

# Einführung in das Textsatzsystem $\text{\LaTeX}$

## Diagramme

Moritz Brinkmann

`moritz.brinkmann@iwr.uni-heidelberg.de`

27. November 2015

# Übersicht

- Ein Diagramm ist eine grafische Darstellung von Daten, Sachverhalten oder Informationen.
- Information sollte dabei im Vordergrund stehen
- Diagramme sollten sich in das Dokument einfügen
  - passende Dimensionen
  - Beschriftung in gleicher Schriftart

Es existieren diverse spezialisierte Pakete

`chronosys` Satz von Zeitstrahlen

`histogr` (sehr simple) Histogramme

`bchart` einfache Balkendiagramme

`gnuplottex` Plots mit gnuplot (siehe Vorlesung Mathematiksat II)

`pgfplots` Umfangreiche Plot-Funktionalität mit TikZ

Es existieren diverse spezialisierte Pakete

`chronosys` Satz von Zeitstrahlen

`histogr` (sehr simple) Histogramme

`bchart` einfache Balkendiagramme

`gnuplottex` Plots mit gnuplot (siehe Vorlesung Mathematiksat II)

`pgfplots` Umfangreiche Plot-Funktionalität mit TikZ

`pgfplots` ist für fast alle Arten von Diagrammen zu empfehlen!

Konfiguration mittels `\pgfplotsset{Optionen}`. Paketautor empfiehlt, für zukünftige Kompatibilität, die aktuelle Version anzugeben.

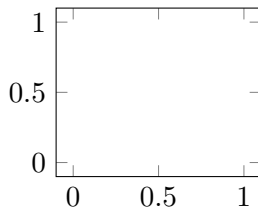
```
\usepackage{pgfplots}  
\pgfplotsset{compat=1.3}
```

Konfiguration mittels `\pgfplotsset{Optionen}`. Paketautor empfiehlt, für zukünftige Kompatibilität, die aktuelle Version anzugeben.

```
\usepackage{pgfplots}  
\pgfplotsset{compat=1.3}
```

**pgfplots** basiert auf TikZ/PGF und steht deshalb innerhalb einer `tikzpicture`:

```
\begin{tikzpicture}  
  \begin{axis}  
    ...  
  \end{axis}  
\end{tikzpicture}
```



# Achsentypen

Verschiedene Achsentypen verfügbar:

```
\begin{<Achsentyp>}[<Optionen>]  
  <Inhalt>  
\end{<Achsentyp>}
```

axis	lineare Koordinatenachsen
semilogyaxis	x-Achse linear, y-Achse logarithmisch
semilogxaxis	x-Achse logarithmisch, y-Achse linear
loglogaxis	beide Achsen logarithmisch
polaraxis	Polarkoordinaten <sup>*</sup>
smithchart	Smith-Diagramm <sup>†</sup>
ternaryaxis	Dreiecksdiagramm <sup>‡</sup>

---

<sup>\*</sup> mit `\usepgfplotslibrary{polar}`

<sup>†</sup> mit `\usepgfplotslibrary{smithchart}`

<sup>‡</sup> mit `\usepgfplotslibrary{ternary}`



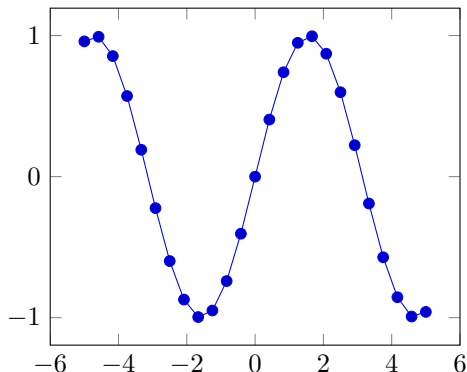
# Daten hinzufügen



```
\addplot [Optionen] {Eingabedaten};
```

```
\addplot+ [Optionen] {Eingabedaten};
```

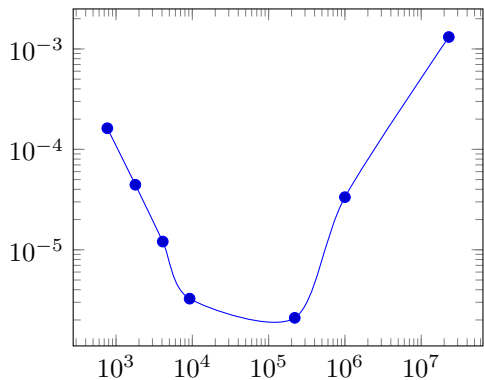
```
\begin{tikzpicture}  
  \begin{axis}  
    \addplot{sin deg(x)};  
  \end{axis}  
\end{tikzpicture}
```



# Koordinaten Eingabe

`\addplot [Optionen] coordinates {Koordinaten};`

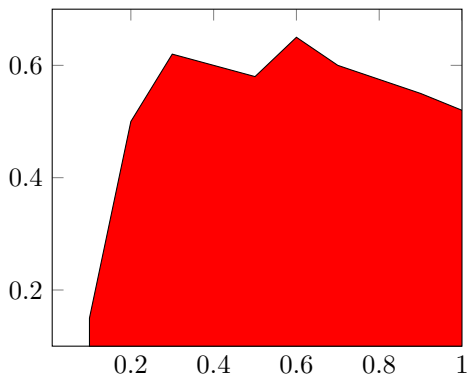
```
\begin{tikzpicture}
  \begin{loglogaxis}
    \addplot+[smooth]
      coordinates {
        (769, 1.6227e-04)
        (1793, 4.4425e-05)
        (4097, 1.2071e-05)
        (9217, 3.2610e-06)
        (2.2e5, 2.1E-6)
        (1e6, 0.00003341)
        (2.3e7, 0.00131415)
      };
  \end{loglogaxis}
\end{tikzpicture}
```



# Nachbearbeitung mit TikZ

`\addplot [Optionen] {Eingabedaten} ggf. TikZ-Befehle;`

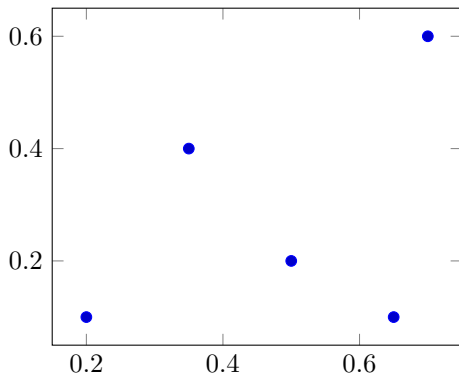
```
\begin{tikzpicture}
  \begin{axis}[xmax=1]
    \addplot [fill=red]
      coordinates
      {(0.1,0.15) (0.2,0.5)
      (0.3,0.62) (0.5,0.58)
      (0.6,0.65) (0.7,0.6)
      (0.9,0.55) (1,0.52)}
    \closedcycle;
  \end{axis}
\end{tikzpicture}
```





```
\addplot [<Optionen>] table [<Spalten-Auswahl>] {<Tabelle>;
```

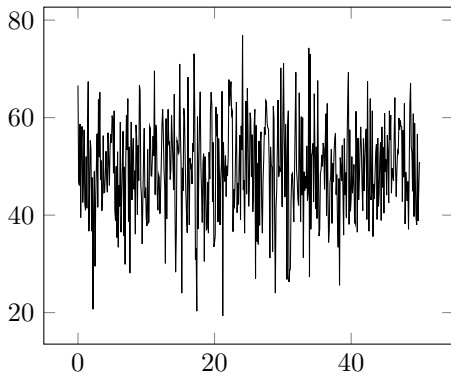
```
\begin{tikzpicture}
  \begin{axis}
    \addplot table [
      only marks,
    ] {
      x      y      myvalue
      0.5    0.2    0.25
      0.2    0.1    1.5
      0.7    0.6    0.75
      0.35   0.4    0.125
      0.65   0.1    2
    };
  \end{axis}
\end{tikzpicture}
```



# Daten in externen Dateien

```
\addplot [Optionen] table [Spalten-Ausw.] {Dateipfad};
```

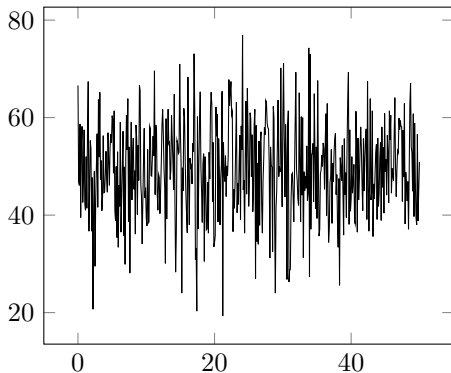
```
\begin{tikzpicture}  
  \begin{axis}  
    \addplot [no markers]  
      table  
        [x=time, y=values]  
        {data.dat};  
  \end{axis}  
\end{tikzpicture}
```



# Daten in externen Dateien

```
\addplot [Optionen] table [Spalten-Ausw.] {Dateipfad};
```

```
\begin{tikzpicture}  
  \begin{axis}  
    \addplot [no markers]  
      table  
        [x=time, y=values]  
        {data.dat};  
  \end{axis}  
\end{tikzpicture}
```



Paket [pgfplotstable](#) erlaubt das Nachbearbeiten vorhandener Tabellen (z. B. Einfügen einer Ausgleichsgerade).

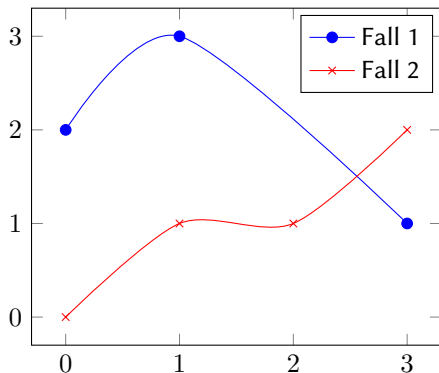
# Beschriftungen

Key	Values	Funktion
title	Text	Titel über dem Diagramm
x/ylabel	bel. Text	Beschriftung der $x$ - bzw. $y$ -Achse
x/ymin/max	Wert	schränkt Achse auf Bereich ein
mark	*, x, +, o, ...	Koordinaten-Marker anpassen
x/ytick	Liste	Koordinatenstriche explizit angeben
minor tick num	Zahl	Anzahl der Zwischenstriche
grid	major, minor	Gitter im Hintergrund einblenden

# Lengenden

`\addlegendentry{Beschreibung}`

```
\begin{tikzpicture}
\begin{axis}
  \addplot[smooth,mark=*,
blue] coordinates {
    (0,2) (1,3) (3,1)
  };
  \addlegendentry{Fall 1}
  \addplot[smooth,color=
red,mark=x] coordinates
  {
    (0,0) (1,1) (2,1)
    (3,2)
  };
  \addlegendentry{Fall 2}
\end{axis}
\end{tikzpicture}
```

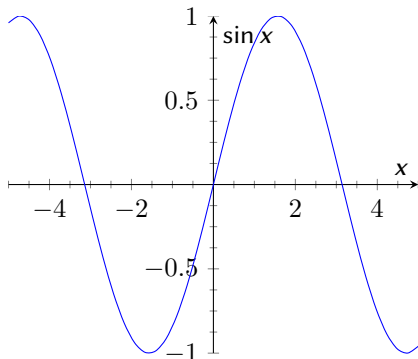




# Platzierung der Achsen

axis y line=*Platzierung*, axis x line=*Platzierung*

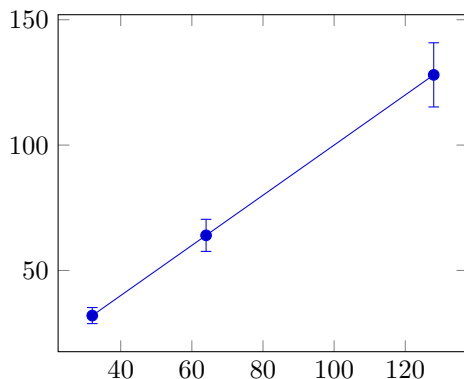
```
\begin{tikzpicture}
\begin{axis}[
minor tick num=3,
axis y line=center,
axis x line=middle,
xlabel=$x$,ylabel=$\sin x$
]
\addplot[smooth,blue,mark
=none,
domain=-5:5,samples=40]
{\sin(deg(x))};
\end{axis}
\end{tikzpicture}
```



# Fehlerbalken

Fehler können mit den Optionen `error bars/⟨Key⟩=⟨Value⟩` gesetzt werden.

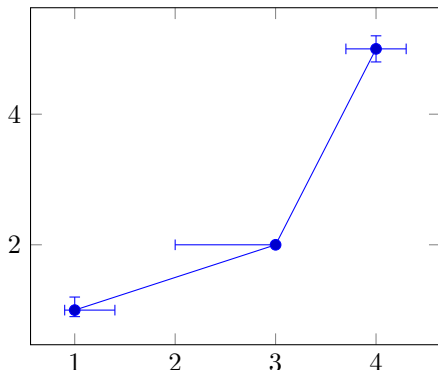
```
\begin{tikzpicture}
\begin{axis}
  \addplot+[
    error bars/y dir=both,
    error bars/y fixed
    relative=.1,
  ] table [x=x,y=y]
  {
    x      y
    32     32
    64     64
    128    128
  };
\end{axis}
\end{tikzpicture}
```



# Fehlerbalken

Individuelle Fehler können mit  $\pm$  (symmetrisch) oder  $+=$  und  $-=$  (asymmetrisch) angegeben werden:

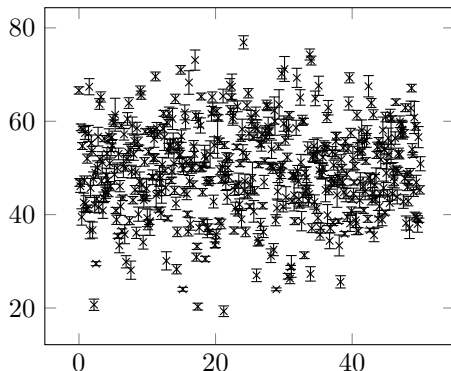
```
\begin{tikzpicture}
\begin{axis}
  \addplot+[
    error bars/.cd,
    x dir=both,
    x explicit,
    y dir=both,
    y explicit,
  ] coordinates {
    (1,1) += (0.4,0.2)
           -= (0.1,0.1)
    (3,2) -= (1,0)
    (4,5) +- (0.3,0.2)
  };
\end{axis}
\end{tikzpicture}
```



# Fehlerbalken

Fehler können auch aus einer Tabelle stammen:

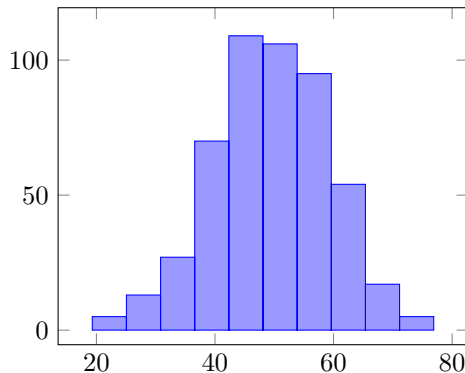
```
\begin{tikzpicture}
  \begin{axis}
    \addplot [only marks,
      mark=x,
      error bars/.cd,
      y dir=both, y
      explicit,]
      table
      [x=time, y=values,
      y error=error]
      {data.dat};
  \end{axis}
\end{tikzpicture}
```



# Histogramme

Histogramme mit Option `hist={\langle Histogram-Optionen \rangle}`

```
\begin{tikzpicture}
  \begin{axis}
    \addplot+[
      fill=blue!40!white,
      mark={},
      hist={
        data=y,
        bins=10
      }
    ] table {data.dat};
  \end{axis}
\end{tikzpicture}
```



Interessante Optionen:

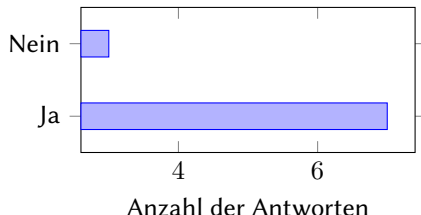
`cummulative` für kummuliertes Histogramm

`density` normiert auf 1

# Balkendiagramme

Option xbar erzeugt Balkendiagramm, ybar erzeugt Säulendiagramm

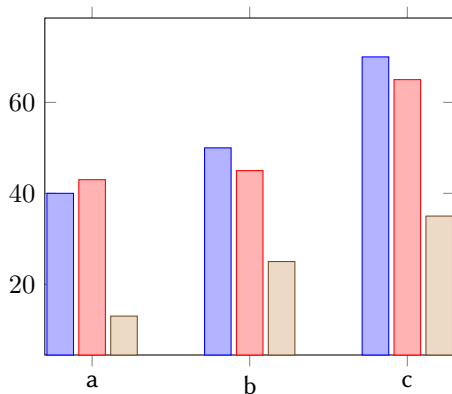
```
\begin{tikzpicture}
\begin{axis}[
  xbar,
  width=6cm, height=3.5cm,
  enlarge y limits=0.5,
  xlabel={Anzahl der
  Antworten},
  symbolic y coords={Ja,
  Nein},
  ytick=data,
]
\addplot coordinates
  {(3,Nein) (7,Ja)};
\end{axis}
\end{tikzpicture}
```



# Balkendiagramme

Option xbar erzeugt Balkendiagramm, ybar erzeugt Säulendiagramm

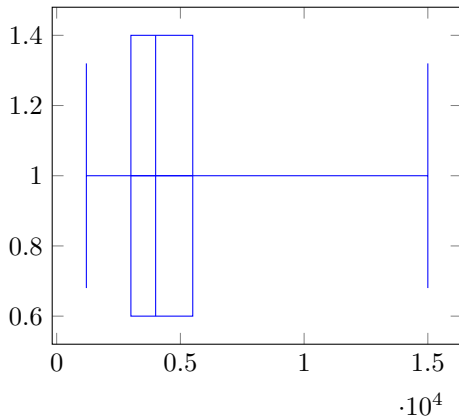
```
\begin{tikzpicture}
\begin{axis}[
  ybar,enlargelimits=0.15,
  symbolic x coords={a,b,c},
  xtick={a,b,c},
]
\addplot coordinates
{(a,40) (b,50) (c,70)};
\addplot coordinates
{(a,43) (b,45) (c,65)};
\addplot coordinates
{(a,13) (b,25) (c,35)};
\end{axis}
\end{tikzpicture}
```



# Boxplots

`\usepgfplotslibrary{statistics}` erlaubt Satz von Boxplots:

```
\begin{tikzpicture}
  \begin{axis}
    \addplot+[
      boxplot prepared={
        median=4000,
        upper quartile
        =5500,
        lower quartile
        =3000,
        upper whisker=1200,
        lower whisker
        =15000,
      } ] coordinates {};
  \end{axis}
\end{tikzpicture}
```



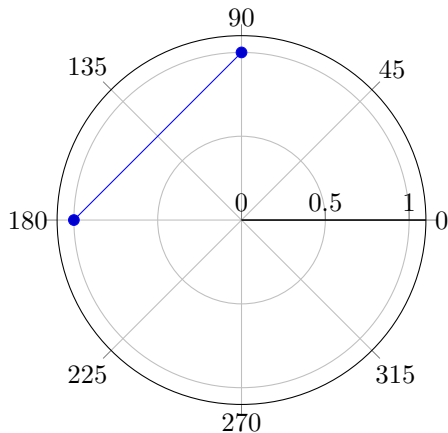


# Polarkoordinaten

Mit `\usepgfplotslibrary{polar}` versteht `pgfplots` Polarkoordinaten.

`polaraxis` geht von Polarkoordinaten aus:

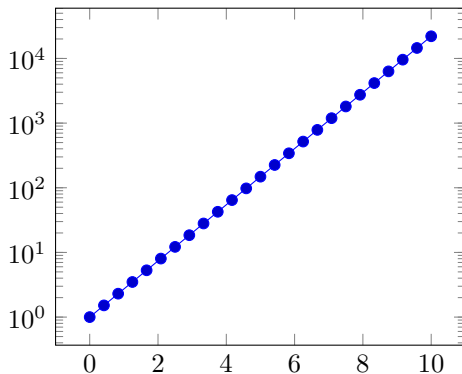
```
\begin{tikzpicture}
  \begin{polaraxis}
    \addplot coordinates
      {(90,1) (180,1)};
  \end{polaraxis}
\end{tikzpicture}
```



# gnuplot in pgfplots

`\addplot gnuplot [Optionen] {gnuplot Befehle};`

```
\begin{tikzpicture}
  \begin{semilogyaxis}
    \addplot gnuplot
      [domain=0:10]
      {exp(x)};
  \end{semilogyaxis}
\end{tikzpicture}
```



# gnuplot in pgfplots

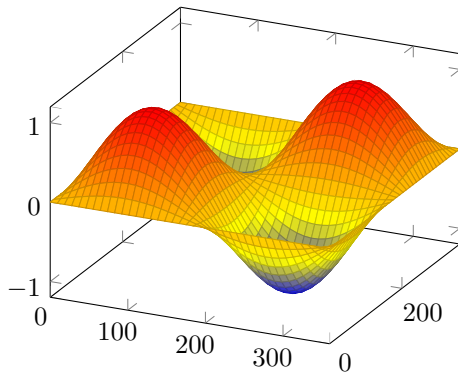
```
\addplot gnuplot [Optionen] {gnuplot Befehle};
```

- **pgfplots** ruft gnuplot auf und speichert das Ergebnis in Hilfsdateien.
- gnuplot wird nur aufgerufen, wenn sich etwas geändert hat.
- gnuplot ist etwas schneller als interne Plots.
- gnuplot stellt mehr mathematische Funktionen zur Verfügung.
- gnuplot nutzt Radiant für Winkel, **pgfplots** nutzt Grad (außer mit Einstellung `trig format=rad`).
- gnuplot und interne Plots haben etwa die selbe Genauigkeit.

# 3D-Plots

`\addplot3 [Optionen] {Eingabedaten};`

```
\begin{tikzpicture}
  \begin{axis}
    \addplot3[
      surf,
      domain=0:360,
      samples=40,
    ]
    {sin(x)*sin(y)};
  \end{axis}
\end{tikzpicture}
```



# Was pgfplots noch so alles kann ...

- extern erzeugte Plots (Bilder) in pgfplot-Koordinatensystem einfügen (auch 3D)
- in Matlab erzeugte Plot importieren
- beliebige Befehle in einer Shell ausführen und das Ergebnis plotten
- Datum oder Uhrzeit als Koordinaten
- automatische Umrechnung von Koordinaten (z. B. polar in kartesisch)
- klickbare Plots (die ein Popup öffnen)
- Plots in einzelne externe Dateien ausgeben
- Flächen zwischen Kurven schraffieren
- Vektorfelder plotten
- ...
- alles das TikZ kann



CHRISTIAN FEUERSÄNGER:

„Manual for Package pgfplots“,  
texdoc pgfplots.