VO Machine Learning 2023

This course aims to give the students the resources to understand fundamental topics in machine learning and the most common algorithms. The schedule can be divided into four parts:

1) *Introduction (1 day)*
2) *Parametric Machine Learning (4+2 days)*
    a) *Supervised Machine Learning*
    b) *Unsupervised Machine Learning*
3) *Nonparametric Machine Learning (1 day)*
4) *Reinforcement Learning (3 days)*

*-- **Note:** The students of VU Data Analysis II: Machine Learning for Data Analysis will skip the reinforcement learning part.*

Detailed Schedule

The following schedule is tentative. There may be some variation of content/topics.

1. 07.03.2023 **Introduction + Basic Linear algebra + Basic of Statistics**
   In this lecture, we introduce the content of the course. We start with a preliminary definition of supervised, unsupervised, and RL problems. We present some linear algebra and statistics.
   - [ ] Problem definition (what is machine learning)
   - [ ] Supervised (regression and classification), Unsupervised (DR, clustering), Reinforcement Learning
   - [ ] Linear algebra
     - [ ] Matrix multiplication, Linear dependency, Rank, Inversion, Determinant *(important for understanding eigenvalues and eigenvectors)*, Eigenspace *(important for PCA)*
2. 8.03.2023 **PS Introduction (exercises)**
3. 14.03.2023 **Statistical Estimators**
   In this lecture, we see that in (statistical) ML, most of the problems can be seen as estimation problems. We focus on the bias and variance of estimators, which will be translated later in the course into overfitting and generalisation. We will see how even a simple estimator of the standard deviation can be biased. We present maximum likelihood and maximum a posteriori estimation.
   - [ ] What is an estimator?
   - [ ] Bias and variance of statistical estimators.
   - [ ] An example of a biased estimator
   - [ ] Maximum likelihood

- [ ] Maximum a posteriori
- [ ] Derive MSE

4. 15.03.2023 **PS Introduction (exercises)**
5. 21.03.2023 **Linear Regression and Logistic Regression**
   This lecture introduces three parameter estimation techniques: maximum likelihood, maximum a posteriori, and Bayesian. We see how the mean-squared error arises from maximum likelihood with the Gaussian assumption. We will discuss linear regression, ridge regression and Logistic Regression.
   - [ ] A closed-form solution for linear regression
   - [ ] Ridge regression
   - [ ] Logistic Regression **[PS]**
6. 22.03.2023 **PS Linear Regression (exercises)**
7. 28.03.2023 **Regression and Classification with Neural network**
   In this lecture, we discuss the limits of linear models and introduce the concept of neural networks. But how can we efficiently train neural networks? GD, SGD, and backprop.
   - [ ] Why and what is a neural network
   - [ ] Classification and regression with neural networks
   - [ ] How to train a neural network (stochastic gradient descent) **[PS]**
   - [ ] (Backpropagation)
8. 29.03.2023 **PS Logistic Regression (exercises)**
9. 18.04.2023 **Overfitting and Generalization**
   Until now, we have seen the regression and classification, but we don't have any "understanding" of how well our parametric model performs. Small models (like linear) generally exhibit high bias and low variance, while larger models exhibit larger variance and lower bias. We see that the mean-square error can be decomposed in bias and variance terms, and to obtain a low MSE, we need to have a good bias-variance tradeoff. The bias/variance can be controlled by the model complexity but also with other techniques, like regularization (e.g., ridge regression), cross-validation, but also model ensembles (they reduce variance), or boosting (which reduces bias).
   - [ ] Overfitting and generalisation
     - [ ] Bias/variance tradeoff (MSE decomposition)
   - [ ] How to obtain a good bias/variance tradeoff?
     - [ ] Regularisation, cross-validation
     - [ ] Train, validation, and test sets
     - [ ] Model ensembles (*Useful later for random forests*)
       - [ ] (average of large regressors - reduces the variance)
     - [ ] Boosting
       - [ ] (sum of weak regressors - reduces the bias)
10. 19.04.2023 **PS Neural Networks (exercises)**
11. 25.04.2023 **Unsupervised ML I**
    At this point, we terminate supervised ML and start unsupervised learning. We start with the problem formalization of unsupervised machine learning. And introduce one of the

most common techniques for dimensionality reduction: (Probabilistic) Principal Component Analysis.

- [ ] Problem definition
- [ ] PCA **[PS]**

12. 26.04.2023 **PS PCA (exercises)**
13. 02.05.2023 **Unsupervised ML II**
    Clustering. Problem definition and techniques
    - [ ] Naive K-means **[PS]**
    - [ ] Fuzzy K-means **[PS]**
    - [ ] Hierarchical Clustering **[PS]**
14. 03.05.2023 **PS Clustering (exercises)**
15. 09.05.2023 **Unsupervised ML III**
    The Gaussian (or linear, if you want) model introduced in Unsupervised ML I are often too simple and cannot deal with complex data. A mixture of Gaussian can achieve higher model complexity and perform clustering too. We conclude this lecture by discussing the advantage of having generative models. We will also discuss the relationship between density estimation, clustering and dimensionality reduction.
    - [ ] GMM and Expectation Maximization **[PS]**
16. 10.05.2023 **PS GMM (exercises)**
17. 16.05.2023 **Nonparametric ML Density Estimation and Regression**
    At this point, we close with parametric ML and start non-parametric statistics. Nonparametric statistics can be seen as parametric statistics where we have a parameter for each sample; hence, the model complexity grows with the number of samples. Computationally, nonparametric techniques have low training time but high query time. We introduce nonparametric techniques, density estimation and regression.
    - [ ] Histogram
    - [ ] K-nearest neighbour density estimation
    - [ ] Simple kernel density estimation **[PS]**
    - [ ] Decision Trees **[PS]**
18. 17.05.2023 **PS Nonparametric (exercises)**
19. 23.05.2023 **Midterm PS + VU final for DiSC!**
20. 30.05.2023 **Reinforcement Learning I: Black-box Optimizers (Policy Search)**
    In this lecture, we briefly introduce the RL problem and “episodic policy gradient”, the most straightforward algorithm for policy improvement.
    - [ ] What is reinforcement learning
    - [ ] Introduction of Markov Decision Processes
    - [ ] Categorization of Reinforcement Learning
        - [ ] Episodic vs step-based
        - [ ] Value-based vs policy gradient vs actor-critic
        - [ ] Online vs offline
        - [ ] On-policy vs off-policy
        - [ ] Tabular vs function approximation

    - ☐ Episodic problem formalization (i.e., no MDP)
        - ☐ Finite difference
        - ☐ Parameter exploration

21. 31.05.2023 **PS Introduction to the Project (exercises)**
22. 06.06.2023 **Reinforcement Learning II: MDP and Policy Evaluation**
    In this lecture, we look more closely at the MDP formalization and introduce the concepts of the Value (and Q) function. We present (discounted) stationary state visitation (useful for policy gradient). We see how to estimate those values using Monte-Carlo and Temporal-Difference. We look at these concepts both with tabular and function approximation.
    - ☐ *The return*
    - ☐ *The value function and the Q-function*
    - ☐ *Monte-Carlo estimation: unbiased but large variance* **[PS]**
    - ☐ *Bellman Equation*
        - ☐ *Bellman operator, contraction property*
    - ☐ *Temporal Difference: biased but low variance* **[PS]**
23. 07.06.2023 **PS Decision Trees (exercises)**
24. 13.06.2023 **Reinforcement Learning III: Policy Improvement**
    In this lecture, we introduce the policy improvement theorem, leading to value-based algorithms like SARSA, Q-Learning, Fitted-Q Iteration, and Deep Q Networks.
    - ☐ *Optimality Bellman Equation and operator*
    - ☐ *Policy Improvement Theorem*
    - ☐ *SARSA, Q-Learning* **[PS]**
    - ☐ *A deep algorithm: DQN*
25. 14.06.2023 **PS Reinforcement Learning (exercises)**
26. 20.06.2023 **Exam simulation**
27. 21.06.2023 **PS Project Work & Questions (exercises)**
28. 27.06.2023 **Final Exam!**

Exams:

For CS students, the exams are divided into a midterm and a final exam. The students from the Minor Digital Science will have only one exam coincidental with the midterm of the CS.

23.05.2023 Midterm for CS, Final for MDS
27.06.2023 Final for CS.

The two exams are closed-book, and no electronic supports (e.g., calculators) are permitted. Theoretical questions and little practical exercises will form the exams. Each exam will score up to 50 points.

For MDS students, those points will sum up the exercises part (which will also take up to 50 points).
For CS students, the two exams will total 100 points.

Material:

For both CS and MDS students, we recommend using:

Title	**Pattern Recognition and Machine Learning**
Author	Christopher M. Bishop
Edition	illustrated
Publisher	Springer, 2006
ISBN	0387310738, 9780387310732

In addition, for the reinforcement learning part (only CS):

Title	**Reinforcement Learning, second edition: An Introduction**
Authors	Richard S. Sutton, Andrew G. Barto
Edition	illustrated
Publisher	MIT Press, 2018
ISBN	0262039249, 9780262039246

Furthermore, I will publish on OLAT both slides for the lecture and teaching material for at-home study.