UNIVERSITY OF LEEDS

Class: Machine Learning

Multi-Layer Neural Networks

Instructor: Matteo Leonetti

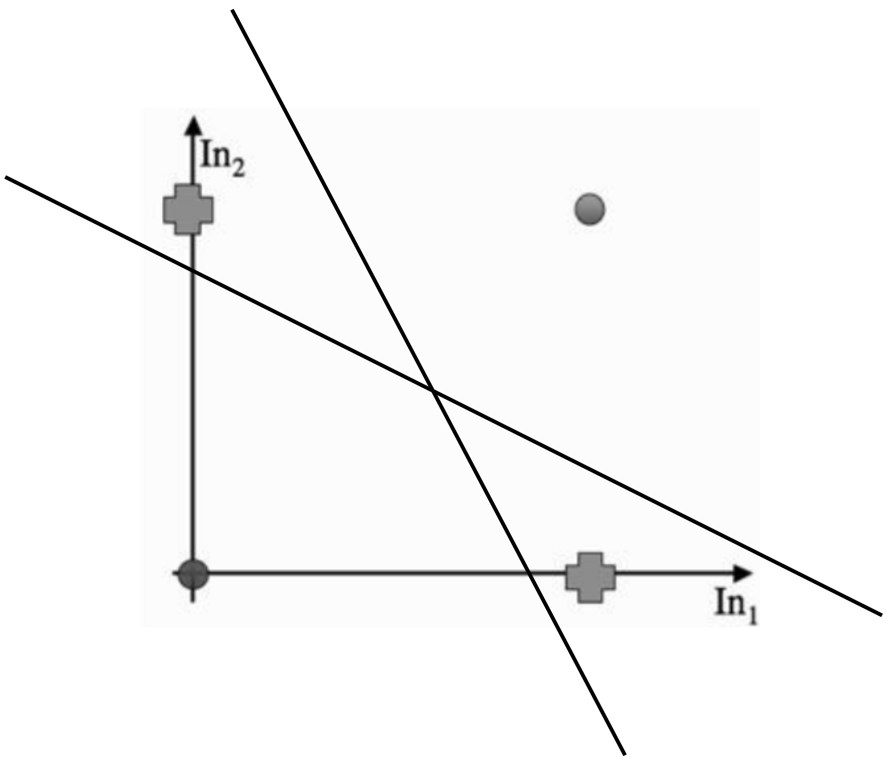
Learning outcomes

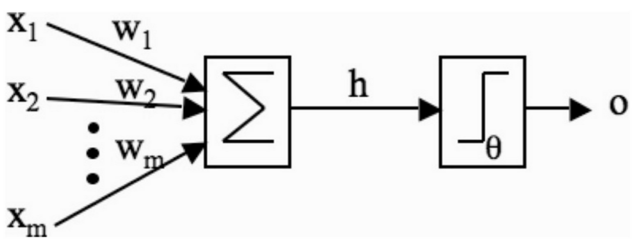
UNIVERSITY OF LEEDS

- Construct a multi-layer neural network that classifies a given dataset in 2D, overcoming the limitation on learning separability.
- Substitute the activation function of the perceptron, with a function amenable to gradient descent.

Perceptron limitations

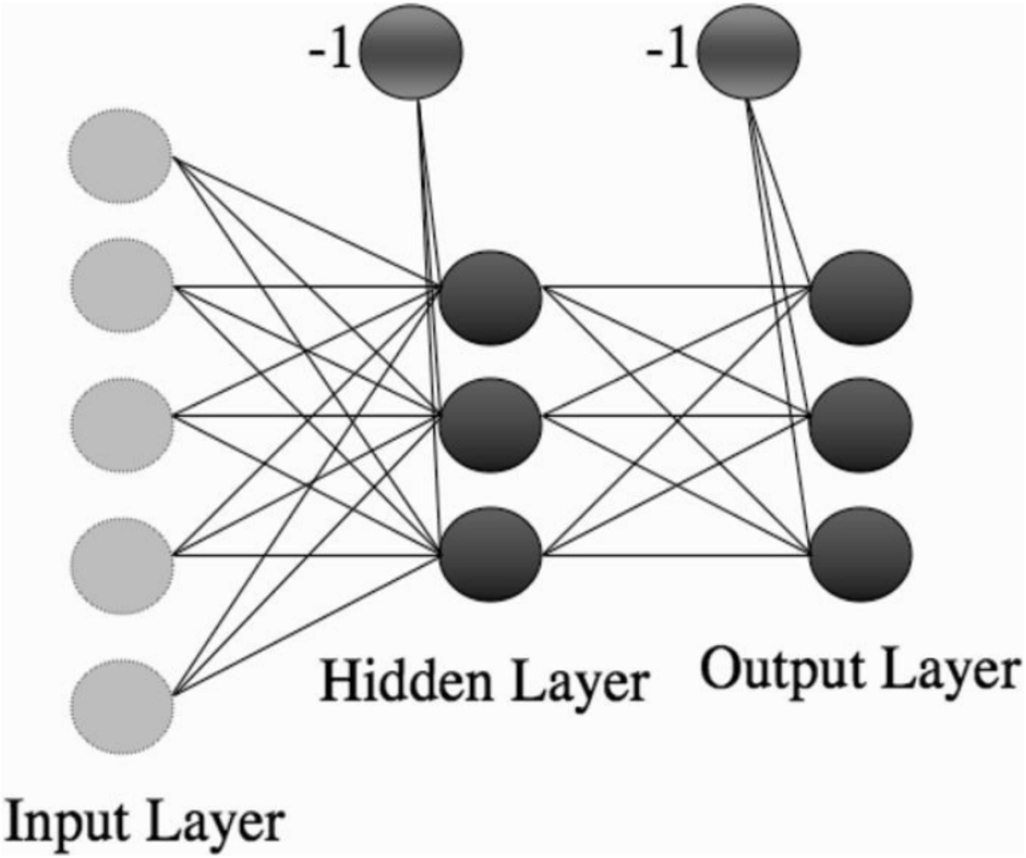
XOR



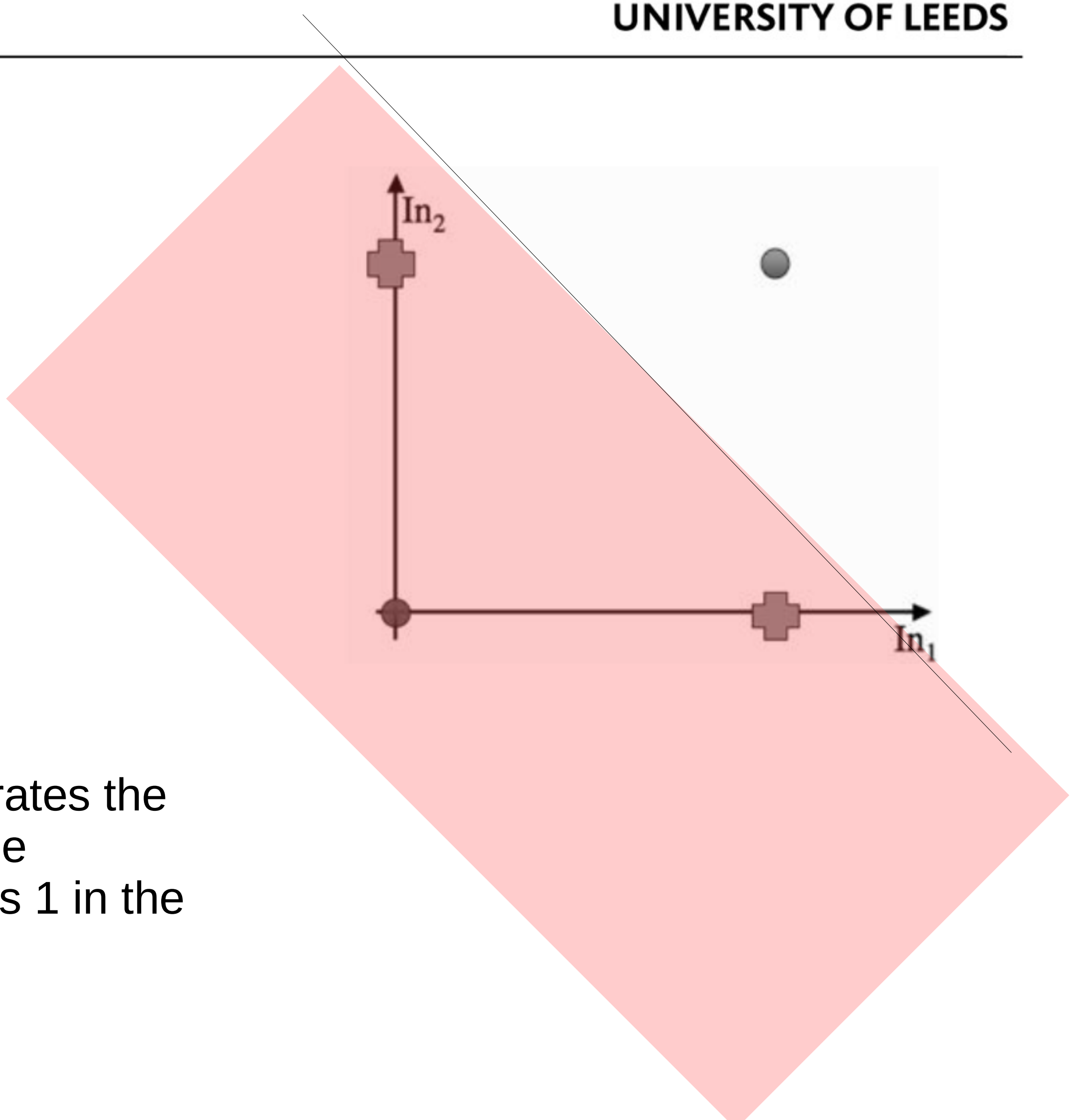


Multi-layer Perceptron

UNIVERSITY OF LEEDS

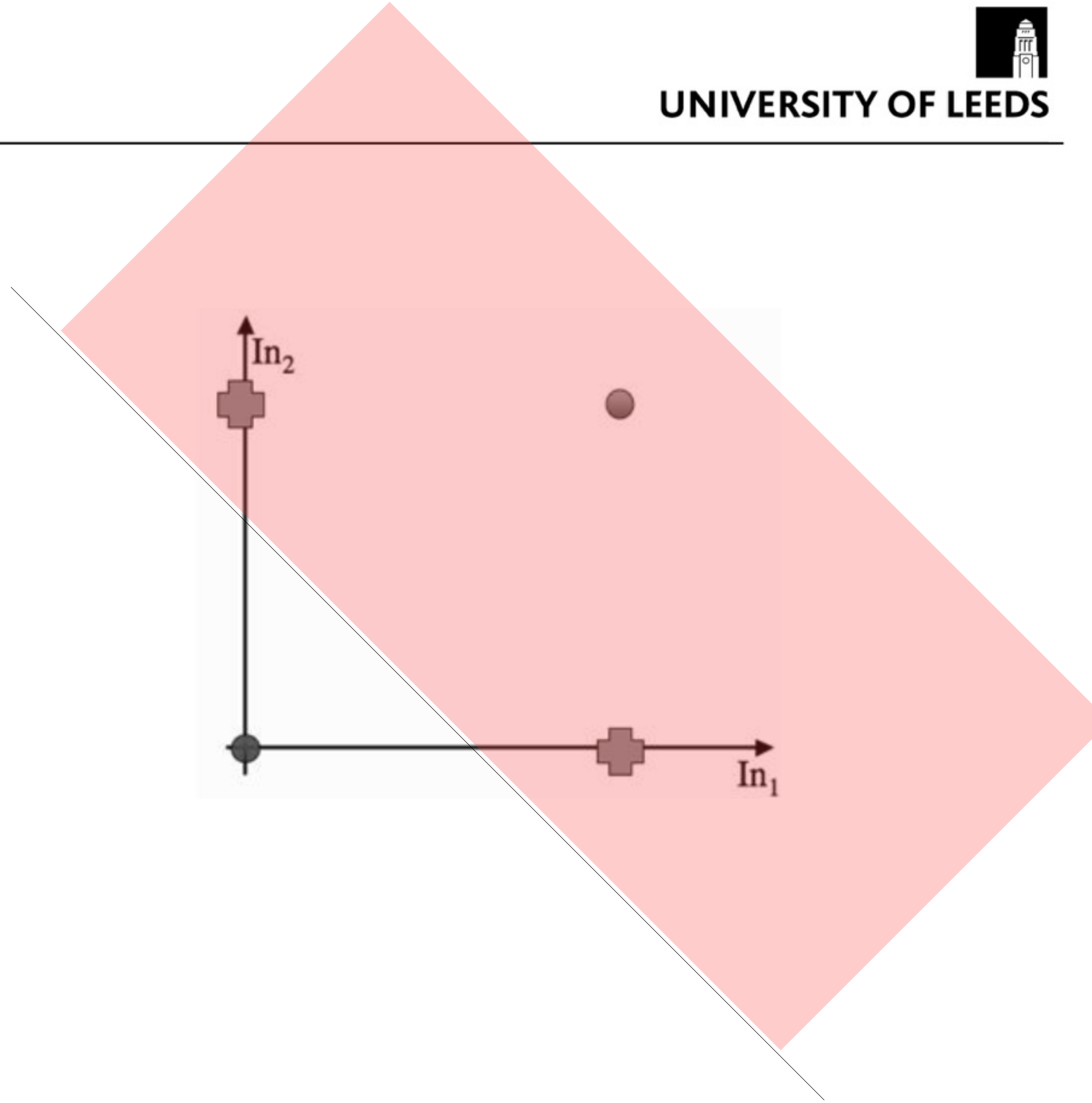


MLP and XOR



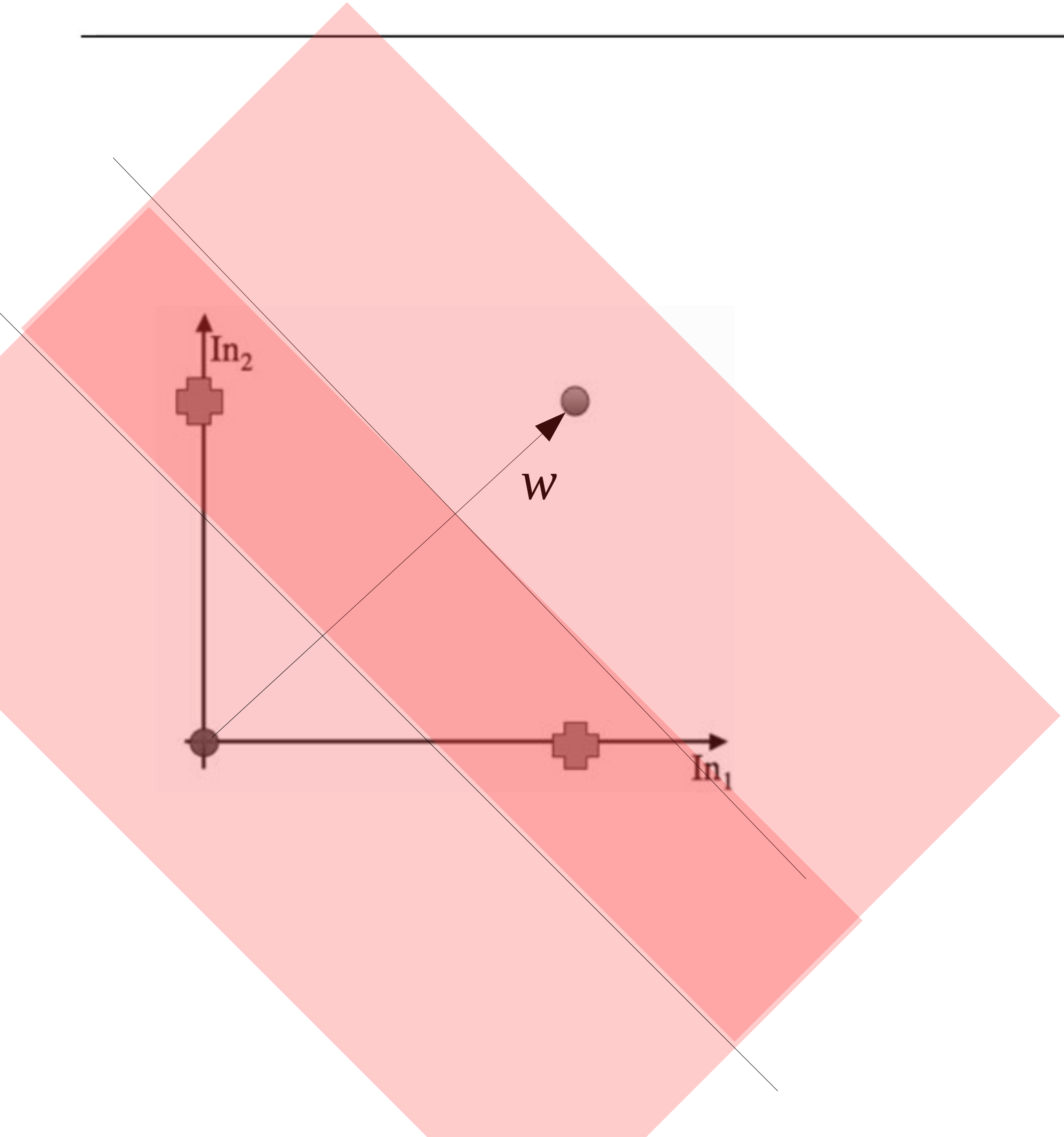
Choose a straight line that separates the points as in the figure, and whose corresponding perceptron returns 1 in the highlighted area

MLP and XOR



Now for the other points

MLP and XOR



Possible solution:

$$-x_1 - x_2 + 2.5 = 0$$

$$w = \langle 2.5, -1, -1 \rangle$$

$$x_1 + x_2 - 0.5 = 0$$

$$w = \langle -0.5, 1, 1 \rangle$$

MLP and XOR

UNIVERSITY OF LEEDS

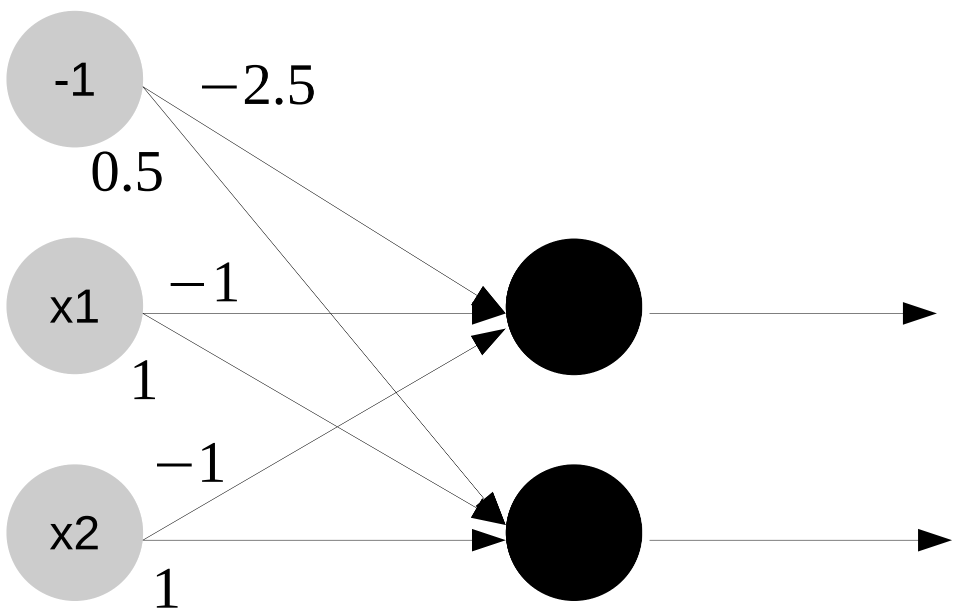
$$-x_1 - x_2 + 2.5 \geq 0$$

$$x_1 + x_2 - 0.5 \geq 0$$

These are 2 perceptrons with weights:

$$\langle 2.5, -1, -1 \rangle$$

$$\langle -0.5, 1, 1 \rangle$$



MLP and XOR

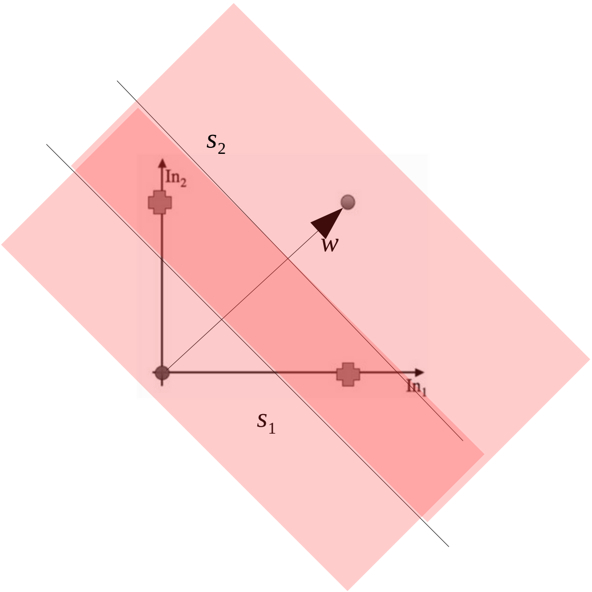
Their outputs are:

What we want

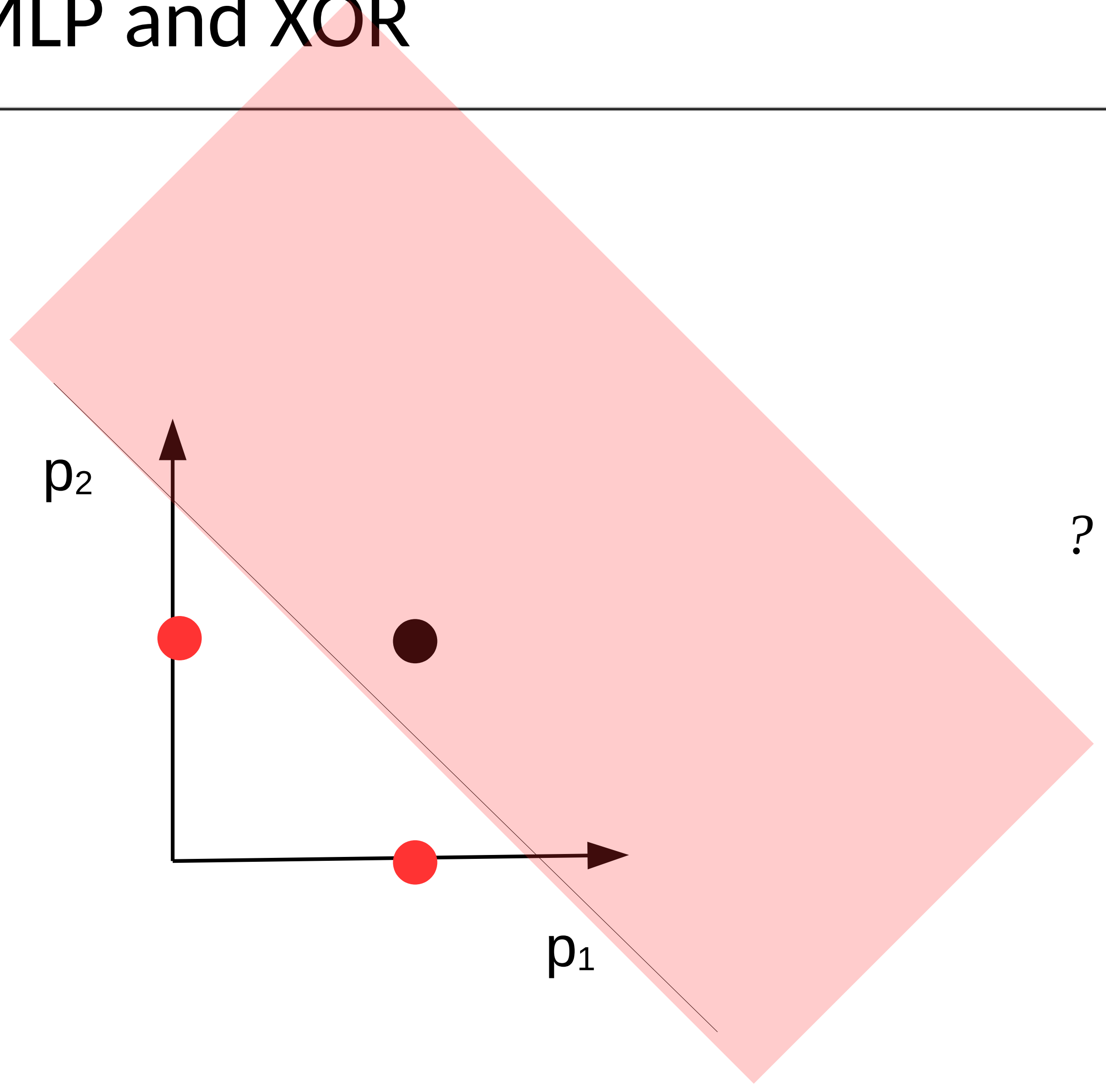
x1	x2	p1	p2	o
0	0	1	0	0
0	1	1	1	1
1	0	1	1	1
1	1	0	1	0

$$p_1 = -x_1 - x_2 + 2.5 \geq 0$$

$$p_2 = x_1 + x_2 - 0.5 \geq 0$$



MLP and XOR



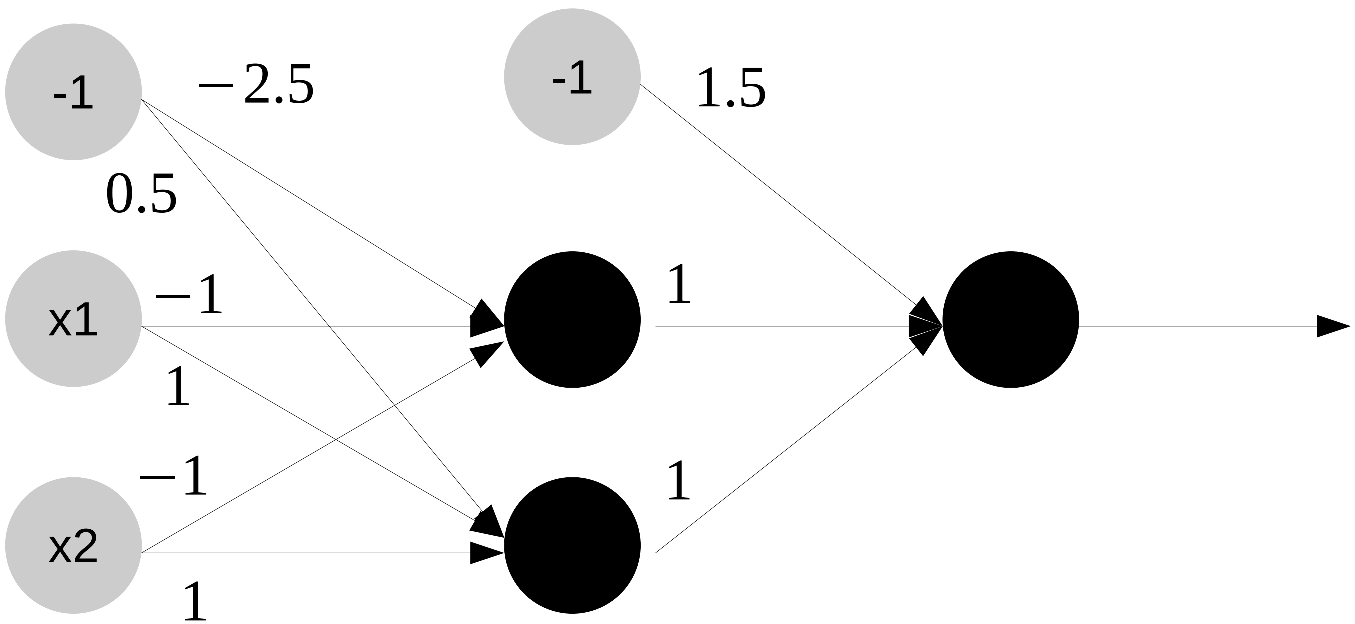
MLP and XOR

UNIVERSITY OF LEEDS

$$p_1 \equiv -x_1 - x_2 + 2.5 \geq 0$$

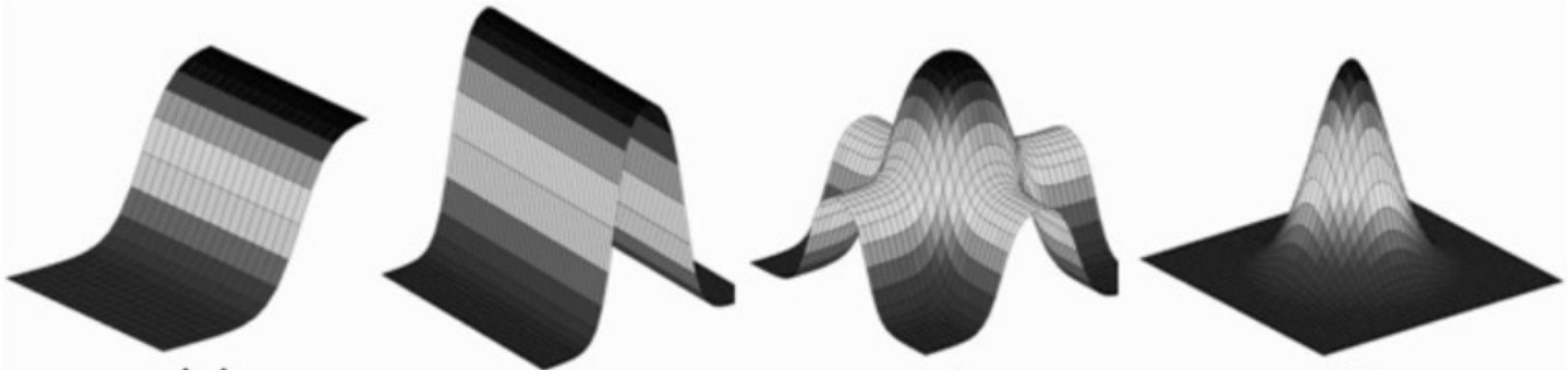
$$p_2 \equiv x_1 + x_2 - 0.5 \geq 0$$

$$o \equiv p_1 + p_2 - 1.5 \geq 0$$



A Universal Approximator

UNIVERSITY OF LEEDS



$$g(x)=\sum_{j}^{N} w_j \sigma(y_j^T x+\theta_j)$$ given $q(x)$ $\epsilon>0$

$$|g(x)-q(x)|<\epsilon$$

Error definition

UNIVERSITY OF LEEDS

$$E(\boldsymbol{X}) = \sum_{\boldsymbol{x}_n \in \boldsymbol{X}} |y_n - t_n|$$

Number of errors on the training set

$$E_p(\boldsymbol{X}) = \sum_{\boldsymbol{x}_n \in \boldsymbol{X}} \boldsymbol{w}^T \boldsymbol{x}_n (y_n - t_n)$$

The Perceptron error

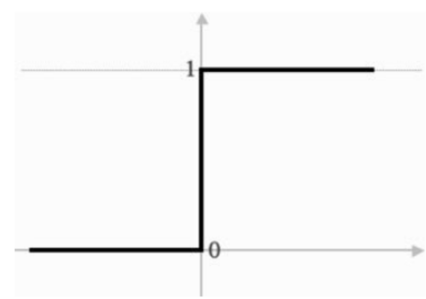
$$E_m(\boldsymbol{X}) = \frac{1}{2} \sum_{\boldsymbol{x}_n \in \boldsymbol{X}} (y_n - t_n)^2$$

Squared error function (differentiable!)
Usually known as the Mean Squared Error (MSE)

Output is differentiable if f is

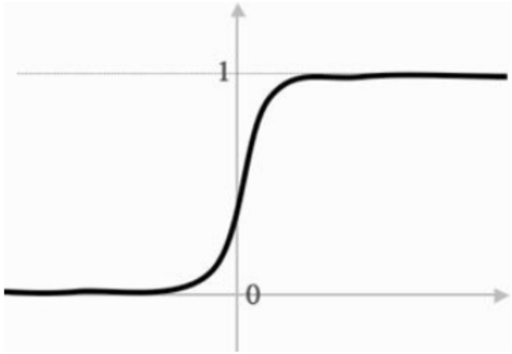
$$y = f\left(\sum_{i=1}^{M} w_i x_i\right)$$

$f =$



Not good

A different activation function

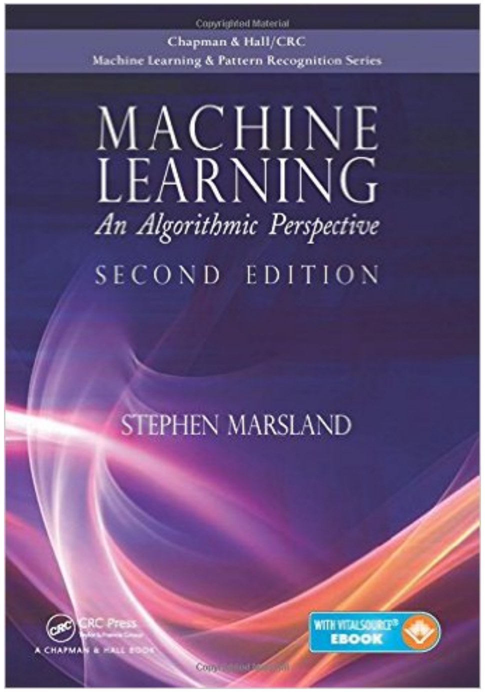


The sigmoid function: $$f(x)=\frac{1}{1+e^{-\beta x}}\equiv\sigma_\beta$$

$$\sigma_\beta'(x)=?$$

Conclusion

UNIVERSITY OF LEEDS



Chapter 4