R introduction

Mandy Vogel

University Leipzig

September 10, 2015

Overview

Reading Data

- `read.table()`

ANOVA

- Data
- Sums of Squares

Permutation Tests

- Permutation Tests
- Rank-Based Permutation Tests

Wilcoxon Test

Linear Model

`lm()`

`glm`

Table of Contents I

Reading Data

The most convenient way of reading data into R is via the function called `read.table()`. It requires that the data is in "ASCII format", or a "flat file" as created with Windows' NotePad or any plain-text editor. The result of `read.table()` is a data frame.

It is expected that each line of the data file corresponds to a subject information, that the variables are separated by blanks or any other separator symbol (e.g., ",", ";"). The first line of the file can contain a header (`header=T`) giving the names of the variables, which is highly recommended!

read.table()

As an example we read in the data contained in the file `fishercats.txt`

Input/Output

```
> read.table("session1data/fishercats.txt",
+            sep=" ",header=T)
  Sex Bwt Hwt
1   F 2.0 7.0
2   F 2.0 7.4
3   F 2.0 9.5
4   F 2.1 7.2
5   F 2.1 7.3
....
```

These data correspond to the heart and body weights of samples of male and female cats (R. A. Fisher, 1947).

read.table()

The first argument corresponds to the data file, the second to the fields separator and the third `header=T` specifies that the first line is a header with variable names. Important: the character variables will be automatically read as factors.
There is a variant for reading data from an url:

Input/Output

```
> winer <- read.table(
+ "http://socserv.socsci.mcmaster.ca/jfox/Courses/R/ICPSR/Wine
+ header=T)
```

read.table()

There are other variants of `read.table` function alike :

- `read.csv()` this function assumes that fields are separated by a comma instead of whites spaces
- `read.csv2()` this function assumes that the separate symbol is the semicolon, but use a comma as the decimal point (some programs, e.g., Microsoft Excel, generate this format when running in European systems)
- the function `scan()` is a powerful, but less friendly, way to read data in R; you may need it, if you want to read files with different numbers ov values per line

Reading data from the clipboard

With the function `read.delim()` or also `read.table()` it is possible to read data directly from the clipboard.
For example mark and copy some columns from an Excel spreadsheet and transfer this content to an R by

Input/Output

```
> mydata <- read.delim("clipboard",na.strings=".")
> str(mydata) # structure of the data
```

Table of Contents I

ANOVA

- a technique we use when all explanatory variables are categorical (factor)
- if there is one factor with three or more levels we use one-way ANOVA (only two levels: t-test should be preferred, would give exactly the same answer since with 2 levels $F = t^2$)
- for more factors there is two-way, three-way anova
- central idea is to compare two or more means by comparing variances
- what do you think is the null hypothesis? what the alternative?

ANOVA: Null Hypothesis

- the null:

$$\mu_1 = \mu_2 = \cdots = \mu_N$$

ANOVA: Null Hypothesis

- the null:

$$\mu_1 = \mu_2 = \cdots = \mu_N$$

- the alternative: there is at least one pair i, j with

$$\mu_i \neq \mu_j$$

The Garden Data

A data frame with 14 observations on 2 variables.

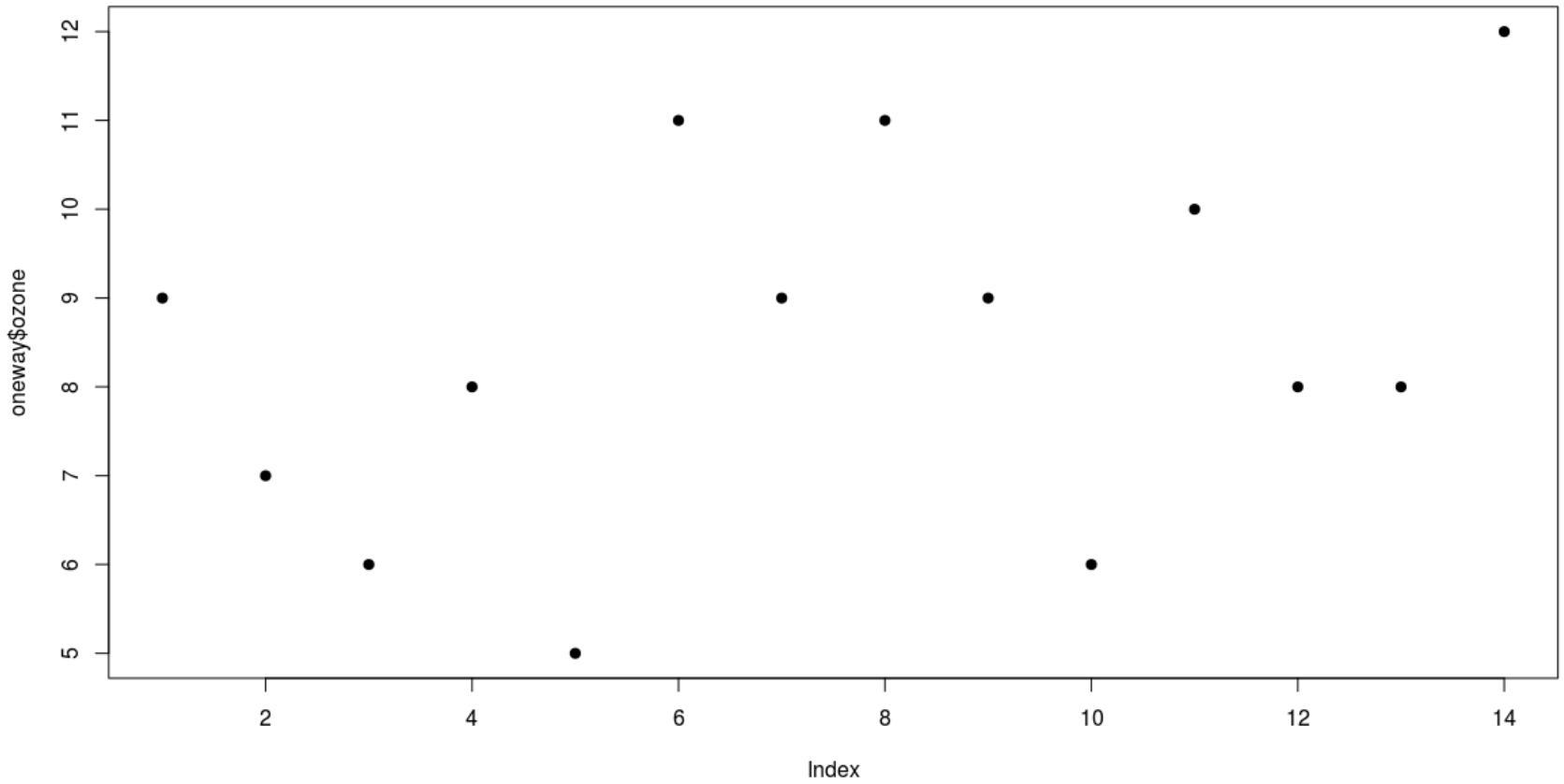
`ozone:`	athmospheric ozone concentration
`garden:`	garden id

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
ozone	9	7	6	8	5	11	9	11	9	6	10	8	8	12
garden	a	a	a	b	a	b	b	b	b	a	b	a	a	b

From: Michæl Crawley, The R-Book

Total Sum of Squares

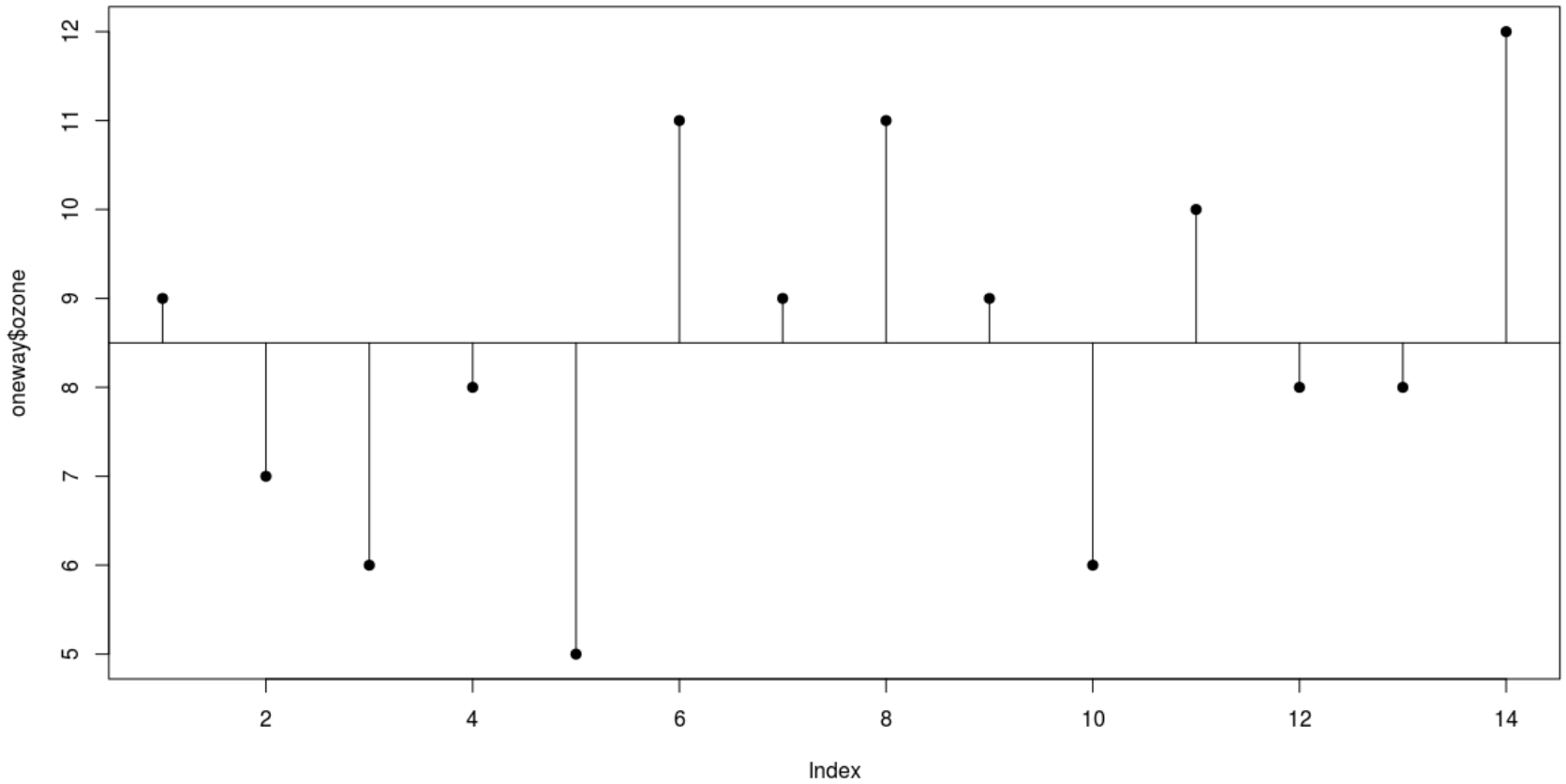
- we plot the values in order they are measured



Total Sum of Squares

- there is a lot of scatter, indicating that the variance in ozone is large
- to get a feel for the overall variance we plot the overall mean (8.5) and indicate each of the residuals by a vertical line

Total Sum of Squares



Total Sum of Squares

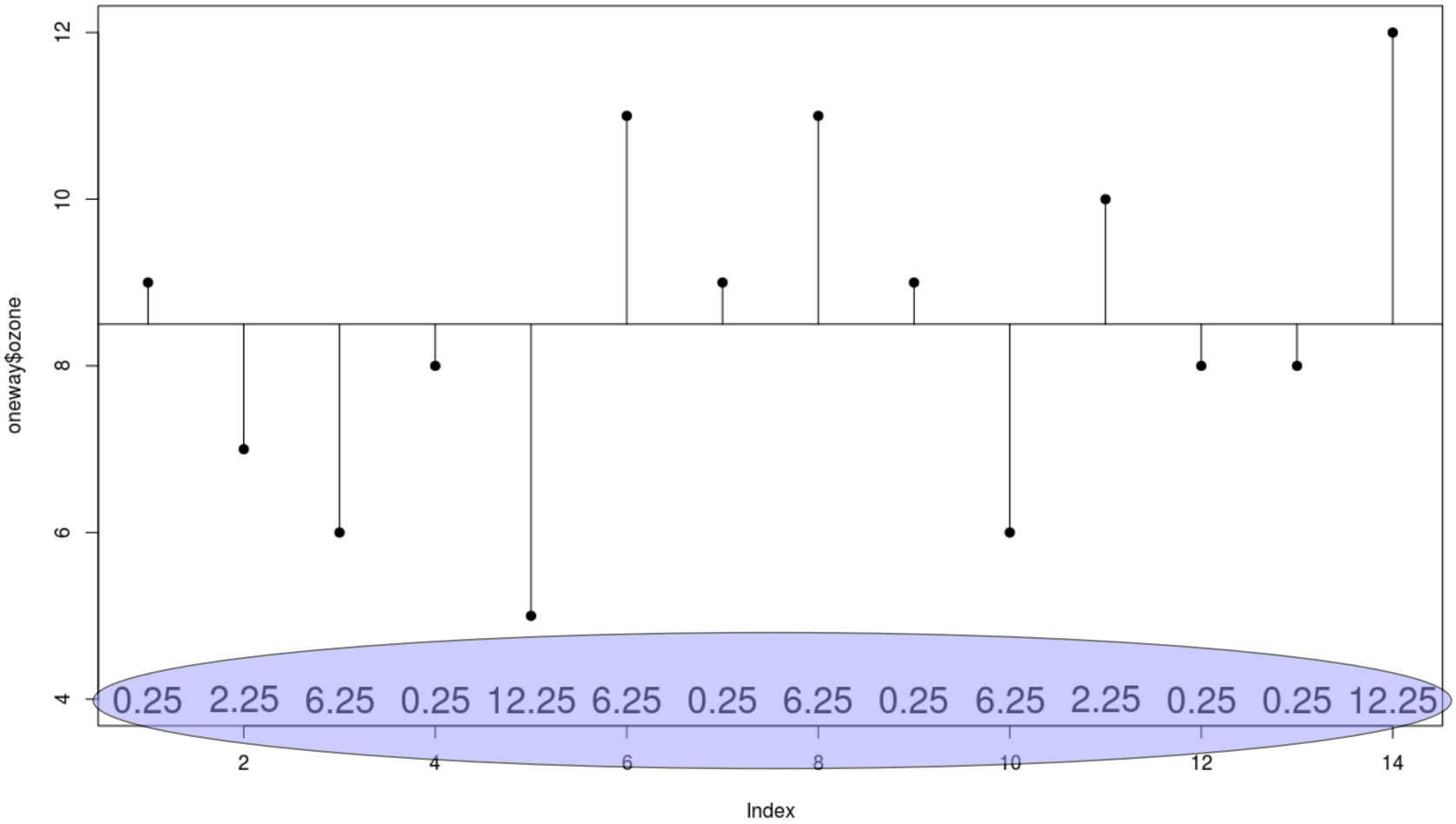
- we refer to this overall variation as the total sum of squares, SSY or TSS

$$SSY = \sum (y - \bar{y})^2$$

Total Sum of Squares

- in this case

$$SSY = 55.5$$

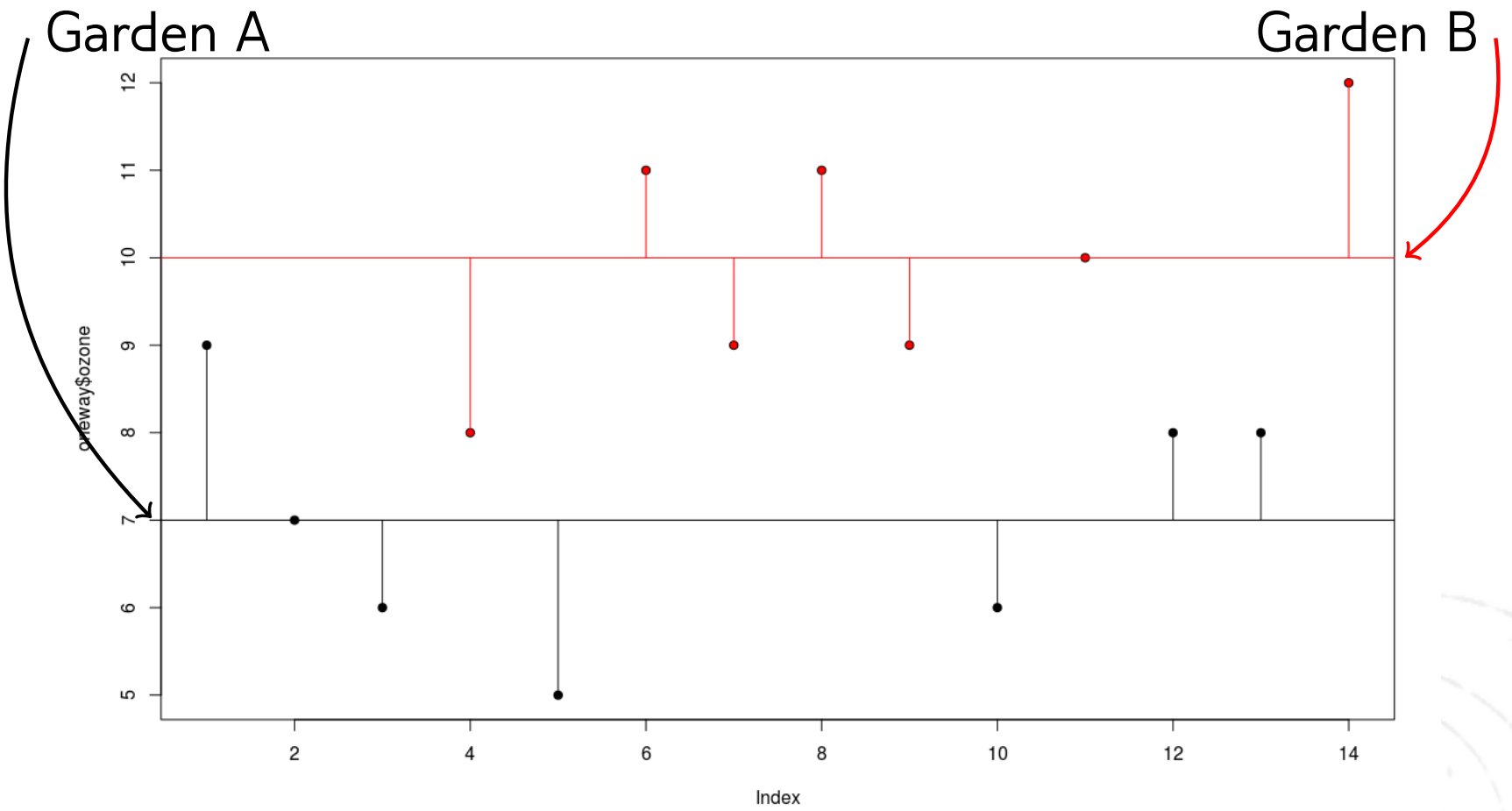


Group Means

- now instead of fitting the overall mean, let us fit the individual garden means

garden	a	b
mean	7	10

Group Means


Garden A
Garden B
oneway$ozone
Index
5
6
7
8
9
10
11
12
2
4
6
8
10
12
14

Group Means

- now we see that the mean ozone concentration is substantially higher in garden B
- the aim of ANOVA is to determine
  - whether it is significantly higher or
  - whether this kind of difference could come by chance alone

Error Sum of Squares

When the means are significantly different then the sum of squares computed from the individual garden means will be also substantially smaller than the sum of squares computed from the overall mean.

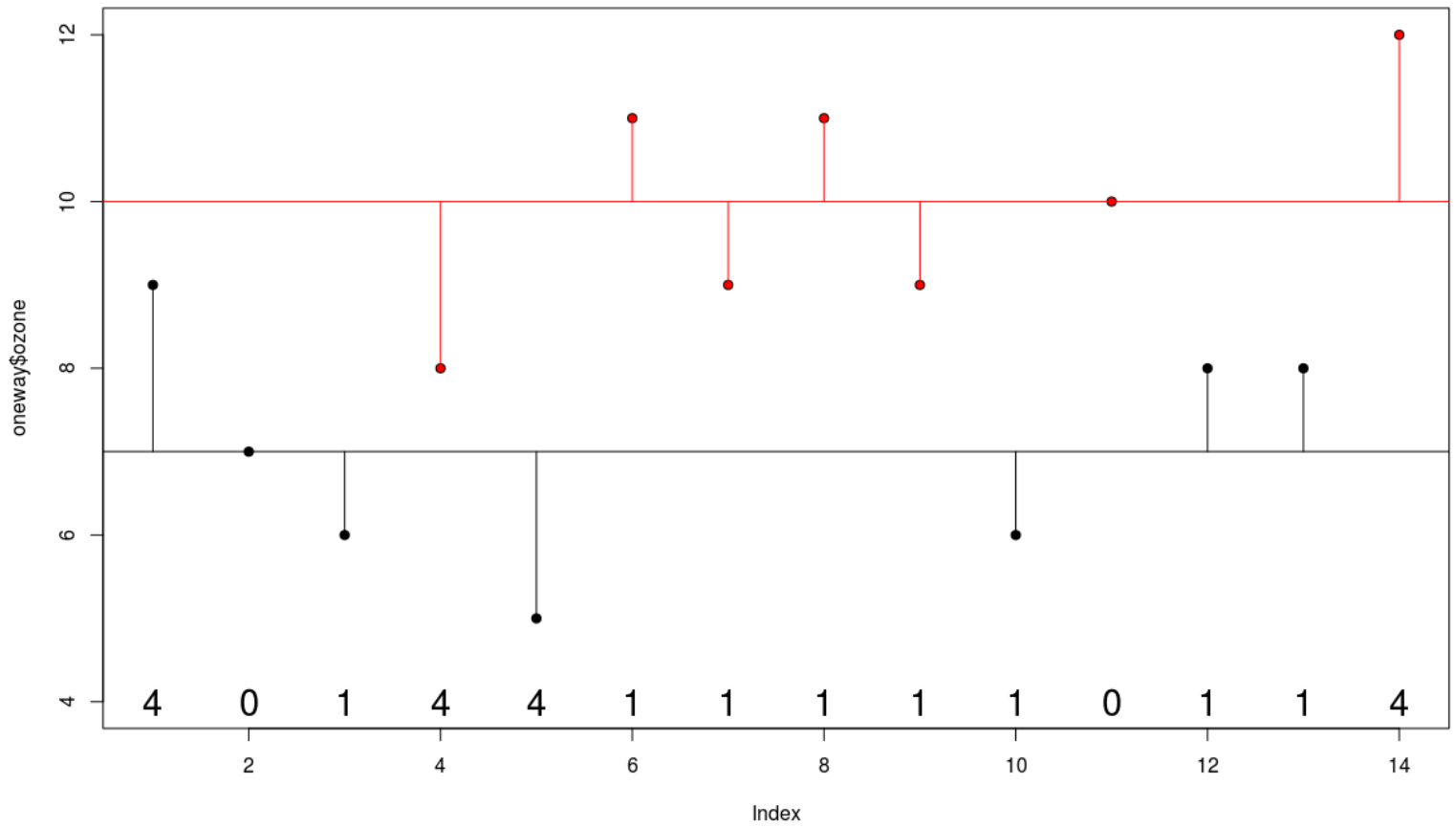
- we define the new sum of squares as the error sum of squares (error in the sense of 'residual')

$$SSE = \sum (y_{gardenA} - \bar{y}_{gardenA})^2 + \sum (y_{gardenB} - \bar{y}_{gardenB})^2$$

Total Sum of Squares

- in this case

$$SSE = 24.0$$



Treatment Sum of Squares

- then the component of the variation that is explained by the difference of the means is called the treatment sum of squares SSA
- analysis of variance is based on the notion that we break down the total sum of squares into useful and informative components

$$SSY = SSE + SSA$$

where

  - SSA = explained variation
  - SSE = unexplained variation

ANOVA table

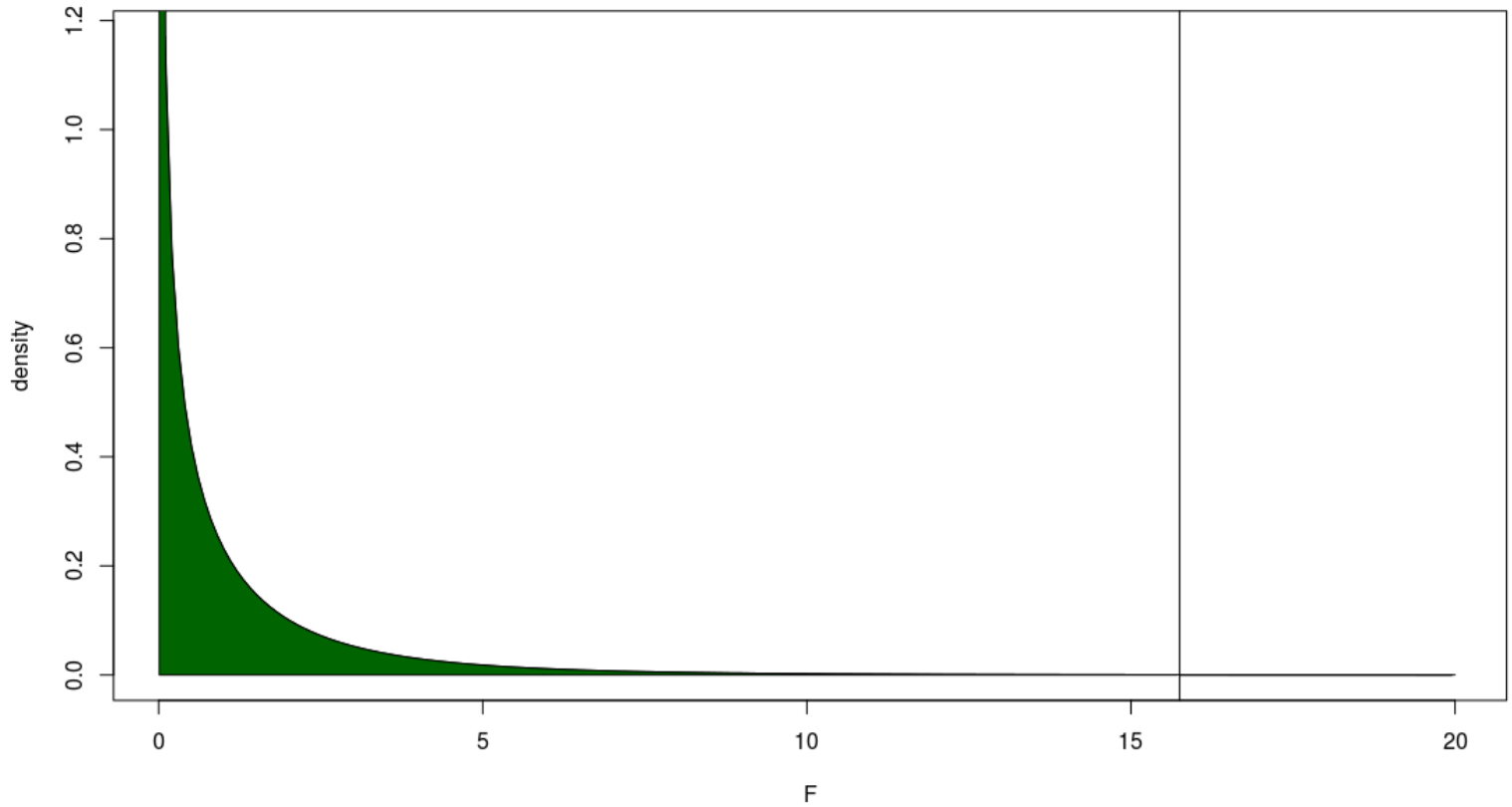
Source	Sum of squares	Degrees of freedom	Mean square	F ratio
Garden	31.5	1	31.5	15.75
Error	24.0	12	$s^2 = 2.0$	
Total	55.5	13		

ANOVA

- now we need to test whether an F ratio of 15.75 is large or small
- we can use a table or software package
- I use here software to calculate the cumulative probability

```
> 1 - pf(15.75,1,12)
[1] 0.001864103
```

ANOVA



ANOVA in R

- in R we use the `lm()` or the `aov()` command and
- the formula syntax `a ~b`
- we assign this to an variable

ANOVA in R

```
mm <- lm(ozone ~ garden, data=oneway)
mm

Call:
lm(formula = ozone ~ garden, data = oneway)

Coefficients:
(Intercept)      gardenb
          7            3
```

ANOVA in R

```
> summary(mm)

Call:
lm(formula = ozone ~ garden, data = oneway)

Residuals:
   Min     1Q Median     3Q    Max
    -2     -1      0      1      2

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.0000     0.5345  13.096 1.82e-08 ***
gardenb       3.0000     0.7559   3.969  0.00186 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.414 on 12 degrees of freedom
Multiple R-squared:  0.5676, Adjusted R-squared:  0.5315
F-statistic: 15.75 on 1 and 12 DF,  p-value: 0.001864
```

ANOVA in R

```
> anova(mm)
Analysis of Variance Table

Response: ozone
          Df Sum Sq Mean Sq F value   Pr(>F)
garden     1   31.5    31.5   15.75 0.001864 **
Residuals 12   24.0     2.0
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

ANOVA in R I

```
> m2 <- aov(ozone ~ garden, data=oneway)
> m2
Call:
   aov(formula = ozone ~ garden, data = oneway)

Terms:
                  garden Residuals
Sum of Squares      31.5      24.0
Deg. of Freedom        1        12

Residual standard error: 1.414214
Estimated effects may be unbalanced
> summary(m2)
             Df Sum Sq Mean Sq F value  Pr(>F)
garden        1   31.5    31.5   15.75 0.00186 **
Residuals    12   24.0     2.0
---
```

ANOVA in R II

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> summary.lm(m2)

Call:
aov(formula = ozone ~ garden, data = oneway)

Residuals:
   Min     1Q Median     3Q    Max
    -2     -1      0      1      2

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.0000     0.5345  13.096 1.82e-08 ***
gardenb       3.0000     0.7559   3.969  0.00186 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.414 on 12 degrees of freedom
Multiple R-squared:  0.5676, Adjusted R-squared:  0.5315
```

ANOVA in R III

```
F-statistic: 15.75 on 1 and 12 DF,  p-value: 0.001864

> summary(m2)
            Df Sum Sq Mean Sq F value  Pr(>F)
garden       1   31.5    31.5   15.75 0.00186 **
Residuals   12   24.0     2.0
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

ANOVA Assumptions

Central Assumptions

- independed, normal distributed errors
- equality of variances (homogeneity)

Welch ANOVA I

- generalization of the Welch t-test
- tests whether the means of the outcome variables are different across the factor levels
- assumes sufficiently large sample (greater than 10 times the number of groups in the calculation, groups of size one are to be excluded)
- sensitive to the existence of outliers (only few are allowed)
- the r command is `oneway.test()`
- non-parametric alternative `kruskal.test()` (Kruskal-Wallis test; assumes same distributional form) or `BDM.test()` - Brunner-Dette-Munk

Exercises

1. Look at the help of the TukeyHSD function. What is its purpose?
2. Execute the code of the example near the end of the help page, interpret the results!
3. install and load the granovaGG package (a package for visualization of ANOVAs), load the `arousal` data frame and use the `stack()` command to bring the data in the long form. Do a anova analysis. Is there a difference at least between 2 of the groups? If indicated do a post-hoc test.
4. Visualize your results

Exercises - Solutions I

1. Look at the help of the TukeyHSD function. What is its purpose?
2. Execute the code of the example near the end of the help page, interpret the results!
3. install and load the granovaGG package (a package for visualization of ANOVAs), load the `arousal` data frame and use the `stack()` command to bring the data in the long form. Do a anova analysis. Is there a difference at least 2 of the groups? If indicated do a post-hoc test.

```
> require(granovaGG)
> data(arousal)
> datalong <- stack(arousal)
> m1 <- aov(values ~ ind, data = datalong)
> summary(m1)
             Df Sum Sq Mean Sq F value   Pr(>F)
ind           3  273.4   91.13   10.51 4.17e-05 ***
Residuals    36  312.3    8.68
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

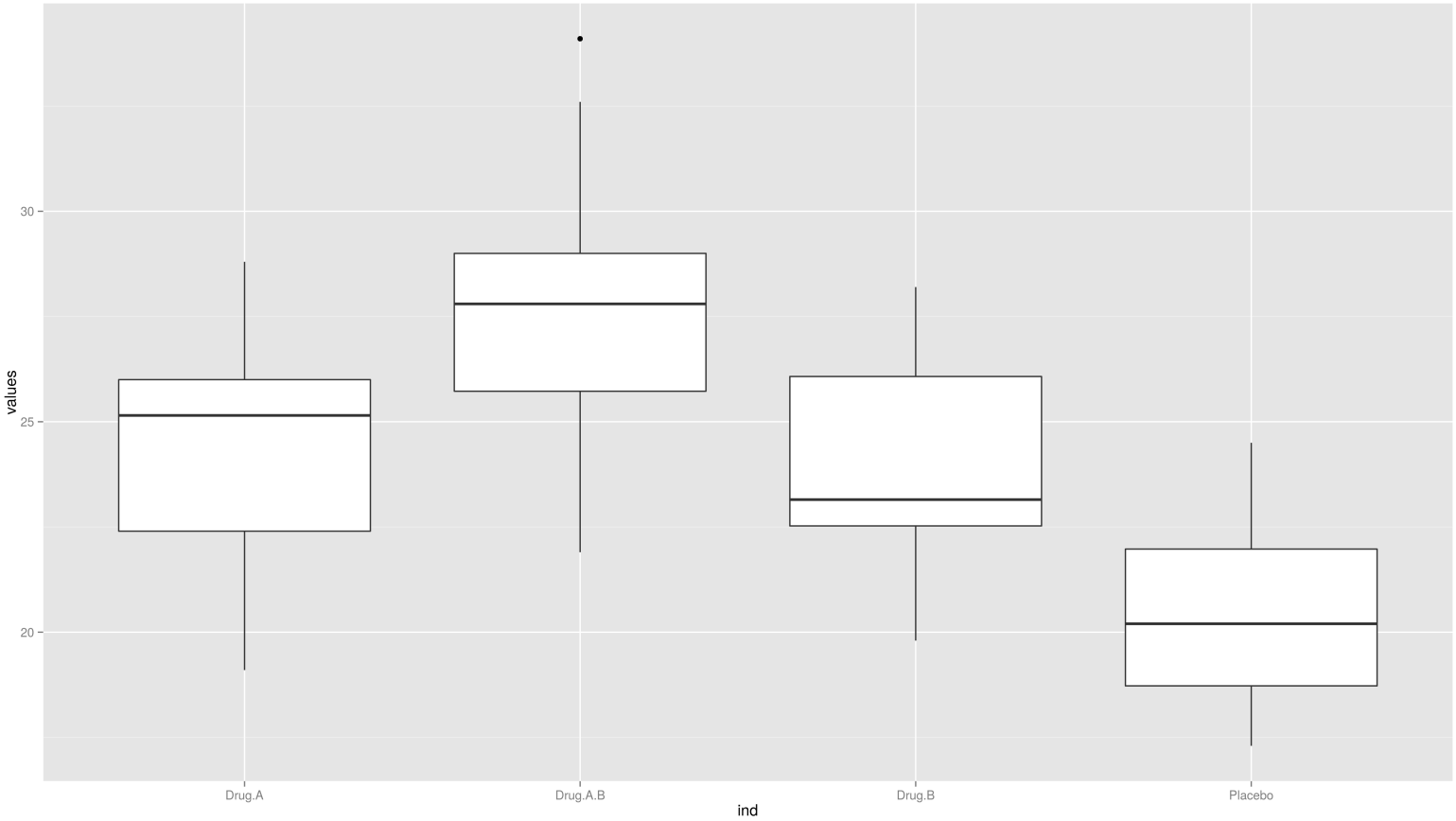
Exercises - Solutions II

```
> TukeyHSD(m1)
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = values ~ ind, data = datalong)

$ind
                    diff           lwr        upr     p adj
Drug.A.B-Drug.A    3.54  -0.007542384  7.0875424 0.0506601
Drug.B-Drug.A     -0.45  -3.997542384  3.0975424 0.9860554
Placebo-Drug.A    -3.84  -7.387542384 -0.2924576 0.0296168
Drug.B-Drug.A.B   -3.99  -7.537542384 -0.4424576 0.0223986
Placebo-Drug.A.B -7.38 -10.927542384 -3.8324576 0.0000137
Placebo-Drug.B    -3.39  -6.937542384  0.1575424 0.0654726
```

4. Visualize your results

```
> ggplot(datalong,aes(x=ind,y=values)) +
+        geom_boxplot()
```

Exercises - Solutions III



Exercises - Solutions IV

```
> granovagg.1w(datalong$values,group = datalong$ind)

By-group summary statistics for your input data (ordered by group means)
     group group.mean trimmed.mean contrast variance standard.deviation
4  Placebo      20.43        20.30    -3.65     5.83                2.41
3   Drug.B      23.82        23.85    -0.26     7.50                2.74
1   Drug.A      24.27        24.45     0.19     7.89                2.81
2 Drug.A.B      27.81        27.52     3.73    13.49                3.67
  group.size
4         10
3         10
1         10
2         10

Below is a linear model summary of your input data

Call:
lm(formula = score ~ group, data = owp$data)

Residuals:
   Min     1Q Median     3Q    Max
-5.910 -2.015 -0.075  1.885  6.290
```

Exercises - Solutions V

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    24.2700     0.9314  26.057  < 2e-16 ***
groupDrug.A.B   3.5400     1.3172   2.688  0.01083 *
groupDrug.B    -0.4500     1.3172  -0.342  0.73461
groupPlacebo   -3.8400     1.3172  -2.915  0.00608 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.945 on 36 degrees of freedom
Multiple R-squared:  0.4668, Adjusted R-squared:  0.4223
F-statistic:  10.5 on 3 and 36 DF,  p-value: 4.173e-05
```

Exercises - Solutions VI

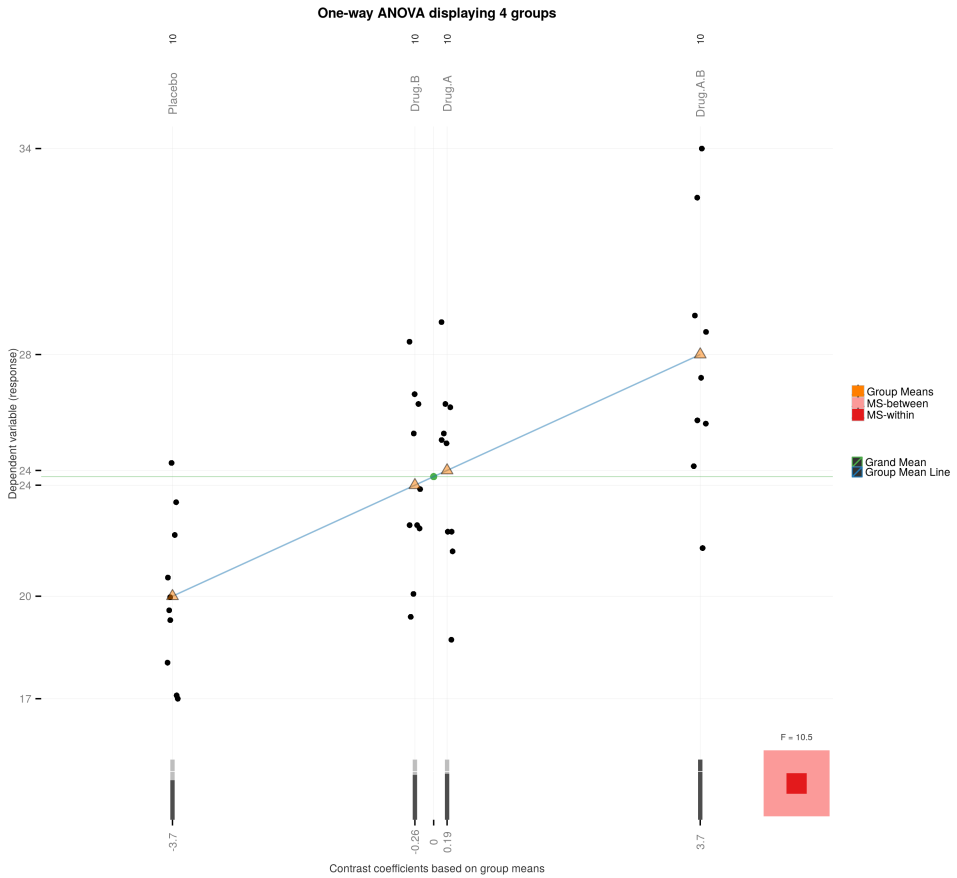


Table of Contents I

Permutation Tests

- dœs not rely on a assumed a priori distribution
- instead a empirical distribution is created from randomization of observed data
- robust against deviations from normality
- sensitive to differences in treatment variances
- scope of inference is limited to the sample (importance has been questioned by a number of authors)

Excurse - tapply()

- tapply() allows you to create tables (hence the "t") of the value of a function on subgroups defined by its second argument, which can be a factor or a list of factors.
- e.g. in the sleep data frame, we can summarize extra (increase in hours of sleep) classify by group (drug given) as follows:

Input/Output

```
> tapply(X = sleep$extra, INDEX = sleep$group, FUN = mean
   1    2
0.75 2.33
> tapply(sleep$extra,sleep$group,mean)
   1    2
0.75 2.33
```

- lapply() is a member of the apply functions
- the function replicate() is also a member of the apply family and used for replication (we have seen this in the

Permutation Tests - Example

- we use the `sleep` data set
- first we calculate the difference of the group means using the functions `lapply()` and `diff()`
- we use the absolute difference (this corresponds to a two-sided test)

Permutation Tests - Example

Input/Output

```
## use tapply to calculate the group means (not necessary
> tapply(sleep$extra,sleep$group,FUN = mean)
   1    2
0.75 2.33
> ## calculate the difference between the group means
> ## (not necessary)
> diff(tapply(sleep$extra,sleep$group,FUN = mean))
   2
1.58
> ## calculate the absolute difference (two-sided test)
> orig.diff <- abs(diff(tapply(sleep$extra,sleep$group,me
> orig.diff
   2
1.58
```

Permutation Tests - Example

Input/Output

```
> ## do it a 10000 times
> res <- replicate(10000,
+     abs(diff(tapply(sleep$extra,sample(sleep$group),mean
> sum(orig.diff <= res)/10000 ## p-value
[1] 0.0724
```

Permutation Tests - Example

Input/Output

```
> ## compare with t-test
> t.test(sleep$extra ~ sleep$group)

Welch Two Sample t-test

data:  sleep$extra by sleep$group
t = -1.8608, df = 17.776, p-value = 0.07939
alternative hypothesis: true difference in means is not e
95 percent confidence interval:
 -3.3654832  0.2054832
sample estimates:
mean in group 1 mean in group 2
           0.75            2.33
```

Rank-Based Permutation Tests

- requiring a single null distribution given a particular sample size
- much less sensitive to outliers compared to parametric methods
- scope of inference is generally less considered an issue
- slightly less powerful than parametric methods if their parametric assumptions hold
- computational problems of ties
- rank transformation throws out a large amount of information
- also sensitive to heteroscedasticity

Rank-Based Permutation Tests - Example

- consider to hypothesized populations X_1 and X_2
- assume two observations for X_1 and three observations for X_2
- one-tailed test: $H0 : X_1 \geq X_2$ versus $H_A : X_1 < X_2$
- in absence of ties we have $(n_1 + n_2)!/(n_1!n_2!) = 10$ possible ranks for X_1 or X_2
- what is the smallest possible p-value?

Table of Contents I

Wilcoxon tests

- the Wilcoxon tests the location of the median
- it is a non-parametric alternative to Student's t test
- it is based on the ranks – and really simple, e.g. for the two sample test: sort your data, give them ranks, sum up the ranks by group, take the smaller sum and look in a table for the appropriate row/column (with ties are dealt with by averaging the appropriate ranks)
- in R it is (not very surprisingly) `wilcox.test()`
- there is also a one and a two sample and a paired version

Wilcoxon Test

- the non-parametric test is much more appropriate when the errors are not normal,
- can be more powerful if the distribution is strongly skewed by the presence of outliers
- typically the t-test will give the lower p-value, so the Wilcoxon test is said to be conservative: if a difference is significant under a Wilcoxon test it would have been even more significant under a t-test.

Wilcoxon Signed Rank Test

- method for one sample or two dependent samples
- calculate the differences between the pairs of observations (or between values and hypothesized value)
- let n be the number of non-zero differences
- rank the absolute values of the n differences
- reassign the signs from step 1
- T_+ is the sum of the positive signed ranks, while T_- is the sum of the negative ranks
- for a two tailed test the minimum of these two is taken as test statistic, in a upper-tailed T_-, in a lower-tailed T_+

Signed Rank: Null Hypothesis

- $H_0 : \tau = 0$
- $H_A : \tau \neq 0$ (two-sided) or $H_A : \tau < 0$ (one-sided)
- where τ is the shift in location (NOT! the mean)

Wilcoxon Signed-Rank Test

```
> pre.test <- c(17,12,20,12,20,21,23,10,15,17,18,18)
> post.test <- c(19,25,18,18,26,19,27,14,20,22,16,18)
> wilcox.test(pre.test,post.test,paired = T)

Wilcoxon signed rank test with continuity correction

data:  pre.test and post.test
V = 7.5, p-value = 0.02527
alternative hypothesis: true location shift is not equal to 0

Warning messages:
1: In wilcox.test.default(pre.test, post.test, paired = T) :
  kann bei Bindungen keinen exakten p-Wert Berechnen
2: In wilcox.test.default(pre.test, post.test, paired = T) :
  kann den exakten p-Wert bei Nullen nicht berechnen
```

Wilcoxon Rank Sum Test

- non-parametric equivalent of a two-sample t-test procedure for independent samples
- hypothesis statements as above
- other alternative: Mann-Whitney-U test (inferential equivalent, i.e. also the same p-val for the same hypotheses)

Wilcoxon Rank Sum Test

Calculation of the test statistic (W) requires three steps

1. rank the data for both samples from the smallest to the largest
2. sum the ranks up for each group separately, get T_1 and T_2
3. calculate W_1 and W_2:

$$W_1 = T_1 - \frac{n_1(n_1+1)}{2}$$

and

$$W_2 = T_2 - \frac{n_2(n_2+1)}{2}$$

4. the test statistic W is the smaller one, the name of the distribution is the Wilcoxon rank sum distribution

Example/Exercise

- load the data `myeloma` data from the asbio package
- test microglobulin dependent on drug
- interpret the results

Example/Exercise

- go back to the `sc.twin` data set from yesterday
- now do the appropriate non-parametric test
- compare the results from both tests

Exercise

1. load the data frame normtemp, which is contained in the UsingR package; it contains the body temperature of several individuals, the gender and the heart rate
2. test if the temperature is different in male (coded as 1) and female (coded as 2), use the appropriate test.
3. test again, compare the results of the t test and the wilcoxon.
4. plot the respective boxplots!

Exercise I

1. load the `UsingR` package
2. The Simple data set `iq` contains simulated scores on a hypothetical IQ test. What analysis is appropriate for measuring the center of the distribution? Why?
3. The Simple data set `slc` contains data on red blood cell sodium-lithium countertransport activity for 190 individuals. Describe the shape of the distribution, estimate the center, state what is an appropriate measure of center for this data.
4. Load the Simple data set `vacation`. This gives the number of paid holidays and vacation taken by workers in the textile industry.
   - 4.1 Is a test for $\bar{y}$ appropriate for this data?
   - 4.2 Does a t-test seem appropriate?
   - 4.3 If so, test the null hypothesis that $\mu = 24$.

Exercise II

5. Repeat the above for the Simple data set `smokyph`. This data set measures pH levels for water samples in the Great Smoky Mountains. Use the waterph column `smokyph['waterph']` to test the null hypothesis that $\mu = 7$.

Table of Contents I

The births data

A data frame with 500 observations on the following 8 variables.

`id:`	Identity number for mother and baby.
`bweight:`	Birth weight of baby.
`lowbw:`	Indicator for birth weight less than 2500 g.
`gestwks:`	Gestation period.
`preterm:`	Indicator for gestation period less than 37 weeks.
`matage:`	Maternal age.
`hyp:`	Indicator for maternal hypertension.
`sex:`	Sex of baby: 1:Male, 2:Female.

From: Michæl Hills and Bianca De Stavola (2002). A Short Introduction to Stata 8 for Biostatistics, Timberlake Consultants Ltd URL: http://www.timberlake.co.uk

Variables in Models

The response variable must be numeric. Main types are
Explanatory variables can be

- Numeric
- Factor

Table of Contents I

Metric Response, Numeric explanatory variable

Assuming that the relationship of `bweight` with `gestwks` is roughly linear we can find the linear effect on `bweight` of a unit increase in `gestwks` with

```
> m <- lm(bweight ~ gestwks, data=births)
```

- `lm()` is the linear model function
- `bweight ~ gestwks` is the model formula
- `m` is a model object (containing all information about our model), there are certain functions to extract these information, e.g.:

```
> coef(m)
(Intercept)     gestwks
 -4489.1398    196.9726
```

One extra week of gestation produces an extra 197g of baby.

Extractor functions

```
> summary(m)

Call:
lm(formula = bweight ~ gestwks, data = births)

Residuals:
     Min       1Q   Median       3Q      Max
-1698.40  -280.14    -3.64   287.61  1382.24

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -4489.140    340.899  -13.17   <2e-16 ***
gestwks       196.973      8.788   22.41   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 449.7 on 488 degrees of freedom
  (10 observations deleted due to missingness)
Multiple R-squared: 0.5073, Adjusted R-squared: 0.5062
F-statistic: 502.4 on 1 and 488 DF,  p-value: < 2.2e-16
```

Extractor functions

```
> coef(m)
(Intercept)     gestwks
 -4489.1398    196.9726
> confint(m)
                  2.5 %     97.5 %
(Intercept) -5158.9503 -3819.3293
gestwks       179.7054   214.2399
```

Other Useful Functions

The model object is a list of different elements each of which can be accessed separately (see `str(m)` for the full list).
Other useful functions:

- `print(m)` simple display
- `plot(m)` produces various diagnostic plots based on residuals
- `fitted(m)` returns a vector of fitted values
- `resid(m)` returns a vector of residuals
- `predict(m, newdata)` predicts the response for new values of the explanatory variables
- `deviance(m)` residual sum of squares
- `df.residual(m)` for the residual degrees of freedom
- `vcov(m)` variance-covariance matrix

Explanatory Variable is a Factor

The effect of `hyp` (2-level factor) on `bweight` is obtained with

```
> m <- lm(bweight ~ hyp, data=births)
> coef(m)
(Intercept)    hyphyper
  3198.9042   -430.6959
```

Omitting the intercept gives the mean `bweight` at the two levels of `hyp`

```
> m <- lm(bweight ~ -1 + hyp, data=births)
> coef(m)
hypnormal  hyphyper
 3198.904  2768.208
```

A Multivariable Model

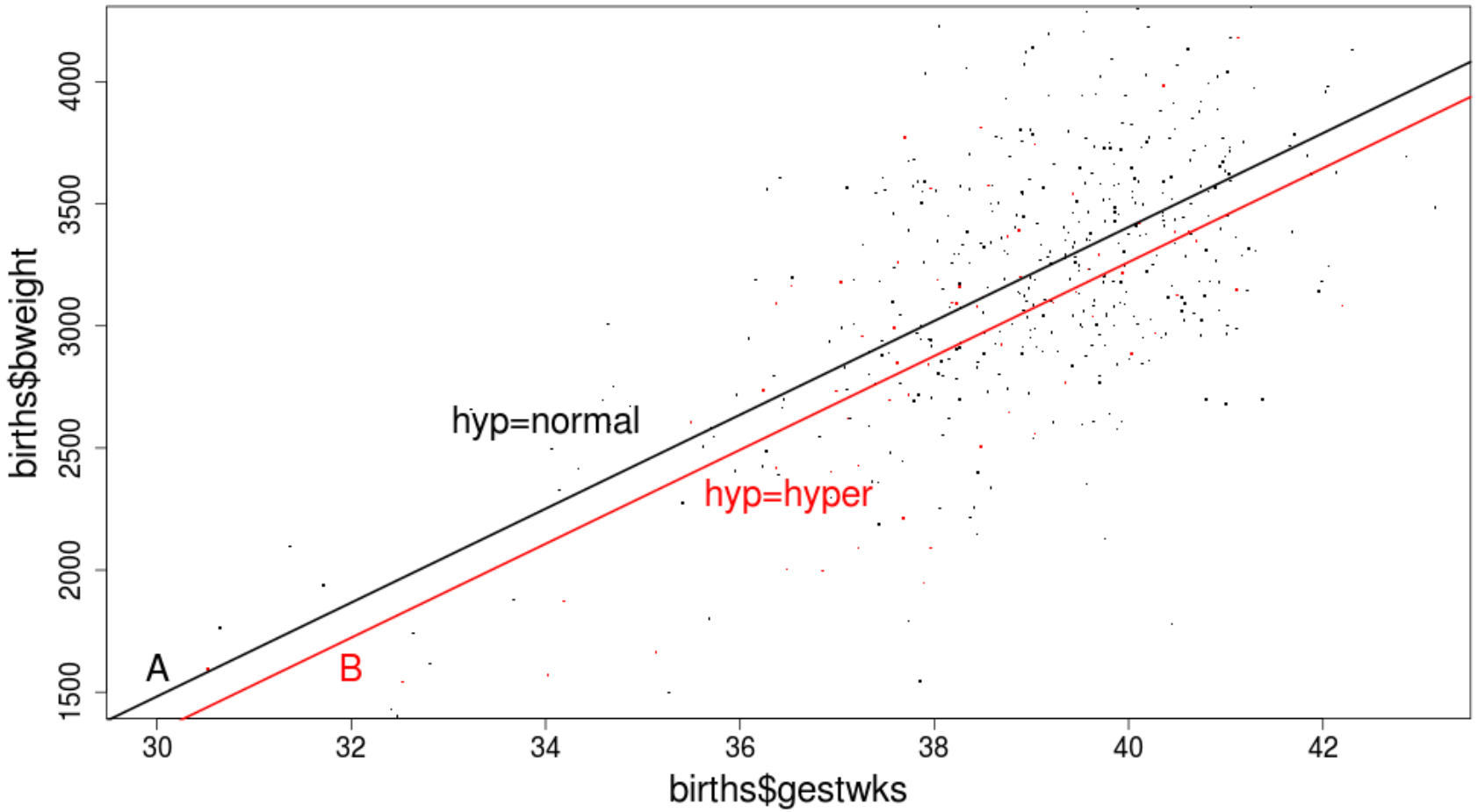
The joint effect of `hyp` and `gestwks` on `bweight` is obtained with

```
> m <- lm(bweight ~ hyp + gestwks, data=births)
```

```
             Estimate
(Intercept) -4285.002
hyphyper     -143.675 (level 2 vs. level 1)
gestwks       192.238 (increase per week)
```

The effect of `hyp` is attenuated (from -430.7 to -143.7). This suggests that much of the effect of hypertension on birth weight is mediated through a shorter gestation period.

A Model With Both `gestwks` and `hyp`



The effect of `gestwks` is the slope of the lines A and B (assumed to be the same). The effect of `hyp` ist the vertical distance between them.

Interaction Models in lm

To specify an interaction term in lm, change the model formula from

Input

```
> m <- lm(bweight ~ hyp + gestwks, data=births)
```

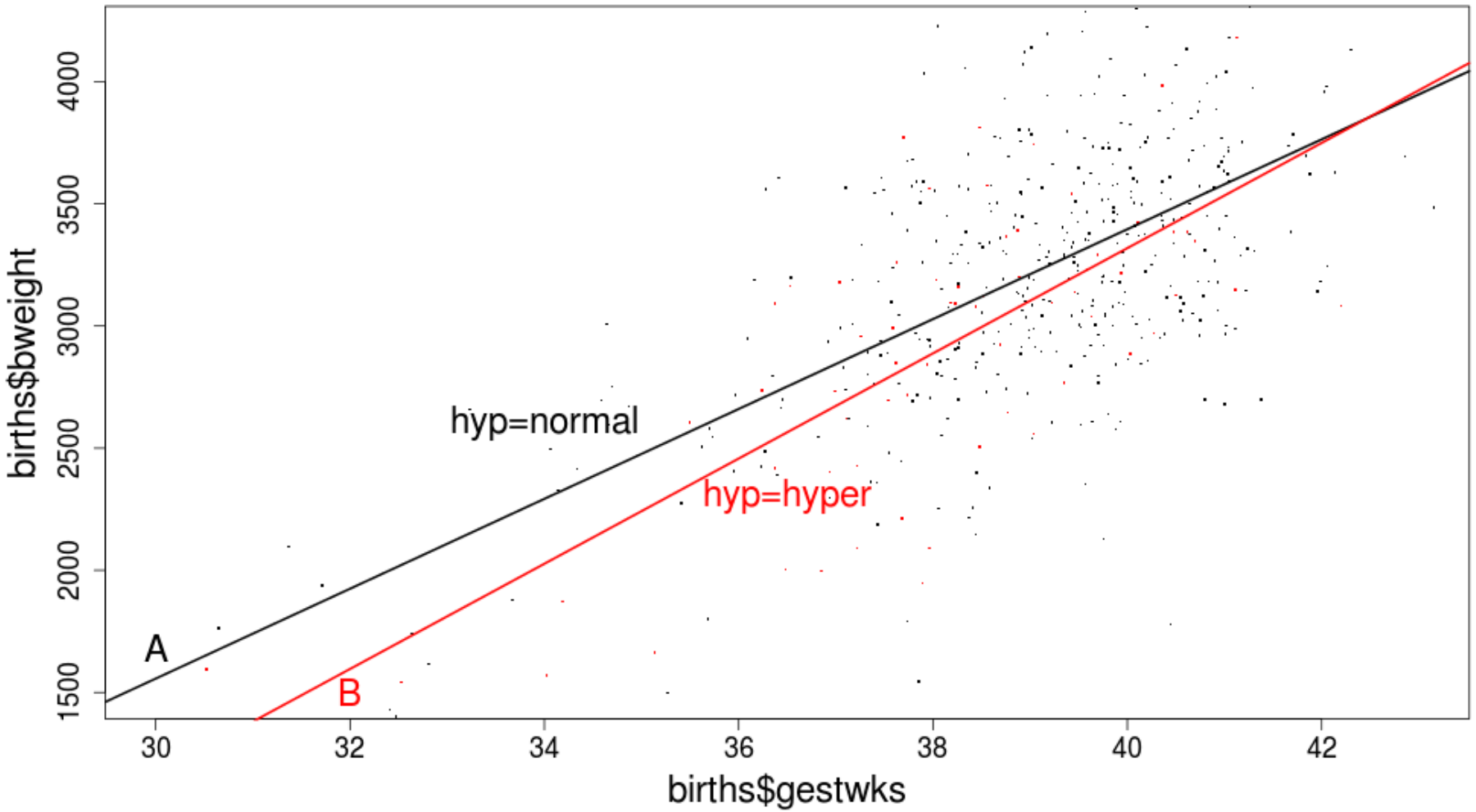
to

```
> m <- lm(bweight ~ hyp + gestwks + hyp:gestwks, data=bi
```

or shorter

```
> m <- lm(bweight ~ hyp * gestwks, data=births)
```

Interaction Between gestwks and hyp



Interactions Models in lm

Output

```
                  Estimate
(Intercept)       -3960.82
hyphyper          -1332.66 (level 2 vs level 1 - inter
gestwks             183.91
hyphyper:gestwks     31.39 (level 2 vs level 1 - slope
```

Now the effect of hyp more difficult to explain, because it is not constant. The effect of -1332 is valid on a hypothetical gestational age of 0. Which dœsn't make sense. You could scale the gestwks variable.

```
> births$gwsc <- births$gestwks-40
> m <- lm(bweight ~ hyp * gwsc, data=births)
```

Interactions Models in lm

Input/Output

```
                Estimate
(Intercept)   3395.60329
hyphyper       -77.25215 (level 2 vs level 1 - interc
gwsc           183.91048
hyphyper:gwsc   31.38510 (level 2 vs level 1 - slope)
```

How much is explained? - aov

In the Null-Model we have seen that $SSE = SSY$ (the error sum of squares is equal to the total sum of squares in y) and therefore the Null-Model explaines nothing of the overall variance. So the fraction how much of the overall variance is explained by our model regarding to the overall variance is a first measure for the fit of the model...

- the simple model with one explanatory variable

```
> m <- lm(bweight ~ gestwks, data=births)
> anova(m)
Analysis of Variance Table

Response: bweight
            Df    Sum Sq   Mean Sq F value    Pr(>F)
gestwks      1 101603845 101603845  502.36 < 2.2e-16 ***
Residuals  488  98698698    202251
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’
```

How much is explained? - aov

- in the second column of the summary we see the regression sum of squares (SSR) in the first line and in the second line the error sum of squares (SSE). So the total sum of squares (SSY - a measure for the overall variation) is the sum of both:

```
> sum(anova(m)$Sum)
[1] 200302543
```

- and the fraction is

```
> anova(m)$Sum[1]/sum(anova(m)$Sum)
[1] 0.5072519
```

How much is explained? - aov

- this is r-squared

```
> summary(m)$r.squared
[1] 0.5072519
```

- which you can extract from the summary of the model

```
> summary(m)
Call:
lm(formula = bweight ~ gestwks, data = births)
Residuals:
     Min       1Q   Median       3Q      Max
-1698.40  -280.14    -3.64   287.61  1382.24
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -4489.140    340.899  -13.17   <2e-16 ***
gestwks       196.973      8.788   22.41   <2e-16 ***

Residual standard error: 449.7 on 488 degrees of freedom
  (10 observations deleted due to missingness)
Multiple R-squared: 0.5073, Adjusted R-squared: 0.5062
F-statistic: 502.4 on 1 and 488 DF,  p-value: < 2.2e-16
```

Exercises I

1. load the nhanes data
2. how many observations, how many variables?
3. how old are the participants (summary statistics, mean, sd)
4. plot waist circumference vs age
5. model the respective data in a linear model, extract and interpret the cœfficients. Extract also the confidence intervals.
6. add sex as a covariate. interpret.

Table of Contents I

Generalized Linear Models

Input

```
> m <- lm(bweight ~ hyp, data=births)
> m <- glm(bweight ~ hyp, family=gaussian, data=birth
```

give the same answer. The model formula is the same for both, but for `glm` it is necessary to specify the family of likelihoods which will be used to fit the model.
The `glm` function allows us to fit other models including logistic regression and Poisson regression.

Predicting Low Birth Weight

We are more interested in predicting birth weight under 2500g (`lowbw`). This requires a model where the outcome is not metric, but binary. For a binary response we use a `glm` with a binomial family.

Input/Output

```
> m <- glm(lowbw ~ hyp, family=binomial, data=births)
> ci.lin(m, Exp=T)[,5:7]
             exp(Est.)       2.5%     97.5%
(Intercept) 0.1030928 0.07445162 0.1427521
hyphyper    3.7307692 2.02747522 6.8650107
```

This returns estimates of the log odds (Intercept) or log odds ratios (for the parameters). To present the results in terms of odds ratios we use the `Exp=TRUE` option to `ci.lin`.

Controlling

Controlling the effect of `hyp` on `lowbw` for `sex`

Input/Output

```
> m <- glm(lowbw ~ hyp+sex, family=binomial, data=births)
> ci.lin(m,Exp=T)
                exp(Est.)
(Intercept)     0.0813691
hyphyper        3.9060041 (hyp controlled for sex)
sexF            1.5641095 (sex controlled for hyp)
```

When you control for a variable you are assuming that any interaction can be ignored.

Interaction (effect modification)

Input/Output

```
> m <- glm(lowbw ~ hyp + sex + hyp:sex,
+          family=binomial, data=births)
> ci.lin(m,Exp=T)[,5:7]
                exp(Est.)
(Intercept)    0.07281553
hyphyper       5.31612903
sexF           1.88644689
hyphyper:sexF  0.52168285
```

Alternatively, use

Input/Output

```
m <- glm(lowbw ~ hyp*sex, family=binomial, data=birth
```

Testing for Interaction

Input/Output

```
> m1 <- glm(lowbw ~ hyp+sex, family=binomial, data=bi
> m2 <- glm(lowbw ~ hyp*sex, family=binomial, data=bi
> anova(m1,m2,test="Chisq")

  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       497     348.34
2       496     347.29  1   1.0561   0.3041
```

The anova function conducts an analysis of variance – an old-fashioned name for a test of significance between two nested models.

Stratified Effects

When there is a strong interaction it may be best to report stratified effects. Omitting the main effect of `hyp` in an interaction model gives us the effect of `hyp` within strata of `sex`.

Input/Output

```
m <- glm(lowbw ~ sex + sex:hyp, family=binomial,
+                                data=births)
> ci.lin(m,Exp=T)[,5:7]
                exp(Est.)
(Intercept)   0.07281553 % 15/206 nur normale Jungen
sexF          1.88644689
sexM:hyphyper 5.31612903
sexF:hyphyper 2.77333333
```

Note that $2.77/5.32 = 0.52$ is the interaction term.

Looking Inside the Black Box

The paradigm is the model

$$\mu = \alpha + \beta X + \gamma Z + \cdots$$

where X,Z,$\cdots$ are numeric explanatory variables. In a glm μ is replaced by some function of mu such as $\log(\mu)$ (link function). When X is a factor, on (say) 3 levels, it is replaced by X_1, X_2, X_3, die indicator variables for the levels of X. Predicted values for $\alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$ are

level	X_1	X_2	X_3	$\alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$
1	1	0	0	$\alpha + \beta_1$
2	0	1	0	$\alpha + \beta_2$
3	0	0	1	$\alpha + \beta_3$

Too Many Parameters

Drop α

level	X_1	X_2	X_3	$\beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$
1	1	0	0	β_1
2	0	1	0	β_2
3	0	0	1	β_3

β_1 is the mean response at level 1, β_2 at level 2, β_3 at level 3.

Drop X_1

level	X_2	X_3	$\alpha + \beta_2 X_2 + \beta_3 X_3$
1	0	0	α
2	1	0	$\alpha + \beta_2$
3	0	1	$\alpha + \beta_3$

α is the mean response at level 1

β_2 und β_3 are the effects of levels 2 and 3 vs level 1. These are called treatment contrasts.

Two Factors

X on 3 levels, Z on 2 levels

$$\mu = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \gamma_1 Z_1 + \gamma_2 Z_2$$

X_1, X_2, X_3 are the indicators for X and Z_1, Z_2 are the indicators for Z. Omitting X_1 and Z_1 the model becomes

$$\mu = \alpha + \beta_2 X_2 + \beta_3 X_3 + \gamma_2 Z_2$$

with predicted means

		Z	
		1	2
X	1	α	$\alpha + \gamma_2$
	2	$\alpha + \beta_2$	$\alpha + \beta_2 + \gamma_2$
	3	$\alpha + \beta_3$	$\alpha + \beta_3 + \gamma_2$

Interaction

Effect of Z the same at each level of X:

		Z	
		1	2
X	1	α	$\alpha + \gamma_2$
	2	$\alpha + \beta_2$	$\alpha + \beta_2 + \gamma_2$
	3	$\alpha + \beta_3$	$\alpha + \beta_3 + \gamma_2$

Effect of Z differs at different levels of X:

		Z	
		1	2
X	1	α	$\alpha + \gamma_2$
	2	$\alpha + \beta_2$	$\alpha + \beta_2 + \gamma_2 + \delta_{22}$
	3	$\alpha + \beta_3$	$\alpha + \beta_3 + \gamma_2 + \delta_{32}$

The δ parameters measure how much the effect of Z changes.

Nested or Stratified Effects

A slightly different way of parameterizing the model gives stratified effects:

		Z	
		1	2
X	1	β_1	$\beta_1\delta_{12}$
	2	β_2	$\beta_2 + \delta_{22}$
	3	β_3	$\beta_3 + \delta_{32}$

Same number of parameters as for interaction, but the δ's now measure the effects of Z at each level of X. In `R` this would be produced by the model formula `Y ~ -1 + X + X:Z`