Artificial Intelligence and Machine Learning (5171020)

<table>
<tr><td>Module name english</td><td colspan="6">Artificial Intelligence and Machine Learning</td></tr>
<tr><td>Type of module</td><td colspan="2">Pflichtmodul</td><td colspan="2">Responsible for module</td><td colspan="2">Prof. Dr. Frank-Michael Schleif</td></tr>
<tr><td>Lecturer</td><td colspan="6">Prof. Dr. Ivan Yamshchikov</td></tr>
<tr><td>Language of instruction, L. of examination</td><td colspan="2">Englisch</td><td colspan="2">Semester</td><td colspan="2">1</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Teaching and learning formats</td><td colspan="2">Seminaristischer Unterricht</td></tr>
<tr><td>ECTS-Credits</td><td colspan="2">5</td><td colspan="2">Type of examination</td><td colspan="2">Schriftliche Prüfung</td></tr>
<tr><td>Bonus benefits</td><td colspan="6"></td></tr>
<tr><td>Workload</td><td>Workload (Total)</td><td>150</td><td>Attendance time</td><td>60</td><td>Self-Study time (incl. exam preparation)</td><td>90</td></tr>
<tr><td>Duration of module</td><td colspan="2">1 Semester</td><td colspan="2">Frequency</td><td colspan="2">Sommersemester</td></tr>
<tr><td>Type of grading</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Artificial Intelligence</td></tr>
<tr><td>Conditions for participation</td><td colspan="6">None</td></tr>
<tr><td>Recommended prerequisites</td><td colspan="6"></td></tr>
<tr><td>Module’s learning outcomes</td><td colspan="6">Upon completion of the module students:
• knowing traditional AI techniques, how the evolved and how they are linked to current approaches
• understand basic types of problems to which machine learning algorithms can be applied and can compare them in terms of data that the algorithm expects to receive and the objectives they use for training
• have a general overview of key machine learning methods, understand their mechanism and major pros and cons, and can use these (relying on existing implementations) to solve typical learning problems by developing own pipelines and models
• can evaluate results of learning exercises and compare different methods in terms of their accuracy as well as computational efficiency and can report on these in oral as well as written form using appropriate tools for expert or more general audience (e.g. via Jupyter Notebooks)
• can follow and grasp formal description of standard machine learning algorithms and translate these into a working implementation in standard machine learning software
• can critically assess data analytical and machine learning exercises in terms of quality of the experimentation pipeline and the clarity and transparency of the experimental protocol</td></tr>
<tr><td>Module content</td><td colspan="6">• Introduction in Artificial Intelligence
- overview of the development of AI within the last few decades
- introduction into symbolic vs sub-symbolic concepts of AI
- classical AI methods (adatron, boltzman machine, hopfield network, cellular automata and alike)
- brief introduction to semantic knowledge representation with links to (fuzzy-) logic, ontologies

• Main concepts and principles of machine learning
- Basic types of machine learning (supervised/ unsupervised / reinforcement learning) and their use
- Main learning goals (prediction - regression/ classification, knowledge discovery – clustering / density estimation, etc.)
- Formalism of the learning problem
- Ethical and societal impacts of machine learning

• Foundations of learning from data
- Objective (loss) function
- Expected/ empirical risk
- Model complexity (over-/ under-fitting)
- Model training/ validation/ testing
- Model evaluation/ selection

• Selected key machine learning algorithms
- Linear models for regression/classification
- Regularization, ridge regression
- Variable selection, sparse models (lasso)
- Mixture models (k-means clustering, Gaussian mixtures)
- Non-parametric methods (kernels, trees, forests)
• Programming for machine learning
- Matlab / Python and packages (Numpy, Pandas, Sci-kit learn, Jupyter Notebooks, and other)</td></tr>
</table>

Literature	1. Bishop, Christopher M. Pattern Recognition and Machine Learning. Information Science and Statistics. New York: Springer, 2006. 2. Murphy, Kevin P. Machine Learning: A Probabilistic Perspective. Adaptive Computation and Machine Learning Series. Cambridge, MA: MIT Press, 2012. 3. Hastie, Trevor, Robert Tibshirani, and JeromeFriedman. The Elements of Statistical Learning. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001. 4. Russel, S, Norwig, P. Artificial Intelligence: A Modern Approach, Pearson, 2022

Artificial Neural Networks and Cognitive Models (5171030)

Module name english	Artificial Neural Networks and Cognitive Models				
Type of module	Pflichtmodul		**Responsible for module**	Prof. Dr. Magda Gregorová	
Lecturer	Prof. Dr. Magda Gregorová				
Language of instruction, L. of examination	Englisch		**Semester**	1	
SWS	4		**Teaching and learning formats**	Seminaristischer Unterricht	
ECTS-Credits	5		**Type of examination**	Portfolio	
Bonus benefits					
Workload	**Workload (Total)**	150	**Attendance time**	60	**Self-Study time (incl. exam preparation)** 90
Duration of module	1 Semester		**Frequency**	Sommersemester	
Type of grading	Differenzierte Note		**Verwendbarkeit**	Artificial Intelligence	
Conditions for participation	None				
Recommended prerequisites					
Module's learning outcomes	Upon completion of the module students: • can place artificial neural networks within the broader area of machine learning, understand their major advantages and disadvantages, and are aware of major applications of ANN as well as selected advanced models under research and their fundamental ideas • understand and assess the critical differences between the basic ANN architectures (MLP, CNN, RNN), can implement them in standard deep learning software packages, and can train, test, and evaluate the ANN models over real data • building on the experience of working with their own ANN implementations, can reuse publicly available implementations of more complex models to carry out experiments over real datasets, can compare the performance of these across various models and their hyperparameter setups • understand the importance of transparency and reproducibility in deep learning experimentation and can present in written as well as oral their learning and evaluation pipeline including relevant description of the selected software and hardware configuration • are aware of the ethical and societal impacts of machine learning and deep learning and can critically assess deep learning reports along these lines				
Module content	• Artificial neural networks (ANN) in machine learning (ML) - Basic concepts of learning algorithms and typical tasks - Model development workflow, hyperparameter tunning, performance measures and model selection - Ethical and societal aspects (open access, data governance, fairness, transparency, reproducibility, safety and robustness, interpretability and human oversight/trust, ecological footprint) • Basic ANN architectures - Multilayer perceptron (feed forward) - Convolutional neural networks - Recurrent neural networks • ANN model regularization - Norm penalties - Data augmentation - Early stopping - Dropout • ANN model optimization - (Stochastic) gradient descent - Backpropagation - Momentum methods - Learning rate scheduling • Major ANN applications and selected advanced models - Computer vision (object detection, image classification, style transfer) - Natural language processing (word2vec, BERT) - Autoencoders - Generative models • Deep learning software packages (one of these) - PyTorch - Tensorflow				
Literature	1. Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016 2. Zhang, Aston, Zachary C. Lipton, Mu Li, and Alexander J. Smola. Dive into Deep Learning. https://d2l.ai/, 2021				

Reasoning and Decision Making under Uncertainty (5171040)

<table>
<tr><td>Module name english</td><td colspan="6">Reasoning and Decision Making under Uncertainty</td></tr>
<tr><td>Type of module</td><td colspan="2">Pflichtmodul</td><td colspan="2">Responsible for module</td><td colspan="2">Prof. Dr. Frank Deinzer</td></tr>
<tr><td>Lecturer</td><td colspan="6">Prof. Dr. Frank Deinzer</td></tr>
<tr><td>Language of instruction, L. of examination</td><td colspan="2">Englisch</td><td colspan="2">Semester</td><td colspan="2">1</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Teaching and learning formats</td><td colspan="2">Seminaristischer Unterricht</td></tr>
<tr><td>ECTS-Credits</td><td colspan="2">5</td><td colspan="2">Type of examination</td><td colspan="2">Portfolio</td></tr>
<tr><td>Bonus benefits</td><td colspan="6"></td></tr>
<tr><td>Workload</td><td>Workload (Total)</td><td>150</td><td>Attendance time</td><td>60</td><td>Self-Study time (incl. exam preparation)</td><td>90</td></tr>
<tr><td>Duration of module</td><td colspan="2">1 Semester</td><td colspan="2">Frequency</td><td colspan="2">Sommersemester</td></tr>
<tr><td>Type of grading</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Artificial Intelligence</td></tr>
<tr><td>Conditions for participation</td><td colspan="6">Keine</td></tr>
<tr><td>Recommended prerequisites</td><td colspan="6"></td></tr>
<tr><td>Module’s learning outcomes</td><td colspan="6">- Students develop further knowledge and skills on the necessary mathematical foundations for understanding and developing algorithms for AI.
- Students can apply the principles of Reinforcement Learning algorithms
- Students can use the principles of modelling gents, environments and rewards.
- Students understand the necessity of function approximations in learning.
- Students understand the concepts of statiscal sensor fusion
- Students can realize sensor fusion applications
- Students build on their acquired knowledge to master learning problems.</td></tr>
<tr><td>Module content</td><td colspan="6">The course is composed of 2 thematic blocks.

Block A: Reinformcement Learning
1. Basic Reinforcement Learning Concepts
- Actions and States
- Goals, Rewards, Returns and Episodes
- Policies and Value Functions
2. Basic Reinforcement Learning Methods
- Finite Markov Decision Processes
- Dynamic Programming
- Monte Carlo Methods
3. Advanced tabular learning Methods
- Temporal-Difference Learning
- Bootstrapping Methods
4. Learning in Continuous State and Action Spaces
- On-Policy Approximation
- Value-function Approximation
- Off-Policy Approximation
- Approximate Eligibility Traces
5. Value Function Approximation Case Studies
- Computer Vision: Action planning
- Mastering Games: Backgammon, Go
6. Applications and Exercises

Block B: Sensor Fusion
1. Using Bayes for Sensor Data Fusion
- Modeling and Estimation of Densities
- Sensor Fusion over Time
2. Hidden Markov Models and Viterbi Algorithm
3. Recursive State Estimation
- Gaussian Filters
- Nonparametric Filters
4. Applications</td></tr>
</table>

Literature	1. Sutton, Barto. Reinforcement Learning - An Introduction. Bradford Books, 2018 2. Thorp. Beat the Dealer. Random House. 1966 3. Mitchell. Data Fusion: Concepts and Ideas. Springer. 2014 4. Thrun, Burgard, Fox: Probabilistic Robotics. MIT Press. 2005 5. Johnson, Freund, Miller. Miller & Freund's Probability and Statistics for Engineers. Pearson Further specialized literature will be announced in the course.

Mathematical Foundations of AI (5172010)

<table>
<tr><td>Module name english</td><td colspan="6">Mathematical Foundations of AI</td></tr>
<tr><td>Type of module</td><td colspan="2">Pflichtmodul</td><td colspan="2">Responsible for module</td><td colspan="2">Prof. Dr. Martin Storath</td></tr>
<tr><td>Lecturer</td><td colspan="6">Prof. Dr. Martin Storath</td></tr>
<tr><td>Language of instruction, L. of examination</td><td colspan="2">Englisch</td><td colspan="2">Semester</td><td colspan="2">1</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Teaching and learning formats</td><td colspan="2">Seminaristischer Unterricht</td></tr>
<tr><td>ECTS-Credits</td><td colspan="2">5</td><td colspan="2">Type of examination</td><td colspan="2">Schriftliche Prüfung</td></tr>
<tr><td>Bonus benefits</td><td colspan="6"></td></tr>
<tr><td>Workload</td><td>Workload (Total)</td><td>150</td><td>Attendance time</td><td>60</td><td>Self-Study time (incl. exam preparation)</td><td>90</td></tr>
<tr><td>Duration of module</td><td colspan="2">1 Semester</td><td colspan="2">Frequency</td><td colspan="2">Sommersemester</td></tr>
<tr><td>Type of grading</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Artificial Intelligence</td></tr>
<tr><td>Conditions for participation</td><td colspan="6">None</td></tr>
<tr><td>Recommended prerequisites</td><td colspan="6"></td></tr>
<tr><td>Module’s learning outcomes</td><td colspan="6">- Students refresh and develop further their knowledge and skills on the necessary mathematical foundations for understanding and developing algorithms for AI; in particular, linear algebra, calculus, probability.
- Students understand the principles of continuous optimization (constrained and unconstrained), are able to select appropriate approaches and they apply them for problems in AI.
- Students are able to apply and evaluate the principles of probabilistic modelling and inference, and they create probabilistic models for frequently occurring kinds of data.
- Students use the acquired mathematical skills to design and create frequently occurring building blocks of AI systems, such as linear regression, PCA, Gaussian mixture models and support vector machines.</td></tr>
<tr><td>Module content</td><td colspan="6">1. Advanced Vector Calculus
• Multivariate derivatives and chain rule
• Backpropagation and automatic differentiation
• Linearization and multivariate Taylor series
2. Advanced Linear Algebra
• Eigenvalues and eigenvectors
• Singular value decomposition
• Matrix approximation
3. Continuous Optimization
• Gradient descent
• Constrained optimization and Lagrange multipliers
• Convex Optimization
4. Models and Data
• Change of variables
• Empirical risk minimization
• Parameter estimation
• Probabilistic modelling and inference
• Model selection</td></tr>
<tr><td>Literature</td><td colspan="6">1. M. P. Deisenroth, A. A. Faisal, Cheng Soon Ong: Mathematics for Machine Learning, Cambridge University Press, 2020
2. C. M. Bishop: Pattern Recognition and Machine Learning, Springer, 2006
3. G. James, D. Witten, T. Hastie, R. Tibshirani: An Introduction to Statistical Learning, Second Edition, Springer, 2021</td></tr>
</table>

Project Module 1 (5172050)

Module name english	Project Module 1				
Type of module	Pflichtmodul		**Responsible for module**	Prof. Dr. Magda Gregorová	
Lecturer	Prof. Dr. Arndt Balzer, Prof. Dr. Magda Gregorová				
Language of instruction, L. of examination	Englisch		**Semester**	1	
SWS	4		**Teaching and learning formats**	Projekt	
ECTS-Credits	5		**Type of examination**	Projektarbeit	
Bonus benefits					
Workload	**Workload (Total)** 150		**Attendance time** 60	**Self-Study time (incl. exam preparation)**	90
Duration of module	1 Semester		**Frequency**	Sommersemester	
Type of grading	Differenzierte Note		**Verwendbarkeit**	Artificial Intelligence	
Conditions for participation	None				
Recommended prerequisites					
Module’s learning outcomes	Students can methodically process and solve comprehensive tasks. The students can develop and implement suitable solution strategies in a team. They know how team processes work and can assess how to contribute their own personality. The students can independently set up, implement, accompany and present a small AI project in a team. They can select and use appropriate development technologies and test and document their code.				
Module content	The students will work in groups to solve projects using AI techniques (supervised by at least one professor). The topics are provided by professors of the FIW, other faculties or external partners. In general the project will contain a software development (potentially accompanied by a technical solution) and a respective documentation or other form or presentation.				
Literature	1. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, A.Geron, O’Reilly, 2019 2. The Data Science Design Manual, S. Skiena, Springer, 2017 3. Deep Learning, I. Goodfellow, MIT Press, 2016 Further literature will be given based on the respective project tasks.				

Ausgewählte Kapitel der Embedded Systems (5071038)

<table>
<tr><td>Englischer Titel</td><td colspan="6">Selected Topics in Embedded Systems</td></tr>
<tr><td>Art des Moduls</td><td colspan="2">Wahlpflichtmodul</td><td colspan="2">Modulverantwortliche(r)</td><td colspan="2">Prof. Dr. Arndt Balzer</td></tr>
<tr><td>Dozent(in)</td><td colspan="6">Prof. Dr. Arndt Balzer</td></tr>
<tr><td>Sprache</td><td colspan="2">Deutsch/Englisch</td><td colspan="2">Studiensemester</td><td colspan="2">1,2</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Lehr- und Lernformen</td><td colspan="2">Seminar</td></tr>
<tr><td>ECTS-Punkte</td><td colspan="2">5</td><td colspan="2">Art der Prüfung</td><td colspan="2">Referat, Kolloquium</td></tr>
<tr><td>Bonusleistungen</td><td colspan="6"></td></tr>
<tr><td>Arbeitsaufwand</td><td>Gesamt</td><td>150</td><td>Präsenzzeit</td><td>60</td><td>Selbststudium</td><td>90</td></tr>
<tr><td>Dauer</td><td colspan="2">1 Semester</td><td colspan="2">Angeboten</td><td colspan="2">Wintersemester</td></tr>
<tr><td>Art der Note</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Informationssysteme, Artificial Intelligence, Digital Business Systems</td></tr>
<tr><td>Voraussetzungen nach SPO</td><td colspan="6">keine</td></tr>
<tr><td>Empfohlende Voraussetzungen</td><td colspan="6"></td></tr>
<tr><td>Lernergebnis des Moduls</td><td colspan="6">Die Studierenden sind in der Lage
- Notwendigkeit, Marktrelevanz und das Potential Eingebetteter (mobiler) Systeme zu bewerten,
- Herausforderungen bei Bau autonomen fahrender Systeme beurteilen und Lösung entwerfen zu können,
- Aufbau und Funktionsweise der Hard- und Software von Regelungssystemen am Beispiel eines Quadrokopters zu beschreiben, einschließlich der Echtzeitanforderungen,
- Teile der Systemsoftware zu implementieren,
- eingesetzte mathematische Methoden zu beurteilen,
- Ansätze zur Verbesserung der Signalverarbeitung zu entwerfen.</td></tr>
<tr><td>Inhalte des Moduls</td><td colspan="6">Die Inhalte der Lehrveranstaltung werden aktuellen Erfordernissen angepasst.

Seit 2020 ist der Schwerpunkt die Entwicklung von Software für ein autonom fahrendes Fahrzeug auf Basis von NVIDIA Hardware
Grundlagen des maschinellen Lernen, dabei u.a. künstliche neuronale Netze
Maschinelles Sehen, \\\"klassische\\\" Bildverarbeitung

Bis 2019 war der Schwerpunkt: Entwicklung von Software zur Steuerung eines Quadrokopters
Programmierung von Embedded Systems
Regelungstechnik, insbesondere PID Regler
Sensorik, Telemetrie
Mathematische Grundlagen: Kartesische und Polar Koordinaten, Euler Winkel, komplexe Zahlen, Quaterionen, Vektoralgebra
Signalverarbeitung: Zustandsschätzer, Bayes-, Gauss-, Kalman-Filter
Lageregelung, Yaw Regelung, Telekommandos

Bei Bedarf: Entwicklung von Software für MCU mit aktuellen IDEs, teil-autonomes Fahren</td></tr>
<tr><td>Literatur</td><td colspan="6">Tom M. Mitchell, Machine Learning, http://www.cs.cmu.edu/~tom/mlbook.html
Christopher M. Bishop, Pattern Recognition and Machine Learning, online
Trevor Hastie et al., The Elements of Statistical Learning, online
Kevin P. Murphy, Machine learning, online
S. Thrun, W. Burgard, D. Fox: Probabilistic Robotics, The MIT Press, 2005

Unterlagen der Uni Würzburg / Emqopter, 2019
A. Gelb, Applied Optimal Estimation, MIT Press, 1974
R. Kalman, A New Approach to Linear Filtering and Prediction Problems, Transaction of the ASME—Journal of Basic Engineering, 1960
P. Marwedel: Embedded System Design - Foundations of Cyber-Physical Systems, Springer, 2011
D. Gajski, F. Vahid: Specification and Design if Embedded Systems, Pearson, 2008
J. McClellan. R. Schafer: Signal Processing First, Pearson, 2003</td></tr>
</table>

Project Module (5171060)

Module name english	Project Module						
Type of module	Pflichtmodul		**Responsible for module**		Prof. Dr. Frank-Michael Schleif		
Lecturer	Prof. Dr. Frank-Michael Schleif						
Language of instruction, L. of examination	Englisch		**Semester**		1,2		
SWS	8		**Teaching and learning formats**		Projekt		
ECTS-Credits	10		**Type of examination**		Projektarbeit		
Bonus benefits							
Workload	**Workload (Total)**	300	**Attendance time**	120	**Self-Study time (incl. exam preparation)**	180	
Duration of module	2 Semester		**Frequency**		Jedes Semester		
Type of grading	Differenzierte Note		**Verwendbarkeit**		Artificial Intelligence		
Conditions for participation	None						
Recommended prerequisites							
Module's learning outcomes	Students can methodically process and solve comprehensive tasks. The students can develop and implement suitable solution strategies in a team. They know how team processes work and can assess how to contribute their own personality. The students can independently set up, implement, accompany and present a small AI project in a team. They can select and use appropriate development technologies and test and document their code.						
Module content	The students will work in groups to solve projects using AI techniques (supervised by at least one professor). The topics are provided by professors of the FIW, other faculties or external partners. In general the project will contain a software development (potentially accompanied by a technical solution) and a respective documentation or other form or presentation.						
Literature	1. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, A.Geron, O'Reilly, 2019 2. The Data Science Design Manual, S. Skiena, Springer, 2017 3. Deep Learning, I. Goodfellow, MIT Press, 2016 Further literature will be given based on the respective project tasks.						

Scientific seminar (5171110)

Module name english	Scientific seminar				
Type of module	Pflichtmodul	**Responsible for module**	Prof. Dr. Magda Gregorová		
Lecturer	Prof. Dr. Magda Gregorová, Hanna Usbeck-Frei				
Language of instruction, L. of examination	Englisch	**Semester**	1,2		
SWS	4	**Teaching and learning formats**	Seminar		
ECTS-Credits	5	**Type of examination**	Portfolio		
Bonus benefits					
Workload	**Workload (Total)** 150	**Attendance time**	60	**Self-Study time (incl. exam preparation)**	90
Duration of module	2 Semester	**Frequency**	Unregelmäßig		
Type of grading	Differenzierte Note	**Verwendbarkeit**	Artificial Intelligence		
Conditions for participation	None				
Recommended prerequisites					
Module’s learning outcomes	Upon completion of the seminar students: • can write English academic texts on AI topics taking into account the expected format, structure, and the target audience; can adapt the language and visual support accordingly (article vs. presentation, etc.). • understand the importance of good academic conduct, the boundaries and consequences of plagiarism, and the benefits of open science, transparency and reproducibility, they can design their communication strategy accordingly (open access / open source, experimental documentation, etc.) • can conduct relevant literature search, analyze the quality of texts, can create and maintain a relevant bibliography in standard software tools and correctly reference previous work in their academic outputs • are aware of selected recent trends in AI research and main opportunities and challenges in transferring them to practical applications • can critically analyse academic text and provide constructive feedback, can interact with senior researchers in an informed discussion				
Module content	Note: In summer semester 2023 exceptionally 2 SWS of the seminar will be offered. The remaining 2 SWS will be offered in winter semester 2023/24. Practical research and scientific work skills and principles of good scientific conduct. • Academic writing on AI topics in English (for non-native speakers) - Standard structure of academic texts – theses, technical reports, research articles, academic CV - Specific grammar features and word choices of English academic text and common pitfalls for non-native speakers - Good conduct in academic writing (citations, acknowledgments, plagiarism), open science, transparency, reproducibility - Literature review (dblp, google scholar, journals and conferences, predatory publishers) - Visual support of technical text (visual display of quantitative data, visual communication), academic presentations and poster design - Analysis of academic text, critical evaluation, peerreview process and principles • Academic and research support software tools The seminar will be enriched by a series of invited talks delivered by external academic researchers and/or AI practitioners. Through these the students will learn about: • Current trends and topics in AI research and applications - Transferability of theoretical research results to practical applications - Opportunities, open questions and challenges for AI research and applications (technical, societal, ethical, etc.) - Academic talk structure, audience targeting, academic exchange of knowledge and experience, constructive feedback and academic research discussion - Networking, establishing and fostering collaborations, formal/ informal interaction with senior researchers and practitioners				
Literature	To be defined in seminar				

Cloud Native (5171512)

Module name english	Cloud Native				
Type of module	Wahlpflichtmodul		**Responsible for module**	Prof. Dr. Frank-Michael Schleif	
Lecturer	Dr. Harald Philipp Gerhards				
Language of instruction, L. of examination	Englisch		**Semester**	1,2	
SWS	4		**Teaching and learning formats**	Seminar	
ECTS-Credits	5		**Type of examination**	Schriftliche Prüfung	
Bonus benefits					
Workload	**Workload (Total)**	150	**Attendance time**	60	**Self-Study time (incl. exam preparation)** 90
Duration of module	1 Semester		**Frequency**	Unregelmäßig	
Type of grading	Differenzierte Note		**Verwendbarkeit**	Artificial Intelligence	
Conditions for participation	none				
Recommended prerequisites					
Module's learning outcomes	Upon completion of the module, students will: • have an overview of the evolution of cloud computing and new architectures. • Be able to understand the architectural patterns of cloud native platforms and applications. • Be able to develop applications for container platforms on behalf of containerization principles. • Be able to understand vertical and horizontal scaling of applications. • Be able to maintain and configure monitoring and security components of Kubernetes platforms. • Be able to critically access approaches to versioning software artifacts and develop appropriate strategies for agile software projects. • Know the concepts of asynchronous communication using Apache Kafka. • Have solidified their knowledge on cloud native tools like Docker, Kubernetes, Helm, Apache Kafka and Git				

Module content	Main Concepts of Cloud Computing • Definition of \"cloud native\" • Historical background • Cloud Native and Open Source • Major players (CNCF, Linux Foundation, Apache Foundation) Cloud Native Architecture • Principles and paradigms • Distributed systems • Representation Concepts (C4, UML) Containerization & Virtualization Principles • Container vs. Virtual Machine • Emergence of Docker • Container Images • Image Build • Composing Containers Container Orchestration • Horizontal and vertical scaling • Kubernetes artifacts • Cluster Network • Persistence in Kubernetes • Templating for Kubernetes • Monitoring and Logging • Kubernetes Management • Service Mesh Pub-Sub-Messaging Concepts • Apache Kafka • Distributed logs • Stream processing Versioning • Commit strategies • Branching strategies Development Operation Principles • DevOps • DevSecOps • CI/CD • GitOps
Literature	Literature will be announced in the course.

Rust programming for safety-critical systems (5171513)

<table>
<tr><td>Module name english</td><td colspan="6">Rust programming for safety-critical systems</td></tr>
<tr><td>Type of module</td><td colspan="2">Wahlpflichtmodul</td><td colspan="2">Responsible for module</td><td colspan="2">Prof. Dr. Daniel Kulesz</td></tr>
<tr><td>Lecturer</td><td colspan="6">Prof. Dr. Sebastian Biedermann, Prof. Dr. Daniel Kulesz</td></tr>
<tr><td>Language of instruction, L. of examination</td><td colspan="2">Englisch</td><td colspan="2">Semester</td><td colspan="2">1,2</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Teaching and learning formats</td><td colspan="2">Seminar</td></tr>
<tr><td>ECTS-Credits</td><td colspan="2">5</td><td colspan="2">Type of examination</td><td colspan="2">Portfolio</td></tr>
<tr><td>Bonus benefits</td><td colspan="6"></td></tr>
<tr><td>Workload</td><td>Workload (Total)</td><td>150</td><td>Attendance time</td><td>60</td><td>Self-Study time (incl. exam preparation)</td><td>90</td></tr>
<tr><td>Duration of module</td><td colspan="2">1 Semester</td><td colspan="2">Frequency</td><td colspan="2">Unregelmäßig</td></tr>
<tr><td>Type of grading</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Informationssysteme, Artificial Intelligence, Digital Business Systems</td></tr>
<tr><td>Conditions for participation</td><td colspan="6">Für die praktischen Arbeiten sollten Studierende einen eigenen Rechner (Laptop) mit Windows, OS X, Linux oder *BSD mitbringen.</td></tr>
<tr><td>Recommended prerequisites</td><td colspan="6"></td></tr>
<tr><td>Module’s learning outcomes</td><td colspan="6">By successful completion of this course, students obtain the following skills:

- They internalized that programming in safety-critical domains is fundamentally different from programming in 'regular' domains.
- They understand how strict programming langauges can contribute to safe and secure programming.
- They can apply basic and advanced concepts of the Rust programming language in practical projects.
- They can build robust Rust applications for use in safety-critical domains.</td></tr>
<tr><td>Module content</td><td colspan="6">Malfunctions of software in safety-critical systems as well as cyberstrikes can lead to severe losses including death and environmental harm. Hence, when building software for such environments the use of safe and secure programming languages is essential. One suitable programming language for this use case is Rust. Moreover, Rust is also continuously gaining popularity and is used in leading open source projects such as the Linux kernel or the Firefox browser. Rust is particularly attractive because it enables both system-level and application-oriented programming while pursuing the goal of making programs safe and secure.

The first part of this course will start with an introduction to safety-critical systems. Afterwards, the basics of Rust (syntax,
concepts) will be explained and comparisons to other programming languages (e.g. Java or C/C++) will be drawn. Here, the focus will be on memory management without a garbage collector and its implications on safety and security.

In the second part of this course, the participants will deepen the theory through practical work on real development projects. The course follows the concept of 'research-based learning' and therefore requires an adequate degree of initiative and willingness to learn. In particular, we expect that students learn independently by means of designated tutorials.</td></tr>
<tr><td>Literature</td><td colspan="6">“Programming Rust: Fast, Safe Systems Development”, Jim Blandy, Jason Orendorff, Leonora Tindall, 2nd. ed, 2021, O'Reily
“Embedded software development for safety-critical systems”, Chris Hobbs, 2nd ed., 2020, CRC Press</td></tr>
</table>

Entrepreneurship for Engineers (5171514)

<table>
<tr><td>Module name english</td><td colspan="6">Entrepreneurship for Engineers</td></tr>
<tr><td>Type of module</td><td colspan="2">Wahlpflichtmodul</td><td colspan="2">Responsible for module</td><td colspan="2">Prof. Dr. Frank-Michael Schleif</td></tr>
<tr><td>Lecturer</td><td colspan="6">Prof. Dr. Ivan Yamshchikov</td></tr>
<tr><td>Language of instruction, L. of examination</td><td colspan="2">Englisch</td><td colspan="2">Semester</td><td colspan="2">1,2</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Teaching and learning formats</td><td colspan="2">Projekt</td></tr>
<tr><td>ECTS-Credits</td><td colspan="2">5</td><td colspan="2">Type of examination</td><td colspan="2">Projektarbeit</td></tr>
<tr><td>Bonus benefits</td><td colspan="6"></td></tr>
<tr><td>Workload</td><td>Workload (Total)</td><td>150</td><td>Attendance time</td><td>60</td><td>Self-Study time (incl. exam preparation)</td><td>90</td></tr>
<tr><td>Duration of module</td><td colspan="2">1 Semester</td><td colspan="2">Frequency</td><td colspan="2">Unregelmäßig</td></tr>
<tr><td>Type of grading</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Artificial Intelligence</td></tr>
<tr><td>Conditions for participation</td><td colspan="6">none</td></tr>
<tr><td>Recommended prerequisites</td><td colspan="6"></td></tr>
<tr><td>Module’s learning outcomes</td><td colspan="6">— Students learn how to apply the principles of technological entrepreneurship.
— Students can create a Minimal Viable Prototype (MVP) by applying principles of paper prototyping.
— Students can create and implement a customer development pipeline can evaluate product market fit and unit economics of the technological product.
— Students can create a pitch deck for their project from scratch, evaluate the quality of the early-stage venture capital, and implement a fund-raising plan.
— Students understand the overall properties of venture capital markets.</td></tr>
<tr><td>Module content</td><td colspan="6">1 What is venture capital?
— a brief history of venture investment
— probabilistic approach to venture investment
— venture capital and technological development
2 What is a product?
— Why is technology not a product?
— Paper prototyping and product market fit
— Customer development for engineers
3 What is a pitch deck?
— What are the key structural components of a good pitch?
— Unit economics
— Storytelling for engineers
4 How do you make decisions under stress?
— Managing small teams
— Trade-off between discipline and creativity
— Empathy for engineers</td></tr>
<tr><td>Literature</td><td colspan="6">B. Horowitz \"The Hard Thing About Hard Things: Building a Business When There Are No Easy Answers\"
P. Thiel \"Zero to One: Notes on Startups, or How to Build the Future\"

Optional literature:
M. Weber \\\\\\"Protestant Ethic and the Spirit of Capitalism\\\\\\"
K.F. Lee \\\\\\"AI Superpowers: China, Silicon Valley and the New Word Order\\\\\\"
B. Christian, T. Griffiths \\\\\\"Algorithms to Live By\\\\\\"</td></tr>
</table>

Trustworthy AI and AI regulations (5171070)

Module name english	Trustworthy AI and AI regulations				
Type of module	Pflichtmodul		**Responsible for module**	Prof. Dr. Oliver Ehret	
Lecturer	Prof. Dr. Oliver Ehret, Prof. Dr. Christian Kraus				
Language of instruction, L. of examination	Englisch		**Semester**	2	
SWS	4		**Teaching and learning formats**	Seminaristischer Unterricht	
ECTS-Credits	5		**Type of examination**	Schriftliche Prüfung	
Bonus benefits					
Workload	**Workload (Total)**	150	**Attendance time** 60	**Self-Study time (incl. exam preparation)**	90
Duration of module	1 Semester		**Frequency**	Wintersemester	
Type of grading	Differenzierte Note		**Verwendbarkeit**	Artificial Intelligence	
Conditions for participation	None				
Recommended prerequisites					
Module's learning outcomes	On successful completion of this module, the learner should be able to: - Understand the challenges of AI systems to existing law - Be able to place AI systems – from a legal standpoint - in civil and intellectual property law - Discuss AI-systems and the risks they are involving in self-driving cars - Outline the role of the selected principles in the context of AI - Evaluate the attempts of regulating AI within the EU to close possible legal gaps - Understand the ongoing measures to give AI systems a place in the legal system - Explain different ethical schools of thought and distinguish their lines of argumentation - Assess the challenges associated with technical innovations against the background of moral values - Evaluate selected applications and dilemmas and argue stringently				
Module content	1. Part Law 1.1. Introduction to law 1.2. AI systems and civil law, e.g. can AI act legally (e.g. by the vicarious agent or proxy) or creating a legal capacity of autonomous systems 1.3. Civil liability of AI systems 1.4. AI and intellectual property 2. Part Ethics 2.1. What is ethics? 2.2. Fairness and trust in AI systems 2.3. Responsibility and liability for AI systems 2.4. Risks of AI for companies 2.5. Human Enhancement 2.6. Autonomous vehicles 2.7. Military applications of AI				

Literature	Bartneck, Christoph, Christoph Lütge, Alan R. Wagner, und Sean Welsh. Ethik in KI und Robotik. München: Hanser, 2019. Coeckelbergh, Mark. AI ethics. The MIT press essential knowledge series. Cambridge, MA: The MIT Press, 2020. Darwall, Stephen L. Philosophical ethics. Dimensions of philosophy series. Boulder, Colo: Westview Press, 1998. European Commission High-level expert group on artificial intelligence, Hrsg. „Ethics guidelines for trustworthy AI“, 8. April 2019. https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai. Loh, Janina. Roboterethik: eine Einführung. Erste Auflage, Originalausgabe. suhrkamp taschenbuch wissenschaft 2277. Berlin: Suhrkamp, 2019. Lütge, Christoph, Hrsg. Handbook of the philosophical foundations of business ethics. Springer reference. Dordrecht?; New York: Springer, 2013. Simanowski, Roberto. Todesalgorithmus: das Dilemma der künstlichen Intelligenz. Deutsche Erstausgabe, 2., Durchgesehene Auflage. Passagen Thema. Wien: Passagen Verlag, 2021. Sparrow, Robert. „Robots and Respect: Assessing the Case Against Autonomous Weapon Systems“. Ethics & International Affairs 30, Nr. 1 (2016): 93–116. https://doi.org/10.1017/S0892679415000647. Taddeo, Mariarosaria, David McNeish, Alexander Blanchard, und Elizabeth Edgar. „Ethical Principles for Artificial Intelligence in National Defence“. Philosophy & Technology, 13. Oktober 2021. https://doi.org/10.1007/s13347-021-00482-3. Wallach, Wendell, und Colin Allen. Moral Machines: Teaching Robots Right from Wrong. First issued as an Oxford University Press paperback. New York, NY: Oxford University Press, 2010. Robbers, An Introduction to German Law, 7. Ed., 2019, Nomos. Barfield and Pagallo, Law and artificial intelligence, 2020, Edward Elgar Publishing Limited. Eidenmüller and Wagner, Law by algorithm, 2021, Mohr Siebeck Tübingen

Artificial Intelligence in Robotics (5171080)

<table>
<tr><td>Module name english</td><td colspan="6">Artificial Intelligence in Robotics</td></tr>
<tr><td>Type of module</td><td colspan="2">Pflichtmodul</td><td colspan="2">Responsible for module</td><td colspan="2">Prof. Dr. Pascal Meißner</td></tr>
<tr><td>Lecturer</td><td colspan="6">Prof. Dr. Pascal Meißner</td></tr>
<tr><td>Language of instruction, L. of examination</td><td colspan="2">Englisch</td><td colspan="2">Semester</td><td colspan="2">2</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Teaching and learning formats</td><td colspan="2">Seminaristischer Unterricht</td></tr>
<tr><td>ECTS-Credits</td><td colspan="2">5</td><td colspan="2">Type of examination</td><td colspan="2">Portfolio</td></tr>
<tr><td>Bonus benefits</td><td colspan="6"></td></tr>
<tr><td>Workload</td><td>Workload (Total)</td><td>150</td><td>Attendance time</td><td>60</td><td>Self-Study time (incl. exam preparation)</td><td>90</td></tr>
<tr><td>Duration of module</td><td colspan="2">1 Semester</td><td colspan="2">Frequency</td><td colspan="2">Wintersemester</td></tr>
<tr><td>Type of grading</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Artificial Intelligence</td></tr>
<tr><td>Conditions for participation</td><td colspan="6">None</td></tr>
<tr><td>Recommended prerequisites</td><td colspan="6"></td></tr>
<tr><td>Module’s learning outcomes</td><td colspan="6">By the end of the module students should be able to:
• Apply the Bayes (filter) formula and sample from probability density functions
• Determine and apply probabilistic sensor and motion models
• Discuss the steps and components of realizations of Bayes filters
• Implement realizations of Bayes filters and compute location estimates for robots
• Build and analyze grid maps
• Differentiate between localisation and SLAM systems as well as outline auxiliary techniques for SLAM solutions
• Assess and implement components of landmark- and grid-based solutions to the SLAM problem
• Differentiate between different path planning techniques and discuss the steps of collision avoidance solutions
• Apply and implement graph-search techniques for path planning
• Assess the Markov Decision Process definition as well as the concepts of Utility and Policy
• Apply dynamic programming on Markov Decision Problems to compute of value functions and optimal policies
• Differentiate between different Reinforcement Learning techniques</td></tr>
<tr><td>Module content</td><td colspan="6">01. Introduction – Nomenclature, history, state of the art, module logistics
02. Linear Algebra and Probability Primer – Vectors and operations, matrices and operations, axioms of probability, independent events, Bayes rule
03. Bayes Filter – Recursive Bayesian updating, state transitions, Markov property, derivation
04. Probabilistic Modeling – Odometry- and velocity-based motion models, beam- and scan- based sensor models
05. Localisation with Nonparametric Filters – Discrete Bayes filter, importance sampling, particle filter
06. Localisation with Gaussian Filters – Kalman filter, Extended Kalman filter
07. Mapping with Known Poses – Occupancy maps, reflection probability maps
08. Landmark-based SLAM – SLAM problem, EKF SLAM, loop closing, Rao-Blackwellization, FastSLAM
09. Grid-based SLAM – Scan matching, FastSLAM, improved proposals, selective resampling
10. Motion and Path Planning – Configuration space, combinatorial planning, graph-based search, collision avoidance
11. Markov Decision Processes – MDP definition, utility, value iteration, policy iteration
12. Reinforcement Learning – Temporal-difference learning, exploration vs exploitation, Q- learning, policy search</td></tr>
<tr><td>Literature</td><td colspan="6">• Probabilistic Robotics, Sebastian Thrun and Wolfram Burgard and Dieter Fox, MIT Press, 978-0262201629, 2005
• Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig, 4th ed. Prentice Hall, 978-0136042594, 2021</td></tr>
</table>

Semantic data processing and representation (5171090)

<table>
<tr><td>Module name english</td><td colspan="6">Semantic data processing and representation</td></tr>
<tr><td>Type of module</td><td colspan="2">Pflichtmodul</td><td colspan="2">Responsible for module</td><td colspan="2">Prof. Dr. Frank-Michael Schleif</td></tr>
<tr><td>Lecturer</td><td colspan="6">Dr. Sebastian Furth</td></tr>
<tr><td>Language of instruction, L. of examination</td><td colspan="2">Englisch</td><td colspan="2">Semester</td><td colspan="2">2</td></tr>
<tr><td>SWS</td><td colspan="2">4</td><td colspan="2">Teaching and learning formats</td><td colspan="2">Seminaristischer Unterricht</td></tr>
<tr><td>ECTS-Credits</td><td colspan="2">5</td><td colspan="2">Type of examination</td><td colspan="2">Portfolio</td></tr>
<tr><td>Bonus benefits</td><td colspan="6"></td></tr>
<tr><td>Workload</td><td>Workload (Total)</td><td>150</td><td>Attendance time</td><td>30</td><td>Self-Study time (incl. exam preparation)</td><td>120</td></tr>
<tr><td>Duration of module</td><td colspan="2">1 Semester</td><td colspan="2">Frequency</td><td colspan="2">Wintersemester</td></tr>
<tr><td>Type of grading</td><td colspan="2">Differenzierte Note</td><td colspan="2">Verwendbarkeit</td><td colspan="2">Artificial Intelligence</td></tr>
<tr><td>Conditions for participation</td><td colspan="6">None</td></tr>
<tr><td>Recommended prerequisites</td><td colspan="6"></td></tr>
<tr><td>Module’s learning outcomes</td><td colspan="6">After successfully completing the module:
• students are able to apply the basic methods of Natural Language Processing and related applications. The students are able to develop result-oriented applications that integrate Natural Language Processing methods. These methods can be based in whole or in part on various forms of artificial neural networks (deep neural networks).
• students are able to analyse concrete tasks in the field of natural language processing from applied science or industrial practice and evaluate and select suitable methods and software components from the field of natural language processing. In particular, students are also able to describe and develop suitable Deep Learning architectures.
• students are also able to describe, implement and present a corresponding overall software architecture. In doing so, they draw on common frameworks from the field of deep learning (e.g. KERAS, TensorFlow, PyTorch, etc.). They organise themselves and their team independently in the application of learned methods of Natural Language
Processing.</td></tr>
<tr><td>Module content</td><td colspan="6">Introduction
• Text and Speech Basics
o Morphological Analysis
o Lexical Representations
o Syntactic Representations
o Semantic Representations
o Discourse Representations
o Language Models
o Distributed Representations / Word Embeddings
• Natural Language Processing Applications
• Deep Learning for Natural Language Processing
o Convolutional Neural Networks and their Application to NLP
o Recurrent Neural Networks and their Application to NLP</td></tr>
<tr><td>Literature</td><td colspan="6">• Kamath, Uday, John Liu, and James Whitaker. Deep learning for NLP and speech recognition. Vol. 84. Cham: Springer, 2019.
• Chris Manning and Hinrich Schütze, Foundations of Statistical Natural Language Processing, MIT Press. Cambridge, MA: May 1999.</td></tr>
</table>

Learning of structured data (5171100)

Module name english	Learning of structured data				
Type of module	Pflichtmodul		**Responsible for module**	Prof. Dr. Frank-Michael Schleif	
Lecturer	Prof. Dr. Frank-Michael Schleif				
Language of instruction, L. of examination	Englisch		**Semester**	2	
SWS	4		**Teaching and learning formats**	Seminaristischer Unterricht	
ECTS-Credits	5		**Type of examination**	Portfolio	
Bonus benefits					
Workload	**Workload (Total)**	150	**Attendance time**	60	**Self-Study time (incl. exam preparation)** 90
Duration of module	1 Semester		**Frequency**	Wintersemester	
Type of grading	Differenzierte Note		**Verwendbarkeit**	Artificial Intelligence	
Conditions for participation	None				
Recommended prerequisites					
Module's learning outcomes	- being able to evaluate and to apply modelling techniques for non-standard data - being able to analyse non-vectorial data and to derive and improve predictive models - knowing how to evaluate and assess respective representation techniques - being able to implement pipelines for non-vectorial data analysis - learn the how-to of proximity based learning - learn how to assess, use and potentially extend the respective frameworks - Students know how to characterize, choose, evaluate, assess and construct practical tools for structured data analysis and respective application fields - learn how to use scientific literature and to understand, derive, implement and potentially extend the presented methods				
Module content	The module explains the generic analysis and processing of non-vectorial or structured data like graphs, trees, sequential data or alike. We discuss algebraic methods as well as neural network based techniques. The algorithmic part is shown in matlab, numpy/python or by use of other numerical frameworks. Exemplary the following key topics are addressed: - Particularities of non-vectorial, compositional and structured data - General proximity measures and implications on mathematical models - Mathematical concepts like information theoretic measures,non-euclidean spaces, local and global embedding approaches - Representation by proximity measures and simple learning methods - Particular algebraic and neural network based Embedding techniques - Evaluation methods for the representation of non-vectorial data - Exemplary implementations and applications				
Literature	- The Dissimilarity Representation for Structural Pattern Recognition, Pekalska & Duin, World Scientific, 2005 - Graph Classification And Clustering Based On Vector Space Embedding, Bunke et al., 2010 - Kernels For Structured Data, Gartner, 2008 - Recent publications on learning of structured data are provided / suggested during the lecture				

Master Thesis (5171130)

Englischer Titel	Master Thesis				
Art des Moduls	Pflichtmodul		**Modulverantwortliche(r)**	Prof. Dr. Frank-Michael Schleif	
Dozent(in)	Prof. Dr. Arndt Balzer, Prof. Dr. Peter Braun, Prof. Dr. Frank Deinzer, Prof. Dr. Frank-Michael Schleif, Prof. Dr. Magda Gregorová				
Sprache	Deutsch/Englisch		**Studiensemester**	3	
SWS	0		**Lehr- und Lernformen**	Undefiniert	
ECTS-Punkte	25		**Art der Prüfung**	Masterarbeit	
Bonusleistungen					
Arbeitsaufwand	**Gesamt**	750	**Präsenzzeit**	0	**Selbststudium** 750
Dauer	1 Semester		**Angeboten**	Jedes Semester	
Art der Note	Differenzierte Note		**Verwendbarkeit**	Artificial Intelligence	
Voraussetzungen nach SPO	50 ECTS points				
Empfohlende Voraussetzungen					
Lernergebnis des Moduls	With the submission of a Master's thesis and the successful assessment, students document that they have understood the teaching content of the previous semesters and are able to apply it to tasks independently and successfully. They are able to derive an innovative research question on a selected research area, which includes a sufficiently significant and as yet unresearched research field. They can work on this research question largely independently with an appropriate and meaningful research design and lead to an objectively comprehensible, reliable and valid result. The written result is at the level of international standards of scientific publications and, upon successful completion, demonstrates the competences in terms of connectivity in the direction of doctoral projects.				
Inhalte des Moduls	Independent preparation of a thesis and processing of a theoretical or practical task according to scientific methods.				
Literatur	Is provided based on the topic, but needs also to be identified by the student as part of the master thesis.				

Design and Analysis of Learning Problems (5171515)

Module name english	Design and Analysis of Learning Problems				
Type of module	Wahlpflichtmodul		**Responsible for module**	Prof. Dr. Frank-Michael Schleif	
Lecturer	Dr. Alex Gößmann				
Language of instruction, L. of examination	Englisch		**Semester**	3	
SWS	4		**Teaching and learning formats**	Seminar	
ECTS-Credits	5		**Type of examination**	Kolloquium	
Bonus benefits					
Workload	**Workload (Total)** 150		**Attendance time** 60		**Self-Study time (incl. exam preparation)** 90
Duration of module	1 Semester		**Frequency**	Sommersemester	
Type of grading	Differenzierte Note		**Verwendbarkeit**	Artificial Intelligence	
Conditions for participation	None				
Recommended prerequisites					
Module’s learning outcomes	1. Students develop a solid intuition about the statistical and numerical principles driving machine learning. Equipped with this intuition they will be able to independently design machine learning approaches and analyse them by classical methods. 2. Students understand the necessity and advantages of regularizing learning methods, based on simple but well understood examples in compressed sensing and sparse regression. 3. Students acquire a numerical understanding of the curse of dimensions, represented by tensors of large orders. They further get familiar with available methods to mitigate the curse of dimensions with carefully designed learning methods such as tensor network based regression. 4. Students get familiar with current approaches towards understanding the success of neural networks.				
Module content	Advanced linear regression: 1. Function spaces, scaler-products and norms 2. Squares risks and their geometrical interpretation 3. Kernel ridge regression and the representer theorem Sparse regression and compressed sensing: 1. l_0 and l_1-regularized learning problems and their algorithmic solutions 2. Compressed sensing and applications 3. Data properties enabling the success of sparse regression Success guarantees and complexities of regression problems: 1. Statistical foundation of learning by risk minimization 2. Complexities of learning architectures and success guarantees 3. Concentration inequalities and uniform concentration bounds Tensor regression: 1. Applications of tensors in machine learning 2. Dimensionality reduction with tensor networks 3. Fitting tensor networks to data Neural network regression: 1. Expressivity and concentration of neural networks 2. Advantages of deep against shallow networks 3. Uniform concentration bounds and Rademacher complexities Accompanying use cases: 1. Prediction of the stability of materials to be used in solar cells 2. Identification of sparse dynamical laws 3. Embedding of knowledge graphs for link predictions				
Literature	Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar: Foundations of Machine Learning, Second Edition. Cambridge, MA: MIT Press 2018 Roman Vershynin: High-Dimensional Probability, An Introduction with Applications in Data Science. Cambridge University Press 2018 Simon Foucart, Holger Rauhut: A Mathematical Introduction to Compressive Sensing. Springer Science & Business Media 2013				