

7. Übungsblatt

Aufgabe 27 Aktivierungsfunktion und Klassifikation

- a) Nehmen Sie an, in einem mehrschichtigen Perzeptron werde der *tangens hyperbolicus* als Aktivierungsfunktion verwendet. Leiten Sie den Faktor in der Fehler-Rückübertragungsformel ab, der sich aus dieser Aktivierungsfunktion ergibt! Versuchen Sie, einen Ausdruck zu finden, der ähnlich einfach ist wie der, der sich für die logistische Funktion ergibt (siehe Ableitung in der Vorlesung)!
- b) Betrachten Sie ein Drei-Klassen-Problem, d.h., einen Datensatz, in dem jeder Datenpunkt genau einer von drei Klassen zugeordnet ist. Der in der Vorlesung besprochene Iris-Datensatz mag als Beispiel dienen. Um dieses Problem mit einem mehrschichtigen Perzeptron zu lösen, d.h., um einen Klassifikator in Form eines neuronalen Netzes zu erzeugen/trainieren, muß man die Zahl der Ausgabeneuronen sowie die gewünschten Ausgabewerte für diese Neuronen wählen. Welche Anzahl von Neuronen würden Sie wählen und welche Ausgabewerte würden Sie zuordnen, wenn die Aktivierungsfunktion (1) die logistische Funktion oder (2) der *tangens hyperbolicus* ist? Ist es eine gute Idee, die gewünschten Werte aus den Sättigungswerten dieser Funktionen abzuleiten? Begründen Sie Ihre Antwort!

Aufgabe 28 Convolutional Neural Networks

In der Bildverarbeitung werden Graustufenbilder oft als zweiparametrische Funktion $f(x, y)$ dargestellt (x und y : Koordinaten eines Bildpunktes, Funktionswert: Graustufe des Bildpunktes) und als Matrix beschrieben. Es sei das folgende Bild \mathbf{A} gegeben:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 10 & 10 & 10 & 0 \\ 0 & 10 & 10 & 10 & 10 & 0 \\ 0 & 10 & 10 & 0 & 0 & 0 \\ 0 & 10 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Convolutional Neural Networks sind in der Lage, Bilddaten effizient und unabhängig von Position und Rotation in den Bildern sichtbarer Objekte zu verarbeiten. Hierfür werden sogenannte Kernfunktionen (*kernel functions*) verwendet, welche mit Ausschnitten der Bildmatrix verrechnet (mit dem Bild gefaltet) werden. Die berechneten Faltungsmerkmale (*convolved features*) dienen dann zur Erkennung von Objekten.

Ein mögliches Faltungsmerkmal ist das Vorliegen einer Kante. Nimmt man an, daß an einer Kante die Helligkeit stark wechselt, so kann man Kanten an den Extremwerten der ersten Ableitung der Bildfunktion f erkennen. Auf dieser Idee beruht der sogenannte Sobel-Operator, der aus zwei Teiloperatoren/Kernfunktionen besteht, nämlich

$$\mathbf{S}_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad \text{und} \quad \mathbf{S}_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}.$$

- a) Berechnen Sie die Faltungsmerkmale \mathbf{G}_x und \mathbf{G}_y durch ausschnittweise Multiplikation (Faltung) der Operatoren \mathbf{S}_x und \mathbf{S}_y mit der Matrix \mathbf{A} , d.h. $\mathbf{G}_x = \mathbf{S}_x * \mathbf{A}$ und $\mathbf{G}_y = \mathbf{S}_y * \mathbf{A}$! (Ein Tabellenkalkulationsprogramm könnte hilfreich sein.)
- b) Aus den erhaltenen richtungsabhängigen Matrizen \mathbf{G}_x und \mathbf{G}_y soll eine richtungsunabhängige Matrix $\mathbf{G} = (g_{ij})_{i,j=1,\dots,6}$ erstellt werden. Dazu werden die Einträge $g_{x,ij}$ und $g_{y,ij}$ der richtungsabhängigen Matrizen \mathbf{G}_x und \mathbf{G}_y jeweils quadriert, dann aufsummiert und aus dem Ergebnis die Wurzel gezogen, d.h. $g_{ij} = \sqrt{g_{x,ij}^2 + g_{y,ij}^2}$. Berechnen sie die Matrix \mathbf{G} mit Hilfe der Teilergebnisse aus Aufgabenteil a) und beschreiben Sie das Endergebnis!

Aufgabe 29 Deep Learning: n -Bit Parität

In der Vorlesung wurde gezeigt, wie die n -Bit Parität durch eine Kette aus Teilnetzen berechnet werden kann, dessen erstes die Biimplikation und deren restliche $n - 2$ das Exklusive Oder berechnen. Zeigen Sie, wie die n -Bit Parität auch durch einen binären Baum von Teilnetzen berechnet werden kann, die jeweils die Biimplikation berechnen! Wie viele Schichten hat das sich ergebende Netzwerk (als Funktion von n)? Wie viele Neuronen hat es insgesamt (als Funktion von n)?

Aufgabe 30 Deep Learning: Dropout

- a) Gegeben sei ein 12-schichtiges Perzeptron mit 10 Eingabeneuronen, 10 Neuronen je versteckter Schicht und einem Ausgabeneuron. Wie viele Gewichte müssen in diesem Netz insgesamt trainiert werden?
- b) Zum Trainieren des Netzes soll der Dropout-Ansatz verwendet werden. Wie viele Gewichte müssen je Trainingsschritt beachtet werden, wenn durch den Dropout (durchschnittlich) 5 Neuronen je Schicht deaktiviert werden?
- c) Wie hängt die (erwartete) Anzahl der zu beachtenden Gewichte von der Dropout-Rate ab, d.h., dem Bruchteil der je Schicht deaktivierten Neuronen?