
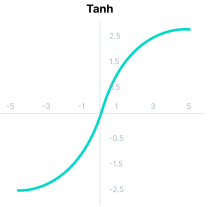
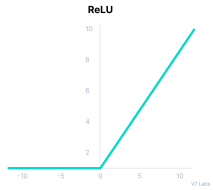
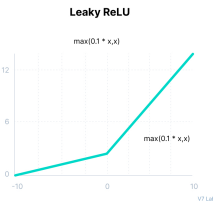
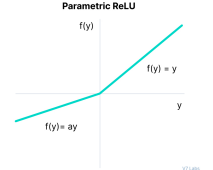
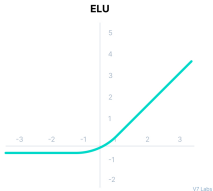
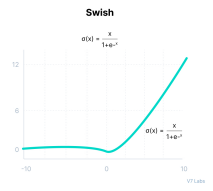
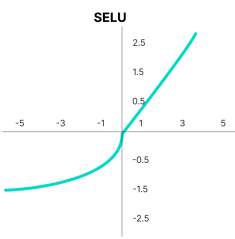
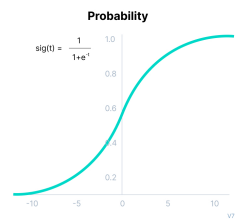


Activation functions:

<p>1. Sigmoid Function: This function maps the input to a range between 0 and 1 and was commonly used in the past as an activation function for neural networks.</p>	<p><i>Sigmoid / Logistic</i></p> $f(x) = \frac{1}{1 + e^{-x}}$	 <p>The graph shows the Sigmoid / Logistic function, which is an S-shaped curve. The x-axis is labeled 'x' and ranges from -6 to 6. The y-axis ranges from 0 to 1. The curve starts near 0 for negative x, passes through 0.5 at x=0, and approaches 1 for positive x.</p>
<p>2. Hyperbolic Tangent (tanh) Function: Similar to the sigmoid function, but it maps the input to a range between -1 and 1, and it is still used in certain situations today.</p>	<p><i>Tanh</i></p> $f(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$	 <p>The graph shows the Hyperbolic Tangent (tanh) function. The x-axis is labeled 'x' and ranges from -5 to 5. The y-axis ranges from -2.5 to 2.5. The curve is an S-shape centered at the origin (0,0), with horizontal asymptotes at y=1 and y=-1.</p>
<p>3. Rectified Linear Unit (ReLU): One of the most popular activation functions, it's simple and effective, replacing negative values with zero to introduce non-linearity.</p>	<p><i>ReLU</i></p> $f(x) = \max(0, x)$	 <p>The graph shows the Rectified Linear Unit (ReLU) function. The x-axis is labeled 'x' and ranges from -10 to 10. The y-axis ranges from 0 to 10. The function is zero for all negative x values and increases linearly with a slope of 1 for all positive x values.</p>
<p>4. Leaky ReLU: A variation of ReLU, it allows a small, non-zero gradient for negative inputs to prevent neurons from getting stuck during training.</p>	<p><i>Leaky ReLU</i></p> $f(x) = \max(0.1x, x)$	 <p>The graph shows the Leaky ReLU function. The x-axis is labeled 'x' and ranges from -10 to 10. The y-axis ranges from 0 to 12. The function is defined as max(0.1 * x, x). For negative x, the function has a small positive slope (0.1). For positive x, the function increases linearly with a slope of 1.</p>
<p>5. Parametric ReLU (PReLU): Similar to Leaky ReLU, but with the slope for negative inputs as a learnable parameter.</p>	<p><i>Parametric ReLU</i></p> $f(x) = \max(ax, x)$	 <p>The graph shows the Parametric ReLU (PReLU) function. The x-axis is labeled 'x' and ranges from -10 to 10. The y-axis is labeled 'f(y)' and ranges from 0 to 10. The function is defined as f(y) = max(ay, y). For negative x, the function has a slope 'a'. For positive x, the function increases linearly with a slope of 1.</p>

<p>6. Exponential Linear Unit (ELU): This function smooths the output for negative values and can speed up learning compared to ReLU.</p>	ELU $\begin{cases} x & \text{for } x \geq 0 \\ \alpha(e^x - 1) & \text{for } x < 0 \end{cases}$	
<p>7. Swish: A newer activation function that combines elements of ReLU and sigmoid to achieve better performance in some cases.</p>	$Swish$ $f(x) = x \cdot \text{sigmoid}(x)$	
<p>8. Scaled Exponential Linear Unit (SELU): A self-normalizing activation function that maintains a constant mean and variance in deep neural networks.</p>	$SELU$ $f(\alpha, x) = \lambda \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	
<p>9. Softmax: Commonly used in the output layer for multi-class classification, it converts the output into a probability distribution.</p>	$Softmax$ $\text{softmax}(z_i) = \frac{\exp(z_i)}{\sum_j \exp(z_j)}$	
<p>10. Maxout: An activation function that generalizes ReLU and Leaky ReLU by taking the maximum value of multiple linear pieces.</p>	$h(x) = \max(Z_1, Z_2, \dots, Z_n)$ $h(x) = \max(W_1 \cdot x + b_1, W_2 \cdot x + b_2, \dots, W_n \cdot x + b_n)$	