**7. Übungsblatt zur Vorlesung** **WS 2017/2018**
**Statistische Methoden der Datenanalyse** **Prof. W. Rhode**

Annika Burkowitz, Sebastian Bange, Alexander Harnisch

---

22) Andere Notation als in der Vorlesung (Transponiert)

$M$: Anzahl der Attribute

$m$: Anzahl der Beispiele bzw. Datenpunkte

$K$: Anzahl der Klassen

a) $x_i \in \mathbb{R}^{M \times 1}$ Beispiele; Trainingsdatenpunkte

$C(f): \mathbb{R}^{K \times m} \to \mathbb{R}^{1 \times 1}$

$\hat{C}(f_i): \mathbb{R}^{K \times 1} \to \mathbb{R}^{1 \times 1}$

$W \in \mathbb{R}^{K \times M}$

$b \in \mathbb{R}^{K \times 1}$

$\nabla_W \hat{C} \in \mathbb{R}^{K \times M}$

$\nabla_b \hat{C} \in \mathbb{R}^{K \times 1}$

$\nabla_{f_i} \hat{C} \in \mathbb{R}^{K \times 1}$

$\frac{\partial f_{k,i}}{\partial W} \in \mathbb{R}^{K \times M}$

$$\frac{\partial A}{\partial W} = \begin{pmatrix} \frac{\partial A}{\partial W_{11}} & \frac{\partial A}{\partial W_{12}} & \cdots \\ \frac{\partial A}{\partial W_{21}} & \ddots & \\ \vdots & & \end{pmatrix}$$

b) $$\nabla_W C(f) = \sum_{k,i} \frac{\partial C}{\partial f_{k,i}} \nabla_W f_{k,i}$$

$$\nabla_{f_{ab}} C(f) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{K} \mathbb{1}(y_i = k) \nabla_{f_{ab}} \log \frac{\exp(f_{k,i})}{\sum_{j=1}^{K} \exp(f_{j,i})}$$

$$\nabla_{f_{ai}} \log \frac{\exp(f_{k,i})}{\sum_{j=1}^{K} \exp(f_{j,i})} = \frac{\partial}{\partial(\exp(f_{k,i}))} \log \frac{\exp(f_{k,i})}{\sum_{j=1}^{K} \exp(f_{j,i})} \frac{\partial \exp(f_{k,i})}{\partial f_{ab}}$$

$$= \frac{\cancel{\sum_{j=1}^{K} \exp(f_{j,i})}}{\exp(f_{k,i})} \left( \frac{1}{\cancel{\sum_{j=1}^{K} \exp(f_{j,i})}} - \frac{\exp(f_{k,i})}{\left(\sum_{j=1}^{K} \exp(f_{j,i})\right)^2} \right) \frac{\partial \exp(f_{k,i})}{\partial f_{ab}}$$

$$= \frac{1}{\cancel{\exp(f_{k,i})}} \left( 1 - \frac{\exp(f_{k,i})}{\sum_{j=1}^{K} \exp(f_{j,i})} \right) \cancel{\exp(f_{k,i})} \delta_{ka} \delta_{ib}$$

$$\Rightarrow \nabla_{f_{ab}} C(f) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{K} \mathbb{1}(y_i = k) \left(1 - \frac{\exp(f_{k,i})}{\sum_{j=1}^{K} \exp(f_{j,i})}\right) \delta_{ka}\, \delta_{ib}$$

$$= \frac{1}{m}\left(\frac{\exp(f_{a,i})}{\sum_{j=1}^{K} \exp(f_{j,i})} - \mathbb{1}(y_b = a)\right)$$

c)

$$W_k x_i = \begin{pmatrix} w_{k1} & w_{k2} & w_{k3} \cdots \end{pmatrix} \begin{pmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \\ \vdots \end{pmatrix} = w_{k1} x_{i1} + w_{k2} x_{i2} + w_{k3} x_{i3} + \ldots$$

$$\Rightarrow \nabla_w f_{k,i} = \begin{pmatrix} \frac{\partial}{\partial w_{11}} & \frac{\partial}{\partial w_{12}} & \cdots \\ \frac{\partial}{\partial w_{21}} & \frac{\partial}{\partial w_{22}} & \cdots \\ \vdots & \ddots & \end{pmatrix} W_k x_i = \begin{pmatrix} \vec{0} \\ \vdots \\ \vec{0} \\ \vec{x}_i^T \\ \vec{0} \\ \vdots \\ \vec{0} \end{pmatrix} \leftarrow \text{k'te Zeile}$$

$$\Rightarrow \left(\nabla_w f_{k,i}\right)_a = \delta_{ka}\, x_i^T$$

$$\nabla_b f_{k,i} = \frac{\partial b_k}{\partial b} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \leftarrow \text{k'te Zeile}$$

$$\Rightarrow \left(\nabla_b f_{k,i}\right)_a = \delta_{ka}$$

e) $f_1 > f_2$ : Datenpunkt gehört zur ersten Klasse

$f_1 < f_2$ : Datenpunkt gehört zur zweiten Klasse

$f_1 = f_2$ : grenz~~fall~~ $\Rightarrow$ Trenngerade

Herleitung:

$f_1 = f_2$

$\Leftrightarrow W_1 \vec{x} + b_1 = W_2 \vec{x} + b_2$

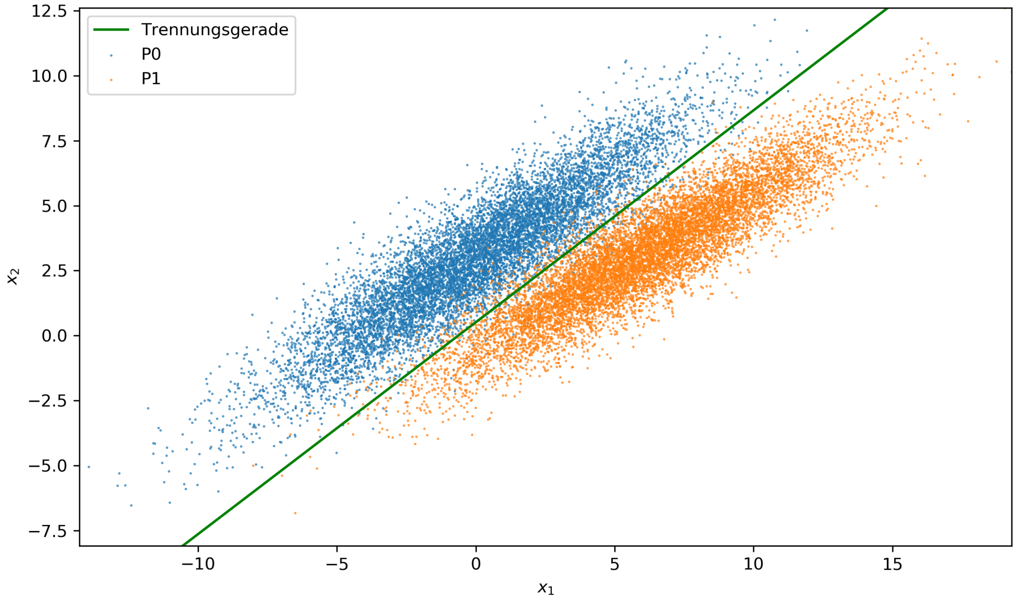
$\Leftrightarrow (W_1 - W_2)\vec{x} + b_1 - b_2 = 0$

$\Leftrightarrow \begin{pmatrix} w_{11} - w_{21} & w_{12} - w_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + b_1 - b_2 = 0$

$\Leftrightarrow x(w_{11} - w_{21}) + y(w_{12} - w_{22}) + b_1 - b_2 = 0$

$\Leftrightarrow y = \dfrac{b_2 - b_1 - x(w_{11} - w_{21})}{w_{12} - w_{22}}$

Achtung im Code: Wie in der Vorlesung genau transponiert definiert:
$w_{12} \Leftrightarrow w_{21}$ ; $b_1 \Leftrightarrow b_2$



Trennungsgerade
P0
P1
12.5
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
$x_2$
−10
−5
0
5
10
15
$x_1$

23) $y = a_0 + a_1 x$

$a_0 = 1{,}0 \pm 0{,}2$ $\qquad \rho = -0{,}8 = \dfrac{cov(a_0, a_1)}{\sigma_{a_0}\,\sigma_{a_1}}$

$a_1 = 1{,}0 \pm 0{,}2$

$\sigma_{a_0} = \sigma_{a_1} = 0{,}2$

allg.: $\sigma_y = \sqrt{\sum_{i=1}^{m} \left(\frac{\partial y}{\partial x_i}\sigma_{x_i}\right)^2 + 2\sum_{i=1}^{m-1}\sum_{k=i+1}^{m}\left(\frac{\partial y}{\partial x_i}\right)\left(\frac{\partial y}{\partial x_k}\right) cov(x_i, x_k)}$

a) $\sigma_y = \sqrt{\left(\frac{\partial y}{\partial a_0}\sigma_{a_0}\right)^2 + \left(\frac{\partial y}{\partial a_1}\sigma_{a_1}\right)^2 + 2\left(\frac{\partial y}{\partial a_0}\right)\left(\frac{\partial y}{\partial a_1}\right) cov(a_0, a_1)}$

$= \sqrt{0{,}2^2 + 0{,}2^2 x^2 + 2x \cdot 0{,}2^2 \cdot (-0{,}8)}$

$= 0{,}2\sqrt{1 + x^2 - 1{,}6x}$

mit $\rho = 0$:

$\sigma_y = 0{,}2\sqrt{1+x^2}$

c) analytisch

$y(-3) = -2{,}0 \pm 0{,}8$

$y(0) = 1{,}0 \pm 0{,}2$

$y(3) = 4{,}0 \pm 0{,}5$

numerisch

$y(-3) = -2{,}0 \pm 0{,}8$

$y(0) = 1{,}0 \pm 0{,}2$

$y(3) = 4{,}0 \pm 0{,}5$

Funktioniert gut!
Klappt halt besser, wenn man mehr Werte für $a_0$ und $a_1$ zieht.

../A23/A23_scatter.png

**Abbildung 1:** Korrelierte Parameter $a_0$ und $a_1$.

../A23/A23_resultat.pdf

**Abbildung 2:** Vergleich des numerisch bestimmten Fehlers mit dem analytisch bestimmten Fehlers.

# Aufgabe 24

## a)

Die Lossfunktion gibt an, wie schlecht die berechnete Vorhersage im Vergleich zu dem korrekten Ergebnis ist. Um die bestmögliche Vorhersage zu treffen, muss die Lossfunktion minimiert werden.

## b)

Wenn die Lossfunktion abgeleitet werden kann, kann sie minimiert werden, in dem die Ableitung der Lossfunktion gleich Null gesetzt wird.

## c)

Aktivierungsfunktionen stellen einen Zusammenhang zwischen dem Input an einem Knoten in einem Neuronalen Netz und dem Output des Knotens her. Beispielsweise realisieren sie einen Schwellwert für den Input, ab dem eine Aktivierung des Neurons stattfindet und somit ein Output generiert wird. Aktivierungsfunktionen sind nicht-linear und ermöglichen damit durch nicht-lineare Kombination des (gewichteten) Inputs die Erzeugung nicht-linearer Entscheidungsgrenzen.
Beispiele

1. Sigmoid:
$$f(x) = \frac{1}{1 + e^{-x}}$$
2. Tangens hyperbolicus:
$$f(x) = \tanh(x)$$
3. Rectified Linear Unit (ReLU) / Softplus:
$$\begin{aligned} f(x)_{\text{ReLU}} &= \max(0,x) \\ f(x)_{\text{Softplus}} &= \ln\left(1 + e^{x}\right) \end{aligned}$$
4. Softmax:
$$q_k(x) = \frac{e^{f_k(x)}}{\sum_j e^{f_j(x)}}$$

## d)

Künstliche Neuronen sind Bestandteile eines Neuronalen Netzes. Jedes Neuron erhält Input, der zunächst (mit individuellen Gewichtungen für jedes Neuron) gewichtet wird und dann durch die Übertragungsfunktion aufsummiert wird. Das Ergebnis ist die sogenannte Netzeingabe. Zusätzlich kann für jedes Neuron ein Schwellwert festgelegt werden. Die Netzeingabe muss dann diesen Schwellwert überschreiten, damit die Aktivierungsfunktion die Eingabe modulieren kann und somit die Ausgabe festlegen kann.

**e)**

Allgemein sind neuronale Netze dann besonders gut geeignet, wenn man selber die Lösung nicht vernünftig mathematisch beschreiben kann. Anwendungsbeispiele für Neuronale Netze:

1. Bilderkennung: Wie definiert man einen Stuhl? Lösung: Das neuronale Netz soll sich das selber benennen.
2. Spiele: Sehr viele mögliche Parameter, Problem wird zu hochdimensional.
3. Sinnerkennung von Sprache (Siri, Cortana, Alexa,..): Ähnlich wie Bilderkennung zur Erkennung der Worte und viele mögliche Parameter, weil jerameter, weil jeder leicht anders formuliert.