Algorithmik zur Optimierung in neuronalen Netzwerken

Gradient Descent und Backpropagation

Tim Hilt

Date: tbd

Hochschule Esslingen — University of Applied Sciences

Gliederung

Training

Loss-Funktion/ Cost-Funktion

- Dient zur Berechnung des Fehlers während dem Training
- Trainingsfehler soll minimiert werden
- $\Rightarrow$ wir suchen den Punkt, an dem die Ableitung der Loss-Funktion 0 wird, der Fehler also nicht mehr abnimmt
- Es gibt eine Vielzahl an Loss- oder Cost-Funktionen, wir betrachten hier die „**Quadratic Cost/ Mean Squared Error (MSE)**“:

$$C(w, b) = \frac{1}{2n} \sum_x ||y(x) - a||^2$$

Loss-Funktion/ Cost-Funktion

- Dient zur Berechnung des Fehlers während dem Training
- Trainingsfehler soll minimiert werden
- $\Rightarrow$ wir suchen den Punkt, an dem die Ableitung der Loss-Funktion 0 wird, der Fehler also nicht mehr abnimmt
- Es gibt eine Vielzahl an Loss- oder Cost-Funktionen, wir betrachten hier die „**Quadratic Cost/ Mean Squared Error (MSE)**“:

$$C(w, b) = \frac{1}{2n} \sum_x ||y(x) - a||^2$$

$C(w, b)$	Cost in Abhängigkeit von w und b
n	Anzahl der Trainingsinstanzen
$y(x)$	Label wenn x Input ist
a	Output des Netzwerkes, bei w und b

Gradient Descent

Backpropagation

- Idee: Divide and conquer; Problem in kleinere, handhabbare Probleme zerlegen

$$f(a, b, c, d) = ((a + b) \cdot (c + d))^2$$

Backpropagation

- Idee: Divide and conquer; Problem in kleinere, handhabbare Probleme zerlegen

$$
\begin{aligned}
f(a, b, c, d) &= ((a + b) \cdot (c + d))^2 \\
g(a, b) &= a + b
\end{aligned}
$$

Backpropagation

- Idee: Divide and conquer; Problem in kleinere, handhabbare Probleme zerlegen

$$
\begin{aligned}
f(a, b, c, d) &= ((a + b) \cdot (c + d))^2 \\
g(a, b) &= a + b \\
h(c, d) &= c + d
\end{aligned}
$$

Backpropagation

- Idee: Divide and conquer; Problem in kleinere, handhabbare Probleme zerlegen

$$
\begin{aligned}
f(a,b,c,d) &= ((a+b)\cdot(c+d))^2 \\
g(a,b) &= a+b \\
h(c,d) &= c+d \\
i(g,h) &= g\cdot h
\end{aligned}
$$

Backpropagation

- Idee: Divide and conquer; Problem in kleinere, handhabbare Probleme zerlegen

$$
\begin{aligned}
f(a,b,c,d) &= ((a+b)\cdot(c+d))^2 \\
g(a,b) &= a+b \\
h(c,d) &= c+d \\
i(g,h) &= g\cdot h \\
f(i) &= i^2
\end{aligned}
$$

Backpropagation

$$
\begin{aligned}
f(a, b, c, d) &= ((a + b) \cdot (c + d))^2 \\
g(a, b) &= a + b \\
h(c, d) &= c + d \\
i(g, h) &= g \cdot h \\
f(i) &= i^2
\end{aligned}
$$

Zuvor beschriebene Architektur

Pass

Optimierte Architektur

- Vorteil: Schnellere Konvergenz
- Verwendung von optimierter Cost-, Activation- und Gradient-Descent-Funktion

Fragen?

Literaturverzeichnis

GÉRON, Aurélien. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, 2019.

BURKOV, Andriy. *The hundred-page machine learning book*. Andriy Burkov Quebec City, Can., 2019.

LECUN, Yann; BOTTOU, Léon; BENGIO, Yoshua; HAFFNER, Patrick. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*. 1998, Jg. 86, Nr. 11, S. 2278–2324.

XIAO, Han; RASUL, Kashif; VOLLGRAF, Roland. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*. 2017.