Introduction to R and RStudio

Eva Herzog, Carola Zenke-Philippi, and Matthias Frisch

1 The R prompt

After starting RStudio, the 'R prompt'[1] shows up in the 'console window'[2]:

```
>
```

At the R prompt, 'R expressions' can be typed in and after pressing the Enter-key, the R expression after the prompt is evaluated. The simplest type of R expressions are algebraic expressions. Algebraic expessions can be built by arithmetic operators, such as

`+`	plus	(addition)
`-`	minus	(subtraction)
`*`	times	(multiplication)
`/`	divided by	(division)
`^`	to the power of	(exponentiation)

Expressions can also include functions, such as

`sqrt()`	square root	(taking the square root)

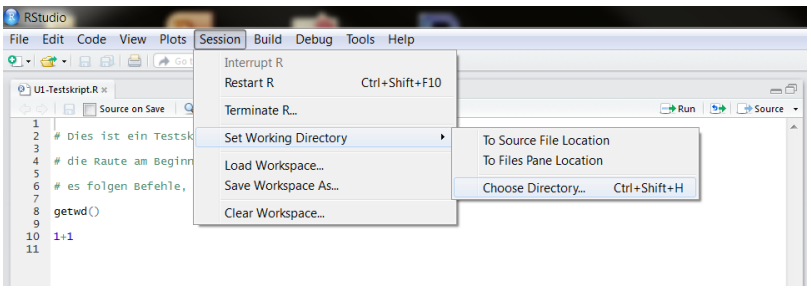
```
> 6+7        # plus (addition)
[1] 13
> 8-4        # minus (subtraction)
[1] 4
> 2*3        # times (multiplication)
[1] 6
> 10/5       # divided by (division)
[1] 2
> 2^3        # to the power of (exponentiation)
[1] 8
> sqrt(25)   # square root (taking the square root)
[1] 5
```

R is case sensitive, this means that `sqrt()` is different from `Sqrt()`, and trying to use `Sqrt()` will result in an error.

2 Setting the working directory

For reading and saving files, R uses its 'working directory'. R will search for data files in this folder and will save graphics files or R scripts there. In RStudio the working directory can be set via the menu system with **Session → Set Working Directory → Choose Directory**:

[1] R specifc terminology and technical terms are marked in single quotes when used for the first time
[2] Input and output in the console window is printed in blue boxes



Alternatively, the working directory can be set with the function `setwd()`. The directory name must be given in double quotes. R and RStudio work on different operating systems, which use different symbols to delimit folders and subfolders. We use always **one forward slash** to separate folders and subfolders **/**. This works for **Windows, MacOS and Linux**. For example:

```
> setwd("C:/Users/gh2347/Desktop")
```

The function `getwd()` can be used to check the current working directory:

```
> getwd()
[1] "C:/Users/gh2347/Desktop"
```

3 Using R scripts

After adjusting the pane layout, a window appears at the upper left corner that shows a file with the name `Untitled1`. This file is an R script that has not been named yet. We save it under the name `U1-Testskript.R`.

If RStudio starts without an empty R script in the left window, we generate an empty script with the menu system using **File** → **New File** → **R script**. R scripts must always be saved with the file extension **".R"**.

> *RStudio detects whether a file is an R script by checking the file extension. Only if the file extension* `.R` *is used, the system will work correctly.*

R scripts are text files that contain sequences of R commands. They can be used to collect the commands that are required to carry out a certain analysis for a certain data set. R scripts can be saved, adjusted, corrected, and re-used to modify analyses or carry out similar analyses for other data sets.

Commands are 'submitted' from the script window to the console window where they are evaluated. The results will also be displayed in the console. In order to submit a command, set the cursor in the corresponding line of the script window and press "Strg + Enter" for Windows and Linux ("Ctrl + Enter" for English keyboards) or "Command + Enter" on a Mac.

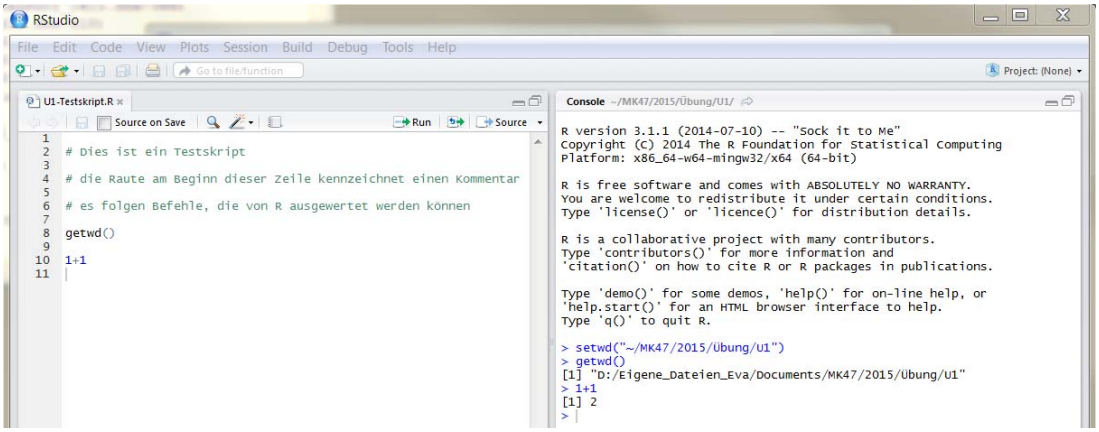
For example, we make an R script called as `U1-Testskript.R` and type in the following commands in the script window[3]

```
getwd()
1 + 1
```

[3] Input in the script window is printed in shaded boxes

then we get after 'submitting' the two lines the following result in the console window.

```
> getwd()
[1] "D:/Eigene_Dateien_Eva/Documents/MK47/2015/Uebung/U1"
> 1 + 1
[1] 2
```



Several lines can be submitted simultaneously by highlighting them in the R script and then sending them to the console.

4 Comments in R scripts

A comment is a text in an R script that does not contain R code and is not evaluated. Comments are used to annotate an R script and make it easier readable and understandable. Typical comments include headlines or explanations of the R code. Comments are marked by the # symbol.

Use spaces and line breaks to structure your code and comments. Separation lines can increase the clarity.

Here is an R script with comments:

```
#*****************************************************************************
# Example for using comments in an R script
#*****************************************************************************

#****************************************
# This will be evaluated:
1 + 1

#****************************************
# This is just a comment:
# 5 + 5

#****************************************
# Code and comments can be combined:
1 + 1       # add two numbers
```

that results in the following output:

```
> #*************************************************************************
> # Example for using comments in an R script
> #*************************************************************************
>
> #***************************************
> # This will be evaluated:
> 1 + 1
[1] 2
>
> #***************************************
> # This is just a comment:
> # 5 + 5
>
> #***************************************
> # Code and comments can be combined:
> 1 + 1       # add two numbers
[1] 2
```

5 Variables

Data that we want to use for calculations and analyses can be stored in variables. For example, we can create a variable with the name `blyton` and assign the value 5 to it by using the 'assignment operator' `<-`. The names of variables (on the left of the assignment operator) can be chosen arbitrarily.

```
blyton <- 5
```

If we submit this assignment (eg. "STRG + Enter"), we get the confirmation that everything worked well in the console window.

```
> blyton <- 5
```

If we submit the name of a variable, the variable is evaluated and its content is printed in the console window

```
blyton
```

```
> blyton
[1] 5
```

The names of variables can be chosen arbitrarily when they are defined. If you want to use the variable, however, you have to take care that you type its name correctly. Submitting

```
Blyton
```

results in

```
> Blyton
Error: object 'Blyton' not found
```

It is possible to do calculations with the data stored in variables. Submitting

```
11 - blyton
```

results in

```
> 11 - blyton
[1] 6
```

6 Data types

The variable `blyton` stores a number, numbers have the data type 'numeric'. The data type of the value that a variable stores can be checked with the function `class()`:

```
class(blyton)
```

```
> class(blyton)
[1] "numeric"
```

Another data type we often need is 'character'. Character variables contain text and are entered in quotes, you can use either `'` or `"`:

```
x <- "Joybrato Mukherjee"
x
class(x)
```

```
> x <- "Joybrato Mukherjee"
> x
[1] "Joybrato Mukherjee"
> class(x)
[1] "character"
```

Examples for other important data types:

```
character: "a", "b", "asdf", "jk;l", ...
numeric: 1, 1.23, 1.23e17, NaN, ...
integer: 1L, 1e3L (=1000L), ...
complex: 1+2i, ...
logical: TRUE,T,FALSE,F,NA
```

7 Data structures

7.1 Vectors

Data elements of the same type can be concatenated to so called vectors with the function `c()`:

```
chain <- c(1,2,3,4,5)
chain
```

```
> chain
[1] 1 2 3 4 5
```

In this example we abbreviated the output in the R console. In the grey input box we have the assignment (`chain <- c(1,2,3,4,5)`) and the evaluation of the variable chain (`chain`), but in the output window we show only the result of the evaluation. From now on we will do this often, which means that always all code in the grey boxes need to be sumitted, but the blue boxes contain only those parts of the output that are necessary to illustrate the use of a certain command.

You can do calculations with vectors. The calculations are usually done element-wise:

```
chain-1
chain-chain
chain^2
sqrt(chain)
```

```
> chain-1
[1] 0 1 2 3 4
> chain-chain
[1] 0 0 0 0 0
> chain^2
[1]  1  4  9 16 25
> sqrt(chain)
[1] 1.000000 1.414214 1.732051 2.000000 2.236068
```

Alphanumeric values can also be combined in vectors:

```
five.friends <- c("George", "Julian", "Richard", "Anne", "Tim")
five.friends
```

```
> five.friends
[1] "George"  "Julian"  "Richard" "Anne"    "Tim"
```

The variable `chain` is used to store a numerical vector and the variable `five.friends` stores an alphanumerical vector.

Single elements of vectors can be accessed with square brackets. Example: The third element of the vector `five.friends` is `Richard`.

```
five.friends[3]
```

```
> five.friends[3]
[1] "Richard"
```

In addition to vectors, there are many other data structures available in R. We will mostly work with vectors, matrices and dataframes.

7.2 Matrices

In vectors, all data are in one row. In matrices, data are arranged in rows and columns.

```
abc <- matrix(c("a", "b", "c", "d",
                "e", "f", "g", "h"), # data
              nrow = 2,              # number of rows
              byrow = TRUE)          # data are ordered row-wise
abc
```

```
> abc
     [,1] [,2] [,3] [,4]
[1,] "a"  "b"  "c"  "d"
[2,] "e"  "f"  "g"  "h"
```

The function `matrix` has two arguments: `nrow` and `byrow`. Arguments allow you to specify what exactly the function is supposed to do. In our case, we want to create a matrix with exactly two rows and the letters are supposed to be written into the matrix row-wise.

For comparison:

```
abc2 <- matrix(c("a", "b", "c", "d",
                 "e", "f", "g", "h"),
               nrow = 4, byrow = FALSE)
abc2
```

```
> abc2
     [,1] [,2]
[1,] "a"  "e"
[2,] "b"  "f"
[3,] "c"  "g"
[4,] "d"  "h"
```

Just like elements of vectors, elements of matrices can be accessed with square brackets. The difference is that it is necessary to indicate the row and column numbers in the format [row, column]. Example: The letter “g” is in the 2nd row and in the 3rd column of the matrix `abc`:

```
abc[2,3]
```

```
> abc[2,3]
[1] "g"
```

It is also possible to access whole rows or columns:

```
abc[2,]
abc[,3]
```

```
> abc[2,]
[1] "e" "f" "g" "h"
> abc[,3]
[1] "c" "g"
```

Note: Vectors as well as matrices can contain EITHER numbers (= `numeric`) OR letters (= `character`). If you mix, R converts the numbers into `character` elements:

```
abc.123 <- matrix(c("a", "b", "c",
                    1 ,  2 ,  3),
                  nrow = 2,  byrow = T)
abc.123
```

```
> abc.123
     [,1] [,2] [,3]
[1,] "a"  "b"  "c"
[2,] "1"  "2"  "3"
```

7.3 Data frames

A data structure that is often used in statistical analyses is the data frame. Data frames are tables. Typically the rows of a data frame contain information for different observation units and the columns of the data frame contain (a) information that describes the observation unit, or (b) measurements of the observation unit.

The function `data.frame()` combines vectors to a data frame. The vectors can have different types but must have the same length.

```
fieldtrial <- data.frame(
  plot       = c( 1,  2,  3,   4,   5,   6,  7,   8,  9,  10,   11,  12) ,
  variety    = c( 1,  1,  1,   1,   1,   1,  2,   2,  2,   2,    2,   2) ,
  fertilizer = c("N","N","N","NK","NK","NK","N", "N","N","NK", "NK","NK"),
  rep        = c( 1,  2,  3,   1,   2,   3,  1,   2,  3,   1,    2,   3) ,
  yield      = c(80, 90, 95,  95, 100,  93, 88, 102, 92,  100, 110, 103)    )

fieldtrial
```

The columns of a data frame are variables, their variable names (`plot`, `variety`, `fertilizer`, `rep` and `yield`), can be chosen arbitrarily when constructing the data frame. The variable names are the headers of the columns.

```
> fieldtrial
   plot variety fertilizer rep yield
1     1       1          N   1    80
2     2       1          N   2    90
3     3       1          N   3    95
4     4       1         NK   1    95
5     5       1         NK   2   100
6     6       1         NK   3    93
7     7       2          N   1    88
8     8       2          N   2   102
9     9       2          N   3    92
10   10       2         NK   1   100
11   11       2         NK   2   110
12   12       2         NK   3   103
```

In this data frame, the first row contains information on the first observation unit, the 12th row, information on the 12th observation unit. Each observation unit is described by the information in the first four columns. The first observation unit was assessed in plot 1, the variety was 1, the

fertilizer was N, and it was the first out of three replications of fertilizer N. The column yield contains the measurement for yield that was assessed at the observation units.

A useful function to get an overview over data frames is `str()`:

```
str(fieldtrial)
```

```
> str(fieldtrial)
'data.frame':        12 obs. of  5 variables:
 $ plot      : num  1 2 3 4 5 6 7 8 9 10 ...
 $ variety   : num  1 1 1 1 1 1 2 2 2 2 ...
 $ fertilizer: chr w/ 2 levels "N","NK": 1 1 1 2 2 2 1 1 1 2 ...
 $ rep       : num  1 2 3 1 2 3 1 2 3 1 ...
 $ yield     : num  80 90 95 95 100 93 88 102 92 100 ...
```

At first sight, we see that the dataframe `fieldtrial` contains 5 columns with 12 rows (= observations). The variables are called `plot`, `variety`, `fertilizer`, `repetition`, and `yield`. The columns `plot`, `variety`, `repetition`, and `yield` contain numbers and therefore have the type `numeric`. The column `fertilizer` contains **character strings**. Compared to older R versions, those character strings are NOT automatically converted to the type `Factor` in R 4.0 (which is required for doing analysis of variance analyses correctly).

It is possible to access the variables in data frames in different ways. The following three commands all lead to the same result, the output of the values of the column `yield` as a vector.

```
fieldtrial$yield
fieldtrial[,"yield"]
fieldtrial[,5]
```

```
> fieldtrial$yield
 [1]  80  90  95  95 100  93  88 102  92 100 110 103

> fieldtrial[,"yield"]
 [1]  80  90  95  95 100  93  88 102  92 100 110 103

> fieldtrial[,5]
 [1]  80  90  95  95 100  93  88 102  92 100 110 103
```

The last line corresponds to the selection of a column of a matrix. Similarly, it is possible to select single lines of a dataframe:

```
fieldtrial[3,]
```

```
> fieldtrial[3,]
  plot variety fertilizer rep yield
3    3       1          N   3    95
```

7.4 Objects

In general software terminology an 'object' is a a combination of program code and data. There is a rule that says that "In R everything is an object", and as consequence different sorts of

objects in R exist. Some sorts of objects actually contain code and data, and for other sorts of objects either the code part or the data part is missing.

- Variables or data structures are objects that consist only of data
- Functions are objects that consist only of program code
- S3 objects and S4 objects are often the result of carrying out analyses. You carry out an analysis with a data set and get back an object with the results. The object contains the results of the analysis and in addition program code that can plot or print the results of the analysis

8 Saving and loading data files

When you created large dataframes from long calculations, it is useful to save the results as text files so that you do not have to re-run the analysis but can just read in the data later. If you do not specify a path, the data are written into the working directory.

```
write.table(fieldtrial, "fieldtrial.txt") # variable name, "filename"
```

The files can be read in with the function `read.table()` and the results should be assigned to a variable.

```
ft <- read.table("fieldtrial.txt")
ft
```

```
> ft
   plot variety fertilizer rep yield
1     1       1          N   1    80
2     2       1          N   2    90
3     3       1          N   3    95
4     4       1         NK   1    95
5     5       1         NK   2   100
6     6       1         NK   3    93
7     7       2          N   1    88
8     8       2          N   2   102
9     9       2          N   3    92
10   10       2         NK   1   100
11   11       2         NK   2   110
12   12       2         NK   3   103
```

(If you do not assign the result of `read.table()` to a variable, it will be just printed the console, which is rather useless).

The function `read.table()` has several arguments for adjustment of the data import. The help function gives you an overview over the available arguments of a function. This always works by entering `?functionname`.

```
?read.table
```

In our course, we only need the arguments `file`, `header`, `sep`, and `dec`.

With `file = filename.extension` you specify which file is supposed to be read from the

working directory. Do not forget the quotation marks! If the data columns have headers, add `header = T`, otherwise `header = F`. With `dec` you can specify whether decimal numbers in the file are encoded with a dot or a comma.

The argument `sep` is especially important. It tells R which symbol is used as a separator for the columns. If you do not know which symbol is used in the data file, open the file in RStudio like you would open an R script (File → Open File) and check. `.txt` files often have a tab stop (`sep="\t"`) or a blank (`sep=" "`), `.csv` files often have a comma (`sep=","`) or a semicolon (`sep=";"`).

9 Reading in data from Excel or OpenOffice files

When you are recording data for your thesis, you will often collect it in Excel or OpenOffice spreadsheets. You can use `read.table()` to read these files into R but first you have to save the xlsx or ods file as a text file, e.g. as csv file.

If you are using **OpenOffice** or **LibreOffice** you may have to change some settings in order to get a file that R can read correctly.

Open the example file `fieldtrial2.xlsx` with Office and have a look at the data. The columns have headers and (in the German version) the decimal separator is a comma. The separation symbol for the columns is invisible. Save the file as `fieldtrial2.csv`. You can now open it in RStudio.

Open, have a look, close. Do not write!

The data have columns headers and the columns are separated with semicolons (";"). The comma is the decimal separator.

```
Parzelle;Sorte;Duenger;Wdh;Ertrag
1;A;N;1;92,7
2;A;N;2;89,3
3;A;N;3;95,2
4;A;NK;1;
5;A;NK;2;105,9
6;A;NK;3;111,4
7;B;N;1;98,8
8;B;N;2;102
9;B;N;3;
10;B;NK;1;96,1
11;B;NK;2;100,5
12;B;NK;3;106,6
```

You now know how to adjust the settings in `read.table()`:

```
ft2 <- read.table(file = "fieldtrial2.csv",  # name of input file
                  dec = ",",                 # decimal separator
                  sep = ";",                 # separator between values
                  header = T)                # data have headers
```

```
> ft2
   Parzelle Sorte Duenger Wdh Ertrag
1         1     A       N   1   92.7
2         2     A       N   2   89.3
3         3     A       N   3   95.2
4         4     A      NK   1     NA
5         5     A      NK   2  105.9
6         6     A      NK   3  111.4
7         7     B       N   1   98.8
8         8     B       N   2  102.0
9         9     B       N   3     NA
10       10     B      NK   1   96.1
11       11     B      NK   2  100.5
12       12     B      NK   3  106.6
```

You can see that two values are missing in the column `yield`. Missing values in R are encoded as `NA` ("not available").

You can find more information on reading data files from OpenOffice in the Apendix.

10 Cleaning up

The following command clears the workspace. All object in the Global Environment are deleted. We recommend doing this after every tutorial in order to save memory and avoid mistakes caused by duplicate variable names.

```
rm(list = ls())
```

Single factor analysis of variance

Matthias Frisch

1 Doughnuts example

R provides lots of different functions for data analysis. Sometimes, however, we need other functions than those available in base R. So we load additional packages that contain the functions we need. The packages have to be loaded each time you restart R.

```
library("dplyr")    # Provides: summarise
```

Data input by hand, method 1 (for balanced data only):

```
amount <- c ( 64, 78, 75, 55,  72, 91, 93, 66,  68, 97, 78, 49,
              77, 82, 71, 64,  56, 85, 63, 70,  95, 77, 76, 68 )

fattype <- gl( n=4, k=1, length=4*6, labels=c("f1","f2","f3","f4") )
```

```
> amount
 [1] 64 78 75 55 72 91 93 66 68 97 78 49 77 82 71 64 56 85 63 70 95 77 76 68
> fattype
 [1] f1 f2 f3 f4 f1 f2 f3 f4 f1 f2 f3 f4 f1 f2 f3 f4 f1 f2 f3 f4 f1 f2 f3 f4
Levels: f1 f2 f3 f4
```

"Balanced" means that we have the same number of replications for each of the factor levels.

Data input by hand, method 2 (can also be used for unbalanced data):

```
fattype2 <- factor( c( rep("f1", times=6), rep("f2", times=6),   # Works also for
                       rep("f3", times=6), rep("f4", times=6)) ) # unbalanced data
```

```
> fattype2
 [1] f1 f1 f1 f1 f1 f1 f2 f2 f2 f2 f2 f2 f3 f3 f3 f3 f3 f3 f4 f4 f4 f4 f4 f4
Levels: f1 f2 f3 f4
```

The order in `fattype2` different from that in `fattype` so we need to change the vector with the amount of fat that is absorbed accordingly:

```
amount2 <- c(64, 72, 68, 77, 56, 95,  # fat type 1
             78, 91, 97, 82, 85, 77,  # fat type 2
             75, 93, 78, 71, 63, 76,  # fat type 3
             55, 66, 49, 64, 70, 68)  # fat type 4

donuts2 <- data.frame(fat = fattype2,      # combine to a data frame
                      absorbed = amount2)
```

However, we stick with the first version and combine both vectors to a data frame:

```
donuts <- data.frame(fat = fattype,      # combine to a data frame
                     absorbed = amount)
```

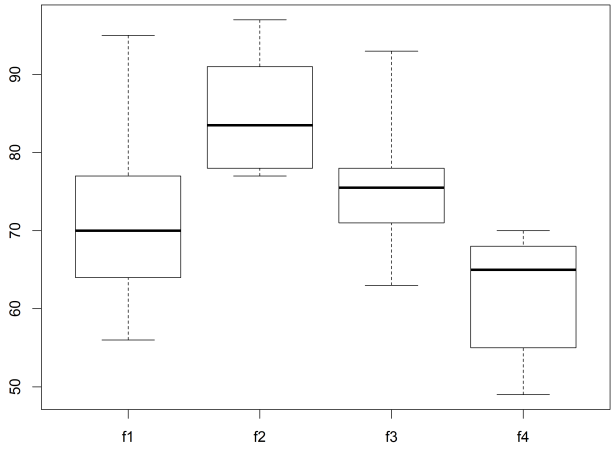
```
> str(donuts)
'data.frame':        24 obs. of  2 variables:
 $ fat     : Factor w/ 4 levels "f1","f2","f3",..: 1 2 3 4 1 2 3 4 1 2 ...
 $ absorbed: num  64 78 75 55 72 91 93 66 68 97 ...
```

`str()` gives us a good overview over the structure of a data frame. This is especially useful if we deal with big data frames that we cannot have a look at in the console. The data frame has two columns with the headers `fat` and `absorbed`. We call each column a variable. The variable `fat` has the data type "factor", the variable `absorbed` has the data type "numeric". Each column contains 24 observations.

Check in the raw data table if the right measurements are assigned to the fat types:

```
> donuts
   fat absorbed
1   f1       64
2   f2       78
3   f3       75
4   f4       55
5   f1       72
6   f2       91
7   f3       93
8   f4       66
9   f1       68
10  f2       97
11  f3       78
12  f4       49
13  f1       77
14  f2       82
15  f3       71
16  f4       64
17  f1       56
18  f2       85
19  f3       63
20  f4       70
21  f1       95
22  f2       77
23  f3       76
24  f4       68
```

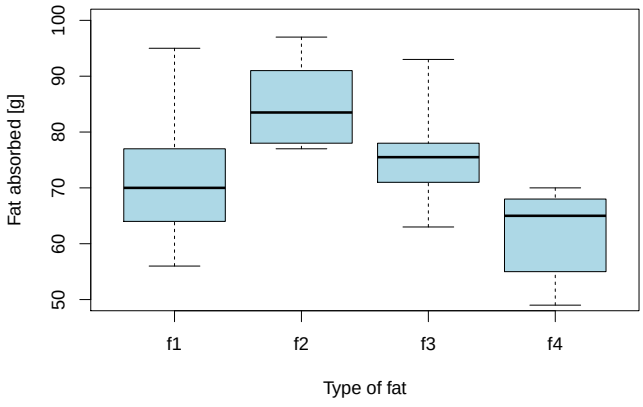
```
boxplot ( absorbed ~ fat,            # Box and Whisker plot
          range = 0,                 # Min, max, quartiles, median
          data=donuts)               # Name of the data set
```



`boxplot()` draws a box and whisker plot of the measured values in `absorbed`. The data are grouped according to the type of fat stated in `fat`. The grouping is caused be the use of the tilde `~`. `data=donuts` tells R where the data are. `range=0` causes the whiskers to be extended to the minimum and maximum values.

We can also make alterations to the color and add labels to the x and y axes.

```
par ( mar=c(4,4,1,1) )                # No. of lines around the plot for
                                      # axis descriptions: bltr
boxplot (absorbed~fat,                # Model
         ylab="Fat absorbed [g]",     # Label for y axis
         xlab="Type of fat",          # Label for x axis
         col ="lightblue",            # Color of boxes
         range=0,                     # Whisker to min and max
         ylim=c(50,100),              # Range of y axis
         data=donuts)
```



Graphics are usually created for use in presentations and publications. We therefore want to export them to a graphics file.

```
png ( file="donuts-boxplot.png",     # Redirect graphics to a png file
      units="in", height=6, width=8, # Size of the picture
      res=360,pointsize=13 )         # Pixel resolution, and font size

par ( mar=c(4,4,1,1) )               # No. of lines around the plot for
                                     # axis descriptions: bltr
boxplot (absorbed~fat,               # Model
         ylab="Fat absorbed [g]",    # Label for y axis
         xlab="Type of fat",         # Label for x axis
         col ="lightblue",           # Color of boxes
         range=0,                    # Whisker to min and max
         ylim=c(50,100),             # Range of y axis
         data=donuts)

dev.off()                            # Important: Close file!!
```

Do not forget `dev.off()` at the end! It closes your graphics file. If you opened several graphics devices, have lost track and nothing seems to work anymore, you can shut them all down with `graphics.off()`.

`dev.off()` will also re-set the options set with `par()`.

Descriptive statistics

```
donuts %>% group_by(fat) %>%
    summarise ( n=n(), mean=mean(absorbed), var=var(absorbed),
                min=min(absorbed), max=max(absorbed))
```

```
  fat       n  mean   var   min   max
  <fct> <int> <dbl> <dbl> <dbl> <dbl>
1 f1        6    72 178      56    95
2 f2        6    85  60.4    77    97
3 f3        6    76  97.6    63    93
4 f4        6    62  67.6    49    70
```

```
attach (donuts)                              # Use this data set
```

Since we do not always want to tell R that the data are stored in the data frame `donuts`, we attach the data frame to the search path. If used properly, attaching a data frame can save you a lot of typing. Do not forget to detach after you are finished!

Bartlett test

```
bartlett.test ( absorbed ~ fat )
```

```
> bartlett.test ( absorbed ~fat )
        Bartlett test of homogeneity of variances
data:  absorbed by fat
Bartlett's K-squared = 1.7504, df = 3, p-value = 0.6258
```

The Bartlett test is one example for a statistical test. The analysis of variance which will follow shortly is another one. The concept of these tests is always the same:

(a) Start with a question you want to answer. We need homogeneous variances of the factor levels for the analysis of variances so we ask: Are the variances of the factor levels different? Note that the question is usually about *differences*. For the doughnuts it would be: Does the amount of fat absorbed vary differently between the fat types?

(b) Choose a statistical test that is designed to answer this question. The Barlett test is a good choice in our case.

(c) Spell out the null and the alternative hypotheses of the test. They depend on the test you selected. Generally, the null hypothesis of any test is "no differences", the alternative hypothesis is "significant differences" or, in the case of one-sided tests, "significantly greater/smaller", between whatever the test is for.

The null hypothesis H_0 of the Barlett test is that the variances of the factor levels are not significantly different. The alternative hypothesis H_1 or H_A is that the variances of the factor levels *are* significantly different. Applied to our question:

Null hypothesis H_0: The variances of the fat uptake are the same for all four types of fat. $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \sigma_4^2$.

Alternative hypothesis H_1 or H_A: At least one of the variances is significantly different from the others. (We do not now which one.)

(d) Apply the test to your data. Note that in the data of your sample(s), there will almost always be differences between your parameters of interest. It is not realistic to expect that the observed variances of the factor levels are exactly the same. The output from the section on descriptive statistics shows exactly that when you look at the column `var`:

```
  fat       n  mean   var   min   max
  <fct> <int> <dbl> <dbl> <dbl> <dbl>
1 f1        6    72 178      56    95
2 f2        6    85  60.4    77    97
3 f3        6    76  97.6    63    93
4 f4        6    62  67.6    49    70
```

However, statistical tests are not about the sample values but about the `true` values which we do not know. The sample variance for fat type 1, $s^2_{\text{fat type 1}}$, is calculated from the sample values for fat type 1 and estimates the true variance of all doughnuts baked in fat type 1, $\sigma^2_{\text{fat type 1}}$. "Estimated" does not mean that it is just a rough guess. We use the term "estimation" when we have the situation that we do not know the true value but need, well, an estimate for it.

Latin letters, here s, are used for the values of the samples and Greek letters, here σ, are used for the true values. Alternatively, the sample values can also be encoded with the $\hat{}$ symbol, as explained in the section in the estimation of effects.

(e) Calculate the test statistic and the p-value. They both show up in the output of the Bartlett test: K^2 is 1.7504381 and p is 0.6258.

(f) Make your test decision.

We make our test decision based on the following rules: If $p < 0.05$, we will reject H_0 and assume that H_1 is true, i.e. that at least one variance is significantly different from

the others.

The value of 0.05 is somewhat arbitrary. It is called the significance level $\alpha_{\max}$ and can be chosen differently (but in our course we always use 0.05 if not stated otherwise). So we will keep the null hypothesis H_0 if $p >= \alpha$ and will reject H_0 and assume that H_1 is true if $p < \alpha$.

For our test, we decide to accept H_0 - very fortunate, actually, since homogeneity of variances is needed in order to conduct the ANOVA.

Analysis of variance

```
summary ( aov(absorbed ~fat) )
```

```
> summary  ( aov(absorbed ~fat) )
            Df Sum Sq Mean Sq F value  Pr(>F)
fat          3   1636   545.5   5.406 0.00688 **
Residuals   20   2018   100.9
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

H_0: $\mu_1 = \mu_2 = \mu_3 = \mu_4$ The means of the fat types are not significantly different, the fat type does not have a significant influence on the amount of fat absorbed. All sample means estimate the true mean μ.

H_1: At least one mean is significantly different from the others. The fat type has a significant influence on the amount of fat absorbed.

If there were no true differences, then the probability that the observed or more extreme data occured is only 0.7%. Therefore we conclude that there are differences between the means, so the type of fat has a significant influence on the amount of absorbed fat ($p = 0.00688$). Or, in a more formal way: $p = 0.00688, \alpha = 0.05, p < \alpha \rightarrow$ reject H_0. At least on type of fat is significantly different from the others regarding the fat uptake.

Note that the F test statistic from the ANOVA table is actually a ratio of variances.

Estimation of effects

It would now be interesting to know which treatment has which effect - that is, to express the effect in numbers.

In the analysis of variance we employ a statistical model which assumes that each observation y_{ij} is composed of the true mean μ, an effect ϑ_i of the factor level i and a random error ϵ_{ij} that is specific to each observation: $y_{ij} = \mu + \vartheta_i + \epsilon_{ij}$

The index i indicates the factor level (*i.e.* type of fat) and the index j indicates the replication within the factor level. This means that the first observation of fat type 4 consists of the following components:

$y_{41} = \mu + \vartheta_4 + \epsilon_{41}$

The true mean μ can be estimated from all data points since the design is balanced:

```
> mean(absorbed)
[1] 73.75
```

Hence $\hat{\mu} = 73.75$.

```
model.tables ( aov(absorbed ~fat) )
```

```
> model.tables (aov (absorbed ~fat) )
fat
     1      2      3      4
 -1.75  11.25   2.25 -11.75
```

The effect on the fat uptake for fat type 1 is $\hat{\vartheta}_1 = -1.75$. This means that the uptake for this fat is estimated to be 1.75 g lower than the average fat uptake of all four fats. The effect on the fat uptake for fat type 2 is $\hat{\vartheta}_1 = 11.25$. This means that donuts fried in this fat type take up 11.25 g more fat than the average fat uptake of all four fats.

The means of the factor levels we estimated in the section on descriptive statistics are a combination of the true mean μ and the effect for the factor level in question:

$\hat{\mu}_i = \hat{\mu} + \hat{\vartheta}_i$

E.g. $\hat{\mu}_4 = \hat{\mu} + \hat{\vartheta}_4 = 73.75 + (-11.75) = 62$

We call these the fitted values. If we only know in which type of fat a batch of donuts is baked, they are our prediction of the amount of fat that will be absorbed. The residuals which will be evaluated in the following sections are the deviations of the individual observations from the predictions, *e.g.* the residual for the first observation of fat type 4 is

$\epsilon_{41} = y_{41} - (\hat{\mu} + \hat{\vartheta}_4) = 55 - (73.75 - 11.75) = 55 - 62 = -7$

because

$y_{41} = \hat{\mu} + \hat{\vartheta}_4 + \epsilon_{41}$

$55 = 73.75 - 11.75 - 7$

More general: $\epsilon_{ij} = y_{ij} - (\hat{\mu} + \hat{\vartheta}_i) = y_{ij} - \hat{\mu}_i$

$y_{ij} = \mu + \vartheta_i + \epsilon_{ij}$ is the standard parametrization for the ANOVA with one fixed effect. The cell means model is:

$y_{41} = \hat{\mu}_4 + \epsilon_{41}$,

in this case:

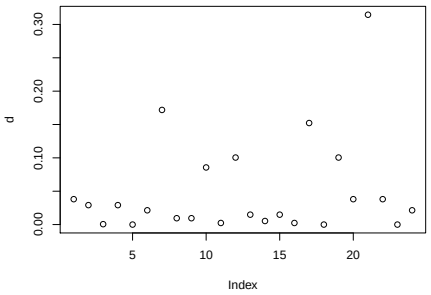
$55 = 62 - 7$

We use Greek letters μ oe ϑ for the values of a model and the $\hat{}$ symbol, $\hat{\mu}$ or $\hat{\theta}$ for the actual numbers in a certain experiment.

Graphical analysis of outliers and residuals

Outliers in the data can influence the model significantly. Cook's distance is a measure to help us judge if an extreme point is indeed an outlier. A value of 1 is often used as a threshold.

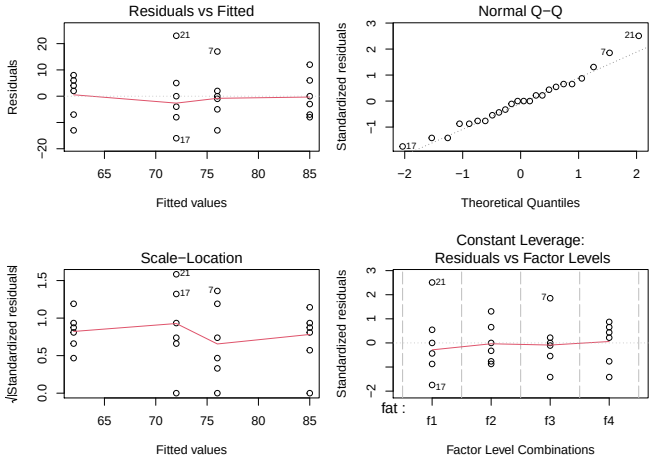
```
plot( cooks.distance( aov(absorbed ~fat) ) ) # Greater than 1?
```



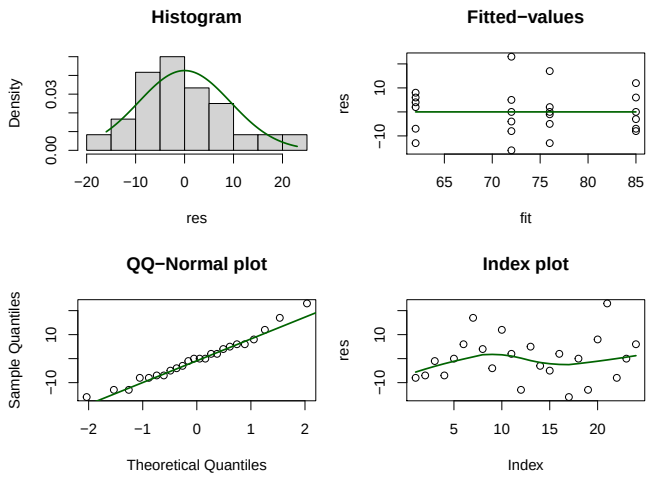
If we find a point with a Cook's distance > 1, we consider it an outlier.

Diagnostic plots can be used to check if the prerequisites of the ANOVA are fulfilled.

```
par( mfrow=c(2,2) )                    # Four displays
plot( aov(absorbed ~fat) )             # Diagnostic plots
```



```
source("v14-u11-00.R")                 # Alternative
check.residuals ( aov(absorbed ~fat) )  # diagnostic plots
```



The histogram and the QQ-Normal plot help us judge if the residuals come from a normal distribution which is another prerequisite for the ANOVA. The plot of the fitted values shows us if the variances are homogeneous. The index plots helps us spot systematic errors in the data collection.

"res" are the residuals, "fit" are the fitted values.

Histogram: The green line is the normal distribution curve for a normal distribution with mean and variance estimated from the data. The histogram should have approximately the same shape.

Fitted values: You can see on the y axis that we have one fitted value for each of the fat types. The first one belongs to fat type 4, the second one to fat type 1, the third one to fat type 3 and the fourth one th fat type 2 (have a look at the table with the means of the factor levels). Each of the 24 points represents one residual and the residuals are grouped to correspond to their fat types. The point on the low left, for example, is the residual for the third replication of fat type 4: $\epsilon_{43} = y_{43} - (\hat{\mu} + \hat{\vartheta}_4) = 49 - (73.75 - 11.75) = 49 - 62 = -13$. The residuals should be randomly distributed around 0 (green line). We do not want to see the shape of a funnel since this would mean that the variances increase if the fitted values in increase and the prerequisite for the ANOVA (homogeneous variances) is violated.

QQ-Normal plot: The x axis represents the quantiles if the residuals came from a normal distribution. The y axis represents the actual quantiles of the sample. If they both are the same, then all the points are on the green line and we assume that the residuals come from a normal distribution. In pratice, they will never be exactly on the green line but they should at least be randomly distributed around it.

Index plot: Any systematic patterns in the index plot which would indicate irregularities in the data collection, *e.g.* different values from day to day or imprecise measurements at the beginning. Ideally, the green trend line should be straight.

The diagnostic plots show no deviations from the assumptions: The data look roughly normally distributed (histogram) as well as the residuals (normal Q-Q plot). The variances seem to be equal (Fitted values) and there are no systematic patterns in the index plot. Our analysis was therefore valid.

Do not forget to detach the data frame again!

```
detach(donuts)                                  # Important!
```

2 Exercises

Analyse the following data sets. (a) What is the response variable (dependent variable)? What is the factor variable (independent variable)? What are the factor levels? How many replications are there per facotr level? What is the sample size N? (b) What is the research question? (c) Is the design balanced or unbalanced? (d) Visualize the data. Describe what you see in the plots. (e) Estimate the means and variances for the factor levels. (f) Carry out a Bartlett test. Do the factor levels have equal variances? (g) Carry out an analysis of variance. What are the null and the alternative hypotheses? What is your conclusion from the analysis? (h) Estimate the effects of the factor levels. What do the numbers mean? (i) Check for outliers.

1 Fertilizer

A field was divided into 12 plots. Each of three different fertilizers were applied to four plots. The yield was measured in [dt/ha].

	Fertilizer		
Replication	Mineral N	Straw	Cover crop
1	18	14	16
2	19	12	14
3	16	13	15
4	17	15	15

Additional questions:

(j) What is the model equation for observation y_{32} in the standard parametrization? What are the names of the variables in the model? (k) What is the model equation for observation y_{32} in the cell means model? What are the names of the variables in the model? Assume that $i = 1$ for mineral N, $i = 2$ for straw and $i = 3$ for cover crop.

2 Pigs

Nineteen pigs were assigned at random to four different groups. Each group was fed with a different diet. After the pigs were grown up they were weighted. Weights in kg (Zar 2009, p. 191).

Diet			
1	2	3	4
60.8	68.7	102.6	87.9
57.0	67.7	102.1	84.2
65.0	74.0	100.2	83.1
58.6	66.3	96.5	85.7
61.7	69.8		90.3

Additional question:

(j) Analyse the residuals. Were the prerequisites of the ANOVA met?

3 Carrots

The yield of four new varieties of carrots was measured [dt/ha].

	Replication				
Variety	**1**	**2**	**3**	**4**	**5**
Mokum	**28.3**	**33.4**	**29.5**	**28.9**	**33.2**
Julia	**24.1**	**24.6**	**25.7**	**23.5**	**26.0**
Neptun	**29.6**	**30.5**	**31.2**	**35.4**	**34.5**
Bolero	**27.3**	**28.1**	**28.6**	**27.0**	**26.8**

Additional questions:

(j) What is the model equation for observation y_{41} in the standard parametrization? What are the names of the variables in the model? (k) What is the model equation for observation y_{41} in the cell means model? What are the names of the variables in the model? Assume that $i = 1$ for Mokum, $i = 2$ for Julia, $i = 3$ for Neptun, and $i = 4$ for Bolero.

4 Blood

Lab animals obtained a diet with four different feeding additives. The time for blood coagulation [sec] was measured.

Additive	**Coagulation time [sec]**							
A	**63**	**67**	**71**	**64**	**65**	**66**		
B	**62**	**60**	**63**	**59**				
C	**56**	**62**	**60**	**61**	**63**	**64**	**63**	**59**
D	**68**	**66**	**71**	**67**	**68**	**68**		

Additional question:

(j) What is the predicted coagulation time for feeding additive B ($i = 2$)? (k) What are the residuals ϵ_{21} and ϵ_{41}?

3 Solutions

1 Fertilizer

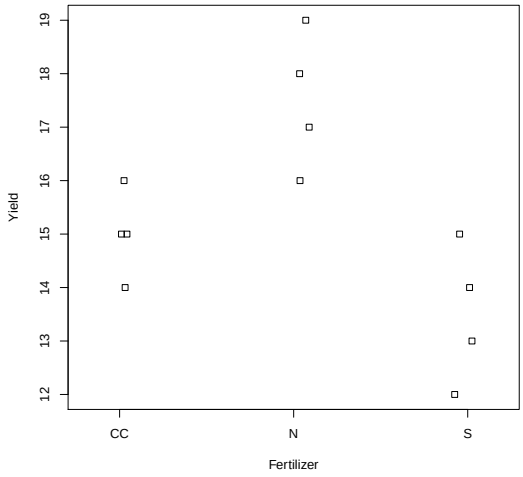
(a) Response variable: yield. Factor variable: fertilizer, 3 factor levels (mineral N, straw, cover crop) with 4 replications each. $N = 12$.

(b) Does the kind of fertilizer have an influence on the yield?

(c) The design is balanced.

(d) There are only four replications per factor level. A boxplot, however, is constructed from five values (min, max, median, Q_{25}, Q_{75}). Since it is not feasible to estimate five values from four observations, a stripchart is a more appropriate visualization here. It shows the single observations.

```
stripchart(yield ~ fert,
           method="jitter",  # make overlapping points visible
           vertical = T)     # rotate the plot
```



The jittering is done randomly so the plots look a bit different each time you create them.

(e)

```
  fert      n  mean   var
  <fct> <int> <dbl> <dbl>
1 CC        4  15   0.667
2 N         4  17.5 1.67
3 S         4  13.5 1.67
```

(f)

```
        Bartlett test of homogeneity of variances

data:  yield by fert
Bartlett's K-squared = 0.64502, df = 2, p-value = 0.724
```

The factor levels have equal variance ($p = 0.7243$).

(g) Null hypothesis: There is no significant difference between the mean yields for different

fertilizers. The fertilizer has no significant influence on the yield. Alternativ hypothesis: At least one of the fertilizers results in a significantly different yield. The fertilizer has a significant influence on the yield.

```
            Df Sum Sq Mean Sq F value Pr(>F)
fert         2  32.67  16.333   12.25 0.0027 **
Residuals    9  12.00   1.333
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusion: The fertilizer has a significant influence on the yield ($p = 0.0027$).

(h)
```
Tables of effects

 fert
fert
     CC       N       S
-0.3333  2.1667 -1.8333
```

The yield of the crop fertilized with mineral N was 2.167 dt/ha higher than average, the yields of the crops fertilized with cover crop and straw were 0.3333 and 1.8333 dt/ha below average, respectively.

(i) There are no observations with a Cook's distance greater than 1. Therefore, we assume that there are no outliers.

(j) $y_{32} = \mu + \vartheta_3 + \epsilon_{32}$
$14 = 15.33333 - 0.33333 - 1$
μ is the true mean, ϑ_3 is the effect of fertilization with cover crop and ϵ_{32} is the residual for observation y_{32}, *i.e.* the deviation of observation y_{32} from the prediction. $j = 2$ means that it is the second replication. (Note that we use the estimators for μ and ϑ_3, that is $\hat{\mu}$ and $\hat{\vartheta}_3$, as a number in our equation.)

(k) $y_{32} = \mu_3 + \epsilon_{32}$
$14 = 15 - 1$
μ_3 is the mean of the third factor level (cover crop) and ϵ_{32} is the residual for observation y_{32}, *i.e.* the deviation of observation y_{32} from the prediction.

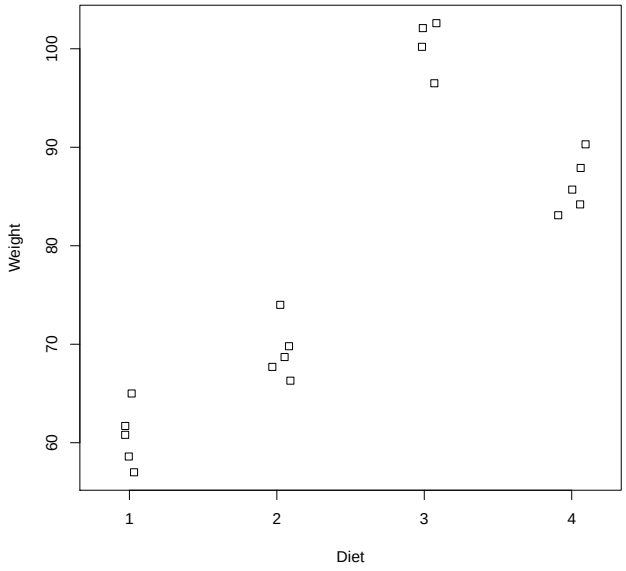
2 Pigs

(a) Response variable: weight. Factor variable: diet, 4 factor levels (diets 1 - 4) with 5, 5, 4, and 5 replications, respectively. $N = 19$.

(b) Does the diet have an influence on the weight gain in pigs?

(c) The design is unbalanced.

(d) A stripchart is appropriate due to the limited number of replications per factor level.



(e)
```
  diet      n  mean   var
  <fct> <int> <dbl> <dbl>
1 1         5  60.6  9.39
2 2         5  69.3  8.57
3 3         4 100.   7.66
4 4         5  86.2  8.39
```

(f)
```
        Bartlett test of homogeneity of variances

data:  weight by diet
Bartlett's K-squared = 0.032842, df = 3, p-value = 0.9984
```

The factor levels have equal variance ($p = 0.9984$).

(g) Null hypothesis: There is no significant difference between the weight gain in pigs on different diets. The diet has no significant influence on the weight gain. Alternativ hypothesis: At least one of the diets results in a significantly different weight. The diet has a significant influence on the weight gain.

```
            Df Sum Sq Mean Sq F value   Pr(>F)
diet         3   4226  1408.8   164.6 1.06e-11 ***
Residuals   15    128     8.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

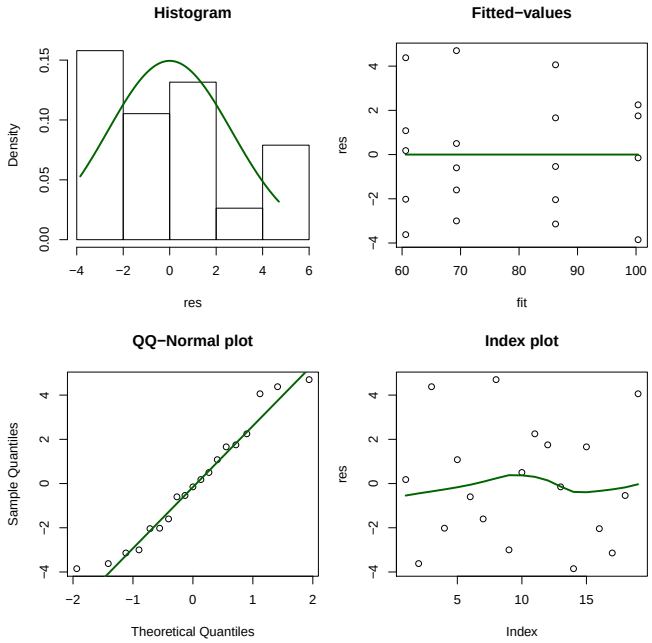
Conclusion: The diet has a significant influence on the weight gain ($p = 1.06 \times 10^{-11}$).

(h)
```
Tables of effects

 diet
        1      2     3     4
   -17.39 -8.711 22.34 8.229
rep   5.00  5.000  4.00 5.000
```

(i) There are no observations with a Cook's distance greater than 1. Therefore, we assume

that there are no outliers.

(j) **Normal distribution of the residuals:** The residuals do not look normally distributed in the histogram but the values are randomly distributed around the green line in the QQ-Normal plot so we assume normally distributed residuals.
Equal variances of the factor levels: There is no funnel in the Fitted-values plot so we assume homoscedasticity. This corresponds to the result of the Bartlett test.
The residuals are randomly distributed around the green line in the index plot and the line is more or less straight. Therefore, there is no indication that something went wrong with the data collection.



Normal distribution of the residuals: The residuals do not look normally distributed in the histogram but the values are randomly distributed around the green line in the QQ-Normal plot so we assume normally distributed residuals.
Equal variances of the factor levels: There is no funnel in the Fitted-values plot so we assume homoscedasticity. This corresponds to the result of the Bartlett test.
The residuals are randomly distributed around the green line in the index plot and the line is more or less straight. Therefore, there is no indication that something went wrong with the data collection.

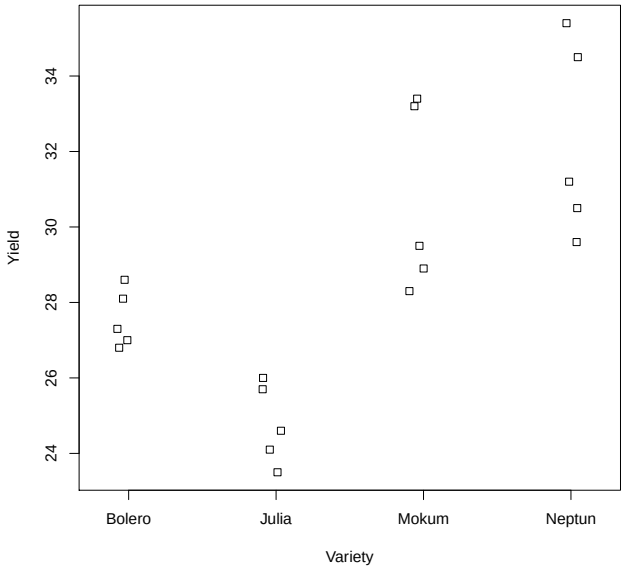
3 Carrots

(a) Response variable: yield. Factor variable: variety, 4 factor levels (Julia, Mokum, Neptun, Bolero) with 5 replications each. $N = 20$.

(b) Does the variety have an influence on the yield in carrots?

(c) The design is balanced.

(d) A stripchart is appropriate due to the limited number of replications per factor level.



(e)
```
  variety     n  mean   var
  <fct>   <int> <dbl> <dbl>
1 Bolero      5  27.6 0.583
2 Julia       5  24.8 1.12
3 Mokum       5  30.7 5.99
4 Neptun      5  32.2 6.54
```

(f)
```
        Bartlett test of homogeneity of variances

data:  yield by variety
Bartlett's K-squared = 6.6579, df = 3, p-value = 0.08364
```

The factor levels equal variance ($p = 0.08364$).

(g) Null hypothesis: There is no significant difference between the mean yields for different varieties. The variety has no significant influence on the yield. Alternativ hypothesis: At least one of the varieties results in a significantly different yield. The variety has a significant influence on the yield.

```
            Df Sum Sq Mean Sq F value   Pr(>F)
variety      3 164.95   54.98   15.45 5.52e-05 ***
Residuals   16  56.94    3.56
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusion: The variety has a significant influence on the yield ($p = 5.52 \times 10^{-5}$).

(h)
```
Tables of effects

 variety
variety
Bolero  Julia  Mokum Neptun
 -1.25  -4.03   1.85   3.43
```

(i) There are no observations with a Cook's distance greater than 1. Therefore, we assume

that there are no outliers.

(j) $y_{41} = \mu + \vartheta_4 + \epsilon_{41}$
$27.3 = 28.81 - 1.25 - 0.26$
μ is the true mean, ϑ_4 is the effect of variety Bolero and ϵ_{41} is the residual for observation y_{41}, *i.e.* the deviation of observation y_{41} from the prediction. $j = 1$ means that it is the first replication. (Note that we use the estimators for μ and ϑ_4, that is $\hat{\mu}$ and $\hat{\vartheta}_3$, as a number in our equation.)

(k) $y_{41} = \mu_4 + \epsilon_{41}$
$27.3 = 27.56 - 0.26$
μ_4 is the mean of the fourth factor level (Bolero) and ϵ_{41} is the residual for observation y_{41}, *i.e.* the deviation of observation y_{41} from the prediction.

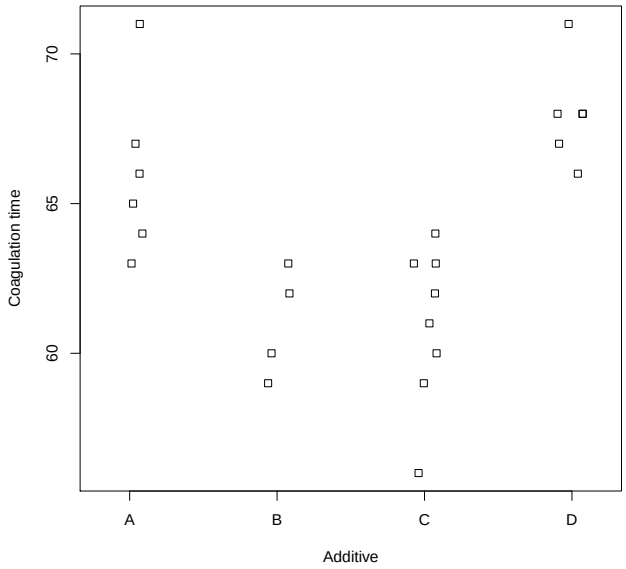
4 Blood

(a) Response variable: coagulation time. Factor variable: feeding additive, 4 factor levels (A, B, C, D) with 6, 4, 8, and 6 replications, respectively. $N = 24$.

(b) Does the feeding additive have an influence on the blood coagulation time?

(c) The design is unbalanced.

(d) A stripchart is appropriate due to the limited number of replications per factor level.



(e)

additive	n	mean	var
<fct>	<int>	<dbl>	<dbl>
1 A	6	66	8
2 B	4	61	3.33
3 C	8	61	6.86
4 D	6	68	2.8

(f)
```
        Bartlett test of homogeneity of variances

data:  ctime by additive
Bartlett's K-squared = 1.668, df = 3, p-value = 0.6441
```

The factor levels have equal variance ($p = 0.6441$).

(g) Null hypothesis: There is no significant difference between the blood coagulation times for different feeding additives. The feeding additive has no significant influence on the blood coagulation time. Alternativ hypothesis: At least one of the feeding additive results in a significantly different blood coagulation time. The feeding additive has a significant influence on the blood coagulation time.

```
            Df Sum Sq Mean Sq F value   Pr(>F)
additive     3    228    76.0   13.57 4.66e-05 ***
Residuals   20    112     5.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusion: The feeding additive has a significant influence on the blood coagulation time ($p = 4.66 \times 10^{-5}$).

(h)
```
Tables of effects

 additive
    A  B  C D
    2 -3 -3 4
rep 6  4  8 6
```

(i) There are no observations with a Cook's distance greater than 1. Therefore, we assume that there are no outliers.

(j) $\mu_2 = \mu + \vartheta_2 = 64 - 3 = 61$
(Note that we use the estimators for μ and ϑ_4, that is $\hat{\mu}$ and $\hat{\vartheta}_2$, as a number in our equation.)

(k) $\epsilon_{21} = y_{21} - \mu_2 = 62 - 61 = 1$
$\epsilon_{41} = y_{41} - \mu_4 = 68 - 68 = 0$
(Note that we use the estimator for μ_2, that is $\hat{\mu}_2$, as a number in our equation.)

Multiple pairwise comparisons and linear contrasts

Matthias Frisch and Carola Zenke-Philippi

1 Pairwise comparisons

Set the working directory first!

Load required packages:

```
library("emmeans")     # Provides: emmeans, test, contrast
library("multcomp")    # Provides: cld
library("dplyr")       # Provides: summarise
```

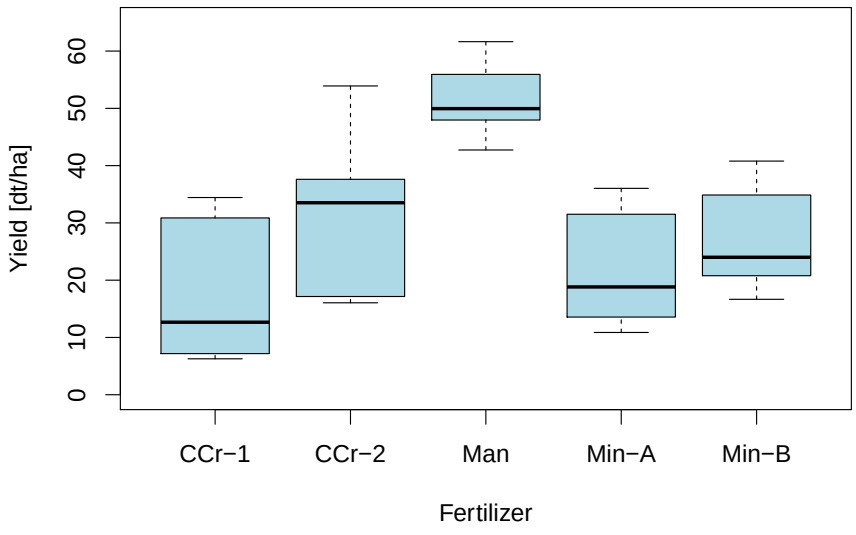
Data input:

```
dueng <- read.table ( "v14-u22-01.csv", header=TRUE, sep=";", dec=",",
                      stringsAsFactors = TRUE)
str (dueng)
```

```
> str(dueng)
'data.frame':        25 obs. of  3 variables:
 $ fert : Factor w/ 5 levels " CCr-1 "," CCr-2 ",..: 4 5 3 1 2 4 5 3 1 2 ...
 $ rep  : Factor w/ 5 levels "r1","r2","r3",..: 1 1 1 1 1 2 2 2 2 2 ...
 $ yield: num  10.87 24 55.93 7.18 53.91 ...
```

The command `read.table()` was explained in the R introduction. Since R 4.0 came out, it is necessary to add an argument `stringsAsFactors = TRUE` that tells R to convert any character variables, also called strings, in the data frame to factors.

```
boxplot( yield ~ fert, data=dueng )
```



Descriptive statistics

```
dueng %>% group_by(fert) %>%
    summarise ( n=n(), mean=mean(yield), var=var(yield),
                min=min(yield), max=max(yield))
```

```
  fert          n  mean   var   min   max
  <fct>     <int> <dbl> <dbl> <dbl> <dbl>
1 " CCr-1 "     5  18.3 180.   6.27  34.4
2 " CCr-2 "     5  31.6 247.  16.0   53.9
3 " Man "       5  51.6  53.5 42.7   61.6
4 " Min-A "     5  22.2 123.  10.9   36.0
5 " Min-B "     5  27.4 101.  16.7   40.8
```

Analysis of variance

Global test of the ANOVA:

```
model.1 <- aov ( yield ~ fert, data=dueng)
summary (model.1)
```

```
> summary (model.1)
            Df Sum Sq Mean Sq F value  Pr(>F)
fert         4   3381   845.4   5.998 0.00243 **
Residuals   20   2819   140.9
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Does the fertilizer have a significant influence on the yield?

Means and standard errors

Calculation of means and respective standard errors with the `emmeans` command.

```
emm.f <- emmeans ( model.1, ~fert )     # Means of factor levels
```

```
> emm.f
 fert    emmean   SE df lower.CL upper.CL
  CCr-1    18.3 5.31 20     7.21     29.4
  CCr-2    31.6 5.31 20    20.57     42.7
  Man      51.6 5.31 20    40.57     62.7
  Min-A    22.2 5.31 20    11.08     33.2
  Min-B    27.4 5.31 20    16.34     38.5

Confidence level used: 0.95
```

The first column shows the factor level of the fertilizer. We find the estimated mean of the factor levels, $\hat{\mu}_i$, in the second column. The third column contains the standard errors for the means (SEM).

Standard errors for a mean (SEM) and for a difference between means (SED), equations on lecture slide v14-02-026:

```
summary(model.1)[[1]][2,3]
( sem <- sqrt(140.9447/5)  )             # Standard error of a mean
( sed <- sqrt(140.9447/5) * sqrt(2) )    # Standard error of a difference between means
```

```
> summary(model.1)[[1]][2,3]
[1] 140.9447
> ( sem <- sqrt(140.9447/5)  )
[1] 5.309326
> ( sed <- sqrt(140.9447/5) * sqrt(2) )
[1] 7.50852
```

The value 140.9447 comes from the analysis of variance. It is the residual variance.

The residual variance is divided by 5 because of the 5 replications.

Note that the equation we use here is only valid if the data are balanced! If the treatment groups have different sample sizes, you need to use a different equation and the SED will be different depending on which comparison you make because of the different number of replications. Here, we only have one SED for all comparisons.

Least significant difference on basis of the t-distribution

We can now estimate pairwise differences between the fertilizers (see the table on lecture slide v14-02-027). The sample means which estimate the true means will never be exactly the same. A small difference might be caused by chance while the true means are in fact equal, a large difference is probably due to the fertilizer effect. But when is a difference "small" or "large"? We use the LSD ("least significant difference") as a threshold: We consider two means as signficantly different if the difference between them is larger than the LSD.

```
alpha <- 0.05
error.df <- 20
( lsd <- qt(1-alpha/2,error.df) * sed)
```

The `error.df` is taken from the ANOVA table: degrees of freedom of the residual.

```
> ( lsd <- qt(1-alpha/2,error.df) * sed)
[1] 15.6625
```

If the difference between the means of two factor levels is greater than 15.6625, then the factor levels are significantly different. The t-distribution does not take multiple testing into account. With the probability of $\alpha = 5\%$ a false positive is found for *each individual comparison.* The *comparison-wise* error rate is 5%.

Least significant difference on basis of the Tukey distribution

```
alpha <- 0.05
factor.levels <- 5
( hsd <- qtukey ( 1-alpha, factor.levels, error.df ) * sem)
```

```
> ( hsd <- qtukey ( 1-alpha, factor.levels, error.df ) * sem)
[1] 22.46831
```

The Tukey distribution takes multiple testing into accout. If two treatment means have a difference of more than 22.46831, they are considered significantly different. With a the probability of $\alpha = 5\%$ a false positive is found in *all comparisons together*. The *family-wise* error rate is 5%.

The LSD value is smaller than the HSD value. Hence, more differences are found using the LSD. This means that the type II error rate (of not finding a difference that actually exists - false negative) is smaller for the LSD than for the HSD. The HSD has a smaller family-wise type I error rate α, which means a lower risk for false-positives, but the tradeoff is a greater type II error rate β. This also means that the power of the HSD is lower: Real differences are more difficult to detect with the HSD than with the LSD.

Note that the HSD is also a least significant difference. Sometimes it is also called "LSD with Tukey adjustment" or something equivalent.

t-test

Comparison of factor levels CCr-2 with Man, using a *t*-test

```
31.648 - 51.642
( t.test.statistic <- -19.994 / sed )
( p.t <-  2 * pt ( abs(-2.663), lower.tail=FALSE, df=20) )
```

```
> 31.648 - 51.642
[1] -19.994
> ( t.test.statistic <- -19.994 / sed )
[1] -2.662842
> ( p.t <-  2 * pt ( abs(-2.663), lower.tail=FALSE, df=20) )
[1] 0.01493745
```

t-tests for all pairwise comparisons of factor levels can be carried out with the `contrast` command:

```
test ( contrast ( emm.f, "pairwise"),
       adjust="none")
```

To get the unadjusted p-values of the t-test the option `adjust="none"` is required.

```
> contrast ( emm.f, "pairwise", adjust="none")
 contrast              estimate   SE df t.ratio p.value
 ( CCr-1 ) - ( CCr-2 )   -13.37 7.51 20 -1.780  0.0903
 ( CCr-1 ) -  Man        -33.36 7.51 20 -4.443  0.0002
 ( CCr-1 ) - ( Min-A )    -3.88 7.51 20 -0.516  0.6114
 ( CCr-1 ) - ( Min-B )    -9.14 7.51 20 -1.217  0.2378
 ( CCr-2 ) -  Man        -19.99 7.51 20 -2.663  0.0149
 ( CCr-2 ) - ( Min-A )     9.49 7.51 20  1.264  0.2208
 ( CCr-2 ) - ( Min-B )     4.23 7.51 20  0.563  0.5796
  Man  - ( Min-A )        29.48 7.51 20  3.927  0.0008
  Man  - ( Min-B )        24.22 7.51 20  3.226  0.0042
 ( Min-A ) - ( Min-B )    -5.26 7.51 20 -0.701  0.4915
```

The column "estimate" contains the estimated difference between the two treatment means. The p value can be compared to our significance level α and tells us whether the two treatments are significantly different. For example, the wheat yields for cover crop 1 and manure are significantly different, the wheat yields for mineral fertilizers A and B are not.

Tukey test

```
  ( tukey.test.statistic <- 19.994 / sem      )
  ( p.tukey <- ptukey (tukey.test.statistic, nmeans=5, df=20, lower.tail=FALSE))
```

```
> ( tukey.test.statistic <- 19.994 / sem      )
[1] 3.765827
> ( p.tukey <- ptukey (tukey.test.statistic, nmeans=5, df=20, lower.tail=FALSE))
[1] 0.09609941
```

The Tukey test for all pairwise comparisons:

```
  test ( contrast ( emm.f, "pairwise"),
         adjust="tukey")
```

```
 contrast              estimate   SE df t.ratio p.value
 ( CCr-1 ) - ( CCr-2 )   -13.37 7.51 20 -1.780  0.4115
 ( CCr-1 ) -  Man        -33.36 7.51 20 -4.443  0.0021
 ( CCr-1 ) - ( Min-A )    -3.88 7.51 20 -0.516  0.9847
 ( CCr-1 ) - ( Min-B )    -9.14 7.51 20 -1.217  0.7419
 ( CCr-2 ) -  Man        -19.99 7.51 20 -2.663  0.0961
 ( CCr-2 ) - ( Min-A )     9.49 7.51 20  1.264  0.7152
 ( CCr-2 ) - ( Min-B )     4.23 7.51 20  0.563  0.9789
  Man  - ( Min-A )        29.48 7.51 20  3.927  0.0066
  Man  - ( Min-B )        24.22 7.51 20  3.226  0.0307
 ( Min-A ) - ( Min-B )    -5.26 7.51 20 -0.701  0.9539

P value adjustment: tukey method for comparing a family of 5 estimates
```

The result of pairwise comparisons can be displayed in a compact letter display. Treatments which are not significantly different are in the same group. They get the same letter or number.

```
cld ( emm.f, adjust="tukey")
```

```
> cld ( emm.f, adjust="tukey")
 fert    emmean   SE df lower.CL upper.CL .group
  CCr-1    18.3 5.31 20     3.22     33.3  1
  Min-A    22.2 5.31 20     7.10     37.2  1
  Min-B    27.4 5.31 20    12.36     42.5  1
  CCr-2    31.6 5.31 20    16.59     46.7  12
  Man      51.6 5.31 20    36.58     66.7   2

Confidence level used: 0.95
Conf-level adjustment: sidak method for 5 estimates
P value adjustment: tukey method for comparing a family of 5 estimates
significance level used: alpha = 0.05
```

Dunnett test

The Dunnett test is used to compare treatments with a control. If there are k treatments and one control, the Dunnett test controls the type I error rate of false positives for all k comparisons of the treatments with the control. If manure is used as control, and the other treatments should be compared with manure, the number of the factor level of manure is required.

```
levels(dueng$fert)
```

```
> levels(dueng$fert)
[1] " CCr-1 " " CCr-2 " " Man "   " Min-A " " Min-B "
```

We use the third factor level as the control. If we can be sure that the treatments must be either greater or smaller than the control, then a one-sided test can be carried out. Here we assume that the treatments can be only smaller than the control and use the option `side="<"` for the one-sided test. The Dunnett test uses the multivariate t-distribution so we use `adjust="mvt"`.

```
test ( contrast (emm.f, "trt.vs.ctrl", ref=3),
       side="<" ,
       adjust="mvt")
```

```
> test ( contrast ( emm.f, "trt.vs.ctrl", ref=3),
+        side="<" ,
+        adjust="mvt")
 contrast          estimate   SE df t.ratio p.value
 ( CCr-1 ) -  Man     -33.4 7.51 20 -4.443  0.0004
 ( CCr-2 ) -  Man     -20.0 7.51 20 -2.663  0.0244
 ( Min-A ) -  Man     -29.5 7.51 20 -3.927  0.0015
 ( Min-B ) -  Man     -24.2 7.51 20 -3.226  0.0071

P value adjustment: mvt method for 4 tests
P values are left-tailed
```

The values of the multivariate t-distribution are calculated with a numerical Monte-Carlo approach, which involves a random number generator. This means that the results can differ slightly each time the command is executed.

2 Linear contrasts

Linear contrasts are used to compare the means of groups of treatments.

Orthogonal contrasts

If there are k factor levels, and $k-1$ orthogonal linear contrasts are tested using a comparison-wise significance level $\alpha_{\max}$, then not only the *comparison-wise* error rate is smaller than $\alpha_{\max}$, but also the *family-wise* error rate is smaller than $\alpha_{\max}$. In other words: For $k-1$ orthogonal contrasts, an adjustment of the p-values for multiple testing is not required.

The order of the factor levels determines the numbers for the contrasts.

```
> levels(dueng$fert)
[1] " CCr-1 " " CCr-2 " " Man "   " Min-A " " Min-B "
```

```
C1 <- list  ( " CCr-1 vs CCr-2" = c( 1, -1,  0,  0,  0 ) ,
              " Min-A vs Min-B" = c( 0,  0,  0,  1, -1 ) ,
              " Org vs Min"     = c( 2,  2,  2, -3, -3 ) ,
              " CCr vs Man"     = c( 1,  1, -2,  0,  0 )  )

test ( contrast ( emm.f, C1 ),
       adjust="none")
```

"Orthogonal" means that if you multiply any of these contrasts by any other, the result is 0.

```
> c( 1, -1,  0,  0,  0 ) %*% c( 2,  2,  2, -3, -3 )
     [,1]
[1,]    0
```

Or by hand: $1 \times 2 + (-1) \times 2 + 0 \times 2 + 0 \times (-3) + 0 \times (-3) = 2 + (-2) + 0 + 0 + 0 = 0$.

```
> test ( contrast ( emm.f, C1 ),
+        adjust="none")
 contrast        estimate    SE df t.ratio p.value
  CCr-1 vs CCr-2   -13.37  7.51 20 -1.780  0.0903
  Min-A vs Min-B    -5.26  7.51 20 -0.701  0.4915
  Org vs Min         54.41 29.08 20  1.871  0.0760
  CCr vs Man        -53.35 13.01 20 -4.103  0.0006
```

Manure vs. other fertilizers

The contrasts are not orthogonal, therefore the p values are adjusted. The Hochberg adjustment (in other programms also called Bonferroni-Holm adjustment) is used here.

```
C2 <- list (  "Man vs CCr1 " = c( -1,  0,  1,  0,  0 ) ,
              "Man vs CCr2 " = c(  0, -1,  1,  0,  0 ) ,
              "Man vs Min-A" = c(  0,  0,  1, -1,  0 ) ,
              "Man vs Min-B" = c(  0,  0,  1,  0, -1 ) )
```

The first two contrasts are not orthogonal:

```
> c( -1, 0,  1,  0,  0 ) %*% c( 0, -1, 1, 0, 0 )
     [,1]
[1,]    1
```

Or by hand: $-1 \times 0 + 0 \times (-1) + 1 \times 1 + 0 \times 0 + 0 \times 0 = 0 + 0 + 1 + 0 + 0 = 1$.

```
  test ( contrast ( emm.f, C2 ),
         adjust="hochberg")
```

```
 contrast     estimate   SE df t.ratio p.value
 Man vs CCr1      33.4 7.51 20 4.443   0.0010
 Man vs CCr2      20.0 7.51 20 2.663   0.0149
 Man vs Min-A     29.5 7.51 20 3.927   0.0025
 Man vs Min-B     24.2 7.51 20 3.226   0.0085

P value adjustment: hochberg method for 4 tests
```

The test can be carried out one-sided, and then can be used as an alternative to the Dunnett test.

```
  test ( contrast ( emm.f, C2 ),
         side=">",
         adjust="hochberg")
```

```
 contrast     estimate   SE df t.ratio p.value
 Man vs CCr1      33.4 7.51 20 4.443   0.0005
 Man vs CCr2      20.0 7.51 20 2.663   0.0075
 Man vs Min-A     29.5 7.51 20 3.927   0.0013
 Man vs Min-B     24.2 7.51 20 3.226   0.0042

P value adjustment: hochberg method for 4 tests
P values are right-tailed
```

Two standards / three treatments

```
  C3 <- list (  " Min-A vs CCr1" = c(-1, 0, 0, 1, 0 ) ,
                " Min-A vs CCr2" = c( 0,-1, 0, 1, 0 ) ,
                " Min-A vs Man " = c( 0, 0,-1, 1, 0 ) ,
                " Min-B vs CCr1" = c(-1, 0, 0, 0, 1 ) ,
                " Min-B vs CCr2" = c( 0,-1, 0, 0, 1 ) ,
                " Min-B vs Man " = c( 0, 0,-1, 0, 1 )  )

  test ( contrast ( emm.f, C3 ),
         adjust="fdr")
```

```
 contrast       estimate   SE df t.ratio p.value
  Min-A vs CCr1     3.88 7.51 20  0.516  0.6114
  Min-A vs CCr2    -9.49 7.51 20 -1.264  0.3566
  Min-A vs Man    -29.48 7.51 20 -3.927  0.0050
  Min-B vs CCr1     9.14 7.51 20  1.217  0.3566
  Min-B vs CCr2    -4.23 7.51 20 -0.563  0.6114
  Min-B vs Man    -24.22 7.51 20 -3.226  0.0127

P value adjustment: fdr method for 6 tests
```

3 Exercises

(a) Multiple comparisons, (b) and (c) linear contrasts.

1 Doughnuts

For each of four types of fat, six batches of doughnuts were baked and the fat uptake was measured in g.

Type of fat			
Aldi	**Lidl**	**Mazola**	**Palmin**
64	78	75	55
72	91	93	66
68	97	78	49
77	82	71	64
56	85	63	70
95	77	76	68

(a) Is it possible that the fat sold by the discounter Aldi is the same as the brand product Palmin? Or is the fat sold by Aldi the same as the brand product Mazola? Is it possible that the fat sold by the discounter Lidl is the same as the brand product Palmin? Answer the questions by carrying out a Tukey test for all pairwise comparisons. (a1) What is the least significant difference ($\alpha = 0.05$) on basis of the Tukey test? (a2) What are the adjusted p-values of all pairwise comparisons?

(b) Check the above hypotheses with linear contrasts. Are the contrasts orthogonal? Carry out a Hochberg adjustment of the p-values if necessary.

(c) Assume that preliminary investigations found that Aldi sells Palmin and Lidl sells Mazola. Is the Aldi/Palmin fat significantly different from the Lidl/Mazola fat?

2 Carrots

The yield of four new varieties of carrots was assessed [dt/ha].

	Replication				
Variety	**1**	**2**	**3**	**4**	**5**
Mokum	28.3	33.4	29.5	28.9	33.2
Julia	24.1	24.6	25.7	23.5	26.0
Neptun	29.6	30.5	31.2	35.4	34.5
Bolero	27.3	28.1	28.6	27.0	26.8

(a) Which of the four carrot varieties differ in yield? Visualize the results of all pairwise comparisons with a display showing groups of treatments that are not significantly different.

(b) The varieties Mokum und Neptun are full-sibs and the varieties Julia and Bolero are also sibs. Test with linear contrasts whether (1) Mokum differs from Neptun, (2) Julia differs from Bolero, or wheter (3) the Mokum/Neptun family is different than the Julia/Bolero family. Are the contrasts orthogonal? Carry out a Hochberg adjustment of the p-values if required.

(c) Comment on the results of a. and b. Where are the differences? Why?

3 Blood

Lab animals obtained a diet with four different feeding additives. The time for blood coagulation [sec] was measured.

Additive	Coagulation time [s]			
A	62	67	65	64
B	62	60	63	59
C	56	62	60	61
D	68	66	71	67

The additive D is the standard additive and the experiment was carried out to check whether the coagulation time is lower with one of the new additives A, B, and C. Suppose that a lower coagulation time is better.

(a) Choose a suitable significance level for this experiment. Give reasons for your choice.

(b) Specify H_0 and H_1.

(c) Carry out a Dunnett-test. Why is this an appropriate test for this scenario?

(d) Compare the additives with unorthogonal linear contrasts and adjust the respective p-values with the false discovery rate.

(e) Comment on the differences in the outcomes of the tests.

4 Terminology

Make sure you understand what the following terms mean:

(a) test statistic,

(b) p-value,

(c) power (Testgüte),

(d) type I and type II error,

(e) conservative (What does it mean when a test is conservative?),

(f) family-wise and comparison-wise error rate,

(g) multiple testing problem.

Talk to your fellow students or ask a question in the Q&A sessions or in the Stud.IP forum if you do not know.

4 Solutions

1 Doughnuts

(a)

```
> hsd
[1] 16.23222

 contrast        estimate  SE df t.ratio p.value
 Aldi - Lidl          -13 5.8 20 -2.242  0.1462
 Aldi - Mazola         -4 5.8 20 -0.690  0.8998
 Aldi - Palmin         10 5.8 20  1.724  0.3378
 Lidl - Mazola          9 5.8 20  1.552  0.4271
 Lidl - Palmin         23 5.8 20  3.966  0.0039
 Mazola - Palmin       14 5.8 20  2.414  0.1066
```

It is possible that Aldi sells Palmin or Mazola: The estimated differences between the means are 4 and 10, respectively, and therefore smaller than the HSD. The p values are larger than 0.05. This means that there are no significant differences between the means and it is possible that Aldi indeed sells a brand product. The means of the fat sold by Lidl and Palmin, however, are significantly different: Their difference exceeds the threshold of 16.2 and the p value is small (0.0039).

(b)

```
 contrast       estimate  SE df t.ratio p.value
 Aldi vs Palmin       10 5.8 20  1.724  0.2002
 Aldi vs Mazola       -4 5.8 20 -0.690  0.4983
 Lidl vs Palmin       23 5.8 20  3.966  0.0023
```

Not all contrasts are orthogonal which means that the p values need to be adjusted. The result is the same as above.

(c)

```
 contrast                  estimate  SE df t.ratio p.value
 Aldi/Palmin vs Lidl/Mazola      -27 8.2 20 -3.292  0.0036
```

Yes, the two groups are significantly different.

2 Carrots

(a)

```
> hsd
[1] 3.414097

 contrast        estimate   SE df t.ratio p.value
 Mokum - Julia       5.88 1.19 16  4.928  0.0008
 Mokum - Neptun     -1.58 1.19 16 -1.324  0.5616
 Mokum - Bolero      3.10 1.19 16  2.598  0.0820
 Julia - Neptun     -7.46 1.19 16 -6.252  0.0001
 Julia - Bolero     -2.78 1.19 16 -2.330  0.1325
 Neptun - Bolero     4.68 1.19 16  3.922  0.0060

 variety emmean    SE df lower.CL upper.CL .group
 Julia     24.8 0.844 16     22.4     27.1  1
 Bolero    27.6 0.844 16     25.2     29.9  12
 Mokum     30.7 0.844 16     28.3     33.0   23
 Neptun    32.2 0.844 16     29.9     34.6    3
```

Julia is significantly different from Mokum and Neptun. Bolero is significantly different from Neptun.

(b)

```
contrast                    estimate   SE df t.ratio p.value
Mokum vs Neptun                -1.58 1.19 16 -1.324  0.2040
Julia vs Bolero                -2.78 1.19 16 -2.330  0.0332
Mokum/Neptun vs Julia/Bolero   10.56 1.69 16  6.258  <.0001
```

(1) Mokum is not significantly different from Neptun. (2) Julia is significantly different from Bolero. (3) The two families are significantly different.

(c) An additional difference between Julia and Bolero was found in b. The reason is that with orthogonal linear contrasts, no adjustment of the p values is necessary. This increases the power of the test.

3 Blood

(a) The standard additive D is probably already known in respect to possible side effects and has proven to be effective and harmless (otherwise it would not be the standard). We therefore want to be sure that a new additive with unknown side effects is *really* better than the standard before we make the switch. This means that we would like to achieve a low probability for a type I error and choose a significance level of 0.05 or even 0.01. (In the exam, you will be told which significance level to use.)

(b) H_0: The treatments A, B, or C are not significantly better than the control. They have a coagulation time that is equal to or larger than the coagulation time of standard additive D. $\mu_{\text{treatment}} \geq \mu_{\text{control}}$.

H_1: The treatments A, B, or C are significantly better than the control. They have a coagulation time that is significantly smaller than the coagulation time of standard additive D. $\mu_{\text{treatment}} < \mu_{\text{control}}$.

(c)

```
 contrast estimate   SE df t.ratio p.value
 A - D       -3.50 1.55 12 -2.256  0.0524
 B - D       -7.00 1.55 12 -4.513  0.0009
 C - D       -8.25 1.55 12 -5.318  0.0002

P value adjustment: mvt method for 3 tests
P values are left-tailed
```

The Dunnett test is appropriate because the new additives A, B, and C are compared with the standard D. It is carried out one-sided. Additive A is not significantly better than the standard, additive B and C are.

(d)

```
 contrast estimate   SE df t.ratio p.value
 A - D       -3.50 1.55 12 -2.256  0.0217
 B - D       -7.00 1.55 12 -4.513  0.0005
 C - D       -8.25 1.55 12 -5.318  0.0003

P value adjustment: fdr method for 3 tests
P values are left-tailed
```

The results is the same as above on a significance level of 0.01. On a significance level of 0.05, an additional significant difference was found between additive A and the standard additive D.

(e) For a significance level of 0.05: The probability that at least one of the differences in .c is

a false positive is 0.05. The probability that a detected difference in .d is a false positive is 0.05. An additional difference was found with the FDR because this adjustment method is less conservative and has a higher power.

Balanced multi-factor ANOVA

Matthias Frisch

1 Generating factor levels for manual data input

The following example shows how to manually enter data if you have two factors.

		Replication			
		1	2	3	4
a	I	69	91	84	76
	II	66	56	58	60
	III	8	4	18	10
b	I	44	24	30	22
	II	118	96	80	106
	III	132	88	101	119

```
yld <- c ( 69,91,84,76,  66,56,58,60,  8,4,18,10, 44,24,30,22,
          118,96,80,106, 132,88,101,119)
var <- gl ( n=2, k=12, length=24, labels=c("a","b") )
cli <- gl ( n=3, k=4,  length=24, labels=c("I","II","III") )
rep <- gl ( n=4, k=1,  length=24, labels=c("r1","r2","r3","r4"))
d2 <- data.frame(var,cli,rep,yld)
```

A detailed explanation for data input can be found at the end for the limpets example.

2 Two-factor ANOVA: Climate experiment

Load required packages:

```
library("GAD" )        # Provides: gad, as.fixed, as.random, estimates
```

The climate experiment has two "treatment" factors, variety with 2 levels ($i = 1$ or $i = 2$) and climate with 3 levels ($j = 1, ..., 3$). There are 4 replications ($k = 1, ..., 4$). It was conducted in a factorial design: Each variety was investigated in each climate. All possible combinations of the factor levels are present.

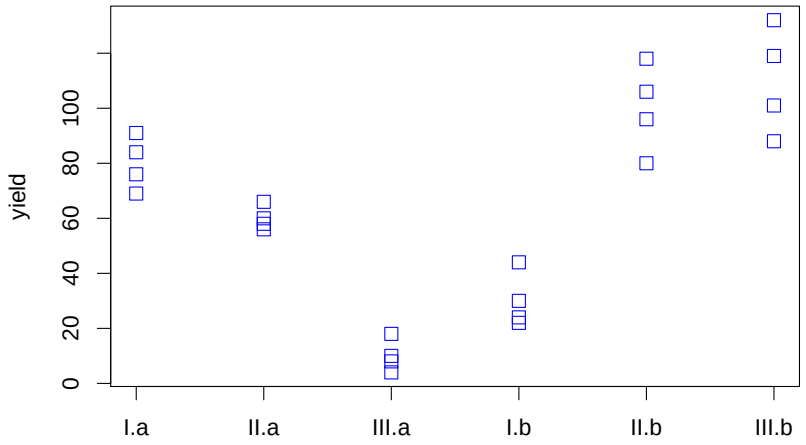
```
cl <- read.table ( "v14-u31-01.csv", header=T, sep=";", dec=",",
                   stringsAsFactors = T )
str(cl)
```

```
> str(cl)
'data.frame':        24 obs. of  5 variables:
 $ variety: Factor w/ 2 levels "a","b": 1 1 1 1 1 1 1 1 1 1 ...
 $ climate: Factor w/ 3 levels "I","II","III": 1 1 1 1 2 2 2 2 3 3 ...
 $ cv     : Factor w/ 6 levels "a-I","a-II","a-III",..: 1 1 1 1 2 2 2 2 3 3 ...
```

```
$ rep    : Factor w/ 4 levels "r1","r2","r3",..: 1 2 3 4 1 2 3 4 1 2 ...
$ yield  : int  69 91 84 76 66 56 58 60 8 4 ...
```

Analysis of variance

```
stripchart ( yield ~ climate:variety,
             vertical=T, col="blue",
             data=cl )
```



The GAD package allows to specify whether a factor is fixed or random:

```
cl$C  <- as.fixed (cl$climate)
cl$V  <- as.fixed (cl$variety)
```

```
m.1 <- lm ( yield ~ C + V + C:V, data=cl)
gad ( m.1 )
```

```
> gad ( m.1 )
Analysis of Variance Table

Response: yield
          Df Sum Sq Mean Sq F value    Pr(>F)
C          2   2800  1400.0  9.5599  0.001482 **
V          1   5400  5400.0 36.8741 9.720e-06 ***
C:V        2  22800 11400.0 77.8452 1.379e-09 ***
Residual 18   2636   146.4
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

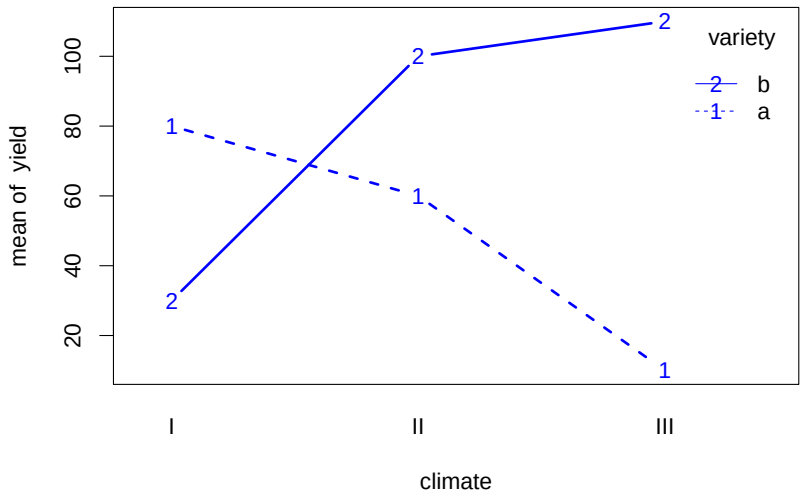
The climate as well as the variety have a significant influence on the yield. Additionally, significant interactions are present.

The model equation is $y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \epsilon_{ijk}$.

Plot interaction

We have to check for interactions if we have two factors in a model:

```
with ( cl, interaction.plot ( climate, variety, yield, type="b" ))
```



We can see that the varieties react differently to the climates.

Treatment means and effects

```
model.tables ( aov(m.1), "means" )
model.tables ( aov(m.1), "effects" )
```

The statistical model behind this trial is: $y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + e_{ijk}$. Each observation y_{ijk} is composed of the true mean μ, an effect of the factor level i of the factor α, an effect of the factor level j of the factor β, an interaction γ between both factor levels and an random error e_{ijk}.

```
> model.tables(aov(m.1),"means")

Grand mean
65

climate
  I  II III
 55  80  60

variety
 a  b
50 80

       variety
climate a   b
    I   80  30
    II  60 100
```

```
> model.tables(aov(m.1),"effects")

climate
  I  II III
-10  15  -5

variety
  a   b
-15  15

       variety
climate a   b
    I   40 -40
    II  -5   5
    III -35  35
```

```
   III  10 110
```

Least significant differences

```
mse <- gad(m.1)$'Mean Sq'[4]
dfe <- gad(m.1)$'Df'[4]
alpha <- 0.05
n.v   <- 12 ; k.v  <- 2
n.c   <-  8 ; k.c  <- 3
n.cv  <-  4 ; k.cv <- 6
sed.c  <- sqrt ( 2*mse / n.c )
sed.v  <- sqrt ( 2*mse / n.v )
sed.cv <- sqrt ( 2*mse / n.cv )
lsd.c  <- qt( 1-alpha/2,dfe ) * sed.c
lsd.v  <- qt( 1-alpha/2,dfe ) * sed.v
lsd.cv <- qt( 1-alpha/2,dfe ) * sed.cv
hsd.c  <- qtukey( 1-alpha,k.c ,dfe ) * sed.c / sqrt(2)
hsd.v  <- qtukey( 1-alpha,k.v ,dfe ) * sed.v / sqrt(2)
hsd.cv <- qtukey( 1-alpha,k.cv,dfe ) * sed.cv/ sqrt(2)
```

k is the number of factor levels, n is the number of replications per factor level. $n \times k$ is the total number of observations, 24 in this case. The MSE comes from the ANOVA table.

```
> sed.c
[1] 6.050712
> sed.v
[1] 4.940385
> sed.cv
[1] 8.556998
> lsd.c
[1] 12.71207
> lsd.v
[1] 10.37936
> lsd.cv
[1] 17.97759
> hsd.c
[1] 15.4424
> hsd.v
[1] 10.37936
> hsd.cv
[1] 27.19444
```

Since we have two treatment factors (variety and climate), we need an LSD and HSD for each factor and a third for each factor level combination. You use `lsd.c` and `hsd.c` for comparisons between two climates, `lsd.v` and `hsd.v` for comparisons between two varieties and `lsd.cv` and `hsd.cv` for comparisons between two specific climate-variety combinations.

3 Hierarchical ANOVA: Soils experiment

This experiment is different: We investigate the yield of six villages in three different regions. However, since a village can only be in *one* region, village 1 in region 1 is different from village 1 in region 2. We call such a design "nested" or "hierarchical" and we need to use a different kind of analysis.

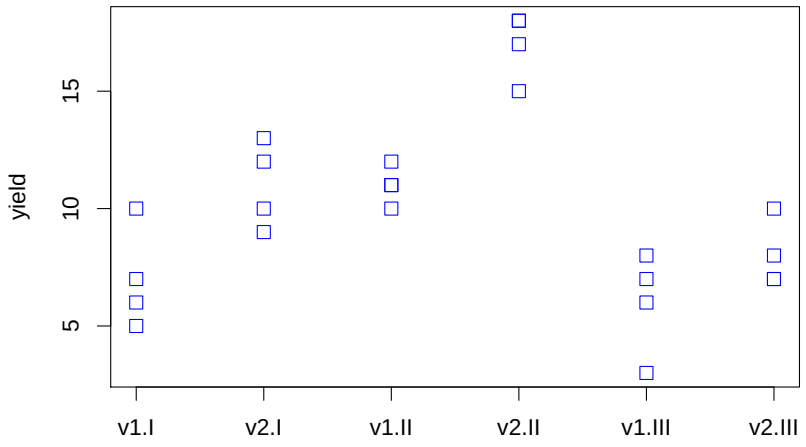
```
soils <- read.table ( "v14-u31-02a.csv", header=T, sep=";", dec=",",
                      stringsAsFactors = T )
str(soils)
```

```
> str(soils)
'data.frame':        24 obs. of  4 variables:
 $ region : Factor w/ 3 levels "I","II","III": 1 1 1 1 1 1 1 1 2 2 ...
 $ village: Factor w/ 2 levels "v1","v2": 1 1 1 1 2 2 2 2 1 1 ...
 $ field  : Factor w/ 4 levels "f1","f2","f3",..: 1 2 3 4 1 2 3 4 1 2 ...
 $ yield  : int  5 6 7 10 9 10 12 13 10 11 ...
```

Analysis of variance

Graphical display:

```
stripchart ( yield ~ village:region,
             vertical=T, col="blue",
             data=soils )
```



Fixed and random factors:

```
soils$R  <- as.fixed (soils$region)
soils$V  <- as.random(soils$village)
```

We are interested in the effects of the particular regions and therefore regard them as fixed. We are generally interested in the effects of the villages but not in the effects of the particular ones we have in our experiment so we regard them as random.

The villages are nested in the regions:

```
m <- lm ( yield ~ R + R:V, data=soils )
gad (m)
```

```
estimates (m)
```

The statistical model behind the analysis is $y_{ijk} = \mu + \alpha_i + \beta_{ij} + e_{ijk}$.

```
> gad (m)
Analysis of Variance Table

Response: yield
         Df Sum Sq Mean Sq F value    Pr(>F)
R         2    208 104.000  2.7857    0.2071
R:V       3    112  37.333 12.9231 9.648e-05 ***
Residual 18     52   2.889
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> estimates(m)
$tm          $mse                              $f.versus
    R V n             Mean square estimates        F-ratio versus
R   0 2 4    R        "Res + R:V + R"          R   "R:V"
R:V 1 1 4    R:V      "Res + R:V"              R:V "Residual"
Res 1 1 1    Residual "Res"
```

The last part of the output shows how the F tests in the ANOVA are done. The hierarchical design means that we cannot simply do them all with the residual variance.

Treatment means and LSD for the first level

Means of regions:

```
model.tables(aov(m),"means")
model.tables(aov(m),"effects")
```

```
> model.tables(aov(m),"means")
Tables of means
Grand mean

10

 R
R
  I  II III
  9  14   7

 R:V
    V
R    v1 v2
  I   7 11
  II 11 17
  III 6  8

> model.tables(aov(m),"effects")
Tables of effects

 R
R
  I  II III
 -1   4  -3
```

```
 R:V
    V
R     v1 v2
  I   -2  2
  II  -3  3
  III -1  1
```

The link between the tables: $\hat{y}_{32} = \hat{\mu} + \hat{\alpha}_2 + \hat{\beta}_{32} = \hat{y}_{32} = 10 + (-3) + 1 = 8$. $\hat{y}_{ij}$ and $\hat{\mu}$ come from the table of means and the effects $\hat{\alpha}$ and $\hat{\beta}$ come from the table of effects.

The LSD can only be estimated for fixed effects.

```
va <- gad(m); dfe <- va$Df[2]; mse <- va$"Mean Sq"[2]
b <- 2 ; n <- 4
sed <- sqrt(2*mse/(b*n))
alpha <- 0.05;
lsd <- sed * qt( 1-alpha/2,dfe )
```

There are $b = 2$ villages in each region (*i.e.* two levels of factor II) and $n = 4$ fields in each village. The equation for the SED can be found on lecture slide `v14-03-227`. Note that the equation for the LSD is always the same but the equation for the SED depends on the experimental design and on the comparisons we want to make.

```
> model.tables(aov(m),"means", cterms="R") # means only for the regions
R
  I  II III
  9  14   7
> sed
[1] 3.05505
> lsd
[1] 9.722534
```

Two regions are considered to be significantly different if the difference in the mean yield is larger than the LSD of 9.72 - which is not the case here. This is consistent with the result of the global test of the ANOVA in which no significance for the regions was detected.

Analysis as single factor ANOVA

```
soils <- read.table ( "v14-u31-02.csv", header=T, sep=";", dec=",",
                      stringsAsFactors = T )
str(soils)
```

```
> str(soils)
'data.frame':        24 obs. of  4 variables:
 $ region : Factor w/ 3 levels "I","II","III": 1 1 1 1 1 1 1 1 2 2 ...
 $ village: Factor w/ 6 levels "a","b","c","d",..: 1 1 1 1 2 2 2 2 3 3 ...
 $ field  : Factor w/ 4 levels "f1","f2","f3",..: 1 2 3 4 1 2 3 4 1 2 ...
 $ yield  : int  5 6 7 10 9 10 12 13 10 11 ...
```

The data are the same, however, the coding is different: In the previous file, the villages were encoded as `v1` and `v2` and we needed additional information about the region to tell them apart.

Now we encode the villages with levels from `a` to `f` so that we know which village is which. Assume that there is no effect of the region and leave it out of the model:

```
soils$V  <- as.fixed(soils$village)
m.4 <- lm ( yield ~ V , data=soils )
gad (m.4)
```

The treatment factor in the single factor ANOVA is fixed.

```
> gad (m.4)
         Df Sum Sq Mean Sq F value    Pr(>F)
V         5    320  64.000  22.154 4.109e-07 ***
Residual 18     52   2.889
```

Treatment means and honestly significant differences

```
va <- gad(m.4)
dfe <- va$Df[2]
mse <- va$"Mean Sq"[2]
alpha <- 0.05
n <- 4
a <- 6
sem <- sqrt(mse/(n))
sed <- sqrt(2) * sem
hsd <- qtukey( 1-alpha,a,dfe ) * sem
```

```
> model.tables(aov(m.4),"means")
V
 a  b  c  d  e  f
 7 11 11 17  6  8
> sed
[1] 1.20185
> hsd
[1] 3.819523
```

Two villages are considered significantly different if their yield difference is larger than 3.82.

Display pairwise differences

```
attach(soils)
m <- tapply(yield,village,mean)
detach(soils)
source("v14-u31-00.R")
comp.lsmeans(m,hsd)
```

`tapply` applies the function `mean` to the yield values and groups by the villages. Then we load an R script written by Prof. Frisch with `source()` and use the function `comp.lsmeans()` that it contains in order to create a compact letter display (CLD). If you are interested, open the file `v14-u31-00.R` and have a look at its contents (do not write!). It shows an example of

how to write your own, customized R functions. We cover that in our profile module MP 100 Bioinformatics.

```
> comp.lsmeans(m,hsd)
      d   b   f
d 17  1
b 11      2
c 11      2
f 8       2   3
a 7           3
e 6           3
```

4 Rape seed experiment

The research question of the rape seed experiment is whether there are rape seed genotypes which show a higher drought tolerance than others.The design of the rapeseed experiment is even more complicated. Have a look at the lecture slides for a graphical display of the experimental design. It is a mixed design with crossed (GxE) and nested factors (E:R). We see that there are $a = 20$ rape seed genotypes which are investigated in $b = 2$ environments, no stress and drought stress. The treatment factors G and E are crossed because each genotype is investigated in each environment. For each environment, there are $r = 3$ replications. Each replication is located in a different part of the greenhouse. The replication therefore carries implicit greenhouse effects. The replication is nested in the environment because replication 1 in environment 1 is not the same as replication 1 in environment 2.

```
raps <- read.table ( "v14-u31-06.csv", header=T, sep=";", dec=",",
                     stringsAsFactors = T )
str(raps)
```

```
> str(raps)
'data.frame':        120 obs. of  4 variables:
 $ geno: Factor w/ 20 levels "l01","l02","l03",..: 1 1 1 1 1 1 2 2 2 2 ...
 $ env : Factor w/ 2 levels "e1","e2": 1 1 1 2 2 2 1 1 1 2 ...
 $ rep : Factor w/ 3 levels "r1","r2","r3": 1 2 3 1 2 3 1 2 3 1 ...
 $ hgt : int  197 234 196 227 234 207 232 220 199 245 ...
```

```
raps$G  <- as.fixed(raps$geno)
raps$E  <- as.fixed(raps$env)
raps$R  <- as.random(raps$rep)
```

We are interested in the effects of the genotypes and the environments in our experiment and therefore regard those as fixed. The effects of the replications are treated as random since the effect of each individual replication is not of interest in the experiment. The six areas can be regarded as a random sample of all possible effects of the greenhouse.

```
m <- lm ( hgt ~ G*E + E:R , data=raps )
gad (m)
estimates(m)
```

We specify two main effects for genotype and environment and their interactions as well as an effect for the replication nested in the environment. `G*E` is short for `G + E + G:E`.

```
> gad (m)
         Df  Sum Sq Mean Sq F value    Pr(>F)
G        19 11425.9   601.4  6.6832 7.866e-10 ***
E         1  9469.6  9469.6 51.5051  0.001996 **
G:E      19  1204.4    63.4  0.7045  0.803141
E:R       4   735.4   183.9  2.0433  0.096709 .
Residual 76  6838.6    90.0
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> estimates(m)
$mse                                  $f.versus
         Mean square estimates            F-ratio versus
G        "Res + G"                    G   "Residual"
E        "Res + E:R + E"              E   "E:R"
G:E      "Res + G:E"                  G:E "Residual"
E:R      "Res + E:R"                  E:R "Residual"
Residual "Res"
```

The model requires modifications in the F tests for the ANOVA. The effects of the genotypes and the environments are significant. There are no significant interactions between genotype and environment. The effect of the replication is not significant.

```
model.tables(aov(m),"means",cterms="G")
```

```
> model.tables(aov(m),"means",cterms="G")
   101    102    103    104    105    106    107    108    109    110    111
215.83 232.17 212.83 223.67 207.83 229.17 209.67 229.33 221.33 211.83 227.00
   112    113    114    115    116    117    118    119    120
217.83 217.00 224.83 210.67 212.83 195.67 225.83 214.17 197.83
```

Here we have estimates of the means of the genotypes.

The statistical model of the experiment is:

$$y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \rho_{k:j} + e_{ijk}$$

α_i: fixed effect of the genotype

β_j: fixed effect of the environment

γ_{ij}: fixed interaction effect between the two crossed factors genotype and environment

$\rho_{k:j}$: random replication effect nested within environments

e_{ijk} : residuals

```
snk.test(m,term ='G')
```

The estimation of the standard error that is required for the LSD and HSD is not as straightforward as before. We use `snk.test()` to determine the standard error and the degrees of freedom. Simply ignore the rest of the output.

```
> snk.test(m,term ='G')
...
Standard error = 2.345
Df = 76
...
```

```
sed <- sqrt(2) * 2.345; dfe <- 76; alpha <- 0.05
( lsd <- sed * qt( 1-alpha/2,dfe ) )
```

```
> ( lsd <- sed * qt( 1-alpha/2,dfe ) )
[1] 6.605045
```

Use this LSD for the comparison between the genotypes.

```
model.tables(aov(m),"means",cterms="G:E")
```

```
> model.tables(aov(m),"means",cterms="G:E")
     E
G     e1     e2
  101 209.00 222.67
  102 217.00 247.33
  103 204.33 221.33
  104 213.33 234.00
  105 200.33 215.33
  106 221.33 237.00
  107 205.00 214.33
  108 222.33 236.33
  109 215.33 227.33
  110 201.00 222.67
  111 216.67 237.33
  112 209.67 226.00
  113 203.00 231.00
  114 218.33 231.33
  115 203.33 218.00
  ...
```

```
snk.test(m,term ='G:E', among ='E', within ='G')
```

```
> snk.test(m,term ='G:E', among ='E', within ='G')
...
Standard error = 3.3163
Df = 76
...
```

```
sed <- sqrt(2)* 3.3163; dfe <- 76; alpha <- 0.05
( lsd <- sed * qt( 1-alpha/2,dfe ) )
```

```
> ( lsd <- sed * qt( 1-alpha/2,dfe ) )
[1] 9.340858
```

Use this LSD for the comparison between specific combinations of genotypes and environments.

5 Exercises

1 Limpets (Napfschnecken)



The oxygen uptake (`sauer`) of two limpet species was assessed in water with different salt concentration (Sokal and Rohlf 1995, p. 333). The following values in μg per mg weight and minute were measured. The species (`art`) and the salt concentrations (`konz`) are considered as fixed factors.

	Species	
Conc.	***A. scabra***	***A. digitalis***
I	7.16 8.26	6.14 6.14
	6.78 14.00	3.86 10.00
	13.60 16.10	10.40 11.60
	8.93 9.66	5.49 5.80
II	5.20 13.20	4.47 4.95
	5.20 8.39	9.90 6.49
	7.18 10.40	5.75 5.44
	6.37 7.18	11.80 9.90
III	11.11 10.50	9.63 14.50
	9.74 14.60	6.38 10.20
	18.80 11.10	13.40 17.70
	9.74 11.80	14.50 12.30

(a) Visualize the data with box-and-whisker plots. Estimate the mean value for each species, each salt concentration, and each combination of species and salt concentration.

(b) What are the treatment factors? Which factor levels do they have?

(c) How many replications are there per factor level combination? Is the design balanced or unbalanced?

(d) Is the experimental design crossed or hierarchical?

(e) We are interested in the specific limpets species and the specific salt concentrations in the experiment. Are the factors fixed or random?

(f) What is the statistical model for the experiment? What do the letters stand for?

(g) Is the oxygen uptake different for the two species? Does the oxygen uptake depend on the salt concentration? Do the two species react differently on a change in the salt concentration?

(h) Which salt concentrations are significantly different? Use a comparison-wise error rate of 5%. Do you get different results for a family-wise error rate of 5% ? Which of the two results would you use in a publication? Why?

(i) What is the least significant difference (LSD) between two concentrations for a comparison-

wise type 1 error rate of 10%?

(j) Estimate the Honestly Significant Difference (HSD) for a family-wise type 1 error rate of 10%. Compare the HSD and the LSD values. Explain the difference with respect to the type 1 and type 2 error rates.

The data are available in the file `v14-u31-03.csv`.

2 Fertilizer

The effect of four fertilizers on the wheat yield should be investigated. To draw conclusions on the wheat yield in general, five varieties were randomly selected from all varieties grown in the area under investigation. For each variety three replications were carried out and the yield in [dt/ha] was assessed.

	Variety				
Fertilizer	**1**	**2**	**3**	**4**	**5**
A	**57**	**26**	**39**	**23**	**48**
	46	**38**	**39**	**36**	**35**
	28	**20**	**43**	**18**	**48**
B	**67**	**44**	**57**	**74**	**61**
	72	**68**	**61**	**47**	**60**
	66	**64**	**61**	**69**	**75**
C	**95**	**92**	**91**	**98**	**78**
	89	**99**	**98**	**85**	**95**
	90	**89**	**82**	**85**	**89**
D	**92**	**96**	**98**	**99**	**99**
	88	**95**	**93**	**90**	**98**
	99	**99**	**98**	**98**	**99**

(a) Visualize the data with an appropriate plot. What do you see?

(b) What are the treatment factors? Which factor levels do they have?

(c) Is the experimental design crossed or hierarchical?

(d) Are the factors fixed or random?

(e) Is the yield different for the different fertilizers? Does the variety have an influence on the yield?

(f) What is a possible research question for the interaction between fertilizer and variety?

(g) Which F ratios are used for the F tests in the ANOVA?

(h) Estimate the mean yields for the fertilizers.

(i) Which fertilizers result in significantly different yields? Use a comparison-wise error rate of 5%.

(j) Compare the fertilizers with a Tukey test (family-wise type 1 error rate of 10%) and display the results in a CLD.

(k) What is the least significant difference (LSD) between two fertilizers for a comparison-wise type 1 error rate of 10%?

(l) Estimate the Honestly Significant Difference (HSD) between the fertilizers for a family-wise type 1 error rate of 10%. Compare the HSD and the LSD values and the result.

(m) Why do we not compare the yield of the different varieties?

The data are available in the file `v14-u31-05.csv`.

3 Chicken

The effect of three diets on the weight gain of chicken was investigated. Twelve chicken houses (Hühnerställe) were randomly chosen. For each diet, four chicken houses were selected and the chicken were fed with the diet. The weight gain of five chicken per chicken house was assessed. Carry out an analysis of the data set.

Diet	1				2				3			
Chicken house	1	2	3	4	5	6	7	8	9	10	11	12
	Weight gain [g]											
Chicken 1	100	70	77	98	94	84	90	90	84	84	88	46
2	51	90	75	85	86	81	88	58	52	78	86	83
3	39	102	73	90	52	97	78	70	103	51	77	106
4	56	81	65	90	81	102	65	65	61	76	66	88
5	56	83	78	22	60	78	45	114	63	32	91	63

(a) Visualize the data with an appropriate plot. What do you see?

(b) What are the treatment factors? Which factor levels do they have?

(c) Is the experimental design crossed or hierarchical?

(d) Are the factors fixed or random?

(e) Is the weight gain different for the different diets? Does the chicken house have an influence on the weight gain?

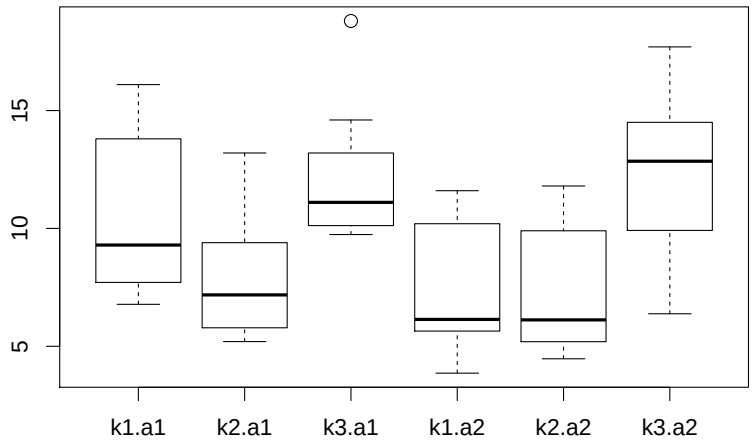
The data are available in the file `v14-u31-04a.csv`.

4 Data input

Practice manual data input by reading in the data tables of Exercises 1, 2 and 3 and assigning factor levels to the data manually.

6 Solutions

1 Limpets



(a)
```
Tables of means
Grand mean

9.619583

 A
A
    a1     a2
10.208  9.031

 K
K
    k1     k2     k3
 8.995  7.614 12.250

 A:K
    K
A    k1     k2     k3
  a1 10.561  7.890 12.174
  a2  7.429  7.338 12.326
```

(b) Factor A: species (factor levels: *A. scabra*, *A. digitalis*
Factor B: salt concentration (factor levels: I, II, III)

It is your decision which of the factors is A and which is B. In the following solutions, we assume that the species is factor A and the salt concentration is factor B.

(c) There are eight replications per factor level combination. The design is therefore balanced.

(d) The design is crossed because each species was investigated in each salt concentration.

(e) The two factors are fixed.

(f) $y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \epsilon_{ijk}$

y_{ijk}: single observation of oxygen uptake

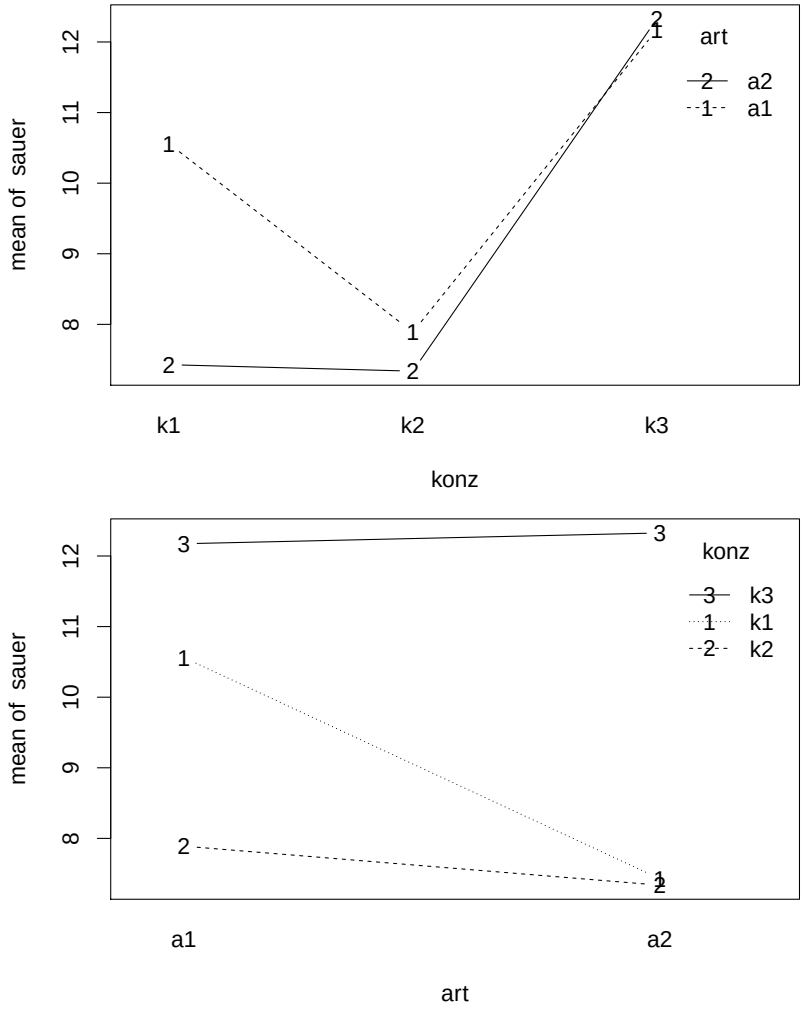
μ: true mean

α_i: fixed effect of the species

β_j: fixed effect of the salt concentration

γ_{ij}: fixed interaction effect between the two crossed factors species and salt concentration

ϵ_{ijk} : residuals



(g) From the interaction plots you might suspect that there are interactions between species and salt concentration because the lines cross/are not parallel.

```
Analysis of Variance Table

Response: sauer
          Df Sum Sq Mean Sq F value    Pr(>F)
A          1  16.64  16.638  1.7404 0.1942381
K          2 181.32  90.661  9.4833 0.0003993 ***
A:K        2  23.93  11.963  1.2514 0.2965616
Residual 42 401.52   9.560
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The ANOVA does not show a significant interaction. The species has no significant influence on the oxygen uptake whereas the salt concentration does.

(h)
```
> lsd.k
[1] 2.206091
> hsd.k
[1] 2.655831
```

The oxygen uptake under the highest salt concentration differs significantly from that under the other two concentrations. The result is the same, no matter whether the CER or the FER is 5%. It depends on the goal of the study which error rate you use. If your study was exploratory, you might want to use the LSD to encourage further investigations. Those might then confirm that the difference you found was a false-positive. If you want to be really sure that there is no false positive, use the HSD.

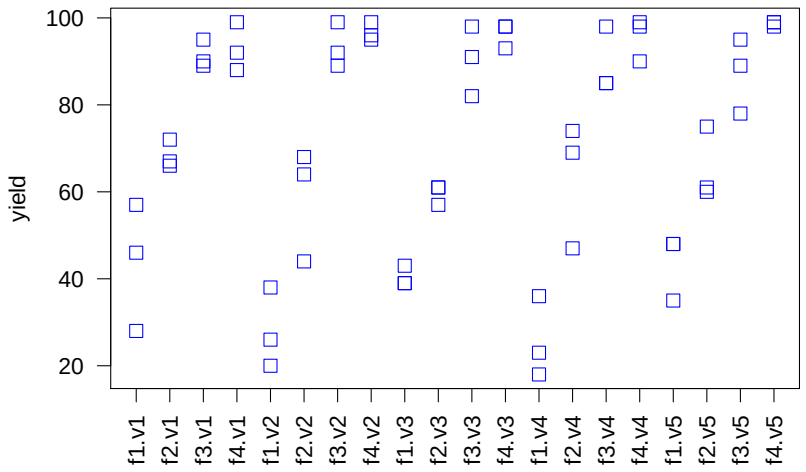
(i)
```
> lsd.k
[1] 1.838647
```

(j)
```
> hsd.k
[1] 2.306321
```

The HSD is larger because it is more conservative ("careful"): You assume two values to be significantly different only when the have a larger difference. This means that it has a lower risk for type I errors but also a lower power (*i.e.*, an increased risk for a type II error).

2 Fertilizer

(a) Boxplots or stripcharts can be used for the display of fertilizers and varieties. Only stripcharts can be used for the factor level combinations because there are only three replications for each combination.



The argument `las=2` within the `stripchart()` command rotates the labels for the tick marks on the x axis by 90 degress.

The yield appears to decrease from fertilizer 1 through 4. The varieties seem to have approximately the same yield.

(b) Treatment factor A: variety (levels: v1, v2, v3, v4, v5) Treatment factor B: fertilizer (levels: f1, f2, f3, f4)

(c) The design is crossed. Each variety is investigated with each fertilizer.

(d) The factor "fertilizer" is fixed because we are interested in the effect of the specific fertilizers in the experiment. The factor "variety" is random because in order to "To draw conclusions on the wheat yield in general, five varieties were randomly selected from all varieties grown in the area under investigation".

No need to worry: You will be told which factor is supposed to be fixed and which is random if such a design is used in the exam. However, you might be asked to give an explanation for the choice of fixed and random.

(e)
```
Analysis of Variance Table

Response: yield
          Df Sum Sq Mean Sq  F value    Pr(>F)
F          3  34061 11353.5 150.0632 8.882e-10 ***
V          4    314    78.6   1.3050    0.2846
F:V       12    908    75.7   1.2568    0.2812
Residual 40   2408    60.2
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There are no significant interactions between fertilizer and variety. At least one of the fertilizers results is a significantly different yield. There is no significant influence of the variety on the yield.

(f) Do the varieties react differently on the fertilizers? Do the fertilizers have different effects, depending on which variety is used?

(g)
```
$f.versus
    F-ratio versus
F   "F:V"
V   "Residual"
F:V "Residual"
```

$$F_{\mathrm{F}} = \frac{\mathrm{MS_F}}{\mathrm{MS_{F:V}}} = \frac{11353.5}{75.7} = 150.0632$$

$$F_{\mathrm{V}} = \frac{\mathrm{MS_V}}{\mathrm{MS_e}} = \frac{78.6}{60.2} = 1.3050$$

$$F_{\mathrm{F}} = \frac{\mathrm{MS_F}}{\mathrm{MS_e}} = \frac{75.7}{60.2} = 1.2568$$

Differences due to rounding errors.

(h) Estimate the mean yields for the fertilizers.

```
Tables of means
Grand mean

71.43333

 F
F
```

```
   f1    f2    f3    f4
36.27 63.07 90.33 96.07
```

(i) Which fertilizers result in significantly different yields? Use a comparison-wise error rate of 5%.

```
> lsd.f
[1] 6.920307
```

Note that you have to take the standard error and the degrees of freedom from `snk.test()`.

Fertilizers 1, 2, and 3 are significantly different from each other. Fertilizer 4 is significantly different from fertilizers 1 and 2 but not from fertilizer 3.

(j) Compare the fertilizers with a Tukey test (family-wise type 1 error rate of 10%) and display the results in a CLD.

```
 contrast   estimate       SE df t.ratio p.value
 f1 - f2  -26.800000 2.833137 40  -9.459  <.0001
 f1 - f3  -54.066667 2.833137 40 -19.084  <.0001
 f1 - f4  -59.800000 2.833137 40 -21.107  <.0001
 f2 - f3  -27.266667 2.833137 40  -9.624  <.0001
 f2 - f4  -33.000000 2.833137 40 -11.648  <.0001
 f3 - f4   -5.733333 2.833137 40  -2.024  0.1965

Results are averaged over the levels of: V
P value adjustment: tukey method for comparing a family of 4 estimates
```

```
 F    emmean       SE df lower.CL  upper.CL .group
 f1 36.26667 2.003331 40 31.04194  41.49139  1
 f2 63.06667 2.003331 40 57.84194  68.29139   2
 f3 90.33333 2.003331 40 85.10861  95.55806    3
 f4 96.06667 2.003331 40 90.84194 101.29139    3

Results are averaged over the levels of: V
Confidence level used: 0.95
Conf-level adjustment: sidak method for 4 estimates
P value adjustment: tukey method for comparing a family of 4 estimates
significance level used: alpha = 0.05
```

The result is the same as before: Fertilizers 1, 2, and 3 are significantly different from each other. Fertilizer 4 is significantly different from fertilizers 1 and 2 but not from fertilizer 3.

Remember to load the required packages!

(k) What is the least significant difference (LSD) between two fertilizers for a comparison-wise type 1 error rate of 10%?

```
> lsd.f
[1] 5.66087
```

(l) Estimate the Honestly Significant Difference (HSD) between the fertilizers for a family-wise type 1 error rate of 10%. Compare the HSD and the LSD values and the result.

```
> hsd.f
```

```
[1] 8.131749
```

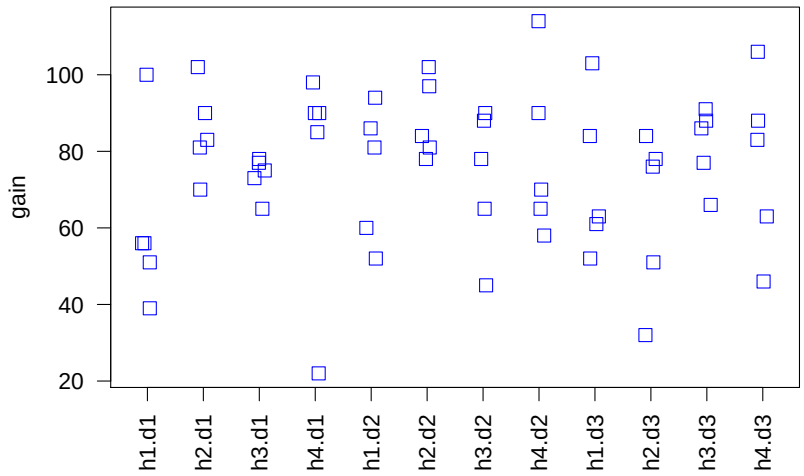
Again, the HSD is larger than the LSD. See the limpets experiment for an explanation. The LSD with a CER of 10% detects an additional significant difference between fertilizers 3 and 4.

(m) Why do we not compare the yield of the different varieties?

The varieties are a random factor. Means are only compared for fixed factors.

3 Chicken

(a) Boxplots or stripcharts can be used for the display of diets and houses. Only stripcharts can be used for the factor level combinations because there are only five replications for each combination.



The argument `method=''jitter''` in the `stripchart()` command randomly jitters the data points left or right so that overlapping points are easier to distinguish.

The plots show no apparent differences in the weight gains between diets and chicken houses.

(b) Treatment factor A: diet (levels: 1, 2, 3) Treatment factor B: chicken house (levels: 1, ..., 12)

(c) The design is hierarchical. The chicken houses are nested within the diets. Not every diet is investigated in each chicken house.

(d) The factor “diet” is fixed because we are interested in the effect of the specific diets in the experiment. The factor “chicken house” is random because t“welve chicken houses (Hühnerställe) were randomly chosen”.

(e) Is the weight gain different for the different diets? Does the chicken house have an influence on the weight gain?

```
Analysis of Variance Table
```

```
Response: gain
          Df  Sum Sq Mean Sq F value Pr(>F)
D          2   323.6  161.82  0.4646 0.6427
D:H        9  3134.9  348.33  0.9260 0.5114
Residual 48 18055.6  376.16
```

Neither the diet nor the chicken house have a significant effect on the weight gain.

4 Data input for the limpets example

There are several possibilites to enter data in R. We start with entering data by hand. First, you have to decide how you want to enter your data: by row, by column or by data blocks.

For entering data row-wise, you start at the upper left corner and then move on to the right. For entering data column-wise, you go through the columns from top to bottom. Another possibility is to enter the data by block: Start with the first block with all the data from concentration I and *A. scabra*, then move on to the next block etc. Here you also have to decide whether you want to enter the blocks row- or column-wise. By the way: "block" has nothing to do with experimental design here!

An example for entering the data row-wise:

```
uptake <- c( 7.16,  8.26,   6.14,  6.14,
             6.78, 14.00,   3.86, 10.00,
            13.60, 16.10,  10.40, 11.60,
             8.93,  9.66,   5.49,  5.80,

             5.20, 13.20,   4.47,  4.95,
             5.20,  8.39,   9.90,  6.49,
             7.18, 10.40,   5.75,  5.44,
             6.37,  7.18,  11.80,  9.90,

            11.11, 10.50,   9.63, 14.50,
             9.74, 14.60,   6.38, 10.20,
            18.80, 11.10,  13.40, 17.70,
             9.74, 11.80,  14.50, 12.30)
```

The advantage is that the data in R look just like the table. However, entering data is a matter of taste. Do not forget to use the point as a decimal separator; no commas!

We then have to enter the factor levels that correspond to our measurements. In our experiment, we have two factors: salt concentration and species.

One way is to simply enter the data in the same order as the values for the oxygen uptake.

```
conc <-   c( "I",    "I",        "I",    "I",
             "I",    "I",        "I",    "I",
             "I",    "I",        "I",    "I",
             "I",    "I",        "I",    "I",

             "II",   "II",       "II",   "II",
             "II",   "II",       "II",   "II",
             "II",   "II",       "II",   "II",
             "II",   "II",       "II",   "II",
```

```
             "III",  "III",      "III",  "III",
             "III",  "III",      "III",  "III",
             "III",  "III",      "III",  "III",
             "III",  "III",      "III",  "III")

spec <-   c( "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",

             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",

             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi")
```

Do not forget the quotation marks!

So far, `conc` and `spec` are just text vectors that we have to convert into factors with

```
conc <- as.factor(conc)
spec <- as.factor(spec)
```

However, entering the data point by point is tedious. `gl()` can be helpful here. Basically, you enter the following things:
`gl(n, length, labels, k)`
with `n` = number of factor levels, `length` = size of the data set (total number of data points), `labels` = names of the factor levels, `k` = how often each level is supposed to be repeated before moving to the next one.

`gl()` repeats the same sequence until the specified length is reached. `n` and `k` have nothing to do with the n and k from our equations from the lecture slides!
Also note that conversion to a factor with `as.factor()` is *not* required because `gl()` directly creates factor variables.

For our data (entered row-wise), the commands look like this:

```
spec <- gl(n=2 , length = 48, labels = c("scab", "digi"), k=2 )
conc <- gl(n=3 , length = 48, labels = c("I", "II", "III"), k=16 )
```

`spec` consists of 2 species, the data set has 48 data points, the factor levels are supposed to be "scab" and "digi" and we want to have "scab" twice, then "digi" twice, then "scab" twice, etc.

We repeat this for `conc` with 3 factor levels. Each level is supposed to be repeated 16 times before the next one starts.

Note that this only works for balanced data sets! For unbalanced ones, you have to enter the factor levels by hand, with the help of `rep()`.

Now combine the data into a data frame and have a look at the result:

```
limpets <- data.frame(uptake, spec, conc)
str(limpets)
limpets
```

```
> limpets <- data.frame(uptake, spec, conc)
> str(limpets)
'data.frame':        48 obs. of  3 variables:
 $ uptake: num  7.16 8.26 6.14 6.14 6.78 14 3.86 10 13.6 16.1 ...
 $ spec  : Factor w/ 2 levels "scab","digi": 1 1 2 2 1 1 2 2 1 1 ...
 $ conc  : Factor w/ 3 levels "I","II","III": 1 1 1 1 1 1 1 1 1 1 ...
> limpets
   uptake spec conc
1    7.16 scab    I
2    8.26 scab    I
3    6.14 digi    I
4    6.14 digi    I
5    6.78 scab    I
6   14.00 scab    I
...
45   9.74 scab  III
46  11.80 scab  III
47  14.50 digi  III
48  12.30 digi  III
```

Side note: `as.factor()` and `gl()` behave differently when it comes to the order of the factor levels:

```
spec <-   c( "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",

             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",

             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi",
             "scab",  "scab",    "digi",  "digi")
spec <- as.factor(spec)
```

```
> spec
 [1] scab scab digi digi scab scab digi digi scab scab digi digi scab
[14] scab digi digi scab scab digi digi scab scab digi digi scab scab
[27] digi digi scab scab digi digi scab scab digi digi scab scab digi
[40] digi scab scab digi digi scab scab digi digi
Levels: digi scab
```

```
spec <- gl(n=2 , length = 48, labels = c("scab", "digi"), k=2 )
```

```
> spec
 [1] scab scab digi digi scab scab digi digi scab scab digi digi scab
```

```
[14] scab digi digi scab scab digi digi scab scab digi digi scab scab
[27] digi digi scab scab digi digi scab scab digi digi scab scab digi
[40] digi scab scab digi digi scab scab digi digi
Levels: scab digi
```

You can see that `as.factor()` orders the factor levels from the former character vector alphanumerically whereas `gl()` keeps the factor levels in the order in which they are entered. This is important for *e.g.* for the linear contrasts.

Experimental Designs

Matthias Frisch

This exercise demonstrates how to apply statistical methods that you already know, like the ANOVA and multiple comparisons, to standard experimental designs.

```
library("lme4"); library("emmeans");
library("multcomp"); library("multcompView")
options ( contrasts = c("contr.sum","contr.sum") )
```

1 Randomized complete block design: Cotton

Recall: Randomization is a core principle of experimental designs and is done to eliminate unstructered environmental variation. Sometimes, however, we know in advance that there is a systematic factor influencing the results of our experiment. If this factor is spatial and has an influence in one direction only (e.g. the soil type on a field), a randomized complete block design is a good way to deal with the influence of this factor. The idea is that the entries of the experiment are grouped into blocks and that the block effect is later estimated in order to eliminate it from the residual error. Since the F-tests in an analysis with fixed factors are done with the MSE, it increases the power of the analysis if we manage to reduce the MSE.

```
cotton <- read.table ("v14-u41-01.csv", header=T, sep=";", dec=",",
                      stringsAsFactors = T)
str(cotton)
```

this command helps strings to change as a factor

```
> str(cotton)
'data.frame':        15 obs. of  3 variables:
 $ Blk     : Factor w/ 3 levels "b1","b2","b3": 1 2 3 1 2 3 1 2 3 1 ...
 $ Fert    : Factor w/ 5 levels "f1","f2","f3",..: 1 1 1 2 2 2 3 3 3 4 ...
 $ Strength: num  7.62 8 7.93 8.14 8.15 7.87 7.76 7.73 7.74 7.17 ...
```

There is one treatment factor (fertilizer) with five factor levels. The design is balanced: All treatments, *i.e.* all five fertilizers, occur equally often. The design is complete: Each treatment occurs in each block.

Analysis of variance

```
boxplot( Strength ~ Fert , data=cotton )
m1 <- aov ( Strength ~ Blk + Fert, data=cotton )
summary ( m1 )
```

The analysis of a RCBD looks pretty similar to a two-factor ANOVA. However, there are no interactions between the blocks and the main factor. In the classical analysis, both the main factor and the block are treated as fixed.

Statistical model: $y_{ij} = \mu + \beta_i + \vartheta_j + \epsilon_{ij}$. β_i is the effect of block i and ϑ_j is the effect of factor level j of the treatment factor. The fertilizer is the treatment factor in our experiment.

```
> summary ( m1 )
            Df Sum Sq Mean Sq F value Pr(>F)
Blk          2 0.0971 0.04856   1.112 0.3750
Fert         4 0.7324 0.18311   4.192 0.0404 *
```

```
Residuals    8 0.3495 0.04369
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

The different fertilizers seem to result in very different yields and indeed, the fertilizer effect is significant in the ANOVA whereas the block effect is not.

Means

```
model.tables ( m1, "means" )
```

model.tables (ml, "effects")

```
Fert
   f1    f2    f3    f4    f5
7.850 8.053 7.743 7.513 7.450
```

Means are estimated for the blocks and for the fertilizers. We are not interested in the means of the blocks but only in the means of the fertilizers.

78=7.7+(-0.012)+(-0.021)+resediual error

Least significant differences

```
va  <- anova ( m1 )              # Analysis of variance
b <- 3; t <- 5                   # Number of blocks and treatments
sed <- sqrt ( 2 * va[3,3] / b)   # Standard error of a difference of means
dfe <- va[3,1]                   # Degrees of freedom residuals
lsd <- qt(0.975,dfe) * sed                  # Least significant difference
hsd <- qtukey(0.95,t,dfe) * sed / sqrt(2)   # Honestly significant diff.
sed; lsd; hsd
```

alfa-0.05/2

alfa-0.05

```
> sed; lsd; hsd
[1] 0.1706556
[1] 0.3935325
[1] 0.5895724
```

Pairwise comparisons, unadjusted

```
e.F <- emmeans ( m1, ~Fert )
cld ( e.F, adjust="none" )
test( contrast (e.F , "pairwise"), adjust="none")
```

```
> cld ( e.F, adjust="none" )
 Fert emmean    SE df lower.CL upper.CL .group
 f5     7.45 0.121  8     7.17     7.73  1
 f4     7.51 0.121  8     7.24     7.79  12
 f3     7.74 0.121  8     7.47     8.02  123
 f1     7.85 0.121  8     7.57     8.13   23
 f2     8.05 0.121  8     7.78     8.33    3
```

```
Results are averaged over the levels of: Blk
Confidence level used: 0.95
significance level used: alpha = 0.05

> test( contrast (e.F , "pairwise"), adjust="none")
 contrast estimate    SE df t.ratio p.value
 f1 - f2   -0.2033 0.171  8 -1.191  0.2676
 f1 - f3    0.1067 0.171  8  0.625  0.5494
 f1 - f4    0.3367 0.171  8  1.973  0.0840
 f1 - f5    0.4000 0.171  8  2.344  0.0471
 f2 - f3    0.3100 0.171  8  1.817  0.1068
 f2 - f4    0.5400 0.171  8  3.164  0.0133
 ...

Results are averaged over the levels of: Blk
```

Pairwise comparisons, adjusted

```
cld ( e.F, adjust="tukey" )
test( contrast (e.F , "pairwise"), adjust="tukey")
```

```
> cld ( e.F, adjust="tukey" )
 Fert emmean    SE df lower.CL upper.CL .group
 f5     7.45 0.121  8     7.05     7.85  1
 f4     7.51 0.121  8     7.11     7.92  12
 f3     7.74 0.121  8     7.34     8.15  12
 f1     7.85 0.121  8     7.45     8.25  12
 f2     8.05 0.121  8     7.65     8.46   2

Results are averaged over the levels of: Blk
Confidence level used: 0.95
Conf-level adjustment: sidak method for 5 estimates
P value adjustment: tukey method for comparing a family of 5 estimates
significance level used: alpha = 0.05

> test( contrast (e.F , "pairwise"), adjust="tukey")
 contrast estimate    SE df t.ratio p.value
 f1 - f2   -0.2033 0.171  8 -1.191  0.7565
 f1 - f3    0.1067 0.171  8  0.625  0.9667
 f1 - f4    0.3367 0.171  8  1.973  0.3563
 f1 - f5    0.4000 0.171  8  2.344  0.2246
 f2 - f3    0.3100 0.171  8  1.817  0.4263
 f2 - f4    0.5400 0.171  8  3.164  0.0743
 ...

Results are averaged over the levels of: Blk
P value adjustment: tukey method for comparing a family of 5 estimates
```

The adjusted pairwise comparisons result in fewer significant differences.

It is also possible to choose a different significance level for the CLD:

```
cld ( e.F, adjust="tukey", alpha=0.1 )
```

```
> cld ( e.F, adjust="tukey", alpha=0.1 ) # changing the significance leve
 Fert   emmean        SE df lower.CL upper.CL .group
 f5   7.450000 0.1206717  8 7.046758 7.853242  1
 f4   7.513333 0.1206717  8 7.110092 7.916575  1
 f3   7.743333 0.1206717  8 7.340092 8.146575  12
 f1   7.850000 0.1206717  8 7.446758 8.253242  12
 f2   8.053333 0.1206717  8 7.650092 8.456575   2

Results are averaged over the levels of: Blk
Confidence level used: 0.95
Conf-level adjustment: sidak method for 5 estimates
P value adjustment: tukey method for comparing a family of 5 estimates
significance level used: alpha = 0.1
```

2 Latin square: Sugar beets

If there is structured environmental variation in two directions, we also do blocking in two directions. In a Latin Square, each treatment occurs in each row and each column. The row and column blocks are therefore complete. The design is balanced.

```
sugar <- read.table ( "v14-u41-02.csv", header=T, sep=";", dec=",",
                      stringsAsFactors = T)
str(sugar)
sugar
```

head(sugar)

```
> str(sugar)
'data.frame':        36 obs. of  4 variables:
 $ Row    : Factor w/ 6 levels "r1","r2","r3",..: 1 1 1 1 1 1 2 2 2 2 ...
 $ Col    : Factor w/ 6 levels "c1","c2","c3",..: 1 2 3 4 5 6 1 2 3 4 ...
 $ Variety: Factor w/ 6 levels "v1","v2","v3",..: 3 6 5 1 2 4 6 2 4 3 ...
 $ Yield  : num  16.2 17 18.1 16.6 17.7 16.3 16 15.3 16 17.1 ...
> sugar
   Row Col Variety Yield
1   r1  c1      v3  16.2
2   r1  c2      v6  17.0
3   r1  c3      v5  18.1
4   r1  c4      v1  16.6
5   r1  c5      v2  17.7
...
```

There is one treatment factor (variety) with six factor levels.

Analysis of variance

```
boxplot ( Yield ~ Variety, data=sugar )  range=0
m2 <- aov ( Yield ~ Row + Col + Variety, data=sugar )
summary ( m2 )
```

```
> summary ( m2 )
            Df Sum Sq Mean Sq F value   Pr(>F)
Row          5  2.292  0.4584   2.545   0.0614 .
Col          5  1.256  0.2511   1.394   0.2687
```

```
Variety      5 12.622  2.5244  14.016 5.89e-06 ***
Residuals   20  3.602  0.1801
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The variety effect is significant, row and column effects are not.

but we include them to reduce our experimental error. they are called blocking factors

Model: $y_{ijk} = \mu + \delta_i + \gamma_j + \vartheta_k + \epsilon_{ijk}$. δ_i is the effect of row i, γ_j is the effect of column j, and ϑ_k is the effect of factor level k of the treatment factor.

and we will use Variety data for LSD

Least significant differences

```
r   <- 6
va  <- anova ( m2 )
sed <- sqrt ( 2* va[4,3] / r)
dfe <- va[4,1]
lsd <- qt(0.975,dfe) * sed
hsd <- qtukey(0.95,r,dfe) * sed / sqrt(2)
sed; lsd; hsd
```

model.tables(m2, "means")

r is the number of treatments (and also the number of rows and columns - why?).

```
> sed; lsd; hsd
[1] 0.2450246
[1] 0.5111123
[1] 0.7701752
```

Means and pairwise comparisons

```
e.V <- emmeans(m2, ~Variety)
cld ( e.V, adjust="none" )
test( contrast ( e.V, "pairwise"), adjust="none" )
```

```
> e.V <- emmeans(m2, ~Variety)
> cld ( e.V, adjust="none" )
 Variety emmean    SE df lower.CL upper.CL .group
 v4        16.2 0.173 20     15.9     16.6  1
 v3        16.4 0.173 20     16.1     16.8  1
 v1        16.4 0.173 20     16.1     16.8  1
 v6        16.5 0.173 20     16.2     16.9  12
 v2        17.0 0.173 20     16.7     17.4   2
 v5        18.0 0.173 20     17.6     18.3    3

> test( contrast ( e.V, "pairwise"), adjust="none" )
 contrast estimate    SE df t.ratio p.value
 v1 - v2    -0.583 0.245 20 -2.381  0.0273
 v1 - v3     0.000 0.245 20  0.000  1.0000
 v1 - v4     0.200 0.245 20  0.816  0.4240
 ...
```

3 Simple lattice: Soy beans

The simple lattice (Zweisatzgitter) is a partially balanced, incomplete design. It contains the first two replications of a balanced lattice, meaning that only some treatment pairs occur together in the same block.

```
soy <- read.table ( "v14-u41-03.csv", header=T, sep=";", dec=",",
                    stringsAsFactors = T)
str(soy)
```

head(soy)

```
> str(soy)
'data.frame':        50 obs. of  5 variables:
 $ Rep    : Factor w/ 2 levels "r1","r2": 1 2 1 2 1 2 1 2 1 2 ...
 $ Block  : Factor w/ 10 levels "b1","b10","b2",..: 1 7 1 8 1 9 1 10 1 2 ...
 $ Plot   : Factor w/ 5 levels "p1","p2","p3",..: 1 1 2 1 3 1 4 1 5 1 ...
 $ Variety: Factor w/ 25 levels "v01","v02","v03",..: 1 1 2 2 3 3 4 4 5 5 ...
 $ Yield  : int  6 24 7 21 5 16 8 17 6 15 ...
```

Analysis of variance

```
soy$Block.in.Rep <- soy$Block:soy$Rep   # Sequence of model terms!!
m3 <- aov ( Yield ~ Rep + Block.in.Rep + Variety , data=soy)
summary(m3)
```

The first line creates a now column in the data frame which helps to identify each block correctly: Block 1 in rep 1 is not the same as block 1 in rep 2.

```
> summary(m3)
             Df Sum Sq Mean Sq F value  Pr(>F)
Rep           1  212.2  212.18  15.539 0.00117 **
Block.in.Rep  8  350.0   43.75   3.204 0.02264 *
Variety      24  711.1   29.63   2.170 0.05640 .
Residuals    16  218.5   13.66
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Apparently, the replication and the block have a significant influence on the yield.

Model: $y_{hij} = \mu + \rho_h + \beta_{hj} + \vartheta_i + \epsilon_{hij}$. ρ_h is the effect of replication h, β_{hj} is the effect of the block j nested in replication h, and ϑ_i is the effect of factor level i.

Least significant differences

```
r <- 2; k <- 5
va <- anova ( m3 )
msb <- va[2,3]; mse <- va[4,3]; dfe <- va[4,1]
w <- (msb-mse) / (k*(r-1)*msb)
sed.l1 <- sqrt( 2*mse/r*(1+(r-1)*w)); lsd.l1 <- qt(0.975,dfe) * sed.l1
sed.l0 <- sqrt( 2*mse/r*(1+r*w))    ; lsd.l0 <- qt(0.975,dfe) * sed.l0
```

For the estimation of the SED, we need the value `w` (see lecture slide `v14-04-172`).

```
> sed.l1; sed.l0
[1] 3.941271
[1] 4.172797

> lsd.l1; lsd.l0
[1] 8.355121
[1] 8.845934
```

We estimate two SEDs because we have so-called first and second associates. First associates occurred together in one block in the trial ($\lambda = 1$), second associates did not ($\lambda = 0$). Consequently, we can estimate the differences between first associates with greater precision than between second associates. This in turn means that the entry means of second associates have to show a greater difference before we consider them to be significantly different.

Look at the field plan to find out which value you need for the comparisons.

Calculation of adjusted entry means using a mixed model

"Adjusted entry means" (also: "adjusted treatment means") mean that the observed means are adjusted by the effect for the replication and the block. We can estimate these means by using the equation from the lecture slides or we can use the function `lmer()` that fits a linear mixed model on the data. The output is then handed over to `emmeans()` which will return the adjusted entry means.

```
m3a <- lmer(Yield ~   (1|Rep) + (1|Block.in.Rep) + Variety , data=soy)
emmeans(m3a, ~Variety)
```

```
> m3a <- lmer(Yield ~   (1|Rep) + (1|Block.in.Rep) + Variety , data=soy)
> emmeans(m3a, ~Variety)
 Variety emmean   SE   df lower.CL upper.CL
 v01      19.07 3.59 7.17 10.62688     27.5
 v02      16.97 3.59 7.17  8.53163     25.4
 v03      14.65 3.59 7.17  6.20512     23.1
 v04      14.77 3.59 7.17  6.32755     23.2
 v05      12.85 3.59 7.17  4.40578     21.3
 v06      13.17 3.59 7.17  4.72887     21.6
 v07       9.07 3.59 7.17  0.63362     17.5
 v08       6.75 3.59 7.17 -1.69290     15.2
 v09       8.37 3.59 7.17 -0.07047     16.8
 v10       8.45 3.59 7.17  0.00776     16.9
 ...
```

`(1|...)` means that the respective effect is included as random. Do not worry too much about this yet, we will deal with mixed linear models in the last part of the course.

The term "adjusted entry means" or "adjusted treatment means" is also used for other designs and is very important.

4 Randomized complete block design with two factors: Cowpeas

A randomized complete block design can also have *two* factors. In this case, each treatment combination occurs in all blocks.

```
peas <- read.table ( "v14-u41-04.csv", header=T, sep=";", dec=",",
                     stringsAsFactors = T)
str(peas)
```

```
> str(peas)
'data.frame':        36 obs. of  4 variables:
 $ Variety: Factor w/ 3 levels "v1","v2","v3": 1 1 1 1 1 1 1 1 1 ...
 $ Spacing: Factor w/ 3 levels "s04","s08","s12": 1 1 1 1 2 2 2 2 3 3 ...
 $ Block  : Factor w/ 4 levels "b1","b2","b3",..: 1 2 3 4 1 2 3 4 1 2 ...
 $ Yield  : int  56 45 43 46 60 50 45 48 66 57 ...
```

There are 3 varieties (variety: treatment factor A), 3 spacings (spacing: treatment factor B) and 4 blocks.

Analysis of variance

The interaction between spacing and variety is included here.

```
boxplot ( Yield ~ Spacing:Variety , data=peas )
m4 <- aov ( Yield ~ Block + Spacing + Variety + Spacing:Variety ,
           data=peas)
summary(m4)
```

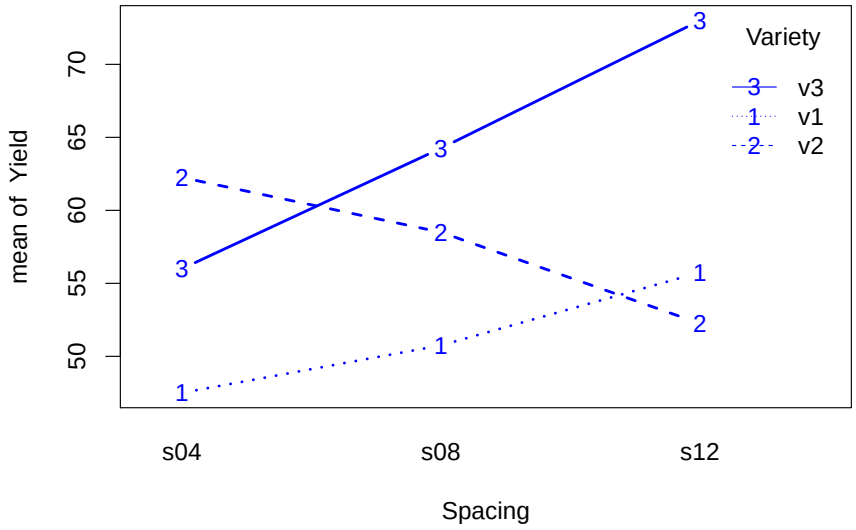
```
> summary(m4)
                Df Sum Sq Mean Sq F value   Pr(>F)
Block            3  255.6    85.2   4.822  0.00912 **
Spacing          2  155.1    77.5   4.387  0.02377 *
Variety          2 1027.4   513.7  29.069 3.87e-07 ***
Spacing:Variety  4  765.4   191.4  10.829 3.68e-05 ***
Residuals       24  424.1    17.7
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All factors, including the interactions, are significant in the analysis.

Model: $y_{hij} = \mu + \rho_h + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{hij}$. ρ_h is the effect of block h, α_i is the effect of factor level i of the treatment factor A, β_j is the effect of factor level j of the treatment factor B, and $(\alpha\beta)_{ij}$ is the effect of the interaction between the levels i and j of the two treatment factors.

Plot interaction

```
with ( peas, interaction.plot ( Spacing, Variety, Yield, type="b"))
```



The interaction plot shows crossing lines, pointing to interactions between variety and spacing.

Least significant differences

```
r <- 4; a <- 3; b <- 3
va <- anova ( m4 )
mse <- va[5,3]; mse
dfe <- va[5,1]; dfe
sed.a <- sqrt( 2*mse/(r*b))
sed.b <- sqrt( 2*mse/(r*a))
sed.ab <- sqrt( 2*mse/(r))
lsd.a <- qt(0.975,dfe) * sed.a
lsd.b <- qt(0.975,dfe) * sed.b
lsd.ab <- qt(0.975,dfe) * sed.ab
```

```
> sed.a; sed.b; sed.ab
[1] 1.716163
[1] 1.716163
[1] 2.972482
> lsd.a; lsd.b; lsd.ab
[1] 3.541987
[1] 3.541987
[1] 6.134901
```

Like in the two-factor ANOVA from the previous chapter, we have to estimate different LSDs for each factor and for the interactions.

`r` is the number of blocks. They are complete so it is also the number of replications. `a` is the number of varieties and `b` is the number of spacings.

Means and pairwise comparisons

Means for each combination of spacing and variety:

```
e.V1 <- emmeans(m4, ~Spacing:Variety)
```

```
> e.V1
 Spacing Variety emmean  SE df lower.CL upper.CL
 s04     v1        47.5 2.1 24     43.2     51.8
 s08     v1        50.8 2.1 24     46.4     55.1
 s12     v1        55.8 2.1 24     51.4     60.1
 s04     v2        62.2 2.1 24     57.9     66.6
 s08     v2        58.5 2.1 24     54.2     62.8
 s12     v2        52.2 2.1 24     47.9     56.6
 s04     v3        56.0 2.1 24     51.7     60.3
 s08     v3        64.2 2.1 24     59.9     68.6
 s12     v3        73.0 2.1 24     68.7     77.3
```

The results are averaged over the levels of block.

Means of the spacings within the varieties:

```
e.V2 <- emmeans(m4, ~Spacing|Variety)
cld ( e.V2, adjust="none", sort=F )
test( contrast ( e.V2, "pairwise"), adjust="none" )
```

```
> cld ( e.V2, adjust="none",sort=F )
Variety = v1:
 Spacing emmean  SE df lower.CL upper.CL .group
 s04       47.5 2.1 24     43.2     51.8  1
 s08       50.8 2.1 24     46.4     55.1  12
 s12       55.8 2.1 24     51.4     60.1   2

Variety = v2:
 Spacing emmean  SE df lower.CL upper.CL .group
 s04       62.2 2.1 24     57.9     66.6  1
 s08       58.5 2.1 24     54.2     62.8  1
 s12       52.2 2.1 24     47.9     56.6   2

Variety = v3:
 Spacing emmean  SE df lower.CL upper.CL .group
 s04       56.0 2.1 24     51.7     60.3  1
 s08       64.2 2.1 24     59.9     68.6   2
 s12       73.0 2.1 24     68.7     77.3    3

Results are averaged over the levels of: Block
Confidence level used: 0.95
significance level used: alpha = 0.05

> test( contrast ( e.V3, "pairwise"), adjust="none" )
Spacing = s04:
 contrast estimate   SE df t.ratio p.value
 v1 - v2    -14.75 2.97 24 -4.962  <.0001
 v1 - v3     -8.50 2.97 24 -2.860  0.0086
 v2 - v3      6.25 2.97 24  2.103  0.0462

Spacing = s08:
 contrast estimate   SE df t.ratio p.value
```

```
 v1 - v2     -7.75 2.97 24 -2.607  0.0154
 v1 - v3    -13.50 2.97 24 -4.542  0.0001
 v2 - v3     -5.75 2.97 24 -1.934  0.0649

Spacing = s12:
 contrast estimate   SE df t.ratio p.value
 v1 - v2      3.50 2.97 24  1.177  0.2506
 v1 - v3    -17.25 2.97 24 -5.803  <.0001
 v2 - v3    -20.75 2.97 24 -6.981  <.0001

Results are averaged over the levels of: Block
```

Comparisons are made between the different spacings within the levels of variety.

Means of the varieties within the spacings:

```
e.V3 <- emmeans(m4, ~Variety|Spacing)
cld ( e.V3, adjust="none",sort=F )
test( contrast ( e.V3, "pairwise"), adjust="none" )
```

```
> cld ( e.V3, adjust="none",sort=F )
Spacing = s04:
 Variety emmean  SE df lower.CL upper.CL .group
 v1        47.5 2.1 24     43.2     51.8  1
 v2        62.2 2.1 24     57.9     66.6   2
 v3        56.0 2.1 24     51.7     60.3    3
 ...

> test( contrast ( e.V3, "pairwise"), adjust="none" )
Spacing = s04:
 contrast estimate       SE df t.ratio p.value
 v1 - v2    -14.75 2.972482 24  -4.962  <.0001
 v1 - v3     -8.50 2.972482 24  -2.860  0.0086
 v2 - v3      6.25 2.972482 24   2.103  0.0462
 ...
```

Comparisons are made between the different varieties within the levels of spacing.

5 Split plot: Alfalfa

Split plot designs (Spaltanlagen) can be used if one of the two factors under investigation is restricted to application to larger plots (`e.g.` tillage) whereas the other factor can be applied to smaller plots and/or if there is more precision needed for one factor than for the other.

```
alf <- read.table ( "v14-u41-05.csv", header=T, sep=";", dec=",",
                    stringsAsFactors = T)
str(alf)
```

```
> str(alf)
'data.frame':        72 obs. of  4 variables:
 $ Variety: Factor w/ 3 levels "Cossack","Ladack",..: 2 2 2 2 2 2 2 2 2 ...
 $ Date   : Factor w/ 4 levels "A","B","C","D": 1 1 1 1 1 1 2 2 2 2 ...
 $ Block  : Factor w/ 6 levels "b1","b2","b3",..: 1 2 3 4 5 6 1 2 3 4 ...
 $ Yield  : num  2.17 1.88 1.62 2.34 1.58 1.66 1.58 1.26 1.22 1.59 ...
```

There are 3 varieties, 4 dates for final cutting and 6 blocks. The main plots (Großparzellen) are the varieties and the sub-plots (Kleinparzellen) are the cutting dates. Main plots as well as sub-plots occur within the blocks!

Research question: Do the different varieties have different yields when harvested at different final cutting dates?

Analysis of variance

Variety and cutting date are included in the analysis as fixed factors. Their interaction is fixed, too.

```
m5 <- aov(Yield ~ Variety + Date + Date:Variety
                  + Error(Block/Variety), data=alf)
summary (m5)
```

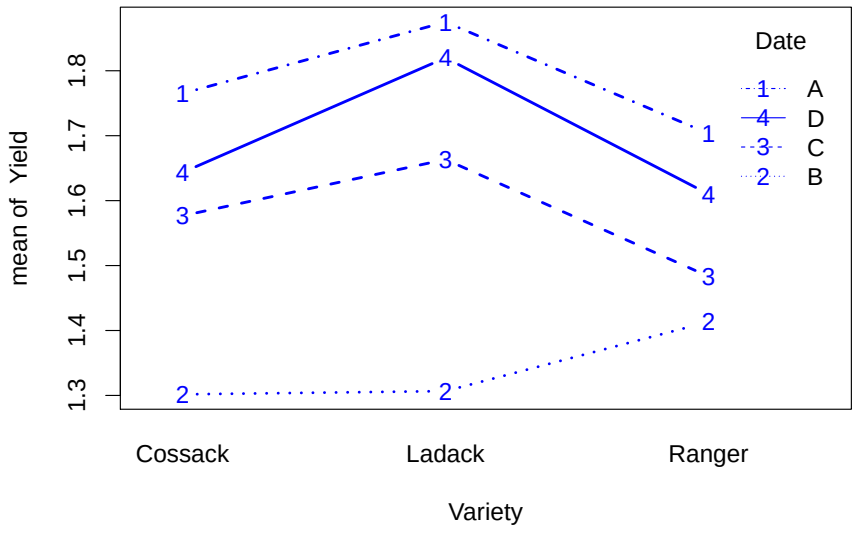
```
> summary (m5)

Error: Block
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals  5   4.15    0.83

Error: Block:Variety
          Df Sum Sq Mean Sq F value Pr(>F)
Variety    2  0.178 0.08901   0.653  0.541
Residuals 10  1.362 0.13623

Error: Within
             Df Sum Sq Mean Sq F value   Pr(>F)
Date          3 1.9625  0.6542  23.390 2.83e-09 ***
Variety:Date  6 0.2106  0.0351   1.255    0.297
Residuals    45 1.2585  0.0280
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The split plot is a balanced design. For this situation, `Error()` within a formula allows the specification of random effects. The block is random here.

Model: $y_{hij} = \mu + \rho_h + \alpha_i + me_{hi} + \beta_j + (\alpha\beta)_{ij} + se_{hij}$. ρ_h is the effect of block h, α_i is the effect of level i of the main plot factor, me_{hi} is the main plot error of main plot i within block h, β_j is the effect of level j of the sub-plot factor, $(\alpha\beta)_{ij}$ is the interaction between the levels i and j of the treatment factors, and se_{hij} is the sub-plot error of sub-plot j within main plot j in block h.

Plot interaction

```
with ( alf, interaction.plot (  Variety, Date, Yield, type="b" ))
```



Least significant differences

The degrees of freedom are needed for the LSD. For the interactions, they cannot be taken from the `summary()`-Output but have to be approximated. One possibility is the Satterthwaite Approximation.

```
r <- 6 ; a <- 3 ; b <- 4
ms.me <- summary(m5)[[2]][[1]][2,3]
ms.se <- summary(m5)[[3]][[1]][3,3]
df.me <- summary(m5)[[2]][[1]][2,1]
df.se <- summary(m5)[[3]][[1]][3,1]
df.ia <- ( ms.me + (b-1)*ms.se ) ^2 /    # Satterthwaite Approximation
        ( ms.me ^2 / df.me  +  ((b-1)*ms.se)^2 / df.se )
sed.1 <- sqrt(2*ms.me/r/b); sed.2 <- sqrt(2*ms.se/r/ a)
sed.3 <- sqrt(2*ms.se/r  ); sed.4 <- sqrt(2*(ms.me+(b-1)*ms.se)/r/b)
lsd.1 <- qt(0.975,df.me) * sed.1 ;  lsd.2 <- qt(0.975,df.se) * sed.2
lsd.3 <- qt(0.975,df.se) * sed.3 ;  lsd.4 <- qt(0.975,df.ia) * sed.4
```

```
> sed.1; sed.2; sed.3; sed.4
[1] 0.10655
[1] 0.05574514
[1] 0.09655341
[1] 0.135443
> lsd.1; lsd.2; lsd.3; lsd.4
[1] 0.2374081
[1] 0.1122765
[1] 0.1944686
[1] 0.2794911
```

Use `lsd.1` ($\text{LSD}_{i,i'(\alpha=0.05)}$) for the comparisons between the varieties, `lsd.2` ($\text{LSD}_{j,j'\ (\alpha=0.05)}$) for the comparisons between the cutting dates, `lsd.3` ($\text{LSD}_{ij,ij'\ (\alpha=0.05)}$) for comparisons between cutting dates within *one* variety, and `lsd.4` ($\text{LSD}_{ij,i'j'\ (\alpha=0.05)}$) for comparisons between cutting dates for different levels of variety. Also see lecture slide `v14-04-275`.

Pairwise comparisons

SED estimates from above can be found in the `contrast()` outputs. The standard error in the `cld()` output is the SEM.

```
e.V <- emmeans(m5, ~Variety)
cld ( e.V, adjust="none",sort=F)
test( contrast ( e.V, "pairwise"), adjust="none" )
```

```
> cld ( e.V, adjust="none",sort=F)
 Variety emmean    SE   df lower.CL upper.CL .group
 Cossack   1.57 0.124 8.37     1.29     1.85  1
 Ladack    1.67 0.124 8.37     1.38     1.95  1
 Ranger    1.55 0.124 8.37     1.27     1.84  1

> test( contrast ( e.V, "pairwise"), adjust="none" )
 contrast         estimate    SE df t.ratio p.value
 Cossack - Ladack  -0.0946 0.107 10 -0.888  0.3956
 Cossack - Ranger   0.0192 0.107 10  0.180  0.8608
 Ladack - Ranger    0.1138 0.107 10  1.068  0.3108
```

```
e.D <- emmeans(m5, ~Date)
cld ( e.D, adjust="none",sort=F)
test( contrast ( e.D, "pairwise"), adjust="none" )
```

```
> cld ( e.D, adjust="none",sort=F)
 Date emmean    SE   df lower.CL upper.CL .group
 A      1.78 0.113 6.06     1.51     2.06  1
 B      1.34 0.113 6.06     1.07     1.62   2
 C      1.57 0.113 6.06     1.30     1.85    3
 D      1.69 0.113 6.06     1.42     1.97  1

> test( contrast ( e.D, "pairwise"), adjust="none" )
 contrast estimate     SE df t.ratio p.value
 A - B       0.441 0.0557 45  7.903  <.0001
 A - C       0.207 0.0557 45  3.707  0.0006
 A - D       0.090 0.0557 45  1.614  0.1134
 B - C      -0.234 0.0557 45 -4.196  0.0001
 B - D      -0.351 0.0557 45 -6.289  <.0001
 C - D      -0.117 0.0557 45 -2.093  0.0420
```

```
e.DV <- emmeans(m5, ~Date|Variety)
cld ( e.DV, adjust="none",sort=F )
test( contrast ( e.DV, "pairwise"), adjust="none" )
```

```
> cld ( e.DV, adjust="none",sort=F )
Variety = Cossack:
 Date emmean    SE   df lower.CL upper.CL .group
 A      1.76 0.137 12.5     1.47     2.06  1
 B      1.30 0.137 12.5     1.00     1.60   2
 C      1.58 0.137 12.5     1.28     1.87  1
 D      1.64 0.137 12.5     1.35     1.94  1

Variety = Ladack:
 Date emmean    SE   df lower.CL upper.CL .group
 A      1.88 0.137 12.5     1.58     2.17  1
```

```
 B      1.31 0.137 12.5     1.01     1.60   2
 C      1.66 0.137 12.5     1.37     1.96    3
 D      1.82 0.137 12.5     1.52     2.12  1 3

Variety = Ranger:
 Date emmean    SE   df lower.CL upper.CL .group
 A      1.70 0.137 12.5     1.41     2.00  1
 B      1.41 0.137 12.5     1.12     1.71   2
 C      1.48 0.137 12.5     1.19     1.78   23
 D      1.61 0.137 12.5     1.31     1.91  1 3

> test( contrast ( e.DV, "pairwise"), adjust="none" )
Variety = Cossack:
 contrast estimate     SE df t.ratio p.value
 A - B      0.4633 0.0966 45  4.799  <.0001
 A - C      0.1883 0.0966 45  1.951  0.0574
 A - D      0.1217 0.0966 45  1.260  0.2141
 B - C     -0.2750 0.0966 45 -2.848  0.0066
 B - D     -0.3417 0.0966 45 -3.539  0.0009
 C - D     -0.0667 0.0966 45 -0.690  0.4934

Variety = Ladack:
 contrast estimate     SE df t.ratio p.value
 A - B      0.5683 0.0966 45  5.886  <.0001
 A - C      0.2117 0.0966 45  2.192  0.0336
 A - D      0.0550 0.0966 45  0.570  0.5718
 B - C     -0.3567 0.0966 45 -3.694  0.0006
 B - D     -0.5133 0.0966 45 -5.317  <.0001
 C - D     -0.1567 0.0966 45 -1.623  0.1117

Variety = Ranger:
 contrast estimate     SE df t.ratio p.value
 A - B      0.2900 0.0966 45  3.004  0.0043
 A - C      0.2200 0.0966 45  2.279  0.0275
 A - D      0.0933 0.0966 45  0.967  0.3389
 B - C     -0.0700 0.0966 45 -0.725  0.4722
 B - D     -0.1967 0.0966 45 -2.037  0.0476
 C - D     -0.1267 0.0966 45 -1.312  0.1962
```

```
e.VD <- emmeans(m5, ~Variety|Date)
cld ( e.VD, adjust="none",sort=F )
test( contrast ( e.VD, "pairwise"), adjust="none" )
```

```
> cld ( e.VD, adjust="none",sort=F )
Date = A:
 Variety emmean    SE   df lower.CL upper.CL .group
 Cossack   1.76 0.137 12.5     1.47     2.06  1
 Ladack    1.88 0.137 12.5     1.58     2.17  1
 Ranger    1.70 0.137 12.5     1.41     2.00  1
...

> test( contrast ( e.VD, "pairwise"), adjust="none" )
Date = A:
 contrast         estimate    SE   df t.ratio p.value
 Cossack - Ladack  -0.1100 0.135 24.1 -0.812  0.4247
 Cossack - Ranger   0.0617 0.135 24.1  0.455  0.6530
 Ladack - Ranger    0.1717 0.135 24.1  1.267  0.2171
...
```

6 Exercises

1 Seed treatments

Five seed treatments were tested and the germination failures were counted. (Data: read in the table and use `gl()` to generate the factors.)

	Block				
Seed treatment	1	2	3	4	5
	Failures				
Check	8	10	12	13	11
Arasan	2	6	7	11	5
Spergon	4	10	9	8	10
Semesan	3	5	9	10	6
Fematexx	9	7	5	5	3

(a) What is the design of the experiment?

(b) What is/are the treatment factor(s)? What are the factor levels?

(c) Does the seed treatment have an influence on the germination rates?

(d) Estimate the treatment means.

(e) Estimate the effect ϑ_3 of the third seed treatment (Spergon).

(f) Estimate the least significant difference with the t distribution ($\alpha = 0.05$).

(g) Estimate the least significant difference with the Tukey distribution ($\alpha = 0.05$).

(h) Which seed treatments have significantly different germination rates ($\alpha = 0.05$)?

2 Germs

The germ content of milk was investigated on five farms. (Data: read in the table and use `gl` to generate the factors.)

	Day				
Time	1	2	3	3	5
08:30	A 1.9	B 1.2	C 0.7	D 2.2	E 2.3
10:00	D 2.3	C 2.0	E 0.6	B 2.6	A 2.3
11:30	C 2.1	A 1.5	D 1.7	E 1.1	B 3.0
14:00	B 2.9	E 1.1	A 1.2	C 1.8	D 2.6
15:30	E 1.8	D 2.1	B 2.0	A 2.4	C 2.5

(a) What is/are the treatment factor(s)? What are the factor levels?

(b) The experimental design is a Latin square. How do you know?

(c) Are the blocks complete or incomplete?

(d) Is the germ contamination different for the five farms?

(e) Estimate the treatment means.

(f) The statistical model of a Latin Square is $y_{ijk} = \mu + \delta_i + \gamma_j + \vartheta_k + \epsilon_{ijk}$. What are the values of the variables for observation $y_{ijk} = y_{412} = y_{14:00,1,B} = 2.9$?

(g) Estimate the standard error of the difference of two treatment means.

(h) Estimate the least significant difference with the t distribution ($\alpha = 0.10$).

(i) Estimate the least significant difference with the Tukey distribution ($\alpha = 0.10$).

(j) Which farms have significantly different contamination rates ($\alpha = 0.10$)?

3 Oats / Alpha lattice

The yield [t/ha] of 24 oat varieties was assessed in an α lattice with three replications (John and Williams, 1995). Estimate the adjusted treatment means and the least significant differences. (Data: `v14-u41-08.csv`)

		Block						Block					
Rep.	Plot	1	2	3	4	5	6	1	2	3	4	5	6
		Variety						Yield					
1	1	11	21	23	13	17	6	4.1172	4.6540	4.2323	4.2530	4.7876	4.7085
	2	4	10	14	3	15	12	4.4461	4.1736	4.7572	3.3420	5.0902	5.2560
	3	5	20	16	19	7	24	5.8757	4.0141	4.4906	4.7269	4.1505	4.9577
	4	22	2	18	8	1	9	4.5784	4.3350	3.9737	4.9989	5.1202	3.3986
2	1	8	24	12	5	2	19	3.9926	3.9039	5.3127	5.1202	5.1566	5.3148
	2	20	15	11	9	18	7	3.6056	4.9114	5.1163	4.2955	5.0988	4.6297
	3	14	3	21	10	13	6	4.5294	3.7999	5.3802	4.9057	5.4840	5.1751
	4	4	23	17	1	22	16	4.3599	4.3042	5.0744	5.7161	5.0969	5.3024
3	1	11	2	17	12	21	3	3.9205	4.0510	4.3234	4.1746	4.4130	2.8873
	2	1	15	18	13	22	5	4.6512	4.6783	4.2486	4.7512	4.2397	4.1972
	3	14	9	4	10	16	20	4.3887	3.1407	4.3960	4.0875	4.3852	3.7349
	4	19	8	6	23	24	7	4.5552	3.9821	4.2474	3.8721	3.5655	3.6096

How to read the table: The factor levels for varieties and the yield values are supposed to be layered. *E.g.* Variety 11 in replication 1, block 1, and plot 1 has a yield of 4.1172.

(a) How many replications were there?

(b) How many blocks were there?

(c) What was the block size?

(d) Were the blocks complete or incomplete?

(e) What was/were the treatment factor(s)?

(f) Does the variety have a significant influence on the yield?

(g) Estimate the adjusted treatment means.

(h) Why are the treatment means "adjusted"?

(i) Estimate the least significant differences with the t distribution ($\alpha = 0.05$).

(j) Why are there two LSDs? Which one is larger and why?

(k) Which LSD do you use to compare (i) varieties 1 and 2, and (ii) varieties 1 and 5? What are the results of the comparisons?

4 Oats / Split plot

The yield of three different varieties of oats was assessed for four different levels of nitrogen fertilizer. The experiment was layed out in a split-plot design with varieties as whole plots and fertilizer as subplots. Do the yields differ depending on variety and fertilizer? (Data: `v14-u41-09.csv`)

		Block					
Variety	**Nitrogen**	**1**	**2**	**3**	**4**	**5**	**6**
Victory	**0 cwt**	**111**	**74**	**61**	**62**	**68**	**53**
	0.2 cwt	**130**	**89**	**91**	**90**	**64**	**74**
	0.4 cwt	**157**	**81**	**97**	**100**	**112**	**118**
	0.6 cwt	**174**	**122**	**100**	**116**	**86**	**113**
Golden Rain	**0 cwt**	**117**	**64**	**70**	**80**	**60**	**89**
	0.2 cwt	**114**	**103**	**108**	**82**	**102**	**82**
	0.4 cwt	**161**	**132**	**126**	**94**	**89**	**86**
	0.6 cwt	**141**	**133**	**149**	**126**	**96**	**104**
Marvellous	**0 cwt**	**105**	**70**	**96**	**63**	**89**	**97**
	0.2 cwt	**140**	**89**	**124**	**70**	**129**	**99**
	0.4 cwt	**118**	**104**	**121**	**109**	**132**	**119**
	0.6 cwt	**156**	**117**	**144**	**99**	**124**	**121**

(a) How many blocks were there?

(b) What was the size of the main plots?

(c) What was/were the treatment factor(s)?

(d) Does the variety and nitrogen level have a significant influence on the yield? Are there significant interactions between variety and nitrogen level?

(e) Estimate the adjusted treatment means for the varieties, the nitrogen levels and each combination of both.

(f) Estimate the least significant differences with the t distribution ($\alpha = 0.05$).

(g) Which LSD do you use to compare (i) varieties “Victory” and “Marvellous”, (ii) nitrogen level 0.2 in the varieties “Victory” and “Marvellous”, (iii) nitrogen levels 0.4 cwt and 0.6 cwt in general, and (iv) nitrogen levels 0 cwt and 0.2 cwt within the variety "Golden Rain"? What are the results of the comparisons?

5 Overview over experimental designs

Make an overview over all the experimental designs covered in the course. It could be in the form of a table and contain answers to the following questions:

(a) What is the name of the design and how is it abbreviated?

(b) How many treatment and environmental factors are there?

(c) Is the design balanced or unbalanced?

(d) Is the design complete or incomplete?

(e) What is the statistical model that is used for the analysis? What do the terms in the model stand for?

(f) How is the randomization conducted?

(g) What is the relationship between the number of treatments t, the block size k and the number of replications r?

(h) How is the SED estimated? Are there different SEDs or just one?

(i) How is the design different from other designs? What are its unique features? How do you recognize the design?

(j) What are the relevant lecture slides?

7 Solutions

1 Seed treatments

(a) The experiment is conducted as a randomized complete block design with five blocks.

(b) The experiment has one treatment factor (seed treatment) with five factor levels (Check, Arasan, Spergon, Semesan, Fematexx).

(c)
```
Analysis of Variance Table
          Df Sum Sq Mean Sq F value  Pr(>F)
Blk        4  49.84   12.46  2.3031 0.10320
Seedtrt    4  83.84   20.96  3.8743 0.02189 *
Residuals 16  86.56    5.41
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The seed treatment has a significant influence on the germination rates.

(d)
```
   Check   Arasan  Spergon  Semesan Fematexx
    10.8      6.2      8.2      6.6      5.8
```

Note that you can use either `model.tables()` or `emmeans()` to answer this question.

(e) $\vartheta_3 = 0.68$

(f)
```
> lsd
[1] 3.118495
```

(g) Estimate the least significant difference with the Tukey distribution.

```
> hsd
[1] 4.506828
```

(h) With the LSD: Check/Arasan, Check/Semesan, Check/Fematexx. With the HSD: Check/Arasan, Check/Fematexx.

2 Germs

(a) The "treatment" factor is the farm.

(b) The milk of each farm is sampled once at each time point (row) and once at each day (column).

(c) The blocks are the time points (row) and days (column). They are both complete.

(d)
```
Analysis of Variance Table
          Df Sum Sq Mean Sq F value    Pr(>F)
Row        4 0.6416  0.1604  1.3434 0.3100851
Column     4 5.2536  1.3134 11.0000 0.0005532 ***
Farm       4 2.7456  0.6864  5.7487 0.0080354 **
Residuals 12 1.4328  0.1194
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

At least one farm has a germ contamination that is significantly different from that of the others.

(e)
```
   A    B    C    D    E
1.86 2.34 1.82 2.18 1.38
```

(f) $2.9 = \mu + \delta_4 + \gamma_1 + \vartheta_2 + \epsilon_{412} = 1.916 + 0.004 + 0.284 + 0.424 + 0.272$

(g)
```
> sed
[1] 0.2185406
```

(h)
```
> lsd
[1] 0.3895022
```

(i)
```
> hsd
[1] 0.605973
```

(j) Which farms have significantly different contamination rates ($\alpha = 0.10$)?

With the LSD: A/B, A/E, B/C, B/E, C/E, D/E. With the HSD: B/E, D/E.

3 Oats / Alpha lattice

(a) There were three replications.

(b) There were six blocks within each replication.

(c) The block size was 4.

(d) The blocks were incomplete.

(e) The treatment factor was the variety.

(f)
```
Analysis of Variance Table

Response: Yield
             Df  Sum Sq Mean Sq F value    Pr(>F)
Rep           2  6.1355 3.06774 36.7557 6.593e-09 ***
Block.in.Rep 15  7.6182 0.50788  6.0851 1.150e-05 ***
Variety      23 10.0619 0.43747  5.2415 1.459e-05 ***
Residuals    31  2.5874 0.08346
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

(g)
```
 Variety emmean    SE   df lower.CL upper.CL
 v01        5.11 0.279 6.20     4.43     5.78
 v02        4.48 0.279 6.20     3.80     5.15
 v03        3.50 0.279 6.20     2.82     4.18
 v04        4.49 0.279 6.20     3.81     5.17
 v05        5.04 0.278 6.19     4.36     5.71
 v06        4.54 0.278 6.19     3.86     5.21
 v07        4.11 0.279 6.20     3.43     4.79
 v08        4.53 0.279 6.20     3.85     5.20
 v09        3.50 0.278 6.19     2.83     4.18
 v10        4.37 0.279 6.20     3.70     5.05
 v11        4.28 0.279 6.20     3.61     4.96
 v12        4.76 0.279 6.20     4.08     5.43
```

```
 v13       4.76 0.278 6.19     4.08     5.43
 v14       4.78 0.278 6.19     4.10     5.45
 v15       4.97 0.278 6.19     4.29     5.65
 v16       4.73 0.279 6.20     4.05     5.41
 v17       4.60 0.278 6.19     3.93     5.28
 v18       4.36 0.279 6.20     3.69     5.04
 v19       4.84 0.278 6.19     4.16     5.52
 v20       4.04 0.278 6.19     3.36     4.72
 v21       4.80 0.278 6.19     4.12     5.47
 v22       4.53 0.278 6.19     3.85     5.20
 v23       4.25 0.278 6.19     3.58     4.93
 v24       4.15 0.279 6.20     3.48     4.83

Degrees-of-freedom method: kenward-roger
Confidence level used: 0.95
```

Note that `model.tables()` cannot be used here because it cannot deal with the experimental design correctly.

(h) The varieties were not all investigated under the same environmental conditions because not all varieties occurred together in the same block. The blocks are incomplete. The treatment means therefore have to be adjusted by the environmental effects of the replications and the blocks within the replications in order to make them comparable.

(i)
```
> w
[1] 0.1044581
> sed.11
[1] 0.259358
> sed.10
[1] 0.2703310
> lsd.11
[1] 0.5289641
> lsd.10
[1] 0.5513436
```

(j) We need two LSDs because of the incomplete blocks. If two varieties occurred together in the same block (first associates, $\lambda = 1$), the difference between their treatment means can be estimated with greater precision than if they did not occur together in one block (second associates, $\lambda = 0$). "With greater precision" means that the SED is smaller for the first associates and therefore the LSD is smaller, too.

(k) (i) Varieties 1 and 2 never occur together in the same block. The estimated difference between their adjusted treatment means is $5.107700 - 4.478532 = 0.629168$ and has to be compared with the $\text{LSD}_{\lambda=0} = 0.5513436$. Varieties 1 and 2 therefore have significantly different yields.

(ii) Varieties 1 and 5 occur together in block 4 in replication 2. The estimated difference between their adjusted treatment means is $5.107700 - 5.037210 = 0.07049$ and has to be compared with the $\text{LSD}_{\lambda=1} = 0.5289641$. Varieties 1 and 5 therefore do not have significantly different yields.

4 Oats / Split plot

(a) There were six blocks.

(b) The size of the main plots was 4.

(c) The treatment factors were “variety” with 3 factor levels and “nitrogen level” with 4 factor levels.

(d)
```
Error: Block
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals  5  15875    3175

Error: Block:Variety
          Df Sum Sq Mean Sq F value Pr(>F)
Variety    2   1786   893.2   1.485  0.272
Residuals 10   6013   601.3

Error: Within
              Df Sum Sq Mean Sq F value   Pr(>F)
Nitro          3  20020    6673  37.686 2.46e-12 ***
Variety:Nitro  6    322      54   0.303    0.932
Residuals     45   7969     177
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

The variety does not have a significant influence on the yield, the nitrogen level does. There are no significant interactions.

(e)
```
Tables of means
Grand mean

103.9722

 Variety
Variety
Golden Rain  Marvellous     Victory
     104.50      109.79       97.63

 Nitro
Nitro
  n0.0   n0.2   n0.4   n0.6
 79.39  98.89 114.22 123.39

 Variety:Nitro
             Nitro
Variety       n0.0   n0.2   n0.4   n0.6
  Golden Rain  80.00  98.50 114.67 124.83
  Marvellous   86.67 108.50 117.17 126.83
  Victory      71.50  89.67 110.83 118.50
```

Since the design of the experiment is balanced, you can use `model.tables()` to estimate the means. A safer way is to use `emmeans()` because it also obtains the correct results for unbalanced data. In this case, the results for both are the same and the `model.tables()` output is printed simply because it is shorter.

(f)
```
> sed.1
[1] 7.078904
>sed.2
[1] 4.435755
> sed.3
[1] 7.682954
> sed.4
[1] 9.715025
> lsd.1
[1] 15.77278
> lsd.2
```

```
[1] 8.93407
> lsd.3
[1] 15.47426
> lsd.4
[1] 19.83438
```

(g) Use `lsd.1` ($\text{LSD}_{i,i'(\alpha=0.05)}$) for the comparisons between the varieties, `lsd.2` ($\text{LSD}_{j,j'\ (\alpha=0.05)}$) for the comparisons between the nitrogen levels, `lsd.3` ($\text{LSD}_{ij,ij'\ (\alpha=0.05)}$) for comparisons between nitrogen levels within *one* variety, and `lsd.4` ($\text{LSD}_{ij,i'j'\ (\alpha=0.05)}$) for comparisons between nitrogen levels for different levels of variety.

(i) `lsd.1` = 15.77278. The estimated difference between the means is $109.79 - 97.63 = 12.16$. The yield of the two varieties is not significantly different.
(ii) `lsd.4` = 19.83438. The estimated difference between the means is $108.50 - 89.67 = 18.83$. The two means are not significantly different.
(iii) `lsd.2` = 8.93407. The estimated difference between the means is $123.39 - 114.22 = 9.17$. The two means are significantly different.
(iv) `lsd.3` = 15.47426. The estimated difference between the means is $98.50 - 80.00 = 18.5$. The two means are significantly different.

Matrix algebra

Matthias Frisch and Carola Zenke-Philippi

Load required packages:

```
library("MASS")      # Provides: ginv()
```

1 Special matrices

Enter a matrix in R:

```
A.m <- matrix(c(3, 0, 0, 1, 0, 0),              # define a matrix
              byrow=T,
              ncol=2)
```

```
> A.m
     [,1] [,2]
[1,]    3    0
[2,]    0    1
[3,]    0    0
```

Matrix $\mathbf{A}$ has dimension 3×2.

```
t(A.m)                                          # transpose a matrix
```

```
> t(A.m)
     [,1] [,2] [,3]
[1,]    3    0    0
[2,]    0    1    0
```

Rows and columns are inverted. The result is called $\mathbf{A}'$ ("A prime").

```
diag(c(2,3,5,11))                               # create a diagonal matrix
```

```
> diag(c(2,3,5,11))
     [,1] [,2] [,3] [,4]
[1,]    2    0    0    0
[2,]    0    3    0    0
[3,]    0    0    5    0
[4,]    0    0    0   11
```

Diagonal matrix (a square matrix): All elements which are not on the main diagonal are 0. The function `diag()` already works if you specify the elements on the diagonal. The rest of the matrix is filled with 0s.

```
diag(1, 4)                                      # create an identity matrix
```

```
> diag(1, 4)
     [,1] [,2] [,3] [,4]
[1,]    1    0    0    0
[2,]    0    1    0    0
[3,]    0    0    1    0
[4,]    0    0    0    1
```

Identity matrix: diagonal matrix with 1s on the main diagonal.

```
diag(A.m)                                       # return elements on the diagonal
```

```
> diag(A.m)
[1] 3 1
```

```
sum(diag(A.m))                                  # trace
```

```
> sum(diag(A.m))
[1] 4
```

The *trace* of a diagonal matrix is the sum of its diagonal elements: $3 + 1 = 4$.

2 Basic matrix operations

```
A.m <- matrix(c(3, 0, 1, 0, 1, 2, 1, 2, 4),
              byrow=T,
              ncol=3); A.m

B.m <- matrix(c(1, 0, 0, 0, 2, 0, 0, 0, 4),
              byrow=T,
              ncol=3); B.m

C.m <- matrix(c(1, 2, 3, 0, 1, 2, 0, 0, 1),
              byrow=T,
              ncol=3); C.m

D.m <- matrix(1:6, byrow=F, ncol=2); D.m

E.m <- matrix(rep(1:4, times=2), byrow=T, ncol=4); E.m
```

```
> A.m
     [,1] [,2] [,3]
[1,]    3    0    1
[2,]    0    1    2
[3,]    1    2    4

> B.m
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    2    0
[3,]    0    0    4

> C.m
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    0    1    2
[3,]    0    0    1

> D.m
     [,1] [,2]
[1,]    1    4
[2,]    2    5
[3,]    3    6

> E.m
     [,1] [,2] [,3] [,4]
[1,]    1    2    3    4
[2,]    1    2    3    4
```

```
A.m + B.m                                                    # Addition
```

```
> A.m + B.m
     [,1] [,2] [,3]
[1,]    4    0    1
[2,]    0    3    2
[3,]    1    2    8
```

Matrices are added element-wise. Addition of two matrices is only possible if the two matrices have the same dimension, *i.e.* the same number of rows and columns.

```
3 * C.m                                                      # Multiplication with a scalar
```

```
> 3 * C.m
     [,1] [,2] [,3]
[1,]    3    6    9
[2,]    0    3    6
[3,]    0    0    3
```

Multiplication with a scalar (a number) is done element-wise, too.

```
D.m %*% E.m                                                  # Matrix multiplication
```

```
> D.m %*% E.m
     [,1] [,2] [,3] [,4]
[1,]    5   10   15   20
[2,]    7   14   21   28
[3,]    9   18   27   36
```

$$\begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix} \times \begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix} = \begin{pmatrix} 1\times1+4\times1 & 1\times2+4\times2 & 1\times3+4\times3+1\times4+4\times4 \\ 2\times1+5\times1 & 2\times2+5\times2 & 2\times3+5\times3+2\times4+5\times4 \\ 3\times1+6\times1 & 3\times2+6\times2 & 3\times3+6\times3+3\times4+6\times4 \end{pmatrix}$$
$$= \begin{pmatrix} 5 & 10 & 15 & 20 \\ 7 & 14 & 21 & 28 \\ 9 & 18 & 27 & 36 \end{pmatrix}$$

Matrices can only be multiplied if the matrices have the right dimensions:

$$\mathbf{D}_{a\times b} \times \mathbf{E}_{b\times c} = \mathbf{D}_{3\times 2} \times \mathbf{E}_{2\times 4} = \mathbf{F}_{3\times 4}$$

On the multiplication of matrices with vectors: Column vectors with all their elements in one column are regarded as matrices with as many rows as the vector has elements and one column, *i.e.* they have the dimension $n \times 1$. Row vectors with all their elements in one row are regarded as matrices with one row and as many columns as the vector has elements, *i.e.* they have the dimension $1 \times n$. Then all the rules for matrix multiplication apply.

Note that R makes no difference between row and column vectors if you enter them with `c()`. You need to use the data structure created with `matrix()` if you want to differentiate between row and column vectors.

3 Inverse of a matrix

```
A.m <- matrix(1:4, byrow=T, ncol=2); A.m
B.m <- matrix(c(2,3,0,0), byrow=T, ncol=2); B.m
```

```
> A.m
     [,1] [,2]
[1,]    1    2
[2,]    3    4

> B.m
     [,1] [,2]
[1,]    2    3
[2,]    0    0
```

```
solve(A.m)                                          # Inverse
```

```
> solve(A.m)
     [,1] [,2]
[1,] -2.0  1.0
[2,]  1.5 -0.5
```

Multiplication of a matrix with its inverse results in the identity matrix. Watch out for rounding errors in the R output.

```
solve(A.m) %*% A.m
A.m %*% solve(A.m)
```

```
> solve(A.m) %*% A.m
     [,1]         [,2]
[1,]    1 4.440892e-16
[2,]    0 1.000000e+00

> A.m %*% solve(A.m)
     [,1]         [,2]
[1,]    1 1.110223e-16
[2,]    0 1.000000e+00
```

```
det(A.m)                                            # Determinant
A.m[1,1]*A.m[2,2]-A.m[1,2]*A.m[2,1]
```

```
> det(A.m)
[1] -2

> A.m[1,1]*A.m[2,2]-A.m[1,2]*A.m[2,1]
[1] -2
```

Calculate the inverse by hand:

```
1/det(A.m) *
   matrix(c(A.m[2,2], -A.m[1,2], -A.m[2,1], A.m[1,1]), byrow=T, ncol=2)
```

```
> 1/det(A.m) *
    matrix(c(A.m[2,2], -A.m[1,2], -A.m[2,1], A.m[1,1]), byrow=T, ncol=2)
     [,1] [,2]
[1,] -2.0  1.0
[2,]  1.5 -0.5
```

Matrix **B** is not invertible:

```
det(B.m)
solve(B.m)
```

```
> det(B.m)
[1] 0

> solve(B.m)
Error in solve.default(B.m) :
  Lapack routine dgesv: system is exactly singular: U[2,2] = 0
```

4 Rank of a matrix

The vectors

$$a_1 = \begin{pmatrix} 2 \\ 0 \\ -3 \end{pmatrix} \quad a_2 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \quad a_3 = \begin{pmatrix} 7 \\ 3 \\ 0 \end{pmatrix}$$

are linear dependent because *e.g.*

$$a_3 = 2a_1 + 3a_2$$

The vectors

$$b_1 = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}; \quad b_2 = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}; \quad b_3 = \begin{pmatrix} 2 \\ 0 \\ 2 \end{pmatrix}$$

are linear independent because there are not coefficients that transform one in the other.

The column rank (row rank) of a matrix is defined a the maximum number of linear independent column vectors (row vectors) of the matrix.

$$\mathbf{A} = \begin{pmatrix} 2 & 1 & 7 \\ 0 & 1 & 3 \\ -3 & 2 & 0 \end{pmatrix}; \quad \mathbf{B} = \begin{pmatrix} 1 & 1 & 2 \\ 0 & 1 & 0 \\ -1 & -1 & 2 \end{pmatrix}$$

```
A.m <- matrix(c(2, 1, 7, 0, 1 , 3, -3, 2, 0), byrow=T, ncol=3)
B.m <- matrix(c(1, 1, 2, 0, 1, 0, -1, -1, 2), byrow=T, ncol=3)
qr(A.m)$rank
qr(B.m)$rank
```

```
> qr(A.m)$rank
[1] 2

> qr(B.m)$rank
[1] 3
```

Matrix **A** does not have full rank (3 columns but only rank 2). Matrix **B** has full rank (3 columns, rank 3). Only matrix **B** is therefore regular. Only matrices that have full rank are invertible.

The rank is also defined for non-square matrices.

5 Moore-Penrose Inverse

Only regular matrices are invertible. However, many algorithms depend on inverses to function. In order to apply these algorithms also to non-invertible matrices, we need to find matrices that have (at least partially) the same characteristics as inverses without actually being "real" inverses. These matrices are called generalized inverses.

```
A.m <- matrix(c(2, 0, 1, 0), byrow=T, ncol=2)
ginv(A.m)
```

```
> ginv(A.m)
     [,1] [,2]
[1,]  0.4  0.2
[2,]  0.0  0.0
```

6 Eigenvalues und eigenvectors

Eigenvalues and eigenvectors can only be determined for quadratic matrices. Each $m \times m$ matrix has a maximum of m eigenvalues. For each eigenvalues, there is an infinite number of eigenvectors.

```
A.m <- matrix(c(3, 0, -9, 6), byrow=T, ncol=2)
eigen(A.m)$values                                  # determine the eigenvalues
```

```
> A.m
     [,1] [,2]
[1,]    3    0
[2,]   -9    6

> eigen(A.m)$values
[1] 6 3
```

Is $\mathbf{e} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ an eigenvector for the eigenvalue 6 of the matrix $\mathbf{A}$?

A vector $\mathbf{x}$ is called eigenvektor to the eigenvalue λ of a square matrix $\mathbf{A}$ if $\mathbf{Ax} = \lambda\mathbf{x}$ (slide `14-31-050`. This means that when the eigenvector of a matrix is multiplied with the matrix, the result is eigenvalue $\times$ the eigenvector. We therefore check if $\mathbf{Ae} = \lambda\mathbf{e}$.

```
e.v <- c(0, 1)
6 %*% e.v
```

```
> A.m %*% e.v
     [,1]
[1,]    0
[2,]    6

> 6 %*% e.v
     [,1] [,2]
[1,]    0    6
```

The product of the eigenvalues of a matrix equals its determinant.

```
det(A.m)
```

```
> det(A.m)
[1] 18
```

The sum of the eigenvalues of a matrix equals its trace.

```
sum(diag(A.m))
```

```
> sum(diag(A.m))
[1] 9
```

The eigenvectors to two different eigenvalues are orthogonal for symmetric matrices.

```
A.m <- matrix(c(3, 0, 0, 6), byrow=T, ncol=2)
eigen(A.m)$values
e.v1 <- c(0, 1)
A.m %*% e.v1
6 %*% e.v1
e.v2 <- c(-1, 0)
A.m %*% e.v2
3 %*% e.v2
```

```
> A.m
     [,1] [,2]
[1,]    3    0
[2,]    0    6

> eigen(A.m)$values
[1] 6 3

> e.v1 %*% e.v2
     [,1]
[1,]    0
```

7 System of linear equations

A plot of wheat is treated with 2.5 kg fertilizer and a second plot is treated with 6.5 kg. This results in a system of linear equations with two equations and two unknows. Graphically: Both points can be on the line.

System of linear equations:

$$22 = b_0 + b_1\ 2.5$$
$$35 = b_0 + b_1\ 6.5$$

In matrix notation:

$$\begin{pmatrix} 22 \\ 35 \end{pmatrix} = \begin{pmatrix} 1 & 2.5 \\ 1 & 6.5 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \end{pmatrix}$$
$$\mathbf{y} = \mathbf{Xb}$$

```
y.yield <- c(22, 35)
X.d <- matrix(c(1, 1, 2.5, 6.5), byrow=F, ncol=2)
```

There is a solution to the system of linear equations because $\mathbf{X}\mathbf{X}^{+}\mathbf{y} = \mathbf{y}$:

```
X.d %*% ginv(X.d) %*% y.yield
```

```
> X.d %*% ginv(X.d) %*% y.yield
     [,1]
[1,]   22
[2,]   35
```

$\mathbf{X}^+ = \mathbf{X}^{-1}$ if $\mathbf{X}$ is regular.

```
solve(X.d)                                          # inverse
ginv(X.d)                                           # Moore-Penrose inverse
```

```
> solve(X.d)
       [,1]   [,2]
[1,]  1.625 -0.625
[2,] -0.250  0.250

> ginv(X.d)
       [,1]   [,2]
[1,]  1.625 -0.625
[2,] -0.250  0.250
```

Determine the parameter vector $\mathbf{b} = \mathbf{X}^{-1}\mathbf{y}$:

```
b.v <- ginv(X.d) %*% y.yield ; b.v
```

```
> b.v
       [,1]
[1,] 13.875
[2,]  3.250
```

Check if the result is true with **Xb**:

```
X.d %*% b.v
```

```
> X.d %*% b.v
     [,1]
[1,]   22
[2,]   35
```

8 Overdetermined systems of linear equations

x	2.5	3.0	3.5	4.0	4.5	5.0	5.5	6.0	6.5
y	22.0	17.5	27.0	23.0	25.0	22.5	33.0	26.0	35.0

$$y = \begin{pmatrix} 22.0 \\ 17.5 \\ 27.0 \\ 23.0 \\ 25.0 \\ 22.5 \\ 33.0 \\ 26.0 \\ 35.0 \end{pmatrix} \quad X = \begin{pmatrix} 1 & 2.5 \\ 1 & 3.0 \\ 1 & 3.5 \\ 1 & 4.0 \\ 1 & 4.5 \\ 1 & 5.0 \\ 1 & 5.5 \\ 1 & 6.0 \\ 1 & 6.5 \end{pmatrix}$$

The system of linear equations is overdetermined because we have more equations than unknowns. There is no single number for each unknown that fulfills the conditions of all equations.

```
y.yield <- c(22, 17.5, 27, 23, 25, 22.5, 33, 26, 35)
X.d <- matrix(c(rep(1, times=9), seq(2.5, 6.5, 0.5)), byrow=F, ncol=2)
```

The system of linear equations $\mathbf{y} = \mathbf{Xb}$ cannot be solved because $\mathbf{XX}^+\mathbf{y} \neq \mathbf{y}$.

```
X.d %*% ginv(X.d) %*% y.yield
```

```
> X.d %*% ginv(X.d) %*% y.yield
         [,1]
 [1,] 19.73333
 [2,] 21.21667
 [3,] 22.70000
 [4,] 24.18333
 [5,] 25.66667
 [6,] 27.15000
 [7,] 28.63333
 [8,] 30.11667
 [9,] 31.60000
```

Approximate solution with the least squares principle:

$$\hat{\mathbf{b}} = (\mathbf{X}'\mathbf{X})^{+}\mathbf{X}'\mathbf{y}$$

```
b.v <- ginv(t(X.d) %*% X.d) %*% t(X.d) %*% y.yield
```

Vector of coefficients:

```
> b.v
          [,1]
[1,] 12.316667
[2,]  2.966667
```

This first value is the intercept $\hat{\beta}_0$ or $\hat{b}_0$, the second value is the slope $\hat{\beta}_1$ or $\hat{b}_1$. Greek letters without hats (β) stand for the true values of the parameters while Greek letters with hats stand for the estimated values. Alternatively, the estimated parameters can be denoted with Latin letters (b) or Latin letters with hats ($\hat{b}$).

Points on the line: estimate the yield values with $\mathbf{y} = \mathbf{Xb}$:

```
X.d %*% b.v
```

```
> X.d %*% b.v
         [,1]
 [1,] 19.73333
 [2,] 21.21667
 [3,] 22.70000
 [4,] 24.18333
 [5,] 25.66667
 [6,] 27.15000
 [7,] 28.63333
 [8,] 30.11667
 [9,] 31.60000
```

9 Quadratic forms

In the method of least squares, the quadratic form

$$q_{\mathbf{H}}(\mathbf{y}) = \mathbf{y}'\mathbf{H}\mathbf{y} \text{ mit } \mathbf{H} = \mathbf{I} - \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{+}\mathbf{X}'$$

describes the sum of the squared deviations of the points from the line.

```
H.m <- diag(1, nrow(X.d)) - X.d %*% ginv(t(X.d) %*% X.d) %*% t(X.d)
q.h.y <- y.yield %*% H.m %*% y.yield
q.h.y
```

```
> q.h.y
 108.4833
```

10 Exercises

10.1 Matrix operations

Consider the following matrices and vectors:

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 0 & 4 \\ 2 & 4 & 0 \end{pmatrix}; \quad \mathbf{B} = \begin{pmatrix} 3 & 2 \\ 2 & 0 \end{pmatrix}; \quad \mathbf{C} = \begin{pmatrix} 4 & 1 & -1 & 1 \\ 1 & 4 & 1 & -1 \\ -1 & 1 & 4 & 1 \\ 1 & -1 & 1 & 4 \end{pmatrix}$$

$$\mathbf{d} = \begin{pmatrix} 0.5 \\ -0.5 \\ 0.5 \\ -0.5 \end{pmatrix}; \quad \mathbf{e} = \begin{pmatrix} -4.3 \\ 1 \\ 3 \\ 4.3 \end{pmatrix}; \quad \mathbf{f} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}; \quad \mathbf{g} = \begin{pmatrix} 3.1 \\ -3.1 \\ 3.1 \\ -3.1 \end{pmatrix}$$

(a) What dimensions do the matrices have?

(b) Use the determinant to check whether the matrices are invertible.

(c) If so, state the inverse. If not, state the Moore-Penrose inverse.

(d) Calculate the trace of each matrix.

(e) Which vectors are pairwise linear independent?

(f) What rank do the matrices have? Which matrices have full rank?

(g) Which matrices are regular, which are singular?

(h) Calculate the eigenvalues of the matrices.

(i) Which of the vectors are eigenvectors of the matrix $\mathbf{C}$?

(j) Which of the matrices are symmetric?

10.2 System of linear equations

Use matrices to determine the solution for the following systems of linear equations:

(a)

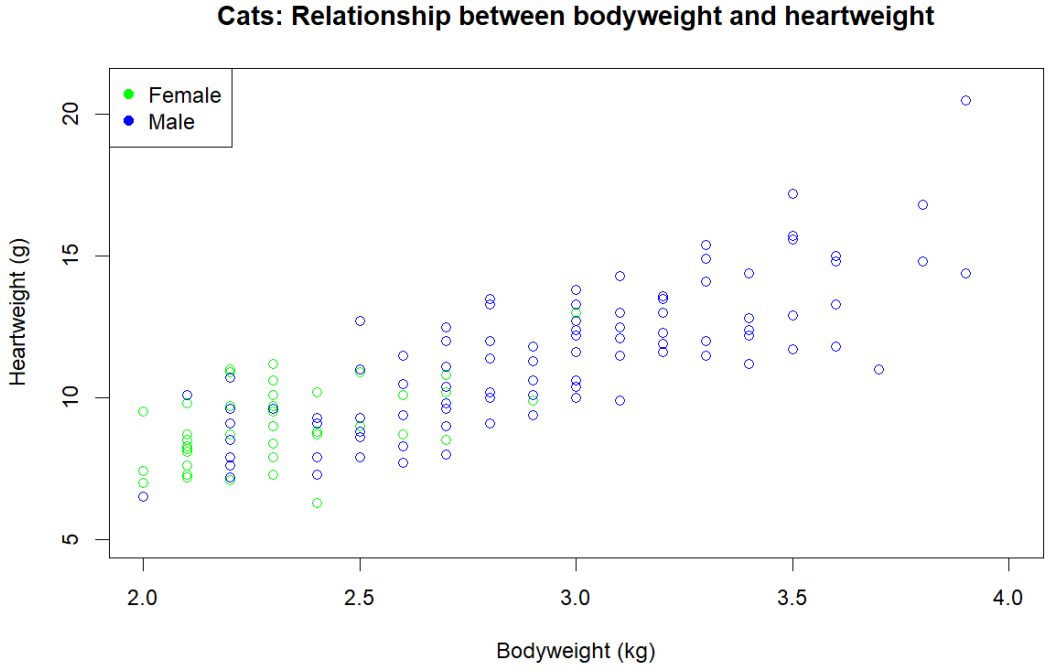
$$\begin{aligned} 40 &= b_0 + 30b_1 \\ -25 &= b_0 - 5b_1 \end{aligned}$$

(b)

$$\begin{aligned} -2.3 &= b_0 - 12b_1 + 4.2b_2 \\ 18.8 &= b_0 + 9.1b_1 + 0.6b_2 \\ 122 &= b_0 + 54.8b_1 - 14.4b_2 \end{aligned}$$

10.3 Overdetermined systems of linear equations

The data set `v14-u61-01.csv` (originally from the R package `MASS`) contains bodyweight in kg (`Bwt`) and heart weight in g (`Hwt`) from adult cats.



The following exercises are supposed to be solved with matrices and vectors in R.

(a) Get an overview over the data set. How many variables does it contain?

(b) The scatterplots points to a linear relationship between bodyweight and heartweight. Create a design matrix for a model that explains the heartweight from the bodyweight.

(c) Solve the equations in order to determine the parameter vector **b**.

(d) What is the matrix equation that describes the model?

(e) Which heartweight does the model estimate for cats with a bodyweight of 2.5 and 3 kg? Create a new design matrix first.

11 Solutions

11.1 Matrix operations

(a) What dimensions do the matrices and vectors have? **A**: 3x3, **B**: 2x2, **c**: 4x4, all vectors: 4x1

(b) Use the determinant to check whether the matrices are invertible.

$\det \mathbf{A} = 0, \quad \det \mathbf{B} = -4, \quad \det \mathbf{C} = 125$

Matrices **B** and **C** are invertible, matrix **A** is not.

(c) If so, state the inverse. If not, state the Moore-Penrose inverse.

$$\mathbf{A}^{+} = \begin{pmatrix} 0.0333 & 0.0833 & 0.0667 \\ 0.0833 & -0.4167 & 0.1667 \\ -0.0167 & 0.2083 & -0.0333 \end{pmatrix}; \quad \mathbf{B}^{-1} = \begin{pmatrix} 0 & 0.5 \\ 0.5 & -0.75 \end{pmatrix};$$

$$\mathbf{C}^{-1} = \begin{pmatrix} 0.4 & -0.2 & 0.2 & -0.2 \\ -0.2 & 0.4 & -0.2 & 0.2 \\ 0.2 & -0.2 & 0.4 & -0.2 \\ -0.2 & 0.2 & -0.2 & 0.4 \end{pmatrix}$$

(d) Calculate the trace of each matrix.

$\operatorname{tr} \mathbf{A} = 1, \quad \operatorname{tr} \mathbf{B} = 3, \quad \operatorname{tr} \mathbf{C} = 16$

(e) Which vectors are pairwise linear independent?

Only **d** und **g** are linear dependent.

(f) What rank do the matrices have? Which matrices have full rank?

$\operatorname{rk} \mathbf{A} = 2, \quad \operatorname{rk} \mathbf{B} = 2, \quad \operatorname{rk} \mathbf{C} = 4$

Matrices **B** and **C** have full rank, matrix **A** does not.

(g) Which matrices are regular, which are singular?

Matrices **B** and **C** are regular, matrix **A** is singular.

(h) Calculate the eigenvalues of the matrices.

Eigenvalues of **A**: $\lambda_1 = 5, \quad \lambda_2 = -4, \quad \lambda_3 = 0$
Eigenvalues of **B**: $\lambda_1 = 4, \quad \lambda_2 = -1$
Eigenvalues of **C**: $\lambda_1 = 5, \quad \lambda_2 = 1$

(i) Which of the vectors are eigenvectors of the matrix **C**?

d, **f**, and **g**

(j) Which of the matrices are symmetric?

B, and **C**

11.2 System of linear equations

Use matrices to determine the solution for the following systems of linear equations:

(a) $b_0 = -15.714286; \quad b_1 = 1.857143$

(b) $b_0 = 26.8842; \quad b_1 = -0.3620; \quad b_2 = -7.9830$

11.3 Overdetermined systems of linear equations

The data set `v14-u61-01.csv` (originally from the R package `MASS`) contains bodyweight in kg (`Bwt`) and heart weight in g (`Hwt`) from adult cats.

(a) Get an overview over the data set. How many variables does it contain?

(b) The scatterplots points to a linear relationship between bodyweight and heartweight. Create a design matrix for a model that explains the heartweight from the bodyweight.

```
> X.d
      [,1] [,2]
  [1,]    1  2.0
  [2,]    1  2.0
  [3,]    1  2.0
...
[143,]    1  3.9
[144,]    1  3.9
```

(c) Solve the equations in order to determine the parameter vector **b**.

```
> b.v
           [,1]
[1,] -0.3566624
[2,]  4.0340627
```

(d) What is the matrix equation that describes the model?

$$\begin{pmatrix} 7.0 \\ 7.4 \\ 9.5 \\ 7.2 \\ \cdots \\ 14.8 \\ 16.8 \\ 14.4 \\ 20.5 \end{pmatrix} = \begin{pmatrix} 1 & 2.0 \\ 1 & 2.0 \\ 1 & 2.0 \\ 1 & 2.1 \\ & \cdots \\ 1 & 3.8 \\ 1 & 3.8 \\ 1 & 3.9 \\ 1 & 3.9 \end{pmatrix} \begin{pmatrix} -0.36 \\ 4.03 \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ \cdots \\ e_{141} \\ e_{142} \\ e_{143} \\ e_{144} \end{pmatrix}$$

This equation is not written on a slide. Rather, it is similar to the equations you encountered in the chapter on experimental designs: Each observation consists of an intercept (-0.36) and a slope for the bodyweight (heartweight increases by 4.03 grams if the boyweight increases by 1 kg).

(e) Which heartweight does the model estimate for cats with a bodyweight of 2.5 and 3 kg? Create a new design matrix first.

```
> X.d <- matrix(c(1, 2.5, 1, 3), byrow=T, ncol=2)
> (v.v <- X.d %*% b.v)
          [,1]
[1,]  9.728494
[2,] 11.745526
```

Test of fixed and random effects

```
mm <- lmer ( Yield ~ Variety + Nitro + Variety:Nitro
                     + (1|Block) + (1|Block:Variety),
                    data=oats )

anova(mm)               # Tests of fixed effects
ranova(mm)              # Tests of random effects
```

`lmer()` fits a mixed linear model. `Variety` and `Nitro` and their interaction are fixed effects. The block effect `(1|Block)` and the main plot error `(1|Block:Variety)` are random effects.

```
> anova(mm)
Type III Analysis of Variance Table with Satterthwaite's method
              Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
Variety        526.1   263.0     2    10  1.4853    0.2724
Nitro        20020.5  6673.5     3    45 37.6856 2.458e-12 ***
Variety:Nitro  321.8    53.6     6    45  0.3028    0.9322
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ranova(mm)
ANOVA-like table for random-effects: Single term deletions

Model:
Yield ~ Variety + Nitro + (1 | Block) + (1 | Block:Variety) +
    Variety:Nitro
                   npar  logLik    AIC    LRT Df Pr(>Chisq)
<none>               15 -273.07 576.14
(1 | Block)          14 -275.56 579.11 4.9782  1   0.025669 *
(1 | Block:Variety)  14 -276.90 581.80 7.6615  1   0.005641 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The nitrogen fertilizer had a significant influence on the yield whereas the variety did not. There was no significant interaction between the nitrogen level and the fertilizer. Both random effects were significant.

Estimation of variance components for the random effects

```
print ( VarCorr(m) , comp="Variance")
```

```
> print ( VarCorr(m) , comp="Variance")
 Groups        Name        Variance
 Block:Variety (Intercept) 106.06
 Block         (Intercept) 214.47
 Residual                  177.08
```

The variance components of random effects quantify the variation that is caused by a certain effect. The residual variance or error variance is the variance that remains unexplained by the model.

Treatment means, pairwise tests, and grouping of treatments

```
e.N <- emmeans ( mm, ~Nitro )
cld ( e.N, adjust="hochberg")
test ( contrast ( e.N , "pairwise"),
       adjust="hochberg" )
```

```
> cld ( e.N, adjust="hochberg")
 Nitro emmean   SE   df lower.CL upper.CL .group
 n0.0    79.4 7.17 6.79     55.2      104  1
 n0.2    98.9 7.17 6.79     74.7      123   2
 n0.4   114.2 7.17 6.79     90.1      138    3
 n0.6   123.4 7.17 6.79     99.2      148     4

> test ( contrast ( e.N , "pairwise"),
+        adjust="hochberg" )
 contrast    estimate   SE df t.ratio p.value
 n0.0 - n0.2   -19.50 4.44 45 -4.396  0.0002
 n0.0 - n0.4   -34.83 4.44 45 -7.853  <.0001
 n0.0 - n0.6   -44.00 4.44 45 -9.919  <.0001
 n0.2 - n0.4   -15.33 4.44 45 -3.457  0.0024
 n0.2 - n0.6   -24.50 4.44 45 -5.523  <.0001
 n0.4 - n0.6    -9.17 4.44 45 -2.067  0.0446
```

```
e.V <- emmeans ( mm, ~Variety )
cld ( e.V, adjust="hochberg")
test ( contrast ( e.V , "pairwise"),
       adjust="hochberg" )
```

```
> cld ( e.V, adjust="hochberg")
 Variety     emmean  SE   df lower.CL upper.CL .group
 Victory       97.6 7.8 8.87     74.7      121  1
 Golden Rain  104.5 7.8 8.87     81.6      127  1
 Marvellous   109.8 7.8 8.87     86.8      133  1

> test ( contrast ( e.V , "pairwise"),
+        adjust="hochberg" )
 contrast                estimate   SE df t.ratio p.value
 Golden Rain - Marvellous    -5.29 7.08 10 -0.748  0.4720
 Golden Rain - Victory        6.88 7.08 10  0.971  0.4720
 Marvellous - Victory        12.17 7.08 10  1.719  0.3492
```

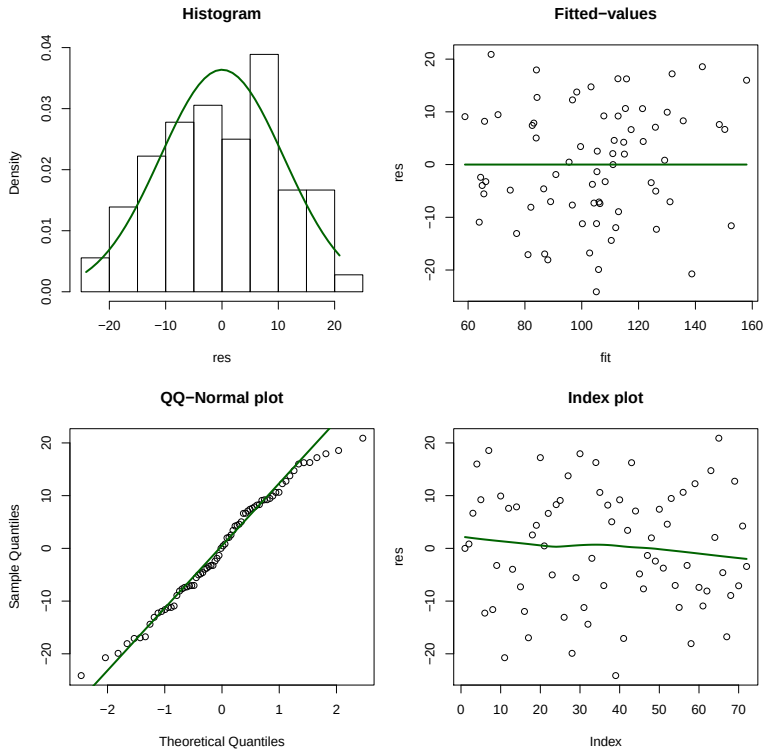
```
e.VN <- emmeans ( mm, ~Variety:Nitro )
cld ( e.VN,  adjust="hochberg", alpha=0.10, sort=F)
test ( contrast ( e.VN , "pairwise"),
       adjust="hochberg")
```

```
> cld ( e.VN,  adjust="hochberg", alpha=0.10, sort=F)
 Variety     Nitro emmean   SE   df lower.CL upper.CL .group
 Golden Rain n0.0    80.0 9.11 16.1     49.6      110  12
 Marvellous  n0.0    86.7 9.11 16.1     56.3      117  1234
 Victory     n0.0    71.5 9.11 16.1     41.1      102  1
 Golden Rain n0.2    98.5 9.11 16.1     68.1      129  123456
 Marvellous  n0.2   108.5 9.11 16.1     78.1      139   234567
 Victory     n0.2    89.7 9.11 16.1     59.3      120  123 5
 Golden Rain n0.4   114.7 9.11 16.1     84.3      145    34567
```

```
 Marvellous  n0.4   117.2 9.11 16.1     86.8      148      567
 Victory     n0.4   110.8 9.11 16.1     80.5      141   234567
 Golden Rain n0.6   124.8 9.11 16.1     94.5      155        7
 Marvellous  n0.6   126.8 9.11 16.1     96.5      157       67
 Victory     n0.6   118.5 9.11 16.1     88.1      149     4 67

Degrees-of-freedom method: kenward-roger
P value adjustment: hochberg method for 66 tests
significance level used: alpha = 0.1

> test ( contrast ( e.VN , "pairwise"),
+        adjust="hochberg")
 contrast                            estimate   SE   df t.ratio p.value
 Golden Rain,n0.0 - Marvellous,n0.0     -6.67 9.71 30.2 -0.686  0.8917
 Golden Rain,n0.0 - Victory,n0.0         8.50 9.71 30.2  0.875  0.8917
 Golden Rain,n0.0 - Golden Rain,n0.2   -18.50 7.68 45.0 -2.408  0.7273
 Golden Rain,n0.0 - Marvellous,n0.2    -28.50 9.71 30.2 -2.934  0.2852
 Golden Rain,n0.0 - Victory,n0.2        -9.67 9.71 30.2 -0.995  0.8917
 Golden Rain,n0.0 - Golden Rain,n0.4   -34.67 7.68 45.0 -4.512  0.0027
 Golden Rain,n0.0 - Marvellous,n0.4    -37.17 9.71 30.2 -3.826  0.0311
 Golden Rain,n0.0 - Victory,n0.4       -30.83 9.71 30.2 -3.174  0.1585
 Golden Rain,n0.0 - Golden Rain,n0.6   -44.83 7.68 45.0 -5.835  <.0001
 Golden Rain,n0.0 - Marvellous,n0.6    -46.83 9.71 30.2 -4.821  0.0023
 Golden Rain,n0.0 - Victory,n0.6       -38.50 9.71 30.2 -3.963  0.0230
 Marvellous,n0.0 - Victory,n0.0         15.17 9.71 30.2  1.561  0.8917
 Marvellous,n0.0 - Golden Rain,n0.2    -11.83 9.71 30.2 -1.218  0.8917
 Marvellous,n0.0 - Marvellous,n0.2     -21.83 7.68 45.0 -2.842  0.2889
 Marvellous,n0.0 - Victory,n0.2         -3.00 9.71 30.2 -0.309  0.8917
 Marvellous,n0.0 - Golden Rain,n0.4    -28.00 9.71 30.2 -2.882  0.3026
 Marvellous,n0.0 - Marvellous,n0.4     -30.50 7.68 45.0 -3.970  0.0143
 Marvellous,n0.0 - Victory,n0.4        -24.17 9.71 30.2 -2.488  0.6882
 Marvellous,n0.0 - Golden Rain,n0.6    -38.17 9.71 30.2 -3.929  0.0248
 Marvellous,n0.0 - Marvellous,n0.6     -40.17 7.68 45.0 -5.228  0.0003
 Marvellous,n0.0 - Victory,n0.6        -31.83 9.71 30.2 -3.277  0.1241
 Victory,n0.0 - Golden Rain,n0.2       -27.00 9.71 30.2 -2.779  0.3620
 Victory,n0.0 - Marvellous,n0.2        -37.00 9.71 30.2 -3.809  0.0319
 Victory,n0.0 - Victory,n0.2           -18.17 7.68 45.0 -2.365  0.7625
...

P value adjustment: hochberg method for 66 tests
```

Analysis of residuals

```
source("v14-u11-00.R")  # Some diagnostic plots, see Appendix
check.residuals(m)
```


Histogram
Density
res
Fitted−values
res
fit
QQ−Normal plot
Sample Quantiles
Theoretical Quantiles
Index plot
res
Index

1.2 Balanced lattice in six replications: Barley

The data is from a field experiment designed to compare the performance of 25 varieties of barley. The experiment was conducted at Slate Hall Farm, UK, in 1976 and was designed as a balanced lattice with 6 replicates laid out in a 10 × 15 rectangular grid. Gilmour AR et al., 1995, Biometrics 51:1440–1450. The data set is used as an example in Gilmour AR et al., 2009, ASReml User Guide, Release 3. The plan for the 5 × 5 balanced lattice design is from Cochran and Cox (1957).

Plan 10.3 5 × 5 balanced lattice

$t = 25, k = 5, r = 6, b = 30, \lambda = 1$

Block	Rep. I					Block	Rep. II					Block	Rep. III				
(1)	1	2	3	4	5	(6)	1	6	11	16	21	(11)	1	7	13	19	25
(2)	6	7	8	9	10	(7)	2	7	12	17	22	(12)	21	2	8	14	20
(3)	11	12	13	14	15	(8)	3	8	13	18	23	(13)	16	22	3	9	15
(4)	16	17	18	19	20	(9)	4	9	14	19	24	(14)	11	17	23	4	10
(5)	21	22	23	24	25	(10)	5	10	15	20	25	(15)	6	12	18	24	5
	Rep. IV						Rep. V						Rep. VI				
(16)	1	12	23	9	20	(21)	1	17	8	24	15	(26)	1	22	18	14	10
(17)	16	2	13	24	10	(22)	11	2	18	9	25	(27)	6	2	23	19	15
(18)	6	17	3	14	25	(23)	21	12	3	19	10	(28)	11	7	3	24	20
(19)	21	7	18	4	15	(24)	6	22	13	4	20	(29)	16	12	8	4	25
(20)	11	22	8	19	5	(25)	16	7	23	14	5	(30)	21	17	13	9	5

*The symbol λ denotes the number of times that two treatments appear in the same block.

Rowblock levels

	Column														
Row	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	1	1	1	11	11	11	11	11	21	21	21	21	21
2	2	2	2	2	2	12	12	12	12	12	22	22	22	22	22
3	3	3	3	3	3	13	13	13	13	13	23	23	23	23	23
4	4	4	4	4	4	14	14	14	14	14	24	24	24	24	24
5	5	5	5	5	5	15	15	15	15	15	25	25	25	25	25
6	6	6	6	6	6	16	16	16	16	16	26	26	26	26	26
7	7	7	7	7	7	17	17	17	17	17	27	27	27	27	27
8	8	8	8	8	8	18	18	18	18	18	28	28	28	28	28
9	9	9	9	9	9	19	19	19	19	19	29	29	29	29	29
10	10	10	10	10	10	20	20	20	20	20	30	30	30	30	30

Columnblock levels

	Column														
Row	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
4	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
5	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
6	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
7	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
8	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
9	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
10	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

```
library("lmerTest"); library("emmeans"); library("multcomp")

barley <- read.table("v14-u51-02.csv",header=T,sep=";",dec=",",
                     stringsAsFactors = T)
str (barley)
```

```
> str(barley)
'data.frame':        150 obs. of  7 variables:
 $ Rep    : Factor w/ 6 levels "re1","re2","re3",..: 1 1 1 1 1 2 2 2 2 2 ...
 $ RowBlk : Factor w/ 30 levels "rb1","rb10","rb11",..: 1 1 1 1 1 3 3 3 3 3 ...
 $ ColBlk : Factor w/ 30 levels "cb1","cb10","cb11",..: 1 12 23 25 26 27 28 29 30 2 ...
 $ Row    : Factor w/ 10 levels "ro1","ro10","ro2",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Column : Factor w/ 15 levels "co1","co10","co11",..: 1 8 9 10 11 12 13 14 15 2 ...
 $ Variety: Factor w/ 25 levels "va01","va02",..: 1 2 4 3 5 19 23 2 6 15 ...
 $ yield  : int  1003 1356 1412 1239 1508 1967 1572 1969 1747 1598 ...
```

Classical lattice analysis

Analysis with all the factors of a classical lattice analysis:

```
m.1 <-lmer ( yield ~ Variety                  # Variety fixed
                   + (1|Rep) + (1|RowBlk),    # Random lattice effects
             data = barley )
print ( VarCorr(m.1) , comp="Variance" )       # Variance components
```

Variance components:

```
> print ( VarCorr(m.1) , comp="Variance" )
 Groups   Name        Variance
 RowBlk   (Intercept) 14385
 Rep      (Intercept)  6882
 Residual             22677
```

We suspect that there was not just environmental variation that was captured by the row blocks but also environmental variation in another direction which can be described if we have a look at the columns of the design. We use the column blocks to describe the direction of that variation.

Added column effect

```
m.2 <-lmer ( yield ~ Variety                          # Variety fixed
                   + (1|Rep) + (1|RowBlk)             # Random lattice effects
                   + (1|ColBlk),                      # Column effect
             control=lmerControl(optimizer="Nelder_Mead"), # Convergence
             data = barley )
print ( VarCorr(m.2) , comp="Variance" )       # Variance components
```

```
> print ( VarCorr(m.2) , comp="Variance" )
 Groups   Name        Variance
 ColBlk   (Intercept) 14811.6
 RowBlk   (Intercept) 15595.0     much smaller
 Rep      (Intercept)  4262.4     residual error
 Residual              8061.8     -> greater power
```

Test of fixed and random effects

```
anova (m)
ranova (m)
```

```
> anova (m.2)
Type III Analysis of Variance Table with Satterthwaite's method
         Sum Sq Mean Sq NumDF  DenDF F value   Pr(>F)
Variety 1711183   71299    24 79.576  8.8441 5.32e-14 ***

> ranova (m.2)
ANOVA-like table for random-effects: Single term deletions

Model:
yield ~ Variety + (1 | Rep) + (1 | RowBlk) + (1 | ColBlk)
             npar  logLik    AIC    LRT Df Pr(>Chisq)
<none>         29 -825.87 1709.7
(1 | Rep)      28 -826.20 1708.4  0.665  1     0.4148
(1 | RowBlk)   28 -852.63 1761.3 53.516  1  2.565e-13 ***
(1 | ColBlk)   28 -851.48 1759.0 51.213  1  8.286e-13 ***
```

The variance component for the residuals is now much smaller, much of the variation went into the column blocks. This leads to a greater power in the multiple comparisons.
The variety has a significant influence on the yield. The effect of the replications is not significant but the effects of row blocks and column blocks are.

Adjusted treatment means, pairwise tests, and grouping of treatments

```
e.V <- emmeans ( m.2, ~Variety )
cld ( e.V, adjust="hochberg" )
test ( contrast (e.V , "pairwise"),
       adjust="hochberg" )
```

```
> cld ( e.V, adjust="hochberg" )
 Variety emmean   SE   df lower.CL upper.CL .group
 va10      1193 60.4 19.4      978     1408  1
 va01      1284 60.4 19.4     1068     1499  12
 va09      1299 60.4 19.4     1084     1514  123
 va14      1327 60.4 19.4     1111     1542  1234
 va11      1327 60.4 19.4     1112     1543  1234
 va23      1329 60.4 19.4     1114     1544  1234
 va16      1346 60.4 19.4     1131     1561  1234
 va07      1401 60.4 19.4     1185     1616  12345
 va03      1421 60.4 19.4     1206     1636  123456
 va04      1452 60.4 19.4     1237     1667   234567
 va08      1457 60.4 19.4     1242     1673   234567
 va12      1484 60.4 19.4     1269     1699   234567
 va21      1493 60.4 19.4     1278     1709   234567
 va15      1498 60.4 19.4     1283     1713   234567
 va17      1498 60.4 19.4     1283     1713   234567
 va06      1527 60.4 19.4     1312     1743    34567
 va05      1533 60.4 19.4     1318     1749    34567
 va24      1546 60.4 19.4     1331     1762     4567
 va02      1549 60.4 19.4     1334     1764     4567
 va18      1592 60.4 19.4     1377     1807      567
 va13      1619 60.4 19.4     1404     1834      567
 va25      1631 60.4 19.4     1415     1846      567
```

```
 va20      1640 60.4 19.4     1425     1855      567
 va22      1644 60.4 19.4     1429     1860       67
 va19      1670 60.4 19.4     1454     1885        7

Degrees-of-freedom method: kenward-roger
Confidence level used: 0.95
Conf-level adjustment: bonferroni method for 25 estimates
P value adjustment: hochberg method for 300 tests
significance level used: alpha = 0.05

> test ( contrast ( e.V , "pairwise" ),
+        adjust="hochberg" )
 contrast    estimate   SE df t.ratio p.value
 va01 - va02 -265.426 62.3 79 -4.258  0.0142
 va01 - va03 -137.344 62.3 79 -2.203  0.9980
 va01 - va04 -168.268 62.3 79 -2.699  0.9980
 va01 - va05 -249.688 62.3 79 -4.005  0.0341
 va01 - va06 -243.820 62.3 79 -3.911  0.0466
 va01 - va07 -117.141 62.3 79 -1.879  0.9980
 va01 - va08 -173.787 62.3 79 -2.788  0.9980
 va01 - va09  -15.272 62.3 79 -0.245  0.9980
 va01 - va10   90.363 62.3 79  1.450  0.9980
 va01 - va11  -43.658 62.3 79 -0.700  0.9980
...
 va23 - va25 -301.520 62.3 79 -4.837  0.0017
 va24 - va25  -84.159 62.3 79 -1.350  0.9980

P value adjustment: hochberg method for 300 tests
```

One sided comparisons with a control

Variety 4 is the variety that is usually used by the farmers. We would like to compare all other varieties to variety 4. `contrast()` needs the position of the reference for the comparisons, so we have to look it up first.

```
levels(barley$Variety)  # Look up treatment number of the reference

test ( contrast ( e.V, "trt.vs.ctrl",ref=4),
       side = ">",
       adjust="none")
```

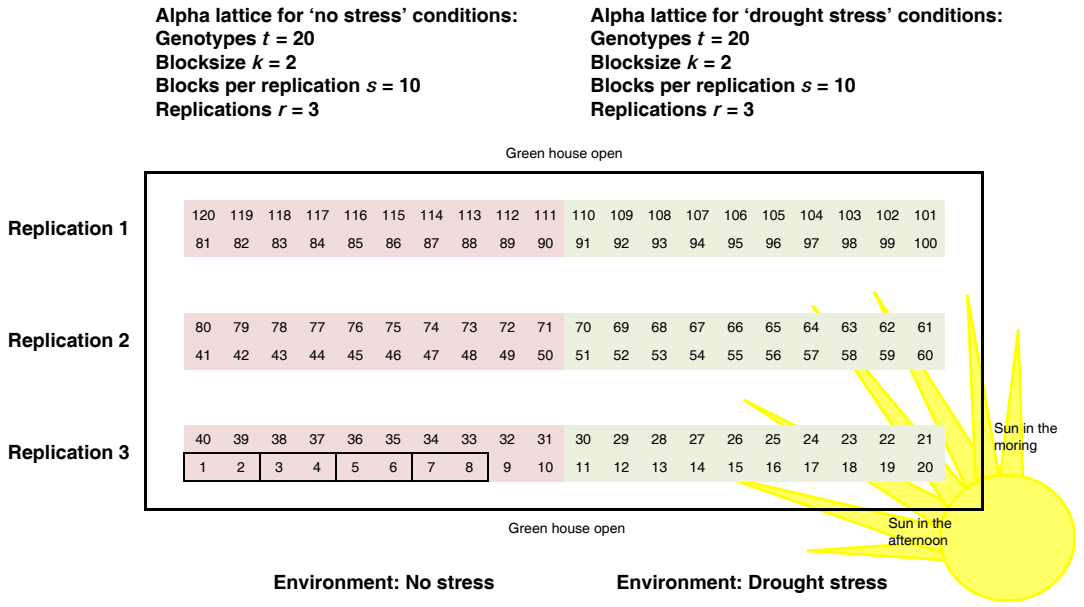
`ref=4` is not for variety 4 but for position 4 in the levels.

```
> levels(barley$Variety) # Look up treatment number of the reference
 [1] "va01" "va02" "va03" "va04" "va05" "va06" "va07" "va08" "va09" "va10"
[11] "va11" "va12" "va13" "va14" "va15" "va16" "va17" "va18" "va19" "va20"
[21] "va21" "va22" "va23" "va24" "va25"

> test ( contrast ( e.V, "trt.vs.ctrl",ref=4),
+        side = ">",
+        adjust="none")
 contrast    estimate   SE df t.ratio p.value
 va01 - va04  -168.27 62.3 79 -2.699  0.9958
 va02 - va04    97.16 62.3 79  1.559  0.0616
 va03 - va04   -30.92 62.3 79 -0.496  0.6894
 va05 - va04    81.42 62.3 79  1.306  0.0977
 va06 - va04    75.55 62.3 79  1.212  0.1146
 va07 - va04   -51.13 62.3 79 -0.820  0.7927
 va08 - va04     5.52 62.3 79  0.089  0.4648
 va09 - va04  -153.00 62.3 79 -2.454  0.9918
```

```
 va10 - va04  -258.63 62.3 79 -4.149  1.0000
 va11 - va04  -124.61 62.3 79 -1.999  0.9755
 va12 - va04    31.93 62.3 79  0.512  0.3050
 va13 - va04   167.19 62.3 79  2.682  0.0045
 va14 - va04  -125.21 62.3 79 -2.009  0.9760
 va15 - va04    46.16 62.3 79  0.740  0.2306
 va16 - va04  -105.71 62.3 79 -1.696  0.9531
 va17 - va04    46.31 62.3 79  0.743  0.2299
 va18 - va04   140.32 62.3 79  2.251  0.0136
 va19 - va04   217.70 62.3 79  3.492  0.0004
 va20 - va04   188.09 62.3 79  3.017  0.0017
 va21 - va04    41.58 62.3 79  0.667  0.2533
 va22 - va04   192.53 62.3 79  3.088  0.0014
 va23 - va04  -122.75 62.3 79 -1.969  0.9738
 va24 - va04    94.61 62.3 79  1.518  0.0665
 va25 - va04   178.77 62.3 79  2.868  0.0026

Degrees-of-freedom method: kenward-roger
P values are right-tailed
```

1.3 Alpha lattice with two factors: Rape seed

20 rape seed genotypes were tested for drought stress tolerance in a lattice design. Plant height was assessed. (Data from M. Hohmann and R. Snowdon, Univ. of Giessen.)




```
library("lmerTest"); library("emmeans"); library("multcomp")

raps <- read.table ( file="v14-u51-03.csv", sep=";",header=T,
                     stringsAsFactors = T)
str(raps)
```

```
> str(raps)
'data.frame':        120 obs. of  7 variables:
 $ rep   : Factor w/ 3 levels "r1","r2","r3": 3 3 3 3 3 3 3 3 3 3 ...
 $ blk   : Factor w/ 10 levels "b01","b02","b03",..: 6 6 7 7 8 8 9 9 10 10 ...
 $ geno  : Factor w/ 20 levels "101","102","103",..: 16 19 10 5 4 1 13 15 3 18 ...
 $ RowBlk: Factor w/ 6 levels "rb01","rb02",..: 6 6 6 6 6 6 6 6 6 6 ...
 $ ColBlk: Factor w/ 20 levels "cb01","cb02",..: 20 19 18 17 16 15 14 13 12 11 ...
 $ env   : Factor w/ 2 levels "k","s": 2 2 2 2 2 2 2 2 2 2 ...
 $ hgt   : int  208 200 184 192 203 196 201 191 200 215 ...
```

Test the fixed and random effects

```
fm <- lmer ( hgt ~ geno + env + geno:env +
             (1|rep:env) + (1|blk:rep:env) , data=raps )
anova (fm)
ranova  (fm)
```

The replication is nested in the environment and the blocks are nested within the replications within the environment.

```
> anova (fm)
Type III Analysis of Variance Table with Satterthwaite's method
         Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
geno     7296.7  384.04    19 53.355  6.4449 3.548e-08 ***
env      3069.1 3069.08     1  4.000 51.5051  0.001996 **
geno:env  881.4   46.39    19 53.355  0.7785  0.720418

> ranova (fm)
ANOVA-like table for random-effects: Single term deletions

Model:
hgt ~ geno + env + (1 | rep:env) + (1 | blk:rep:env) + geno:env
                  npar  logLik    AIC     LRT Df Pr(>Chisq)
<none>              43 -315.41 716.83
(1 | rep:env)       42 -315.56 715.13 0.29766  1    0.58535
(1 | blk:rep:env)   42 -316.90 717.80 2.97319  1    0.08465 .
```

The genotype and the environment had a significant effect on the plant height. There were no significant interactions between genotype and environment and the replications and the blocks did not have a significant effect.

Variance components

```
print ( VarCorr(fm) , comp="Variance" )
```

```
> print ( VarCorr(fm) , comp="Variance" )
 Groups      Name        Variance
 blk:rep:env (Intercept) 32.5104
 rep:env     (Intercept)  2.9625
 Residual                59.5880
```

Efficiency of the lattice

```
m.rcbd  <- lmer ( hgt ~ geno + env + geno:env +  # Analyse as
                        (1|rep:env),             # randomized complete
                  data=raps )                    # block design

print ( VarCorr(m.rcbd) , comp="Variance" )      # Error in the RCBD

(lattice.efficiency = 89.98 / 59.59 * 100)       # Ratio of errors
```

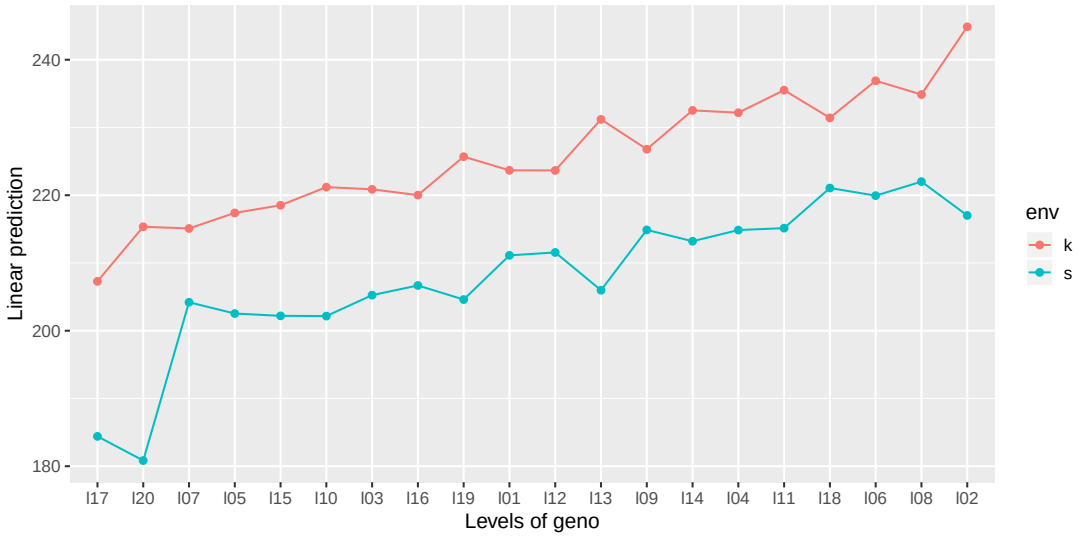
We analyse the experiment as a randomized complete block design in order to judge the efficiency of the lattice. The replications are now considered as the blocks of the lattice.

```
> print ( VarCorr(m.rcbd) , comp="Variance" )
 Groups   Name        Variance
 rep:env  (Intercept)  4.6935
 Residual             89.9813

> (lattice.efficiency = 89.98 / 59.59 * 100)
[1] 150.9985
```

Interaction plot of the adjusted entry means

It would be interesting if we had genotypes which reacted differently to drought stress than others - in other words, interactions.

```
l <- data.frame ( summary ( emmeans(fm,~geno ) ) )   # Means of genotypes
idx <- l[order(l$emmean),]$geno                      # Sort the levels of
raps$geno  <-  factor ( raps$geno , levels =  idx)   # genotypes

fm <- lmer ( hgt ~ geno + env + geno:env +           # Refit model with
                   (1|rep:env) + (1|blk:rep:env) ,   # ordered data
             data=raps )                             # set

lsmip ( fm, env ~ geno)         # Interaction plot on basis of emmeans
```



Difference between genotypes under stress

```
e.ge <- emmeans ( fm, ~geno:env, at=list(env="s") )
cld ( e.ge, adjust="none", alpha=0.1)
```

```
> cld ( e.ge, adjust="none", alpha=0.1)
 geno env emmean   SE   df lower.CL upper.CL .group
 I20  s      181 5.52 68.9      170      192  1
 I17  s      184 5.52 68.9      173      195  1
 I10  s      202 5.52 68.9      191      213   23
 I15  s      202 5.52 68.9      191      213   23
 I05  s      203 5.52 68.9      192      214   2 4
 I07  s      204 5.52 68.9      193      215   2345
 I19  s      205 5.52 68.9      194      216   23456
 I03  s      205 5.52 68.9      194      216   23456
 I13  s      206 5.52 68.9      195      217   23456
 I16  s      207 5.52 68.9      196      218   23456
 I01  s      211 5.52 68.9      200      222   234567
 I12  s      212 5.52 68.9      201      223   234567
 I14  s      213 5.52 68.9      202      224   234567
 I04  s      215 5.52 68.9      204      226   234567
 I09  s      215 5.52 68.9      204      226    3 567
 I11  s      215 5.52 68.9      204      226     4567
 I02  s      217 5.52 68.9      206      228       67
```

```
106  s      220 5.52 68.9      209      231        7
118  s      221 5.52 68.9      210      232        7
108  s      222 5.52 68.9      211      233        7
```

1.4 Exercises

1 Cropping systems

Maize yields from 14 on-farm trials in the Phalombe Project region of south-eastern Malawi. The farms were located near two different villages. On each farm, four different cropping systems were tested. The systems were: LM = Local Maize, LMF = Local Maize with Fertilizer, CCA = Improved Composite, CCAF = Improved Composite with Fertilizer. The data set `v14-u51-05.csv` contains the following variables: yield, system, village, farm. Source: Hildebrand PE (1984) Modified Stability Analysis of Farmer Managed On-Farm Trials. Agronomy Journal 76:271–274. Data set from the R package `agridat`.

(a) Visualize the data.

(b) Do the cropping system or the village have a significant effect on the yield? Do all cropping systems work equally well in both villages? Analyze the data with a mixed linear model. Assume that the cropping system, the village, and the interaction between cropping system and village are fixed, and the field is a random effect. Test for the fixed effects. Use a significance level of 0.1.

(c) What does the classification of cropping system and village as fixed and the field as random tell us about the interest of the researcher?

(d) What is the average yield for the four cropping systems? Calculate the least squares means for the cropping systems.

(e) Is there a difference between the local maize (LM) and the improved composite (CCA)? Is there a difference between local maize with fertilizer and improved composite with fertilizer? Carry out a pairwise comparison of the least square means for cropping system. Use unadjusted p-values.

(f) Is there a difference between local maize in village 2 and improved composite in village 2? Carry out a pairwise comparison of all combinations of factors system and village. Use unadjusted p-values.

2 Split plot: Oats

This exercise extends the analysis from section 1.1. Suppose that 'Victory' is the standard variety.

(a) Was is the (adjusted) entry mean of this variety?

(b) Conduct a Dunnett test to find out whether the two other varieties are better than the standard ($\alpha = 0.1$).

3 Balanced lattice in six replications: Barley

This exercise extends the analysis from section 1.2.
What is the efficiency of the model with the added column effect (`m`) compared to the classical lattice analysis (`m.1`)?

4 Mixed model analysis of experiments with block structure

Experiments that employ blocks can be analyzed with mixed linear models by assuming that the block effects are random.

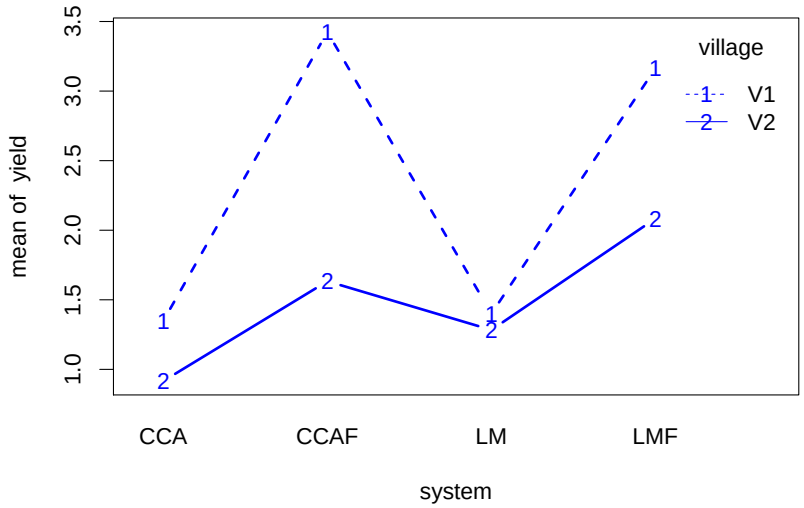
(a) Analyse the RCBD with cotton from the session on experimental designs with a mixed linear model. Data: `v14-u41-01.csv`. Conduct pairwise comparisons with an FDR adjustment. Make a compact letter display ($\alpha = 0.1$). What dimensions do the design matrices and the parameter vectors have?

(b) Analyse the Latin square with sugar beets from the session on experimental designs with a mixed linear model. Data: `v14-u41-02.csv`. Conduct pairwise comparisons with a Tukey adjustment. Make a compact letter display ($\alpha = 0.1$). What do the design matrices and the vectors with the coefficients for fixed and random effects look like?

1.5 Solutions

1 Cropping systems

(a)



(b)

```
                Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
system         24.9297  8.3099     3    36 36.9543 4.371e-11 ***
village         0.8449  0.8449     1    12  3.7571 0.0764677 .
system:village  5.6590  1.8863     3    36  8.3885 0.0002333 ***
```

The systems do have a significant influence on the yield and the villages have a significant effect, too. There are significant interactions between cropping system and village so apparently the systems do not work equally well in the villages.

(c) The researcher is interested in the effects of the particular villages and cropping systems in the experiment while the fields are considered a random sample of all the fields in the villages.

(d)

```
system emmean    SE   df lower.CL upper.CL
CCA      1.13 0.247 18.4    0.615     1.65
CCAF     2.53 0.247 18.4    2.011     3.05
LM       1.34 0.247 18.4    0.824     1.86
LMF      2.62 0.247 18.4    2.105     3.14
```

The average yields are in the column `emmean`.

(e)

```
contrast   estimate    SE df t.ratio p.value
CCA - CCAF  -1.3958 0.181 36 -7.708  <.0001
CCA - LM    -0.2083 0.181 36 -1.150  0.2575
CCA - LMF   -1.4896 0.181 36 -8.226  <.0001
CCAF - LM    1.1875 0.181 36  6.558  <.0001
CCAF - LMF  -0.0938 0.181 36 -0.518  0.6078
LM - LMF    -1.2812 0.181 36 -7.075  <.0001
```

(f)

```
contrast          estimate    SE   df t.ratio p.value
CCA V1 - CCAF V1   -2.0750 0.237 36.0 -8.752  <.0001
CCA V1 - LM V1     -0.0500 0.237 36.0 -0.211  0.8342
CCA V1 - LMF V1    -1.8125 0.237 36.0 -7.644  <.0001
CCA V1 - CCA V2     0.4333 0.494 18.4  0.878  0.3915
CCA V1 - CCAF V2   -0.2833 0.494 18.4 -0.574  0.5731
CCA V1 - LM V2      0.0667 0.494 18.4  0.135  0.8941
CCA V1 - LMF V2    -0.7333 0.494 18.4 -1.485  0.1544
CCAF V1 - LM V1     2.0250 0.237 36.0  8.541  <.0001
```

```
CCAF V1 - LMF V1    0.2625 0.237 36.0  1.107  0.2756
CCAF V1 - CCA V2    2.5083 0.494 18.4  5.079  0.0001
CCAF V1 - CCAF V2   1.7917 0.494 18.4  3.628  0.0019
CCAF V1 - LM V2     2.1417 0.494 18.4  4.337  0.0004
CCAF V1 - LMF V2    1.3417 0.494 18.4  2.717  0.0139
LM V1 - LMF V1     -1.7625 0.237 36.0 -7.434  <.0001
LM V1 - CCA V2      0.4833 0.494 18.4  0.979  0.3404
LM V1 - CCAF V2    -0.2333 0.494 18.4 -0.473  0.6421
LM V1 - LM V2       0.1167 0.494 18.4  0.236  0.8158
LM V1 - LMF V2     -0.6833 0.494 18.4 -1.384  0.1830
LMF V1 - CCA V2     2.2458 0.494 18.4  4.548  0.0002
LMF V1 - CCAF V2    1.5292 0.494 18.4  3.097  0.0061
LMF V1 - LM V2      1.8792 0.494 18.4  3.805  0.0012
LMF V1 - LMF V2     1.0792 0.494 18.4  2.185  0.0420
CCA V2 - CCAF V2   -0.7167 0.274 36.0 -2.618  0.0129
CCA V2 - LM V2     -0.3667 0.274 36.0 -1.339  0.1889
CCA V2 - LMF V2    -1.1667 0.274 36.0 -4.261  0.0001
CCAF V2 - LM V2     0.3500 0.274 36.0  1.278  0.2093
CCAF V2 - LMF V2   -0.4500 0.274 36.0 -1.644  0.1090
LM V2 - LMF V2     -0.8000 0.274 36.0 -2.922  0.0060
```

2 Split plot: Oats

(a)

```
Variety     emmean  SE   df lower.CL upper.CL
Golden Rain  104.5 7.8 8.87     86.8      122
Marvellous   109.8 7.8 8.87     92.1      127
Victory       97.6 7.8 8.87     79.9      115
```

The adjusted entry mean of the variety “Victory” is 97.625.

(b)

```
contrast              estimate   SE df t.ratio p.value
Golden Rain - Victory     6.88 7.08 10 0.971   0.2789
Marvellous - Victory     12.17 7.08 10 1.719   0.0992
```

The yield of both “Golden Rain” and “Marvellous is higher than that of “Victory” (the estimates of the differences are positive). However, only “Marvellous” has a significantly higher yield than “Victory”.

3 Balanced lattice in six replications: Barley

```
> 22677.3/8061.8 * 100
[1] 281.2933
```

The efficiency is 281 %.

4 Mixed model analysis of experiments with block structure

(a)

```
> anova(m) # Tests of fixed effects
Type III Analysis of Variance Table with Satterthwaite's method
      Sum Sq Mean Sq NumDF DenDF F value  Pr(>F)
Fert 0.73244 0.18311     4     8  4.1916 0.04037 *

> ranova (m) # Tests of random effects
ANOVA-like table for random-effects: Single term deletions

Model:
Strength ~ Fert + (1 | Blk)
          npar  logLik    AIC       LRT Df Pr(>Chisq)
<none>       7 -1.3880 16.776
(1 | Blk)    6 -1.3925 14.785 0.0091438  1     0.9238
```

The effect of the fertilizer is significant, the block effect is not.

```
contrast    estimate        SE df t.ratio p.value
f1 - f2   -0.2033 0.171  8 -1.191  0.3345
f1 - f3    0.1067 0.171  8  0.625  0.6104
f1 - f4    0.3367 0.171  8  1.973  0.2066
f1 - f5    0.4000 0.171  8  2.344  0.1571
f2 - f3    0.3100 0.171  8  1.817  0.2066
f2 - f4    0.5400 0.171  8  3.164  0.0665
f2 - f5    0.6033 0.171  8  3.535  0.0665
f3 - f4    0.2300 0.171  8  1.348  0.3067
f3 - f5    0.2933 0.171  8  1.719  0.2066
f4 - f5    0.0633 0.171  8  0.371  0.7202
```

```
Fert emmean    SE   df lower.CL upper.CL .group
f5     7.45 0.122 9.98     7.06     7.84  1
f4     7.51 0.122 9.98     7.13     7.90  1
f3     7.74 0.122 9.98     7.36     8.13  12
f1     7.85 0.122 9.98     7.46     8.24  12
f2     8.05 0.122 9.98     7.67     8.44   2
```

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \mathbf{e}$$

$$\begin{pmatrix} 7.62 \\ 8.00 \\ 7.93 \\ 8.14 \\ 8.15 \\ 7.87 \\ 7.76 \\ 7.73 \\ 7.74 \\ 7.17 \\ 7.57 \\ 7.80 \\ 7.46 \\ 7.68 \\ 7.21 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_{f1} \\ \beta_{f2} \\ \beta_{f3} \\ \beta_{f4} \\ \beta_{f5} \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_{b1} \\ u_{b2} \\ u_{b3} \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ e_9 \\ e_{10} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \\ e_{15} \end{pmatrix}$$

(b)

```
> anova(m) # Tests of fixed effects
Type III Analysis of Variance Table with Satterthwaite's method
        Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
Variety 12.622  2.5244     5    20  14.016 5.889e-06 ***

> ranova (m) # Tests of random effects
ANOVA-like table for random-effects: Single term deletions

Model:
Yield ~ Variety + (1 | Row) + (1 | Col)
          npar  logLik    AIC     LRT Df Pr(>Chisq)
<none>       9 -25.397 68.794
(1 | Row)    8 -26.428 68.856 2.06157  1     0.1511
(1 | Col)    8 -25.515 67.030 0.23556  1     0.6274
```

```
 contrast estimate    SE df t.ratio p.value
 v1 - v2    -0.583 0.245 20 -2.381  0.2098
 v1 - v3     0.000 0.245 20  0.000  1.0000
 v1 - v4     0.200 0.245 20  0.816  0.9611
 v1 - v5    -1.550 0.245 20 -6.326  <.0001
 v1 - v6    -0.100 0.245 20 -0.408  0.9983
 v2 - v3     0.583 0.245 20  2.381  0.2098
 v2 - v4     0.783 0.245 20  3.197  0.0448
 v2 - v5    -0.967 0.245 20 -3.945  0.0090
 v2 - v6     0.483 0.245 20  1.973  0.3909
 v3 - v4     0.200 0.245 20  0.816  0.9611
 v3 - v5    -1.550 0.245 20 -6.326  <.0001
 v3 - v6    -0.100 0.245 20 -0.408  0.9983
 v4 - v5    -1.750 0.245 20 -7.142  <.0001
 v4 - v6    -0.300 0.245 20 -1.224  0.8200
 v5 - v6     1.450 0.245 20  5.918  0.0001
```

```
 Variety emmean    SE   df lower.CL upper.CL .group
 v4        16.2 0.199 25.4     15.7     16.8  1
 v1        16.4 0.199 25.4     15.9     17.0  12
 v3        16.4 0.199 25.4     15.9     17.0  12
 v6        16.5 0.199 25.4     16.0     17.1  12
 v2        17.0 0.199 25.4     16.4     17.6   2
 v5        18.0 0.199 25.4     17.4     18.6    3
```

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \mathbf{e}$$

$$\begin{pmatrix} 16.2 \\ 17.0 \\ 18.1 \\ 16.6 \\ 17.7 \\ 16.3 \\ 16.0 \\ 15.3 \\ \cdots \\ 17.8 \\ 16.2 \\ 18.3 \\ 16.6 \\ 16.4 \\ 17.6 \\ 17.1 \\ 16.5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ & & & \cdots & & & \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_{v1} \\ \beta_{v2} \\ \beta_{v3} \\ \beta_{v4} \\ \beta_{v5} \\ \beta_{v6} \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ & & & & & & \cdots & & & & & \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ \cdots \\ e_{29} \\ e_{30} \\ e_{31} \\ e_{32} \\ e_{33} \\ e_{34} \\ e_{35} \\ e_{36} \end{pmatrix}$$

2 Heterogeneous error variances

2.1 Series of alpha lattices at six locations

Multi-environment yield trial of maize laid out in a lattice design. 64 corn hybrids were tested in six counties in North Carolina. For each county one location was chosen where the experiment was layed out as a lattice with three replicates. The data set `v14-u51-04.csv` contains the following variables: yield, gen (genotype, variety), county, rep (replication), block. Source: Besag J, Higdon D (1999) Bayesian Analysis of Agricultural Field Experiments, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61:691–746. Data from the R package `agridat`.

```
library("lmerTest"); library("nlme")

met <- read.table ( "v14-u51-04.csv", sep=";", dec=",", header=T,
                    stringsAsFactors = T)
str(met)
```

```
> str(met)
'data.frame':        1152 obs. of  7 variables:
 $ county: Factor w/ 6 levels "C1","C2","C3",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ row   : Factor w/ 18 levels "row01","row02",..: 16 12 4 7 14 1 3 16 9 12 ...
 $ col   : Factor w/ 11 levels "col01","col02",..: 10 9 6 11 8 7 1 3 8 1 ...
 $ rep   : Factor w/ 3 levels "R1","R2","R3": 3 2 1 2 3 1 1 3 2 3 ...
 $ block : Factor w/ 8 levels "B1","B2","B3",..: 1 1 1 2 2 2 3 3 3 4 ...
 $ yield : num  175 158 101 146 168 ...
 $ gen   : Factor w/ 64 levels "G01","G02","G03",..: 1 1 1 2 2 2 3 3 3 4 ...
```

Lattice analysis with homogeneous error variances at the locations (counties)

```
mm <- lmer ( yield ~ gen + county + gen:county
             + (1|county:rep) + (1|county:rep:block), data=met )
print ( VarCorr(mm) , comp="Variance" )
```

```
> print ( VarCorr(mm) , comp="Variance" )
 Groups           Name        Variance
 county:rep:block (Intercept)  60.257
 county:rep       (Intercept) 120.584
 Residual                     197.420
```

The same model can be fit with the fit with the `lme()` function of the `nlme` package. It can assume heterogeneous error variances for the locations which `lmerTest` cannot do.

```
met$countyrep <- met$county:met$rep
levels(met$countyrep)

m.lme1 <- lme ( fixed   = yield ~ gen + county + gen:county,
                random  = ~ 1|countyrep/block,
                data    = met )

print ( VarCorr(m.lme1) )
```

```
> print ( VarCorr(m.lme1) )
            Variance     StdDev
countyrep = pdLogChol(1)
(Intercept) 120.58569    10.981151
block =     pdLogChol(1)
(Intercept)  60.25696     7.762536
Residual    197.41992    14.050620
```

With the `weights` option the model uses heteroscedastic error variances.

```
m.lme2 <- lme ( fixed   = yield ~ gen + county + gen:county,
                random  = ~ 1|countyrep/block,
                weights = varIdent (form=~1|county),  # Heteroscedacity
                data    = met )
```

Each county gets its own error variance, the residual variance printed in the overview is the error variance in of the first level of county.

```
print ( VarCorr(m.lme2) )
```

```
> print ( VarCorr(m.lme2) )
            Variance     StdDev
countyrep = pdLogChol(1)
(Intercept) 119.12590    10.914481
block =     pdLogChol(1)
(Intercept)  58.56319     7.652659
Residual    149.14597    12.212533
```

The ratios of the standard deviations at the counties can be obtained with:

```
m.lme2$modelStruct$varStruct
```

```
> m.lme2$modelStruct$varStruct
Variance function structure of class varIdent representing
       C1        C2        C3        C4        C5        C6
1.0000000 1.1582761 0.9913856 1.2647221 0.9053222 1.4917732
```

These ratios can be used to calculate the error variances at the counties:

```
vc <- data.frame(VarCorr(m.lme2)[,])
v.e <- as.numeric(as.character(vc["Residual"==rownames(vc),"Variance"]))
vw <- varWeights(m.lme2$modelStruct$varStruct)
tt <- unique(data.frame(comp=names(vw),w=vw))
tt$var <- 1/tt$w/tt$w * v.e
tt[,-2]
```

```
> tt[,-2]
    comp      var
1     C1 149.1460
193   C2 200.0943
385   C3 146.5875
577   C4 238.5624
769   C5 122.2415
961   C6 331.9079
```

The newer `sommer` package can also be used for this task.

```
library("sommer")

m2 <- mmer ( fixed  = yield ~ gen + county + gen:county ,
             random = ~ county:rep + county:rep:block ,
             rcov   = ~ vs(ds(county),units),          # Heteroscedasticity
             data=met, verbose = FALSE)

summary(m2)$varcomp
detach(package:sommer)
```

```
> summary(m2)$varcomp
                              VarComp VarCompSE   Zratio Constraint
county:rep.yield-yield       119.12849  52.89702 2.252083   Positive
county:rep:block.yield-yield  58.55889  11.90284 4.919740   Positive
C1:units.yield-yield         149.14506  20.41373 7.306115   Positive
C2:units.yield-yield         200.10574  27.27392 7.336889   Positive
C3:units.yield-yield         146.59077  20.07087 7.303658   Positive
C4:units.yield-yield         238.55898  32.40302 7.362244   Positive
C5:units.yield-yield         122.24083  16.76788 7.290177   Positive
C6:units.yield-yield         331.90658  44.74932 7.417020   Positive
```

The result is the same. `sommer` has advantages over the other packages in terms of flexibility and speed but it is still under heavy development, and the `emmeans` package supports `sommer` only partially. This is why we go back to our analysis with `nlme`.
The homoscedastic and heteroscedastic models fitted with `nlme` can be compared with `anova` to check whether the error variances of the models are significantly different.
A test of fixed effects can also be carried out with `anova`.

```
anova ( m.lme1, m.lme2 )  # Compare homoscedastic with heteoscedastic
anova ( m.lme2 )          # Test fixed effects
```

```
> anova ( m.lme1, m.lme2 ) # Compare homoscedastic with heteoscedastic model
       Model  df      AIC     BIC    logLik   Test L.Ratio p-value
m.lme1     1 387 7601.233 9398.38 -3413.616
m.lme2     2 392 7572.684 9393.05 -3394.342 1 vs 2  38.549  <.0001

> anova ( m.lme2 ) # Test fixed effects
            numDF denDF   F-value p-value
(Intercept)     1   630 1586.8849  <.0001
gen            63   630    5.2534  <.0001
county          5    12   30.5556  <.0001
gen:county    315   630    1.2205  0.0191
```

Adjusted entry means and comparisons

```
library("emmeans")
library("multcomp")
e.g <- emmeans ( m.lme2, ~ gen )
cld (e.g, adjust="none", alpha=0.1)
```

```
> cld (e.g, adjust="none", alpha=0.1)
 gen emmean   SE df lower.CL upper.CL .group
 G57   88.4 4.37 12     78.8     97.9  1
 G29   90.8 4.37 12     81.2    100.3  12
```

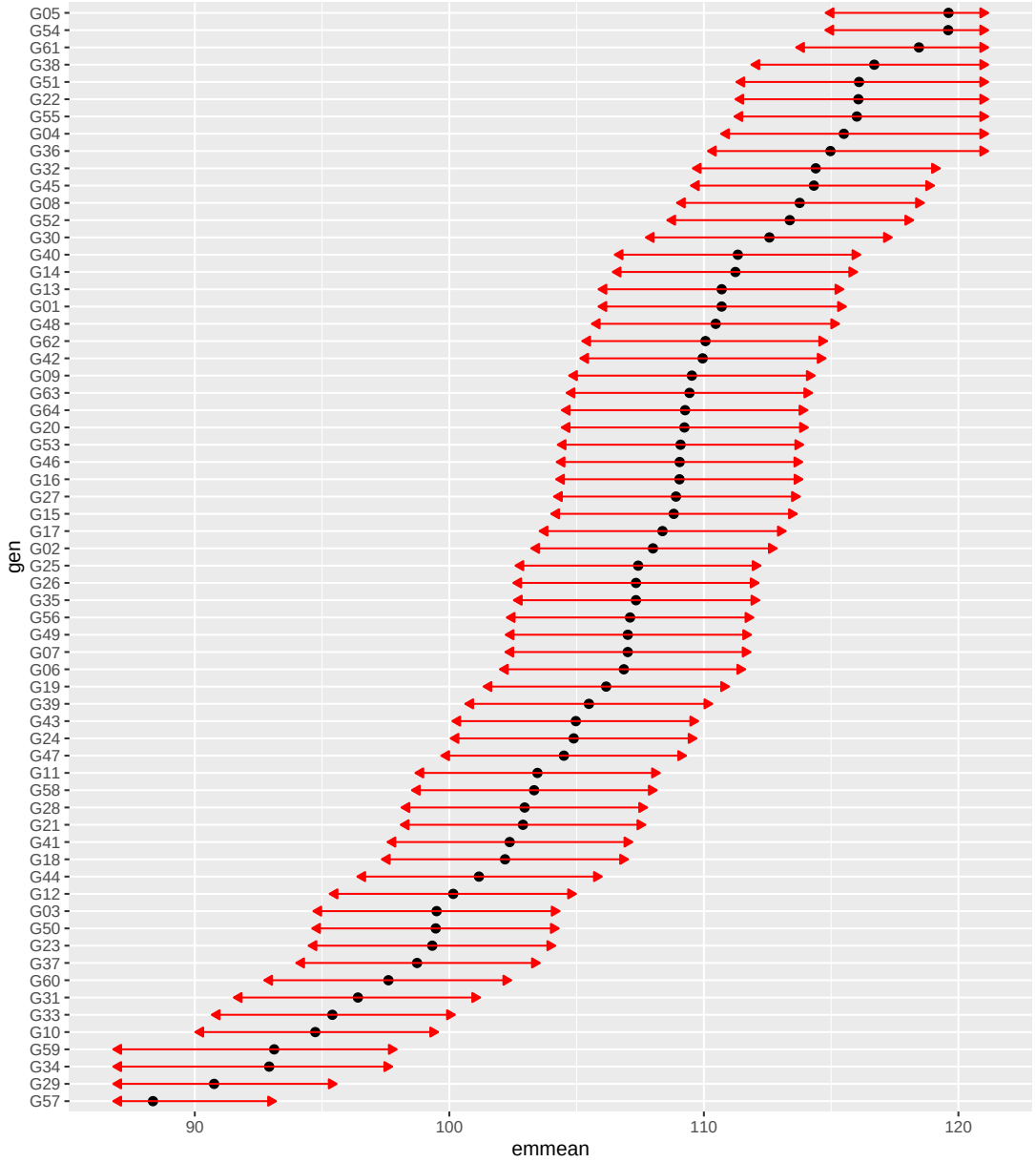
```
 G34   92.9 4.37 12     83.4    102.4  123
 G59   93.1 4.37 12     83.6    102.6  1234
...
```

To generate a sorted plot of the emmeans, the factor levels of gen are sorted according to the emmean. Then the model is refit and the emmeans are calculated with ordered factor levels of the genotypes.

```
emt <- summary(e.g)
met$gen <- factor(met$gen,levels=emt$gen[order(emt$emmean)])
m.lme2 <- lme ( fixed  = yield ~ gen + county + gen:county,
                random = ~ 1| countyrep/block,
                weights = varIdent (form=~1|county),
                data   = met )
e.g <- emmeans ( m.lme2, ~ gen )
```

The emmeans can now be ploted with the factor levels ordered. Non-overlapping arrows indicate significantly different genotypes.

```
plot ( e.g, comparisons=TRUE, intervals=FALSE, adjust="none", alpha=0.1)
```

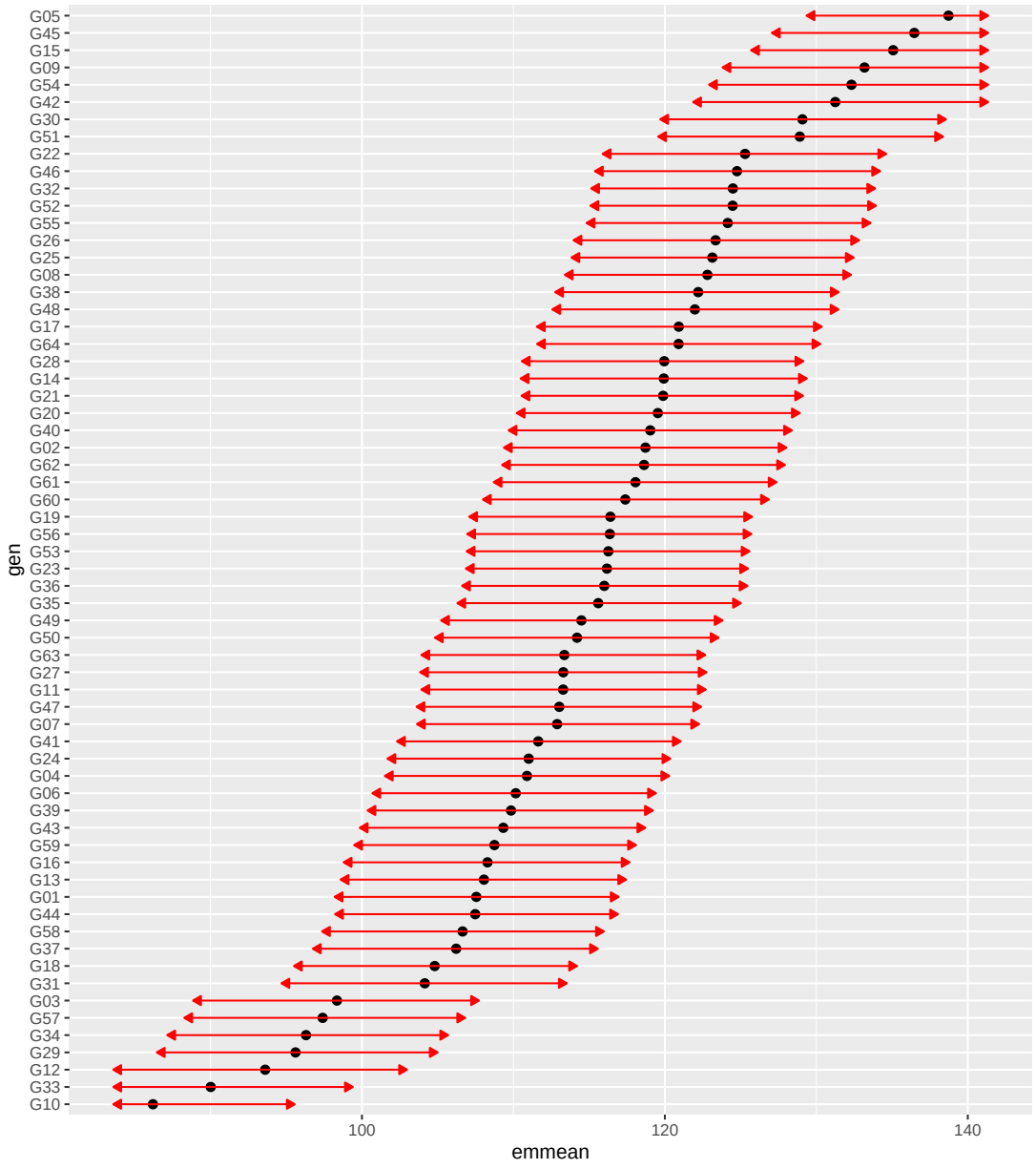

G05
G54
G61
G38
G51
G22
G55
G04
G36
G32
G45
G08
G52
G30
G40
G14
G13
G01
G48
G62
G42
G09
G63
G64
G20
G53
G46
G16
G27
G15
G17
G02
G25
G26
G35
G56
G49
G07
G06
G19
G39
G43
G24
G47
G11
G58
G28
G21
G41
G18
G44
G12
G03
G50
G23
G37
G60
G31
G33
G10
G59
G34
G29
G57
gen
90
100
110
120
emmean

Comparison of the treatments in county #5:

```
e.c5 <- emmeans ( m.lme2, ~ gen, at=list(county="C5") ) # County #5
cld ( e.c5, adjust="none", alpha=0.1 )
```

Reorder factor leves, refit the model and plot emmeans:

```
emt <- summary(e.c5)
met$gen <- factor(met$gen,levels=emt$gen[order(emt$emmean)])
m.lme2 <- lme ( fixed  = yield ~ gen + county + gen:county,
                random = ~ 1| countyrep/block,
                weights = varIdent (form=~1|county),
                data   = met )
e.c5 <- emmeans ( m.lme2, ~ gen, at=list(county="C5") ) # County #5

plot ( e.c5, comparisons=TRUE, intervals=FALSE, adjust="none", aplha=0.1)
```



2.2 Exercises

1 Rye

Chromosome segments from the primitive rye 'Altev' were introgressed into the genetic background of the elite line 'L2053'. The test cross performance of the introgression lines 'L2101'–'L2180' was assesed with in a lattice design with two replications at three locations. The data set contains the following variables: WUH (plant height), gen (genotype), loc (location), rep (replication), blk (block). Source: Falke et al. (2008) Establishment of introgression libraries in hybrid rye (Secale cereale L.) from an Iranian primitive accession as a new tool for rye breeding and genomics. Theor Appl Genet 117:641–652. Data from Dr. Th. Miedaner Univ. of Hohenheim. The data are stored in the internal R format. Read in the data with

```
load("v14-u51-06.Rdta")
str(rye)
```

To remove data lines with missing values use

```
rye <- subset ( rye, !is.na(rye$WUH) )
```

(a) Visualize the data with a boxplot. Use the help system to find out how to format the boxplot such that the boxes are horizontal instead of vertical. Graphic parameters can be set using the command 'par': Use the help system to find out how you can use par to get horizontal labels for the lines on the y-axis.

(b) The 'str' command shows that the genotype is an ordered factor. Use the help for 'factor' to find out what an ordered factor is and why it is useful for the present data set.

(c) Use the 'table' command to get the following overview of the data. What do the numbers mean?

	T2*Altev	T2*L2053	T2*2101	T2*2102	T2*2103	T2*2104	T2*2105	T2*2106
EWE555	4	20	2	2	2	2	2	2
HOH555	4	19	2	2	2	2	2	2
OLI555	4	20	2	2	2	2	2	2

(d) The experimental design is an alpha lattice with two replications that was planted at three locations. Consider genotype and location as fixed effects. Test whether a model with heterogeneous variances at the three locations fits the data better then a homoscedastic model.

(e) Is there a significant genotype by environment interaction?

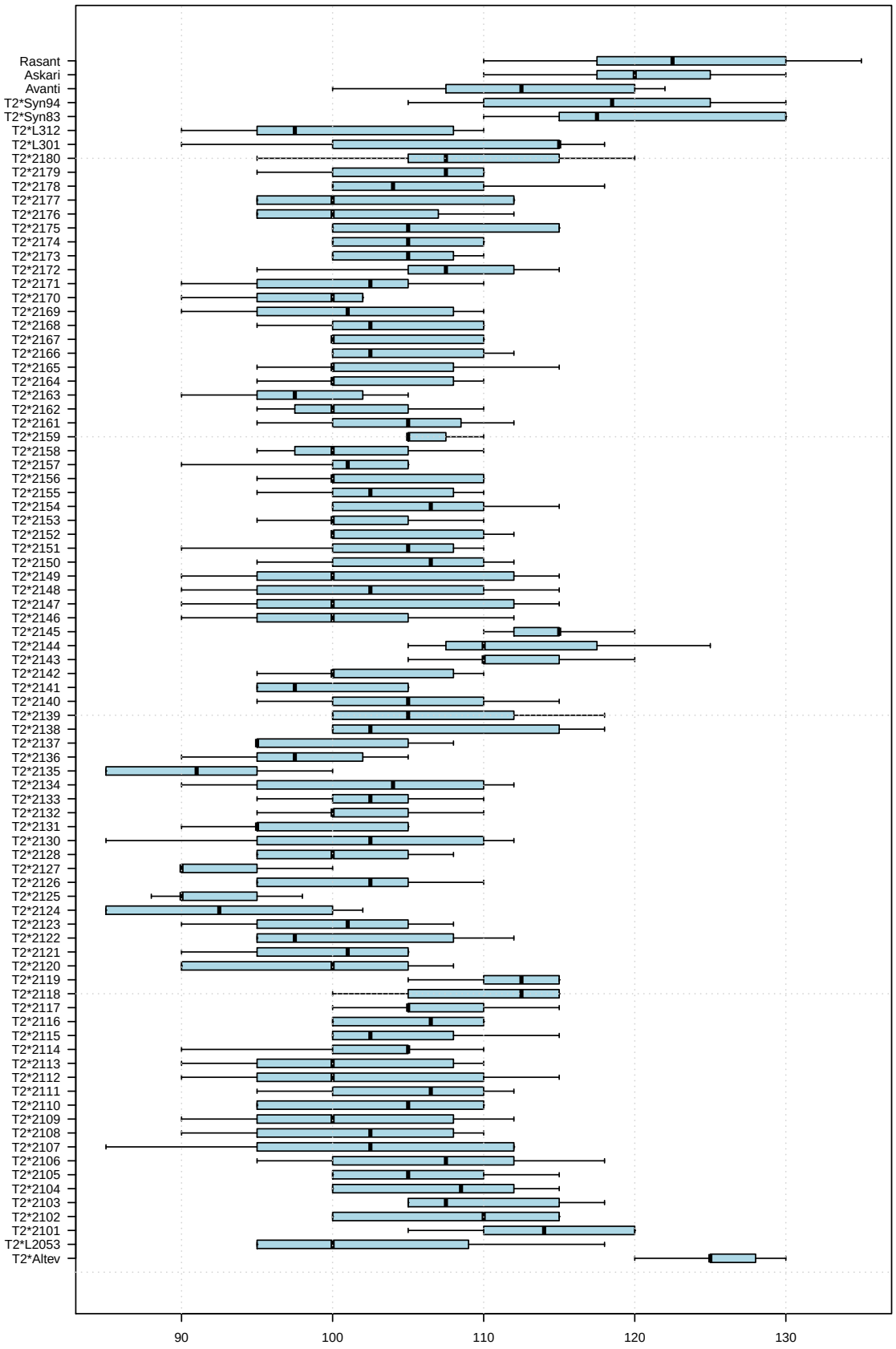
(f) Calculate the adjusted treatment means of the genotypes over all three locations.

(g) Short plant height is favorable and the elite line 'L2053' is smaller than the primitive rye 'Altev'. If there are introgression lines that are smaller than the elite line, then the primitive rye possesses alleles that shorten plant heigt, even if the primitive material itself is higher than the elite material. Are there introgression lines that are significantly smaller than the elite line? To answer this question, compare the plant height for 'T2*L2053' with all other treatments with a one sided test and adjust for multiple testing by employing a false discovery rate of 0.05. Note that 'T2*L2053' (the control) is the second genotype.

2.3 Solutions

1 Rye

(a)



(b) Normally, factor levels are ordered alphanumerically. However, with ordered factors, the given order determines their sequence. This is helpful here because the first two factor levels are the ones that all the others are supposed to be compared to: "T2*Altev" is the cross between the tester and the primitive rye "Altev" and "T2*L2052" is the cross between the tester and the elite line.

(c) The numbers are the number of times the genotypes were replicated at each location.

(d)
```
> anova ( m.lme1, m.lme2 )
       Model  df      AIC      BIC    logLik   Test   L.Ratio p-value
m.lme1     1 264 2409.104 3403.941 -940.5522
m.lme2     2 266 2412.643 3415.016 -940.3215 1 vs 2 0.4613432   0.794
```

The model with the heterogeneous error variances is not significantly better than the simpler model.

(e)
```
> anova(m.lme1)
            numDF denDF   F-value p-value
(Intercept)     1   263 20402.280  <.0001
gen            86   263    30.131  <.0001
loc             2     3    24.616  0.0138
gen:loc       172   263     1.501  0.0015
```

The GxE interaction is significant. The genotypes perform differently in the three environments.

(f)
```
 gen      emmean    SE df lower.CL upper.CL
 T2*Altev  125.4 1.157  3    121.7    129.1
 T2*L2053  102.3 0.833  3     99.7    105.0
 T2*2101   114.3 1.477  3    109.6    119.0
 T2*2102   108.2 1.478  3    103.5    112.9
 T2*2103   109.3 1.480  3    104.6    114.0
 T2*2104   106.6 1.473  3    101.9    111.3
 T2*2105   106.6 1.476  3    101.9    111.3
 T2*2106   106.5 1.479  3    101.8    111.2
 T2*2107   101.3 1.478  3     96.6    106.0
 T2*2108   101.5 1.481  3     96.8    106.2
 T2*2109   101.6 1.483  3     96.9    106.3
 T2*2110   102.7 1.480  3     98.0    107.4
 T2*2111   104.4 1.478  3     99.7    109.1
 ...
 Avanti    113.3 1.175  3    109.6    117.1
 Askari    120.6 1.158  3    117.0    124.3
 Rasant    122.1 1.157  3    118.4    125.8
```

(g)
```
 contrast                estimate    SE  df t.ratio p.value
 (T2*Altev) - (T2*L2053)  23.0811 0.994 263 23.216  1.0000
 (T2*2101) - (T2*L2053)   11.9106 1.354 263  8.794  1.0000
 (T2*2102) - (T2*L2053)    5.8853 1.355 263  4.344  1.0000
 (T2*2103) - (T2*L2053)    6.9281 1.343 263  5.160  1.0000
 (T2*2104) - (T2*L2053)    4.2638 1.346 263  3.167  1.0000
 (T2*2105) - (T2*L2053)    4.2325 1.356 263  3.122  1.0000
 (T2*2106) - (T2*L2053)    4.1830 1.365 263  3.065  1.0000
 (T2*2107) - (T2*L2053)   -1.0856 1.349 263 -0.805  0.7552
 (T2*2108) - (T2*L2053)   -0.8532 1.363 263 -0.626  0.9018
 (T2*2109) - (T2*L2053)   -0.7420 1.365 263 -0.544  0.9018
 (T2*2110) - (T2*L2053)    0.3153 1.367 263  0.231  1.0000
 (T2*2111) - (T2*L2053)    2.0128 1.363 263  1.477  1.0000
 (T2*2112) - (T2*L2053)   -0.5627 1.363 263 -0.413  0.9746
 (T2*2113) - (T2*L2053)   -2.1443 1.357 263 -1.580  0.3538
...
```

Appendix

The file `v14-u11-00.R` contains a routine for diagnostic plots to check the resiuals for model assumptions (adopted from the corresponding Genstat routine.)

```
check.residuals <- function ( m )  {
  res <- residuals ( m )
  fit <- fitted    ( m )
  par(mfrow=c(2,2),mar=c(4,4,4,1))
  hist  ( res , probability=T, main="Histogram" )
  x <- seq(min(res),max(res),length.out=100)
  lines ( x,dnorm(x,mean(res),sd(res)),lwd=2,col="darkgreen")
  plot  ( res~fit, main="Fitted-values" )
  lines ( c(min(fit),max(fit)) , c(0,0),col="darkgreen",lwd=2 )
  qqnorm( res, main="QQ-Normal plot" )
  qqline( res, col="darkgreen", lwd=2 )
  plot  ( res, main="Index plot" )
  lines (lowess((1:length(fit)), res),col="darkgreen",lwd=2 )
}
```

Installation of R und RStudio

Carola Zenke-Philippi und Matthias Frisch

1 R and RStudio

R is a software for statistical analyses and graphics. It is open-source and freely available from the internet. R is extensible and there exist many additional software packages that extend the functionality of the R core software.

RStudio is a graphical user interface (GUI) for R, which is available for Windows, Linux, and MacOS. It provides a uniform look and a uniform handling for the three operating systems.

R and RStudio are constantly being developed and expanded, and different software versions might use different syntax, might use different mathematical solutions to the same problem, and might give different results.

> *It is therefore important that you install R exactly as described in this handout. Only then our examples are guaranteed to work and give the correct results.*

Do not use older versions of R from previous classes or downloads from another webpage.

2 Uninstalling old versions of R and RStudio

If you already had R and RStudio installed, these *must* be uninstalled first. Under Windows, it is also required that you delete following folders if they exist:

- the folder `C:\Programme\R`
- the folder `C:\Programme (x86)\R`,
- the folder `C:\Users\zenke-c\Documents\R`
  (replace `zenke-c` by your own user name).

3 Downloading R and RStudio

We use **R 4.2.1** or higher. It can be downloaded from

> `https://ftp.gwdg.de/pub/misc/cran/`

– please choose the platform you are using.

RStudio Desktop (Open Source License) Version 2022.07.2+576 or higher can be downloaded from

> `www.rstudio.com/products/rstudio/download/#download`

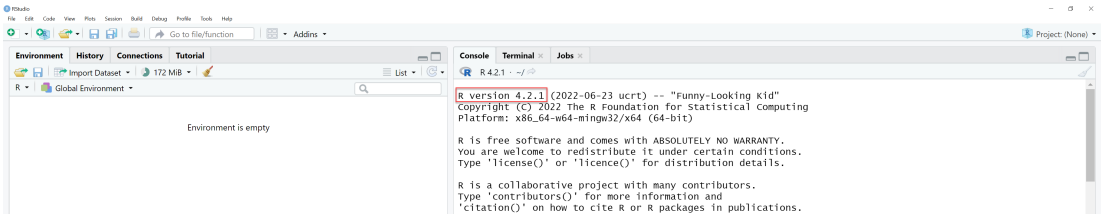
Choose the free Desktop version for your platform.

Note that we tested our R code with **R 4.2.1** and the R packages available at the beginning of October 2022. We cannot guarantee that everything also works with newer versions so please complete the installation as early in the semester as possible.

4 Installing R and RStudio

R must be installed before RStudio is installed. During the installation of R use the default settings.

For the installation of RStudio you use the default settings. After the installations are complete, start RStudio and check whether the startup message contains the information that you are using **R 4.2.1** or higher:



5 Installing extension packages

Make sure that you read the following instructions carefully and follow them step by step.

We need a set of R extension packages, which is installed with one of the following scripts:

`v14-c02.R` for courses MK-002, MK-002-EN, and MK-002-EN-DI

You can find the script in Stud.IP. The installation process is the same for all operating systems. Make sure you are connected to the internet.

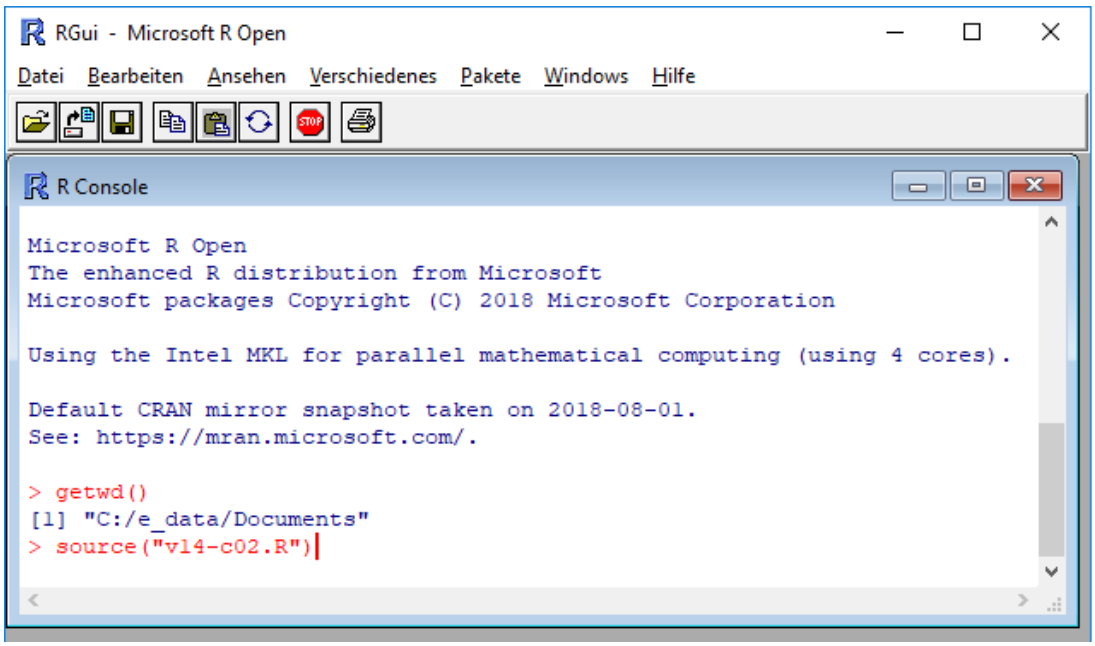
Click right on the R icon on your desktop (*not* the RStudio icon).

Execute R as admin.

Type the command `getwd()` in the console after the symbol “>” and press “Enter”.

Copy the file `v14-c02.R` in the folder that is displayed.

Then use the command `source("v14-c02.R")` to execute the script by pressing “Enter”. The following picture shows an example.

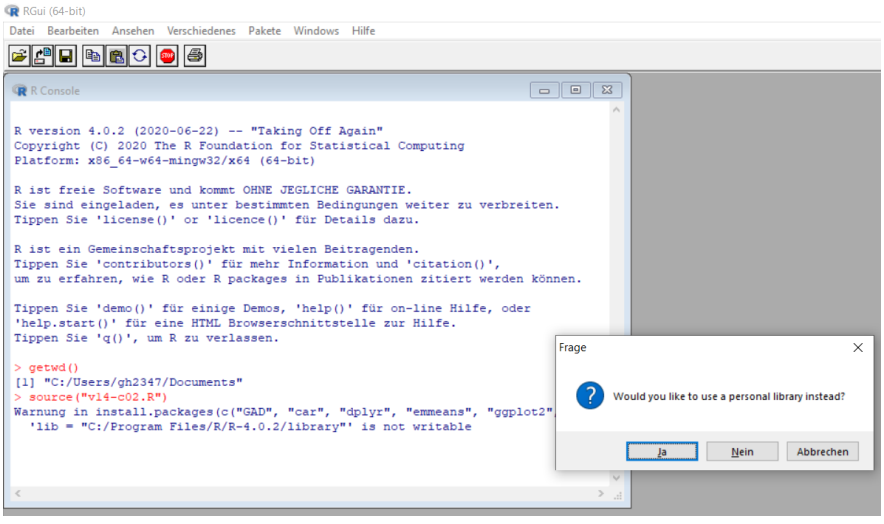


Here, the file `v14-c02.R` must be copied into the folder

```
"C:/e_data/Documents"
```

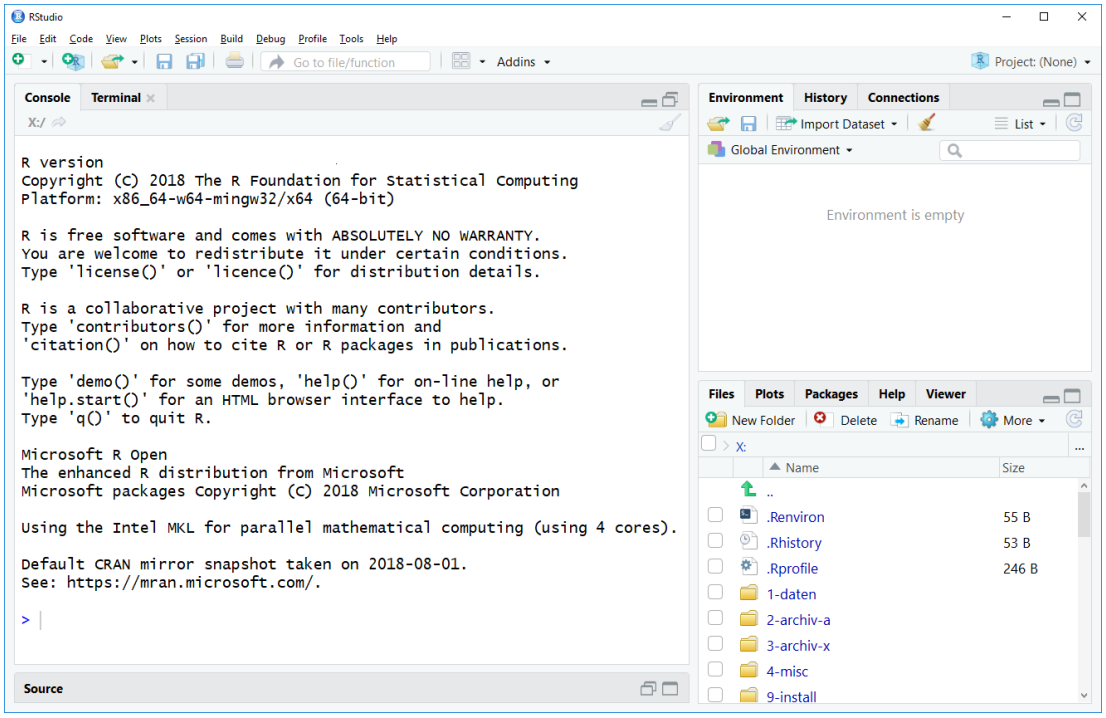
When the symbol ">" appears in the console again, the installation process is finished. Now clean the console by pressing "Ctrl + L". Check whether the installation has worked by repeating all the steps above for the R scripts `v14-c03.R` (MK-002, MK-002-EN, and MK-002-EN-DI). If you do not receive any error messages (you may have to scroll upwards in the console), then your installation was successful.

Troubleshooting: If you receive the following error message, you forgot to execute R Gui as admin. Please shut the R window and start the package installation from step 1.

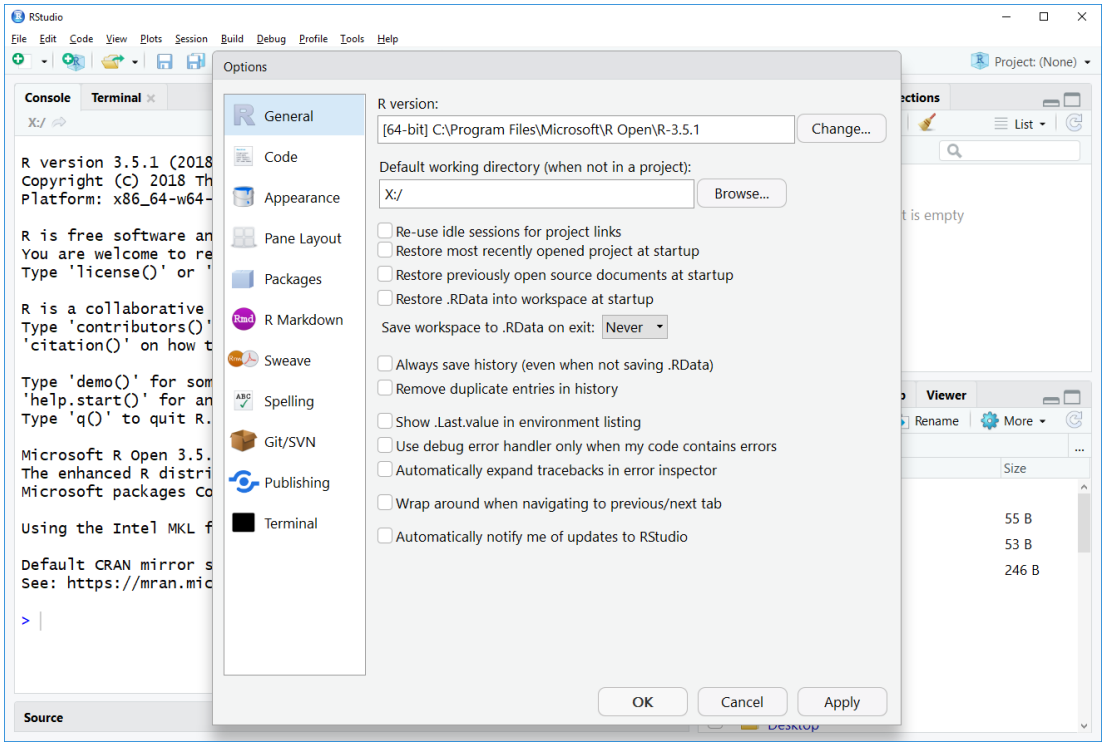


6 Starting and adjusting RStudio

When RStudio starts for the first time after the installation, it opens three windows:



In the menu under **"Tools → Global Options"** the appearance and options of RStudio can be set up.



Remove all check marks in the General section, select "Never" at "Save workspace to .RData on exit". With this setting, data from a session is always discarded on closing RStudio. If the defaults are kept, then an R session is saved on exit and reloaded the next time RStudio is started (which will at best cause no trouble and at worst spoil your analyses, and restore any errors you have ever made for all eternity).

If you are using Windows and see only a white screen when first starting RStudio, execute RStudio as administrator once. You can find this option by clicking right on the desktop icon.

R packages

Carola Zenke-Philippi und Matthias Frisch

The file `v14-c02.R` has the following content:

```
.libPaths("")

repos <- "https://ftp.gwdg.de/pub/misc/cran/"

install.packages(c("GAD",
                   "car",
                   "dplyr",
                   "emmeans",
                   "ggplot2",
                   "knitr",
                   "lattice",
                   "lme4",
                   "lmerTest",
                   "lsmeans",
                   "MASS",
                   "MCPMod",
                   "multcomp",
                   "multcompView",
                   "mvtnorm",
                   "nlme",
                   "pbkrtest",
                   "shiny",
                   "sommer"),
                 dependencies = T,
                                repos = repos)
```

The file `v14-c03.R` has the following content:

```
library("GAD", warn.conflicts=F)
library("car", warn.conflicts=F)
library("dplyr", warn.conflicts=F)
library("emmeans", warn.conflicts=F)
library("ggplot2", warn.conflicts=F)
library("knitr", warn.conflicts=F)
library("lattice", warn.conflicts=F)
library("lme4", warn.conflicts=F)
library("lmerTest", warn.conflicts=F)
library("lsmeans", warn.conflicts=F)
library("MASS", warn.conflicts=F)
library("MCPMod", warn.conflicts=F)
library("multcomp", warn.conflicts=F)
library("multcompView", warn.conflicts=F)
library("mvtnorm", warn.conflicts=F)
library("nlme", warn.conflicts=F)
library("pbkrtest", warn.conflicts=F)
library("shiny", warn.conflicts=F)
library("sommer", warn.conflicts=F)
```

RStudio Server

Screenshot Instructions
WiSe 2022/23

The RStudio server is intended for students who cannot install R or the required packages on their own PCs or who only have a tablet. Everyone else can ignore these instructions.

1 How to get a user account

You can get a user account for the RStudio server from Mr. Hans Sachs. Send him an e-mail requesting an account together with a photo of your student ID card or a photo of your confirmation of student status/matriculation certificate (from which he can see your name and your matriculation number - file format: PDF, PNG or JPG/JPEG) and the ID of your course (MK-002-EN, MK-002-EN-DI, or MK-119-EN). Mr. Sachs will then send you your username and password. His e-mail address is hans.sachs@agrar.uni-giessen.de

Note that Mr. Sachs will only reply to e-mails sent from your official university e-mail account.

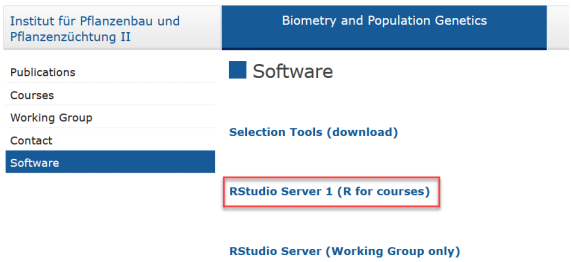
Caution:

- The server account will be deleted after one year without notification.
- Maintenance may require us to reboot the server. This may result in loss of data.

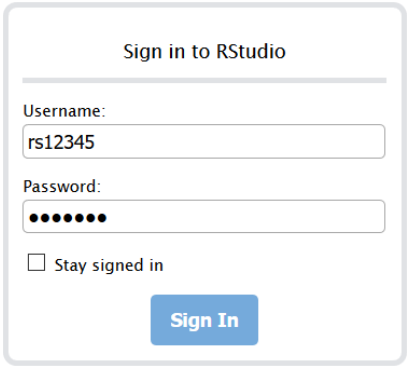
Save a regular backup of all your data on your local hard drive or USB stick.

2 Log-in on the RStudio server

The RStudio server works in the web browser. Use this link to log in: rstudio-srv1.uni-giessen.de. Click on the blue link “Sign in to RStudio”.



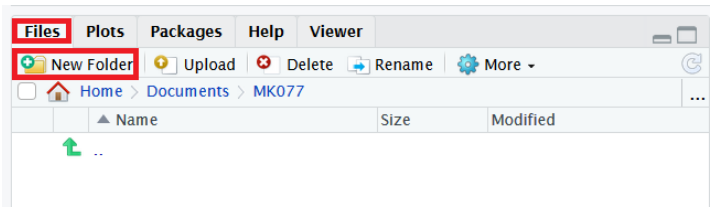
Enter your user name and password into the window displayed on the right.

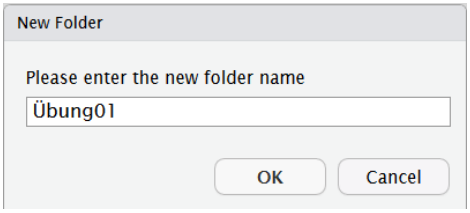


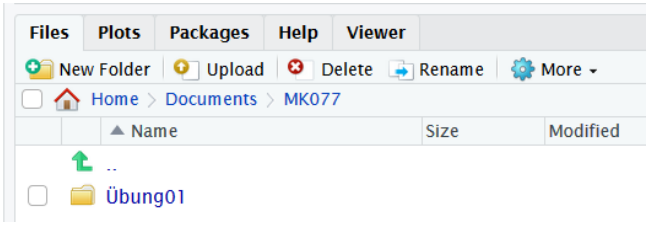
RStudio opens. Use **"Tools → Global Options"** to set it up as described in the instruction for the installation.

3 Creating a new folder

Create a new folder on the server by clicking the button **"New Folder"** in the RStudio window **"Files"**:

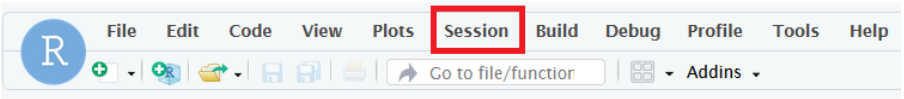


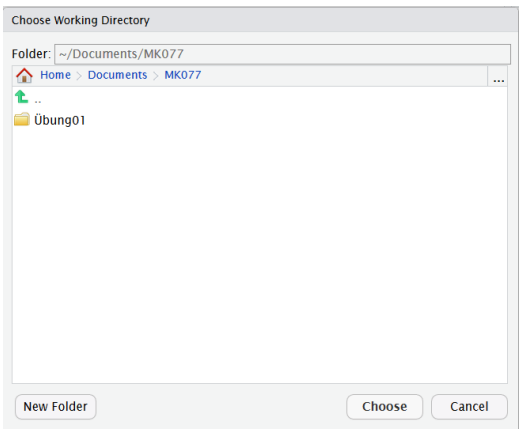




4 Setting the working directory

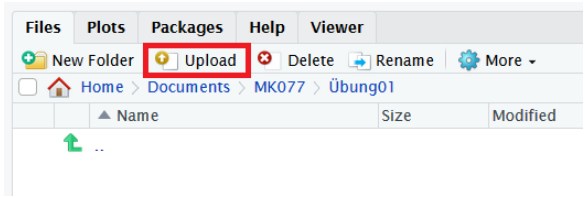
You can set the working directory to your new folders by clicking on **"Session → Set Working Directory → Choose Directory"**.

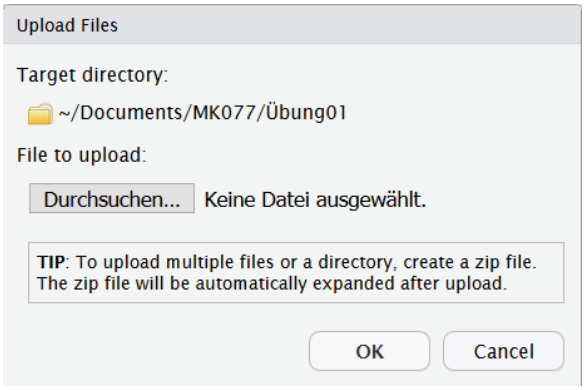


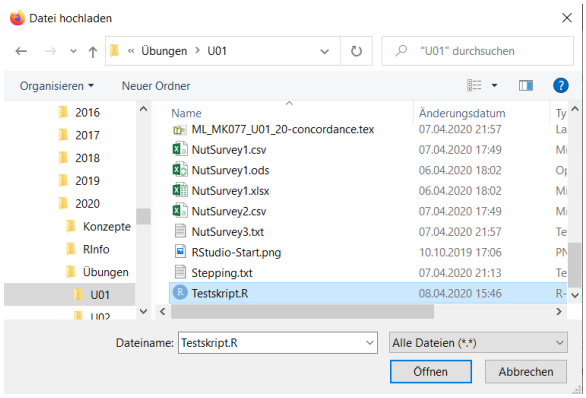


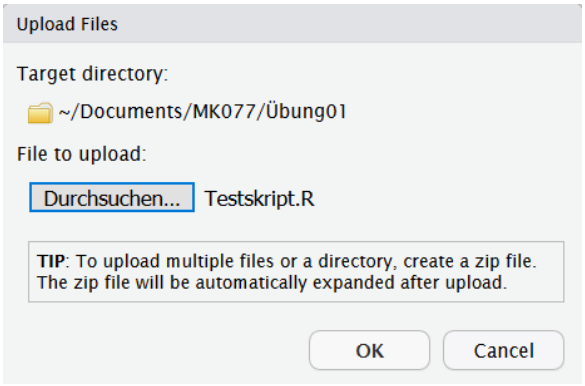
5 Importing files

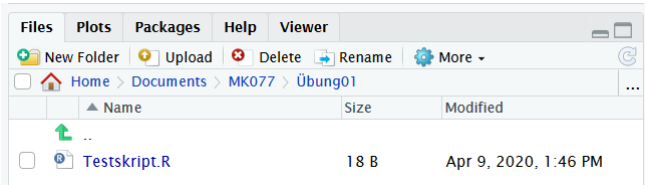
You need to upload data files and R scripts from your local PC to the server in order to work with them. It is not possible to get access to your local files via the server!
Select the folder into which you want to upload the files in the "'Files" window. Then click on "Upload".



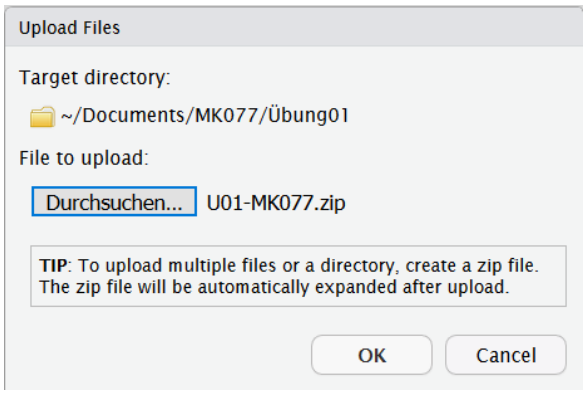


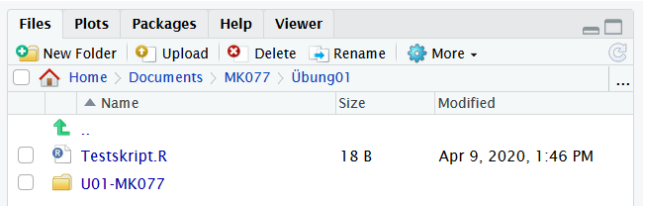






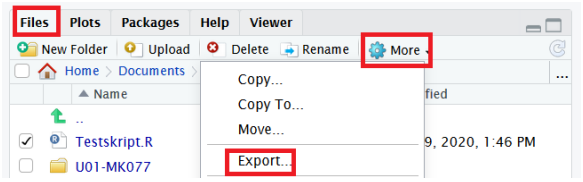
.zip folders are automatically unzipped when uploaded and appear as new folders:

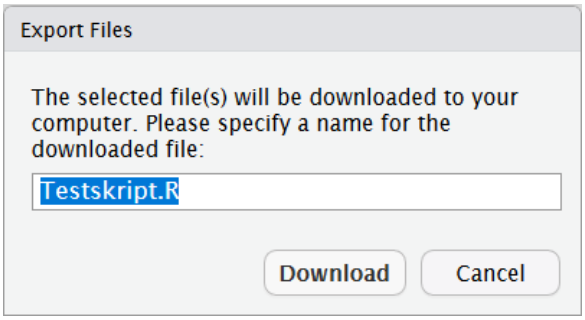


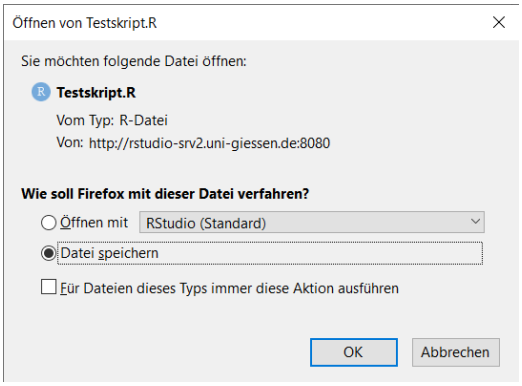


6 Downloading files from the server

Select the file(s) (put a check mark), click on "More → Export → Download" in the "Files" window and save the file(s) on your computer.

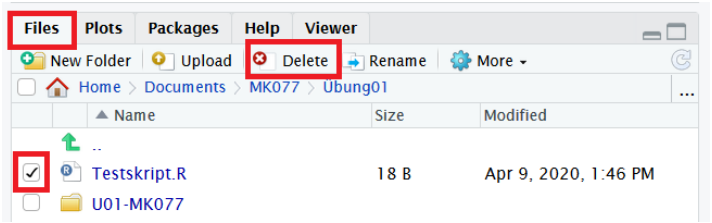


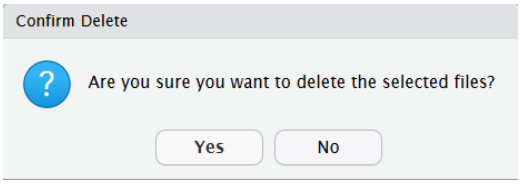




7 Deleting files

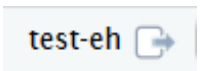
Select the file(s) (put a check mark), click "Delete" in the "Files" window and confirm that you want to delete the file(s).





8 Log-out

Log out by clicking on the little arrow next to your username in the upper right corner:

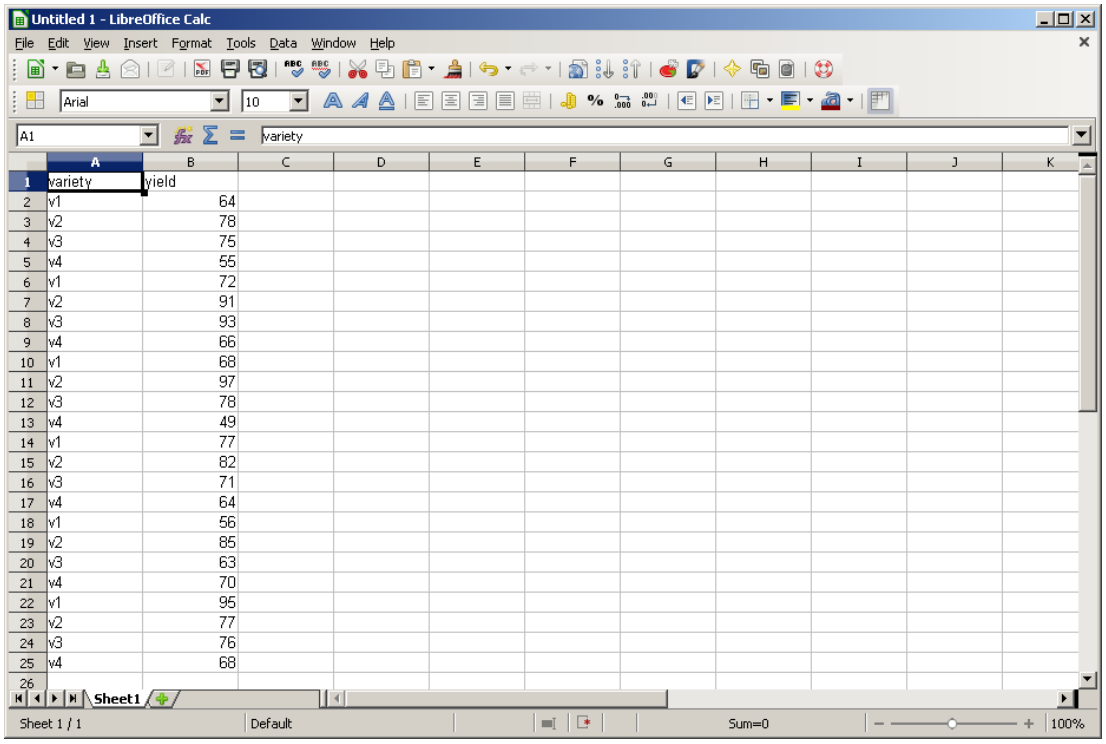


Exercises: R and LibreOffice, Data Import and Export

Matthias Frisch

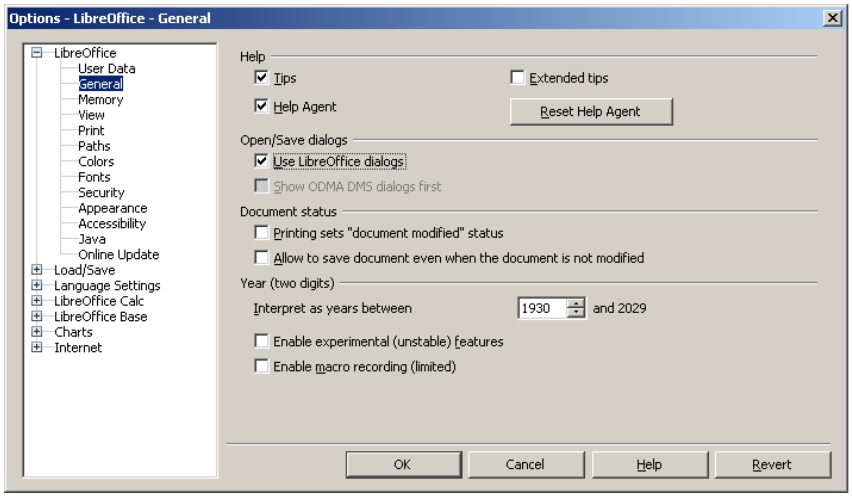
Using LibreOffice to prepare data sets for import in R

Open a new spreadsheet and enter the data. To avoid mistakes in the data analysis, precede factor levels always with a character. For example, do not name the four varieties just 1, 2, 3, 4, but name them v1, v2, v3, v4.

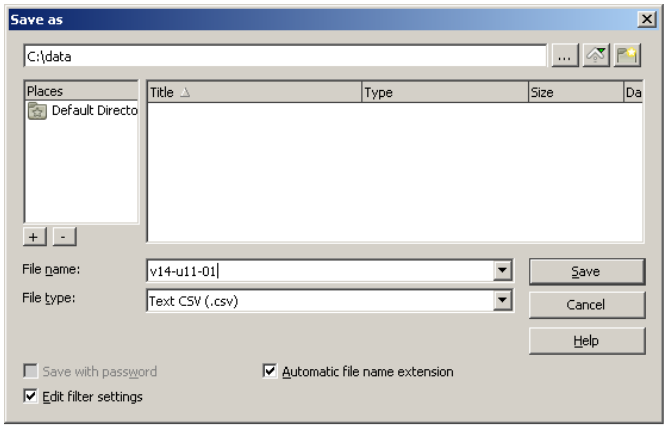


	A	B
1	variety	yield
2	v1	64
3	v2	78
4	v3	75
5	v4	55
6	v1	72
7	v2	91
8	v3	93
9	v4	66
10	v1	68
11	v2	97
12	v3	78
13	v4	49
14	v1	77
15	v2	82
16	v3	71
17	v4	64
18	v1	56
19	v2	85
20	v3	63
21	v4	70
22	v1	95
23	v2	77
24	v3	76
25	v4	68
26		

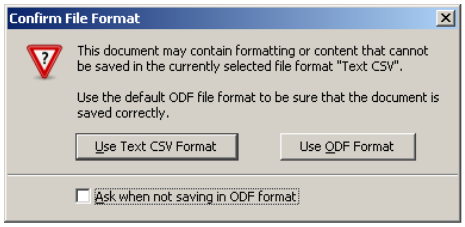
Select ‘Tools/Options’, click on ‘LibreOffice/General’, check the checkbox ‘Use LibreOffice dialogs’, click OK.



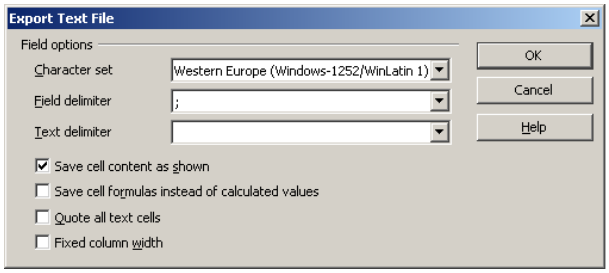
Select ‘File/Save As’, chose ‘Text CSV’ as file type and check the checkbox ‘Edit filter settings’.



Confirm that you want to ‘Use Text CSV Format’. If you uncheck the box at the bottom of the dialog you are not asked this question in the future.



Chose the `;` as ‘Field delimiter’, leave the ‘Text delimiter’ empty and click OK.



Then the data can be imported in R. Note: If you have data with decimals, then depending on the language of your operating system, it is required to specify whether `.` or `,` is the decimal delimiter.

```
setwd("c://data")
maize <- read.table ( "v14-u11-01.csv", header=T, sep=";", dec=".")
```

```
> str(maize)
'data.frame':        24 obs. of  2 variables:
 $ variety: Factor w/ 4 levels "v1","v2","v3",..: 1 2 3 4 1 2 3 4 1 2 ...
 $ yield  : int  64 78 75 55 72 91 93 66 68 97 ...
```

Importing R graphics in LibreOffice text documents

Open an new text document. Chose ‘Insert/Picture/From File’ and chose the saved png file.

Mixed Linear Models: Growth Curves with nlme

Matthias Frisch

Growth curves of chicken

```
library(nlme); library(lsmeans); library(multcomp); library(lattice)
huhn <- read.table("v14-u51-21.csv",header=T,sep=";",dec=",",
                   stringsAsFactors = TRUE)
str(huhn)
bwplot (Time   ~ weight|Diet, data=huhn)          # Boxplots using the
bwplot (weight ~ as.factor(Time)|Diet, data=huhn) # Lattice package
bwplot (weight ~ Diet|Time, data=huhn)            # Grouped by time
```

```
> str(huhn)
'data.frame':        578 obs. of  4 variables:
 $ weight: int  42 51 59 64 76 93 106 125 149 171 ...
 $ Time  : int  0 2 4 6 8 10 12 14 16 18 ...
 $ Chick : Factor w/ 50 levels "c01","c02","c03",..: 15 15 15 15 15 15 15 15 15 15 ...
 $ Diet  : Factor w/ 4 levels "d1","d2","d3",..: 1 1 1 1 1 1 1 1 1 1 ...
```

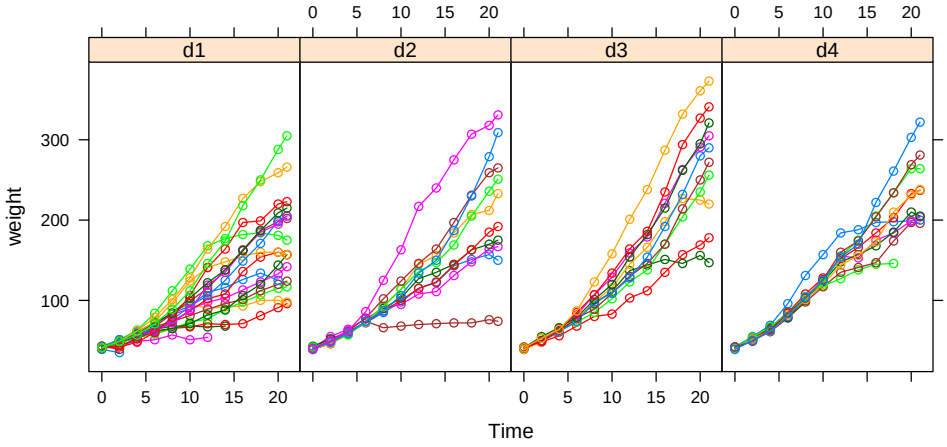
Single factor anova for the last day

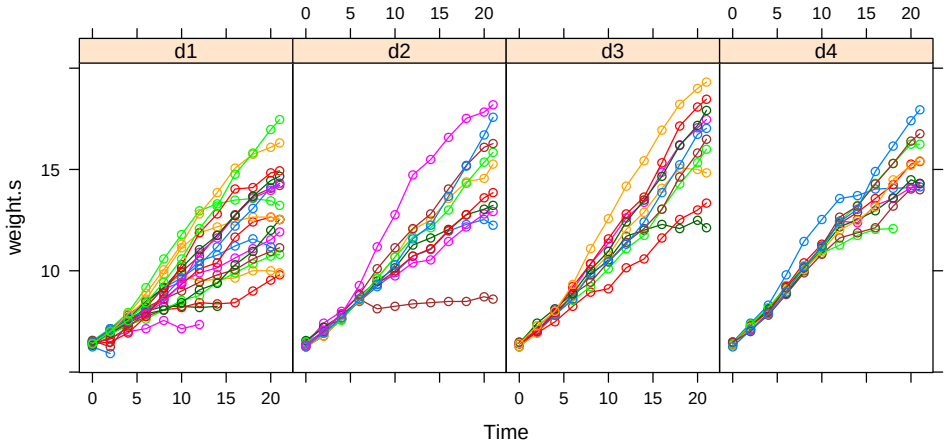
```
huhn.21 <- subset ( huhn, Time == 21 )
m.21 <- ( aov ( weight ~ Diet, data=huhn.21) )
summary (m.21)
model.tables (m.21,"means")
summary ( glht( m.21, lsm(pairwise~Diet) ), adjusted("none") )
```

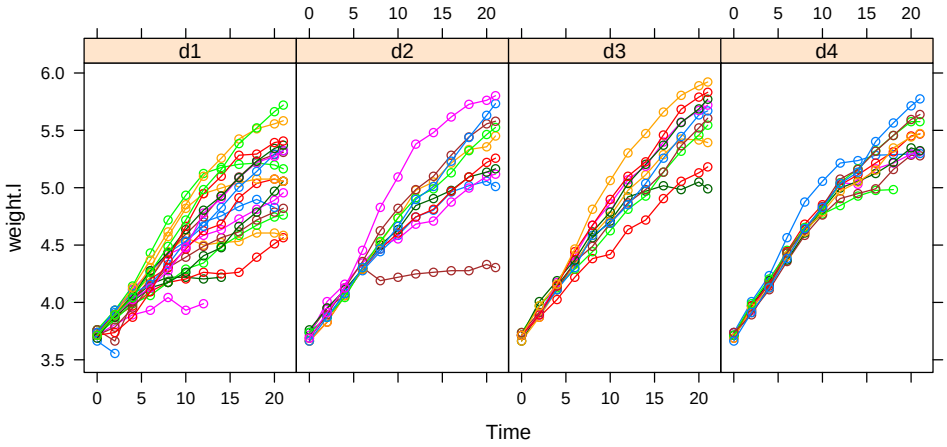
```
> summary (m.21)
            Df Sum Sq Mean Sq F value  Pr(>F)
Diet         3  57164   19055   4.655 0.00686 **
Residuals   41 167839    4094
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> model.tables (m.21,"means")
      d1    d2    d3    d4
   177.7 214.7 270.3 238.6

> summary ( glht( m.21, lsm(pairwise~Diet)), adjusted("none") )
             Estimate Std. Error t value Pr(>|t|)
d1 - d2 == 0   -36.95      25.79  -1.433  0.15955
d1 - d3 == 0   -92.55      25.79  -3.588  0.00088 ***
d1 - d4 == 0   -60.81      26.66  -2.281  0.02782 *
d2 - d3 == 0   -55.60      28.61  -1.943  0.05889 .
d2 - d4 == 0   -23.86      29.40  -0.811  0.42178
d3 - d4 == 0    31.74      29.40   1.080  0.28653
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- none method)
```

Check growth curves for linearity

```
xyplot ( weight ~ Time|Diet,    # For each diet one plot
         groups=Chick,          # Each chick gets an individual color
         type="b",              # Plot joined symbols
         data=huhn)             # Lattice package

huhn$weight.s <- sqrt (huhn$weight)        # Square root transformation
xyplot ( weight.s ~ Time|Diet, groups=Chick, type="b", data=huhn)

huhn$weight.l <- log (huhn$weight)         # Logarithmic transformation
xyplot ( weight.l ~ Time|Diet, groups=Chick, type ="b", data=huhn)
```







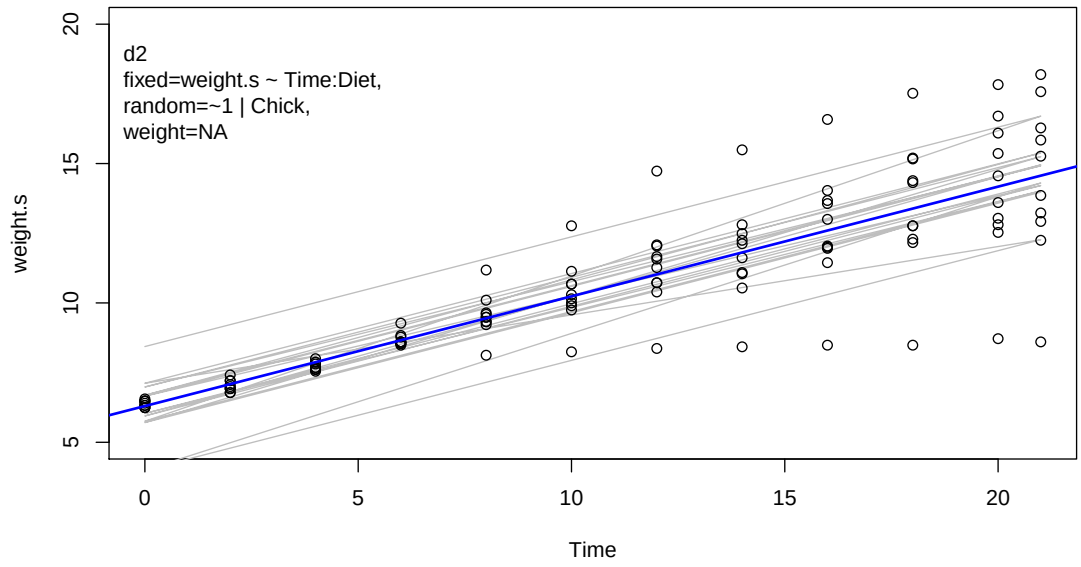
Comparison of alternative models

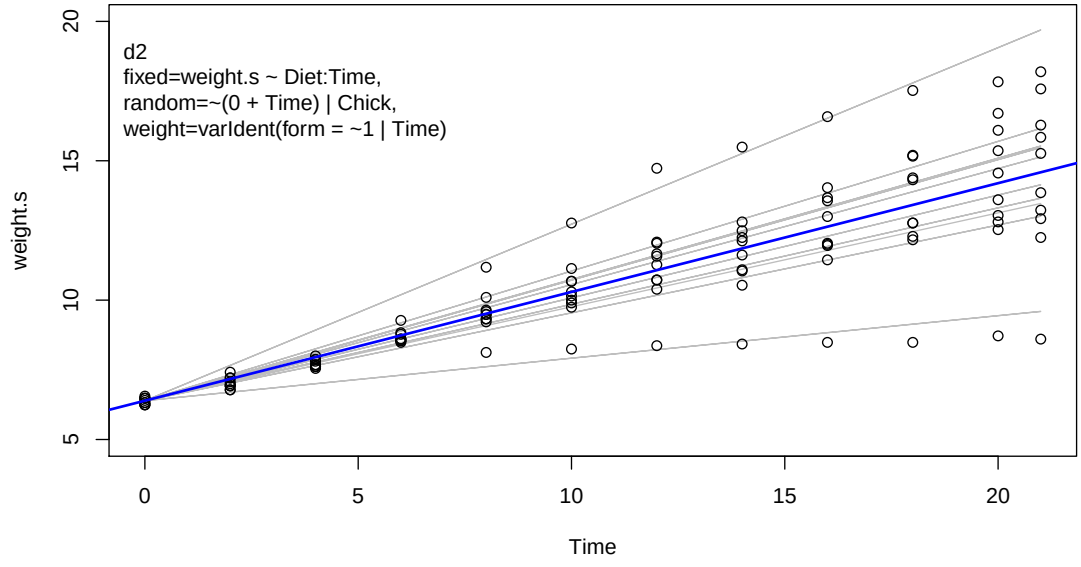
```
m.1 <- lm ( weight.s ~ Time:Diet , # Fixed linear model: Separate slope
            data = huhn )          # for each Diet. Common intercept
AIC(m.1)                           # is default (no need to specify)

m.2 <- lme ( fixed  = weight.s ~ Time:Diet,  # Mixed linear model 'lme'
             random = ~ 1 | Chick,           # Random effect for each
             data   = huhn)                  # chicken. Column of '1's
AIC(m.2)                                     # in the design matrix Z

m.3 <- lme ( fixed  = weight.s ~ 0 +    # '0+' Removes the default
                      Diet + Time:Diet, # intercept. 'Diet' adds an
             random = ~ 1 | Chick,      # separate intercept for each
             data   = huhn )            # diet. Hence, separate reg-
AIC(m.3)                                # ression line for each chick

m.4 <- lme ( fixed  = weight.s ~ Diet:Time, # Fixed effects as in m.2
             random = ~ Time | Chick,       # Random regression slope
             data   = huhn )                # for each chicken. Column
AIC(m.4)                                    # with days in Z

m.5 <- lme ( fixed  = weight.s ~  0 +     # Fixed effects as in m.3
                      Diet + Diet:Time,   # but with a with regression
             random = ~ Time | Chick,     # slope replacing the simple
             data   = huhn )              # random effect
AIC(m.5)

m.6 <- lme ( fixed   = weight.s ~ Diet:Time,   # Heteroscedacity
             random  = ~ Time | Chick,         # Each Time gets
             weights = varIdent(form=~1|Time), # a separate error
             data    = huhn )                  # variance
AIC(m.6)

m.7 <- lme ( fixed   = weight.s ~ Diet:Time,   # Remove the intercept
             random  = ~ (0+Time) | Chick,     # from the regression
             weights = varIdent(form=~1|Time), # lines: Each chicken
             data    = huhn )                  # has the same starting
AIC(m.7)                                       # point
```

```
> AIC(m.1)
[1] 1941.806
> AIC(m.2)
[1] 1700.255
> AIC(m.3)
[1] 1705.319
> AIC(m.4)
[1] 983.8665
> AIC(m.5)
[1] 988.8991
> AIC(m.6)
[1] 860.8408
> AIC(m.7)
[1] 884.4627
```

Visual check of the model fit

```
mo <- m.7      # m.2, m.4, m.6, m.7
di <- "d3"     # "d1","d2","d3","d4"

coeff <- mo$coefficients$fixed
huhn$pred <- predict(mo)
h.tmp <- subset(huhn, Diet == di)
plot ( weight.s ~ Time, type="n", ylim=c(5,20), data=h.tmp )
points ( pred ~ Time, type="l", col="gray",data=h.tmp )
points ( weight.s ~ Time, type="p", data=h.tmp )
id <- 1 + as.numeric(substr(di,2,2))
abline ( a=coeff[1], b=coeff[id], col="blue", lwd=2 )
x <- as.character(mo$call)
t <- paste (di,"\n","fixed=",x[2], ",\nrandom=",x[4],
            ", \nweight=", x[5],sep="")
text(-0.8,17.00,t,pos=4)
```





Comparision of the regression slopes for the diets

```
anova(m.7)
m.7$coefficients$fixed
K <- rbind ( "d1-d2" = c(0, -1,  1, 0, 0),
             "d1-d3" = c(0, -1,  0, 1, 0),
             "d1-d4" = c(0, -1,  0, 0, 1),
             "d2-d3" = c(0,  0, -1, 1, 0),
             "d2-d4" = c(0,  0, -1, 0, 1),
             "d3-d4" = c(0,  0,  0,-1, 1) )
summary ( glht(m.7, K), adjusted("none") )
```

```
> anova(m.7)
            numDF denDF   F-value p-value
(Intercept)     1   524 281818.50  <.0001
Diet:Time       4   524    161.21  <.0001

> m.7$coefficients$fixed
(Intercept) Dietd1:Time Dietd2:Time Dietd3:Time Dietd4:Time
  6.3915243   0.3074578   0.3902115   0.4597000   0.4531695

> summary ( glht(m.7, K ),adjusted("none") )
Linear Hypotheses:
           Estimate Std. Error z value Pr(>|z|)
d1-d2 == 0  0.08275    0.04198   1.971 0.048714 *
d1-d3 == 0  0.15224    0.04198   3.626 0.000288 ***
d1-d4 == 0  0.14571    0.04199   3.471 0.000519 ***
d2-d3 == 0  0.06949    0.04818   1.442 0.149189
d2-d4 == 0  0.06296    0.04818   1.307 0.191272
d3-d4 == 0 -0.00653    0.04818  -0.136 0.892174
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- none method)
```

Plot back transformed growth curve

```
diet1 = subset(huhn, Diet == "d1");
r.c <- coeff.m.7

plot (weight~Time, type = "p", main = "", ylim=c(0,400), data=diet1)
l.s <- ( r.c[1] + r.c[2] * 0:21 ) ^2
points(0:21,l.s, col = "blue",lwd=2,type="l")
text(2.5,350,"Diet 1")
```

Efficient data analysis and documentation with R

Matthias Frisch

1 Data input

For data input csv-files are useful. csv-files are text-only files, they can be prepared with a text editor, by a data base routine, or with spreadsheet software. Often they are structured such that the data from each observation unit are stored in one line of the file. A first set of columns identify the observation unit, and a second set of columns contain the observed data. This csv-file was prepared with a text editor:

```
loc     year  plot gen      rep trt DHE DMA PLH GYLD TKW
Holetta y2005 p1   EH1877   r1  t6  73  124 107 3821 47.7
Holetta y2005 p2   EH1505   r1  t3  73  125 107 3120 47.4
Holetta y2005 p3   EH1900   r1  t2  70  120 123 4175 46.4
Holetta y2005 p4   IBON2796 r1  t11 73  122 100 3980 54.3
Holetta y2005 p5   EH1869   r1  t7  78  127 117 3895 50.3
...
```

It is good practice to follow conventions for variable naming. For example:

- Use abbreviations for column headings, not long words
- Use always the same abbreviations in all data files. E.g., gen is genotype, loc is year
- Use lowercase letters for columns containing 'identifier' variables that describe the observation units. Use capital letters for columns that contain measurements
- If a column contains a numeric factor level, place an alphanumeric symbol in front of the number. E.g., write y2005 but not 2005. Alphanumeric values are interpreted as factors by analysis routines. This avoids the need of an explicit conversion to a factor and avoids the danger of using discrete factor levels in an analysis as continuous regressors (like a linear regression on the year)

Another rule that simplifies life is:

- Never use spaces, or special characters, or characters that occur only in some languages in data files, file names, or folder names. This will cause severe problems, and they may not occur immediately but only if you try to read your data on another computer, with another version of R, or another operating system, or whatever. The gold standard was defined by Céline, a master student in Crop Sciences: "Königin Céline!". We have a German umlaut, a space, a French accent-daigu, and an exclamation mark. That won't work

csv-files are characterized by three important properties:

- Are alphanumeric values quoted or not?
- Which symbol separates two columns?
- Which symbol is used as a decimal delimiter for numeric values?

In the above example, no quotation was used, columns are separated by spaces, and the decimal delimiter is a ".". If the data file is prepared by a spreadsheet software, it is important to be aware of which column separators and decimal delimiters were used. A very good option to prepare data files is the free LibreOffice software. It has the option to define the column separator and the decimal delimiter. Popular is also Excel. In newer versions of Excel the language of the operating system determines the column separator and the decimal delimiter. A German version of Excel will export:

```
"loc";"year";"plot";"gen";"rep";"trt";"DHE";"DMA";"PLH";"GYLD";"TKW"
"Holetta";"y2005";"p1";"EH1877";"r1";"t6";73;124;107;3821;47,7
"Holetta";"y2005";"p2";"EH1505";"r1";"t3";73;125;107;3120;47,4
"Holetta";"y2005";"p3";"EH1900";"r1";"t2";70;120;123;4175;46,4
"Holetta";"y2005";"p4";"IBON2796";"r1";"t11";73;122;100;3980;54,3
"Holetta";"y2005";"p5";"EH1869";"r1";"t7";78;127;117;3895;50,3
...
```

and an English version

```
"loc","year","plot","gen","rep","trt","DHE","DMA","PLH","GYLD","TKW"
"Holetta","y2005","p1","EH1877","r1","t6",73,124,107,3821,47.7
"Holetta","y2005","p2","EH1505","r1","t3",73,125,107,3120,47.4
"Holetta","y2005","p3","EH1900","r1","t2",70,120,123,4175,46.4
"Holetta","y2005","p4","IBON2796","r1","t11",73,122,100,3980,54.3
"Holetta","y2005","p5","EH1869","r1","t7",78,127,117,3895,50.3
...
```

The German version is read in with

```
met <- read.table ("v14-y10-DE-data.csv",header=T,sep=";",dec=","
                   stringsAsFactors = TRUE)
str(met)
table(met$year,met$loc)
```

```
> table(met$year,met$loc)

      Adet Asasa Bekoji Holetta SGonder
  y2005   60    60     60      60      60
  y2006   60    60     60      60      60
  y2007   60    60     60      60      59

> str(met)
'data.frame':        899 obs. of  11 variables:
 $ loc : Factor w/ 5 levels "Adet","Asasa",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ year: Factor w/ 3 levels "y2005","y2006",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ plot: Factor w/ 60 levels "p1","p10","p11",..: 1 12 23 34 45 56 58 59 60 2 ...
 $ gen : Factor w/ 15 levels "Beka","BNEth019",..: 9 3 11 14 8 10 7 5 15 12 ...
 $ rep : Factor w/ 4 levels "r1","r2","r3",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ trt : Factor w/ 15 levels "t1","t10","t11",..: 12 9 8 3 13 10 2 1 11 4 ...
 $ DHE : int  73 73 70 73 78 76 77 73 70 73 ...
 $ DMA : int  124 125 120 122 127 121 125 120 123 125 ...
 $ PLH : num  107 107 123 100 117 120 118 122 103 115 ...
 $ GYLD: int  3821 3120 4175 3980 3895 3880 3605 4460 2760 3880 ...
 $ TKW : num  47.7 47.4 46.4 54.3 50.3 42.8 51.9 51 49.8 53.6 ...
```

2 Carrying out an analysis for several columns of a data set

We use the package emmeans, which can be used to estimate means in balanced experiments and least-square means or expected marginal means for unbalanced data.

```
suppressMessages(library("emmeans"))
emm_options(msg.interaction=FALSE,msg.nesting=FALSE)
```

A simple analysis to obtain the mean yield (GYLD) of the 15 genotypes (gen) is

```
m.f  <- lm (GYLD ~ gen*loc*year + year:loc:rep, data = met)
l <- emmeans ( m.f, "gen",mode="a")
CLD (l, sort=TRUE, reversed=TRUE, alpha=0.10, adjust="none")
```

```
> CLD(l,sort=TRUE,reversed=TRUE,alpha=0.10, adjust="none")
 gen          emmean      SE  df lower.CL upper.CL .group
 EH1847     3575.133 65.8823 629 3445.757 3704.509  1
 EH1877     3327.817 65.8823 629 3198.441 3457.193   2
 EH1900     3290.600 65.8823 629 3161.224 3419.976   2
 EH1893     3267.267 65.8823 629 3137.891 3396.643   23
 BNEth019   3267.100 65.8823 629 3137.724 3396.476   23
 IBON2796   3221.000 65.8823 629 3091.624 3350.376   23
 IBON4499   3216.667 65.8823 629 3087.291 3346.043   23
 F2SXS13395 3207.333 65.8823 629 3077.957 3336.709   23
 EH1505     3128.267 65.8823 629 2998.891 3257.643    34
 EH1869     3033.133 65.8823 629 2903.757 3162.509     4
 EH18472    2982.783 65.8823 629 2853.407 3112.159     45
 HB120      2874.500 65.8823 629 2745.124 3003.876      56
 EH1864     2784.000 65.8823 629 2654.624 2913.376       6
 Beka       2581.217 65.8823 629 2451.841 2710.593        7
 EH1551     2534.631 66.6620 629 2403.724 2665.538        7
```

It is good practice not to repeat R code by copy-and-paste within one file for different parts of an analysis. This is very prone to errors if the code is changed in a later stage of the analysis. To carry out the same analysis for other variables in the data set without replicating the code, we define the name of the column to be analyzed in a variable and then generate the R code that runs the analysis.

```
Trait <- "GYLD"
paste("m.fixed <- lm(",Trait,"~ gen*loc*year + year:loc:rep,
                                data=met)")
```

```
> paste("m.fixed  <- lm (",Trait,"~ gen*loc*year + year:loc:rep, data = met)")
[1] "m.fixed  <- lm ( GYLD ~ gen*loc*year + year:loc:rep, data = met)"
```

Now we can use the two commands eval and parse to run commands stored in a textstring

```
eval(parse(text=paste(
 "m.fixed  <- lm (",Trait,"~ gen*loc*year + year:loc:rep,
                             data = met)")))
```

Hence running the same analysis with TKW can be done with

```
Trait <- "TKW"
eval(parse(text=paste(
  "m.fixed  <- lm (",Trait,"~ gen*loc*year + year:loc:rep,
                              data = met)")))
l <- emmeans ( m.fixed, "gen",mode="a")
CLD(l,sort=TRUE,reversed=TRUE,alpha=0.10, adjust="none")
```

and running the commands for several columns is

```
Traits <- c("GYLD","TKW","PLH")
for (Trait in Traits) {
  eval(parse(text=paste(
    "m.fixed  <- lm (",Trait,"~ gen*loc*year + year:loc:rep,
```

```
                                  data = met)")))
    l <- emmeans ( m.fixed, "gen",mode="a")
    comp <- CLD(l,sort=TRUE,reversed=TRUE,alpha=0.10, adjust="none")
    print(Trait)
    print(comp)
  }
```

```
[1] "GYLD"
 gen          emmean      SE  df lower.CL upper.CL .group
 EH1847     3575.133 65.8823 629 3445.757 3704.509  1
 EH1877     3327.817 65.8823 629 3198.441 3457.193   2
 EH1900     3290.600 65.8823 629 3161.224 3419.976   2
 ...
[1] "TKW"
 gen          emmean        SE  df lower.CL upper.CL .group
 IBON4499   48.04667 0.3712298 629 47.31767 48.77567  1
 EH1505     47.30333 0.3712298 629 46.57433 48.03233  1
 IBON2796   45.72000 0.3712298 629 44.99100 46.44900   2
 F2SXS13395 45.34167 0.3712298 629 44.61267 46.07067   2
 ...

[1] "PLH"
 gen           emmean        SE  df  lower.CL  upper.CL .group
 EH18472    112.62667 0.7853768 629 111.08439 114.16894  1
 HB120      110.70333 0.7853768 629 109.16106 112.24561   2
 Beka       110.18333 0.7853768 629 108.64106 111.72561   2
 EH1847     106.08333 0.7853768 629 104.54106 107.62561    3
 ...
```

We can also save the results of the analyses to a named list

```
  Traits <- c("GYLD","TKW","PLH")
  Result <- vector(mode="list",length=length(Traits))
  names(Result) <- Traits
  for (Trait in Traits) {
    eval(parse(text=paste(
      "m.fixed  <- lm (",Trait,"~ gen*loc*year + year:loc:rep,
                                 data = met)")))
    l <- emmeans ( m.fixed, "gen",mode="a")
    Result[[Trait]] <- CLD (l, sort=TRUE, reversed=TRUE,
                            alpha=0.10, adjust="none")
  }
  Result
  detach(package:emmeans)
```

```
> Result
$GYLD
 gen          emmean      SE  df lower.CL upper.CL .group
 EH1847     3575.133 65.8823 629 3445.757 3704.509  1
 ...
$TKW
 gen          emmean        SE  df lower.CL upper.CL .group
 IBON4499   48.04667 0.3712298 629 47.31767 48.77567  1
 ...
$PLH
 gen           emmean        SE  df  lower.CL  upper.CL .group
 EH18472    112.62667 0.7853768 629 111.08439 114.16894  1
...
```

3 Carrying out an analysis for several factor level combinations

The data set contains a multi-environment trial, for which we want to determine the single-environment heritabilities.

```
met <- read.table ( "v14-y10-DE-data.csv", header=T, sep=";", dec=",",
                    stringsAsFactors = TRUE )
suppressMessages(library("lme4"))
met$env <- met$year:met$loc
Environments <- levels(met$env)
```

```
> Environments
 [1] "y2005:Adet"    "y2005:Asasa"   "y2005:Bekoji"  "y2005:Holetta"
 [5] "y2005:SGonder" "y2006:Adet"    "y2006:Asasa"   "y2006:Bekoji"
 [9] "y2006:Holetta" "y2006:SGonder" "y2007:Adet"    "y2007:Asasa"
[13] "y2007:Bekoji"  "y2007:Holetta" "y2007:SGonder"
```

We calculate the heritability for one environment.

```
E <- Environments[1]
SL <- subset(met,met$env==E)
m.SL  <- lmer ( GYLD ~ (1|gen) + (1|rep), data = SL)
s <- summary(m.SL)
v.e <- s$sigma^2
v.g <- as.numeric(s$varcor$gen)
hsq <- v.g/(v.g+v.e)
```

```
> hsq
[1] 0.5390138
```

To calculate the heritabilities for all environments we use a loop and store the result in a vector.

```
n <- length(Environments)
SLhsq <- vector("numeric",n)
names(SLhsq) <- Environments
for (i in 1:n){
  SL <- subset(met,met$env==Environments[i])
  m.SL  <- lmer ( GYLD ~ (1|gen) + (1|rep), data = SL)
  s <- summary(m.SL)
  v.e <- s$sigma^2
  v.g <- as.numeric(s$varcor$gen)
  SLhsq[i] <- v.g/(v.g+v.e)
}
```

```
> data.frame(SLhsq)
                SLhsq
y2005:Adet    0.5390138
y2005:Asasa   0.5055993
y2005:Bekoji  0.3191383
y2005:Holetta 0.2549625
...
```

We use eval and parse to specify the trait.

```
Trait <- "TKW"
for (i in 1:n){
  SL <- subset(met,met$env==Environments[i])
  eval(parse(text=paste(
    "m.SL  <- lmer (", Trait ,"~ (1|gen) + (1|rep), data = SL)")))
  s <- summary(m.SL)
  v.e <- s$sigma^2
  v.g <- as.numeric(s$varcor$gen)
  SLhsq[i] <- v.g/(v.g+v.e)
}
```

Two nested loops for traits and environments

```
Environments <- levels(met$env)
Traits       <- c("GYLD","TKW","PLH")

nEnv <- length(Environments)
nTrt <- length(Traits)

Heritability <- matrix(nrow=nEnv,ncol=nTrt)
rownames(Heritability) <- Environments
colnames(Heritability) <- Traits

for (i in 1:nEnv){
  SL <- subset(met,met$env==Environments[i])
  for (j in 1:nTrt) {
    Trait <- Traits[j]
    eval(parse(text=paste(
      "m.SL  <- lmer (", Trait ,"~ (1|gen) + (1|rep), data = SL)")))
    s <- summary(m.SL)
    v.e <- s$sigma^2
    v.g <- as.numeric(s$varcor$gen)
    Heritability[i,j] <- v.g/(v.g+v.e)
  }
}
```

```
> Heritability
                  GYLD       TKW       PLH
y2005:Adet    0.5390138 0.8599472 0.5499248
y2005:Asasa   0.5055993 0.9000222 0.6726486
y2005:Bekoji  0.3191383 0.9257919 0.7544237
y2005:Holetta 0.2549625 0.8344078 0.4091020
y2005:SGonder 0.4208843 0.7758101 0.6554413
...
```

4 Combining code and results in a PFD file: stitch

R code and results can be combined in a PDF file using the command stitch from the package knitr. This requires the text processing system LaTeX. On Windows Miktex is a TeX system that can be easily installed. After downloading the Miktex basic distribution (file name basic-miktex-2.9nn-x64.exe), and running the installer, it is important to check the option to download automatically missing packages without confirmation. This is necessary to use the stitch command, because several LaTeX packages that are required and not included in the Miktex basic distribution.

Generate a comment line in the R file and add the two commands `library("knitr")` and `stitch("filename.R")` in the comment line.

```
# library("knitr"); stitch("v14-y10-4.R")
```

Then mark the two commands without the comment sign and submit them. The PDF is generated and stored in the working directory.

```
> library("knitr"); stitch("v14-y10-4.R")
processing file
...
output file: v14-y10-4.tex
PDF output at: v14-y10-4.pdf
```

To control the format of the output an .Rnw file that contains a template can be used.

```
# library("knitr"); stitch ( "v14-y10-4.R", template="mf.Rnw" )
```

The resulting PDF file is shown in Figure 1 on p. 9.

5 Combine code and results in a Word file: RMarkdown

RMarkdown is an extension of RStudio. It can be used to combine code and result in either a Word file, an HTML file or a pdf file. Save an R file with the extension .Rmd. Then surround each block of R commands that ends with an output with ```` ```{r} ```` and ```` ``` ```` (three accent-grave, quotes don't work).

````
```{r}
# v14-y10-5.Rmd
met <- read.table ( "v14-y10-DE-data.csv", header=T, sep=";", dec=",",
                    stringsAsFactors = TRUE )
met$gen <- reorder(met$gen, met$GYLD, mean)
suppressMessages(library("emmeans"))
emm_options(msg.interaction=FALSE,msg.nesting=FALSE)
m.fixed  <- lm (GYLD ~ gen*loc*year + year:loc:rep, data = met)
l <- emmeans ( m.fixed, "gen",mode="a")
CLD(l,sort=TRUE,reversed=TRUE,alpha=0.10, adjust="none")
```

```{r}
plot(l,comparisons=TRUE,alpha=0.10,adjust="none")
```
````

Then click on "Knit to Word". The resulting Word file is saved in the working directory, it is shown in Figure 2 on p. 10.

```
##############################################################################
# v14-y10-4.R
##############################################################################
# library(knitr); stitch("v14-y10-4.R",template="mf.Rnw")

met <- read.table ( "v14-y10-DE-data.csv", header=T, sep=";", dec="," )
met$gen <- reorder(met$gen, met$GYLD, mean)
suppressMessages(library("emmeans"))
emm_options(msg.interaction=FALSE,msg.nesting=FALSE)
m.fixed  <- lm (GYLD ~ gen*loc*year + year:loc:rep, data = met)
l <- emmeans ( m.fixed, "gen",mode="a")
CLD(l,sort=TRUE,reversed=TRUE,alpha=0.10, adjust="none")

##  gen        emmean   SE  df lower.CL upper.CL .group
##  EH1847       3575 65.9 629     3446     3705  1
##  EH1877       3328 65.9 629     3198     3457   2
##  EH1900       3291 65.9 629     3161     3420   2
##  EH1893       3267 65.9 629     3138     3397   23
##  BNEth019     3267 65.9 629     3138     3396   23
##  IBON2796     3221 65.9 629     3092     3350   23
##  IBON4499     3217 65.9 629     3087     3346   23
##  F2SXS13395   3207 65.9 629     3078     3337   23
##  EH1505       3128 65.9 629     2999     3258    34
##  EH1869       3033 65.9 629     2904     3163     4
##  EH18472      2983 65.9 629     2853     3112     45
##  HB120        2874 65.9 629     2745     3004      56
##  EH1864       2784 65.9 629     2655     2913       6
##  Beka         2581 65.9 629     2452     2711        7
##  EH1551       2535 66.7 629     2404     2666        7
##
## Results are averaged over the levels of: rep, loc, year
## Confidence level used: 0.95
## significance level used: alpha = 0.1

plot(l,comparisons=TRUE,alpha=0.10,adjust="none")
```

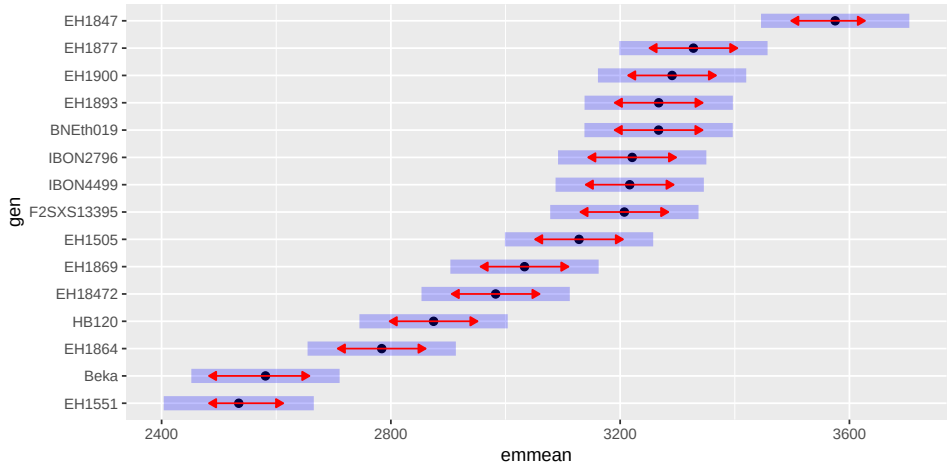


Figure 1: PDF file generated with stitch

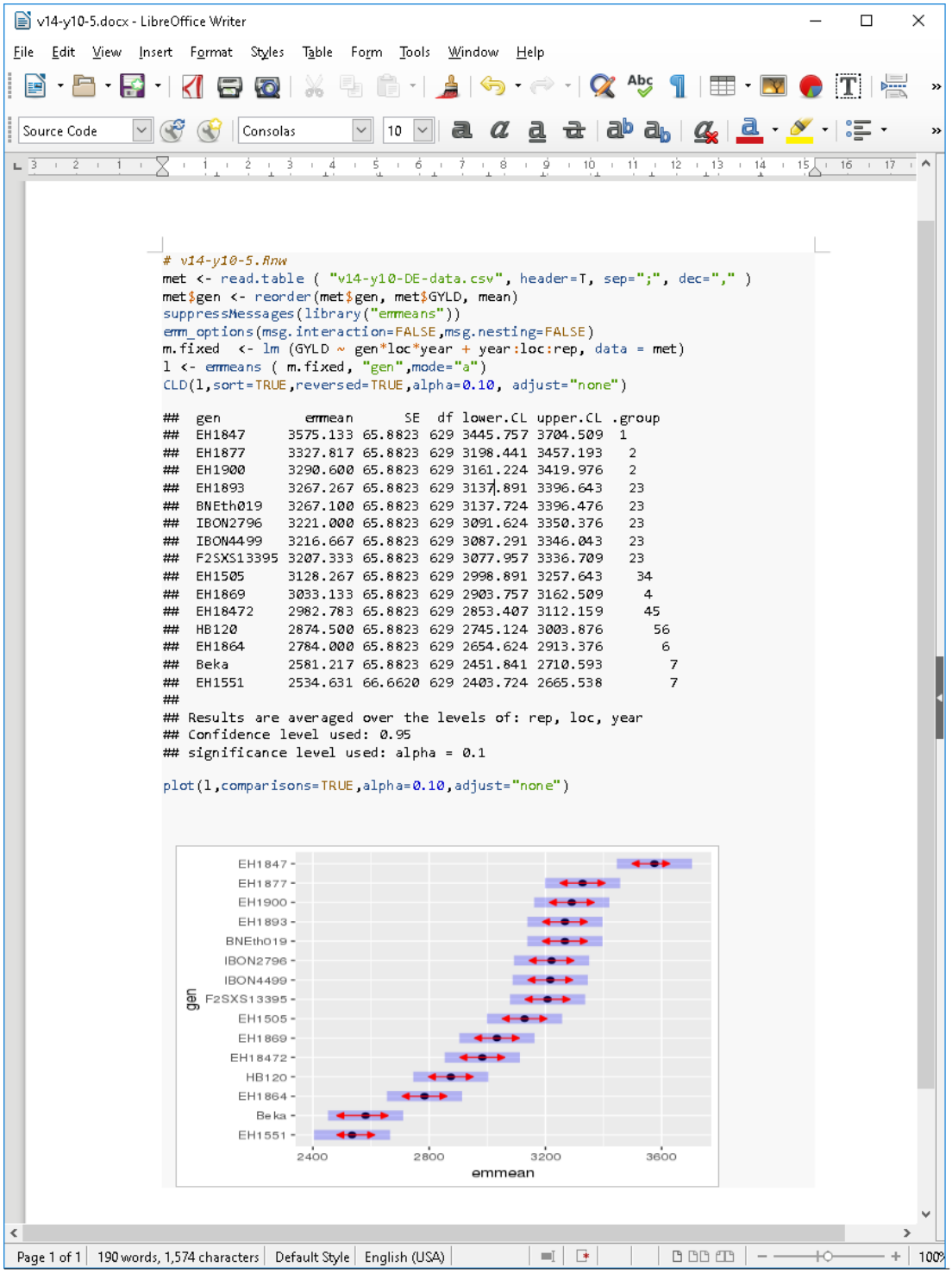


Figure 2: Word file generated with RMarkdown

6 Writing results to an Excel workbook: openxlsx

We compare the yield of the barley lines.

```
met <- read.table ( "v14-y10-DE-data.csv", header=T, sep=";", dec=",",
                    stringsAsFactors = TRUE )
met$gen <- reorder(met$gen, met$GYLD, mean)
suppressMessages(library("emmeans"))
suppressMessages(library("openxlsx"))
emm_options(msg.interaction=FALSE,msg.nesting=FALSE)
m.fixed  <- lm (GYLD ~ gen*loc*year + year:loc:rep, data = met)
l <- emmeans ( m.fixed, "gen",mode="a")
EMM <- CLD(l, sort=TRUE, reversed=TRUE, alpha=0.10, adjust="none")
```

The basic commands are illustrated by the follwing example: We create an new workbook, add a new a worksheet to the workbook, write the data of the emmeans table to the sheet, and save the workbook.

```
wb <-createWorkbook()
addWorksheet (wb,"Table")
writeData ( wb, "Table", EMM, colNames=TRUE, rowNames=FALSE )
saveWorkbook ( wb, "EMMeansGYLD.xlsx", overwrite = TRUE )
```

We can structure the output in the workbook: We write a title in the first row and format it bold, we write the data table starting in row 3. Then we use a format that rounds the numbers in the columns 2, 3, 5, and 6 to two digits after the decimal delimiter.

```
wb <-createWorkbook()
addWorksheet ( wb, "GYLD" )
writeData ( wb, "GYLD","EMMeans for Grain Yield", startRow=1)
bold <- createStyle ( textDecoration="bold" )
addStyle( wb, "GYLD", bold, 1, 1)
writeData ( wb, "GYLD", EMM,colNames=TRUE, rowNames=FALSE, startRow=3)
twoDigits <- createStyle(numFmt="0.00")
addStyle ( wb, "GYLD", twoDigits, 1:(3+nrow(EMM)), c(2:3,5:6),
           gridExpand=TRUE, stack=TRUE)
```

No we add a plot to the worksheet, in the rows below the table.

```
par (  mar=c(6,4,1,1))
png("plot.png", height=1000, width=1500, res=300, pointsize=12)
plot(l,comparisons=TRUE,alpha=0.10,adjust="none")
r<-dev.off()
insertImage(wb, "GYLD", "plot.png", width = 5, height = 5,
            startRow=(5+nrow(EMM)))

saveWorkbook(wb,"EMMeansGYLD.xlsx",overwrite = TRUE)
```

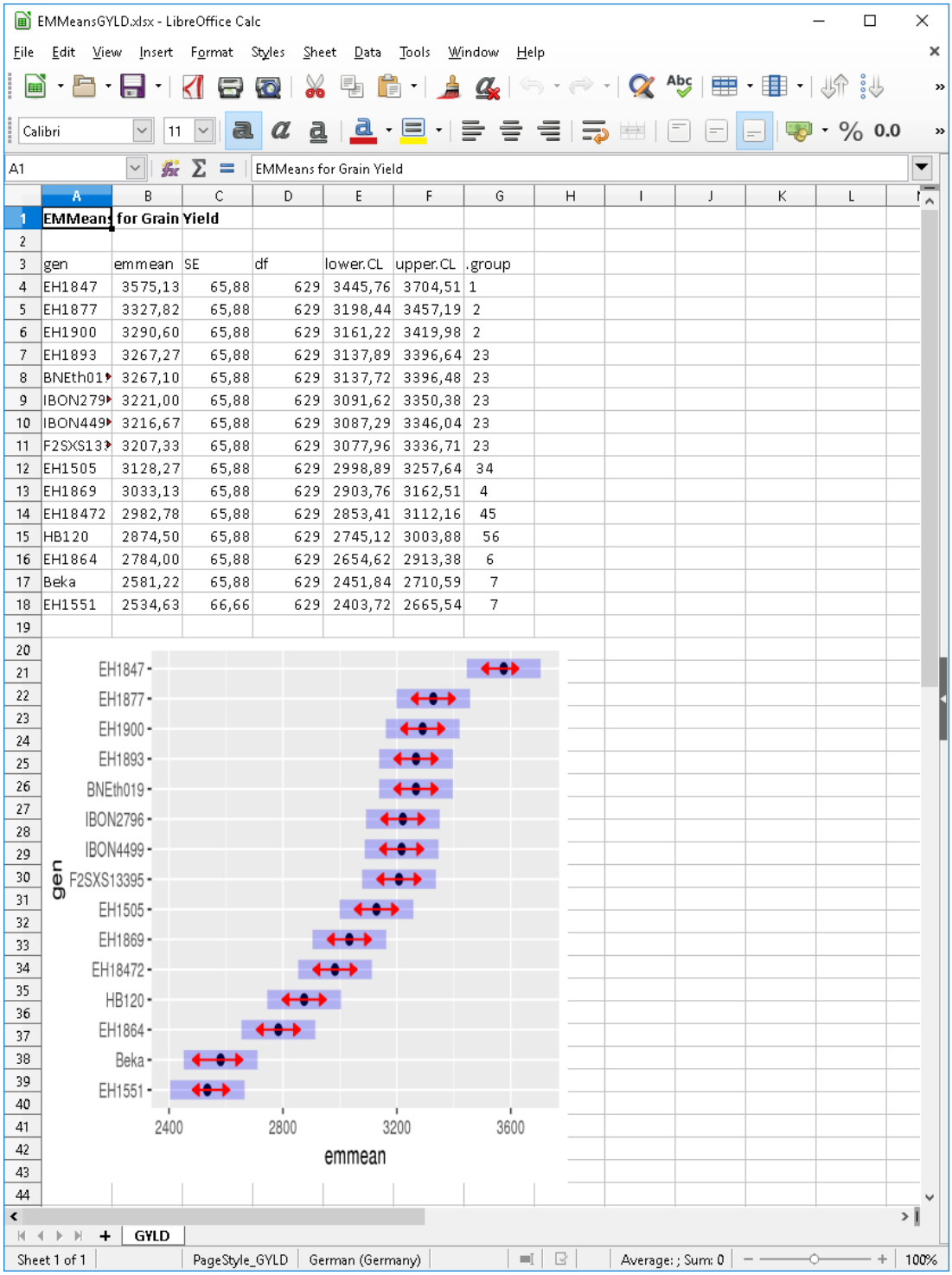


Figure 3: Excel file generated with openxlsx

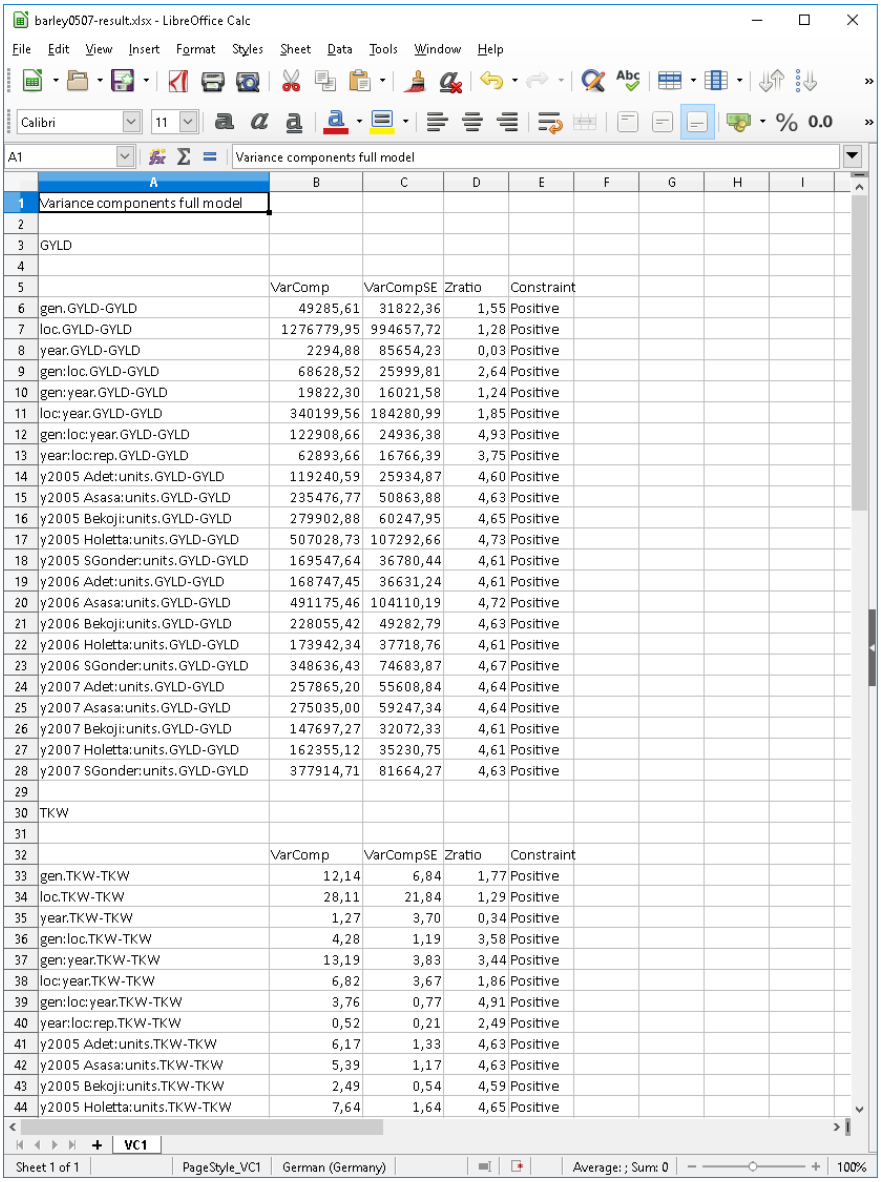
7 Exercises

1 Variance components

The following commands carry out a variance component analysis for the trait GYLD.

```
library("sommer")
met <- read.table ( "v14-y10-DE-data.csv", header=T, sep=";", dec=",",
                    stringsAsFactors = TRUE )
met$env <- paste(met$year,met$loc)
m.vc1  <- mmer ( GYLD ~ 1,
                 random = ~ gen + loc + year
                            + gen:loc + gen:year + loc:year
                            + gen:loc:year
                            + year:loc:rep,
                rcov=~vs(at(env),units),
                data = met,verbose=F)
summary(m.vc1)$varcomp
```

Estimate the variance comonents for the traits GYLD, TKW, and PLLH and write the results to an excel sheet:



	A	B	C	D	E
1	Variance components full model				
2					
3	GYLD				
4					
5		VarComp	VarCompSE	Zratio	Constraint
6	gen.GYLD-GYLD	49285,61	31822,36	1,55	Positive
7	loc.GYLD-GYLD	1276779,95	994657,72	1,28	Positive
8	year.GYLD-GYLD	2294,88	85654,23	0,03	Positive
9	gen:loc.GYLD-GYLD	68628,52	25999,81	2,64	Positive
10	gen:year.GYLD-GYLD	19822,30	16021,58	1,24	Positive
11	loc:year.GYLD-GYLD	340199,56	184280,99	1,85	Positive
12	gen:loc:year.GYLD-GYLD	122908,66	24936,38	4,93	Positive
13	year:loc:rep.GYLD-GYLD	62893,66	16766,39	3,75	Positive
14	y2005 Adet:units.GYLD-GYLD	119240,59	25934,87	4,60	Positive
15	y2005 Asasa:units.GYLD-GYLD	235476,77	50863,88	4,63	Positive
16	y2005 Bekoji:units.GYLD-GYLD	279902,88	60247,95	4,65	Positive
17	y2005 Holetta:units.GYLD-GYLD	507028,73	107292,66	4,73	Positive
18	y2005 SGonder:units.GYLD-GYLD	169547,64	36780,44	4,61	Positive
19	y2006 Adet:units.GYLD-GYLD	168747,45	36631,24	4,61	Positive
20	y2006 Asasa:units.GYLD-GYLD	491175,46	104110,19	4,72	Positive
21	y2006 Bekoji:units.GYLD-GYLD	228055,42	49282,79	4,63	Positive
22	y2006 Holetta:units.GYLD-GYLD	173942,34	37718,76	4,61	Positive
23	y2006 SGonder:units.GYLD-GYLD	348636,43	74683,87	4,67	Positive
24	y2007 Adet:units.GYLD-GYLD	257865,20	55608,84	4,64	Positive
25	y2007 Asasa:units.GYLD-GYLD	275035,00	59247,34	4,64	Positive
26	y2007 Bekoji:units.GYLD-GYLD	147697,27	32072,33	4,61	Positive
27	y2007 Holetta:units.GYLD-GYLD	162355,12	35230,75	4,61	Positive
28	y2007 SGonder:units.GYLD-GYLD	377914,71	81664,27	4,63	Positive
29					
30	TKW				
31					
32		VarComp	VarCompSE	Zratio	Constraint
33	gen.TKW-TKW	12,14	6,84	1,77	Positive
34	loc.TKW-TKW	28,11	21,84	1,29	Positive
35	year.TKW-TKW	1,27	3,70	0,34	Positive
36	gen:loc.TKW-TKW	4,28	1,19	3,58	Positive
37	gen:year.TKW-TKW	13,19	3,83	3,44	Positive
38	loc:year.TKW-TKW	6,82	3,67	1,86	Positive
39	gen:loc:year.TKW-TKW	3,76	0,77	4,91	Positive
40	year:loc:rep.TKW-TKW	0,52	0,21	2,49	Positive
41	y2005 Adet:units.TKW-TKW	6,17	1,33	4,63	Positive
42	y2005 Asasa:units.TKW-TKW	5,39	1,17	4,63	Positive
43	y2005 Bekoji:units.TKW-TKW	2,49	0,54	4,59	Positive
44	y2005 Holetta:units.TKW-TKW	7,64	1,64	4,65	Positive