Introduction

Artificial neurons form the basic computational unit of neural networks; they receive weighted inputs, compute a summed signal, and use an **activation function** to produce an output that is forwarded to later units. This discussion explains how an artificial neuron works, defines activation functions, and focuses on the sigmoid activation: why it matters in artificial networks and how its behavior loosely parallels aspects of biological neurons. The analysis draws on university library materials and current online sources.

How an artificial neuron function

An artificial neuron (also called a perception in its simplest form) models three components: (1) inputs (x_i) that represent signals from other neurons or data features, (2) weights (w_i) that scale each input, and (3) a bias (b) that shifts the activation threshold. The neuron computes a weighted sum ($z = \sum_i w_i x_i + b$) and then applies an activation function ($f(z)$) to produce the output ($y = f(z)$). This two-stage process — linear aggregation followed by a non-linear transform — gives neural networks the ability to approximate complex, non-linear functions rather than just linear mappings (Datacamp, 2025).

The activation function therefore determines how the neuron responds to its input: it can squash values into a bounded range (e.g., 0–1), produce signed outputs (e.g., −1 to 1), or preserve sparsity and piecewise linearity (e.g., ReLU). Because training uses gradient-based optimization, the mathematical properties of the activation (continuity and differentiability) directly affect learning dynamics and convergence. Practical choices of activation functions trade off interpretability, gradient behavior, and computational cost (GeeksforGeeks, 2025).

What an activation function is — concise definition

An activation function (f) is a (usually) nonlinear function applied to the neuron's pre-activation (z). Its primary roles are to (a) introduce non-linearity so networks can learn non-linear mappings, (b) bound outputs to ranges useful for interpretation (probabilities, signed signals), and (c) shape gradients used by backpropagation so that weights update in a stable, informative way. Common families include sigmoid-like functions (logistic, tanh), rectified functions (ReLU and variants), and piecewise or normalized functions (softmax for multi-class output layers). (DataCamp, 2025)

Why the sigmoid function is important in artificial neurons

The logistic sigmoid ($\sigma(z) = \frac{1}{1+e^{-z}}$) maps any real input into (0,1), producing an S-shaped curve that approaches 0 for large negative (z) and 1 for large positive (z). Historically, sigmoid served two important roles:

1. **Probability-like output for binary decisions.** Because sigmoid outputs fall in (0,1), they naturally interpret as probabilities in binary classification and in logistic regression models (Coursera, 2025).

2. **Smooth, differentiable nonlinearity.** The logistic function is differentiable everywhere, enabling gradient-based learning (backpropagation) without the discontinuities that block gradient flow. Early neural networks relied heavily on sigmoids for this reason (MachineLearningMastery.com, 2021; Built In, 2023)

However, practitioners now often prefer ReLU and its variants for hidden layers because sigmoids can suffer from **vanishing gradients**: when inputs push the sigmoid toward its flat tails (near 0 or 1), the derivative becomes very small, and weight updates slow dramatically in deep networks. Despite that limitation, sigmoid remains useful for output neurons in binary prediction tasks and in shallow architectures where vanishing gradients are less severe (V7 Labs, 2021).

Relation between sigmoid and biological neurons

Biological neurons integrate synaptic inputs to change membrane potential and produce action potential (spikes) when the membrane potential crosses a threshold. This thresholded, saturating behavior inspires the sigmoid's S-shape: small, summed inputs yield near-zero firing probability, large inputs saturate the response, and intermediate inputs produce a graded increase in firing likelihood. In modeling terms, the sigmoid approximates the relationship between membrane potential (input) and firing rate (output) when the system is considered in rate-coded rather than spike-timed terms (Wikipedia; neurocomputing lecture notes). While biological neurons ultimately transmit discrete spikes, many high-level neural models treat firing rate as a continuous variable that follows a non-linear (often sigmoid-like) input–output curve, which explains the conceptual link between the logistic activation and neural physiology (Wikipedia, 2025).

Conclusion

An artificial neuron computes a weighted sum and applies an activation function; that function determines non-linearity, output range, and gradient properties that shape learning. The sigmoid function historically played a central role because it is smooth, bounded, and interpretable as a probability, and it also parallels the saturating, threshold-like behavior of biological neurons when those neurons are viewed through a firing-rate lens. Modern practice balances these benefits against training issues such as vanishing gradients, which is why alternative activations (e.g., ReLU) often replace sigmoid in deep hidden layers while sigmoid remains appropriate for binary outputs.

References

Datacamp. (2025). *Sigmoid activation function: Deep learning basics*. https://www.datacamp.com/articles/sigmoid-activation-function.

GeeksforGeeks. (2025). *Activation functions in neural networks*. https://www.geeksforgeeks.org/machine-learning/activation-functions-neural-networks/.

Coursera. (2025). *Sigmoid activation function: An introduction*. https://www.coursera.org/articles/sigmoid-activation-function.

Wikipedia. (2025). *Biological neuron model*. https://en.wikipedia.org/wiki/Biological_neuron_model.

Built In. (2023). *Sigmoid activation function: An introduction*. https://builtin.com/machine-learning/sigmoid-activation-function.

MachineLearningMastery. (2021). *A gentle introduction to sigmoid function*. https://machinelearningmastery.com/a-gentle-introduction-to-sigmoid-function/.

V7 Labs. (2021). *Activation functions in neural networks [12 types & use]*. https://www.v7labs.com/blog/neural-networks-activation-functions.

Word count: 745