

# Stochastik für Informatiker

## Eine Vorlage für Texte



Hochschule Kaiserslautern

Manfred Brill

Sommersemester 2018



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Pakete und Style Files</b>	<b>3</b>
2.1	Aufbau der Hauptdatei . . . . .	3
2.2	Die Datei setup.tex . . . . .	5
2.2.1	Verwendete L <sup>A</sup> T <sub>E</sub> X-Pakete . . . . .	5
2.2.2	Einstellungen und Funktionen . . . . .	6
<b>3</b>	<b>Ein Kapitel mit Beispielen</b>	<b>11</b>
3.1	Marginalien . . . . .	11
3.2	Lese-Anleitungen . . . . .	12
3.3	Kontrollfragen und Übungsaufgaben . . . . .	13
3.3.1	Kontrollfragen . . . . .	13
3.3.2	Übungsaufgaben . . . . .	14
<b>4</b>	<b>Bewegliche Objekte</b>	<b>19</b>
4.1	Abbildungen . . . . .	19
4.2	Tabellen . . . . .	22
<b>5</b>	<b>Folien mit dem Beamer-Package</b>	<b>23</b>
<b>6</b>	<b>Verzeichnisse und Index</b>	<b>31</b>
6.1	Literaturverzeichnis . . . . .	31
6.2	Index . . . . .	32
	<b>Literaturverzeichnis</b>	<b>33</b>
	<b>Index</b>	<b>37</b>



# Kapitel 1

## Einleitung

Dieses Dokument ist eine kurze Einführung in eine Vorlage, die für Lehrbriefe in den Modulen *Stochastik für Informatiker*, *Software Management Grundlagen*, *Software Qualitätsmanagement* und *Datenanalyse mit R* erstellt wurde. Grundlage der Vorlage waren Empfehlungen für die Formatierung von Lehrbriefen der ZFH für Microsoft Word ([ZFH11]).

Das Dokument setzt voraus, dass Kenntnisse in  $\text{\LaTeX}$  vorhanden sind. Für das Literaturverzeichnis wird BibTeX eingesetzt. Weitere Hinweise zu  $\text{\LaTeX}$  findet man in [MG05] und [Kop02].

Getestet wurden die Einstellungen auf Windows 10 mit MiKTeX 2.9. Die Version aus dem Jahr 2015 wurde auch auf Fedora 17 mit TeXLive getestet.



# Kapitel 2

## Pakete und Style Files

### 2.1 Aufbau der Hauptdatei

Um den Header der Haupt-Datei nicht unnötig lang zu machen wurden alle Einstellungen in die Datei `setup.tex` verschoben. Dort werden die nötigen Pakete geladen und alle weiteren Einstellungen durchgeführt.

Angaben zum Semester, zum Modul oder zum Dozenten sind in einer vorher zu ladenden Datei definiert. In diesem vorliegenden Dokument wird die Datei `stammdatens.tex` verwendet. An dieser Stelle werden auch die Pfade für die Bilder und die Übungsaufgaben definiert. Hier die Werte die für dieses Dokument verwendet wurden:

```
\newcommand{\theSemester}{Sommersemester~2018}
% Variable für die Vorlesung
\newcommand{\theClass}{Stochastik~für~Informatiker}
% Variable für den Studiengang
\newcommand{\theCourse}{IT-Analyst}
% Variable für den Hochschul-Namen
\newcommand{\theSchool}{Hochschule Kaiserslautern}
% Variable für den Dozenten
\newcommand{\theTeacher}{Manfred~Brill}

% Verzeichnis für die Übungsaufgaben
\newcommand{\exercisePath}{../../Uebungsaufgaben/Stochastik}
% Verzeichnis für Bilder
\newcommand{\imagePath}{../../images}
% Verzeichnis für allgemeine Bilder
\newcommand{\generalImagePath}{../../generalimages}
% Default-Titelbild der Lehrveranstaltung
```

```
\newcommand{\titleImage}{\imagePath/Misc/statistiklogo}
```

Makros und Funktionen verwenden diese Variablen, so dass Namen, Semester und anderes einfach zu verändern ist, indem man diese Datei verändert. Die hier dargestellten Werte findet man unter anderem auf der Titelseite.

Weiter gibt es die Datei `pdfsetup.tex`, in der Angaben für die PDF-Dateien gemacht werden. Insbesondere wird mit `\hypersetup` eingestellt, dass Links in grün im PDF dargestellt werden, und dass URLs im Text mit Zeilenumbruch dargestellt werden:

```
\usepackage[breaklinks]{hyperref}
\usepackage{breakurl}

\hypersetup{
  pdftitle = {\theCourse{} -- \theClass{}},
  pdfauthor = {\theTeacher{}},
  pdfsubject = {\theCourse{}, \theClass{} Studienbrief},
  pdfkeywords = {\theCourse{}, \theSchool{}},
  pdfdisplaydoctitle = true,
  pdfpagemode = UseThumbs,
  colorlinks = false,
  linkcolor = green,
  linkbordercolor = {0 1 0},
  pdfpagelayout = {SinglePage}
}

\urlstyle{same}
```

Vom Ergebnis kann man sich im Acrobat Reader überzeugen, die Angaben in der Datei werden in die Eigenschaften der PDF-Datei geschrieben. Wichtig ist, dass die Funktion `\hypersetup` als letztes im Header durchgeführt wird.

Die Files für die Erstellung eines Studienbriefs oder ähnlicher Dokumente für die Lehre sind immer auf die gleiche Weise organisiert. Im Stammverzeichnis befinden sich

- `setup.tex`,
- `pdfsetup.tex`.



Die Hauptdatei für ein Dokument steht im Verzeichnis `main`. Spezifische Variable werden in der Datei `stammdaten.tex` definiert, die vor `setup.tex` ins Dokument eingefügt wird.

Der Header dieses Dokuments lautet folgendermaßen:

```
\documentclass[enabledeprecatfontcommands, fontsize=12pt,
  open=right, a4paper,
  twoside, DIV=11,
  abstractoff,
  headsepline,
  numbers=noenddot,
  BCOR=15mm,
  headings=standardclasses,
  headings=big]{scrbook}
\KOMAOptions{cleardoublepage=empty}
% Header für Variablen wie Vorlesung, Semester,
% Pfade zu Aufgaben, Bildern, ...
\input{stammdaten}
% Einstellungen
\input{../setup}
% Spezifische Definitionen für die Stochastik
\input{LVDefinitionen}
%
\input{../pdfsetup}
%
```

Die einzelnen Kapitel der Dokumente werden mit `\input` in das Dokument eingefügt. Dabei stehen die einzelnen Kapitel in einem Verzeichnis `chapter`, das je nach Lehrveranstaltung weitere Unterverzeichnisse enthält.

## 2.2 Die Datei setup.tex

### 2.2.1 Verwendete L<sup>A</sup>T<sub>E</sub>X-Pakete

Die Datei `setup.tex` verwendet style files, die im Rahmen von Büchern entstanden sind: `mb.sty` und `mbmath.sty`. Eine Dokumentation dazu findet man in [Bri13a, Bri13b]. In `mb.sty` werden die grundlegenden Pakete geladen, Fonts festgelegt und so weiter. In `mbmath.sty` gibt es auf der Basis der AMS-Pakete weitere Definitionen für Funktionen und andere mathe-

matischen mathematischen Symbole. Werden diese nicht benötigt kann man diesen Eintrag löschen.

Darüber hinaus werden die folgenden Pakete geladen:

- `textcomp`
- `array`
- `ifthen`
- `pifont`
- `pstricks`
- `enumitem` mit der Option `shortlabels`
- `imakeidx`, mit  

```
\makeindex[title=Index,columns=2,options=-s german,intoc=true]
```
- `sclayer-scrpage` mit den Optionen `automark` und `headsepline`
- `hyperref`

In `\makeindex` wird der Index so eingestellt, dass er mit deutschen Umlauten arbeiten kann und zweispaltig formatiert wird. Je nachdem wie der Index erstellt werden soll kann man dieses Paket weglassen und mit dem „normalen“ Index arbeiten.

**Tipp:**

Wichtig ist, dass `hyperref` als *letztes* Paket geladen wird. Deshalb die Auslagerung für dieses Paket in die Datei `pdfsetup.tex`.

## 2.2.2 Einstellungen und Funktionen

Nach dem Laden der  $\text{\LaTeX}$ -Pakete werden noch einige Dateien geladen, die entweder im aktuellen Verzeichnis oder zentral, in `texmf-local`, liegen können. Aktuell werden die folgenden Dateien geladen:

- `variablen` für Begriffe wie Programmiersprachen, Software-Pakete, Versionen und anderen Angaben,

- `colors` für Farben (z.b. die aktuelle Fachbereichsfarbe), und
- `coordinateSystems` für  $\text{\LaTeX}$ -Makros zum Zeichnen von Koordinatensystemen mit  $\text{\LaTeX}$ -Vektorgrafik.

Beispiele für Definitionen, die in diesen Dateien enthalten sind findet man weiter unten im Text.

Mit `\raggedbottom` wird eingestellt, dass  $\text{\LaTeX}$  die Zeilen nicht auseinander zieht um eine Seite zu füllen. Diese Einstellung stammt aus der Zusammenarbeit mit dem Hanser-Verlag. Damit erhält man gleiche Zeilenabstände. Verwendet man den  $\text{\LaTeX}$ -Default werden die Zeilen etwas verzogen, um auf jede Seite den exakt gleichen Satzspiegel zu haben. Das mögen Buchbinder gar nicht. Auch Absatzabstände, Einzüge und Einstellungen zu *Hurenkindern* und *Schusterjungen* – Begriffe aus dem Buchdruck, die sich auf Absatz-Beginn und -ende beziehen, die nur aus einer Zeile bestehen.

**Tipp:**

Die kursiv gedruckten Begriffe im letzten Absatz wurden mit der Funktion `\begriff` formatiert. Mit dieser Funktion wird nicht nur hervorgehoben, sondern auch ein Index-Eintrag erstellt!

Die Tiefe des Inhaltsverzeichnisses wird auf 2 eingestellt. Mit

```
\renewcommand{\topfraction}{0.99}  
\renewcommand{\textfraction}{0.001}
```

wird gewährleistet, dass eine Seite bis zu 99% aus einer Abbildung bestehen kann. Dies stammt aus Texten zur Computergrafik und ist bei sehr vielen Abbildungen in einem Text äußerst nützlich.

Dann erfolgen die Einstellung für Kopf- und Fußzeilen:

```
\pagestyle{scrheadings}  
\clearscrheadfoot  
\ihead{\headmark}  
\ohead[]{\pagemark}  
\chead{}  
\pagestyle{scrheadings}
```

Quelltexte wie der gerade gezeigte werden mit Hilfe des Pakets `lstlisting` ausgegeben. Gesetzt werden Eigenschaften für die Sprachen Java und C++, die Hintergrundfarbe aus ein 85-% Grau und weitere Eigenschaften.

**Tipp:**

Java und C++ wurden mit den Funktionen `\java` und `\cpp` aus der Datei `variablen.tex` gesetzt!

In der Datei werden eine Reihe von Funktionen definiert, um Formatierungen einheitlich zu gestalten. Es gibt Funktionen für die Formatierung von Tasten auf dem Keyboard, von Namen, von Dateinamen, die in Tabelle 2.1 zusammengefasst sind. Diese wurden auch bereits im Text verwendet.

**Tabelle 2.1:** Funktionen in `setup.tex`

Bedeutung	Beispiel mit Aufruf
Taste auf dem Keyboard	<code>\taste{Esc}</code> ergibt Esc
Datei- oder Verzeichnis-Name	<code>\datei{setup.tex}</code> ergibt <code>setup.tex</code>
(Geschützte) Namen	<code>\name{Microsoft}</code> ergibt <i>Microsoft</i> Nicht in einer Überschrift verwenden!
Begriffe beim ersten Erwähnen kursiv formatieren und in den Index eintragen	<code>\begriff{Beispiel}</code> ergibt <i>Beispiel</i> und einen Index-Eintrag

Mit `\tip{Text}` kann eine Box ausgegeben werden, in der Tipps für die Leser ausgegeben werden, wie das folgende Beispiel zeigt.

**Tipp:**

In dieser Box steht ein Tipp, der mit `\tip` erzeugt wurde!

Für die Darstellung von Algorithmen gibt es die Umgebung `\algorithmus`. Sie erhält zwei Argumente, eine Überschrift und das eigentliche Verfahren, mit den verschiedenen durchgeführten Schritten. Dafür wird die Nummerierung `schritte` definiert und verwendet.

Hier ein Beispiel aus der Mathematik:

```
\algorithmus{LU}--Zerlegung ohne Zeilenvertauschungen
{
\begin{schritte}
\item Reduzieren Sie  $SA$  ohne Zeilenvertauschung auf
```

```

Stufenform  $U$  und speichern Sie die Multiplikatoren,
die die führenden Einsen und die Nullen unterhalb
der Diagonale erzeugen.
\item Belegen Sie die Diagonale von  $L$  mit den Kehrwerten
der Multiplikatoren, die an der entsprechenden Position
in  $U$  eine führende Eins geliefert haben.
\end{schritte}
}

```

Und hier das Ergebnis:

### LU-Zerlegung ohne Zeilenvertauschungen

Schritt 1: Reduzieren Sie  $A$  ohne Zeilenvertauschung auf Stufenform  $U$  und speichern Sie die Multiplikatoren, die die führenden Einsen und die Nullen unterhalb der Diagonale erzeugen.

Schritt 2: Belegen Sie die Diagonale von  $L$  mit den Kehrwerten der Multiplikatoren, die an der entsprechenden Position in  $U$  eine führende Eins geliefert haben.

Für die Gestaltung einer Titelseite gibt es die Funktion `\titelseite`. Abhängig von der Variablen `\titleImage` wird ein Default-Bild auf der Seite eingefügt. Diese Defaulteinstellung kann wie unten im Beispiel gezeigt mit Hilfe eines optionalen Arguments verändert werden.

```
\titelseite{<Pfad zu Bild>}{Dokumentation}
```

Das zweite Argument ist der Titel des Dokuments. Die Titelseite dieses Dokuments wurde durch

```

\titelseite[\imagePath/Misc/tugboat]
{Eine Vorlage für Studienbriefe}

```

erzeugt. Es muss keine Endung angegeben werden.  $\LaTeX$  sucht dann, je nachdem wie wir die PDF-Datei erzeugen, das passende Format aus. Durch diese beiden Funktionen wird gewährleistet, dass die Titelseiten aller Studienbriefe zu einer Lehrveranstaltung immer das gleiche Layout aufweisen.

Der Rest der Definitionen in `setup.tex` realisiert Vorschläge für die Gestaltung von Lehrbriefen. Das folgende Kapitel zeigt die Verwendung dieser Funktionen und Umgebungen an Hand von Beispielen.



# Kapitel 3

## Ein Kapitel mit Beispielen

### 3.1 Marginalien

Die ZFH schlägt vor *Marginalien* zu verwenden. Dadurch wird das Wiederfinden von Begriffen und die Navigation im Dokument erleichtert. Um solche Marginalien zu unterstützen sieht der Satzspiegel einen relativ großen äußeren Rand vor, was dem Leser sicher schon aufgefallen ist. Die Marginalien der ZFH sollen fett gedruckt werden; dafür gibt es die Funktion `\randnotiz{Randnotiz}`. Das Ergebnis sieht man hier am Rand.

**Randnotiz**

#### **Tipp:**

Die Funktion sollte nicht zu spät im Absatz platziert werden. Die Marginalie wandert mit, was bei Seitenumbrüchen ungünstig ist!

### Praxisbezug

Für Abschnitte, die einen Bezug zur Praxis darstellen gibt es eine spezielle Marginalie mit dem Zeichen wie in Abbildung 3.1. Dieser Abschnitt ist bereits so markiert worden. Hier der Absatz als L<sup>A</sup>T<sub>E</sub>X-Quelle:



```
Für Abschnitte, die \praxisbezug{}  
einen Bezug zur Praxis darstellen gibt es eine spezielle  
Marginalie mit dem Zeichen wie in  
Abbildung \ref{praxisbezug_icon}.  
Dieser Abschnitt ist bereits so markiert worden.
```



Abbildung 3.1: Die Grafik für den „Praxisbezug“

## 3.2 Lese-Anleitungen

In den Modulen gibt es Lehrbriefe, die als Lese-Anleitungen verfasst sind. Deshalb gibt es Funktionen, die grafisch kennzeichnen, dass jetzt eine *Lese-Anleitung* im Text folgt. Insgesamt gibt es zwei Funktionen. Einmal ein Symbol für einen Hinweis auf einen Begleittext oder ein Lehrbuch, das jetzt „sofort“ und unbedingt gelesen werden soll. Dazu gibt es die Funktion `\lesen`:

```
\lesen{Lesen Sie diese Anleitung, um die \LaTeX{}-Vorlage
zu verwenden!}
```

Und hier das Ergebnis:



Lesen Sie diese Anleitung, um die  $\text{\LaTeX}$ -Vorlage zu verwenden!

### **Tipp:**

Wichtig: vor der Umgebung muss eine Leerzeile stehen!

Für Hinweise auf Literatur, die als Vertiefung gelesen werden kann gibt es die Funktion `\vertiefen{}`. Sie unterscheidet sich von der `\lesen`-Funktion durch das verwendete Symbol. Hier ein Ergebnis:



Wichtig bei der Formulierung von Richtlinien ist, dass man nicht zu viele Regeln aufstellt. Dadurch erreicht man häufig das Gegenteil dessen, was eigentlich erreicht werden sollte. Holzmann schlägt 10 Regeln vor. Dieses interessante PDF finden Sie in OLAT!

Und die  $\text{\LaTeX}$ -Quelle:

```
\vertiefen{Wichtig bei der Formulierung von Richtlinien ist,
dass man nicht zu viele Regeln aufstellt. Dadurch erreicht
man häufig das Gegenteil dessen, was eigentlich
erreicht werden sollte. Holzmann schlägt 10 Regeln
vor. Dieses interessante PDF finden Sie in OLAT!
}
```



## 3.3 Kontrollfragen und Übungsaufgaben

In den Lehrbriefen der ZFH gibt es Kontrollfragen und Übungsaufgaben. Kontrollfragen sind kleine Fragen im Text, über die die Leser sofort, beim Lesen, nachdenken sollen. Der Lesefluss soll also unterbrochen werden. Die Antwort ist typischer Weise im darauf folgenden Text zu finden. Übungsaufgaben sind meist am Ende eines Abschnitts oder eines Kapitels aufgeführt, und im Anhang oder in einem separaten Dokument gibt es Lösungshinweise dazu.

### 3.3.1 Kontrollfragen

Für die *Kontrollfragen* gibt es die Funktion `\kontrollfrage` und einen eigenen Zähler, `\kontroll`. Der Zähler ist kapitelweise definiert. Hier ein Beispiel:

**Kontrollfragen**

```
\kontrollfrage{
\item[\kontroll] Wie fügt man eine Kontrollfrage ein?
\item[\kontroll] Wie fügt man eine Abbildung ein?
\item[\kontroll] Und drittens \ldots
}
```

Und so sieht das Ergebnis aus:



3.1 Wie fügt man eine Kontrollfrage ein?

3.2 Wie fügt man eine Abbildung ein?

3.3 und drittens ...

Und noch eine weitere Frage, um zu sehen ob die Nummerierung funktioniert:



3.4 Das ist eine blöde Frage!

**Tipp:**

Unbedingt darauf achten, dass *vor* und *nach* den Kontrollfragen in der L<sup>A</sup>T<sub>E</sub>X-Quelle Leerzeilen enthalten sind!

Wollen wir uns auf Kontrollfragen beziehen ist die Lösung aktuell etwas umständlich. Die Kontrollfrage 4 im dritten Kapitel erzeugen wir durch die Eingabe von `\ref{kontrolle:3:4}`. Allgemein ist

```
\ref{kontrolle:x:y}
```

ein Bezug auf eine Kontrollfrage im Kapitel  $x$ , mit der Nummer  $y$ .

Bei der Ausgabe wird nur die Nummer der Frage ausgegeben, das Kapitel muss manuell hinzugefügt werden. Dies ist ein Bezug auf die Kontrollfrage 3 im Kapitel 4: 4. Die Nummer des Kapitels im Bezug: 3.4 auf Seite 13 erhalten wir durch `3.\ref{kontrolle:3:4}`.

### 3.3.2 Übungsaufgaben

#### Übungsaufgaben

Die ZFH möchte *Übungsaufgaben* nach einer Überschrift, die auch grafisch kennzeichnet, dass jetzt Übungsaufgaben folgen. Die Aufgaben sollten kapitelweise nummeriert werden.

Zur Kennzeichnung gibt es die Funktion `\uebung`. Die einzelnen Aufgaben und die *Lösungen* stehen in einzelnen  $\text{\LaTeX}$ -Dateien. Dies stammt aus einem System, das ich seit Jahren für die Mathematik-Übungen einsetze. Ob eine Lösung dargestellt wird kann mit einer logischen Variable, `solutions`, in  $\text{\LaTeX}$  entschieden werden. Damit können wir leicht Musterlösungen von Übungsblättern oder Lösungs-Folien für die Vorlesung erzeugen, im gleichen oder in einem weiteren Dokument.

#### Verzeichnisse

Die Aufgabenstellungen und Lösungen liegen je nach Lehrveranstaltung relativ zur Haupt-Datei. Wo die Texte liegen wird durch `\exercisePath` angegeben, die im vorliegenden Text in `stammdaten` definiert ist. Alle Dateien für die Aufgabenstellung und die Lösungen stehen in den Unterverzeichnissen `aufgaben` und `loesungen`. Jede Aufgabe hat einen eindeutigen Namen. Als Beispiel verwenden wir im Folgenden `stat17.tex`. Diese Datei existiert in jedem der beiden Unterverzeichnisse. Der Dateiname ohne Suffix ist der Schlüssel zur Aufgabe. In der Funktion, die die Aufgabenstellung einliest wird auch ein *Label* gesetzt, das mit dem Namen der Datei übereinstimmt. So kann man später mit Hilfe von `\label{stat17}` Bezug auf die Nummer der gestellten Aufgabe Bezug nehmen. Werden in der Aufgabenstellung oder der Lösung  $\text{\LaTeX}$ -Labels generiert, werden diese als Regel immer mit dem Dateinamen begonnen. Dadurch wird sichergestellt dass es keine doppelten Marken gibt. Eine Abbildung in der Datei `stat17.tex` hat

ein Label wie `stat17:abbildung`. In den Lehrbriefen werden die Lösungshinweise im Anhang gesammelt. Deshalb findet man im Header dieser Datei die Zeile

```
\setboolean{solutions}{true}
```

Setzt man diese Variable auf `false` werden die Lösungen ausgeblendet.

Für die Darstellung der Aufgabenstellung gibt es die Funktion `\aufgabe`, die als Argument den Dateinamen der Aufgabenstellung erhält, ohne die Endung `.tex`. Für die Lösungen im Anhang gibt es die Funktion `\hinweis`, wieder mit dem Dateinamen als Argument. Zwei Übungsaufgaben in den Dateien `stat17.tex` und `stat22.tex` werden so als Übungsaufgaben in den Text eingeführt:

```
\uebung  
\aufgabe{stat17}  
\aufgabe{stat22}
```

Und hier das Ergebnis:



## Übungsaufgaben

### Aufgabe 3.1

Die Berechnungen für eine Stichprobe aus 200 Werten ergeben ein arithmetisches Mittel von  $\bar{x} = 1\,500$  € und  $s = 300$  €.

- Welche Aussagen können Sie über die Anzahl der Werte zwischen 900 € und 2 100 € machen?
- Welche Aussagen können Sie über die Anzahl der Werte zwischen 600 € und 2 400 € machen?

### Aufgabe 3.2

Berechnen Sie den Bravais-Pearson Korrelationskoeffizienten für den Datensatz „FAZ Index“!

Teilaufgaben sollen alphabetisch nummeriert werden. Dazu wird das Paket `enumitem` verwendet, um zu verhindern, dass jedes Mal für eine Aufgabe die Standard-Labels umgestellt werden. Um „normale“ Aufzählungen zu erhalten müsste man eine eigene Umgebung definieren, oder jedes Mal den Stil des Standard-Labels umstellen.

Die Lösungen der beiden Aufgaben findet man im Anhang 6.2, ab Seite 35. Im PDF werden Links erstellt, so dass man von den Lösungs-Seiten wieder auf die Aufgabenstellung springen kann.

Möchte man sich auf eine Aufgabe beziehen kann man, wie bereits erwähnt, ein Label verwenden, das in der Funktion `\aufgabe` erzeugt wird. Jede Aufgabe hat ein Label das mit dem Dateinamen übereinstimmt. Wir können uns also auf die zweite Aufgabe im Beispiel beziehen, sie hat die Nummer 3.2. Diesen Bezug erhalten wir mit Hilfe von `\ref{stat22}`.

**Tipp:**

Es gibt parallel zu diesem System der Übungsaufgaben auch eine Vorlage `setup_artcl` für Übungsblätter oder andere Dokumente, die mit `scrartcl` formatiert werden. Dort gibt es nur eine Funktion `\aufgabe`. Wird die Variable `solutions` im Header der Datei auf `true` gesetzt wird in diesem Kontext eine Musterlösung erzeugt, dabei steht die Lösung immer direkt nach der Aufgabenstellung.

Für die Aufgaben und Lösungen wird das Paket `theorem` eingesetzt. Deshalb werden an dieser Stelle in der Datei `setup.tex` drei Umgebungen *Fakt*, *Satz* und *Definition* definiert. Diese Umgebungen sind kapitelweise nummeriert.

Je nach Lehrveranstaltung gibt es weitere Fragmente, die einem Dokument hinzugefügt werden. in *Software Management* gibt es *Szenarien* für Aufgaben, in der Stochastik und in Lehrveranstaltungen zur Datenanalyse wird mit Datensätzen gearbeitet. Solche spezifische Funktionen werden nach `setup.tex` in die Haupt-Datei eingefügt, hier mit `LVDefinitionen.tex`. Im vorliegenden Fall definiert sie die Funktion `dataset`:

```
\newcommand{\dataset}[2]{\textbf{\large Datensatz #1}}
\vspace*{0.3cm}\hrule\label{#2}
\input{\exercisePath/datasets/#2}}
```

Auch hier gibt es für jeden Datensatz eine ASCII-Datei. Jeder Datensatz hat eine Kurzbezeichnung, die als erstes Argument angegeben wird, gefolgt vom Namen der Datei ohne die Endung. Der folgende Text wird durch

```
\dataset{Beruf und Sport}{berufsport}
```

in das Dokument eingefügt.

## Datensatz Beruf und Sport

### Beschreibung

In einer Befragung wurden 1 000 berufstätige Personen nach der Berufsgruppe und der sportlichen Betätigung gefragt. Quelle ist [BB93].

### Format

Die Kontingenztabelle der Ergebnisse ist in Tabelle 3.1 zusammengefasst.

**Tabelle 3.1:** Die Ergebnisse für den Datensatz „Beruf und Sport“

Berufsgruppe	Sportliche Betätigung			Randhäufigkeit
	Nie	Gelegentlich	Regelmäßig	
Arbeiter	240	120	70	430
Angestellter	160	90	90	340
Beamter	30	30	30	90
Landwirt	37	7	6	50
Freiberuflich	40	32	18	90
Randhäufigkeit	507	279	214	1 000

### Dateien

#### berufsport.csv

Die Datei enthält die Kontingenztabelle wie in Tabelle 3.1.

#### berufsport.json

Die Werte für beide Merkmale der Kontingenztabelle sind in der Liste `Daten` abgelegt. Dabei wurde die Matrix zeilenweise abgespeichert. Die Ausprägungen des Merkmals „Berufsgruppe“ sind in der Liste `KategorienX` abgelegt, in der Reihenfolge wie in der ersten Spalte der Kontingenztabelle. Die Ausprägungen des Merkmals „Sportliche Betätigung“ befinden sich in der Liste `KategorienY`. Die Reihenfolge hält sich an die Angaben in der zweiten Zeile der Kontingenztabelle.



# Kapitel 4

## Bewegliche Objekte

### 4.1 Abbildungen

Abbildungen im Text werden mit Hilfe von `includegraphics` in das Dokument eingefügt. Das hat den Vorteil, dass man den Suffix, also den Dateityp, bei der Angabe des Namens weglassen kann. In den meisten Fällen, bis auf ganz alte Abbildungen aus den Quellen für das Buch [Bri01] stammen, gibt es für Grafiken eine Version im `png`- oder im `eps`-Format. Damit wird sichergestellt, dass die Erstellung eines PDF-Dokuments mit Hilfe von *dvips* oder mit *pdflatex* möglich ist. Stand Juli 2018 wurde auf PDF<sub>L</sub>ATEX umgestellt, so dass es für neue Abbildung ausreicht das `png`-Format anzubieten.

Auf Abbildungen gibt es immer einen Bezug im Text, sie werden zentriert und die Beschriftung steht unterhalb der Grafik. Die Abbildung 4.1 wurde so in das Dokument eingefügt:

```
\begin{figure}[ht]
\centering
\includegraphics[width=10cm]{\imagePath/Misc/tugboat}
\caption{\label{tugboat}Ein schönes
Beispiel für eine Abbildung}
\end{figure}
```

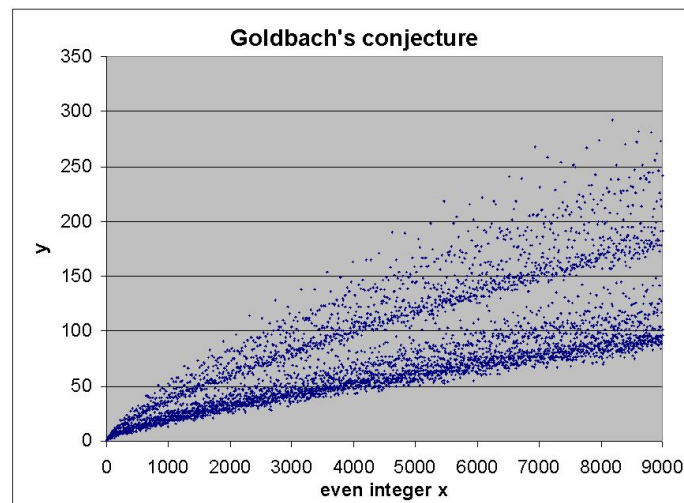
Zentral für alle Mathematik-Texte gibt es ein Verzeichnis `images`, das nochmals in verschiedene Unterverzeichnisse untergliedert ist. Neben dem Verzeichnis `Misc`, das Abbildungen enthält, die allgemein verwendet werden, gibt es Verzeichnisse wie `Graphentheorie` oder `Funktionen`.

**Tipp:**

Der Name `Misc` des Verzeichnisses stammt vom englischen Begriff *miscellaneous*. Die deutsche Übersetzung dafür ist „Sonstiges“.

**Tipp:**

Eine mögliche Fehlerquelle, die noch nicht überall beseitigt wurde ist die Tatsache, dass Verzeichnis-Namen in der Vergangenheit nicht eindeutig nach Groß- und Kleinschreibung vergeben wurden. Das ist unter Windows kein Problem, führt aber bei Verwendung auf Linux zu Problemen und muss sukzessive korrigiert werden!



**Abbildung 4.1:** Eine Abbildung mit einer Bitmap aus dem Verzeichnis `imagePath`

Daneben gibt es ein weiteres Verzeichnis mit Abbildungen, die in vielen Projekten oder Lehrveranstaltungen zum Einsatz kommen und nicht möglichst nicht dupliziert werden. Dafür gibt es die Variable `generalImagePath`, die wieder als spezifische Variable in `stammdaten` definiert ist. In Abbildung 4.2 finden wir ein solches Beispiel, gefolgt vom Quelltext für das Einfügen.

```
\begin{figure}[ht]
\centering
\includegraphics[width=10cm]{\generalImagePath/Misc/tugboat}
\caption{\label{tugboat}Ein schönes
Beispiel für eine Abbildung}
\end{figure}
```





Abbildung 4.2: Ein schönes Beispiel für eine Abbildung

Neben Abbildungen, die mit EPS-Grafiken oder Bitmaps erstellt werden bietet  $\text{\LaTeX}$  die Möglichkeit mit Vektorgrafik-Funktionen selbst Abbildungen zu zeichnen. Nicht nur in der Mathematik gibt es häufig Abbildungen mit zweidimensionalen Abbildungen. Diese Koordinatensysteme sind als Box in der Datei `coordinateSystems.tex` definiert. Abbildung 4.3 zeigt eines der Beispiele, ein Koordinatensystem mit einem Gitter, das alle vier Quadranten anzeigt. Die linke untere Ecke liegt bei  $(-4.25, -2.2)$ , die Länge ist 8.5 und die Höhe ist 5.5. Der Abstand der Gitterlinien beträgt 0.5.

**Koordinaten-  
systeme**

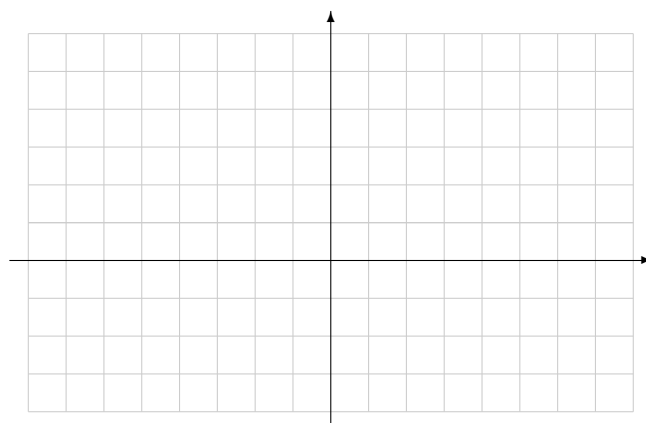


Abbildung 4.3: Ein Koordinatensystem mit allen Quadranten und Gitter

Eine Abbildung mit einem solchen Koordinatensystem erzeugen wir mit

```
\begin{picture}(8.5,5.5)(-4,-2.2)
\put(-4.25,-2.2){\usebox{\centeredCS}}
\end{picture}
```

## 4.2 Tabellen

*Tabellen* werden mit `table` eingefügt und haben immer eine Beschriftung, um im Text darauf Bezug zu nehmen. Die Beschriftung wird als Tabellen-Überschrift eingefügt, also oberhalb der Tabelle. Das stammt aus den Projekten mit dem Hanser-Verlag und wurde seitdem beibehalten. Bei den Büchern wurde die Quelle für jede Tabelle sogar in separate Dateien ausgelagert und in einem Verzeichnis `tables` zusammengefasst. Dies ist sinnvoll wenn Tabellen sowohl in einem Text als auch in Folien, die man mit dem *Beamer*-Paket erstellt, verwendet. So ist sicher gestellt, dass die Tabelle nur an einer Stelle steht und dass Korrekturen so immer für alle Dokumente wirken, in denen die Tabelle verwendet wird.

**Tabelle 4.1:** Ein Beispiel für eine Tabelle

Erste Spalte	Zweite Spalte
1	Eins
2	Zwei

Die Tabelle 4.1 wurde mit dem folgenden Quelltext eingefügt:

```
\begin{table}[ht]
\centering
\begin{tabular}{ll}\hline
\textbf{Erste Spalte}&\textbf{Zweite Spalte}\\
\hline
Erste Spalte&Zweite Spalte\\
$1$&Eins\\
$2$&Zwei\\
\hline
\end{tabular}
\end{table}
```

Einträge in den Spalten sind meist linksbündig. Das muss man nicht beibehalten, aber die Ausrichtung von Einträgen sollte immer konsistent sein.

# Kapitel 5

## Folien mit dem Beamer-Package

Folien für eine Lehrveranstaltung werden mit Hilfe des Beamer-Package erstellt. Dabei stehen die Quellen der Folien parallel zu den Texten und Übungsaufgaben. Damit wird gewährleistet, dass Abbildungen und andere Assets möglichst zentral stehen und wiederverwendet werden. Die Folien sind bei Bedarf unterteilt nach den verschiedenen Lehrveranstaltungen und Fachgebieten wie *Analysis*, *Lineare Algebra* oder *Stochastik*.

Wie schon für die Texte gibt es auch hier eine Datei mit Angaben, die abhängig von der jeweiligen Vorlesung sind. Diese Datei heißt `header.tex`. Das folgende Beispiel stammt aus den Folien für die Präsenzphase des Fachs *Stochastik für Informatiker* im Studiengang IT-Analyst:

```
\newcommand{\lectureName}{Stochastik für Informatiker}
% Titel der Folie
\title{\lectureName}
\author{Prof. Dr. Manfred Brill}
\institute{Hochschule Kaiserslautern\\
    Fachbereich Informatik und Mikrosystemtechnik}
\date[]{}
% Variablen wie Semester ...
%
\newcommand{\theSemester}{Sommersemester~2018}
% Standardverzeichnis für das Basisverzeichnis der Bilder
%
\newcommand{\imagePath}{../../images}
%
% Name der Bild-Datei, die auf die Frage-Folie kommen soll
\newcommand{\questionImage}{Misc/Answer_to_Life}
```

Die Bezeichnung der Lehrveranstaltung wird definiert und anschließend werden die Variablen für die Titelfolie gesetzt. Anschließend definieren wir das aktuelle Semester und die Pfade für das Verzeichnis, in dem Bilder für die Lehrveranstaltung liegen. Jede Folie enthält eine letzte Folie mit einem Bild. Das Bild, das verwendet werden soll wird in der Variable `questionImage` angegeben.

An die Stelle der Datei `setup.tex` tritt `slidesheader.tex`, die im Verzeichnis `texmf-local` steht und unabhängig von den Lehrveranstaltungen verwendet werden kann. Diese Datei wird im Folgenden noch beschrieben. Bevor wir auf `slidesheader.tex` eingehen werfen wir einen Blick auf `keywords`. Hier werden Schlagwörter und andere Informationen für die Eigenschaften des erstellten PDF-Dokuments gespeichert. Das Vorgehen ist ähnlich wie bei den Texten. Hier ein Beispiel, wieder aus der gleichen Veranstaltung:

```
\subject{Stochastik für Informatiker}
\keywords{Wahrscheinlichkeitsrechnung und Statistik,
          Medieninformatik, Hochschule Kaiserslautern}
%
\usepackage{breakurl}
\hypersetup{
pdfauthor = {Manfred Brill},
pdfsubject = {Stochastik für Informatiker},
pdftitle = {Stochastik für Informatiker},
pdfkeywords = {Stochastik für Informatiker, IT-Analyst,
               Hochschule Kaiserslautern},
pdfpagelayout = SinglePage,
pdfpagemode = UseThumbs,
pdfdisplaydoctitle = true
}

\urlstyle{same}
```

Die Datei `slidesheader.tex` lädt ähnlich wie `setup.tex` einige Packages. Da es bei Folien durchaus vorkommen kann, dass Medien eingebunden werden wird das Package `multimedia` geladen. Da es neben den Folien auch Handouts geben soll wird das Package `pgfpages` verwendet. Um Smileys und andere Abbildungen in Folien zu verwenden wird das Package `marvosym` geladen.

Wie für die anderen Dokumente werden die Einstellungen in den Dateien `colors.tex`, `variablen.tex` und `coordinateSystems.tex` geladen. Auch das `listings`-Paket wird geladen und konfiguriert. Auch für Folien wird die Variable `solutions` definiert, die anschließend in den Folien und auch in `slidesheader` verwendet wird.

Das `beamer`-Package bietet verschiedene Styles an, die angepasst werden können. Als *theme* wird `CambridgeUS` ausgewählt. Die weiteren Anpassungen finden in der Datei `beamer.tex` statt. Wichtig ist die Verwendung der Farbe `colorDepartment`, die in `colors.tex` definiert ist. Hier wird die Farbe des Fachbereichs aus dem Hochschul-Logo verwendet. Das Ergebnis ist in Abbildung 5.4 zu sehen.

Folien die nur aus Abbildung und einer Überschrift bestehen kommen häufig in Präsentationen vor. Deshalb ist eine Funktion definiert, mit deren Hilfe man eine solche Folie erstellen kann:

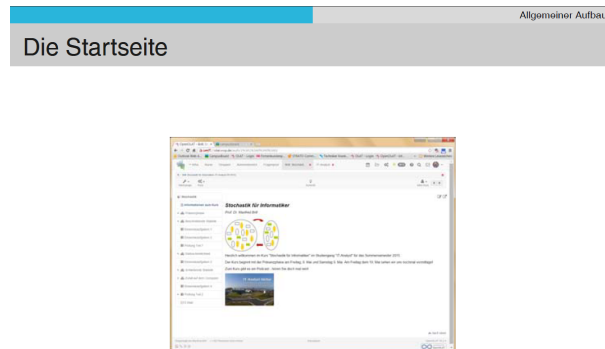
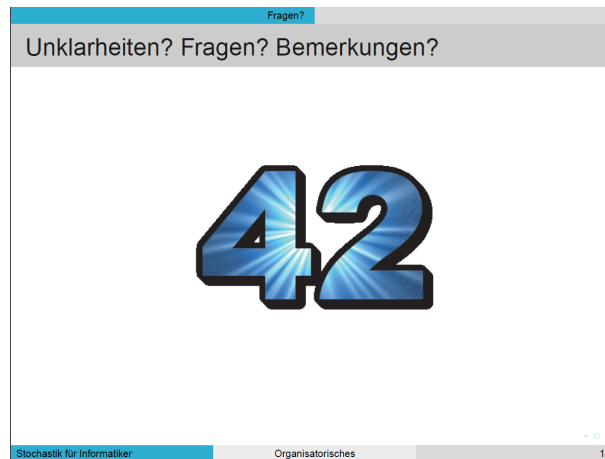
```
\newcommand{\imageslide}[3]{%
\begin{frame}{#1}%
  \begin{center}%
    \includegraphics[width = #2]{\imagePath/#3}%
  \end{center}%
\end{frame}}
```

Diese Funktion hat drei Argumente. Das erste Argument ist der Titel der Folie, gefolgt von der Breite, mit der die Abbildung in die Folie eingeführt werden soll. Das letzte Argument ist der Pfad zur Bild-Datei, relativ zum Wert der Variable `imagePath`.

```
\imageslide{Die Startseite}{6cm}{OLAT/itastochastik_index}
```

In Abbildung 5.1 ist das Ergebnis dieser Anweisung zu sehen.

Am Ende jeder Datei mit Folien wird eine Folie eingefügt, die die Zuhörer zu Fragen auffordert. Die Hörer erkennen so immer, dass das Thema abgeschlossen ist. Dafür gibt es die Funktion `questions`. Als Bild wird die Datei verwendet, die in `header.tex` in der Variable `\questionImage` definiert ist. Die Funktion `questions` reagiert auf die Variable `solutions`. Die Folie wird nur erzeugt, wenn `solutions` auf `TRUE` gesetzt ist. So kann die Fragefolie bei Handouts und anderen Dokumenten ausgeblendet werden. Für die Studierenden macht es sicher keinen Sinn, in 50 PDF-Dateien jedes Mal die Fragefolie zu drucken. Abbildung 5.2 zeigt ein Ergebnis der Funktion.

Abbildung 5.1: Eine Folie, erzeugt mit `imageslide`Abbildung 5.2: Eine Folie, erzeugt mit `questions`

Die Breite des eingefügten Bilds kann als optionales Argument übergeben werden; der Default-Wert dafür ist 5 cm. Ein Bild mit einer Breite von 8 cm wird so eingefügt:

```
\questions[8cm]
```

Mit den beschriebenen Konfigurationen sieht der Kopf der  $\text{\TeX}$ -Datei für die Folien immer gleich aus:

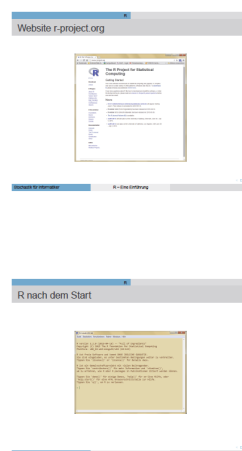
```
\documentclass{beamer}
\input{header}
\input{slidesheader}
```

```
%
\subtitle{Organisatorisches}
%
\input{keywords}
% Kommentieren der folgenden Zeilen, je nach Kontext
\setbeameroption{show notes}
\setboolean{solutions}{true}
\pgfpagesuselayout{2 on 1}[a4paper, border shrink=5mm]
%
```

Das Thema der Folien wird mit `subtitle` auf die Titel-folie eingefügt. Die Variable `solutions` wird gesetzt, damit können wir Folien aus- und einblenden. Das `beamer`-Package bietet die Möglichkeit, Notizen in die Datei einzufügen. Diese werden als Default ausgeblendet, mit der Zeile

```
setbeameroptions{show notes}
```

werden sie angezeigt. Die Zeile mit der Funktion `pgfpagesuselayout` wird für Handouts eingesetzt, bei denen zwei Folien pro Seite ausgegeben werden. Abbildung 5.3 zeigt ein solches Beispiel.



**Abbildung 5.3:** Die Informationen aus unserem Beispiel

Insgesamt werden immer drei Versionen des PDF-Dokuments erzeugt. Dafür gibt es im Verzeichnis mit den Quelldateien ein Verzeichnis mit dem Namen `fertig` und drei Unterverzeichnisse für Folien, Handouts und Notizen. Alle Folien und alle Notizen liegen in `notes`, die Folien, die angezeigt werden

sollen im Verzeichnis `folien`, und die Handouts, die über OLAT oder andere Seiten an die Hörer verteilt werden liegen im Ordner `handouts`. Die Handouts werden mit `\setboolean{solutions}{false}` und ohne Notizen erzeugt.



Abbildung 5.4: Die Titel-Folie zu unserem Beispiel

Um in den Notizen eine Überschrift zu formatieren gibt es in `slidesheader` die Funktion `\noteshead`, an die ein Text für die Überschrift übergeben wird. Die ersten beiden Einträge in jeder Folien-Datei besteht dann aus der Titel-Folie und einer Notizen-Seite. Auf dieser Notizen-Seite sind Angaben über den Inhalt, den Namen der Datei und über die Rangfolge enthalten. Hier ein Beispiel, das zu der Titel-folie in Abbildung 5.4 führt:

```
\begin{frame}
  \titlepage
  \begin{center}
    \includegraphics[width = 3.5cm]
      {\imagePath/statistiklogo}
  \end{center}
\end{frame}

\note{
\noteshead{Information}

\begin{description}
\item[Thema:] Einführung
\item[Version:] \theSemester{}
\item[Dateiname:] organisatorisches.tex
```



```
\item[Reihenfolge:] $1$  
\end{description}  
}
```

Abbildung 5.5 zeigt die so erzeugte Notizen-Seite.

### **Information**

Thema: Einführung

Version: Sommersemester 2018

Dateiname: organisatorisches.tex

Reihenfolge: 1

**Abbildung 5.5:** Die Informationen aus unserem Beispiel



# Kapitel 6

## Verzeichnisse und Index

### 6.1 Literaturverzeichnis

Das *Literaturverzeichnis* wird mit Hilfe von `BIBTEX` aufgebaut. Dazu gibt es eine ganze Reihe von Dateien mit Einträgen, die für verschiedene Themen wie `LATEX`, Mathematik oder Computergrafik aufgebaut wurden. Diese Dateien stehen zentral in einem Verzeichnis und werden in die Dokumente mit Hilfe von relativen Pfaden eingefügt. `BIBTEX` ist relativ alt und unhandlich, das macht sich häufig bemerkbar.

Die einzelnen Eingabe-Dateien für die Quellen werden mit Hilfe von `JabRef`, einer Java-Anwendung, verwaltet ([Kop18]). Vorteil dieser Anwendung ist, dass man in den Dateien suchen kann und die Einträge als Formular-Einträge bearbeitet werden können. Das Literaturverzeichnis, das wir auf Seite 33 finden wurde so in das Dokument eingefügt: **JabRef**

```
\cleardoublepage
\phantomsection
\addcontentsline{toc}{chapter}{Literaturverzeichnis}
\chaptermark{Literaturverzeichnis}
\sectionmark{Literatur}\label{literatur}
\bibliography{
../.../BibTex/latex,../.../BibTex/mathematikBooks}
```

Interessant ist die Funktion `phantomsection`, die dafür sorgt, dass das Literaturverzeichnis richtig behandelt wird und die Verweise in der Inhaltsangabe bei der PDF-Anzeige auch korrekt auf den Anfang des Literaturverzeichnisses verweisen.

**Tipp:**

Die einzelnen relativen Pfade zu den BibTeX-Dateien müssen so wie oben angegeben in der Datei stehen. Das Einfügen von Blanks führt bei mir zu Fehlermeldungen!

## 6.2 Index

Der *Index* wird mit dem Paket `imakeidx` erzeugt. Hier ist wichtig, dass die deutschen Umlaute korrekt dargestellt wird. Dies gelingt durch die Option `-g` für die Funktion `makeindex`, die während der Erstellung des PDF-Dokuments aufgerufen wird. Damit diese Option korrekt arbeitet und deutsche Umlaute wie gewohnt im Editor eingegeben werden können müssen die Quotes umgestellt werden. Dazu gibt es die Datei `german.ist`, die sinnvoller Weise in `texmf-local/makeindex/german` gespeichert ist:

```
% Voraussetzung: makeindex mit Option -g aufrufen!
quote '>'      % > ersetzt "
%
delim_0 "\\hspace{2ex} "
delim_1 "\\hspace{2ex} "
delim_2 "\\hspace{2ex} "
```

Für den Index ist eingestellt, dass ein Eintrag im Inhaltsverzeichnis erstellt wird und dass der Index, der mit `printindex` ausgegeben wird, zweispaltig ist. In diesem Dokument wurde der Index so eingefügt:

```
% Index
\clearevenpage
\phantomsection
\small
\printindex
\normalsize
```

# Literaturverzeichnis

- [BB93] BAMBERG, GÜNTER und BAUR, FRANZ: *Statistik*. Oldenbourg, 8. Auflage, 1993.
- [Bri01] BRILL, MANFRED: *Mathematik für Informatiker*. Hanser Verlag, 2001.
- [Bri13a] BRILL, MANFRED: *mbmath.sty - Ein mathematisches L<sup>A</sup>T<sub>E</sub>X-Package*. Hochschule Kaiserslautern, 2013.
- [Bri13b] BRILL, MANFRED: *mb.sty - L<sup>A</sup>T<sub>E</sub>X-Style für Texte an der Fachhochschule Kaiserslautern*. Hochschule Kaiserslautern, 2013.
- [Kop02] KOPKA, HELMUT: *LaTeX - Band 1: Einführung*. Addison-Wesley, 2002.
- [Kop18] KOPP, OLIVER: *JabRef*. <http://www.jabref.org/>, 2018. Zuletzt gesehen am 26. Februar 2018.
- [MG05] MITTELBACH, FRANK und GOOSSENS, MICHEL: *Der LaTeX-Begleiter*. Addison-Wesley, 2005.
- [ZFH11] ZFH: *Vorgaben zur Gestaltung von Lehrtexten*. ZFH, 2011.



# Lösungshinweise

Hier finden sich die beiden Beispiele für die Lösungshinweise.

## Aufgabe 3.1

Wir verwenden die Tschebyscheff'sche Ungleichung und die empirischen Einschätzungen.

- a) Die gesuchten Angaben passen zum Wert  $k = 2$ . Mit Tschebyscheff erhalten wir die Voraussage, dass 75 % in diesem Bereich liegen; die empirische Einschätzung von McClave-Benson sagt sogar 95 % voraus.
- b) Die gesuchten Angaben passen zum Wert  $k = 3$ . Mit Tschebyscheff erhalten wir die Voraussage, dass ungefähr 90 % in diesem Bereich liegen; die empirische Einschätzung von McClave-Benson sagt sogar 100 % voraus.

## Aufgabe 3.2

Die Ergebnisse wurden mit OpenOffice Calc berechnet!

Den FAZ-Index nennen wir  $X$ , die Renditen  $Y$ . Dann müssen wir die arithmetischen Mittelwert für  $X$  und  $Y$  berechnen; wir erhalten

$$\bar{x} = \frac{2\,968}{8} = 371, \bar{y} = \frac{61,6}{8} = 7,7.$$

Als empirische Standardabweichung für  $X$  und  $Y$  erhalten wir

$$s_x \approx 88,776, s_y \approx 0,93.$$

Für die Kovarianz ergibt sich

$$s_{xy} = -70,$$

und insgesamt gilt

$$r_{xy} \approx -0,848.$$

Wir können auf der Basis dieser Daten zu Recht einen negativen linearen Zusammenhang vermuten.



# Index

Übungsaufgaben 14  
Verzeichnisstruktur 14

Beamer 22  
Beispiel 8  
Beruf und Sport 17

Datensätze 16  
Definition 16  
dvips 19

Fakt 16

Hurenkinder 7

Index 32  
deutsche Umlaute 32

JabRef 31

Kontrollfragen 13  
Koordinatensysteme 21

Label 14  
Lese-Anleitung 12  
Literaturverzeichnis 31

Marginalien 11  
miscellaneous 20

pdflatex 19  
Praxisbezug 11

Satz 16  
Schusterjunge 7  
setup.tex 3  
Szenarien 16

Tabellen 22