Neural Networks – Activation Functions, Loss Functions and Model Behavior

1. Gradients of Activation Functions

ReLU (Rectified Linear Unit)

ReLU is defined as ReLU(x) = max(0, x).

The derivative of ReLU is: - 1 when x is greater than 0 - 0 when x is less than or equal to 0

In practice, the derivative at x = 0 is taken as 0. ReLU helps neural networks train faster because it avoids the vanishing gradient problem. However, neurons can sometimes stop learning completely if they keep receiving negative inputs. This issue is known as the dying ReLU problem.

Sigmoid Function

The sigmoid function is defined as $\sigma(x) = 1 / (1 + e^{-x})$.

The derivative of sigmoid is: $\sigma'(x) = \sigma(x)(1 – \sigma(x))$

When the input value is very large or very small, the gradient becomes close to zero. This causes very slow learning in deep networks and is known as the vanishing gradient problem.

2. Loss Functions and Their Effect on Convergence

Mean Squared Error (MSE)

Used mainly for regression problems. It measures the average squared difference between predicted and actual values. MSE is sensitive to outliers and usually converges slowly for classification tasks.

Binary Cross Entropy

Used for binary classification. It penalizes wrong predictions heavily and provides strong gradients, which helps the model converge faster than MSE.

Categorical Cross Entropy

Used for multi-class classification problems along with the softmax activation function. It works well with probability outputs and provides stable and fast convergence.

The choice of loss function directly affects how fast and how well a neural network learns. A mismatch between loss function and task can lead to poor convergence.

3. Choosing Activation Functions

Hidden Layers

ReLU is the most commonly used activation function for hidden layers because it is computationally efficient and allows gradients to flow easily. Leaky ReLU is preferred when standard ReLU causes neurons to die. Tanh is sometimes used when zero-centered outputs are needed. Sigmoid is generally avoided in hidden layers due to vanishing gradients.

Output Layers

For binary classification, sigmoid activation is used. For multi-class classification, softmax is used. For regression problems, a linear activation function is preferred.

4. Underfitting and Overfitting

Underfitting

Underfitting occurs when the model is too simple to capture patterns in the data. Both training loss and validation loss remain high. This usually happens when the model has insufficient capacity or is not trained long enough.

Overfitting
Overfitting occurs when the model learns the training data too well and fails to generalize. Training loss becomes very low, but validation loss increases. This is caused by overly complex models or insufficient data.

Good Model Fit
A well-fitted model shows low training loss and low validation loss, with both curves remaining close to each other. Such a model generalizes well to unseen data.

Summary
ReLU enables fast learning but may suffer from dead neurons. Sigmoid provides probability outputs but causes vanishing gradients. Cross-entropy loss functions are preferred for classification tasks. Underfitting indicates an overly simple model, while overfitting indicates excessive complexity.