**HW1 – Question 1, 2 & 3**

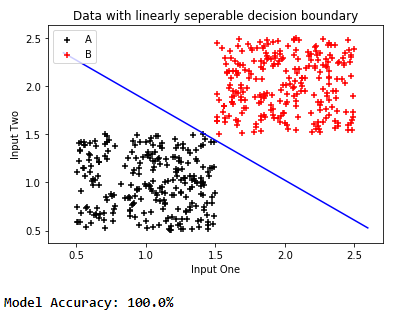
ID: eo9232

Name: Md Reza

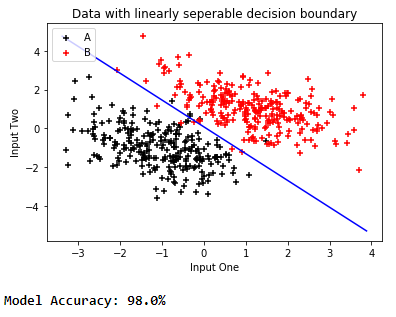
IE7860 – Winter 2022 2021

**Question 1:** Choice #1: Implementation of a Perceptron Network

Two-dimension decision boundary for **two-class uniform** data: It turned out that the perceptron does converge on data and is linearly separable.



Two-dimension decision boundary for **two-class gaussian** data: It turned out that the perceptron does not converge on data and is linearly not separable.



The programming codes are submitted through the canvas.

**Question 2:**

In theory, if a hyper-planar decision boundary exists that correctly classifies all training data into two classes, the training data are linearly separable.

The Single-Layer Perceptron is the most straightforward neural network used for pattern classification. It comprises one layer of input neurons and one layer of output neurons. It expresses only linear dependences between its input neurons and each output neuron, where a set of input patterns with their expected outputs. Learning rules determine the connection weights between neurons to perform an error-free classification. The best weight is leveraged to minimize the generalization error. Many error-free solutions are generally possible if the learning set is linearly separable with a non-linear activation function.

Multi-layer feed-forward networks with hidden layers were introduced & developed to overcome the shortcomings of a single-layer perceptron network. Hence, this justifies the validity of the thesis mentioned in question-2 about a single-layer perceptron.

**Question 3:**

**AND operation:**

|  |  |  |  |
| --- | --- | --- | --- |
| Input | |  | Output |
| *x*1 | *x*2 | Bias *(b)* | *y* |
| 0 | 0 | -1.5 | 0 |
| 1 | 1 | -1.5 | 1 |
| 1 | 0 | -1.5 | 0 |
| 0 | 1 | -1.5 | 0 |

*Where v* = *w*1*x*1 + *w*2*x*2 + *b*

= *x*1 + *x*2 – 1.5

Therefore,

If *x*1 = *x*2 = 0, then *v* = -1.5, and *y* = 0

If *x*1 = *x*2 = 1, then *v* = 0.5, and *y* = 1

If *x*1 = 1, and *x*2 = 0, then *v* = -0.5, and *y* = 0

If *x*1 = 0, and *x*2 = 1, then *v* = -0.5, and *y* = 0

**OR operation:**

|  |  |  |  |
| --- | --- | --- | --- |
| Input | |  | Output |
| *x*1 | *x*2 | Bias *(b)* | *y* |
| 0 | 0 | -0.5 | 0 |
| 1 | 1 | -0.5 | 1 |
| 1 | 0 | -0.5 | 1 |
| 0 | 1 | -0.5 | 1 |

*Where v* = *x*1 + *x*2 + *b*

= *x*1 + *x*2 – 0.5

Therefore,

If *x*1 = *x*2 = 0, then *v* = -0.5, and *y* = 0

If *x*1 = *x*2 = 1, then *v* = 1.5, and *y* = 1

If *x*1 = 1, and *x*2 = 0, then *v* = 0.5, and *y* = 1

If *x*1 = 0, and *x*2 = 1, then *v* = 0.5, and *y* = 1

**COMPLEMENT operation:**

|  |  |  |
| --- | --- | --- |
| Input *x* | Bias *(b)* | *v & y* |
| 0 | 0.5 | 0.5, 1 |
| 1 | 0.5 | -0.5, 0 |

*Where v* = w*x* + *b = -x + 0.5*

Therefore,

If x = 0, then v = 0.5, and y = 1

If x = 1, then v = -0.5, and y = 0

**EXCLUSIVE OR operation:**

|  |  |  |
| --- | --- | --- |
| Input | | Output |
| *x*1 | *x*2 | *y* |
| 0 | 0 | 0 |
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |

The table above indicates that this operation is nonlinearly separable, which justifies the perceptron's limitations because it cannot implement the EXCLUSIVE OR function.