Statistics and Data analysis using \texttt{R}

Valentin Wimmer
contact: Valentin.Wimmer@wzw.tum.de
Lehrstuhl für Pflanzenzüchtung, TUM

December 2010
Contents

1 Introduction 2
  1.1 Installing R . . . . . . . . . . . . . 3
  1.2 Working with R . . . . . . . . . . 3
  1.3 First steps . . . . . . . . . . . . . 4
  1.4 R help . . . . . . . . . . . . . . 5

2 Data management 6
  2.1 Data structure . . . . . . . . . . 7
  2.2 Read in data . . . . . . . . . . . . 11
  2.3 Data manipulation . . . . . . . . 14
  2.4 Data export . . . . . . . . . . . . 16

3 Descriptive Statistics 16

4 Graphics 19
  4.1 Classic graphics . . . . . . . . . 19
  4.2 Controlling the appearance of
      a plot . . . . . . . . . . . . . . . 22

5 Probability Distributions 27
  5.1 R as a set of statistical tables . . 27

6 Hypothesis Testing (Inference) 32
  6.1 Two-sample t-test for equality
      of means . . . . . . . . . . . . . . 34
  6.2 Wilcoxon test . . . . . . . . . . . . 36
  6.3 Chi-squared-test . . . . . . . . . . 36
  6.4 Kolmogorov-Smirnov-Test . . . . 38

7 ANOVA and linear Regression 40
  7.1 ANOVA . . . . . . . . . . . . . . . 40
  7.2 Linear Regression . . . . . . . . 44
  7.3 Multiple Linear Regression . . . 53
  7.4 Categorial Covariates . . . . . . 59

8 Generalized Linear Regression 64
  8.1 Logistic Regression . . . . . . . . 64
  8.2 The Generalized Linear Model . 74

9 Linear Mixed Models 79

Aims of this course

- Basic knowledge about R
- Overview over important and useful R functions
- Getting a feeling for the possibilities of R for data analysis and statistics
- Learn to analyse own data with R
- Writing own R functions

About this course

R commands are displayed as

```
R> 5 + 5
```

R output is displayed as

```
[1] 10
```

important notes

useful hints

description of datasets
1 Introduction

What is R?

- R is an environment for statistical computing and graphics
- The root of R is the S-Language
- Today R is the most important statistical software
- Open source, everybody can contribute to R
- Library of add-on R packages
- R package: collection of functions, documentations, data and examples
- The main distribution is maintained by the R Development Core Team

Advantages of R

- Rapid development
- Open Source and thus no "Black Box". All algorithms and functions are visible in the source code.
- No fees
- Many systems: Macintosh, Windows, Linux, ...
- Quite fast
- Many new statistical methods available as R-packages

Disadvantages of R

- Open Source: compatibility with older versions not so important
- No real quality check of the contents of R-packages
- No graphical user interface
- Error message often not useful
- There exist faster programs for special problems
1.1 Installing R

Installing R

- The base distribution is available under www.r-project.org
- The packages can be installed direct with R or www.CRAN.r-project.org or www.bioconductor.org
- There exist different versions for each system
- *.zip for Windows and *.tar.gz for Unix
- For windows, use the Precompiled binary distributions of the base system

1.2 Working with R

Working with R

- Use a text-editor, e.g. Tinn-R to save your scripts
- Make comments to your code (comment lines start with #):
- Set work directories
  
  \[ R> \text{getwd}() \]
  \[ R> \text{setwd}(\"\") \]

- Install an add-on packages
  
  \[ R> \text{install.packages}(\"doBy\") \]

Specifying a path in R

The path has to be given in the form "C:\..." or "C://..."

Working with R

- In R commands are evaluated directly and result printed on screen
- Your input may have more than one line
- Different input in the same line is separated with ";"
- Available packages, functions or objects are available in the R workspace
- To list the available objects, type
  
  \[ R> \text{ls}() \]

- Workspace can be saved every time using \texttt{save} or when R session is finished
1.3 First steps

First steps

- Use R as a calculator

  \[
  \texttt{R> print(sqrt(5) * 2)}
  \]

  \[
  \texttt{[1]} \texttt{ 4.472136}
  \]

- Assign a value to an object, return value of an object

  \[
  \texttt{R> a <- 2}
  \]

  \[
  \texttt{R> a + 3}
  \]

  \[
  \texttt{[1]} \texttt{ 5}
  \]

  \[
  \texttt{R> b <- a}
  \]

  \[
  \texttt{R> b}
  \]

  \[
  \texttt{[1]} \texttt{ 2}
  \]

- Load an add-on R package

  \[
  \texttt{R> library(MASS)}
  \]

First steps

- Operators: +, –, *, /, & & , |, ||

- Comparison: ==, !=, >, >=, <, <=

- Allocation

  \[
  \texttt{R> a <- 2}
  \]

  \[
  \texttt{R> a <- 2}
  \]

  \[
  \texttt{R> a = 2}
  \]

Note that R is case sensitive. Names of objects have to start with a letter.

Other numeric functions
1.4 R help

**R help**

- The `help-function`
  
  ```r
  R> help(mean)
  R> `?` (mean)
  ```

- Overview over an add-on package
  
  ```r
  R> help(package = "MASS")
  ```

- **R manuals**
  - *An Introduction to R*
  - *R Data Import/Export*
  - *R Installation and Administration*
  - *Writing R Extensions*

- [www.rseek.org](http://www.rseek.org)

better use [www.rseek.org](http://www.rseek.org) than [www.google.de](http://www.google.de)

**Exercises**

- Set your work directory
- Define an object `course` and assign the value "Using R"
- Type `ls()` to see the content of your workspace
- Close your R session and save the workspace
- Open the saved workspace with R
- Type `ls()` to see the content of your workspace
- Install add-on package `lme4` from CRAN
- Load the package in your workspace
- Look at the help of the package
Exercises

- Calculate the following with R

\[(\sqrt{5 - 2})^2 - 6 + 5 \cdot e^2\]

- Look at the help file of `floor()` and `ceiling()`.

- What is the result?

```r
R> a <- (4 <= floor(4.1))
R> b <- (5 > ceiling(4.1))
R> a != b
```

- What is the result (give a solution)?

```r
R> (5 < 3) | (3 < 5)
R> (5 < 3) || (3 < 5)
R> (5 < 3) | c((3 < 5), c(5 < 3))
R> (5 < 3) | c((3 < 5), c(5 < 3))
R> (5 < 3) & c((3 < 5), c(5 < 3))
R> (5 < 3) && c((3 < 5), c(5 < 7))
R> (5 < 3) && c((3 < 5), c(8 < 7))
```

2 Data management

**iris dataset**

Throughout this course we will use several datasets coming with R. The most famous one is Fisher’s `iris` data. To load it, type

```r
R> data(iris)
R> head(iris)
```

```
Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1 5.1 3.5 1.4 0.2 setosa
2 4.9 3.0 1.4 0.2 setosa
3 4.7 3.2 1.3 0.2 setosa
4 4.6 3.1 1.5 0.2 setosa
5 5.0 3.6 1.4 0.2 setosa
6 5.4 3.9 1.7 0.4 setosa
```

For more information, type

```r
R> help(iris)
```
The iris data

iris data
This data set contains the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica.

- Sepal.Length [metric]
- Sepal.Width [metric]
- Petal.Length [metric]
- Petal.Width [metric]
- Species [nominal]

2.1 Data structure

Data structure
R is an object-oriented programming language, so every object is instance of a class. The name of the class can be determined by

R> class(iris)

[1] "data.frame"

The most important classes are
- character
- numeric
- logical
- data.frame
- matrix
- list
- factor
Data structure

- Multiple numbers can be combined in a vector
  \[
  R> c(4, 2, 5)
  \[
  [1] 4 2 5
  \]
- Another important object is a matrix
- A matrix is a rectangular pattern with \( n \) rows and \( p \) columns
- A matrix is created as
  \[
  R> (m <- matrix(data = c(2, 5, 6, 3, 4, 3), nrow = 2, ncol = 3))
  \[
  [,1] [,2] [,3]
  [1,]  2  6  4
  [2,]  5  3  3
  \]

Computations with matrices

- transpose the matrix: \( m^T \)
  \[
  R> t(m)
  \[
  [,1] [,2]
  [1,]  2  5
  [2,]  6  3
  [3,]  4  3
  \]
- product with a constant
  \[
  R> m * 4
  \[
  [,1] [,2] [,3]
  [1,]  8 24 16
  [2,] 20 12 12
  \]
- matrix product
  \[
  R> t(m) %*% m
  \[
  [,1] [,2] [,3]
  [1,] 29 27 23
  [2,] 27 45 33
  [3,] 23 33 25
  \]
- Alternative crossprod() method
• inverse of a matrix $m^{-1}$

```r
R> (m <- matrix(c(1, 0, 0, 0, 1, 0, 0, 1, 1), ncol = 3))

[,1] [,2] [,3]
[1,] 1 0 0
[2,] 0 1 1
[3,] 0 0 1
```

```r
R> solve(m)

[,1] [,2] [,3]
[1,] 1 0 0
[2,] 0 1 -1
[3,] 0 0 1
```

• diagonal elements of a matrix

```r
R> diag(crossprod(m))

[1] 1 1 2
```

### Assess the data structure

<table>
<thead>
<tr>
<th>command</th>
<th>result</th>
<th>useful for</th>
</tr>
</thead>
<tbody>
<tr>
<td>str</td>
<td>general structure of the object</td>
<td>list, data.frame</td>
</tr>
<tr>
<td>head</td>
<td>show the first 6 lines</td>
<td>data.frame, matrix</td>
</tr>
<tr>
<td>dim</td>
<td>dimension of the object (rows x columns)</td>
<td>data.frame, matrix</td>
</tr>
<tr>
<td>length</td>
<td>length of the object</td>
<td>list, numeric</td>
</tr>
<tr>
<td>nchar</td>
<td>number of characters in one string</td>
<td>character</td>
</tr>
<tr>
<td>summary</td>
<td>important statistical parameters</td>
<td>list, data.frame</td>
</tr>
<tr>
<td>names</td>
<td>variable names</td>
<td>matrix, data.frame</td>
</tr>
<tr>
<td>rownames</td>
<td>row names</td>
<td>matrix, data.frame</td>
</tr>
<tr>
<td>colnames</td>
<td>column names</td>
<td>matrix, data.frame</td>
</tr>
</tbody>
</table>

Table 1: useful commands to assess the structure of an R object

### Class of a S3 object

• To check, if an object has an special class, use

```r
R> obj <- 3
R> is.numeric(obj)

[1] TRUE
```
\textit{R} > \texttt{is.character(obj)}

[1] \texttt{FALSE}

- The class of an object can simply be changed (no formal check of attributes)

\texttt{R} > \texttt{obj2} <- \texttt{obj}
\texttt{R} > \texttt{(obj2} <- \texttt{as.character(obj2))}

[1] "3"

\texttt{R} > \texttt{is.character(obj2)}

[1] \texttt{TRUE}

\texttt{R} > \texttt{is.numeric(obj2)}

[1] \texttt{FALSE}

\textbf{Handling factor-objects}

- A \texttt{factor} is used for objects with repeated levels

- A \texttt{factor} has two arguments: levels and labels

\texttt{R} > \texttt{levels(iris$Species)}

[1] "setosa" "versicolor" "virginica"

\texttt{R} > \texttt{head(labels(iris$Species))}

[1] "1" "2" "3" "4" "5" "6"

- A numeric or a character variable can be converted to a factor with the \texttt{factor()} command and the arguments
  - labels
  - levels
Exercises

Basic

• What is the difference between

\[
R> v <- c("An", "Introduction", "to", "R")
R> nchar(v)
R> length(v)
\]

• What are the first three values of Sepal.Length?

• Define the following matrix in R:

\[
X = \begin{pmatrix}
4 & 12 & 20 & 0.8 \\
5 & 3 & 7 & 1 \\
9 & 11 & 10 & 8
\end{pmatrix}
\]

• What are the dimensions of \(X\)?

• What is the sum of all elements of \(X\)?

Exercises

Advanced

• What are the last three values of Sepal.Length?

• Look at the help file of \texttt{matrix}. Try to find out the use of the \texttt{byrow} argument.

• The variable \texttt{Sepal.Length} should be categorised. We want to have two categories, one up 6 cm and one above 6cm. Create a \texttt{factor} and give labels for the categories.

• How many observations are in each of the three groups?

2.2 Read in data

Read in data

To read in tabulated data, use

\[
R> \text{read.table(file, header = TRUE, ...)}
\]

were \texttt{file} is the name (+path) of the data or use the url. Use \texttt{header=TRUE} if the first row of the data contains the variable names. This function is the default reading function and is suitable for \texttt{txt, dat,...} and returns an object of class \texttt{data.frame}. Use the \texttt{csv}-format for Excel files and use function:

\[
R> \text{read.csv2(file, header = TRUE, sep = ",", dec = ".", ...)}
\]

look at the data \texttt{before} reading it with \texttt{R} (header, decimal symbol, missing values,...)
Creating and indexing complex objects

- Objects can be shared in a vector
- A vector is created with the `c()` command
  
  ```
  R> (myvec <- c(1, "Hallo", 3, 7))
  [1] "1" "Hallo" "3" "7"
  ```
- A vector can contain objects with different classes
- Consecutive numbers are obtained by
  
  ```
  R> 3:9
  [1] 3 4 5 6 7 8 9
  R> 6:-4
  [1] 6 5 4 3 2 1 0 -1 -2 -3 -4
  ```
- Vectors with a regular structure are constructed with the functions `rep` and `seq`
  
  ```
  R> rep(c(1, 5), each = 2, times = 3)
  [1] 1 1 5 5 1 1 5 5 1 1 5 5
  ```
- Entries of vector objects
  
  ```
  R> (o1 <- c(3, 6:8, 10))
  [1] 3 6 7 8 10
  R> o1[2]
  [1] 6
  ```
- Entries of a matrix
  
  ```
  R> (o2 <- matrix(data = 1:12, nrow = 3, ncol = 4))
  ```
Entries of a data.frame

R> iris[2, 5]

[1] setosa
Levels: setosa versicolor virginica

R> iris[3:5, "Species"]

[1] setosa setosa setosa
Levels: setosa versicolor virginica

R> head(iris$Petal.Width)

[1] 0.2 0.2 0.2 0.2 0.2 0.4

Entries of a list

R> mod <- lm(Petal.Width ~ Species, data = iris)
R> mod[["coefficients"]]

(Intercept) Speciesversicolor Speciesvirginica
 0.246       1.080       1.780

R> mod[[1]]

(Intercept) Speciesversicolor Speciesvirginica
 0.246       1.080       1.780
**attaching a data.frame**
The attach() command can be used for datasets. Afterwards the variable of the dataset are attached to the R search path.

You only need to type

```R
R> Petal.Width
```

instead of

```R
R> iris$Petal.Width
```

to refer to the variable Petal.Width in the dataset iris then.

Note that objects can be masked by attaching.

**Reading xls-files**

To read in Excel files, you need the RODBC-library and the following commands:

```R
R> library(RODBC)
R> trac <- odbcConnectExcel("Table.xls")
R> Tnames <- sqlTables(trac)$TABLE_NAME
R> df1 <- sqlFetch(trac, Tnames[1], as.is = TRUE)
R> odbcClose(trac)
```

**Exercises**

- Load the file data2read from the website. Have a look at the file.
- Read in data using the read.table function and save the data as an object with name mydata. What is the class of object mydata?
- Assess the data structure of mydata with the methods str, head, dim, colnames and rownames.
- What are the values of the third row of mydata?
- Attach mydata and give the rows of mydata where x ≥ 8.

**2.3 Data manipulation**

**The swiss data**

**swiss dataset**

Standardized fertility measure and socio-economic indicators for each of 47 French-speaking provinces of Switzerland at about 1888. A data frame with 47 observations on 6 variables, each of which is in percent, i.e., in [0,100].
• **Fertility**: common standardized fertility measure
• **Agriculture**: % of males involved in agriculture as occupation
• **Examination**: % draftees receiving highest mark on army examination
• **Education**: % education beyond primary school for draftees.
• **Catholic**: % catholic (as opposed to protestant).
• **Infant.Mortality**: live births who live less than 1 year.

All variables but Fertility give proportions of the population.

**Data manipulation**

• Replacement in objects
  
  ```R
  R> (v <- 1:9)
  [1] 1 2 3 4 5 6 7 8 9
  R> (v[3] <- 5)
  [1] 5
  ```

• A **data.frame** can also be manipulated with the `fix()` command

• Subset of a data frame. With the following command only the individuals of species "setosa" are selected from the **iris** dataset
  
  ```R
  R> iris.Setosa <- subset(iris, iris$Species == "Setosa")
  ```

• Order the elements of a numeric vector
  
  ```R
  R> order.edu <- order(swiss$Education, decreasing = TRUE)
  ```

• Give the the 3 districts with the highest Education
  
  ```R
  R> swiss[order.edu[1:3], 1:5]
  ```

<table>
<thead>
<tr>
<th>Fertility</th>
<th>Agriculture</th>
<th>Examination</th>
<th>Education</th>
<th>Catholic</th>
</tr>
</thead>
<tbody>
<tr>
<td>V. De Geneve</td>
<td>35.0</td>
<td>1.2</td>
<td>37</td>
<td>53</td>
</tr>
<tr>
<td>Neuchatel</td>
<td>64.4</td>
<td>17.6</td>
<td>35</td>
<td>32</td>
</tr>
<tr>
<td>Rive Droite</td>
<td>44.7</td>
<td>46.6</td>
<td>16</td>
<td>29</td>
</tr>
</tbody>
</table>
2.4 Data export

Data export

- Data of R can be written into a file with the `write.table()` command
- The R workspace can be saved with `save()` command and can be reloaded with `load()`
- Function `xtable()` to create LaTeX-table output

```r
R> library(xtable)
R> xtable(summary(iris), caption = "Summary statistic of iris data")
```

<table>
<thead>
<tr>
<th>Sepal.Length</th>
<th>Sepal.Width</th>
<th>Petal.Length</th>
<th>Petal.Width</th>
<th>Species</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Min. :4.300</td>
<td>Min. :2.000</td>
<td>Min. :1.000</td>
<td>Min. :0.100</td>
<td>setosa:50</td>
</tr>
<tr>
<td>2 1st Qu.:5.100</td>
<td>1st Qu.:2.800</td>
<td>1st Qu.:1.600</td>
<td>1st Qu.:0.300</td>
<td>versicolor:50</td>
</tr>
<tr>
<td>3 Median :5.800</td>
<td>Median :3.000</td>
<td>Median :4.350</td>
<td>Median :1.300</td>
<td>virginica:50</td>
</tr>
<tr>
<td>4 Mean :5.843</td>
<td>Mean :3.057</td>
<td>Mean :3.758</td>
<td>Mean :1.199</td>
<td></td>
</tr>
<tr>
<td>5 3rd Qu.:6.400</td>
<td>3rd Qu.:3.300</td>
<td>3rd Qu.:5.100</td>
<td>3rd Qu.:1.800</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Summary statistic of iris data

Exercises

Construct the following sequences using the functions `rep` or `seq`:

```r
[1] 3 4 5 6 7 3 4 5 6 7 3 4 5 6 7
```

```r
[1] -2.5 -2.0 -1.5 -1.0 -0.5 0.0 0.5 1.0 1.5 2.0
```

```r
[1] 2 2 2 2 3 3 3 3 4 4 4 4 2 2 2 2 3 3 3 3 4 4 4 4
```

```r
[1] 10.0 10.5 11.0 11.5 12.0 12.5 13.0 13.5 14.0 14.5 15.0
```

3 Descriptive Statistics

The C02 data

The `C02` dataset

The C02 data frame has 84 rows and 5 columns of data from an experiment on
the cold tolerance of the grass species Echinochloa crus-galli.
mean(x) compute (trimmed) mean
median(x) compute median (50% quantile)
min(x) compute minimum
max(x) compute maximum
range(x) compute minimum and maximum
quantile(x, probs=c()) compute the probs 100% quantile
sd(x), var(x) compute the standard deviation/variation
cov(x, y), cor(x, y)) compute the covariance/correlation

Table 3: Standard descriptive functions

- **Plant**: an ordered factor with levels $Qn1 < Qn2 < Qn3 < ... < Mc1$ giving a unique identifier for each plant.
- **Type** a factor with levels Quebec Mississippi giving the origin of the plant
- **Treatment** a factor with levels "nonchilled" and "chilled"
- **conc** a numeric vector of ambient carbon dioxide concentrations (mL/L).
- **uptake** a numeric vector of carbon dioxide uptake rates (umol/m^2 sec).

**Cross Tables**

- The `table` command gives the frequencies of a categorial variable, typically it is used with one or more factor-variables

  \[
  R \texttt{> table(CO2$Type)}
  \]

  Quebec Mississippi
  42 42

- A cross-table of two factors is obtained by

  \[
  R \texttt{> table(CO2$Type, CO2$Treatment)}
  \]

  nonchilled chilled
  Quebec 21 21
  Mississippi 21 21

- We have a balanced experimental design

More than 2 factors are possible too

**Descriptive Statistics**
Missing values (NA)

- Missing values are always coded with the symbol NA
- The function `is.na` is used to check if a value is missing
- The function `complete.cases` takes a `data.frame` and returns a logical vector being TRUE when the corresponding observation contains no missing values

```r
R> table(complete.cases(airquality))
FALSE TRUE
   42  111
```

- Missing values often should not be considered in the calculation

```r
R> mean(c(1:5, NA, 8))
[1] NA
```

- The missing values are removed with the `na.rm` argument in many functions as those in Table 3

```r
R> mean(c(1:5, NA, 8), na.rm = TRUE)
[1] 3.833333
```

Excercises

Basic

- Compute the 5%, the 10% and the 75% quantile of the education rate in the `swiss` data.
- Is the distribution of the education rate in the `swiss` data symmetric, left skewed or right skewed?
- Compute the covariance and the correlation between the ambient carbon dioxide concentrations and the carbon dioxide uptake rates in the `CO2` data. How to you interpret the result?
- Compute the correlation coefficient for each Treatment group separately.

Advanced

- Compute the correlation between the following two variables: \( v1=[11,5,6,3,*35,4,4] \), \( v2=[4,6,2,1,6,*7,2,3] \), where "*" indicates a missing value.
- Give ranks to `uptake` and `conc` and compute spearman's correlation coefficient.
4 Graphics

Graphics with R

- Many graphical possibilities and settings in R
- Division in high-level and low-level graphics
- Standard graphics for each object type (try `plot` for a `data.frame`)
- Additional graphics system in the library `lattice`
- Standard graphical function: `plot` to plot y against x
- You can either use `plot(x,y)` or `plot(y ~ x)`

4.1 Classic graphics

Graphics with R: `plot` for metric and metric variables

```r
R> plot(Education ~ Agriculture, data = swiss)
```

![Graph](image)

Boxplot for metric (+ factor)

```r
R> boxplot(Petal.Width ~ Species, data = iris)
```

![Boxplot](image)
Histogram for metric variates

R> hist(swiss$Education)

Histogram of swiss$Education

Bar plot for factors

R> barplot(rowSums(HairEyeColor), col = c("black", "brown", "red", + "gold"))
Pie charts for factors

R> pie(rowSums(HairEyeColor), col = c("black", "brown", "red", "gold"))
Mosaicplot for cross-tables

R> mosaicplot(HairEyeColor[, , "Male"], shade = TRUE, main = "")

4.2 Controlling the appearance of a plot

Main graphic arguments

The following arguments fit for all types of graphics:

- **type**: a string specifying the type of plot. Possible is
  - "p" for points
  - "l" for lines
  - "b" for both
  - "h" for histogram like vertical lines,
  - "s"/"S" for stair steps
  - "n" for no plotting
- **main**: a string specifying the plot title
- **sub**: a string specifying the subs title
- **xlab**: a string specifying the title for the x axis
- **ylab**: a string specifying the title for the y axis
Additional graphic arguments

The following arguments specify the appearance of a plot:

- `xlim`: a vector of 2 specifying minimum and maximum of x axis
- `ylim`: a vector of 2 specifying minimum and maximum of y axis
- `lty`: a numeric giving the line type (1: solid line, 2: dashed line, ...)
- `lwd`: a numeric giving the line width
- `pch`: a string or a numeric giving the plotting character (pch=21 for a filled circle, pch=19 for a solid circle,...)
- `cex`: a numeric specifying the character (or symbol) expansion
- `col`: a vector of strings or numbers specifying the color of the plotting symbols (to use red color you can use `col="red"` or `col=2`)

Set graphical parameters

- To set or query graphical parameters use `par`

  ```r
  R> par(mfrow = c(1, 2))
  ```

- By this commands, the graphics window is divided into 2 parts (1 row, 2 columns)

- Following arguments are useful (additional to the ones on the previous page)
  - `mfrow=c(r,c)` divide the graphics window into r rows and c columns
  - `mar=c(b,l,u,r)` set the figure margin at the bottom (b), left (l), upper(u) and right (r).
  - `oma=c(b,l,u,r)` set the outer figure margin
  - `bg=col` set col as a color for the figure background
  - `las`: set the axis style (0: always parallel to the axis [default], 1: always horizontal, 2: always perpendicular to the axis, 3: always vertical)

Set graphical parameters

Set graphical parameters
Figure 1: The R Plot Area with Outer Margin Area

Figure 2: Multiple Figures by Row (mfrow) with Outer Margin Area
Adding low-level graphics to a plot

- Low-level output can be added to an existing plot
- Other useful low-level functions are:
  - `points` adding points [args: x,y,pch,...]
  - `lines` adding lines [args: x,y,lty,...]
  - `text` adding text [args: x,y,labels]
  - `title` adding a title
  - `abline` adding a line [args: a,b,h,v]
  - `axis` adding an axis [args: side,at,labels,las,...]

Adding low-level graphics to a plot

Adding a legend

```R
R> legend("topright", legend = c("setosa", "versicolor", "virginica"),
+     pch = c(21, 21, 21), col = c(14, "lightgreen", "blue"))
```
Saving graphics

There are two possibilities to save graphics in R:

- Right click in graphic an "save as"
- better alternative: functions **pdf**, **png**, **postscript**

Usage:

```r
R> pdf("C://mypath", width = 5, height = 4)
R> plot(Sepal.Width ~ Sepal.Length, data = iris)
R> title("Iris data")
R> dev.off()
```

- With **pdf()** a graphic device is opened
- and closed with **dev.off()**

Exercises

Basic

- Make a histogram of each of **Sepal.Length**, **Petal.Length**, **Sepal.Width** and **Petal.Width** in one plot for the **iris** data. Give proper axis labels, a title for each subbplot and a title for the whole plot.

- Now we use the **swiss** data. Plot the **Education** rate against the **Agriculture** rate. Use different symbols for the provinces with more catholics and those with more protestant inhabitants. Choose axis limits from 0 to 100. Add a legend to your plot.

- Save this plot as a **pdf**-file.
Add two vertical lines and two horizontal lines to the plot which represent the means of the Agriculture and the Education for both subgroups (Catholic vs Protestant).

Look at the layout function. This is an alternative for par(mfrow).

5 Probability Distributions

5.1 R as a set of statistical tables

Probability Distributions
Several theoretical distributions are included in R

- The probability density function is defined as \( P(X = x) \)
- The cumulative distribution function (CDF) is defined as \( P(X \leq x) \)
- The quantile function is the smallest \( x \) for a given \( q \) where \( P((X \leq x) > q) \)

The functions names consists of a prefix and the probability name. The prefix name "d" is for the density, "p" for the CDF, "q" for the quantile function and "r" for simulation (random samples).

Probability Distributions
The name of the distribution is usually an abbreviation

\[ \text{R> pnorm}(0.5) \]

[1] 0.6914625 gives the value of the CDF of a standard normal distribution \( N(0,1) \) at point 0.5.

Additional arguments can passed to the distribution functions. For example,

\[ \text{R> pnorm}(0.5, \text{mean} = 2, \text{sd} = 3) \]

[1] 0.3085375 gives the value of the CDF of \( N(2,3) \) at point 0.5.

Probability Distributions
Following (and others) distributions are available in R

Visualising probability functions
For visualisations, the curve function is useful
<table>
<thead>
<tr>
<th>Distribution</th>
<th>R name</th>
<th>additional arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal</td>
<td>norm</td>
<td>mean, sd</td>
</tr>
<tr>
<td>binomial</td>
<td>binom</td>
<td>size, prob</td>
</tr>
<tr>
<td>hypergeometric</td>
<td>hyper</td>
<td>m, n, k</td>
</tr>
<tr>
<td>Poisson</td>
<td>pois</td>
<td>lambda</td>
</tr>
<tr>
<td>geometric</td>
<td>geom</td>
<td>prob</td>
</tr>
<tr>
<td>Student’s t</td>
<td>t</td>
<td>df, ncp</td>
</tr>
<tr>
<td>chi-squared</td>
<td>chisq</td>
<td>df, ncp</td>
</tr>
<tr>
<td>F</td>
<td>f</td>
<td>df1, df2, ncp</td>
</tr>
<tr>
<td>gamma</td>
<td>gamma</td>
<td>shape, scale</td>
</tr>
<tr>
<td>beta</td>
<td>beta</td>
<td>shape1, shape2, ncp</td>
</tr>
</tbody>
</table>

Table 4: Probability distributions in R

R> curve(dnorm(x, mean = 4, sd = 0.5), from = 2, to = 8, main = "N(4,0.5) distribution")

Computing quantiles

To compute the 95% quantile of the N(0,2)-distribution, use qnorm()

R> qnorm(0.95, mean = 0, sd = 2)
[1] 3.289707

5.2 Examining the distribution of a set of data

Examining the distribution of a set of data

Given a (univariate) set of data we can examine its distribution in a large number of ways:

- to get a basic summary use summary or fivenum
Quantiles of $N(0,2)$ distribution

```
x
qnorm(x, mean=0, sd=2)
```

$0.95$

3.28

- to display the numbers by stem (a "stem and leaf" plot) use `stem`
- to make a density estimation use `density`
- to plot the empirical cumulative distribution function, use `cdf`
- to compare the quantiles of two univariate data sets $x$ and $y$, use `qqplot(x,y)`
- to compare the empirical quantiles with a normal distribution, use `qqnorm`

Density estimation

The *faithful* dataset
Waiting time between eruptions and the duration of the eruption for the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.

A data frame with 272 observations on 2 variables:

- **eruptions** numeric Eruption time in mins
- **waiting** numeric Waiting time to next eruption (in mins)

The function `density` computes kernel density estimates

An important argument for density estimation is the bandwidth (`bw`)

The `rug` function can be used to add a rug representation to the plot

Density estimation
```r
R> data(faithful)
R> plot(density(faithful$eruptions))
R> rug(faithful$eruptions)

Stem and Leaf plot

R> stem(faithful$eruptions)

The decimal point is 1 digit(s) to the left of the |

16 | 070355555588
18 | 00002223333355577777778888822335777888
20 | 0002223378800035778
22 | 000233578023578
24 | 00228
26 | 23
28 | 080
30 | 7
32 | 2337
34 | 250077
36 | 0000823577
38 | 2333335582225577
40 | 00000335778888800223355557778
42 | 033355557788023335555577778
44 | 0222233557780000000023333357778888
46 | 0000233357700000023578
48 | 0000022335800333
50 | 0370

Empirical cumulative distribution
```
R> plot(ecdf(faithful$eruptions), do.points = FALSE, verticals = TRUE)

R> qqnorm(faithful$eruptions)
R> qqline(faithful$eruptions)
Normal Q-Q plot

Exercises
Basic
• You randomly select a sample of 10 persons in a class. There are 20 women and 30 men in that class. What is the probability to have exactly 3 woman in that sample? Suppose you are sampling with replacement.
• What is the probability if you are sampling without replacement?
• What is $P(X > 3)$ if $X \sim N(0,2)$?
• Would you say, that the eruptions of Old Faithful are normal distributed?
• Plot a histogram together with a density estimation for the eruptions data. Try different values for the bandwidth.

Exercises
Advanced
• Continuous distributions can be visualized with curve. How can categorial distributions be visualized?
• The distribution of the eruption data can be seen as a mixture of two normal distributions. Try to find the mean and variance of these normals manually and plot a histogram together with the distribution of the mixture of normals.

6 Hypothesis Testing (Inference)

HairEyeColor dataset

The HairEyeColor data
Distribution of hair and eye color and sex in 592 statistics students at the University of Delaware. A 3-dimensional array resulting from cross-tabulating 592 observations on 3 variables. The variables and their levels are as follows:

1. Hair levels: Black, Brown, Red, Blond
2. Eye levels: Brown, Blue, Hazel, Green
3. Sex levels: Male, Female
**Statistical Tests**

- *Inference* = process of drawing conclusions about a population based on a sample from the whole population

- Example: Comparison of two samples of randomly selected 20 plants from two fields. The population are all plants on each field

- A test is used to assess the hypotheses
  - $H_0$: null hypothesis (can only be rejected, not validated)
  - $H_1$: alternative hypothesis

- Note that $H_0 \not\in H_1$

- Decision is done after computing a test statistic and comparing it with tabulated theoretical values of the distribution under $H_0$

**Errors in hypothesis testing**

<table>
<thead>
<tr>
<th>Truth</th>
<th>Test</th>
<th>Non-significant</th>
<th>Significant</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H_0$</td>
<td></td>
<td>True Negative</td>
<td>Type-1 Error</td>
</tr>
<tr>
<td>$H_1$</td>
<td></td>
<td>Type-2 Error</td>
<td>True Positive</td>
</tr>
</tbody>
</table>

Table 5: Type-1 and Type-2 errors in hypothesis testing

- Statistical testing relies on control of the type-1 error rate

- the level of test, usually denoted by $\alpha$ is equal to 0.05

- Def. *p-value*: Probability of observing a more extreme than the observed test statistic given $H_0$ is true

**Statistical Tests**

- The type of suitable test depends on the population distribution the question setting:
  - comparing a measure in two normal Populations
    - Student’s t-Test
  - comparing a measure in two non-normal Populations
    - Wilcoxon test
  - testing independence in Contingency Tables
    - Chi-squared-test
  - comparing one sample with a theoretical distribution
    - Kolmogorv-Smirnoff test
6.1 Two-sample t-test for equality of means

Student’s t-Test

Used to test the null hypothesis that the means of two independent populations of size \( n_1 \) respectively \( n_2 \) are the same

\[ H_0 : \mu_1 = \mu_2. \]

The test statistic is computed as

\[ t = \frac{\bar{y}_1 - \bar{y}_2}{s\sqrt{1/n_1 + 1/n_2}}, \]

where \( s \) is the pooled standard deviation.

Under the null hypothesis, the \( t \)-statistic has a Student’s \( t \)-distribution with \( n_1 + n_2 - 2 \) degrees of freedom, thus \( H_0 \) is rejected if \(|t| \geq t_{\alpha/2, n_1+n_2-2} \) (two sided test) or \(|t| \geq t_{\alpha, n_1+n_2-2} \) (one sided test)

Student’s t-Test with R

\[ R> boxplot(uptake \sim Treatment, data = CO2, ylab = "uptake") \]

![Boxplot of estimates of amount of uptake in both treatments.](Image)

Student’s t-Test with R

Student’s t-Test is done with the function \( t.test \). It has the following arguments
Figure 4: Normal Probability plots of estimates of amount of uptake in treatment levels "chilled" and "nonchilled".

- x a (non-empty) numeric vector of data values
- y an optional (non-empty) numeric vector of data values.
- alternative a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less".
- paired a logical indicating whether you want a paired t-test (for dependent samples).
- conf.level confidence level of the interval (default = 0.95).
- formula a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs a factor with two levels giving the corresponding groups.
- data an optional matrix or data frame

Student’s t-Test with R

R> t.test(uptake ~ Treatment, data = CO2, var.equal = FALSE)
**Welch Two Sample t-test**

- data: uptake by Treatment
- \( t = 3.0485, \ df = 80.945, \ p\text{-value} = 0.003107 \)
- alternative hypothesis: true difference in means is not equal to 0
- 95 percent confidence interval:
  - \( 2.382366 \) \( 11.336682 \)
- sample estimates:
  - mean in group nonchilled: 30.64286
  - mean in group chilled: 23.78333

### 6.2 Wilcoxon test

**Wilcoxon test with R**

- In opposite to Student’s \( t\)-Test the Wilcoxon test uses only the ranks of the observations
- No distribution assumptions for the variables

```r
R> wilcox.test(uptake ~ Treatment, data = CO2)
```

**Wilcoxon rank sum test with continuity correction**

- data: uptake by Treatment
- \( W = 1187.5, \ p\text{-value} = 0.006358 \)
- alternative hypothesis: true location shift is not equal to 0

This test is more conservative than a \( t\)-Test with the same data

### 6.3 Chi-squared-test

**Chi-squared-test**

A sample of \( n \) observations in two nominal (categorial) variables is arranged in a Contingency Table Under \( H_0 \) the row variable \( x \) and the column variable \( y \) are independent, thus the expected values \( E_{jk} \) for cell \((j,k)\) can be computed as \( E_{jk} = n_j.n_k/n \).

<table>
<thead>
<tr>
<th>( x )</th>
<th>1</th>
<th>...</th>
<th>( c )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( n_{11} )</td>
<td>...</td>
<td>( n_{1c} )</td>
</tr>
<tr>
<td>2</td>
<td>( n_{21} )</td>
<td>...</td>
<td>( n_{2c} )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>( r )</td>
<td>( n_{r1} )</td>
<td>...</td>
<td>( n_{rc} )</td>
</tr>
<tr>
<td>( n_1 )</td>
<td>( ... )</td>
<td>( n_c )</td>
<td>( n )</td>
</tr>
</tbody>
</table>

Table 6: The general \( r \times c \) Contingency Table.

- \( y \) are independent,
**Chi-squared-test**

The test statistic for the chi-squared-test is

\[ \chi^2 = \sum_{j=1}^{r} \sum_{k=1}^{c} \frac{(n_{jk} - E_{jk})^2}{E_{jk}}. \]

This value is compared with a \( \chi^2 \)-distribution with \((r - 1)(c - 1)\) degrees of freedom.

\( \chi^2 \)-distribution
Chi-squared-test with R

Contingency Table of Hair and Eye Color of male statistic students

```r
R> HairEyeColor[, , "Male"]

<table>
<thead>
<tr>
<th>Eye</th>
<th>Hair</th>
<th>Brown</th>
<th>Blue</th>
<th>Hazel</th>
<th>Green</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Black</td>
<td>32</td>
<td>11</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Brown</td>
<td>53</td>
<td>50</td>
<td>25</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Red</td>
<td>10</td>
<td>10</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Blond</td>
<td>3</td>
<td>30</td>
<td>5</td>
<td>8</td>
</tr>
</tbody>
</table>
```

The chi-squared-test is performed with the function `chisq.test`. The argument is simply a contingency table (obtained with the function `table`)

```r
R> chisq.test(HairEyeColor[, , "Male"])

Pearson's Chi-squared test

data: HairEyeColor[, , "Male"]
X-squared = 41.2803, df = 9, p-value = 4.447e-06
```

Chi-squared-test with R

The expected values $E_{jk}, j = 1, \ldots, r, k = 1, \ldots, c$ are obtained as follows:

```r
R> chisq.test(HairEyeColor[, , "Male"])$expected

<table>
<thead>
<tr>
<th>Eye</th>
<th>Hair</th>
<th>Brown</th>
<th>Blue</th>
<th>Hazel</th>
<th>Green</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Brown</td>
<td>50.22939</td>
<td>51.76703</td>
<td>24.089606</td>
<td>16.913978</td>
</tr>
<tr>
<td></td>
<td>Red</td>
<td>11.94265</td>
<td>12.30824</td>
<td>5.727599</td>
<td>4.021505</td>
</tr>
<tr>
<td></td>
<td>Blond</td>
<td>16.15771</td>
<td>16.65233</td>
<td>7.749104</td>
<td>5.440860</td>
</tr>
</tbody>
</table>
```

6.4 Kolmogorov-Smirnov-Test

Kolmogorov-Smirnov-Test

This test is either used to compare to variables $x$ and $y$ or to compare one variable with a theoretical distribution, e.g. the normal distribution. In R, this test is implemented in the function `ks.test()`. The arguments are

- **x** a numeric vector of data values.
- **y** either a numeric vector of data values, or a character string naming a cumulative distribution function or an actual cumulative distribution function such as `pnorm`.
- **alternative** indicates the alternative hypothesis and must be one of "two.sided" (default), "less", or "greater".
- ... further arguments passed to the distribution function
Kolmogorov-Smirnov-Test with R

The Fertility-rate in the swiss data should be compared with a normal distribution. Simply use the `ks.test()` for vector of the 47 measures

```r
R> ks.test(swiss$Fertility, "pnorm")
```

**One-sample Kolmogorov-Smirnov test**

data: swiss$Fertility

D = 1, p-value < 2.2e-16

alternative hypothesis: two-sided

- The null hypothesis "distribution is normal" is rejected with $\alpha=0.05$ because data is not scaled
- The KS-Test with scaled data results in

```r
R> ks.test(scale(swiss$Fertility), "pnorm")
```

**One-sample Kolmogorov-Smirnov test**

Data: scale(swiss$Fertility)

D = 0.1015, p-value = 0.7179

alternative hypothesis: two-sided

- Now $H_0$ is not rejected

Exercises

Basic

- The means of `Sepal.Length` should be compared for the species "setosa" and "virginica" in the `iris` data. Think about the assumptions (normal distributions, equal variances) and choose the right test. Is there a significant difference in mean?
  - Compute the p-value with help of the t-statistic on slide 35 on your own. What value would we have with a one-sided hypothesis?
  - Why do we have df=9 in the output of the $\chi^2$-test on slide 38?

Advanced

- It was necessary to scale the data before using the `ks.test`. How can we get the right result without using the `scale` method? Have a look at the help for the function and think about the ...-argument.
- Have a look at slide 38 again and compute the value of the $\chi^2$-statistic on our own.

39
Exercises
Advanced

• You have given the following data. Are the two factors \( m_1 \) [levels "A" and "B"] and \( m_2 \) [levels "A", "B", and "C"] independent?

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>A</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>A</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>B</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>B</td>
<td>B</td>
<td>13</td>
</tr>
<tr>
<td>6</td>
<td>B</td>
<td>C</td>
<td>14</td>
</tr>
<tr>
<td>7</td>
<td>A</td>
<td>C</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>B</td>
<td>B</td>
<td>16</td>
</tr>
</tbody>
</table>

7 ANOVA and linear Regression

ANOVA & linear Regression

• Both ANOVA (Analysis of Variance) and linear Regression are used to analyse the effect of one independent explanatory covariate on a dependent response variable.

• If there is more than one covariate, the expression multiple linear regression is used.

• ANOVA and linear Regression have in common, that the response is assumed to be normal distributed.

• The difference is, that the covariates in ANOVA are factors but in linear Regression no assumption is made for the covariates.

7.1 ANOVA

ANOVA

ANOVA is used in so called factorial designs. Following distinction can be done:

• balanced designs: same number of observations in each cell

• unbalanced designs: different number of observations in each cell
Table 7: balanced design with two treatments $A$ and $B$ with two levels 1 (low) and 2 (high) with 3 measurements for each cell

\[
\begin{array}{ccc}
  & A & \\
  B & \text{low} & \text{high} & \text{Total} \\
  \text{low} & (3.5,5.5,4.5) & (5.5,6.6) & 30.5 \\
  \text{high} & (2,4,1.5) & (8,11,9) & 35.5 \\
  & 20.5 & 45.5 & 66 \\
\end{array}
\]

**ANOVA**

The model for ANOVA with two factors and interaction is

\[
y_{ijk} = \mu + \gamma_i + \beta_j + (\gamma\beta)_{ij} + \epsilon_{ijk}, \tag{1}
\]

with

- $y_{ijk}$ the $k$th measurement made in cell $(i, j)$
- $\mu$ the overall mean
- $\gamma_i$ main effect of the first factor
- $\beta_j$ the main effect of the second factor
- $(\gamma\beta)_{ij}$ the interaction effect of the two factors
- $\epsilon_{ijk}$ the residual error term with $\epsilon_{ijk} \sim N(0, \sigma^2)$

**ANOVA formula in R**

The ANOVA model is specified with a model formula. The *two-way layout with interactions* of equation (1) is written in R as

\[
y \sim \text{A} + \text{B} + \text{A:B},
\]

where $\text{A}$ is the first and $\text{B}$ the second factor. The interaction term is denoted by $\text{A:B}$. An equivalent model formula is

\[
y \sim \text{A*B}.
\]

In both cases, a mean $\mu$ is implicitly included in the model. In case $\mu = 0$ use

\[
y \sim \text{A + B + A:B -1}.
\]
ANOVA

In a ANOVA, the total variance of the observations is partitioned into parts due to the main effects and interaction plus an error term. The hypothesis that there is no significant effect is tested with an $F$-test. The assumptions for the $F$-test are

- The observations are independent of each other.
- The observations in each cell arise from a population having a normal distribution.
- The observations in each cell are from populations having the same variance.

$F$-distribution

ANOVA with R

- Now we want to perform ANOVA with two main effects and interaction for the example from above
- Let us first have a look at the data

ANOVA with R

- The mean plot suggested that there are differences in the means, especially for factor A
- To apply ANOVA to the data we use the function `aov`
- The results is shown in a nice way with the `summary` method
R> plot.design(y ~ A + B + A:B)

Figure 5: Plot of the means for each level of the factors.

R> summary(aov(y ~ A + B + A:B))

Df Sum Sq Mean Sq  F value Pr(>F)
A     1  52.083 52.083 43.8596  0.0001654 ***
B     1   2.083 2.083  1.7544  0.2219123
A:B   1  21.333 21.333 17.9649  0.0028433 **
Residuals 8   9.500 1.187
---
Signif. codes:  0 ***  0.001 **  0.01 *  0.05 .  0.1  1

ANOVA with R

• The resulting ANOVA-table shows that there is a highly significant effect of factor A
• The interaction effect is significant too
• The main effect of B is not significant on 5% level
• The estimated effects are obtained with the coef method

R> coef(aov(y ~ A + B + A:B))

(Intercept)  Ahigh  Bhhigh Ahhigh:Bhhigh
 4.333333   1.500000 -1.833333  5.333333

• Note that the level "low" is treated as reference category due to identification restrictions

43
\( \text{R> interaction.plot}(A, B, y) \)

Figure 6: Interaction plot of \( A \times B \)

**ANOVA with R**

To understand the interaction effect, the function \texttt{interaction.plot()} is very useful.

**Exercises**

**Basis**

- Compute an ANOVA for the \texttt{CO2} dataset. The effect of Treatment and Type on the carbon dioxide uptake rates should be examined. Give an interpretation for the result. Would you use a model with or without interaction?

- Think about the model assumptions. A useful method is to use the \texttt{plot} method with the result of an ANOVA.

**Advanced**

- In a second step, we want to look if the ambient carbon dioxide concentr \texttt{conc} has an effect on \texttt{uptake} too. Check this with an ANOVA. The variable should be dichotomized at the median before using it as a factor.

**7.2 Linear Regression**

**Linear Regression**

Linear Regression model:

\[
y_i = \alpha + x_i \beta + \epsilon_i, \ i = 1, \ldots, n \tag{2}\]
with

- $y_i$ response measure for individual $i$
- $x_i$ covariate measure for individual $i$
- $\alpha$ regression coefficient for the intercept
- $\beta$ regression coefficient for the slope
- $\epsilon_i$ error term with $\epsilon_i \sim N(0, \sigma^2)$

The coefficients $\alpha$ and $\beta$ are estimated by $\hat{\alpha}$ and $\hat{\beta}$ which is the least squares solution:

$$
\hat{\beta} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{3}
$$

$$
\hat{\alpha} = \bar{y} - \beta \bar{x} \tag{4}
$$

One can see that

$$
\hat{\beta} = \frac{\text{Cov}(x, y)}{\text{Var}(x)} = r_{xy} \frac{s_y}{s_x}, \tag{5}
$$

where $r_{xy}$ is the empiric correlation coefficient between $x$ and $y$ and $s_x$ is the standard deviation for $x$ and $s_y$ is the standard deviation for $y$. The coefficient of determination $R^2$ is the proportion of variance that is explained by the statistical model. It gives information about the goodness of fit. It holds that

$$
R^2 = r_{xy}^2 \in [0, 1]
$$

Note, that $R^2$ does not tell whether

- the independent variables are a true cause of the changes in the dependent variable
- the correct regression was used
- the most appropriate independent variable has been chosen
- the model might be improved by using transforming the independent variable
The estimated values are obtained as

$$\hat{y}_i = \hat{\alpha} + x_i \hat{\beta}, \; i = 1, \ldots, n$$  \hspace{1cm} (6)

The variability of the data set is measured through different sums of squares:

$$SS_{\text{tot}} = \sum_{i=1}^{n} (y_i - \bar{y})^2, \text{ the total sum of squares} \hspace{1cm} (7)$$

$$SS_{\text{reg}} = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2, \text{ the regression sum of squares} \hspace{1cm} (8)$$

$$SS_{\text{err}} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2, \text{ the sum of squares of residuals} \hspace{1cm} (9)$$

It holds that

$$SS_{\text{err}} + SS_{\text{reg}} = SS_{\text{tot}} \text{ and } R^2 = \frac{SS_{\text{reg}}}{SS_{\text{tot}}}.$$  

Linear Regression - Residuals

The standard residuals are defined as

$$r_i = \hat{y}_i - y_i, \; i = 1, \ldots, n$$  \hspace{1cm} (10)

- The sum of squares of the residuals is a measure for the fit of the model.
- Residuals are the vertical distances of each point from the regression line.
- The regression is unbiased, so $E(\epsilon) = 0$.
- There should be no structure in the distribution of the residuals $\rightarrow$ look at the residuals.

New York Air Quality Measurements

airquality dataset

Daily air quality measurements in New York, May to September 1973. A data frame with 154 observations on 6 variables:

- **Ozone** [numeric] Mean ozone in parts per billion from 13:00 to 15:00 o’clock at Roosevelt Island
- **Solar.R** [numeric] Solar radiation in Langleys from 08:00 to 12:00 o’clock at Central Park
- **Wind** [numeric] Average wind speed in miles per hour at 07:00 and 10:00 o’clock at LaGuardia Airport
• **Temp** [numeric] Maximum daily temperature in degrees Fahrenheit at La Guardia Airport

• **Month** [numeric] Month (1 - 12)

• **Day** [numeric] Day of month (1 - 31)

### Linear Regression with R

- In our study we want to measure the influence of the temperature on the ozone level
- We use the `airquality` dataset
- We have to think about the causal relationship
- Here **Temp** is the independent and **Ozone** is the dependent variable
- We should make some descriptive analyses before
- An important measure is the correlation coefficient $r$. It can be calculated as

  $$ R > \text{cor(} \text{airquality}\$\text{Temp, airquality}\$\text{Ozone, use = "complete.obs"}) $$

  
  [1] 0.6983603

- A linear Regression can be done with the `lm` function

  $$ R > \text{lm(Ozone} \sim \text{Temp, data = airquality}) $$

  \text{Call:}
  
  \text{lm(formula = Ozone} \sim \text{Temp, data = airquality})

  \text{Coefficients:}
  
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>Temp</td>
</tr>
<tr>
<td>-146.995</td>
<td>2.429</td>
</tr>
</tbody>
</table>

- The "Coefficients" give the least squares estimates for $\alpha$ (Intercept) and $\beta$
- A detailed model summary is obtained with the `summary` method
R> plot(Ozone ~ Temp, data = airquality)

Figure 7: Scatterplot of Temp and Ozone

R> summary(lm(Ozone ~ Temp, data = airquality))

Call:
  lm(formula = Ozone ~ Temp, data = airquality)

Residuals:
   Min     1Q  Median     3Q    Max
-40.7295 -17.4086  -0.5869  11.3062 118.2705

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -146.9955   18.2872  -8.038  9.37e-13 ***
Temp         2.4287    0.2331   10.418  < 2e-16 ***

---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1  1

Residual standard error: 23.71 on 114 degrees of freedom
(37 observations deleted due to missingness)
Multiple R-squared: 0.4877,     Adjusted R-squared: 0.4832
F-statistic: 108.5 on 1 and 114 DF,  p-value: < 2.2e-16
R> plot(Ozone ~ Temp, data = airquality)
R> abline(lm(Ozone ~ Temp, data = airquality))

![Scatterplot](image)

Figure 8: Scatterplot of Temp and Ozone with regression curve

The Regression curve can be displayed in the scatterplot

- It is useful to save the model as an object
  
  ```R
  R> myMod <- lm(Ozone ~ Temp, data = airquality)
  ```

- myMod is of class "lm"
  
  ```R
  R> class(myMod)
  ```

  ```R
  [1] "lm"
  ```

- The structure of myMod can be seen with the `str` method
- Now several method can be applied directly to myMod, e.g.
  
  ```R
  R> coef(myMod)
  ```

  ```R
  (Intercept)     Temp
  -146.995491   2.428703
  ```

- This gives the regression coefficients
The following methods can be applied to an object of class "lm"

- **coef**: returns a vector of length 2 containing the regressions coefficients
- **fitted.values**: returns a vector of length n containing the predicted values
- **residuals**: returns a vector of length n containing the residuals
- **predict**: a method for predicting values with the model
- **summary**: a model summary
- **plot**: a plot for the model diagnostics

Model diagnostics

```r
R> layout(mat = matrix(1:4, 2, 2))
R> plot(myMod)
```

- Three outliers can be identified: observations 30, 62 and 117
- Look at the data

```r
R> airquality[c(30, 62, 117), ]
```

<table>
<thead>
<tr>
<th>Ozone</th>
<th>Solar.R</th>
<th>Wind</th>
<th>Temp</th>
<th>Month</th>
<th>Day</th>
<th>qTemp</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>115</td>
<td>223</td>
<td>5.7</td>
<td>79</td>
<td>5</td>
<td>30</td>
</tr>
<tr>
<td>62</td>
<td>135</td>
<td>269</td>
<td>4.1</td>
<td>84</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>117</td>
<td>168</td>
<td>238</td>
<td>3.4</td>
<td>81</td>
<td>8</td>
<td>25</td>
</tr>
</tbody>
</table>

50
• Compare with the other values

```r
R> fivenum(airquality$Ozone, na.rm = TRUE)
[1] 1.0 18.0 31.5 63.5 168.0
R> fivenum(airquality$Temp, na.rm = TRUE)
[1] 56 72 79 85 97
```

Mark the observations in the scatterplot

```r
R> plot(Ozone ~ Temp, data = airquality[-c(30, 62, 117), ], ylim = c(-10, + 210), xlim = c(40, 120))
R> abline(myMod)
R> points(airquality[c(30, 62, 117), ]$Ozone ~ airquality[c(30, + 62, 117), ]$Temp, col = 2)
```

• Observations with a high Temp are underestimated by our model
• Maybe the true relationship between is not linear but, e.g. the true functional relationship is quadratic
• We fit a second model where the temperature is considered quadratic

```r
R> airquality$qTemp <- airquality$Temp^2
R> cor(airquality$qTemp, airquality$Ozone, use = "complete.obs")
```
R> cor(airquality$Temp, airquality$Ozone, use = "complete.obs")

[1] 0.6983603

R> myMod2 <- lm(Ozone ~ qTemp, data = airquality)
R> myMod2

Call:
  lm(formula = Ozone ~ qTemp, data = airquality)

Coefficients:
      (Intercept)       qTemp
  -57.074043    0.016123

R> summary(myMod2)

Call:
  lm(formula = Ozone ~ qTemp, data = airquality)

Residuals:
   Min     1Q   Median     3Q    Max
-39.7067 -16.7409  -0.9534  10.5436 119.2933

Coefficients:  
  Estimate Std. Error t value Pr(>|t|)
  (Intercept) -57.074043   9.38633  -6.081 1.63e-08 ***
  qTemp       0.016123   0.00148  10.859  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 23.23 on 114 degrees of freedom
(37 observations deleted due to missingness)
Multiple R-squared: 0.5085,    Adjusted R-squared: 0.5042
F-statistic: 117.9 on 1 and 114 DF,  p-value: < 2.2e-16

The new curve displayed in the scatterplot

R> plot(Ozone ~ Temp, data = airquality, ylim = c(-10, 210), xlim = c(40, + 120))
R> x <- 1:120
R> lines(coef(myMod2)[1] + coef(myMod2)[2] * x^2)
**Exercises**

**Basic**

- Which of the models `myMod` or `myMod2` fits better to the data? Give reasons for your choice.

- What could be a problem of the second model?

- In second analysis we want to analyse the effect of the temperature on the wind. Use the `airquality` data, make descriptive analyses and fit a model. Give interpretations for the results.

- Look at the model diagnostics. Are there outliers in the data? Highlight the outliers.

**Advanced**

- Look at the `airquality` data. As you can see, there are 37 missing values for `Ozone`. Predict this values with the `predict` method and with our model `myMod1` and plot them into the scatterplot.

- We want to estimate the effect of the agriculture rate on the education rate in the swiss data. What could be a problem here?

7.3 **Multiple Linear Regression**

**Multiple Linear Regression**

If more than one dependent variable should be considered in one model we need *Multiple Linear Regression*. Assume that $y_i$ represents the value of the response variable of the $i$th observations and that $x_{i1}, x_{i2}, \ldots, x_{iq}$ represent the values of the $q$ explanatory variables. The model is given by

$$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_q x_{iq} + \epsilon_i, \; i = 1, \ldots, n$$

(11)
with $\epsilon_i \sim N(0, \sigma^2)$. It holds that
\[ E(y|x_1, \ldots, x_q) = \beta_0 + \beta_1 x_1 + \ldots + \beta_q x_q. \]

Note, that the linear in multiple linear regression applies to the regression parameters, not the response or independent variables. So also quadratic functions of independent variables fit in this model class.

Multiple Linear Regression

The multiple linear regression model can be written as a common model for all observations as
\[
    y = X\beta + \epsilon, \tag{12}
\]
with
- $y = (y_1, \ldots, y_n)^T$ as the vector of response variables
- $\beta = (\beta_1, \ldots, \beta_q)^T$ as vector of regression coefficients
- $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^T$ as vector of error terms

The design or model matrix $X$ has the following form:
\[
    X = \begin{pmatrix}
    1 & x_{11} & x_{12} & \cdots & x_{1q} \\
    1 & x_{21} & x_{22} & \cdots & x_{2q} \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    1 & x_{n1} & x_{n2} & \cdots & x_{nq}
    \end{pmatrix}
\]
with $n$ rows and $q + 1$ columns.

Multiple Linear Regression

The least squares estimator for $\beta$ is calculated as
\[
    \hat{\beta} = (X^TX)^{-1}X^Ty. \tag{13}
\]
The expectation for $\hat{\beta}$ is given as
\[
    E(\hat{\beta}) = \beta
\]
and the variance is
\[
    \text{Var}(\hat{\beta}) = \sigma^2(X^TX)^{-1}.
\]
The predicted values are
\[
    \hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_q x_{iq}.
\]
Multiple Linear Regression

- In a multiple linear regression model, the parameter $\beta_j$ associated with the explanatory variable $x_j$ is such that $\beta_j$ is the increase in the response variable $y$ when $x_j$ increases by 1, conditional on the other explanatory variables remaining constant.

- If $\beta_j > 0$, the response variable increases if $x_j$ increases, if $\beta_j < 0$ the response variable decreases if $x_j$ increases.

Multiple Linear Regression - Residuals

- The standard residuals are defined as
  $$\hat{e}_i = y_i - \hat{y}_i, \ i = 1, \ldots, n.$$

- The symmetric $n \times n$ prediction-matrix or hat-matrix is defined as
  $$H = X(X^TX)^{-1}X$$

- The standardized residuals are defined as ($h_{ii}$ is the $i$th diagonal element of $H$)
  $$r_i = \frac{\hat{e}_i}{\hat{\sigma} \sqrt{1 - h_{ii}}}.$$

- The studentized residuals are defined as
  $$r_i^* = r_i \left( \frac{n - q - 1}{n - q - r_i^2} \right)^{1/2}$$

<table>
<thead>
<tr>
<th>Source of variation</th>
<th>Sum of squares</th>
<th>Degrees of freedom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td>$\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$</td>
<td>$q$</td>
</tr>
<tr>
<td>Residual</td>
<td>$\sum_{i=1}^{n} (\hat{y}_i - y_i)^2$</td>
<td>$n-q-1$</td>
</tr>
<tr>
<td>Total</td>
<td>$\sum_{i=1}^{n} (y_i - \bar{y})^2$</td>
<td>$n-1$</td>
</tr>
</tbody>
</table>

Table 8: Analysis of variance table for the multiple linear regression model.

The mean square ratio

$$F = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 / q}{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2 / (n - q - 1)}$$

provides an $F$-test of the general hypothesis

$$H_0 : \beta_1 = \ldots = \beta_q = 0.$$ 

Under $H_0$, the statistic $F$ has an $F$-distribution with $q$ and $n - q - 1$ degrees of freedom.
- As in Linear Regression $R^2$ is a measure for the fit of the model. It is computed as

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} = 1 - \frac{\sum_{i=1}^{n} \hat{\epsilon}_i^2}{\sum_{i=1}^{n} (y_i - \hat{y})^2}.$$  

- In linear Regression $R^2 = r_{xy}^2$ and in multiple linear Regression $R^2 = r_{y\hat{y}}^2$ holds.

- Hierarchical ordered models can be compared with $R^2$. Suppose we have two models

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \epsilon_i \quad (M1) \tag{14}$$

$$y_i = \beta_0 + \beta_1 x_{i1} + \epsilon_i \quad (M2). \tag{15}$$

- Now, $R^2_{M1} \geq R^2_{M2}$

- Different models can only be compared with $R^2$, if the have the same response, the same number of parameters and an intercept.

**Multiple Linear Regression**

- Adjusted $R^2$ is a modification of $R^2$ that adjusts for the number of explanatory terms in a model.

- Unlike $R^2$, the adjusted $R^2$ increases only if the new term improves the model more than would be expected by chance.

- The adjusted $R^2$ can be negative, and will always be less than or equal to $R^2$.

- Adjusted $R^2$ is computed as

$$R^2_{adj} = 1 - (1 - R^2) \frac{n - 1}{n - q - 1}.$$  

- Adjusted $R^2$ is useful for model selection.

- In the R summary output, the $R^2$ measure is called Multiple $R$-squared and the adjusted $R^2$ Adjusted $R$-squared.
Multiple Linear Regression with R

- We want to extend the simple Model $\text{Ozone} \sim \text{Temp}$ by an additional quadratic effect of $\text{Temp}$.

- The model call in R is

```r
R> airquality$qTemp <- airquality$Temp^2
R> (mod <- lm(Ozone ~ Temp + qTemp, data = airquality))
```

```r
Call:
  lm(formula = Ozone ~ Temp + qTemp, data = airquality)

Coefficients:
(Intercept)   Temp    qTemp
 305.48577   -9.55060   0.07807
```

- Significant effect of both $\text{Temp}$ and $\text{qTemp}$

```r
R> summary(mod)
```

```r
Call:
  lm(formula = Ozone ~ Temp + qTemp, data = airquality)

Residuals:
  Min     1Q   Median     3Q    Max
-37.619 -12.513  -2.736   9.676 123.909

Coefficients:     Estimate Std. Error t value Pr(>|t|)
(Intercept) 305.48577   122.12182  2.501   0.01380 *
Temp       -9.55060    3.20805  -2.977  0.003561 **
qTemp       0.07807     0.02086   3.743  0.000288 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 22.47 on 113 degrees of freedom
(37 observations deleted due to missingness)
Multiple R-squared: 0.5442,  Adjusted R-squared: 0.5362
F-statistic: 67.46 on 2 and 113 DF,  p-value: < 2.2e-16
```

- The $F$-statistic in the model output compares the model with an intercept model. This is an test for a general effect of the model.
Different, nested model can be compared also. The smaller model has to be part of the bigger model.

If we want to know, if the additional quadratic effect of \texttt{Temp} is meaningful, we compare the models:

\begin{verbatim}
R> mod1 <- lm(Ozone ~ Temp, data = airquality)
R> mod2 <- lm(Ozone ~ Temp + qTemp, data = airquality)
\end{verbatim}

In R, this is done with the \texttt{anova} command:

\begin{verbatim}
R> anova(mod1, mod2)
\end{verbatim}

\textit{Analysis of Variance Table}

\begin{verbatim}
Model 1: Ozone ~ Temp
Model 2: Ozone ~ Temp + qTemp

Res.Df RSS Df Sum of Sq F Pr(>F)
1 114 64110
2 113 57038 1 7071.8 14.010 0.0002876 ***
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
\end{verbatim}

As we can see, the quadratic effect is meaningful.

\textbf{Model Selection}

- More covariates lead always to a equal or better model adaption and thus a higher $SS_{\text{reg}}$.
- Adding random numbers as covariates would also improve the model adaption.
- The task is to find a compromise between model adaption and model complexity.
- This is called \textit{model selection}.

\textbf{Model Selection - Deviance & AIC}

- Deviance: is a quality of fit statistic for a model.
- AIC (Akaike's information criterion): is a measure of the goodness of fit of an estimated statistical model.

\[ AIC = 2k + n(\ln(SS_{\text{Reg}}/n)) \]

- with $k =$ number of explanatory variables + intercept in the model and $n =$ number of observations.
- The idea is to penalize the number of explanatory variables to favour sparse models.
Exercises

Basic

- We want to find the best model to explain the amount of Ozone in the airquality data. Make some descriptive statistic, e.g. scatterplots for each possible covariate and the response and compute the correlation matrix.

- Try different models and compare the with the anova method. Which model would you choose. Take $\alpha = 0.05$. Give an interpretation for the effects.

- Check the model assumptions and look for outliers in the data.

Advanced

- Look at the help of the function step and try to find the best model with this function. Compare the results (Note: look especially at the direction argument).

7.4 Categorial Covariates

Including categorial covariates

- Up to now we have considered only metric covariates in the models.

- As in ANOVA, also categorial covarites are allowed as covariates.

- Additional, interactions between metric and categorial and two or more categorial covariates are possible.

- Covariates are treated as categorial if they are declared as a factor.

- factor covarites are coded to use them in models

- Two main types of coding: dummy coding and effect coding

- A Linear Regression Model with inly categorial covarites is the same as an ANOVA.

Dummy coding

- Dummy coding uses only ones and zeros to convey all of the necessary information on group membership.

- A dummy variable is defined as

\[
    d = \begin{cases} 
    0 & \text{no group member} \\
    1 & \text{group member} 
    \end{cases}
\]
For a factor with \( k \) groups (levels) \( k - 1 \) dummy variables \( d_1, d_2, ..., d_{k-1} \) have to be introduced.

For \( d_1 \), every observation in group 1 will be coded as 1 and 0 for all other groups it will be coded as zero.

We then code \( d_2 \) with 1 if the observation is in group 2 and zero otherwise.

\( d_k \) is not needed because \( d_1, ..., d_{k-1} \) has all of the information needed to determine which observation is in which group.

**Dummy coding**

Consider the following example with observations of a factor variable \( \text{grp} \) with 4 groups (each replicated 2 times) and a metric response measure \( y \).

We have to create \( 4 - 1 = 3 \) dummy variables \( d_1, d_2 \) and \( d_3 \).

The resulting coding is the following:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>grp</td>
<td>d1</td>
<td>d2</td>
<td>d3</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 9: Example of dummy coding

\( d_1 \) is called the reference category.

**Dummy coding**

Means of \( y \) in each group

\[
\begin{array}{cccc}
1 & 2 & 3 & 4 \\
5.5 & 3.5 & 8.0 & 4.5 \\
\end{array}
\]

We fit now a linear model with \( R \). Note that dummy coding is the default for factor covariates and the first level is used as reference category.

```r
R> y <- c(5, 4, 7, 4, 6, 3, 9, 5)
R> grp <- factor(rep(1:4, times = 2))
R> str(grp)
Factor w/ 4 levels "1","2","3","4": 1 2 3 4 1 2 3 4
R> mod <- lm(y ~ grp)
R> coef(mod)
```

60
The intercept is the mean of group 1.

The estimated regression coefficients are just the difference of the intercept.

**Dummy coding**

- We could also define the dummy variables on our own:

  ```R
  R> d1 <- c(0, 1, 0, 0, 1, 0, 0, 0)
  R> d2 <- c(0, 0, 1, 0, 0, 1, 0, 0)
  R> d3 <- c(0, 0, 0, 1, 0, 0, 0, 1)
  ```

- We set up a model and estimate the effects:

  ```R
  R> lm(y ~ d1 + d2 + d3)
  ```

  Call:
  `lm(formula = y ~ d1 + d2 + d3)`

  Coefficients:
  (Intercept)  d1  d2  d3
  5.5  -2.0  2.5 -1.0

- The results are the same

- Dummy coding is implemented in the function `contr.treatment`

**Effect coding**

- An alternative coding scheme is effect coding.

- Effect coding uses only ones, zeros and minus ones to convert all of the necessary information on group membership.

- A *effect variable* for *k* groups is defined as

  \[
  e_j = \begin{cases} 
  1 & \text{member of group } j \\
  -1 & \text{member of group } k & \text{for } j = 1, \ldots, k - 1 \\
  0 & \text{else} 
  \end{cases}
  \]

- For a *factor* with *k* groups (levels) *k - 1* effect variables *e1, e2,...,ek-1* have to be introduced.

- For *e1*, every observation in group 1 will be coded as 1, as -1 for group *k* and 0 for all other groups it will be coded as zero.

- *ek* is not needed because *e1,...,ek-1* has all of the information needed to determine which observation is in which group.
Effect coding

- In our example, the effect coding scheme is the following:

<table>
<thead>
<tr>
<th>y</th>
<th>grp</th>
<th>d1</th>
<th>d2</th>
<th>d3</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

Table 10: Example of effect coding

- Effect coding is implemented in the function `contr.sum`

Effect coding

- We fit now a linear model with R and use the effect coding scheme. This can be specified in the `contrasts` argument of `lm`.

```R
R> lm(y ~ grp, contrasts = list(grp = contr.sum(4)))
```

Call:
`lm(formula = y ~ grp, contrasts = list(grp = contr.sum(4)))`

Coefficients:
```
(Intercept)   gr1   gr2   gr3
5.375  0.125 -1.875  2.625
```

- With effect coding the intercept is equal to the grand mean of all of the observations.
- The coefficients of each of the effect variables is equal to the difference between the mean of the group coded 1 and the grand mean.

Interaction

- As in ANOVA interactions can be used in regression models.
- An interaction term can be specified in the model formula.
- Following notations are possible in a formula:
Exercises

Basic

- We look again to the `airquality` data. We want to analyse the effect of the date on the `Ozone` level. Make first some descriptive plots for `Ozone` and `Month` and `Day`.

- Create a new factor `season` for the season. It should have two levels: spring: from 01-05 to 15-06 and summer from 16-06 to 31-09. Add this new variable to the `airquality` dataset. How many observations are in spring and how many are in summer?

- What is the mean value of `Ozone` in spring, what is the mean value in summer?

- Fit a linear model with `Ozone` as response and `season` as a categorical covariate. Use dummy coding and `spring` as the reference category. Give an interpretation for the result.

Advanced

- Now fit the `lm` and use dummy coding with `summer` as reference category.

- Fit the `lm` and use effect coding.

Exercises

Advanced

- There is the theory, that the ozone level does not depend on the `Month` only but additional also on the day of the week. One assumes that there is a difference between weekdays and weekend.

- Create a new factor variable which contains the information if an observation is made on a weekday or nor not (weekend). Perhaps the functions `as.Date` and `weekdays` are useful.

- Fit a `lm` with the effect of season and weekday + interaction and give an interpretation for the results.
8 Generalized Linear Regression

Dust dataset
The dust data is available under the archive of datasets of the department of statistics at the LMU: http://www.stat.uni-muenchen.de/service/datenarchiv/dust/dust.asc.

The dust data
The data was collected by the Deutsche Forschungsgemeinschaft in the years 1960 to 1977 in a munich factory with 1246 workers. The data.frame with one observation for each worker contains the following variables

- cbr [factor] chronical reaction of bronchia [0: no, 1:yes]
- dust [numeric] dust exposure at workplace (mg/m³)
- smoking [factor] Is the worker a smoker? [0: no, 1:yes]
- expo [numeric] time of exposure (years)

8.1 Logistic Regression

Logistic Regression

- In the dust data our response variable is cbr.
- It has only two possible categories yes and no.
- Multiple linear regression model can be written as

\[
\begin{align*}
\mathbf{y} & \sim N(\mu, \sigma^2) \\
\mu &= \beta_0 + \beta_1 x_1 + \ldots + \beta_q x_q
\end{align*}
\]

- This model is only suitable for continuous response variables with, conditional on the values of the explanatory variables, a normal distribution with constant variance.
- So the model is not suitable for the response variable cbr which is binary.
- Other situations where linear regression fails are count data responses, metric but not normal responses or categorial responses

Logistic Regression

- What we need is a model which takes the special type of the response variable into account
- For binary data \{0,1\} we want to model the response probability \(\pi\) of taking the value 1 directly as a linear function of the explanatory variables.
For example we want the probability that somebody has a chronical reaction of bronchia depending on

- the amount of dust exposure
- the time of exposure
- the smoking status

We have to ensure that our model does not lead to fitted values for the response probability outside $[0,1]$.

**Logistic Regression**

- A suitable transformation is the logistic or logit function which ensures a range of transformed values in $[0,1]$.

- The logistic regression model for the response probability $\pi$ is

$$\text{logit}(\pi) = \log \left( \frac{\pi}{1 - \pi} \right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_q x_q$$  \hspace{1cm} (16)

- The logit of a probability is simply the log of the odds of the response taking the value one.

- Equation (16) can be also write as

$$\pi(x_1, x_2, \ldots, x_q) = \frac{\exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_q x_q)}{1 + \exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_q x_q)}$$

- The logit function can take any real value, but the associated probability always lies in the required $[0,1]$ interval.

**Logistic Regression**

- The regression coefficients in a logist regression model have a different interpretation as in linear regression.

- Equation (16) can be also write as

$$\frac{\pi}{1 - \pi} = \exp(\beta_0) \exp(\beta_1 x_1) \cdot \ldots \cdot \exp(\beta_q x_q)$$

- In a logistic regression model, the parameter $\beta_j$ associated with the explanatory variable $x_j$ is such that $\exp(\beta_j)$ is the odds that the response variable takes one when $x_j$ increases by one conditional on the other explanatory variables remaining constant.

- The regression coefficients are estimated by maximum likelihood (ML).
Logistic Regression with R

- First, we load the data
  
  \[R\> dust \leftarrow \text{read.table}(\text{"http://www.stat.uni-muenchen.de/service/datenarchiv/dust/dust.asc"},\]
  
  \[\quad +\quad \text{header} = \text{\textbf{TRUE}})\]

- We declare `cbr` and `smoking` as factor variables
  
  \[R\> \text{dust}$cbr \leftarrow \text{factor(dust}$cbr, \text{levels} = c(0, 1), \text{labels} = c(\text{"no"},\]
  
  \[\quad +\quad \text{\"yes\")})\]

  \[R\> \text{dust}$smoking \leftarrow \text{factor(dust}$smoking, \text{levels} = c(0, 1), \text{labels} = c(\text{"no"},\]
  
  \[\quad +\quad \text{\"yes\")})\]

- A short summary of the data
  
  \[R\> \text{summary(dust)}\]

<table>
<thead>
<tr>
<th></th>
<th>dust</th>
<th>smoking</th>
<th>expo</th>
</tr>
</thead>
<tbody>
<tr>
<td>no :954</td>
<td>Min. : 0.2000</td>
<td>no :325</td>
<td>Min. : 3.00</td>
</tr>
<tr>
<td>yes :292</td>
<td>1st Qu.: 0.4925</td>
<td>yes :921</td>
<td>1st Qu.: 16.00</td>
</tr>
<tr>
<td>Median : 1.4050</td>
<td>Median :25.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean : 2.8154</td>
<td>Mean :25.06</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3rd Qu.: 5.2475</td>
<td>3rd Qu.: 33.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Max. :24.0000</td>
<td>Max. :66.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- We now fit a logistic model with `cbr` as response variable and `expo` as explanatory variable using the `glm` (generalized linear model) function.

- The code to fit the model is
  
  \[R\> \text{dust glm 1} \leftarrow \text{glm(cbr \sim expo, data = dust, family = binomial())}\]

  \[R\> \text{class(dust glm 1)}\]

  \[[1] \text{"glm" "lm"}\]

- The class "glm" inherits of "lm".
- This model implies a global mean automatically \[\sim 1\]
- The argument `family=binomial()` is used to specify, that logistic regression should be used.
Figure 9: Descriptive boxplots for the dust data.

Figure 10: Descriptive plots for the dust data.
R> layout(matrix(1:2, ncol = 2))
R> cdplot(cbr ~ dust, data = dust)
R> cdplot(cbr ~ expo, data = dust)

Figure 11: Conditional plots for the dust data.

R> mosaicplot(cbr ~ smoking, data = dust, shade = TRUE)
• A description of the model is obtained with the `summary` method

```r
R> summary(dust_glm_1)
```

Call:
`glm(formula = cbr ~ expo, family = binomial(), data = dust)`

Deviance Residuals:
```
Min 1Q Median 3Q Max
-1.1681 -0.7751 -0.6270 -0.4909 2.0688
```

Coefficients:
```
Estimate Std. Error z value Pr(|z|)
(Intercept) -2.258885 0.181545 -12.443 < 2e-16 ***
expo 0.040673 0.006066 6.705 2.01e-11 ***
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
```

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1356.8 on 1245 degrees of freedom
Residual deviance: 1309.9 on 1244 degrees of freedom
AIC: 1313.9

Number of Fisher Scoring iterations: 4

• From the results we see that `expo` has a significant positive effect on the probability of `cbr=="yes"` at the 5% level.

• The value of the regression coefficient is

```r
R> coef(dust_glm_1)["expo"]
```

```
expo
0.04067309
```

• This value is more helpful if we convert to the corresponding values for the odds themselves by exponentiating the estimate

```r
R> exp(coef(dust_glm_1)["expo"])
```

```
expo
1.041512
```

• This means, if the years of exposure is increased by 1 the odds that `cbr=="yes"` increases by 0.38.
The probability of getting a chronic reaction of bronchia with a dust exposure of 10 years is computed as:

\[
R > a <- \text{coef(dust glm 1)[1]}
R > b <- \text{coef(dust glm 1)["expo"]}
R > \exp(a + b \times 10)/(1 + \exp(a + b \times 10))
\]

\[
\begin{array}{l}
\text{(Intercept)} \\
0.1356202
\end{array}
\]

A 95% confidence interval for the not transformed effect is:

\[
R > \text{confint(dust glm 1, parm = "expo")}
\]

\[
\begin{array}{cc}
2.5 \% & 97.5 \% \\
0.02887967 & 0.05267799
\end{array}
\]

Using the exponential transformation, we get for the 95% confidence interval:

\[
R > \exp(\text{confint(dust glm 1, parm = "expo")})
\]

\[
\begin{array}{cc}
2.5 \% & 97.5 \% \\
1.029301 & 1.054090
\end{array}
\]

The fitted probabilities for the `dust glm 1` are fitted as a line in the conditional densityplot using the following code:

\[
R > \text{cdplot(cbr ~ expo, data = dust, ylevels = c("yes", "no"))}
R > x <- \text{seq(from = min(dust$expo), to = max(dust$expo), length = 200)}
R > \text{lines(x, exp(a + b * x)/(1 + exp(a + b * x)), lwd = 2, col = 2)}
\]

The fitted probabilities are computed using the inverse of the logistic function as transformation.
In the next step, we include the next covariate from the \textbf{dust} data. This is the amount of dust exposure at workplace.

We can now fit a logistic regression model that includes both explanatory variables using the code

\begin{verbatim}
R> dust_glm_2 <- glm(cbr ~ expo + dust, data = dust, family = binomial())
\end{verbatim}

The transformed 95\% confidence intervals for the regression coefficients are:

\begin{verbatim}
R> exp(confint(dust_glm_2, parm = c("expo", "dust")))
\end{verbatim}

\begin{verbatim}
  2.5 %  97.5 %
expo 1.027912 1.052967
dust 1.046549 1.145827
\end{verbatim}

Both explanatory variables have a positive effect on the probability of getting a chronic reaction of bronchia

\begin{verbatim}
R> summary(dust_glm_2)
\end{verbatim}
Call:
glm(formula = cbr ~ expo + dust, family = binomial(), data = dust)

Deviance Residuals:
     Min       1Q   Median       3Q      Max
-1.3550  -0.7685  -0.6007  -0.4592   2.1620

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.502380  0.196077 -12.762  < 2e-16 ***
  expo        0.039467  0.006138  6.429   1.28e-10 ***
   dust       0.090799  0.023076  3.935   8.33e-05 ***
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1356.8 on 1245 degrees of freedom
Residual deviance: 1294.7 on 1243 degrees of freedom
AIC: 1300.7

Number of Fisher Scoring iterations: 4

- The predicted values of the second model can be plotted against both explanatory variables in a bubble plot using the following code:

  R> prob <- predict(dust_glm_2, type = "response")
  R> plot(dust ~ expo, data = dust, pch = 21, ylim = c(0, 30), xlim = c(0, 80), cex = prob * 8)

- The size of the points corresponds to the probability of getting a chronic reaction of bronchia.

- To compare dust_glm_1 and dust_glm_2 we use again the anova method.

- This is possible, because we want to compare two nested models.

- For two glm-objects the argument test="Chisq" is necessary to compare to models as two lm objects

  R> anova(dust_glm_1, dust_glm_2, test = "Chisq")
By which factor does the odds of getting a cbr increase, when the exposure increases by 3 years?

Figure 12: Bubble plot of fitted values for a logistic regression model fitted to the dust data

Analysis of Deviance Table

Model 1: cbr ~ expo
Model 2: cbr ~ expo + dust

Resid. Df Resid. Dev Resid. Deviance P(>|Chi|)
1    1244 1309.9
2    1243 1294.7 1  15.267 9.333e-05 ***
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

- Including dust as explanatory variable leads to a significant increase in model adaption.

Exercises

Basic

- After how many years of exposure is the probability of having a chronic reaction of bronchia greater 0.5, 0.9 and 0.95? Use the estimations of model dust(glm_1).

- By which factor does the odds of getting a cbr increase, when the exposure increases by 3 years?
We extend model dust_glm_1 by including the factor smoking as explanatory variable. What is the effect of smoking and is it useful to include this variable in the model?

Advanced

- Plot the fitted probability values against the years of exposure. Fit two lines, one for smokers and one for nonsmokers. Fit the observed values using the function rug.

- Include an interaction of smoking and exposure in the model. Give an interpretation for the results. Is it useful to model this interaction?

Exercises

Advanced

- Plot the fitted probability values against the years of exposure. Fit again two lines, one for smokers and one for nonsmokers.

- If you compare the three explanatory variables and dust, smoking and expo, which ones and which interactions would you include in a model (model selection)?

- Using the predict method on a glm object gives us the probabilities of cbr="yes". We want to predict with our logistic regression model if an person is likely to have cbr or not. For that purpose, every person with a fitted probability of > 0.5 is predicted to have cbr and vice versa. This predicted values can be compared with the real values, e.g. making a 2×2 cross table:

<table>
<thead>
<tr>
<th>real value of cbr</th>
<th>predicted value of cbr</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>yes</td>
<td>yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>True Negative</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>False Negative</td>
<td>True Positive</td>
</tr>
</tbody>
</table>

- Compare the different models by comparing the counts of True Negatives, False Positives, False Negatives and True Positive.

8.2 The Generalized Linear Model

Generalized Linear Model

- The generalized linear model (GLM) is a flexible generalization of ordinary least squares regression.

- The GLM generalizes linear regression by allowing the linear model to be related to the response variable via a link function and by allowing the magnitude of the variance of each measurement to be a function of its predicted value
The GLM consists of three elements.

1. A probability distribution from the exponential family (normal, binomial, Poisson,...).
2. A linear predictor $\eta = X\beta = \beta_0 + \beta_1 x_1 + ... \beta_q x_q$.
3. A link function $g$ such that $E(y) = \mu = g^{-1}(\eta)$.

The link function $g$ describes how the mean response $E(y) = \mu$ is kinked to the explanatory variables through the linear predictor $\eta = g(\mu)$.

### Generalized Linear Models

- Each member of a the exponential family distribution has an own link function and variance function.
- The link function ensures the right range of predicted values for every distribution, i.e. $y > 0$ for $y \sim N(\lambda)$ or $y \in [0, 1]$ for $y \sim \text{Bin}(n, \pi)$.

<table>
<thead>
<tr>
<th>Family</th>
<th>Link g</th>
<th>Variance Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>$\eta = \mu$</td>
<td>$1$</td>
</tr>
<tr>
<td>Poisson</td>
<td>$\eta = \log \mu$</td>
<td>$\mu$</td>
</tr>
<tr>
<td>Binomial</td>
<td>$\eta = \log(\mu/(1 - \mu))$</td>
<td>$\mu(1 - \mu)$</td>
</tr>
<tr>
<td>Gamma</td>
<td>$\eta = \mu^{-1}$</td>
<td>$\mu^2$</td>
</tr>
<tr>
<td>Inverse Gaussian</td>
<td>$\eta = \mu^{-2}$</td>
<td>$\mu^3$</td>
</tr>
</tbody>
</table>

### GLM - Residuals

- The Pearson residual is comparable to the standardized residuals used for linear models and is defined as
  $$r_p = \frac{y - \hat{\mu}}{\sqrt{V(\hat{\mu})}}$$
- The deviance residuals are computed as
  $$r_D = \text{sign}(y - \hat{\mu}) \sqrt{d_i}$$
- with $\sum r_D^2 = \text{Deviance} \ 0 \sum d_i$
Ships Damage Data

The ships data
Data frame giving the number of damage incidents and aggregate months of service by ship type, year of construction, and period of operation for 40 ships. This data set is part of the library MASS

- **type** [factor] levels "A" to "E"
- **year** [factor] year of construction: levels 1960 - 64, 65 - 69, 70 - 74, 75 - 79 (coded as "60", "65", "70", "75").
- **period** [factor] period of operation: levels 1960 - 74, 75 - 79 (coded as "60", "75").
- **service** [numeric] aggregate months of service.
- **incidents** number of damage incidents (response).

Formatting the data

```r
R> library(MASS)
R> data(ships)
R> ships$year <- factor(ships$year)
R> ships$period <- factor(ships$period)
```

Summary of the data

```r
R> summary(ships)
```

<table>
<thead>
<tr>
<th>type</th>
<th>year</th>
<th>period</th>
<th>service</th>
<th>incidents</th>
</tr>
</thead>
<tbody>
<tr>
<td>A:8</td>
<td>60:10</td>
<td>60:20</td>
<td>Min. : 0.0</td>
<td>Min. : 0.0</td>
</tr>
<tr>
<td>B:8</td>
<td>65:10</td>
<td>75:20</td>
<td>1st Qu.: 175.8</td>
<td>1st Qu.: 0.0</td>
</tr>
<tr>
<td>C:8</td>
<td>70:10</td>
<td>Median : 782.0</td>
<td>Median : 2.0</td>
<td></td>
</tr>
<tr>
<td>D:8</td>
<td>75:10</td>
<td>Mean : 4089.3</td>
<td>Mean : 8.9</td>
<td></td>
</tr>
<tr>
<td>E:8</td>
<td>3rd Qu.: 2078.5</td>
<td>3rd Qu.: 11.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Max. : 44882.0</td>
<td>Max. : 58.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Poisson regression

- The response variable **incidents** is count data.
- The response can only take positive values.
- Such a variable is unlikely to have a normal distribution.
• We model a Poisson distribution:

\[ \text{Pois}(y) = \frac{e^{-\lambda \lambda^y}}{y!}. \]

• This type of GLM is known as \textit{Poisson regression}.

• The default link function is the log function.

• We want to measure the effect of aggregated months of service on the number of incidents.

\begin{shaded}
\textbf{Generalized Linear Models with R}
\end{shaded}

\[ \text{R} > \text{plot(incidents ~ service, data = ships, type = "h")} \]

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{plot.png}
\caption{Incidents vs Service}
\end{figure}

\textbf{Generalized Linear Models with R}

• We can apply the model using

\[ \text{R} \> \text{ships.glm.1} \leftarrow \text{glm(incidents ~ service, data = ships, family = poisson())} \]

• The residuals can be obtained by

\[ \text{R} \> \text{res} \leftarrow \text{residuals(ships.glm.1, type = "pearson")} \]

• The \texttt{predict} method for an object of class "glm2" returns by default the predicted values on scale of the linear predictor (\texttt{type="link"}).

• The predicted values on response scale are obtained with the argument \texttt{type="response"}

\[ \text{R} \> \text{pred} \leftarrow \text{predict(ships.glm.1, type = "response")} \]
R> summary(ships_glm_1)

Call:
glm(formula = incidents ~ service, family = poisson(), data = ships)

Deviance Residuals:
     Min       1Q   Median       3Q      Max
-6.0040  -3.1674  -2.0055   0.9155   7.2372

Coefficients:           Estimate  Std. Error   z value Pr(>|z|)
(Intercept) 1.613e+00   7.150e-02  22.55   <2e-16 ***
service     6.417e-05   2.870e-06  22.36   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 730.25  on 39  degrees of freedom
Residual deviance: 374.55  on 38  degrees of freedom
AIC: 476.41

Number of Fisher Scoring iterations: 6

Exercises

Basic

• What is the interpretation of the regression coefficients of model ships_glm_1?
  Think of the interpretation in a logistic regression model here.

• How many incidents would you expect after 10, 100, 1000, 10000 and 100000 month of service?

• Make a residuals vs. fitted values scatterplot for the deviance residuals.

• Look at a fitted values vs. original values scatterplot for the ships_glm_1 model. What could be a problem?

• Add other explanatory variables to the model. Try to find the best model by comparing the models with the anova method?
Exercises

Advanced

• Both the Poisson and binomial distribution have variance functions that are completely determined by the mean. There is no free parameter for the variance (see in the summary output: Dispersion parameter for poisson family taken to be 1).

• Often the empirical variance is bigger than the theoretical variance. This is called overdispersion.

• To deal with this problem, we can use a quasi-likelihood approach. This is implemented in R in the function quasipoisson() which can be passed as a family argument for glm.

• Refit model ships glm 1 with a quasi-likelihood approach and give an interpretation of the results.

9 Linear Mixed Models

Wheat Yield Trial data

The Wheat2 data

The Wheat2 data frame has 224 rows and 5 columns. It summarizes the results of a field trial. This dataset is part of the package nlme.
- **Block** [factor] an ordered factor with levels $4 < 2 < 3 < 1$
- **variety** a factor with 56 levels [ARAPAHOE, BRULE, BUCKSKIN, CENTURA,...]
- **yield** [numeric] the yield of the crop
- **latitude** [numeric] specifies the latitude of the crop on the field
- **longitude** [numeric] specifies the longitude of the crop on the field

- Load the data

  ```r
  R> library(nlme)
  R> data(Wheat2)
  ```

- Summary of the data

  ```r
  R> summary(Wheat2)
  ```

<table>
<thead>
<tr>
<th>Block</th>
<th>variety</th>
<th>yield</th>
<th>latitude</th>
<th>longitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>4:56</td>
<td>ARAPAHOE : 4</td>
<td>Min. : 1.05</td>
<td>Min. : 4.30</td>
<td>Min. : 1.20</td>
</tr>
<tr>
<td>2:56</td>
<td>BRULE : 4</td>
<td>1st Qu.:23.52</td>
<td>1st Qu.:17.20</td>
<td>1st Qu.: 7.20</td>
</tr>
<tr>
<td>1:56</td>
<td>CENTURA : 4</td>
<td>Mean :25.53</td>
<td>Mean :27.22</td>
<td>Mean :14.08</td>
</tr>
<tr>
<td>CENTURK78: 4</td>
<td>3rd Qu.:30.39</td>
<td>3rd Qu.:38.70</td>
<td>3rd Qu.:20.40</td>
<td></td>
</tr>
<tr>
<td>CHEYENNE : 4</td>
<td>Max. :42.00</td>
<td>Max. :47.30</td>
<td>Max. :26.40</td>
<td></td>
</tr>
<tr>
<td>(Other) :200</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- The trial consits of 4 Blocks, each variety is repeated once on each block

  ```r
  R> all(table(Wheat2$Block, Wheat2$variety) == 1)
  [1] TRUE
  ```

**Data situation**

- We have *repeated measures*, i.e. clustered data or a longitudinal study
- The data has the following form

  \[
  (y_{i1}, \ldots, y_{ij}, \ldots, y_{in_j}, x_{i1}, \ldots, x_{ij}, \ldots, x_{in_j}), \quad i = 1, \ldots, m
  \]
- Here we have $m$ clusters and $n_i$ observations for each cluster.
- $y_{ij}$ is the values from observation $j$ in cluster $i$. 

80
Result

- Correlated data
- Observations of the same cluster tend to be more equal than observations of different clusters
- Different sources of variety in the data
  - Within clusters based on the repeated measures. This means the departures of the measures from the mean of the cluster.
  - Between clusters. This means the departures of the means of the clusters from the overall mean.

Goals

Estimation of

- Cluster-effects
- Population-effects
- Correlation structure

Linear Mixed Models

The linear mixed model for repeated measures is given by

\[ y_{ij} = x_{ij}^T \beta + u_{ij}^T b_i + \epsilon_{ij} i = 1, \ldots, m, j = 1, \ldots, n_i \]  \hspace{1cm} (17)

or in matrix notation

\[ y_i = X_i \beta + U_i b_i + \epsilon_i i = 1, \ldots, m \]  \hspace{1cm} (18)

The random effects are assumed to be iid distributed normal

\[ b_i \sim N(0, D), \]
where \( D \) is the covariance matrix of random effects and 
\[
\epsilon_i \sim N(0, \Sigma_i).
\]

Experimental Design

- In an experimental design often blocks are defined.
- We are not interested in the block effects specifically but must account for their effect.
- Therefore, blocks are treated as random effects.
- This can be done in the framework of linear mixed models.

Linear Mixed Models with R

- There are two packages to compute linear mixed models: \texttt{nlme} and \texttt{lme4}
- Both have several advantages and disadvantages
- We first have a look at the \texttt{nlme}-package
- Here, a special object class named "groupedData" exists which is special for cluster data with repeated measures
- We make a groupedData object for the field trial

\[
R> gD <- groupedData(yield \sim 1 \mid Block, data = Wheat2)
\]

\[
R> plot(gD)
\]

- We now fit an linear mixed model using the function \texttt{lme}

\[
R> lmmWheat2 <- lme(gD)
R> summary(lmmWheat2)
\]

\[
\text{Linear mixed-effects model fit by REML}
\text{Data: gD}
\begin{align*}
\text{AIC} & \quad 1515.307 \\
\text{BIC} & \quad 1525.528 \\
\text{logLik} & \quad -754.6535
\end{align*}
\]

\[
\text{Random effects:}
\text{Formula: } ^\sim 1 \mid Block
\]
The random effects are estimated as

\[ R > \text{random.effects(lmmWheat2)} \]

\[
\begin{array}{l}
(\text{Intercept}) \\
4 & -3.8623327 \\
2 & 2.8511949 \\
3 & -0.8737036 \\
1 & 1.8848413 \\
\end{array}
\]

If we compare this with the block means minus the overall mean

\[ R > \text{tapply(Wheat2$yield, Wheat2$Block, mean) - mean(Wheat2$yield)} \]
Effects are shrunken to zero

In the next step we include the position of the crops on the field as fixed effect to your model. This is done by including an main effect for latitude and longitude and an interaction term. Note the blocks are of equal size and arranged in columns with the plants sowed in 3-4 columns in each block. We need an equal range of latitude and longitude in each block, otherwise latitude would be a surrogate for the block effect.

```r
R> Wheat2$latitude2 <- Wheat2$latitude - rep(c(4.3, 17.2, 25.8, 38.7), each = 56)
R> unique(Wheat2$latitude2)
[1] 0.0 4.3 8.6 12.9 4.3 8.6 12.9 4.3 8.6
```

The model is now specified as

```r
R> lmmWheat2_2 <- lme(yield ~ latitude2 * longitude, random = ~1 | Block, data = Wheat2)
```

We do not use a "groupedData"-object now but instead we specify a model formula for the fixed effect and a formula for the random effect. 1|Block means, that a random intercept is fitted for each level of the grouping variable (here Block)

Note, that the grouping variable must be a factor

The estimated random effects are now

```r
R> random.effects(lmmWheat2_2)
(Intercept)
4 -4.73121459
2  2.61928461
3 -0.09549262
1  2.20742260
```

84
**Graphical visualisation of the results**

- We want to visualize our results
- We plot the fitted values for the first block

```r
R> wb1 <- wheat2[wheat2$Block == 1, ]
R> fitb1 <- matrix(data = 0, ncol = length(unique(wb1$latitude2)), + nrow = length(unique(wb1$longitude)), byrow = FALSE)
R> fitb1[16:71] <- fitted(lmmwheat2_2)[1:56]
```

- Graphical visualisation is done with the functions `filled.contour` and `persp` using the following commands:
Figure 15: Fitted values for Wheat yield trial with model \texttt{lmmWheat2_2}.

\begin{verbatim}
R> filled.contour(x = unique(WB1$latitude2), y = sort(unique(WB1$longitude)),
+                z = t(fitB1))
R> persp(x = unique(WB1$latitude2), y = sort(unique(WB1$longitude)),
+       z = t(fitB1), theta = 25, phi = 25)
\end{verbatim}

**package lme4**

- In the \texttt{lme4} library linear mixed models can be estimated with the \texttt{lmer} function

\begin{verbatim}
R> library(lme4)
R> `? (lmer)
\end{verbatim}

- This function uses different arguments
- Fixed and random effects are specified in the same formula
- Fixed effects are given as usual, random effect are given in brackets and the grouping variable is given by a vertical line
- The model from above can be fitted with \texttt{lmer} typing
R> Wheat2_lme4 <- lmer(yield ~ latitude2 * longitude + (1 | Block),
+   data = Wheat2)
R> class(Wheat2_lme4)

[1] "mer"
attr("package")
[1] "lme4"

- No "groupeData"-objects can be used

R> summary(Wheat2_lme4)

Linear mixed model fit by REML
Formula: yield ~ latitude2 * longitude + (1 | Block)
Data: Wheat2
AIC  BIC logLik deviance REMLdev
1432 1453 -710.2 1410 1420
Random effects:
Groups   Name        Variance  Std.Dev.
Block    (Intercept) 11.916     3.4519
          Residual     30.524     5.5248
Number of obs: 224, groups: Block, 4

Fixed effects:
Figure 17: Perspective plot for the fitted values of block 1.

<table>
<thead>
<tr>
<th>Estimate</th>
<th>Std. Error</th>
<th>t value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>23.97464066</td>
<td>2.49785</td>
</tr>
<tr>
<td>latitude2</td>
<td>-0.82743079</td>
<td>0.23206</td>
</tr>
<tr>
<td>longitude</td>
<td>0.22562615</td>
<td>0.09740</td>
</tr>
<tr>
<td>latitude2:longitude</td>
<td>0.04422476</td>
<td>0.01384</td>
</tr>
</tbody>
</table>

Correlation of Fixed Effects:

<table>
<thead>
<tr>
<th>(Intr)</th>
<th>lattd2</th>
<th>longtd</th>
</tr>
</thead>
<tbody>
<tr>
<td>latitude2</td>
<td>-0.651</td>
<td></td>
</tr>
<tr>
<td>longitude</td>
<td>-0.661</td>
<td>0.831</td>
</tr>
<tr>
<td>lttd2:lngtd</td>
<td>0.534</td>
<td>-0.865</td>
</tr>
</tbody>
</table>

- To extract the estimated fixed effects use

\[
R> \text{fixef(Wheat2_lme4)}
\]

\[
\begin{array}{ccc}
\text{(Intercept)} & \text{latitude2} & \text{longitude} \\
23.97464066 & -0.82743079 & 0.22562615 \\
\end{array}
\begin{array}{c}
\text{latitude2:longitude} \\
0.04422476 \\
\end{array}
\]

- To extract the estimated random effects use

\[
R> \text{ranef(Wheat2_lme4)}
\]

88
To extract the covariance matrix of random effects use

\[ \text{R} > \text{VarCorr(Wheat2_lme4)} \]

\[
\begin{array}{r}
\$Block \\
\text{(Intercept)} \\
4 & -4.73226084 \\
2 & 2.61986444 \\
3 & -0.09551437 \\
1 & 2.20791077
\end{array}
\]

- To extract the covariance matrix of random effects use

\[ \text{R} > \text{VarCorr(Wheat2_lme4)} \]

\[
\begin{array}{r}
\$Block \\
\text{(Intercept)} \\
\text{(Intercept) } 11.91566 \\
\text{attr(,"stddev")} \\
\text{(Intercept) } 3.451907 \\
\text{attr(,"correlation")} \\
\text{(Intercept) } 1 \\
\text{attr(,"sc")} \\
\sigma_{\text{REML}} \\
5.524812
\end{array}
\]

**Growth of soybean plants**

**The Soybean dataset**

Data from an experiment to compare growth patterns of two genotypes of soybeans: Plant Introduction (P), an experimental strain, and Forrest (F), a commercial variety. The Soybean data frame has 412 rows and 5 columns. The data set is part of the library \textbf{nlme}.

- **Plot** a factor giving a unique identifier for each plot.
- **Variety** a factor indicating the variety; Forrest (F) or Plant Introduction (P).
- **Year** a factor indicating the year the plot was planted.
- **Time** a numeric vector giving the time the sample was taken (days after planting).
- **weight** a numeric vector giving the average leaf weight per plant (g).
Exercises

- Make some descriptive analyses for the Soybean data.
- We want to compare the growth of soybeans for the different Plots. What is your response variable, what are fixed and what are random effects? Set up a model and give an interpretation for the results.
- We want to fit two random effects for each plot: a random intercept and a random slope for the time. The model should also include the fixed effects Variety and Year. Fit this model using the lme4 package.
- Look at the covariance matrix of the random effects and give an interpretation.

10 Writing own functions

Repetitive execution

- Functions for repetitive execution: for loops, repeat and while
- There is also a for loop construction which has the form
  
  ```R
  for (name in expr_1) expr_2
  ```

  where name is the loop variable, expr_1 is a vector expression, (often a sequence like 1:20) and expr_2 is repeatedly evaluated as name ranges through the values in the vector result of expr_1.

  Example
  ```R
  x <- rep(NA, times = 6)
  for (i in 1:6) x[i] <- factorial(i)
  x
  ```

  ```R
  [1] 1 2 6 24 120 720
  ```

Writing own functions

- The R language allows the user to create objects of mode "function". These are true R functions that are stored in a special internal form and may be used in further expressions
- A function is defined by an assignment of the form
  ```R
  name <- function(arg_1, arg_2, ...) expression
  ```

  The expression is an R expression, (usually a grouped expression), that uses the arguments, arg_i, to calculate a value. The value of the expression is the value returned for the function
- A call to the function then usually takes the form
  ```R
  name(expr_1, expr_2, ...)
  ```