

Erratum Stand 30. Januar 2020

Für den ersten Druck des Buches vom August 2018

- **Abschnitt 4.2.2:** Kein Fehler im eigentlichen Sinne, aber hier ich wollte ich eigentlich die grobe Unterscheidung bzgl. Klassifikation und Regression aus Abschnitt 2.4.1 noch durch die Skalenniveaus der Zielmengen ergänzen. Klassifikation als Ansatz für primär Nominal und Ordinal-Skalen und Regression für die die Rationalskala.
- **Seite 36** ... bei einer Zuweisung `a = b` ~~kopiert werden~~verhalten, ist ... Die Variablennamen sind ~~hierin~~ Python lediglich ... und ~~die~~eine Zuweisung
- **Seite 47** Im vorletzten Listingblock steht -1 richtig wäre natürlich 10.
- **Seite 57:** ... Parameter `lower` `right` in die untere ~~linke~~ rechte Ecke.
- **Seite 65** erster Absatz Zeile 2: „Zeilen 18-~~21~~20“
- **Seite 86:** Das Verfahren heißt zwar Kreuzvalidierung, aber um es im Buch verwendeten Definition besser zu beschreiben wäre es besser die folgenden Sätze wie folgt zu ändern: Dann muss die Abfolge aus Training und ~~Validierung~~Test k -fach durchgeführt werden. Hierbei wird jeweils eine Teilmenge zum ~~Validieren~~Testen verwendet, und mit den restlichen $k - 1$ Teilmengen wird trainiert.
- **Seite 104** zweiter Absatz Zeile 9: „... vollen Rank~~g~~ ...“
- **Seite 102:** In der letzten Zeile der Tabelle muss es heißen:
~~%~~ der Bevölkerung mit niedrigem Status
- **Seite 111 und 115:** Im Quellcode wird `np.random.choice` ohne `replace=False` aufgerufen. Dadurch werden die Trainings- und Testmenge ggf. nicht sauber getrennt.
- **Seite 111** Im Code-Zeile 25 steht `YTrain[...]`, es müsste aber `yTrain[...]` heißen. Skript läuft zwar so, ist aber nicht so gedacht.
- **Seite 112:** ... Hierbei haben wir einen einfachen Mehrheitsentscheid in Zeile ~~55~~26 verwendet, ...
- **Seite 114 und 115:** Bei der Gewichtung des k -nn sind in den Formel zwei Fehler. Einmal wurde nicht mit dem Kehrwert gearbeitet und zum anderen steht dort ein x wo ein y stehen müsste. Richtig lautet der Abschnitt auf Seite 114:

$$y_p(x) = \sum_{i=1}^k \hat{\omega}_i y_i \quad \text{mit} \quad \hat{\omega}_i = \frac{d_i^{-1}}{\hat{d}} \quad \text{und} \quad \hat{d} = \sum_{i=1}^k d_i^{-1}$$

Ein Problem, mit dem wir dabei umgehen müssen, ist u. a. die Ausnahme, bei der einer der k Nachbarn den Abstand Null von unserem Ansatzpunkt hat.

bzw. auf Seite 115

Die Regression über die k Nachbarn x_i ($i = 1 \dots k$) ergibt sich dann durch die folgende Formel mit den angepassten Gewichten ω_i :

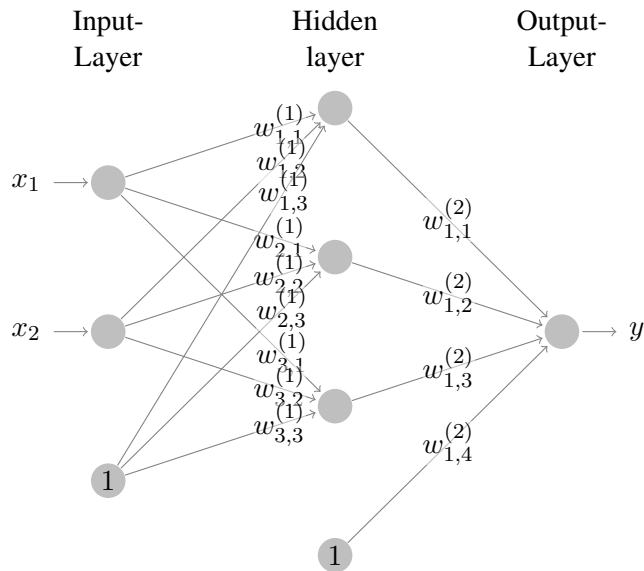
$$y_p(x) = \sum_{i=1}^k \omega_i y_i \quad \text{mit} \quad \omega_i = \frac{\left(d_i + \frac{\text{smear}}{k}\right)^{-1}}{d} \quad \text{und} \quad d = \sum_{i=1}^k \left(d_i + \frac{\text{smear}}{k}\right)^{-1}$$

Leider ist die falsche Formel ohne Umkehrwert auch im Quellcode umgesetzt. Bitte laden Sie den korrigierten Quellcode von der Webseite herunter. Die Werte und Plots im Buch ändern sich jedoch nicht wesentlich, da für `mysmear = 0.5` dieser Wert den Output dominiert.

- **Seite 123 ff:** Die Tabelle 6.1 wird durch einen fehlerhaften Verweis oft als 6.2 bezeichnet. Tatsächlich ist 6.2 nur auf Seite 145 richtig.
- **Seite 124:** Damit erhalten wir für unser Beispiel eine gewichtete Pfadlängensumme 21, eine Pfadlängensumme von 12,...
...die Baumhöhe gleich 4, die Pfadlängensumme 14 und die gewichtete Pfadlängensumme 22.
- **Seite 128:** Die Formel in der Beschriftung von Abb. 6.7 ist falsch. Richtig ist $H(V) = -p \cdot \log_2(p) - (1-p) \cdot \log_2(1-p)$
- **Seite 132** `tree` `binaryTree` sollten Sie in Zeile 45 natürlich durch ...
- **Seite 133** Der so erzeugte Baum inklusive **Zeilennummern**Knoten-Nummern ist in ...
- **Seite 134** Die Anzahl der Blätter ist 6, die Baumhöhe 4, die Pfadlänge 17 und ...
- **Seite 142** Im Quellcode Zeile 105 und 107, wird nicht auf die Merkmale eingeschränkt. Es muss `dataset[Trainingset, 0:4]` bzw. `dataset[Testset, 0:4]` heißen.
- **Seite 142** Eigentlich kein Fehler, aber eine eher sinnlose Bezeichnung für eine Größe. Es wäre besser in der Formel auf Seite 142 \bar{y} zu verwenden. Also:

$$RSS = \sum_{i=1}^n (\bar{y} - f(x_i))^2$$
$$\bar{y} = \frac{1}{n} \sum_{i=1}^n f(x_i)$$

- **Seite 172:** Die Abbildung 7.11 enthält einen Notationsfehler, was die Angaben der Gewichte angeht. Richtig sind die folgenden Angaben:



- **Seite 175** Didaktisch ungünstig Deep-Network und *Universal approximation theorem* kombiniert. Besser: Man kann alle stetigen Funktionen schon mit einem Hidden-Layern approximieren, wenn man in dem Hidden-Layer...
- **Seite 177ff** Formaler Lapsus: Es wird an verschiedenen Stellen $D = \{X, Y\}$ angegeben. In den Fehlerfunktionen wird dann aber immer angegeben:

$$J(W) = \sum_{y_D \in D} \dots$$

Formal korrekt ist natürlich:

$$J(W) = \sum_{y_D \in Y} \dots$$

- **Seite 184:** Abbildung 7.17 . In der Abbildung steht *huge*. Das hat dort keinen Sinn, sondern hätte durch den \LaTeX -Befehl `\huge` die Schrift größer machen sollen.
- **Seite 211:** Der Quellcode unten passt nicht zum Plot, da ein Vorzeichenfehler drin ist. Es muss richtig heißen:
`y = np.maximum(1 - np.maximum(-x, 0), 0)`
- **Seite 226:** Wir skalieren formatieren also die 28×28 -Matrix in einen Vektor mit 784 Einträgen um.
- **Seite 235:** ... und daher als quasi selbst die Feature lernt... und daher als quasi selbst die Feature lernt.
- **Seite 237** [Wort fehlt] ... Wenn wir den Classification Error betrachten ...

- **Seite 242:** Kein Fehler, aber ggf. didaktisch ungünstig. Die Architektur des CNN auf der Seite ist ungewöhnlich, weil ich am Schluss kein abschließendes Pooling eingebaut habe. Wenn Sie die typische Folge von Faltung+Pooling nutzen wollen, fügen Sie zwischen Zeile 36 und 37 im Listing noch

```
CNN.add(layers.MaxPool2D(pool_size=(2, 2), padding='same'))
```

hinzu. Die Freiheitsgrade im Netz nehmen dadurch ab und die Genauigkeit bei gleicher Anzahl von Trainingszyklen wird etwas schlechter.

- **Seite 243:** Copy & Paste-Fehler, in der zweiten Formel wurde vergessen den letzten Faktor zu löschen. Nach der ersten Schicht gibt es natürlich keine Farbkanäle mehr. Die Formel lautet also:

$$9248 = 32 \text{ Bias-Neuronen} + 32 \cdot 32 \text{ Feature Maps} \cdot (3 \cdot 3) \text{ Kernelgewichte}$$

- **Seite 263:** in Formel (9.4) ist das $1/n$ vor der Summe im Zähler zu viel.
- **Seite 263:** Copy & Paste-Fehler:

$$R = \begin{pmatrix} \text{Kor}(x, x) & \text{Kor}(x, y) \\ \text{Kor}(y, x) & \text{Kor}(y, y) \end{pmatrix} .$$

- **Seite 272:** ... und verwenden anschließend die schon bekannten ~~Schwertlinien~~ PKW-Daten.
- **Seite 282f:** Die Zahl der Merkmale im Fließtext stimmt nicht mit denen im Code bzw. der Auswertung überein. Es wurde überall mit 24 Merkmalen gearbeitet, nicht mit 32. ... ursprünglich 784 auf z. B. ~~32~~24 Merkmale zu ...
... Als Mensch kann man die auf ~~32~~24 Merkmale reduzierte ...
⋮

- **Seite 297** ... zwei wählbaren Parametern $\kappa\gamma$ und $\delta\tau$:
- **Seite 308** C statt S in der Formel:

$$J = \sum_{i=1}^k \sum_{x_j \in C_i} d(x_j, \mu_i)$$

- **Seite 309** Einmal steht dort d für die Dimension, es müsste aber r sein.
- **Seite 319** im Pseudocode steht `eps` im Aufruf, es müsste aber ε sein, damit es mit dem Code unten konsistent ist.
- **Seite 319** im Pseudocode steht N' in Zeile 20. Das muss durch `NExpand` ersetzt werden.
- **Seite 320** Die Einrückung im Pseudocode in Zeile 23 bis 25 ist eine Ebene zu tief.
- **Seite 343:** In der abgesetzten Formel zwischen (12.5) und (12.6) fehlt ein Argument. Hier steht $\max_{a'} Q(\delta(s, a'))$, richtig ist $\max_{a'} Q(\delta(s, a'), a')$

- **Seite 348:** Abb. 12.6 ~~Noden~~ \rightsquigarrow Norden
- **Seite 353:** ... mit einer ein klein wenig höheren Belohnung ...
(besser) \rightsquigarrow
... mit einer ein klein wenig höheren kumulativen Belohnung ...
- **Seite 354:** Auf Seite 354 fehlt der Hinweis, dass $x(a)$ in der Formel (12.9) in diesem Anwendungsfall dem Q-Wert von a in der gegebenen Situation s entspricht.
- **Seite 382 und 392** Die Zeile 143 muss lauten:

```
def CoopData(self, X, Y, nX, nY) :
```

Die gleiche Definition muss auch auf der Seite 392 korrigiert werden.