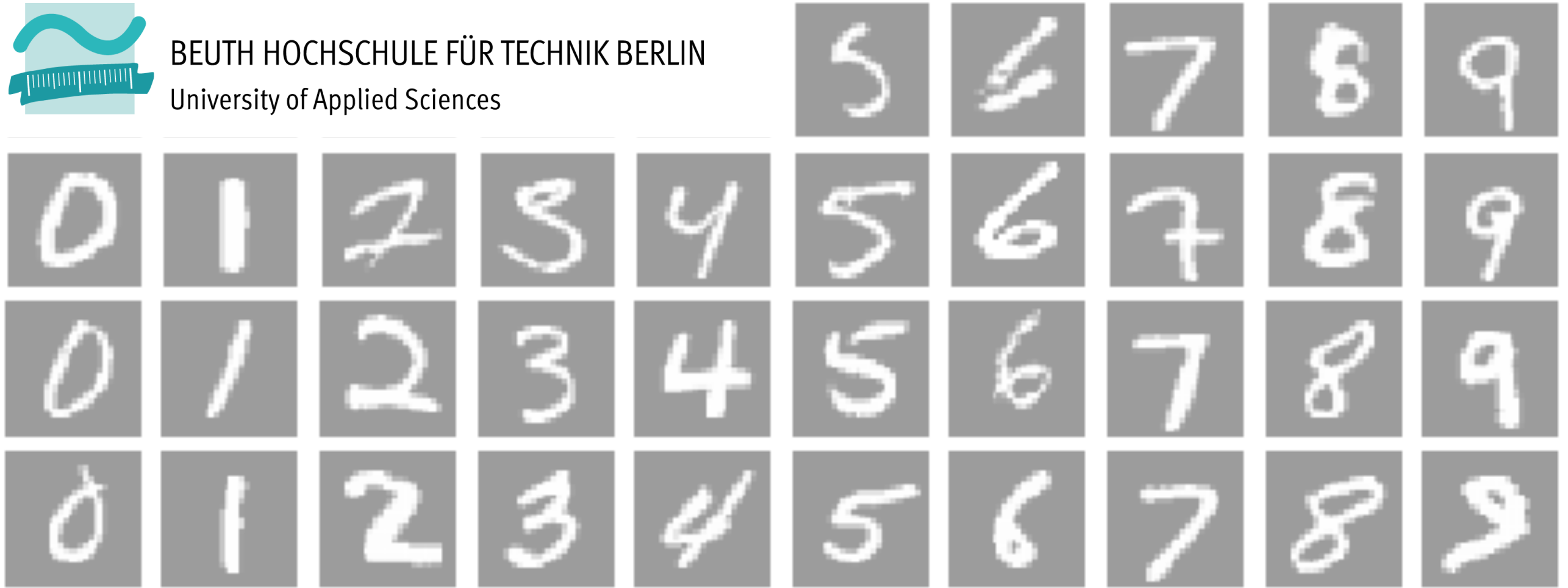




BEUTH HOCHSCHULE FÜR TECHNIK BERLIN
University of Applied Sciences



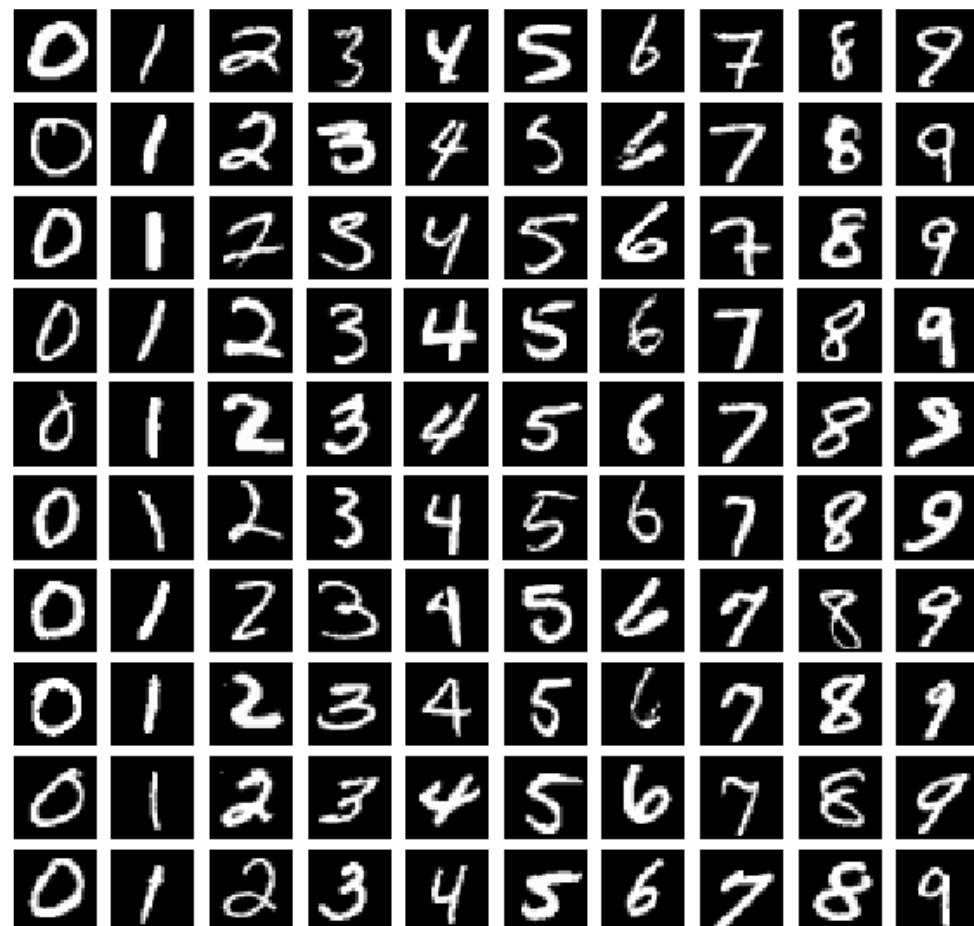
INTERACTIVE SYSTEMS – PROJEKTVORSTELLUNG

IMPLEMENTIERUNG EINER ERKENNUNG VON ZIFFERN

AGENDA

1. Projektidee – Rückblick
2. Support Vector Machines
3. Neuronale Netze
4. Convolutional Neural Networks
5. Vergleich von SVM, NN und CNN

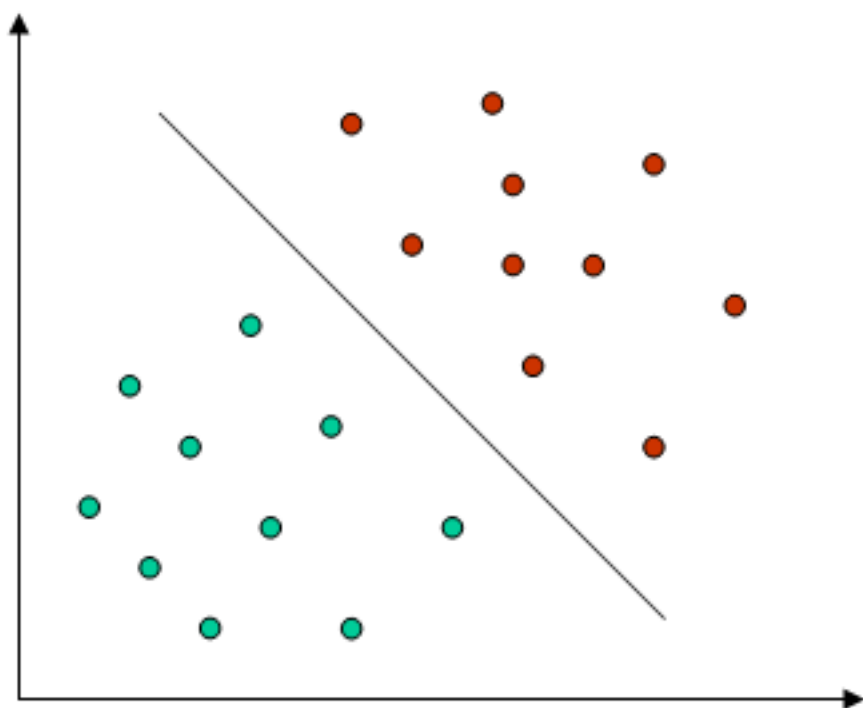
1. PROJEKTIDEE – RÜCKBLICK



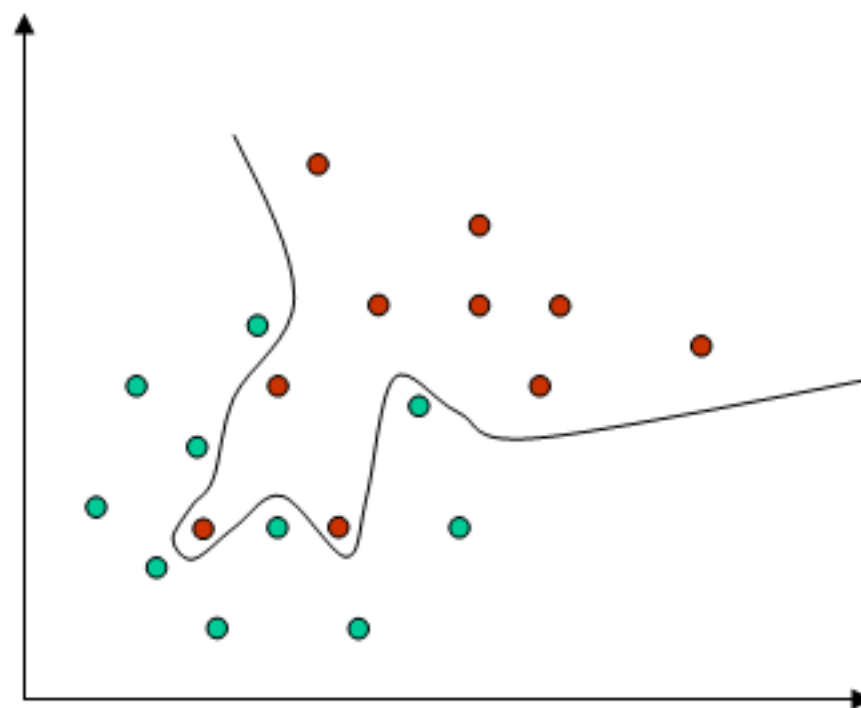
- ▶ Datensatz MNIST
- ▶ Testen von verschiedenen Modellen
- ▶ Support Vector Machine, Neuronale Netze, CNNs
- ▶ Hyperparameter, z. B. Anzahl der Neuronen / Layer

2. SUPPORT VECTOR MACHINE

- ▶ Hyperparameter: Trennfläche
 - ▶ Linear: Erkennungsrate 84 %
 - ▶ Polynom 2., 4., 9. Grades



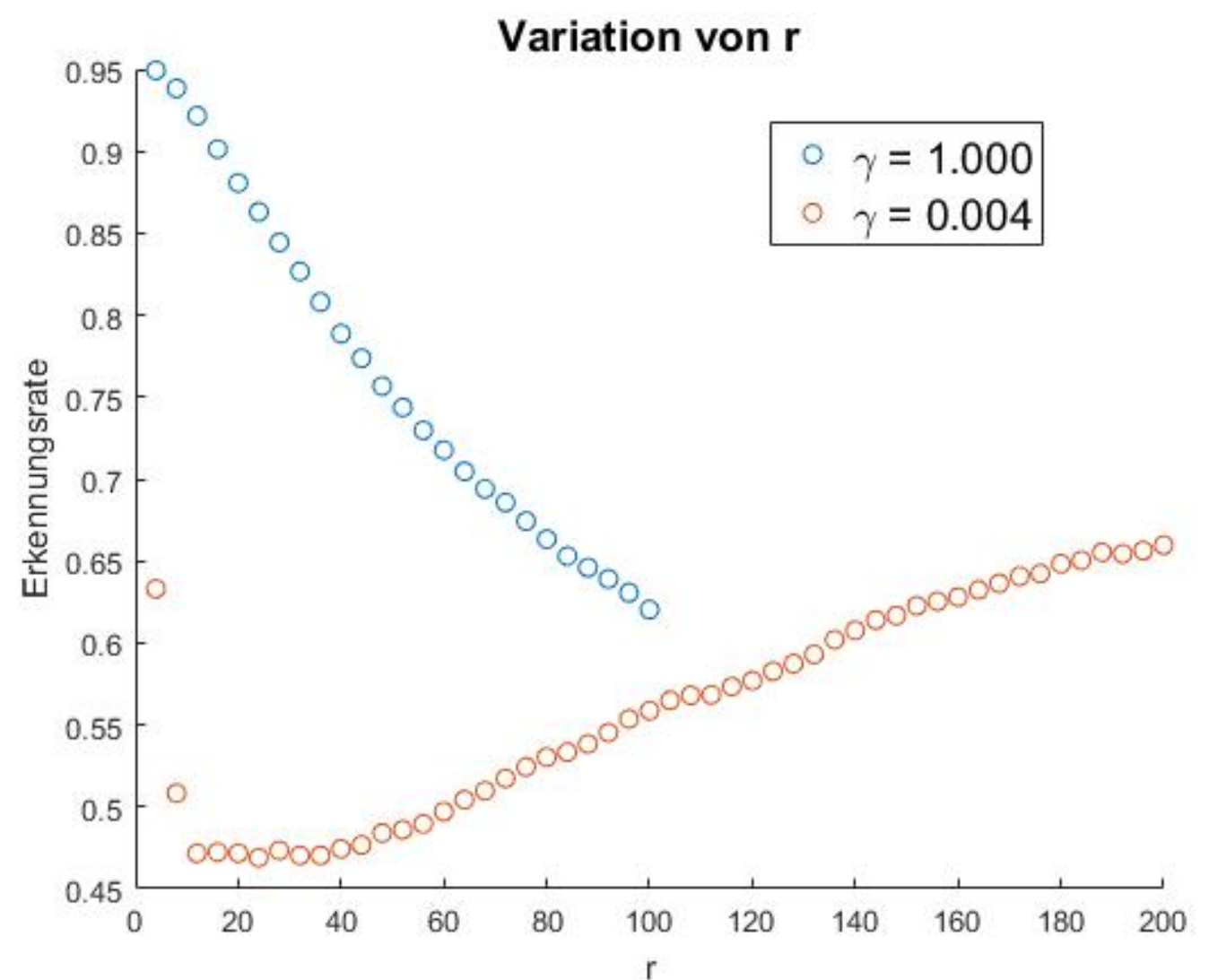
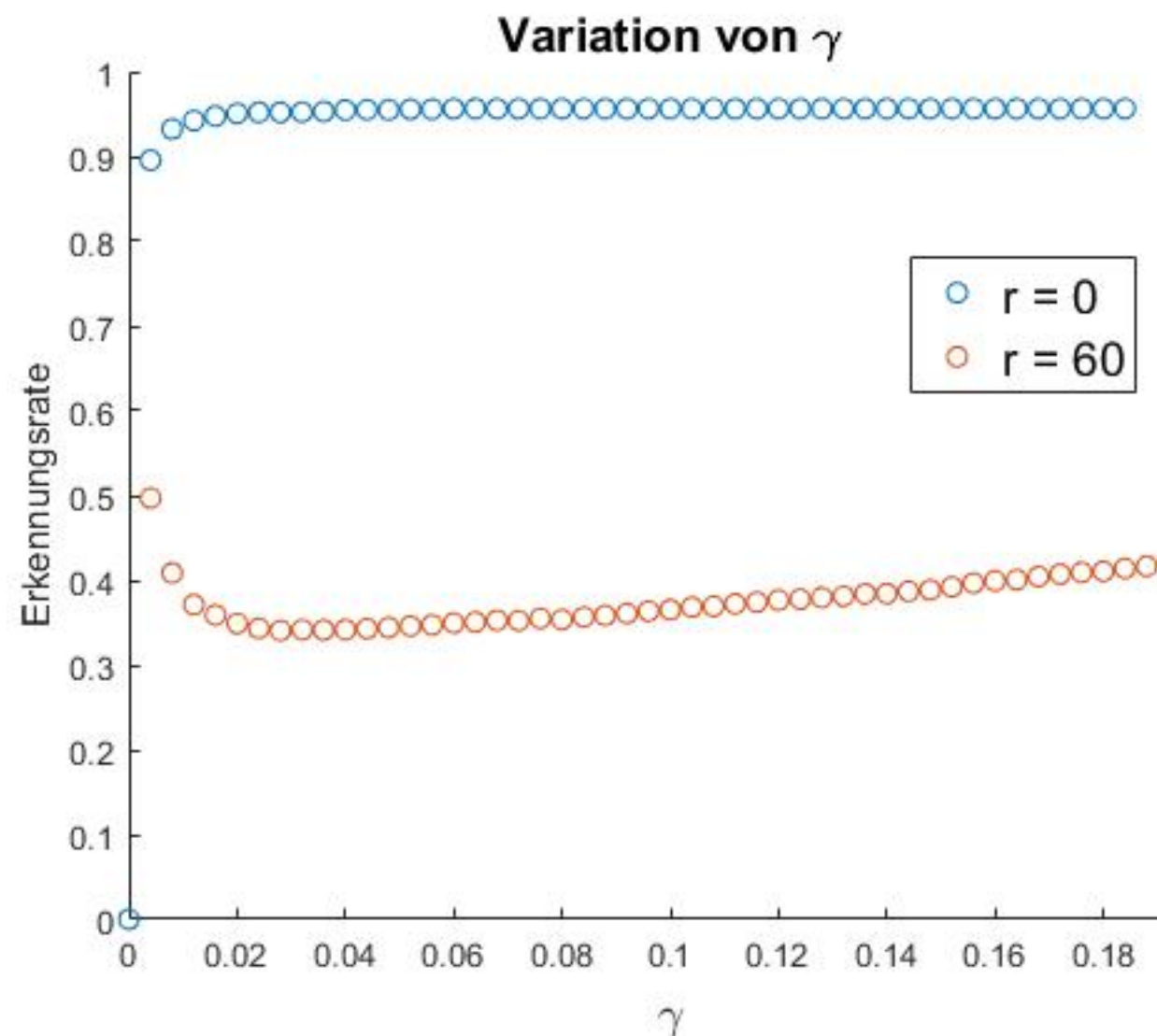
Linear trennbar



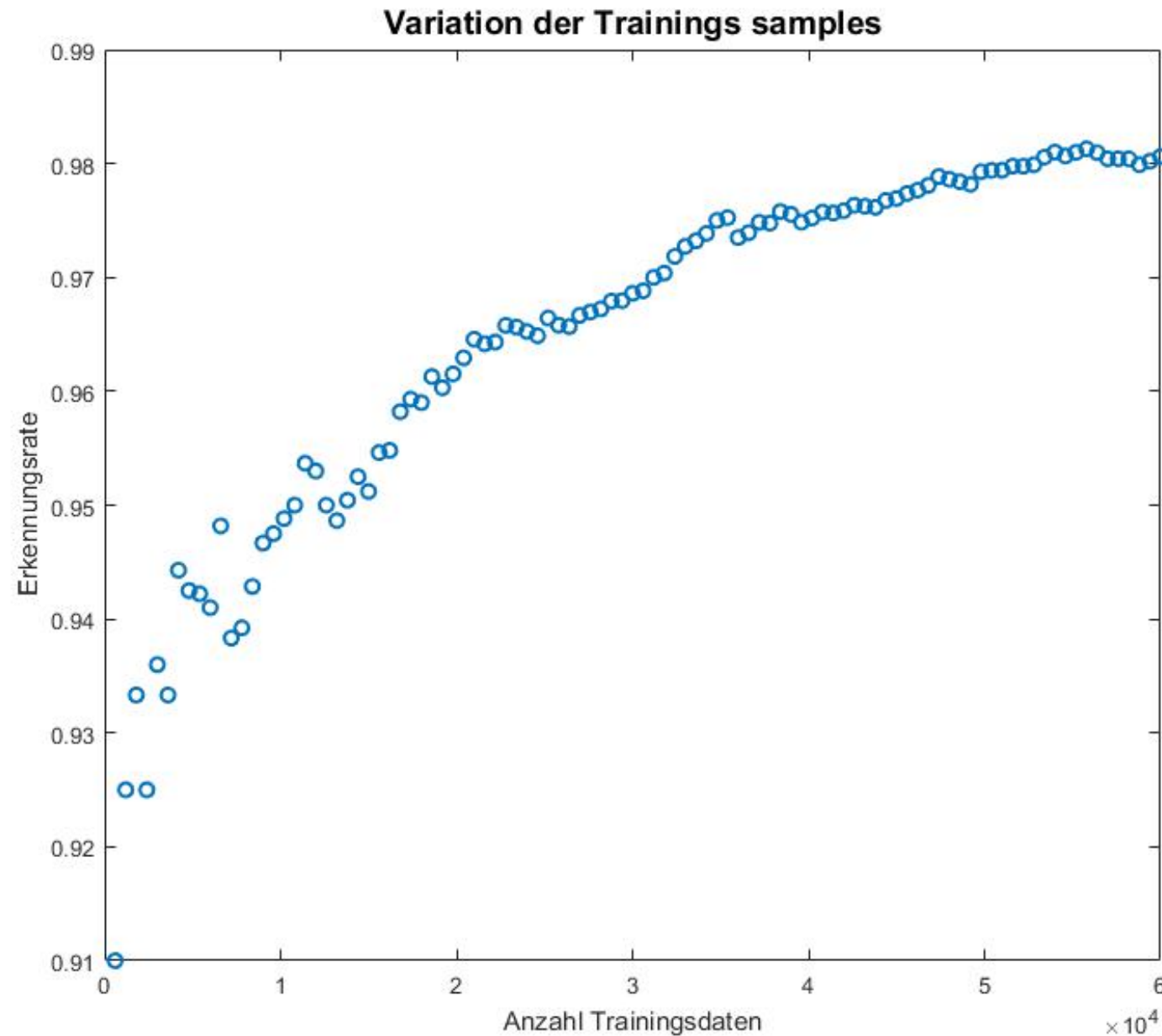
Nicht linear trennbar

NICHT-LINEARE HYPERFLÄCHE – POLYNOM 2. GRADES

- γ und r beeinflussen die Form der Trennfläche
- Training: 18000 Samples



NICHT-LINEARE HYPERFLÄCHE – POLYNOM 2. GRADES



- Je mehr Trainings-Samples, desto bessere Erkennungsrate

NICHT-LINEARE HYPERFLÄCHE – POLYNOM N. GRADES

► Polynom 4. Grades

gamma	r	Erkennungsrate
0,0001	1	0,87
0,001	1	0,63
0,0013	1	0,56
0,01	1	0,42
0,1	1	0,93
1	1	0,95
1	0	0,94
1	10	0,94
1	100	0,43

► Polynom 9. Grades

gamma	r	Erkennungsrate
0.0013	1	0.2
0.1	1	0.94
0.0001	1	0.81
1	0	0.87
10	0	0.87
0.01	0	0.87
0.001	0	0.77

► höherer Grad N führt zu längerer Trainingsdauer!

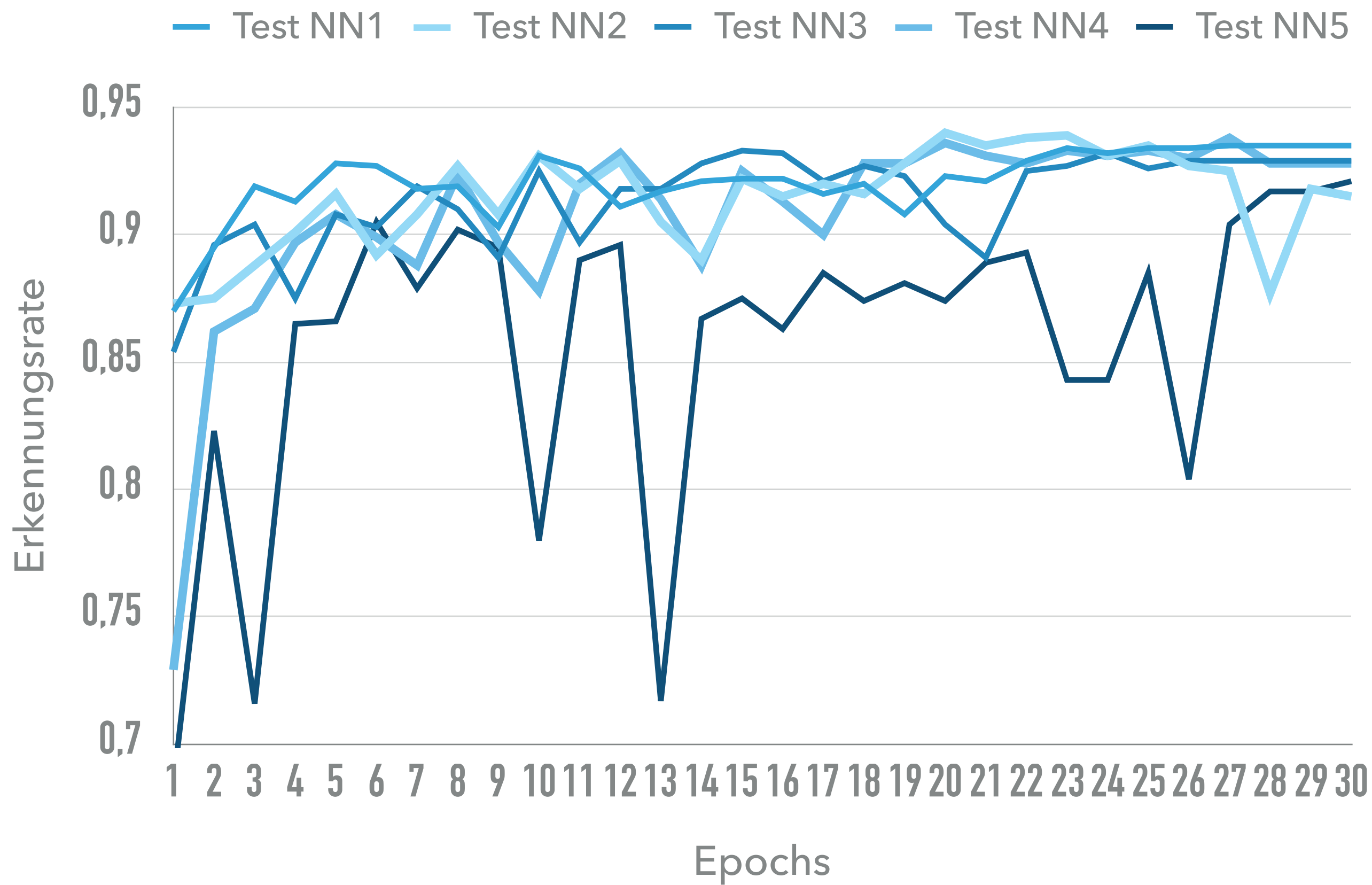
3. NEURONALE NETZE

► 5 verschiedene Netzarchitekturen

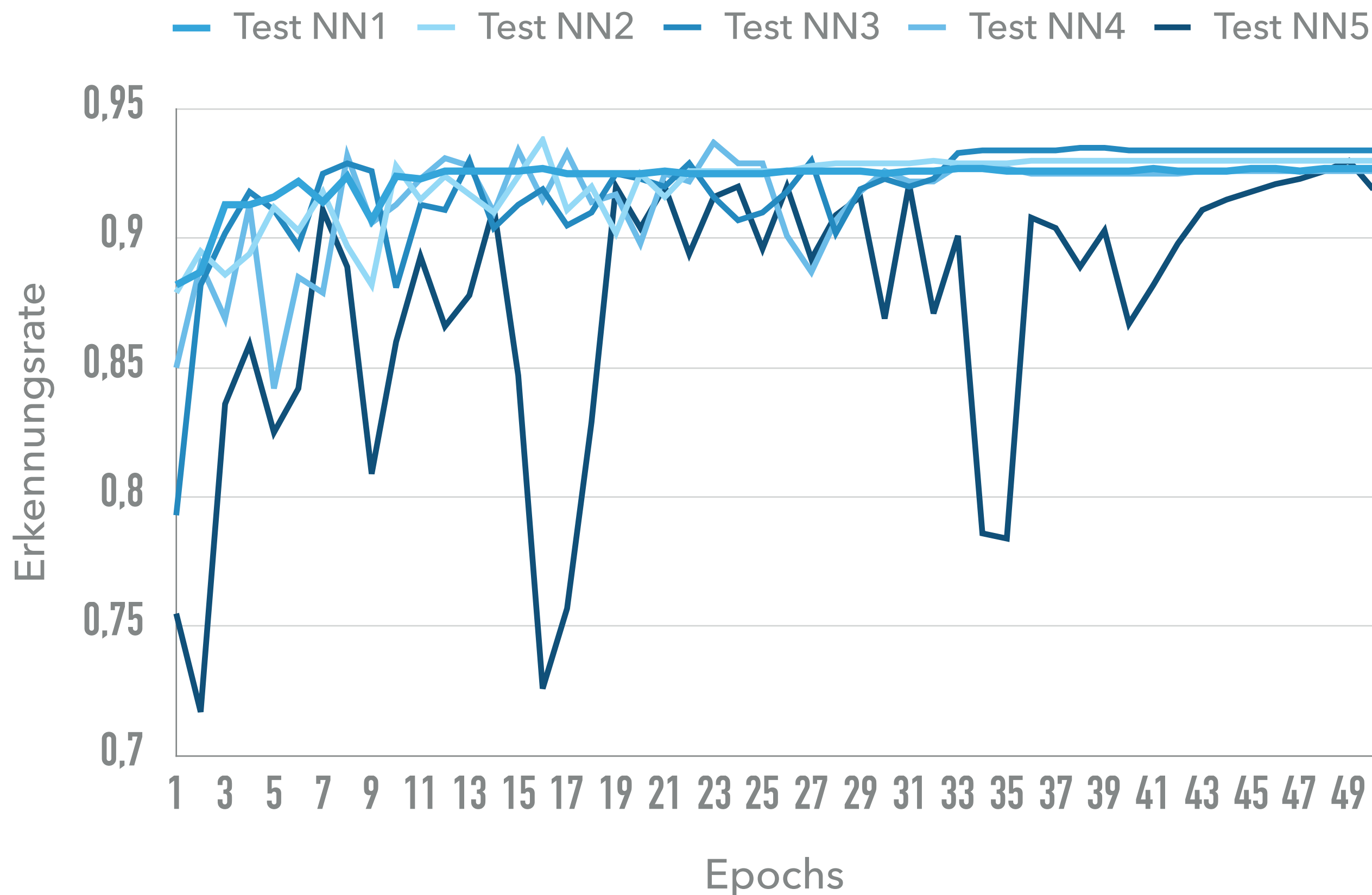
ID	architecture (number of neurons in each layer)	test error for best validation [%]	simulation time [h]	weights [milions]
1	1000, 500, 10	0.49	23.4	1.34
2	1500, 1000, 500, 10	0.46	44.2	3.26
3	2000, 1500, 1000, 500, 10	0.41	66.7	6.69
4	2500, 2000, 1500, 1000, 500, 10	0.35	114.5	12.11
5	9×1000 , 10	0.44	107.7	8.86

Quelle: "Deep Big Simple Neural Nets Excel on Handwritten Digit Recognition"
<https://arxiv.org/pdf/1003.0358.pdf>

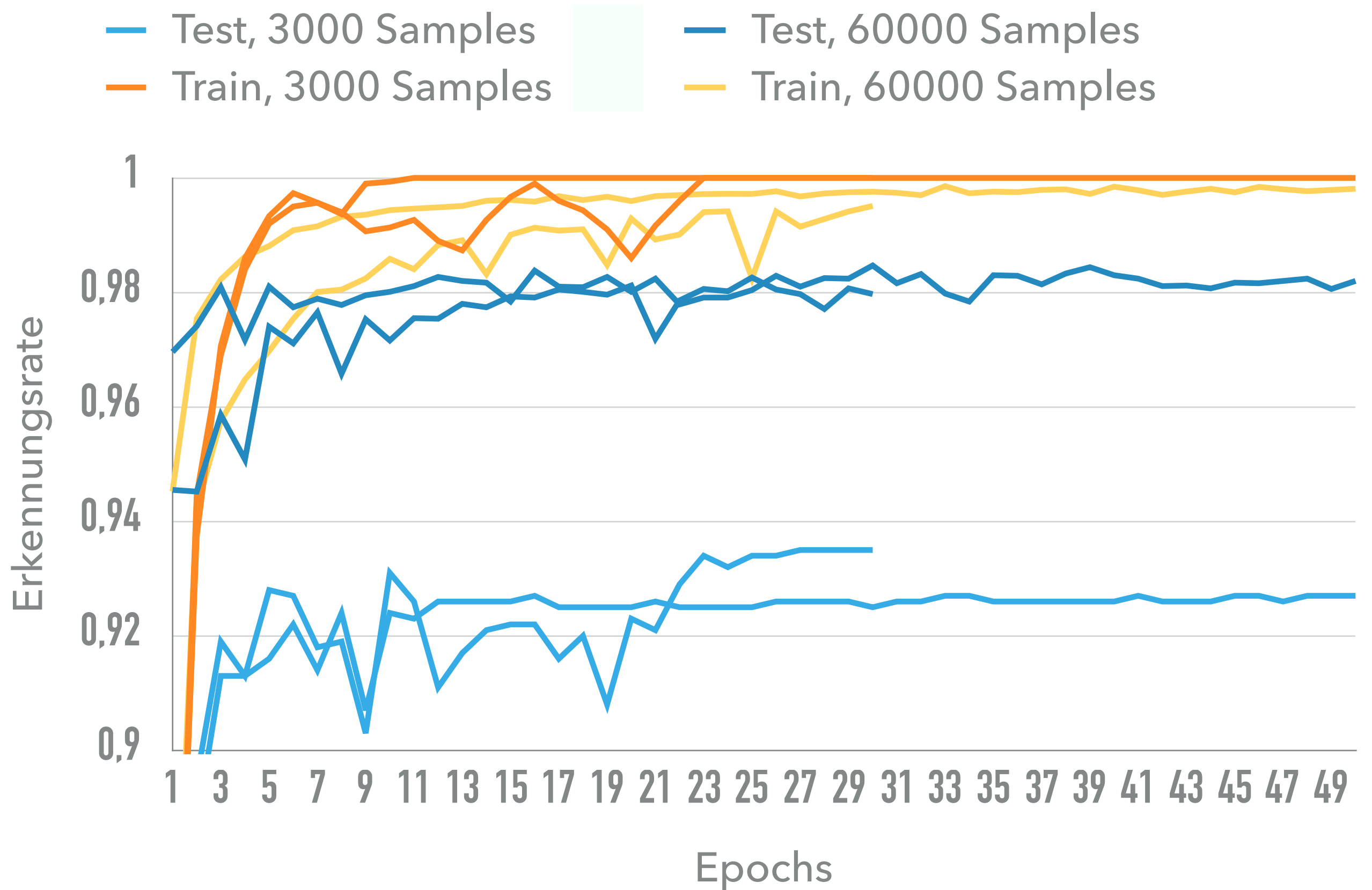
3. NN – 3000 TRAINING, 1000 TEST, 30 EPOCHEN



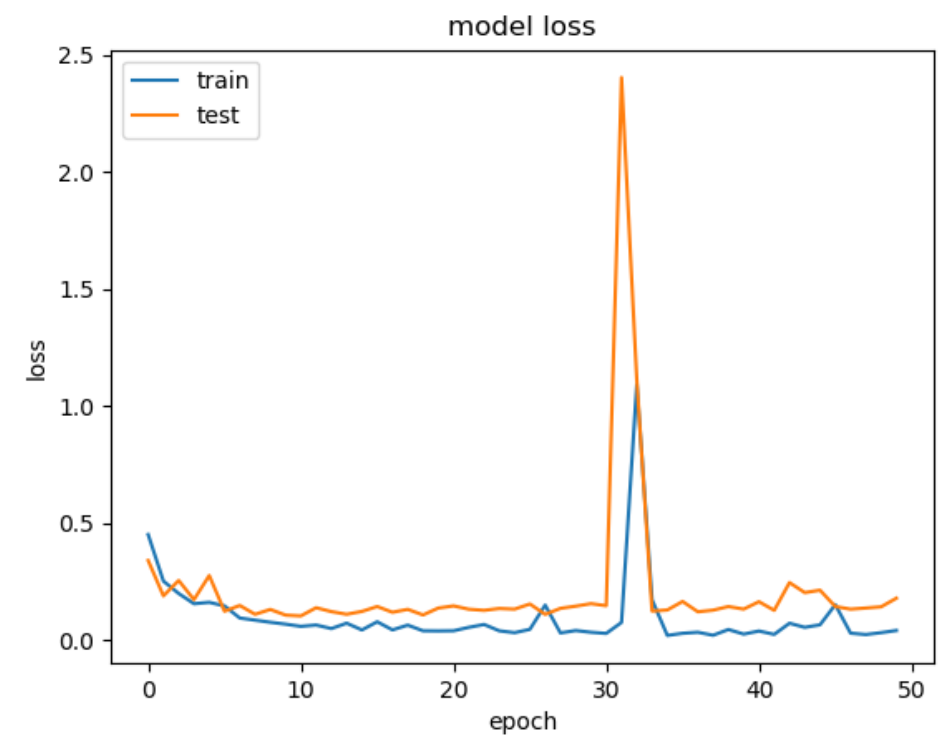
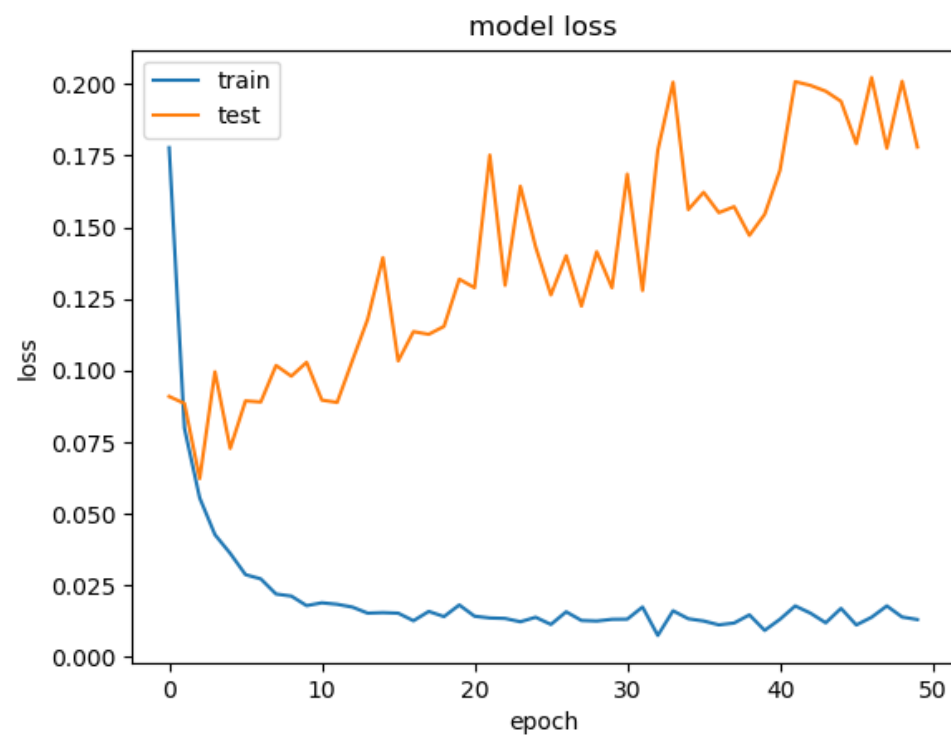
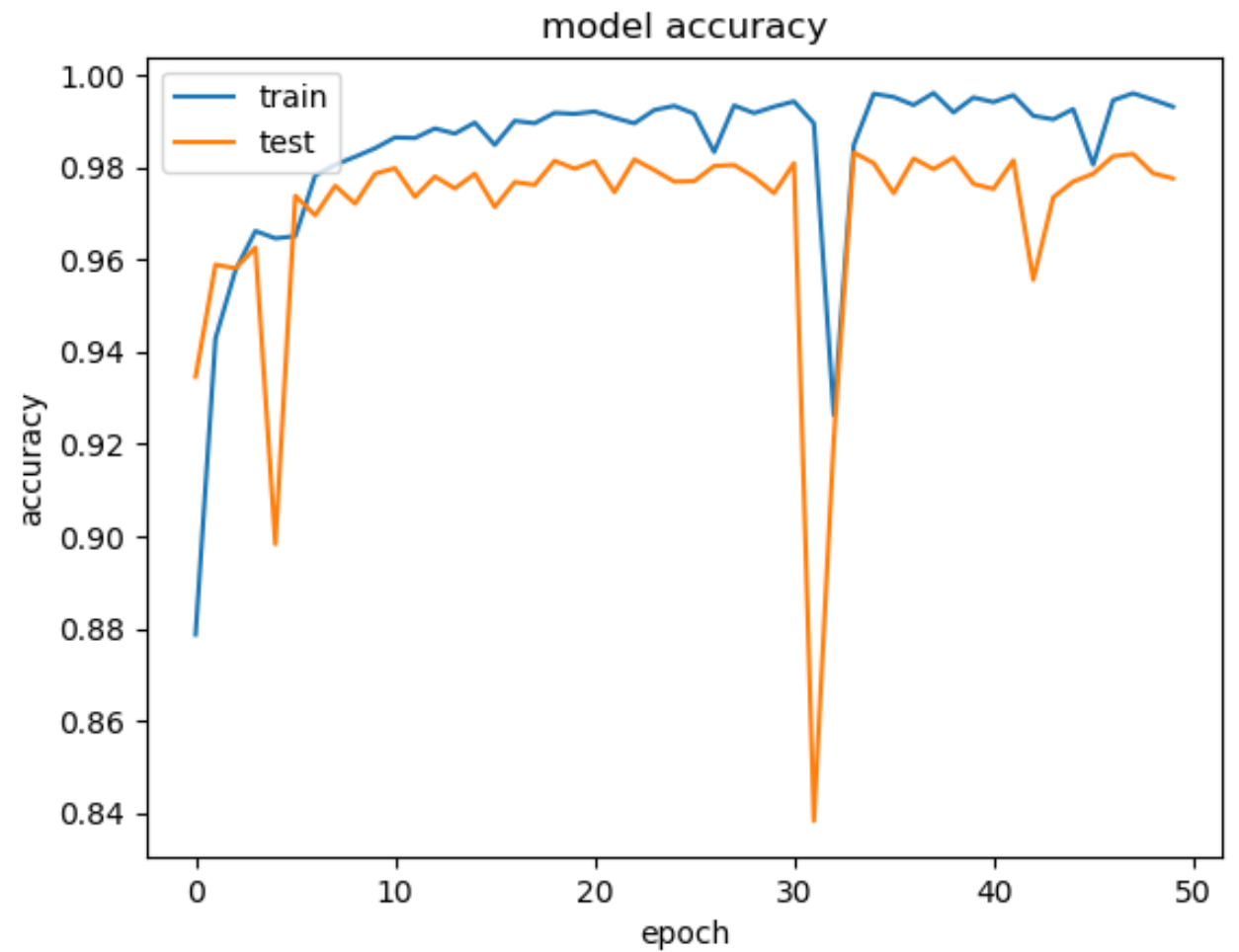
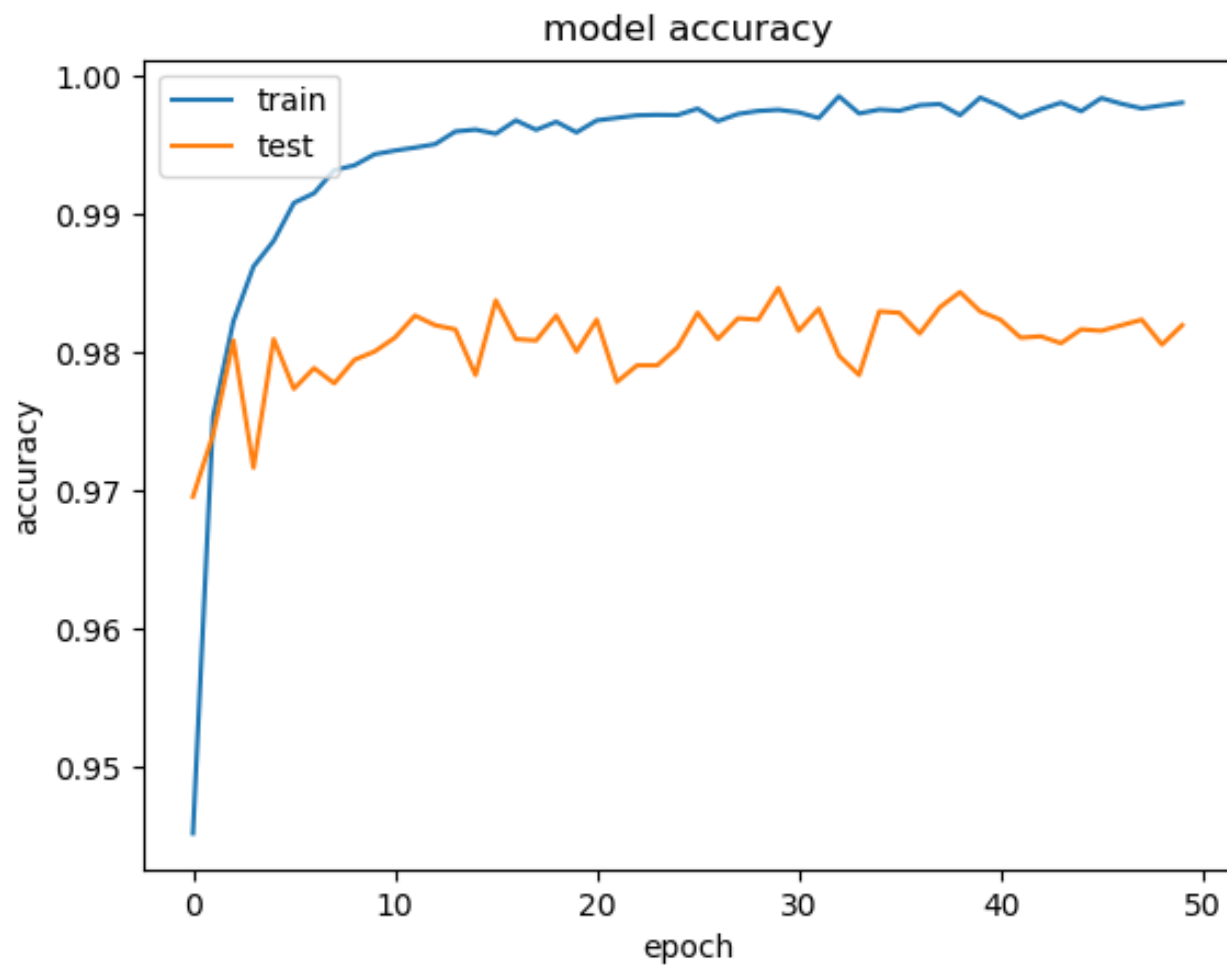
3. NN – 3000 TRAINING, 1000 TEST, 50 EPOCHEN



3. NN1 – 60000 / 3000 TRAINING, 10000 / 1000 TEST, 30 / 50 EPOCHEN



3. NN – 60000 TRAINING, 10000 TEST, NN1 VS NN5



3. NEURONALE NETZE – ERGEBNISSE

► Erkennungsraten der verschiedenen Architekturen

	3000 Trainingsdaten		60000 Trainingsdaten	
	30 Epochen	50 Epochen	30 Epochen*	50 Epochen*
NN1	93,5 %	92,7 %	98,1 %	98,2 %
NN2	91,5 %	93,0 %	-	-
NN3	92,9 %	93,4 %	-	-
NN4	92,8 %	92,6 %	-	-
NN5	92,1 %	91,9 %	98,0 %	97,8 %

* 60000 Test Samples

3. NEURONALE NETZE – ERGEBNISSE

► Vergleich der Ergebnisse Paper vs. eigene Tests

ID	architecture (number of neurons in each layer)	test error for best validation [%]	Eigene Ergebnisse best test error [%]
1	1000, 500, 10	0.49	1,9*
2	1500, 1000, 500, 10	0.46	7,0
3	2000, 1500, 1000, 500, 10	0.41	6,6
4	2500, 2000, 1500, 1000, 500, 10	0.35	7,2
5	9 × 1000, 10	0.44	2,0*

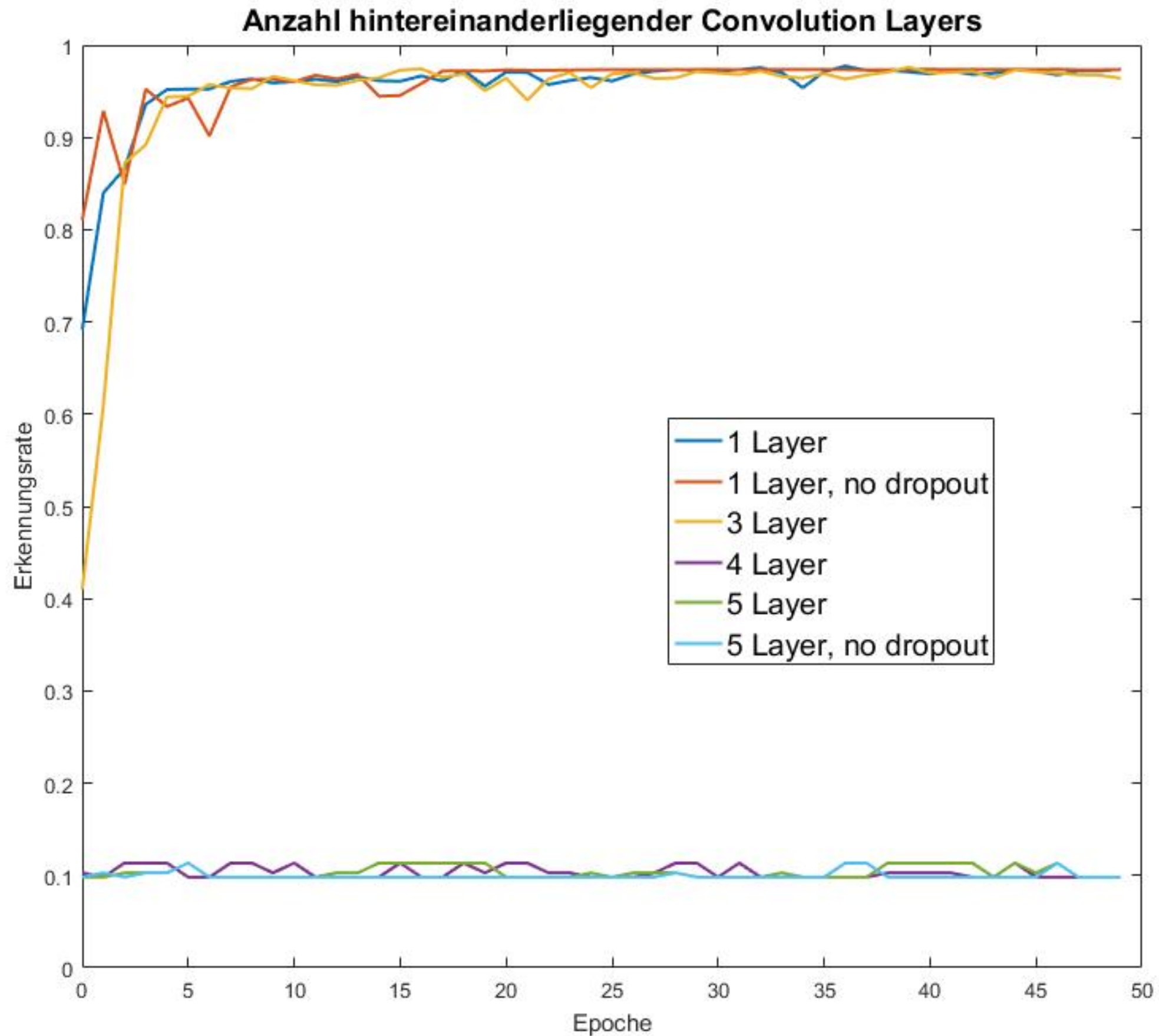
* 60000 Test Samples

4. CONVOLUTIONAL NEURAL NETWORKS

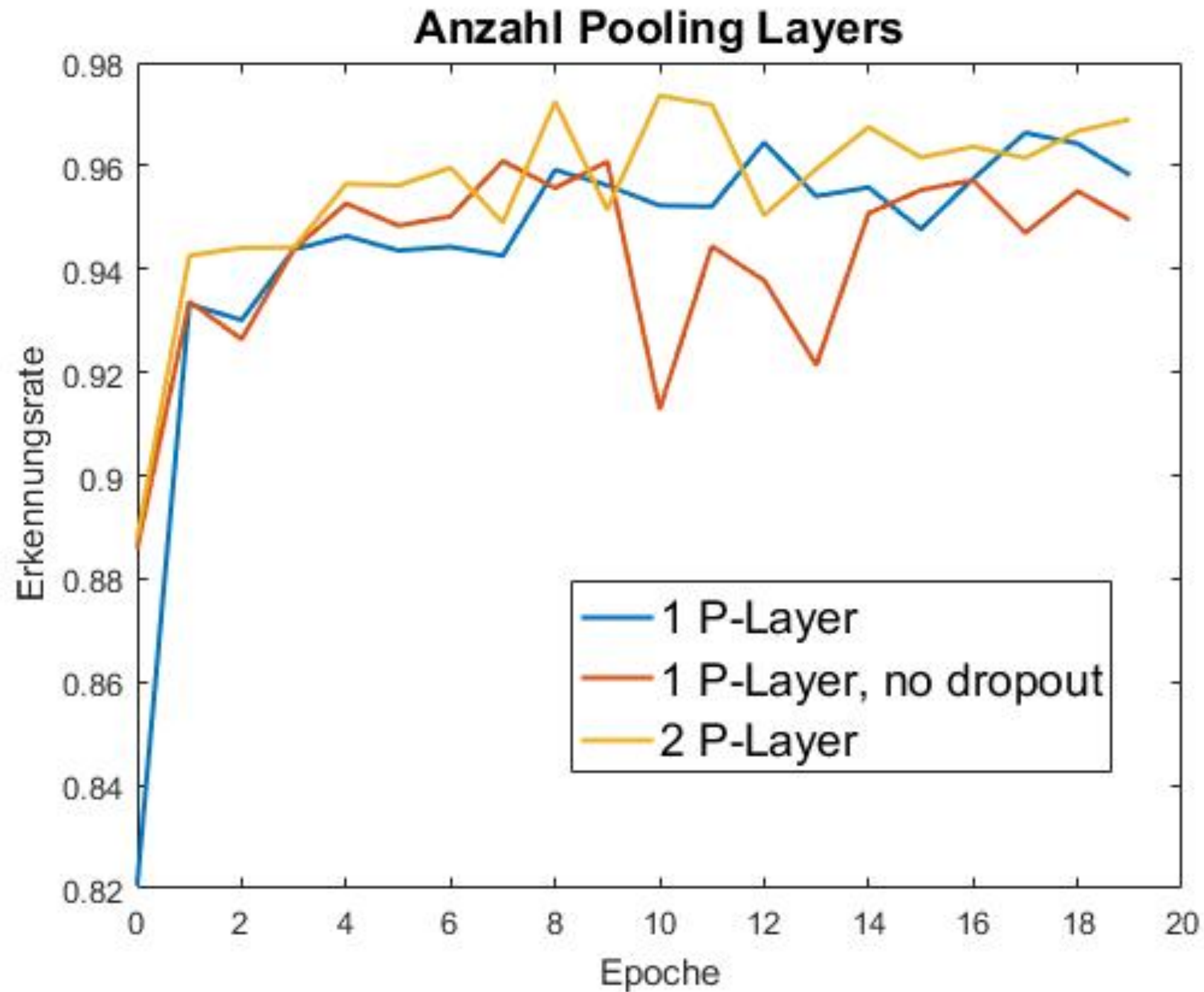
Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 28, 28, 32)	320
conv2d_2 (Conv2D)	(None, 28, 28, 32)	9248
max_pooling2d_1(MaxPooling2	(None, 14, 14, 32)	0
dropout_1 (Dropout)	(None, 14, 14, 32)	0
conv2d_3 (Conv2D)	(None, 14, 14, 64)	18496
conv2d_4 (Conv2D)	(None, 14, 14, 64)	36928
max_pooling2d_2 (MaxPooling2	(None, 7, 7, 64)	0
dropout_2 (Dropout)	(None, 7, 7, 64)	0
flatten_1 (Flatten)	(None, 3136)	0
dense_1 (Dense)	(None, 1024)	3212288
dense_2 (Dense)	(None, 1024)	1049600
dense_3 (Dense)	(None, 1024)	1049600
dense_4 (Dense)	(None, 1024)	1049600
dense_5 (Dense)	(None, 10)	10250
Total params: 6,436,330 (trainable)		

- Vergleich verschiedener Architekturen
- Anzahl convolutional layers
- Anzahl Pooling Layers
- Poolsize
- Trainingssamples: 2500

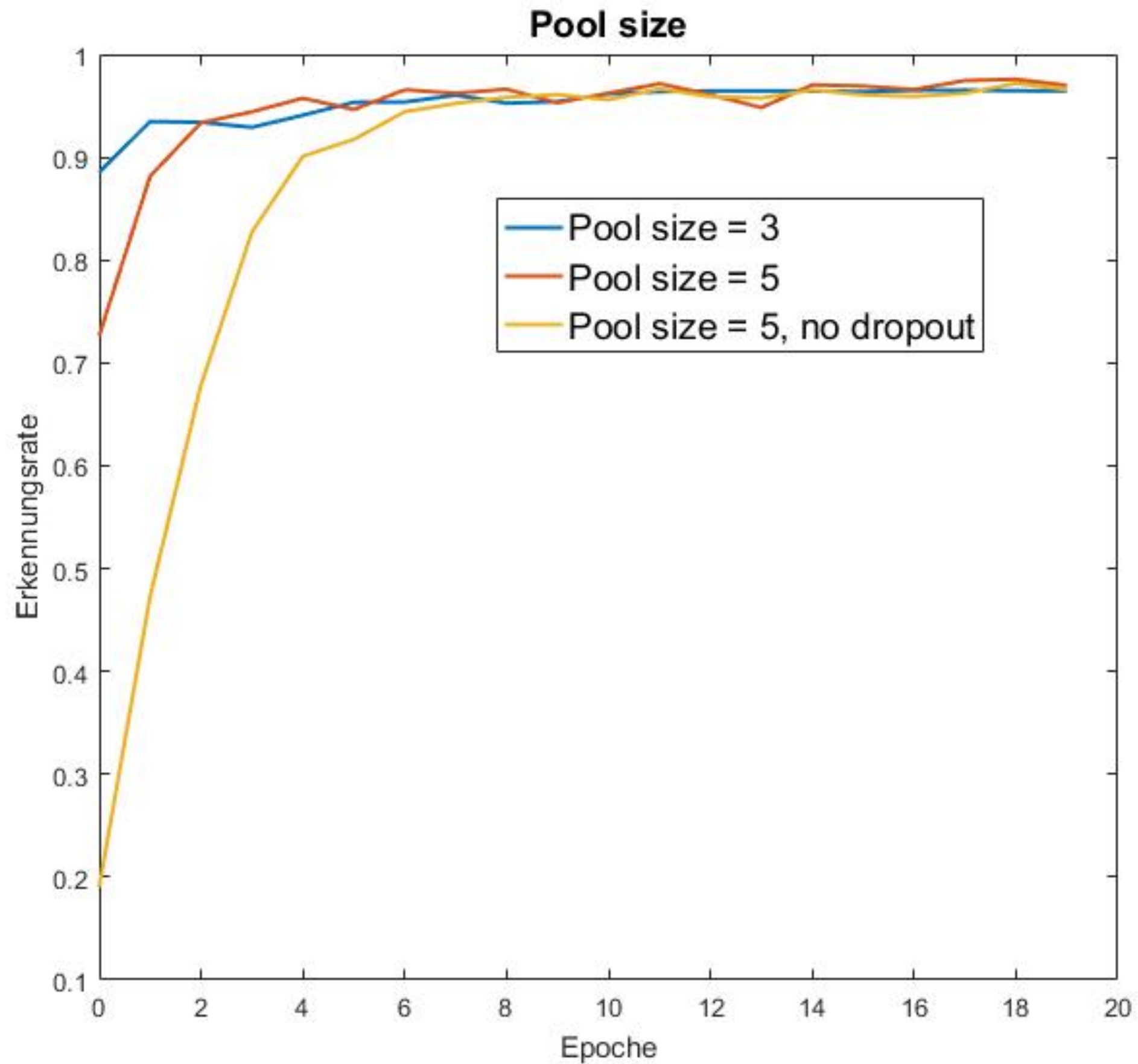
4. CONVOLUTIONAL NEURAL NETWORKS – ERGEBNISSE



4. CONVOLUTIONAL NEURAL NETWORKS – ERGEBNISSE



4. CONVOLUTIONAL NEURAL NETWORKS – ERGEBNISSE



4. VERGLEICH VON SVM, NN UND CNN

▶ Beste Erkennungsraten:

- ▶ SVM: 98,1 % (Polynom 2. Grades, 60000 Trainingsdaten)
- ▶ NN: 98,2 % (NN1, 50 Epochen, 60000 Trainingsdaten)
- ▶ CNN: 97,4 % (20 Epochen, 2500 Trainingsdaten)

▶ Beste Netze (Papers):

- ▶ SVM: 99,4 % (Virtual SVM, deg-9 poly, 2-pixel jittered)
- ▶ NN: 99,65 % (6-layer, 784-2500-2000-1500-1000-500-10 (on GPU) [elastic distortions])
- ▶ CNN: 99,7 % (35 conv. net, 1-20-P-40-P-150-10 [elastic distortions])

4. VERGLEICH VON SVM, NN UND CNN

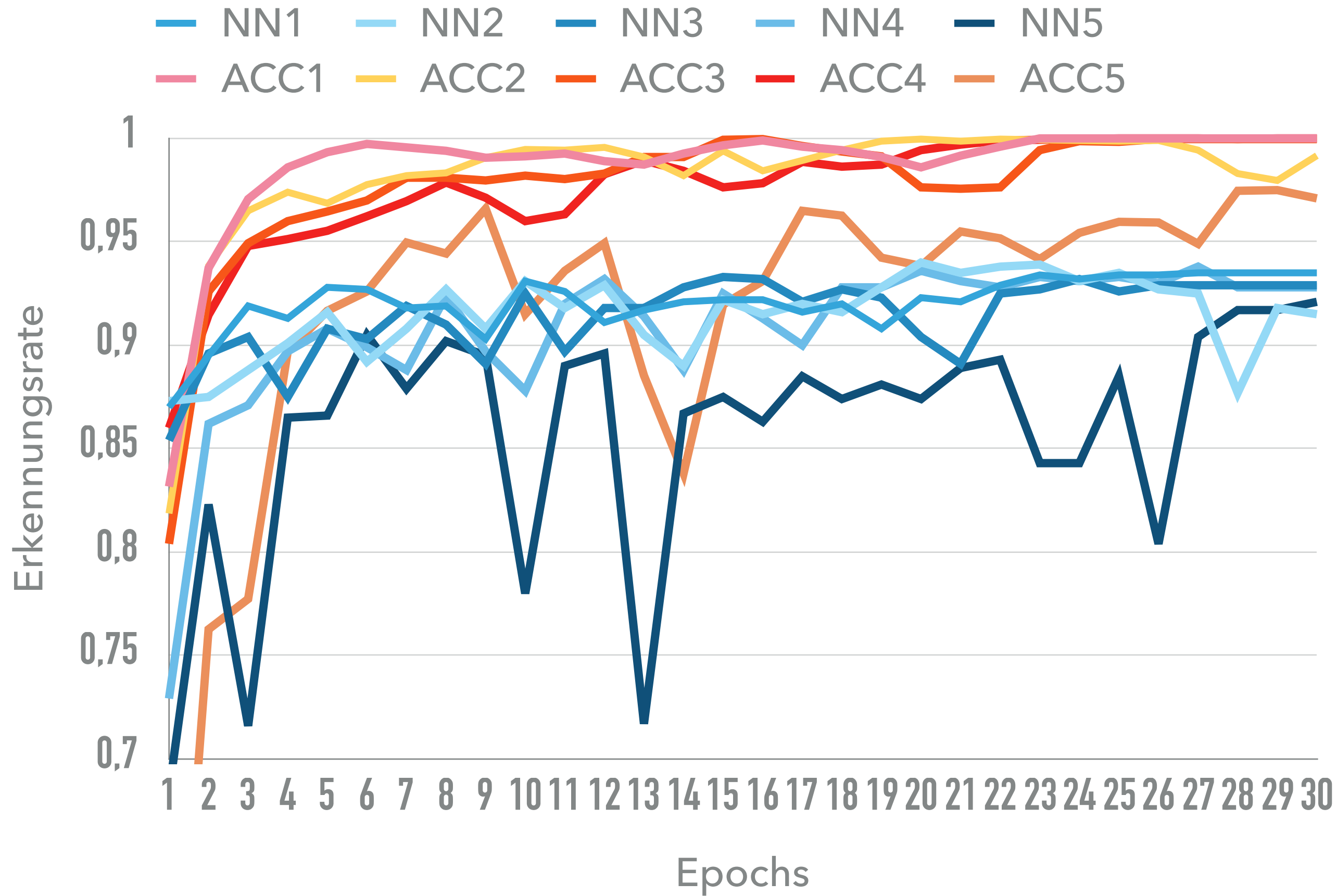
- ▶ Support Vector Machines:
 - ▶ Erkennungsrate variiert stark mit verwendeten Kernelparametern
 - ▶ Einzigartiges Ergebnis, weniger anfällig für Overfitting
 - ▶ Schnell, wenige Hyperparameter
- ▶ Neuronale Netze:
 - ▶ Klassischer Ansatz für Klassifizierung von verschiedenen Datenstrukturen
 - ▶ Flexible Architektur
- ▶ CNN:
 - ▶ Viele Architekturen liefern Prediction Modelle mit hohen Erkennungsraten
 - ▶ Besonders geeignet für Bilddaten

VIELEN DANK.

Andreas Hallmann - s70261@beuth-hochschule.de

Katharina Krebs - s67624@beuth-hochschule.de

3. NN – 3000 TRAINING, 1000 TEST, 30 EPOCHEN



3. NN – 3000 TRAINING, 1000 TEST, 50 EPOCHEN

