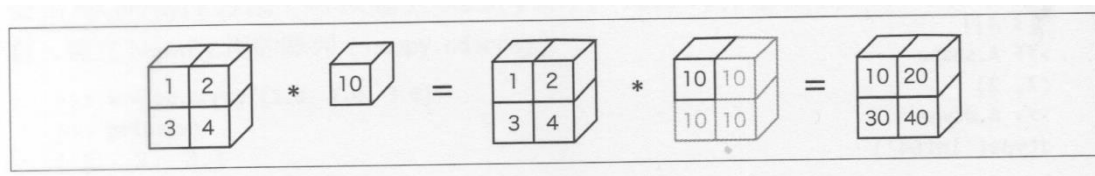
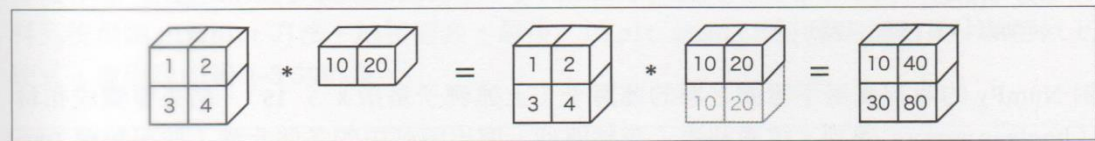


Numpy 廣播



由於 NumPy 具有廣播功能，所以能用聰明的方式，對不同形狀的陣列進行運算。



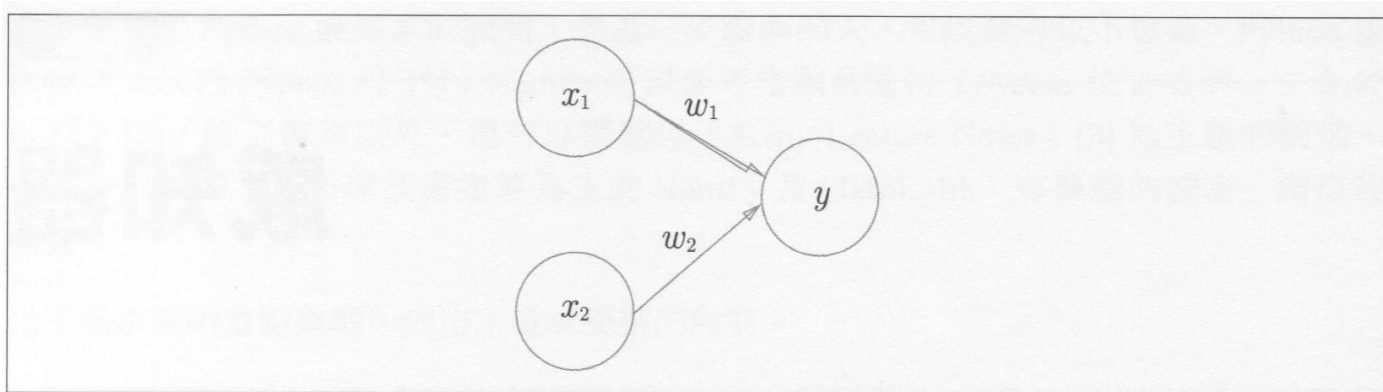


圖 2-1 接收 2 個輸入訊號的感知器

$$y = \begin{cases} 0 & (w_1x_1 + w_2x_2 \leq \theta) \\ 1 & (w_1x_1 + w_2x_2 > \theta) \end{cases}$$

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

圖 2-2 及閾的真值表

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

圖 2-3 反及閾的真值表

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

圖 2-4 或閘的真值表

$$y = \begin{cases} 0 & (b + w_1x_1 + w_2x_2 \leq 0) \\ 1 & (b + w_1x_1 + w_2x_2 > 0) \end{cases}$$

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

圖 2.5 互斥或閘的真值表

$$y = \begin{cases} 0 & (-0.5 + x_1 + x_2 \leq 0) \\ 1 & (-0.5 + x_1 + x_2 > 0) \end{cases}$$

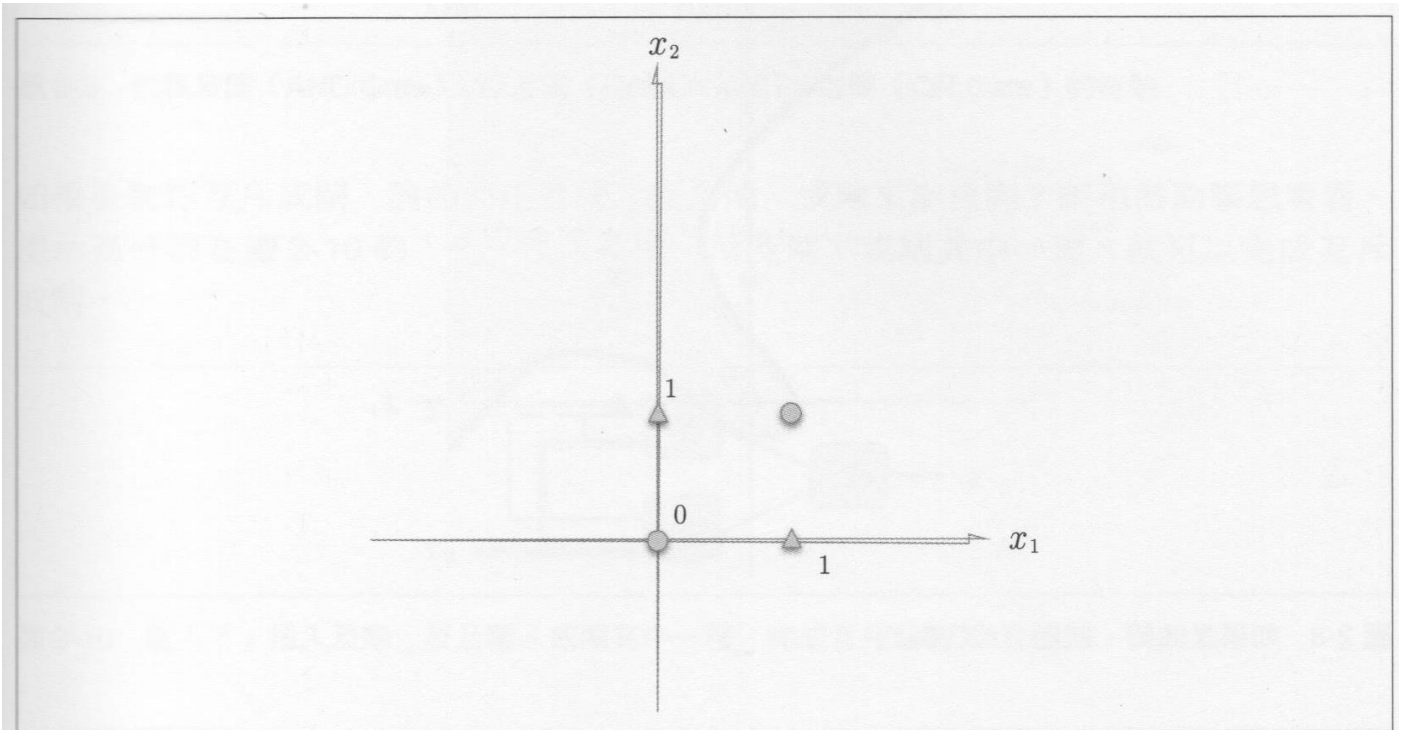


圖 2-7 ○與△代表互斥或閘的輸出。利用直線可以分成○與△區域嗎？

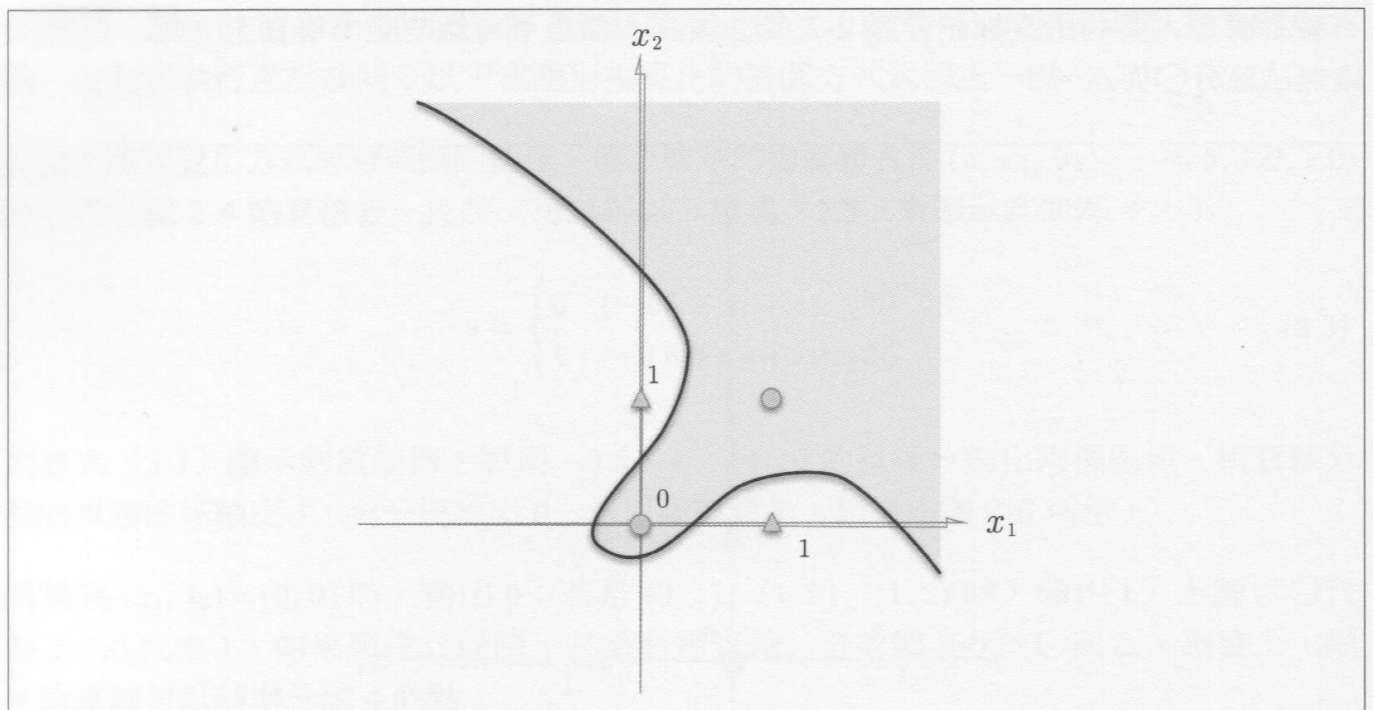


圖 2-8 如果是曲線，就能分成○與△

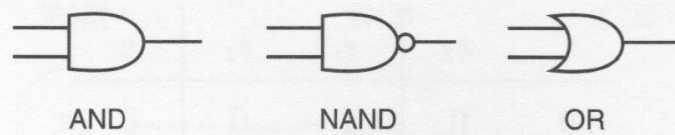


圖 2-9 代表及閘（AND Gate）、反及閘（NAND Gate）、或閘（OR Gate）的符號

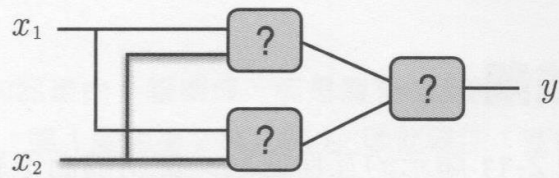


圖 2-10 在「？」插入及閘、反及閘、或閘其中一種，完成互斥或閘！

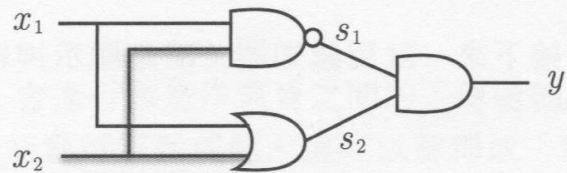


圖 2-11 搭配及閘、反及閘、或閘，完成互斥或閘

x_1	x_2	s_1	s_2	y
0	0	1	0	0
1	0	1	1	1
0	1	1	1	1
1	1	0	1	0

圖 2-12 互斥或閘的真值表

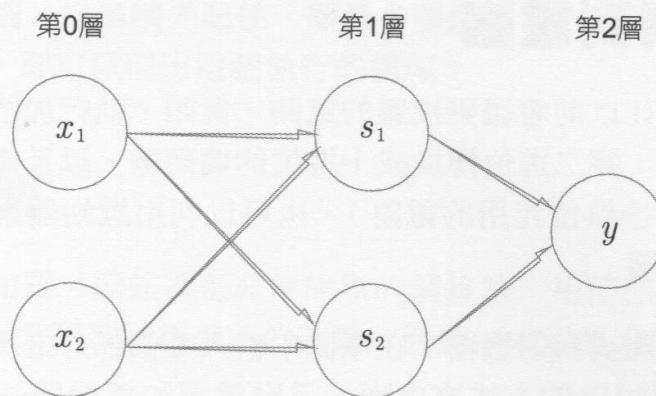


圖 2-13 利用 XOR 的感知器來顯示