Deep Learning (del) – Mini Challenge 1

Inhaltverzeichnis

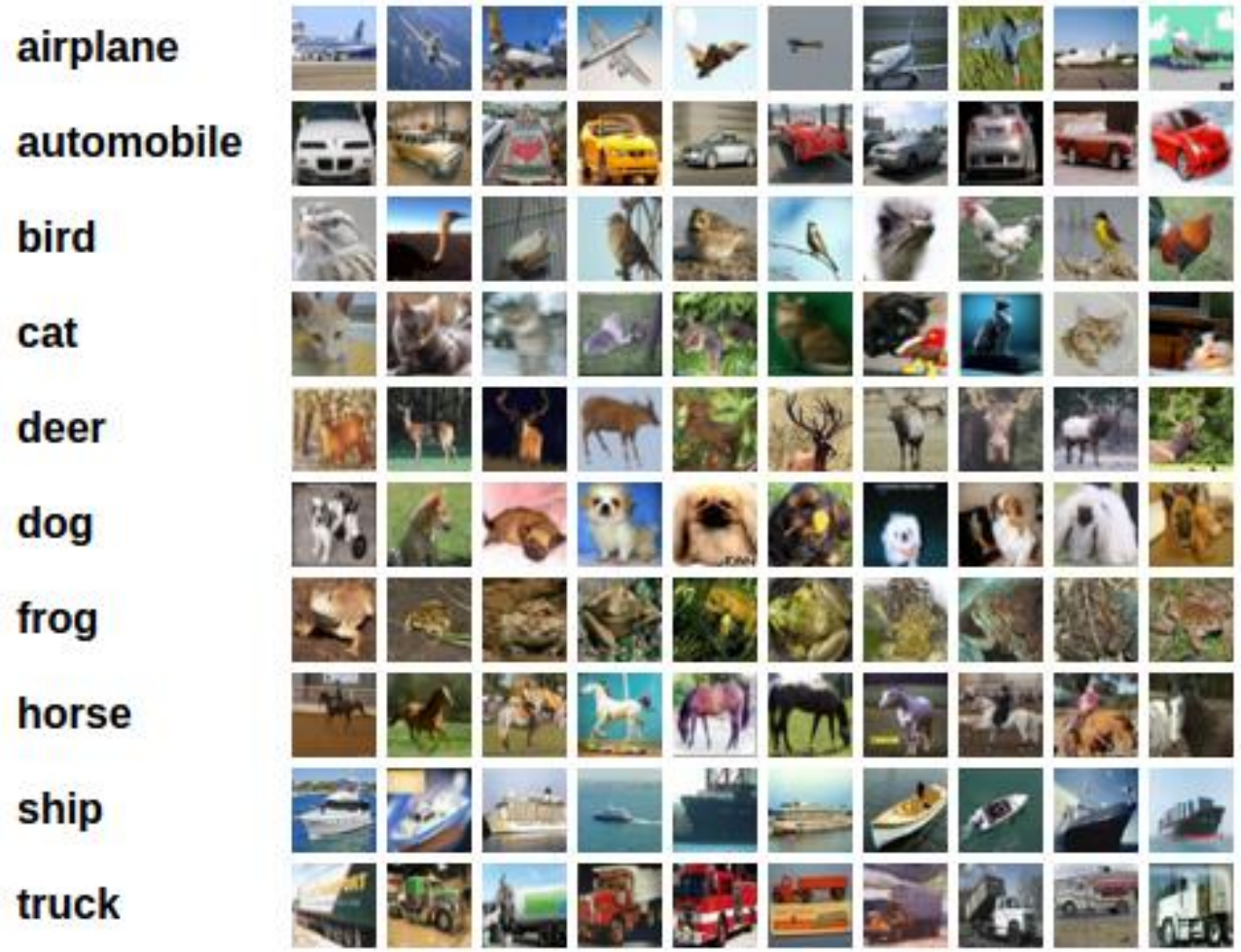
del-mc1 Workspace – Weights & Biases (wandb.ai)

Datensatz

Cifar-10

Umfasst 50‘000 Trainingsbilder und 10‘000 Testbilder. Dimension 32x32x3 Pixel.

- Verteilung der Klassen untersucht (Accuracy)
- Mittelwerte und Standardabweichung berechnet
- Daten bereits normiert [0;1]



Übersicht Challenge Notebook

1

Aufgabenstellung: Eine Blaue Box beschreibt die Aufgabe aus der Aufgabenstellung 'SGDS_DEL_MC1.pdf'

Antworte: Eine Grüne Box beschreibt die Bearbeitung / Reflektion der Aufgabenstellung

2

Experiment	Lernrate	Grösse hL1	Batchsize
1	1e-1, 1e-2, 1e-3, 1e-4, 1e-5	128*	32*
2	1e-2**	64, 128, 256, 512, 1024	32*
3	1e-2**	1024**	4, 16, 32, 64, 128

3

Ziele der Challenge

- Ein Gefühl zum Verhalten von Hyperparameter entwickeln
- Theorie in der Praxis mit Pytorch anwenden und nachvollziehen
- Experimente definieren und beurteilen

Modellkomplexität und Hyperparameter

MLP hL1 / hL2

1. mit einem hidden Layer
2. mit zwei hidden Layer

Experiment	Lernrate	Grösse hL1	Grösse hL2	Batchsize
1	1e-1, 1e-2, 1e-3, 1e-4, 1e-5	1024*	256*	32*
2	1e-2**	1024, 2048	256*	32*
3	1e-2**	1024**	64, 128, 256, 512	32*
4	1e-2**	1024**	512*	4, 16, 32, 64, 128

CNN Simple

- mit zwei Convolution Layers
- einem linearen hidden Layer

Experiment	Lernrate	Grösse Filter1	Grösse Filter2	Batchsize	Kernel-Grösse	Stride	Padding
1	1e-1, 1e-2, 1e-3, 1e-4, 1e-5	32*	64*	32*	3*	1*	1*
2	1e-2**	16, 32, 64	64*	32*	3*	1*	1*
3	1e-2**	32**	32, 64, 96	32*	3*	1*	1*
4	1e-2**	32**	64**	8, 16, 32, 64	3*	1*	1*
5	1e-2**	32**	64**	32**	3, 5, 7	1*	1*
6	1e-2**	32**	64**	32**	3**	1, 2, 3	1*
7	1e-2**	32**	64**	32**	3**	1**	0, 1, 2

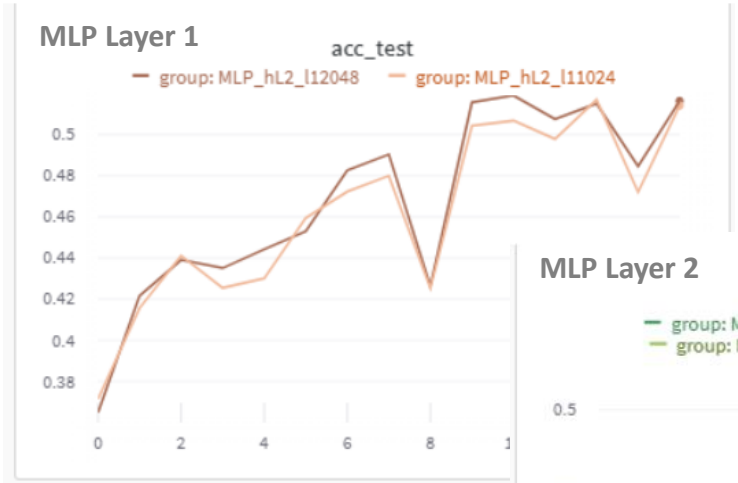
(*) gewählter Startwert

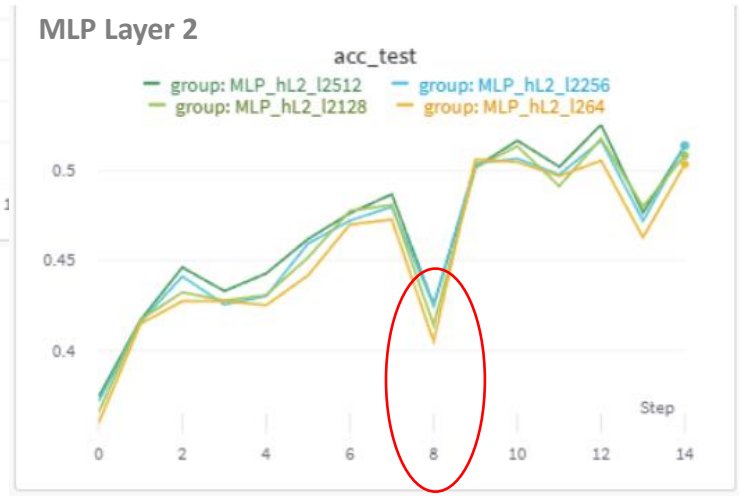
(**) optimaler Wert

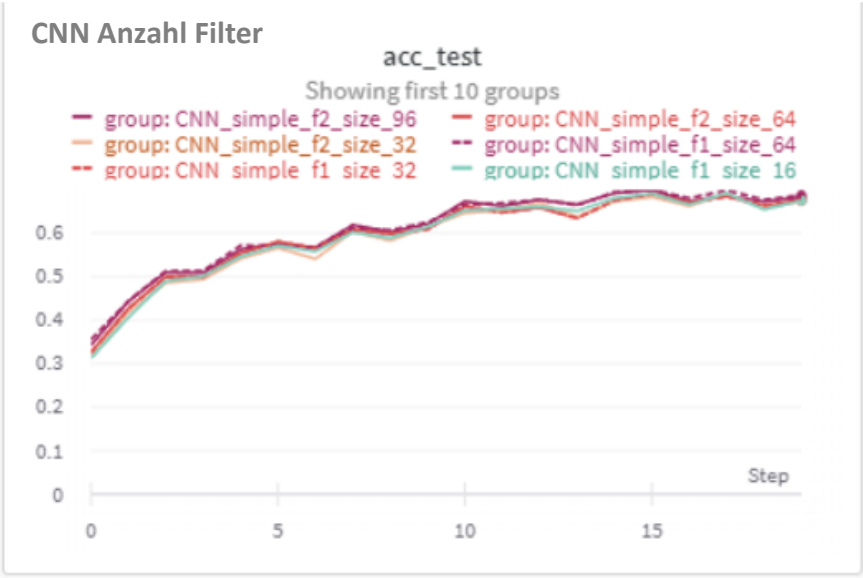
Modelkomplexität

Erkenntnisse

- MLP: Die Layergrösse hatten nur geringen Einfluss
  - (32x32x3) 3072 -> hL1 -> hL2 -> 10
  - hL1 : 1024, 2048
  - hL2 : 64, 128, 256, 512
- CNN: Die Anzahl der Filter hatte wenig Einfluss
  - F1: 16, 32, 64
  - F2: 32, 64, 96

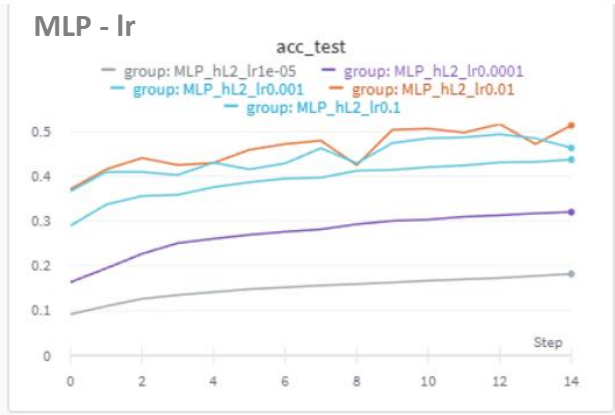






Bei allen Trainings gab es bei Epoche 8 eine starke Schwankung (MLP).

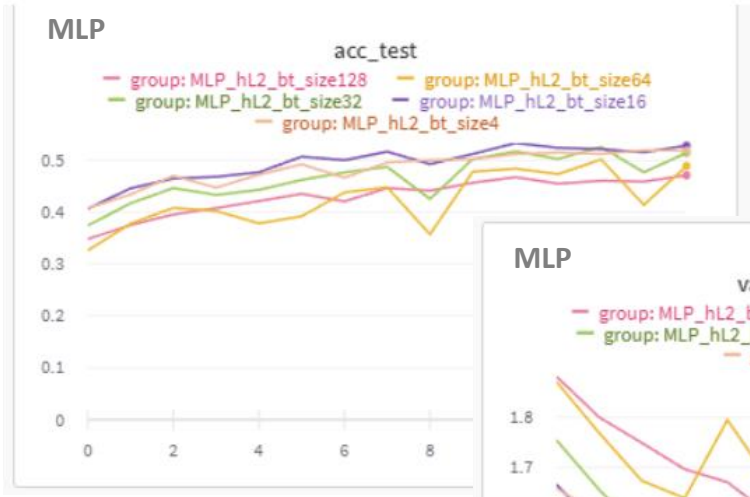
Eine hohe Lernrate ist weniger stabil.
(0.01 wurde verwendet, orange

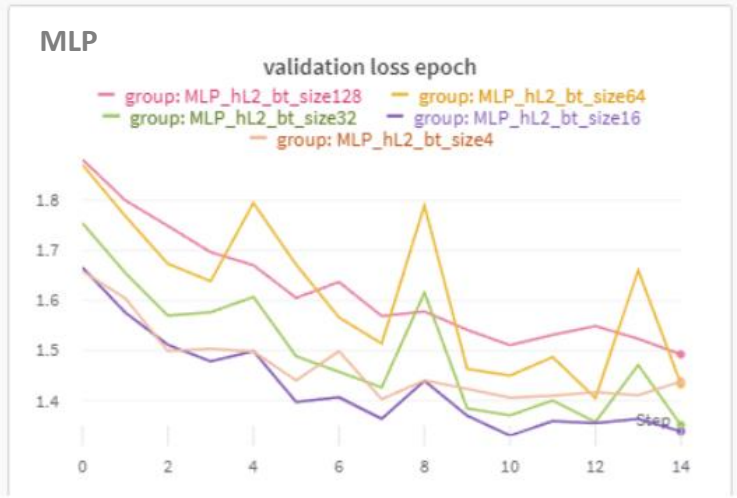


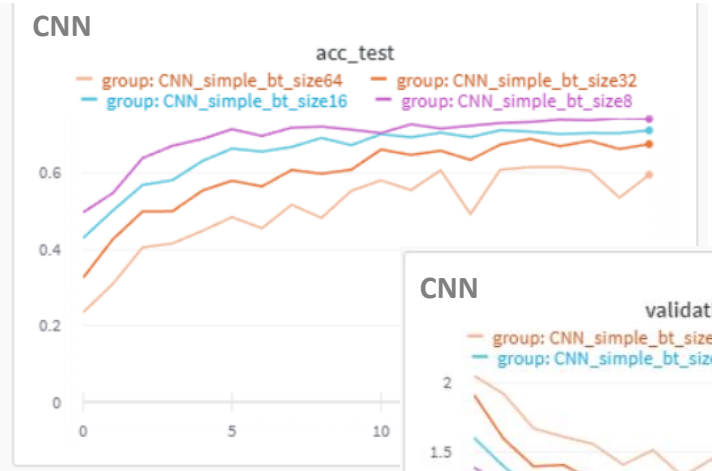
Batchgrösse

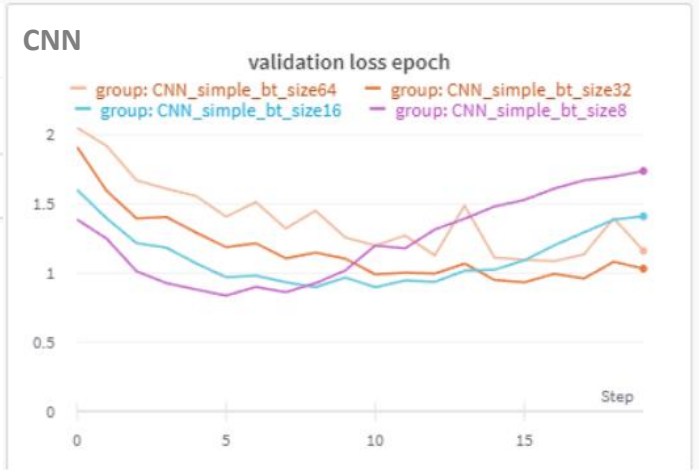
Erkenntnisse

- Die Batchsize hat Einfluss auf die Trainingsdauer
- sowie auf die Stabilität des Trainings
  - Überraschend dass die Stabilität bei tiefen Batchgrössen trotzdem vorhanden ist









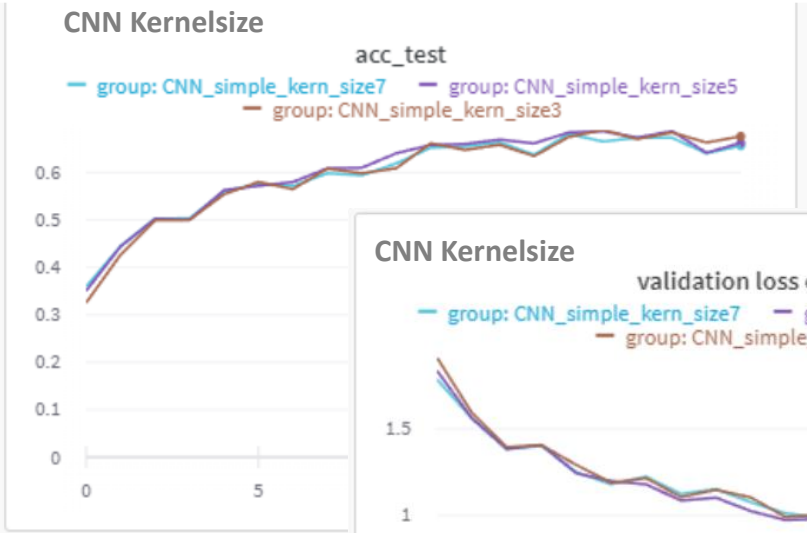
deutlich schnelleres Overfitting bei tiefer Batchgrösse

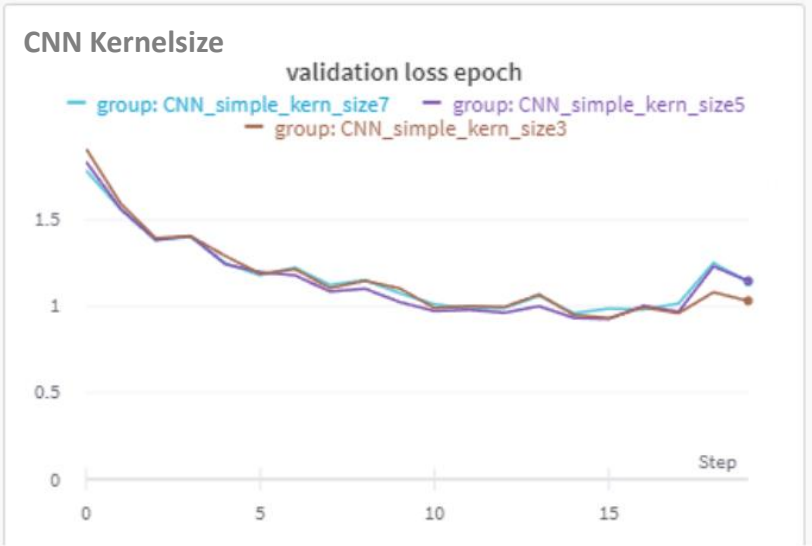
MLP-Batchsize = 4: 25 Minuten
MLP-Batchsize = 64: 4 Minuten
CNN-Batchsize = 8: 22 Minuten
CNN-Batchsize = 64: 6 Minuten

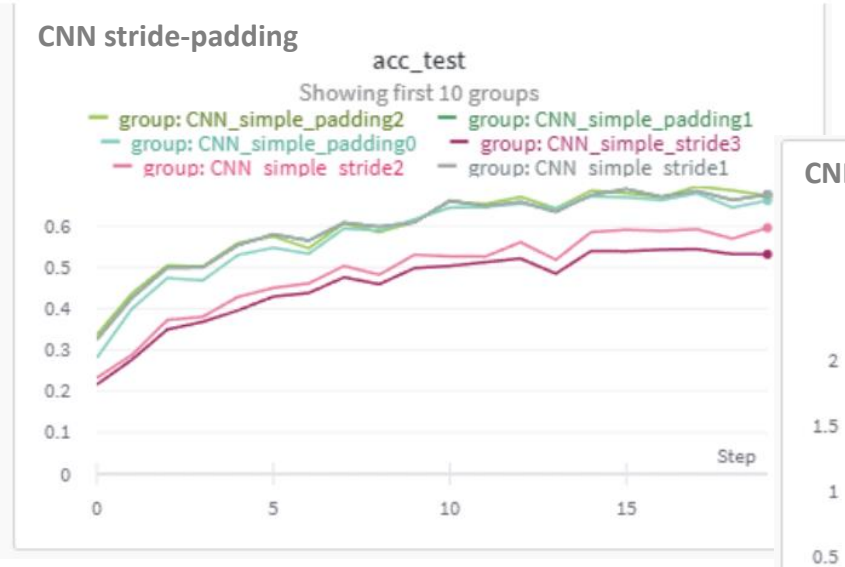
Kernelgrösse, Stride, Padding

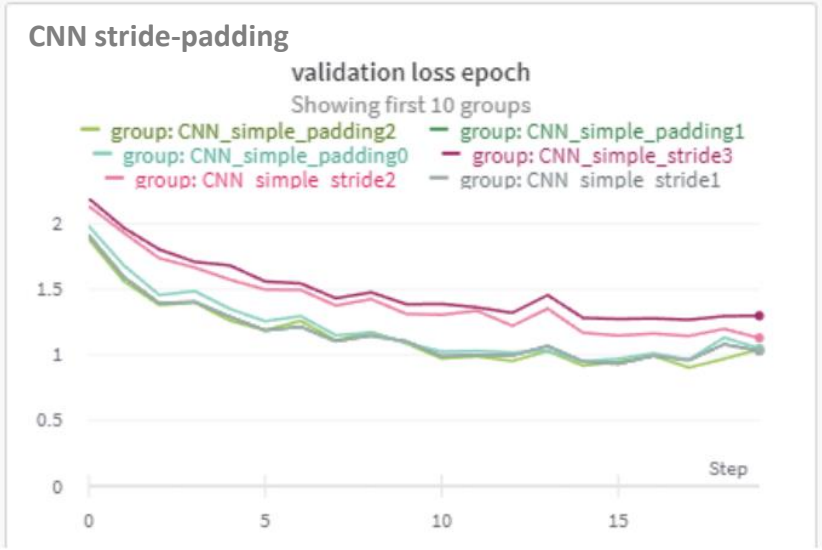
Erkenntnisse

- Leichtes Overfitting mit höherer Kernelgrösse
  - mehr globale Informationen
- Stride grösser 1 trainiert weniger schnell
  - Evtl. auf maxPooling verzichten
- Padding von 2 mit leicht höherer Accuracy
  - Bilder sind aber eher zentriert





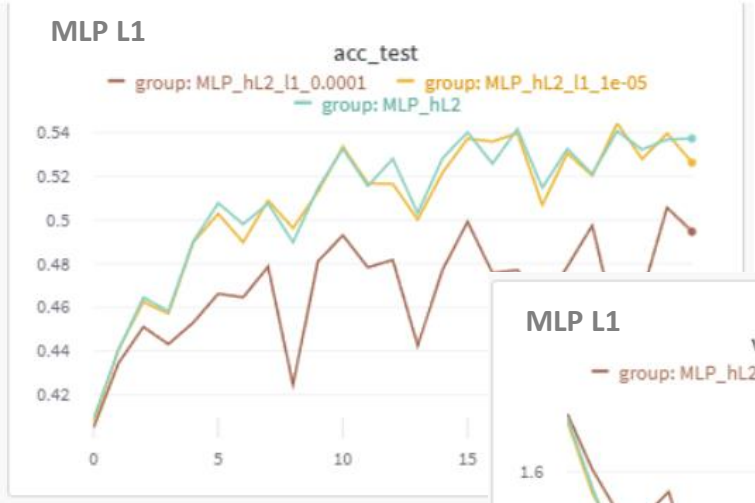


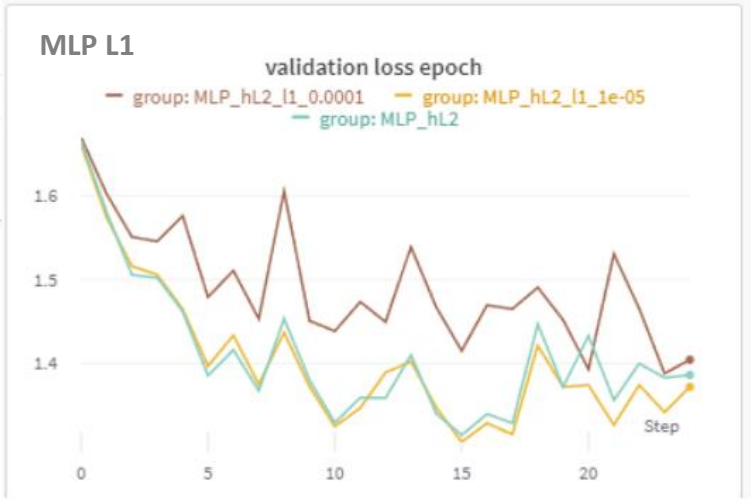


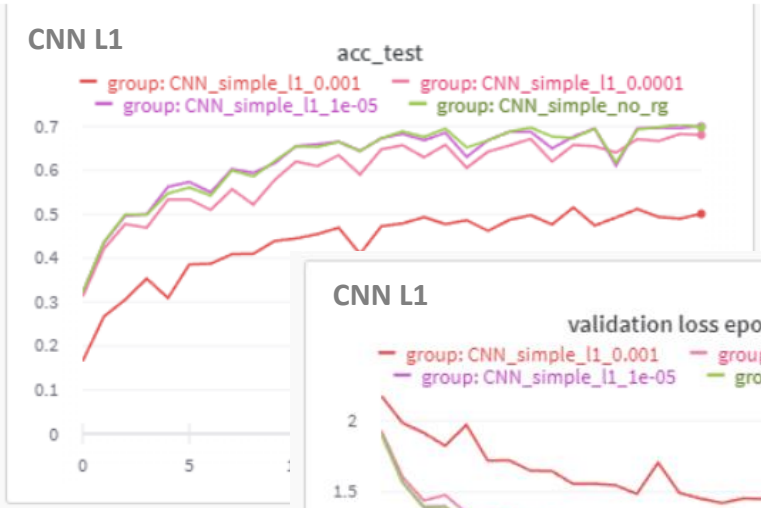
Regularisierung

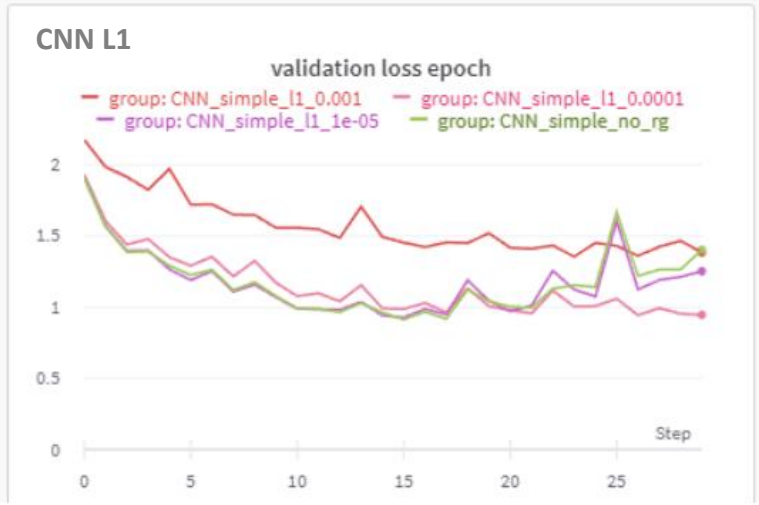
Erkenntnisse

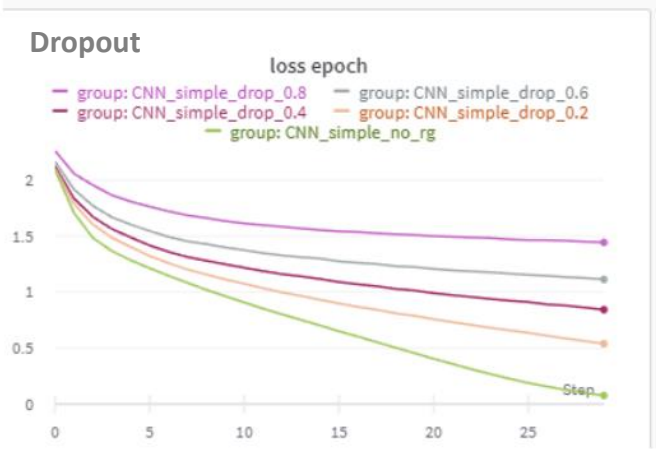
- L1 hat starken Einfluss auf das Training. Auch die Stabilität leidet (MLP). Evtl. Lernrate verkleinern.
- L2 greift weniger stark ein
- Dropout lässt das Netzwerk weniger schnell trainieren
- Early Stopping auch nützlich, weniger übersichtlich

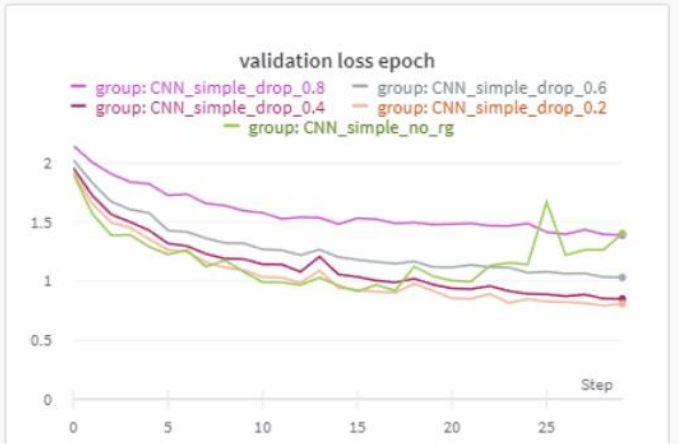








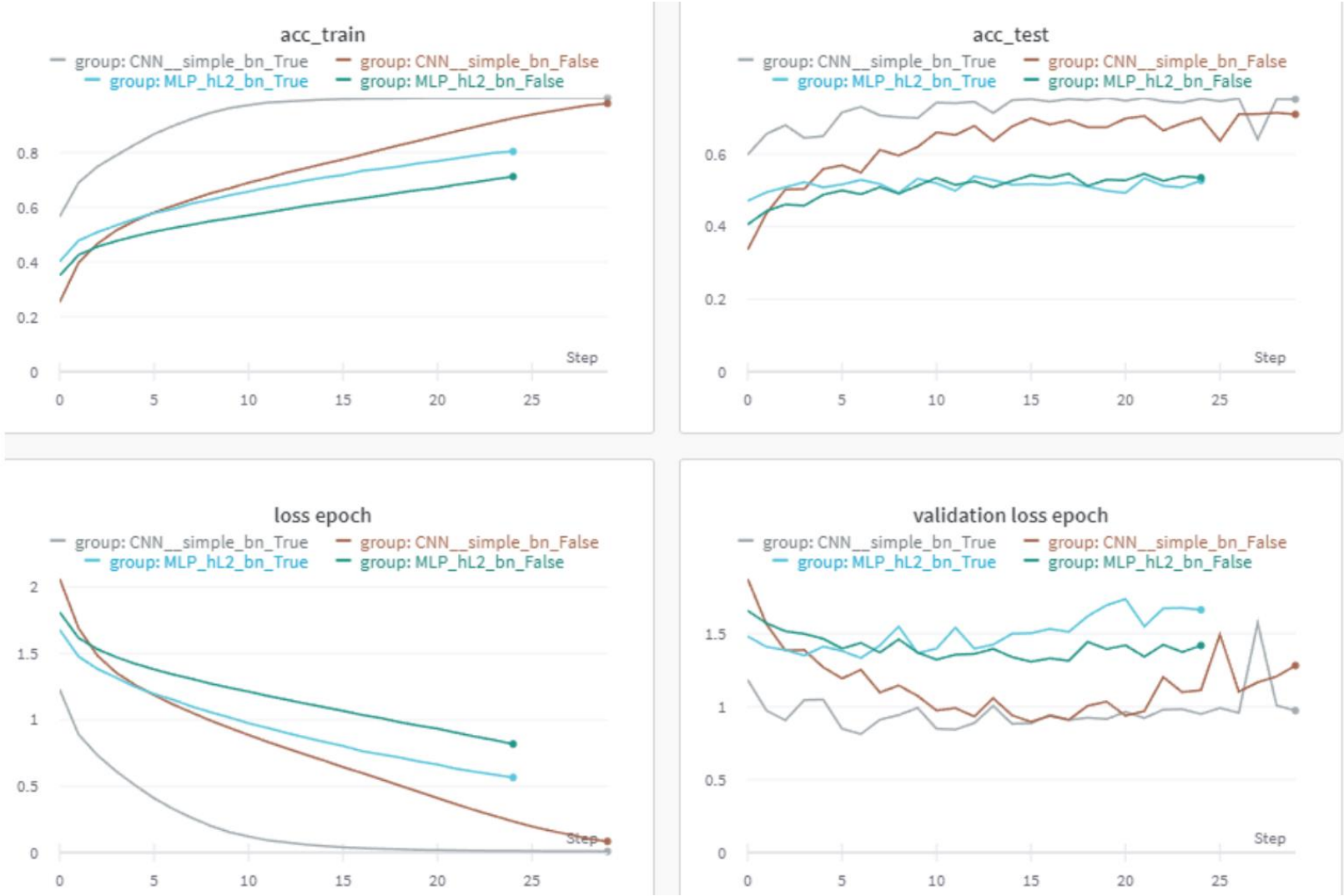




Batchnorm

Erkenntnisse

- Versprechen der Theorie korrekt
  - deutlich schnelleres Training
  - leichte Regularisierung
- Achtung Overfitting



Fazit

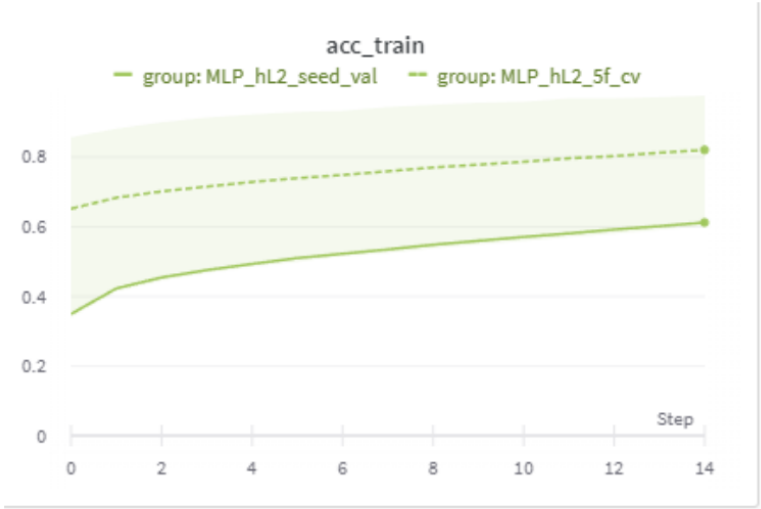
Erkenntnisse

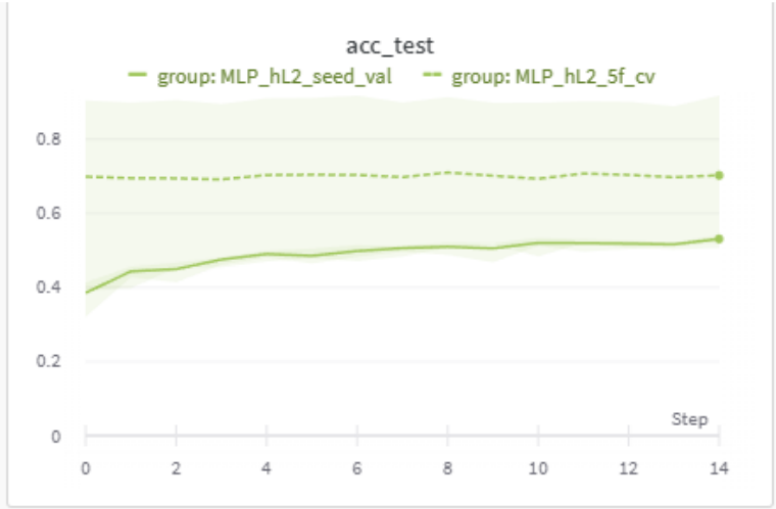
- Modelkomplexität anhand der Aufgabe wählen (MLP < CNN)
  - Hyperparameter Tuning kompensiert eine schlechte Wahl nicht
- Erstelle einen Ablauf für das Tuning
  - Starten mit verschiedenen Lernraten (Adam)
  - Testen verschiedener Anzahl Layer und deren Grössen (MLP)
  - Teste die Anzahl der Filter (CNN)
  - Einfluss der Batchsize testen, sofern dass die GPU zulässt
  - Regularisierungs-Methoden einsetzen
  - Verwendung von Batchnorm
- Modellvergleiche mit Cross-Validation oder unterschiedlichen Seeds

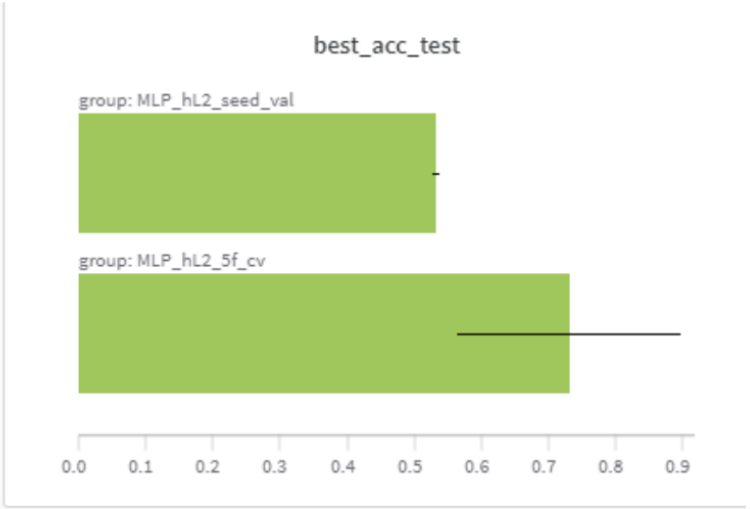
Offene Fragen

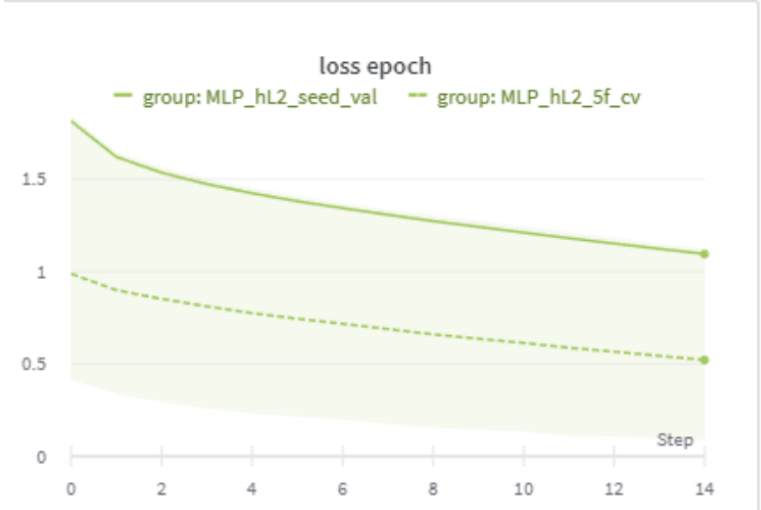
- Wie verhalten sich Layergrösse (MLP) und Anzahl Filter (CNN) bei komplexeren Architekturen
- Welche Regularisierungs-Methoden können/dürfen kombiniert werden
  - L1/L2 + Dropout
  - Early Stopping + Dropout
  - L1 + L2
- Adam mit lr=0.0001 nicht trainiert, kleinere lr verwenden und Momentum aufbauen (MLP)
- Cross-Validation Error in der Challenge finden

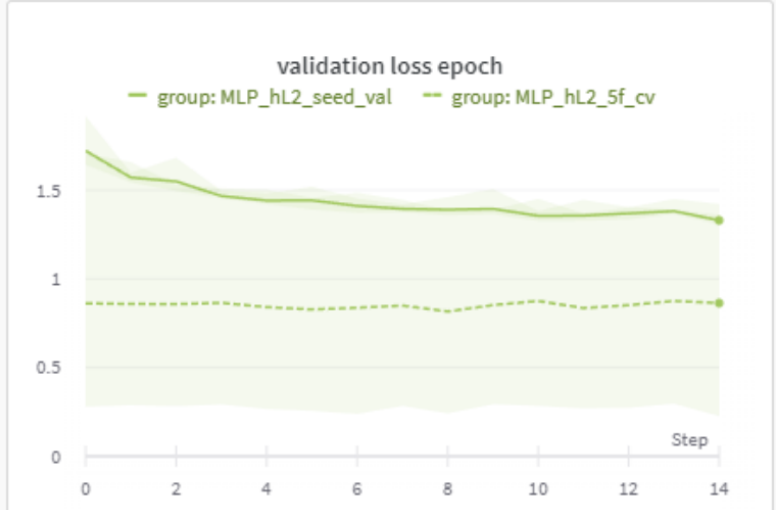
Cross-Validation

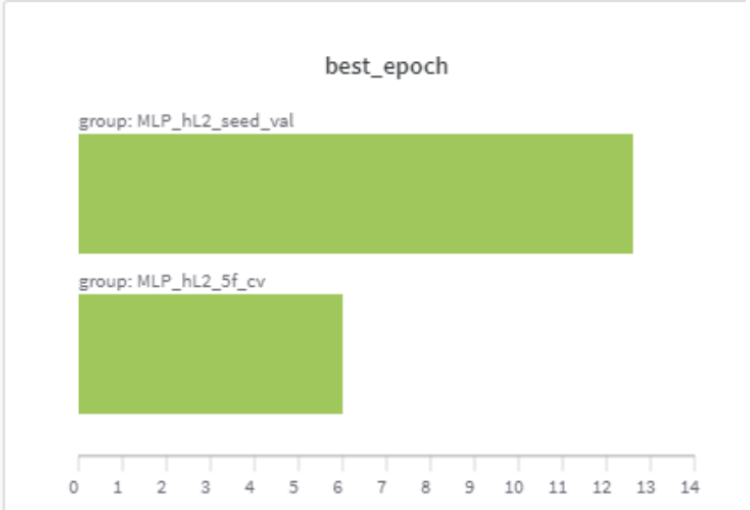












Modell	Lernrate	Grösse hL1	Grösse hL1	Batchsize	Train Acc	Test Acc	Test Schätzfehler (σ)
MLP-hl2-5f-cv (--)	1e-2	1024	512	16	0.8208	0.7026	0.1646
MLP-hl2-seeds (-)	1e-2	1024	512	16	0.6124	0.5310	0.0070