

Perceptron Activity

A perceptron is a simple type of neural network used for binary classification. It takes input values, multiplies them by weights, adds a bias, and passes the result through an activation function to produce an output. The perceptron learns by adjusting its weights based on errors, improving its accuracy over time. It works best for problems where the data can be separated with a straight line (Singh and Banerjee, 2019).

Exercise 1

In this exercise, I was supposed to change input values and weights to observe their effect on the sum function. Increasing inputs raised the sum, while decreasing them lowered it. Negative inputs reversed the result's sign. Changing the weights showed how they control how much each input contributes to the sum—bigger weights made the inputs matter more, and smaller weights made them matter less. I learned that both inputs and weights work together to decide the final result, and even small changes can have a big impact.

Exercise 2

In this exercise, I was expected to run the code and understand how the AND operator works. The AND operator gives an output of 1 only when both inputs are 1, and 0 for all other combinations. The program trained the model to follow this rule by adjusting values step by step until it got the outputs right. After training, I tested the model and saw that it correctly identified all the input combinations of the AND operator. This exercise helped me see how a simple system can be trained to mimic logical rules like the AND operator.

Exercise 3

In this exercise, I was expected to run the code to understand the sigmoid function. The sigmoid function turns any number into a value between 0 and 1. When the input is large, the output is close to 1, and when it is small, the output is close to 0. I learned that the sigmoid function is useful for normalising values and making them easier to work with in neural networks. It helps in calculating errors, updating weights during training, and ensures the outputs stay within a predictable range, which is important for improving accuracy over time. Additionally, the sigmoid function plays a key role in handling the complexity of the XOR operator dataset by enabling multi-layer neural networks to model nonlinear relationships, which a single-layer perceptron cannot achieve (Glasgow, 2023).

References:

- Singh, J. and Banerjee, R., (2019). *A study on single and multi-layer perceptron neural network*. 2019 3rd International Conference on Computing Methodologies and Communication, pp.35–40. Available at: <https://doi.org/10.1109/ICCMC.2019.8819775> (Accessed 23 Jan. 2025)
- Glasgow, M., (2023). *SGD Finds then Tunes Features in Two-Layer Neural Networks with near-Optimal Sample Complexity: A Case Study in the XOR problem*. Available at: <https://arxiv.org/abs/2309.15111> (Accessed 25 January 2025)