

**9. Aufgabe (2): Perzeptron-Lernregel**

Die verallgemeinerte Perzeptron-Lernregel unterscheidet sich von der Standard Perzeptron-Lernregel durch die zusätzliche Lernrate  $\eta$ . Zeigen Sie, dass bei einer Initialisierung der Gewichte und des Schwellwertes mit Null die Lernrate aus theoretischer Sicht keinen Einfluss auf das Lernverhalten haben kann!

**10. Aufgabe (3): 2-Schicht Perzeptron**

Es sei die folgende Boole'sche Funktion  $F : \{0, 1\}^3 \rightarrow \{0, 1\}$  gegeben:

$$(x_1, x_2, x_3) \xrightarrow{F} \begin{cases} 1 & x_1 + x_2 + x_3 = 2 \\ 0 & \text{sonst} \end{cases}$$

Zeigen Sie:  $F$  ist mit einem einzelnen Perzeptron nicht berechenbar, aber durch eine 2-Schicht-Architektur aus 2 Perzeptronen in der Zwischenschicht (und einem Perzeptron in der Ausgabeschicht).

Bestimmen Sie dazu (von Hand!) geeignete Gewichtswerte und Schwellwerte für alle Neuronen und erläutern Sie, warum diese Architektur  $F$  implementiert.

**11. Aufgabe (5): MLP und Error Backpropagation**

1. Implementieren Sie den Error Backpropagation Algorithmus im Online-Modus in `matlab` für ein  $n$ - $h$ - $p$  MLP. Die Neuronen der Zwischenschicht und der Ausgabeschicht sollen einen zusätzlichen konstanten Input (*Bias*) erhalten. Die Transferfunktion soll möglichst als Parameter übergeben werden können und muss zusammen mit der zugehörigen Ableitung ggf. spezifiziert werden. Gewichte und Bias sollen mit (gleichverteilten) (Pseudo-)Zufallszahlen im Bereich  $[-a, a]$ ,  $a > 0$  initialisiert werden.
2. Zum Test Ihrer Implementation generieren Sie eine Datei für  $F$  aus Aufgabe 10 mit drei binären Eingabevariablen und passendem Lehrersignal. Trainieren Sie ein 3-2-1 MLP über maximal 10000 Lernepochen mit der Lernrate 0.01. Wählen Sie die `tanh`-Funktion als Transferfunktion und berechnen Sie den quadratischen Fehler nach jeder Lernepoche. Plotten Sie den Verlauf des Fehlers.