Neural Networks

Exercise 1 : Perceptron Learning

In this exercise, you design a single perceptron with two inputs x_1 and x_2. This perceptron shall implement the boolean formula $A \wedge \neg B$ with a suitable function $y(x_1, x_2)$. Use the values 0 for *false* and 1 for *true*.

(a) Draw all possible examples and a suitable decision boundary in a coordinate system.

(b) Draw the graph of the perceptron. The schematic must include x_1, x_2, and all model weights.

(c) Manually determine a set of suitable weights $\mathbf{w} = (w_0, w_1, w_2)$ from your drawings.

Exercise 2 : Perceptron Learning

Why can the boolean formula A *XOR* B not be learned by a single perceptron? Justify your answer with a drawing.

Exercise 3 : Parameters of the Multilayer Perceptrons

In this exercise, you analyze the number of weights (parameters) of multilayer perceptrons. We use the notation from the lecture (e.g., slide ML:IV-104), where multilayer perceptrons have d layers, p attributes, hidden layer i with l_i units, and an output layer with k units.

(a) Let $d = 4$, $p = 7$, $l_1 = 5$, $l_2 = 3$, $l_3 = 3$, and $k = 4$. Draw the graph of the multilayer perceptron.

(b) Calculate the number of weights in the multilayer perceptron of (a).

(c) Calculate the number of weights in the multilayer perceptron of (a) but with each l_i doubled, i.e., $l_1 = 10$, $l_2 = 6$, $l_3 = 6$. Has the number of weights doubled as well?

(d) Let $f(p, l_1, \ldots, l_{d-1}, k)$ be a function that computes the number of weights in the general case. Write down an expression for f.

