



UNIVERSITÄT  
DES  
SAARLANDES

FAKULTÄT FÜR MATHEMATIK UND INFORMATIK

MODULHANDBUCH

# **Data Science and Artificial Intelligence BSc**

19. Dezember 2023

---

## **Liste der Modulbereiche und Module**

<b>1 Ringvorlesungen</b>	<b>3</b>
1.1 Perspektiven der Informatik . . . . .	4
<b>2 Grundlagen der Mathematik</b>	<b>5</b>
2.1 Mathematik für Informatiker 1 . . . . .	6
2.2 Mathematik für Informatiker 2 . . . . .	8
2.3 Mathematik für Informatiker 3 . . . . .	10
<b>3 Grundlagen der Informatik</b>	<b>12</b>
3.1 Big Data Engineering . . . . .	13
3.2 Grundzüge der Theoretischen Informatik . . . . .	16
3.3 Grundzüge von Algorithmen und Datenstrukturen . . . . .	18
3.4 Programmierung 1 . . . . .	19
3.5 Programmierung 2 . . . . .	21
<b>4 Spezialisierter Bereich DSAI</b>	<b>23</b>
4.1 Elements of Data Science and Artificial Intelligence . . . . .	24
4.2 Elements of Machine Learning . . . . .	25
4.3 Statistics Lab . . . . .	27
<b>5 Stammvorlesungen DSAI</b>	<b>29</b>
5.1 Artificial Intelligence . . . . .	30
5.2 Automated Reasoning . . . . .	32
5.3 Database Systems . . . . .	33
5.4 Image Processing and Computer Vision . . . . .	35
5.5 Information Retrieval and Data Mining . . . . .	37
5.6 Machine Learning . . . . .	38
5.7 Neural Networks: Theory and Implementation . . . . .	39
<b>6 Vertiefungsvorlesungen DSAI</b>	<b>40</b>
6.1 AI Planning . . . . .	41

6.2	Mathematische Statistik . . . . .	42
6.3	Security . . . . .	43
6.4	Stochastik I . . . . .	44
6.5	Stochastik II . . . . .	45
<b>7</b>	<b>Seminare DSAI</b>	<b>46</b>
7.1	Seminar . . . . .	47
<b>8</b>	<b>Projektseminare DSAI</b>	<b>49</b>
8.1	Projektseminar DSAI . . . . .	50
<b>9</b>	<b>Bachelor-Seminar und Arbeit</b>	<b>51</b>
9.1	Bachelor-Seminar . . . . .	52
9.2	Bachelor-Arbeit . . . . .	53

***Modulbereich 1***

---

***Ringvorlesungen***

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>1</b>	<b>6</b>	<b>jedes Wintersemester</b>	<b>1 Semester</b>	<b>2</b>	<b>2</b>

**Modulverantwortliche/r** Studiendekan der Fakultät Mathematik und Informatik  
Studienbeauftragter der Informatik

**Dozent/inn/en** Dozent/inn/en der Fachrichtung

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen** Nachweis über das inhaltliche Verständnis von mindestens drei Vorträgen, z.B. durch schriftliche Ausarbeitung oder Test.

**Lehrveranstaltungen / SWS** 2 SWS Vorlesung

**Arbeitsaufwand** 30 h Präsenzstudium  
+ 30 h Eigenstudium  
= 60 h (= 2 ECTS)

**Modulnote** Das Modul ist insgesamt bestanden, wenn die Prüfungsleistung bestanden wurde (unbenotet).

**Sprache** Deutsch / English

## Lernziele / Kompetenzen

Frühzeitige Motivierung und Überblick über die zentralen wissenschaftlichen Fragestellungen der Informatik, sowie über die Kompetenzen der Saarbrücker Informatik.

## Inhalt

Vorträge durch wöchentlich wechselnde Dozent/inn/en bieten einen Querschnitt durch die Forschungsthemen der Saarbrücker Informatik. Die Themen spannen einen attraktiven Bogen von aktuellster Forschung zu anspruchsvollen Problemen der industriellen Praxis.

## Literaturhinweise

Material wird passend zu den zu den einzelnen Vorträgen bereitgestellt.

## Weitere Informationen

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Perspectives in Computer Science*.

**Modulbereich 2**

---

***Grundlagen der Mathematik***

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>1</b>	<b>6</b>	<b>jedes Wintersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Joachim Weickert

**Dozent/inn/en** Prof. Dr. Joachim Weickert  
 Prof. Dr. Mark Groves  
 Prof. Dr. Henryk Zähle  
 Prof. Dr. Christian Bender

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen**

- Teilnahme an den Übungen und Bearbeitung der wöchentlichen Übungsaufgaben (50 Prozent der Übungspunkte werden zur Klausurteilnahme benötigt)
- Bestehen der Abschlussklausur oder der Nachklausur

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
 + 2 SWS Übung  
 = 6 SWS

**Arbeitsaufwand** 90 h Präsenzstudium  
 + 180 h Eigenstudium  
 = 270 h (= 9 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen und praktischen Aufgaben ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben.

**Sprache** Deutsch und Englisch

## Lernziele / Kompetenzen

- Erarbeitung von mathematischem Grundlagenwissen, das im Rahmen eines Informatik- bzw. Bioinformatikstudiums benötigt wird
- Fähigkeit zur Formalisierung und Abstraktion
- Befähigung zur Aneignung weiteren mathematischen Wissens mit Hilfe von Lehrbüchern

## Inhalt

Die Zahlen geben die Gesamtzahl der Doppelstunden an.

### DISKRETE MATHEMATIK UND EINDIMENSIONALE ANALYSIS

- A. Grundlagen der diskreten Mathematik (8)
1. Mengen (1)
  2. Logik (1)
  3. Beweisprinzipien, incl. vollst. Induktion (1)
  4. Relationen (1)
  5. Abbildungen (2)
    - injektiv, surjektiv, bijektiv
    - Mächtigkeit, Abzählbarkeit
    - Schubfachprinzip
  6. Primzahlen und Teiler (1)
  7. Modulare Arithmetik (1)

## B. Eindimensionale Analysis (22)

### B.1 Zahlen, Folgen und Reihen (8)

8. Axiomatik der reellen Zahlen, sup, inf (1)
9. Komplexe Zahlen (1)
10. Folgen (1 1/2)
11. Landau'sche Symbole (1/2)
12. Reihen: Konvergenzkriterien, absolute Kgz. (2)
13. Potenzreihen (1/2)
14. Zahlendarstellungen (1/2)
15. Binomialkoeffizienten und Binomialreihe (1)

### B.2 Eindimensionale Differentialrechnung (8)

16. Stetigkeit (1)
17. Elementare Funktionen (1)
18. Differenzierbarkeit (1 1/2)
19. Mittelwertsätze und L'Hospital (1/2)
20. Satz von Taylor (1)
21. Lokale Extrema, Konvexität, Kurvendiskussion (2)
22. Numerische Differentiation (1)

### B.3 Eindimensionale Integralrechnung (6)

23. Das bestimmte Integral (2)
24. Das unbestimmte Integral und die Stammfunktion (1)
25. Uneigentliche Integrale (1)
26. Numerische Verfahren zur Integration (1)
27. Kurven und Bogenlänge (1)

## Literaturhinweise

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet

## Weitere Informationen

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Mathematics for Computer Scientists 1*.



Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>2</b>	<b>6</b>	<b>jedes Sommersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Joachim Weickert

**Dozent/inn/en** Prof. Dr. Joachim Weickert  
 Prof. Dr. Mark Groves  
 Prof. Dr. Henryk Zähle  
 Prof. Dr. Christian Bender

**Zulassungsvoraussetzungen** Mathematik für Informatiker 1 (empfohlen)

**Leistungskontrollen / Prüfungen**

- Teilnahme an den Übungen und Bearbeitung der wöchentlichen Übungsaufgaben (50 Prozent der Übungspunkte werden zur Klausurteilnahme benötigt)
- Bestehen der Abschlussklausur oder der Nachklausur

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
 + 2 SWS Übung  
 = 6 SWS

**Arbeitsaufwand** 90 h Präsenzstudium  
 + 180 h Eigenstudium  
 = 270 h (= 9 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen und praktischen Aufgaben ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben.

**Sprache** Deutsch und Englisch

## Lernziele / Kompetenzen

- Erarbeitung von mathematischem Grundlagenwissen, das im Rahmen eines Informatik- bzw. Bioinformatikstudiums benötigt wird
- Fähigkeit zur Formalisierung und Abstraktion
- Befähigung zur Aneignung weiteren mathematischen Wissens mit Hilfe von Lehrbüchern

## Inhalt

Die Zahlen geben die Gesamtzahl der Doppelstunden an.

### LINEARE ALGEBRA

#### C. Algebraische Strukturen (5)

- 29. Gruppen (2)
- 30. Ringe und Körper (1)
- 31. Polynomringe über allgemeinen Körpern (1/2)
- 32. Boole'sche Algebren (1/2)

#### D. Lineare Algebra (21)

- 33. Vektorräume (2)
  - Def., Bsp.,
  - lineare Abb.
  - Unterraum,
  - Erzeugnis, lineare Abhängigkeit, Basis, Austauschatz

34. Lineare Abb. (Bild, Kern) (1)
35. Matrixschreibweise für lineare Abbildungen (1 1/2)
  - Interpretation als lineare Abbildungen
  - Multiplikation durch Hintereinanderausführung
  - Ringstruktur
  - Inverses
36. Rang einer Matrix (1/2)
37. Gauss-Algorithmus für lineare Gleichungssysteme: (2)
  - Gausselimination (1)
  - Lösungstheorie (1)
38. Iterative Verfahren für lineare Gleichungssysteme (1)
39. Determinanten (1)
40. Euklidische Vektorräume, Skalarprodukt (1)
41. Funktionalanalytische Verallgemeinerungen (1)
42. Orthogonalität (2)
43. Fourierreihen (1)
44. Orthogonale Matrizen (1)
45. Eigenwerte und Eigenvektoren (1)
46. Eigenwerte und Eigenvektoren symmetrischer Matrizen (1)
47. Quadratische Formen und positiv definite Matrizen (1)
48. Quadriken (1)
50. Matrixnormen und Eigenwertabschätzungen (1)
51. Numerische Berechnung von Eigenwerten und Eigenvektoren (1)

## Literaturhinweise

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet

## Weitere Informationen

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Mathematics for Computer Scientists 2*.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>3</b>	<b>6</b>	<b>jedes Wintersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Joachim Weickert

**Dozent/inn/en** Prof. Dr. Joachim Weickert  
 Prof. Dr. Mark Groves  
 Prof. Dr. Henryk Zähle  
 Prof. Dr. Christian Bender

**Zulassungsvoraussetzungen** Mathematik für Informatiker 1 und 2 (empfohlen)

**Leistungskontrollen / Prüfungen**

- Teilnahme an den Übungen und Bearbeitung der wöchentlichen Übungsaufgaben (50 Prozent der Übungspunkte werden zur Klausurteilnahme benötigt)
- Bestehen der Abschlussklausur oder der Nachklausur

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
 + 2 SWS Übung  
 = 6 SWS

**Arbeitsaufwand** 90 h Präsenzstudium  
 + 180 h Eigenstudium  
 = 270 h (= 9 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen und praktischen Aufgaben ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben.

**Sprache** Englisch

## Lernziele / Kompetenzen

- Erarbeitung von mathematischem Grundlagenwissen, das im Rahmen eines Informatik- bzw. Bioinformatikstudiums benötigt wird
- Fähigkeit zur Formalisierung und Abstraktion
- Befähigung zur Aneignung weiteren mathematischen Wissens mit Hilfe von Lehrbüchern

## Inhalt

Die Zahlen geben die Gesamtzahl der Doppelstunden an.

### STOCHASTIK, NUMERIK UND MEHRDIMENSIONALE ANALYSIS

#### E. NUMERISCHE ERGÄNZUNGEN (3)

- 52. Banachscher Fixpunktsatz (1)
- 53. Interpolation, incl. Splines (2)

#### F. MEHRDIMENSIONALE ANALYSIS UND NUMERIK (11)

- 54. Stetigkeit und Differentialoperatoren für skalarwertige Funktionen (2)
- 55. Differentialoperatoren für vektorwertige Funktionen (1)
- 56. Totale Differenzierbarkeit (1/2)
- 57. Mittelwertsatz und Satz von Taylor (1 1/2)
- 58. Extrema von Funktionen mehrerer Variabler (1)
- 59. Das Newton-Verfahren (1)
- 60. Extrema mit Nebenbedingungen (1)

- 61. Mehrfachintegrale (1)
- 62. Die Umkehrfunktion und die Transformationsregel (1)
- 63. Variationsrechnung (1)
  
- G. STOCHASTIK (16)
  - 64. Grundbegriffe (Ws., Stichprobenraum) (1/3)
  - 65. Kombinatorik (2/3)
  - 66. Erzeugende Funktionen (1)
  - 67. Bedingte Wahrscheinlichkeiten (1)
  - 68. Zufallsvariable, Erwartungswert, Varianz (2) (Systemzuverlässigkeit, Varianz, Kovarianz, Jensen)
  - 69. Abschätzungen für Abweichungen vom Mittelwert (1) (Momente, Schranken von Markov, Chebyshev, Chernoff, schwaches Gesetz der grossen Zahlen)
  - 70. Wichtige diskrete Verteilungen (1)
  - 71. Wichtige kontinuierliche Verteilungen (1) (incl. zentraler Grenzwertsatz)
  - 72. Multivariate Verteilungen und Summen von Zufallsvariablen (1)
  - 73. Parameterschätzung und Konfidenzintervalle (1)
  - 74. Hypothesentests (1)
  - 75. Methode der kleinsten Quadrate (1)
  - 76. Robuste Statistik (2/3)
  - 77. Fehlerfortpflanzung (1/3)
  - 78. Markowketten (2)
  - 79. Pseudozufallszahlen und Monte-Carlo-Simulation (1)

## **Literaturhinweise**

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet

## **Weitere Informationen**

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul \*Mathematics for Computer Scientists 3.



Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4</b>	<b>6</b>	<b>jedes Sommersemester</b>	<b>1 Semester</b>	<b>4</b>	<b>6</b>

**Modulverantwortliche/r** Prof. Dr. Jens Dittrich

**Dozent/inn/en** Prof. Dr. Jens Dittrich

**Zulassungsvoraussetzungen** *Programmierung 1, Programmierung 2, Softwarepraktikum oder Projektpraktikum, Mathematik für Informatiker 1, sowie Grundzüge von Algorithmen und Datenstrukturen (jeweils empfohlen).*

**Leistungskontrollen / Prüfungen** Erfolgreiche Teilnahme an den Übungen/Projekt berechtigt zur Teilnahme an der Abschlussklausur.

**Lehrveranstaltungen / SWS** 2 SWS Vorlesung  
+ 2 SWS Übung  
= 4 SWS

**Arbeitsaufwand** 60 h Präsenzstudium  
+ 120 h Eigenstudium  
= 180 h (= 6 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen, und ggf. Projekt ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekanntgegeben.

**Sprache** Englisch

## Lernziele / Kompetenzen

Die Vorlesung vermittelt grundlegende Kenntnisse über fundamentalen Konzepte von Datenmanagement und Datenanalyse im Kontext von Big Data und Data Science.

Im Rahmen der Übungen kann während des Semesters ein durchgehendes Projekt durchgeführt werden. Dies kann zum Beispiel ein soziales Netzwerk (im Stil von Facebook) sein bzw. jedes andere Projekt, in dem Techniken des Datenmanagements eingeübt werden können (z.B. naturwissenschaftliche Daten, Bilddaten, andere Webapplikationen, etc.). Zunächst wird dieses Projekt in E/R modelliert, dann umgesetzt und implementiert in einem Datenbankschema. Danach wird das Projekt erweitert, um auch unstrukturierte Daten verwalten und analysieren zu können. Insgesamt werden so an einem einzigen Projekt alle fundamentalen Techniken gezeigt, die für das Verwalten und Analysieren von Daten wichtig sind.

## Inhalt

### 1 Einführung und Einordnung

- Einordnung und Abgrenzung: Data Science
- Wert von Daten: Das Gold des 21. Jahrhunderts
- Bedeutung von Datenbanksystemen
- Architekturen: 2-Tier, 3-Tier, etc
- Was sind eigentlich Daten?
- Modellierung vs Realität
- Kosten mangelhafter Modellierung
- Datenbanksystem nutzen vs selbst entwickeln
- Positive Beispiele für Apps
- Anforderungen
- Literaturhinweise
- Vorlesungsmodus

- 2 Datenmodellierung
  - Motivation
  - E/R
  - Relationales Modell
  - Hierarchische Daten
  - Graphen und RDF
  - Redundanz, Normalisierung, Denormalisierung
  - Objektrelationale DBMS
- 3 Anfragesprachen
  - Relationale Algebra
  - Hierarchische Anfragesprachen
  - Graphorientierte Anfragesprachen
- 4 SQL
  - Grundlagen
  - Zusammenhang mit relationaler Algebra
  - PostgreSQL
  - Integritätsbedingungen
  - Transaktionskonzept
  - ACID
  - Sichten (und access control lists)
- 5 Implementierungstechniken
  - Übersicht
  - vom WAS zum WIE
  - Kosten verschiedener Operationen
  - EXPLAIN
  - Physisches Design
  - Indexe, Tuning
  - Datenbank-Tuning
  - Regelbasierte Anfrageoptimierung
  - Kostenbasierte Anfrageoptimierung
  - Machine Learning als Anfrageoptimierungstechnik
- 6 Zeitliche und räumliche Daten
  - als Teil des Schemas
  - as of/time travel
  - append-only und Streaming
  - Versioning
  - Snapshotting (Software und OS-basiert)
  - Differential Files/LSM et al
  - Publish/Subscribe
  - Indexstrukturen
- 7 Recovery, Durability, Archivierung
  - Grundproblematik
  - Vergessen vs Komprimieren vs Kondensieren
  - Heiße vs kalte Daten
  - Archivierung
  - Redundanz
  - Implementierungsaspekte
  - UNDO/REDO
  - Logging
- 8 Nebenläufigkeitskontrolle
  - Serialisierbarkeitstheorie
  - Isolationslevels
  - Verteilte Datenbanksysteme: Sharding, HP, VP, permissioned Blockchains
  - Implementierungsaspekte

## 9 ETL und Data Cleaning

- Datenbankschnittstellen: JDBC et al
- Textdatenbanken: CSV, Sqlite
- Data Warehousing
- Schema Matching
- Reporting
- Data Cleaning
- Denormalisierung, Caching, Materialisierung
- Workflows
- ETL und Data Science in Data Science und Machine Learning

## 10 Big Data

- Was ist eigentlich Big Data?
- Big Data vs Privatheit
- Beispiele: Zusammenführen von Daten
- Physische Barrieren

## 11 NoSQL

- Key/Value Stores
- KeyDocument Stores: MongoDB
- MapReduce
- Flink
- Spark

## Literaturhinweise

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet.

## Weitere Informationen

Dieses Modul wurde früher auch unter dem Namen *Informationssysteme* geführt. Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Big Data Engineering*.



Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>3</b>	<b>6</b>	<b>jedes Wintersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Raimund Seidel

**Dozent/inn/en** Prof. Dr. Raimund Seidel  
 Prof. Dr. Bernd Finkbeiner  
 Prof. Dr. Kurt Mehlhorn  
 Prof. Dr. Markus Bläser

**Zulassungsvoraussetzungen** *Programmierung 1 und 2 und Mathematik für Informatiker 1 und 2* oder vergleichbare Veranstaltungen der Mathematik sind empfohlen.

**Leistungskontrollen / Prüfungen** Erfolgreiche Bearbeitung der Übungsaufgaben berechtigt zur Klausurteilnahme.

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
 + 2 SWS Übung  
 = 6 SWS

**Arbeitsaufwand** 90 h Präsenzstudium  
 + 180 h Eigenstudium  
 = 270 h (= 9 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen und praktischen Aufgaben ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben.

**Sprache** Englisch

## Lernziele / Kompetenzen

Die Studierenden kennen verschiedene Rechenmodelle und ihre relativen Stärken und Mächtigkeiten.

Sie können für ausgewählte Probleme zeigen, ob diese in bestimmten Rechenmodellen lösbar sind oder nicht.

Sie verstehen den formalen Begriff der Berechenbarkeit wie auch der Nicht-Berechenbarkeit.

Sie können Probleme aufeinander reduzieren.

Sie sind vertraut mit den Grundzügen der Ressourcenbeschränkung (Zeit, Platz) für Berechnungen und der sich daraus ergebenden Komplexitätstheorie.

## Inhalt

Die Sprachen der Chomsky Hierarchie und ihre verschiedenen Definitionen über Grammatiken und Automaten; Abschlusseigenschaften; Klassifikation von bestimmten Sprachen („Pumping lemmas“);

Determinismus und Nicht-Determinismus;

Turing Maschinen und äquivalente Modelle von allgemeiner Berechenbarkeit (z.B.  $\mu$ -rekursive Funktionen, Random Access Machines) Reduzierbarkeit, Entscheidbarkeit, Nicht-Entscheidbarkeit;

Die Komplexitätsmaße Zeit und Platz; die Komplexitätsklassen P und NP;

Grundzüge der Theorie der NP-Vollständigkeit

## **Literaturhinweise**

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet.

## **Weitere Informationen**

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Introduction to Theoretical Computer Science*.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>3</b>	<b>6</b>	<b>jedes Wintersemester</b>	<b>1 Semester</b>	<b>4</b>	<b>6</b>

**Modulverantwortliche/r** Prof. Dr. Raimund Seidel

**Dozent/inn/en** Prof. Dr. Raimund Seidel  
 Prof. Dr. Kurt Mehlhorn  
 Prof. Dr. Markus Bläser

**Zulassungsvoraussetzungen** *Programmierung 1 und 2, und Mathematik für Informatiker 1 und 2* oder vergleichbare Veranstaltungen der Mathematik sind empfohlen.

**Leistungskontrollen / Prüfungen** Erfolgreiche Bearbeitung der Übungsblätter berechtigt zur Klausurteilnahme.

**Lehrveranstaltungen / SWS** 2 SWS Vorlesung  
 + 2 SWS Übung  
 = 4 SWS

**Arbeitsaufwand** 60 h Präsenzstudium  
 + 120 h Eigenstudium  
 = 180 h (= 6 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen und praktischen Aufgaben ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben.

**Sprache** Englisch

## Lernziele / Kompetenzen

Die Studierenden lernen die wichtigsten Methoden des Entwurfs von Algorithmen und Datenstrukturen kennen: Teile-und-Herrsche, Dynamische Programmierung, inkrementelle Konstruktion, „Greedy“, Dezimierung, Hierarchisierung, Randomisierung. Sie lernen Algorithmen und Datenstrukturen bzgl. Zeit- und Platzverbrauch für das übliche RAM Maschinenmodell zu analysieren und auf Basis dieser Analysen zu vergleichen. Sie lernen verschiedene Arten der Analyse (schlechtester Fall, amortisiert, erwartet) einzusetzen.

Die Studierenden lernen wichtige effiziente Datenstrukturen und Algorithmen kennen. Sie sollen die Fähigkeit erwerben, vorhandene Methoden durch theoretische Analysen und Abwägungen für ihre Verwendbarkeit in tatsächlich auftretenden Szenarien zu prüfen. Ferner sollen die Studierenden die Fähigkeit trainieren, Algorithmen und Datenstrukturen unter dem Aspekt von Performanzgarantien zu entwickeln oder anzupassen

## Inhalt

### Literaturhinweise

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet.

### Weitere Informationen

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Fundamentals of Data Structures and Algorithms*.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>1</b>	<b>6</b>	<b>jedes Wintersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Gert Smolka

**Dozent/inn/en** Prof. Dr. Gert Smolka  
 Prof. Dr.-Ing. Holger Hermanns  
 Prof. Bernd Finkbeiner, Ph.D

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen**

- zwei Klausuren (Mitte und Ende der Vorlesungszeit)
- Die Note wird aus den Klausuren gemittelt und kann durch Leistungen in den Übungen verbessert werden.
- Eine Nachklausur findet innerhalb der letzten beiden Wochen vor Vorlesungsbeginn des Folgesemesters statt.

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
 + 2 SWS Übung  
 = 6 SWS

**Arbeitsaufwand** 90 h Präsenzstudium  
 + 180 h Eigenstudium  
 = 270 h (= 9 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen und praktischen Aufgaben ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben.

**Sprache** Deutsch und Englisch

## Lernziele / Kompetenzen

- höherstufige, getypte funktionale Programmierung anwenden können
- Verständnis rekursiver Datenstrukturen und Algorithmen, Zusammenhänge mit Mengenlehre
- Korrektheit beweisen und Laufzeit abschätzen
- Typabstraktion und Modularisierung verstehen
- Struktur von Programmiersprachen verstehen
- einfache Programmiersprachen formal beschreiben können
- einfache Programmiersprachen implementieren können
- anwendungsnahe Rechenmodelle mit maschinennahen Rechenmodellen realisieren können
- Praktische Programmiererfahrung, Routine im Umgang mit Interpretern und Übersetzern

## Inhalt

- Funktionale Programmierung
- Algorithmen und Datenstrukturen (Listen, Bäume, Graphen; Korrektheitsbeweise; asymptotische Laufzeit)
- Typabstraktion und Module
- Programmieren mit Ausnahmen
- Datenstrukturen mit Zustand
- Struktur von Programmiersprachen (konkrete und abstrakte Syntax, statische und dynamische Syntax)
- Realisierung von Programmiersprachen (Interpreter, virtuelle Maschinen, Übersetzer)

## **Literaturhinweise**

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet

## **Weitere Informationen**

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Programming 1*.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>2</b>	<b>6</b>	<b>jedes Sommersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Sebastian Hack

**Dozent/inn/en** Prof. Dr. Sebastian Hack  
Prof. Dr. Jörg Hoffmann

**Zulassungsvoraussetzungen** *Programmierung 1* und *Mathematik für Informatiker 1* und Mathematikveranstaltungen im Studiensemester oder vergleichbare Kenntnisse aus sonstigen Mathematikveranstaltungen (empfohlen)

**Leistungskontrollen / Prüfungen** Prüfungsleistungen werden in zwei Teilen erbracht, die zu gleichen Teilen in die Endnote eingehen. Um die Gesamtveranstaltung zu bestehen, muss jeder Teil einzeln bestanden werden.

Im **Praktikumsteil** müssen die Studierenden eine Reihe von Programmieraufgaben selbstständig implementieren. Diese Programmieraufgaben ermöglichen das Einüben der Sprachkonzepte und führen außerdem komplexere Algorithmen und Datenstrukturen ein. Automatische Tests prüfen die Qualität der Implementierungen. Die Note des Praktikumsteils wird maßgeblich durch die Testergebnisse bestimmt.

Im **Vorlesungsteil** müssen die Studierenden Klausuren absolvieren und Übungsaufgaben bearbeiten. Die Aufgaben vertiefen dabei den Stoff der Vorlesung. Die Zulassung zu der Klausur hängt von der erfolgreichen Bearbeitung der Übungsaufgaben ab.

Im Praktikumsteil kann eine Nachaufgabe angeboten werden

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
+ 2 SWS Übung  
= 6 SWS

**Arbeitsaufwand** 90 h Präsenzstudium  
+ 180 h Eigenstudium  
= 270 h (= 9 ECTS)

**Modulnote** Wird aus Leistungen in Klausuren, Übungen und praktischen Aufgaben ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben

**Sprache** Deutsch und Englisch

## Lernziele / Kompetenzen

Die Studierenden lernen die Grundprinzipien der imperativen /objektorientierten Programmierung kennen. Dabei wird primär Java als Programmiersprache verwendet.

In dieser Vorlesung lernen sie:

- wie Rechner Programme ausführen
- Die Grundlagen imperativer und objektorientierter Sprachen
- kleinere, wohlstrukturierte Programme in C zu schreiben
- mittelgroße objektorientierte Systeme in Java zu implementieren und zu testen
- sich in wenigen Tagen eine neue imperative/objektorientierte Sprache anzueignen, um sich in ein bestehendes Projekt einzuarbeiten

## **Inhalt**

- Imperatives Programmieren
- Objekte und Klassen
- Klassendefinitionen
- Objektinteraktion
- Objektsammlungen
- Objekte nutzen und testen
- Vererbung
- Dynamische Bindung
- Fehlerbehandlung
- Klassendesign und Modularität
- Systemnahe Programmierung

sowie spezifische Vorlesungen für die Programmieraufgaben.

## **Literaturhinweise**

Bekanntgabe jeweils vor Beginn der Vorlesung auf der Vorlesungsseite im Internet

## **Weitere Informationen**

Dieses Modul ist inhaltsgleich mit dem englischsprachigen Modul *Programming 2*.

**Modulbereich 4**

---

***Spezialisierter Bereich DSAI***



Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>1</b>	<b>6</b>	<b>every winter semester</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Jörg Hoffmann  
 Prof. Dr. Jens Dittrich  
 Prof. Dr. Bernt Schiele  
 Prof. Dr. Vera Demberg

**Dozent/inn/en** Prof. Dr. Jörg Hoffmann  
 Prof. Dr. Jens Dittrich  
 Prof. Dr. Bernt Schiele  
 Prof. Dr. Vera Demberg

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen** Weekly assignments,  
 Exam (qualification for exam depends on performance in assignments)

**Lehrveranstaltungen / SWS** 4 h lectures  
 + 2 h tutorial  
 = 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
 + 180 h private study  
 = 270 h (= 9 ECTS)

**Modulnote** Based on exam. The exact modalities are specified by the lecturers.

**Sprache** English

## Lernziele / Kompetenzen

Overview of challenges and methods in Data Science and AI. Basic knowledge of key concepts and algorithms.

## Inhalt

Introduction to history and concepts of Data Science and AI

- Machine Learning (supervised, unsupervised, reinforcement, neural networks)
- (adversarial) Search, Planning
- Reasoning
- Modeling and Simulation
- Data Management, Big Data Engineering, and Analytics

The methods will be covered in the context of applications, such as Game Playing, Computer Vision, Autonomous Driving, Language Processing, Social Networks.

The exercises will cover methodological, algorithmic, as well as practical aspects. Where basic programming or scripting skills are required, the lecture and exercises will introduce these skills.

## Literaturhinweise

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>3</b>	<b>6</b>	<b>every winter semester</b>	<b>1 semester</b>	<b>4</b>	<b>6</b>

**Modulverantwortliche/r** Prof. Dr. Jilles Vreeken  
Prof. Dr. Isabel Valera

**Dozent/inn/en** Prof. Dr. Jilles Vreeken  
Prof. Dr. Isabel Valera

**Zulassungsvoraussetzungen** The lecture assumes basic knowledge in statistics, linear algebra, and programming. It is advisable to have successfully completed *Mathematics for Computer Scientists 2* and *Statistics Lab*. The exercises use the programming language R. We will give a basic introduction to R in the first tutorial. In addition, for preparation the following materials are useful: *R for Beginners* by Emmanuel Paradis (especially chapters 1, 2, 3 and 6) and *An introduction to R* (Venables/Smith).

**Leistungskontrollen / Prüfungen** Prerequisite for admission to the examination is a cumulative 50% of the points of the theoretical and a cumulative 50% of the points of the practical tasks on the exercise sheets. Depending on the number of participants, the examinations are either written or oral. The final modality will be announced in the first two weeks of the lecture.

**Lehrveranstaltungen / SWS** 2 h lectures  
+ 2 h tutorial  
= 4 h (weekly)

**Arbeitsaufwand** 60 h of classes  
+ 120 h private study  
= 180 h (= 6 ECTS)

**Modulnote** Will be determined from performance in exams.

**Sprache** English

## Lernziele / Kompetenzen

In this course we will discuss the foundations – the elements – of machine learning. In particular, we will focus on the ability of, given a data set, to choose an appropriate method for analyzing it, to select the appropriate parameters for the model generated by that method and to assess the quality of the resulting model. Both theoretical and practical aspects will be covered. What we cover will be relevant for computer scientists in general as well as for other scientists involved in data analysis and modeling.

## Inhalt

The lecture covers basic machine learning methods, in particular the following contents:

- Introduction to statistical learning
- Overview over Supervised Learning
- Linear Regression
- Linear Classification
- Splines
- Model selection and estimation of the test errors
- Maximum-Likelihood Methods
- Additive Models
- Decision trees

- Boosting
- Dimensionality reduction
- Unsupervised learning
- Clustering
- Visualization

## **Literaturhinweise**

The course broadly follows the book *An Introduction to Statistical Learning with Applications in R*, Springer (2013). In some cases, the course receives additional material from the book *The Elements of Statistical Learning*, Springer (second edition, 2009). The first book is the introductory text, the second covers more advanced topics. Both books are available as free PDFs. Any change of, or additional material will be announced before the start of the course on the course webpage.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>2</b>	<b>6</b>	<b>jedes Sommersemester</b>	<b>1 Semester</b>	<b>4</b>	<b>6</b>

**Modulverantwortliche/r** Prof. Dr. Verena Wolf  
Prof. Dr. Vera Demberg

**Dozent/inn/en** Prof. Dr. Verena Wolf  
Prof. Dr. Vera Demberg

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen** mündliche oder schriftliche Prüfung

**Lehrveranstaltungen / SWS** 2 SWS Vorlesung  
+ 2 SWS Übung  
= 4 SWS

**Arbeitsaufwand** 60 h Präsenzstudium  
+ 120 h Eigenstudium  
= 180 h (= 6 ECTS)

**Modulnote** Wird aus Leistungen in der Klausur, sowie den Prüfungsvorleistungen ermittelt. Die genauen Modalitäten werden vom Modulverantwortlichen bekannt gegeben. Alle Modulelemente sind innerhalb eines Prüfungszeitraumes erfolgreich zu absolvieren.

**Sprache** Deutsch oder Englisch

## Lernziele / Kompetenzen

- Verständnis der mathematischen Konzepte von Zufallsvariablen und Verteilungen
- Verständnis und Anwendung von Methoden der Punkt- und Intervallschätzung, statistischer Tests
- Verständnis der mathematischen Konzepte von Zustandsdiskreten Markovprozessen und Verwendung solcher Prozesse zur Beschreibung von realen Phänomenen

## Inhalt

### Probabilities and Discrete Random Variables

- Probability
- discrete RVs
- expectation, variance and quantiles (also visualization of them)
- higher moments
- important discrete probability distributions
- Generating discrete random variates Continuous Random Variables and Laws of Large Numbers
- $\sigma$ -algebras (very lightweight)
- Continuous Random Variables
- Important Continuous Distributions
- generating continuous random variates
- Chebyshev's inequality
- Weak/Strong Law of Large Numbers
- Central Limit Theorem

### Multidimensional Probability Distributions

- joint probability distribution

- conditional probability distribution
- Bayes' Theorem
- covariance and correlation
- independence
- important multidimensional probability distributions

#### Point Estimation

- (generalized) method of moments
- maximum likelihood estimation
- Bayesian inference (posterior mean/median, MAP)
- Kernel density estimation
- OLS estimator (this is simple regression but should be mentioned here!)
- (shortly: model selection)

#### Interval Estimation

- confidence intervals for sample mean/variance
- confidence intervals for MLE
- bootstrap confidence interval
- Bayesian credible interval

#### Statistical Testing

- Level  $\alpha$  tests (Z-Test, T-Test)
- p-value
- chi-squared tests, Fisher test
- multiple testing (Bonferroni correction, Holm-Bonferroni method, Benjamini-Hochberg, etc)

#### Discrete-time Markov chains (only if time)

- transient distributions
- equilibrium distributions
- Monte-Carlo simulation

#### HMMs

- Baum-Welch-Algorithmus
- Viterbi-Algorithmus

### **Literaturhinweise**

*Modulbereich 5*

---

*Stammvorlesungen DSAI*

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-5</b>	<b>6</b>	<b>at least every two years</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Jörg Hoffmann

**Dozent/inn/en** Prof. Dr. Jörg Hoffmann

**Zulassungsvoraussetzungen** *Programming 1, Programming 2, Fundamentals of Data Structures and Algorithms, and Elements of Machine Learning* or other courses in machine learning are recommended.

**Leistungskontrollen / Prüfungen**

- Regular attendance of classes and tutorials
- Solving of weekly assignments
- Passing the final written exam
- A re-exam takes place during the last two weeks before the start of lectures in the following semester.

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Will be determined from the performance in exams. The exact modalities will be announced at the beginning of the module.

**Sprache** English

## Lernziele / Kompetenzen

Knowledge about basic methods in Artificial Intelligence

## Inhalt

Search:

- Uninformed- and informed search procedures
- Monte-Carlo tree search

Planning:

- Formalism and complexity
- Critical-path heuristics
- Delete relaxation heuristics
- Abstraction heuristics

Markov decision processes:

- Discounted reward and expected cost
- Value iteration
- Informed search
- Reinforcement learning

Games:

- Adversarial search
- Learning from self-play

## **Literaturhinweise**

Russel & Norvig Artificial Intelligence: A Modern Approach;  
further reading will be announced before the start of the course on the course page on the Internet.



Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-5</b>	<b>6</b>	<b>at least every two years</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Christoph Weidenbach

**Dozent/inn/en** Prof. Dr. Christoph Weidenbach

**Zulassungsvoraussetzungen** *Introduction to Computational Logic*

**Leistungskontrollen / Prüfungen**

- Regular attendance of classes and tutorials
- Weekly assignments
- Practical work with systems
- Passing the final and mid-term exam
- A re-exam takes place during the last two weeks before the start of lectures in the following semester.

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Will be determined from performance in exams, exercises and practical tasks. The exact modalities will be announced at the beginning of the module.

**Sprache** English

## Lernziele / Kompetenzen

The goal of this course is to provide familiarity with logics, calculi, implementation techniques, and systems providing automated reasoning.

## Inhalt

Propositional Logic – CDCL, Superposition - Watched Literals  
First-Order Logic without Equality – (Ordered) Resolution,  
Equations with Variables – Completion, Termination  
First-Order Logic with Equality – Superposition (SUP) - Indexing

## Literaturhinweise

Will be announced before the start of the course on the course page on the Internet.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-5</b>	<b>6</b>	<b>at least every two years</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Jens Dittrich

**Dozent/inn/en** Prof. Dr. Jens Dittrich

**Zulassungsvoraussetzungen** especially Saarland University CS department's undergraduate lecture *Big Data Engineering* (former *Informationssysteme*), *Programmierung 1* and *2*, *Algorithmen und Datenstrukturen* as well as *Nebenläufige Programmierung*

For graduate students:

- motivation for databases and database management systems;
- the relational data model;
- relational query languages, particularly relational algebra and SQL;
- **solid** programming skills in Java and/or C++
- undergrad courses in algorithms and data structures, concurrent programming

**Leistungskontrollen / Prüfungen**

- Passing a two-hour written exam at the end of the semester
- Successful demonstration of programming project (teams of up to three students are allowed); the project may be integrated to be part of the weekly assignments

Grades are based on written exam; 50% in weekly assignments (in paper and additionally paper or electronic quizzes) must be passed to participate in the final and repetition exams.

A repetition exam takes place during the last two weeks before the start of lectures in the following semester.

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

This class may be run as a flipped classroom, i.e. 2 hours of lectures may be replaced by self-study of videos/papers; the other 2 hours may be used to run a group exercise supervised by the professor called "the LAB")

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Will be determined based on project, midterm and best of endterm and reexam.

**Sprache** English

## Lernziele / Kompetenzen

Database systems are the backbone of most modern information systems and a core technology without which today's economy – as well as many other aspects of our lives – would be impossible in their present forms. The course teaches the architectural and algorithmic foundations of modern database management systems (DBMS), focussing on database systems internals rather than applications. Emphasis is made on robust and time-tested techniques that have led databases to be considered a mature technology and one of the greatest success stories in computer science. At the same time, opportunities for exciting research in this field will be pointed out.

In the exercise part of the course, important components of a DBMS will be treated and where possible implemented and their performance evaluated. The goal this is to work with the techniques introduced in the lecture and to understand them and their practical implications to a depth that would not be attainable by purely theoretical study.

## Inhalt

The course "Database Systems" will introduce students to the internal workings of a DBMS, in particular:

- storage media (disk, flash, main memory, caches, and any other future storage medium)
- data managing architectures (DBMS, streams, file systems, clouds, appliances)
- storage management (DB-file systems, raw devices, write-strategies, differential files, buffer management)
- data layouts (horizontal and vertical partitioning, columns, hybrid mappings, compression, defragmentation)
- indexing (one- and multidimensional, tree-structured, hash-, partition-based, bulk-loading and external sorting, differential indexing, read- and write-optimized indexing, data warehouse indexing, main-memory indexes, sparse and dense, direct and indirect, clustered and unclustered, main memory versus disk and/or flash-based)
- processing models (operator model, pipeline models, push and pull, block-based iteration, vectorization, query compilation)
- processing implementations (join algorithms for relational data, grouping and early aggregation, filtering)
- query processing (scanning, plan computation, SIMD)
- query optimization (query rewrite, cost models, cost-based optimization, join order, join graph, plan enumeration)
- data recovery (single versus multiple instance, logging, ARIES)
- parallelization of data and queries (horizontal and vertical partitioning, shared-nothing, replication, distributed query processing, NoSQL, MapReduce, Hadoop and/or similar and/or future systems)
- read-optimized system concepts (search engines, data warehouses, OLAP)
- write-optimized system concepts (OLTP, streaming data)
- management of geographical data (GIS, google maps and similar tools)
- main-memory techniques

## Literaturhinweise

Will be announced before the start of the course on the course page on the Internet.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-5</b>	<b>6</b>	<b>at least every two years</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Joachim Weickert

**Dozent/inn/en** Prof. Dr. Joachim Weickert

**Zulassungsvoraussetzungen** Undergraduate mathematics (e.g. Mathematik für Informatiker I-III) and elementary programming knowledge in C

**Leistungskontrollen / Prüfungen**

- For the homework assignments one can obtain up to 24 points per week. Actively participating in the classroom assignments gives 12 more points per week, regardless of the correctness of the solutions. To qualify for both exams one needs 2/3 of all possible points.
- Passing the final exam or the re-exam.
- A re-exam takes place during the last two weeks before the start of lectures in the following semester.

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Will be determined from the performance in the exam or the re-exam. The better grade counts.

**Sprache** English

## Lernziele / Kompetenzen

Broad introduction to mathematical methods in image processing and computer vision. The lecture qualifies students for a bachelor thesis in this field. Together with the completion of advanced or specialised lectures (9 credits at least) it is the basis for a master thesis in this field.

## Inhalt

Inhalt

1. Basics
  - 1.1 Image Types and Discretisation
  - 1.2 Degradations in Digital Images
2. Colour Perception and Colour Spaces
3. Image Transformations
  - 3.1 Continuous Fourier Transform
  - 3.2 Discrete Fourier Transform
  - 3.3 Image Pyramids
  - 3.4 Wavelet Transform
4. Image Compression
5. Image Interpolation
6. Image Enhancement
  - 6.1 Point Operations

- 6.2 Linear Filtering and Feature Detection
- 6.3 Morphology and Median Filters
- 6.3 Wavelet Shrinkage, Bilateral Filters, NL Means
- 6.5 Diffusion Filtering
- 6.6 Variational Methods
- 6.7 Deconvolution Methods
- 7. Texture Analysis
- 8. Segmentation
  - 8.1 Classical Methods
  - 8.2 Variational Methods
- 9. Image Sequence Analysis
  - 9.1 Local Methods
  - 9.2 Variational Methods
- 10. 3-D Reconstruction
  - 10.1 Camera Geometry
  - 10.2 Stereo
  - 10.3 Shape-from-Shading
- 11. Object Recognition
  - 11.1 Hough Transform
  - 11.2 Invariants
  - 11.3 Eigenspace Methods

## **Literaturhinweise**

Will be announced before the start of the course on the course page on the Internet.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-5</b>	<b>6</b>	<b>at least every two years</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Gerhard Weikum

**Dozent/inn/en** Prof. Dr. Gerhard Weikum

**Zulassungsvoraussetzungen** Good knowledge of undergraduate mathematics (linear algebra, probability theory) and basic algorithms.

**Leistungskontrollen / Prüfungen**

- Regular attendance of classes and tutor groups
- Presentation of solutions in tutor groups
- Passing 2 of 3 written tests (after each third of the semester)
- Passing the final exam (at the end of the semester)

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Will be determined by the performance in written tests, tutor groups, and the final exam. Details will be announced on the course web site.

**Sprache** English

## Lernziele / Kompetenzen

The lecture teaches models and algorithms that form the basis for search engines and for data mining and data analysis tools.

## Inhalt

Information Retrieval (IR) and Data Mining (DM) are methodologies for organizing, searching and analyzing digital contents from the web, social media and enterprises as well as multivariate datasets in these contexts. IR models and algorithms include text indexing, query processing, search result ranking, and information extraction for semantic search. DM models and algorithms include pattern mining, rule mining, classification and recommendation. Both fields build on mathematical foundations from the areas of linear algebra, graph theory, and probability and statistics.

## Literaturhinweise

Will be announced on the course web site.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-5</b>	<b>6</b>	<b>at least every two years</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Isabel Valera

**Dozent/inn/en** Prof. Dr. Isabel Valera

**Zulassungsvoraussetzungen** The lecture gives a broad introduction into machine learning methods. After the lecture the students should be able to solve and analyze learning problems.

**Leistungskontrollen / Prüfungen**

- Regular attendance of classes and tutorials.
- 50% of all points of the exercises have to be obtained in order to qualify for the exam.
- Passing 1 out of 2 exams (final, re-exam).

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Determined from the results of the exams, exercises and potential projects. The exact grading modalities are announced at the beginning of the course.

**Sprache** English

## Lernziele / Kompetenzen

The lecture gives a broad introduction into machine learning methods. After the lecture the students should be able to solve and analyze learning problems.

## Inhalt

- Bayesian decision theory
- Linear classification and regression
- Kernel methods
- Bayesian learning
- Semi-supervised learning
- Unsupervised learning
- Model selection and evaluation of learning methods
- Statistical learning theory
- Other current research topics

## Literaturhinweise

Will be announced before the start of the course on the course page on the Internet.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-5</b>	<b>6</b>	<b>every winter semester</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Dietrich Klakow

**Dozent/inn/en** Prof. Dr. Dietrich Klakow

**Zulassungsvoraussetzungen** *Mathematik für Informatiker I - III* or comparable; good programming skills.

**Leistungskontrollen / Prüfungen** Written Exam

**Lehrveranstaltungen / SWS** 2 h lectures  
+ 2 h tutorial  
+ 2 h project work  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Written exam and graded projects. Exact details will be announced in the first lecture.

**Sprache** English

## Lernziele / Kompetenzen

The participants will be introduced to the key ideas of basic classification algorithms and in particular neural networks. A focus is also the implementation and applications to relevant problems. To achieve this, there will be theoretical exercises as well as project work.

## Inhalt

- Classification
- Regression
- Linear Classifiers
- Perceptron
- Support Vector Machines
- Multy-Layer Perceptrons
- Deep Learning Software
- Autoencoders
- LSTMs
- Recurrent Neural Networks
- Sequence-to-sequence learning

## Literaturhinweise

Ian Goodfellow and Yoshua Bengio and Aaron Courville  
Deep Learning  
MIT Press, 2016  
<http://www.deeplearningbook.org>



**Modulbereich 6**

---

**Vertiefungsvorlesungen DSAI**

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-6</b>	<b>6</b>	<b>winter semester</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Jörg Hoffmann

**Dozent/inn/en** Prof. Dr. Jörg Hoffmann

**Zulassungsvoraussetzungen** For graduate students: none

**Leistungskontrollen / Prüfungen** Regular attendance of classes and tutorial  
Paper as well as programming exercises for exam qualification  
Final exam  
A re-exam takes place before the start of lectures in the following semester.

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Will be determined from performance in exams, exercises and practical tasks. The exact modalities will be announced at the beginning of the module.

**Sprache** English

## Lernziele / Kompetenzen

The students will gain a deep understanding of algorithms used in Automatic Planning for the efficient exploration of large state spaces, from both a theoretical and practical point of view. The programming exercises will familiarize them with the main implementation basis in Automatic Planning. The search algorithms are generic and are relevant also in other CS sub-areas in which large transition systems need to be analyzed.

## Inhalt

Automatic Planning is one of the fundamental sub-areas of Artificial Intelligence, concerned with algorithms that can generate strategies of action for arbitrary autonomous agents in arbitrary environments. The course examines the technical core of the current research on solving this kind of problem, consisting of paradigms for automatically generating heuristic functions (lower bound solution cost estimators), as well as optimality-preserving pruning methods. Apart from understanding these techniques themselves, the course explains how to analyze, combine, and compare them.

Starting from an implementation basis provided, students implement their own planning system as part of the course. The course is concluded by a competition between these student systems.

## Literaturhinweise

Will be announced before the start of the course on the course page on the Internet.

# Mathematische Statistik

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-6</b>	<b>6</b>	<b>unregelmässig</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Henryk Zähle

**Dozent/inn/en** Dozent/inn/en der Mathematik

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen** Regelmäßige, aktive Teilnahme an der Vorlesung und an den begleitenden Übungen; Abschlussprüfung

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
+ 2 SWS Übung  
= 6 SWS

**Arbeitsaufwand** 60 h Vorlesung  
+ 30 h Übungen  
+ 180 h Eigenstudium  
(Vor- und Nachbereitung, Bearbeitung Übungsaufgaben)  
= 270 h (= 9 ECTS)

**Modulnote** Durch Klausur(en) und mündliche Prüfung. Der Modus wird zu Beginn der Vorlesung bekannt gegeben.

**Sprache** Deutsch

## Lernziele / Kompetenzen

### Inhalt

- Statistische Modelle; Modellwahl und Modellüberprüfung
- Punktschätzungen
- Bereichsschätzungen
- Hypothesentests

### Literaturhinweise

Bekanntgabe jeweils vor der Vorlesung auf der Vorlesungsseite im Internet

### Weitere Informationen

*Methoden:* Information durch Vorlesung; Vertiefung durch Eigentätigkeit (Nacharbeit, Übungen).

*Anmeldung:* Bekanntgabe jeweils rechtzeitig vor Semesterbeginn durch Aushang und im Internet.

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-6</b>	<b>6</b>	<b>at least every two years</b>	<b>1 semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Michael Backes

**Dozent/inn/en** Prof. Dr. Michael Backes  
Prof. Dr. Cas Cremers

**Zulassungsvoraussetzungen** For graduate students: none

**Leistungskontrollen / Prüfungen**

- Regular attendance of classes and tutorials
- Passing the final exam
- A re-exam is normally provided (as written or oral examination).

**Lehrveranstaltungen / SWS** 4 h lectures  
+ 2 h tutorial  
= 6 h (weekly)

**Arbeitsaufwand** 90 h of classes  
+ 180 h private study  
= 270 h (= 9 ECTS)

**Modulnote** Will be determined by the performance in exams, tutor groups, and practical tasks. Details will be announced by the lecturer at the beginning of the course.

**Sprache** English

## Lernziele / Kompetenzen

Description, assessment, development and application of security mechanisms, techniques and tools.

## Inhalt

- Basic Cryptography,
- Specification and verification of security protocols,
- Security policies: access control, information flow analysis,
- Network security,
- Media security,
- Security engineering

## Literaturhinweise

Will be announced on the course website

# Stochastik I

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-6</b>	<b>6</b>	<b>jedes Sommersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Henryk Zähle  
Prof. Dr. Christian Bender

**Dozent/inn/en** Dozent/inn/en der Mathematik

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen** Regelmäßige, aktive Teilnahme an der Vorlesung und an den begleitenden Übungen; Abschlussprüfung

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
+ 2 SWS Übung  
= 6 SWS

**Arbeitsaufwand** 60 h Vorlesung  
+ 30 h Übungen  
+ 180 h Eigenstudium  
(Vor- und Nachbereitung, Bearbeitung Übungsaufgaben)  
= 270 h (= 9 ECTS)

**Modulnote** Durch Klausur(en) und mündliche Prüfung. Der Modus wird zu Beginn der Vorlesung bekannt gegeben.

**Sprache** Deutsch

## Lernziele / Kompetenzen

### Inhalt

- Maß- und Integrationstheorie
- Allgemeine Wahrscheinlichkeitsräume
- Zufallsvariablen und deren Verteilungen
- Bedingen auf Ereignisse
- Unabhängigkeit
- Erwartungswert, Varianz, Kovarianz, Korrelation
- Charakterisieren von Verteilungen auf euklidischen Räumen (Verteilungsfunktion, erzeugende Funktionen)
- Summen unabhängiger Zufallsvariablen
- Konvergenzbegriffe für Folgen von Wahrscheinlichkeitsmaßen und Folgen von Zufallsvariablen
- Grenzwertsätze für Summen unabhängiger reellwertiger Zufallsvariablen (Gesetze der großen Zahlen, zentraler Grenzwertsatz)
- Multivariate Normalverteilung, multivariater zentraler Grenzwertsatz

### Literaturhinweise

Bekanntgabe jeweils vor der Vorlesung auf der Vorlesungsseite im Internet.

### Weitere Informationen

*Methoden:* Information durch Vorlesung; Vertiefung durch Eigentätigkeit (Nacharbeit, Übungen).

*Anmeldung:* Bekanntgabe jeweils rechtzeitig vor Semesterbeginn durch Aushang und im Internet.

# Stochastik II

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4-6</b>	<b>6</b>	<b>jedes Wintersemester</b>	<b>1 Semester</b>	<b>6</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Henryk Zähle  
Prof. Dr. Christian Bender

**Dozent/inn/en** Dozent/inn/en der Mathematik

**Zulassungsvoraussetzungen** keine

**Leistungskontrollen / Prüfungen** Regelmäßige, aktive Teilnahme an der Vorlesung und an den begleitenden Übungen; Abschlussprüfung

**Lehrveranstaltungen / SWS** 4 SWS Vorlesung  
+ 2 SWS Übung  
= 6 SWS

**Arbeitsaufwand** 60 h Vorlesung  
+ 30 h Übungen  
+ 180 h Eigenstudium  
(Vor- und Nachbereitung, Bearbeitung Übungsaufgaben)  
= 270 h (= 9 ECTS)

**Modulnote** Durch Klausur(en) und mündliche Prüfung. Der Modus wird zu Beginn der Vorlesung bekannt gegeben.

**Sprache** Deutsch

## Lernziele / Kompetenzen

### Inhalt

- Bedingen auf Sigma-Algebren
- Grundlagen stochastischer Prozesse
- Poisson-Prozess
- Brown'sche Bewegung
- Martingaleigenschaft
- Markoveigenschaft

### Literaturhinweise

Bekanntgabe jeweils vor der Vorlesung auf der Vorlesungsseite im Internet.

### Weitere Informationen

*Methoden:* Information durch Vorlesung; Vertiefung durch Eigentätigkeit (Nacharbeit, Übungen).

*Anmeldung:* Bekanntgabe jeweils rechtzeitig vor Semesterbeginn durch Aushang und im Internet.

*Modulbereich 7*

---

*Seminare DSAI*

# Seminar

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>5</b>	<b>6</b>	<b>jedes Semester</b>	<b>1 Semester</b>	<b>2</b>	<b>7</b>

**Modulverantwortliche/r** Studiendekan der Fakultät Mathematik und Informatik  
Studienbeauftragter der Informatik

**Dozent/inn/en** Dozent/inn/en der Fachrichtung

**Zulassungsvoraussetzungen** Grundlegende Kenntnisse im jeweiligen Teilbereich des Studienganges.

**Leistungskontrollen / Prüfungen**

- Thematischer Vortrag mit anschließender Diskussion
- Aktive Teilnahme an der Diskussion
- Gegebenenfalls schriftliche Ausarbeitung oder Projekt

**Lehrveranstaltungen / SWS** 2 SWS Seminar

**Arbeitsaufwand** 30 h Präsenzstudium  
+ 180 h Eigenstudium  
= 210 h (= 7 ECTS)

**Modulnote** Wird aus den Leistungen im Vortrag und der schriftlichen Ausarbeitung und/oder dem Seminarprojekt ermittelt. Die genauen Modalitäten werden von dem/der jeweiligen Dozenten/in bekannt gegeben.

**Sprache** Deutsch oder Englisch

## Lernziele / Kompetenzen

Die Studierenden haben am Ende der Veranstaltung vor allem ein tiefes Verständnis aktueller oder fundamentaler Aspekte eines spezifischen Teilbereiches der Informatik erlangt.

Sie haben weitere Kompetenz im eigenständigen wissenschaftlichen Recherchieren, Einordnen, Zusammenfassen, Diskutieren, Kritisieren und Präsentieren von wissenschaftlichen Erkenntnissen gewonnen.

## Inhalt

Weitgehend selbstständiges Erarbeiten des Seminarthemas:

- Lesen und Verstehen wissenschaftlicher Arbeiten
- Analyse und Bewertung wissenschaftlicher Aufsätze
- Diskutieren der Arbeiten in der Gruppe
- Analysieren, Zusammenfassen und Wiedergeben des spezifischen Themas
- Erarbeiten gemeinsamer Standards für wissenschaftliches Arbeit
- Präsentationstechnik

Spezifische Vertiefung in Bezug auf das individuelle Thema des Seminars.

Der typische Ablauf eines Seminars ist üblicherweise wie folgt:

- Vorbereitende Gespräche zur Themenauswahl
- Regelmäßige Treffen mit Diskussion ausgewählter Beiträge
- ggf. Bearbeitung eines themenbegleitenden Projekts
- Vortrag und ggf. Ausarbeitung zu einem der Beiträge



## **Literaturhinweise**

Material wird dem Thema entsprechend ausgewählt.

## **Weitere Informationen**

Die jeweils zur Verfügung stehenden Seminare werden vor Beginn des Semesters angekündigt und unterscheiden sich je nach Studiengang.

**Modulbereich 8**

---

***Projektseminare DSAI***

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>4</b>	<b>6</b>	<b>jedes Sommersemester</b>	<b>1 Semester</b>	<b>2</b>	<b>9</b>

**Modulverantwortliche/r** Prof. Dr. Jens Dittrich

**Dozent/inn/en** Prof. Dr. Jens Dittrich  
Dozent/inn/en der Fachrichtung

**Zulassungsvoraussetzungen** *Elements of Data Science and Artificial Intelligence, Elements of Machine Learning, Statistics Lab, Programmierung 1 und 2*

**Leistungskontrollen / Prüfungen**

- regelmäßige (mindestens jede zweite Woche) Treffen mit AnwenderInnen und BetreuerInnen,
- Abschlussvortrag am Ende des Semesters,
- Abgabe des erstellten Projektes inklusive Source Code

**Lehrveranstaltungen / SWS** 2 SWS Seminar

**Arbeitsaufwand** 30 h Meetings mit DSAI-AnwenderInnen und BetreuerInnen  
+ 249 h Eigenstudium  
= 270 h (= 9 ECTS)

**Modulnote** unbenotet

**Sprache** Deutsch oder Englisch

## Lernziele / Kompetenzen

- selbständiges Bearbeiten eines kleine DSAI-Projektes
- selbständiges Präsentation von Projektergebnissen
- Arbeit im Team
- Arbeit mit AuftraggeberInnen

## Inhalt

In diesem Projektseminar sollen Studierende im Team von 2-3 Studierenden die DSAI-Aufgabenstellung eines/r DSAI-AnwenderIn bearbeiten.

## Literaturhinweise

Material wird dem Thema entsprechend ausgewählt.



# Bachelor-Seminar

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>6</b>	<b>6</b>	<b>jedes Semester</b>	<b>variabel</b>	<b>2</b>	<b>9</b>

**Modulverantwortliche/r** Studiendekan der Fakultät Mathematik und Informatik  
Studienbeauftragter der Informatik

**Dozent/inn/en** Dozent/inn/en der Fachrichtung

**Zulassungsvoraussetzungen** Erwerb von mindestens 120 CP

**Leistungskontrollen / Prüfungen**

- Schriftliche Ausarbeitung der Aufgabenstellung der Bachelorarbeit und der relevanten wissenschaftlichen Literatur
- Vortrag über die geplante Aufgabenstellung mit anschließender Diskussion
- Aktive Teilnahme an der Diskussion

**Lehrveranstaltungen / SWS** 2 SWS Seminar

**Arbeitsaufwand** 30 h Präsenzstudium (Seminar)  
+ 30 h Betreuung durch den Lehrstuhl  
+ 210 h Eigenstudium  
= 270 h (= 9 ECTS)

**Modulnote** Wird aus den Leistungen im Vortrag und der schriftlichen Ausarbeitung ermittelt. Die genauen Modalitäten werden von dem/der jeweiligen Dozenten/in bekannt gegeben.

**Sprache** Deutsch oder Englisch

## Lernziele / Kompetenzen

Im Bachelorseminar erwirbt der Studierende unter Anleitung die Fähigkeit zum wissenschaftlichen Arbeiten im Kontext eines angemessenen Themengebietes.

Am Ende des Bachelorseminars sind die Grundlagen für eine erfolgreiche Anfertigung der Bachelorarbeit gelegt, und wesentliche Lösungsansätze bereits eruiert.

Das Bachelorseminar bereitet somit die Themenstellung und Ausführung der Bachelorarbeit vor.

Es vermittelt darüber hinaus praktische Fähigkeiten des wissenschaftlichen Diskurses. Diese Fähigkeiten werden durch die aktive Teilnahme an einem Lesekreis vermittelt, in welchem die Auseinandersetzung mit wissenschaftlich anspruchsvollen Themen geübt wird.

## Inhalt

Einarbeitung in ein wissenschaftliches Themengebiet innerhalb der Informatik.

Anfertigung einer schriftlichen Ausarbeitung der Aufgabenstellung der Bachelorarbeit und der relevanten wissenschaftlichen Literatur.

Fachvortrag über das Themengebiet und die geplante Aufgabenstellung der Bachelorarbeit.

Das Thema wird in enger Absprache mit dem anleitenden Dozenten definiert.

## Literaturhinweise

Dem Themengebiet entsprechende wissenschaftliche Artikel in enger Absprache mit dem Dozenten

# Bachelor-Arbeit

Studiensem.	Regelst.sem.	Turnus	Dauer	SWS	ECTS
<b>6</b>	<b>6</b>	<b>jedes Semester</b>	<b>3 Monate</b>	<b>-</b>	<b>12</b>

**Modulverantwortliche/r** Studiendekan der Fakultät Mathematik und Informatik  
Studienbeauftragter der Informatik

**Dozent/inn/en** Professoren der Fachrichtung

**Zulassungsvoraussetzungen** Erfolgreicher Abschluss des *Bachelor-Seminars*

**Leistungskontrollen / Prüfungen** Schriftliche Ausarbeitung. Sie beschreibt sowohl das Ergebnis der Arbeit als auch den Weg, der zu dem Ergebnis führte. Der eigene Anteil an den Ergebnissen muss klar erkennbar sein. Außerdem Präsentation der Bachelorarbeit in einem Kolloquium, in dem auch die Eigenständigkeit der Leistung des Studierenden überprüft wird.

**Lehrveranstaltungen / SWS** keine

**Arbeitsaufwand** 30 h Betreuung durch den Lehrstuhl  
+ 330 h Eigenstudium  
= 360 h (= 12 ECTS)

**Modulnote** Beurteilung der Bachelorarbeit durch die Gutachter.

**Sprache** Deutsch oder Englisch

## Lernziele / Kompetenzen

Die Bachelor-Arbeit ist eine Projektarbeit, die unter Anleitung ausgeführt wird. Sie soll der Kandidaten/die Kandidatin in der Lage versetzen, innerhalb einer vorgegebenen Frist ein Problem aus dem Gebiet der Informatik selbständig zu lösen und die Ergebnisse in wissenschaftlich angemessener Form zu dokumentieren.

## Inhalt

Bearbeitung einer aktuellen Problemstellung aus der Informatik unter Anleitung. Adäquate Dokumentation der Ergebnisse in Form einer wissenschaftlichen Abschlussarbeit.

Das Thema wird in enger Absprache mit dem anleitenden Dozenten definiert.

## Literaturhinweise

Dem Themengebiet entsprechende wissenschaftliche Artikel in enger Absprache mit dem anleitenden Dozenten.