Ein wiederbelebbares Buch zur Statistik

Version: 1.01, Dezember 2004

H. P. Wolf

Statistik und Informatik
Fakultät für Wirtschaftswissenschaften
Universität Bielefeld

Inhalt

1 Vorbemerkung

1.1 Zielsetzung und Idee.

Konzept: Wie lernt ein Fahrschüler das Autofahren? Nach den bei uns gültigen Gesetzen muß er in einer Fahrschule einen Unterricht besuchen, er muß Fahrstunden nehmen — dabei sind besondere unter speziellen Bedingungen zu absolvieren —, und er muß eine theoretische und eine praktische Prüfung bestehen. Jeder, der Autofahren gelernt hat, wird bestätigen, daß damit der Lernprozeß noch lange nicht abgeschlossen ist. Auch nach bestandener Prüfung sollte man weiter bereit sein, fremde und eigene Erfahrungen anzusammeln und einzusetzen.
Wie bekommt man einen Führerschein für statistische Werkzeuge? Leider (?) gibt es so einen Schein bei uns noch nicht, und es gibt demzufolge auch keine allgemeinen gesetzlichen Regelungen, wie ein solcher zu erlangen ist. Dennoch benötigt man für einen geeigneten Umgang mit statistischen Werkzeugen sowohl theoretisches Grundwissen wie auch praktische Erfahrung. Wird nur Theorie vermittelt, wird der Studierende in realen Problemsituationen die Übertragungsleistung nicht bewältigen können und kapitulieren müssen. Lernt er an Beispielen, Kochrezepte einzusetzen, bleiben Zusammenhänge wie auch die Bedeutung von Annahmen unreflektiert, so daß eine adäquate Interpretation der Ergebnisse unterbleiben wird. Ziel der statistischen Ausbildung muß es also sein, statistische Konzepte theoretisch wie auch in ihrer Anwendung zu vermitteln.
Die klassische Vorlesung begleitet von Kleingruppenübungen kann die Einführung in das theoretische Gebäude leisten. Auch können an kleinen Beispielen die Konzepte verdeutlicht werden. Reale Problemfälle, die in der Regel mit größeren Datenmengen verbunden sind, lassen sich aber ohne weiteres nicht diskutieren. Hierfür wird eine Unterstützung durch Rechner mit statistischer Software notwendig, die aber nicht für Ausbildungssituationen geschaffen sind. Mit diesem vorliegenden elektronischen Buch, dem `revbook`, wird ein Versuch gemacht, den beschriebenen Mangel zu mildern. Durch dieses Buch, das in gespeicherter Form auf Rechnern vielerlei Funktionalitäten bietet, erhält der Lernende ein neues Medium, mit dem und in dem er werkeln und erfahren kann.
Um Erfahrungen an einem konkreten Gegenstand zu machen, beginnen die Gedanken dieser Abhandlung in einem realen Problembereich. In diesem werden Fragen aufgeworfen und unter Einsatz von statistischen Konzepten und Werkzeugen beantwortet.

Praxis: Als interessantes Problemfeld dient die Belastung eines WWW-Servers. Dieses ist ein (virtueller) Rechner, der das Internet neben einigen anderen Diensten auf Anforderung mit Dateien versorgt. Durch die enormen Wachstumsprozesse des Internets ergeben sich verschiedene Fragen, die sich um die Belastung des Servers drehen: Wie mißt man Belastung? Wann ist die Belastung am größten? Worauf müssen wir uns in der Zukunft einstellen?

Konzept: Zur Motivation werden in jedem Kapitel einige Fragen aufgelistet und Daten bereitgestellt. Mit Hilfe der in diesem Buch verfügbaren Werkzeuge entdeckt der aktive Leser dann eine Reihe von Antworten. Dabei werden wieder neue Fragen aufgeworfen, die tiefer in die Statistik hineinführen und zu den folgenden Kapiteln überleiten. Die Fragen sind von ihrer Natur her auf sehr viele andere Bereiche übertragbar, so daß sich die Relevanz der Ausführungen nicht nur auf das Feld des Internet beschränkt.
Dieses Buch dient damit zwei unterschiedlichen Zielen:

1. Erfahrungsbildung durch Einsatz von Werkzeugen der Statistik an Beispieldatensätzen,
2. Beantwortung von Fragen eines realen Problembereiches unter Anwendung statistischer Werkzeuge.

Das Buch soll als zusätzliches Angebot eine Einführungsvorlesung begleiten. Deshalb werden nicht alle Grundkonzepte der Statistik hier eingeführt. So erfährt zum Beispiel der Wahrscheinlichkeitsbegriff keine nähere Problematisierung, sondern es wird davon ausgegangen, daß solche theoretischen Konzepte an anderer Stelle vertieft erklärt werden. Zur Unterstützung enthält der Text aber Literaturhinweise.

Ebenso findet keine Einführung in R statt. R ist die Software, die dieses elektronische Buch als Fundament hat. Für einen Anfänger dürfte die Sprache R sehr viele kleine technische Probleme aufwerfen, die ihn von der eigentlichen Auseinandersetzung ablenken können. Deshalb ist von der Beherrschung der Sprache vorsetzlich auch nicht ausgegangen worden. Alle R-Anweisungsfolgen sind für ihre Benutzung zu Modulen, von denen der Anfänger nur die satzähnlichen Namen erfährt, zusammengefaßt. Die Definitionen der Module kann der Kenner von R im Anhang finden und kann natürlich von seinem Wissen profitieren. Der Neugierige kann stöbern und über die Definitionen der Module viele Dinge über die doch sehr eingängige Sprache R herausfinden.

1.2 Einige Bemerkungen zur Technik

Konzept: In diesem Abschnitt werden einige Hinweise zur eingesetzten Technik gegeben.

1.2.1 Lesen des REVBOOK-Buches

Praxis: Eine ausgedruckte Version dieses Buches kann wie ein Buch gelesen werden. Gedanken, Verfahrensaufrufe, Ergebnisse und Interpretationen sind so aufgeschrieben, daß sie einen geschlossenen Gedankengang ergeben (sollen). Wenn das Buch im PDF- oder im PS-Format vorliegt, kann es mit einem entsprechendem Viewer betrachtet werden. Unter dem Betriebssystem UNIX könnte der Aufruf dann so aussehen:

```
$ acroread revbook.pdf  oder  $ ghostview revbook.ps
```

Jedoch ist das Lesen eines Buches an einem Bildschirm eine Zumutung und auch hier nicht angestrebt. Hauptanliegen der Rechnerversion besteht darin, interaktiv mit dem Buch arbeiten zu können. Dazu läßt sich das Buch als Abfolge von Textstücken (text chunks) und Anweisungsblöcken (code chunks) auffassen. Die Texte sind für den menschlichen Leser bestimmt und beinhalten, wie gesagt, Gedanken, Ergebnisse und Interpretationen. Die Anweisungsblöcke, die in den verschiedenen Kapiteln auftauchen, bestehen im wesentlichen aus Modulaufrufen, deren Definitionen über die eingetragenen Referenznummern im Anhang zu finden sind.

1.2.2 Voraussetzungen für die interaktive Arbeit

Voraussetzung für das oben skizzierte Tun sind die Statistik-Software R, die `relax`-Bibliothek sowie die Quelldatei zu dem Buch `revbook.rev`.

1.2.3 Start der interaktiven Arbeit mit dem Buch

Die genaue Form des Starts kann von Version zu Version ein wenig unterschiedlich sein. Folgende Schritte sind in der Regel nötig.

1. Rechner anstellen und einloggen.
2. R starten durch Click auf die R-Ikone.

   R meldet sich dann mit dem Eingabeaufforderungszeichen (Prompt) >.

3. `relax` öffnen `relax()`-Funktion aufrufen:

   ```
   library(relax)
   relax()
   ```

   Hierfür muß das Paket `relax` unter R installiert sein. Es öffnet sich ein neues Fenster, in das das Revbook geladen werden muss.

4. Laden des Revbook geschieht über den Menüpunkt `OpenReport` des `File`-Menüs. Dazu muss auf der Anlage die Datei `revbook.rev` zu finden sein. Außerdem ist eine Datei mit speziellen Objekten (`robj.R`) erforderlich, die im Arbeitsverzeichnis, im Paket `relax` oder sogar über Netz gesucht und geladen wird. Sollen die Code-Chunks dieses Dokuments per Hand eingegeben und ohne `relax` zur Ausführung gebracht werden, sind diese Objekte ebenfalls für viele Code-Chunks erforderlich. Hierfür können diese Objekte – sofern lokal verfügbar – geladen werden über durch die Anweisung:

   ```
   source("robj.R")
   ```

   oder über das Internet beschafft werden:

   ```
   source("http://www.wiwi.uni-bielefeld.de/~wolf/software/revbook/robj.R")
   ```

   Nach einer solchen Anweisung befinden sich die Objekte in der Arbeitsumgebung. Damit eigene Objekte gleichen Namens nicht zerstört werden, darf man beim Verlassen von R die Umgebung nicht oder nicht unter dem gewohnten Namen speichern.

1.2.4 Die interaktive Arbeit

Das elektronische Buch enthält eine ganze Reihe von Anweisungen zur Aktivierung statistischer Verfahren. Diese können durch Positionierung der Maus selektiv ausgewählt und dann urch Druck des Knopfes `EvalRCode` gestartet werden. Mit ein wenig Geschick (und etwas R-Wissen) lassen sich eigene Wünschen umsetzen. Ohne weiteres Wissen können Operationen über die Auswahlmenüs unter `Revbook` gestartet werden.

1.2.5 Beendigung der interaktiven Arbeit

`relax()` wird verlassen über

```
--> File
   --> Exit
```

R wird durch Aufruf der Funktion `q()` beendet.

1.3 Gliederungsprinzip

Konzept: Der Inhalt ist primär nach statistischen Elementen gegliedert, da dieses Suchprozesse beim zweiten Lesen und auch sprunghafte Arbeitsstile besser unterstützt. Im kleinen tauchen die drei Hinweiswörter **Konzept**, **Praxis** und **Technik** auf. Hierdurch soll dem Leser angedeutet werden, daß die folgenden Gedanken entweder mehr allgemeiner, abstrakter, konzeptionell oder aber speziell, am Beispiel orientiert sind oder zum dritten sich eher auf rein technische Belange beziehen.

1.4 Literatur

Autorenkollektiv, ohne Jahr: Grundausbildung in Statistik für Wirtschaftswissenschaftler,
FU Berlin, Universität Bielefeld, TU Hannover, Ruck-Zuck-Druck, Bielefeld.
BAMBERG, G., BAUR, F., 1980: Statistik,
Oldenbourg Verlag, München/Wien.
CHAMBERS, J.M., CLEVELAND, W.S., KLEINER, B., TUKEY, P.A., 1983:
Graphical Methods for Data Analysis, Duxbury Press, Boston.
HOAGLIN, D.C., MOSTELLER, F., TUKEY, J.W., (EDS.), 1982:
Understanding Robust & Exploratory Data Analysis.
John Wiley & Sons, New York.
SCHLITTGEN, R., 1990: Einführung in die Statistik,
Oldenbourg Verlag, München/Wien.
VENABLES, W.N., RIPLEY, B.D., 1994 u.a.: Modern Applied Statistics with S-Plus,
Springer-Verlag, New York/Berlin.
BECKER, R.A., CHAMBERS, J.M., WILKS, A.R., 1980: The new S Language,
Wadsworth & Brooks/Cole, Pacific Grove.
WOLF, P., 1995: Eine Reportschmiede für den Datenanalytiker.
Diskussionspapier Nr. 301 der Fakultät für Wirtschaftswissenschaften
der Universität Bielefeld, Bielefeld.

1.5 Noch nicht enthalten

Noch mehr Experimente, Behandlung von kategoriellen Daten, Kapitel Schätzen, Kapitel Testen, Kapitel Regression, Diskussion der Fehlbenutzung statistischer Werkzeuge, weitere spezielle Lernwerkzeuge.

2 Ein Problemfeld

2.1 Fragen

Praxis: Mit dem vielgerühmten neuen Instrument *Internet* handeln sich deren Bediener auch Probleme ein. Neben der Anschaffung von Hardware und Software müssen verschiedene Pflegedienste erledigt werden. Dazu gehören das Beschließen und Umsetzen von organisatorischen Maßnahmen wie auch die Planung von Kapazitäten. Notwendig ist hierfür eine gute Abschätzung der zukünftigen Belastung, die eine Kenntnis der aktuellen voraussetzt. Dadurch stellen sich zum Beispiel folgende Belastungsfragen ein:

- Wie groß ist die Belastung?
- Wie stark schwanken Belastungen?
- Wie läßt sich die Belastung darstellen?
- Wie ändert sich die Belastung im Tagesablauf oder auch im Laufe einer Woche?
- Wie lassen sich Gesetzmäßigkeiten modellieren?

Solche Fragen sind schnell formuliert. Ihre Beantwortung erfordert jedoch statistischen Sachverstand, einiges aus der Wahrscheinlichkeitstheorie und zunächst die Definition dessen, was unter Belastung verstanden werden soll.

2.2 Daten

Praxis: Wir wollen uns auf einen realen WWW-Server konzentrieren und die Belastung dieses Servers anhand der Zugriffe, die von außen kommen, studieren. Eine hohe Anzahl von Zugriffen (pro Zeiteinheit) wie auch große Übertragungsmengen sind als hohe Belastung zu deuten. Eine Erhebung hat für den 03.02.1997 und für den 17.02.1997 die Zugriffszeitpunkte (in Sekunden seit Monatsbeginn) mit den jeweiligen Übertragungsmengen (in Bytes) hervorgebracht. Damit stehen für die erste Analyse ausreichend Daten zur Verfügung.

2.3 Relevanz

Konzept: Zwei Bemerkungen seien zur Relevanz der Diskussion angefügt:

1. Führt die Analyse des Datenmaterials zu brauchbaren Erkenntnissen, läßt sich die Methodik auf ähnliche Datensätze anwenden. Damit wird zum Beispiel die Analyse der Server-Daten eines anderen Tages auch gelingen.

2. Kapazitätsfragen sind an vielen Stellen in der Welt relevant. Man denke nur an unsere zunehmend verstopften Autobahnen. Aber auch für die Planung betrieblicher Ressourcen spielen zufällig eintretende Ereignisse (wie eintreffende Kunden, Aufträge, Produktionsausfälle) und Belastungsfragen (zum Beispiel bei der Planung eines Lagers oder von Bearbeitungs- und Abfertigungsstellen) eine erhebliche Rolle. Insofern lassen sich viele der hier gewonnenen methodischen Erkenntnisse auf ganz andere Bereiche übertragen. Diese Übertragung erfordert einen Abstraktionsprozeß, der dem zukünftigen Problemlöser aber nicht abgenommen werden kann.

Technik: Es sei darauf hingewiesen, daß auf den Variablen `zeitpunkte.03.02.1997` und `mengen.03.02.1997` die Zugriffszeitpunkte sowie die Datenmengen vom 03.02.1997 abgelegt sind, `dzeitpunkte.03.02.1997` hält die Zeitdifferenzen zwischen den Zeitpunkten. Die Zeitpunkte vom 17.02.1997 sind der Variablen `zeitpunkte.17.02.1997`, die zugehörigen Mengen `mengen.17.02.1997` und die Zwischenzeiten `dzeitpunkte.17.02.1997` zu entnehmen. Dieser Hinweis ist für R-Kenner gedacht. Mit diesem `revbook` kann aber auch über die Auswahlmenus auf diese Daten zugegriffen werden. Hierfür ist es nicht erforderlich, die Namen der Variablen zu kennen.

3 Maßzahlen zur Beschreibung von Lage und Variabilität

Technik: Am Ende dieses Absatzes befindet sich die erste Stelle, die vom Rechner gelesen werden kann. Solche Stellen sind, wie oben beschrieben, mit einer laufenden Nummer versehen. So findet man neben dem ersten Block die Zahl 1. Die Zeichen ⟨* 1⟩+≡ zeigen den Beginn des Anweisungsblockes an. Der Anweisungsblock selbst besteht aus einer Anweisung zum Ausdruck einer Überschrift, die für die Orientierung hilfreich sein soll. Es finden also noch keine weiteren Berechnungen statt.

1 ⟨* 1⟩≡
```
cat("Revbook:1.a:start.mz:Kapitel Masszahlen\n")
```

Zum Literaturstudium sei hingewiesen auf: [Autorenkollektiv: Abschnitte über Lokalisationsmaße und Maßzahlen der Variabilität], [Bamberg, Baur: Abschnitte über Lageparameter und Streuungsparameter], [Schlittgen: Abschnitte über Lageparameter und Streuungsparameter].

Konzept: Die ersten beiden Fragen des letzten Kapitels lauteten: *Wie groß ist die Belastung?* und *Wie stark schwanken die Belastungen?* Zur Beantwortung müssen die Begriffe *groß* und *schwanken* mit Bedeutung gefüllt und präzisiert werden. Hierzu bietet die Statistik die vagen Begriffe *Lage* und *Variabilität* sowie verschiedene Maßzahlen zu deren Formalisierung an. In diesem Abschnitt werden solche Maßzahlen zur Lage und zur Variabilität behandelt.

Konzept: Durch den Übergang zur Maßzahl versucht der Statistiker, einen Datensatz bezüglich eines speziellen Aspektes maximal zu verdichten (Datenreduktion). Hierdurch erhofft man sich einen klaren Blick auf die gesuchte Information. Jedoch ist die Definition von Maßzahlen nicht unproblematisch. Dieser Abschnitt zeigt verschiedene Festlegungen und ihre Probleme auf und soll ein vertieftes Methoden-Verständnis fördern.

Die treibenden Fragen in allgemeiner Formulierung zu diesem Kapitel sind:

- Wo liegt das Zentrum der Beobachtungen?
- Wie stark streuen die Beobachtungen?
- Sind die Beobachtungen symmetrisch verteilt?
- Wie sehen die Verteilungen in den Randbereichen aus?
- Gibt es Ausreißer?

Maßzahlen können Beiträge zu den ersten drei Fragen beisteuern. Für die weiteren sei auf den Abschnitt über graphische Werkzeuge verwiesen. Zuerst werden das Mittel, der Median und Verwandte, die als Lageparameter Verwendung finden, eingesetzt. Dann werden Maße zur Charakterisierung der Streuung betrachtet. Abschließend werden Box-Cox-Transformationen diskutiert, die in vielen Fällen das Datenmaterial symmetrischer machen können.
Für die zulässige Anwendung eines Werkzeuges müssen im allgemeinen bestimmte Bedingungen erfüllt sein. Dies gilt auch für Werkzeuge der Statistik. So kann man nicht jedes Datenmaterial mit jedem Werkzeug bearbeiten und dann die resultierenden Rechenergebnisse interpretieren. Statistische Verfahren, in denen mit Daten *gerechnet* wird, erwarten, daß die Rechenoperationen für die Daten geeignet sind. Für die Beobachtungen *männlich* und *weiblich* ist die Addition keine geeignete Operation, da für eine Addition ein nominales Skalenniveau nicht ausreicht. Auch geordnete Größenklassen erfüllen diese Forderung nicht (Beispiel hierfür: *winzig*, *klein*, *normal*, *groß*, *riesig*). Erst eine metrisch skalierte Variable erlaubt Summation und Durchschnittsbildung; hier lassen sich auch Differenzen der Beobachtungen sinnvoll interpretieren. Für die im folgenden eingesetzten Operationen wird implizit die Zulässigkeit unterstellt. Fragen der Metrik werden nicht näher diskutiert. Hierzu sei auf die Literatur verwiesen.

Praxis: Woran läßt sich die Server-Belastung festmachen? Ein Aspekt von Belastung kann an den Zeiten zwischen den Zugriffsereignissen abgelesen werden. Sind die Zeiten zwischen zwei Zugriffsereignissen klein, dann ist die Belastung groß, sind sie groß, dann ist die Belastung klein. Wir wollen in diesem Kapitel die Zwischenzeiten vom 03.02.97 näher unter die Lupe nehmen. Für diese Daten lassen sich die fünf Fragen übersetzen in:

- Welches Belastungsniveau herrscht am 03.02.97? oder Welche Wartezeiten zwischen Zugriffen können als typisch angesehen werden?
- Wie stark schwanken die Zeiten zwischen einzelnen Zugriffen?
- Ist die Verteilung der Zugriffszeiten symmetrisch oder gibt es zum Beispiel wenige lange Wartezeiten und viele kurze?
- Welche Verteilung haben die sehr langen und die sehr kurzen Wartezeiten?
- Gibt es besonders kurze oder besonders lange Zwischenzeiten?

Technik. Zunächst legen wir die Zwischenzeiten vom 03.02.97, unsere Urliste, auf der Variablen `x` ab. Hierdurch können alle möglichen Operationen mit `x` durchgeführt werden, ohne daß der Originaldatensatz verloren gehen kann. Da alle Verfahrensbeschreibungen dieses Kapitels auf `x` zugreifen, können so durch Neubelegung von `x` alle eingesetzten Verfahren auf den neuen Datensatz angewendet werden.
Zusätzlich drucken wir die Daten, da es nicht zu viele sind, einmal aus.

2 ⟨*lege Zwischenzeiten vom 03.02.97 auf* `x` *ab und drucke* `x` *aus* 2⟩≡

```
x<-dzeitpunkte.03.02.97
x
```

Praxis: Wir erhalten:

```
  [1]  1077  4862 24449   554   164  1667    67  3670   639   350    83   524
 [13]  2236   403   240     6   131     2     1    44   351   615    14    12
 [25]     8   106   312  1183   697  1021   771    46   153    29   672   516
 [37]   116   285    73    15   120    14   490   788   600   117  2772   197
 [49]     6    11     2     1    19    21   503  1374   133   337   320  1754
 [61]   119    44   433   718   158    46   712    58   139    40   165   344
 [73]  1378   588   455   870  2409  1021   547    10     1     8    40    67
 [85]   166  1145   778   260  1515   540  1281  1965   207  1913  2330   174
 [97]   573  1702    49   107   290   362  1381
```

3.1 Maße zur Charakterisierung der Lage

Konzept: Die Frage nach der Größe der Belastung ist eine Niveaufrage. Denkt man sich eine Achse, auf der sich Belastungen abtragen lassen, so wird nach der Stelle der Achse gefragt, die die allgemeine Belastung charakterisiert. Es wird also die Lage gesucht: Wo liegt die Stelle der typischen Belastung auf der Belastungsachse?

Praxis: Die Frage kann zum Beispiel zur mittleren Zwischenzeit als Indikator führen. Bei der Übertragung auf die Zwischenzeiten darf natürlich der inverse Zusammenhang zwischen Belastung und Zwischenzeiten nicht vergessen werden.

Konzept: Als erste Variante einer charakterisierenden Zahl wird fast immer der Mittelwert vorgeschlagen. Also berechnen wir zunächst das arithmetische Mittel $\sum_{i=1}^{n} x_i/n$ der n Beobachtungen $(x_1, \ldots, x_n)$.

Algorithmus: `mean(x)` — *arithmetisches Mittel von* `x` $= (x_1, \ldots, x_n)$

```
mean(x) == sum(x)/n
```

Technik: An sich dürfte hier, da allgemein bekannt, eine formale Definition des Mittelwertes nicht nötig sein. Jedoch zeigt der Kasten, daß R auch als Notationsinstrument eingesetzt werden kann. `sum(x)` ist eine R-Anweisung, die die Summe der auf `x` abgelegten Zahlen berechnet. `/` steht für Division, `+`, `-` und `*` für Addition, Substraktion und Multiplikation. Außerdem sei noch bemerkt, daß R die Funktion `mean()` kennt, die das arithmetische Mittel der ihr übergebenen Variablen berechnet. Hier ist das Modul zur Mittelwertberechnung von `x`:

3 ⟨*berechne Mittelwert von* `x` 3⟩≡

```
mean(x)
```

Praxis: Wir erhalten:

```
Mittelwert
[1] 823.8
```

Da die Daten als Einheit Sekunden haben, beträgt die durchschnittliche Wartezeit auf den nächsten Zugriff eine knappe Viertelstunde. Das scheint nicht sehr viel zu sein.

Konzept: Alternativ kann man zur Charakterisierung der Lage die Stelle berechnen, die den Datensatz bezüglich der Anzahlen in zwei gleich große Teile teilt. Diese Stelle heißt (bekanntlich) Median.

Algorithmus: `median(x)` — *Median von* `x` $= (x_1, \ldots, x_n)$

$$\texttt{median(x)} \;==\; \begin{cases} x_{(\frac{n+1}{2})} & \textit{für} \quad n \textit{ ungerade} \\ 0.5x_{(n/2)} + 0.5x_{(n/2+1)} & \textit{für} \quad n \textit{ gerade} \end{cases}$$

Hierbei ist $x_{(i)}$ *die i-t kleinste Beobachtung.*

Technik: Der R-Aufruf `median(x)` berechnet den Median von `x`. Um auch für Nicht-R-Fans verständlich zu sein und nicht zu sehr abzulenken, wird im Anweisungsblock zur Berechnung des Medians ein Modulaufruf verwendet. Aus diesem Grund tauchen in den Kapiteln des `revbook` fast nur Modulaufrufe auf. Die Anweisungen der Module sind über die kleine Referenznummer hinter dem Modulnamen im Anhang zu finden.

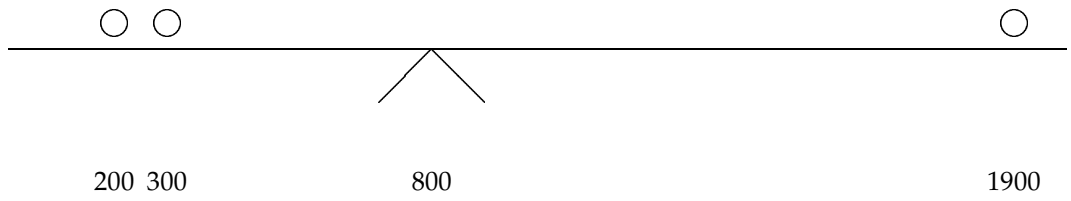
4 ⟨*berechne Median von* `x` 4⟩≡
```
median(x)
```

Praxis: Wir erhalten:

```
Median
[1] 312
```

Fast jedes zweite Mal beträgt also der Abstand zwischen zwei Zugriffsversuchen weniger als 6 Minuten. Damit erscheint die Belastung viel höher zu sein, als das Mittel vermuten ließ.

Konzept: Irrt eine der beiden Maßzahlen? Wann liefern Mittel und Median gleiche Werte? Für die drei Werte 200, 300, 400 sind Mittel und Median gleich (300). Für 200, 300, 1900 erhalten wir wieder einen Median von 300, jedoch ein Mittel von 800. 800 ist übrigens gerade die Stelle des Schwerpunkts eines physikalischen Systems, bei dem sich an den Stellen 200, 300 und 1900 gleich große Massen befinden.



Wie man sieht, haben Extremwerte, die eventuell auf Fehlern beruhen, einen erheblichen Einfluß auf das Mittel, der Median bleibt von ihnen unberührt. Wir wollen die Extremwerte aufspüren.

5 ⟨*berechne Extrema von* `x` 5⟩≡
```
cat("Minimum:",min(x),"\nMaximum:",max(x),"\n")
```

Praxis:

```
Minimum:
[1] 1
Maximum:
[1] 24449
```

In der Tat weicht der größte Wert erheblich von dem Bereich (bis 1000) ab, in dem sich die bisher berechneten Lageparameter befinden. Damit wird klar, daß man das Mittel in dem vorliegenden Fall nicht ohne weiteres als Argument für Belastungsfragen heranziehen darf.

Konzept: Auch die Berechnung der Mitte der Extremwerte — genannt Midrange — als Lagemaß weist denselben Defekt auf.

Algorithmus: — Midrange von $\texttt{x} = (x_1, \ldots, x_n)$

```
0.5 * (min(x) + max(x))
```

Technik: Die R-Funktionen `min()` und `max()` berechnen Minimum und Maximum eines Datensatzes.

Praxis: Unser Modul mit dem sprechenden Namen folgt:

6 ⟨*berechne Midrange von* x 6⟩≡
```
(min(x)+max(x))/2
```

```
Midrange
[1] 12225
```

Konzept: Nun wird noch eine Alternative zum arithmetischen Mittel und zu $x_{0.5}$ präsentiert, die die Auswirkung der Ausreißer entschärft:

Algorithmus: — α-getrimmtes Mittel von $\texttt{x} = (x_1, \ldots, x_n)$

Das α-getrimmte Mittel ist das arithmetische Mittel der um die $\alpha \times 100\%$ *kleinsten und größten Werte reduzierten Daten. (Man kann natürlich nur weniger als 50 % an jedem Ende entfernen!)*

Technik: Das Modul zur Berechnung des getrimmten Mittels fragt nach dem Prozentsatz der an jeder Seite zu entfernenden Werte. Die Frage ist durch Eingabe einer Prozentzahl zu beantworten. (Für absolute Rechnerneulinge: Für den Wunsch 10 % erst die beiden Tasten 1 und 0 einmal drücken, dann die Eingabetaste.)

7 ⟨*berechne getrimmtes Mittel von* x 7⟩≡
```
cat("getrimmtes Mittel\n")
cat("Wieviel Prozent sollen an jeder Seite enfernt werden?\n")
cat("  Eingabe:\n")
result<-mean(x,trim=min(.5,.01*c(scan(,0,n=1),0)[1]))
cat("  ergibt:\n")
result
```

Praxis: Hier das Musterbeispiel:

```
getrimmtes Mittel
Wieviel Prozent sollen an jeder Seite enfernt werden?
  Eingabe:
[1] 10
    ergibt:
[1] 439.11
```

Das getrimmte Mittel liegt offensichtlich dem Median viel näher als das nicht getrimmte. Haben Sie hierfür eine Erklärung?

Konzept: Es gibt natürlich weitere Vorschläge, die Lage durch eine Statistik auszudrücken. Stellvertretend sei deshalb noch das Trimean vorgestellt. Dieses berechnet sich als gewogenes Mittel von Median und den Quartilen. Dafür benötigen wir die Berechnungsvorschrift von Quartilen, die wieder auf die der Quantile zurückgeführt wird. Das Trimean selbst ist relativ unüblich, führt uns aber zu den wichtigen Konzepten *Quartile* und *Quantile*. (Siehe auch [Hoaglin et al.].)

Algorithmus: `quantile(x,p)` *— das p-Quantil von* $\mathtt{x} = (x_1, \ldots, x_n)$

Das p-Quantil ist grob der Beobachtungswert, der größer ist als $p \times 100$ *% der Beobachtungen und kleiner als die verbleibenden* $(100 - p \times 100)$ *% Beobachtungen.*

(Die Definition eines empirischen Quantils ist nicht genormt, so daß kleine Unterschiede auftreten können. Diese spielen für größere Stichprobenumfänge kaum eine Rolle. Werden jedoch von verschiedenen Software-Produkte unterschiedliche Werte ermittelt, kann das auf unterschiedliche Definitionen zurückzuführen sein. R sieht die Welt so: Das p-Quantil ist gegeben durch $g^{-1}(p)$, wobei g^{-1} die Umkehrfunktion von $g(x)$ ist; $g(x)$ ist die Funktion, die sich aus dem Polygonzug der Punkte $(x_{(i)}, (i-1)/(n-1))$ ergibt. $x_{(i)}$ ist dabei der i-te Wert der Rangwertreihe (sortierte Folge der Werte).)

Technik: `quantile(x,p)` berechnet das p-Quantil von `x`. Für `p` ist dabei der gewünschte Anteil einzusetzen. Quartile sind besondere Quantile. Sie teilen den Datensatz in vier ungefähr gleich große Teile.

Konzept:

Algorithmus: Quartile von $\mathtt{x} = (x_1, \ldots, x_n)$

`quantile(x,0.25)` *und* `quantile(x,0.75)` *werden auch als das untere (oder als erstes) Quartil bzw. das obere (oder als drittes) Quartil bezeichnet.*
Das untere Quartil entspricht grob dem Median derjenigen Beobachtungen, die unterhalb des Medians aller Beobachtungen liegen, das obere Quartil grob dem Median aller Beobachtungen oberhalb des Gesamtmedians.

Auch diese Begriffe sind nicht genormt.

Technik: `quantile(x,0.25)` berechnet also das untere Quartil von `x`, `quantile(x,0.75)` das obere.

Konzept: Nun läßt sich das Trimean, das Mittel aus dem Median und dem Mittel aus erstem und drittem Quartil, festlegen.

Algorithmus: — Trimean von $\mathtt{x} = (x_1, \ldots, x_n)$

```
sum(c(0.25,0.5,0.25)*quantile(x,c(0.25,0.5,0.75)))

==mean(c(median(x),mean(quantile(x,c(0.25,0.75)))))
```

Technik: Die Notation deutet darauf hin, daß innerhalb von R Funktionsaufrufe ineinander verschachtelt werden könne. Die Funktion `c()` verkettet mehrere Dinge zu einer großen Einheit.

8 ⟨*berechne Trimean von* x 8⟩≡
```
sum(c(0.25,0.5,0.25)*quantile(x,c(0.25,0.5,0.75)))
```

Praxis: Der Datensatz `x` besitzt ein Trimean von

```
Trimean
[1] 357.75
```

Der Leser kann sich nun überlegen, welche Vorteile das Trimean aufweist.

Konzept: Einzelne Lagemaße vermögen nur etwas über die Lage zu mitzuteilen. Mit Hilfe von mehreren Maßen lassen sich schon weitere Aspekte aufspüren. Neben der Lage wird in der Statistik als zweite Frage die nach der Variabilität gestellt. Mit dieser beschäftigt sich der nächste Abschnitt.

3.2 Maße zur Charakterisierung der Variabilität

Praxis: Wenn im Mittel alle 15 Minuten ein Zugriff auf den Server durchgeführt wird, ist damit noch nicht klar, ob Zwischenzeiten typischerweise im Intervall von [14,16], von [10,20] oder gar von [0,100] anzutreffen sind. Sind die beobachteten Zwischenzeiten alle fast gleich, oder wenn sie eher unterschiedlich sind, wie unterschiedlich sind sie? Diese Fragen haben etwas mit *Variabilität* zu tun. Der Variabilität der Zwischenzeiten `dzeiten.03.02.97` wollen wir nun nachgehen.

Konzept: Wenn die Lage (z.B. durch das Mittel) bekannt ist, ist damit noch nicht klar, in welchem Bereich die einzelnen Daten streuen. Die Stärke, in der die Beobachtungen streuen, wird durch Maße der Variabilität beschrieben. Im Prinzip ist im letzten Abschnitt schon ein Variabilitäts-Maß aufgeführt worden: die Spannweite, die sich als Differenz zwischen Maximum und Minimum ergibt.

Algorithmus: — Spannweite von `x` $= (x_1, \ldots, x_n)$

```
max(x) - min(x)
```

9 ⟨*berechne Spannweite von* `x` 9⟩≡

```
max(x)-min(x)
```

Praxis:

```
Spannweite
[1] 24448
```

24448 Sekunden sind ungefähr 400 Minuten oder 6 Stunden und 40 Minuten. Die Wartezeiten auf den nächsten Serverzugriff schwanken also erheblich. Wie schon oben festgestellt wurde, kann es sein, daß man nur 1 Sekunde auf den nächsten Zugriff warten muß. Jedoch gab es auch eine Phase von über 6 Stunden Länge, in der keine Tätigkeit stattfand. Könnten solche Schwankungen zufällig entstanden sein oder gibt Erklärungen? Überlegen Sie einmal, inwieweit sich Lebensgewohnheiten auf die Serverbelastung auswirken könnten! (Übrigens können Serverbelastungen auch auf die zu zahlenden Telefongebühren zurückwirken!)

Konzept: Gegen die Spannweite spricht, daß sie von zwei extremen Werten festgelegt werden kann, die für den Datensatz keine große Repräsentativität zu haben brauchen. Deshalb findet man in der Regel als ersten Vorschlag die Stichprobenvarianz sowie deren Wurzel, die Standardabweichung. Diese Maßzahlen gründen sich auf folgenden Gedanken. Wird von jedem Datum das berechnete Mittel abgezogen, erhält man Werte, aus denen man Streuungseigenschaften ermitteln kann. Durch Summation ihrer Quadrate erhält man einen Indikator, der noch von der Anzahl der Daten stark beeinflußt wird. (Die Summation der Abweichung selbst ergibt immer 0 und ist deshalb ungeeignet.) Sind die Daten durch $x_1, \ldots, x_n$ gegeben, gelangt man zu der bekannten Formel der Stichprobenvarianz: $s^2 = \sum_{i=1}^{n}(x_i - \bar{x})^2/(n-1)$.

Algorithmus: `var(x)` — *Stichprobenvarianz von* `x` $= (x_1, \ldots, x_n)$

```
var(x) == sum((x-mean(x))^2)/(n-1)
```

Technik: Die Funktion `var()` berechnet die Stichprobenvarianz.

Konzept: Für die Standardabweichung folgt:

Algorithmus: — *Standardabweichung von* `x` $= (x_1, \ldots, x_n)$

```
var(x)^0.5
```

Technik: Die Operation `x^y` bedeutet: x^y (Exponentiation) und `x^0.5` ist die Wurzel aus `x`.

Konzept: Die Standardabweichung besitzt den Vorteil, daß ihre Dimension mit der der Daten übereinstimmt, so daß sie unmittelbar interpretiert werden kann. Vom Informationsgehalt her ist die Stichprobenvarianz gleichwertig. (Übrigens sei darauf hingewiesen, daß in obiger Formel nicht durch die Anzahl der Beobachtungen, sondern nur durch $(n-1)$ dividiert wird. Dieses dürfte beim Lernenden der Statistik Verwunderung erzeugen. Doch besitzt die Festlegung angenehme theoretische Eigenschaften und ist gegenüber der mittleren quadratischen Abweichung — $d^2 = s^2(n-1)/n = \sum_{i=1}^{n}(x_i - \bar{x})^2/n$ — mehr verbreitet.

Praxis: Was ergibt sich für die Beispieldaten?

10 ⟨*berechne Stichprobenvarianz von* `x` 10⟩≡
```
var(x)
```

```
Stichprobenvarianz
[1] 6187227
```

Diese Zahl erzeugt wirklich Interpretationsprobleme.

11 ⟨*berechne Standardabweichung von* `x` 11⟩≡
```
var(x)^0.5
```

```
Stichprobenstandardabweichung
[1] 2487.4
```

Hiermit läßt sich vielleicht vermuten, daß sehr viele Beobachtungen in einem Bereich von [`mean(x)-2500`, `mean(x)+2500`] anzutreffen sind. Überlegen Sie, welche Datenkonstellationen zu einer Stichprobenstandardabweichung von 2500 führen.

Konzept: Eine Inspektion der Formeln zeigt, daß beide Vorschläge noch stärker auf Ausreißer reagieren als das arithmetische Mittel. Vor dem Einsatz sollte man offensichtliche Ausreißer entfernen. Dennoch ist eine getrimmte Standardabweichung unüblich. Sicher auch deswegen, weil es wieder unempfindlichere Maße gibt. Hier soll nur noch der Quartilsabstand genannt werden, der sich als Differenz des 0.75- und des 0.25-Quantils ergibt.

Algorithmus: Der Interquartils-Abstand von $\mathtt{x} = (x_1, \ldots, x_n)$
ergibt sich als Differenz des dritten und des ersten Quartils.

```
quantile(x,0.75)-quantile(x,0.25)
```

12 ⟨*berechne Inter-Quartilsabstand von* x 12⟩≡
```
as.vector(quantile(x,0.75)-quantile(x,0.25))
```

Praxis:

```
Inter-Quartilsabstand
[1] 682
```

Der Größenordnung nach paßt dieses Ergebnis besser zu den Lokalisationsmaßen Median und getrimmtes arithmetisches Mittel.
Als Antwort auf die Belastungsfragen können wir formulieren: Jede zweite Zwischenzeit zwischen zwei Zugriffszeitpunkten liegt in etwa unter 6 Minuten. 50 % der Zeitabstände liegen in einem Intervall mit einer Länge von etwas weniger als 11 Minuten.

3.3 Weitere Überlegungen

Technik: Es ist ein wenig langweilig, die verschiedenen Maßzahlen einzeln zu berechnen. Deshalb lassen sich eine Reihe charakteristischer Werte mit folgendem Modul ermitteln. Betrachten wir das Beispiel:

13 ⟨*berechne zusammenfassende Statistiken von* x 13⟩≡
```
summary.stats(x)
```

Praxis:

```
Zusammenfassende Statistiken
 Min. 1st Qu. Median  Mean 3rd Qu.  Max. Std.Dev.   n
    1    62.5    312 823.8   744.5 24450   2487.4 103
```

Wir sehen die Extremwerte, den Mittelwert, den Median und die beiden Quartile. Außerdem werden die Stichprobenstandardabweichung (`Std.Dev.`) und der Stichprobenumfang mit ausgegeben. Damit wird der Datensatz auf acht Zahlen reduziert, aus denen vieles bezüglich Lage und Variabilität abzulesen ist. Mit diesen Zahlen ist der Datensatz recht gut beschrieben. Im besonderen erkennt man, daß die Daten asymmetrisch verteilt sind. Median und Mittel stimmen nicht gut überein und beide liegen nicht in der Mitte zwischen den Extremwerten. (Übrigens werden manchmal zur Datenanalyse sogenannte Letter-Value-Diagramme erstellt. Diese zeigen den Median, die Extremwerte und die Quartile sowie den Stichprobenumfang in tabellarischer Form. Versuchen Sie, in der Literatur Beispiele zu finden! Überlegen Sie den Gebrauchswert dieser Diagramme!)

Konzept: Verteilungen mit solchen Eigenschaften werden in der Statistik als schief bezeichnet. Natürlich gibt es auch Maßzahlen, um die Schiefe zu beschreiben. Solche werden hier nicht näher betrachtet. Es sei jedoch darauf hingewiesen, daß schiefe Verteilungen für weitergehende Analysen problematischer sein können als symmetrische. Deshalb findet man Transformationsvorschläge, mit denen Datensätze in weniger asymmetrische überführbar sind.

Eine Klasse von Transformationen ist durch die Box-Cox-Transformationen gegeben.

Algorithmus: Box-Cox-Transformationen $T(x)$ lassen sich beschreiben durch:

$$T(x) = \begin{cases} (x^\lambda - 1)/\lambda & \textit{für} \quad \lambda \neq 0 \\ \ln(x) & \textit{für} \quad \lambda = 0 \end{cases}$$

Je nach Wahl des Parameters λ ergibt sich eine andere Transformation.

Praxis: Damit stellt sich die Frage, welches in einer spezifischen Datensatzsituation der passende Wert für λ ist. Sie können nun verschiedene Werte ausprobieren. Häufig wird empfohlen, nacheinander $\cdots, 3, 2, 1, 0.5, 0, -0.5, -1, -2, \cdots$ auszuprobieren.

Technik: Der gewünschte Wert für λ (`lambda`) wird erfragt. Außerdem wird der Bediener gefragt, ob er mit den transformierten Werten die alten `x`-Werten überschreiben möchte.

14 ⟨*transformiere* x *mittels Box-Cox-Transformation, zeige Statistiken* 14⟩≡
```
xtrans<-box.cox.transformation(x)
print(summary.stats(xtrans))
print("Wollen Sie das Transformationsergebnis auf x ablegen (j/n)?")
if(readline()[1]=="j") x<-xtrans
```

Praxis: Mit $\lambda = 0$ erhält man schon eher symmetrisch zu nennende transformierte Daten, zum Beispiel das neue Mittel recht gut in der Mitte zwischen den Extremwerten liegt:

```
Statistiken der transformierten Daten
[1] "Bitte lambda eingeben:"
1: 0
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.   n
    0   4.133  5.743 5.265   6.612 10.1   2.0082 103
```

3.4 Offene Fragen

Praxis: Nachdem wir nun in der Lage sind, bestimmte Eigenschaften durch Maßzahlen einzufangen, bleiben aber noch viele Fragen übrig.

- Wie kann man ein umfassenderes Bild der Belastung, die an einem Tag herrschte, erhalten?
- Wie lassen sich Belastungen verschiedener Tage in Beziehung setzen?
- Wie lassen sich Fragen der Art beantworten, ob im allgemeinen die Belastung schon eine bestimmte Größe überschritten hat?

3.5 Aufgaben

Praxis: Die folgenden Aufgaben sollen mittels der Operationsmöglichkeiten, die mit dem folgenden Menü-Modul bereitgestellt werden, bearbeitet werden.

15 ⟨* 1⟩+≡
```
cat("Revbook:1.m:menu.mz:Menue Masszahlen\n")
if(exists("auswahl.masszahlen")) auswahl.masszahlen()
```

Zur Information wird das Auswahlmenü präsentiert:

```
lege Daten auf  x  ab
zeige  x  an
zeige Rangwertreihe von  x  an
berechne Mittelwert
berechne Median
berechne Extrema
berechne Midrange
berechne getrimmtes Mittel
berechne Trimean
berechne Stichprobenvarianz
berechne Standardabweichung
berechne Inter-Quartilsabstand
berechne zusammenfassende Statistiken
transformiere mittels Box-Cox-Transformation
```

a) Welche allgemeinen Erwartungen haben Sie bezüglich Lage und Variabilität von Zwischenzeiten entwickelt? Formulieren Sie diese und überprüfen Sie Ihre Einschätzung mit Hilfe der Daten vom 17.02.97! Aktivieren Sie dazu den geeigneten Code!

b) Hängt Ihrer Meinung nach die Belastung auch von den transportierten Mengen ab? Wenn ja, warum? Was erwarten Sie? Überprüfen Sie Ihre Ansichten mit den Mengen, die für den 3.02.97 festgestellt wurden!

c) Die letzte Übung sollten Sie mit den Mengen vom 17.02.97 wiederholen. Falls Ihnen die Verteilung unsymmetrisch erscheint, sollten Sie versuchen, durch Transformation dieses zu beheben.

d) Weitere Experimente können Sie anstellen, wenn Sie die transportierten Mengen durch die Zwischenzeiten dividieren. Hierdurch erhalten Sie einen neuen Ansatz, um die Belastung zu erfassen. Was spricht für diese Idee? Welche Lage und welche Variabilität besitzen die Quotienten?

e) Auch hier lassen sich die Ergebnisse mittels des zweiten Datensatzes validieren.

4 Graphische Beschreibungstechniken

Technik: Das Kapitel beginnt wieder mit einer Anweisung für die Menüsteuerung, mit deren Hilfe der Leser am Rechner sofort zum Start dieses Kapitels springen kann. Es sei daran erinnert, daß der Leser mit: `s MENU` die Menüsteuerung starten kann.

16 ⟨* 1⟩+≡

```
cat("Revbook:2.a:start.gr:Kapitel Graphische Beschreibungstechniken\n")
```

Zum Literaturstudium sei hingewiesen auf: [Autorenkollektiv: Abschnitt über Graphische Darstellungen des Inhalts von Häufigkeitstabellen sowie Abschnitt über empirische Verteilungsfunktionen], [Bamberg, Baur: Kapitel über Auswertungsmethoden für eindimensionales Datenmaterial], [Schlittgen: Kapitel über Darstellung eindimensionaler Datensätze].

Konzept: So schön wie der Versuch ist, einen großen Datensatz mit wenigen Statistiken zu beschreiben, ist er aber auch gleichzeitig sehr unbefriedigend. Man möchte einen Eindruck von den Daten gewinnen und sich nicht mit wenigen zusammenfassenden Werten abspeisen lassen. Am liebsten möchte man die Daten sehen. Deshalb ist nichts naheliegender, als graphische Antworten auf seine Fragen zu bekommen. So könnte dieser Abschnitt überschrieben werden mit: *Eine Lanze für graphische Analysewerkzeuge.* Maßzahlen besitzen zwar den Vorteil, daß man deren Eigenschaften formal beschreiben und mit ihnen leicht Vergleiche anstellen kann. Demgegenüber besteht allerdings der Nachteil, daß besondere Charakteristika der Daten übersehen werden können. Bei einer exploratorischen Analyse gilt es jedoch beispielsweise, auch Besonderheiten von Datensätzen festzustellen. Dieses geht mit graphischen Methoden viel leichter. Leider waren viele dieser Methoden in der Zeit, in denen es noch keine Rechner mit geeigneter Software gab, nur mühsam anzuwenden. In der heutigen Zeit mit ihren neuen Instrumenten ist dieses Problem (fast) verschwunden, so daß exploratorische Vorgehensweisen großen Rückenwind erhalten. In diesem Kapitel sollen Möglichkeiten graphischer Methoden für eindimensionale Datensätze aufgezeigt werden, aber mit dem Hinweis, daß gedankenloses Ausprobieren nicht unter die Überschrift Statistik fällt.
Zu den oben aufgeführten allgemeinen Fragen

- Wo liegt das Zentrum der Beobachtungen?
- Wie stark streuen die Beobachtungen?
- Sind die Beobachtungen symmetrisch verteilt?
- Wie sehen die Verteilungen in den Randbereichen aus?
- Gibt es Ausreißer?

kommen hinzu

- Wie lassen sich wesentliche Maßzahlen visualisieren?
- Wie erhält man einen ersten Eindruck von der Verteilung der Daten?
- In welchen Bereichen besitzen die Daten welche Häufungen?
- Wieviel Prozent der Daten liegen unterhalb eines vorgegebenen Wertes?

Im folgenden werden die zum Teil um den Einsatz konkurrienden Instrumente Boxplot, Jitterplot, Stabdiagramm, Stem-and-Leaf-Diagramm, Histogramm, Dichtespur und der Plot der empirischen Verteilungsfunktion vorgestellt.

Praxis: Gegenstand bildet wieder der Datensatz `dzeitpunkte.17.02.97` zusammen mit den oben gestellten Fragen zur Serverbelastung.

Technik: Damit die Modulstruktur dieses Kapitels zu der des letzten paßt, legen wir die Zwischenzeiten auf den Variablen `x` ab.

17 ⟨*lege Zwischen-Zeitpunkte vom 03.02.97 auf* `x` *ab* 17⟩≡
```
x<-dzeitpunkte.03.02.97
```

4.1 Der Boxplot als verbesserte graphische 5-Zahlen-Zusammenfassung

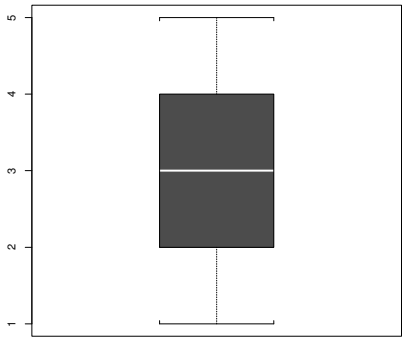
Konzept: Die fünf Größen Minimum, unteres und oberes Quartil, Median, Maximum werden gern unter Angabe der Werteanzahl als 5-Zahlen-Zusammenfassung bezeichnet. Wie lassen sich diese den Datensatz beschreibende Zahlen für den Analytiker visualisieren?

Algorithmus: Graphische 5-Zahlen-Zusammenfassung
Zeichne eine vertikale Achse, die zu dem Wertebereich der Beobachtungen paßt. Zeichne eine Box, die in vertikaler Richtung vom ersten Quartil bis zum dritten Quartil reicht. Die Breite der Box ist beliebig. In der Höhe des Medians trage in die Box eine horizontale Trennlinie ein. Markiere die Extremwerte durch weitere horizontale Linien. Füge zwischen den Extremwerten und der Box vertikale Verbindungslinien ein, um die Zusammengehörigkeit anzudeuten.

Die Zahlen von eins bis fünf besitzen die Extrema eins und fünf, den Median drei und die Quartile zwei und vier. Als graphische 5-Zahlen-Zusammenfassung ergibt sich mit dem Modul:

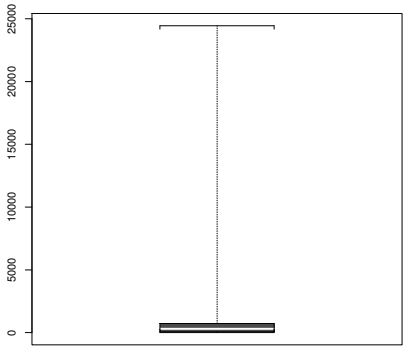
18 ⟨*erstelle graphische 5-Zahlen-Zusammenfassung von 1, 2, 3, 4, 5* 18⟩≡
```
boxplot(1:5,range=0)
```

... folgendes Bild:



Praxis: Wir wollen den entsprechenden Plot zu unserem Datensatz sehen.

19 ⟨*erstelle graphische 5-Zahlen-Zusammenfassung von* x 19⟩≡
```
boxplot(x,range=0)
```



Die Box, in der 50 % der Daten liegen sollen, ist nicht mehr als Box zu erkennen. Sie befindet sich im unteren Bereich des Bildes, fast zu einem Strich degeneriert. Der größte Teil der Daten muß unter 1000 gesucht werden. Damit hat man eine grobe Vorstellung von der Lage der Daten. Besonders auffällig ist die starke Asymmetrie des Datensatzes, für die das obere Viertel der Werte verantwortlich zu sein scheint. Diese großen Werte beanspruchen soviel von dem gesamten Darstellungsbereich, daß sich selbst das Minimum nicht mehr von der Box abhebt. Für eine präzisere Antwort auf die Lagefrage müßte man eine Ausschnittsvergrößerung betrachten.

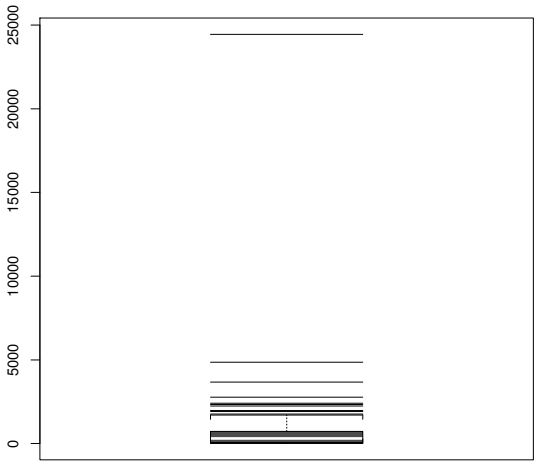
Konzept: Solche Plots sollen einen ersten Gesamteindruck vermitteln. Zu dem ersten Eindruck gehören Hinweise darüber, wie stark die Werte streuen. Hierzu lassen sich die Ausmaße der Box und die Lage der Extremwerte betrachten. Der Beispielplot sollte übrigens nicht als Gegenbeispiel gewertet werden, denn er hat doch wohl einen deutlichen Eindruck hervorgerufen. Wer hat sich nach dem letzten Kapitel eine Vorstellung von dem Datensatz gebildet, die mit dem Eindruck aufgrund des letzten Plots übereinstimmt?
Werden die vertikalen Linien (whiskers), die von der Box ausgehen, höchstens eine standardisierte Spanne lang und weiter entfernt liegende Werte isoliert gezeichnet, erhält man den Boxplot.

Algorithmus: Boxplot

1. *Zeichne eine Box mit Medianlinie nach dem Algorithmus* Graphische 5-Zahlen-Zusammenfassung
2. *Bestimme als Grenzen die beiden Stellen, die sich 1.5×Interquartils-Abstand über beziehungsweise unter der Box befinden.*
3. *Bestimme die beiden extremsten Beobachtungen, die gerade noch nicht jenseits der in Schritt 2 ermittelten Grenzen liegen, markiere diese durch horizontale Striche und verbinde diese Striche durch vertikale Linien mit der Box.*
4. *Markiere alle außerhalb der Grenzen liegenden Punkte durch horizontale Striche.*

Praxis: Ein Boxplot kann vielleicht das Viertel der größeren Werte etwas aufsplitten. Wir wollen den Boxplot erstellen.

20 ⟨*erstelle Boxplot zu* x 20⟩≡
```
boxplot(x)
```



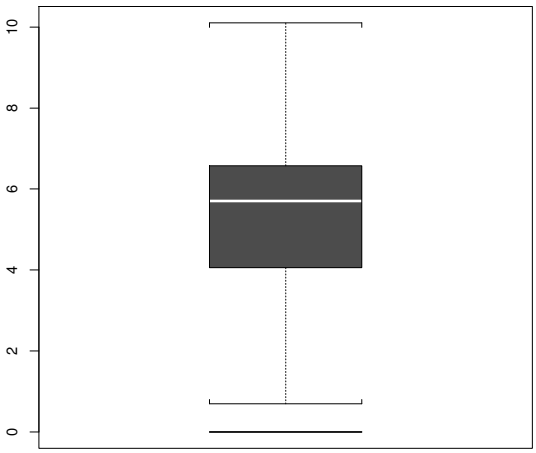
Wesentlich wirkt sich auf das Erscheinungsbild wieder der größte Beobachtungswert aus. Dieser liegt weit von den übrigen Werten entfernt. Man erahnt den Einfluß der großen Zwischenzeiten auf Mittel (Schwerpunkt) und Stichprobenvarianz. Möglicherweise sind mehrere Werte von der Größe 24449 vorhanden. Wir wollen uns die 10 größten und die 10 kleinsten Werte (das sind die ersten und die letzten 10 Werte der Rangwertreihe) ausdrucken.

21 ⟨*zeige die 10 kleinsten und die 10 größten Werte von* `x` *an* 21⟩≡
```
cat("10 kleinsten Werte von x:\n")
print(sort(x)[1:10])
cat("10 groessten Werte von x:\n")
print(rev(sort(x))[1:10])
```

```
10 kleinsten Werte von x:
 [1]  1  1  1  2  2  6  6  8  8 10
10 groessten Werte von x:
 [1] 24449  4862  3670  2772  2409  2330  2236  1965  1913  1754
```

Offensichtlich gibt es nur einen Wert der Größe 24449. Der zweitgrößte ist von dem Maximum schon erheblich entfernt. Weiter sieht man, daß die Abstände zwischen den anderen großen Werten noch um ein Vielfaches größer sind als die Abstände zwischen den kleinsten Werten.
Die enorme Asymmetrie unseres Datensatzes kann vielleicht durch eine Transformation gemildert werden. Versuchen wir das Logarithmieren. Wir wollen deshalb einen Boxplot der logarithmierten Daten erstellen.

22 ⟨*erstelle Boxplot zu* `log(x)` 22⟩≡
```
boxplot(log(x))
```



In der Tat ist das Erscheinungsbild jetzt deutlich symmetrischer. Das logarithmierte Minimum wird isoliert als möglicher Ausreißer dargestellt. Der horizontale Strich markiert übrigens gleich drei Beobachtungen (vgl. oben die kleinsten 10 Werte).
Wir wollen das Maximum als Ausreißer einstufen und für die weiteren Betrachtungen aus dem Datensatz `x` entfernen.

Konzept: Eine Beobachtung wird nicht durch eine technische Definition zu einem Ausreißer, sondern durch eine vorsetzliche Einstufung. Insofern liefert ein Boxplot höchstens Indizien auf eventuell als Ausreißer einzustufende Werte. Große Werte können auf Verarbeitungsfehler zurückgehen und damit falsche Werte sein, sie können aber auch sehr wohl korrekt und von großer Bedeutung sein. Werden Wasserstände untersucht, darf man das Hochwasser, das hinter einem sehr großen Wert steht, nicht einfach ignorieren.

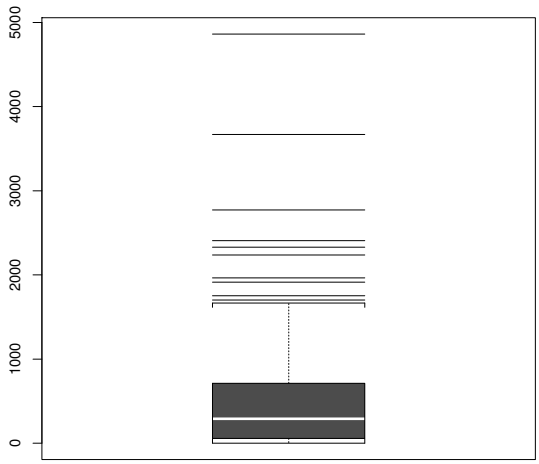
Technik: Das folgende Modul zur Festlegung eines Datenausschnitts erfragt Grenzen, jenseits derer Beobachtungen ausgesondert werden sollen. Durch geeignete Eingaben wird nur das Maximum 24449 entfernt.

23 ⟨*bilde Ausschnitt der Daten* x 23⟩≡
```
x<-remove.extreme.values(x)
```

Wir wollen den Boxplot für die verbleibenden Daten erstellen.

24 ⟨* 1⟩+≡
```
boxplot(x)
```

Wir erhalten:



Dieser Boxplot erscheint für die Belastungsfragen eher geeignet zu sein. Wir sehen, daß 75 % der Daten unterhalb von ca. 800 liegen, mehr als 50 % liegen unter 500. Zum Maximum hin dünnen die Werte immer mehr aus. Das Minimum liegt nicht weit vom Zentrum der Daten entfernt.

Konzept: Ein Boxplot zeigt Lage und Variabilität eines Datensatzes, wir erkennen die Schiefe der Verteilung und das Verhalten an den Enden (Schwänzen) der Verteilung. Mögliche Ausreißer (gemäß einer technischen Festlegung) werden besonders hervorgehoben. Damit ist ein Boxplot einfach, informativ und ermöglicht übrigens auch schnell Vergleiche zwischen Datensätzen.

Praxis: Wer noch ein wenig Erfahrungen mit dem Instrument Boxplot sammeln möchte, für den steht ein Boxplot-Erfahrungs-Sammel-Modul zur Verfügung. Hier gilt: Probieren geht über Studieren.

25 ⟨*starte Erfahrungssammelfunktion zum Boxplot* 25⟩≡
```
cat("Gib Daten ein!\n")
boxplot(as.numeric(scan(what="")))
```

Praxis: Zu dem Werkzeug Boxplot soll noch bemerkt werden, daß die Grenzen der Box unterschiedlich definiert werden, manchmal auch mittels sogenannter hinges (Fourths, Angeln). Deren Definition weicht ein wenig von unserer Festlegung der Quartile ab. Hier wollen wir nicht näher auf die geringen Unterschiede eingehen, die sich im wesentlichen bei sehr kleinen Datensätzen zeigen. Für weitere Fragen sei verwiesen auf [Hoaglin et al.]. Boxplots, die mit anderen Programmen erstellt werden, können deshalb etwas voneinander abweichen. Auch läßt sich oft das genaue Erscheinungsbild der Boxplots über weitere Parameter beeinflussen, so daß die Diskussion hier noch fortgeführt werden könnte. Wir wollen uns jedoch lieber anderen graphischen Instrumenten zuwenden.

4.2 Der Jitterplot zur vollständigen Darstellung

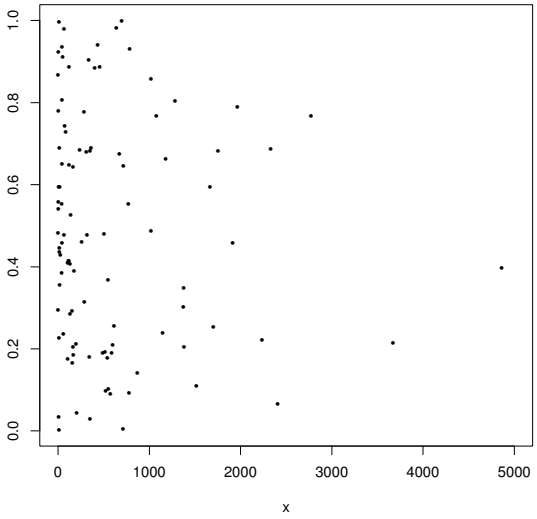
Konzept: Der Zweck von graphischen Werkzeugen liegt natürlich nicht nur darin, Verdichtungsergebnisse (zusammenfassende Statistiken) zu visualisieren, sondern es besteht auch die Chance, alle Daten sprechen zu lassen, zum Beispiel mit einem Jitterplot. Ziel eines Jitterplots ist es, alle Beobachtungen mit Hinweisen auf unterschiedliche Datendichte darzustellen.

Algorithmus: `jitterplot(x)` — *Jitterplot von* $\mathtt{x} = (x_1, \ldots, x_n)$

Im Jitterplot werden Zufallszahlen gegen Beobachtungen abgetragen.

Zufallszahlen haben Eigenschaften wie Zahlen, die zufällig ausgewählt werden. Die Zufallszahlen übermitteln keine weiteren Informationen, ihre Bedeutung liegt nur in der Generierung des Erscheinungsbildes, wie es in der nächsten Abbildung zu sehen ist.

26 ⟨*erstelle Jitterplot zu* x 26⟩≡
```
jitterplot(x)
```



Praxis: In diesem Plot sind alle noch auf x befindlichen Beobachtungen abgedruckt.
Durch die zufälligen y-Werte wirkt das Ergebnis wie ein Mückenschwarm oder die Bestandteile eines Gases, wobei die Dichte am linken Bildrand am größten zu sein scheint. Die bisherigen Ergebnisse bezüglich der Lage werden unterstützt.

Konzept: Die den Daten innewohnende Variabilität wird offensichtlich. Auch dürfte noch einmal deutlich werden, daß eine einzige Zahl nicht als Antwort auf die gestellten Belastungsfragen ausreicht.

4.3 Stab- und Balkendiagramm als graphisches Zählergebnis

Konzept: Neben den Darstellungen, in denen Beobachtungen als einzelne Punkte auftauchen, gibt es Vorschläge, die Daten für bestimmte Fragen aufzubereiten. Ein einfaches Werkzeug ist das Stabdiagramm, an dem abgelesen werden kann, wie häufig eine bestimmte Merkmalsausprägung beobachtet worden ist.

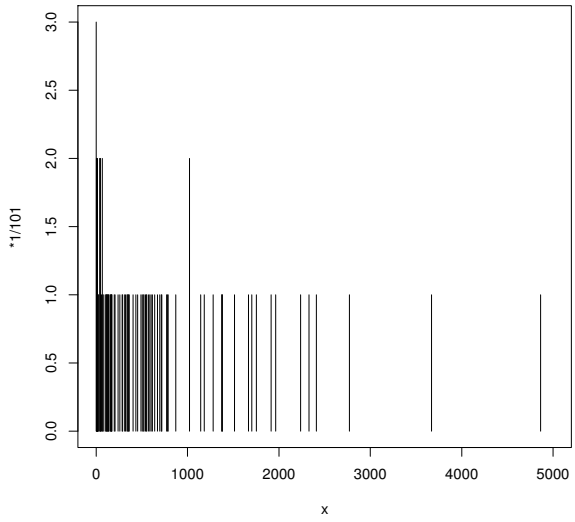
Algorithmus: Stabdiagramm

Im Stabdiagramm werden die relativen Häufigkeiten der verschiedenen Ausprägungen durch entsprechend lange Stäbe über den jeweiligen Beobachtungen dargestellt.

Technik: Damit aufgrund des Bildes schnell absolute Häufigkeiten ermittelt werden können, sind auf der vertikalen Achse ganze Zahlen notiert. Werden diese gemäß der Achsenbeschriftung mit 1/101 multipliziert, erhält man die relativen Häufigkeiten.

27 ⟨*erstelle Stabdiagramm zu* x 27⟩≡
```
stabdiagramm(x)
```

Praxis:



Im wesentlichen werden die Eindrücke aus dem Jitterplot bestätigt. Bezüglich Fragen der Dichtheit stellt das Stabdiagramm gegenüber dem Jitterplot keine Verbesserung dar. Wir sehen, daß die meisten Werte nur einzeln vorkommen, einige Ausprägungen wurden doppelt getroffen und eine trat dreifach auf. Zum Überprüfung läßt sich schnell die Häufigkeitstabelle berechnen.

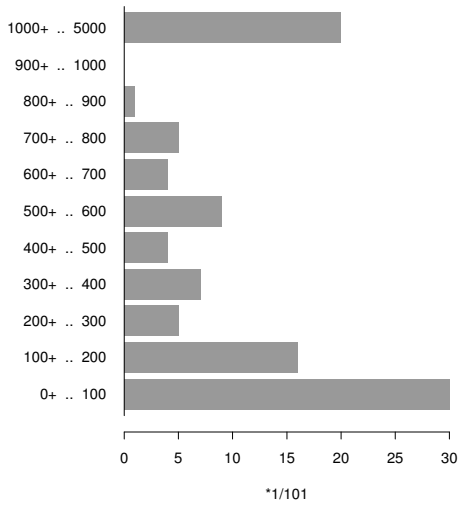
28 ⟨*erstelle Häufigkeitstabelle zu* x 28⟩≡
```
table(x)
```

```
1 2 6 8 10 11 12 14 15 19 21 29 40 44 46 49 58 67 73 83 106 107 116 117 119
3 2 2 2  1  1  1  2  1  1  1  1  2  2  2  1  1  2  1  1   1   1   1   1   1

120 131 133 139 153 158 164 165 166 174 197 207 240 260 285 290 312 320 337
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1

344 350 351 362 403 433 455 490 503 516 524 540 547 554 573 588 600 615 639
  1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1

672 697 712 718 771 778 788 870 1021 1145 1183 1281 1374 1378 1381 1515 1667
  1   1   1   1   1   1   1   1    2    1    1    1    1    1    1    1    1

1702 1754 1913 1965 2236 2330 2409 2772 3670 4862
   1    1    1    1    1    1    1    1    1    1
```

Ein Stabdiagramm ist für den Problembereich der Zeitdifferenzen offensichtlich nicht besonders hilfreich. Eher interessiert die Frage, wie häufig sich Ausprägungen in bestimmten Abschnitten oder Intervallen eingestellt haben.

Praxis: Zur Ermittlung, wie viele Vorkommnisse in den Klassen 0..100, 100..200, 200..300, ..., 1000..5000 liegen, kann ein aggregiertes Diagramm, nennen wir es Balkendiagramm, hilfreich sein. (Ein solcher Plot ist allgemein unüblich und dient hier nur zu didaktischen Zwecken.)

29 ⟨*erstelle spezielles Balkendiagramm zu* x 29⟩≡
```
balkendiagramm(x,c(0,100,200,300,400,500,600,700,800,900,1000,5000))
```



Konzept: Neben der Bestätigung bisheriger Erkenntnisse lassen sich die Anzahlen in den einzelnen Klassen leicht ablesen. Man muß jedoch beachten, daß die Klassen unterschiedliche Breiten aufweisen. Dadurch könnte es zu Fehlinterpretationen kommen. Der Stem-and-Leaf-Plot und das Histogramm als Modifikation dieser Balkengraphik sind von solchen Problemen befreit. An dieser Stelle soll noch einmal die Problematik des Stabdiagramms reflektiert werden. Denn die mangelnde Eignung des Diagramms liegt nicht in den speziellen Werten des Datensatzes begründet, sondern darin, daß das Merkmal *Zeitdifferenzen* als ein kontinuierliches Merkmal einzustufen ist. Alle Punkte aus einem Zeit*intervall* sind nämlich als Realisation denkbar. Bei solchen Gegebenheiten geht man in der Regel zur Betrachtung, Darstellung (siehe zum Beispiel Histogramm) und Interpretation von Intervallen über. Lassen sich nur Ausprägungen vorstellen, die nicht zu Intervallen zusammengefaßt werden können, spricht man naheliegenderweise von diskreten Merkmalen. Bei diesen kann ein Stabdiagramm sehr sinnvoll sein. Dazu denke man nur an die Realisationen von 100 Würfelwürfen, bei denen die relative Häufigkeit der möglichen Ausprägungen $\{1, 2, 3, 4, 5, 6\}$ sehr wohl interessiert. Es könnte jetzt eingewendet werden, daß die Zeiten und Zeitdifferenzen nur in ganzen Sekunden erhoben worden sind und damit ein diskretes Merkmal zugrundeliegt. Dem ist zu entgegnen, daß zum ersten auch Zwischenwerte interpretierbar sind und zum zweiten diskrete Merkmale mit sehr vielen Ausprägungen oft wie kontinuierliche Merkmale behandelt werden.

4.4 Der Stem-and-Leaf-Plot als numerisches Balkendiagramm

Konzept: Der Stem-and-Leaf-Plot versucht, Bereiche unterschiedlicher Datendichte aufzudecken. Damit lassen sich Häufigkeiten (der Belastung) eher erkennen als mit einem Boxplot, einem Jitterplot oder einem Stabdiagramm.
Dieser aus Textzeichen aufgebaute Plot zeigt Beobachtungsanzahl, Median, Quartile und ein numerisches Balkendiagramm, eventuell gefolgt von einigen speziellen Werten.

Praxis: Betrachten wir ein Beispiel:

30 ⟨*erstelle Stem-and-Leaf-Plot zu* x 30⟩≡
```
stem.leaf(x)
```

Es sei daran erinnert, daß von x der extrem große Wert 24449 entfernt worden war.

```
N = 102   Median = 301
Quartiles = 58, 718

Decimal point is 2 places to the right of the colon

    0 : 000001111111111223444455567778
    1 : 112222334566677
    2 : 014689
    3 : 1244556
    4 : 0359
    5 : 02245579
    6 : 0147
    7 : 012789
    8 : 7
    9 :
   10 : 228
   11 : 48
   12 : 8
   13 : 788
   14 :
   15 : 1
   16 : 7
   17 : 05
   18 :
   19 : 16

High:   2236   2330   2409   2772   3670   4862
```

Konzept:

> *Algorithmus: Stem-and-Leaf-Diagramm*
>
> *Zunächst wird festgestellt, in welchem Bereich die Beobachtungen liegen. Dann wird der relevante Bereich in ausreichend viele Klassen zerlegt. Nun wird eine aus zwei durch einen Doppelpunkt getrennte Spalten bestehende Tabelle erstellt, die so viele Zeilen besitzt, wie Klassen definiert worden sind. Links von den* `":"`*-Zeichen werden Klassenbezeichnungen notiert. Sie werden aus der ersten oder den ersten Ziffern der Klassenuntergrenzen gebildet. Rechts neben dem* `":"`*-Zeichen wird für jede Beobachtung in der passenden Tabellenzeile eine weitere signifikante Ziffer eingetragen. So ergeben sich in den Zeilen Ziffernbalken, aus deren Längen die Besetzungszahl der Klassen folgt. Innerhalb der Balken werden die einzelnen Ziffern sortiert aufgeführt. Die Klassenbezeichnungen werden als Stamm (stem) und die Beobachtungsrepräsentanten als Blätter (leafs) bezeichnet. Über dem Diagramm wird ein Hinweis zur Lage des Dezimalpunktes bezüglich des Stammes gegeben. Unter dem Diagramm werden besonders große und kleine Werte aufgeführt, die die Darstellung zu sehr in die Länge ziehen würden; sie sind eventuell als Ausreißer einzustufen.*

Praxis: Mit diesem Plot erhalten wir zum ersten Mal eine Abschätzung der Datendichte in den verschiedenen Bereichen. Es sieht so aus, daß die Klassenhäufigkeiten mit der Intervallänge kontinuierlich abnehmen. Sehr kleine Zwischenzeiten sind am häufigsten.
Letzteres ist deshalb bemerkenswert, weil bei kleinen Zwischenzeiten zwischen Zugriffen die Belastung am größten ist. Den WEB-Master dürfte diese Erkenntnis nicht zu sehr erfreuen oder doch?

Konzept: Schwierig ist beim Stem-and-Leaf-Plot die Festlegung der Klassenanzahl und damit der Klassenbreiten. Als Vorteile des Diagramms ergeben sich: Es kann relativ schnell per Hand erstellt werden, wenn der Datenumfang nicht zu groß ist. Es erfordert keine hochauflösende Graphik. Es ist aussagekräfig bezüglich Lage, Variabilität, Schiefe und möglicher Ausreißer. Es eignet sich auch zum Ordnen der Werte.
In gewisser Weise ist das Histogramm, obwohl älter, eine optische, aber nicht unbedingt qualitative Weiterentwicklung des Stem-and-Leaf-Diagramms. Es versucht, die im Jitterplot gesehene Dichtheit der Daten mit Hilfe relativer Häufigkeiten darzustellen.

4.5 Das Histogramm zur Abschätzung der Dichte

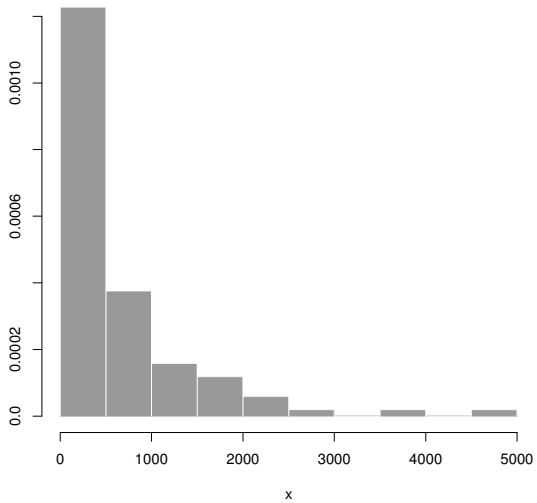
Konzept: Das Histogramm ist eine Darstellung der relativen Häufigkeiten von klassierten Daten.

> *Algorithmus: Histogramm*
>
> *Zunächst werden Klassen gebildet, und es wird festgestellt, wie viele Beobachtungen in den Klassen liegen. Über jeder Klasse wird dann ein Rechteck gezeichnet, dessen Flächeninhalt der relativen Häufigkeit entspricht.*

Damit wird die Dichtevorstellung, die der Jitterplot implizit geliefert hat, explizit darstellt.

Praxis: Aus dem Datensatz `x` wird folgendes Bild erstellt.

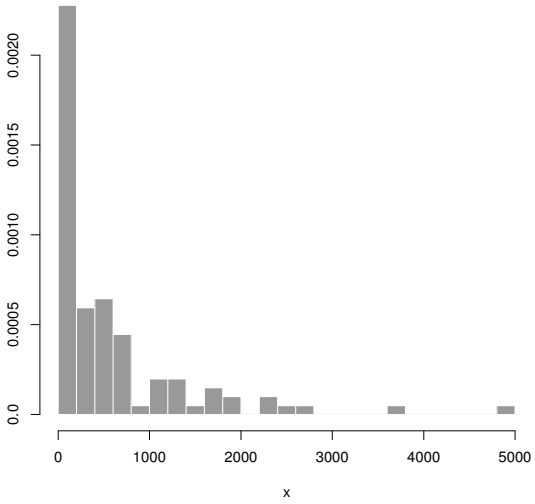
31 ⟨*erstelle Histogramm zu* x 31⟩≡
```
histogramm(x)
```



Wir können an diesem Diagramm ablesen, daß in 0.0012*500 = 60 % aller Fälle die Zeitdifferenzen kleiner als 500 waren (Höhe des ersten Rechtecks: 0.0012, Rechteckbreite: 500). Die Klassengrenzen wurden automatisch gewählt. Sie können aber auch vorgegeben werden.

32 ⟨*erstelle Histogramm zu* x *mit Grenzen 0, 200, 400, ..., 5000* 32⟩≡
```
hxh<-pmin(5000,pmax(0,x))
if(length(hxh)!=length(x))
    cat("Warnung: Werte wurden auf den Bereich [0,5000] reduziert!\n")
histogramm(hxh,breaks=seq(from=0,to=5000,by=200))
```

Dieses Modul liefert folgendes Bild:

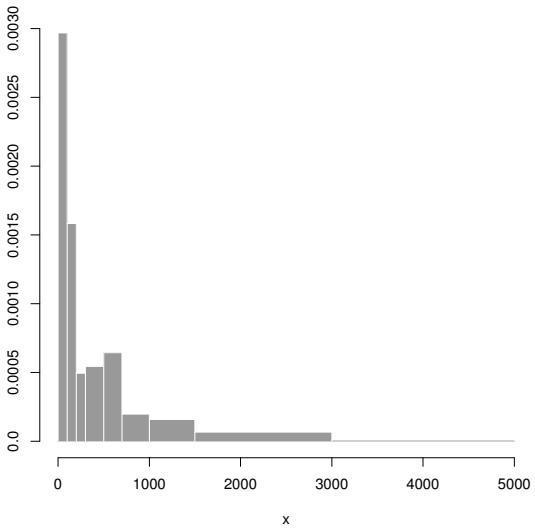


Wir wollen noch ein Modul bereitstellen, mit dem Klassengrenzen per Hand eingegeben werden können.

33 ⟨*erstelle Histogramm von* x *mit manueller Wahl der Grenzen* 33⟩≡

```
print("Grenzen nach einander eingeben")
print("Nach Eingabe der letzten Klasse noch einmal: -> ENTER")
print("Hinweis: Alle Werte muessen sich innerhalb der Grenzen befinden!")
histogramm(x,breaks=sort(scan(,0)))
```

Das folgende Histogramm erhält man für die Eingabe der Grenzen: 0, 100, 200, 300, 500, 700, 1000, 1500, 3000, 5000.



Konzept: Die Wahl der Klassengrenzen ist auch für die Zweckmäßigkeit des Histogramms entscheidend. Werden die Klassen noch enger gewählt, wird das Erscheinungsbild noch zittriger und dem Stabdiagramm ähnlicher werden. Gibt es zu wenige Klassen, wird zuviel Information weggeglättet. Die beste Lösung dürfte, wie häufig, in der Mitte liegen.
Es sei darauf hingewiesen, daß bei kontinuierlichen Merkmalen das Histogramm die Zwecke erfüllt, für die im diskreten Fall das Stabdiagramm herangezogen wird.

Unzufriedenheit mit dem Histogramm führt zu Dichtespuren:

4.6 Dichtespuren als Alternativen zu Histogrammen

Konzept: Histogramme lassen sich über Häufigkeitstabellen schnell erstellen. Das ist ein Vorteil. Sie besitzen eckige Silhouetten. Das ist für den Betrachter ein Nachteil. Denn man hat das Gefühl (siehe Jitterplot), daß die rechteckige Struktur nicht zu rechtfertigen ist. Dichtespuren stellen hier eine Antwort dar. Der kleine Nachteil des erhöhten Rechenaufwandes wird durch die heutigen Rechner relativiert. Beim Histogramm werden Klassen festgesetzt, in denen dann die zugehörigen Beobachtungen gezählt werden. Zur Berechnung der Dichtespur geht man ähnlich vor, nur daß zu jeder Stelle der x-Achse eine eigene Klasse oder besser Umgebung festgelegt wird und die darin gefundenen Beobachtungen gewichtet verarbeitet werden.
Man sollte sich nicht durch die folgende Formel abschrecken lassen. Die wesentlichen Aspekte sind folgende: Es werden zu einer Stelle y die Abstände zu den Beobachtungen ermittelt und verarbeitet. Die Funktion $K(\cdot)$ — auch Kernel genannt — gewichtet nah zu y liegende Beobachtungen stark und weit entfernte schwach. Die Gewichtungsergebnisse werden zusammengezählt und durch n dividiert, wie wir dies von der Durchschnittsbildung her kennen. Weitere Details sollen uns hier nicht näher interessieren.

Algorithmus: `density(x,w)` — *Dichtespur* $\hat{f}(y;x,w)$

$$\hat{f}(y;x,w) = \frac{1}{nw}\sum_{i=1}^{n} K\left(\frac{y-x_i}{w}\right) \quad == \texttt{sum(K((y - x)/w))/w}$$

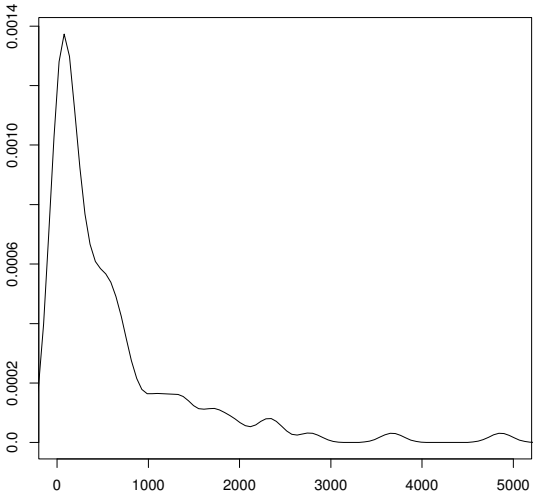
Hierbei ist w die sogenannte Fensterbreite und

$$K(t) = \sqrt{\frac{8}{w\pi}}\exp\left(-\left(\frac{4t}{w}\right)^2\right) \quad == \texttt{(8/(w * pi))\^{}0.5 * exp(-(4 * t)\^{}2/w\^{}2)}$$

Technik: Das folgende Modul erstellt eine Dichtespur. Die Fensterbreite wird dabei erfragt.

34 ⟨*erstelle Dichtespur zu* x 34⟩≡
```
dichtespur(x)
```

Praxis: Mit einer Fensterbreite, die ein Zehntel der Spannweite umfaßt, erhält man folgende Dichtespur:



Man erkennt sehr schön, wie $\hat{f}$ sein Maximum nahe bei 0 hat und dann allmählich nach rechts abfällt.

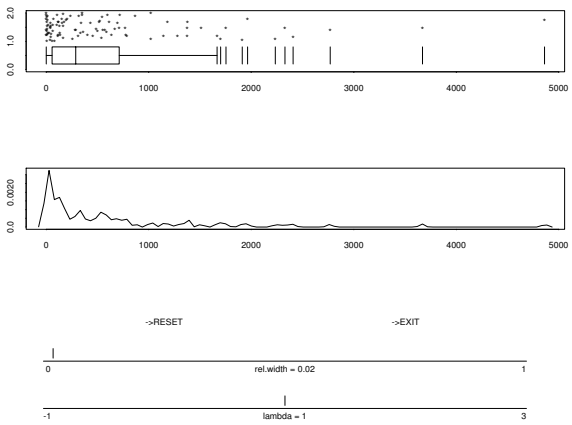
Konzept: Eine Verkleinerung der Fensterbreite führt zu einem zitterigen Bild, eine vergrößerte Breite verstärkt die Glättung. Um ein angemessenes `w` zu finden, muß man in der Regel Versuche mit verschiedenen Fensterbreiten anstellen.

Praxis: Deshalb erhalten Sie nun die Möglichkeit, die Fensterbreite beliebig zu variieren.

Technik: Dabei müssen die Fensterbreiten als Anteil der Spannweite fixiert werden. Zusätzlich gestattet das Instrument die Wirkung verschiedener Box-Cox-Transformationen zu studieren.

35 ⟨*aktiviere interaktives Dichtespur-Sektion zu* `x` 35⟩≡
```
dichtespur.interaktiv(x)
```

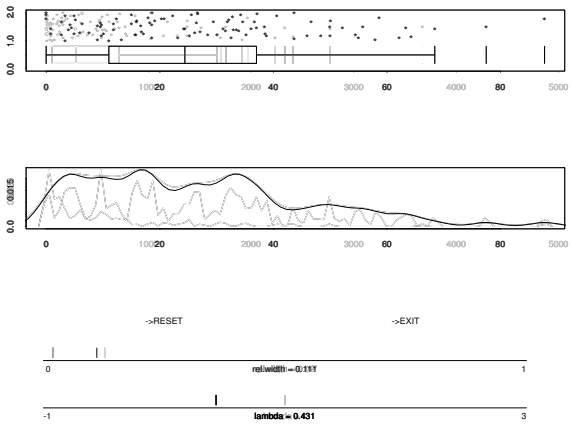
Technik: Das Modul erzeugt eine dreigeteilte Graphik. Im oberen Drittel sieht man einen Jitterplot zusammen mit einem Boxplot der transformierten Daten. In der Mitte folgt die Dichtespur. Sie hängt von der gewählten Transformation und der gewählten Fensterbreite ab. Am Anfang sieht das Bild so aus (Zu sehen ist hier das Erscheinungsbild. Am Bildschirm sind die Details deutlicher zu erkennen als im Druck. Besonders dann, wenn man das graphische Fenster vergrößert (−> Vollbild).):



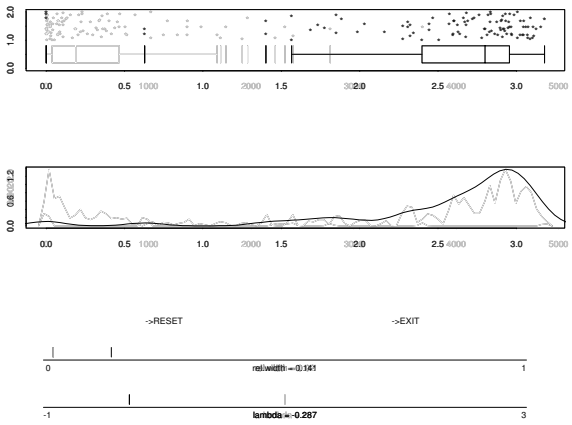
Zum Verlassen dieses kleinen Programms muß man mit der Maus EXIT anklicken (Mauszeiger auf EXIT bewegen und linke Maustaste drücken). Eine andere Fensterbreite kann durch einen Mausklick direkt über der Linie, unter der `rel.width` steht, gewählt werden. Ganz links ist die Fensterbreite 0 zugeordnet, ganz rechts 100 % der Spanneweite. Mit dem zweiten Schieberegler (unterste horizontale Linie) läßt sich ein λ für die Box-Cox-Transformation wählen, mit der der ursprüngliche Datensatz `x` transformiert werden sollen.

Konzept: Ziel des interaktiven Moduls ist es, ein Gespür für die Auswirkung der Fensterbreite und des Transformationsparameters zu bekommen. Im Rahmen einer Datenanalyse sind übrigens Parameter erstrebenswert, für die eine Interpretation gefunden werden kann. Dieses bedeutet, daß dort zu krumme λ-Werte auf jeden Fall unbrauchbar sind. Daneben gibt es auch Einsatzfelder für Box-Cox-Transformationen, bei denen es nicht unbedingt auf eine Interpretation von λ ankommt. Dann versucht man manchmal, den Wert zu finden, der zu den besten transformierten Daten gemäß eines vorgegebenen Kriteriums führt.

Praxis: Nach einigen Modifikation kann sich als Resultat ($\lambda = 0.431$, $w = 0.111$)



oder ($\lambda = 0.287, w = 0.141$)



ergeben.

Konzept: Das Histogramm gibt Antworten darüber, wie groß die relativen Häufigkeiten pro Einheit der x-Achse sind. Es beantwortet nicht Fragen, die sich auf beliebige Intervalle beziehen. Solche Fragen lassen sich mit Hilfe der empirischen Verteilungsfunktion beantworten.

4.7 Die empirische Verteilungsfunktion

Konzept: Das Histogramm stellt einen Versuch dar, die unterschiedlichen Klassenhäufigkeiten explizit darzustellen. Damit bekommt man einen Eindruck, in welchen Bereichen mehr Beobachtungen anzutreffen sind und in welchen Bereichen weniger. Es lassen sich aber nicht direkt Fragen der folgenden Art beantworten.

- Wieviel Prozent der Beobachtungen liegen unter x_p, über x_p oder zwischen x_1 und x_2?
- In welchem Bereich liegen die $p \times 100$ % kleinsten oder größten Beobachtungen?

Die Fragen beziehen sich offensichtlich auf Flächen, die sich über Teile des Histogramms finden ließen. Direkt lassen sich Antworten mit dem Histogramm aber nicht geben. Günstiger ist es, die empirische Verteilungsfunktion zu heranzuziehen. Sie zeigt zu jedem Wert x die relative Anzahl $\hat{F}(x)$ der Beobachtungen an, die nicht größer als x sind.

Algorithmus: Empirische Verteilungsfunktion $\hat{F}(x)$.

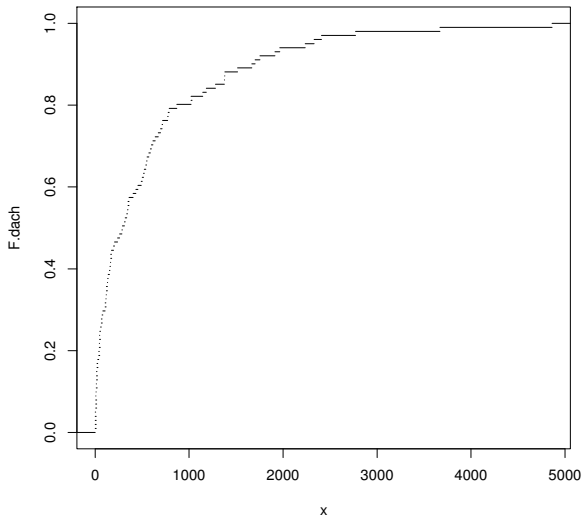
$$\hat{F}(x) = \frac{\textit{Anzahl der Beobachtungen kleiner gleich } x}{\textit{Anzahl der Beobachtungen}}$$

Verteilungsfunktionen sind für den Statistiker besonders statistische Instrumente. An dieser Stelle wollen wir aber keine tiefere Diskussion anstellen.

Praxis:

36 ⟨*erstelle F.dach von* x 36⟩≡
```
F.dach(x)
```

Wir bekommen folgendes Bild aufgrund der Daten `x`.



Konzept: Geht man von klassierten Daten oder von der Information aus, die in einem Histogramm enthalten ist, so werden die ursprünglichen Beobachtungen innerhalb jeder Klasse als gleichverteilt angenommen. Dieses führt zu abschnittsweise linearen Verläufen.

Algorithmus: Empirische Verteilungsfunktion $\hat{F}(x)$ für klassierte Daten.

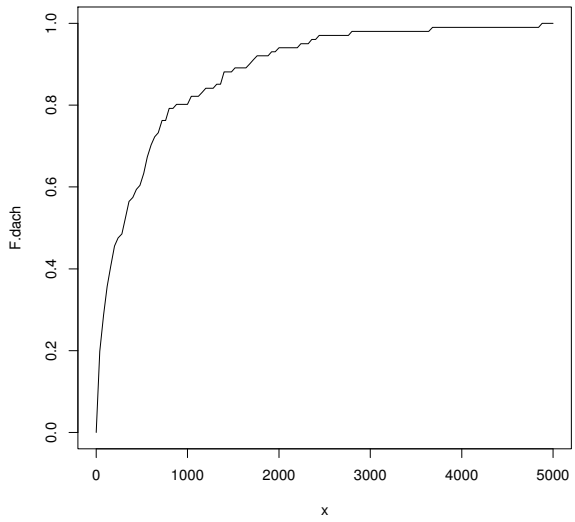
$$\hat{F}(x) = \begin{cases} 0 & x \leq \tilde{x}_0 \\ \hat{F}(\tilde{x}_{i-1}) + (x - \tilde{x}_{i-1}) n_i / (\Delta x_i n) & \tilde{x}_{i-1} < x \leq \tilde{x}_i \\ 1 & \tilde{x}_k < x \end{cases}$$

$\tilde{x}_k$ ist die größte Klassenobergrenze und $\hat{F}(\tilde{x}_i)$ ist definiert durch $\sum_{j=1}^{i} n_j/n$. n_j/n ist die relative Häufigkeit von Beobachtungen in der j-ten Klasse.

37 ⟨*erstelle F.dach für klassierte Daten zu* `x` 37⟩≡

```
F.dach(x,breaks=seq(from=0,to=5000,by=40))
```

Praxis: Die Klassengrenzen 0, 40, 80, ..., 5000 führen zu folgendem Bild:

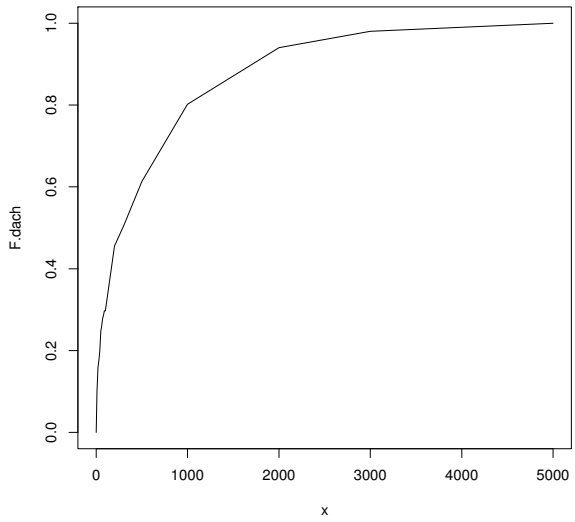


Mit Hilfe dieser Graphiken lassen sich die Fragen nach Anteilen und Häufigkeiten von Bereichen beantworten. Man sieht, bei zirka 300 wird der Datensatz halbiert ($\hat{F}(300) \approx 0.5$). 20 % aller Zwischenzeiten liegen unterhalb von 50 Sekunden (sehen ist hier etwas übertrieben). Hiervon ausgehend dürfte ein Experte für den Serverbetrieb bei wachsender Belastung Engpässe erwarten.
Wir wollen die Möglichkeit der selbständigen Wahl der Grenzen anbieten:

38 ⟨*erstelle emp. Verteilungsfunktion mit manueller Wahl der Grenzen zu* x 38⟩≡

```
print("Grenzen nach einander eingeben")
print("Nach Eingabe der letzten Klasse noch einmal: -> ENTER")
F.dach(x,breaks=sort(scan(,0)))
```

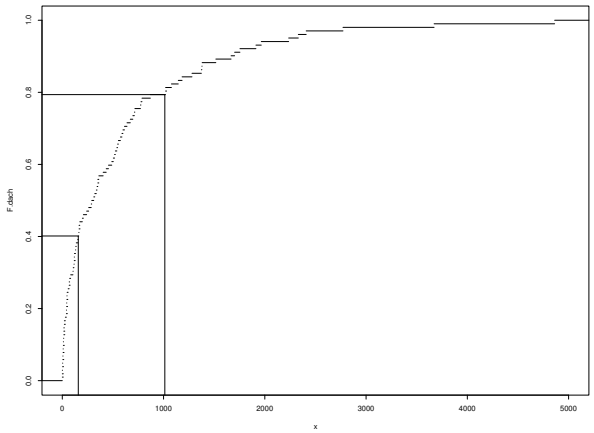
Das folgende Bild erhält man für die Eingabe der Grenzen: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 200, 300, 500, 1000, 2000, 3000, 5000



Für genauere Abschätzungen soll der Zugriff auf die Verteilungsfunktion auch graphisch unterstützt werden. Dazu kann man mit der Maus zwei Punkte im Verteilungsfunktionsplot anklicken. Probieren geht hier über Studieren!

39 ⟨*befrage F.dach zu* x *graphisch* 39⟩≡
```
F.dach(x)
F.dach.look.up(x)
```

Ein mögliches Bild könnte folgendes sein:



Dies führt zu dem numerischen Ergebnis:

```
[1] "bitte zweimal Punkt im F.dach-Plot anklicken!"
  x1 F.dach(x1) 1-F.dach(x1)       x2 F.dach(x2) 1-F.dach(x2)
 158  0.4019608    0.5980392 1011.577  0.7941176    0.2058824
 F.dach(x2)-F.dach(x1)
             0.3921569
```

Für Zugriffsfragen mit Hilfe der Verteilungsfunktion sei hier an den Satz 16 aus dem Kapitel 2 des Skriptes [Autorenkollektiv] erinnert.

4.8 Offene Fragen

Konzept: Die graphischen Werkzeuge erlauben es, zusammenfassende Statistiken wie auch die Einzeldaten zu visualisieren, ausreißerverdächtige Beobachtungen zu erkennen und Häufigkeiten einzelner Ausprägungen sowie Häufigkeiten der Realisationen in bestimmten Intervallen festzustellen.

Praxis: Für die Serverproblematik wurde herausgefunden, daß kleine Zwischenzeiten zwischen aufeinanderfolgenden Zugriffen sehr häufig sind und damit für die Belastungseinschätzung eine erhebliche Bedeutung haben. Weitere Belastungsaspekte werden im folgenden Aufgabenabschnitt angesprochen. Jedoch lassen sich folgende Fragen noch nicht beantworten:

- Weichen Belastungen an verschiedenen Tagen voneinander ab oder folgen sie denselben Gesetzmäßigkeiten?
- Unterscheidet sich ein spezieller Tag von einem Standardtag?
- Unter welchen Umständen ist mit einer hohen Belastung zu rechnen?
- Wie läßt sich die Belastung modellieren?

Die Fragen zeigen, daß es für die nächsten Kapitel noch einiges zu tun gibt.

4.9 Aufgaben

Praxis: Die Aufgaben sollen mit den Operationen des folgenden Angebotes bearbeitet werden.

40 ⟨*1⟩+≡
```
cat("Revbook:2.m:menu.gr:Menue Graphische Beschreibungstechniken\n")
if(exists("auswahl.graphiken")) auswahl.graphiken()
```

Zur Information wird das Auswahlmenü gezeigt:

```
items:
  lege Daten auf x ab
  bilde Ausschnitt der Daten
  erstelle graphische 5-Zahlen-Zusammenfassung
  erstelle Boxplot
  erstelle Jitterplot
  erstelle Stabdiagramm
  erstelle Haeufigkeitstabelle
  erstelle Stem-and-Leaf-Plot
  erstelle Histogramm
  erstelle Histogramm mit manuellen Grenzen
  erstelle Dichtespur
  aktiviere interaktives Dichtespur-Sektion
  erstelle F.dach
  erstelle F.dach mit manuellen Grenzen
  befrage F.dach graphisch
  transformiere mittels Box-Cox-Transformation
```

a) Erstellen Sie zu den Mengen der am 17.02.97 übertragenen Daten verschiedene Graphiken. Beschreiben Sie den Datensatz verbal aufgrund Ihrer Beobachtungen.

b) Erstellen Sie zu den logarithmierten Mengen der am 17.02.97 übertragenen Daten verschiedene Graphiken. Hat die Transformation Ihrer Meinung nach einen symmetrischen Datensatz hervorgebracht?

c) Transformieren Sie den Datensatz so, daß ein möglichst symmetrisches Ergebnis entsteht.

d) Stellen Sie Vermutungen über die Mengen vom 3.02.97 an. Schreiben Sie diese nieder. Überprüfen Sie diese dann mit Hilfe der graphischen Werkzeuge.

e) Was vermuten Sie, wie die Verteilung der Zugriffs-Zeitpunkte vom 17.2.97 aussehen wird. Begründen Sie Ihre Vermutung. Erstellen Sie dann von diesen einen Jitterplot, ein Stabdiagramm und ein Histogramm.

f) Zeigen die Zeitpunkte vom 3.02.97 andere Eigenschaften? Wann ist die Belastung am größten: nachts, morgens, mittags, nachmittags oder abends? Suchen Sie Gründe für die Gestalt der Verteilung der Zugriffszeitpunkte.

g) Besitzen die Zeitdifferenzen vom 3.02.97 unerwartete Eigenschaften?

h) *Menge pro Zeitdifferenz* könnte auch ein Belastungsmaß sein. Experimentieren Sie mit diesem!

5 Multivariate Beschreibungstechniken

41 ⟨*1⟩+≡
```
cat("Revbook:3.a:start.mu:Kapitel Multivariate Beschreibungstechniken\n")
```

Zum Literaturstudium sei hingewiesen auf: [Bamberg, Baur: Kapitel über Auswertungsmethoden für mehrdimensionales Datenmaterial], [Chambers et al., Kapitel: Comparing Data Distributions, Studying Two-Dimensional Data, Plotting Mulitvariate Data], [Schlittgen: Kapitel über Darstellung multivariater Datensätze].

Konzept: In den ersten Kapiteln sind nur eindimensionale Datensätze Gegenstand der Betrachtung gewesen. Interessante Probleme führen jedoch meistens zu mehrdimensionalen Fragestellungen, so daß die multivariate Statistik eine äußerst große Bedeutung besitzt. Als Voraussetzung für angemessene multivariate Analysen müssen die wesentlichen Konzepte der eindimensionalen Statistik verstanden worden sein. Deshalb beschäftigen sich die ersten Diskussionen in einer statistischen Grundausbildung hauptsächlich mit eindimensionalen Problemen. Bevor wir die einfachen Beschreibungstechniken verlassen, wollen wir jedoch einen kurzen Einblick in *Multivariate Beschreibungstechniken* geben.

Praxis: Auch in unserem Problemfeld erkennt man multivariate Fragestellungen:

1. Sind Belastungen verschiedener Tage als sehr ähnlich anzusehen oder unterscheiden sie sich deutlich?
2. Ist die Belastung innerhalb eines Tages gleichmäßig verteilt oder verändert sie sich im Zeitablauf?
3. Besteht ein Zusammenhang zwischen den Zwischenzeiten und den übertragenen Mengen?

Konzept: Die Situation zur Frage 1 führt zu dem Vergleich (zweier) inhaltlich vergleichbarer aber als unabhängig anzunehmender Datensätze. Diese Frage gehört in die Klasse: *Das Zweistichprobenproblem für unabhängige Stichproben*. Die einzelnen Beobachtungen der beiden Datensätze stehen in keiner Beziehung zueinander.
Demgegenüber sind in der Situation der Frage 3 die Daten der Merkmale *Zwischenzeiten* und *übertragene Mengen* über Zugriffsereignisse miteinander verbunden. Man kann sich vorstellen, daß die Beobachtungen zur Frage 3 in Form von Paaren (Wartezeit, nach der Wartezeit übertragene Menge) vorliegen. Der Statistiker vergleicht in der Situation der ersten Frage Statistiken der einzelnen Datensätze und die Gestalten der empirischen Verteilungen anhand von geeigneten Graphiken. In der verbundenen Situation wird zunächst die Vermutung der Unabhängigkeit untersucht. Eine Ablehnung der Vermutung führt dann zur Beschreibung der Abhängigkeiten. (Für Interessierte sei an dieser Stelle auf die Kapitel über Regressionsanalyse in der Einführungsliteratur hingewiesen.)
Die Klassifikation der zweiten Frage hängt von ihrer Umsetzung ab. Eine Möglichkeit besteht darin, die Zwischenzeiten eines Tages bezüglich der Anzahl von Ereignissen in gleich große Teile zu zerlegen und die Teilstichproben zu vergleichen. Die laufende Nummer kann dann als Bindungsmerkmal interpretiert werden. Dann lassen sich die jeweils i-ten Beobachtungen der einzelnen Teile gegenüberstellen. Werden die Belastungsdaten jedoch in gleich große Zeitklassen — wie: nachts, morgens, nachmittags und abends — eingeteilt, so ergeben sich neue, kleinere Datensätze, deren Beobachtungen als unverbunden angesehen werden müssen. Als drittes bietet es sich an, die Zugriffszeitpunkte einer Inspektion zu unterziehen; dieser Weg wird zu eindimensionalen Techniken zurückführen.
Anhand der Fragen wird der Leser in diesem Kapitel mit Techniken vertraut gemacht, die zur Beantwortung der gestellten Fragen hilfreich sind. Zu diesen gehören: vergleichende Boxplots, Vergleichsplot der empirischen Verteilungsfunktionen, Scatterplots mit konvexen Hüllen, Draftsman's Displays. Zusätzlich wird die Idee von Bootstrap-Verfahren vorgestellt.

Technik: Datensätze, die in diesem Kapitel als Input dienen, werden auf der Variablen `xy` abgelegt. An dem Variablennamen soll auch optisch sichtbar werden, daß die nicht eindimensional sind. Für R-Kenner sei darauf hingewiesen, daß `xy` ein Listenobjekt ist.

5.1 Vergleichende Boxplots

Praxis: Betrachten wir die erste Frage: Sind Belastungen verschiedener Tage als sehr ähnlich anzusehen oder unterscheiden sie sich deutlich? Das Merkmal *Belastungen* muß zunächst mit Inhalt gefüllt werden. Wir wollen für die Belastungen wieder die Zwischenzeiten heranziehen. Da uns die Zwischenzeiten vom 03.02.97 und vom 17.02.97 vorliegen, werden wir mit diesen unser Glück versuchen.

Technik: Die Datensätze, die zu den beiden Tagen gehören, kombinieren wir zu dem Objekt `xy`.

42 ⟨*lege Zwischenzeiten vom 03.02.97 und vom 17.02.97 auf* `xy` *ab* 42⟩≡
```
xy<-list(dz.03.02.97=dzeitpunkte.03.02.97,
         dz.17.02.97=dzeitpunkte.17.02.97)
```

Konzept: Wenn die Vergleichs-Frage zwischen den Tagen nur auf die Lage der Datensätze abzielt, können wir für beide Datensätze die Mediane oder Mittelwerte berechnen und vergleichen. Interessiert daneben auch die Gestalt der Verteilung, empfehlen sich graphische Darstellungen.

Praxis: Da keine weiteren Hinweise zu den beiden Referenztagen vorliegen, vermuten wir sicher eine ähnliche Lage, eine ähnliche Variabilität und eine ähnliche Gestalt. Berechnen wir zunächst die zusammenfassenden Statistiken.

43 ⟨*berechne zusammenfassende Statistiken zu* `xy` 43⟩≡
```
summary.stats(xy)
```

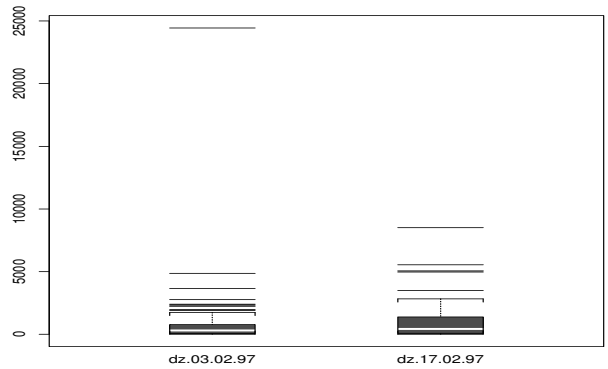
```
$dz.03.02.97:
 Min. 1st Qu. Median  Mean 3rd Qu.  Max. Std.Dev.   n
    1    62.5    312 823.8   744.5 24450   2487.4 103

$dz.17.02.97:
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.  n
    1      85    428 970.9    1356 8513   1415.5 87
```

Die Quartile und das Mittel vom 17.02.97 sind größer als die entsprechenden Größen vom 03.02.97. Dagegen gehört das Gesamtmaximum zu den Daten vom 03.02.97. Damit deuten sich Unterschiede zwischen den Tagen an.

Konzept: Vergleichende Boxplots entstehen dadurch, daß mehrere Boxplots mit einer gemeinsamen Achse nebeneinander gezeichnet werden. Die einzelnen Boxplots zeigen sowohl einige Statistiken (Welche sind das noch? Wer sich nicht mehr genau erinnert, sollte unbedingt zurückblättern.) wie auch die grobe Gestalt der Verteilungen an.

44 ⟨*erstelle Boxplot zu* `xy` 44⟩≡
```
boxplot(xy)
```

Praxis: Plot Output:



Wieder einmal wird der Gesamteindruck erheblich von den möglichen Ausreißern beherrscht. Deshalb wollen wir die Werte oberhalb von 5000 entfernen und den Plot wiederholen.

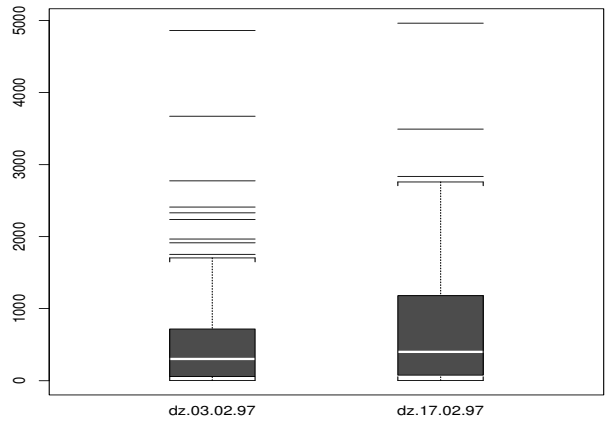
45 ⟨*bilde Ausschnitte der Komponenten von* `xy` 45⟩≡
```
xy<-remove.extreme.values(xy)
```

```
[1] "Komponente:  dz.03.02.97"
[1] "Die Werte der 1 -ten Komponente liegen in"
[1]     1 24449
[1] "Bis zu welcher Untergrenze sollen Werte von der"
[1] "1 -ten Komponente entfernt werden?"
1: 0
[1] "Ab welcher Obergrenze sollen Werte von der"
[1] "1 -ten Komponente entfernt werden?"
1: 5000
[1] "Komponente:  dz.17.02.97"
[1] "Die Werte der 2 -ten Komponente liegen in"
[1]    1 8513
[1] "Bis zu welcher Untergrenze sollen Werte von der"
[1] "2 -ten Komponente entfernt werden?"
1: 0
[1] "Ab welcher Obergrenze sollen Werte von der"
[1] "2 -ten Komponente entfernt werden?"
1: 5000
```

Die so definierten Ausschnitte führen mit dem Modul ...

46 ⟨* 1⟩+≡
```
boxplot(xy)
```

... zu den Boxplots:



Jetzt lassen sich die Verteilungen schon besser vergleichen. Die Mediane und der Bereich unterhalb des Medians sind nicht sehr unterschiedlich. Der Teil der Verteilungen zwischen Median und drittem Quartil ist für den 17.02.97 stärker auseinandergezogen. Die stark unterschiedliche Lage der dritten Quartile konnte natürlich schon den zusammenfassenden Statistiken entnommen werden.

5.2 Histogramme

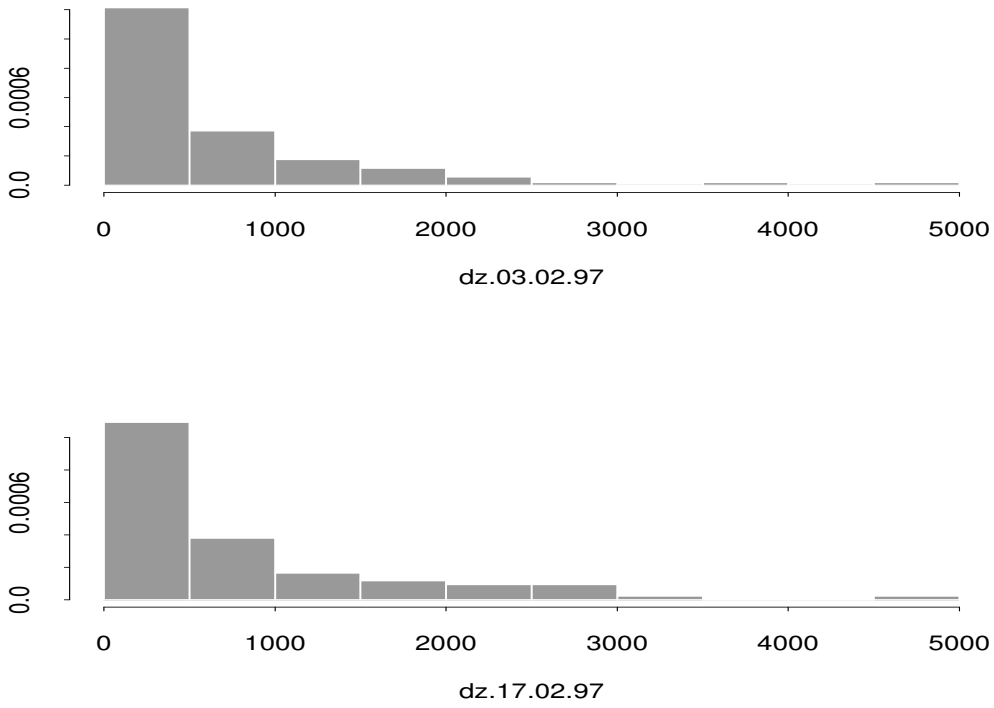
Praxis: Zur Unterstützung wollen wir die Histogramme betrachten.

Technik: Um die Vergleichbarkeit zu erhöhen, werden sich die horizontalen Achsen entsprechen. Die Anzahl der gewünschten Klassen muß eingegeben werden.

47 ⟨*zeichne Histogramme zu* `xy` 47⟩≡

```
print("Wie viele Klassen sind sollen gebildet werden?")
nclass<-c(scan(,0,n=1),7)[1]
histogramm(xy,nclass=nclass)
```

Für zehn Klassen erhält man folgendes Bild.



Unterschiede sind wieder nur im rechten Bereich der Verteilungen zu erkennen. Für den Vergleich bestimmter Intervalle helfen die Histogramme nicht sehr weiter.

5.3 Verteilungsfunktionen und Dichtespuren

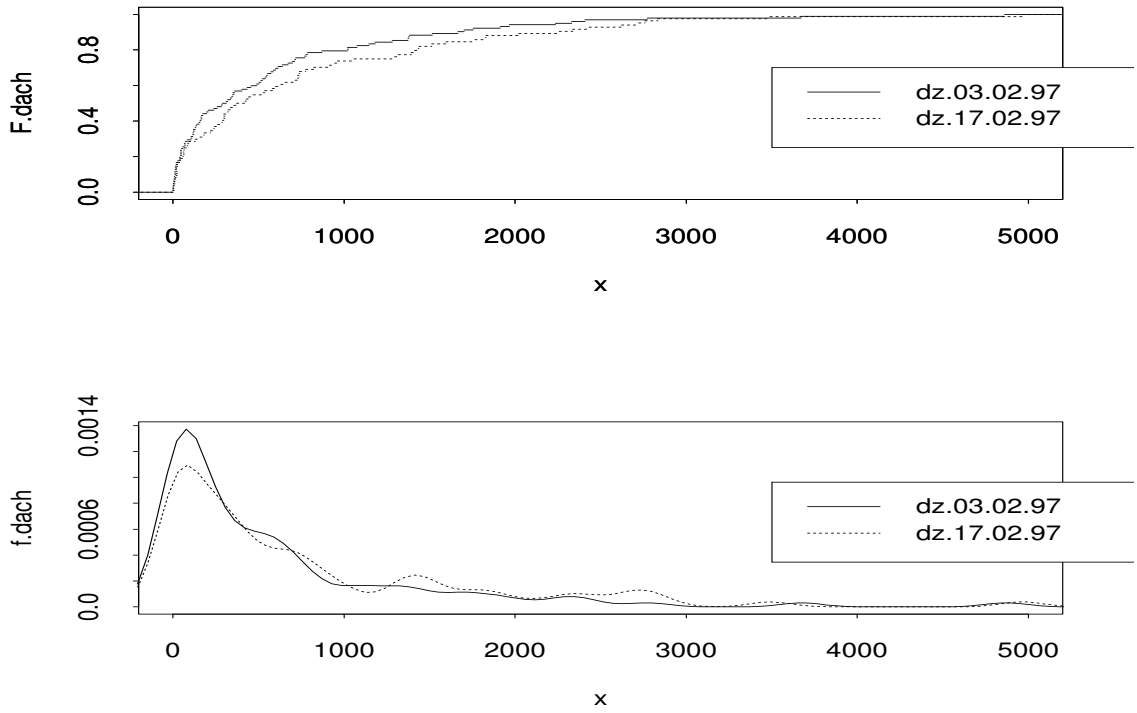
Hier können vielleicht die in einem Plot eingetragenen empirischen Verteilungsfunktionen oder Dichtespuren mehr leisten.

Technik: Für die Dichtespur ist die Fensterbreite anzugeben.

48 ⟨*zeichne zu* `xy` *F.dach und f.dach* 48⟩≡

```
cat("F.dach und f.dach zu xy[1:2]\n")
opar<-par(no.readonly=T); par(mfrow=2:1)
F.dach(xy)
dichtespur(xy)
par(opar)
```

Praxis: Für die Eingabe 500 als Fensterbreite erhalten wir die folgende Graphiken.



Das Bild hat nicht viel mehr Aussagekraft als die vergleichenden Boxplots. Für die Daten vom 03.02.97 (durchgezogene Linie) werden im Intervall von 0 bis zirka 300 jedoch erhöhte Häufigkeiten ausgewiesen. Die lokalen Maxima sollten nicht überinterpretiert werden.

5.4 Der QQ-Plot

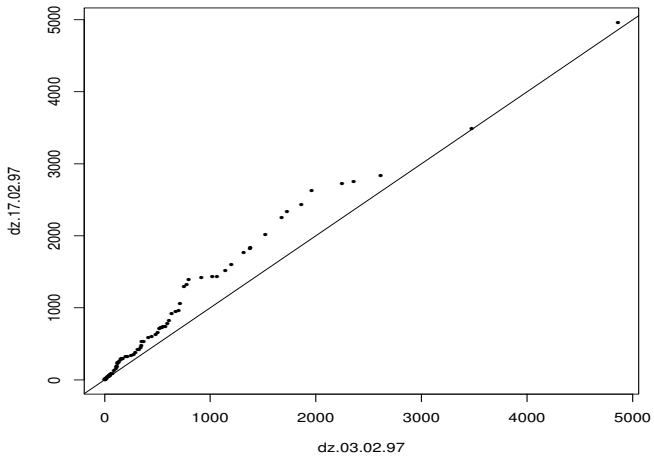
Konzept: Als letztes Werkzeug zu der Frage des Vergleichs zwischen Tagen wird noch der *QQ-Plot* vorgestellt. Mit diesem Plot wird versucht, zentrale Informationen zweier empirischer Verteilungsfunktionen in einer Darstellung zusammenzufassen. Dazu werden die zugehörigen zwei Prozentpunkte zu vorgegebenen Funktionswerten der empirischen Verteilungsfunktionen gegeneinander abgetragen. Das bedeutet für zwei gleich große Datensätze, daß die kleinste Beobachtung des einen gegen die kleinste Beobachtung des anderen, die zweitkleinste Beobachtung des einen gegen die zweitkleinste Beobachtung des anderen usw. abgetragen wird. Bei verschieden großen Umfängen kommen Interpolationsverfahren zum Einsatz. Sind die beiden Stichproben (fast) identisch, ergeben sich Punktepaare, die (fast) auf der Diametralen ($f(x) = x$) liegen.

Algorithmus: — QQ-Plot zu $\mathtt{x} = (x_1, \ldots, x_n)$ *und* $\mathtt{y} = (y_1, \ldots, y_m)$

In einem QQ-Plot werden zu ausgewählten Werten p *die Quantile* y_p *gegen die Quantile* x_p *abgetragen.*

49 ⟨*erstelle zu den ersten beiden Elementen von* `xy` *einen QQ-Plot* 49⟩≡

```
qqplot(xy[[1]],xy[[2]],xlab=getnames(xy)[1],ylab=getnames(xy)[2])
abline(0,1)
```



Praxis: Deutlich weichen die Punkte von der Diametralen ab. Dies kann als ein Indiz für eine unterschiedliche Verteilung angesehen werden.

Konzept: Liegen die Punkte des QQ-Plots auf einer zur Diametralen parallelen Geraden, so weist dies auf eine Lage-Verschiebung hin. Eine von Eins verschiedene Steigung zeigt, daß die relativen Häufigkeiten in den zugehörigen Abschnitten unterschiedlich sind (Dies kann zum Beispiel an unterschiedlichen Skalierungsfaktoren liegen — Beispiel: ein ähnlicher Sachverhalt ist einmal in cm und einmal in inch gemessen worden.).

Praxis: Hier sehen wir, daß der Datensatz vom 03.02.97 (bis zirka 800) mehr kleinere Beobachtungen besitzt, denn die Punkte entfernen sich von (0, 0) ausgehend zunächst von der Diametralen nach oben. Dann verlaufen die Punkte bis zirka 2000 fast auf einer Parallelen zur Referenzlinie. Der rechte Schwanz des Datensatzes vom 03.02.97 ist also im Vergleich zu dem vom 17.02.97 etwas in Richtung kleinerer Werte verschoben. Die größten Werte liegen wieder gut auf der gezeichneten Geraden.
Damit sind Unterschiede erkennbar geworden. Es ist jedoch noch nicht klar, ob man sie als bemerkenswert einstufen darf oder ob sie sich auch aufgrund von Zufalls-Schwankungen einstellen würden.

5.5 Bootstrap-Verfahren

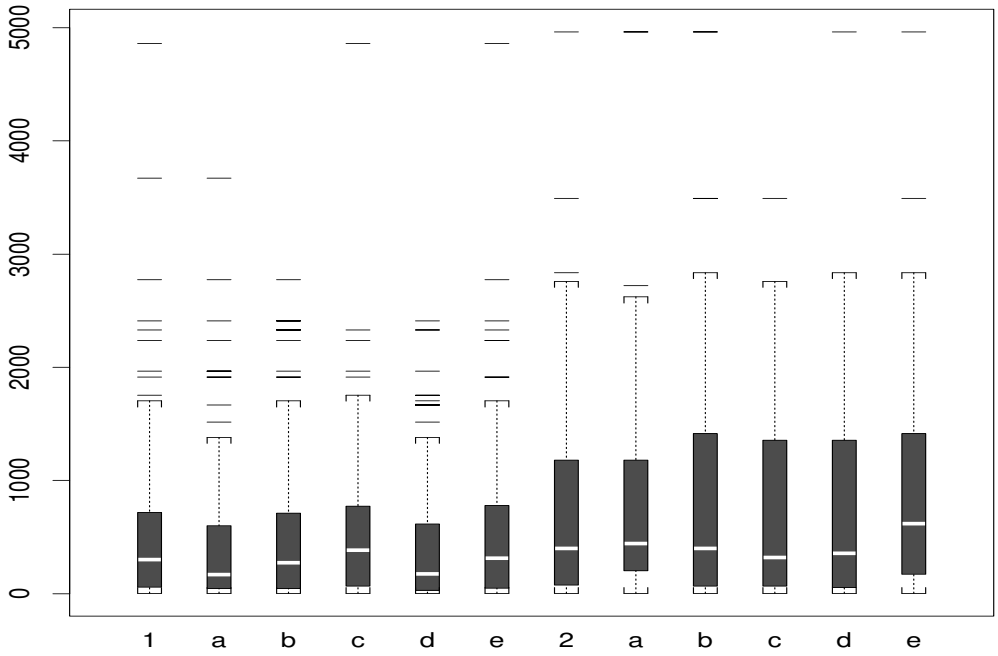
Konzept: Bevor wir uns der nächsten der drei oben gestellten Fragen zuwenden, wollen wir herausfinden, in welchem Umfang der Zufall Schwankungen, wie im letzten Absatz erwähnt, hervorrufen kann. Dazu wollen wir ein sogenanntes Bootstrap-Experiment durchführen.

Praxis: Das Experiment nimmt die auf `xy` abgelegten Datensätze als Grundgesamtheiten an. Dann werden aus ihnen jeweils `n`-mal Stichproben nach dem Prinzip *Ziehen mit Zurücklegen* gezogen. Jeder ursprüngliche Datensatz und jede zugehörige Stichprobe wird in Form eines Boxplots dargestellt.

Technik: Die Anzahl `n` ist anzugeben. Für das Experiment ist außerdem ein Startpunkt für den eingebauten Zufallszahlengenerator in Form einer ganzen Zahl zwischen 0 und 1000 einzugeben. Verschiedene Zufallsstarts führen zu verschiedenen Ergebnissen.

50 ⟨*zeige Bootstrap-Stichprobenverteilungen mit* `xy` 50⟩≡
```
bootstrap.experiment(xy)
```

Praxis: Mit den Setzungen `n=5` und `ZZ=13` erhält man



Die ursprünglichen Datensätze werden durch Zahlen markiert, die Boxplots der zugehörigen Bootstrap-Stichproben folgen jeweils rechts daneben und sind durch kleine Buchstaben gekennzeichnet. Alle dritten Quartile der Wiederholungen zu dem zweiten Datensatz `2` (das sieht man an den fünf rechten Boxplots) sind größer als diejenigen des ersten Datensatzes. Für die Mediane ist eine solch klare Aussage nicht zu treffen.

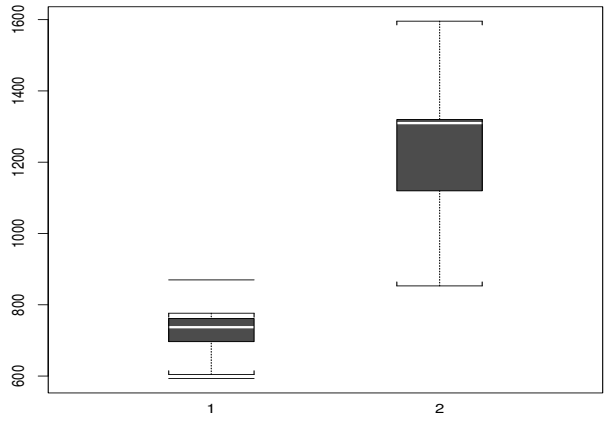
Konzept: Verfahren, bei denen wie im durchgeführten Experiment aus einem gegebenen Datensatz neue Stichproben gezogen werden, heißen *Bootstrap*-Verfahren. Mit ihnen lassen sich Variabilitäten von Stichproben und aus Stichproben berechneten Größen (Statistiken) studieren.

Praxis: Um letzteres beispielhaft zu demonstrieren, können mit dem folgenden Modul zu jedem Datensatz Boxplots erstellt werden, die die Realisationen einer gewählten Statistik in den einzelnen Bootstrap-Stichproben zusammenfassend darstellen.

Technik: In dem folgenden Experimentier-Modul eine Statistik ausgewählt werden. Als Alternativen werden angeboten: `median`, `mean`,`3. Quartil`,`1. Quartil`, `s`,`var`, `Interquartils-Abstand` eine ausgewählt werden.

51 ⟨*zeige Verteilung einer Statistik von Bootstrap-Stichproben aus* `xy` 51⟩≡
```
bootstrap.experiment(xy,type="?")
```

Praxis: Für `n=10` und die Wahl `3. Quartil` erhält man das überzeugende Ergebnis:



Die dritten Quartile der Stichproben zu dem Datensatz vom 17.02.97 sind alle größer als die der anderen Stichproben. Damit erscheint ein zufälliger Unterschied eher unwahrscheinlich.

Konzept: Übrigens sei erwähnt, daß man unter dem Stichwort Zwei-Stichproben-Problem für unabhängige Stichproben in der Regel zur Abteilung *Testen* geschickt wird. Dort findet man dann Testprozeduren, die zum Beispiel unter bestimmten Annahmen die Unterschiedlichkeit der Mittel der zugehörigen Grundgesamtheiten oder ihrer Verteilungsfunktionen mit einer bestimmten Sicherheit herausfinden.

Praxis: Bevor wir die Auseinandersetzung mit dem Tagesvergleich verlassen, sei darauf hingewiesen, daß die Summe der Zeitdifferenzen ungefähr 24 Stunden beträgt. Damit hängt das Mittel der Zeitdifferenzen fast nur noch von der Anzahl der gezählten Ereignisse ab.

5.6 Der Scatterplot

Mit der dritten Frage soll der Zusammenhang zwischen den Merkmalen *Zwischenzeit* und *übertragene Menge* geklärt werden: Besteht ein Zusammenhang zwischen den Zwischenzeiten und den übertragenen Mengen? Die Beobachtungen können — wie oben bereits erwähnt — in Form von Beobachtungspaaren angeordnet werden.

Technik: Wir wollen diese Paare wieder auf `xy` ablegen. (Bei genauerer Überlegung könnte man zu dem Ergebnis kommen, daß die Anzahl der Zwischenzeiten um eins größer sein müßte als die Anzahl der Zwischenzeiten. Jedoch wurde als erste Wartezeit die Zeit von Tagesbeginn bis zum ersten Ereignis ergänzt.)

52 ⟨*lege Zwischenzeiten und Mengen vom 03.02.97 auf* `xy` *ab* 52⟩≡

```
xy<-cbind(dz.03.02.97=dzeitpunkte.03.02.97,m.03.02.97=mengen.03.02.97)
xy<-split(xy,col(xy))
```

Konzept: Das naheliegendste Werkzeug für die Darstellung der beiden verbundenen Merkmale ist der Scatterplot (Streudiagramm).

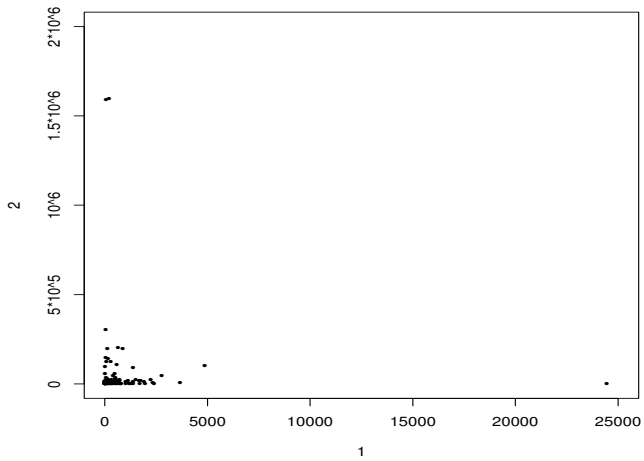
> *Algorithmus: — Scatterplot eines zweidimensionalen Datensatzes.*
>
> *Im Scatterplot werden die Komponenten der Beobachtungspaare gegeneinander geplottet.*

In einem Scatterplot läßt sich erkennen, ob die Daten eher wie zufällig verteilt sind oder ob bestimmte Strukturen beziehungsweise Beziehungen zwischen den Merkmalen zu erkennen sind. So können sich die Punkte an bestimmten Stellen häufen oder sie können um eine Linie herum streuen. Strukturen irgendwelcher Form sprechen gegen eine Zufälligkeitsvermutung. In Fällen, in den es gelingt, aufgrund der Ausprägung des einen Merkmals Schlüsse für die Realisation des anderen zu ziehen, werden die Merkmale als nicht unabhängig, d.h. abhängig bezeichnet. Dann versucht man als nächstes, die Abhängigkeitsstrukturen zu beschreiben. Bei der Gegenüberstellung der Merkmale *Körpergröße* und *Gewicht* von Studierenden würde man zum Beispiel deutliche Abhängigkeitsstrukturen erkennen können.

Praxis: Für eine Abhängigkeit der Merkmale *Zwischenzeit* und *Menge* lassen sich zwar nicht sofort Begründungen finden, doch würde ein negativer Zusammenhang für die effektive Serverbelastung von erheblicher Bedeutung sein. Deshalb wollen wir einen Blick auf die Daten werfen und dazu einen Scatterplot erstellen.

53 ⟨*erstelle Scatterplot zu den ersten beiden Elementen von* `xy` 53⟩≡

```
scatter(xy)
```



Praxis: Leider wird das Bild wieder durch die wenigen sehr großen Werte beherrscht, so daß die meisten Punkte sich in der linken unteren Ecke ballen. Wir wollen deshalb alle x- und y-Werte logarithmieren.

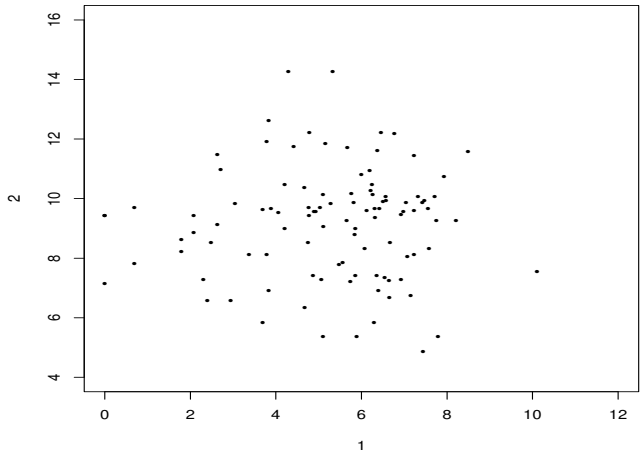
Konzept: Solche Veränderungsoperationen heißen in der Statistik *Transformationen*. Sie dienen dazu, bestimmte Eigenschaften von Datensätzen stärker hervortreten zu lassen, oder zur Aufbereitung des Datenmaterials, um Bedingungen statistischer Verfahren, die man gerne einsetzen möchte, zu erfüllen. Durch Logarithmierung werden extrem große Werte näher an die übrigen Punkte herangeholt und eine Punktewolke kleiner Beobachtungen auseinandergezogen. Abhängigkeitsstrukturen, die im letzten Plot verdeckt waren, könnten hierdurch sichtbar werden.

Technik: Technisch realisieren wir die Logarithmierung durch entsprechende Box-Cox-Transformationen. Das folgende Modul fragt die Transformationsparameter λ ab. Für weitere Erklärungen blättere man zurück zum Ende des Kapitels über Maßzahlen.

54 ⟨*transformiere Elemente von* `xy` *mittels Box-Cox-Transformation* 54⟩≡
```
xy<-box.cox.transformation(xy)
```

Durch zweimalige Eingabe von "`0`" logarithmiert das Modul die beiden Datensätze. Nun kann der Scatterplot wiederholt werden.

55 ⟨*1⟩+≡
```
scatter(xy)
```



Praxis: Wir erkennen, daß die Wolke schön aufgelockert ist. Die weit entfernt liegenden Werte sind nun am Rand der Wolke zu finden. Besondere Strukturen, die als Abhängigkeiten der beiden Merkmale deutbar sein könnten, treten aber nicht hervor.

5.7 Konvexe Hüllen

Konzept: Wir wollen noch einen Versuch der optischen Unterstützung unternehmen, indem wir den Plot mit konvexen Hüllen versehen.

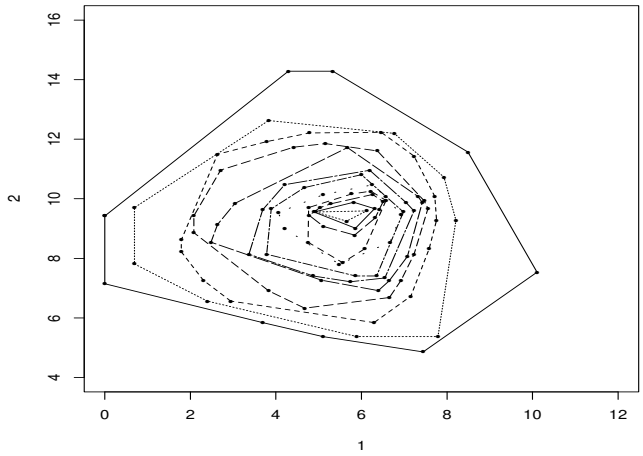
> *Algorithmus: — Konvexe Hülle*
> *Die konvexe Hülle ist das kleinste die Punktewolke umschließende Polygon.*

Notwendigerweise werden die Eckpunkte des Polygons durch Beobachtungspunkte definiert. Entfernt man die Datenpunkte, die zum Polygon gehören, aus dem Datensatz, läßt sich wieder die konvexe Hülle der verbliebenen Punkte finden. Nach dieser Melodie, auch data peeling genannt, kann man solange fortfahren, bis (fast) alle Punkte entfernt sind.

Praxis: Wir wollen die Folge der konvexen Hüllen sehen.

56 ⟨*erstelle Scatterplot zu* `xy` *und zeichne konvexe Hüllen* 56⟩≡
```
scatter(xy, hull=100)
```

Praxis: Wir erhalten damit dieses Bild:



Allenfalls läßt sich ablesen, daß bei sehr kleinen Zeitdifferenzen nicht die größten Datenmengen transportiert werden. Eine Struktur nach dem Motto je größer das eine, umso größer (oder kleiner) das andere, ist nicht zu erkennen.

5.8 Der Korrelationskoeffizient

Konzept: Zur Beschreibung der linearen Abhängigkeit lassen sich auch Maßzahlen heranziehen. Die bekannteste ist mit dem Korrelationskoeffizienten (genauer: Korrelationskoeffizienten von Bravais-Pearson) gegeben. Wir haben diese Maßzahl nur aus Gründen ihres häufigen Gebrauchs mit aufgenommen, wollen sie hier aber nicht umfassend diskutieren, da sie nicht ganz in das Kapitel über Graphiken paßt.

Konzept: Das Formale:

Algorithmus: `cor(x,y)` *— Korrelationskoeffizient der Ausprägungen* (x_i, y_i) *mit* $i = 1, \ldots, n$.

```
cor(x,y) = sum((x-mean(x))*(y-mean(y)))/
           (sum((x-mean(x))^2)^0.5 * sum((y-mean(y))^2)^0.5)
```

oder

$$\texttt{cor(x,y)} = \frac{\sum_i ((x_i - \bar{x}) \times (y_i - \bar{y}))}{\sqrt{\sum_i (x_i - \bar{x})^2 \times \sum_i (y_i - \bar{y})^2}}$$

Ist der Korrelationskoeffizient nahe bei 0, liegt kein linearer Zusammenhang vor. Liegen alle Beobachtungen sehr nahe einer Geraden mit positiver (negativer) Steigung, so nimmt er Werte in der Nähe von +1 (-1) an.
Wie viele andere Maße auch, besitzt der Korrelationskoeffizient die Schwäche der Ausreißerempfindlichkeit. Außerdem mißt er, wie gesagt, nur den Grad des linearen Zusammenhangs, so daß zum Beispiel bei Punkten, die nahe einer Kreislinie liegen, die Abhängigkeitsstruktur nicht erkannt werden kann.

Praxis: Wir wollen den Korrelationskoeffizienten berechnen. Vielleicht bestätigt der Korrelationskoeffizient unsere Einsichten.

57 ⟨*berechne Korrelationskoeffizienten der ersten beiden Elementen von* `xy` 57⟩≡

```
cor(xy[[1]],xy[[2]])
```

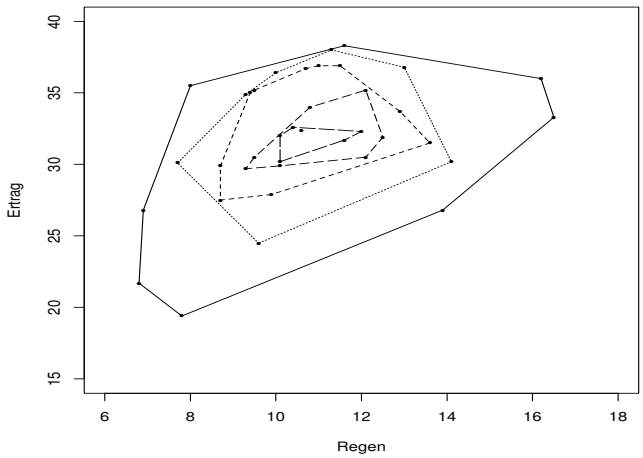
```
[1] 0.02028
```

Das Ergebnis weicht nicht wesentlich von 0 ab. Das haben wir auch nach dem letzten Bild erwartet, da sich dort auch kein linearer Zusammenhang gezeigt hat.

Praxis: Wir können also zusammenfassen, daß zu der Frage: *Besteht ein Zusammenhang zwischen den Zwischenzeiten und den übertragenen Mengen?* kein Zusammenhang festgestellt werden konnte. Um ein Beispiel zu zeigen, bei dem ein gewisser Zusammenhang zu erkennen ist, legen wir auf `xy` Kornerträge und festgestellte Niederschlagsmengen ab und setzen die letzten beiden Module noch einmal ein. Man kann sich schon vorstellen, daß es zwischen den beiden Merkmalen *Regenmenge* und *Ertrag* Zusammenhänge gibt. (Siehe [Becker et al., Datensatz `corn`, p. 647].)

58 ⟨*zeichne zu Regenmengen und Kornerträge konvexe Hüllen und berechne Korrelationskoeffizienten* 58⟩≡
```
xy<-list("Regen"=corn.rain,"Ertrag"=corn.yield)
scatter(xy, hull=100)
cor(xy[[1]],xy[[2]])
```

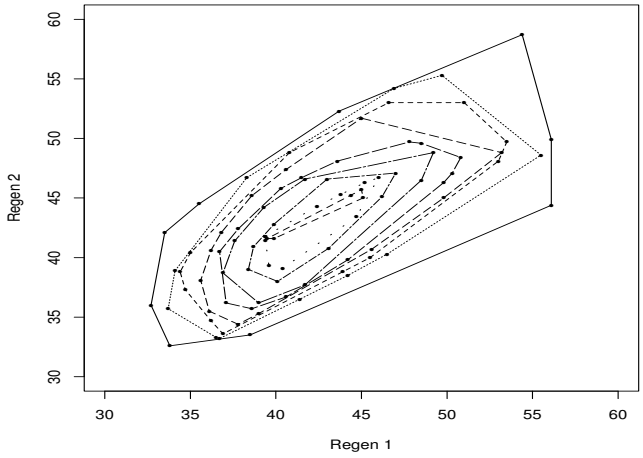
Wir erhalten:




```
Berechnung des Korrelationskoeffizienten zu xy
[1] 0.402629
```

Ein Wert von 0.40 zeigt nur einen sehr schwachen linearen Zusammenhang an. Jedoch ist aus dem Scatterplot abzulesen, daß bei großen Regenmengen keine kleinen Erträge beobachtet wurden. Die Hüllen haben auch alle eine gewisse Tendenz von links unten nach rechts oben, so daß wohl doch von einem Zusammenhang auszugehen ist.
Als zweites Beispiel seien zwei Datensätze vorgeführt, die Regendaten aus zwei verschiedenen Quellen, die sich beide auf New York und die gleichen Zeiträume beziehen, zeigen. (Siehe [Becker et al., Datensatz `rain`, p. 656].)

59 ⟨*zeichne zu Regendatensätzen konvexe Hüllen und berechne Korrelationskoeffizienten* 59⟩≡
```
xy<-list("Regen 1"=rain.nyc1,"Regen 2"=rain.nyc2)
scatter(xy, hull=100)
cor(xy[[1]],xy[[2]])
```




```
Berechnung des Korrelationskoeffizienten zu xy
[1] 0.7133686
```

Hier ist der Korrelationskoeffizient schon größer und auch das Bild unterstützt die Abhängigkeitsvermutung. Es ist aber erstaunlich, daß sich die Abhängigkeit als nicht noch stärker erweist. Spürnasen können ja über die angegebene Literaturstelle weitere Forschungen anstellen.

5.9 Draftsman's Display

Praxis: Zur zweiten Frage *Ist die Belastung innerhalb eines Tages gleichmäßig verteilt oder verändert sie sich im Zeitablauf?* wurden oben schon drei Vorgehensweisen angedeutet. Wir wollen diese ein wenig näher betrachten. Wenn wir fragen, was zu erwarten ist, dann dürften Antworten wie: *in der Nacht wird die Belastung geringer ausfallen als am Tage* oder *eine relativ hohe Belastung ist mittags und vielleicht am frühen Abend zu erwarten* anzutreffen sein. Wir wollen diesen Vermutungen nachgehen.
Kommen wir zum ersten Vorgehensvorschlag, bei dem die Daten anzahlenmäßig in gleich große Stücke geteilt werden. Damit steht über die laufende Nummer ein Bindungsmerkmal zur Verfügung, mittels dessen Gegenüberstellungen vorgenommen werden können.
Zunächst führen wir die Aufteilung durch.

60 ⟨*lege Zwischenzeiten vom 03.02.97 auf* `xy` *ab und ordne Daten in* `anz` *gleich große Klassen ein* 60⟩≡

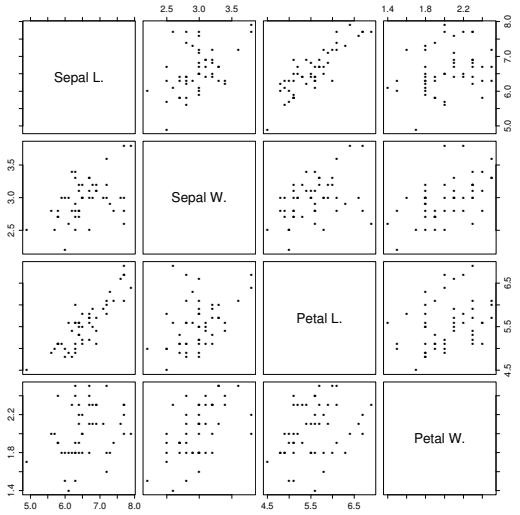
```
xy<-list(dzeitpunkte.03.02.97)
cat("xy in gleich grosse Klassen einteilen: xy\n")
xy<-split.in.eq.classes(xy)
```

Technik: Die Anzahl `anz` wird erfragt. Für die Übereinstimmung mit den abgedruckten Ergebnissen muß man `anz` auf 4 setzen.

Konzept: Jetzt haben wir `anz` Stichproben von gleicher Länge vorliegen. Diese könnten wir uns im `anz`-dimensionalen Raum vorstellen, aber nicht zu Papier bringen. Jedoch ist eine Gegenüberstellung von jeweils zwei Stichproben in einem Scatterplot möglich. Damit ergeben sich `0.5*(anz-1)*anz` verschiedene Scatterplots. Werden diese in einer bestimmten Weise angeordnet und dann noch an der gedachten Diagonalen von links oben nach rechts unten gespiegelt, erhält man einen sogenannten Draftsman's Display.

Algorithmus: — Draftsman's Display der verbundenen Stichproben
`xy` = $(x_{\bullet 1}, x_{\bullet 2}, \cdots, x_{\bullet \texttt{anz}})$
Die schachbrettartige Anordnung von Scatterplots heißt Draftsman's Display, wenn in dem Feld (i, j) mit $i \neq j$ der Scatterplot der i-ten Stichprobe und der j-ten Stichprobe zu finden ist.

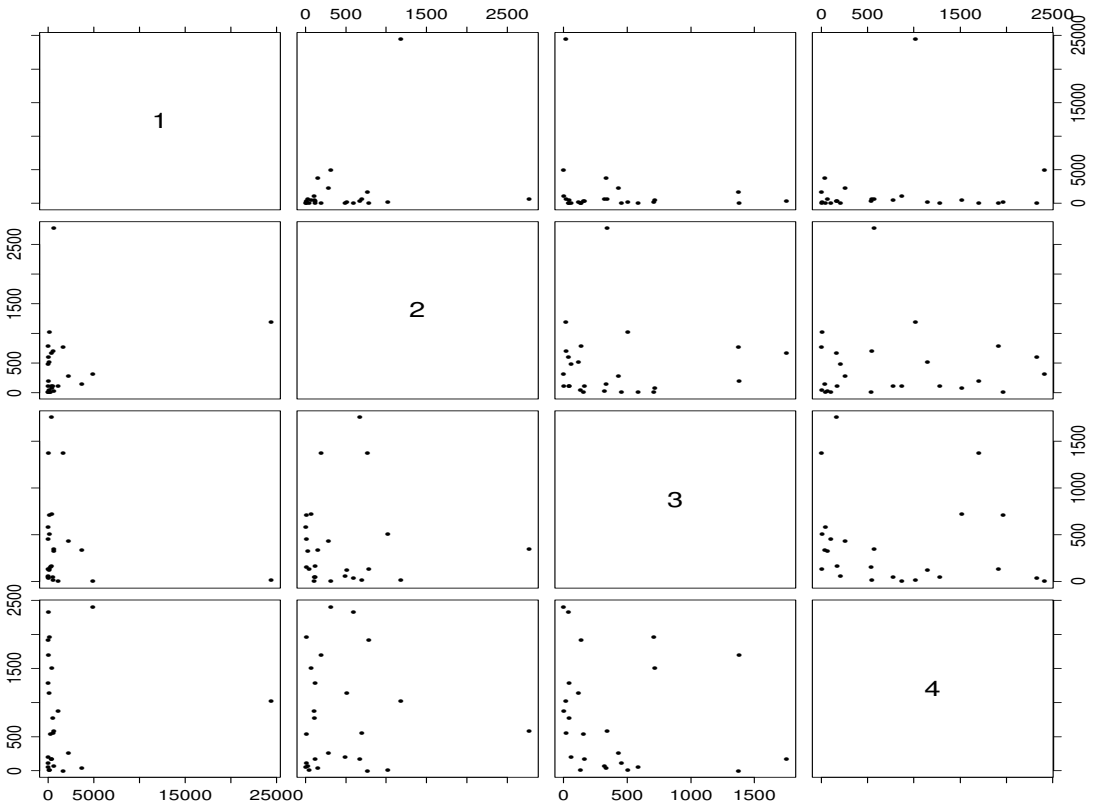
Hier ist ein Beispiel, das in der statistischen Literatur häufig diskutiert wird. (Siehe [Becker et al., Datensatz: `iris`, p. 650].)



Der Plot ist sicher hilfreich bei der Strukturensuche.

Praxis: Die Bindung ist in erster Linie hier eine technische. Eine substanzielle Begründung ist bisher noch nicht gegeben worden. Jedoch ist klar, daß sich zum Beispiel ein Trend in den Daten an unterschiedlichen Achsenbereichen oder an bestimmten Strukturen innerhalb der Scatterplots zeigen würde. Wir wollen dieses Instrument, das in anderen Situationen viel wirkungsvoller ist, nun einsetzen.

61 ⟨*erstelle Draftsman's Display zu* `xy` 61⟩≡
```
pairs(matrix(unlist(xy),ncol=length(xy)),labels=as.character(1:length(xy)))
```



Das Auffälligste sind die unterschiedlichen Bereiche, in denen die Werte der einzelnen Teildatensätze liegen. Das erste Viertel der Daten beansprucht im wesentlichen einen Bereich von 0 bis zirka 5000 (Sekunden), wobei zusätzlich der Extremwert von knapp 25000 im Ausschnitt untergebracht ist. Das Fenster des zweiten Viertels geht von 0 bis knapp unter 3000, das des nächsten bis etwas über 1500 und das des vierten bis 2500. Betrachtet man die größten Zeitdifferenzen als Indikator, so zeigen diese die größten Zugriffspausen an und lassen damit Folgerungen für die Belastung zu, die in Richtung der oben angestellten Vermutungen gehen. Sehr überzeugend ist diese Argumentation aber nicht, da die meisten Beobachtungen (Bildpunkte) nicht verarbeitet worden sind und die Zeit nur implizit aus dem graphischen Ergebnis zu erkennen ist.
In den kleinen Plots ist übrigens keine Beziehung zwischen den vier Vierteln zu erkennen. Wir wollen an dieser Stelle weitere Gedanken dem Leser/Experimentator überlassen und uns dem zweiten Weg widmen. Immerhin hat uns der erste Versuch das Werkzeug Draftsman's Display beschert.

5.10 Noch einmal vergleichende Boxplots

Konzept: Der zweite Weg besteht darin, den Tag in verschiedene, zeitlich gleichlange Zeitabschnitte zu teilen, um dann die Beobachtungen der einzelnen Abschnitte zu vergleichen. Wird der gesamte Datensatz in `anz` Abschnitte eingeteilt, ergeben sich `anz` neue Datensätze, die mit oben dargestellten Techniken verarbeitet werden können.

Praxis: Wir wollen die Zwischenzeiten vom 03.02.97 gemäß den zugehörigen Zeitpunkten in `anz` Klassen einteilen.

Technik: Die Anzahl der Abschnitte wird erfragt.

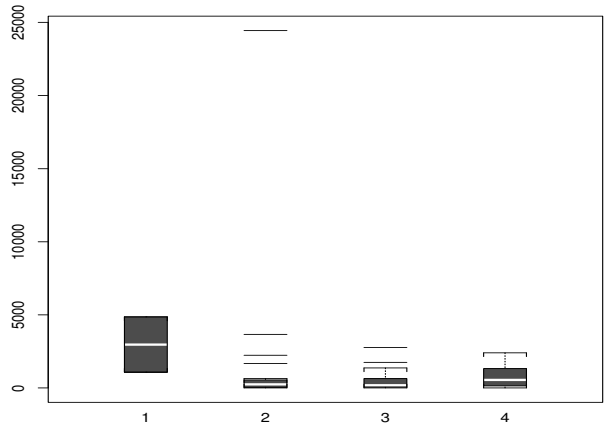
62 ⟨*lege Zwischenzeiten und Zeiten vom 03.02.97 auf* `xy` *ab, ordne* `xy[1]` *gemäß* `xy[2]` *in* `anz` *Klassen ein:* `xy` 62⟩≡

```
xy<-list(dzeitpunkte.03.02.97, zeitpunkte.03.02.97)
xy<-split.x.by.y(xy[[1]],xy[[2]])
```

Praxis: Nun können wir mit Leichtigkeit vergleichende Boxplots zu den vier Zeitabschnitten erstellen.

63 ⟨* 1⟩+≡
```
boxplot(xy)
```

Für vier zeitlich gleich große Teile erhalten wir:

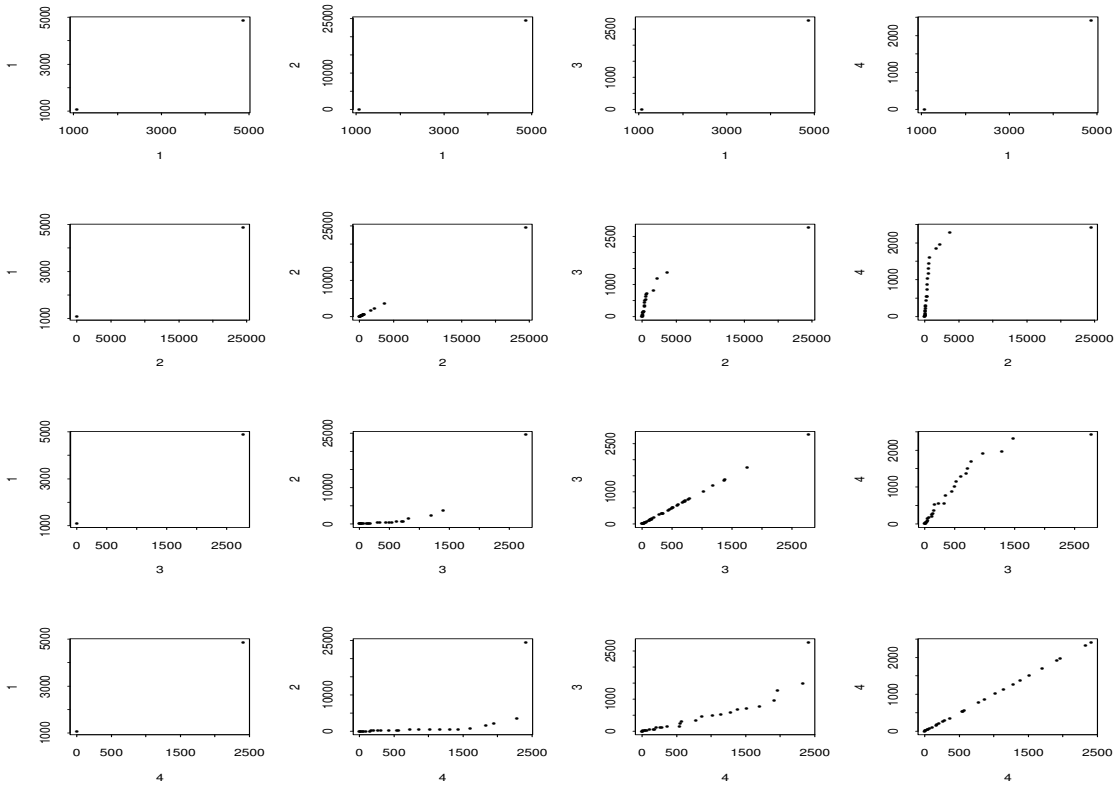


Jeder Boxplot repräsentiert 6 Stunden des Tages 03.02.97. Auf den ersten Blick sehen die vier einzelnen Plots recht ähnlich aus. Doch besteht der erste nur aus einer Box, deren Niveau über den anderen drei Boxen liegt. Der zweite Teildatensatz beinhaltet die größte Zwischenzeit, die knapp unter 25000 Sekunden abgebildet ist. Der dritte und der vierte Datensatz scheinen sich von der Struktur her nicht zu sehr zu unterscheiden. Jetzt haben wir zwar eine zeitgemäße Zerteilung der Daten vorgenommen, doch fördert das gewählte Instrument *Boxplots* Unterschiede nicht sehr deutlich zu Tage. Wir wollen mit QQ-Plots die vier generierten Datensätze gegenüberstellen.

5.11 QQ-Plots, paarweise

Oben haben wir zum direkten Vergleich verschiedener Stichproben QQ-Plots kennengelernt. Wir wollen nun zum Vergleich der vier Teildatensätze QQ-Plots erstellen und wie beim Draftsman's Display schachbrettartig anordnen.

64 ⟨*erstelle zum paarweisen Vergleich der Datensätze von* `xy` *QQ-Plots* 64⟩≡
```
qqpairsplot(xy)
```



Die sicher etwas zu kleinen Bilder der ersten Bilderspalte unterstützen die erste Vermutung: Der Plot oben links zeigt den QQ-Plot der Zwischenankunftszeiten, die den ersten sechs Stunden zuzurechnen sind, mit sich selbst. Es sind nur zwei Punkte in dem Plot zu erkennen. Die vier Plots in der Süd-Ost-Ecke der Gesamtgraphik weisen dagegen eine Vielzahl von Punkten auf. Mit den unterschiedlichen Punktezahlen ist offensichtlich eine unterschiedliche Zugriffsintensität belegt.
Zur Bestätigung wollen wir die Anzahlen der einzelnen Teildatensätze genau berechnen.

65 ⟨*errechne Anzahlen der Stichproben von* `xy` 65⟩≡
```
lapply(xy,length)
```

Wir erhalten:

```
Anzahlen der Stichproben von xy
$"1":
[1] 2

$"2":
[1] 25

$"3":
[1] 48

$"4":
[1] 28
```

Die größte Belastung herrscht demnach im Zeitraum zwischen 12.00 und 18.00. Weiter zeigen die Teildatensätze drei und vier recht ähnliche Strukturen. Vergleiche mit dem zweiten Plot lassen sich aufgrund des Ausreißers nicht gut anstellen. Wir wollen zum Abschluß versuchen, mögliche Belastungsunterschiede, die sich im Laufe eines Tages einstellen, deutlicher zusammenhängend darzustellen. Dazu wollen wir uns nur bekannter Instrumente bedienen.

5.12 Jitterplot, F.dach und f.dach

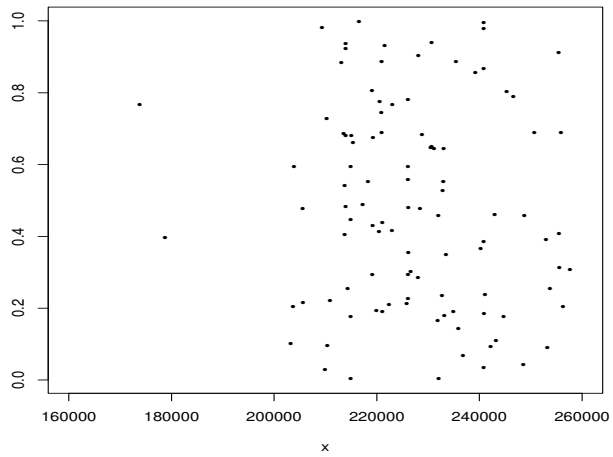
Praxis: Der dritte Vorschlag bestand darin, nicht die Zwischenzeiten, sondern die Zugriffszeiten selbst zu untersuchen.

66 ⟨*lege Zeitpunkte vom 03.02.97 auf* `xy` *ab* 66⟩≡
```
xy<-list(zeitpunkte.03.02.97)
```

Zunächst erstellen wir einen Jitterplot der Zugriffszeitpunkte.

67 ⟨*zeichne Jitterplot zu* `xy` 67⟩≡
```
jitterplot(xy)
```

Wir erhalten:



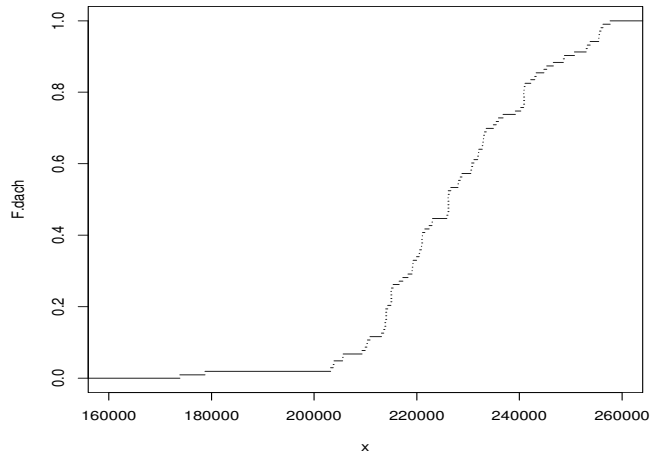
Jetzt sieht man den Belastungsunterschied schon sehr deutlich. Um eine quantitative Vorstellung zu bekommen, hilft uns die empirische Verteilungsfunktion weiter.

Konzept: Aus der empirischen Verteilungsfunktion läßt sich ermitteln, in welchen Intervallen welche Anteile von Beobachtungen liegen.

Praxis: Wir wollen das Instrument einsetzen, um die Anteile der Zugriffszeitpunkte zu betrachten, die vor einem beliebigen Zeitpunkt x liegen. Damit läßt sich aus dem Bild ablesen, in welchen Bereichen erhöhte Nachfrage geherrscht hat.

68 ⟨*zeichne* `F.dach` *zu* `xy` 68⟩≡
```
F.dach(xy)
```

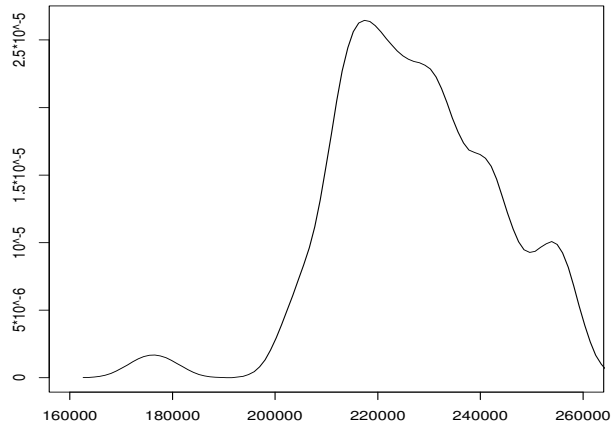
Die empirische Verteilungsfunktion zeigt eine klare Zweiteilung. Im ersten Teil des Bildes steigt die Kurve kaum an, im zweiten dafür um so kräftiger.



Damit ist die Vermutung untermauert, daß die Belastung in der Nacht erheblich geringer ist als am Tage. Spitzenbelastungen, die mit einem starken Anstieg der Kurve einhergehen, lassen sich jedoch nur vage abschätzen.
Zur Ermittlung der dichtesten Bereiche empfehlen sich Dichtespuren als Instrument. Na, dann auf!

69 ⟨*zeichne Dichtespuren zu* `xy` 69⟩≡
```
dichtespur(xy)
```

Für die nach einigen Versuchen gewählte Fensterbreite von 15000 erhält man:



Nun haben wir wirklich eine plastische Vorstellung von der Belastung im Tagesablauf bekommen. Es muß natürlich bei der Interpretation berücksichtigt werden, daß die Kurve am rechten Rand fallen muß, da jenseits der Tagesgrenze keine Werte eingegangen sind. Kritisch ist auch anzumerken, daß für eine Präsentation die Achsenbeschriftung nicht sehr geeignet ist. Wer kann schon 180000 Sekunden in Kopf umrechnen (180000 Sekunden = 3000 Minuten = 50 Stunden = 2 Tage und 2 Stunden, entsprechend: 216000 Sekunden = 3600 Minuten = 60 Stunden = 2 Tage und 12 Stunden)?

Technik: Eine Rechenhilfe könnte für Umrechnungen helfen — sie funktioniert leider nur am Rechner.

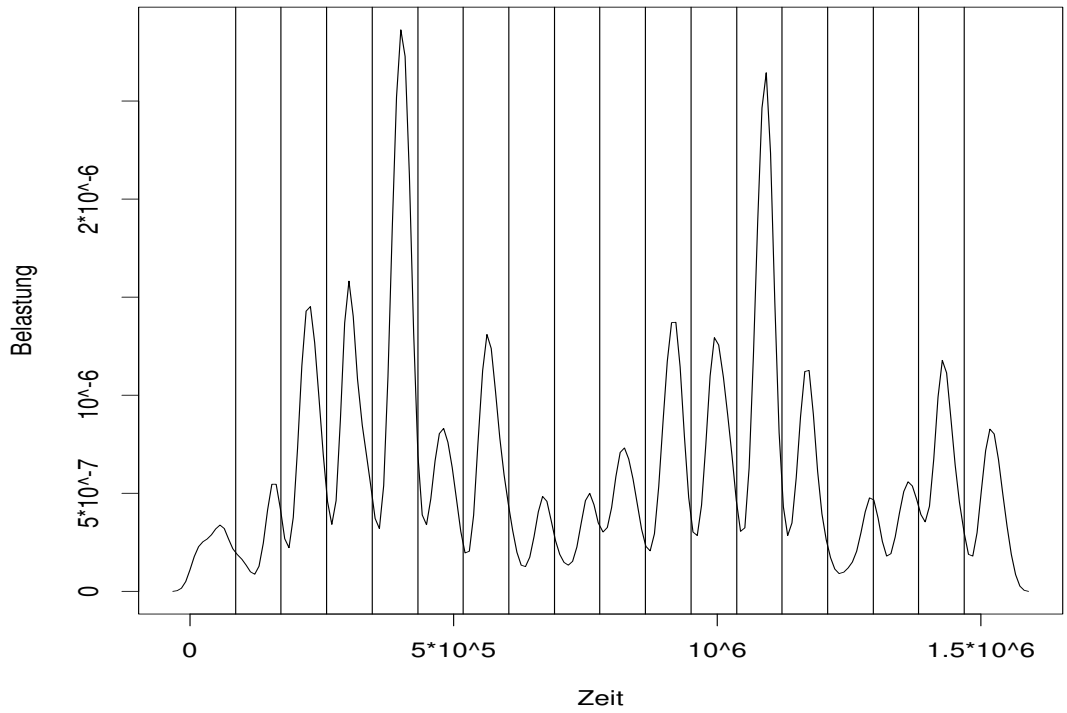
70 ⟨*rechne Sekunden in Tage, Stunden und Minuten um* 70⟩≡

```
print("Geben Sie die Sekunden ein!")
h<-c(scan(,0,n=1),0)[1]; ta<-floor(h/86400); h<-h%%86400;
st<-floor(h/3600); h<-h%%3600; mi<-floor(h/60); h<-h%%60
print(paste(ta,"Tage",st,"Stunden",mi,"Minuten",h,"Sekunden"))
```

Konzept: Es ist festzustellen, daß nicht immer noch mehr Instrumente her müssen, sondern daß es viel wichtiger ist, die bekannten geschickt einzusetzen.

5.13 Dichtespur über den ganzen Monat

Die Belastungsfragen und die Strukturensuche ist noch lange nicht ausgereizt. Doch erscheint an dieser Stelle eine Unterbrechung angebracht zu sein. Als Zugabe wird noch eine Graphik präsentiert, die eine Dichtespur über die ersten 18 Tage des Februars zeigt. Die einzelnen Tagesgrenzen sind durch senkrechte Linien hervorgehoben. Als Input wurden die Zeitpunkte der Serverzugriffe verwendet. Deutlich sieht man, daß an jedem Tag ein lokales Maximum anzutreffen ist. Außerdem erkennt man ohne weiteres die Wochenstruktur. Zur Erinnerung sei gesagt: der Tag 01.02.97 war ein Samstag.



Leider ist die Achsenbeschriftung in einem unschönen Format. Da die einzelnen Werte hier aber keine Rolle spielen, ist keine zusätzliche Mühe zur Verschönerung aufgewendet worden. Zur Vermeidung von Mißverständnissen sei am Beispiel erklärt, daß `1.5*10^6`=1.5×10^6= 1 500 000.

5.14 Was haben wir gelernt?

Konzept: Auf der einen Seite wurde eine ganze Reihe statistischer Werkzeuge vorgestellt, mit denen man im Rahmen des Übungsteil noch viele Versuche anstellen kann.

Praxis: Auf der anderen haben wir zur Serverbelastung gelernt, daß die Belastungen verschiedener Tage in der Beziehung Ähnlichkeiten aufweisen, als daß zu Tagesbeginn eine geringe Last vorherrscht. Die Hauptbelastungen dürften nachmittags anzutreffen sein. Wie das letzte Bild zeigt, hängt das Niveau der Last stark vom Wochentag ab. Der Unterschied zwischen dem 03.02.97 und dem 17.02.97 fiel nicht so groß aus, da es sich in beiden Fällen um einen Montag gehandelt hat. Ein Zusammenhang zwischen dem Merkmal Mengen und der Zeit oder der durch kurze Zwischenzeiten gemessenen Belastung konnte nicht festgestellt werden.

5.15 Offene Fragen

Als offene Fragen können angeführt werden:

- Wie lassen sich die beobachteten Belastungsphänomene modellieren? Die Beantwortung dieser Frage ist Voraussetzung für die Beantwortung der folgenden beiden Fragen:
- Wie lassen sich zu erwartende Belastungen abschätzen? Diese Frage führt zu dem Kapitel *Schätzen*.
- Wie lassen sich Vergleichsaussagen präziser abhandeln? Dieses führt zum Kapitel *Testen*.

Das folgende Kapitel befaßt sich deshalb mit Modellierungsfragen.

5.16 Kontingenztabellen

Viele Datensätze haben nur ein ordinales oder auch nur ein nominales Meßniveau. Für solche sind die oben beschriebenen Techniken wohl nicht geeignet. Dieser Abschnitt soll daran erinnern, daß man dann nach anderen als den hier beschriebenen Methoden Ausschau halten muß. Als Tropfen auf den heißen Stein ist im Auswahlmenue zu diesem Kapitel ein Punkt zu finden, mit dem man Kontingenztabelle ein wenig untersuchen kann.

5.17 Aufgaben

Die Aufgaben sollen mit folgenden unterstützenden Möglichkeiten bearbeitet werden.

71 ⟨*1⟩+≡
```
cat("Revbook:3.m:menu.mu:Menue Multivariate Beschreibungstechniken\n")
if(exists("auswahl.mgraphiken")) auswahl.mgraphiken()
```

Zur Information wird wieder das Auswahlmenü gezeigt:

```
lege Datensaetze zur Bearbeitung auf xy ab
modifiziere Datensaetze xy
lege einen Datensatz von xy auf x ab
ermittle einige Ma{\ss}zahlen zu xy
erstelle Boxplot zu xy
erstelle Jitterplot zu xy
zeichne Histogramme zu xy
zeichne Dichtespuren zu xy
zeichne F.dach zu xy
zeichne zu xy F.dach und f.dach
erstelle Scatterplot zu den ersten beiden Elementen von xy
erstelle Scatterplot zu xy und zeichne konvexe Huellen
erstelle zu xy[1:2] einen QQ-Plot
erstelle Draftsmans Display zu xy
erstelle zum paarweisen Vergleich von xy QQ-Plots
mache mit xy Bootstrap-Experimente
Kontingenztabellenfunktion mit xy
```

a.) Erstellen Sie für die übertragenen Datenmengen vom 03.02.97 und 17.02.97 vergleichende Boxplots! Dazu müssen Sie zunächst die entsprechenden Daten auf dem Arbeitsobjekt `xy` ablegen. Versuchen Sie, Ausreißer zu entfernen (oder die Daten zu logarithmieren) und das Bild zu wiederholen. Scheinen Ihnen die Verteilungen sehr unterschiedlich zu sein?

b.) Erstellen Sie Histogramme für die übertragenen Datenmengen vom 03.02.97 und 17.02.97, nachdem vorher die Ausreißer entfernt worden sind. Bei welcher Klassenanzahl sehen die Verteilungen sehr ähnlich aus, bei welcher sehr unterschiedlich?

c.) Versuchen Sie, mit den Daten der letzten Übung verschiedene Dichtespuren zu erstellen. Welche Fensterbreite führt Ihrer Meinung nach zu einem brauchbaren Ergebnis?

d.) Was lesen Sie aus dem QQ-Plot der Daten aus b.) ab? (Wieso ist übrigens ein Scatterplot nicht geeignet/möglich?)

e.) Legen Sie auf dem Arbeitsobjekt `xy` einen der Datensätze doppelt ab. Modifizieren Sie eine Komponente von `xy` durch eine Box-Cox-Transformation. Wie wirken sich verschiedene Parameter der Box-Cox-Transformationen auf das Erscheinungsbild der QQ-Plots aus?

f.) Wiederholen Sie das geschilderte Bootstrap-Experiment für die entsprechenden Daten vom 17.02.97!

g.) Geben Sie manuell `x`-Werte und `y`-Werte mit dem Ziel ein, einen Korrelationskoeffizienten von 0, von 0.5, von -0.99, von 0.90 zu erhalten!

h.) Wiederholen Sie die Analyse der Belastung im Tagesablauf mit den Daten des Tages 17.02.97! Weist der 17.02.97 viele Ähnlichkeiten auf? Sehen Sie Unterschiede?

i.) Legen Sie `Auto1` auf `xy` ab. Dieser Datensatz besteht aus drei Komponenten, die sich auf Autos beziehen. Die erste enthält die Anzahl der Tage seit dem Kauf des Autos, die zweite die Anzahl der gefahrenen Meilen zwischen den letzten beiden Tankstellenbesuchen und die dritte die zuletzt getankte Menge in Gallonen. Die untersuchten Autos waren Neuwagen aus dem Jahr 1974, ([Becker et al., Datensatz `car`, p. 645]). Erstellen Sie einen Draftsman's Display und interpretieren Sie ihn!

j.) Legen Sie `Auto2` auf `xy` ab. Dieser Datensatz besteht aus drei Komponenten, die sich auf Autos beziehen. Ihre Bedeutung ist: Preis, Verbrauch, Gewicht, Länge, Wendekreis. Die Daten beziehen sich auf verschiedene Automodelle. Analysieren Sie die Daten! ([Becker et al., Datensatz `auto`, p. 644]).

6 Modellierungsschritte

72 ⟨* 1⟩+≡
```
cat("Revbook:4.a:start.mo:Kapitel Modellierungsschritte\n")
```

Zum Literaturstudium sei hingewiesen auf: [Autorenkollektiv: Abschnitt über das Verteilungsmodell], [Bamberg, Baur: Kapitel über Zufallsvariablen und ihre Verteilungen], [Schlittgen: Kapitel über diskrete Verteilungsmodelle und über stetige Verteilungsmodelle].

6.1 Motivation — noch unbeantwortete Fragen

Konzept: Die ersten Kapitel dieses Buches beschäftigen sich mit Verfahren zur Beschreibung von Datensätzen. Diese Verfahren wurden zugleich dafür benutzt, in den Problembereich *Belastung eines* WWW*-Servers* auf eine Reihe von Fragen Antworten zu finden. Beschreibende Verfahren stoßen aber an ihre Grenzen, wenn noch nicht eingetretene Phänomene abgeschätzt werden sollen. Dann müssen Erwartungen und Wahrscheinlichkeitsaussagen begründet abgeleitet werden. Der Statistiker muß für solche weitergehenden Wünsche die vorliegenden Gegebenheiten unter Berücksichtigung der Zielsetzung modellieren. Bei der Rückübertragung der Modell-Ergebnisse in den ursprünglichen Problembereich sind etwaige Verstöße gegen Modellannahmen zu berücksichtigen. In diesem Kapitel sollen typische Fragestellungen mit dazu passenden einfachen Modellen vorgestellt und anhand der Serverproblematik demonstriert werden.
Der Gedankengang dieses Kapitels wird durch folgende Fragen geleitet.

Praxis:

- Wie groß ist die Wahrscheinlichkeit, daß in der nächsten Minute ein Server-Zugriff stattfindet?
- In wie vielen Minuten einer Stunde ist mit Serverzugriffen zu rechnen?
- Wie viele Minuten muß man auf den nächsten Serverzugriff warten?
- Wie lange muß man auf den nächsten Zugriff warten?
- Wie viele Zugriffe werden sich in den nächsten 60 Minuten einstellen?

Konzept: Zur Beantwortung dieser Fragen werden die Bernoulli-Verteilung, die Binomial-Verteilung, die geometrische Verteilung, die Exponential-Verteilung und die Poisson-Verteilung vorgestellt und eingesetzt. Es werden Werkzeuge angegeben, mit denen man Stichproben aus diesen Modellen ziehen und die Eignung der Modelle prüfen kann. Zusätzlich werden Kalkulatoren bereitgestellt, mit denen sich konkrete Fragen an die Modelle beantworten lassen.

6.2 Die Bernoulli-Verteilung — Wie groß ist die Wahrscheinlichkeit, daß in der nächsten Minute ein Server-Zugriff stattfindet?

73 ⟨*1⟩+≡
```
cat("Revbook:4.d1:v.bern:Abschnitt Bernoulli-Verteilung\n")
```

Konzept: Wieviel würden Sie darauf wetten, daß in der nächsten Minute ein Serverzugriff stattfindet? Das hängt sicher von der Wahrscheinlichkeit ab, mit der in der nächsten Minute auf den Server zugriffen wird. Leider ist diese Wahrscheinlichkeit unbekannt. Um diese abzuschätzen, muß ein passendes (Verteilungs-)Modell ausgewählt und eingesetzt werden. Die Frage hier führt uns zur Bernoulli-Verteilung bzw. zum Bernoulli-Prozeß. Im folgenden werden Bernoulli-Experimente durchgeführt, eine Abschätzung der gewünschten Wahrscheinlichkeit vorgeschlagen und die Übertragbarkeit auf unsere Server-Probleme überlegt.

Konzept: Wir wollen von der vorliegenden Situation abstrahieren. Die Überschrift erinnert an die Frage: *Wie groß ist die Chance, daß eine zufällig geworfene Münze mit seiner Zahl nach oben landet?* und damit an das einfachste Experiment des Statistikers, das *Bernoulli-Experiment*.

Bernoulli-Experiment

Ein Experiment mit den beiden möglichen Ergebnissen Erfolg (1) *und* Mißerfolg (0)*, wobei die Wahrscheinlichkeit für* Erfolg p *und für* Mißerfolg $q = 1 - p$ *beträgt, heißt Bernoulli-Experiment.*

Mit einem Bernoulli-Experiment läßt sich der Wurf einer Münze modellieren. Von einer fairen Münze wird man $p = 0.5$ fordern. Die Zufallsvariable, die den möglichen Ausgang eines Bernoulli-Experimentes beschreibt, heißt Bernoulli-verteilt:

Bernoulli-Verteilung

X heißt Bernoulli-verteilt mit dem Parameter p, wenn gilt: $P(X = 1) = p$ *und* $P(X = 0) = 1 - p = q$.
Wir schreiben kurz: $X \sim Bernoulli(p)$ *und es gilt:* $E(X) = p$ *und* $Var(X) = pq$.

$E(X)$ ist der Erwartungswert von X und $\mathrm{Var}(X)$ seine Varianz. Hinweis: In diesem Kapitel wird keine ausführliche Einführung in die Wahrscheinlichkeitstheorie gegeben. Alle abgedruckten diesbezüglichen Definitionen, Sätze usw. dienen der Erinnerung und können eine gründliche Einführung nicht ersetzen.

Konzept: Zur Überprüfung, ob eine Münze fair ist, würde man sie wiederholt werfen und die Ergebnisse untersuchen.

Praxis: Ob in der nächsten Minute ein Serverzugriff stattfindet oder nicht, kann als Bernoulli-Experiment angesehen werden. Ein Zugriff werde als Erfolg angesehen ($X = 1$), kein Zugriff in der nächsten Minute als Mißerfolg ($X = 0$). Die gesuchte Wahrscheinlichkeit entspricht dann dem Parameter p der Bernoulli-Verteilung:

$$P(\text{Zugriff in der nächsten Minute}) = P(X = 1) = p$$

Konzept: Welches ist ein geeigneter Vorschlag für p? Um einen Eindruck von den Gegebenheiten zu bekommen, empfiehlt sich zunächst eine Datenbeschau. Dann werden wir nach einem passendem p Ausschau halten, indem wir Eigenschaften des realen Datensatzes mit denen von Modelldaten gegeneinanderstellen.

Praxis: Für die ersten Minuten des 03.02.97 läßt sich herausfinden, ob Serverzugriffe durchgeführt worden sind (Ergebnis: $X = 1$) oder nicht (Ergebnis: $X = 0$):

74 ⟨*ermittle aus Sekundendaten vom 03.02.97 Minuten mit Aktivität ->x und zeige erste Werte von* x *an* 74⟩≡

```
x<-zeitpunkte.03.02.97
# Aktivitaet pro Minute aus Sekundendaten ermitteln
x<-(x%%86400)/60
x<-pmin(1,as.vector(table(cut(x,0:1440))))
# Anfang von x anzeigen
x[1:min(100,length(x))]
```

Dieses liefert:

```
Wie viele Minuten sollen untersucht werden (Default=100)?
1: 100
 [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[38] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[75] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
```

Es zeigen sich viele 0-Ergebnisse. Das zugehörige p dürfte wohl von 0.5 verschieden sein. Aber welcher Wert ist geeignet? Wir wollen die Auswirkung verschiedener p-Werte experimentell untersuchen.

6.2.1 Realisierte Bernoulli-Prozesse

Konzept: Entsprechend zu solchen realen 0-1-Ergebnissen läßt sich ein Bernoulli-Experiment wiederholt durchführen. Falls sich die einzelnen Bernoulli-Experimente nicht beeinflussen und der Parameter p konstant bleibt, heißt eine solche Abfolge von Experimenten *Bernoulli-Prozeß*.
Wir wollen computergestützt Bernoulli-Prozesse mit vorgegebenen p realisieren und studieren, ob das p wiederentdeckt werden kann.

Praxis: Wir wollen zuerst Experimentierbedingungen festsetzen, dann die einzelnen Experimente ablaufen lassen und zum Schluß die Ergebnisse untersuchen.

Technik: Als Experimentierbedingungen sind die Anzahl der Bernoulli-Experimente (`n`), die Erfolgswahrscheinlichkeit (`p`) sowie der Start des internen Zufallszahlengenerators (`ZZ`) festzulegen. Unterschiedliche Zufallszahlengenerator-Starts (durch `ZZ` ausgelöst) führen zu unterschiedlichen Ausgangssituationen, so daß in Abhängigkeit von `ZZ` unterschiedliche 0-1-Ergebnisfolgen resultieren. Nach der Festlegung der Parameter werden die Experimente durchgeführt.

75 ⟨* 1⟩+≡
```
# lege Parameter [[p,n,ZZ]] f\"ur wiederholte Bernoulli-Experimente fest#
cat("Festlegung der Bedingungen fuer wiederholte Bernoulli-Experimente\n")
cat("Legen Sie die Erfolgswahrscheinlichkeit p fest (Default: 0.5): p=?\n")
p<-c(scan(,0,n=1),0.5)[1]
cat("Legen Sie die Anzahl n der Bernoulli-Experimente fest\n")
cat("(Default: 200): n=?\n")
n<-c(scan(,0,n=1),200)[1]
cat("Zufallsstart: Geben Sie eine ganze Zahl zwischen 0 und 1000 ein\n")
cat("(Default: 17): ZZ=?\n")
ZZ<-c(scan(,0,n=1),17)[1]
#starte Bernoulli-Experimente, Bedingungen: [[n, p, ZZ]], Ergebnis: [[x]]#
set.seed(ZZ)
x<-rbinom(n,1,p)
```

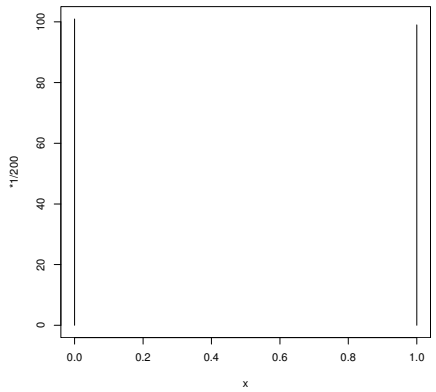
Praxis: Wir wollen die folgenden Festlegungen, die mit den Default-Vorschlägen (das sind die voreingestellten Werte für leere Eingaben) übereinstimmen, ausprobieren:

```
Festlegung der Bedingungen fuer wiederholte Bernoulli-Experimente
Legen Sie die Erfolgswahrscheinlichkeit p fest (Default: 0.5): p=?
[1] 0.5
Legen Sie die Anzahl n der Bernoulli-Experimente fest
(Default: 200): n=?
[1] 200
Zufallsstart: Geben Sie eine ganze Zahl zwischen 0 und 1000 ein
(Default: 17): ZZ=?
[1] 17
```

Nun können wir die erzeugte 0-1-Folge (`x`) mit den deskriptiven Techniken der Statistik beschreiben. Ein Stabdiagramm zeigt die relative Häufigkeit der beiden Ergebnisse an.

76 ⟨* 1⟩+≡
```
stabdiagramm(x)
```

Für die oben abgedruckten Setzungen erhält man das Stabdiagramm:



Die relative Häufigkeit für einen Erfolg beträgt nicht ganz 50 %.

Konzept: Eine genaue Übereinstimmung mit $p = 0.5$ wird sicher auch niemand erwartet haben, da Zufallsergebnisse und Zufallsstichproben eben von Zufällen abhängen. Andererseits liegt der realisierte Erfolgsanteil ziemlich nah an der vorgegebenen Wahrscheinlichkeit p. Dieses erscheint durchaus plausibel zu sein und führt zu dem Vorschlag, die Wahrscheinlichkeit durch das empirische Gegenstück, die relative Häufigkeit, abzuschätzen.
Der Mittelwert der Ergebnisse gleicht übrigens der relativen Häufigkeit oder dem Anteil der Erfolgsausgänge. Warum ist das so?

$$\bar{x} = \sum_i x_i / n = (0 \times \text{Anzahl der Nullen} + 1 \times \text{Anzahl der Einsen}) / n = \text{Anteil der Einsen}$$

Praxis: Wir wollen den Erfolgsanteil bzw. Mittelwert genau berechnen.

77 ⟨*1⟩+≡
```
mean(x)
```

Wir berechnen aus den Ergebnissen der Bernoulli-Experimente den

```
Mittelwert
[1] 0.495
```

und können dies als Schätzvorschlag in die Diskussion einbringen.

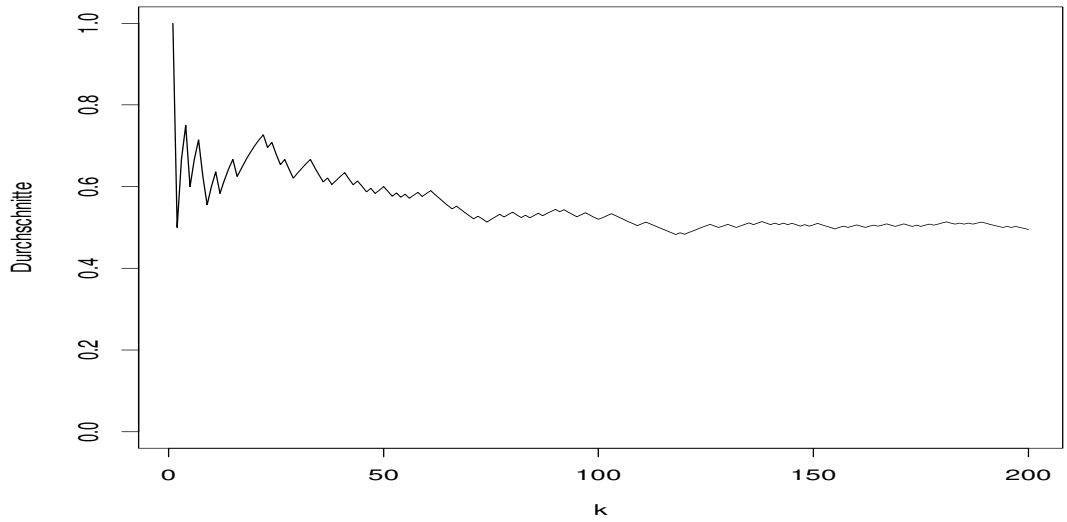
6.2.2 Entwicklung von Durchschnitten

Konzept: Jedoch sollten wir Genauigkeitsfragen bedenken. Intuitiv erwartet man genauere Ergebnisse mit wachsendem n.

Praxis: Dieses läßt sich dadurch überprüfen, daß man den Durchschnitt der ersten k Ergebnisse gegen k abträgt. Auf geht's.

78 ⟨*plotte Durchschnitte für die ersten k Elemente von* x *gegen k* 78⟩≡
```
cat("Plot der Durchschnitte der ersten k Elemente gegen k\n")
h<-cumsum(x)/(1:length(x))
plot(c(1,length(x)),c(max(1,max(h)),min(0,min(h))),
     type="n",xlab="k",ylab="Durchschnitte")
lines(1:length(x), h)
```

Aufgrund der Daten der Bernoulli-Experimente erhalten wir folgende Durchschnittskurve:



Die Folge der Mittel schwanken mit wachsendem n immer weniger und ihr Pfad in der Darstellung strebt wohl gegen $p = 0.5$. Die relative Häufigkeit geht gegen die vorher festgesetzte Wahrscheinlichkeit. Würde man den zutreffenden Parameter p nicht kennen, könnte man auch aufgrund dieses Bildes $p = 0.495$ oder gerundet $p = 0.5$ schätzen.

Konzept: Das vorgeschlagene Schätzverfahren läßt sich auch unter der Überschrift Methode der Momente einführen, denn es wurde, wie folgt, vorgegangen: Setze das empirische Moment $\bar{x}$ in eine Beziehung ein, in der sowohl das theoretische Moment $\mathrm{E}(X)$ als auch der unbekannte Parameter vorkommen, und löse diese nach dem gesuchten Parameter auf:

$$\mathrm{E}(X) = p \quad \Rightarrow \quad p = \mathrm{E}(X) \quad \Rightarrow \quad p_{\text{geschätzt}} = \bar{x}.$$

Praxis: Aus der Graphik läßt sich auch die Qualität der Schätzung erahnen. Für kleine Stichprobenumfänge ($n < 20$) ist der Plot noch sehr unruhig. Damit wirkt ein entsprechender Schätzvorschlag nicht sehr vertrauensvoll. Erst mit größer werdenden n wird die Kurve glatter.

Konzept: In der allgemeinen Situation ist p unbekannt. Dann will man wissen, wie gut ein Schätzvorschlag ist. Die Genauigkeit läßt sich durch die w-fache Wiederholung des beschriebenen Vorgehens demonstrieren.

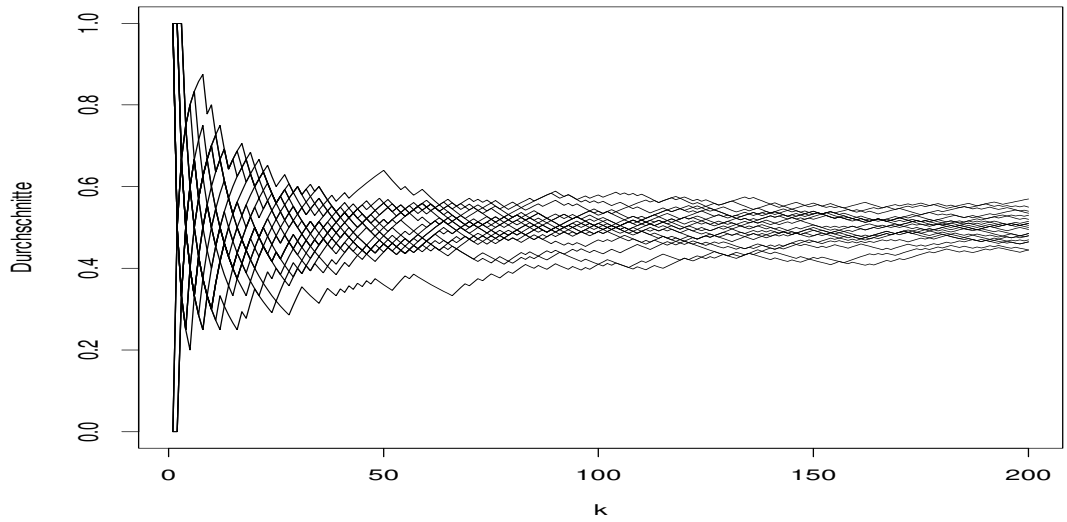
Praxis: Dazu läßt sich das letzte Experiment `w`-mal wiederholen. Für das folgende Bild erhielt `w` den Wert 20 und die anderen Parameter wurden wie oben festgesetzt.

79 ⟨*1⟩+≡

```
# lege Parameter [[p,n,ZZ]] f\"ur wiederholte Bernoulli-Experimente fest #
cat("Legen Sie die Erfolgswahrscheinlichkeit p fest (Default: 0.5): p=?\n")
p<-c(scan(,0,n=1),0.5)[1]
cat("Legen Sie die Anzahl n der Bernoulli-Experimente fest\n")
cat("(Default: 200): n=?\n")
n<-c(scan(,0,n=1),200)[1]
cat("Zufallsstart: Geben Sie eine ganze Zahl zwischen 0 und 1000 ein\n")
cat("(Default: 17): ZZ=?\n")
ZZ<-c(scan(,0,n=1),17)[1]

# lege Wiederholungsanzahl [[w]] fest #
cat("Legen Sie die Anzahl w der Wiederholungen der n Versuche fest\n")
cat("(Default: 20): w=?\n")
w<-c(scan(,0,n=1),20)[1]

# ziehe Stichprobe, zeichne Durchschnittskurven und ermittle Statistiken  #
set.seed(ZZ)
x<-matrix(rbinom(n*(w+1),1,p),(w+1),n)
cat("(w+1)-facher Plot der Durchschnitte der ersten k Elemente gegen k\n")
plot(c(1,n),c(max(1,max(x)),0),type="n",xlab="k",ylab="Durchschnitte")
for(i in 1:(w+1)) lines(1:n, cumsum(x[i,])/(1:n))
cat("Berechnung der zusammenfassender Statistiken\n")
summary.stats(apply(x,1,sum)/n)
```

```
Festlegung der Bedingungen fuer wiederholte Bernoulli-Experimente
Legen Sie die Erfolgswahrscheinlichkeit p fest (Default: 0.5): p=?
[1] 0.5
Legen Sie die Anzahl n der Bernoulli-Experimente fest
(Default: 200): n=?
[1] 200
Zufallsstart: Geben Sie eine ganze Zahl zwischen 0 und 1000 ein
(Default: 17): ZZ=?
[1] 17
Legen Sie die Anzahl w der Wiederholungen der n Versuche fest
(Default: 20): w=?
[1] 20
```

Mit diesen Setzungen erhalten wir `w+1=21` Pfade:



und außerdem für die Stelle $k = 200$ die zusammenfassenden Statistiken:

```
  Min. 1st Qu. Median Mean 3rd Qu. Max.   Std.Dev.  n
 0.445    0.48    0.5  0.5    0.52 0.57 0.03365264 21
```

Konzept: Der Plot zeigt die unvermeidliche Variabilität, die Stichproben innewohnt. Die einzelnen Pfade scheinen alle in etwa gegen das gemeinsame Ziel 0.5 zu streben. Jedoch ist für eine kleine Abweichung von $p = 0.5$ schon ein erheblicher Stichprobenumfang einzusetzen. Die 21 berechneten durchschnittlichen Erfolgszahlen liegen in [0.445, 0.57] und erstrecken sich damit noch über einen recht großen Bereich. Zur näheren Untersuchung der Variabilität der Durchschnitte für ein festes k muß die Binomial-Verteilung herangezogen werden — siehe dazu den nächsten Abschnitt. (Für $n \to \infty$ geht $p_{\text{geschätzt}}$ gegen p und die Variabilität gegen 0, so wie man es sich wünscht. Der Schätzer gehört in die Klasse der konsistenten Schätzer — man wird diese Klasse im weiteren Verlauf der Statistik-Ausbildung kennenlernen.)

6.2.3 Serverdaten

Praxis: Wir wollen uns jetzt wieder — gestärkt durch eine Reihe von Erfahrungen mit Bernoulli-Experimenten — den vorliegenden Daten zuwenden. Zunächst wollen wir `p` durch das Mittel der Minutendaten abschätzen.

80 ⟨*berechne zusammenfassende Statistiken für Minutenaktivitäten (03.02.97)* 80⟩≡

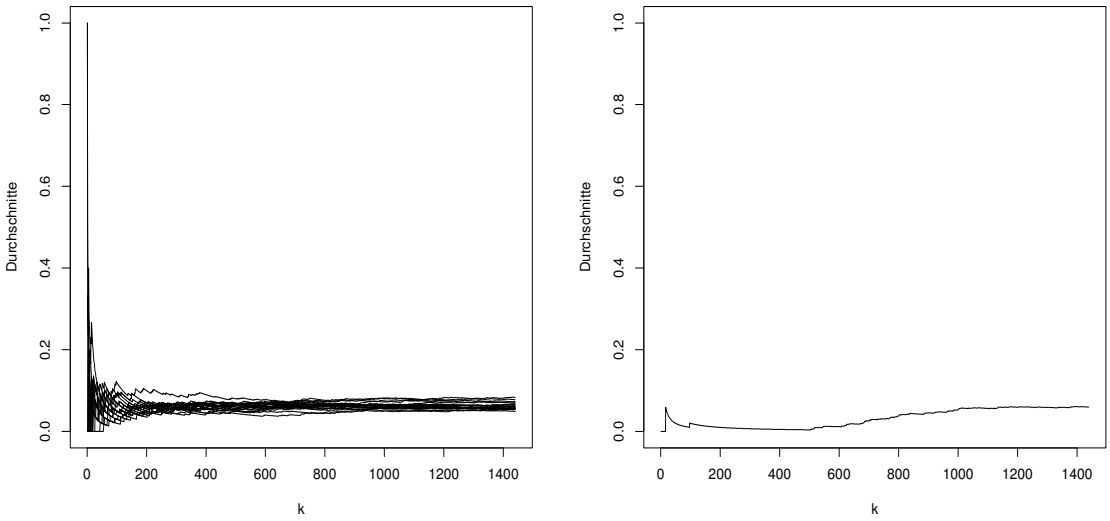
```
x<-zeitpunkte.03.02.97
x<-(x%%86400)/60
x<-pmin(1,as.vector(table(cut(x,0:1440))))
summary.stats(x)
```

Wir erhalten:

```
 Min. 1st Qu. Median    Mean 3rd Qu. Max.  Std.Dev.    n
    0       0      0 0.05972       0    1 0.2370538 1440
```

und können als Wert für den Parameter `p` den Wert 0.06 vorschlagen. Mit diesem `p`, mit `n=1440=24*60`, mit `ZZ=17` und mit `w=20` können wir das letzte Bild wiederholen.

Wir erhalten dann als graphisches Ergebnis den linken der beiden folgenden Plots. (Die Bedeutung des rechten Plots wird etwas weiter unten dargelegt.)



Als Berechnungsergebnisse erhalten wir:

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    Std.Dev.  n
0.04931 0.05625 0.05972 0.06214 0.06597 0.08333 0.008569059 21
```

Die Pfade streben im Mittel einem Wert von ungefähr 6 % entgegen, wobei selbst für den großen Stichprobenumfang ($n = 1440$) die errechneten Durchschnitte noch zwischen zirka 0.05 und 0.08 schwanken.

Konzept: *Die Wahrscheinlichkeit beträgt 6 %!* oder *Die Wahrscheinlichkeit liegt zwischen 0.05 und 0.08!* kann nun auf die gestellte Frage *Wie groß ist die Wahrscheinlichkeit, daß in der nächsten Minute ein Server-Zugriff stattfindet?* geantwortet werden. Wir wollen jedoch noch untersuchen, ob die Modellierung durch wiederholte Bernoulli-Experiment mit konstantem p haltbar ist.

Praxis: Dazu wollen wir den Pfad zeichnen, der sich aus den vorliegenden Daten ergibt, und mit den Pfaden aus dem letzten Experiment vergleichen. Zum besseren Vergleich ist dieser Plot schon rechts neben dem vorherigen abgedruckt worden. Hier folgen die notwendigen Anweisungen.

81 ⟨*plotte Anteil der Minuten mit Zugriff gegen betrachtete Minutenanzahlen* 81⟩≡

```
x<-zeitpunkte.03.02.97
x<-(x%%86400)/60
x<-pmin(1,as.vector(table(cut(x,0:1440))))
h<-cumsum(x)/(1:length(x))
plot(c(1,length(x)),c(max(1,max(h)),min(0,min(h))),
     type="n",xlab="k",ylab="Durchschnitte")
lines(1:length(x), h)
```

Gegenüber der Antwort müssen wir aufgrund des letzten Bildes einige Vorbehalte anmelden. Erstens zeigt das Bild bei genauer Inspektion einen Anstieg der Kurve ab $k = 500$. Dieses stimmt mit der Beobachtung aus dem letzten Kapitel überein, daß die Verteilung der Belastung über den Tag hinweg nicht gleichmäßig ist. Ein p-Wert für morgens wird sich von einem für nachmittags oder abends unterscheiden. Damit ist die Modellierung der Minuten eines ganzen Tages mit Hilfe eines Bernoulli-Prozesses mit konstantem p nicht sehr geeignet. Zweitens dürften sich von Tag zu Tag Schwankungen ergeben, so daß nicht das Phänomen der Variabilität zwischen verschiedenen Tagen übersehen werden darf. Auf die gestellte Frage ist darum die Gegenfrage nach Tag und Uhrzeit berechtigt. Erhält man keine zusätzlichen Hinweise, muß man sich mit der dargestellten Modellierung zufrieden geben. Erst mit genaueren Angaben könnten bessere Daten(ausschnitte) ausgewählt und ein tages- und stundenspezifisches p vorgeschlagen werden.
Wenden wir uns nun der zweiten Frage dieses Kapitels zu. Damit erhalten wir nebenbei ein Instrumentarium zur Quantifizierung der Variabilitäten der Schätzungen.

6.3 Die Binomial-Verteilung — In wie vielen Minuten einer Stunde ist mit Serverzugriffen zu rechnen?

82 ⟨*1⟩+≡
```
cat("Revbook:4.d2:v.binom:Abschnitt Binomial-Verteilung\n")
```

Konzept: Die Binomial-Verteilung liefert die Wahrscheinlichkeiten dafür, daß bei einer festen Anzahl n von unabhängigen Bernoulli-Experimenten mit gleichem p genau x Erfolge eintreten. Werden die Zufallsexperimente *Servertätigkeit in der Minute* $1, 2, \ldots$ durch Bernoulli-Experimente modelliert und wird zunächst unterstellt, daß diese unabhängig sind und sich der Parameter p nicht ändert, so führt die Frage der Überschrift zur Binomial-Verteilung. In diesem Abschnitt werden experimentell einige Eigenschaften der Binomial-Verteilung erkundet, und es wird dargestellt, welche Antwort sich ergeben würde, wenn die Minuten mit Serverzugriffen durch einen Bernoulli-Prozeß beschreibbar wären. Dann werden wieder Abweichungen zur realen Situation diskutiert. Zum Schluß werden noch einige Überlegung zu Schätzgenauigkeiten angestellt, wobei ein Kalkulator für Binomial-Verteilungen wertvolle Dienste leistet.

Praxis: Die Bilder des letzten Abschnitts, in denen verschiedene Pfade eingezeichnet sind, weisen einen Weg zur Beantwortung der gestellten Frage. Eine Stunde hat 60 Minuten. Betrachtet man in den Pfadbildern jeweils die Stelle $k = 60$, so sieht man für das gewählte p realisierte Anteile (in der Semantik des Beispiels: Anzahl der Minuten mit Zugriff / 60 Minuten). Die Anzahlen der Minuten — und hiernach wird gefragt — ergeben sich durch Multiplikation mit 60.

Konzept: Solche Anzahlen sind unter gewissen Bedingungen binomialverteilt. Hier ist der Steckbrief der Binomial-Verteilung:

Binomial-Verteilung

Sind die Zufallsvariablen $X_1, \ldots, X_n$ mit identischem p unabhängig Bernoulli-verteilt, so ist ihre Summe binomialverteilt mit den Parametern n und p:

$$X_1, \ldots, X_n \sim \textit{ unabhängig Bernoulli}(p) \quad \Rightarrow \quad S_n = \sum_{i=1}^{n} X_i \sim \textit{ Binomial}(n, p)$$

S_n hat die Wahrscheinlichkeitsfunktion:

$$f(x) = \binom{n}{x} p^x (1-p)^{n-x} \quad x = 0, 1, \ldots, n$$

und es gilt $E(S_n) = np$ und $Var(S_n) = npq$.

6.3.1 Stichproben aus dem Binomial-Modell

Wir wollen zunächst einige Erfahrungen mit der Binomial-Verteilung sammeln und dann zu der gestellten Frage zurückkommen. Bei 60 Bernoulli-Experimenten werden für $p = 0.5$ ungefähr $np = 60 \times 0.5 = 30$ Experimente mit Erfolg ausgehen. Doch mit wie starken Abweichungen muß gerechnet werden?

Praxis: Wir wollen diese Frage wieder experimentell untersuchen. Dazu ziehen wir eine Stichprobe aus einer Binomial-Verteilung mit `n=60` und `p=0.5`. Die Stichprobe können wir mit uns schon vertrauten Beschreibungstechniken darstellen.

83 ⟨*ziehe* `r` *binomialverteilte Zufallszahlen mit* `n` *und* `p`*:* `x` 83⟩≡
```
cat("Stichprobe aus Binomial-Verteilung\n")
print("Wie gross soll der Parameter n sein (Default: n=1)?")
n<-c(scan(,0,n=1),1)[1]
print("Wie gross soll der Parameter p sein (Default: p=0.5)?")
p<-c(scan(,0,n=1),0.5)[1]
print("Wie viele Zufallszahlen wollen Sie (Default: r=100)?")
r<-c(scan(,0,n=1),100)[1]
print("Zufallsstartgeneratorstart (Default: ZZ=17)?")
ZZ<-c(scan(,0,n=1),17)[1]
set.seed(ZZ)
x<-rbinom(r,n,p)
```

Praxis: Wir wollen folgende Einstellungen wählen.

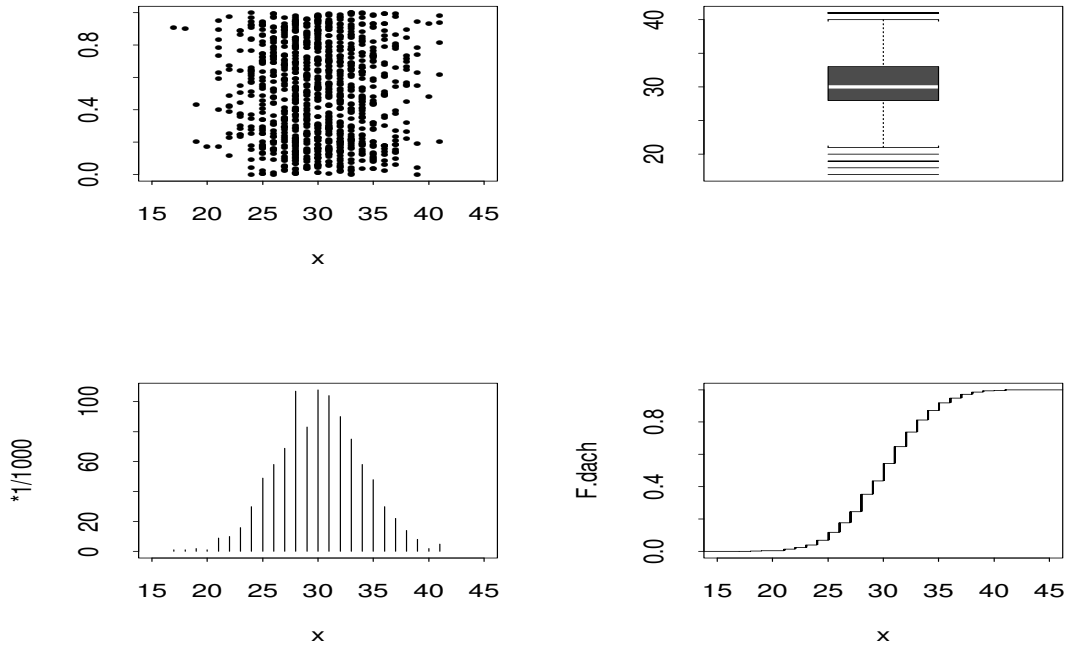
```
Stichprobe aus Binomial-Verteilung
[1] "Wie gross soll der Parameter n sein (Default: n=1)?"
1: 60
[1] "Wie gross soll der Parameter p sein (Default: p=0.5)?"
1: 0.5
[1] "Wie viele Zufallszahlen wollen Sie (Default: r=100)?"
1: 1000
[1] "Zufallsstartgeneratorstart (Default: ZZ=17)?"
1: 17
```

Wir ermitteln die Übersichtsstatistiken und erstellen Graphiken: Jitterplot, Boxplot, Stabdiagramm und die empirische Verteilungsfunktion.

84 ⟨*plotte Stichprobe, berechne Statistiken zu* `x` *— diskret* 84⟩≡
```
par(mfrow=c(2,2))
jitterplot(x); boxplot(x)
stabdiagramm(x); F.dach(x)
par(mfrow=c(1,1))
```

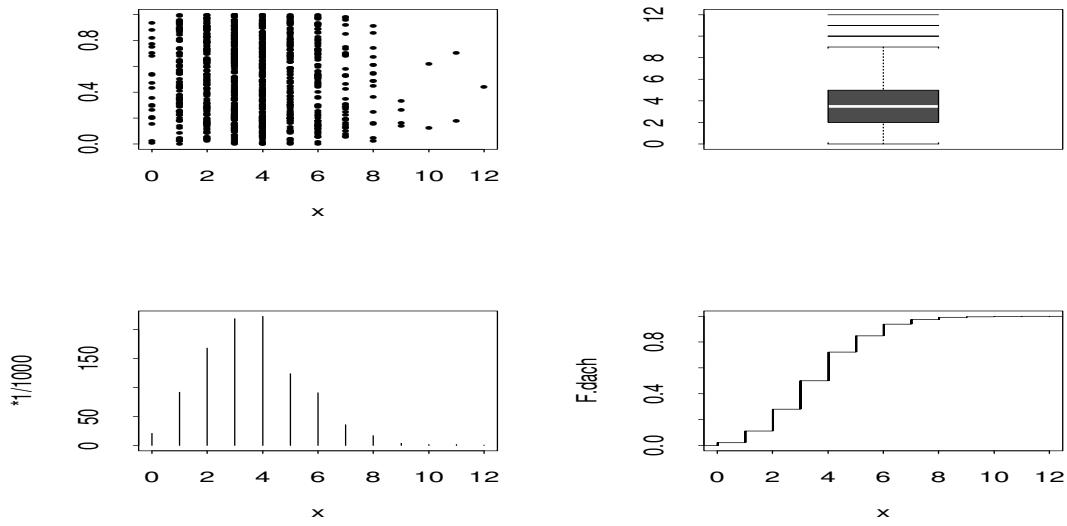
```
Zusammenfassende Statistiken
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.    n
   17      28     30 30.08      33   41 3.867178 1000
```

Zu den oben genannten Experimentbedingungen erhält man die Graphik:



Die Verteilung scheint sehr symmetrisch zu sein. Die mittlere Erfolgsanzahl liegt (wie erwartet) bei 30 Erfolgen. Extreme Ausreißer scheinen nicht aufzutreten. Die meisten Realisationen liegen grob zwischen 20 und 40.
Nach dieser Vorübung steht ein auf unsere Situation angepaßtes Experiment auf der Tagesordnung. Eine Stichprobe aus einer Binomial-Verteilung mit $n = 60$ und $p = 0.06$, dem festgestellten Anteil von Minuten mit Serverzugriffen, führt zu folgendem Übersichtsplot.

Technik: Dazu müssen natürlich die letzten zwei Sektionen geeignet wiederholt werden.



Praxis: Im Unterschied zu den Ergebnissen mit `p=0.5` zeigt der Boxplot eine gewisse Asymmetrie, wie dem Stabdiagramm zu entnehmen ist. Mittel und Median dürften deshalb auch deutlich voneinander abweichen.

Hier sind die Zahlen:

```
Zusammenfassende Statistiken
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.    n
    0       2    3.5 3.621       5   12 1.825566 1000
```

Der Mittelwert und der Median sind fast gleich, dies ist eher ein Zeichen für Symmetrie. Da der Median genau in der Mitte zwischen den Quartilen liegt, geht die angedeutete Asymmetrie wesentlich auf die Schwänze der Verteilung zurück. Weiter läßt sich feststellen, daß die Spanne zwischen den Quartilen 3 beträgt, bei dem Experiment mit $p = 0.5$ dagegen 6. Die Stichprobenstreuungen unterscheiden sich um einen Faktor größer als 2.

Konzept: Kurzes Nachrechnen belegt, daß für $p = 0.5$ die Wurzel aus der Varianz

$$\sqrt{np(1-p)} = \sqrt{60 \times 0.5 \times 0.5} = 3.873$$

ergibt, für $p = 0.06$ stellt sich dagegen nur

$$\sqrt{np(1-p)} = \sqrt{60 \times 0.06 \times 0.94} = 1.840$$

ein. Damit ist die unterschiedliche Variabilität erklärlich.

Praxis: Aus dem Stabdiagramm ist zu ersehen, daß die häufigsten Ergebnisse im Bereich von 1 bis 6 liegen. Überträgt man dieses auf das Serverproblem, so müßte eine zufällig gewählte Stunde zirka eine Minute bis 6 Minuten mit Servertätigkeit besitzen. Hierbei ist dann angenommen, daß $p = 0.06$ der wahre Wert des Parameters ist, sich nicht ändert und keine Abhängigkeiten bestehen. Unter diesen Bedingungen stellen sich im Durchschnitt 3.6 Zugriffe pro Stunde ein.

6.3.2 Serverdaten

Wir wollen sehen, wie gut die Wirklichkeit damit erfaßt ist. Dazu werden wir die 24 Stunden vom 03.02.97 betrachten. Jede der 24 Stunden, die eine beobachtbare Anzahl der Minuten mit Servertätigkeit hervorbringt, kann als ein empirisches Experiment angesehen werden. Die Ergebnisse dieser Experimente lassen sich wie oben die Stichprobe aus der Binomial-Verteilung darstellen.

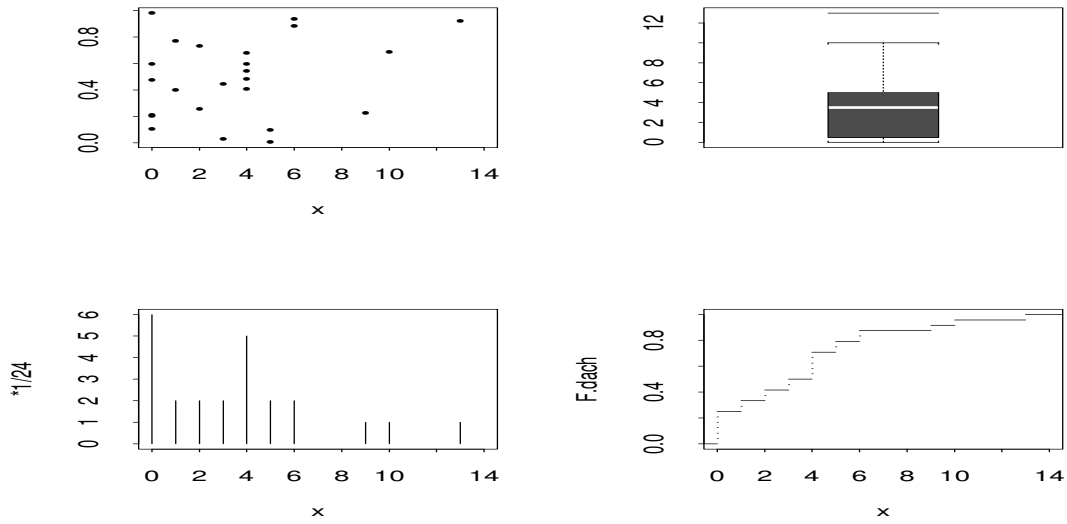
85 ⟨*berechne und plotte Minuten mit Zugriffen der Stunden vom 03.02.97* 85⟩≡

```
x<-zeitpunkte.03.02.97
```

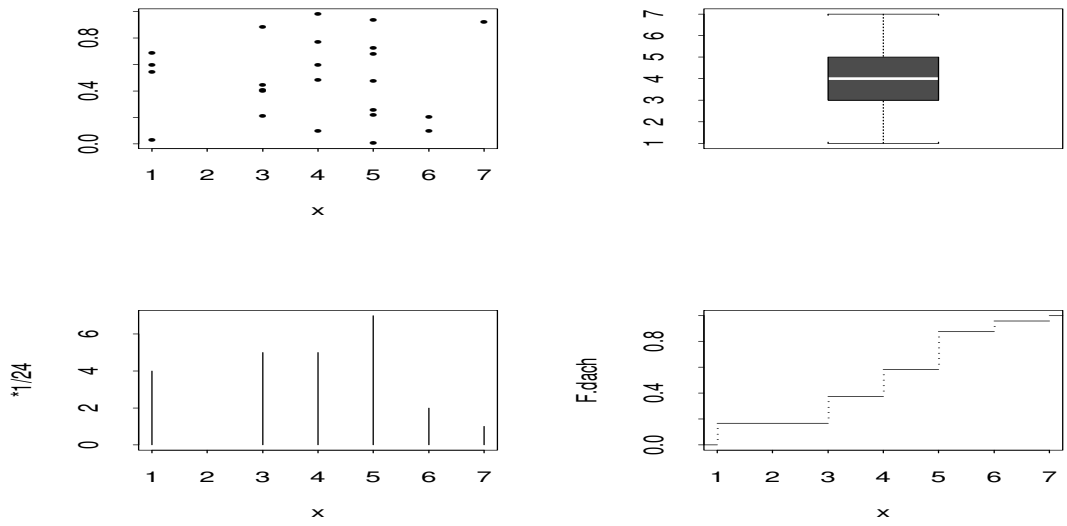
Praxis: Wir erhalten

```
Zusammenfassende Statistiken
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.  n
    0    0.75    3.5 3.583       5   13 3.437854 24
```

sowie



Die Realität sieht — wie wir schon erwartet haben — wieder einmal etwas anders aus als die Theorie. Zwar gehören die Ergebnisse von 1 bis 6 zu den häufigeren Ergebnissen, jedoch ist das Ergebnis 0 extrem oft vertreten, und es gibt drei Stunden mit mehr als 8 Minuten, in denen Zugriffe stattfanden. Bis auf die Streuung belegen die berechneten Statistiken dagegen den Unterschied nicht.
Es könnte der Hinweis kommen, daß für einen fairen Vergleich die Anzahl der Wiederholungen `r` im letzten Experiment mit der Stichprobe aus einer Binomial-Verteilung auf 24 festgesetzt werden muß. Deshalb wollen wir auch noch ein Bild und die zusammenfassenden Statistiken einer Stichprobe vom Umfang 24 aus *Binomial(60,0.06)* beisteuern.



```
Zusammenfassende Statistiken
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.  n
    1       3      4 3.875       5    7 1.676241 24
```

In dem Bild ist gar kein Ergebnis mit 0 oder mehr als 7 Erfolgen zu erkennen. Damit ist die Spanne viel kleiner als bei den realen Daten, die ja auch verglichen mit dem Binomial-Verteilungsmodell eine viel zu große Streuung besitzen. Gibt es dafür eine Erklärung?

Konzept: Durch ein Gedankenexperiment läßt sich ein Erklärungsversuch entwickeln. Wenn in jeder Stunde eines Tages der Parameter p einen unterschiedlichen Wert annehmen würde, so ergäben sich 24 Realisationen aus 24 verschiedenen Binomial-Verteilungen. Die unterschiedlichen Parameter und damit auch die unterschiedlichen Erwartungswerte der beteiligten Binomial-Verteilungen führen zu einer erhöhten Variabilität für die Mischung und wahrscheinlich auch für die konkrete Stichprobe. Dieses könnte sich in einem weiter auseinandergezogenen Stabdiagramm und einer erhöhten Stichprobenstreuung zeigen. Damit sind genau die Phänomene zu erwarten, die oben aufgedeckt worden sind. Sind die verschiedenen p-Werte sehr unterschiedlich, ist die Binomial-Verteilung kein gutes Modell. Jedoch in Phasen, in denen sich der Parameter p nicht oder kaum ändert, kann sie zur Beantwortung von Fragen über Wahrscheinlichkeiten für Erfolgsanzahlen eingesetzt werden. Auf die Diskussion der Unabhängigkeitsannahme wird an dieser Stelle verzichtet.

Konzept: Wir wollen zum Abschluß dieses Abschnitts noch ein wenig über Genauigkeiten von Abschätzungen nachdenken. Hierdurch soll die oben aufgeworfene Frage zur Genauigkeit der Anteilsschätzung mit beantwortet werden. Anteile ergaben sich aus den Erfolgsanzahlen durch eine Division durch n. Die Erfolgsanzahlen sind bei Gültigkeit der Voraussetzungen binomialverteilt und besitzen eine Varianz von $np(1-p)$. Da gilt: $\mathrm{Var}(aX) = a^2\mathrm{Var}(X)$ ist die Varianz der Anteile gegeben durch

$$\sigma_A^2 = \mathrm{Var}(\text{Anteile}) = \mathrm{Var}(\text{Erfolgsanzahlen}/\mathtt{n}) = np(1-p)/n^2 = p(1-p)/n$$

Als interpretierbares Maß ist die Standardabweichung σ_A, die Wurzel aus der Varianz, besser geeignet: $\sigma_A = \sqrt{p(1-p)/n}$.

Praxis: Für $n = 200$ — wie in den ersten Graphiken — und $p = 0.06$ erhalten wir gerundet einen Wert von $\sigma_A = 0.0168$.

Konzept: Sehr viele Realisationen werden innerhalb des Intervalls $p \pm 2\sigma_A$ liegen. (Diese Konstruktion ist ein durchaus üblicher Vorschlag.)

Praxis: Für $n = 200$ erhält man damit das Intervall (0.0264,0.0936) und für $n = 1440$ (0.0475,0.0725). An diesen Zahlen läßt sich auch ablesen, daß man für eine hohe Genauigkeit (kleine Intervalle) einen großen Stichprobenumfang benötigt.

6.3.3 Ein Kalkulator für die Binomial-Verteilung

Technik: Mit dem Werkzeug `binomial.calculator` lassen sich solche Berechnungen nachrechnen. Versuchen Sie, die Zahlen des letzten Absatzes zu verifizieren. (Vielleicht ist es eine Hilfe zu bemerken, daß der Kehrwert von 1440 ungefähr 0.0006944444 beträgt.)

86 ⟨*aktiviere* `binomial.calculator` 86⟩≡
```
binomial.calculator()
```

Praxis: Mit diesem Werkzeug läßt sich natürlich auch ermitteln, wie groß die Wahrscheinlichkeit dafür ist, daß für $n = 60$ und $p = 0.06$ eine Realisation von 0 oder größer als 6 zustande kommt. Der folgende Report soll das verdeutlichen:

```
[1] "binomial.calculator"
[1] "Bitte n eingeben! (Default: n=1) n=?"
1: 60
[1] "Bitte p eingeben! (Default: p=0.5) p=?"
1: 0.06
 E(X) Var(X)     sigma  E(X)-2sigma E(X)-sigma E(X)+sigma E(X)+2sigma
  3.6  3.384 1.8395652 -0.079130332  1.7604348  5.4395652   7.2791303
[1] "Auswahl von binomial.calculator"
items:
1:n eingeben
2:p eingeben
3:f(x) und F(x) berechnen
4:Quantile berechnen
5:Statistiken von a*X berechnen
6:Plot erstellen
Selection: 3
[1] "Bitte x eingeben! x=?"
1: 0 6
3:
[1] "Werte der Wahrscheinlichkeitsfunktion:"
     x        f(x)
[1,] 0 0.024415814
[2,] 6 0.082667714
[1] "Werte der Verteilungsfunktion:"
     x        F(x)
[1,] 0 0.024415814
[2,] 6 0.932863863
[1] "Auswahl von binomial.calculator"
items:
1:n eingeben
2:p eingeben
3:f(x) und F(x) berechnen
4:Quantile berechnen
5:Statistiken von a*X berechnen
6:Plot erstellen
Selection: 0
```

Die Wahrscheinlichkeit der Realisation 0 beträgt also ungefähr 2.5 % und die von einem Ergebnis größer 6 ungefähr 6.7 %.

Praxis: Wir wollen die erarbeiteten Antworten kurz zusammenfassen. Wenn man nichts über den genauen Zeitpunkt der Stunde und über den Wochentag weiß, wird man zur Beantwortung der Frage *In wie vielen Minuten einer Stunde ist mit Serverzugriffen zu rechnen?* eine Binomial-Verteilung mit den Parametern $p = 0.06$ und $n = 60$ heranziehen. Mit diesem Modell kommt man zu der Antwort, daß ungefähr 3.6 Minuten mit Serverzugriffen zu erwarten sind. In über 90 % der Fälle wird sich bei Richtigkeit des Modells ein Ergebnis zwischen einer und sechs Minuten einstellen. Weiß man mehr über die fragliche Stunde, kann man ein verbessertes p wählen und wird zu besseren Ergebnissen kommen.

Konzept: Die Binomial-Verteilung hat sich im letzten Teil der Ausführung als Instrument zur Abschätzung von der Qualität der Schätzung von p entpuppt. Weitergehende Gedanken sind aber im Rahmen dieser Einführung nicht vorgesehen.

6.4 Die geometrische Verteilung — Wie viele Minuten muß man auf den nächsten Serverzugriff warten?

87 ⟨*1⟩+≡
```
cat("Revbook:4.d3:v.geom:Abschnitt geometrische Verteilung\n")
```

Konzept: Vom Mensch-ärgere-Dich-nicht-Spiel her kennt jeder die Frage: *Wann kommt denn endlich die (nächste) Sechs?* Andere Situationen — wie: sechs Richtige beim Lotto, ein Sonnentag im Urlaub, der nächste Aktienkursanstieg, die erste Einladung zum Vorstellungsgespräch, der erste Nichtausfall eines Formel-1-Rennwagens, das Bestehen einer schweren Klausur, der erste männliche Nachwuchs und so weiter — führen auch zu solchen Wartefragen. Deshalb ist die Beschäftigung mit Warteproblemen an sich überaus praxisrelevant. Wir wollen in diesem Abschnitt einige Eigenschaften der geometrischen Verteilung, die als erstes bei Fragen nach der Anzahl von *Fehlversuchen* bis zum ersten Erfolg ins Spiel gebracht wird, aufdecken und dann zu einer Antwort auf die Frage der Überschrift kommen. Im Rahmen der Diskussion wird ein Geometrischer Verteilungs-Kalkulator und ein Werkzeug zur optischen Prüfung der Verteilungsannahme zur Verfügung gestellt.
In der einfachsten Form der Wartezeitmodellierung wird die Zeit in Versuchen gemessen, und es wird für die Versuche ein Bernoulli-Prozeß unterstellt. In diesem Fall ist Wartezeit bis zum nächsten Erfolg geometrisch verteilt.

Geometrische Verteilung

Sind die Zufallsvariablen $X_1, X_2, \ldots$ mit identischem p unabhängig Bernoulli-verteilt, so ist die Anzahl der Fehlversuche W vor dem ersten Erfolg geometrisch verteilt mit dem Parameter p. W besitzt die Wahrscheinlichkeitsfunktion:

$$f(w) = pq^w \quad w = 0, 1, \ldots, n \quad q = 1 - p \quad ,$$

und es gilt $E(W) = q/p$ und $Var(W) = q/p^2$.

Für $p = 0.5$ müssen wir im Durchschnitt mit einem Fehlversuch (q/p=1) rechnen. Ist $p = 0.1$, folgt: $E(W) = q/p = 0.9/0.1 = 9$ und wir müssen uns auf eine Wartezeit von im Mittel 9 einstellen.

6.4.1 Stichproben aus geometrisch verteilten Grundgesamtheiten

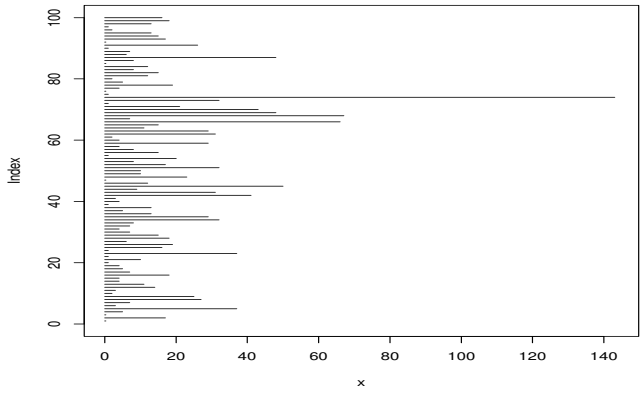
Praxis: Wir wollen zunächst für die Erfahrungssammlung Realisationen aus einer geometrisch verteilten Grundgesamtheit erzeugen und untersuchen. Dazu betrachten wir ein kleines graphisches Experiment, bei dem die Erfolgswahrscheinlichkeit `p`, die Anzahl der Wiederholungen `n` und der Start des Zufallszahlengenerators `ZZ` gewählt werden müssen. Wir wollen im Rahmen der bisherigen Überlegungen `p=0.06` sowie die Defaultwerte `ZZ=17` und `n=100` wählen.

88 ⟨*ziehe* `n` *geometrisch verteilte Zufallszahlen mit* `p`*:* `x` 88⟩≡
```
cat("Stichprobe aus geometrischer Verteilung\n")
print("Welches p wuenschen Sie (Default=0.5)? p=?")
p<-c(scan(,0,n=1),.5)[1]
print("Welche Wiederholungsanzahl n wuenschen Sie (Default=100)? n=?")
n<-c(scan(,0,n=1),100)[1]
print("Welchen Zufallsgeneratorstart ZZ wuenschen Sie (Default=17)? ZZ=?")
ZZ<-c(scan(,0,n=1),17)[1]
set.seed(ZZ)
x<-rgeom(n,p)
```

Nun lassen sich die Wartezeiten als horizontale Linien darstellen.

89 ⟨*plotte* x *als horizontale Stäbe* 89⟩≡

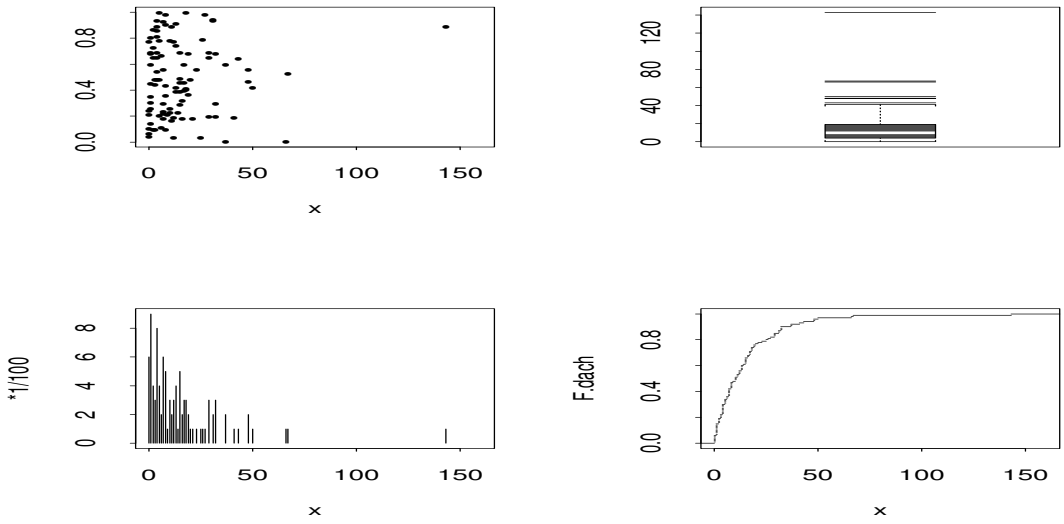
```
cat("Plot von x als Staebe\n")
h<-length(x)
plot(c(0,max(x)+1),c(0,h), type="n",xlab="x",ylab="Index")
segments(0,1:h,x,1:h)
```



Zur Darstellung der simulierten Verteilung helfen einige schon eingesetzte Anweisungen.

90 ⟨* 1⟩+≡

```
par(mfrow=c(2,2))
jitterplot(x); boxplot(x)
stabdiagramm(x); F.dach(x)
par(mfrow=c(1,1))
```



Mit Hilfe der zusammenfassenden Statistiken bekommen wir einen quantitativen Zugang.

```
Zusammenfassende Statistiken
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.   n
    0       4     10 15.52      19  143 19.22513 100
```

Zunächst fällt die starke Asymmetrie der simulierten Verteilung ins Auge. Sowohl Boxplot als auch Jitterplot und Stabdiagramm sprechen eine eindeutige Sprache. In ganz wenigen Fällen muß man sehr lange warten. Da der Median 10 ist, liegt in 50 % die Wartezeit unter 11 Versuchen. Die Differenz zwischen Mittel und Median ist beträchtlich und ist zum Teil dem Ausreißer (?) 143 anzulasten. An dieser Stelle sei daran erinnert, daß die Darstellungen der Zwischenzugriffszeiten zu verwandten Bildern geführt haben. Wir werden darauf zurückkommen.
Falls die Annahmen des Modells zutreffen, kommen wir aufgrund der Simulationen zu der Antwort: Im Durchschnitt müssen wir ungefähr 16 Minuten warten, bis eine Minute mit einem Serverzugriff eintritt. In ungefähr 25 % der Fälle wird die Wartezeit jedoch weniger als 5 Minuten betragen.

Konzept: Ob die Annahme des Verteilungsmodells gerechtfertigt ist, läßt sich einerseits an der Gegenüberstellung von simulierten und empirischen Plots begutachten. Andererseits können dazu spezielle Werkzeuge eingesetzt werden (siehe unten).

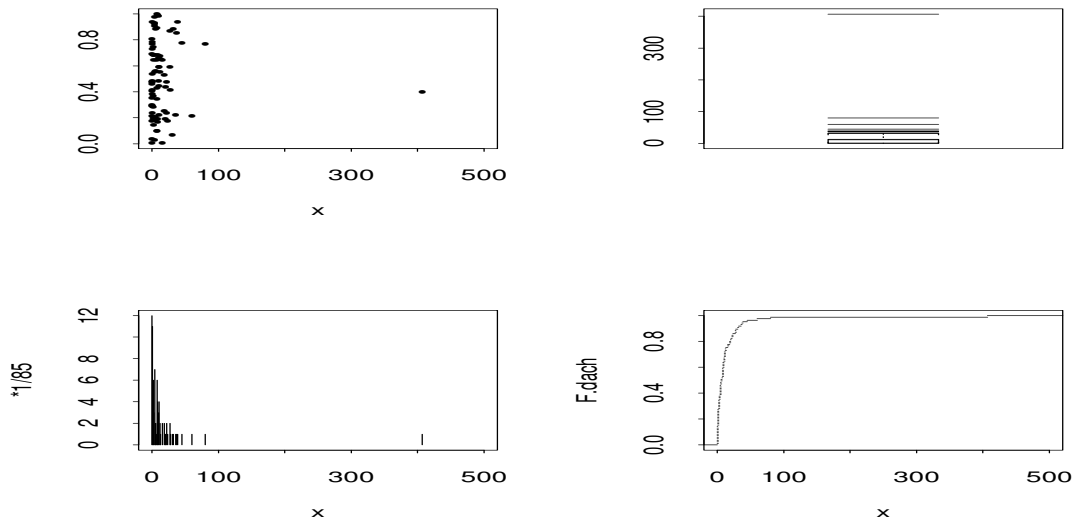
6.4.2 Serverdaten

Praxis: Wie im vorherigen Abschnitt wollen wir zunächst schauen, wie die empirischen Gegenstücke zu den letzten Plots aussehen.

91 ⟨*1⟩+≡

```
cat("Zeitpunkte vom 03.02.97 auf x ablegen\n")
x<-zeitpunkte.03.02.97

cat("Aktivitaet pro Minute aus Sekundendaten ermitteln\n")
x<-(x%%86400)/60
x<-pmin(1,as.vector(table(cut(x,0:1440))))
x<-diff((1:length(x))[x==1])-1

par(mfrow=c(2,2))
jitterplot(x); boxplot(x)
stabdiagramm(x); F.dach(x)
par(mfrow=c(1,1))
```

Wir erhalten:




```
Zusammenfassende Statistiken
 Min. 1st Qu. Median  Mean 3rd Qu. Max. Std.Dev.  n
    0       1      6 15.44      13  407 45.21758 85
```

Die Grundstruktur stimmt in etwa. Jedoch ist die Variabilität doch noch um einiges größer als im Modell. Im besonderen ist das Maximum erheblich vom Zentrum der Daten entfernt.

Konzept: Geht man von $p = 0.06$ aus, so folgt für die Wahrscheinlichkeit einer Wartezeit von 407 Minuten:

$$P(W = 407 \mid p = 0.06) = p(1-p)^{407} = 0.00000000365$$

Dieses relativiert die Brauchbarkeit des Modells wieder ein wenig. Trotzdem wollen wir die geometrische Verteilung noch ein wenig näher studieren.
Da jedoch jedes passendere Modell auch komplizierter werden wird, wollen wir die geometrische Verteilung noch näher studieren.

6.4.3 Ein Kalkulator für die geometrische Verteilung

Technik: Für Wahrscheinlichkeits-Berechnungen wie im letzten Absatz wollen wir wieder ein Kalkulationsprogramm anbieten. Probieren geht hier über studieren!

92 ⟨*aktiviere* `geometric.calculator` 92⟩≡

```
geometric.calculator()
```

6.4.4 Ein Erkennungsplot zur geometrischen Verteilung

Konzept: Als Alternative zur Überprüfung, ob das geometrische Verteilungsmodell paßt, wollen wir noch ein Werkzeug vorstellen. Für dieses verwenden wir zum einen eine interessante Eigenschaft der geometrischen Verteilung:

Die geometrische Verteilung besitzt kein Gedächtnis!

Bei dieser Aussage handelt es sich um eine verbale Interpretation eines formalen Zusammenhangs, denn für die geometrische Verteilung gilt für $x \geq x_0$:

$$\begin{aligned}
P(W \leq x \mid W \geq x_0) &= 1 - P(W > x \mid W \geq x_0) \\
&= 1 - \frac{P(W > x \wedge W \geq x_0)}{P(W \geq x_0)} \\
&= 1 - \frac{q^{-x_0} P(W > x)}{q^{-x_0} P(W \geq x_0)} \\
&= 1 - \frac{q^{-x_0} \sum_{i=x+1}^{\infty} pq^i}{q^{-x_0} \sum_{i=x_0}^{\infty} pq^i} \\
&= 1 - \frac{\sum_{i=x+1}^{\infty} pq^{i-x_0}}{\sum_{i=x_0}^{\infty} pq^{i-x_0}} \\
&= 1 - \frac{\sum_{i=x+1-x_0}^{\infty} pq^i}{\sum_{i=x_0-x_0}^{\infty} pq^i} \\
&= 1 - \frac{P(W > x - x_0)}{1} \\
&= P(W \leq x - x_0)
\end{aligned}$$

Wenn man also schon $x_0 - 1$ Versuche erfolglos gewartet hat, ist die Wahrscheinlichkeit, weniger als weitere $x - x_0$ Versuche warten zu müssen, genauso groß wie die Wahrscheinlichkeit, ohne Bedingung $x - x_0$ Versuche zu warten.
Zum anderen gilt:

$$P(W = x) = pq^x \quad \Rightarrow \quad \ln(P(W = x)) = \ln(p) + \ln(1 - p) * x$$

Hiernach müßten die logarithmierten relativen Häufigkeiten, berechnet aus einer Stichprobe einer geometrisch verteilten Grundgesamtheit, gegen die Anzahl der Fehlversuche `x` geplottet um eine Gerade mit dem Achsenabschnitt $a = \log(p)$ und der Steigung $b = \log(1 - p)$ streuen. Anhand des Plots können damit schon völlig zur geometrischen Verteilung unpassende Datensätze aussortiert werden.
Entfernt man dann die Wartezeitbeobachtungen, die kleiner als ein vorgegebener Wert x_0 sind, so muß sich für den Rest nach Verschiebung um x_0 wieder ein in etwa ähnliches Bild mit einer ähnlich gelagerten Geraden ergeben.

Praxis: Wir wollen dieses Verfahren zuerst auf Daten mit bekannten Eigenschaften anwenden. Dazu ziehen wir eine Stichprobe vom Umfang `n` aus einer geometrisch verteilten Grundgesamtheit mit dem Parameter `p`, erstellen den beschriebenen Plot und schätzen aus dem Plot p ab. Dann wollen wir die Überschreitung der Wartezeit `x.0` betrachten und Plot und Abschätzung wiederholen. Es müßte im wesentlichen dasselbe herauskommen.

Technik: Die Experimentparameter `n`, `p`, `x.0` und `ZZ` werden bei Ausführung der folgenden Anweisungen erfragt.

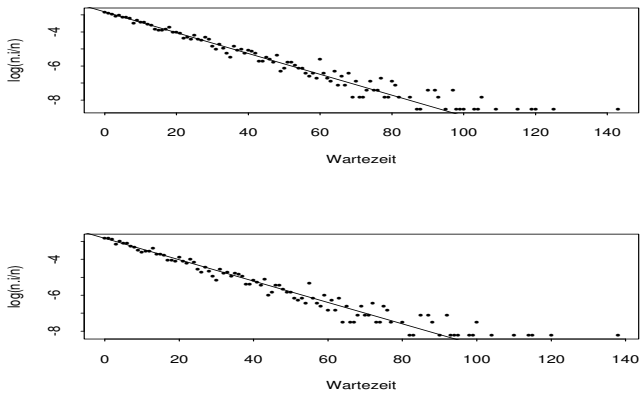
93 ⟨*bereite Gedächnisexperiment vor und führe es durch* 93⟩≡

```
cat("Stichprobe aus geometrischer Verteilung\n")
print("Welches p wuenschen Sie (Default=0.5)? p=?")
p<-c(scan(,0,n=1),.5)[1]
print("Welche Wiederholungsanzahl n wuenschen Sie (Default=100)? n=?")
n<-c(scan(,0,n=1),100)[1]
print("Welchen Zufallsgeneratorstart ZZ wuenschen Sie (Default=17)? ZZ=?")
ZZ<-c(scan(,0,n=1),17)[1]
set.seed(ZZ)
x<-rgeom(n,p)

print("Wie gross soll x.0 sein (Default: x.0=5)?")
x.0<-c(scan(,0,n=1),5)[1]
par(mfrow=2:1)

cat("Erkennungsplot fuer geometrische Verteilung\n")
print(geom.p.est(x))

x<-x[x>=x.0] - x.0
print("Plot und Daten nach Modifikation")

cat("Erkennungsplot fuer geometrische Verteilung\n")
print(geom.p.est(x))

par(mfrow=c(1,1))
```

Praxis: Mit den Setzungen

```
Stichprobe aus geometrischer Verteilung
[1] "Welches p wuenschen Sie (Default=0.5)? p=?"
1: .06
[1] "Welche Wiederholungsanzahl n wuenschen Sie (Default=100)? n=?"
1: 5000
[1] "Welchen Zufallsgeneratorstart ZZ wuenschen Sie (Default=17)? ZZ=?"
1: 17
[1] "Wie gross soll x.0 sein (Default: x.0=5)?"
1: 5
```

erhalten wir folgende Ergebnisse:




```
[1] "Plot und Daten aufgrund der Stichprobe"
 p aus Achsenabschnitt p aus Steigung
            0.05957634     0.05932194
[1] "Plot und Daten nach Modifikation"
 p aus Achsenabschnitt p aus Steigung
            0.05901169     0.05779475
```

Zum ersten Plot läßt sich feststellen, daß sehr viele Punkte ganz gut in der Nähe einer Geraden liegen. Dies kann als Beleg der Eignung des geometrischen Modells für die Stichprobendaten gelten. Im Bereich kleiner Wartezeiten streuen die Punkte weniger als in demjenigem großer Zeiten. Die Schätzungen aus dem ersten Plot kommen dem vorgegebenen p sehr nahe.
Die Ergebnisse zu dem zweiten Plot sind qualitativ nicht von den ersten zu unterscheiden. Also auch zu der modifizierten Stichprobe paßt — wie erwartet — ein geometrisches Modell und die Schätzungen des unbekannten Parameters fallen sehr ähnlich aus.

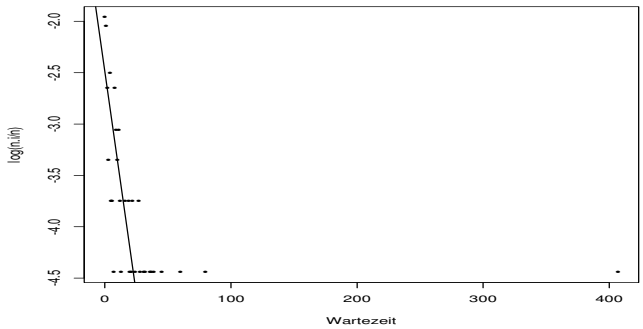
6.4.5 Serverdaten

Konzept: Mit diesem Werkzeug wollen wir zum Abschluß der Diskussion der geometrischen Verteilung prüfen, wie gut die Wartezeit in Minuten auf die nächste Minute mit Serverzugriff zum geometrischen Modell paßt.

Praxis:

94 ⟨*1⟩+≡

```
cat("Zeitpunkte vom 03.02.97 auf x ablegen\n")
x<-zeitpunkte.03.02.97

cat("Aktivitaet pro Minute aus Sekundendaten ermitteln\n")
x<-(x%%86400)/60
x<-pmin(1,as.vector(table(cut(x,0:1440))))
x<-diff((1:length(x))[x==1])-1

cat("Erkennungsplot fuer geometrische Verteilung\n")
print(geom.p.est(x))
```




```
[1] "Plot und Daten aufgrund der Stichprobe"
 p aus Achsenabschnitt p aus Steigung
           0.083944023    0.082995951
```

Die Punkte liegen nicht sehr überzeugend auf einer Geraden. Besonders der Ausreißer jenseits 400 paßt nicht ins Bild. Er wird natürlich wieder durch die lange Nachtpause verursacht.

Konzept: Es sei angemerkt, daß sich die gewählte Prozedur zur Plazierung der Geraden von den Ausreißern nicht beeinflussen läßt. Bei dem oben verwendeten Vorschlag $p = 0.06$ ging die lange Wartezeit jedoch ein, so daß hier das etwas größere Ergebnis 0.084 durchaus plausibel ist.

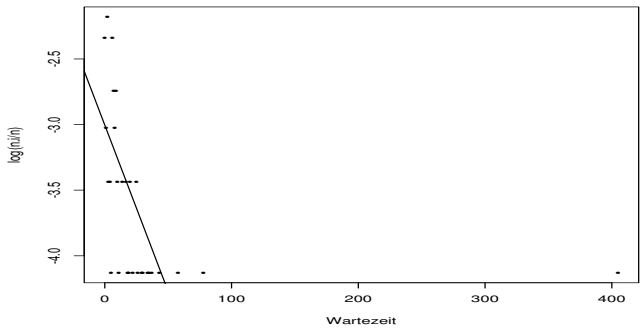
Praxis: Nun wollen wir die Datenreduktions-Idee umsetzen.

95 ⟨*1⟩+≡
```
cat("Zeitpunkte vom 03.02.97 auf x ablegen\n")
x<-zeitpunkte.03.02.97

cat("Aktivitaet pro Minute aus Sekundendaten ermitteln\n")
x<-(x%%86400)/60
x<-pmin(1,as.vector(table(cut(x,0:1440))))
x<-diff((1:length(x))[x==1])-1

print("Wie gross soll x.0 sein (Default: x.0=5)?")
x.0<-c(scan(,0,n=1),5)[1]
x<-x[x>=x.0] - x.0

cat("Erkennungsplot fuer geometrische Verteilung\n")
print(geom.p.est(x))
```

Wie man schnell durch Probieren herausfindet, macht nur ein kleiner `x.0`-Wert Sinn. Für die Wahl `x.0=2`:

```
Zeitpunkte vom 03.02.97 auf x ablegen
Serverzugriff pro Minute ermitteln
[1] "Welchen Wert fuer x.0 wuenschen Sie?"
1: 2
```

erhalten wir



sowie

```
Erkennungsplot fuer geometrische Verteilung
[1] "Plot und Daten aufgrund der Stichprobe"
 p aus Achsenabschnitt p aus Steigung
           0.049628984    0.025023162
```

Das Verfahren errechnet jetzt ganz andere Werte. Damit ist wieder eine gewisse Skepsis gegen die Modellierung mittels einer geometrischen Verteilung angebracht. Es sollte bei weitergehenden Interpretationen berücksichtigt werden, daß im Simulationsexperiment ein Stichprobenumfang von 5000 gewählt werden konnte, nach der Modifikation mit `x.0=2` für den letzten Plot aber nur noch 63 Daten zur Verfügung stehen.
Besonders die letzten beiden Bilder regen sicher dazu an, die Modellierung zu verbessern. Wenn man sich vor Augen hält, daß die ursprünglichen Sekunden-Daten in Minuteninformationen überführt worden sind, könnte eine Modellierung mit feineren Zeiteinteilungen Verbesserungen bringen. Im nächsten Abschnitt wird diese Idee durch Unterstellung einer kontinuierlichen Zeit umgesetzt. Damit entfällt dann auch das Problem, mehrere (Zugriffs-)Ereignisse in einem Intervall nicht auswerten zu können.

In diesem Abschnitt wurde verschiedenes Material zur geometrischen Verteilung zur Verfügung gestellt. Mit diesem lassen sich geometrisch verteilte Zufallszahlen erzeugen und darstellen. Der `geometric.calculator` ermöglicht Berechnungen rund um die geometrische Verteilung. Ein graphisches Werkzeug zur Prüfung des Verteilungsmodells wurde vorgestellt. Zur Frage der Überschrift *Wie viele Minuten muß man auf den nächsten Serverzugriff warten?* läßt sich antworten: Nach dem angepaßten Modell im Mittel etwas mehr als eine Viertelstunde, wobei in 50 % der Fälle 10 Minuten ausreichen sollten. Diese Ergebnisse werden dadurch relativiert, daß die Daten zum Teil erheblich unterschiedlicher Eigenschaften aufweisen als aus einer geometrischen Verteilung gezogene Daten.

6.5 Die Exponentialverteilung — Wie lange muß man auf den nächsten Zugriff warten?

96 ⟨* 1⟩+≡
```
cat("Revbook:4.d4:v.exp:Abschnitt Exponentialverteilung\n")
```

Konzept: Die Frage der Überschrift zwingt uns zur Auseinandersetzung mit einem kontinuierlichen Zeitbegriff. Wurde im letzten Abschnitt *Zeit* als eine Abfolge von Minuten aufgefaßt, wird sie nun als kontinuierliche Größe angenommen. Dieses führt formal zu einem Wechsel von diskreten zu stetigen Verteilungen.
In diesem Abschnitt wird als Pendant zur geometrischen Verteilung die Exponentialverteilung vorgestellt. Diese Verteilung eignet sich dazu, Wartezeitsituation zu modellieren, für die einige einfache Bedingungen erfüllt sind. Zunächst wird die Exponentialverteilung motiviert, dann werden Stichproben aus exponentialverteilten Grundgesamtheiten untersucht und ein Kalkulator für die Exponentialverteilung angeboten. Für die Zwischenzeiten des Servers wird untersucht, inwieweit eine Modellierung mit Hilfe der Exponentialverteilung angemessen ist und was letztlich auf die gestellte Frage geantwortet werden kann.
Die geometrische Verteilung zeichnet sich durch zwei sich bedingende Eigenschaften aus:

1. Die Wahrscheinlichkeiten der geometrischen Verteilung bilden eine geometrische Folge.
2. Die geometrische Verteilung besitzt kein Gedächnis.

Eine stetige Verteilung mit entsprechenden Eigenschaften könnte ein geeignetes Modell zur Modellierung von Wartezeiten sein. Wir wollen eine solche Verteilung aufspüren. Aus der Umformung

$$pq^x = pe^{\ln q^x} = pe^{x \ln q} = pe^{dx}$$

erkennt man, daß die Wahrscheinlichkeiten der geometrischen Verteilung auf einer Exponentialfunktion liegen. Es ist naheliegend, als Dichte $f(x) = pe^{dx}$ zu probieren. Damit $f(x)$ eine Dichte ist, muß gelten:

$$\int_0^\infty f(x)dx = 1 \qquad \Rightarrow \qquad p > 0 \quad \text{und} \quad d = -p$$

Verwenden wir statt der Konstanten c den griechischen Buchstaben λ, so erhalten wir die übliche Schreibweise der Dichte der Exponentialverteilung.

Exponentialverteilung

Eine Zufallsvariablen X mit der Dichte

$$f(x) = \lambda e^{-\lambda x} \qquad x \geq 0$$

heißt exponentialverteilt mit dem Parameter $\lambda > 0$.
Es gilt: $F(x) = 1 - e^{-\lambda x}$, $E(X) = 1/\lambda$ *und* $Var(X) = 1/\lambda^2$.

Wir wollen prüfen, ob diese Verteilung auch kein *Gedächnis* besitzt. Für $x > x_0$ ergibt sich:

$$\begin{aligned} P(X \leq x \mid X > x_0) &= 1 - P(X > x \mid X > x_0) \\ &= 1 - \frac{P(X > x)}{P(X > x_0)} \\ &= 1 - \frac{e^{-\lambda x}}{e^{-\lambda x_0}} \\ &= 1 - e^{-\lambda(x - x_0)} \\ &= F(x - x_0) \\ &= P(X \leq x - x_0) \end{aligned}$$

Damit besitzt die Exponentialverteilung die gewünschten Eigenschaften.

6.5.1 Stichproben aus dem Exponential-Modell

Praxis: Wir wollen zunächst eine Stichprobe aus einer exponentialverteilten Grundgesamtheit ziehen und untersuchen.

97 ⟨*ziehe Stichprobe aus einer Exponentialverteilung mit λ:* x 97⟩≡

```
cat("Stichprobenziehung -- Exponentialverteilung\n")
print("Welches lambda wuenschen Sie (Default=1)? lambda=?")
lambda<-c(scan(,0,n=1),1)[1]
print("Welchen Stichprobenumfang n wuenschen Sie (Default=100)? n=?")
n<-c(scan(,0,n=1),100)[1]
print("Welchen Zufallsgeneratorstart ZZ wuenschen Sie (Default=17)? ZZ=?")
ZZ<-c(scan(,0,n=1),17)[1]
set.seed(ZZ)
x<-rexp(n,lambda)
```

Mit den Setzungen

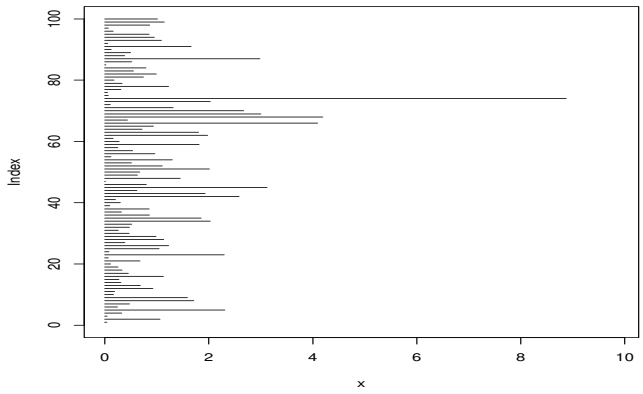
```
Stichprobenziehung -- Exponentialverteilung
[1] "Welches lambda wuenschen Sie (Default=1)? lambda=?"
[1] 1
[1] "Welchen Stichprobenumfang n wuenschen Sie (Default=100)? n=?"
[1] 100
[1] "Welchen Zufallsgeneratorstart ZZ wuenschen Sie (Default=17)? ZZ=?"
[1] 17
```

erhalten wir aufgrund der Anweisungen

98 ⟨* 1⟩+≡

```
cat("Plot von x als Staebe\n")
h<-length(x)
plot(c(0,max(x)+1),c(0,h), type="n",xlab="x",ylab="Index")
segments(0,1:h,x,1:h)
```

den Plot Output:



Zur ersten Untersuchung der Stichprobe lassen sich bereits bekannte Anweisungen verwenden.

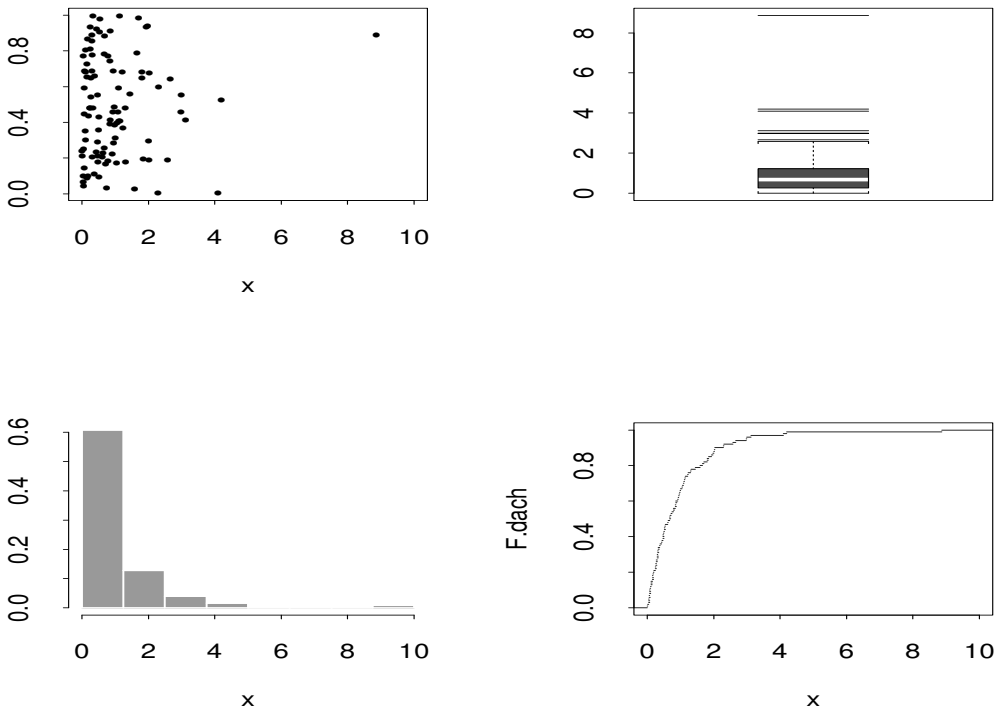
99 ⟨*plotte Stichprobe, berechne Statistiken zu* x *— kontinuierlich* 99⟩≡

```
par(mfrow=c(2,2))
jitterplot(x);boxplot(x);histogramm(x);F.dach(x)
par(mfrow=c(1,1))
summary.stats(x)
```

Wir erhalten als

```
Zusammenfassende Statistiken
      Min. 1st Qu. Median   Mean 3rd Qu.  Max. Std.Dev.   n
 0.0001549  0.2668 0.6735 0.9914   1.225 8.872 1.188518 100
```

und als Plot:



6.5.2 Ein Kalkulator für die Exponentialverteilung

Für Berechnungen rund um die Exponentialverteilung ist ein Werkzeug wünschenswert, mit dem sich Kalkulationen anstellen lassen. Hier ist es.

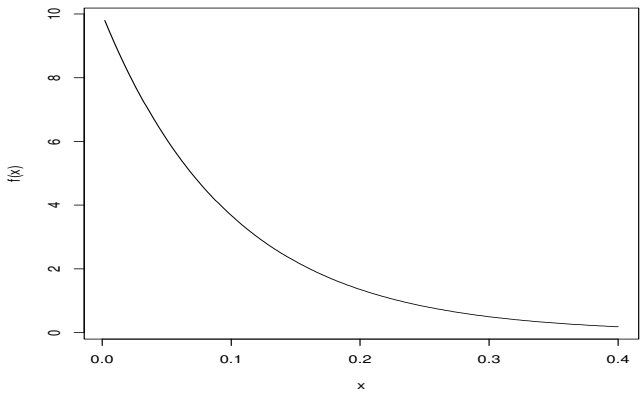
100 ⟨*1⟩+≡

```
cat("Exponential-Kalkulator\n")
exponential.calculator()
```

Die Auswahlpunkte, die dieses Werkzeug bietet, lassen sich aus folgendem Protokoll entnehmen.

```
[1] "exponential.calculator start"
[1] "Bitte lambda eingeben! (Default: lambda=1) lambda=?"
1: 10
E(X) Var(X) sigma E(X)-2sigma E(X)-sigma E(X)+sigma E(X)+2sigma
 0.1   0.01   0.1        -0.1          0        0.2         0.3
[1] "Auswahl von exponential.calculator"
items:
1:lambda eingeben
2:f(x) und F(x) berechnen
3:Quantile berechnen
4:Statistiken berechnen
5:Plot erstellen
Selection: 2
[1] "Bitte x eingeben! x=?"
1: 0.1 0.5 1
4:
[1] "Werte der Dichtefunktion und der Verteilungsfunktion:"
       x          f(x)       F(x)
[1,] 0.1 3.6787944e+00 0.63212056
[2,] 0.5 6.7379470e-02 0.99326205
[3,] 1.0 4.5399930e-04 0.99995460
[1] "Auswahl von exponential.calculator"
items:
1:lambda eingeben
2:f(x) und F(x) berechnen
3:Quantile berechnen
4:Statistiken berechnen
5:Plot erstellen
Selection: 5
```

Die letzte Anweisung liefert eine Darstellung der Dichte.



6.5.3 Ein Erkennungsplot zur Exponentialverteilung

Konzept: Nachdem wir uns mit dem bisherigen Angebot einen Eindruck von dem Verteilungsmodell und von Stichproben aus exponentialverteilten Grundgesamtheiten verschaffen konnten, wollen wir schauen, ob die Exponentialverteilung ein geeignetes Modell für die Zwischenzugriffszeiten vom 03.02.97 darstellt. Aufgrund der folgenden Umformungen erhalten wir ein Instrument für diese Fragen.

$$F(x) = 1 - e^{-\lambda x} \quad \Rightarrow \quad 1 - F(x) = e^{-\lambda x} \quad \Rightarrow \quad -\ln(1 - F(x)) = \lambda x$$

Setzt man für $F(x)$ die empirische Verteilungsfunktion ein und plottet die linke Seite gegen x, also $-\log(1 - \hat{F}(x))$ gegen die sortierten Beobachtungen, so werden bei Eignung der Exponentialverteilung die Punkte nahe einer Geraden liegen. Die Steigung der Gerade sollte in etwa λ entsprechen.

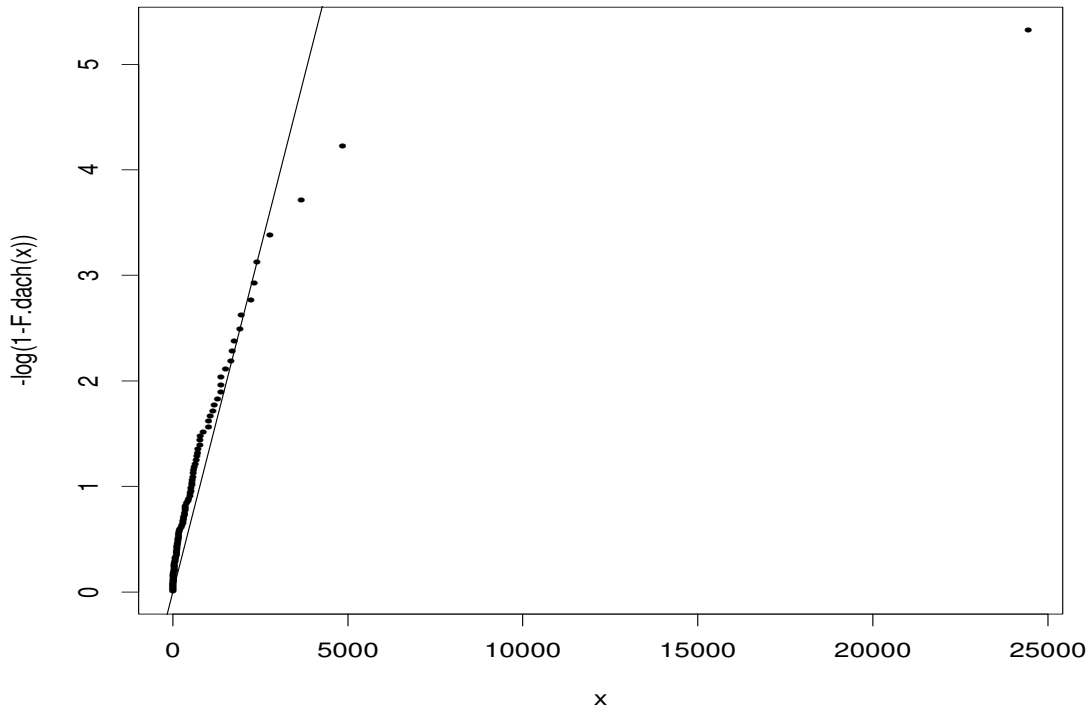
6.5.4 Serverdaten

Praxis: In Anlehnung an diesen Gedankengang können wir mit den folgenden Anweisungen einen Erkennungsplot für die Zwischenzeiten vom 03.02.97 erstellen. Dabei werden für die Geradenschätzung nur die 80 % kleineren Beobachtungen verwendet.

101 ⟨*erstelle für Zwischenzeiten Erkennungsplot zur Exponentialverteilung* 101⟩≡

```
x<-dzeitpunkte.03.02.97
cat("Exponential-Erkennungsplot\n")
print(identify.exp(x))
```

Die Anweisungen führen zu dem Bild:



sowie einem Vorschlag für λ:

```
Exponential-Erkennungsplot
 lambda.dach
  0.001908848
```

Wieder sehen wir ein Bild, bei dem der bekannte Ausreißer eine gute Ausnutzung der Darstellungsfläche verhindert. Die Punkte liegen zum großen Teil in der Nähe der eingezeichneten Geraden.

Konzept: Wir wollen den vorgeschlagenen Wert für λ näher unter die Lupe nehmen. λ läßt sich als Rate, als durchschnittliche Anzahl von Ereignissen interpretieren, die pro Zeiteinheit eintreten. Die *kleine* Größenordnung der Zahlen sollte übrigens keine Bedenken erregen, da eine Umskalierung der Zeitachse Abhilfe schaffen kann. Würden wir als Einheit Stunden wählen, so erhielten wir z.B. die Steigungen $3600 \times 0.0019 = 6.84$. Bei $\lambda = 6.84$ sind 6.84 Ereignisse pro Zeiteinheit (Stunde) zu erwarten. Als Wartezeit ist der Kehrwert der Rate λ zu erwarten: $1/0.0019 = 526.32$ Sekunden $= 8.77$ Minuten. Ist λ groß, muß man in der Regel nur kurz auf das nächste Ereignis warten. Ist λ klein, muß man eher länger warten und die Anzahl von eintreffenden Ereignissen pro Zeiteinheit ist eher klein. Über den ganzen Tag gesehen erhalten wir also ein Rate von knapp 7 Ereignissen pro Stunde. Hierbei sind — wie oben erwähnt — die 20 % längsten Pausen außer acht gelassen worden.
Nach den letzten Überlegungen müßte der Durchschnitt der Wartezeiten in dem Bereich um 500 Sekunden liegen. Wir wollen deshalb schnell noch einmal die wesentlichen Statistiken zu den Zwischenzeiten vom 03.02.97 auflisten.

102 ⟨* 1⟩+≡
```
summary.stats(x)
```

```
 Min. 1st Qu. Median  Mean 3rd Qu.  Max. Std.Dev.   n
    1    62.5    312 823.8   744.5 24450 2487.414 103
```

In der Tat liegt der oben berechnete Wert zwischen dem Median und dem Mittel des Datensatzes. Berechnet man umgekehrt den Kehrwert des Mittels, so erhält man $1/823.8 = 0.0012$. Zur Rate 0.0012 gehört also eine durchschnittliche Wartezeit von 823.8 Sekunden oder zirka 13.7 Minuten. Über das Mittel aller Zwischenzeiten würde man also eine Rate 0.0012 vorschlagen.
Die Erkenntnis, die hinter den Berechnungen steckt, mag als eine Wiederholung schon bekannter Dinge angesehen werden. Jedoch können wir nun begrifflich besser mit den Phänomenen umgehen. Die Belastungsrate verkörpert die konzentrierte Antwort auf die zugrundeliegende Belastungsfrage.
Mit einem Exponential-Modell und einer Belastungsrate von λ=0.0019 können wir die Daten (bezogen auf Sekunden) einigermaßen beschreiben. Leider befriedigt dieses noch nicht völlig, da die Unterschiedlichkeit der Belastung über den Tag hinweg noch nicht faßbar wird. Verfeinerungen könnten darin bestehen, eine kompliziertere Verteilung heranzuziehen oder aber die Erkenntnis umzusetzen, daß es über einen Tag hinweg verschiedene Belastungsraten gibt. Wir wollen den zweiten dieser beiden Wege gehen und die Belastungsrate im Zeitablauf abschätzen.
Um sich verändernde Raten zu ermitteln, benötigen wir noch ein weiteres Instrument. Der hier angebotene Vorschlag geht nach folgenden Schritten vor:

1. Betrachte `anz` aufeinanderfolgende Zugriffszeitpunkte.
2. Ermittle die Zeitspanne vom ersten bis zum `anz`-ten Ereignis.
3. Berechne hieraus eine Belastungsrate als Kehrwert der Zeitspanne.
4. Wende die ersten drei Punkte auf alle möglichen Mengen von `anz` aufeinanderfolgenden Zugriffszeitpunkten an.
5. Wiederhole die ersten vier Punkte für unterschiedliche `anz`-Werte.

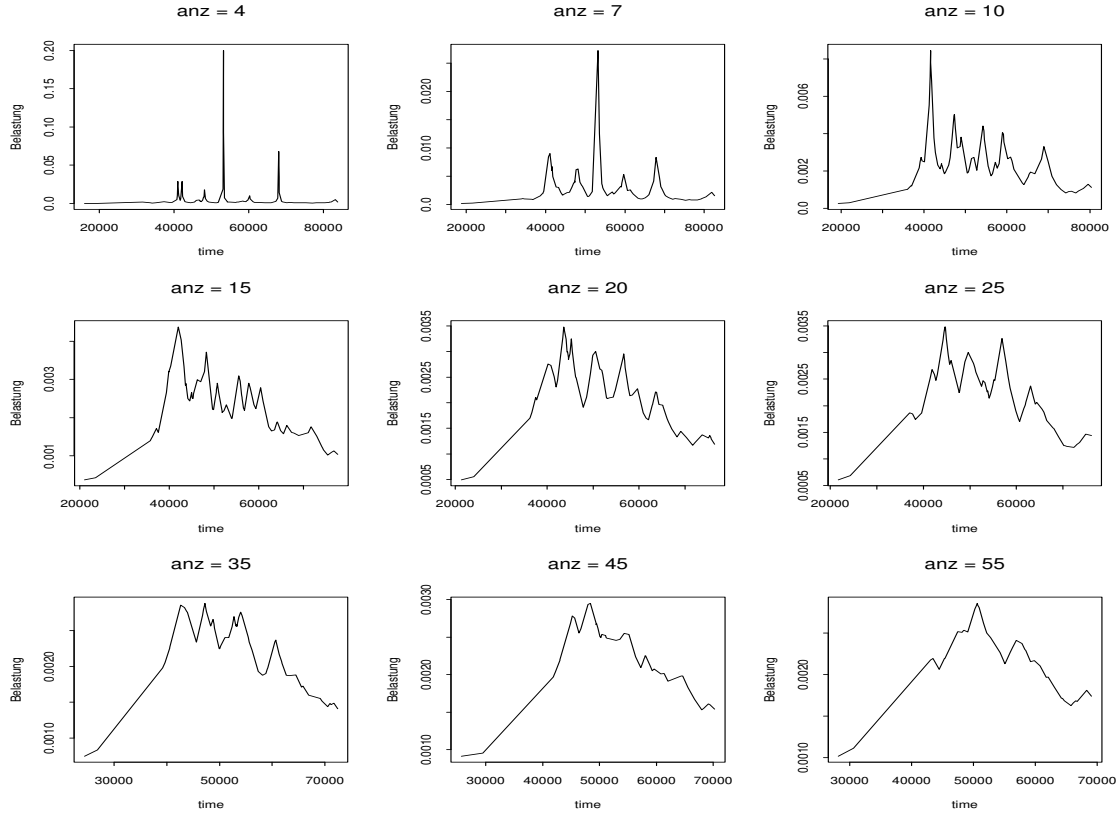
Die Idee wird in den folgenden Anweisungen umgesetzt. Die gewünschten Anzahlen `anz` sind einzugeben. Die berechneten Belastungen werden dann graphisch dargestellt.

103 ⟨*ermittele Belastung anhand zusammenhängender Zugriffszeitpunkte* 103⟩≡
```
interval.est.lambda(x)
```

Praxis: Mit den folgenden, systematisch variierten Anzahlen:

```
Belastungsrate im Zeitablauf
[1] "Geben Sie die Zahl der Ereignisse an, die zusammen betrachtet"
[1] "werden sollen! Es duerfen auch mehrere Werte (maximal 9 Werte)"
[1] "Zahlen eingegeben werden. (Default: 5 10 20)"
1: 4 7 10 15 20 25 35 45 55
```

erhalten wir 9 Plots:



Konzept: Die Plots zeigen Belastungsschätzungen im Zeitablauf an. Wir sehen, daß die Anzahl der berücksichtigten Ereignisse einen großen Einfluß auf das Ergebnis hat. Je geringer die Anzahl `anz`, desto spitzer und höher werden die Spitzen. Je größer die Anzahl `anz` gewählt wird, umso glatter und flacher wird der Plot. Wird sie zu klein gewählt, wird das Bild von zufällig nahe beieinanderliegenden Ereignissen beherrscht. Ist die Anzahl zu groß, werden Belastungswechsel verdeckt. Leider scheint es keine ausgezeichnete oder richtige Anzahl zu geben.
An dieser Stelle könnte das Werkzeug noch weiteren Verbesserungen unterzogen und es könnte auch ein Bezug zum Dichteschätzinstrument am Ende des Kapitels über multivariate Beschreibungstechniken gezogen werden. Dieses würde jedoch zu umfangreich sein und unterbleibt deshalb.

Praxis: Wir wollen zum Abschluß dieses Abschnitts die gestellte Frage beantworten: *Wie lange muß man auf den nächsten Zugriff warten?* Aufgrund des Datenmaterials vom 03.02.97 läßt sich zunächst entgegnen, daß die Belastung und damit die Wartezeit stark von der Tageszeit abhängt. Unterbleibt eine nähere Zeitangabe, lautet die Antwort im Mittel 13.7 Minuten. Wird nach der Spitzenbelastung, also nach der Belastung zur Mittagszeit gefragt, kommen wir auf ein $\lambda = 0.0035$ und damit auf eine durchschnittliche Wartezeit von $\lambda^{-1} = 285$ Sekunden oder 4 Minuten und 45 Sekunden.

6.6 Die Poisson-Verteilung — Wie viele Zugriffe werden sich in den nächsten 60 Minuten einstellen?

104 ⟨*1⟩+≡
```
cat("Revbook:4.d5:v.pois:Abschnitt Poisson-Verteilung\n")
```

Für einen Systemverwalter ist die Frage nach der Wartezeit bis zum nächsten Ereignis höchstens eine vorgelagerte Frage. Wichtiger ist ihm, in welchen Zeiträumen sich wieviel tut. Anhand der Belastungsrate im Zeitablauf kann man schon stark belastete Zeiten ausmachen. Jedoch fehlt noch die Abschätzung für die Stärke der Beanspruchung oder spezieller für Ereignishäufigkeiten in vorgegebenen Zeitintervallen.
Schaut man zurück, so stellt man fest, daß in dem Abschnitt zur Binomial-Verteilung eine entsprechende Frage beantwortet wurde. Die Binomial-Verteilung liefert Wahrscheinlichkeiten für die Anzahlen von Erfolgen bei einer festen Anzahl von Versuchen. Hier interessieren wir uns für die Wahrscheinlichkeiten der Anzahlen von Ereignissen in einem festen Zeitintervall. Die gesuchte Verteilung müßte sich also wie folgt ergeben:

1. Zerlege das Zeitintervall in kleinere Intervalle,
2. Ermittele Erfolgswahrscheinlichkeiten für die kleineren Intervalle,
3. Bestimme Wahrscheinlichkeit für die Erfolgsanzahl,
4. Lasse die Länge der kleineren Intervalle gegen 0 gehen.

Bei diesem Prozeß wächst in der üblichen Symbolik der Binomial-Verteilung n an. Entsprechend muß p_n, die Wahrscheinlichkeit, in einem kleinen Intervall ein Ereignis anzutreffen, kleiner werden. Da np_n die erwartete Anzahl der Erfolge im Gesamtintervall ist, wollen wir sie auf λ festsetzen. Dann folgt formal $p_n = \lambda/n$ und

$$\begin{aligned} f(x \mid n, p_n) = \binom{n}{x} p_n^x (1-p_n)^{n-x} &= \frac{n!}{x!(n-x)!} \left(\frac{\lambda}{n}\right)^x \left(1-\frac{\lambda}{n}\right)^{n-x} \\ &= \frac{\lambda^x}{x!} \frac{n!}{(n-x)!n^x} \left(1-\frac{\lambda}{n}\right)^n \left(1-\frac{\lambda}{n}\right)^{-x} \end{aligned}$$

Geht jetzt $n \to \infty$, folgt:

$$\begin{aligned} \frac{n!}{(n-x)!n^x} = \frac{n}{n}\frac{(n-1)}{n}\frac{(n-2)}{n}\cdots\frac{(n-x+1)}{n}\frac{(n-x)!}{(n-x)!} &\to 1 \\ \left(1-\frac{\lambda}{n}\right)^n = \left(\left(1+\frac{1}{n/(-\lambda)}\right)^{n/(-\lambda)}\right)^{-\lambda} &\to e^{-\lambda} \\ \left(1-\frac{\lambda}{n}\right)^{-x} &\to 1 \end{aligned}$$

Insgesamt erhält man also:

$$\lim_{n\to\infty, p_n\to 0, np_n\to\lambda} \binom{n}{x} p_n^x (1-p_n)^{n-x} = \frac{\lambda^x}{x!} e^{-\lambda}$$

Dieses ist die Wahrscheinlichkeitsfunktion der Poisson-Verteilung.

Poisson-Verteilung

Eine Zufallsvariablen X mit der Wahrscheinlichkeitsfunktion

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} \qquad x = 0, 1, \ldots \qquad \lambda > 0$$

heißt Poisson-verteilt mit dem Parameter λ.
Es gilt: $E(X) = \lambda$ und $Var(X) = \lambda$.

Sind die Zwischenankunftszeiten exponentialverteilt mit dem Parameter λ, so lassen sich die Wahrscheinlichkeiten für die Anzahl von Ereignissen in einem Zeitintervall mit Hilfe der Poisson-Verteilung mit Parameter λ berechnen. Dabei ist die Länge des Zeitintervalls durch das Intervall vorgegeben, auf das sich die Rate der Exponentialverteilung bezieht.

6.6.1 Stichproben aus dem Poisson-Modell

Auch hier wollen wir die Möglichkeit geben, Stichproben aus dem vorgestellten Modell zu ziehen.

105 ⟨*ziehe Stichprobe aus einer Poisson-Verteilung mit λ:* x 105⟩≡

```
cat("Stichprobenziehung -- Poisson-Verteilung\n")
print("Welches lambda wuenschen Sie (Default=1)? lambda=?")
lambda<-c(scan(,0,n=1),1)[1]
print("Welchen Stichprobenumfang n wuenschen Sie (Default=100)? n=?")
n<-c(scan(,0,n=1),100)[1]
print("Welchen Zufallsgeneratorstart ZZ wuenschen Sie (Default=17)? ZZ=?")
ZZ<-c(scan(,0,n=1),17)[1]
set.seed(ZZ)
x<-rpois(n,lambda)
```

Wir wollen im Anschluß an obige Diskussionen einmal einen großen Wert für λ wählen und die Auswirkungen verfolgen. Genauer wähle man:

```
Stichprobenziehung -- Poisson-Verteilung
[1] "Welches lambda wuenschen Sie (Default=1)? lambda=?"
1: 10
[1] "Welchen Stichprobenumfang n wuenschen Sie (Default=100)? n=?"
1: 200
[1] "Welchen Zufallsgeneratorstart ZZ wuenschen Sie (Default=17)? ZZ=?"
1: 17
```

Zur Untersuchung der Stichprobe hilft:

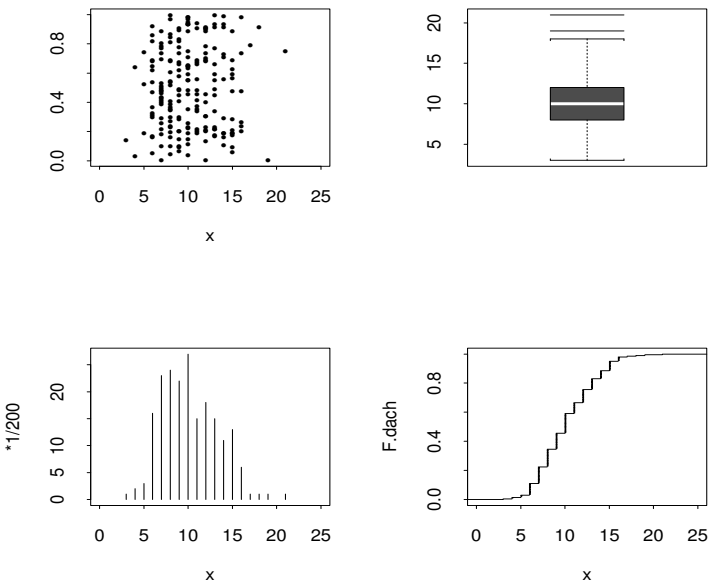
106 ⟨*1⟩+≡

```
par(mfrow=c(2,2))
jitterplot(x); boxplot(x)
stabdiagramm(x); F.dach(x)
par(mfrow=c(1,1))
```

Diese Anweisungen liefern:

```
 Min. 1st Qu. Median Mean 3rd Qu. Max. Std.Dev.   n
    3       8     10 10.2      12   21 3.188701 200
```

und das Bild:



Der Mittelwert ist mit 10.195 dem Erwartungswert von 10 sehr nah. Die Verteilung wirkt fast symmetrisch, doch sind im Boxplot zwei Ausreißer ausgewiesen, die beide auf einer Seite liegen. Die relativen Häufigkeiten führen im Gegensatz zur empirischen Verteilungsfunktion der simulierten Daten nicht zu einem so gleichmäßigen Stabdiagramm, wie man das aufgrund der Wahrscheinlichkeitsfunktion vermutet kann (siehe dazu unten: `poisson.calculator`).

6.6.2 Ein Erkennungsplot zur Poisson-Verteilung

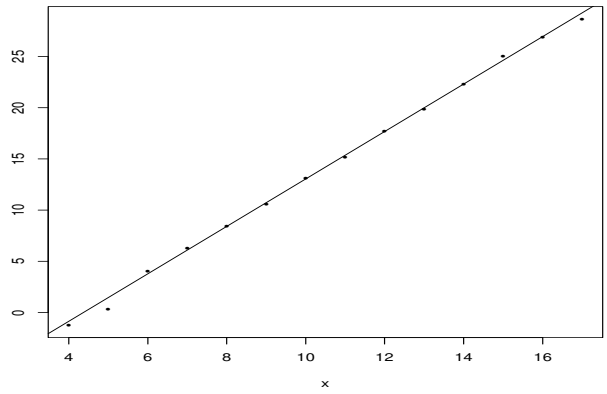
Mit den Umformungen

$$P(X=x)=\frac{\lambda^x e^{-\lambda}}{x!} \quad \Rightarrow \quad \ln P(X=x)=x\ln\lambda-\lambda-\ln(x!)$$
$$\Rightarrow \quad \ln P(X=x)+\ln(x!)=-\lambda+x\ln\lambda$$

läßt sich wieder von einer Stichprobe ausgehend ein Plot erstellen, in dem bei Eignung der Poisson-Verteilung die Punkte nahe der Geraden mit dem Achsenabschnitt $-\lambda$ und der Steigung $\ln\lambda$ liegen.
Hier sind die zugehörigen Anweisungen.

107 ⟨*erstelle für* x *Erkennungsplot zur Poisson-Verteilung* 107⟩≡
```
cat("Poisson-Erkennungsplot\n")
print(identify.pois(x))
```

Wir erhalten für die Stichprobe aus der Poisson-Verteilung:



sowie

```
Poisson-Erkennungsplot
 lambda.dach1 lambda.dach2
    10.124575    10.141854
```

Für die Stichprobe finden wir den uns ja bekannten Parameter ziemlich genau wieder.

6.6.3 Ein Kalkulator für die Poisson-Verteilung

Zur Beantwortung der Frage, die diesem Abschnitt voransteht, benötigen wir einen Kalkulator zur Poisson-Verteilung. Dieser wird hier zur Verfügung gestellt. Sein Funktionsumfang entspricht dem der bisher vorgestellten Kalkulatoren.

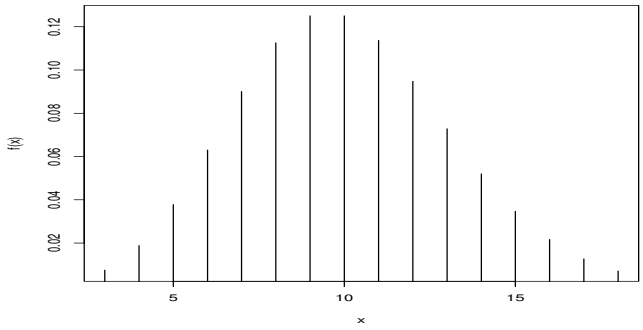
108 ⟨*aktiviere* `poisson.calculator` 108⟩≡

```
cat("Poisson-Kalkulator\n")
poisson.calculator()
```

Gehen wir einmal von der hohen Belastungsrate $\lambda = 10$ (10 zu erwartende Ereignisse pro Stunde) aus, so erhalten wir folgende Wahrscheinlichkeiten.

```
[1] "poisson.calculator start"
[1] "Bitte lambda eingeben! (Default: lambda=1) lambda=?"
1: 10
 E(X) Var(X)    sigma E(X)-2sigma E(X)-sigma E(X)+sigma E(X)+2sigma
   10     10 3.162278    3.675445   6.837722   13.16228    16.32456
[1] "Auswahl von poisson.calculator"
items:
1:lambda eingeben
2:f(x) und F(x) berechnen
3:Quantile berechnen
4:Statistiken berechnen
5:Plot erstellen
 Selection: 2
[1] "Bitte x eingeben! x=?"
1:

[1] "Werte der Wahrscheinlichkeits- und der Verteilungsfunktion:"
       x        f(x)        F(x)
 [1,]  3 0.007566655 0.010336051
 [2,]  4 0.018916637 0.029252688
 [3,]  5 0.037833275 0.067085963
 [4,]  6 0.063055458 0.130141421
 [5,]  7 0.090079226 0.220220647
 [6,]  8 0.112599032 0.332819679
 [7,]  9 0.125110036 0.457929714
 [8,] 10 0.125110036 0.583039750
 [9,] 11 0.113736396 0.696776146
[10,] 12 0.094780330 0.791556476
[11,] 13 0.072907946 0.864464423
[12,] 14 0.052077104 0.916541527
[13,] 15 0.034718070 0.951259597
[14,] 16 0.021698794 0.972958390
[15,] 17 0.012763996 0.985722386
[16,] 18 0.007091109 0.992813495
[1] "Auswahl von poisson.calculator"
items:
1:lambda eingeben
2:f(x) und F(x) berechnen
3:Quantile berechnen
4:Statistiken berechnen
5:Plot erstellen
Selection: 5
```

Der erstellte Plot sieht schon sehr *normal* aus ...



... *Normal* in dem Sinn, daß die Gestalt sehr stark der von der Normalverteilung entspricht. Mehr zur Normalverteilung erfahren Sie im nächsten Kapitel.

6.6.4 Serverdaten

Mit diesen Instrumenten kann der WWW-Server-Verwalter mit einem seiner Meinung nach geeigneten λ eigene Berechnungen anstellen und die Frage beantworten: *Wie viele Zugriffe werden sich in den nächsten 60 Minuten einstellen?*

6.7 QQ-Plots

Konzept: Es verwundert ein wenig, daß zum Vergleich von Datensätzen QQ-Plots eingeführt worden sind, in diesem Kapitel aber noch keinen Einsatz erfahren haben. Mit diesen läßt sich doch wohl auch prüfen, ob ein Datensatz zu einer bestimmten Verteilung paßt. In einem solchen QQ-Plot werden die empirischen Quantile ihren theoretischen Entsprechungen gegenübergestellt und als Punkte in einem Scatterplot dargestellt.

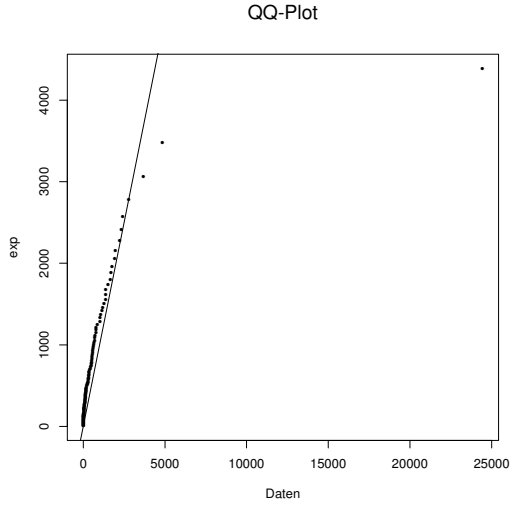
Praxis: Wir wollen dieses Versäumnis nachholen und hierzu zum Abschluß des Kapitels ein Modul anbieten. Das Modul wird nach der gewünschten Verteilung fragen. Geeignete Parameter werden aus den Daten geschätzt, da sie in der Regel unbekannt sind.

Technik: Ist das Modell vollkommen unpassend, kann es übrigens auch zu einem Fehler kommen, der hoffentlich keine weiteren Schwierigkeiten verursacht.

109 ⟨*erstelle zu Zwischenzeiten einen QQ-Plot zur Modellprüfung* 109⟩≡

```
cat("x <- dzeitpunkte.03.02.97\n")
x<-dzeitpunkte.03.02.97
cat("QQ-Plot\n")
qq.x.model(x)
```

Hier ist der Plot zu unserem Datensatz, der bei der Wahl `Exponentialverteilung` entsteht.



Das entstandene Bild sieht dem Erkennungs-Plot aus dem Abschnitt zur Exponentialverteilung äußerst ähnlich. Warum ist das wohl so?

6.8 Aufgaben

Die Aufgaben sollen im wesentlichen unterstützt mit den Angeboten folgender Auswahl bearbeitet werden.

110 ⟨*1⟩+≡

```
cat("Revbook:4.m:menu.mo:Menue Modellierungsschritte\n")
if(exists("auswahl.modelle")) auswahl.modelle()
```

Zur Information wieder das Auswahlmenü:

```
items:
  lege Datensaetze zur Bearbeitung auf x ab
  erzeuge Stichprobe x
  modifiziere Datensaetze x
  zeige x an, berechne Statistiken, stelle x dar
  aktiviere Verteilungskalkulator
  erstelle Verteilungs-Erkennungsplots
  erstelle QQ-Plots zur Modellueberpruefung
  erstelle Durchschnittskurven
  ermittle Belastung im Zeitablauf
```

a) Berechnen Sie mit dem Binomial-Kalkulator die Wahrscheinlichkeit dafür, daß bei 10 Münzwürfen genau 5 mal Kopf fällt, daß bei 3 Würfelwürfen keine 6 erscheint und daß von 100 gebraucht gekauften Disketten höchstens 10 defekt sind, wenn eine einzelne mit einer Wahrscheinlichkeit von $p = 0.90$ gebrauchsfähig ist.

b) Wie viele Versuche benötigen Sie im Durchschnitt beim Mensch-ärgere-Dich-nicht-Spiel, um eine 6 zu bekommen. Wie groß ist die Wahrscheinlichkeit, länger als 3 Runden pausieren zu müssen.

c) Telefongespräche treten mit einer Rate von $\lambda = 5$ pro Stunde ein. Wie lange müssen Sie bis zum nächsten Anruf in 90 % der Fälle höchstens warten?

d) Ein Skript habe im Schnitt 5 Druckfehler pro Seite. Wie groß ist die Wahrscheinlichkeit, eine fehlerfreie Seite zu bekommen? Wie viele fehlerfreie Seiten erwarten Sie für ein Skript vom Umfang 100 Seiten? Wie groß ist die Wahrscheinlichkeit, maximal 5 fehlerfreie Seiten zu bekommen?

e) Ziehen Sie wiederholt Stichproben aus einer geometrischen Verteilung vom Umfang 100 mit dem Parameter $p = 0.5$ und berechnen Sie jeweils die durchschnittlichen Fehlversuchsanzahlen. Berechnen Sie das Mittel dieser Durchschnitte. Liefert der zuletzt berechnete Durchschnitt einen Hinweis auf den Parameter p? Wieso? Welches p würden Sie vorschlagen?

f) Ziehen Sie eine Stichprobe vom Umfang 1000 aus einer binomialverteilten Grundgesamtheit mit $n = 100$ und $p = 0.7$. Welche der drei Verteilungsmodelle: Geometrische Verteilung, Exponentialverteilung, Poisson-Verteilung paßt Ihrer Meinung nach am besten zu der Stichprobe. Fällt Ihnen auch eine Begründung ein?

g) Ziehen Sie eine Stichprobe aus einer Exponentialverteilung mit $\lambda = 0.1$. Zeichnen Sie einen Graphen, in dem die Durchschnitte bis einschließlich des k-ten Elementes der Stichprobe dargestellt werden (Durchschnittskurve). Gegen welchen Wert strebt Ihr gezeichneter Pfad? Warum? Variieren Sie den Stichprobenumfang, Zufallsstart und λ. Machen Sie einen Vorschlag für den erforderlichen Stichprobenumfang, wenn aus dem Durchschnitt der Stichprobenwerte λ ermittelt werden soll.

h) Wiederholen Sie mit den Daten vom 17.02.97 möglichst viele der Schritte, die in diesem Kapitel mit den Daten vom 03.02.97 vorgeführt wurden. Sammeln Sie Punkte der Bestätigung wie auch abweichende Erkenntnisse.

i) Zählen Sie die Autos, die sich in 20 verschiedenen Rotphasen vor einer Ampel sammeln. Halten Sie außerdem die Zwischenzeiten zwischen den Ankünften fest. Welche Verteilung ist für die Anzahlen angemessen? Welche für die Wartezeiten auf das nächste Auto? Machen Sie Vorschläge für die Parameter.

7 Experimente zur Normalverteilung

111 ⟨*1⟩+≡
```
cat("Revbook:5.a:start.nv:Kapitel Normalverteilung\n")
```

Zum Literaturstudium sei hingewiesen auf: [Autorenkollektiv: Kapitel 6], [Bamberg, Baur: Abschnitte über Binomialverteilung, Normalverteilung, den zentralen Grenzwertsatz], [Schlittgen: Kapitel über die Normalverteilung].

Konzept: In diesem Kapitel wird nicht auf die Serverdaten zurückgegriffen, da im wesentlichen abstraktere Zusammenhänge den Gegenstand bilden.

Konzept: Was wäre die Statistik ohne die Normalverteilung? Anders und wahrscheinlich ärmer! Wir wollen in diesem Abschnitt einige Argumente für ihre herausragende Rolle präsentieren und mit Hilfe von Experimenten demonstrieren.

7.1 Die Normalverteilung zur Approximation der Binomialverteilung

Praxis: Stellen Sie sich vor, Sie würden das nächste Jahr jede Woche zwei Lottotips abgeben. Was meinen Sie, wie groß die Anzahl Ihrer Gewinne sein wird? Wie groß wird sie für 5 Jahre sein?

Konzept: Da die Gewinnchance p für einen Tip ungefähr 0.02 beträgt, können Sie im Prinzip alle Fragen mit der *Binomial*($n = 104, p = 0.02$)-Verteilung bzw. der *Binomial*($n = 520, p = 0.02$)-Verteilung beantworten. (Sie können die folgenden Überlegungen gern mit einem genaueren Wert für p durchführen, den Sie dazu aber erst bestimmen müssen. Viel Spaß!) Ohne Rechner dürfte das jedoch schon einige Schwierigkeiten bereiten. Man benötigt dann ein Approximationswerkzeug. Mit der Normalverteilung steht ein solches zur Verfügung:

> **Theorem** (De Moivre-Laplace): Es sei X_n binomialverteilt mit den Parametern n und p. Dann geht für $n \to \infty$ die Verteilung von
>
> $$Y_n = \frac{X_n - np}{\sqrt{np(1-p)}}$$
>
> gegen die Standardnormalverteilung:
>
> $$P(Y_n \leq x) \to \Phi(x) \qquad \text{für alle } x \in \mathcal{R}$$
>
> Der Quotient $\frac{X_n - np}{\sqrt{np(1-p)}}$ ist also approximativ N(0,1)-verteilt.

Damit ist die Anzahl der Gewinntips approximativ normalverteilt mit den Parametern $\mu = n \cdot p = 1.04$ und $\sigma^2 = n \cdot p(1-p) = 1.02$.

Praxis: Als nächstes sei daran erinnert, daß im Zusammenhang mit den Pfadplots des letzten Kapitels auf die Binomialverteilung verwiesen wurde. In einem solchen Plot wurden die durchschnittlichen Erfolgsanzahlen gegen die Anzahl der durchgeführten Bernoulli-Experimente als Linienzug abgetragen. Wiederholungen von Experimenten führten zu einer Vielzahl von Pfaden, die dann die Variabilität der Erfolgsquote visualisierten.

Konzept: Es wurde argumentiert, daß zur Abschätzung der Variabilität der Erfolgsquote für ein bestimmtes n die Binomialverteilung hilfreich wäre. Dies gilt auch für die Beantwortung der Frage, wie groß n gewählt werden muß, um ein unbekanntes p ausreichend genau abschätzen zu können. Da bei solchen Fragestellungen das fragliche n relativ groß werden kann, ist die Approximation durch die Normalverteilung interessant.

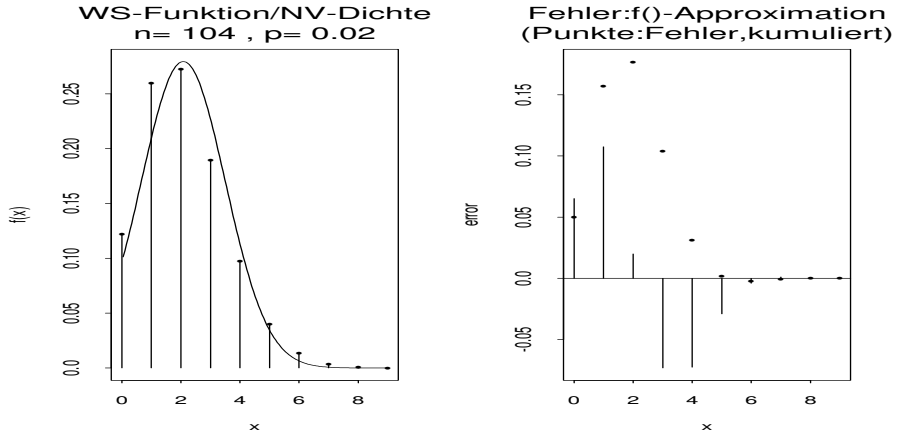
Praxis: Für den Anwender ist es sehr wichtig zu wissen, wie gut eigentlich die Approximation ist. Deshalb wollen wir hier ein Demonstrationsexperiment anbieten.

Technik: In dem Experiment lassen sich die Parameter n und p der Binomialverteilung setzen. Dann werden die Wahrscheinlichkeitsfunktion der Binomialverteilung und der Dichtefunktion der Normalverteilung N(np, $\sqrt{np(1-p)}$) in einer Graphik gezeigt. In einem zweiten Plot werden die Differenzen zwischen der Binomialverteilung und der Normalverteilung als senkrechte Stäbe visualisiert. Zwar macht das Theorem eine Aussage über Verteilungsfunktionen, doch sind verschiedene Betrachtungen auf der Ebene der Wahrscheinlichkeitsfunktion bzw. Dichte anschaulicher. Deshalb sind im ersten Plot nicht die Verteilungsfunktionen abgebildet worden. Für einen korrekten Vergleich muß man einen *Stab* der Wahrscheinlichkeitsfunktion (an der Stelle x_0) mit der Fläche eines *Streifens* der Dichtefunktion (der von $x_0 - 1$ bis x_0 reicht) vergleichen. Für große Werte von np unterscheiden sich übrigens die Flächen der Streifen nicht sonderlich von den Werten der Dichtefunktion, so daß man durchaus den Fit von Wahrscheinlichkeitsfunktion und Dichtefunktion betrachten kann. In dem zweiten Plot sind die Differenzen zwischen den *Stäben* und den zugehörigen *Streifen* als vertikale Linien dargestellt. Zusätzlich ist im zweiten Plot durch Punkte eingetragen worden, welcher Fehler sich an den entsprechenden Stellen zwischen den Verteilungsfunktionen einstellt. Dieser Fehler ergibt sich aus Kumulation der einzelnen Fehler.

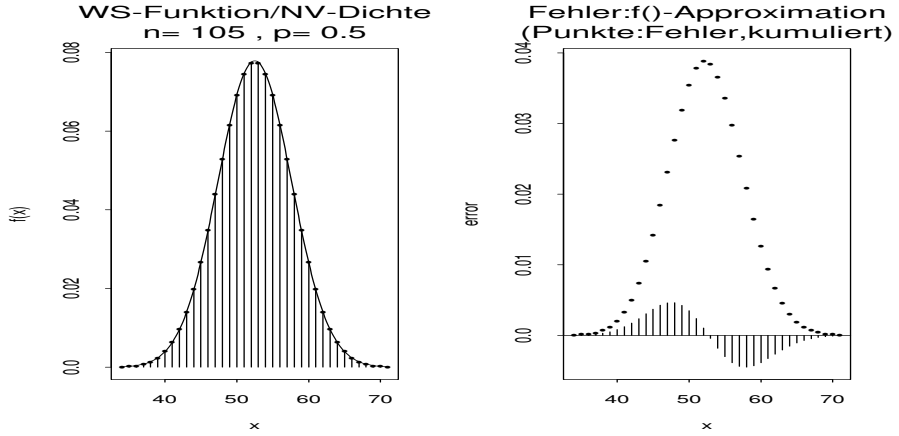
112 ⟨*aktiviere Experiment zur Approximation von* Binom(n,p) *durch NV* 112⟩≡

```
cat("Demonstration der Approximation der Binomialverteilung durch\n")
cat("die Normalverteilung\n")
demo.Laplace()
```

Praxis: Zu dem Beispiel mit $n = 104$ und $p = 0.02$ erhalten wir den folgenden graphischen Output.



Die Approximation könnte besser sein. Diese Erkenntnis steht nicht im Widerspruch zu allgemeinen Empfehlungen. Zum Beispiel werden im Statistik-Skript die Approximationsvoraussetzungen $np \geq 10$ und $n(1-p) \geq 10$ genannt. Offensichtlich ist die erste nicht erfüllt. Für 5 Jahre ist die Bedingung erfüllt und könnte verwendet werden. Dieses sollte der Leser am Rechner tun. Wie man schnell prüfen kann, führen p-Werte nahe 0.5 zu viel besseren Ergebnissen. Für $n = 104$ und $p = 0.5$ würde man zum Beispiel erhalten:



Konzept: Binomialverteilungen mit $p = 0.5$ besitzen Wahrscheinlichkeitsfunktionen, die um die Stelle $x = np$ symmetrisch sind. Für gegebenes n ist die Verteilung um so asymmetrischer je weiter p von 0.5 entfernt liegt. Außerdem gilt, daß für einen festen Wert von p mit wachsendem n weniger asymmetrische Verteilungen resultieren. Hierdurch wird verständlich, daß für sehr kleine p-Werte für eine gute Approximation ein großes n erforderlich ist. Zum Schluß sei erwähnt, daß die Approximation durch eine sogenannte Stetigkeitskorrektur noch ein wenig verbessert werden kann. Was es damit auf sich hat, sei dem Leser als Literaturstudiums-Aufgabe gestellt.

7.2 Zur standardisierten Summe von Zufallsvariablen

Konzept: Wenn man die Normalverteilung nur zur Approximation der Binomialverteilung verwenden könnte, wäre sie sicher nicht als Motiv für den 10-DM-Schein ausgesucht worden. Sie besitzt natürlich noch wichtigere Eigenschaften. Dazu stellen wir folgende Überlegung an: Eine binomialverteilte Zufallsvariable kann als Summe von Bernoulli-verteilten Zufallsvariablen aufgefaßt werden. Damit läßt sich mit der Normalverteilung die Verteilung einer Summe von speziellen Zufallsvariablen annähern. Hieraus ergibt sich die Frage, ob auch für andere Zufallsvariablen ein solcher Zusammenhang gefunden werden kann. Die Antwort liefert der zentrale Grenzwertsatz:

Theorem (Zentraler Grenzwertsatz): Es seien $X_1, X_2, \ldots, X_n$ identisch verteilte, unabhängige Zufallsvariablen mit $\mathrm{E}(X_i) = a$ und $\mathrm{Var}(X_i) = b^2$. Dann gilt:

$$\lim_{n\to\infty} P(Z \leq z) = \Phi(z)$$

mit

$$Z = \frac{\sum X_i - na}{b\sqrt{n}} = \frac{1}{\sqrt{n}} \sum \frac{X_i - a}{b} = \frac{\bar{X} - a}{b/\sqrt{n}};$$

es konvergiert also die Verteilungsfunktion der standardisierten Summe bzw. des standardisierten Mittels dieser Zufallsvariablen mit steigender Summandenzahl gegen die Verteilungsfunktion der standardisierten Normalverteilung.

Die Prämissen dieses Satzes sind äußerst allgemein. Außerdem gibt es in der Statistik viele Situationen (wie zum Beispiel in der Testtheorie), in denen eine Frage auf die Summe von unabhängigen und identisch verteilten Zufallsvariablen zurückgeführt werden kann. Die Normalverteilung erlangt aus diesem Grund über den zentralen Grenzwertsatz aber auch aufgrund ihrer mathematischen Eigenschaften ihre herausragende Bedeutung.

Praxis: In praktischen Situationen kann der zentrale Grenzwertsatz wieder als Approximationswerkzeug dienen. Stellen Sie sich vor, daß ein ordentlicher Buchhalter nach jedem Tanken die getankte Benzinmenge ordentlich in sein Fahrtenbuch schreibt, jedoch leider immer auf ganze Liter rundet. Nach 48 Tankvorgängen möchte er gern wissen, wie stark sich wohl die getankte Menge von der Summe seiner Volumeneinträgen unterscheidet, wie groß also der Fehler ist. Einzelne Fehler können mit einer Gleichverteilung modelliert werden, deren Erwartungswert als 0 und deren Varianz als $1/12$ angenommen werden kann. Nach diesen Angaben läßt sich für die Summe der 48 Einzelfehler eine Normalverteilung mit Erwartung 0 und Standardabweichung 2 verwenden.
Die Qualität der Approximation hängt vom Stichprobenumfang wie auch von der Verteilung der Stichprobenvariablen ab. Um hierfür ein Gespür zu bekommen, folgt nun ein Experiment, in dem diese beiden Einflußgrößen gewählt werden können.

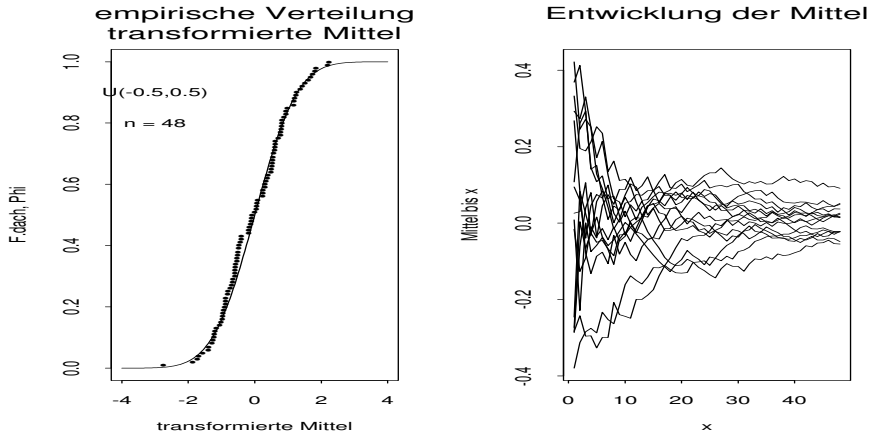
Technik: Der Grundgedanke des Experimentes besteht darin, zu der gewählten Verteilung wiederholt Stichproben zu generieren und dann die empirische Verteilungsfunktion der standardisierten Stichprobenmittel und die Verteilungsfunktion der Standardnormalverteilung Φ in einem Plot darzustellen. In einem zweiten Plot werden zu den einzelnen Stichproben (aber höchstens 15) die Entwicklungen der Mittel dargestellt. Dazu wird zu jeder Stichprobe $X_1, \ldots, X_n$ jeweils $\sum_{j=1}^{i} X_j / i$ gegen i abgetragen. Auf diese Weise erhält man zu jeder Stichprobe einen Pfad. Aus der Menge der Pfade lassen sich Genauigkeitsvorstellungen entwickeln.
Hier ist das Modul:

113 ⟨*aktiviere Experiment zum zentralen Grenzwertsatz* 113⟩≡

```
cat("Demonstration der Approximation von standardisierten\n")
cat("Stichprobenmitteln durch die Normalverteilung\n")
demo.zgws()
```

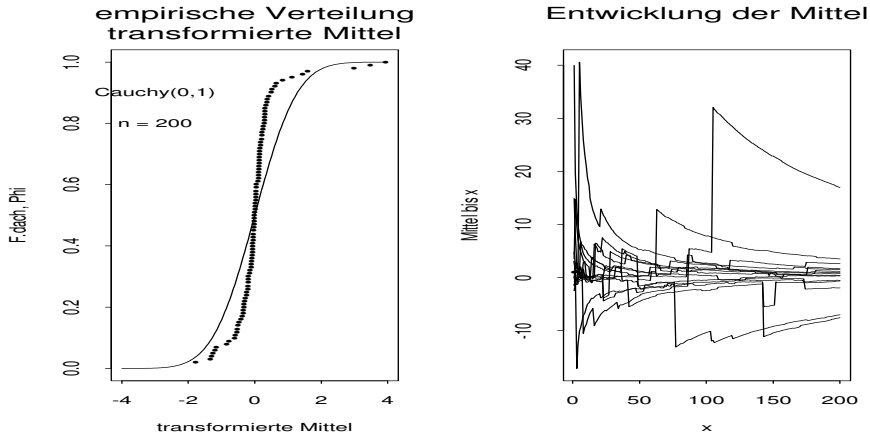
Praxis: Zu der Situation mit den Tankfüllungen ergibt sich als Dialog und Resultat:

```
Demonstration der Approximation von standardisierten
Stichprobenmitteln durch die Normalverteilung
Modell / Aus welcher Verteilung sollen
Stichproben gezogen werden (Default: Normalverteilung)?
items:
1:Normalverteilung
2:Exponentialverteilung
3:Gleichverteilung
4:Cauchy-Verteilung
5:Binomialverteilung
6:Poisson-Verteilung
Selection: 3
Bitte Untergrenze der Gleichverteilung eingeben (Default: 0)!
1: -0.5
Bitte Obergrenze eingeben (Default: 1)!
1: 0.5
Bitte Umfang der einzelnen Stichproben eingeben! (Default: 5)
1: 48
Bitte Umfang der Wiederholungen eingeben! (Default: 100)
1: 100
Bitte Start des Zufallszahlengenerators eingeben! (Default: 13)
1: 13
...
```



Die empirische Verteilungsfunktion (durch Punkte markiert) liegt sehr nahe bei der von der Standardnormalverteilung (durchgezogene Linie). Daß dieses bei Verletzung der Annahmen des Satzes nicht so ist, zeigt der Plot, der sich bei der Cauchy-Verteilung mit zum Beispiel folgenden Setzungen ergibt:

```
Bitte Zentrum der Cauchy-Verteilung eingeben (Default: 0)!
1: 0
Bitte Skalenparameter eingeben (Default: 1)!
1: 1
Bitte Umfang der einzelnen Stichproben eingeben! (Default: 5)
1: 200
Bitte Umfang der Wiederholungen eingeben! (Default: 100)
1: 100
Bitte Start des Zufallszahlengenerators eingeben! (Default: 13)
1: 13
...
```



Man sieht, daß die Pfade immer wieder wie durch einen Impuls Sprünge machen. Am Plot der Verteilungsfunktionen erkennt man, daß sich besonders die Schwänze unterscheiden. Dazu sei angemerkt, daß bei vielen realen Problemsituationen den Schwänzen der Verteilungen eine besondere Bedeutung zukommt. Zum Beispiel werden sich die zentralen Überlegungen beim Testen von Hypothesen ausschließlich auf die Randbereiche von Verteilungen beziehen, so daß die Qualitäten einer Approximation wesentlich an den Schwanzenden festgemacht werden sollten.
Wir wollen es bei dieser Demonstration belassen und keine weiteren theoretischen Ausführungen anstellen. Es empfiehlt sich, Versuche mit der Exponenentialverteilung und kleinen Stichprobenumfängen und auch den diskreten Verteilungen mit kleineren und größeren Umfängen anzustellen.

7.3 Ein Normalverteilungskalkulator

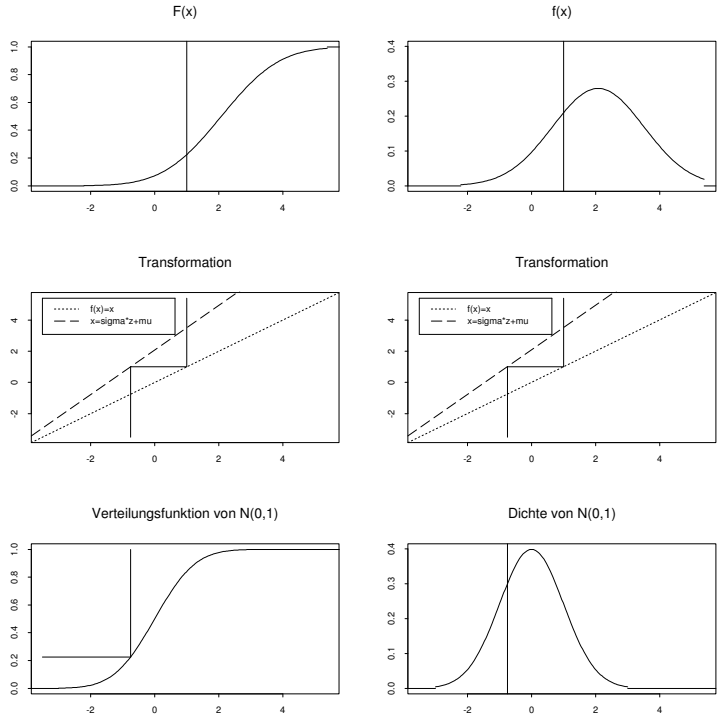
Technik: In diesem Abschnitt wollen wir ein Instrument bereitstellen, mit dem einige Berechnungen rund um die Normalverteilung durchgeführt werden können. Dieses Instrument wird mit dem folgenden Modul bereitgestellt. Zunächst sind Mittel und Standardabweichung/Varianz festzulegen, worauf ein kurzer Steckbrief des gewählten Modells ausgegeben wird. Danach kann man die Parameter neu wählen, Dichte- und Verteilungsfunktionswerte zu vorgegebenen x-Werten ermitteln, Quantile berechnen lassen oder das gewählte Modell verschieben oder skalieren. Im graphischen Fenster wird dabei immer der Zusammenhang zwischen dem gewählten Modell und der Standardnormalverteilung dargestellt. Die oberen beiden Graphiken beziehen sich auf das gewählte Normalverteilungsmodell, die unteren beiden auf die Standardnormalverteilung. In der Mitte wird der Transformationsprozeß visualisiert.

114 ⟨*aktiviere Normalverteilungskalkulator* 114⟩≡
```
cat("Aktivierung des Normalverteilungskalkulators\n")
normal.calculator()
```

Praxis: Als Beispiel wird im folgenden der Dialog gezeigt, der zu der Frage der Anzahl der zu erwartenden Gewinntips entstehen könnte. Dabei wurde auf eine Normalverteilung mit den Parametern $\mu = np = 2.08$ und $\sigma^2 = np(1-p) = 2.04$ verwiesen. (Zwar ist unter diesen Bedingungen eine Normalapproximation schlecht, doch stört das hier nicht die Verwendung als Beispiel.)

```
[1] "normal.calculator start"
[1] "Bitte Mittel eingeben! (Default: mu=0) mu=?"
1:  2.08
[1] "sigma ist festzulegen!"
[1] "Wenn Sie"
[1] "  Standardabweichung eingeben wollen, bitte 0 eingeben,"
[1] "  Varianz            eingeben wollen, bitte 1 eingeben!"
[1] "  (Default -- leere Eingabe: 0)"
1:  1
[1] "Bitte VARIANZ festlegen! (Default: 1) sigma^2=?"
1:  2.04
 E(X) Var(X)    sigma E(X)-2sigma E(X)-sigma E(X)+sigma E(X)+2sigma
 2.08   2.04 1.428286  -0.7765714  0.6517143   3.508286    4.936571


[1] "Auswahl von normal.calculator"
items:
1:Mittel festsetzen
2:Variabilitaet festlegen
3:f(x), F(x), (1-F(x)) berechnen
4:Quantile berechnen
5:X verschieben
6:X skalieren
Selection: 3


[1] "Bitte x eingeben! x=?"
1:  1
 E(X) Var(X)    sigma E(X)-2sigma E(X)-sigma E(X)+sigma E(X)+2sigma
 2.08   2.04 1.428286  -0.7765714  0.6517143   3.508286    4.936571
[1] "Dichte- und Verteilungsfunktion:"
     x     f(x)      F(x)    1-F(x)
[1,] 1 0.209864 0.2247793 0.7752207
[1] "Auswahl von normal.calculator"
```

In dem Dialog wurde zuerst das Modell fixiert. Dann wurde ermittelt, wie groß die Wahrscheinlichkeit ist, ein Gewinnlos zu bekommen. Der Kalkulator berechnete einen Wert von zirka 78 %. Das zugehörige Bild sieht wie folgt aus:

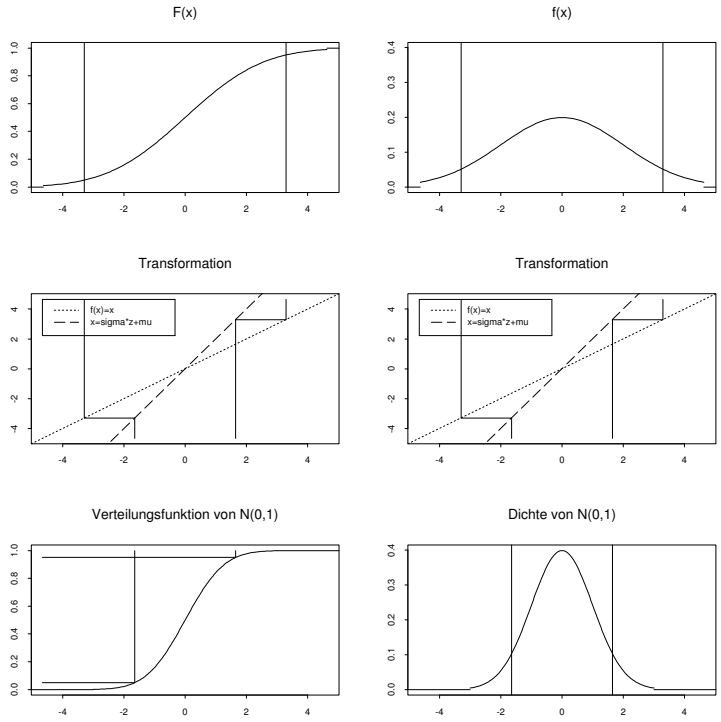


Es wurde oben schon deutlich, daß die Approximation nicht allzu gut ist. Natürlich sollte man bei dieser Konstellation nicht zur Approximation greifen. Genauere Ergebnisse bekommt man hier mit dem Binomial-Kalkulator. Der Fall *Gewinnlose in 5 Jahren* bleibt wieder dem Leser überlassen. Für das Beispiel mit den Tankmengenfehlern ist eine Normalverteilung mit den Parameter $\mu = 0$ und $\sigma = 2$ zu verwenden. Mit dem folgenden Dialog wurde festgestellt, daß sich in 90 % aller Fälle der Fehler grob zwischen -3.3 und 3.3 bewegen wird.

```
[1] "normal.calculator start"
[1] "Bitte Mittel eingeben! (Default: mu=0) mu=?"
1:  0
[1] "sigma ist festzulegen!"
[1] "Wenn Sie"
[1] "  Standardabweichung eingeben wollen, bitte 0 eingeben,"
[1] "  Varianz            eingeben wollen, bitte 1 eingeben!"
[1] "  (Default -- leere Eingabe: 0)"
1: 0
[1] "Bitte STANDARDABWEICHUNG festlegen! (Default: 1) sigma=?"
1:  2
 E(X) Var(X) sigma E(X)-2sigma E(X)-sigma E(X)+sigma E(X)+2sigma
    0      4     2          -4         -2          2           4


[1] "Auswahl von normal.calculator"
items:
1:Mittel festsetzen
2:Variabilitaet festlegen
3:f(x), F(x), (1-F(x)) berechnen
4:Quantile berechnen
5:X verschieben
6:X skalieren
Selection: 4
```

```
[1] "Bitte Wahrscheinlichkeiten eingeben! F(x)=?"
[1] 0.05 0.95
 E(X) Var(X) sigma E(X)-2sigma E(X)-sigma E(X)+sigma E(X)+2sigma
    0      4     2          -4         -2          2          4
[1] "Quantile:"
     F(x)         x
[1,] 0.05 -3.289707
[2,] 0.95  3.289707
[1] "Auswahl von normal.calculator"
...
```

Dazu gehört das folgende Bild:



7.4 Aufgaben

Der aufmerksame Leser hat sicher schon verschiedene Aufgaben im Text entdeckt. Außerdem wimmelt es in den Lehrbüchern zur Statistik von Aufgaben zur Normalverteilung, so daß dem hier nichts mehr hinzuzufügen ist. Da in diesem Kapitel nur drei Module eingesetzt wurden, erscheint auch eine eigene Auswahlfunktion ein wenig übertrieben zu sein. So endet das Kapitel hier nun etwas abrupt.

8 Anhang

8.1 Endehinweis

115 ⟨*1⟩+≡

```
cat("-----------------------------------\n")
cat(" Sie sind am Ende angekommen!")
cat(" Waehlen Sie einen neuen Startpunkt! \n")
```

8.2 Liste der verwendeten Module

⟨* 1⟩ 1, 15, 16, 24, 40, 41, 46, 55, 63, 71, 72, 73, 75, 76, 77, 79, 82, 87, 90, 91, 94, 95, 96, 98, 100, 102, 104, 106, 110, 111, 115
⟨*aktiviere Experiment zum zentralen Grenzwertsatz* 113⟩ 113
⟨*aktiviere Experiment zur Approximation von* Binom(n,p) *durch NV* 112⟩ 112
⟨*aktiviere Normalverteilungskalkulator* 114⟩ 114
⟨*aktiviere* `binomial.calculator` 86⟩ 86
⟨*aktiviere* `geometric.calculator` 92⟩ 92
⟨*aktiviere interaktives Dichtespur-Sektion zu* `x` 35⟩ 35
⟨*aktiviere* `poisson.calculator` 108⟩ 108
⟨*befrage F.dach zu* `x` *graphisch* 39⟩ 39
⟨*berechne Extrema von* `x` 5⟩ 5
⟨*berechne Inter-Quartilsabstand von* `x` 12⟩ 12
⟨*berechne Korrelationskoeffizienten der ersten beiden Elementen von* `xy` 57⟩ 57
⟨*berechne Median von* `x` 4⟩ 4
⟨*berechne Midrange von* `x` 6⟩ 6
⟨*berechne Mittelwert von* `x` 3⟩ 3
⟨*berechne Spannweite von* `x` 9⟩ 9
⟨*berechne Standardabweichung von* `x` 11⟩ 11
⟨*berechne Stichprobenvarianz von* `x` 10⟩ 10
⟨*berechne Trimean von* `x` 8⟩ 8
⟨*berechne getrimmtes Mittel von* `x` 7⟩ 7
⟨*berechne und plotte Minuten mit Zugriffen der Stunden vom 03.02.97* 85⟩ 85
⟨*berechne zusammenfassende Statistiken für Minutenaktivitäten (03.02.97)* 80⟩ 80
⟨*berechne zusammenfassende Statistiken von* `x` 13⟩ 13
⟨*berechne zusammenfassende Statistiken zu* `xy` 43⟩ 43
⟨*bereite Gedächnisexperiment vor und führe es durch* 93⟩ 93
⟨*bilde Ausschnitt der Daten* `x` 23⟩ 23
⟨*bilde Ausschnitte der Komponenten von* `xy` 45⟩ 45
⟨*ermittele Belastung anhand zusammenhängender Zugriffszeitpunkte* 103⟩ 103
⟨*ermittle aus Sekundendaten vom 03.02.97 Minuten mit Aktivität* `->x` *und zeige erste Werte von* `x` *an* 74⟩ 74
⟨*errechne Anzahlen der Stichproben von* `xy` 65⟩ 65
⟨*erstelle Boxplot zu* `log(x)` 22⟩ 22
⟨*erstelle Boxplot zu* `x` 20⟩ 20
⟨*erstelle Boxplot zu* `xy` 44⟩ 44
⟨*erstelle Dichtespur zu* `x` 34⟩ 34
⟨*erstelle Draftsman's Display zu* `xy` 61⟩ 61
⟨*erstelle F.dach für klassierte Daten zu* `x` 37⟩ 37
⟨*erstelle F.dach von* `x` 36⟩ 36
⟨*erstelle Histogramm von* `x` *mit manueller Wahl der Grenzen* 33⟩ 33
⟨*erstelle Histogramm zu* `x` 31⟩ 31
⟨*erstelle Histogramm zu* `x` *mit Grenzen 0, 200, 400, ..., 5000* 32⟩ 32
⟨*erstelle Häufigkeitstabelle zu* `x` 28⟩ 28
⟨*erstelle Jitterplot zu* `x` 26⟩ 26
⟨*erstelle Scatterplot zu den ersten beiden Elementen von* `xy` 53⟩ 53
⟨*erstelle Scatterplot zu* `xy` *und zeichne konvexe Hüllen* 56⟩ 56
⟨*erstelle Stabdiagramm zu* `x` 27⟩ 27
⟨*erstelle Stem-and-Leaf-Plot zu* `x` 30⟩ 30
⟨*erstelle emp. Verteilungsfunktion mit manueller Wahl der Grenzen zu* `x` 38⟩ 38
⟨*erstelle für Zwischenzeiten Erkennungsplot zur Exponentialverteilung* 101⟩ 101
⟨*erstelle für* `x` *Erkennungsplot zur Poisson-Verteilung* 107⟩ 107
⟨*erstelle graphische 5-Zahlen-Zusammenfassung von 1, 2, 3, 4, 5* 18⟩ 18
⟨*erstelle graphische 5-Zahlen-Zusammenfassung von* `x` 19⟩ 19
⟨*erstelle spezielles Balkendiagramm zu* `x` 29⟩ 29
⟨*erstelle zu Zwischenzeiten einen QQ-Plot zur Modellprüfung* 109⟩ 109
⟨*erstelle zu den ersten beiden Elementen von* `xy` *einen QQ-Plot* 49⟩ 49
⟨*erstelle zum paarweisen Vergleich der Datensätze von* `xy` *QQ-Plots* 64⟩ 64
⟨*lege Zeitpunkte vom 03.02.97 auf* `xy` *ab* 66⟩ 66
⟨*lege Zwischen-Zeitpunkte vom 03.02.97 auf* `x` *ab* 17⟩ 17
⟨*lege Zwischenzeiten und Mengen vom 03.02.97 auf* `xy` *ab* 52⟩ 52
⟨*lege Zwischenzeiten und Zeiten vom 03.02.97 auf* `xy` *ab, ordne* `xy[1]` *gemäß* `xy[2]` *in* `anz` *Klassen ein:* `xy` 62⟩ 62
⟨*lege Zwischenzeiten vom 03.02.97 auf* `x` *ab und drucke* `x` *aus* 2⟩ 2
⟨*lege Zwischenzeiten vom 03.02.97 auf* `xy` *ab und ordne Daten in* `anz` *gleich große Klassen ein* 60⟩ 60
⟨*lege Zwischenzeiten vom 03.02.97 und vom 17.02.97 auf* `xy` *ab* 42⟩ 42
⟨*plotte Anteil der Minuten mit Zugriff gegen betrachtete Minutenanzahlen* 81⟩ 81
⟨*plotte Durchschnitte für die ersten k Elemente von* `x` *gegen k* 78⟩ 78
⟨*plotte Stichprobe, berechne Statistiken zu* `x` *— diskret* 84⟩ 84
⟨*plotte Stichprobe, berechne Statistiken zu* `x` *— kontinuierlich* 99⟩ 99
⟨*plotte* `x` *als horizontale Stäbe* 89⟩ 89
⟨*rechne Sekunden in Tage, Stunden und Minuten um* 70⟩ 70
⟨*start* 116⟩ 116, 117, 118

⟨*starte Erfahrungssammelfunktion zum Boxplot* 25⟩ 25
⟨*transformiere Elemente von* xy *mittels Box-Cox-Transformation* 54⟩ 54
⟨*transformiere* x *mittels Box-Cox-Transformation, zeige Statistiken* 14⟩ 14
⟨*zeichne Dichtespuren zu* xy 69⟩ 69
⟨*zeichne* F.dach *zu* xy 68⟩ 68
⟨*zeichne Histogramme zu* xy 47⟩ 47
⟨*zeichne Jitterplot zu* xy 67⟩ 67
⟨*zeichne zu Regendatensätzen konvexe Hüllen und berechne Korrelationskoeffizienten* 59⟩ 59
⟨*zeichne zu Regenmengen und Kornerträge konvexe Hüllen und berechne Korrelationskoeffizienten* 58⟩ 58
⟨*zeichne zu* xy *F.dach und f.dach* 48⟩ 48
⟨*zeige Bootstrap-Stichprobenverteilungen mit* xy 50⟩ 50
⟨*zeige Verteilung einer Statistik von Bootstrap-Stichproben aus* xy 51⟩ 51
⟨*zeige die 10 kleinsten und die 10 größten Werte von* x *an* 21⟩ 21
⟨*ziehe Stichprobe aus einer Exponentialverteilung mit λ:* x 97⟩ 97
⟨*ziehe Stichprobe aus einer Poisson-Verteilung mit λ:* x 105⟩ 105
⟨*ziehe* n *geometrisch verteilte Zufallszahlen mit* p*:* x 88⟩ 88
⟨*ziehe* r *binomialverteilte Zufallszahlen mit* n *und* p*:* x 83⟩ 83

8.3 Initialisierung

116 ⟨*start* 116⟩≡

```
cat("---------------------------------------\n")
print("Start der revbook-Initialisierung")
options(digits=5)
if(file.exists("robj.R")) source("robj.R",local=T) else {
  if(grep("relax",.path.package())){
    if(file.exists(paste(.path.package("relax"),"rev/robj.R",sep="/"))){
      source(paste(.path.package("relax"),"rev/robj.R",sep="/"),local=T)
    } else {
      source("http://www.wiwi.uni-bielefeld.de/~wolf/software/revbook/robj.R",local=T)
    }
  }
}
```

117 ⟨*start* 116⟩+≡

```
# myhead.menu(rm.menu=T)
myhead.menu(item="Menue: Masszahlen",
   code=auswahl.masszahlen, title="Revbook",menu.no=2)
myhead.menu(item="Menue: eindimensionale Graphiken", code=auswahl.graphiken,menu.no=2)
myhead.menu(item="Menue: zweidimensionale Graphiken", code=auswahl.mgraphiken,menu.no=2)
myhead.menu(item="Menue: Modelle",         code=auswahl.modelle,menu.no=2)
```

8.4 Abschluß der Initialisierungsbefehle

118 ⟨*start* 116⟩+≡

```
cat("--- REVBOOK Initialisierung beendet ---\n")
if(file.exists(paste(.path.package("relax"),"rev/rrevbook.pdf",sep="/")))
cat("Zur Ansicht betrachte:\n",paste(.path.package("relax"),
        "rev/rrevbook.pdf",sep="/"),
        "\n---------------------------------------\n")
```