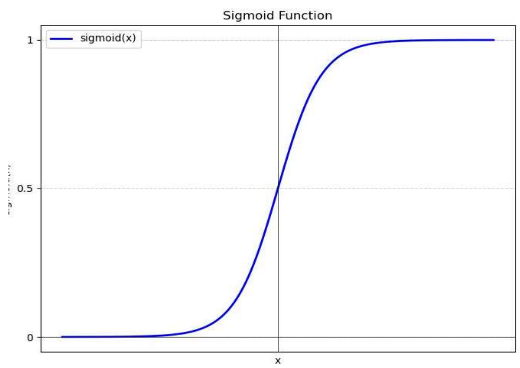
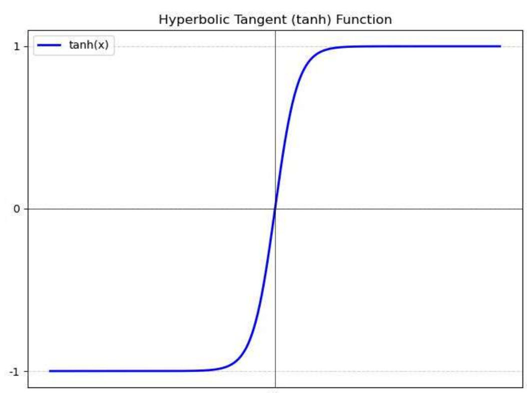
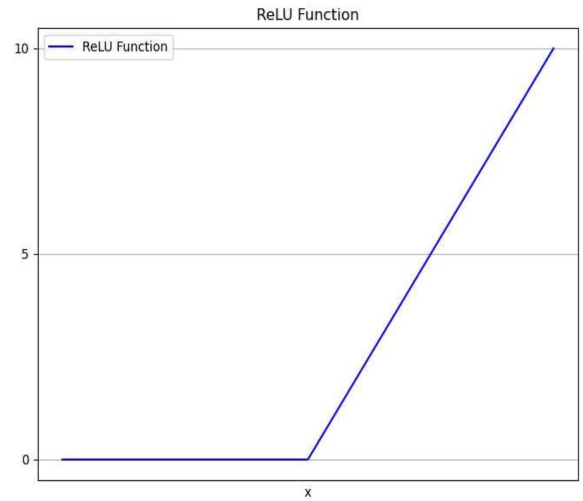
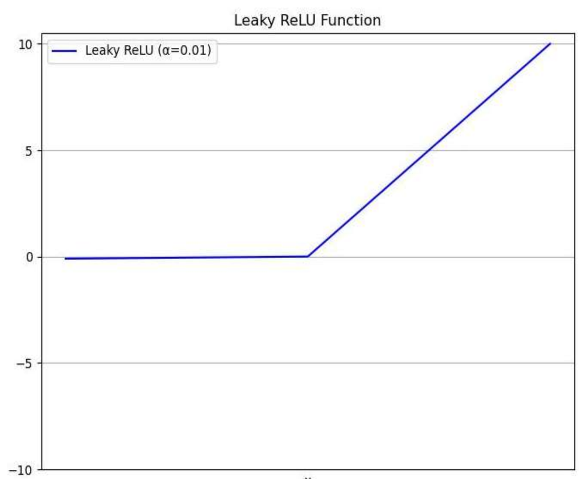
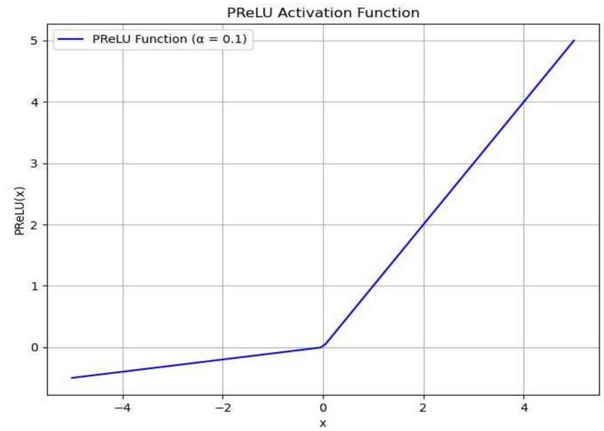
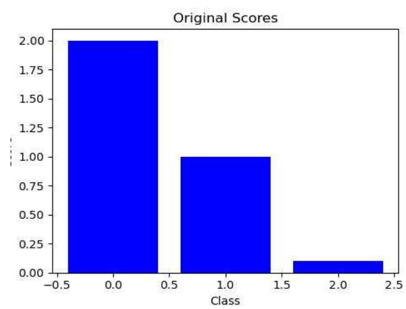
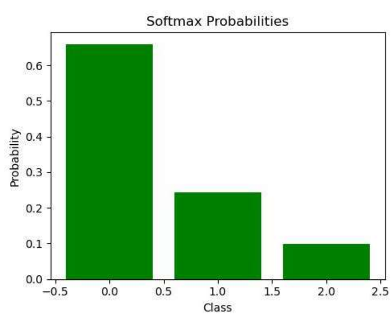
**<u>Fundamentals of Deep Learning</u>**

# Activation Functions

| Function | Graph | Characteristics |
|---|---|---|
| **Sigmoid Function**<br><br>$\sigma(z) = \frac{1}{1+e^{-z}}$ |  | 1. Values between 0 and 1<br>2. Suitable for outputs with binary classification or probabilities<br>3. Suffers from vanishing gradient problem |
| **Hyperbolic Tangent Function**<br><br>$tanh(z) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$ |  | 1. Values between -1 and +1<br>2. Convergence is a bit faster<br>3. Suffers from vanishing gradient problem<br>4. Smooth and differentiable |

| | | |
|---|---|---|
| **Rectified Linear Unit (ReLU)**<br><br>$$\phi(x) = \begin{cases} 0 & x \leq 0 \\ x & x > 0 \end{cases}$$ |  | 1. Values can range from 0 to $\infty$<br>2. Some nodes with little information may be zeroed out (Spare activation)<br>3. Avoids vanishing gradient problem for positive inputs<br>4. Can suffer from dying ReLU problem due to high learning rates, biases etc.<br>5. Useful to capture large effects |
| **Leaky Rectified Linear Unit ( Leaky ReLU)**<br><br>$$\phi(x) = \begin{cases} \alpha x & x \leq 0 \\ x & x > 0 \end{cases}$$ |  | 1. Similar to ReLU but allows negative outcomes<br>2. Values can range from **-$\infty$** to **+$\infty$**<br>3. Outputs are scaled by a factor **$\alpha$** (learning rate)<br>4. Fixed Dying ReLU problem<br>5. No nodes are zeroed out* |

| | | |
|---|---|---|
| **Parametric Rectified Linear Unit (PReLU)**<br>$\mathrm{PReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha \cdot x & \text{if } x \le 0 \end{cases}$ |  | 1. Extension of Leaky ReLU<br>2. Learns optimal slope of negative values in training using parameter **α**<br>3. Used in Convolutional Neural Networks |
| **Softmax Function**<br>$s(x_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}$ |  | 1. Returns a vector of probabilities<br>2. Each element represents the probability of the corresponding class<br>3. Suitable for multi-class classification tasks, often in output layer<br>4. Outputs are scaled down<br>5. Smooth and differentiable |