Neural Network Forward Propagation - Example 2

In this example, we'll consider another simple neural network with a different combination of activation functions:

- **Input Layer:** 3 neurons
- **Hidden Layer:** 2 neurons with Sigmoid activation
- **Output Layer:** 1 neuron with ReLU activation

Step 1: Initialize Input

Let the input vector be:

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Step 2: Define Weights and Biases

Initialize the weights and biases for each layer.

Weights and Biases for Hidden Layer

$$\mathbf{W}^{(1)} = \begin{bmatrix} w_{11}^{(1)} & w_{12}^{(1)} & w_{13}^{(1)} \\ w_{21}^{(1)} & w_{22}^{(1)} & w_{23}^{(1)} \end{bmatrix}, \mathbf{b}^{(1)} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \end{bmatrix}$$

Weights and Biases for Output Layer

$$\mathbf{W}^{(2)} = \begin{bmatrix} w_{11}^{(2)} & w_{12}^{(2)} \end{bmatrix}, \mathbf{b}^{(2)} = b^{(2)}$$

Step 3: Forward Propagation

3.1 Compute Weighted Sum for Hidden Layer

The weighted input to the hidden layer neurons is calculated as:

$$Z^{(1)} = \mathbf{W}^{(1)}\mathbf{X} + \mathbf{b}^{(1)}$$

3.2 Apply Sigmoid Activation Function

The Sigmoid activation function is defined as:

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

Applying Sigmoid to each element of $Z^{(1)}$:

$$A^{(1)} = \sigma(Z^{(1)})$$

3.3 Compute Weighted Sum for Output Layer

The weighted input to the output neuron is:

$$Z^{(2)} = \mathbf{W}^{(2)} A^{(1)} + b^{(2)}$$

3.4 Apply ReLU Activation Function

The ReLU activation function is defined as:

$$ReLU(z) = \max(0, z)$$

Applying ReLU to $Z^{(2)}$:

$$A^{(2)} = ReLU(Z^{(2)})$$

Step 4: Summary of Forward Propagation

1. Input: $\mathbf{X}$
2. Hidden Layer:
   i. Weighted Sum: $Z^{(1)} = \mathbf{W}^{(1)} \mathbf{X} + \mathbf{b}^{(1)}$
   ii. Activation: $A^{(1)} = \sigma(Z^{(1)})$
3. Output Layer:
   i. Weighted Sum: $Z^{(2)} = \mathbf{W}^{(2)} A^{(1)} + b^{(2)}$
   ii. Activation: $A^{(2)} = ReLU(Z^{(2)})$

Example Calculation

Let's assign specific values to the inputs, weights, and biases for a concrete example.

Given:

- $x_1 = 0.5$
- $x_2 = -1.5$
- $x_3 = 2.0$

- $w_{11}^{(1)} = 0.2$
- $w_{12}^{(1)} = -0.5$
- $w_{13}^{(1)} = 0.3$
- $w_{21}^{(1)} = 1.0$
- $w_{22}^{(1)} = 0.7$
- $w_{23}^{(1)} = -1.2$
- $b_1^{(1)} = 0.1$
- $b_2^{(1)} = -0.2$
- $w_{11}^{(2)} = 1.5$
- $w_{12}^{(2)} = -2.0$
- $b^{(2)} = 0.5$

4.1 Compute $Z^{(1)}$

$Z^{(1)} = \mathbf{W}^{(1)}\mathbf{X} + \mathbf{b}^{(1)}$

Calculating each component:

- $Z_1^{(1)} = 0.2 \times 0.5 + (-0.5) \times (-1.5) + 0.3 \times 2.0 + 0.1$

  Calculation:
  $Z_1^{(1)} = 0.1 + 0.75 + 0.6 + 0.1 = 1.55$

- $Z_2^{(1)} = 1.0 \times 0.5 + 0.7 \times (-1.5) + (-1.2) \times 2.0 + (-0.2)$

  Calculation:
  $Z_2^{(1)} = 0.5 - 1.05 - 2.4 - 0.2 = -3.15$

Thus:

$$Z^{(1)} = \begin{bmatrix} 1.55 \\ -3.15 \end{bmatrix}$$

4.2 Apply Sigmoid to Get $A^{(1)}$

$$A^{(1)} = \sigma(Z^{(1)}) = \begin{bmatrix} \sigma(1.55) \\ \sigma(-3.15) \end{bmatrix}$$

Calculating each component:

- $\sigma(1.55) = \dfrac{1}{1 + e^{-1.55}}$

Calculation:
$e^{-1.55} \approx 0.212$
$\sigma(1.55) = \frac{1}{1+0.212} = \frac{1}{1.212} \approx 0.825$

- $\sigma(-3.15) = \frac{1}{1+e^{3.15}}$

  Calculation:
  $e^{3.15} \approx 23.344$
  $\sigma(-3.15) = \frac{1}{1+23.344} = \frac{1}{24.344} \approx 0.041$

Thus:

$$A^{(1)} = \begin{bmatrix} 0.825 \\ 0.041 \end{bmatrix}$$

4.3 Compute $Z^{(2)}$

$$Z^{(2)} = \mathbf{W}^{(2)} A^{(1)} + b^{(2)}$$

Calculating:
<

- $Z^{(2)} = 1.5 \times 0.825 + (-2.0) \times 0.041 + 0.5$

  Calculation:
  $Z^{(2)} = 1.2375 - 0.082 + 0.5 = 1.6555$

Thus:

$$Z^{(2)} = 1.6555$$

4.4 Apply ReLU to Get $A^{(2)}$

$$A^{(2)} = ReLU(Z^{(2)}) = ReLU(1.6555) = \max(0, 1.6555) = 1.6555$$

Step 5: Final Output

The output of the network is:

$$A^{(2)} = 1.6555$$

Conclusion

Through forward propagation, the neural network processes the input $\mathbf{X} = \begin{bmatrix} 0.5 \\ -1.5 \\ 2.0 \end{bmatrix}$ and produces an output $A^{(2)} = 1.6555$ using a combination of Sigmoid and ReLU activation functions.

General Forward Propagation Equations

For a neural network with multiple layers, the forward propagation can be generalized as:

- **Layer l:**
  - Weighted Sum: $Z^{(l)} = \mathbf{W}^{(l)} A^{(l-1)} + \mathbf{b}^{(l)}$
  - Activation: $A^{(l)} = \text{activation}(Z^{(l)})$

The final output layer applies an appropriate activation function based on the task (e.g., ReLU for regression or other purposes).

Activation Functions Used

Sigmoid

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

Sigmoid squashes the input into a range between 0 and 1, making it suitable for probability estimation in binary classification tasks.

ReLU (Rectified Linear Unit)

$$ReLU(z) = \max(0, z)$$

ReLU introduces non-linearity by outputting zero for negative inputs and the input itself for positive inputs.

Final Notes

This example demonstrates how different activation functions can be combined within a neural network to leverage their unique properties. Sigmoid activation in the hidden layer allows the network to model non-linear relationships, while ReLU in the output layer ensures that the output remains non-negative, which can be beneficial for certain tasks like regression where negative values may not be meaningful.