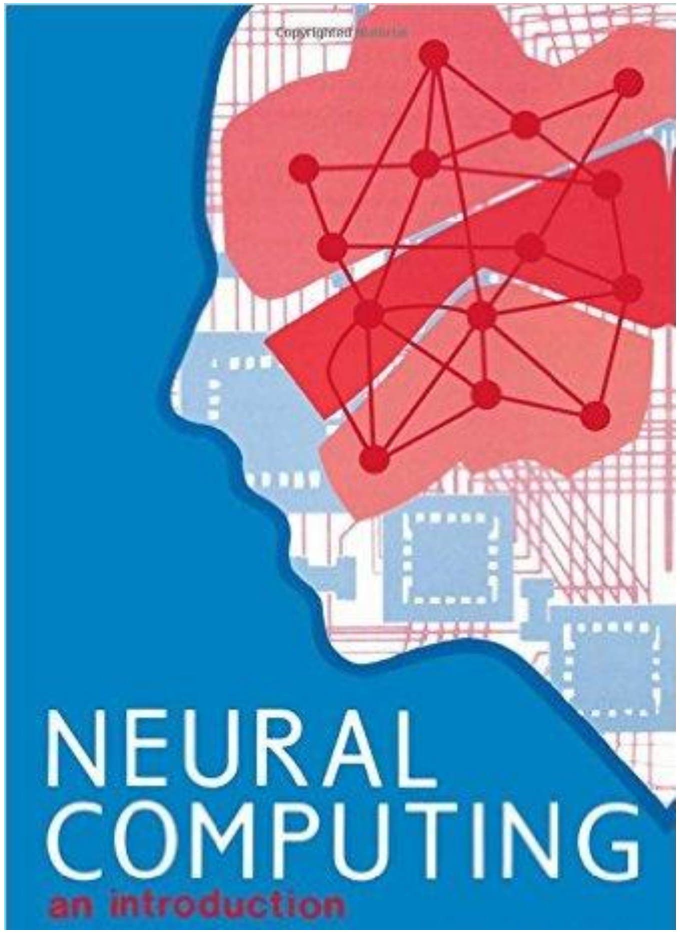
College of Computing
and Information Technology

CS366 Introduction to A.I.

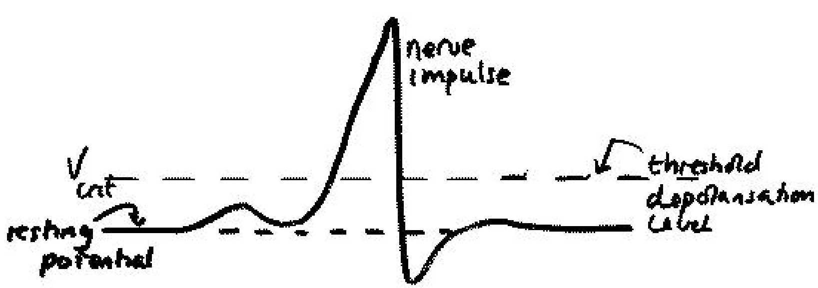
Dr. Mohamed Farouk.



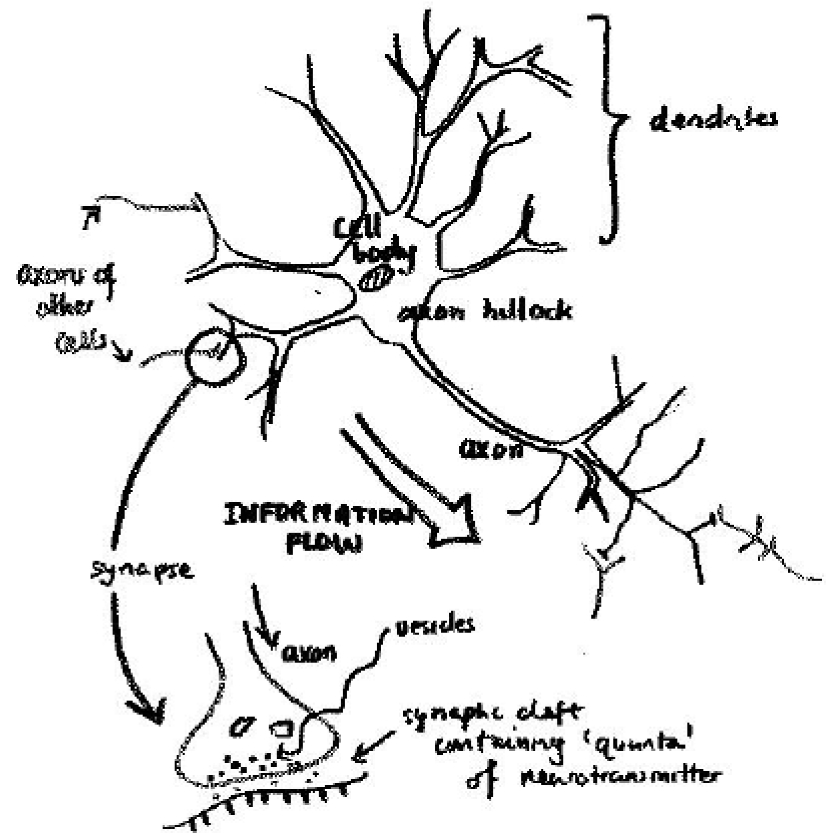
The effects of presynaptic ('input') neurons are summed at the axon hillock.

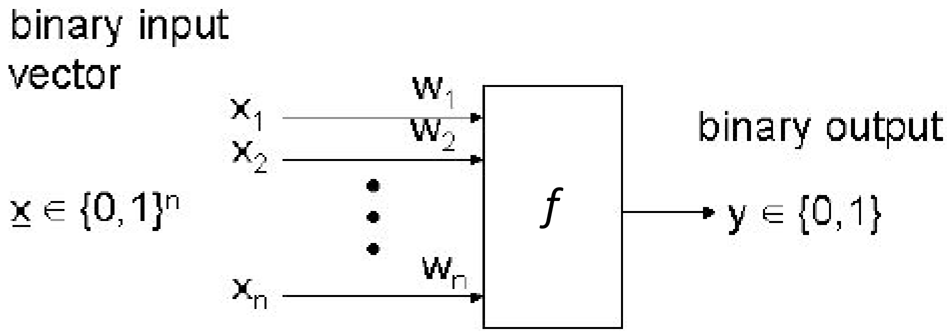
Some of these effects are excitatory (making the neuron more likely to become active itself), some are inhibitory (making the neuron receiving the signal less likely to be active).

A neuron is a decision unit. It fires (transmits an electrical signal down its axon which travels without decrement to the dentritic trees of other neurons) if the electrical potential V at the axon hillock exceeds a threshold value Vcrit of about 15mV.



THE BIOLOGICAL PROTOTYPE



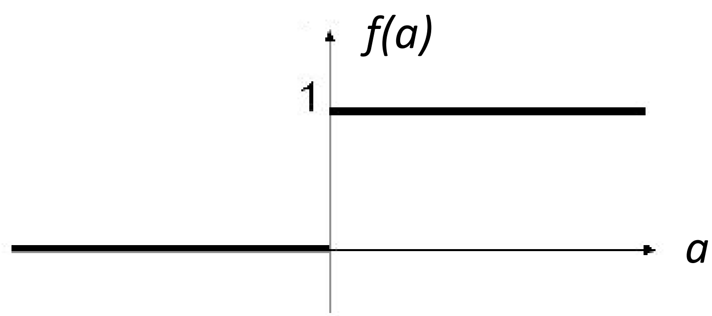


The weighted signals are summed, and this sum is then compared to a threshold s:

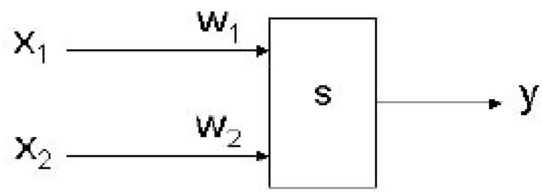
$$y = 0 \text{ if } \sum_{j=1}^{n} w_j x_j \le s$$

$$y = 1 \text{ if } \sum_{j=1}^{n} w_j x_j > s$$

$$a = \sum_{j=1}^{n} w_j x_j - s$$



An example for the activation *f*unction is the step *f*unction



Example: AND function $x_1 \wedge x_2$

A possible choice of parameters is $w_1 = w_2 = 1.0,\ s = 1.0$

x_1	x_2	a	$y=\theta(a)$
0	0	-1.0	0
0	1	0.0	0
1	0	0.0	0
1	1	1.0	1

Example: OR function $x_1 \vee x_2$

A possible choice of parameters is $w_1 = w_2 = 1.0,\ s = 0.0$

x_1	x_2	a	$y=\theta(a)$
0	0	0.0	0
0	1	1.0	1
1	0	1.0	1
1	1	2.0	1

Consider the example $y = x_1$ OR x_2 for which the two pattern classes are

$A = \{ (0,1), (1,0), (1,1) \}$
$B = \{ (0,0) \}$

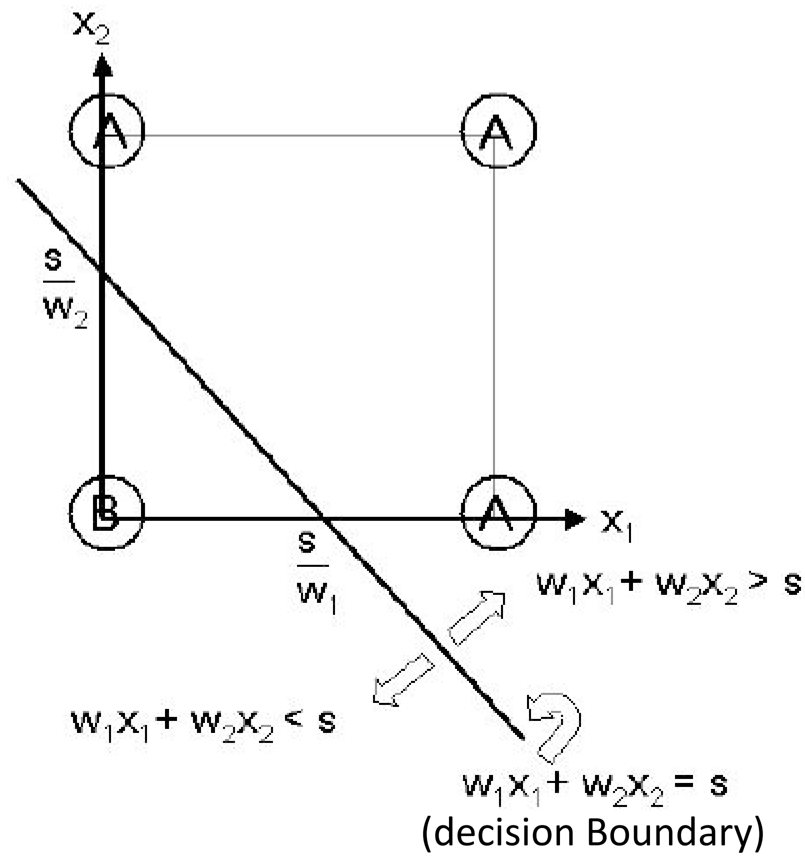
The line

$$w_1 x_1 + w_2 x_2 = s$$

or equivalently

$$x_2 = \frac{s}{w_2} - \frac{w_1}{w_2} x_1$$

The line divides the plane into two regions, one of which contains the 'class A' vertices, one the 'class B' vertices.

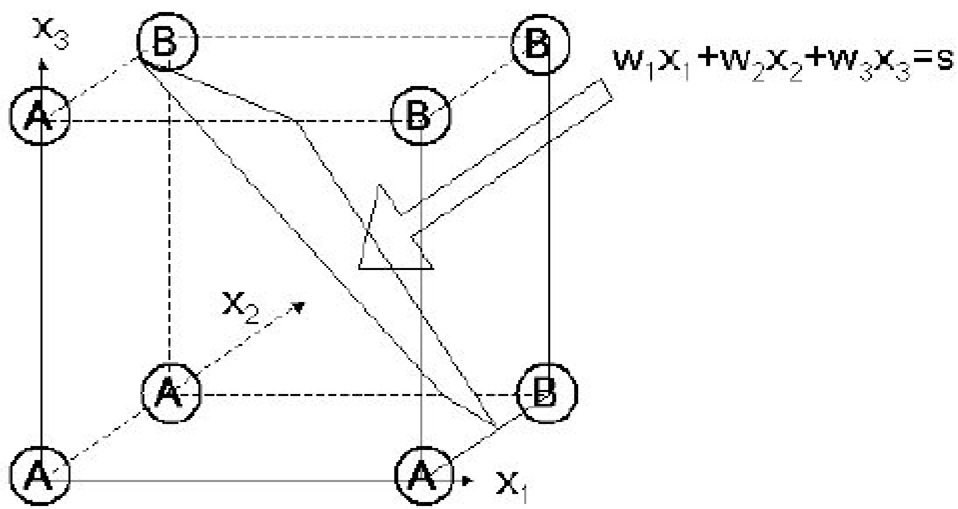


LINEAR SEPARABILITY

Example: for the 3-bit classifier of p.24

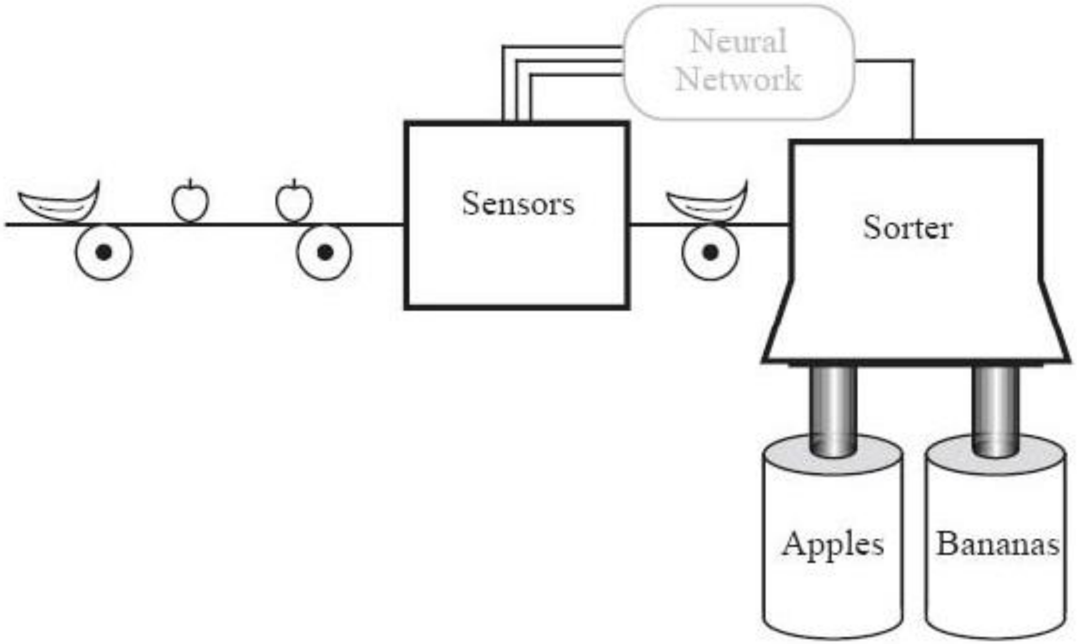
A = { (0,0,0), (0,0,1), (0,1,0), (1,0,0) }
B = { (1,1,1), (1,1,0), (1,0,1), (0,1,1) }

and the pattern separation after training, with the weights then acquired, looks in 3D like

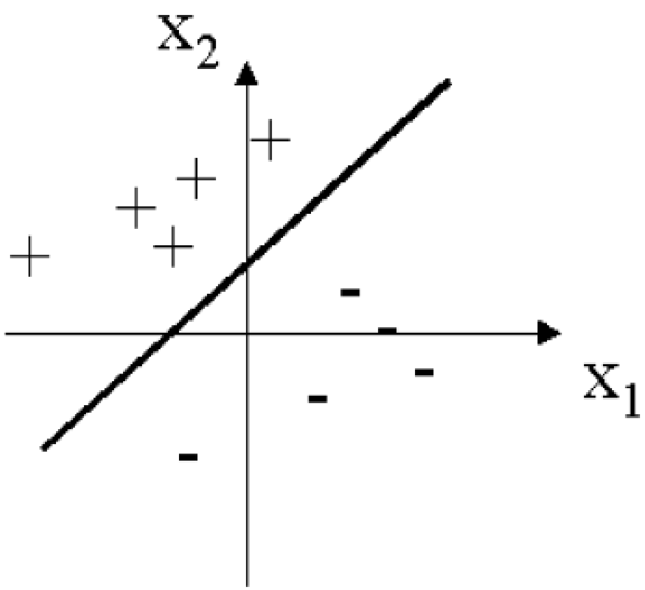


For an n-input perceptron unit the input patterns occupy the vertices of the **n-dimensional hypercube $[0,1]^n$**. Pattern separation is achieved by means of an **(n-1)-dimensional hyperplane $w_1x_1+\ldots+w_nx_n = s$**.

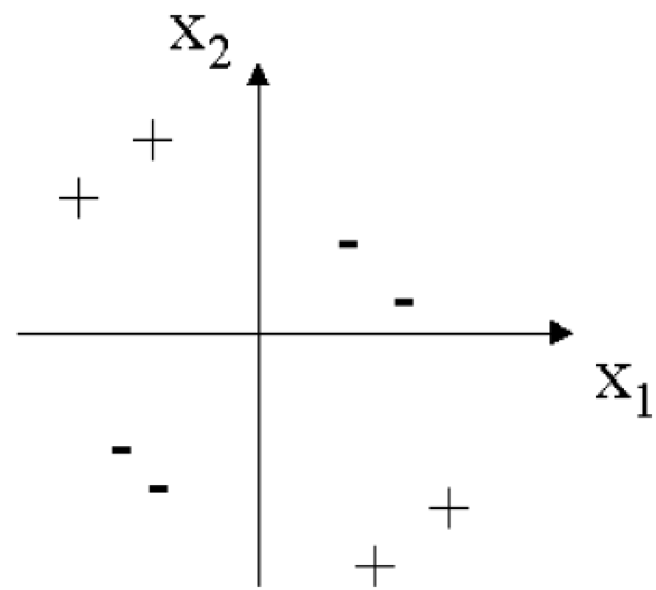
A more complex example



Linearly Separable



Linearly Separable



Not Linearly Separable

$$w_1 x_1 + w_2 x_2 + \theta = 0$$

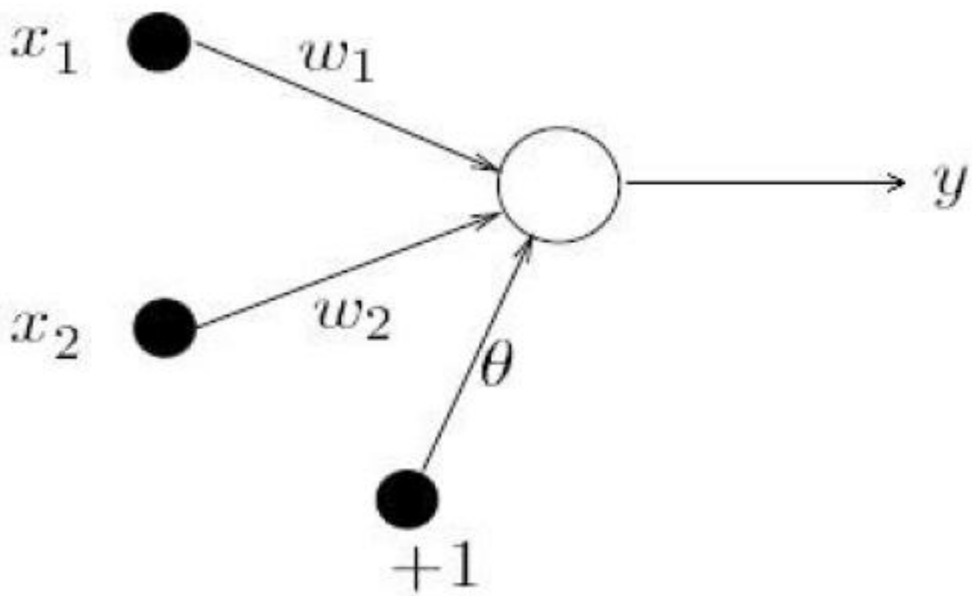
Perceptron Learning Algorithm

- We want to train the perceptron to classify inputs correctly
- Accomplished by adjusting the connecting weights and the bias
- Can only properly handle linearly separable sets

Perceptron Learning Algorithm

- We have a "training set" which is a set of input vectors used to train the perceptron.
- During training both w_i and θ *(bias)* are modified for convenience, let $w_0 = \theta$ and $x_0 = 1$
- *Let,* η, the *learning rate*, be a small positive number (small steps lessen the possibility of destroying correct classifications)
- Initialise w_i to some values

Simple Perceptron



Perceptron Learning Algorithm

$$\textit{Desired output} \qquad d(n) = \begin{cases} +1 & if\ x(n) \in set\ A \\ -1 & if\ x(n) \in set\ B \end{cases}$$

1. Select random sample from training set as input
2. If classification is correct, do nothing
3. If classification is incorrect, modify the weight vector *w* using

$$w_i = w_i + \eta d(n) x_i(n)$$

Repeat this procedure until the entire training set is classified correctly

Simple Perceptron

- Simplest output function

$$y = \operatorname{sgn}\left(\sum_{i=1}^{2} w_i x_i + \theta\right)$$

$$\operatorname{sgn}(s) = \begin{cases} 1 & \text{if } s > 0 \\ -1 & \text{otherwise.} \end{cases}$$

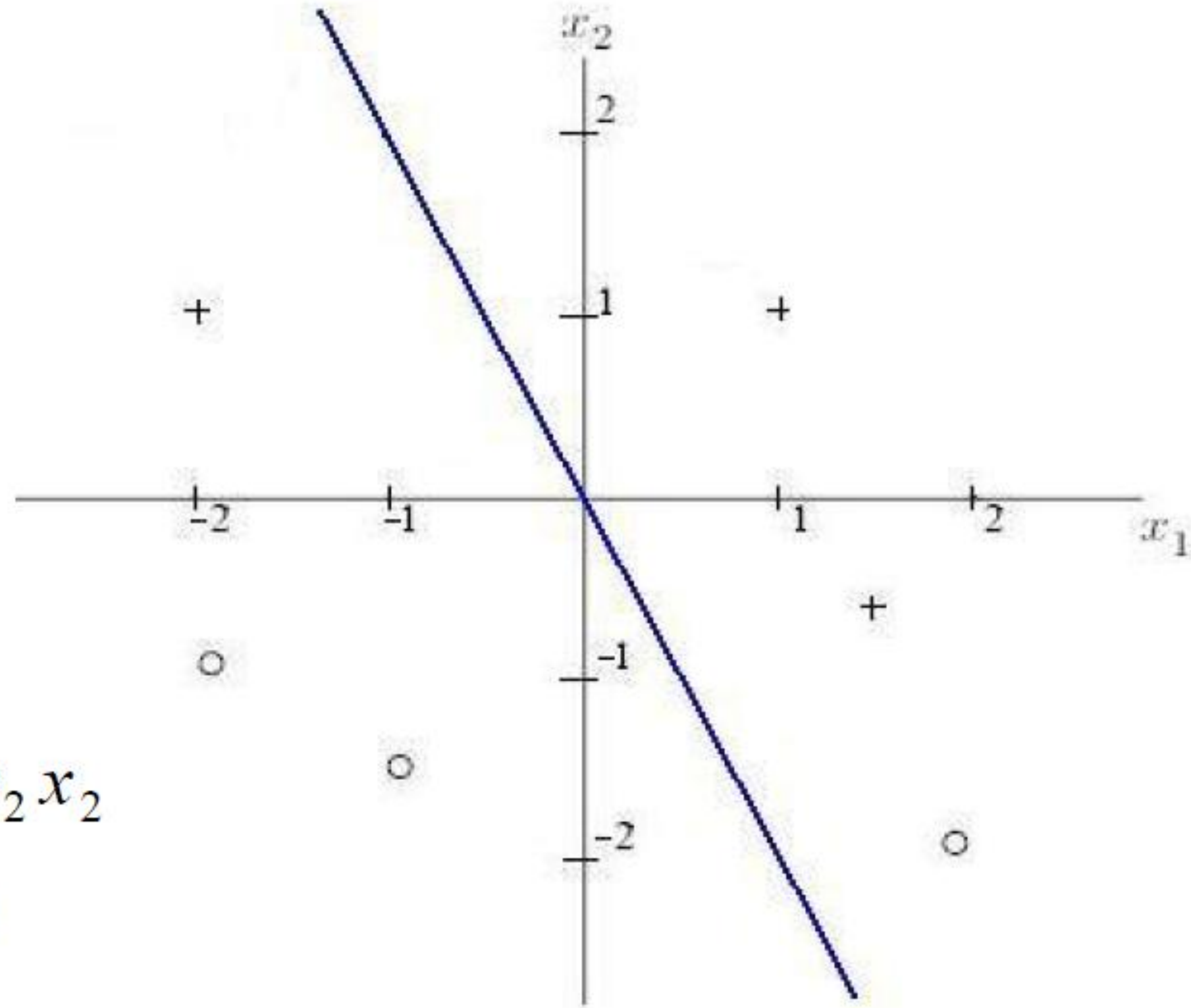
- Used to classify patterns said to be linearly separable

Learning Example

Initial Values:

$\eta = 0.2$

$$w = \begin{pmatrix} 0 \\ 1 \\ 0.5 \end{pmatrix}$$

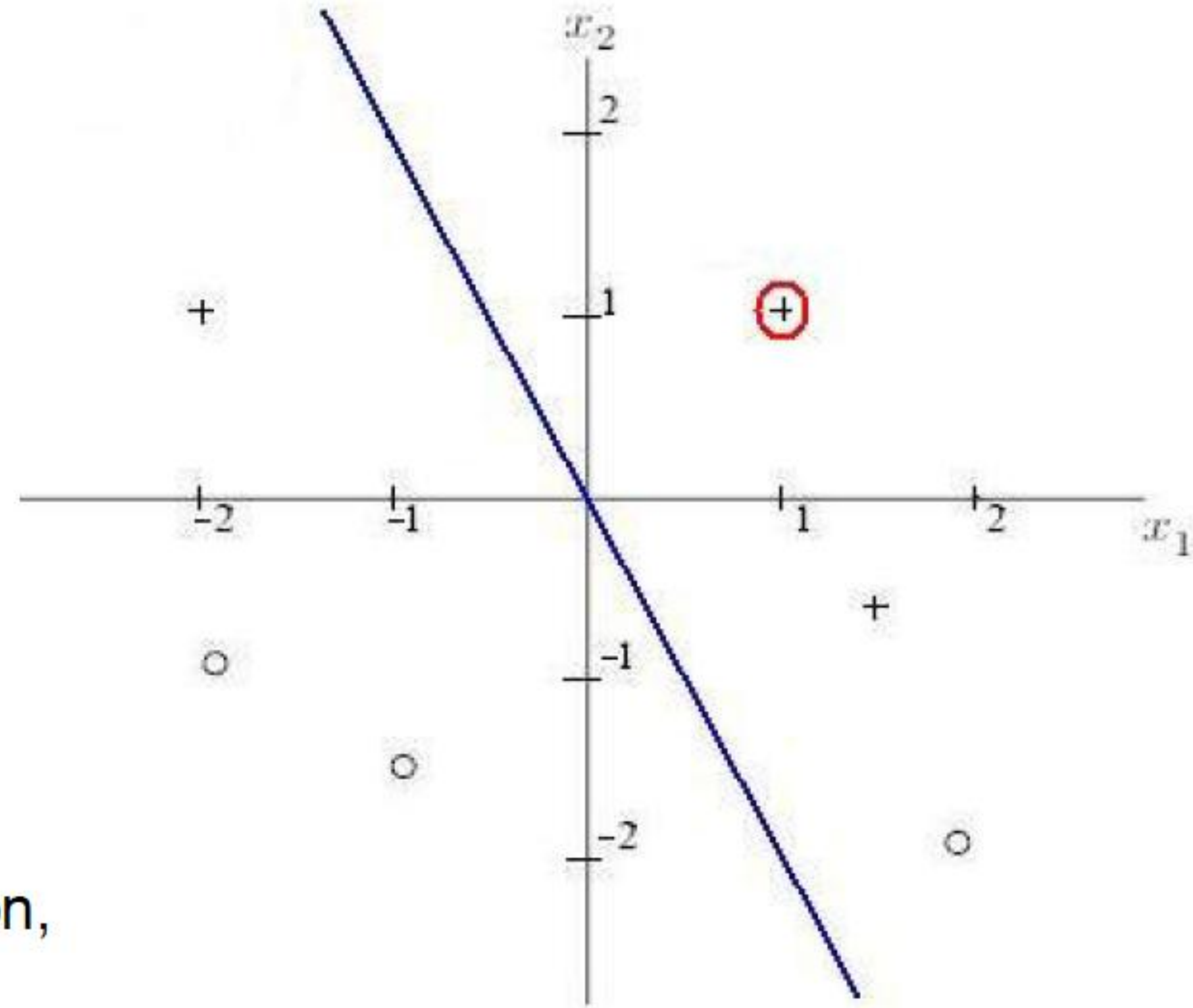


$$\begin{aligned} 0 &= w_0 + w_1 x_1 + w_2 x_2 \\ &= 0 + x_1 + 0.5 x_2 \\ \Rightarrow \ x_2 &= -2 x_1 \end{aligned}$$

Learning Example

$\eta = 0.2$

$$w = \begin{pmatrix} 0 \\ 1 \\ 0.5 \end{pmatrix}$$



$x_1 = 1,\ x_2 = 1$
$w^T x > 0$

Correct classification, no action

Learning Example

$\eta = 0.2$

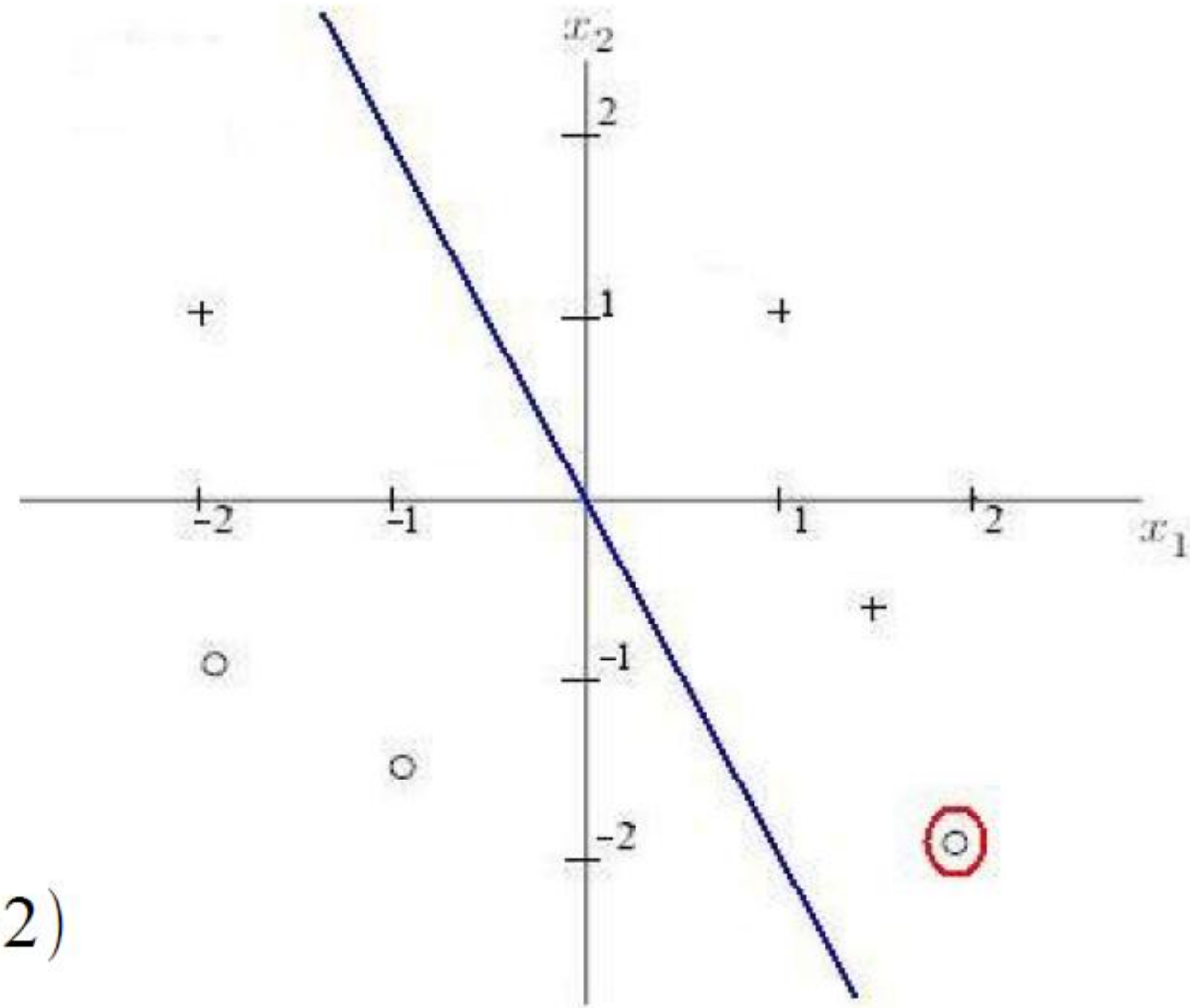
$$w = \begin{pmatrix} 0 \\ 1 \\ 0.5 \end{pmatrix}$$

$x_1 = 2, x_2 = -2$

$$w_0 = w_0 - 0.2 * 1$$

$$w_1 = w_1 - 0.2 * 2$$

$$w_2 = w_2 - 0.2 * (-2)$$



Learning Example

$\eta = 0.2$

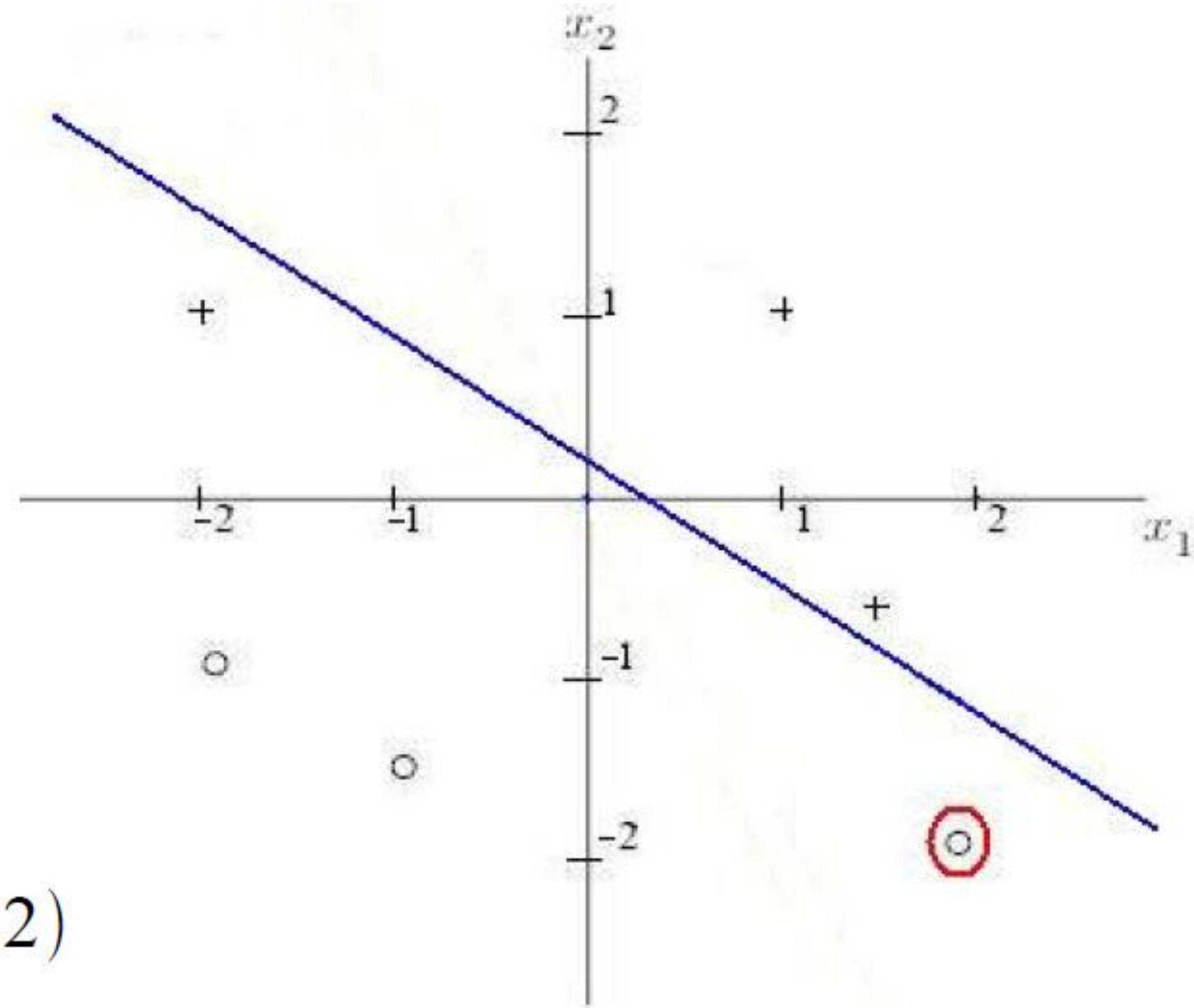
$$w = \begin{pmatrix} -0.2 \\ 0.6 \\ 0.9 \end{pmatrix}$$

$x_1 = 2, x_2 = -2$

$$w_0 = w_0 - 0.2 * 1$$

$$w_1 = w_1 - 0.2 * 2$$

$$w_2 = w_2 - 0.2 * (-2)$$



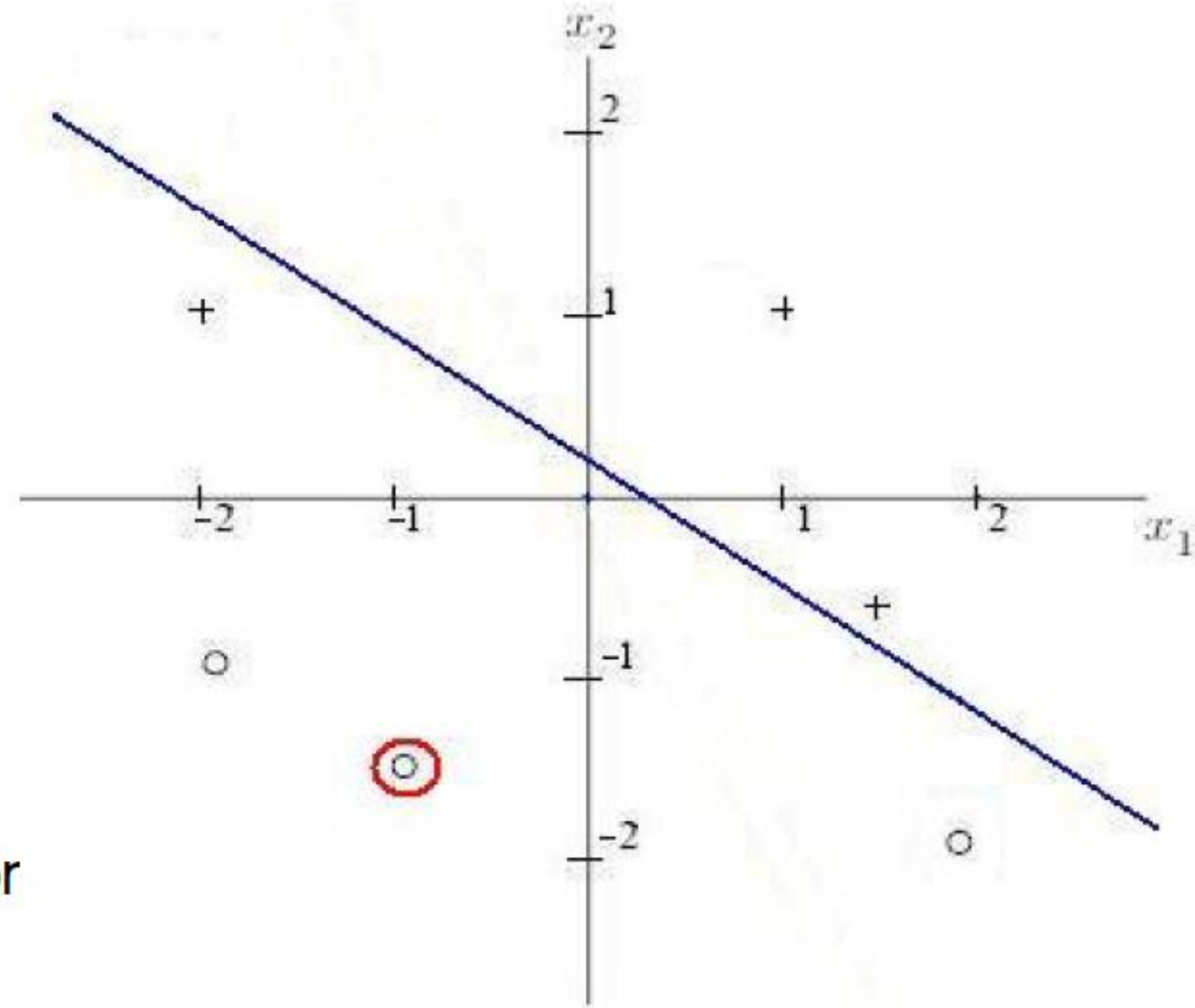
Learning Example

$\eta = 0.2$

$$w = \begin{pmatrix} -0.2 \\ 0.6 \\ 0.9 \end{pmatrix}$$

$x_1 = -1, x_2 = -1.5$
$w^T x < 0$

Correct classificatior
no action



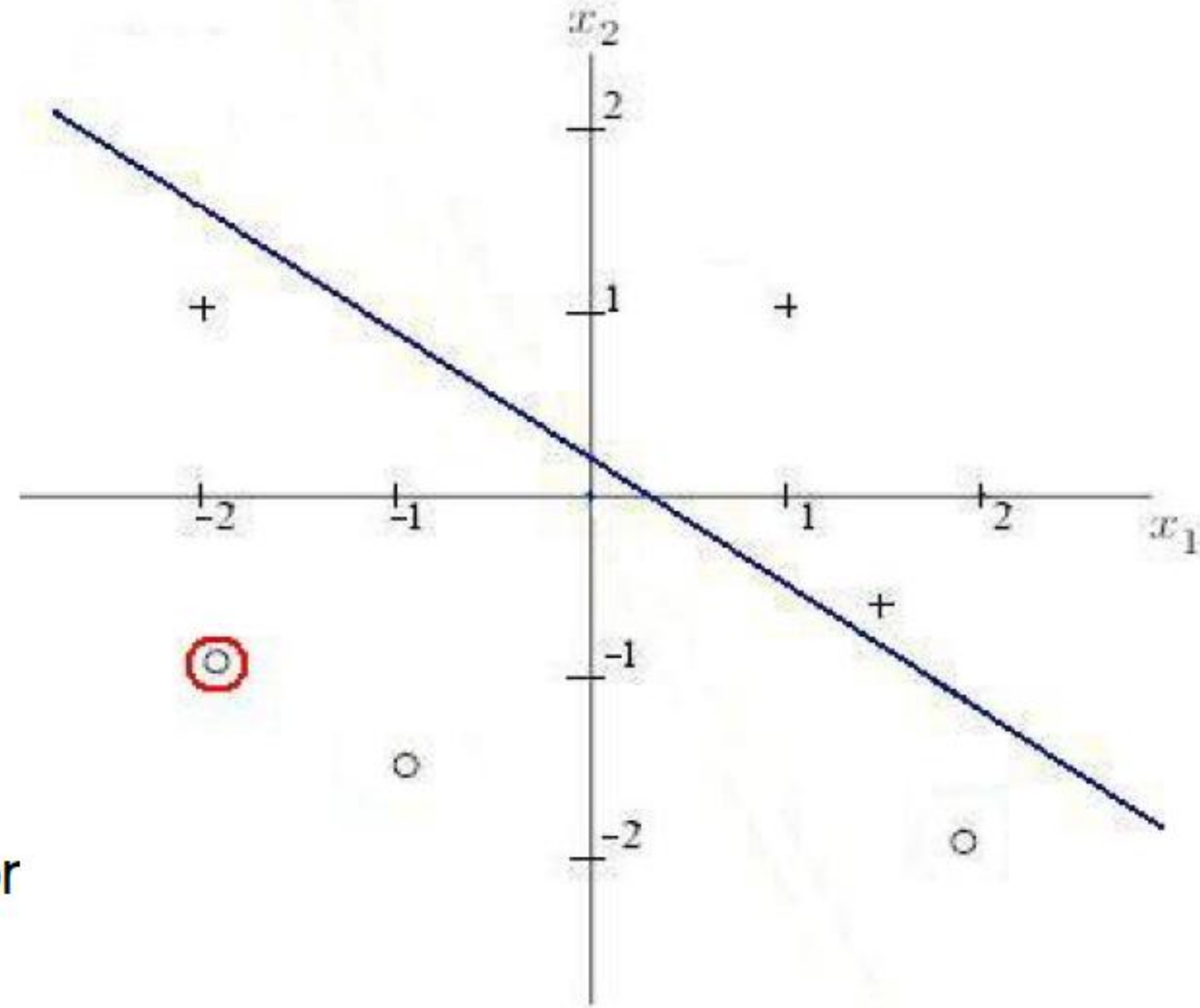
Learning Example

$\eta = 0.2$

$$w = \begin{pmatrix} -0.2 \\ 0.6 \\ 0.9 \end{pmatrix}$$

$x_1 = -2,\ x_2 = -1$
$w^T x < 0$

Correct classificatior
no action



Learning Example

$\eta = 0.2$

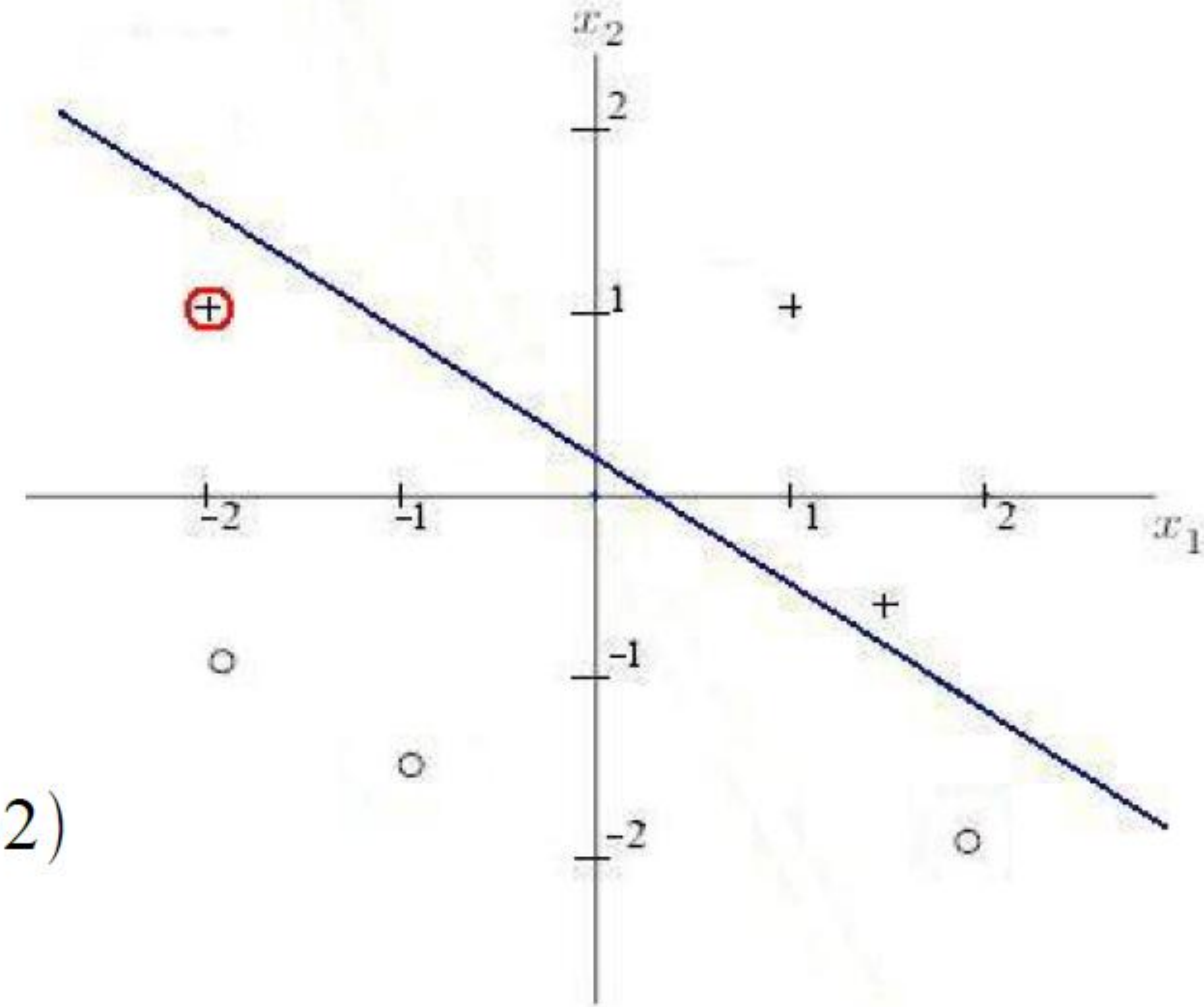
$$w = \begin{pmatrix} -0.2 \\ 0.6 \\ 0.9 \end{pmatrix}$$

$x_1 = -2,\ x_2 = 1$

$$w_0 = w_0 + 0.2 * 1$$

$$w_1 = w_1 + 0.2 * (-2)$$

$$w_2 = w_2 + 0.2 * 1$$



Learning Example

$\eta = 0.2$

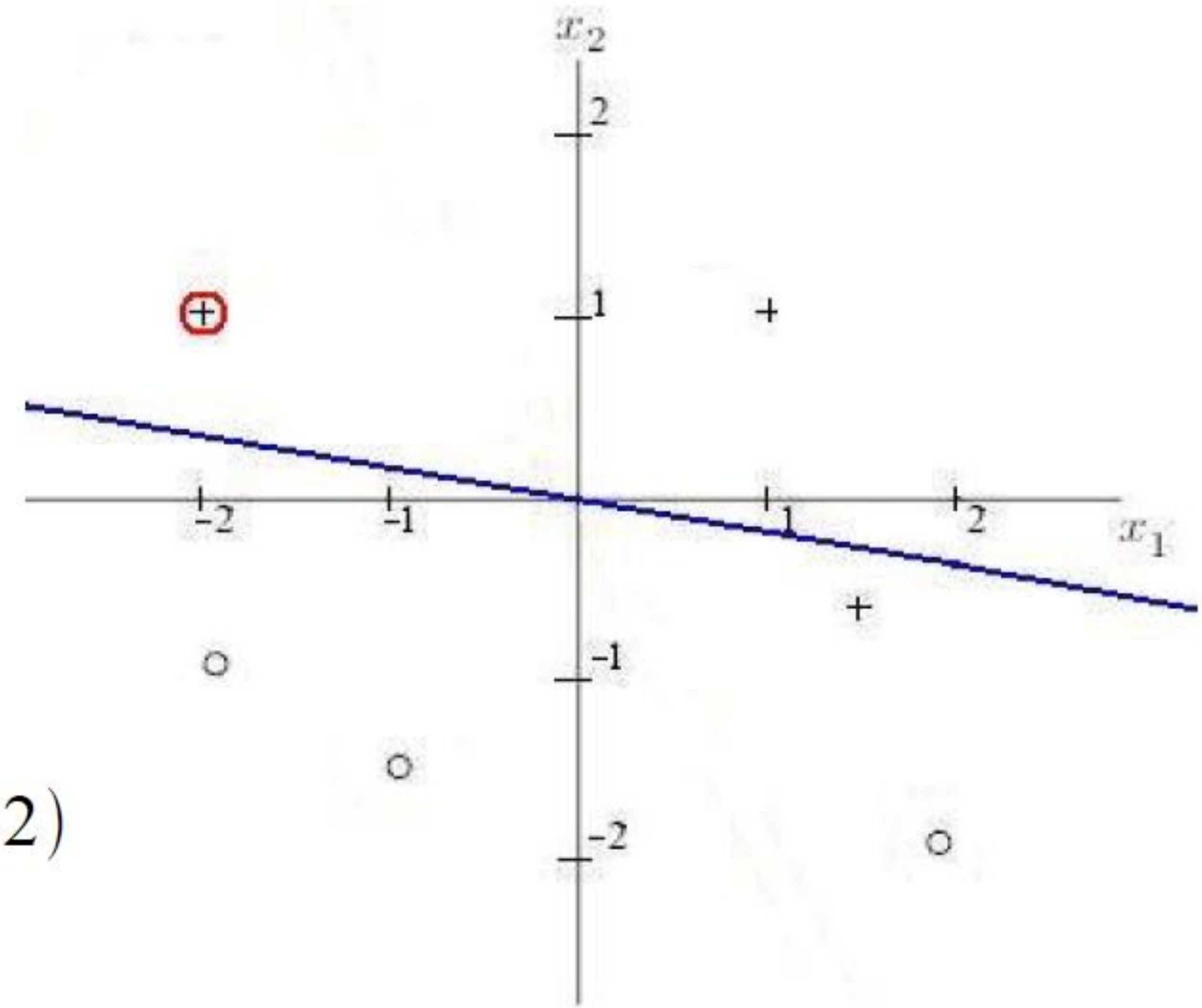
$$w = \begin{pmatrix} 0 \\ 0.2 \\ 1.1 \end{pmatrix}$$

x_1 = -2, x_2 = 1

$$w_0 = w_0 + 0.2 * 1$$

$$w_1 = w_1 + 0.2 * (-2)$$

$$w_2 = w_2 + 0.2 * 1$$



Learning Example

$\eta = 0.2$

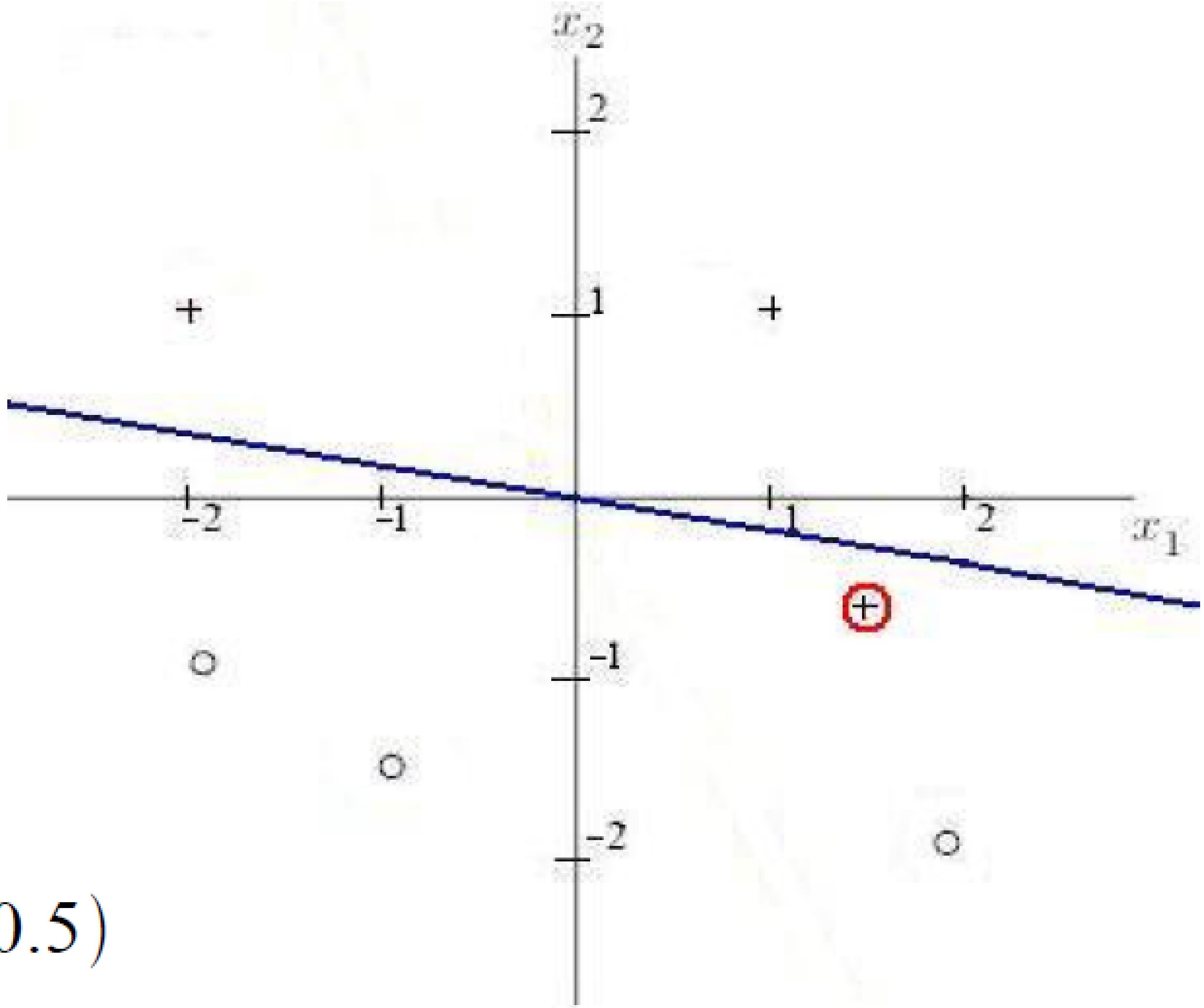
$$w = \begin{pmatrix} 0 \\ 0.2 \\ 1.1 \end{pmatrix}$$

$x_1 = 1.5,\ x_2 = -0.5$

$$w_0 = w_0 + 0.2 * 1$$

$$w_1 = w_1 + 0.2 * 1.5$$

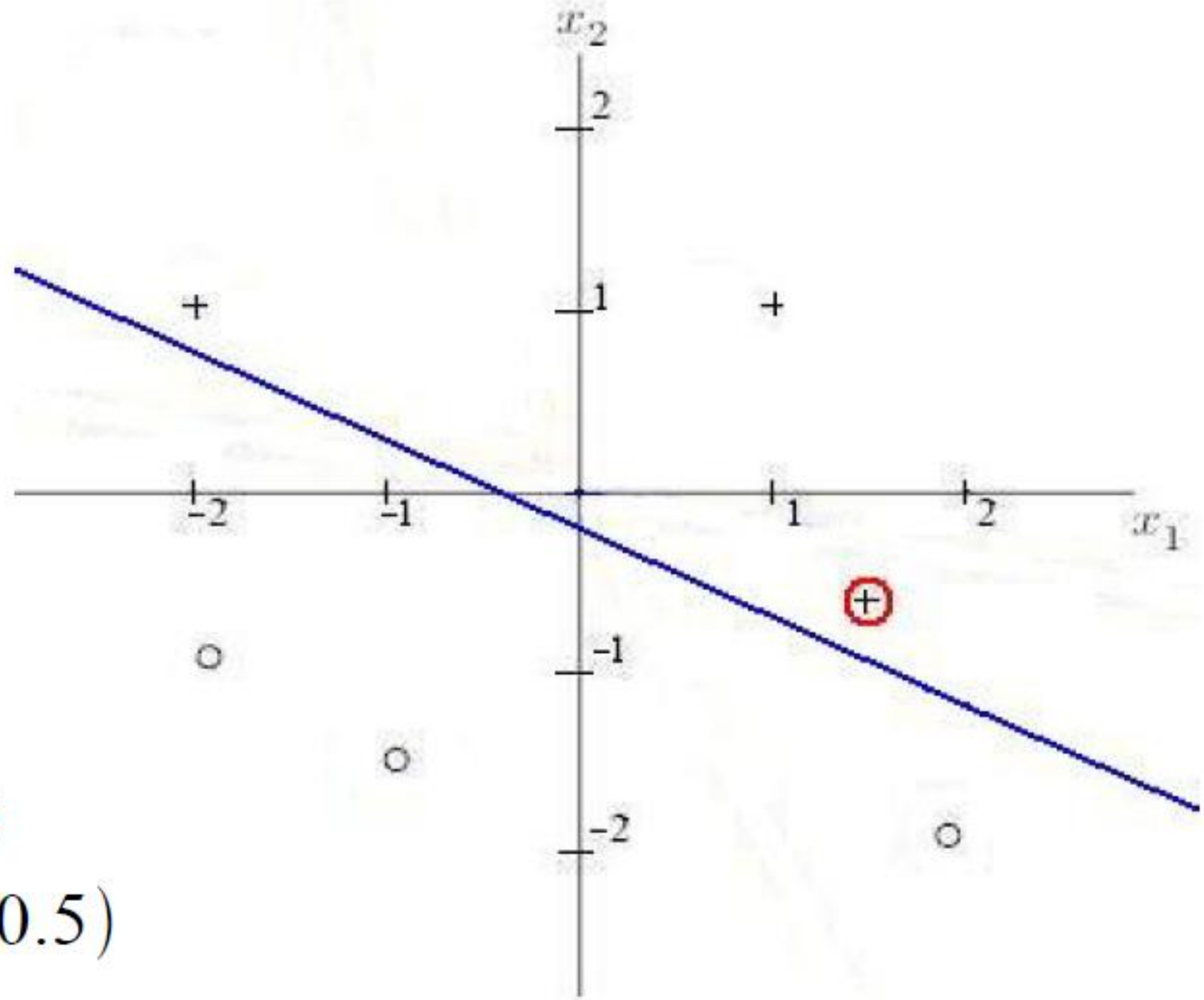
$$w_2 = w_2 + 0.2 * (-0.5)$$



Learning Example

$\eta = 0.2$

$$w = \begin{pmatrix} 0.2 \\ 0.5 \\ 1 \end{pmatrix}$$



$x_1 = 1.5$, $x_2 = -0.5$

$$w_0 = w_0 + 0.2 * 1$$

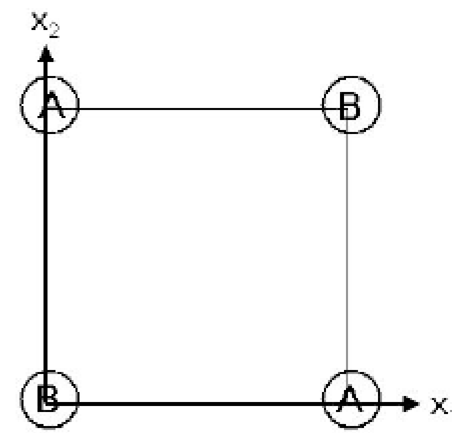
$$w_1 = w_1 + 0.2 * 1.5$$

$$w_2 = w_2 + 0.2 * (-0.5)$$

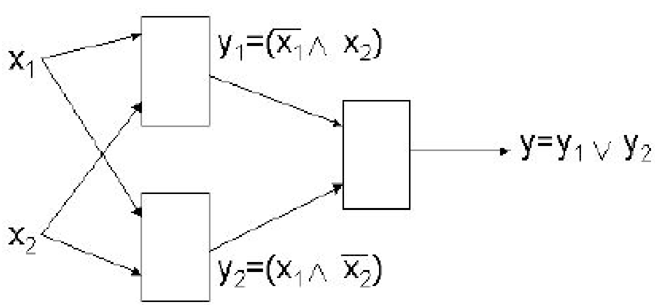
LIMITATIONS OF PERCEPTRONS

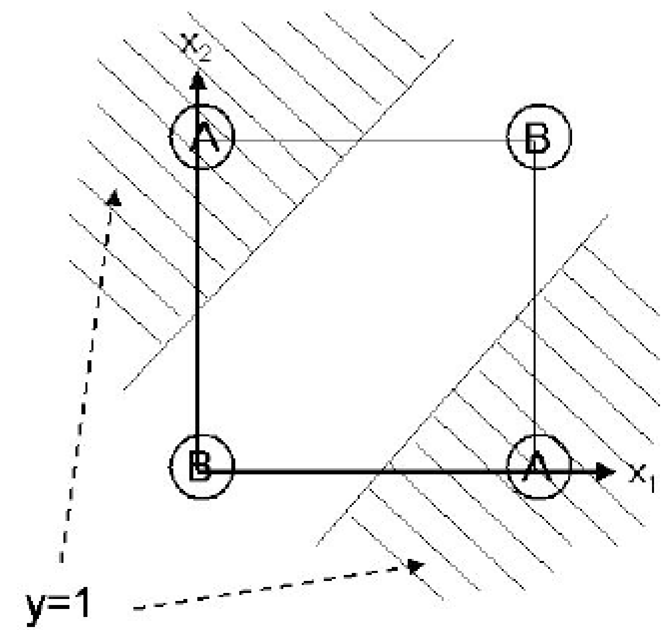
The root of the problem can be seen in the simplest such example, the XOR problem (2-bit 'not-parity'), defined by the truth table

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

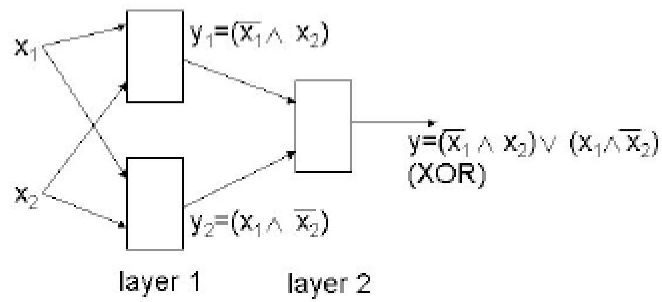


Overcoming the linear separability restriction using multiple layers

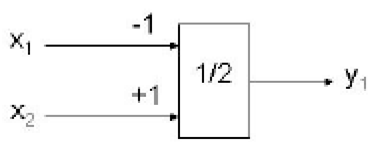


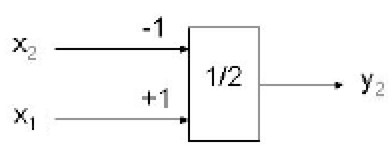


Layer 1 **Layer 2**



x_1	x_2	$w_{11}x_1 + w_{12}x_2 - s_1$	output y_1
0	0	$-\frac{1}{2}$	0
0	1	$\frac{1}{2}$	1
1	0	$-\frac{3}{2}$	0
1	1	$-\frac{1}{2}$	0





x_1	x_2	y_1	y_2	$w_1y_1 + w_2y_2 - s$	final output y
0 1	0 1	0	0	$-\frac{1}{2}$	0
1	0	0	1	$\frac{1}{2}$	1
0	1	1	0	$\frac{1}{2}$	1
-	-	1	1	$\frac{3}{2}$	1