

Zeit	Raum	Abgabe im Moodle; Mails mit Betreff: [SMD19]
Di. 16–18	CP-03–150	dominik.baak@udo.edu und noah.biederbeck@udo.edu
Do. 14–16	CP-03–150	kevin3.schmidt@udo.edu und maximilian.sackel@udo.edu
Fr. 16–18	CP-03–150	felix.geyer@udo.edu und rune.dominik@udo.edu

Aufgabe 19: *DeepLearning Kurzfragen*

5 P.

- (a) Was beschreibt die Lossfunktion und wofür wird sie benötigt?
- (b) Wie kann die Lossfunktion minimiert werden?
- (c) Welche Funktion haben die Aktivierungsfunktionen bzw. welches Problem wird durch diese gelöst? Nennen Sie drei gängige Aktivierungsfunktionen.
- (d) Was ist ein Neuron?
- (e) Nennen Sie drei Anwendungsbeispiele für Neuronale Netze und beschreiben Sie kurz warum sie für diese Beispiele besonders geeignet sind.

Aufgabe 20: *Lineare Klassifikation mit Softmax*

15 P.

Für eine Parameteranpassung bei der Klassifikation mittels der Softmax-Funktion muss der Gradient der Lossfunktion für alle anzupassenden Parameter bestimmt werden. Die Lossfunktion C ist wie aus der Vorlesung bekannt, gegeben durch:

$$C(f) = \frac{1}{m} \sum_{i=1}^m \hat{C}(f_i) = \frac{1}{m} \sum_{i=1}^m \left[- \sum_{k=1}^K \mathbf{1}(y_i = k) \log \frac{\exp(f_{k,i})}{\sum_j \exp(f_{j,i})} \right]. \quad (1)$$

Zur Ableitung der Lossfunktion wird die Kettenregel verwendet:

$$\nabla_W \hat{C} = \sum_{k=1}^K \frac{\partial \hat{C}}{\partial f_{k,i}} \cdot \frac{\partial f_{k,i}}{\partial W} \quad (2)$$

$$\nabla_b \hat{C} = \sum_{k=1}^K \frac{\partial \hat{C}}{\partial f_{k,i}} \cdot \frac{\partial f_{k,i}}{\partial b}. \quad (3)$$

- (a) Gegeben seien K Klassen und m Beispiele x_i jeweils mit M Komponenten. Welche Dimension haben die einzelnen Komponenten x_i , C , W , b , $\nabla_W \hat{C}$, $\nabla_{f_i} \hat{C}$, $\frac{\partial f_{k,i}}{\partial W}$, $\frac{\partial f_{k,i}}{\partial b}$? Unterscheiden Sie dabei auch zwischen Zeilen- und Spaltenvektoren.

- (b) Zeigen Sie, dass sich für die Ableitung der Lossfunktion nach den Scores für die Klasse a folgendes ergibt:

$$\nabla_{f_a} C(f) = \frac{1}{m} \sum_{i=1}^m \left[\frac{\exp(f_{a,i})}{\sum_j \exp(f_{j,i})} - \mathbf{1}(y_i = a) \right]. \quad (4)$$

- (c) Bestimmen Sie als zweiten Schritt der Kettenregel die Ableitungen von $f_{k,i}$ nach W und b mit $f_{k,i} = W_k x_i + b_k$.
- (d) Implementieren Sie die lineare Klassifikation mit Softmax für die zwei Populationen P_0 und P_1 aus der im moodle zu findenden Datei `populationen.hdf5`. Verfahren Sie dabei wie folgt:

- Lesen Sie die Populationen aus den Keys P_0 und P_1 ein.
- Führen Sie beide Populationen zusammen und erstellen Sie die entsprechenden Label (P1 hat dabei Label 1).
- Initialisieren Sie die Gewichtsmatrix und den Bias-Vektor.
- Nutzen Sie eine learning-rate von 0,5 und trainieren Sie 100 Epochen.
- Implementieren Sie die folgenden Schritte vektorisiert (Nutzen Sie `np.matmul`).
 - Implementieren Sie die Softmax Funktion und die Indikatorfunktion.
 - Iterieren Sie über die Anzahl der Epochen. Berechnen Sie in jeder Iteration die Softmax-Funktion für die aktuellen Parameter W und b . Bestimmen Sie mithilfe der erhaltenen Werte den Gradienten der Lossfunktion bzgl. W und b . Und führen Sie schließlich ein Parameterupdate durch.

Tipp: Der Gradient nach W lässt sich auch nach $\nabla_W C = \nabla_f C \cdot x_i^T$ berechnen.

- (e) Stellen Sie das Resultat (die trennende Gerade) zusammen mit den beiden Populationen in einem Scatterplot dar. Zur Herleitung der Geradengleichung nutzen Sie die Bedingung $f_1 = f_2$. Warum gilt das?