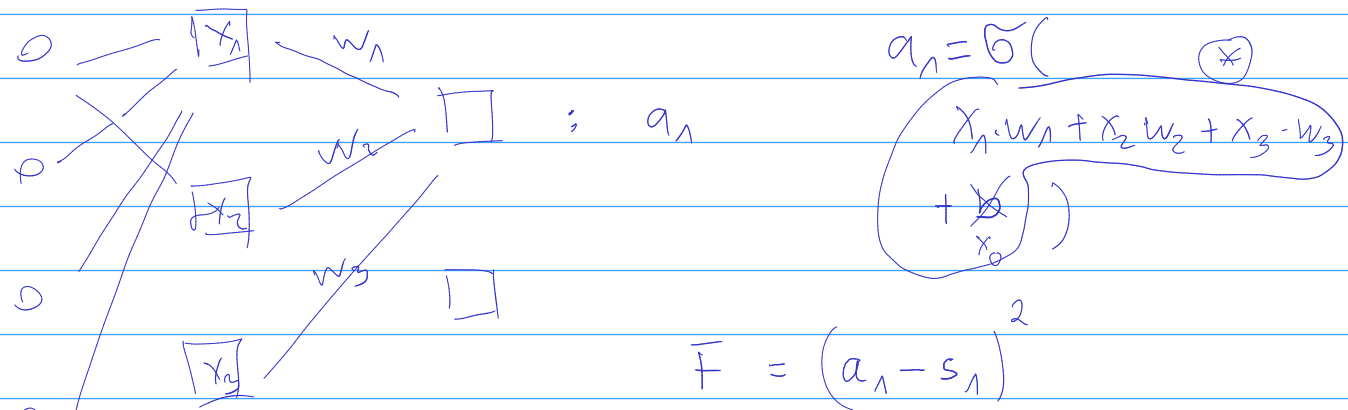


Problem: Backpropagation kompliziert

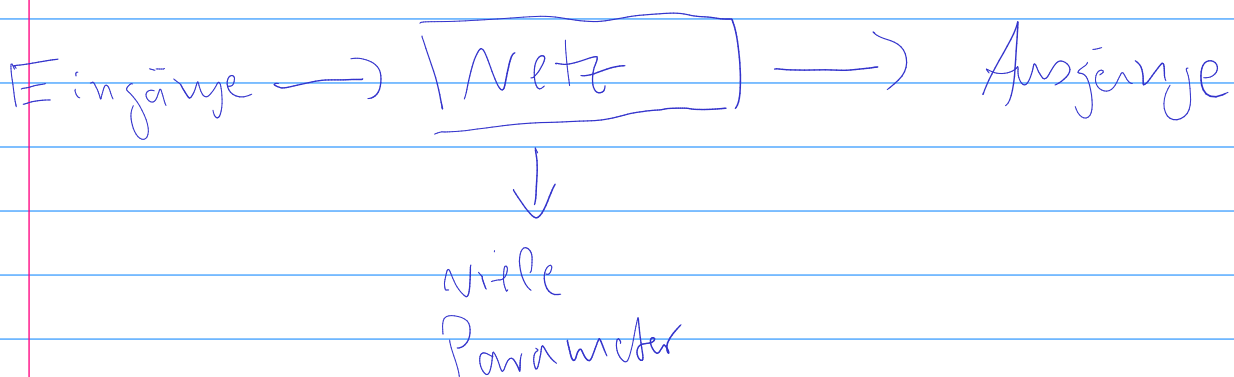


$$\frac{\partial F}{\partial w_1} = 2 \cdot (a_1 - s_1) \cdot \frac{\partial a_1}{\partial w_1}$$

$$= \sigma'(\otimes) \cdot x_1$$

Problem: Hidden Layer. Jeder Parameter im Hidden Layer beeinflusst alle (!) Ausgangsneuronen.

Vorschlag für heute: Wir arbeiten mit der Näherungsweise Berechnung der partiellen Ableitung oder ihr habt den Algorithmus irgendwo einfach übernommen.



Näherungsweise Berechnung des Gradienten:

1. Berechne für eine Eingabe die Ausgabe und den Fehler (Forward pass / Forward propagation)
2. Ändere einen (!) Parameter um einen festen Wert (etwa 0,01) und berechne wieder die Ausgabe und den Fehler, berechne ferner $\Delta y = \text{fehler aus 2} - \text{fehler aus 1}$
3. Berechne die näherungsweise partielle Ableitung für den in 2 geänderten Parameter als $\Delta y /$

4. *Speichere sinnvoll die berechnete genäherte partielle Ableitung (etwa im entspr. Neuron in einem zweiten Array mit dem gleichen Index wie den vorher geänderten Parameter).*
5. *Gehe zurück zu 2, ändere den nächsten Parameter, bis man für jeden Parameter eine (genäherte) partielle Ableitung hat.*
6. *Jetzt können wir jeden Parameter anpassen mit $w_{\text{neu}} = w_{\text{alt}} - \text{Lernfaktor} * \text{partielle Ableitung zu } w$*
7. *Gehe zurück zu 1, bis das Netz gut genug ist (oder bis man verzweifelt den Algorithmus abbricht).*

Verbesserung: Führe 1 - 5 für mehrere Eingaben durch, ändere dann in 6 den Parameter mit einem Mittelwert (!) der für die verschiedenen Eingaben berechneten partiellen Ableitungen.