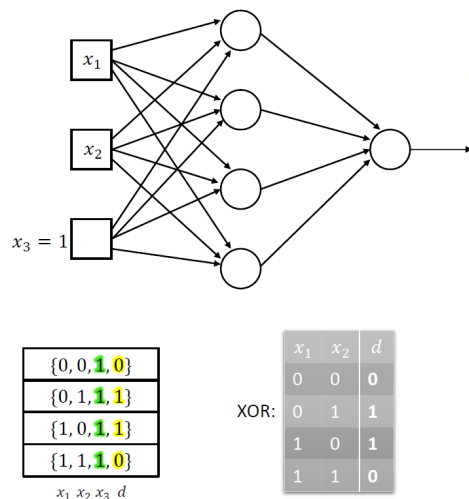


## Lab 11: Training von Neuronalen Netzen

In der Vorlesung haben Sie gesehen, dass ein einlagiges neuronales Netz (z.B. ein Perceptron) nur in der Lage ist linear-separierbare Probleme zu lösen. Mit einer entsprechenden Anzahl Neuronen sind mehrlagige neuronale Netze dagegen (theoretisch) in der Lage beliebige stetige Entscheidungsregionen zu lernen.

### Aufgabe 1

Im Folgenden sollen Sie ein neuronales Netz, mit zwei hidden Layers, die XOR-Funktion lernen lassen. Nehmen Sie hierfür die unten angegebene Wahrheitstabelle der XOR-Funktion als Trainingsdaten.



Wählen Sie eine geeignete Fehlerfunktion, sowie geeignete Aktivierungsfunktionen. Verwenden Sie für das Training z.B. 2000 Epochen bzw. Iterationen des Backpropagation-Algorithmus und einer (Mini-) Batch-Grösse von  $N = 4$ .

Als Startpunkt können Sie von `train_two_layer_network_template.py` ausgehen.

### Aufgabe 2

Installieren Sie das Deep Learning Framework Pytorch (<https://pytorch.org/>). Definieren Sie das oben eingezeichnete neuronale Netz in Pytorch und trainieren Sie dieses. Falls Sie Erfahrung mit einem anderen Deep Learning Framework haben (z.B. Keras mit Tensorflow Backend) dürfen Sie gerne dieses benutzen.