

## Aufgabenblatt

Abgabe: bis Montag, 16.11.2009; 10:00 Uhr

Diese Zettel sollen Sie mit dem Statistikprogramm R vertraut machen. Zunächst werden die grundlegende Bedienung und wichtige Befehle erklärt. Im Anschluss daran finden sich Aufgaben.

**Durch das regelmäßige Bearbeiten dieser Aufgaben kann ein Teilnahmechein erworben werden.**

Alle Aufgaben sind Programmieraufgaben. Bitte kopieren Sie die eingegebenen Befehle in eine Textdatei (Kommentare können hinter der Raute # eingegeben werden), und senden Sie diese zur Abgabe per e-Mail an

`mentemeier@uni-muenster.de`

Eine schriftliche Abgabe ist nicht erforderlich.

Das Statistikprogramm R ist frei verfügbar, und kann von <http://www.r-project.org/> heruntergeladen werden. Zum Starten in der Uni siehe gesonderte Anleitung.

## Einführung in R

### Grundlegende Befehle

Befehle werden direkt nach dem > eingegeben, und mit „Enter“ abgeschlossen. So gibt 2+6 beispielsweise das Ergebnis [1] 6. Die führende [1] ist die Nummerierung der Ausgabe. Dezimalstellen werden durch den Punkt abgetrennt.

R beherrscht alle grundlegenden Rechenarten, diese werden mittels +, -, \*, /, ^ angesprochen. Auch zahlreiche (reelle) Funktionen sind bereits integriert, wie z.B. `sin`, `exp`, `log`, `sqrt`. Diese werden - wie alle Funktionen in R - in der Form `Funktion(Argumente)` aufgerufen, z.B. `exp(1)`. Oder auch die Hilfsfunktion `help(Funktionsname)`.

### Vektoren

Vektoren werden mit der Funktion `c()` eingegeben, Einträge werden durch Kommata getrennt. *Variablen* werden mittels des Pfeils `<-` zugewiesen, und werden durch Eingabe ihres Namens aufgerufen, z.B.

```
vektor<-c(1,2,3,4)
```

```
vektor
```

Operationen werden grundsätzlich komponentenweise auf Vektoren angewendet, Skalare werden entsprechend wiederholt. Dies gilt auch für logische Operatoren, wie `==` (gleich), `<=` (kleiner gleich), `>` (größer), `!=` (ungleich) usw. Das Ergebnis ist dann ein logischer Vektor mit den Einträgen "TRUE" und "FALSE".

Die Einträge werden mit eckigen Klammern indiziert; Indexmenge kann wiederum ein Vektor sein (dann werden mehrere Einträge ausgegeben), oder auch ein logischer Vektor, z.B. `vektor[vektor!=2]`. Vektoren mit ganzzahligen Einträgen erzeugt auch der Kurzbefehl `-2:10`, dies würde einen Vektor mit den ganzen Zahlen von -2 bis 10 erzeugen. Weitere nützliche Funktionen für Vektoren sind `sum()`, `max()`, `min()`, `sort()`, `length()`, `cumsum()`. Qualitative Merkmale, wie z.B. Namen, werden in Anführungsstrichen eingegeben. Ein solcher Vektor sollte mit dem Befehl `factor()` in einen Faktor umgewandelt werden, damit R erkennt, dass es sich um qualitative Merkmale handelt. Mit dem Befehl `levels()` werden die Ausprägungen eines Faktors angezeigt.

## Datentabellen eingeben und einlesen

Datentabellen werden in R meist als `data.frame` abgelegt. Ein `data.frame` besteht aus mehreren Vektoren, die seine Spalten bilden. Jede Spalte ist benannt. Der Befehl

```
Tabelle <- data.frame(Geschlecht = c("m", "w", "w"), Alter = c(24,32,20))
```

erzeugt ein `data.frame` namens `Tabelle`, bestehend aus den Spalten `Geschlecht` und `Alter`. Auf die Spalte `Alter` kann mittels `Tabelle$Alter` zugegriffen werden. Der Befehl `attach(Tabelle)` macht die einzelnen Spalten als Variablen verfügbar.

Externe Tabellen können im Wesentlichen aus `.csv`- und `.txt`-Dateien eingelesen werden. Der Befehl

```
Tabelle2<-read.table("http://www.mathe.de/beispiel.txt",  
  sep=";", header=TRUE, dec=",")
```

weist der Variable `Tabelle2` den Inhalt der angegebenen Datei zu, in der die Spalten durch Semikola getrennt sind (`sep=";"`), die in der ersten Zeile die Spaltenentitel enthält (`header=TRUE`), und in der Dezimalzahlen durch ein Komma getrennt werden (`dec=","`). Die Zeilen der Tabelle entsprechen den Zeilen in der Datei. Alle Argumente außer dem Dateinamen sind optional. Werden diese weggelassen, nimmt R Standardwerte dafür an, z.B. werden Leerzeichen und Tabulatoren als Separatoren zwischen Spalten angesehen (dies entspräche der Angabe von `sep=""`). Welche Standardwerte R einsetzt, kann der Hilfedatei entnommen werden, die über `help(read.table)` aufgerufen wird.

## Zeitreihen

Bei einer Zeitreihe wird ein Objekt hinsichtlich eines Merkmals über einen Zeitraum beobachtet. In R erzeugt man eine Zeitreihe mit dem Befehl `ts(data,start,frequency)`.

`data` ist dabei der Datenvektor mit den Beobachtungen, `start` gibt den Startzeitpunkt der Zeitreihe an, `frequency` die Anzahl der Beobachtungen innerhalb einer Zeiteinheit.

Beispiel: Wird ab Februar 1997 jeweils am Monatsanfang die Temperatur um 9 Uhr gemessen, so setzt man `start = c(1997,2)` (Start im 2. Monat im Jahr 1997) und `frequency = 12` (weil es 12 Monate im Jahr gibt).

Der Verlauf einer Zeitreihe  $x$  lässt sich mit Hilfe der Funktion `plot.ts(x)` visualisieren. Die Beschriftung der x- bzw. y-Achse geschieht durch die optionalen Argumente `xlab` bzw. `ylab`.

## Histogramme

Ein Histogramm ist eine graphische Darstellung der Häufigkeitsverteilung von Messwerten. Die Messwerte werden dabei in Klassen eingeteilt (x-Achse) und die Anzahl der Messwerte pro Klasse werden als Balken auf die y-Achse eingetragen. In R erzeugt man ein Histogramm mit dem Befehl `hist(x, breaks)`.

$x$  ist dabei der Vektor der Messwerte und `breaks` kann entweder ein Vektor sein, der die Klassengrenzen festlegt, oder eine natürliche Zahl, die die Anzahl der Klassen angibt. Die Beschriftung der x- bzw. y-Achse geschieht analog zum `plot.ts` Befehl. Die Überschrift kann durch das Argument `main` geändert werden.

## Weitere wichtige Befehle

Eine Übersicht über wichtige R-Befehle findet sich auf der Homepage des Praktikums. Auf den folgenden Zetteln werden ebenfalls weitere Befehle vorgestellt. Bei Fragen wenden Sie sich gerne an die Betreuer.

## Aufgaben

### Aufgabe 1. (5 Punkte)

Das dritte Keplersche Gesetz über die Planetenbewegungen besagt, dass die Quadrate der Umlaufzeiten der Planeten proportional zur dritten Potenz ihrer großen Bahnhalbachsen sind. Für unser Sonnensystem liegen folgende Werte vor:

Planet	Große Halbachse in km	Umlaufzeit in Erdenjahren
Merkur	57909175	0,205
Venus	108208930	0,615
Erde	149597890	1,000
Mars	227936640	1,881
Jupiter	778412020	11,862
Saturn	1426725400	29,447
Uranus	2870972200	84,017
Neptun	4498252900	164,791

- Geben Sie die Daten als `data.frame` ein, den sie der Variablen `Kepler` zuweisen. Benennen Sie die Spalten wie in der Tabelle (verwenden sie Punkte anstelle von Leerzeichen in den Spaltennamen).
- Überprüfen Sie das Keplersche Gesetz, in dem Sie alle 8 Quotienten berechnen.
- In der Literatur wird die Keplersche Konstante für unser Sonnensystem mit dem Wert

$$C = 2,97 \cdot 10^{-19} \frac{s^2}{m^3}$$

angegeben. Rechnen Sie die gegebenen Daten in die entsprechenden Einheiten um (Ein Jahr habe dafür 365,25 Tage). Bilden Sie einen Vektor mit den absoluten Abweichungen der errechneten Werte vom Literaturwert.

### Aufgabe 2. (5 Punkte)

Auf der Praktikums-Homepage finden Sie mehrere Dateien.

- (a) Lesen Sie die Dateien `Test1.txt`, `Test2.txt` und `Test3.txt` in R ein. Geben Sie an, welche Parameter genutzt werden müssen, damit Sie jeweils ein `data.frame` mit drei Spalten, bezeichnet mit `A`, `B` und `C` erhalten, in dessen Spalten die natürlichen Zahlen von 1 bis 7, ihre Quadrate und ihre dritten Potenzen stehen.
- (b) Berechnen Sie die Summen der ersten 7
  - natürlichen Zahlen (ohne Null...),
  - Quadratzahlen,
  - dritten Potenzen von natürlichen Zahlen.
- (c) Lesen Sie die Datei `Liste.txt` von der Praktikums-Homepage ein, und weisen Sie sie der Variablen `Liste` zu. Machen sie `Spalte1` als Variable verfügbar. Nutzen Sie R-Funktionen, um anzuzeigen
  - wieviele Einträge `Spalte1` besitzt
  - wieviele Einträge von `Spalte1` größer als 1 sind
  - welche Werte der größte bzw. kleinste Eintrag von `Spalte1` haben.

### Aufgabe 3. (5 Punkte)

Auf der Praktikums-Homepage finden Sie die Datei `deutsche_bank.csv`. Sie enthält die wöchentlichen Aktienkurse der Deutschen Bank vom 1.1.2003 bis 2.9.2009.

1. Lesen Sie die Datei mittels `read.csv` ein und weisen Sie ihr einen Variablennamen zu. Lassen Sie sich anschließend eine Zusammenfassung der Daten mit Hilfe von `summary` ausgeben.
2. Erzeugen Sie aus der Spalte `Close` eine Zeitreihe. Geben Sie dabei auch die Argumente `start` und `frequency` an. Plotten Sie anschließend den Verlauf des Aktienkurses und beschriften Sie die Achsen geeignet.
3. Berechnen Sie die Renditen  $R(t) = \frac{A(t) - A(t-1)}{A(t-1)}$  und erstellen Sie ein Histogramm aus diesen Werten. Wählen Sie dabei geeignete Klassengrenzen, um ein aussagekräftiges Histogramm zu erhalten. Beschriften Sie dabei die Achsen wieder geeignet und wählen sie eine passende Überschrift.