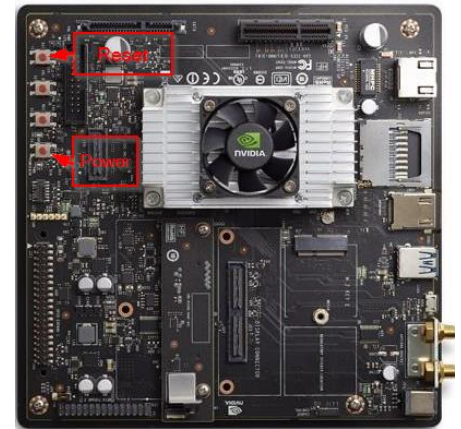


Deep Learning Praktikum

Klaus Dorer, Ralph Isenmann – Hochschule Offenburg



Die Jetson Boards sind komplette Computer mit integrierter Grafikkarte. Damit lassen sich Deep Convolutional Netzwerke besonders effizient trainieren und ausführen. Starten sie, wenn Bildschirmkabel, Tastatur-Dongle und Strom angesteckt sind, zunächst das Jetson Board mit dem Power Knopf.



1 Backpropagation Netzwerke

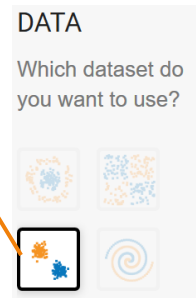
Die Übungen finden im Tensorflow Playground statt: <https://playground.tensorflow.org>

1.1 Daten

Es gibt vier Datensätze, die jeweils in zwei Dimensionen x und y liegen. Das Netzwerk soll die orangenen Punkte als orange und die blauen als blau klassifizieren. Starten sie mit dem Datensatz links unten.



Gestartet und zurück gesetzt wird das Netz mit den Control Buttons



Das Ergebnis des Lernens wird im Output Bereich grafisch angezeigt. Zusätzlich zeigt der Test loss, wie gut das Netzwerk unbekannte Daten richtig klassifiziert, bzw. wie klein der Fehler der Klassifikation noch ist.

1.2 Aufgabenstellung

Aktivierungsfunktion

- Wählen sie Activation->Linear. Wie groß muss das Netz mindestens sein, um dieses Muster zu lernen?
- Setzen sie das Netz zurück (2 Hidden Layers, 4, 2 Neurons). Welche der Inputmuster (Data) können mit linearer Aktivierungsfunktion gelernt werden?
- Testen sie jetzt jedes Muster mit jeder Aktivierungsfunktion

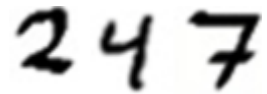
Netzwerkgröße

Wählen sie jetzt das spiralförmige Muster rechts unten mit Aktivierungsfunktion Tanh und allen Features.

- Untersuchen sie systematisch, welchen Einfluss die Anzahl und Größe der Hiddenschichten auf den Lernerfolg hat.
- Versuchen sie mit allen auf der Seite zur Verfügung stehenden Mitteln, ein möglichst gutes Ergebnis (Test loss) beim Spiralmuster zu erzielen.

2 Ziffern erkennen

Eine der ersten Anwendungen im Bereich Bilderkennung und quasi das HelloWorld von Deep Neural Networks ist das Erkennen von handschriftlichen Ziffern.

The image shows three handwritten digits: '2', '4', and '7', written in a dark, slightly blurred ink on a white background.

2.1 Daten

Der MNIST Datensatz enthält 70.000 Bilder von handschriftlich erstellten Ziffern. Er ist bereits in die Keras Umgebung integriert bzw. wird beim ersten Lernen aus dem Netz geladen.

2.2 Aufgabenstellung

Öffnen sie zunächst ein Terminal (Ctrl-Alt-T) und wechseln sie in das Verzeichnis ml2_deeplearning_with_keras:

```
cd ./dev/ml2_deeplearning_with_keras /
```

Aktivieren sie die Keras Umgebung:

```
py3_keras
```

Starten sie den Jupyter Notebook Server

```
jupyter notebook
```

Navigieren sie in der im Browser geöffneten Seite zu MnistClassification /MnistClassifier.ipynb und öffnen sie das Buch durch einfachen Click. Ein neues Browser Tab öffnet sich.

- Arbeiten sie die Zellen nacheinander ab, indem sie sie mit Shift-Enter ausführen
- Prüfen sie jetzt, wie gut er von ihnen geschriebene Ziffern erkennt. Öffnen sie dazu eines der Bilder aus dem data Ordner mit dem Programm Pinta. Übermalen sie die vorhandene Ziffer in Schwarz und zeichnen dann eine Ziffer in Weiß. Speichern sie das neue Bild als jpg unter einem neuen Namen im selben Ordner und führen sie die entsprechende Zelle im Notebook erneut aus.

3 RoboCup Objekte erkennen

Auf dem Jetson Board sollen mit Hilfe von Deep Convolutional Networks Objekte auf Bildern erkannt werden, die unser Fußballroboter Sweaty erkennen muss.

3.1 Daten

- Die im Workflow verwendeten Daten sind bereits auf dem Jetson Board installiert. Man kann sie aber auch mit Notebook MagmaDataset.ipynb neu erzeugen. Die Daten sind Bilder aus der RoboCup Domäne und zeigen verschiedene Objekte die dort vorkommen.
- Es gibt 4917 Trainingsbilder in Farbe
- Die Bilder können augmentiert werden (Rotation, Translation, Scherung, Skalierung, Spiegelung)

Klasse	Kategorie
0	Ball
1	Torpfosten
2	Hindernis
3	L-Linie
4	X-Linie
5	T-Linie
6	11m-Punkt
7	Fuß

3.2 Aufgabenstellung

Wenn sie den Jupyter Notebook Server von Aufgabe 2 noch am Laufen haben, können sie den einfach weiter verwenden. Ansonsten die Schritte zum Starten des Servers aus Aufgabe 2 durchführen.

Navigieren sie in der im Browser geöffneten Seite zu HsoMagmaClassification /MagmaClassifier.ipynb und öffnen sie das Buch durch einfachen Click. Ein neues Browser Tab öffnet sich.

Untersuchen sie systematisch:

- Den Einfluss der Menge an Trainingsdaten auf den Lernerfolg. Die Menge an Trainingsdaten kann im Knoten „Partitioning“ geändert werden.
- Den Einfluss des Optimizers (SGD, ADAM, ADADELTA, ADAGRAD, RMSPROP, NADAM, ADAMAX) auf den Lernerfolg. Der Optimizer kann im Unterworkflow „Keras“ im Knoten „Keras Network Learner“ im Tab „Options“ eingestellt werden

Konfigurieren sie immer nur einen entsprechenden Knoten im Workflow und Starten sie das Training. **Vergessen** Sie nicht die Standardeinstellung der Parameter wiederherzustellen für den zweiten Teil.

Untersuchen sie den Einfluss der folgenden Parameter und Konfigurationen systematisch:

- Augmentierung
- Schichten im Netz löschen oder hinzufügen
- Denselayer verkleinern/vergrößern
- Zahl der Epochen erhöhen
- Batchsize variieren
- <Ihr eigener Vorschlag> (am besten im Labor kurz fragen, ob ok)