

Einführung in das Textsatzsystem \LaTeX für Psychologen

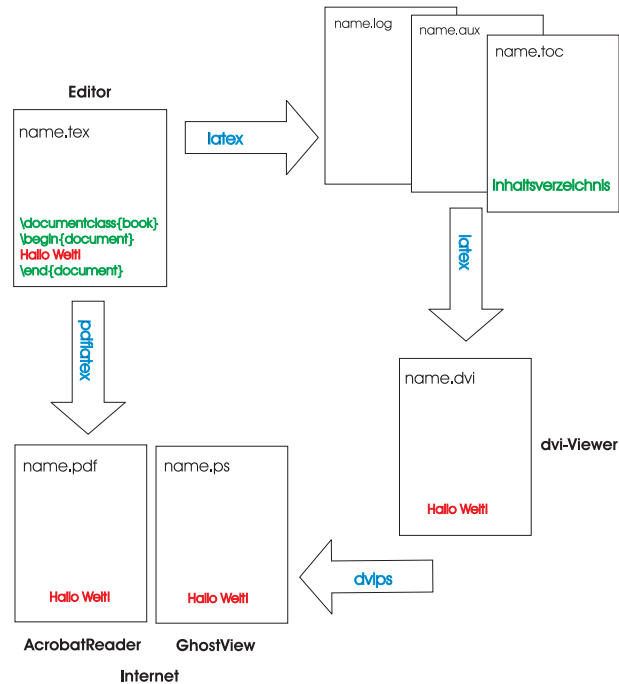
Prof. Dr. Carolin Strobl
Psychologische Methodenlehre
HS 2012

Inhaltsverzeichnis

1	Wie funktioniert \LaTeX?	3
2	Arbeiten mit einem \LaTeX-Editor	4
3	Grundlegende \LaTeX-Befehle	4
3.1	Kommentare	4
3.2	Dokumentklassen - notwendig!	6
3.3	Pakete	6
3.3.1	Pakete für Umlaute	6
3.3.2	Weitere Pakete	7
3.4	Weitere häufig genutzte Befehle in der Präambel	7
3.5	Dokumentbeginn und Ende - notwendig!	7
3.6	Titelseite	8
3.7	Beispiel Präambel und Dokumentbeginn	8
3.8	Gliederungsbefehle	8
3.9	Fußnoten	9
3.10	Aufzählungen	9
3.11	Abstände, Zeilenwechsel, Absätze	11
3.12	Seitenaufteilung	11
3.13	Schriftgrößen und Schriftarten	13
3.14	Farben	13

EINFÜHRUNG IN L ^A T _E X	2
4 Mathematische Formeln	15
4.1 Matrizen	16
4.2 Komplizierte Gleichungen	18
4.3 Mehrzeilige Gleichungssysteme	19
5 Gleitobjekte	20
5.1 Tabellen	20
5.1.1 Verschachtelte Tabellen	21
5.1.2 Tabellen als Gleitobjekte	23
5.2 Grafiken	24
5.2.1 Exkurs: Grafiken in R erzeugen	25
5.2.2 Grafiken direkt im L ^A T _E X Dokument erzeugen	26
5.3 Rotierte Gleitobjekte	26
6 Verzeichnisse	28
6.1 Literaturverzeichnis	28
6.1.1 Literaturverzeichnis im Dokument	28
6.1.2 Literaturdatenbank mit BibT _E X	29
6.1.3 Aussehen der Literaturangaben verändern	30
6.2 APA-Zitierstil	32
6.3 Andere Verzeichnisse	33
7 Vortragsfolien mit L^AT_EX	33
8 Zusammengesetzte Dokumente	34
8.1 Einbinden von L ^A T _E X-Quelltext	34
8.2 Einbinden von R-Quelltext	34
9 Kreuzworträtsel	36
A Farb-Namen für die usenames Option	37
B Parameter zur Seitenaufteilung	38
C Kreuzworträtsel-Lösung	39

1 Wie funktioniert L^AT_EX?



L^AT_EX ist ein Textsatzsystem, mit dem man sehr einfach sehr schöne Dokumente erzeugen kann, wenn man ein bisschen übt und weiss, wo man weiter nachlesen kann...

Bei jedem L^AT_EX Dokument kann man am Anfang festlegen, wie das fertige Dokument aussehen soll. Dazu muss man die entsprechende Befehls-Syntax kennen, die man zusätzlich zum Inhalt des Dokuments in die Quelltext-Datei schreibt. Wenn man dann das Programm L^AT_EX aufruft, wird der Quelltext automatisch so gesetzt, dass das Dokument gut aussieht.

Das passiert in folgenden Schritten:

1. Quelltext erzeugen (Datei **name.tex**)
2. Setzen mit L^AT_EX (erzeugt eine dvi-Datei **name.dvi**, aus der man in einem weiteren Schritt eine ps- oder pdf-Datei erzeugen kann) oder (dieses Vorgehen wird empfohlen:) mit PDFL^AT_EX (erzeugt direkt eine pdf-Datei **name.pdf**)
außerdem werden automatisch noch weitere Hilfsdateien erzeugt (**name.log**, **name.aux** etc.), die zusätzliche Informationen enthalten und die L^AT_EX zum Setzen benötigt
3. fertiges Dokument ankucken
4. zum Verändern ggf. wieder den Quelltext bearbeiten

2 Arbeiten mit einem L^AT_EX-Editor

L^AT_EX Dokumente können in jedem beliebigen Editor oder Textverarbeitungsprogramm erstellt werden. Es gibt aber auch spezielle Editoren, die besondere Hilfestellung (Syntax-Highlighting, Vervollständigung von Befehlen, graphische Oberfläche mit Buttons zum Ausführen der Befehle etc.) bieten.

Ein plattformunabhängiger (d.h. für Windows, Unix und Mac geeigneter) Editor ist Texmaker.

Vorgehen zum Erzeugen eines Dokuments:

1. neue Datei anlegen mit File-New oder Neue-Seite-Symbol
2. Quelltext mit L^AT_EX-Befehls-Syntax schreiben
3. abspeichern mit der Dateiondung `.tex`
4. oben rechts PDFL^AT_EX auswählen und mit dem Pfeil links daneben aufrufen; pdf-Datei und Hilfsdateien werden erzeugt und landem im selben Ordner wie die tex-Datei
5. bei Fehlermeldungen (erscheinen in rot im log-Fenster): im Quelltext in der angegebenen Zeile (wird angezeigt wenn man die Fehlermeldung anklickt) nach dem Fehler suchen und verbessern
6. wieder PDFL^AT_EX aufrufen
7. die pdf-Datei kann man mit dem eingebauten Viewer rechts oben oder unter Tools-View PDF (F7) (oder durch Doppelklick auf das Dateisymbol im Ordner z.B. mit dem AcrobatReader) öffnen (unter Windows kann die Anzeige aber vermutlich nicht bei geöffneten AcrobatReader aktualisiert werden).

Für einige der im Folgenden beschriebenen Inhalte ist es u.a. notwendig, den PDFL^AT_EX-Befehl mehrfach auszuführen. Damit man das nicht immer durch mehrfaches Anklicken machen muss, gibt es z.B. bei Texmaker die Möglichkeit, im Menü unter Options-Configure Texmaker eine Abfolge von Befehlen zusammenzustellen und als sog. Quick Build abzuspeichern. Ruft man danach Quick Build auf, werden alle Befehle automatisch ausgeführt.

3 Grundlegende L^AT_EX-Befehle

3.1 Kommentare

Es kann nie schaden, in ein Dokument erklärende Kommentare einzufügen, damit man nach einer Woche immernoch weiß, was man mit diesem kryptischen Quelltext erzeugen wollte...

Die Kommentare werden von L^AT_EX beim Übersetzen einfach ignoriert. Jede Kommentarzeile wird folgendermaßen gekennzeichnet:

```
Das hier wird gezeigt.  
% Das hier nicht.  
Das hier wird wieder gezeigt.
```

Der Output sieht dann so aus:

```
Das hier wird gezeigt.  
Das hier wird wieder gezeigt.
```

Um ganze Absätze auszukommentieren kann man auch folgende Befehle verwenden:

```
Das hier wird gezeigt.
```

```
\iffalse  
Das hier nicht.  
Das ist alles Kommentar.  
\fi
```

```
Das hier wird wieder gezeigt.
```

oder die `comment`-Umgebung aus dem `verbatim`-Paket, das vor Beginn des Dokuments in der Präambel eingebunden werden muss (s.u.):

```
\usepackage{verbatim}
```

```
Das hier wird gezeigt.
```

```
\begin{comment}  
Das hier nicht.  
Das ist alles Kommentar.  
\end{comment}
```

```
Das hier wird wieder gezeigt.
```

Ergebnis:

```
Das hier wird gezeigt.  
Das hier wird wieder gezeigt.
```

3.2 Dokumentklassen - notwendig!

Zuerst muss man sich entscheiden, wie das fertige Dokument aussehen soll. Es stehen verschiedene Dokumentklassen zu Verfügung, z.B.:

- `article` und `report` für Artikel
- `book` für Bücher
- `thesis` für Abschlussarbeiten
- `slides`, `foils` und `beamer` für Folien

Zusätzlich zur Dokumentklasse kann man (muss aber nicht) weitere Optionen wählen, z.B.:

- `a4paper` für das Papierformat
- `oneside` für einseitigen Druck (d.h. bei `book` z.B. auch einseitige Kopfzeilen)
- `10pt` oder `12pt` für die Schriftgröße

Ein kompletter Befehl sieht dann z.B. so aus:

```
\documentclass[a4paper,oneside,12pt]{book}
```

3.3 Pakete

Zusätzlich zu den Standardfunktionen bietet L^AT_EX noch weitere Pakete, die man bei Bedarf, z.B. für deutsche Umlaute, spezielle mathematische Symbole oder Grafik-Einbindung, verwenden kann. Die Pakete werden vor Dokumentbeginn in der sog. Präambel aufgerufen mit:

```
\usepackage{paketname}
```

3.3.1 Pakete für Umlaute

Mit dem veralteten Aufruf (vor Dokumentbeginn)

```
\usepackage{german}
```

ergeben die Kombinationen

```
"a, "o, "u, "s
```

die Umlaute ä, ö, ü, ß. (In älteren T_EX Dokumenten und Büchern wird diese Schreibweise manchmal noch verwendet.) Inzwischen können Umlaute z.B. mit

```
\usepackage[german]{babel}
\usepackage[latin1]{inputenc}
```

aber auch direkt getippt werden.

Durch die Option `german` wird z.B. auch statt dem englischen *Contents* das Wort *Inhaltsverzeichnis* verwendet.

Neben `latin1` gibt es noch andere sog. Encodings (z.B. `utf8`, das standardmäßig von vielen Unix-Betriebssystemen verwendet wird).

3.3.2 Weitere Pakete

Weitere häufig genutzte Pakete sind z.B.:

- `xcolor` für Farben
- `graphicx` zum Einfügen von Grafiken
- `amsmath` für bestimmte mathematische Symbole

Diese und unzählige weitere Pakete stehen kostenlos zum Download zur Verfügung unter:

<http://www.ctan.org/>

Alle Pakete, die man in einem Dokument verwenden will, müssen aber jedes mal vor dem Dokumentbeginn in der sog. Präambel durch den Befehl `usepackage` geladen werden (Beispiel s.u.)!

3.4 Weitere häufig genutzte Befehle in der Präambel

Um zu Verhindern, daß L^AT_EX (wie im Buchdruck üblich) bei jedem neuen Absatz den Text etwas einrückt, kann man den Befehl

```
\setlength{\parindent}{0pt}
```

verwenden; für einen größeren Zeilenabstand z.B.:

```
\renewcommand{\baselinestretch}{1.5}
```

3.5 Dokumentbeginn und Ende - notwendig!

Das eigentliche Dokument beginnt mit dem Befehl

```
\begin{document}
```

und endet mit dem Befehl

```
\end{document}.
```

Wenn man den abschließenden Befehl (oder z.B. eine Klammer) vergißt, passiert Schreckliches! Clevere L^AT_EX Editoren wie Texmaker schlagen aber automatisch den passenden `end` Befehl vor.

3.6 Titelseite

Um eine Titelseite mit automatischer Hervorhebung von Titel und Autor (zentriert und größer gesetzt) zu erzeugen, werden üblicherweise in der Präambel die Befehle

```
\title{Ein iteressanter Aufsatz}
\author{Ein begnadeter Schriftsteller}
\date{1. April, 2004}
```

festgelegt, und direkt nach Dokumentbeginn wird mit `maketitle` die Titelseite ausgegeben (bei `book` und `report` auf einer extra Seite, bei `article` fortlaufend).

Wird kein Datum eingegeben, wird meist das aktuelle Datum verwendet; möchte man kein Datum angeben, lässt man im `date`-Befehl ein Leerzeichen zwischen den geschweiften Klammern.

3.7 Beispiel Präambel und Dokumentbeginn

```
\documentclass[oneside]{article}

\usepackage{a4wide}
\usepackage[german]{babel}
\usepackage[latin1]{inputenc}
\usepackage{verbatim}
\usepackage{graphicx}

\renewcommand{\baselinestretch}{1.5}

\title{Einführung in das Textsatzsystem \LaTeX \ für Psychologen}
\date{HS 2012}
\author{Prof. Dr. Carolin Strobl\ \Psychologische Methodenlehre}

\begin{document}
\maketitle
...
```

3.8 Gliederungsbefehle

Zur Gliederung kann man in das Dokument verschiedene Überschriften und Unter-Überschriften einbauen:

```
\section{Text als Überschrift}
```


Zur Verfügung stehen `section`, `subsection`, `subsubsection` etc. und in den Dokumentklassen `book` und `report` zusätzlich `chapter`. (Bei der Dokumentklasse `book` werden außerdem automatisch die Kapitelüberschriften oben auf jeder Seite angezeigt - sehr ästhetisch!)

Die Überschriften müssen nur im fließenden Text eingesetzt werden, L^AT_EX übernimmt sie dann automatisch in das Inhaltsverzeichnis, das mit dem Befehl

```
\tableofcontents
```

am Anfang des Dokuments ausgegeben wird.

Bei Verzeichnissen, wie z.B. auch beim Inhaltsverzeichnis, werden Referenzen und Seitenzahlen beim ersten Durchlauf von L^AT_EX gesammelt und erst beim zweiten Durchlauf ins Verzeichnis übernommen. Deshalb sind, bis alle Einträge bzw. Änderungen sichtbar sind, mehrere Durchläufe von L^AT_EX oder PDFL^AT_EX nötig. Im Texmaker besteht aber wie oben beschrieben die Möglichkeit, eine ganze Serie von Befehlen unter Quick Build abzuspeichern.

3.9 Fußnoten

Fußnoten kann man in den fließenden Text mit dem Befehl `footnote` einfügen. Sie werden von L^AT_EX automatisch durchnummeriert.

Der Befehl wird so eingesetzt:

```
Gleich kommt eine Fußnote\footnote{Dieser Text erscheint als
Fußnote unten auf der Seite!}.
```

und hat folgende Wirkung (siehe auch unten):

Gleich kommt eine Fußnote¹.

3.10 Aufzählungen

L^AT_EX kann einfache und nummerierte Listen erzeugen:

```
\begin{itemize}
  \item AAAAA
  \item BBBBB
  \item CCCCC
\end{itemize}
```

- AAAAA
- BBBBB

¹Dieser Text erscheint als Fußnote unten auf der Seite!

- CCCCC

```
\begin{enumerate}
  \item AAAAA
  \item BBBBB
  \item CCCCC
\end{enumerate}
```

1. AAAAA
2. BBBBB
3. CCCCC

Die Aufzählungssymbole können auch selbst festgelegt werden:

```
\begin{itemize}
  \item[a] AAAAA
  \item[b] BBBBB
  \item[c] CCCCC
\end{itemize}
```

- a) AAAAA
- b) BBBBB
- c) CCCCC

oder automatisch mit dem `enumerate` Paket:

```
\begin{enumerate}[a]
  \item AAAAA
  \item BBBBB
  \item CCCCC
\end{enumerate}
```

- a) AAAAA
- b) BBBBB
- c) CCCCC

3.11 Abstände, Zeilenwechsel, Absätze

L^AT_EX ignoriert von Hand getippte Abstände und Zeilenwechsel, z.B. wird aus

```
Das sind Leerzeichen    und das ist ein Zeilenwechsel
der ignoriert wird. Erst eine Leerzeile
```

führt zu einem Absatz mit Zeilenwechsel.

Das sind Leerzeichen und das ist ein Zeilenwechsel der ignoriert wird. Erst eine Leerzeile führt zu einem Absatz mit Zeilenwechsel.

Deshalb gibt es entsprechende L^AT_EX Befehle:

<code>\vspace{3cm}</code> , <code>\medskip</code>	vertikaler Abstand
<code>\hspace{3cm}</code>	horizontaler Abstand
<code>\newline</code> oder <code>\\</code>	Zeilenwechsel
Leerzeile	neuer Absatz (ggf. mit Einrückung) und Zeilenwechsel
<code>\newpage</code>	Seitenwechsel

In der Mathematik-Umgebung gibt es noch besondere Abstands-Befehle (s.u.)!

3.12 Seitenaufteilung

Soll der Text in mehreren Spalten, wie bei einem Zeitungsartikel, ausgegeben werden, verwendet man den Befehl `multicols`. Dazu muss auch das Paket `multicol` geladen werden, also vor Dokumentbeginn:

```
\usepackage{multicol}
```

und dann

```
\begin{multicols}{2}
links\\ links\\ links\\
```

```
rechts\\ rechts\\ rechts\\
\end{multicols}
```

ergibt zweispaltigen Druck:

links	rechts
links	rechts
links	rechts

Die Parameter für die Seiteneinteilung lassen sich festlegen, z.B. die Breite:

```
\setlength{\textwidth}{10cm}
```

oder vom Ausgangswert ausgehend verändern:

```
\addtolength{\textwidth}{10cm}
```

bzw.

```
\addtolength{\textwidth}{-4cm}
```

(Im Anhang zu diesem Skript sind alle Parameter für die Seiteneinteilung dargestellt.)

Das Paket `a4wide` erzeugt automatisch einen breiteren Textsatz.

Das `geometry`-Paket bietet eine noch flexiblere Seiteneinteilung.

Noch mehr Möglichkeiten zur kreativen Seitenaufteilung bieten Boxen (s.u.) und die `minipage`-Umgebung:

```
\begin{center}
```

```
\framebox{
```

```
\begin{minipage}[t]{7cm}
```

der Inhalt der `mini`-page wird auf eine kleinere Seite geschrieben

und entsprechend umgebrochen

```
\end{minipage}
```

```
}
```

```
\end{center}
```

der Inhalt der `minipage` wird auf eine kleinere Seite geschrieben und entsprechend umgebrochen

3.13 Schriftgrößen und Schriftarten

L^AT_EX wählt als Voreinstellung (außer in der Dokumentklasse `slides`) eine Serifenschrift der Größe 10pt. (Die Voreinstellung der Größe haben wir oben schonmal als Option der Dokumentklasse von 10pt auf 12pt verändert.)

Alle eigenen Änderungen der Schrift im Dokument beziehen sich auf diese Voreinstellung, z.B. werden Größenangaben relativ zur Voreinstellung benannt:

<code>normale Schrift</code>	normale Schrift
<code>{\tiny winzig}</code>	<small>winzig</small>
<code>{\small klein}</code>	<small>klein</small>
<code>{\large groß}</code>	groß
<code>{\Large größer}</code>	größer
<code>{\huge riesig}</code>	riesig
<code>{\Huge gigantisch}</code>	gigantisch

Dabei dienen die Befehle als Schalter zum Umschalten in eine andere Schriftgröße, und wirken auf den durch geschweifte Klammern begrenzten Text.

Auch die Schriftart lässt sich im Dokument verändern:

<code>normale Schrift</code>	normale Schrift
<code>\texttt{Maschinenschrift}</code>	Maschinenschrift
<code>\textbf{fettgedruckt}</code>	fettgedruckt
<code>\textit{Kursiv}</code>	<i>kursiv</i>
<code>\emph{hervorgehoben}</code>	<i>hervorgehoben</i>
	im Vergleich zur normalen Schrift

```
\textit{Wenn die normale Schrift kursiv ist, wird \emph{andersrum
hervorgehoben}!}}
```

Wenn die normale Schrift kursiv ist, wird andersrum hervorgehoben!

3.14 Farben

Wenn man das Paket `color` oder besser das neuere Paket `xcolor` verwendet, kann man Text farbig schreiben mit:

Text in `\textcolor{blue}{blau}` und schwarz

Text in blau und schwarz

oder farbig hinterlegen mit:

Text vor `\colorbox{yellow}{gelb}` und weiß

Text vor `gelb` und weiß

oder farbig hinterlegen mit Rand:

Text vor `\fcolorbox{red}{yellow}{gelb mit rotem Rand}` und weiß

Text vor `gelb mit rotem Rand` und weiß

Wenn man das `xcolor` Paket mit der Option `usenames,dvipsnames` verwendet, können auch vordefinierte Farben, wie z.B.

`\textcolor{BrickRed}{Ziegelrot}` und `\textcolor{CornflowerBlue}{Kornblumenblau}`

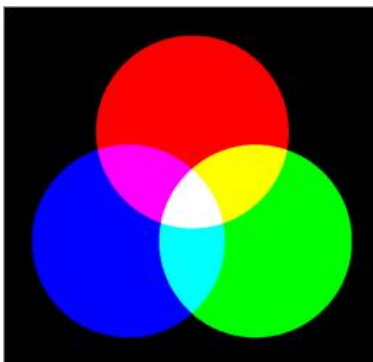
Ziegelrot und Kornblumenblau

verwendet werden (vgl. Anhang oder z.B. http://en.wikibooks.org/wiki/LaTeX/Colors#Predefined_colors).

Außerdem kann man auch Farben selbst definieren mit:

`\definecolor{Name der neuen Farbe}{rgb}{1,1,0}`

Farben werden so mit additiver Farbmischung im RGB-System (rot, grün, blau) erzeugt, d.h. als ob man buntes Licht übereinander projiziert:



Setzt man alle Regler der neuen Farbe auf 0 (niedrigster Wert)

`\definecolor{schwarz}{rgb}{0,0,0}`

erhält man schwarz; setzt man alle auf 1 (höchster Wert) erhält man weiß. Dazwischen sind der Kreativität keine Grenzen gestetzt. Die neuen Farben können später einfach mit ihren Namen verwendet werden, z.B.

```
\definecolor{meineFarbe}{rgb}{0.6,1,0.7}
\colorbox{meineFarbe}{Hintergrund in meiner Farbe.}
```

Hintergrund in meiner Farbe.

4 Mathematische Formeln

Mit L^AT_EX kann man besonders schöne Formeln schreiben. Zur Verfügung stehen alle möglichen Sonderzeichen, die allerdings nur in der Mathematik-Umgebung funktionieren:

```
\begin{displaymath}
```

...hier können Formeln stehen...

```
\end{displaymath}
```

Dazu äquivalent ist die Abkürzung

```
$$
```

...hier können Formeln stehen...

```
$$
```

Sollen die Formeln Durchnumeriert werden verwendet man:

```
\begin{equation}
```

...hier können nummerierte Formeln stehen...

```
\end{equation}
```

Wenn man im fließenden Text mathematische Symbole verwenden will, müssen sie von Dollarzeichen umschlossen sein, z.B. wird aus

```
...das $\alpha$-Männchen pirschte sich langsam an...
```

...das α -Männchen pirschte sich langsam an!

Exponenten und Indizes werden folgendermaßen angehängt (wenn sie mehrere Zeichen umfassen sollen unbedingt in geschweiften Klammern!):

`$X^2, X_2 \;` und `\;`; `X_{1-\alpha}`

ergeben X^2, X_2 und $X_{1-\alpha}$.

Abstände müssen in der Mathematik-Umgebung gekennzeichnet werden durch:

<code>\,</code>	kleiner Abstand
<code>\;</code>	mittlerer Abstand
<code>\quad</code>	großer Abstand

Wichtige mathematische Symbole:

(+, -, =, <, >, (und) tippt man normal auf der Tastatur)

<code>\alpha \;</code> <code>\beta \;</code> <code>\gamma</code>	$\alpha \beta \gamma$
<code>\cdot \;</code> <code>\div</code>	$\cdot \div$
<code>\le \;</code> <code>\ge \;</code> <code>\preceq \;</code> <code>\succeq</code>	$\leq \geq \preceq \succeq$
<code>\cap \;</code> <code>\cup</code>	$\cap \cup$
<code>\prod \;</code> <code>\sum</code>	$\prod \sum$
<code>\langle \;</code> <code>\rangle</code>	$\langle \rangle$
<code>\lbrace \;</code> <code>\rbrace</code>	$\{ \}$
<code>\emptyset \;</code> <code>\forall \;</code> <code>\infty</code>	$\emptyset \forall \infty$
<code>\mathds{R} \;</code> <code>\mathds{N}</code>	$\mathbb{R} \mathbb{N}$
<code>\hat{\pi} \;</code> <code>\widehat{P}</code>	$\hat{\pi} \hat{P}$
<code>\stackrel{a}{\sim}</code>	$\overset{a}{\sim}$
<code>\frac{a}{b}</code>	$\frac{a}{b}$

Für \mathbb{R} und \mathbb{N} braucht man das Paket `dsfont`.

Werden Zähler und Nenner eines Bruches (z.B. bei verschachtelten Brüchen) so klein, dass man sie mit `frac` nicht mehr lesen kann, kann man `dfrac` aus dem Paket `amsmath` verwenden (nicht vergessen: dazu muß das Paket in der Präambel geladen werden).

Um z.B. ein fett gedrucktes α zu erzeugen, braucht man auch das Paket `amsmath` und den Befehl:

`\boldsymbol{\alpha}`

4.1 Matrizen

Matrizen lassen sich mit L^AT_EX sehr übersichtlich schreiben - solange man selbst im Quelltext den Überblick behält...

Eine Matrix ist ein Feld mit einer bestimmten Anzahl Zeilen und Spalten. Im Quelltext müssen wir festlegen:

1. ob das Feld mit Klammern o.ä. begrenzt werden soll
2. wie viele Spalten es haben soll
3. wie viele Zeilen es haben soll

Das Grundgerüst einer Matrix sieht so aus:

```
\begin{displaymath}
Begrenzung links
  \begin{array}{Anzahl der Spalten festlegen}
    Zeile 1
    Zeile 2
    Zeile 3
  \end{array}
Begrenzung rechts
\end{displaymath}
```

Als **Begrenzung** wählt man bei Matrizen Klammern, bei Determinanten Striche. Sie werden gesetzt mit den Befehlen

```
\left( und \right)
```

bzw.

```
\left| und \right|
```

Die **Anzahl der Spalten** wird festgelegt, indem man für jede Spalte einen der Buchstaben **c**, **r**, oder **l** einsetzt, und zwar:

c (für zentrierte Einträge)

r (für rechtsbündige)

l (für linksbündige)

Also z.B. für drei Spalten mit zentrierten Einträgen:

```
\begin{array}{ccc}
```

Die Zeilen werden jetzt in die **array**-Umgebung eingesetzt. Dabei ist wichtig, dass die Anzahl der Einträge mit der festgelegten Anzahl der Spalten übereinstimmt. Die Einträge werden durch **&** als Trennzeichen getrennt; nach jeder Zeile muss der Zeilenumbruch durch **** gekennzeichnet werden, z.B.:

A & B & C \\\

Also nochmal im Ganzen:

```
\begin{displaymath}
\left|
\begin{array}{cccc}
A & B & C & D \\\
E & F & G & H \\\
I & J & K & L \\\
\end{array}
\right.
\end{displaymath}
```

ergibt:

$$\left| \begin{array}{cccc} A & B & C & D \\ E & F & G & H \\ I & J & K & L \end{array} \right|$$

4.2 Komplizierte Gleichungen

Was wir gerade über Matrizen gelernt haben, kann man sich auch zunutze machen, um z.B. Gleichungen mit Fallunterscheidungen zu schreiben.

Diese frei erfundene Gleichung

$$f(z) = \begin{cases} c_0 + c_1(z-a) + c_2(z-a)^2 + c_n(z-a)^n \\ c_2(z-a)^2 + c_n(z-a)^n \\ c_n(z-a)^n \end{cases}$$

enthält nämlich auch ein `array`:

```
\begin{displaymath}
f(z)=\left\{
\begin{array}{r}
c_0+c_1(z-a) + c_2(z-a)^2 + c_n(z-a)^n \\\
c_{-1}(z-a)^{-1} + c_{-2}(z-a)^{-2} \\\
c_{-n}(z-a)^{-n}
\end{array}
\right.
\end{displaymath}
```

Es hat eine Spalte mit rechtsbündigen Einträgen. Links wird es durch eine geschweifte Klammer begrenzt (deren Symbol erst mit `\` entwertet werden muss, weil es normalerweise für Befehle reserviert ist). Rechts soll es gar nicht begrenzt werden; damit die Umgebung abgeschlossen ist, wird sie mit einem Punkt beendet, der aber nicht im Output erscheint.

4.3 Mehrzeilige Gleichungssysteme

Gleichungssysteme, bei denen z.B. das Gleichheitszeichen immer übereinander stehen soll, werden als `eqnarray` geschrieben. dabei werden die selben Trennzeichen verwendet wie bei einem `array`.

```
\begin{eqnarray}
  y&=&ax^2+bx+c\\
  &=&az^3+\gamma
\end{eqnarray}
```

$$y = ax^2 + bx + c \tag{1}$$

$$= az^3 + \gamma \tag{2}$$

Die Aufteilung entspricht dabei einem `array` mit der Spalten-Einteilung `{rcl}`.

Sollen einzelne Zeilen nicht numeriert werden, verwendet man den `nonumber`-Befehl vor dem Zeilenende, um die Numerierung zu unterdrücken.

```
\begin{eqnarray}
  y&=&ax^2+bx+c \nonumber\\
  &=&az^3+\gamma
\end{eqnarray}
```

$$y = ax^2 + bx + c$$

$$= az^3 + \gamma \tag{3}$$

Um die Numerierung zu ändern, kann auf den Zählerstand zugegriffen werden:

```
\setcounter{equation}{0}
```

```
\begin{eqnarray}
  y&=&ax^2+bx+c \nonumber\\
  &=&az^3+\gamma
\end{eqnarray}
```

$$\begin{aligned}
 y &= ax^2 + bx + c \\
 &= az^3 + \gamma
 \end{aligned}
 \tag{1}$$

Soll die Numerierung ganz weggelassen werden, verwendet man den Befehl `eqnarray*`.

5 Gleitobjekte

Gleitobjekte werden in den laufenden Quelltext geschrieben, aber von L^AT_EX so positioniert, dass die Seitenaufteilung des Dokumentes optimal ist. Tabellen und Grafiken können als Gleitobjekte eingebunden werden.

5.1 Tabellen

Hier wird zunächst erklärt, wie Tabellen erzeugt werden, so dass sie an der Stelle im Text erscheinen, an der sie stehen; danach wird gezeigt, wie sie als Gleitobjekte eingebunden werden.

Tabellen werden ähnlich wie mathematische Arrays erzeugt:

```

\begin{tabular}{Anzahl der Spalten festlegen}
  Zeile 1
  Zeile 2
  Zeile 3
\end{tabular}

```

Jede Spalte wird wieder durch einen der Buchstaben `c`, `l` oder `r` festgelegt (siehe oben). Zwischen den Spalten können eine oder zwei vertikale Linien gesetzt werden, durch `|` und `||`, z.B.:

```

\begin{tabular}{|c|c|c||r|}

```

Die Zeileneinträge werden wieder durch `&` getrennt; jede Zeile wird mit dem Zeilenumbruch abgeschlossen. Am oberen und unteren Rand der Tabelle und nach jeder Zeile können horizontale Linien gezogen werden, mit dem Befehl:

```

\hline

```

Beispiel:

```

\begin{tabular}{|c|c|c||r|}
  \hline
  3&4&1&8\\
  \hline
  4&5&6&15\\
  \hline
  \hline
  7&9&7&23\\
  \hline
\end{tabular}

```

3	4	1	8
4	5	6	15
7	9	7	23

Für längeren Text gibt es auch die Möglichkeit, eine Spalte mit fester Breite zu verwenden, in der der Text automatisch umgebrochen wird:

```

\begin{tabular}{|l|p{6cm}|}
  \hline
  Stufe 1 & In Stufe 1 werden die Grundlagen gelegt.\\
  \hline
  Stufe 2 & In Stufe 2 werden aufbauend auf den Grundlagen
             die genauen Abläufe spezifiziert.\\
  \hline
\end{tabular}

```

Stufe 1	In Stufe 1 werden die Grundlagen gelegt.
Stufe 2	In Stufe 2 werden aufbauend auf den Grundlagen die genauen Abläufe spezifiziert.

5.1.1 Verschachtelte Tabellen

Soll ein Eintrag, z.B. eine Überschrift, mehrere Zeilen überspannen, nimmt man:

```
\multicolumn{Anzahl Spalten}{Aussehen}{Inhalt}
```

also z.B.:

```

\begin{tabular}{c|c|c|c}
  A&B&\multicolumn{2}{c}{C}\\
  \hline
  1&2&3&4\\
\end{tabular}

```

```

\hline
5&6&7&8\\
\end{tabular}

```

A	B	C	
1	2	3	4
5	6	7	8

Horizontale Linien, die nicht über die gesamte Breite der Tabelle gehen, werden mit `cline` erzeugt:

```

\begin{tabular}{c|c|c|c}
A&B&\multicolumn{2}{c}{C}\\
\cline{3-4}
1&2&3&4\\
\hline
5&6&7&8\\
\end{tabular}

```

A	B	C	
1	2	3	4
5	6	7	8

Soll ein Eintrag, z.B. eine seitliche Überschrift, mittig zwischen zwei Zeilen gesetzt werden, wird er aus der unteren Zeile nach oben gerückt:

```

\begin{tabular}{c|cc|cc}
&A&B&\multicolumn{2}{c}{C}\\
\hline
&1&2&3&4\\
\raisebox{1.5ex}{-1.5ex}{D}&5&6&7&8\\
\end{tabular}

```

	A	B	C	
	1	2	3	4
D	5	6	7	8

Weitere gute Beispiele für komplexe Tabellen siehe: <http://en.wikibooks.org/wiki/LaTeX/Tables>

5.1.2 Tabellen als Gleitobjekte

Tabellen können mit L^AT_EX folgendermaßen als Gleitobjekte eingesetzt werden:

Tabellen werden durch die `table`-Umgebung zu Gleitobjekten:

```
\begin{table}[Positionierungs-Vorschläge]
  hier steht die Tabelle
\end{table}
```

Man kann bei Gleitobjekten Positionierungs-Vorschläge angeben, die eingehalten werden, wenn dadurch keine Probleme beim Seitenumbruch entstehen:

h (here - möglichst hier ausgeben)
t (top - auf der nächstmöglichen Seite oben)
b (bottom - auf der nächstmöglichen Seite unten)
p (page - alle auf einer extra Seite ausgeben).

Mit dem Befehl `caption` kann eine Tabellenüber- oder -unterschrift gewählt werden. Durch ein Label kann auf die Tabelle im text und in einem Verzeichnis Bezug genommen werden.

```
\begin{table}[h]
\centering
\caption{\label{ersteTab} Tabelle Nr. 1}
  \begin{tabular}{|c|}
    \hline
    mal sehen, wo Tabelle 1 landet ...\\
    \hline
  \end{tabular}
\end{table}
```

Tabelle 1: Tabelle Nr. 1

mal sehen, wo Tabelle 1 landet ...

... und wo Tabelle 2 landet

Tabelle 2: Tabelle Nr. 2

```

\begin{table}[t]
\centering
  \begin{tabular}{|c|}
    \hline
    ... und wo Tabelle 2 landet \\
    \hline
  \end{tabular}
\caption{\label{zweiteTab} Tabelle Nr. 2}
\end{table}

```

Im Text kann auf die Tabellen Nummer `\ref{ersteTab}` und `\ref{zweiteTab}` Bezug genommen werden.

Im Text kann auf die Tabellen Nummer 1 und 2 Bezug genommen werden. Auch hier kann der Zähler z.B. mit

```
\setcounter{table}{42}
```

verändert werden.

5.2 Grafiken

Auch Grafiken können als Gleitobjekte eingefügt werden. Um sie in einer dvi- oder ps-Datei zu verwenden, müssen sie im Format `.eps` vorliegen, für eine pdf-Datei im Format `.pdf` oder auch `.jpg`, wenn man zum setzen den Befehl `PDFLATEX` verwendet.

Bei Programmen zum einscannen oder bearbeiten von Bildern kann man viele Dateiformate zum abspeichern auswählen, oder in sie exportieren (die Erzeugung von Grafiken in R wird unten kurz exemplarisch behandelt). Die Bild-Datei muss beim Setzen im selben Verzeichnis liegen wie der Quelltext, ansonsten muss man den Datei-Pfad mit angeben!

Um Grafiken einbinden zu können braucht das Paket `graphicx`.

In der `figure`-Umgebung gelten die selben Positionierungs-Vorschläge wie in der `table`-Umgebung (siehe oben). Die Bild-Datei wird mit `includegraphics` eingefügt; dabei kann man auch die gewünschte Breite des Bildes, und damit die Gesamtgröße, bestimmen.

Auch Bildunterschriften können wider mit `caption` und einem Label eingefügt werden.

Sinnvolle Angewohnheit: nur der Name der Bild-Datei angeben, nicht die Datei-Endung, damit man aus dem Quelltext sowohl `.ps` als auch `.pdf` Dateien erzeugen kann (z.B.: die Bild-Datei liegt als `bild.eps` und als `bild.pdf` vor - im Quelltext wird sie als `bild` eingesetzt):

```
\begin{figure}[h]
\centering
\includegraphics[width=2.5cm]{bild}
\caption{\label{Bild1} Bild Nr. 1}
\end{figure}
```

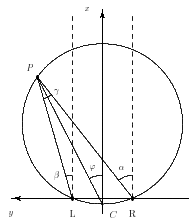


Abbildung 1: Bild Nr. 1

Im Text kann auf Bild `\ref{Bild1}` Bezug genommen werden.

Im Text kann auf Bild 1 Bezug genommen werden.

5.2.1 Exkurs: Grafiken in R erzeugen

Im Statistik-Paket R können dort erzeugte Diagramme und Plots einfach z.B. in eine pdf-Datei geschrieben, und dann in L^AT_EX eingebunden werden.

Eine neue pdf-Datei wird zur Ausgabe geöffnet mit:

```
pdf("name.pdf")
```

Dann werden mit R Befehlen Grafiken erzeugt.

Abschließend wird die Datei (als sog. device für die Ausgabe) wieder geschlossen mit:

```
dev.off()
```

Beispiel:

```
>daten <- read.table("daten.txt",header=T)
>daten
      x  y
1    2  5
2    3  8
3    1  4
>pdf("streudiagramm.pdf")
>plot(x,y)
>dev.off()
```

5.2.2 Grafiken direkt im L^AT_EX Dokument erzeugen

Innerhalb eines L^AT_EX Dokuments kann man mit PicT_EX einfache Zeichnungen oder mit dem umfangreicheren TikZ auch komplexe Schaubilder (sowie z.B. den Text überlagernde Pfeile und Umrahmungen) einfügen. Beispiele siehe: <http://www.texample.net/tikz/examples/all/>

5.3 Rotierte Gleitobjekte

Wenn z.B. eine Tabelle zu breit für die Seite ist, kann man sie (mitsamt der Tabellenunterschrift) drehen. Dafür stehen mit dem Paket `rotating` die folgenden Befehle zur Verfügung, z.B. um die gesamte Tabelle um 90° im Uhrzeigersinn zu drehen:

```
\begin{sidewaystable}
\centering
  \begin{tabular}{...}
    ...
  \end{tabular}
  \caption{\label{dritteTab} Tabelle Nr. 3}
\end{sidewaystable}
```

Entsprechend gibt es auch den Befehl `sidewaysfigure`, und die Umgebungen `sideways`, die alle eingeschlossenen Elemente um 90° im Uhrzeigersinn dreht, und `rotate`, bei der man den Rotationswinkel selbst mit angeben kann:

```
\begin{rotate}{40}
das wird rotiert
\end{rotate}
```

das wird rotiert

class probability p		0.5			0.2			0.05					
allele probabilities (π_1, π_2)		(0.6,0.6)	(0.6,0.8)	(0.6,0.95)	(0.8,0.8)	(0.6,0.6)	(0.6,0.8)	(0.6,0.95)	(0.8,0.8)	(0.6,0.6)	(0.6,0.8)	(0.6,0.95)	(0.8,0.8)
categorical	naive	0.4814	0.4710	0.4211	0.4303	0.4921	0.4708	0.4182	0.4379	0.4897	0.4496	0.4250	0.4376
	corrected	0.0531	0.0510	0.0417	0.0472	0.0569	0.0541	0.0552	0.0488	0.0549	0.0545	0.0909	0.0612
ordered	naive	0.3514	0.3083	0.2552	0.2625	0.3545	0.3118	0.2555	0.2700	0.3481	0.3054	0.2653	0.2686
	corrected	0.0510	0.0511	0.0507	0.0492	0.0511	0.0530	0.0497	0.0524	0.0529	0.0520	0.0506	0.0503

Tabelle 3: Tabelle Nr. 3

6 Verzeichnisse

Bei Verzeichnissen, wie z.B. auch beim Inhaltverzeichnis, werden Referenzen und Seitenzahlen von L^AT_EX gesammelt und erst beim zweiten Durchlauf des Programms ins Verzeichnis übernommen (z.B. beim Texmaker kann man sich aber wie oben beschrieben eine Abfolge von Befehlen zusammenstellen und als Quick Build abspeichern).

6.1 Literaturverzeichnis

Hier werden zunächst die technischen Prinzipien zum Einbinden von Referenzen besprochen; weiter unten werden verschiedene Zitierstile zur optischen Verschönerung der Einträge vorgestellt. Es besteht die Möglichkeit, die (wenigen) Literaturangaben direkt im aktuellen Dokument anzugeben. Für längere Literaturlisten ist es aber sinnvoller, eine Literaturdatenbank anzulegen, die man nach und nach ergänzen und weiter verwenden kann.

6.1.1 Literaturverzeichnis im Dokument

Das Verzeichnis wird wie eine Aufzählung angelegt:

```
\begin{thebibliography}{Einrücktiefe-Mustertext}
  \bibitem[Name1]{Kennwort1}Literaturangabe 1
  \bibitem[Name2]{Kennwort2}Literaturangabe 2
\end{thebibliography}
```

Die Literaturangaben werden im Text mit

```
\cite{Kennwort1}
```

fortlaufende Nummer, wenn [Name1] weggelassen wurde, oder mit der in eckigen Klammern verwendeten Kurzbezeichnung

Das Literaturverzeichnis erhält automatisch eine Überschrift. Die Einrücktiefe des Literaturverzeichnisses wird durch einen Mustertext festgelegt, z.B. den längsten als Name verwendeten Ausdruck.

```
\begin{thebibliography}{C.Cleo, 1993}
  \bibitem[Alfred:1991]A.Alfred, 1991, \emph{Ameisen}: Akazien-Verlag
  \bibitem[Bert:1992]B.Bert, 1992, \emph{Biohazards}: Buch-Verlag
  \bibitem[C.Cleo, 1993]{Cleo:1993}C.Cleo, 1993, \emph{Chemie}: Cyan-Verlag
\end{thebibliography}
```

Literatur

[1] A.Alfred, 1991, *Ameisen*: Akazien-Verlag

[2] B.Bert, 1992, *Biohazards*: Buch-Verlag

[C.Cleo, 1993] C.Cleo, 1993, *Chemie*: Cyan-Verlag

Hier zitiere ich zunächst Autor 1 `\cite{Alfred:1991}`, und dann Autor 2 `\cite{Bert:1992}` und Autor 3 `\cite{Cleo:1993}`.

Hier zitiere ich zunächst Autor 1 [1], und dann Autor 2 [2] und Autor 3 [C.Cleo, 1993].

6.1.2 Literaturdatenbank mit BibT_EX

Wenn mehrere Dokumente bzw. mehrere Benutzer auf Einträge in einer Literaturdatenbank zurückgreifen können sollen, verwendet man BibT_EX.

Dazu wird eine extra Literaturdatenbank als Datei `datenbankname.bib` angelegt. Die Einträge darin müssen folgendem Format, z.B. für Bücher, entsprechen:

```
@BOOK{Kennwort1,
  AUTHOR={Autor1},
  TITLE={Titel1},
  PUBLISHER={Verlag1},
  ADDRESS={Ort1},
  YEAR={Jahr1}
}
```

(Weitere Beispiele finden Sie in den Kursmaterialien.)

Im eigentliche Dokument legt man z.B. mit

```
\bibliographystyle{plain}
```

den Zitierstil fest und zitiert wieder mit

```
\cite{Kennwort1}
```

Das Literaturverzeichnis, das nur die hier zitierten Einträge aus der Literaturdatenbank enthält, wird eingesetzt mit:

```
\bibliography{datenbankname}
```

Möchte man aus irgendeinem Grund Einträge aus der Literaturdatenbank im Literaturverzeichnis ausgeben lassen, die nicht im Text zitiert werden, verwendet man den Befehl:

`\nocite{Kennwort1}`

Um dann das Dokument zu erzeugen muss man vom Prinzip her

1. PDFL^AT_EX aufrufen, damit der Dateiname der Literatudatenbank `datenbankname` aus dem `bibliography`-Befehl gelesen wird
2. Bib_TE_X aufrufen, um die Einträge aus der Literturdatenbank zu übernehmen
3. mit PDFL^AT_EX (zweimal!) das Dokument mit Literaturangaben und Literaturverzeichnis erzeugen.

Diese Abfolge von Befehlen kann man sich z.B. bei Texmaker aber auch als Quick Build abspeichern.

6.1.3 Aussehen der Literaturangaben verändern

Zur Darstellung von Literaturangaben stehen verschiedene Zitierstile (Dateien der Art `bibstylename.bst` für sog. Bibliographystyles), zur Verfügung.

`\bibliographystyle{plain}`

mit dem Befehl

`\cite{Kennwort1}`

ergibt: [1].

bei Verwendung von

`\usepackage{natbib}`

stehen folgende Darstellungsformen bereit:

`\bibliographystyle{plainnat}`

mit dem Befehl

`\citep{Kennwort1}`

ergibt: [Author1, Jahr1].

`p` steht hier für parenthesis (Klammern). Dieser Befehl ist für Literaturangaben der Form

Das XY-Paradigma [Skinner, 1950] hatte großen Einfluß...

geeignet. Die alternative Form für Literaturangaben im fortlaufenden Text ist

`\bibliographystyle{plainnat}`

mit dem Befehl

`\citet{Kennwort1}`

ergibt: Author1 [Jahr1].

`t` steht hier für Text. Dieser Befehl ist für Literaturangaben der Form

Wie schon Skinner [1950] berichtet...

geeignet.

Runde Klammern gibt es z.B. bei den `bibliographystyles` `chicago.bst`.

Der `bibliographystyle` legt auch fest, in welcher Reihenfolge und Darstellung die Einträge im Literaturverzeichnis erscheinen. Manchmal wird dabei besonderer Wert auf den Zitierstil der Artikel- und Buchtitel gelegt:

Man unterscheidet zwischen *title style* (Großschreibung wie z.B. bei Zeitungüberschriften) und *sentence style* (normale Groß- und Kleinschreibung). Da Großbuchstaben einfach in Kleinbuchstaben umzusetzen sind, man umgekehrt aber Probleme kriegen kann, sollte man beim anlegen der Literaturdatenbank

1. alle Titel im *title style* aufnehmen, z.B.:

The Unbearable Lightness of Being

2. wählt man einen `bibliographystyle` mit *title style* (z.B. `jss.bst` der online-Zeitschrift Journal of Statistical Software) wird der Titel auch genau so ausgegeben
3. wählt man hingegen einen `bibliographystyle` mit *sentence style* (z.B. `chicago.bst`) erhält man

The unbearable lightness of being

4. wichtig: Buchstaben, die weiterhin groß gesetzt werden sollen, wie z.B. der erste Buchstaben eines Substantives im Deutschen, müssen extra durch geschweifte Klammern geschützt werden, sonst wird aus

Applied Statistics with R im *title style*

Applied statistics with r im *sentence style*

Wer einen ganz besonderen Zitierstil braucht bzw. will, kann in einer Shell bzw. Eingabeaufforderung auch mit

```
latex makebst
```

einen eigenen `bibliographystyle` erzeugen: anhand eines Dialogs werden die Präferenzen zum Zitierstil abgefragt, und eine Datei `meinstil.dbj` erzeugt. Diese Datei wird mit

```
latex meistil.dbj
```

in die Datei `meinstil.bst` übersetzt, die man wie gewohnt als `bibliographystyle` verwenden kann.

6.2 APA-Zitierstil

Speziell für Abschlussarbeiten und Artikel, die im Stil der American Psychological Association (APA) formatiert werden sollen, empfiehlt sich die Verwendung des Zitierstils `apacite`.

Um diesen Stil zu verwenden bindet man in der Präambel das `apacite`-Paket ein:

```
\usepackage{apacite}
```

Am Ende des Dokuments erzeugt man das Literaturverzeichnis mit dem `apacite` Zitierstil:

```
\bibliographystyle{apacite}
\bibliography{datenbankname}
```

Die wichtigsten Befehle zum Zitieren mit `apacite` sind für die Zitierung in Klammern:

```
\cite<siehe>[S.~11]{Kennwort1}           (siehe Autor1, Jahr1, S. 11)
```

wobei hier ein etwaiger Text vor der Referenz in Pfeilkammern, ein Text nach der Referenz in eckigen Klammern angegeben wird, oder für die Zitierung im laufenden Text:

```
wie schon \citeA[S.~12]{Kennwort2} sagt   wie schon Autor1 (Jahr1, S. 12) sagt
```

Wird eine Referenz mit mehreren Autoren mehrfach verwendet wird bei `apacite` und den meisten anderen Zitierstilen ab der zweiten Nennung automatisch die Abkürzung *et al.* eingesetzt.

Eine detaillierte Beschreibung mit vielen weiteren Optionen liefert die Dokumentation zum `apacite`-Paket. Es unterstützt auch das Anlegen von Autorenverzeichnissen u.ä.

Neben dem APA-Zitierstil gibt es auch die `apa`-Dokumentklasse für die Einreichung von Artikeln bei Zeitschriften, die der APA-Norm folgt.

6.3 Andere Verzeichnisse

Mit den Befehlen

```
\listoffigures
```

und

```
\listoftables
```

können ganz einfach von allen Gleitobjekten, die mit einer `caption` versehen sind, Verzeichnisse angelegt werden (vgl. Abschnitt zu Gleitobjekten).

Tabellenverzeichnis

1	Tabelle Nr. 1	23
2	Tabelle Nr. 2	24
3	Tabelle Nr. 3	27

Abbildungsverzeichnis

1	Bild Nr. 1	25
---	----------------------	----

Mit den Paketen `apacite` und `index` können auch Autoren- und Stichwortverzeichnisse (z.B. für Bücher) erstellt werden.

7 Vortragsfolien mit L^AT_EX

Zusätzlich zu den Dokumentklassen für Schriftstücke gibt es die Klassen `slides`, `foils` und die umfangreiche Klasse `beamer` für Vortragsfolien.

Die Dokumentklasse `foils` bietet z.B. Folienüberschriften, die automatisch zentriert gesetzt werden. Außerdem stehen Felder für Namen, Titel o.ä. zur Verfügung.

Beispiel:

```
\documentclass[dvips,landscape]{foils}
\setlength{\parindent}{0pt}
\lefthead{Carolin Strobl}
\righthead{LMU München}
\MyLogo{Ein interessanter Vortrag}
\rightfoot{\thepage} %Seitennummerierung
```

```
\begin{document}
\foilhead{Vortrag}
...
\end{document}
```

(Das Resultat finden Sie im Beispiel-Ordner der Kursmaterialien.)

Bunte Folien lassen sich durch

```
\pagecolor{blue} Seitenhintergrund
\color{yellow} Textfarbe
```

erzeugen (wobei die Hintergrundfarbe nicht in allen Viewern, aber z.B. im AdobeReader angezeigt werden kann).

Professionellere Vortragsfolien lassen sich mit der Dokumentklasse `beamer` erzeugen (vgl. Beispiel-Präsentation in den Kursmaterialien).

8 Zusammengesetzte Dokumente

Bei längeren Dokumenten oder sich wiederholenden Elementen kann es sinnvoll sein, ein L^AT_EX Dokument aus verschiedenen Elementen zusammenzusetzen.

8.1 Einbinden von L^AT_EX-Quelltext

Wird z.B. immer der selbe Dokumentkopf mit Dokumentklasse und Pakete verwendet, oder sollen Teile einer anderen Datei `anderedatei.tex` im Dokument verwendet werden, eignet sich der Befehl:

```
\input{anderedatei}
```

8.2 Einbinden von R-Quelltext

Mit der Funktion `Sweave` in R können R-Programmcode, Output und Grafiken in ein L^AT_EX Dokument übernommen werden.

Das kann sehr praktisch sein, wenn z.B. im Datensatz etwas verändert wird: man muss dann nicht nochmal alles berechnen und nochmal alle Ergebnisse in das L^AT_EX Dokument kopieren, sondern kann in einem Schritt das neue Dokument erzeugen.

Ablauf:

- man erzeugt in einem Editor ein Dokument mit dem Namen `name.Snw`, das L^AT_EX Code und R-Code enthält (vgl. Beispiel in den Kursmaterialien)
- in R werden die enthaltenen R-Befehle ausgeführt

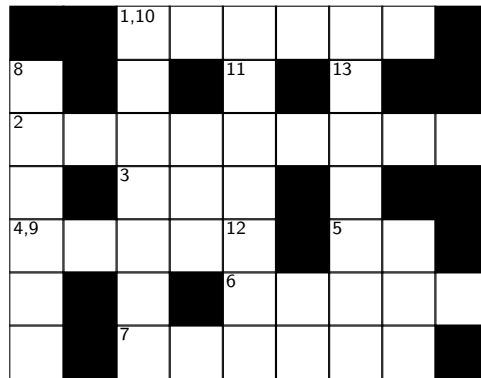
```
Sweave("name.Snw")
```

dadurch wird eine `.tex`-Datei erzeugt, die den Output, Grafiken, Text, Tabellen etc. enthält.

- diese `.tex`-Datei kann man dann ganz normal mit L^AT_EX oder PDFL^AT_EX setzen lassen

Wichtig: Wenn man an dem Dokument etwas ändern möchte, muss man das in der `Snw`-Datei tun, nicht in der `tex`-Datei – sonst sind nach dem nächsten `Sweave` alle Änderungen futsch, weil diese immer wieder überschrieben wird!

9 Kreuzworträtsel



waagerecht: 1 zentriert bzw. Zentrum 2 Studienfach mit Seltenheitswert 3 Normalverteilungssymbol; Klein-... macht auch Mist 4 Textsatzprogramm für Genießer 5 Mittelwert, griechischer Buchstabe und Kuh-Sprache 6 Befehl zum Einfügen einer anderen L^AT_EX Datei 7 Dokumentklasse für Berichte

senkrecht: 8 die Mistgabel unter den Schriftzeichen 9 kleiner oder gleich 10 höchster Gliederungsbeehl bei `book` 11 wahre Wahrscheinlichkeit und Kreiszahl 12 verschnörkelter griechischer Buchstabe; Schweineschwänzchen 13 Listenpunkt auch bei Aufzählungen

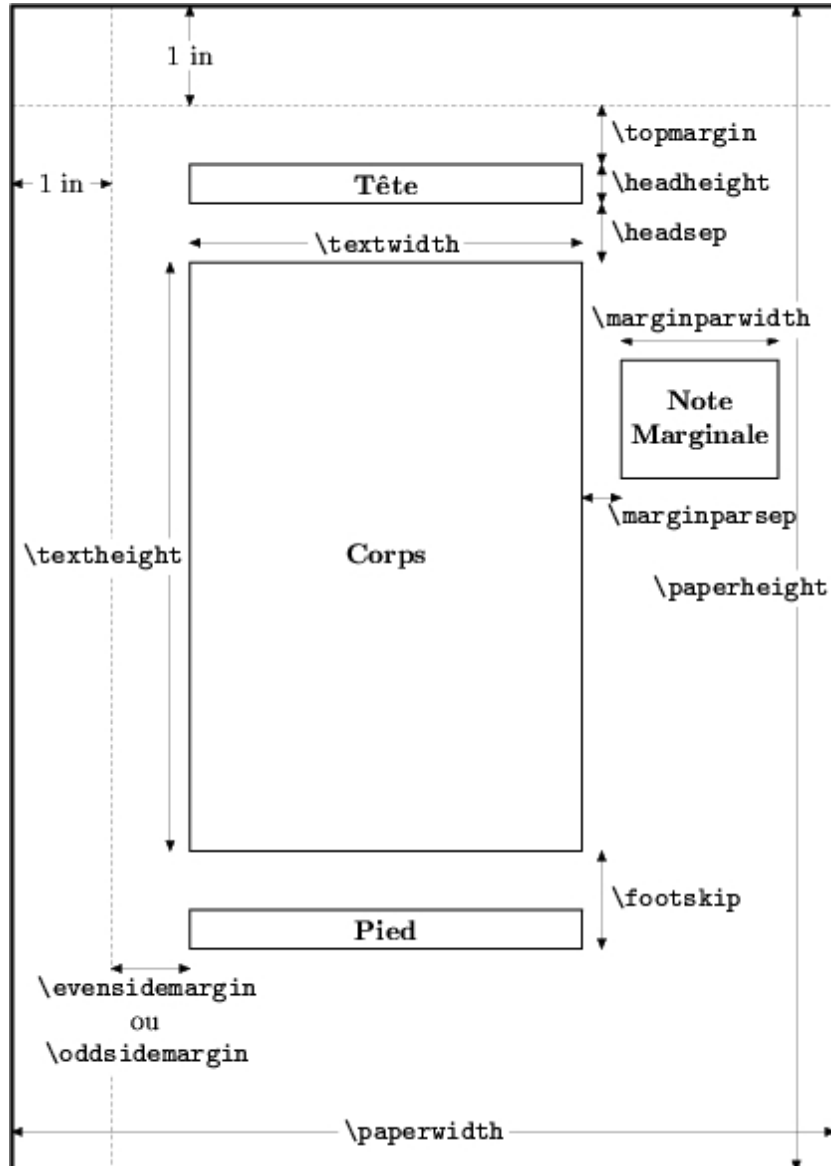
(Quelletext und Lösung im Anhang.)

Viel Spaß beim Ausprobieren!

A Farb-Namen für die `usenames` Option

Apricot	Aquamarine	Bittersweet	Black
Blue	BlueGreen	BlueViolet	BrickRed
Brown	BurntOrange	CadetBlue	CarnationPink
Cerulean	CornflowerBlue	Cyan	Dandelion
DarkOrchid	Emerald	ForestGreen	Fuchsia
Goldenrod	Gray	Green	GreenYellow
JungleGreen	Lavender	LimeGreen	Magenta
Mahogany	Maroon	Melon	MidnightBlue
Mulberry	NavyBlue	OliveGreen	Orange
OrangeRed	Orchid	Peach	Periwinkle
PineGreen	Plum	ProcessBlue	Purple
RawSienna	Red	RedOrange	RedViolet
Rhodamine	RoyalBlue	RoyalPurple	RubineRed
Salmon	SeaGreen	Sepia	SkyBlue
SpringGreen	Tan	TealBlue	Thistle
Turquoise	Violet	VioletRed	
WildStrawberry	Yellow	YellowGreen	YellowOrange

B Parameter zur Seitenaufteilung



C Kreuzworträtsel-Lösung

		^{1,10} C	E	N	T	E	R	
⁸ P		H		¹¹ P		¹³ I		
² S	T	A	T	I	S	T	I	K
		³ P	H	I		E		
^{4,9} L	A	T	E	¹² X		⁵ M	U	
		E		⁶ I	N	P	U	T
		⁷ R	E	P	O	R	T	

```

%im Dokumentkopf:
\usepackage{cwpuzzle}

%im Dokument:
%Kreuzworträtsel erzeugen:
\begin{Puzzle}{9}{7}
  |*|[1,10]C|E|N|T|E|R|*|. |[8]P|*|H|*|[11]P|*|[13]I|*|*|.
  |[2]S|T|A|T|I|S|T|I|K|. |I|*|[3]P|H|I|*|E|*|*|.
  |[4,9]L|A|T|E|[12]X|*|[5]M|U|*|. |E|*|E|*|[6]I|N|P|U|T|.
  |Q|*|[7]R|E|P|O|R|T|*|.
\end{Puzzle}
\begin{PuzzleClues}{\textbf{waagerecht:}}
  \Clue{1}{CENTER}{zentriert bzw. Zentrum}
  \Clue{2}{STATISTIK}{Studienfach mit Seltenheitswert}
  \Clue{3}{PHI}{Normalverteilungssymbol; Klein-... macht auch Mist}
  \Clue{4}{LATEX}{Textsatzprogramm für Genießer}
  \Clue{5}{MU}{Mittelwert, griechischer Buchstabe und Kuh-Sprache}
  \Clue{6}{INPUT}{Befehl zum Einfügen einer anderen \LaTeX\ Datei}
  \Clue{7}{REPORT}{Dokumentklasse für Berichte}
\end{PuzzleClues}
\begin{PuzzleClues}{\textbf{senkrecht:}}

```

```

\Clue{8}{PSI}{die Mistgabel unter den Schriftzeichen}
\Clue{9}{LEQ}{kleiner oder gleich}
\Clue{10}{CHAPTER}{höchster Gliederungsbefehl bei \texttt{book}}
\Clue{11}{PI}{wahre Wahrscheinlichkeit und Kreiszahl}
\Clue{12}{XI}{verschnörkelter griechischer Buchstabe;
Schweineschwänzchen}
\Clue{13}{ITEM}{Listenpunkt auch bei Aufzählungen}
\end{PuzzleClues}

```

%Lösung anzeigen:

```

\PuzzleSolution[true]
\begin{Puzzle}{9}{7}
  |*|[2]C|*|*|*|*|*|*|. |*|*|H|*|[3]P|*|[4]I|*|*|.
  |[1]S|T|A|T|I|S|T|I|K|. |*|*|P|*|*|*|E|*|*|.
  |[5]L|A|T|E|[6]X|*|[7]M|U|*|. |*|*|E|*|I|*|*|*|*|.
  |*|*|R|*|*|*|*|*|*|.
\end{Puzzle}

```