoid Activation Function**

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

Given raw scores:

$X = [2.0, -1.0, 0.5]$

Calculate Sigmoid for Each Score:

Calculate Sigmoid for Each Score:

1. For $x = 2.0$:

$$\sigma(2.0) = \frac{1}{1 + e^{-2.0}} \approx \frac{1}{1 + 0.135} \approx \frac{1}{1.135} \approx 0.881$$

2. For $x = -1.0$:

$$\sigma(-1.0) = \frac{1}{1 + e^{-(-1.0)}} = \frac{1}{1 + e^{1}} \approx \frac{1}{1 + 2.718} \approx \frac{1}{3.718} \approx 0.269$$

3. For $x = 0.5$:

$$\sigma(0.5) = \frac{1}{1 + e^{-0.5}} \approx \frac{1}{1 + 0.607} \approx \frac{1}{1.607} \approx 0.622$$

Interpretation:

Sigmoid Function: The sigmoid function maps any input to a value between 0 and 1. It is often used for binary classification tasks or as the activation function in the output layer of a binary classification model.

3) ReLU (Rectified Linear Unit) Activation Function

The ReLU function is defined as:

ReLU(x) = max(0,x)

Calculate ReLU for Each Score:

1. For $x = 2.0$:

$$\mathrm{ReLU}(2.0) = \max(0, 2.0) = 2.0$$

2. For $x = -1.0$:

$$\mathrm{ReLU}(-1.0) = \max(0, -1.0) = 0$$

3. For $x = 0.5$:

$$\mathrm{ReLU}(0.5) = \max(0, 0.5) = 0.5$$

Interpretation:

ReLU Function: ReLU outputs the input directly if it is positive; otherwise, it outputs zero. It is widely used in hidden layers of neural networks to introduce non-linearity while maintaining computational efficiency.

4) Tanh (Hyperbolic Tangent) Activation Function

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

1. For $x = 2.0$:

$$\tanh(2.0) = \frac{e^2 - e^{-2}}{e^2 + e^{-2}} \approx \frac{7.389 - 0.135}{7.389 + 0.135} \approx \frac{7.254}{7.524} \approx 0.966$$

2. For $x = -1.0$:

$$\tanh(-1.0) = \frac{e^{-1} - e^1}{e^{-1} + e^1} \approx \frac{0.368 - 2.718}{0.368 + 2.718} \approx \frac{-2.350}{3.086} \approx -0.761$$

3. For $x = 0.5$:

$$\tanh(0.5) = \frac{e^{0.5} - e^{-0.5}}{e^{0.5} + e^{-0.5}} \approx \frac{1.649 - 0.607}{1.649 + 0.607} \approx \frac{1.042}{2.256} \approx 0.461$$

Interpretation:

Tanh Function: The Tanh function maps input values to the range [-1, 1]. It is often used in hidden layers of neural networks and is preferred over sigmoid in some cases because it centers the data around zero, which can make learning easier.