

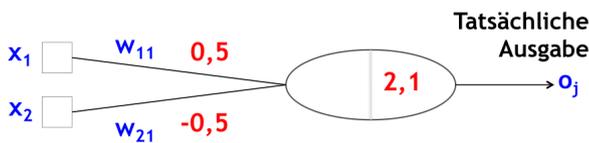
# Klassifikation mit einer boole'schen Funktion in MemBrain

In MemBrain soll die folgende zweistellige boole'sche Funktion selbständig von einem künstlichen neuronalen Netz gelernt werden:

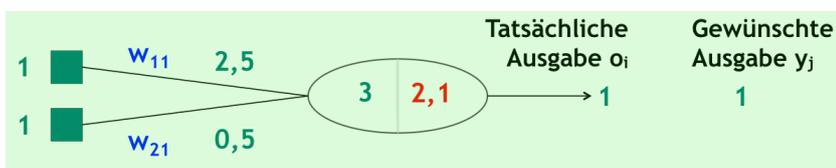
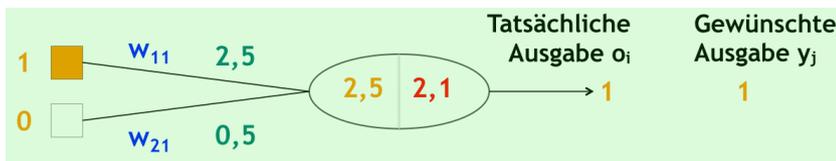
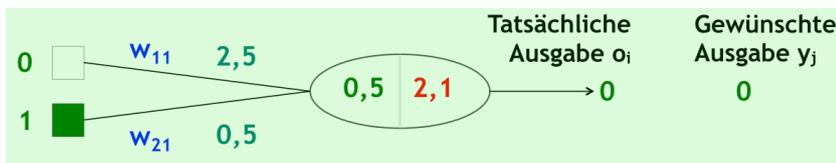
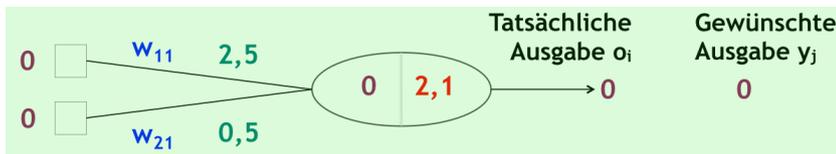
$x_1$	$x_2$	$y$
0	0	0
0	1	0
1	0	1
1	1	1

Dazu benötigen wir prinzipiell nur ein Neuron und zwei Eingänge, der Einfachheit halber sprechen wir im Folgenden trotzdem von einem neuronalen Netz (mit drei Neuronen).

Das Netz wird mit „zufällig“ von uns festgelegten Werten initialisiert. Wir wählen -0,5 und 0,5 als Synapsengewichte und 2,1 als Schwellwert. Mit diesen Startwerten kann das neuronale Netz die Funktion noch nicht korrekt wiedergaben. Wir werden es jedoch trainieren und überprüfen, ob das Netz in der Lage ist, seine Gewichte so zu ändern, dass es am Ende des Trainings die Funktion korrekt gelernt hat. Das Modell für das Netz sieht also wie folgt aus:

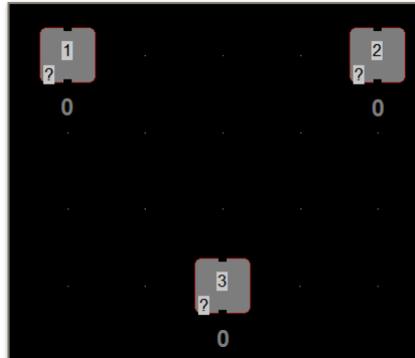


Am Ende des Trainings sollten entsprechend der Hebb'schen Lernregel die Gewichte auf folgende Werte automatisch aktualisiert worden sein (gezeigt anhand der vier möglichen Eingaben):

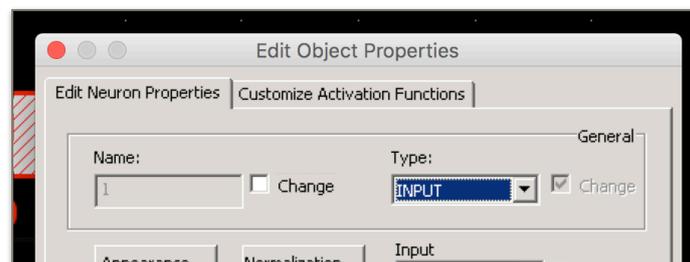


## A. Erstellen des neuronalen Netzes in MemBrain

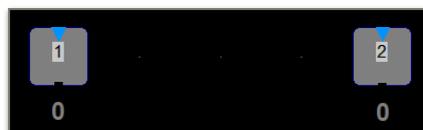
1. Zunächst erfolgt die Modellierung: Mit dem [Insert new neurons](#)-Symbol können nun per Klick Neuronen auf dem Bildschirm verteilt werden. Wir benötigen zwei Eingabeneuronen und ein Ausgabeneuron, wenn wir zwei Eingaben ( $x_1$  und  $x_2$ ) auf eine Ausgabe  $Y$  abbilden wollen.



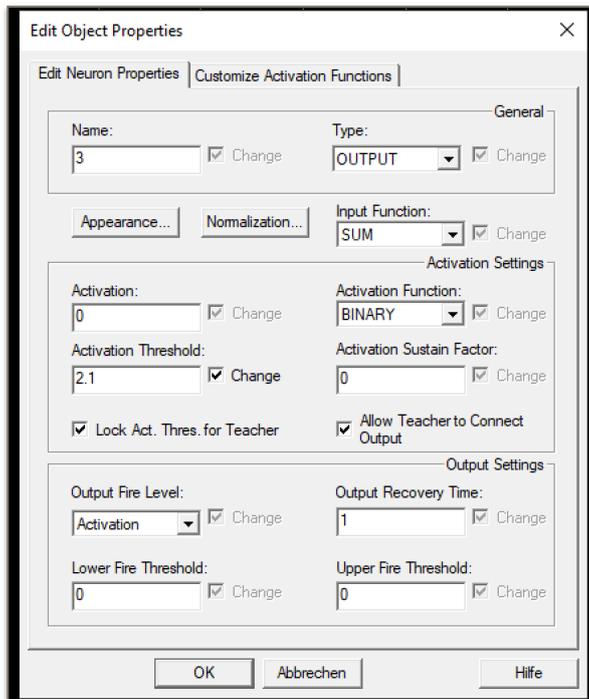
2. Da die Neuronen unterschiedliche **Aufgaben** haben (Eingabe, Ausgabe), müssen diese noch festgelegt werden. Dazu kann man in MemBrain bei gedrückter Maustaste gleich mehrere Neuronen auswählen und dann auf eines mit einem Doppelklick klicken, um die Eigenschaften für alle ausgewählten Neuronen gleichzeitig festzulegen. In der oberen Reihe muss dazu im aufspringenden Menü unter [Type](#) der Eintrag [INPUT](#) gewählt werden.



Die Eingangsschicht sollte dann wie folgt aussehen:



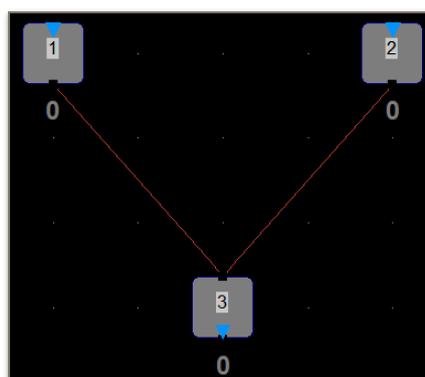
Das einzelne Ausgabeneuron wird per Doppelklick ausgewählt und es wird unter [Type](#) als [OUTPUT](#) deklariert. Unter [Activation Function](#) wird [BINARY](#) ausgewählt (also nur 0 oder 1 als Ausgabe) und der Schwellwert des Neurons wird unter [Activation Threshold](#) auf den Wert [2.1](#) gesetzt. Zusätzlich muss noch der Haken bei [Lock Act. Thres. for Teacher](#) gesetzt werden, damit der Schwellwert selbst nicht durch das Training geändert wird, sondern bei 2.1 bleibt. Die folgende Übersicht zeigt alle notwendigen Einstellungen in der Übersicht:



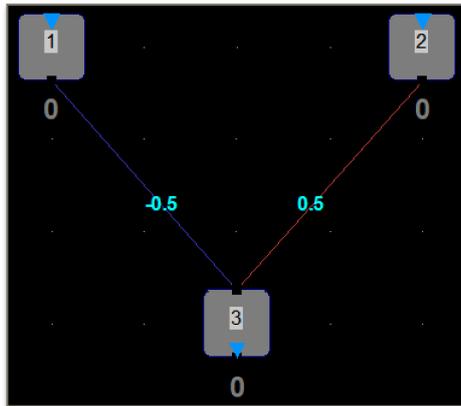
Das Ausgabeneuron sieht nach einem Klick auf **OK** nun also wie folgt aus (die kleinen blauen Pfeile zeigen übrigens den Verwendungszweck an):



3. Damit auch wirklich ein Netz entsteht, müssen die Neuronen miteinander verbunden werden. Dazu zieht man mit der Maus bei gedrückter Maustaste zwischen den schwarzen kleinen Vierecken von den Ausgängen der oberen Neuronen bis zum Eingang des unteren Neurons Verbindungen.

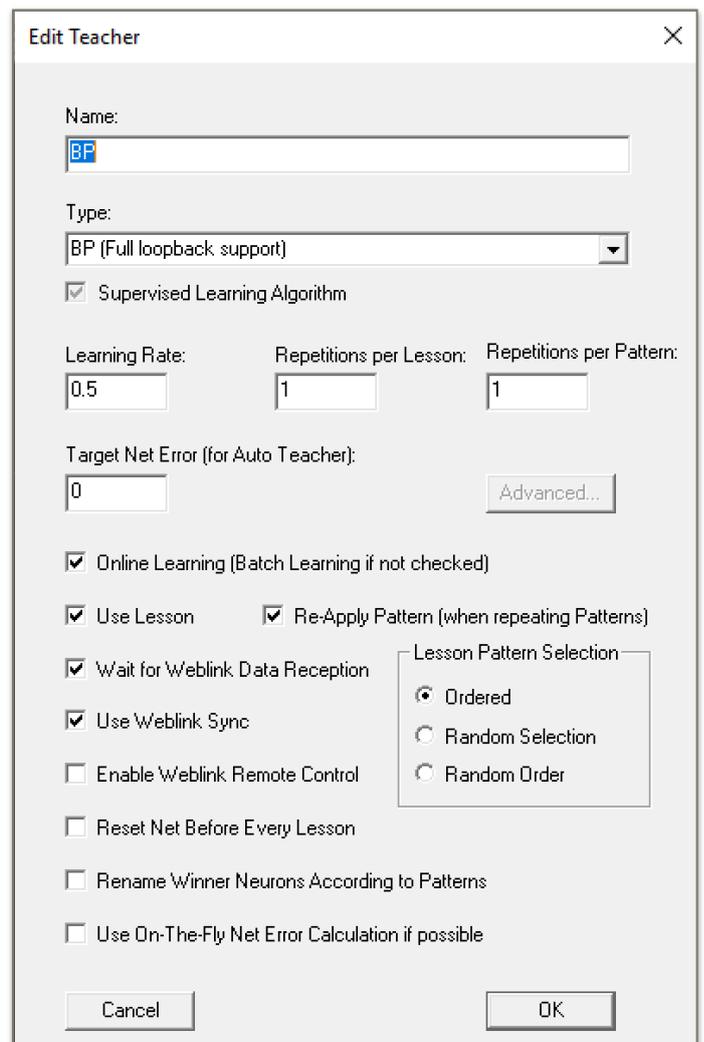
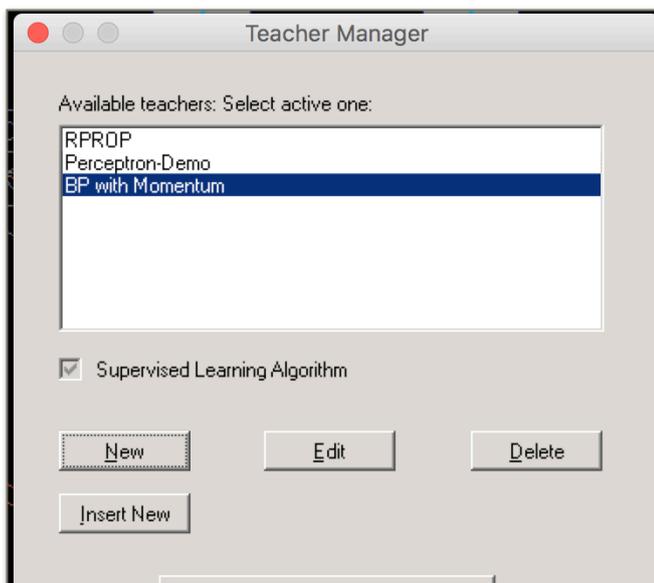


Im Anschluss klickt man doppelt auf jede Verbindung und hakt den Punkt **Display Weight** an, damit die Gewichte im MemBrain auch angezeigt werden. Beim ersten (linken) Gewicht muss der Wert des Gewichts unter **Weight** noch auf -0,5 gestellt werden, so wie im folgenden Bild dargestellt:



4. Der **Teacher Manager**, der im Menü **Teach** zu finden ist, ist Steuerzentrale für unterschiedliche Lernalgorithmen. Um die **richtige Lernregel** festzulegen, geht ihr bitte wie folgt vor:

- Im Menü **Teach** auf den Eintrag **Teacher Manager** klicken. Es öffnet sich ein Dialog wie in der linken Abbildung (mit eventuell weniger Einträgen bei einer Neuinstallation des Programms):



- Klickt nun auf **New**, um eine neue Lernregel zu erstellen und wählt dann **BP (Full loopback support)** oben aus der Liste des neuen Dialogfensters.
- Stellt danach alle anderen Parameter, wenn nötig, so ein wie auf dem rechten Bild gezeigt (**Wichtig: unter Learning Rate muss der Wert auf 0,5 geändert werden**):

- OK klicken, das neue Lernverfahren auswählen und nochmals OK drücken. Die Einstellung bleibt gespeichert

## B. Trainieren und Testen des neuronalen Netzes

1. Mit einem Klick auf das Symbol [Show Lesson Editor](#) wird ein Dialogfenster geöffnet, in das man die boole'sche Funktion eingeben kann, die gelernt werden soll.



Dazu klickt ihr drei Mal auf den Button [New Pattern](#), um insgesamt vier Muster anzulegen, durch die man mit den Pfeil-Buttons am rechten Rand durchklicken kann. Nun müsst ihr [Pattern No. 1](#) auswählen. Bei [Input data](#) sollten die Zahlen für  $x_1$  und  $x_2$  bereits 0 und 0 sein - und bei [Output Data](#) sollte ebenso eine 0 stehen, was bereits der ersten Zeile der Tabelle auf S.1 richtig entspricht.

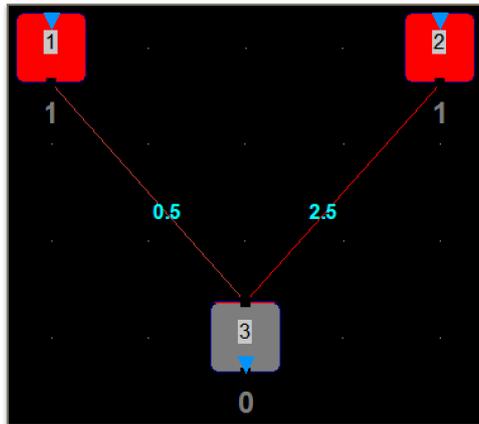
- dann wird [Pattern No. 2](#) mit den Pfeiltasten ausgewählt, bei [Input data](#) werden die Zahlen 1 und 0 (in dieser Reihenfolge) eingetragen und bei [Output Data](#) eine 0.
- dann wird [Pattern No. 3](#) mit den Pfeiltasten ausgewählt, bei [Input data](#) werden die Zahlen 0 und 1 (in dieser Reihenfolge) eingetragen und bei [Output Data](#) eine 1.
- dann wird [Pattern No. 2](#) mit den Pfeiltasten ausgewählt, bei [Input data](#) werden die Zahlen 1 und 1 eingetragen und bei [Output Data](#) eine 1.

Dementsprechend haben wir die boole'sche Funktion mit den beiden Eingaben und der Ausgabe korrekt von der Tabelle auf S.1 in den Teacher Manager übertragen. Es bietet sich an, mit den Pfeiltasten noch einmal durch die Muster zu navigieren und zu überprüfen, ob alles korrekt übernommen wurde. Ein Klick auf [Close](#) schließt den Teacher Manager schließlich.

2. Das Netz kann nun schrittweise mit einem Klick auf das Symbol [Teach Step](#) trainiert werden.



Bitte das Netz nicht randomisieren, wenn ihr danach gefragt werdet. Nach zwei Durchgängen sollten die Gewichte des Netzes denen des fertig trainierten Netzes auf S.1 entsprechen. Beachtet: das Netz ist um  $90^\circ$  im Uhrzeigersinn gedreht. Unser neuronales Netz hat entsprechend der Hebb'schen Lernregel die boole'sche Funktion richtig gelernt.



3. Zum Testen, ob das Netz korrekt funktioniert, kann man die Eingabeneuronen einzeln anklicken und wahlweise 0 oder 1 auf der Tastatur drücken. So ändert man die anliegende Eingabe des jeweiligen Neurons. Mit einem Klick auf [Think on Input](#) kann dann wiederum berechnet werden, was die Ausgabe für den jeweiligen Eingabevektor ist, so dass man manuell leicht überprüfen kann, dass alle vier Eingabemöglichkeiten zu einer richtigen Ausgabe des neuronalen Netzes führen.

