

# Modulhandbuch

für den Studiengang

**Data Science (B.Sc.)**

**Wintersemester 2021/22**

Hinweise:

- Weitere Informationen zu den einzelnen Studiengängen (Studien- und Prüfungsordnungen, Studienberatung, etc.) finden Sie auf [www.math.fau.de/studium](http://www.math.fau.de/studium)
- Semesteraktuelle Informationen zu den angebotenen Lehrveranstaltungen finden Sie im UnivIS-Vorlesungsverzeichnis.
- Module eines Studiengangs sind in der jeweiligen Prüfungsordnung festgelegt. Diese Sammlung umfasst die Module, die vom Department Mathematik in den jeweiligen Studiengängen verwendet werden.

## Inhaltsverzeichnis

Modul AuD-MT-V: Algorithmen und Datenstrukturen für MT .....	4
Modul BFS: Berechenbarkeit und Formale Sprachen.....	6
Modul NumMath: Einführung in die Numerik.....	8
Modul GLoIn: Grundlagen der Logik in der Informatik .....	10
Modul IntroML: Introduction to Machine Learning .....	12
Modul LKOpt: Lineare und Kombinatorische Optimierung.....	14
Modul MDS-1: Mathematik für Data Science 1 .....	16
Modul DSFI: Seminar Data Science in Forschung und Industrie.....	18

1	<b>Modulbezeichnung</b>	<b>Modul AuD-MT-V: Algorithmen und Datenstrukturen für MT</b> (englische Bezeichnung: Algorithms and Data Structures (for Medical Engineering))	<b>ECTS 5</b>
2	<b>Lehrveranstaltungen</b>	Algorithmen und Datenstrukturen (f. Medizintechnik) (V) (4 SWS) Algorithmen und Datenstrukturen (f. Medizintechnik) Tafelübung (Ü) (2 SWS) Algorithmen und Datenstrukturen (f. Medizintechnik) Rechnerübung (Ü) (2 SWS)	
3	<b>Lehrende</b>	Prof. Dr. Tobias Reichenbach <a href="mailto:tobias.j.reichenbach@fau.de">tobias.j.reichenbach@fau.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr. Tobias Reichenbach <a href="mailto:tobias.j.reichenbach@fau.de">tobias.j.reichenbach@fau.de</a>	
5	<b>Inhalt</b>	Die Vorlesung AuD-MT richtet sich an Studierende der Studiengänge Medizintechnik und Data Sciences und zählt dort zu den Grundlagenvorlesungen im Bereich Informatik. Neben einer Einführung in die (objektorientierte) Programmierung in Java werden verschiedene Datenstrukturen wie verkettete Listen, Bäume und Graphen behandelt. Ein weiterer Schwerpunkt liegt auf dem Entwurf von Algorithmen. Dazu zählt Rekursion, Sortierverfahren und Graphalgorithmen, sowie Aufwandsabschätzung von Algorithmen.	
6	<b>Lernziele und Kompetenzen</b>	Die Studierenden <ul style="list-style-type: none"> <li>• lösen objektorientierte Programmieraufgaben in der Programmiersprache Java</li> <li>• veranschaulichen Programmstrukturen mit Hilfe einer Untermenge der Unified Modelling Language</li> <li>• vergleichen die Aufwände verschiedener Algorithmen hinsichtlich der Laufzeit und des Speicherbedarfs</li> <li>• implementieren grundlegende kombinatorische Algorithmen, insbesondere Such- und Sortierverfahren, binäre Bäume und grundlegende Graphalgorithmen</li> <li>• verstehen und benutzen Rekursion als Bindeglied zwischen mathematischen Problembeschreibungen und programmiererischer Umsetzung</li> <li>• übersetzen rekursive Problembeschreibungen in iterative</li> <li>• planen und bearbeiten Programmieraufgaben so, dass sie zeitgerecht fertig gestellt werden</li> </ul>	
7	<b>Voraussetzungen für die Teilnahme</b>	Die Beispiele in der Vorlesung sind nicht Medizintechnik-spezifisch. Vorkenntnisse in dieser Richtung sind nicht notwendig. Der Bezeichnungszusatz „MT“ ist historisch bedingt und dient zur Unterscheidung vom Modul „Algorithmen und Datenstrukturen“.	
8	<b>Einpassung in Musterstudienplan</b>	1. Semester	
9	<b>Verwendbarkeit des Moduls</b>	Pflichtmodul in: <ul style="list-style-type: none"> <li>• B.Sc. Data Science (Grundlagenmodul Informatik)</li> <li>• B.Sc. Medizintechnik</li> <li>• B.Sc. Wirtschaftsinformatik</li> </ul>	
10	<b>Studien- und Prüfungsleistung</b>	Als Prüfungsleistung ist eine Klausur von 120min Dauer geplant. Auf Grund der erwarteten Teilnehmerzahl von über 500 Hörern sind Änderungen im Rahmen der Corona-Satzung vorbehalten.	
11	<b>Berechnung Modulnote</b>	Klausur 100%	
12	<b>Turnus des Angebots</b>	Vorlesung: 1 x jährlich nur im WiSe Übungen: 2 x jährlich im SoSe und WiSe	

13	<b>Arbeitsaufwand</b>	Workload 150 h davon <ul style="list-style-type: none"> <li>• Vorlesung: 4 SWS x 15 = 60h</li> <li>• Rechnerübung: 2 SWS x 15 = 30h</li> <li>• Tafelübung: 2 SWS x 15 = 30h</li> <li>• Selbststudium: 30 h</li> </ul>
14	<b>Dauer des Moduls</b>	1 Semester
15	<b>Unterrichts- und Prüfungssprache</b>	deutsch
16	<b>Literaturhinweise</b>	In der Vorlesung werden zu den einzelnen Kapiteln passende Lehrbücher vorgeschlagen.

1	<b>Modulbezeichnung</b>	<b>Modul BFS: Berechenbarkeit und Formale Sprachen</b> (englische Bezeichnung: Theory of Computation and Formal Languages)	<b>ECTS 7,5</b>
2	<b>Lehrveranstaltungen</b>	Vorlesung Berechenbarkeit und Formale Sprachen (4 SWS) Übung Berechenbarkeit und Formale Sprachen (2 SWS)	ECTS 5 ECTS 2,5
3	<b>Lehrende</b>	Prof. Dr. Rolf Wanka <a href="mailto:rolf.wanka@fau.de">rolf.wanka@fau.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr. Rolf Wanka <a href="mailto:rolf.wanka@fau.de">rolf.wanka@fau.de</a>	
5	<b>Inhalt</b>	<ul style="list-style-type: none"> <li>• Registermaschinen und Turingmaschinen als Modelle des Berechenbaren, die Church-Turing-These und unentscheidbare Probleme</li> <li>• NP-Vollständigkeit und das P-NP-Problem</li> <li>• Endliche Automaten</li> <li>• Grammatiken und die Chomsky-Hierarchie</li> <li>• Kontextfreie Grammatiken und Kontextfreie Sprachen</li> <li>• Kellerautomaten</li> </ul>	
6	<b>Lernziele und Kompetenzen</b>	Die Studierenden <ul style="list-style-type: none"> <li>• erwerben fundierte Kenntnisse über die Grenzen der Berechenbaren, insbesondere lernen sie, wie man beweist, dass bestimmte Aufgaben unlösbar sind bzw. dass sie vermutlich nicht schnell gelöst werden können;</li> <li>• lernen die wesentlichen Techniken kennen, mit denen man Programmiersprachen beschreiben und syntaktisch korrekte Programme erkennen kann;</li> <li>• erwerben fundierte Kenntnisse in den Beweis- und Analyse-Methoden der algorithmisch orientierten Theoretischen Informatik</li> </ul>	
7	<b>Voraussetzungen für die Teilnahme</b>	Grundlagenwissen in Algorithmen, wie es in den Modulen „Algorithmen und Datenstrukturen“ und „Algorithmen und Datenstrukturen für MT“ vermittelt wird.	
8	<b>Einpassung in Musterstudienplan</b>	Ab 3. Semester B.Sc.	
9	<b>Verwendbarkeit des Moduls</b>	Pflichtmodul <ul style="list-style-type: none"> <li>• B. Sc. Informatik, 3. Semester</li> </ul> Wahlpflichtmodul in: <ul style="list-style-type: none"> <li>• B.Sc. Data Science, ab 3. Semester</li> </ul> Vertiefungsmodul in <ul style="list-style-type: none"> <li>• B. Sc. Mathematik (Nebenfach Informatik), ab 3. Semester</li> </ul>	
10	<b>Studien- und Prüfungsleistung</b>	Mindestens 50% der Übungspunkte und Klausur von 90min Dauer.	
11	<b>Berechnung Modulnote</b>	Klausur 100%	
12	<b>Turnus des Angebots</b>	Vorlesung und Übungen: 1 x jährlich nur im WiSe	
13	<b>Arbeitsaufwand</b>	Workload 225 h davon <ul style="list-style-type: none"> <li>• Vorlesung: 4 SWS x 15 = 60 h</li> <li>• Übung: 2SWS x 15 = 30 h</li> <li>• Selbststudium: 135 h</li> </ul>	
14	<b>Dauer des Moduls</b>	1 Semester	
15	<b>Unterrichts- und Prüfungssprache</b>	deutsch	

16	<b>Literaturhinweise</b>	<ul style="list-style-type: none"><li>• I. Wegener. Theoretische Informatik. Teubner</li><li>• J. Hopcroft, J. Ullman. Introduction to Automata Theory, Languages and Computation, Addison-Wesley</li></ul> <p>Weitere Literatur wird in der Vorlesung genannt.</p>
----	--------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1	<b>Modulbezeichnung</b>	<b>Modul NumMath: Einführung in die Numerik</b> (englische Bezeichnung: Numerical Mathematics)	<b>ECTS 10</b>
2	<b>Lehrveranstaltungen</b>	Vorlesung Einführung in die Numerik (4 SWS) Übungen zur Einführung in die Numerik (2 SWS) Tutorium zur Einführung in die Numerik (1 SWS)	
3	<b>Lehrende</b>	Prof. Dr. Martin Burger <a href="mailto:martin.burger@fau.de">martin.burger@fau.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr. Eberhard Bänsch <a href="mailto:baensch@math.fau.de">baensch@math.fau.de</a>	
5	<b>Inhalt</b>	<ul style="list-style-type: none"> <li>• Direkte Eliminationsverfahren für lineare Gleichungssysteme [Gauß mit Pivotsuche (Erinnerung), Cholesky, LR-Zerlegung für vollbesetzte (Erinnerung) Bandmatrizen]</li> <li>• Linear stationäre iterative Verfahren: Erinnerung und SOR-Verfahren</li> <li>• Verfahren für Eigenwertaufgaben (QR-Verfahren)</li> <li>• Fehleranalyse und Störungsrechnung (Gleitpunktarithmetik, Konditionsanalyse, schlechtgestellte Probleme)</li> <li>• Lineare Ausgleichsrechnung (Orthogonalisierungsverfahren, Numerik der Pseudoinverse)</li> <li>• Iterative Verfahren für nicht-lineare Gleichungssysteme (Fixpunktiteration, Newton-Verfahren, Gauß-Newton)</li> <li>• Interpolation (Polynome, Polynomialsplines, FFT)</li> <li>• Numerische Integration (Newton-Cotes, Gauß, Extrapolation, Adaption)</li> </ul> <p>Die Präsentation des Stoffes erfolgt in Vorlesungsform. Die weitere Aneignung der wesentlichen Begriffe und Techniken erfolgt durch wöchentliche Hausaufgaben.</p>	
6	<b>Lernziele und Kompetenzen</b>	<p>Die Studierenden</p> <ul style="list-style-type: none"> <li>• verwenden algorithmische Zugänge für Probleme der linearen Algebra und Analysis und erklären und bewerten diese;</li> <li>• urteilen insbesondere über die Stabilität und Effizienz eines numerischen Verfahrens;</li> <li>• setzen mit eigener oder gegebener Software Verfahren um und bewerten deren Ergebnisse kritisch;</li> <li>• erläutern und verwenden ein breites Problem- und Verfahrensspektrum: (Direkte und) iterative Verfahren für lineare Gleichungssysteme, nicht-lineare Gleichungssysteme, insbesondere Newton-Verfahren, (nicht)lineare Ausgleichsrechnung, Interpolation und Integration, Numerik von Eigenwertaufgaben;</li> <li>• sammeln und bewerten relevante Informationen und erkennen Zusammenhänge</li> </ul>	
7	<b>Voraussetzungen für die Teilnahme</b>	<p>empfohlen:</p> <ul style="list-style-type: none"> <li>• Module zur Analysis und Linearen Algebra</li> <li>• Kenntnisse in MATLAB sind zwingend. Diese können in einem jeweils vor Semesterbeginn stattfindenden Kurs erworben werden.</li> </ul>	



8	<b>Einpassung in Musterstudienplan</b>	ab dem 3. Semester
9	<b>Verwendbarkeit des Moduls</b>	Pflichtmodul in <ul style="list-style-type: none"> <li>• B.Sc. Technomathematik (Aufbaumodul)</li> </ul> Wahlpflichtmodul in <ul style="list-style-type: none"> <li>• B. Sc. Data Science</li> <li>• B. Sc. Mathematik (Angewandte Mathematik)</li> <li>• B.Sc. Wirtschaftsmathematik (Mathematisches Wahlpflichtmodul)</li> </ul>
10	<b>Studien- und Prüfungsleistung</b>	<ul style="list-style-type: none"> <li>• Übungsleistungen (wöchentliche Hausaufgaben, unbenotet)</li> <li>• Klausur (90 Min.)</li> </ul>
11	<b>Berechnung Modulnote</b>	Klausur (100 %)
12	<b>Turnus des Angebots</b>	jährlich im Wintersemester
13	<b>Arbeitsaufwand</b>	Workload 300 h davon <ul style="list-style-type: none"> <li>• Vorlesung: 4 SWS x 15 = 60 h</li> <li>• Übung: 2 SWS x 15 = 30 h</li> <li>• Tutorium: 1 SWS x 15 = 15 h</li> <li>• Selbststudium: 195 h</li> </ul>
14	<b>Dauer des Moduls</b>	ein Semester
15	<b>Unterrichts- und Prüfungssprache</b>	deutsch
16	<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>• R. Schaback und H. Wendland: Numerische Mathematik; Springer, Berlin, 2005</li> <li>• A. Quarteroni, R. Sacco, F. Saleri: Numerische Mathematik I, II; Springer, Berlin, 2002</li> <li>• P. Deuffhard und A. Hohmann: Numerische Mathematik I; de Gruyter, Berlin 2002</li> <li>• J. Stoer: Numerische Mathematik I; Springer, Berlin, 2005</li> <li>• J. Stoer und R. Bulirsch: Numerische Mathematik I; Springer, Berlin, 2005</li> <li>• Vorlesungsskript auf der Homepage des Bereichs Modellierung, Simulation und Optimierung des Departments Mathematik, ständig neu an die Vorlesung angepasst</li> </ul>

1	<b>Modulbezeichnung</b>	<b>Modul GLOIn: Grundlagen der Logik in der Informatik</b>	<b>ECTS 5</b>
2	<b>Lehrveranstaltungen</b>	Vorlesung Grundlagen der Logik in der Informatik (2 SWS) Übung Grundlagen der Logik in der Informatik (2 SWS) InÜ: Intensivübung Grundlagen der Logik in der Informatik (6 SWS)	
3	<b>Lehrende</b>	Prof. Dr. Lutz Schröder <a href="mailto:lutz.schroeder@fau.de">lutz.schroeder@fau.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr. Lutz Schröder <a href="mailto:lutz.schroeder@fau.de">lutz.schroeder@fau.de</a>	
5	<b>Inhalt</b>	<p><b>Aussagenlogik:</b></p> <ul style="list-style-type: none"> <li>• Syntax und Semantik</li> <li>• Automatisches Schließen: Resolution</li> <li>• Formale Deduktion: Korrektheit, Vollständigkeit</li> </ul> <p><b>Prädikatenlogik erster Stufe:</b></p> <ul style="list-style-type: none"> <li>• Syntax und Semantik</li> <li>• Automatisches Schließen: Unifikation, Resolution</li> <li>• Quantorenelimination</li> <li>• Anwendung automatischer Beweiser</li> <li>• Formale Deduktion: Korrektheit, Vollständigkeit</li> </ul>	
6	<b>Lernziele und Kompetenzen</b>	<p>Wissen: Die Studierenden</p> <ul style="list-style-type: none"> <li>• geben Definitionen zur Syntax und Semantik der verwendeten Logiken wieder</li> <li>• beschreiben grundlegende Deduktionsalgorithmen</li> <li>• geben Regeln der verwendeten formalen Deduktionssysteme wieder</li> <li>• Verstehen: Die Studierenden</li> <li>• erläutern das Verhältnis zwischen Syntax, Semantik und Beweistheorie der verwendeten Logiken</li> <li>• erklären die Funktionsprinzipien grundlegender Deduktionsalgorithmen</li> <li>• erläutern die Funktionsweise automatischer Beweiser</li> <li>• erläutern grundlegende Resultate der Metatheorie der verwendeten Logiken und deren Bedeutung</li> <li>• Anwenden: Die Studierenden</li> <li>• wenden Deduktionsalgorithmen auf konkrete Deduktionsprobleme an</li> <li>• formalisieren Anwendungsprobleme in logischer Form</li> <li>• verwenden automatische Beweiser zur Erledigung entstehender Beweisziele</li> <li>• führen einfache formale Beweise manuell</li> <li>• Analysieren: Die Studierenden</li> <li>• führen einfache metatheoretische Beweise, insbes. durch syntaktische Induktion</li> <li>• Lern- bzw. Methodenkompetenz: Die Studierenden</li> <li>• beherrschen das grundsätzliche Konzept des Beweises als hauptsächliche Methode des Erkenntnisgewinns in der theoretischen Informatik. Sie überblicken abstrakte Begriffsarchitekturen.</li> <li>• Sozialkompetenz: Die Studierenden</li> <li>• lösen abstrakte Probleme in Gruppenarbeit</li> </ul>	
7	<b>Voraussetzungen für die Teilnahme</b>	keine	
8	<b>Einpassung in Musterstudienplan</b>	3. Semester	

9	<b>Verwendbarkeit des Moduls</b>	Bachelor Informatik <ul style="list-style-type: none"> <li>Studierende aller Fächer / Studium Generale: als Schlüsselqualifikation geeignet</li> </ul>
10	<b>Studien- und Prüfungsleistung</b>	Die Modulprüfung besteht aus einer 90-minütigen Klausur. Die im Rahmen der Übungen gestellten Übungsaufgaben können abgegeben werden und werden in diesem Fall bewertet. Auf Basis des Ergebnisses dieser Bewertungen können bis zu 15 % Bonuspunkte erworben werden, die zu dem Ergebnis einer bestandenen Klausur hinzugerechnet werden.
11	<b>Berechnung Modulnote</b>	100 % der schriftl. Prüfungsnote. In die Prüfungsnote fließen Leistungen aus dem Übungsbetrieb in Form eines Bonus mit ein (s.o.).
12	<b>Turnus des Angebots</b>	1 x jährlich jeweils im WiSe
13	<b>Arbeitsaufwand</b>	Workload 150h davon <ul style="list-style-type: none"> <li>Vorlesung: 2 SWS x 15 = 30 h</li> <li>Übung: 2 SWS x 15 = 30 h</li> <li>Eigenstudium: 90 h</li> </ul>
14	<b>Dauer des Moduls</b>	1 Semester
15	<b>Unterrichts- und Prüfungssprache</b>	deutsch
16	<b>Literaturhinweise</b>	Geeignete Literatur wird in der Vorlesung bekannt gegeben.

1	<b>Modulbezeichnung</b>	<b>Modul IntroML: Introduction to Machine Learning</b>	<b>ECTS 5</b>
2	<b>Lehrveranstaltungen</b>	V: Introduction to Machine Learning (2 SWS) Ü: Introduction to Machine Learning Exercises (2 SWS)	ECTS 3,75 ECTS 1,25
3	<b>Lehrende</b>	Dr.-Ing. Vincent Christlein <a href="mailto:vincent.christlein@fau.de">vincent.christlein@fau.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr.-Ing. Andreas Maier <a href="mailto:andreas.maier@fau.de">andreas.maier@fau.de</a>	
5	<b>Inhalt</b>	Die Vorlesung hat zum Ziel, die Studierenden mit dem prinzipiellen Aufbau eines Mustererkennungssystems vertraut zu machen. Es werden die einzelnen Schritte von der Aufnahme der Daten bis hin zur Klassifikation von Mustern erläutert. Die Vorlesung beginnt dabei mit einer kurzen Einführung, bei der auch die verwendete Nomenklatur eingeführt wird. Die Analog-Digital-Wandlung wird vorgestellt, wobei der Schwerpunkt auf deren Auswirkungen auf die weitere Signalanalyse liegt. Im Anschluss werden gebräuchliche Methoden der Vorverarbeitung beschrieben. Ein wesentlicher Bestandteil eines Mustererkennungssystems ist die Merkmalsextraktion. Verschiedene Ansätze zur Merkmalsberechnung/-transformation werden gezeigt, darunter Momente, Hauptkomponentenanalyse und Lineare Diskriminanzanalyse. Darüber hinaus werden Möglichkeiten vorgestellt, Merkmalsrepräsentationen direkt aus den Daten zu lernen. Die Vorlesung schließt mit einer Einführung in die maschinelle Klassifikation. In diesem Kontext wird der Bayes- und der Gauss-Klassifikator besprochen.	
6	<b>Lernziele und Kompetenzen</b>	Die Studierenden <ul style="list-style-type: none"> <li>• erklären die Stufen eines allgemeinen Mustererkennungssystems</li> <li>• verstehen Abtastung, das Abtasttheorem und Quantisierung</li> <li>• verstehen und implementieren Histogrammequalisierung und -dehnung</li> <li>• vergleichen verschiedene Schwellwertmethoden</li> <li>• verstehen lineare, verschiebungsinvariante Filter und Faltung</li> <li>• wenden verschiedene Tief- und Hochpassfilter sowie nichtlineare Filter an</li> <li>• wenden verschiedene Normierungsmethoden an</li> <li>• verstehen den Fluch der Dimensionalität</li> <li>• erklären verschiedene heuristische Merkmalsberechnungsmethoden, z.B. Projektion auf einen orthogonalen Basisraum, geometrische Momente, Merkmale basierend auf Filterung</li> <li>• verstehen analytische Merkmalsberechnungsmethoden, z.B. Hauptkomponentenanalyse, Lineare Diskriminanzanalyse</li> <li>• verstehen die Basis von Repräsentationslernen</li> <li>• erläutern die Grundlagen der statistischen Klassifikation (Bayes-Klassifikator)</li> <li>• benutzen die Programmiersprache Python, um die vorgestellten Verfahren der Mustererkennung anzuwenden</li> <li>• lernen praktische Anwendungen kennen und wenden die vorgestellten Algorithmen auf konkrete Probleme an</li> </ul>	
7	<b>Voraussetzungen für die Teilnahme</b>	Ein Mustererkennungssystem besteht aus den folgenden Stufen: Aufnahme von Sensordaten, Vorverarbeitung, Merkmalsextraktion und maschinelle Klassifikation. Diese Vorlesung beschäftigt sich in erster Linie mit den ersten drei Stufen und schafft damit die Grundlage für weiterführende Lehrveranstaltungen (Pattern Recognition und Pattern Analysis).	
8	<b>Einpassung in Musterstudienplan</b>	ab 3. Semester B.Sc.	

9	<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul in: <ul style="list-style-type: none"> <li>• B.Sc. Data Science, ab 3. Semester</li> </ul> M.Sc. Informatik, ab 1. Semester Vertiefungsmodul in <ul style="list-style-type: none"> <li>• B. Sc. Informatik, ab 5. Semester</li> </ul>
10	<b>Studien- und Prüfungsleistung</b>	Bestehen der Klausur
11	<b>Berechnung Modulnote</b>	Klausur 100%
12	<b>Turnus des Angebots</b>	Vorlesung und Übungen: 1 x jährlich nur im WiSe
13	<b>Arbeitsaufwand</b>	Workload 150 h davon <ul style="list-style-type: none"> <li>• Vorlesung: 2 SWS x 15 = 30 h</li> <li>• Übung: 2 SWS x 15h = 30 h</li> <li>• Selbststudium: 90 h</li> </ul>
14	<b>Dauer des Moduls</b>	1 Semester
15	<b>Unterrichts- und Prüfungssprache</b>	englisch
16	<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>• Vorlesungsfolien</li> <li>• Heinrich Niemann: Klassifikation von Mustern, 2. überarbeitete Auflage, 2003</li> <li>• Sergios Theodoridis, Konstantinos Koutroumbas: Pattern Recognition, 4. Auflage, Academic Press, Burlington, 2009</li> <li>• Richard O. Duda, Peter E. Hart, David G. Stock: Pattern Classification, 2. Auflage, John Wiley &amp; Sons, New York, 2001</li> </ul>

1	<b>Modulbezeichnung</b>	<b>Modul LKOpt: Lineare und Kombinatorische Optimierung</b> (englische Bezeichnung: Linear and Combinatorial Optimization)	<b>ECTS 10</b>
2	<b>Lehrveranstaltungen</b>	Vorlesung Lineare und Kombinatorische Optimierung (4 SWS) Übungen zur Linearen und Kombinatorischen Optimierung (2 SWS)	
3	<b>Lehrende</b>	Dr. Dieter Weninger <a href="mailto:dieter.weninger@math.uni-erlangen.de">dieter.weninger@math.uni-erlangen.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr. Alexander Martin <a href="mailto:alexander.martin@math.uni-erlangen.de">alexander.martin@math.uni-erlangen.de</a>	
5	<b>Inhalt</b>	Schwerpunkt dieser Vorlesung ist die Theorie und Lösung kombinatorischer und in diesem Kontext linearer Optimierungsprobleme. Wir behandeln klassische Probleme auf Graphen, wie das Kürzeste-Wege-Problem, das Aufspannende-Baum-Problem oder das Max-Flow-Min-Cut-Theorem. Zum Vorlesungsumfang gehört auch das Simplexverfahren für lineare Programme und das Studium algorithmischer Grundprinzipien wie Sortieren, Greedy, Tiefen- und Breitensuche sowie Heuristiken  Neben der vierstündigen Vorlesung werden zweistündige Übungen angeboten. Anhand von Präsenz- und Hausaufgaben werden wesentliche Lerninhalte geübt. Zusätzlich werden kleinere Softwareübungen angeboten.	
6	<b>Lernziele und Kompetenzen</b>	Die Studierenden <ul style="list-style-type: none"> <li>• erkennen und analysieren selbstständig kombinatorische Optimierungsprobleme;</li> <li>• erläutern algorithmische Grundprinzipien und wenden diese zielorientiert an;</li> <li>• klassifizieren komplexe Verfahren des Lerngebietes;</li> <li>• sammeln und bewerten relevante Informationen und stellen Zusammenhänge her</li> </ul>	
7	<b>Voraussetzungen für die Teilnahme</b>	empfohlen: Lineare Algebra	
8	<b>Einpassung in Musterstudienplan</b>	ab dem 3. Semester	
9	<b>Verwendbarkeit des Moduls</b>	Pflichtmodul in <ul style="list-style-type: none"> <li>• B. Sc. Wirtschaftsmathematik (Aufbaumodul)</li> </ul> Wahlpflichtmodul in <ul style="list-style-type: none"> <li>• B. Sc. Mathematik (Angewandte Mathematik)</li> <li>• B.Sc. Technomathematik (Numerische Mathematik, Modellierung und Optimierung)</li> </ul>	
10	<b>Studien- und Prüfungsleistung</b>	<ul style="list-style-type: none"> <li>• Klausur (90 Min.)</li> </ul>	
11	<b>Berechnung Modulnote</b>	Klausur (100 %)	
12	<b>Turnus des Angebots</b>	jährlich im Wintersemester	

13	<b>Arbeitsaufwand</b>	Workload 300 h davon <ul style="list-style-type: none"> <li>• Vorlesung: 4 SWS x 15 = 60 h</li> <li>• Übung: 2 SWS x 15 = 30 h</li> <li>• Selbststudium: 210 h</li> </ul>
14	<b>Dauer des Moduls</b>	ein Semester
15	<b>Unterrichts- und Prüfungssprache</b>	deutsch
16	<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>• Vorlesungsskript zu diesem Modul</li> <li>• Schrijver: Combinatorial Optimization Vol. A – C; Springer, 2003</li> <li>• Korte, J. Vygen: Combinatorial Optimization; Springer, 2005</li> </ul>

1	<b>Modulbezeichnung</b>	<b>Modul MDS-1: Mathematik für Data Science 1</b> (englische Bezeichnung: Mathematics for Data Science 1)	<b>ECTS 10</b>
2	<b>Lehrveranstaltungen</b>	Vorlesung Mathematik für Data Science 1 (V) Übung Mathematik für Data Science 1 (Ü) Tafelübung Mathematik für Data Science 1 (Ü)	
3	<b>Lehrende</b>	Prof. Dr. Jan Heiland <a href="mailto:jan.heiland@fau.de">jan.heiland@fau.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr. Martin Burger <a href="mailto:martin.burger@fau.de">martin.burger@fau.de</a>	
5	<b>Inhalt</b>	<p><b>Analysis I:</b></p> <ul style="list-style-type: none"> <li>• Naive Mengenlehre und Logik</li> <li>• Grundeigenschaften der natürlichen, rationalen und reellen Zahlen: Vollständige Induktion, Körper- und Anordnungsaxiome, Vollständigkeit, untere / obere Grenzen, Dichtheit von <math>\mathbb{Q}</math> in <math>\mathbb{R}</math>, abzählbare und überabzählbare Mengen</li> <li>• Komplexe Zahlen: Rechenregeln und ihre geometrische Interpretation, quadratische Gleichungen</li> <li>• Konvergenz, Cauchy-Folgen, Vollständigkeit</li> <li>• Zahlenfolgen und Reihen: Konvergenzkriterien und Rechenregeln, absolute Konvergenz, Potenzreihen, unendliche Produkte</li> <li>• Elementare Funktionen, rationale Funktionen, Potenzen mit reellen Exponenten, Exponentialfunktion, Hyperbelfunktionen, trigonometrische Funktionen, Monotonie und Umkehrfunktion, Logarithmus</li> <li>• Stetige reellwertige Funktionen: Zwischenwertsatz, Existenz von Minimum und Maximum auf kompakten Mengen, stetige Bilder von Intervallen und Umkehrbarkeit, gleichmäßige Stetigkeit, gleichmäßige Konvergenz</li> <li>• Differential- und Integralrechnung in einer reellen Veränderlichen: Rechenregeln für Differentiation, Mittelwertsatz der Differentialrechnung, Taylorformel, Extremwerte und Kurvendiskussion, Definition des Integrals und Rechenregeln, gliedweise Differentiation, Hauptsatz der Differential- und Integralrechnung, Mittelwertsatz der Integralrechnung</li> </ul> <p><b>Lineare Algebra I:</b></p> <ul style="list-style-type: none"> <li>• Lineare Gleichungssysteme</li> <li>• Vektorräume</li> <li>• Euklidische Vektorräume (Orthonormalisierung, Orthogonalprojektion)</li> <li>• Lineare Abbildungen</li> <li>• Gruppen und Körper</li> <li>• Lineare Abbildungen, Matrizen, Gauss-Algorithmus, Determinanten, Eigenwerte und Eigenvektoren,</li> <li>• Diagonalisierung Hauptachsentransformation</li> <li>• Elemente der numerischen linearen Algebra (LR und QR-Zerlegung)</li> </ul> <p>Die Präsentation des Stoffes erfolgt in Vorlesungsform. Die weitere Aneignung der wesentlichen Begriffe und Techniken erfolgt durch wöchentliche Hausaufgaben.</p>	



6	<b>Lernziele und Kompetenzen</b>	Die Studierenden <ul style="list-style-type: none"> <li>• definieren und erklären grundlegende Begriffe der Analysis und linearen Algebra;</li> <li>• diskutieren einfache Funktionen;</li> <li>• bewerten Folgen und Reihen;</li> <li>• analysieren lineare Abbildungen und Matrizen;</li> </ul> reproduzieren grundlegende Prinzipien und Techniken.
7	<b>Voraussetzungen für die Teilnahme</b>	
8	<b>Einpassung in Musterstudienplan</b>	1. Semester
9	<b>Verwendbarkeit des Moduls</b>	Pflichtmodul: <ul style="list-style-type: none"> <li>• B.Sc. Data Science (Grundlagenmodul)</li> </ul>
10	<b>Studien- und Prüfungsleistung</b>	Klausur (120 min.) und praktische Übungsleistung (pÜL)
11	<b>Berechnung Modulnote</b>	Klausur 80% und praktische Übungsleistung 20%
12	<b>Turnus des Angebots</b>	1 x jährlich jeweils im WiSe
13	<b>Arbeitsaufwand</b>	Workload 300 h davon <ul style="list-style-type: none"> <li>• Vorlesung: 4 SWS x 15 = 60h</li> <li>• Übung: 2 SWS x 15 = 30h</li> <li>• Tafelübung: 2 SWS x 15 = 30h</li> <li>• Selbststudium: 180 h</li> </ul>
14	<b>Dauer des Moduls</b>	1 Semester
15	<b>Unterrichts- und Prüfungssprache</b>	deutsch
16	<b>Literaturhinweise</b>	<ul style="list-style-type: none"> <li>• O. Forster: Analysis 1</li> <li>• S. Hildebrandt: Analysis I</li> <li>• G. Fischer: Lineare Algebra</li> </ul>

1	<b>Modulbezeichnung</b>	<b>Modul DSFI: Seminar Data Science in Forschung und Industrie</b> (Seminar Data Science in Research and Industry)	<b>ECTS 5</b>
2	<b>Lehrveranstaltungen</b>	Seminar Data Sciences in Forschung und Industrie (S) (2 SWS)	
3	<b>Lehrende</b>	Prof. Dr. Frauke Liers <a href="mailto:frauke.liers@math.uni-erlangen.de">frauke.liers@math.uni-erlangen.de</a> Dr. Andreas Bäermann <a href="mailto:andreas.baermann@math.uni-erlangen.de">andreas.baermann@math.uni-erlangen.de</a>	
4	<b>Modulverantwortung</b>	Prof. Dr. Frauke Liers <a href="mailto:frauke.liers@math.uni-erlangen.de">frauke.liers@math.uni-erlangen.de</a>	
5	<b>Inhalt</b>	<ul style="list-style-type: none"> <li>• Anwendungsperspektiven von Data Science</li> <li>• Überblick über relevante Data Science Industriebranchen und die Berufsaussichten eines Data Scientists</li> <li>• Fragestellungen und Diskussionen mit den derzeit verwendeten Lösungsansätzen im Anwendungskontext der Data Sciences (mit Expertinnen und Experten aus verschiedenen Bereichen z.B. der Naturwissenschaften, Technikwissenschaften, Geisteswissenschaften, Wirtschaftswissenschaften, etc.)</li> </ul>	
6	<b>Lernziele und Kompetenzen</b>	<p>Die Studierenden</p> <ul style="list-style-type: none"> <li>• kennen Data Science Anwendungsfelder und die in verschiedenen Kontexten auftretenden Bereiche</li> <li>• Sie kennen die Berufsanforderungen eines Data Scientists und können diese wiedergeben.</li> <li>• Können strukturiert mit Expertinnen und Experten aus verschiedenen Bereichen z.B. der Naturwissenschaften, Technikwissenschaften, Geisteswissenschaften, Wirtschaftswissenschaften diskutieren</li> </ul>	
7	<b>Voraussetzungen für die Teilnahme</b>		
8	<b>Einpassung in Musterstudienplan</b>	1. Semester	
9	<b>Verwendbarkeit des Moduls</b>	Pflichtmodul in: <ul style="list-style-type: none"> <li>• B. Sc. Data Sciences (Kernmodul)</li> </ul>	
10	<b>Studien- und Prüfungsleistung</b>	Klausur mit Multiple-Choice (90 min.)	
11	<b>Berechnung Modulnote</b>	Klausur 100%	
12	<b>Turnus des Angebots</b>	1 x jährlich im WiSe	
13	<b>Arbeitsaufwand</b>	Workload 150 h davon Präsenzzeit: 60 h <ul style="list-style-type: none"> <li>• Seminar: 2 SWS x 15 = 30 'h</li> </ul> Eigenstudium: 90 h	
14	<b>Dauer des Moduls</b>	1 Semester	
15	<b>Unterrichts- und Prüfungssprache</b>	deutsch	
16	<b>Literaturhinweise</b>	wird von den jeweiligen Dozentinnen/Dozenten bekannt gegeben.	