Statistische Software (R)

Paul Fink, M.Sc.

Institut für Statistik
Ludwig-Maximilians-Universität München

Daten einlesen

Daten

Daten als eine Zusammenstellung von Informationen zu einem bestimmten Thema.

Beispiel:
Am Sonntag 11.05.2014 hatte es in München um Punkt 12 Uhr 15°C, während es am Samstag 11.05.2014 in München um Punkt 12 Uhr noch 20°C waren. Am Freitag zuvor waren es zur selben Zeit am selben Ort allerdings erst 12°C. . . .

Ist das ein gutes Format um die relevanten Informationen zu beschreiben?

Viele *unnötige* Wörter im Bezug auf Information

Informationen doppelt ⇒ **DRY-Prinzip**

Daten

Es geht hier um Temperaturbeobachtungen (gemessen in °C)
an einem bestimmten Ort (irgendwo in München)
zu jeweils einer bestimmten Zeit (12 Uhr)

Diese Informationen sind fest und ändern sich nicht mit der Beobachtung. Solche Art von Informationen heißen **Metadaten**

Die eigentlichen variablen Daten kann man nun einfach in einer Tabelle darstellen, jede Beobachtung eine Zeile

Wochentag	Datum	Temperatur
Sonntag	11.05.2014	15
Samstag	10.05.2014	20
Freitag	09.05.2014	12
⋮	⋮	⋮

Von statistischem Interesse meist nur die eigentlichen variablen Daten.

Datenformate - Textformat

Jedes Zeichen der Information wird als Text gespeichert.

Spezielle Zeichen trennen die Spalten innerhalb einer Zeile

Vorteil: Sehr einfaches Format, einfach zu editieren

Nachteil: Geht in der Regel nur für Datensätze im sog. Rechteck-Schema

Beispiel: CSV- oder Fixed-Width-Format

Datenformate - Binärformat

Zeichen werden unterschiedlich gespeichert, je nach Typ: Ganze Zahl, reelle Zahl, Text, ... $\Rightarrow$ *Intelligentes* Speichern

(Sehr) Kleine Ungenauigkeiten beim Speichern von Zahlen

Vorteil: Spart Platz, kann zur Vermeidung von Redundanz verwendet werden

Nachteil: Man braucht spezielle Software dazu, die auflöst, was als was gespeichert ist.

Beispiel: SPSS sav-Dateien, Excel-Spreadsheets

Wiederholung: Data.frame

Die wohl wichtigste Struktur zur Haltung von Daten im üblichen Rechteckschema, wo die Beobachtungen in den Zeilen und die Variablen in den Spalten dargestellt werden, ist die Datenmatrix. In R wird diese `data.frame` genannt.

Data.frames sind spezielle Listen, deren Elemente wiederum Vektoren gleicher Länge sind. Data.frames sind **DIE** typische Datenstruktur in R.

Allerdings kann man auch auf einzelne Elemente / Blöcke wie bei einer Matrix zugreifen.

Pfadangaben

Jede Datei auf dem Computer liegt an einem Ort in der Ordner-Baum-Struktur

Verweis dahin über den sogenannten *Pfad*

Jeder Ordnerebene wird über Pfadtrenner verbunden, bei MS Windows \, bei Mac und *nix-Systemen /

```
> pfad <- "Der/Pfad/zu/meiner/Datei"
> pfad
[1] "Der/Pfad/zu/meiner/Datei"
> pfad_ms <- "Der\\Pfad\\zu\\meiner\\Datei" # geht nur unter Windows
> pfad_ms
[1] "Der\\Pfad\\zu\\meiner\\Datei"
> pfad_r <- file.path("Der", "Pfad", "zu", "meiner", "Datei")
> pfad_r
[1] "Der/Pfad/zu/meiner/Datei"
```


Relative vs. Absolute Pfadangabe

Bei der absoluten Angabe muss man immer im Wurzelverzeichnis starten (Laufwerksbuchstabe bei Windows)

```
> "C:/Der/absoulte/Pfad/zu/meiner/Datei"
[1] "C:/Der/absoulte/Pfad/zu/meiner/Datei"
```

Bei der relativen Angabe nimmt R das aktuelle Arbeitsverzeichnis und geht dann herunter

```
> "Der/relative/Pfad/zu/meiner/Datei"
[1] "Der/relative/Pfad/zu/meiner/Datei"
> # entspricht
> # file.path(getwd(),"Der/relative/Pfad/zu/meiner/Datei"")
```

Spezielle Verzeichnisnamen:

`..` : Ordnernamen der Ebene oberhalb der aktuellen

`.` : Aktueller Ordner

Einlesen in R - Textformat

R kann gut umgehen mit Textformaten, für Binärformate braucht man in der Regel spezielle Packages.

`Tipp`: Vermeiden Sie das Einlesen von Binärformaten!

DIE Funktion in R zum Einlesen von Textformaten ist `read.table`

Es gibt weitere Funktionen zum einlesen von anderen Textformaten, die aber in der Regel nur die Funktion `read.table` mit vorgegeben Argumenten aufrufen, zum Beispiel `read.csv`

Hilfe: `?read.table`

Die Funktion liefert nach dem Einlesen ein Objekt von Typ `data.frame` zurück

Einlesen in R - Textformat

Standardmäßig wandelt R beim Einlesen die Spalten in geeignete Formate um.

Spalten mit nur Zahlen werden zu `numeric`, Text wird in einen Faktor umgewandelt.

Verhalten kann man ändern über z.B. `as.is` oder/und `col.classes` Argument beim Aufruf von `read.table`

Über das `header` Argument kann man steuern ob die erste Zeile als Name der Spalten interpretiert werden soll

Wichtig beim Einlesen ist, dass das richtige Trennzeichen verwendet wird - anzugeben über `sep`-Argument

Einlesen in R - Binärformat

Häufig sind die Daten in Excel-Files abgespeichert. Dann kann man sie als `csv`-Datei speichern und mit den Textformat-Funktionen einlesen (empfohlen), oder direkt einlesen mit der Funktion `read.xls` aus dem Package `gdata`

Daten aus SPSS sav-Dateien kann man mit der Funktion `read.spss` aus dem Paket `foreign` einlesen

Wenn nur irgendwie möglich, vermeiden aus Binärformaten einzulesen!