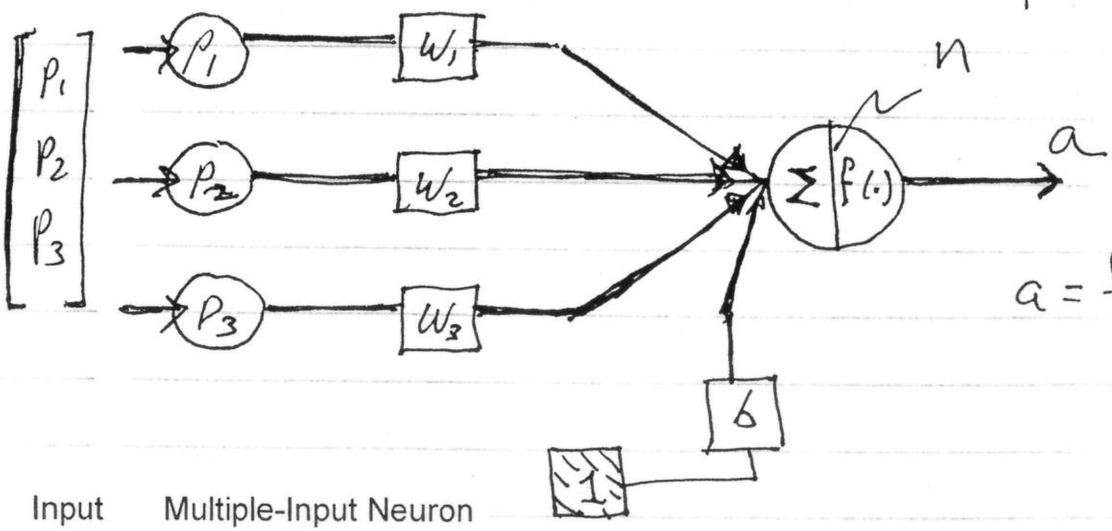
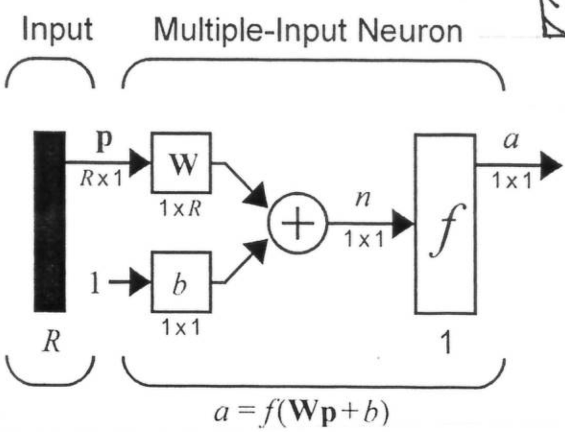
EACH P.E. (Perceptron) implements $\underline{1}$ LINEAR BOUNDARY

$\bar{p}$ $\qquad \bar{W} = [w_1, w_2, w_3] \qquad n = \bar{W}\bar{p} + b$



$a = f(n)$



this is because n is a linear (1st order) combination of the p_i features.

Example:

$\bar{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$; $\bar{W} = [w_{11} \;\; w_{12}]$, then: $n = \bar{W}\bar{p} + b$

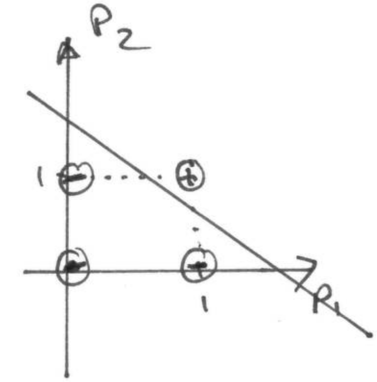
$n = 0 = w_{11} p_1 + w_{12} p_2 + b$, solving for p_2:

$p_2 = \left(\frac{-w_{11}}{w_{12}}\right) p_1 + \left(\frac{-b}{w_{12}}\right)$ } straight line, NO second order terms or higher

"$y = mx + b$"

Example: 2-in binary AND

p_1	p_2	a
0	0	0
0	1	0
1	0	0
1	1	1



"pattern" or "input" SPACE

Example: 3-in binary AND:

p_1	p_2	p_3	a
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

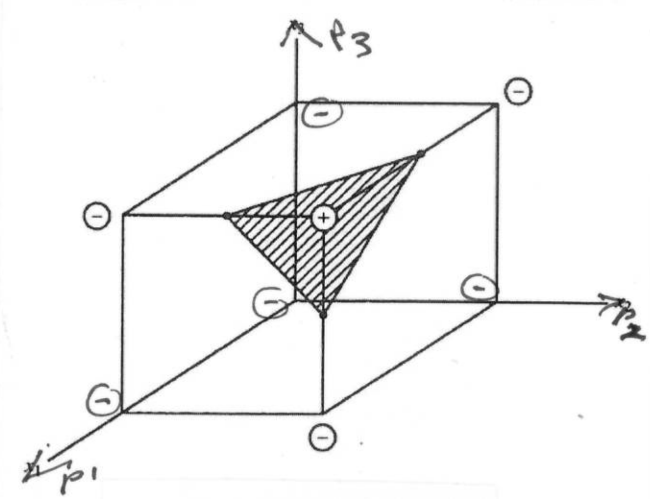


Figure 2.12 Linear separation of binary training inputs.