Perceptron and Single-Layer Neural Network

Perceptron

A Perceptron is the most basic unit of a neural network. It performs a linear transformation on the input and applies an activation function to produce a binary output.

Mathematical Expression:

*Output= X*W + B*

Where:

- **X** = Input vector (e.g., features from an image)
- **W** = Weight vector
- **B** = Bias (a constant value)
- **Output =** Result after linear combination (often passed to activation function step())

Logic in Verilog:

A Perceptron in hardware multiplies each input with a weight, adds them up with a bias, and then applies a threshold (e.g., output = 1 if sum > 0, else 0).

Single-Layer Neural Network (Dense Layer)

A Single-Layer Neural Network, also called a Dense Layer or Fully Connected Layer, consists of multiple Perceptrons working in parallel.

Each perceptron:

- **Takes the same input vector**
- **Has its own set of weights and bias**
- **Produces one output**

Mathematical Form:

*Y=X*W+B*

Where:

- **X =** Input vector of size NN
- **W =** Weight matrix of size N×MN \times M (for M perceptrons)

- **B =** Bias vector of size MM
- **Y =** Output vector of size MM (one output per perceptron)

Output:

The outputs are typically passed through an activation function to introduce non-linearity.

Example:

If:

- Input vector X = [x1, x2]
- Layer has 6 perceptrons

Then:

- **W =** 2x6 matrix (each perceptron has 2 weights)
- **B =** [b1, b2, b3, b4, b5, b6]
- **Outputs:**

$$y1=x1\cdot w11 + x2\cdot w21 + b1$$

$$y2=x1\cdot w12 + x2\cdot w22 + b2$$

$$y3=x1\cdot w13 + x2\cdot w23 + b3$$

$$y4=x1\cdot w14 + x2\cdot w24 + b4$$

$$y5=x1\cdot w15 + x2\cdot w25 + b5$$

$$y6=x1\cdot w16 + x2\cdot w26 + b6$$

In this project, I implemented a basic perceptron and a single dense layer using Verilog. It has 2 inputs and 6 perceptron's, each with 2 weights and a bias.

To run the project, create a Verilog project, add the source files to **Design Sources** and the testbench files to **Simulation Sources**, then click **Run Simulation**.