Berechnungen der Biases nach 2 Iterationen

Hermann Völlinger

13.9.2024

Vorwärts- und Rückwärtsdurchlauf (Iteration 1)

Vorwärtsdurchlauf

Hidden Layer Aktivierungen:

$$\begin{aligned} z_{\text{hidden},1} &= W_1[0] \cdot x + b_1[0] \\ &= (0.15 \times 0.05) + (0.20 \times 0.10) + 0.35 \\ &= 0.3775 \\ h_1 &= \sigma(0.3775) \approx 0.59327 \end{aligned}$$

$$\begin{aligned} z_{\text{hidden},2} &= W_1[1] \cdot x + b_1[1] \\ &= (0.25 \times 0.05) + (0.30 \times 0.10) + 0.35 \\ &= 0.3925 \\ h_2 &= \sigma(0.3925) \approx 0.59688 \end{aligned}$$

Output Layer Aktivierungen:

$$\begin{aligned} z_{\text{output},1} &= W_2[0] \cdot h + b_2[0] \\ &= (0.40 \times 0.59327) + (0.45 \times 0.59688) + 0.60 \\ &\approx 1.1059 \\ y_1 &= \sigma(1.1059) \approx 0.7514 \end{aligned}$$

$$\begin{aligned} z_{\text{output},2} &= W_2[1] \cdot h + b_2[1] \\ &= (0.50 \times 0.59327) + (0.55 \times 0.59688) + 0.60 \\ &\approx 1.2249 \\ y_2 &= \sigma(1.2249) \approx 0.7729 \end{aligned}$$

Rückwärtsdurchlauf

Fehler im Output Layer:

$$\begin{aligned}\text{error}_{\text{output},1} &= y_1 - y_{\text{target},1} \\ &= 0.7514 - 0.01 \\ &= 0.7414\end{aligned}$$

$$\begin{aligned}\text{error}_{\text{output},2} &= y_2 - y_{\text{target},2} \\ &= 0.7729 - 0.99 \\ &= -0.2171\end{aligned}$$

Delta für den Output Layer:

$$\begin{aligned}\delta_{\text{output},1} &= \text{error}_{\text{output},1} \times \sigma'(z_{\text{output},1}) \\ &= 0.7414 \times \sigma(1.1059) \times (1 - \sigma(1.1059)) \\ &= 0.7414 \times 0.7514 \times (1 - 0.7514) \\ &\approx 0.1385\end{aligned}$$

$$\begin{aligned}\delta_{\text{output},2} &= \text{error}_{\text{output},2} \times \sigma'(z_{\text{output},2}) \\ &= -0.2171 \times \sigma(1.2249) \times (1 - \sigma(1.2249)) \\ &\approx -0.0381\end{aligned}$$

Fehler im Hidden Layer:

$$\begin{aligned}\text{error}_{\text{hidden},1} &= W_2[0,0] \times \delta_{\text{output},1} + W_2[1,0] \times \delta_{\text{output},2} \\ &= (0.40 \times 0.1385) + (0.50 \times -0.0381) \\ &\approx 0.0452\end{aligned}$$

$$\begin{aligned}\text{error}_{\text{hidden},2} &= W_2[0,1] \times \delta_{\text{output},1} + W_2[1,1] \times \delta_{\text{output},2} \\ &= (0.45 \times 0.1385) + (0.55 \times -0.0381) \\ &\approx 0.0500\end{aligned}$$

Delta für den Hidden Layer: Begründung: Wie berechnet sich der Gradient des Biases?

Der Gradient des Biases wird während des Backpropagation-Prozesses berechnet. Um zu verstehen, wie sich der Gradient des Biases ergibt, müssen wir die Rolle des Biases im neuronalen Netzwerk und die Kettenregel der Ableitung betrachten. Hier ist die Begründung in mehreren Schritten:

1. Vorwärtsdurchlauf und Rolle des Biases In einem neuronalen Netzwerk ist der Bias ein zusätzlicher Parameter, der für jedes Neuron addiert wird, um die Ausgabe des Neurons zu verschieben und Flexibilität bei der Modellierung zu bieten. Für eine Neuronenausgabe in der versteckten oder Ausgabeschicht haben wir:

$$z = \sum_i w_i \cdot x_i + b$$

wobei z der Nettoeingang für das Neuron ist, w_i die Gewichtungen, x_i die Eingaben und b der Bias ist. Diese Aktivierung wird dann durch eine Aktivierungsfunktion (wie die Sigmoid-Funktion) transformiert:

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

2. Fehlerfunktion Der Fehler (oder Verlust) für das Netzwerk wird häufig durch die *quadratische Fehlerfunktion* oder die *Cross-Entropy* berechnet. Bei der quadratischen Fehlerfunktion lautet die Fehlerfunktion für eine Ausgabe y und Zielausgabe y_{target}:

$$E = \frac{1}{2}(y_{\text{target}} - y)^2$$

3. Backpropagation: Berechnung des Gradienten Während der Backpropagation geht es darum, den Gradienten des Fehlers E in Bezug auf die Gewichte und Biases zu berechnen. Dieser Gradient gibt an, wie stark der Fehler E in Bezug auf eine kleine Änderung des Biases b ansteigt. Die Kettenregel wird verwendet, um dies zu berechnen.

Der Gradient des Fehlers in Bezug auf den Bias b ist:

$$\frac{\partial E}{\partial b}$$

Um diesen Gradient zu berechnen, wenden wir die Kettenregel an. Der Fehler E hängt vom Bias b über z und y ab. Also:

$$\frac{\partial E}{\partial b} = \frac{\partial E}{\partial y} \cdot \frac{\partial y}{\partial z} \cdot \frac{\partial z}{\partial b}$$

4. Ableitungen Schritt für Schritt

- $\frac{\partial E}{\partial y}$: Dies ist die Ableitung des Fehlers in Bezug auf die Ausgabe y. Bei einer quadratischen Fehlerfunktion lautet sie:

$$\frac{\partial E}{\partial y} = -(y_{\text{target}} - y)$$

- $\frac{\partial y}{\partial z}$: Dies ist die Ableitung der Sigmoid-Aktivierungsfunktion in Bezug auf z:

$$\frac{\partial y}{\partial z} = \sigma(z)(1 - \sigma(z)) = y(1 - y)$$

- $\frac{\partial z}{\partial b}$: Da $z = \sum w_i \cdot x_i + b$, ist die Ableitung von z in Bezug auf den Bias b einfach:

$$\frac{\partial z}{\partial b} = 1$$

5. Endergebnis Der Gradient des Biases ergibt sich also zu:

$$\frac{\partial E}{\partial b} = \frac{\partial E}{\partial y} \cdot \frac{\partial y}{\partial z} \cdot \frac{\partial z}{\partial b}$$

Das bedeutet:

$$\frac{\partial E}{\partial b} = -(y_{\text{target}} - y) \cdot y(1 - y)$$

Da $\frac{\partial z}{\partial b} = 1$, fällt dieser Term weg, und es bleibt der Einfluss der Aktivierung und des Fehlers übrig.

6. Gradienten und Bias-Update Im Backpropagation-Algorithmus wird der Bias mit einem Lernschritt basierend auf dem berechneten Gradienten aktualisiert:

$$b_{\text{neu}} = b_{\text{alt}} - \eta \cdot \frac{\partial E}{\partial b}$$

wobei η die Lernrate ist.

Zusammenfassung

- Der Gradient des Biases zeigt an, wie der Fehler in Bezug auf den Bias ansteigt.
- Er wird durch die Ableitung der Fehlerfunktion und der Aktivierungsfunktion berechnet.
- Der Bias wird in jeder Iteration durch den Gradienten aktualisiert, um den Fehler zu minimieren.

Durch diesen Prozess lernt das neuronale Netzwerk, seine Biases so anzupassen, dass der Fehler minimiert wird und die Zielausgaben besser erreicht werden.

$$\begin{aligned}
\delta_{\text{hidden},1} &= \text{error}_{\text{hidden},1} \times \sigma'(z_{\text{hidden},1}) \\
&= 0.0452 \times \sigma(0.3775) \times (1 - \sigma(0.3775)) \\
&\approx 0.0109
\end{aligned}$$

$$
\begin{aligned}
\delta_{\text{hidden},2} &= \text{error}_{\text{hidden},2} \times \sigma'(z_{\text{hidden},2}) \\
&= 0.0500 \times \sigma(0.3925) \times (1 - \sigma(0.3925)) \\
&\approx 0.0120
\end{aligned}
$$

Aktualisierung der Biases (Iteration 1)

Neue Biases für den Hidden Layer b_1:

$$
\begin{aligned}
b_1[0] &= b_1[0] - \eta \times \delta_{\text{hidden},1} \\
&= 0.35 - 0.5 \times 0.0109 \\
&= 0.3446
\end{aligned}
$$

$$
\begin{aligned}
b_1[1] &= b_1[1] - \eta \times \delta_{\text{hidden},2} \\
&= 0.35 - 0.5 \times 0.0120 \\
&= 0.3440
\end{aligned}
$$

Neue Biases für den Output Layer b_2:

$$
\begin{aligned}
b_2[0] &= b_2[0] - \eta \times \delta_{\text{output},1} \\
&= 0.60 - 0.5 \times 0.1385 \\
&= 0.5307
\end{aligned}
$$

$$
\begin{aligned}
b_2[1] &= b_2[1] - \eta \times \delta_{\text{output},2} \\
&= 0.60 - 0.5 \times (-0.0381) \\
&= 0.6191
\end{aligned}
$$

Vorwärts- und Rückwärtsdurchlauf (Iteration 2)

1. Ergebnisse der ersten Iteration

Nach der ersten Iteration lauten die aktualisierten Bias-Werte wie folgt:

$$b_1^{(1)} = 0.3447, \quad b_2^{(1)} = 0.3431$$
$$b_1^{(2)} = 0.5304, \quad b_2^{(2)} = 0.6788$$

Diese Werte wurden mit der Lernrate $\eta = 0.5$ und den Gradienten für die jeweiligen Biases aktualisiert.

2. Vorwärtsdurchlauf der zweiten Iteration

Im Vorwärtsdurchlauf der zweiten Iteration werden die neuen Bias-Werte verwendet, um die Aktivierungen in der versteckten und der Ausgabeschicht zu berechnen.

2.1 Versteckte Schicht

Die Nettoeingänge für die Neuronen in der versteckten Schicht berechnen sich wie folgt:

$$z_1 = x_1 \cdot w_{11} + x_2 \cdot w_{21} + b_1^{(1)}$$
$$z_2 = x_1 \cdot w_{12} + x_2 \cdot w_{22} + b_2^{(1)}$$

Die berechneten Werte sind:

$$z_1 = 0.05 \cdot 0.15 + 0.10 \cdot 0.20 + 0.3447 = 0.3557$$
$$z_2 = 0.05 \cdot 0.25 + 0.10 \cdot 0.30 + 0.3431 = 0.3671$$

Die Aktivierungen in der versteckten Schicht nach Anwendung der Sigmoid-Funktion sind:

$$h_1 = \sigma(z_1) = \frac{1}{1 + e^{-0.3557}} = 0.5889$$
$$h_2 = \sigma(z_2) = \frac{1}{1 + e^{-0.3671}} = 0.5909$$

2.2 Ausgabeschicht

Die Nettoeingänge für die Neuronen in der Ausgabeschicht werden wie folgt berechnet:

$$z_1^{(2)} = h_1 \cdot w_{11}^{(2)} + h_2 \cdot w_{21}^{(2)} + b_1^{(2)}$$
$$z_2^{(2)} = h_1 \cdot w_{12}^{(2)} + h_2 \cdot w_{22}^{(2)} + b_2^{(2)}$$

Die berechneten Werte sind:

$$z_1^{(2)} = 0.5889 \cdot 0.40 + 0.5909 \cdot 0.45 + 0.5304 = 1.1779$$
$$z_2^{(2)} = 0.5889 \cdot 0.50 + 0.5909 \cdot 0.55 + 0.6788 = 1.4039$$

Die Aktivierungen in der Ausgabeschicht nach Anwendung der Sigmoid-Funktion sind:

$$y_1 = \sigma(z_1^{(2)}) = \frac{1}{1 + e^{-1.1779}} = 0.7646$$
$$y_2 = \sigma(z_2^{(2)}) = \frac{1}{1 + e^{-1.4039}} = 0.8028$$

3. Fehlerberechnung in der zweiten Iteration

Die Fehler in der zweiten Iteration sind:

$$\delta_{y_1} = y_1 - y_{\text{target},1} = 0.7646 - 0.01 = 0.7546$$
$$\delta_{y_2} = y_2 - y_{\text{target},2} = 0.8028 - 0.99 = -0.1872$$

4. Rückwärtsdurchlauf (Backpropagation)

4.1 Gradienten der Ausgabeschicht-Biases

Die Gradienten der Fehlerfunktion in Bezug auf die Biases der Ausgabeschicht werden mit der Sigmoid-Ableitung berechnet:

$$\frac{\partial E}{\partial b_1^{(2)}} = \delta_{y_1} \cdot \sigma'(z_1^{(2)})$$
$$\frac{\partial E}{\partial b_2^{(2)}} = \delta_{y_2} \cdot \sigma'(z_2^{(2)})$$

Dabei gilt:

$$\sigma'(z) = \sigma(z) \cdot (1 - \sigma(z))$$

Die Ableitungen der Sigmoid-Funktion sind:

$$\sigma'(z_1^{(2)}) = 0.7646 \cdot (1 - 0.7646) = 0.1797$$

$$\sigma'(z_2^{(2)}) = 0.8028 \cdot (1 - 0.8028) = 0.1584$$

Die Gradienten der Biases in der Ausgabeschicht sind:

$$\frac{\partial E}{\partial b_1^{(2)}} = 0.7546 \cdot 0.1797 = 0.1356$$

$$\frac{\partial E}{\partial b_2^{(2)}} = -0.1872 \cdot 0.1584 = -0.0296$$

4.2 Gradienten der versteckten Schicht-Biases

Der Fehler in der versteckten Schicht wird durch den Fehler in der Ausgabeschicht beeinflusst. Der Gradient der Fehlerfunktion in Bezug auf die Biases der versteckten Schicht lautet:

$$\frac{\partial E}{\partial b_1^{(1)}} = \left(\delta_{y_1} \cdot w_{11}^{(2)} + \delta_{y_2} \cdot w_{12}^{(2)}\right) \cdot \sigma'(z_1)$$

$$\frac{\partial E}{\partial b_2^{(1)}} = \left(\delta_{y_1} \cdot w_{21}^{(2)} + \delta_{y_2} \cdot w_{22}^{(2)}\right) \cdot \sigma'(z_2)$$

Zuerst berechnen wir die Ableitungen der Sigmoid-Funktion für die versteckte Schicht:

$$\sigma'(z_1) = 0.5889 \cdot (1 - 0.5889) = 0.2421$$

$$\sigma'(z_2) = 0.5909 \cdot (1 - 0.5909) = 0.2417$$

Dann berechnen wir die Gradienten:

$$\frac{\partial E}{\partial b_1^{(1)}} = (0.7546 \cdot 0.40 + -0.1872 \cdot 0.50) \cdot 0.2421 = 0.0188$$

$$\frac{\partial E}{\partial b_2^{(1)}} = (0.7546 \cdot 0.45 + -0.1872 \cdot 0.55) \cdot 0.2417 = 0.0228$$

5. Bias-Updates (zweite Iteration)

Die neuen Bias-Werte nach der zweiten Iteration werden mit der Lernrate $\eta = 0.5$ wie folgt aktualisiert:

$$b_1^{(1)} = b_1^{(1)} - \eta \cdot \frac{\partial E}{\partial b_1^{(1)}} = 0.3447 - 0.5 \cdot 0.0188 = 0.3411$$

$$b_2^{(1)} = b_2^{(1)} - \eta \cdot \frac{\partial E}{\partial b_2^{(1)}} = 0.3431 - 0.5 \cdot 0.0228 = 0.3399$$

$$b_1^{(2)} = b_1^{(2)} - \eta \cdot \frac{\partial E}{\partial b_1^{(2)}} = 0.5304 - 0.5 \cdot 0.1356 = 0.4604$$

$$b_2^{(2)} = b_2^{(2)} - \eta \cdot \frac{\partial E}{\partial b_2^{(2)}} = 0.6788 + 0.5 \cdot 0.0296 = 0.6376$$

6. Endergebnisse der zweiten Iteration

Die neuen Bias-Werte nach der zweiten Iteration sind:

$$b_1^{(1)} = 0.3411, \quad b_2^{(1)} = 0.3399$$

$$b_1^{(2)} = 0.4604, \quad b_2^{(2)} = 0.6376$$