Formalities

- Presentation of 45 min followed by discussions
- Discussion with tutors one week in advance
- Assignment of a **nice** jupyter notebook with an code example and comments (text boxes) explaining the theory: at the day of presentation
- Distribution to fellow students
- Potential dates: 16.05, 23.05, 06.06, 13.06, 20.06, 27.06
- To get started with Python / Jupyter a suggestion is Anaconda: Link

The perceptron

- Learning vs. fitting: model complexity
- McCulloch-Pitts-Neuron
- Gradient descent: convergence?
- Classification task, linear separability, XOR-problem
- General approximation theorem
- Storage capacity
- Inference: Bayes' theorem
- Soft and hard classifiers
- Linear- and logistic regression

[Metha Chap. 1-7], [MacKay Chap. 39-41], [CKS Chap. 1]

Link to example code

Deep and convolutional NN

- Supervised learning
- Feed-forward algorithm
- Back-propagation algorithm
- Bias-variance trade-off
- Classification task
  - Image: (28x28) matrix with values $\in [0, 1]$
  - Label: integer $\in [1, 9]$
  - Fit function mapping images to labels
- Lit: [Metha]; [GF]; [Bishop]

Python package: Keras

MNIST: Link to documentation

Fashion MNIST: Link to Colab Notebook

Hopfield model

- Hebbian learning
- Storage capacity
- Attractors
- Stochastic dynamics $T > 0$
- Similar to spin glasses
- Mean field and replica theories
- Lit: [CKS Chap. 21]; [MacKay Chap. 42]; [Orhan]; [Mehlig Sec. I]

Easy to implement from scratch
(Traveling salesman problem (and simulated annealing): Link)

Restricted Boltzmann machine

- Unsupervised learning technique
- Max-Ent-Models: statistical physiscs
- Generative: learns PDF from data, that can be sampled
- Hidden variables: Hubbard-Stratonovich transformation from field theories
- Higher order correlations
- Lit: [Mehta]; [MacKay]

Python package PyTorch:

Link to example 1

Link to example 2

Reinforcement learning

- Markov-Decision-Process
- Temporal difference learning
- Tabular solution methods
- Q-Learning
- Exploration vs exploitation
- Lit: [Plaat Chapt. 3]; [Sutton]

Python package Gymnasium: Link
Example in Keras Lerning to Play Atari Game Breakout: Link
Learning to play (taxi world): Link

Applications in Soft Matter?

E. Boattini et al. *Unsupervised learning for local structure detection in colloidal systems* JCP, 2019
Talk with overview on this paper: Soft matter and machine learning by Laura Filion Link

- Auto-encoder to detect different crystalline orders
- Fit a Gaussian Mixture Model to Steinhardt-order-parameter distribution

Bibliography - link collection

[CKS] Coolen, Kühn, Sollich; *Theory of Neural Information Processing Systems* Link zu KonSearch, als PDF von uns zu haben

[MacKay] David MacKay; *Information Theory, Inference, and Learning Algorithms* Link

[Mehta] P. Mehta et al.; *A high-bias, low-variance introduction to Machine Learning for physicists* doi:10.1016/j.physrep.2019.03.001

[GF] Goodfellow: *Deep learning* Link

[Bishop] Christoper M. Bishop: *Pattern Recognition and Machine Learning*

[Orhan] Emin Orhan: *The Hopfield Model* Link

[Mehlig] Bernhard Mehlig: *Machine learning with neural networks* Link

[Plaat] Aske Plaat: *Learning to play* Link

[Sutton] Sutton, Barto: *Reinforcement learning: An Introduction* Link