Task 1:

(1)

Um das Perzeptron für die boolesche Funktion A ∧ ¬B abzubilden, darf das Perzeptron nur feuern, wenn für die 2 Eingaben A=1 und B=0 gilt. Siehe Tabelle.

A	B	A ∧ ¬B
0	0	0
0	1	0
1	**0**	**1**
1	1	0

Als Ausgabe-Funktion wird die Heaviside-Funktion genutzt.

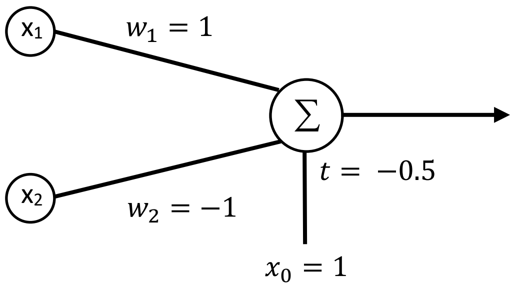
$$H: \mathbb{R} \rightarrow \{0,1\}, H(x) = \begin{cases} 0 : x < 0 \\ 1 : x \geq 0 \end{cases}$$

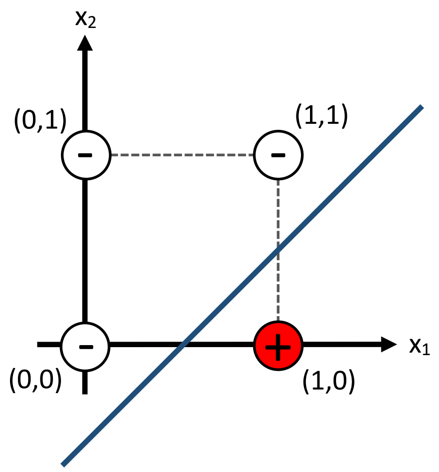
Für das Perzeptron mit x_1=A, x_2=B und t=Threshold muss also gelten

$$\begin{array}{lcl} x_1 = x_2 = 0 & \rightarrow & t < 0 \\ x_1 = 0,\ x_2 = 1 & \rightarrow & t + w_2 < 0 \\ x_1 = 1,\ x_2 = 0 & \rightarrow & t + w_1 \geq 0 \\ x_1 = x_2 = 1 & \rightarrow & t + w_1 + w_2 < 0 \end{array}$$

damit das Perzeptron die richtige Ausgabe erzeugt.

Um die passenden Gewichte zu erhalten setzen wir t=-0.5 und passen w_1 und w_2 so an, dass die Gleichungen erfüllt werden. Z.B. für w_1=1 und w_2=-1. Damit erhält man das Perzeptron mit Decision Boundary:

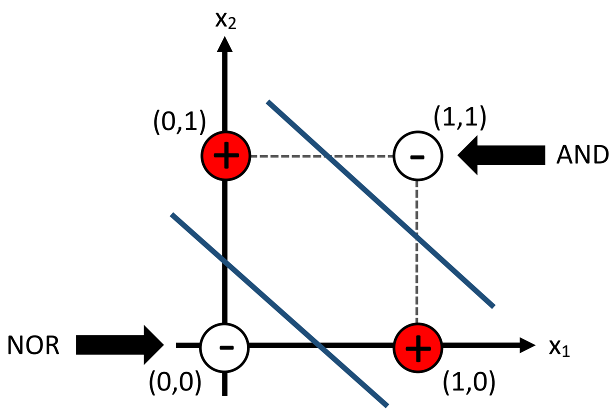




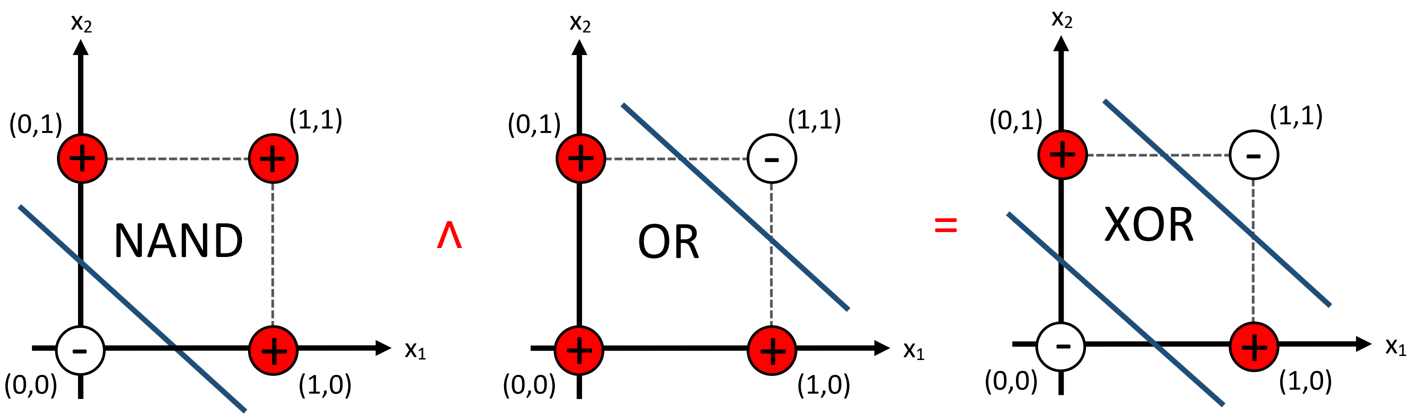
(2)

Das Problem um XOR abzubilden liegt darin, dass 2 Entscheidungsgrenzen (Siehe Bild) benötigt werden. Deswegen ist es nicht mit einem Perzeptron abbildbar. Aber wie auf dem Bild zu sehen, lässt sich XOR über die Negation von AND und NOR erreichen. D.h.

$$\begin{aligned} XOR &= \neg(AND \vee NOR) \\ &= NAND \wedge OR \end{aligned}$$

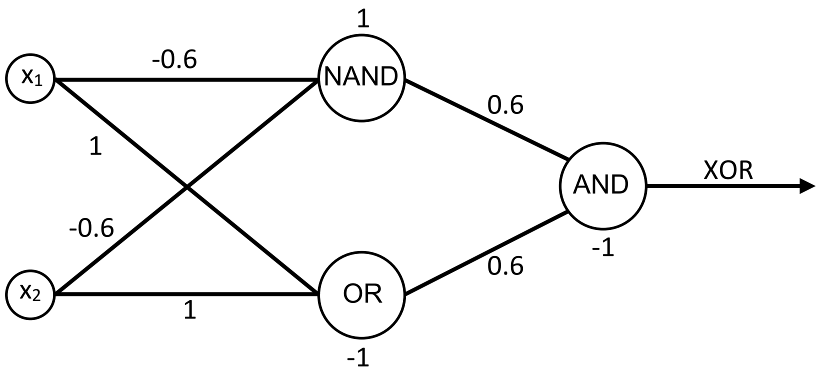


Was auch schnell ersichtlich ist, wenn man sich die Entscheidungs-grenzen anschaut.



Da NAND bei (0,0) und OR bei (1,1) niemals 1 wird ergibt sich durch Kombination dieser Perzeptronen die boolesche XOR-Funktion.

Das endgültige Perzeptronen-Netzwerk sieht dann folgendermaßen aus:



Die Anpassungen der Gewichte und Thresholds werden dann genauso gemacht, wie in (1) beschrieben. Dies geschieht völlig unabhängig von den anderen Perzeptronen. Je nach boolescher Funktion muss das Perzeptron eine entsprechende Schwelle übersteigen, damit eine 1 ausgegeben wird.