

Name: Florian
Datum: 21.10.2021

Testprotokoll

Im Folgenden werden Sie Ihre Implementation des neuronalen Netzes, sowie das neuronale Netz selbst auf ihre Funktionalität überprüfen. Notieren Sie sich zuerst zu jedem Testfall Ihre Erwartungen. Öffnen Sie danach Ihre Implementation der `KonsolenAnwendung.py` und erstellen das zu testende neuronale Netz.

Vergleichen Sie Ihre Erwartungen mit den Ergebnissen.

1. Testfälle zur Funktionalität der Implementierung:

Was passiert beim Erstellen eines neuronalen Netzes...

a. ohne Input-Layer?

Erwartung: Es sollte kein neuronales Netz erstellt werden.

Beobachtung: Fehlermeldung: „Erstellung des Netzwerkes konnte nicht abgeschlossen werden. Ein Inputlayer fehlt.“

b. ohne Hidden-Layer?

Erwartung: Hidden-Layer = optional => Erstellung eines Netzes

Beobachtung: Erfolg! Erstellung des Netzwerkes wurde abgeschlossen. Erwartung erfüllt.

c. ohne Output-Layer?

Erwartung: Output-Layer ist notwendig => es sollte kein Netz erstellt werden.

Beobachtung: Fehlermeldung: „Erstellung des Netzwerkes konnte nicht abgeschlossen werden. Ein Outputlayer fehlt.“

d. mit einem Output-Layer ohne Neuronen?

Erwartung: Ein Layer ohne Neuronen = kein Layer => kein Netz

Beobachtung: Funktioniert. Auch für Input- & Hidden-Layer.

Training nicht möglich.

=> Abfrage bei der Frage nach Neuronen-Anzahl

Auswertung der Implementation

Name: Florian
Datum: 21.10.21

Zur Überprüfung der Genauigkeit der folgenden Netzwerkkonfigurationen gehen Sie wie folgt vor:

- I. Erstellen Sie das neuronale Netzwerk wie unten angegeben.
- II. Trainieren Sie das neuronale Netzwerk.
- III. Speichern Sie das neuronale Netzwerk unter dem angegebenen Namen.
- IV. Testen Sie die Genauigkeit mithilfe der Testdaten und notieren Sie die Ergebnisse.
- V. Testen Sie als letztes die manuelle Eingabe folgender Zahlen: 0, 8, 6, 5, 7.

2. Testfälle verschiedener Konfigurationen des Neuronalen Netzes:

a. Speichername: standard

Input-Layer: 128; Hidden-Layer: 128; Output-Layer: 10

Beobachtung: Bei der 0 las es eine 2; bei 8 $\Rightarrow$ 6; bei 6 $\Rightarrow$ 6
bei 5 $\Rightarrow$ 5; bei 7 $\Rightarrow$ 3 [Genauigkeit: 100%]
Funktioniert nicht perfekt. 8 und 6 sehen ähnlich aus sowie
7 und 3, aber 0 und 2 nicht $\Rightarrow$ 2/5 richtig

b. Speichername: doppelter_hidden

Input-Layer: 128; Hidden-Layer: 128; Hidden-Layer: 128; Output-Layer: 10

Beobachtung: Bei der 0 las es eine 3; 8 $\Rightarrow$ 3; 6 $\Rightarrow$ 5;
5 $\Rightarrow$ 9; 7 $\Rightarrow$ 3; 0/5 komplett versagt, andere Fehler als
Netz dazu, obwohl Genauigkeit mit den Testdaten bei
100% lag

c. Speichername: kein_hidden

Input-Layer: 128; Hidden-Layer: ---; Output-Layer: 10

Beobachtung: Genauigkeit: 100,0%
Es las überall eine 3. Beim zweiten Versuch: 0 $\Rightarrow$ 7;
8 $\Rightarrow$ 3; 6 $\Rightarrow$ 3; 5 $\Rightarrow$ 5; 7 $\Rightarrow$ 6; 1/5.

d. Speichername: mehr_output

Input-Layer: 128; Hidden-Layer: ---; Output-Layer: 11

Beobachtung: Genauigkeit: 99,99%, aber las 0 $\Rightarrow$ 0; 8 $\Rightarrow$ 3; 6 $\Rightarrow$ 7;
5 $\Rightarrow$ 5; 7 $\Rightarrow$ 6, was mit der Genauigkeit nicht übereinstimmt
2/5.

3. Beschreibe Möglichkeiten für weitere Testfälle, bei dem die neuronalen Netzwerke versagen. Beispiel: 5 wird als 5 interpretiert.