

# Experimental Design and Data Analysis

## Lecture 2

Eduard Belitser

VU Amsterdam

# Lecture overview

- ① bootstrap confidence intervals
- ② bootstrap tests
- ③ one sample (two paired samples) tests for normal and not normal samples
  - $t$ -test
  - sign test
  - Wilcoxon signed rank test

bootstrap CI  
●oooooooo

bootstrap tests  
ooooooo

t-test: one sample/paired samples  
oooooooooo

one sample/paired samples, not normal  
oooooooo

## bootstrap confidence intervals

# Confidence interval for normal data

A **point estimate** for an unknown parameter  $\mu$  is some function of the data.

**EXAMPLE** For a sample  $X_1, \dots, X_n \sim N(\mu, \sigma^2)$ , we can **estimate**  $\mu$  using the estimating statistic  $\bar{X}$ . The point estimate for  $\mu$  is thus  $\hat{\mu} = \bar{X}$ .

Recall that a **confidence interval** for an unknown parameter  $\mu$  is a **random interval** around the point estimate, containing  $\mu$  with, e.g., 95% confidence.

**EXAMPLE (continued)** An (asymptotic) 95%-confidence interval for  $\mu$  is the interval  $[\bar{X} - m, \bar{X} + m]$ , where  $m = 1.96s/\sqrt{n}$ .

The margin  $m = 1.96s/\sqrt{n}$  is based on the asymptotic normality of  $\bar{X}$  and the fact that  $s$  is a good estimator of  $\sigma$ . If in the CI we use the upper  $t$ -quantile  $t_{0.025, n-1}$  instead of  $z_{0.025} \approx 1.96$ , the CI will be bigger (i.e., more “conservative”) because always  $t_{\alpha, n-1} > z_{\alpha}$ , but  $t_{\alpha, n-1} \rightarrow z_{\alpha}$  as  $n \rightarrow \infty$ .

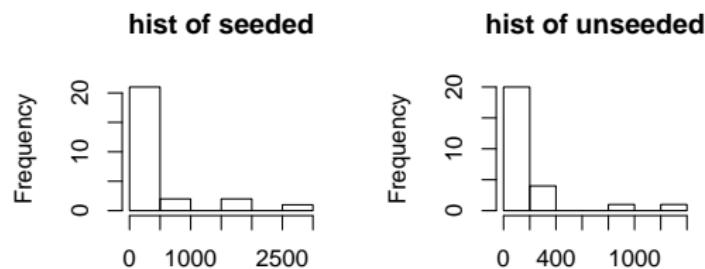
## Example of non-normal sample: clouds data sets

**EXAMPLE** Cloud seeding is a technique used to change the amount and type of precipitation, by dispersing substances into clouds. Precipitation values of **seeded** (with a chemical, silver nitrate, to cause a rainfall) and **unseeded** clouds were measured. We want to construct CI's for the rainfall means of the two clouds data sets.

Not all data can be assumed to come from a (appr.) normal distribution.

Histograms and QQ-plots can be used to [check the normality assumption](#).

```
> c1=clouds[,1];hist(c1) #seeded  
> c2=clouds[,2];hist(c2) #unseeded  
> T1=mean(c1); T2=mean(c2)  
> T1 # rainfall mean of seeded  
[1] 441.9846  
> T2 # rainfall mean of unseeded  
[1] 164.5619
```



Assuming normality here is clearly [wrong](#). It is not always reasonable to rely on the CI based on (asymp.) normality. [Is there an alternative to determine CI's?](#)

# Bootstrap confidence interval

- Suppose we have a data sample  $X = (X_1, \dots, X_n)$  and an estimating statistic  $T = T(X_1, \dots, X_n)$  for a parameter, say,  $\theta$ .
- We use **simulation** to find the distribution of the estimating statistic  $T(X)$ . The **bootstrap CI** is then found from this simulated distribution.
- The bootstrap method estimates the distribution of  $T$  by **creating a sample of representative values**  $T_1^*, \dots, T_B^*$  with  $B$  large.
- The **basic bootstrap confidence interval** of level  $1 - \alpha$  is

$$[2T - T_{(1-\alpha/2)}^*, 2T - T_{(\alpha/2)}^*],$$

where  $T_{(\beta)}^*$  is the  $T^*$ -value such that  $\beta \times 100\%$  of the  $T^*$ -values are lower than  $T_{(\beta)}^*$ .  $T_{(\beta)}^*$  is called the **sample  $\beta$ -quantile** of the sample  $T_1^*, \dots, T_B^*$ .

In R: the sample  $\beta$ -quantile of  $T^* = (T_1^*, \dots, T_B^*)$  is  $T_{(\beta)}^* = \text{quantile}(T^*, \beta)$ .

- The bootstrap estimate for the variance of statistics  $T(X)$  is given by

$$\widehat{\text{Var}}(T) = S_{T^*}^2 = \frac{1}{B-1} \sum_{b=1}^B (T_b^* - \overline{T^*})^2. \quad \text{In R: } S_{T^*}^2 = \text{var}(T^*).$$

This bootstrap CI is constructed in such a way that it uses  $T$ . A simpler version of bootstrap CI (called **percentile bootstrap CI**):  $[T_{(\alpha/2)}^*, T_{(1-\alpha/2)}^*]$ .

bootstrap CI  
oooo●ooo

bootstrap tests  
ooooooo

t-test: one sample/paired samples  
oooooooo

one sample/paired samples, not normal  
oooooooo

## Heuristics for basic bootstrap CI

We interpret  $T_1^*, \dots, T_B^*$  as realizations of some random variable  $T^*$ . Then

$$\begin{aligned}1 - \alpha &\approx P\left(T_{(\alpha/2)}^* \leq T^* \leq T_{(1-\alpha/2)}^*\right) \quad (\text{percentile bootstrap CI } [T_{(\alpha/2)}^*, T_{(1-\alpha/2)}^*]) \\&= P\left(T_{(\alpha/2)}^* - T \leq T^* - T \leq T_{(1-\alpha/2)}^* - T\right) \\&\approx P\left(T_{(\alpha/2)}^* - T \leq T - \theta \leq T_{(1-\alpha/2)}^* - T\right) \\&= P\left(2T - T_{(1-\alpha/2)}^* \leq \theta \leq 2T - T_{(\alpha/2)}^*\right),\end{aligned}$$

which gives us the basic bootstrap confidence interval for  $\theta$ :

$$[2T - T_{(1-\alpha/2)}^*, 2T - T_{(\alpha/2)}^*].$$

A simpler version of bootstrap CI (**percentile bootstrap CI**):  $[T_{(\alpha/2)}^*, T_{(1-\alpha/2)}^*]$ .

# How to generate $T^*$ -values

The generation of  $T^*$  values is as follows.

Repeat  $B$  times ( $i = 1, \dots, B$ ):

- generate a surrogate data set  $X_1^*, \dots, X_n^*$  by sampling  $n$  values from the original data set  $X_1, \dots, X_n$  **with replacement**,
- compute  $T_i^* = T(X_1^*, \dots, X_n^*)$  for the surrogate sample.

This procedure yields  $T_1^*, \dots, T_B^*$ .

Notice that we sample from the data that we have. Some data points  $X_i$  may be chosen more than once amongst the  $X^*$ -values, whereas other data points  $X_i$  may not be chosen at all. We do not introduce any new  $X$ -values, we only determine new  $T^*$ -values. This bootstrap procedure is called **empirical bootstrap**.

How many different resamples are possible from a sample of size  $n$ ? The number of ways to place  $n$  objects into  $n$  bins (some bins may be empty,  $i$ -th bin contains the copies of  $X_i$ ). The method of stars and bars yields  $\binom{2n-1}{n-1} = \binom{2n-1}{n}$ .

If you want a reference and a rule of thumb for  $B$ , Wilcox (2010) writes "599 is recommended for general use."

# Bootstrap CI in R: example with cloud sets

EXAMPLE (continued) Determine this interval for the seeded clouds (c1):

```
> B=1000
> Tstar=numeric(B)
> for(i in 1:B) {
+   Xstar=sample(c1,replace=TRUE)
+   Tstar[i]=mean(Xstar) }
> Tstar25=quantile(Tstar,0.025)
> Tstar975=quantile(Tstar,0.975)
> sum(Tstar<Tstar25)
[1] 25
> c(2*T1-Tstar975,2*T1-Tstar25)
176.8857 668.9462
```

generate  $X_1^*, \dots, X_n^*$   
compute  $T_b^*, b = 1, \dots, B$   
determine  $T_{(\alpha/2)}^*$   
determine  $T_{(1-\alpha/2)}^*$

The 95% bootstrap confidence interval for the population mean of seeded clouds is [177, 669] around its mean T1=442.

For unseeded clouds the interval is [42, 254] around its mean T2=165.

## Example with cloud sets: discussion

- The smaller the CI, the better. The obtained two CI's are very large, because the estimating statistic  $\bar{X}$  is not robust against outliers.
- A **robust** estimator for location is `median(X)`, estimating the **population median**. For the clouds data, the median is **smaller** than the mean.
- The 95% bootstrap CI for the **median** of seeded clouds is [139, 326] ([177, 669] for the mean); unseeded clouds: [-20, 62] ([42, 254] for the mean).
- For both data sets: the CI for the median is **shorter** and contains **lower** values. This confirms that the median is more robust than the mean.

### General discussion on bootstrap confidence intervals

- Repeating the computation of a bootstrap confidence interval will always yield a different interval. Enlarging  $B$  will reduce the variation.
- The bootstrap interval still depends only on the **sample**  $X_1, \dots, X_n$ .
- If the original data  $X_1, \dots, X_n$  carries little information about the parameter  $\theta$ , the bootstrap interval will be off as well.

bootstrap CI  
ooooooooo

bootstrap tests  
●oooooo

t-test: one sample/paired samples  
oooooooooooo

one sample/paired samples, not normal  
oooooooooooo

## bootstrap tests

# Idea

- Suppose we are given
  - a sample  $X_1, \dots, X_n$ ,
  - a null hypothesis  $H_0$ : some claim about the population distribution,
  - a (sensible) test statistic  $T = T(X_1, \dots, X_n)$ ,
- but we lack
  - the distribution of  $T$  under  $H_0$ .
- Then we cannot perform the test, because we do not have a critical value for  $T$ , that acts as border between rejecting and not rejecting  $H_0$ .
- But if we somehow can simulate “pseudo-observations” characterizing  $H_0$ , we can use a bootstrap test.
- It uses simulations to "mimic" the distribution of  $T$  under  $H_0$ .

For a bootstrap test, no standard R-command — we have to program it ourselves.

# Set up of a bootstrap test

Given our sample  $X_1, \dots, X_n$ , we can compute the test statistic  $T = T(X_1, \dots, X_n)$  based on our sample.

Simulating the distribution of  $T$  under  $H_0$  in the bootstrap fashion means generate a bunch of surrogate  $T$ -values  $(T_1^*, \dots, T_B^*)$  that are representative values for  $T$  under  $H_0$ .

The simulation set up is

- repeat  $B$  times ( $i = 1, \dots, B$ ):
  - ① generate a surrogate data sample  $X_1^*, \dots, X_n^*$  (of the same sample size as original data set) according to  $H_0$ ,
  - ② Compute the test statistic  $T_i^* = T(X_1^*, \dots, X_n^*)$  for the surrogate sample.
- compare the  $T$ -value of the original data to the surrogate  $T^*$ -values and determine a  $p$ -value.

By simulating the unknown distribution we make an estimation error. This error can be made arbitrarily small by choosing  $B$  large enough.

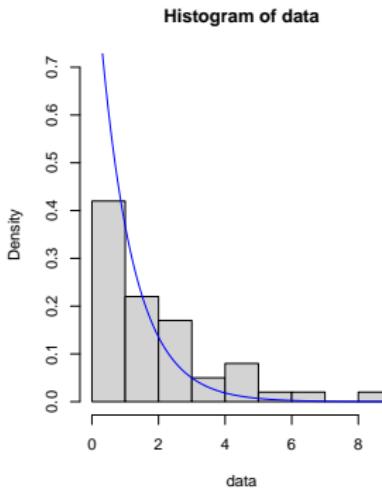
# Bootstrap test: implementation in R (1)

We wish to test  $H_0 : X_i \stackrel{iid}{\sim} \text{Exp}(1)$ ,  $i = 1 \dots, n$ , i.e. the data are a random sample from the standard exponential distribution.

First plot a histogram of the data and the density  $\text{Exp}(1)$  corresponding to  $H_0$ .

```
> hist(data,prob=T,ylim=c(0,0.7))
> x=seq(0,max(data),length=1000)
> lines(x,dexp(x),type="l",col="blue")
```

$H_0$  seems plausible.



## Bootstrap test: implementation in R (2)

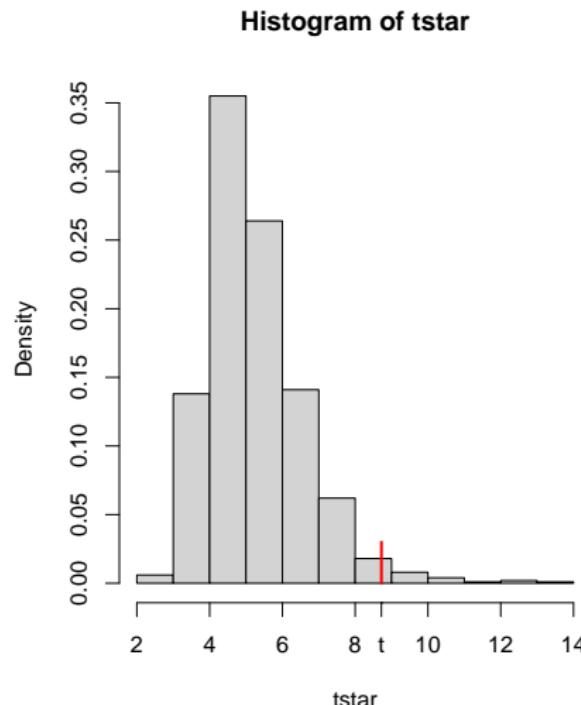
As test statistic we use  $T(X_1, \dots, X_n) = \max(X_1, \dots, X_n)$ . The  $p$ -value is computed as the proportion of  $T^*$ -values exceeding the  $T$ -value.

```
> n=length(data); t=max(data); t  
[1] 8.72055  
> B=1000; tstar=numeric(B)  
> for (i in 1:B){xstar=rexp(n,1)  
+ tstar[i]=max(xstar)}  
> pl=sum(tstar<t)/B;pr=sum(tstar>t)/B  
> p=2*min(pl,pr); p  
[1] 0.038
```

Since  $p$ -value=0.038,  $H_0$  is rejected.

The R-code for the histogram

```
> hist(tstar,prob=T,  
+ main="Histogram of tstar")  
> lines(rep(t,2),c(0,0.03),  
+ col="red",lwd=2)  
> axis(1,t,expression(paste("t")))
```



## Bootstrap test: discussion

- The resulting  $p$ -value depends on the realised  $T^*$ -values. It is recommended to repeat a bootstrap test a few times to see whether the  $p$ -value is stable.
- When  $B$  is too small, there is a lot of variation in the  $p$ -value, in that case  $B$  should be increased. In most cases  $B = 1000$  is adequate.
- A bootstrap test can be performed with any test statistic. E.g., in the example taking `min` as a test statistic yields a bootstrap  $p$ -value of about 0.19 (check this yourselves!) and does not lead to rejecting  $H_0$ .
- The **difference** between the simulation of  $T^*$ -values for bootstrap confidence intervals and bootstrap tests is in the way the  $X_1^*, \dots, X_n^*$  are generated. For confidence intervals you draw  $X_i^*$ 's from the original sample, whereas for tests you generate  $X_i^*$ 's according to  $H_0$ .

bootstrap CI  
ooooooooo

bootstrap tests  
ooooooo

t-test: one sample/paired samples  
●ooooooooo

one sample/paired samples, not normal  
ooooooooo

## t-test: one sample/paired samples

# t-test for one sample

## Setting:

the **data**  $(X_1, \dots, X_n)$  is a result of an experiment with one **numerical outcome** per experimental unit. Interest is in the **location** of the population distribution.

## Design:

- Take a random sample of experimental units from the relevant population
- Measure the outcome on each unit

**EXAMPLE** Measurement of the **height** of 4 years old children.

**EXAMPLE** Measurement of the **yearly amount of sun hours** in diff. countries.

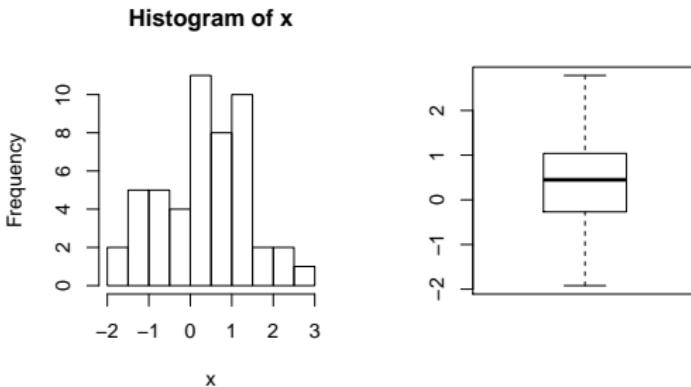
## Analysis:

- **t-test** assumes that the data  $(X_1, \dots, X_n)$  stems from a **normal distribution** (or, at least, **approximately normal**).
- **Test** about the population mean  $\mu$ :  $H_0 : \mu \left\{ \begin{array}{l} = \\ \leq \\ \geq \end{array} \right\} \mu_0$  vs.  $H_1 : \mu \left\{ \begin{array}{l} \neq \\ > \\ < \end{array} \right\} \mu_0$ .
- The **test statistic**  $T = \sqrt{n}(\bar{X} - \mu_0)/s$  has the  $t_{n-1}$ -distribution under  $H_0$ .

# One sample t-test in R

Generate data:

```
> mu=0.2
> x=rnorm(50,mu,1)
> par(mfrow=c(1,2))
> hist(x)
> boxplot(x)
```



```
> t.test(x) # by default H0: mu=0
One Sample t-test
data: x
t = 2.2701, df = 49, p-value = 0.02764.
[ some output deleted ]
```

Conclusion: reject  $H_0 : \mu = 0$ .

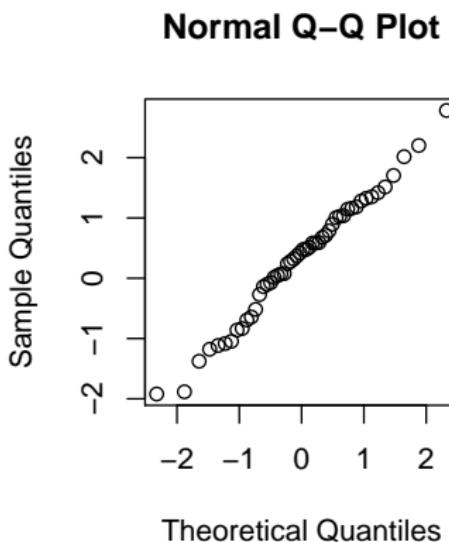
# One sample t-test in R: diagnostics

- t-test is based on the (appr.) normality assumption, need to check this.
- If the data do not follow a normal distribution, the *p*-value from the t-test **cannot be trusted**.

```
> qqnorm(x)
```

Besides `qqnorm`, one can also look at `hist`, `shapiro.test` and `boxplot`.

The main normality checks in this course are `histogram` and `qqnorm`. Sometimes, the Shapiro-Wilk normality test `shapiro.test` is also to be reported (especially when it rejects normality).



# Setting and design for two paired samples

## Setting:

An experiment with **two numerical outcomes** per experimental unit. Interest is in a possible **difference** between the two outcomes.

**EXAMPLE** Comparing **pain relief** by a dedicated drug or by a placebo. Both treatments are applied to every individual (with recovery time in between).

**EXAMPLE** Comparing two **car tire brands** by putting both brands of tire on the same car and measuring the tires' wear.

## Design:

- Take a random sample of experimental units from the relevant population.
- Measure the two outcomes on each unit (which are clearly related).
- The experiment should be set up so that any other type of “dependence” is eliminated and a difference in outcomes is due to the “treatment” only.

**Remark.** If subjects must perform two tasks, then they should be allowed sufficient time between the tasks to recover and forget. If a **learning effect** (the first measurement influences the second) is suspected, then, if possible, **randomize the order** of the two treatments within the units. The analysis must then follow the **cross over design** (studied later), not the paired samples design as discussed here.

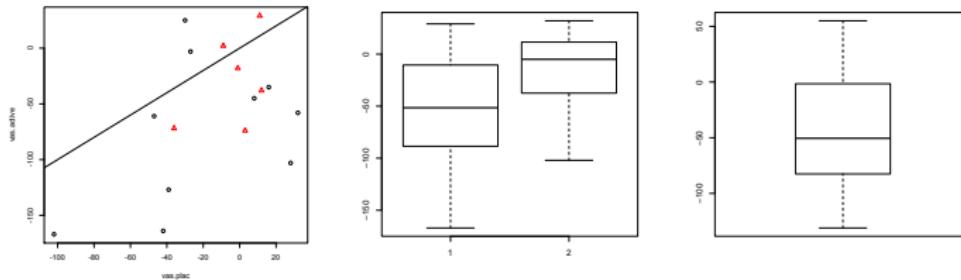
# Paired t-test: analysis

- Data  $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$ .
- In the **paired t-test** the **differences**  $Z_1 = X_1 - Y_1, \dots, Z_n = X_n - Y_n$  are assumed to be (approx.) from a **normal** distribution  $N(\mu, \sigma^2)$ .
- Test about the mean difference  $H_0 : \mu \left\{ \begin{array}{l} = \\ \leq \\ \geq \end{array} \right\} 0$  versus  $H_1 : \mu \left\{ \begin{array}{l} \neq \\ > \\ < \end{array} \right\} 0$ .
- **Test statistic**  $T = \frac{\bar{Z}}{s_Z / \sqrt{n}}$ , with  $\bar{Z} = \frac{1}{n} \sum_{i=1}^n Z_i$ ,  $s_Z^2 = \frac{1}{n-1} \sum_{i=1}^n (Z_i - \bar{Z})^2$ . Under  $H_0$ ,  $T$  has the  $t_{n-1}$ -distribution.
- The analysis is simply a **one sample analysis** on the differences, and  $\mu$  is the difference of the means of the  $X$ -population and the  $Y$ -population.

# Paired t-test in R: graphics

The rows of the data set `ashina.txt` correspond to 16 subjects and give measures of pain (for chronic headache) when treated with an active drug or a placebo.

```
> ashina=read.table("ashina.txt",header=TRUE); ashina
   vas.active vas.plac grp
1       -167      -102   1
2       -127      -39    1
[ some output deleted ]
16      -72       -36   2
> plot(vas.active~vas.plac,pch=grp,col=grp,data=ashina); abline(0,1)
> boxplot(ashina[,1],ashina[,2]); boxplot(ashina[,1]-ashina[,2])
```



The third column of the data frame `ashina` indicates the order of measurement (1=placebo first, 2=active first). This is used in the first plot (only) to determine the plotting character. A possible effect of the ordering of the measurements is ignored.

# Paired t-test in R: estimation and testing

```
> t.test(ashina[,1],ashina[,2],paired=TRUE) # two sample paired t-test
   Paired t-test
data: ashina[, 1] and ashina[, 2]
t = -3.2269, df = 15, p-value = 0.005644 # conclusion: H0 is rejected
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-71.1946 -14.5554
sample estimates:
mean of the differences
-42.875
```

Without paired=TRUE, `t.test` with 2 arguments treats 2 samples **as independent**.  
With 1 argument `t.test` performs a one sample t-test. Applied to the differences this  
is **equivalent to a paired two sample t-test**.

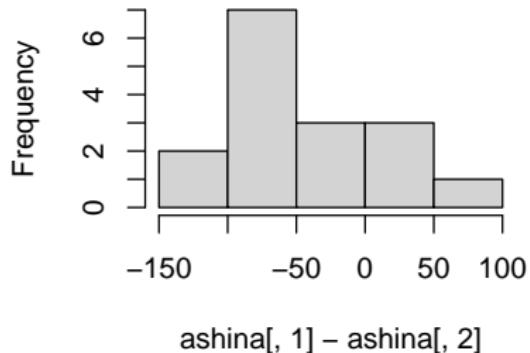
```
> t.test(ashina[,1]-ashina[,2]) # one sample t-test for differences
   One Sample t-test
data: ashina[, 1] - ashina[, 2]
t = -3.2269, df = 15, p-value = 0.005644 # conclusion: H0 is rejected
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
-71.1946 -14.5554
[ some output deleted ]
```

# Paired t-test in R: diagnostics

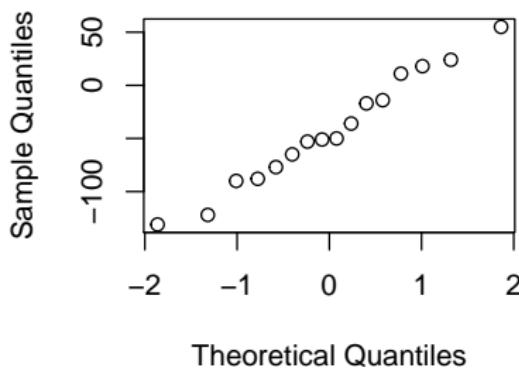
Conclusion from the above analysis:  $H_0$  is rejected, i.e., the mean of the differences is different from 0. Recall that we assumed (appr.) normality of the data. Check the normality assumption on the differences (not original samples):

```
> par(mfrow=c(1,2));hist(ashina[,1]-ashina[,2]);qqnorm(ashina[,1]-ashina[,2])
> shapiro.test(ashina[,1]-ashina[,2])  ## gives $p$-value 0.9377
```

Histogram of ashina[, 1] – ashina[,



Normal Q-Q Plot



Here no reason to suspect that the differences are not taken from a normal population.

bootstrap CI  
ooooooooo

bootstrap tests  
ooooooo

t-test: one sample/paired samples  
oooooooooo

one sample/paired samples, not normal  
●oooooooo

one sample (or two paired samples) from a nonnormal distribution

bootstrap CI  
ooooooooo

bootstrap tests  
ooooooo

t-test: one sample/paired samples  
oooooooooo

one sample/paired samples, not normal  
o●oooooooo

# One sample/paired samples: setting and design

## Setting:

- An experiment with **one numerical outcome** per experimental unit.  
Interest is in the **location** (e.g., median) of the population distribution.
- An experiment with **two numerical outcomes** per experimental unit.  
Interest is in a possible **difference between the locations** of the two outcomes. This setting is called **two paired samples** (or, **matched pairs**).

## Design:

- Take a random sample of experimental units from the relevant population.
- Measure the outcome on each unit, or measure the two outcomes on each unit (will be clearly related as they are measured on the same unit).

**EXAMPLE** The **exam grades** for a certain course.

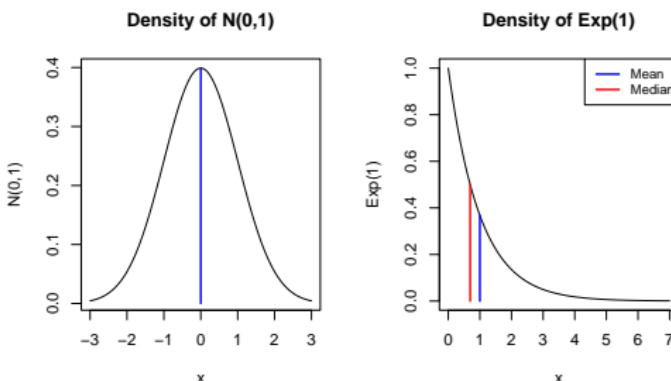
**EXAMPLE** The **blood pressure** of a person before and after a drug treatment.

# The median: recap

The **median** of a population is the middle value in the sorted populat. values.  
Formally:  $m$  is the median of a (contin.) random variable  $X$  if  $P(X \leq m) = \frac{1}{2}$ .

For median  $m$ , we have that  $P(X < m) = P(X > m) = \frac{1}{2}$ , so being bigger or smaller than the median is like **tossing a fair coin**.

For skewed distributions the mean is highly influenced by the high/low values.  
In such cases it is better to test location in terms of **median** instead of **mean**.



The more skewed, the bigger the distance between median and mean.

# Sign test for one sample or matched pairs

## Setting:

- A sample  $X_1, \dots, X_n$  from some population. We want to test about the population **median  $m$** .
- A sample  $(Z_1, Y_1), \dots, (Z_n, Y_n)$  of matched pairs from some population. We want to test about the **median  $m$  of the differences  $X_i = Z_i - Y_i$** .

**Hypotheses:** we test  $H_0 : m \left\{ \begin{array}{l} \leq \\ \geq \end{array} \right\} m_0$  versus  $H_1 : m \left\{ \begin{array}{l} \neq \\ < \end{array} \right\} m_0$ .

**Test statistic:**  $T = \#(i : X_i > m_0)$  ("#" means "the number of".)

**Distribution of  $T$  under  $H_0$ :** exactly  $\text{Bin}(n, \frac{1}{2})$  (a norm. approx. is possible).

Depending on  $H_1$  the test is one-sided or two-sided.

**In R:** `binom.test(t,n,p=0.5,alt=...)` (e.g., `alt="g"` if  $H_1 : m > m_0$ )

If  $m = m_0$ , about  $\frac{n}{2}$  values are expected to be bigger/smaller than  $m_0$ . Large deviations from this indicate that  $H_0$  may not be true.

One needs to formulate the right alternative in the binomial: for example, if  $H_1 : m > m_0$ , put `alt="g"` in `binom.test`.

## Sign test in R: example

We want to test whether the median exam grade is 6. Because of the small sample size, we are not sure about normality. (Grades are not always normally distributed!) Data are the exam grades of 13 randomly selected students.

```
> examresults=c(3.7,5.2,6.9,7.2,6.4,9.3,4.3,8.4,6.5,8.1,7.3,6.1,5.8)
> sum(examresults>6)
[1] 9
> binom.test(9,13,p=0.5)    # exact binomial test
[ some output is deleted ]
p-value = 0.2668
```

Conclusion from the above output of `binom.test`:  $H_0$  is not rejected.

To test the claim of interest correctly, one should reduce to the right version of the binomial test: the relevant one-sided or two sided version. For example, to test whether the exam is not too difficult, we can set  $H_1 : m > m_0 = 6$  leading to test `binom.test(t,n,p=0.5,alt="g")`. Other choices of statistics  $T$ : e.g., for  $T = \#\{i : X_i \leq m_0\}$ , testing  $H_1 : m > m_0$  leads to `binom.test(t,n,p=0.5,alt="l")`.

# Wilcoxon signed rank test for one sample or matched pairs

## Setting:

- A sample  $X_1, \dots, X_n$  from a **symmetric** population (a stronger assumption than for the sign test!). Want to test about the population **median**  $m$ .
- A sample  $(Z_1, Y_1), \dots, (Z_n, Y_n)$  of matched pairs from some population. Test about the **median  $m$  of the (symm.) differences**  $X_i = Z_i - Y_i$ .

Hypotheses:  $H_0 : m \left\{ \begin{array}{l} \leq \\ \geq \end{array} \right\} m_0$  versus  $H_1 : m \left\{ \begin{array}{l} \neq \\ > \\ < \end{array} \right\} m_0$ .

Test statistic:  $T = \sum_{i: X_i > m_0} R_i$  of the ranks of  $|X_i - m_0|$  over such  $i$  for which  $X_i > m_0$ . Large values of  $T$  indicate that  $m > m_0$ , small  $T$  that  $m < m_0$ .

Distribution of  $T$  under  $H_0$ : known in R (normal approximation for large  $n$ ).

In R: `wilcox.test(data, mu=m0, alt=...)` Dep. on  $H_1$ , one- or two-sided test.

Rank of an observation is the order number assigned to it if the observations are ordered from smallest to largest. For example, the ranks of observations  $X_1 = 3$ ,  $X_2 = 5$ ,  $X_3 = 2$ ,  $X_4 = 7$  are  $R_1 = 2$ ,  $R_2 = 3$ ,  $R_3 = 1$ ,  $R_4 = 4$  resp. In R, the ranks of the sample  $x$  are computed by `rank(x)`. Norm. approx.:  $\frac{T - n(n+1)/4}{\sqrt{n(n+1)(2n+1)/24}} \sim N(0, 1)$ .

bootstrap CI  
ooooooooo

bootstrap tests  
ooooooo

t-test: one sample/paired samples  
oooooooooo

one sample/paired samples, not normal  
oooooooo●○

## Wilcoxon signed rank test in R: example

The [Wilcoxon signed rank test](#) takes into account the **ranks** of the deviations from the proposed median  $m_0$ . If the data is symmetric around  $m_0$ , the ranks at both sides should be approximately equal.

```
> sum(rank(abs(examresults-6))[examresults-6>0]) # value test statistics
[1] 64
> wilcox.test(examresults,mu=6)

Wilcoxon signed rank test

data: examresults
V = 64, p-value = 0.2163
alternative hypothesis: true location is not equal to 6
```

**Conclusion:**  $H_0$  is not rejected.

# To finish

Today we discussed:

- ① bootstrap confidence intervals
- ② bootstrap tests
- ③ one sample (two paired samples) tests for normal and not normal samples
  - t-test
  - sign test
  - Wilcoxon signed rank test

Next time: two sample tests.

# Experimental Design and Data Analysis, Lecture 3

Eduard Belitser

VU Amsterdam

# Lecture overview

## ① two paired samples (normal and not normal)

- permutation test
- dependence in two paired samples
  - Pearson's correlation test
  - Spearman's rank correlation test

## ② two independent samples (normal and not normal)

- two samples  $t$ -test
- Mann-Whitney test
- Kolmogorov-Smirnov test

permutation test for two paired samples

●oooooooo

Dependence in two paired samples

oooooooooo

two independent samples

oooooooooooooooooooo

permutation tests for two paired samples

## Reminder: setting and design for two paired samples

### Setting:

- An experiment with a numerical outcome measured according to **two conditions** per experimental unit (hence **two paired samples**);
- Interest is in a possible **difference** between the two outcomes per unit.

**EXAMPLE** Difference in **average course grade** for **mathematical courses** and **informatics courses** for BA-students at the VU.

**EXAMPLE** Difference in **pain relief** by an **active drug** and a **placebo** for patients.

### Design (the standard paired samples design):

- Take a random sample of experimental units from the relevant population.
- Measure the two outcomes on each unit.

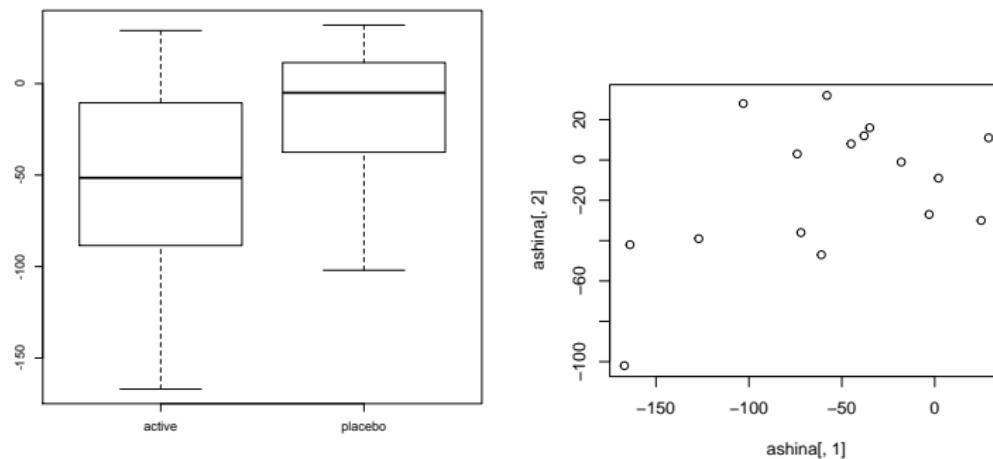
# Idea of permutation technique

- Data  $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$  must be **two paired samples**.
- In a permutation test we do **not assume normality**.
- We use **any test statistic**  $T = T(X_1, Y_1, \dots, X_n, Y_n)$  to test  $H_0$ : no difference between the distributions of  $X_j$ 's and that of  $Y_j$ 's. The choice of test statistics should **express somehow the difference conjectured**.
- Like in a bootstrap test, we simulate the distribution of  $T$  under  $H_0$ , using  $B$  surrogate  $T^*$ -values. Repeat  $B$  times (for  $i = 1, \dots, B$ ):
  - generate each  $(X_j^*, Y_j^*)$ ,  $j = 1, \dots, n$ , by applying a random **permutation** of the original  $(X_j, Y_j)$ , i.e., choose between  $(X_j, Y_j)$  and  $(Y_j, X_j)$  with equal probability;
  - next, compute  $T_i^* = T(X_1^*, Y_1^*, \dots, X_n^*, Y_n^*)$ .
- Under  $H_0$  of no difference between the distributions of  $X$  and  $Y$  within pairs permuting the labels should not change the distribution of  $T$ .

# Permutation test in R: data input and graphics

Recall dataset ashina.txt (headache after drug or placebo for 16 subjects).

```
> ashina=read.table("ashina.txt",header=TRUE)
> boxplot(ashina[,1],ashina[,2],names=c("active","placebo"))
> plot(ashina[,1],ashina[,2])
```



Based on the boxplots, we expect the active medicine to yield better pain relief.

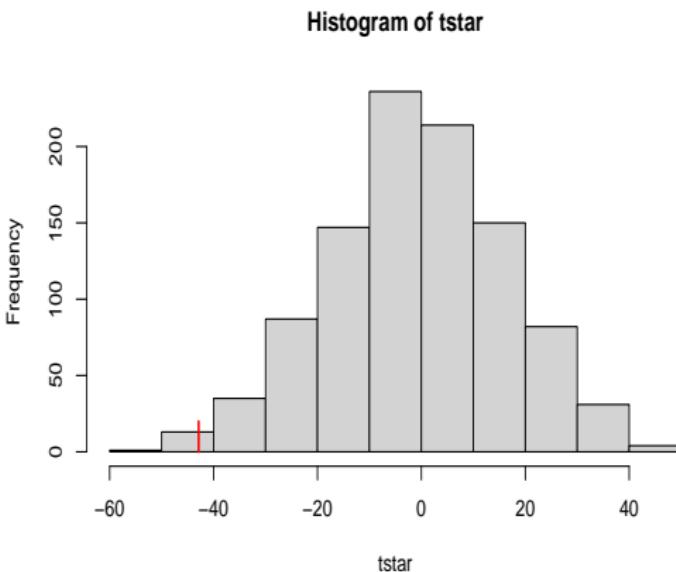
# Permutation test in R: testing (1)

```
> mystat=function(x,y) {mean(x-y)}  
> B=1000; tstar=numeric(B)  
> for (i in 1:B) {  
+   ashinastar=t(apply(cbind(ashina[,1],ashina[,2]),1,sample))  
+   tstar[i]=mystat(ashinastar[,1],ashinastar[,2]) }  
> myt=mystat(ashina[,1],ashina[,2])
```

Instead of computing all  $2^{16} = 65536$  possible permutations, we generate 1000 randomly chosen permutations to estimate the distribution of our test statistic under  $H_0$ . The function `apply` applies a function to either all rows or all columns in a matrix (parameter 1 indicates rows), `t(matrix)` means transposition of `matrix`.

## Permutation test in R: testing (2)

```
> myt  
[1] -42.875  
> hist(tstar)  
> lines(rep(myt,2),c(0,20),  
+ col="red",lwd=2)  
> pl=sum(tstar<myt)/B  
> pr=sum(tstar>myt)/B  
> p=2*min(pl,pr); p  
[1] 0.008
```



**Conclusion:** there is indeed a significant difference between the active drug and the placebo.

## Permutation test: discussion

- A permutation test for two paired samples can be performed with **any test statistic** that expresses difference between the  $X$  and  $Y$  within pairs.  
(The mean of differences  $Z_i = X_i - Y_i$  is most common to consider, but one may as well consider the median of the  $Z_i$ 's.)
- Alternatives to the permutation test for two paired samples are the sign test and the Wilcoxon signed rank test applied to the differences.

permutation test for two paired samples  
oooooooo

Dependence in two paired samples  
●oooooooo

two independent samples  
oooooooooooo

## Dependence in two paired samples

# Dependence between two paired samples

## Setting:

An experiment with two **numerical outcomes** (say  $X$  and  $Y$ ) per experimental unit. Interest is in a possible **dependence** between the two outcomes per unit.

**EXAMPLE** Relation between **shoe size** and **body mass index** of a person.

**EXAMPLE** Relation between **average course grade** and **number of students taking the course** for courses at the VU.

**EXAMPLE** Relation between amount of **precipitation** and **sun hours** for different cities in Europe.

## Design:

- Take a random sample of experimental units from the relevant population.
- Measure the two quantities on each unit. (The two outcomes are in principle related, because measured on the same experimental unit.)
- However, we possibly have measured unrelated quantities of the units and we want to test whether these quantities are **correlated**.

# Pearson's correlation test

- Data  $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$ .
- The Pearson correlation test assumes normality of the both  $X_i$ 's and  $Y_i$ 's.  
(Rather, the asympt. normality of the sample correlation  $\hat{\rho}$ .)
- The test is based on the sample correlation coefficient (which estimates the “true” correlation  $\rho = Cor(X, Y)$ ):

$$\hat{\rho} = \hat{\rho}_{X,Y} = \frac{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)(Y_i - \bar{Y}_n)}{\sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2} \sqrt{\frac{1}{n-1} \sum_{i=1}^n (Y_i - \bar{Y}_n)^2}}.$$

- We test the null hypothesis  $H_0 : \rho = \rho_0 = 0$  that the correlation between the two populations is  $\rho_0 = 0$ . The test statistic is given by

$$T_\rho = \frac{\hat{\rho} - \rho_0}{\sqrt{(1 - \hat{\rho}^2)/(n - 2)}} = \frac{\hat{\rho}}{\sqrt{(1 - \hat{\rho}^2)/(n - 2)}},$$

which has under  $H_0 : \rho = 0$  a t-distribution with  $n - 2$  degrees of freedom.

# Spearman's rank correlation test

- Data  $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$ .
- Spearman's rank correlation test does not assume normality. The test considers the ranks  $R(X_i)$  and  $R(Y_i)$  in the two samples, and compares the ordering of the ranks in the  $X_i$  and the  $Y_i$ .
- If the data are rank correlated, these sequences of ranks will run (approximately) in parallel or in opposite order.
- The test statistic is the sample correlation  $\tilde{\rho}$  between the the rank vectors.
- We test the null hypothesis  $H_0 : \rho_s = 0$ . (Correlation of the rank variables.)

If all  $n$  ranks are distinct integers, the test statistic can be computed as

$$\tilde{\rho} = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2-1)}$$
 where  $d_i = R(X_i) - R(Y_i)$  is the difference between the two ranks of observations  $X_i$  and  $Y_i$ .

This test is useful in testing whether variable  $Y$  is a monotone transformation of variable  $X$  (or vice versa) in which case the true rank correlation is  $\rho_s = 1$ .

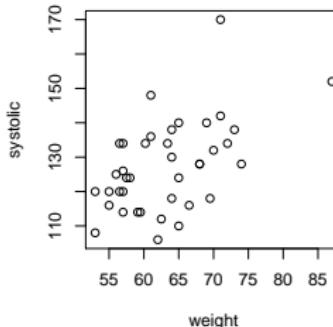
## Correlation tests in R: example

Consider the data frame `peruvians.txt`, where the rows correspond to 39 Peruvian men that moved from a native culture to a modern society. Amongst others, years since migration, systolic and diastolic blood pressure, heart rate (column `wrist`), weight, length were measured.

```
> peruvians=read.table("peruvians.txt",header=TRUE); peruvians
   age migration weight length chin arm calf wrist systolic diastolic
1   21           1    71.0   1629   8.0  7.0 12.7    88      170       76
2   22           6    56.5   1569   3.3  5.0  8.0    64      120       60
[ some output deleted ]
39  54          40    87.0   1542  11.3 11.7 11.3    92      152       88
```

```
> attach(peruvians)
> plot(systolic~weight)
```

Based on this picture, we expect dependence between systolic and weight.



## Pearson's test in R: example

```
> cor.test(systolic,weight)
```

Pearson's product-moment correlation

data: systolic and weight

t = 3.7164, df = 37, p-value = 0.0006654

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

0.2463759 0.7186619

sample estimates:

cor

0.5213643

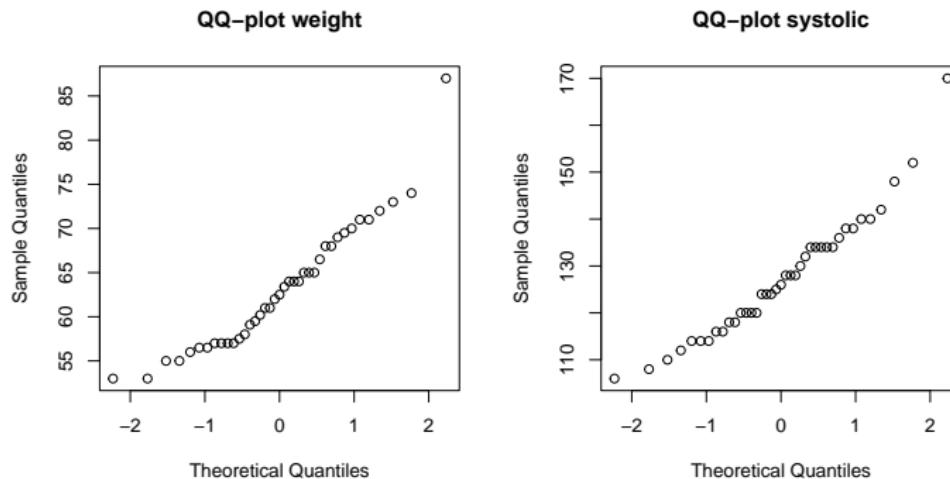
**Conclusion:** there is significant correlation, if normality is assumed.

The default for cor.test is Pearson's correlation test, based on normality.

# Correlation tests: diagnostics

Check the normality assumption on the two samples:

```
> par(mfrow=c(1,2)); qqnorm(peruvians$weight,main="QQ-plot weight")
> qqnorm(peruvians$systolic,main="QQ-plot systolic")
```



QQ-plots show that normality is doubtful for the weight sample. Hence, we use the rank correlation test of Spearman (and **not** Pearson's correlation test).

## Spearman's test in R: example

```
> cor.test(systolic,weight,method="spearman")

  Spearman's rank correlation rho

data: systolic and weight
S = 5322.352, p-value = 0.003119
alternative hypothesis: true rho is not equal to 0
sample estimates:
rho
0.4613004

Warning message:
In cor.test.default(systolic, weight, method = "spearman") :
  Cannot compute exact p-values with ties
```

**Conclusion:** there is indeed significant rank correlation.

There is a warning about ties, which means that some values occur multiple times in weight and/or systolic. Therefore *R* uses an approximation for the *p*-value.

permutation test for two paired samples  
oooooooo

Dependence in two paired samples  
ooooooooo

two independent samples  
●oooooooooooo

two independent samples

## Two independent samples: setting and design

Setting: an experiment with

- one numerical outcome per experimental unit,
- two independent groups of experimental units.

Interest is in a possible difference between the two populations. medskip

**EXAMPLE** Comparing the weight of newborn children in two countries.

**EXAMPLE** Total yield from an agricultural plot for two different fertilizers.

Design:

- Take a random sample of experimental units of size  $M$  from the first population and a random sample of size  $N$  from the second population;
- Measure the outcome on each unit.

The numbers  $M$  and  $N$  need not be the same.

# t-test for two independent samples

Recall the t-test for two independent samples.

- Data  $(X_1, \dots, X_M)$  and  $(Y_1, \dots, Y_N)$ .
- The **two samples t-test** assumes that both samples  $X_1, \dots, X_M$  and  $Y_1, \dots, Y_N$  come from independent **normal** populations. Denote the mean of the first population by  $\mu$  and the mean of the second by  $\nu$ .
- We **test** about the relation between the population means  $\mu$  and  $\nu$ :

$$H_0 : \mu \left\{ \begin{array}{l} = \\ \leq \\ \geq \end{array} \right\} \nu \quad \text{versus} \quad H_1 : \mu \left\{ \begin{array}{l} \neq \\ > \\ < \end{array} \right\} \nu.$$

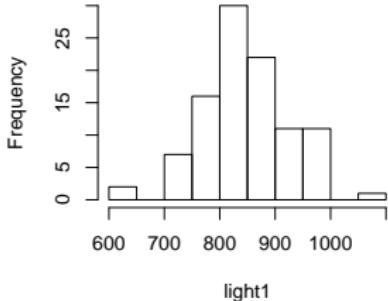
- The **test statistic**  $T = \frac{\bar{X}_M - \bar{Y}_N}{S_{N,M}} \sim t_{N+M-2}$  under  $H_0$ .

# t-test in R: data input and graphics

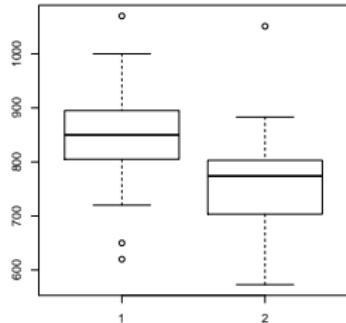
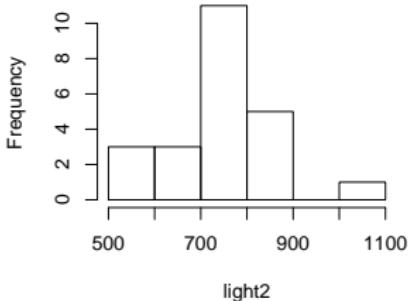
Consider two data sets of measurements of the speed of light (minus 299000) by Michelson in 1879 and in 1882.

```
>light1=scan("light1879.txt"); light2=scan("light1882.txt")
> hist(light1); hist(light2); boxplot(light1,light2)
```

Histogram of light1



Histogram of light2



# t-test in R: estimation and testing

The two samples *t*-test:

```
> t.test(light1,light2)
```

```
Welch Two Sample t-test
```

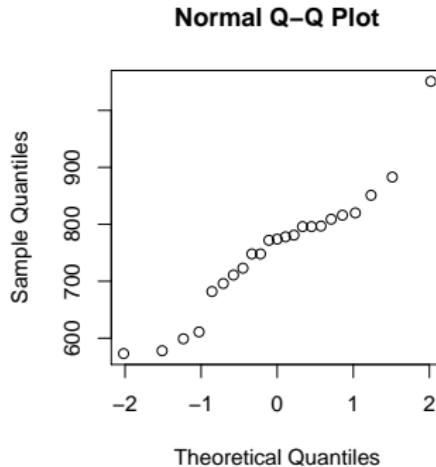
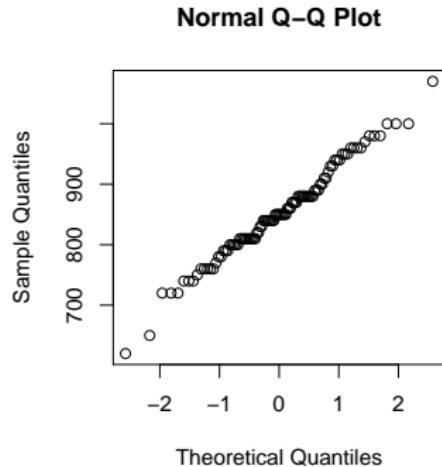
```
data: light1 and light2
t = 4.0598, df = 27.754, p-value = 0.0003625
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 47.63387 144.73135
sample estimates:
mean of x mean of y
 852.4000 756.2174
```

**Conclusion:**  $H_0$  of equal means is rejected.

By default `t.test` with two arguments performs the two samples *t*-test for independent samples.

# t-test in R: diagnostics

```
> qqnorm(light1)  
> qqnorm(light2)
```



Normality of the second sample is actually doubtful.

# Mann-Whitney test

- Data: two **independent** samples  $(X_1, \dots, X_M)$  and  $(Y_1, \dots, Y_N)$ .
- The **Mann-Whitney test** assumes that the sample  $X_1, \dots, X_M$  stems from population  $F$  and sample  $Y_1, \dots, Y_N$  stems from population  $G$ .
- We **test** the null hypothesis  $H_0 : F = G$  (the distributions are the same).
- The Mann-Whitney test is again based on ranks. It considers the  $M$  ranks  $R_1, \dots, R_M$  of  $X_1, \dots, X_M$  in the combined sample  $(X_1, \dots, X_M, Y_1, \dots, Y_N)$  of length  $M + N$ . If  $F = G$  these  $M$  rank numbers should lie randomly between 1 and  $M + N$ . The test statistic is

$$T = \sum_{i=1}^M R_i, \quad \text{the distribution of } T \text{ under } H_0 \text{ is (approximately) known.}$$

- Large values of  $T$  indicate that  $F$  is shifted towards the right from  $G$ , i.e. that  $X$ -values are bigger than  $Y$ -values.

If responses are continuous, a significant result of Mann-Whitney test shows a difference in medians, actually this test is only consistent against the alternative  $H_1 : P(X > Y) \neq P(Y > X)$ .

# Mann-Whitney test in R: testing

```
> wilcox.test(light1,light2)

Wilcoxon rank sum test with continuity correction

data: light1 and light2
W = 1829, p-value = 1.056e-05
alternative hypothesis: true location shift is not equal to 0
```

**Conclusion:**  $H_0$  of equal medians is rejected. The underlying distribution of light1 is shifted to the right from that of light2.

When given two arguments `wilcox.test` will perform the Mann-Whitney test for two samples. The Mann-Whitney test is especially suited for detecting shift differences — differences in location — between two populations.

One-sided alternatives are also possible to test by the Mann-Whitney test. For example, to test whether the distribution of light1 is on the bigger values than the distribution of light2, we use `wilcox.test(light1,light2,alt="g")`.

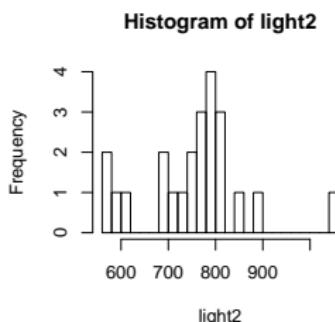
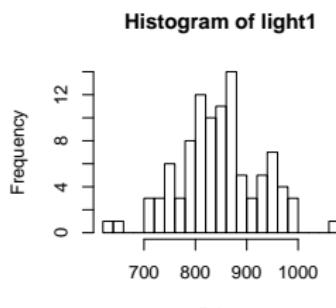
# Kolmogorov-Smirnov test

- Data: two **independent** samples  $(X_1, \dots, X_M)$  and  $(Y_1, \dots, Y_N)$ .
- The **Kolmogorov-Smirnov test** assumes that the sample  $X_1, \dots, X_M$  stems from distribution  $F_X$  and sample  $Y_1, \dots, Y_N$  stems from distribution  $F_Y$ .
- We test the null hypothesis  $H_0 : F_X = F_Y$  (the distributions are the same).
- The Kolmogorov-Smirnov test is based on the maximal difference of the two empirical distribution functions for two samples.
- The **test statistic** computes the maximal vertical difference in **empirical distribution functions** (summed histograms). Its **distribution** under  $H_0$  is known (e.g., in R).

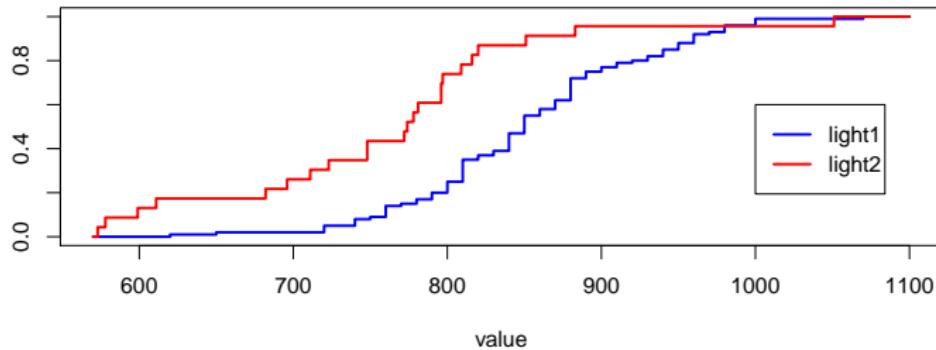
The empirical distribution function for a sample  $Z_1, \dots, Z_n$  is defined as  $\hat{F}_n(x) = \frac{\#\{i: Z_i \leq x\}}{n}$  for all  $x \in \mathbb{R}$ . This is a non-decreasing from 0 to 1 step function making jumps of size  $\frac{1}{n}$  in points  $Z_{(1)}, \dots, Z_{(n)}$ .

# Kolmogorov-Smirnov test in R: graphics

```
> hist(light1)  
> hist(light2)
```



summed histogram



# Testing in R by the Kolmogorov-Smirnov test

```
> ks.test(light1,light2)
```

Two-sample Kolmogorov-Smirnov test

```
data: light1 and light2
D = 0.5391, p-value = 3.803e-05
alternative hypothesis: two-sided
```

Warning message:

```
In ks.test(light1, light2) : cannot compute exact p-values with ties
```

A warning about ties again: R uses an approximation for computing the *p*-value.

**Conclusion:**  $H_0 : F_X = F_Y$  is rejected (in fact, `light1` is larger than `light2`).

**One-sided alternatives** are also possible to test by the Kolmogorov-Smirnov test. But take into account the **counterintuitive interpretation**: `ks.test(x,y,alt="g")` tests the alternative  $F_X(x) \geq F_Y(x)$  which means that  $F_Y$  is shifted towards the right from  $F_X$ , i.e., that **Y-values are bigger than X-values**. For example, we suspect that `light1` is larger than `light2`. To test this, `ks.test(light1,light2,alternative="less")` (or `ks.test(light2,light1,alternative="greater")`).

# Goodness-of-fit by the Kolmogorov-Smirnov test

- Data: a sample from a (unknown) distribution:  $(X_1, \dots, X_n) \sim F_X$ .
- For some (known)  $F_0$ , test  $(X_1, \dots, X_n) \sim F_0$ , i.e.,  $H_0 : F_X = F_0$ .
- The KS-test is again based on the test statistic that computes the maximal vertical difference between the empirical distrib. function of the sample and the distrib. function  $F_0$ . Its distribution under  $H_0$  is known.

Verify by the KS-test whether the sample `light1` comes from  $N(850, 10000)$ .

```
> ks.test(light1, pnorm, 850, 100) # specify the parameters of F0
```

Asymptotic one-sample Kolmogorov-Smirnov test

```
data: light1
D = 0.10854, p-value = 0.1894
alternative hypothesis: two-sided
```

Warning message:

```
In ks.test.default(light1, pnorm, 850, 100) :
ties should not be present for the Kolmogorov-Smirnov test
```

Do not reject  $H_0$ .

permutation test for two paired samples  
oooooooo

Dependence in two paired samples  
ooooooooo

two independent samples  
oooooooooooo●

## To finish

Today we discussed: two samples tests (including permutation test); for paired and independent samples, for normal and not normal cases.

Next time:  $k$  samples, one way ANOVA.

# Experimental Design and Data Analysis, Lecture 4

Eduard Belitser

VU Amsterdam

# Lecture overview

- ① Analysis of Variance (one-way ANOVA)
- ② Kruskal-Wallis test
- ③ permutation tests in the setting of one-way ANOVA

1-way ANOVA

●oooooooooooooo

Kruskal-Wallis

ooooooo

permutation test in 1-way ANOVA

ooooooo

## one way ANOVA (analysis of variance) completely randomized design

# Setting

An experiment with:

- a **numerical outcome**  $Y$ ;
- a **factor** that can be fixed at  $I$  **levels** ("treatment").

If  $I = 2$ , this is just the two-sample problem, and we could perform a t-test.

**EXAMPLE** Agricultural experiment with outcome **total yield** from a plot and treatment **type of fertilizer**.

**EXAMPLE** Quality of a genetic algorithm to determine the minimal value of a criterion function with outcome **CPU time needed to find true minimum** and treatment **mutation probability** set to 0.01, 0.02, 0.03, 0.04 or 0.05.

**EXAMPLE** Outcome **time to develop mold** on bread and treatment **temperature of the environment** fixed to 15, 19 or 22 degrees (garage, bedroom, living room).

# Design

- Select  $NI$  experimental units randomly from the population of interest.
- Assign level  $i$  of the factor to a random set of  $N$  units ( $i = 1, 2, \dots, I$ ).
- Perform the experiment  $NI$  times, independently.

Randomization in R.

```
> I=4; N=5
> rep(1:I,N)
[1] 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4
> sample(rep(1:I,N))
[1] 3 4 2 1 1 4 3 4 3 1 3 2 3 2 1 4 2 4 2 1
```

Use level 3 for unit 1, level 4 for unit 2, etc.

Using an equal number of units  $N$  for each level (called **balanced design**) is preferable, but not necessary.

# One-way ANOVA

## Data

sample 1:  $Y_{11}, Y_{12}, \dots, Y_{1N}$

sample 2:  $Y_{21}, Y_{22}, \dots, Y_{2N}$

⋮

sample  $I$ :  $Y_{I1}, Y_{I2}, \dots, Y_{IN}$ .

Assume that these samples are obtained independently from  $I$  normal populations with (possibly different) population means  $\mu_1, \mu_2, \dots, \mu_I$ , and with equal variances.

We want to test the null hypothesis  $H_A : \mu_1 = \mu_2 = \dots = \mu_I$  versus the alternative  $H_1 : \mu_i \neq \mu_j$  for some  $(i, j)$ .

The test statistic is a bit complicated, see below. It is, together with its distribution under  $H_A$ , implemented in R.

# One-way ANOVA model

A categorical explanatory variable (also called **factor**) with  $I$  different categories/levels corresponds to  $I$  groups/populations/levels.

The **one-way ANOVA** model is: with  $\mu_i = \mu + \alpha_i$ ,

$$Y_{ik} = \mu_i + e_{ik} = \mu + \alpha_i + e_{ik}, \quad i = 1, \dots, I, \quad k = 1, \dots, n_i,$$

- $Y_{ik}$  is the  $k$ -th response measured in group  $i$ ,
- $\mu$  is the common mean,  $\alpha_i$  is the contribution of level  $i$ ,  $i = 1, \dots, I$ ,

**Assumption:** the indep. errors  $e_{ik} \sim N(0, \sigma^2)$ , with unknown variance  $\sigma^2$ .

**Balanced design:** the same number of observations per group  $n_i = N$ ,  
 $i = 1, \dots, I$ , so that the total number of observations is  $n = \sum_{i=1}^I n_i = NI$ .

**Note:** if  $I = 2$ , this is the setting for the two sample  $t$ -test with equal variances.

Parameters  $\mu, \alpha_1, \dots, \alpha_I$  are not uniquely defined, one needs to specify one linear restriction on the parameters. Default parametrization (treatment parametrization) in R is  $\alpha_1 = 0$ , meaning that group 1 is the reference class. Other common parametrizations are  $\mu = 0$  (then  $\mu_i = \alpha_i$ ) or  $\sum_{i=1}^I \alpha_i = 0$ . The parametrization in R can be set by the command `contrasts`.

The one-way ANOVA can be written in the matrix form  $Y = X\beta + e$  for appropriate **design matrix**  $X$  and parameter vector  $\beta$ , depending on a chosen parametrization.

# One-way ANOVA test

**Setting:** a one-way ANOVA model:  $Y_{ij} = \mu + \alpha_i + e_{ij}$ .

**Hypotheses:**  $H_A : \alpha_1 = \dots = \alpha_k = 0$  (no factor effect) versus  $H_1 : \text{at least one } \alpha_i \neq 0$  (factor effect is present).

**Test statistic:** with  $\bar{Y}_{i\cdot} = \frac{1}{n_i} \sum_{k=1}^{n_i} Y_{ik}$  and  $\bar{Y}_{..} = \frac{1}{I} \sum_{i=1}^I \frac{1}{n_i} \sum_{k=1}^{n_i} Y_{ik}$ , under  $H_0$ ,

$$F = \frac{\text{between-groups SS}}{\text{within-groups SS}} = \frac{\sum_{i=1}^I n_i (\bar{Y}_{i\cdot} - \bar{Y}_{..})^2 / (I-1)}{\sum_{i=1}^I \sum_{j=1}^{n_i} (Y_{ij} - \bar{Y}_{i\cdot})^2 / (n-I)} \sim F_{I-1, n-I},$$

the **F-distribution** with  $I-1$  and  $n-I$  degrees of freedom.

Larger values of  $F = f$  give **more evidence against  $H_0$**  in favor of  $H_1$ , hence we only reject  $H_A$  if  $F$  is large. The test is therefore **always right-sided**: compare the  $p$ -value  $p_{right} = P(F > f)$  with a significance level  $\alpha$ .

**In R:** the  $p$ -value is in `anova(lm(y~f), data=...)`,  $f$  is the factor.

**In R:** `summary(lm(y~f, data=...))` shows the coefficient estimates  $\hat{\alpha}_i$ 's in the treatment parameterization, to get these in the sum parametrization use (before `lm` command) `contrasts(f)=contr.sum`.

# One-way ANOVA table

One-way ANOVA results are usually presented in an one-way [ANOVA table](#):

Source	Df	Sum Sq	Mean Sq	F value	p-value
Factor A	$I - 1$	$SS_A$	$SS_A/(I - 1)$	$f = \frac{SS_A/(I-1)}{RSS/(n-I)}$	$P(F > f)$
Residuals	$n - I$	$RSS$	$RSS/(n - I)$		
Total	$n - 1$	$SS_T$			

In R it looks as follows:

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Factor	--	-----	-----	-----	-----
Residuals	--	-----	-----		

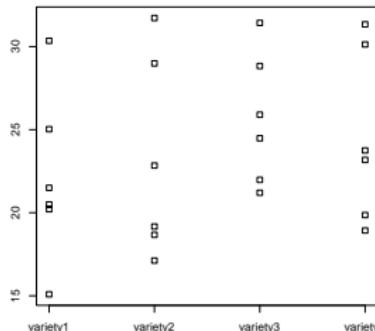
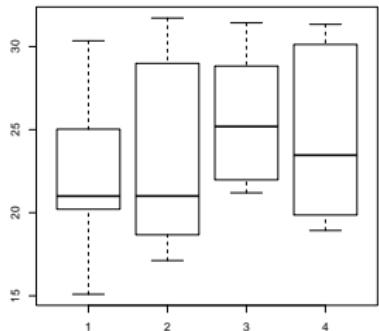
Here  $RSS = \sum_{i=1}^I \sum_{j=1}^{n_i} (Y_{ij} - \bar{Y}_{i\cdot})^2$ ,  $SS_A = \sum_{i=1}^I n_i(\bar{Y}_{i\cdot} - \bar{Y}_{..})^2$ ,  
 $SS_T = RSS + SS_A$ .

The denominator  $\frac{RSS}{n-I}$  of the test statistics  $F$  is an unbiased estimator of  $\sigma^2$ :

$$\hat{\sigma}^2 = S^2 = \frac{RSS}{n-I}.$$

# One-way ANOVA in R: graphics

```
> melon=read.table("melon.txt",header=TRUE)
> melon
  variety1 variety2 variety3 variety4
1    15.09    17.12    21.20    18.93
2    20.21    19.17    28.83    31.34
3    30.35    28.99    31.43    30.13
4    25.03    22.84    25.90    23.18
5    20.50    31.72    21.98    19.86
6    21.50    18.67    24.48    23.75
> boxplot(melon); stripchart(melon,vertical=TRUE)
```



# One-way ANOVA in R: data input

If needed, create a data frame with a numeric column of responses  $Y_{in}$  and a second factor column of the corresponding factor levels.

```
> melon
  variety1 variety2 variety3 variety4
1    15.09    17.12    21.20    18.93
2    20.21    19.17    28.83    31.34
3    30.35    28.99    31.43    30.13
4    25.03    22.84    25.90    23.18
5    20.50    31.72    21.98    19.86
6    21.50    18.67    24.48    23.75
> melonframe=data.frame(yield=as.vector(as.matrix(melon)),
+ variety=factor(rep(1:4,each=6))) #create a data frame in the right format
> melonframe[1:5,]
  yield variety
1 15.09      1
2 20.21      1
3 30.35      1
4 25.03      1
5 20.50      1
> is.factor(melonframe$variety); is.numeric(melonframe$variety)
[1] TRUE
[1] FALSE
```

# One-way ANOVA in R: testing

```
> melonaov=lm(yield~variety,data=melonframe)
> anova(melonaov)
```

## Analysis of Variance Table

Response: yield

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
variety	3	43.55	14.516	0.5543	0.6512
Residuals	20	523.73	26.186		

The command `lm` creates an object of type linear model (many things can be extracted from it by using other functions), `yield~variety` is a **model formula**. Read it as: “explain yield using variety”. The *p*-value for  $H_A : \alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0$  (which is the same as  $H_A : \mu_1 = \mu_2 = \mu_3 = \mu_4$ ) is 0.6512, hence  $H_A$  is not rejected, i.e., factor variety is not significant.

The best estimates  $\hat{\mu}_i$  for  $\mu_i$ ,  $i = 1, \dots, I$ , are the means over  $(Y_{i1}, \dots, Y_{in_i})$ :

$\hat{\mu}_i = \bar{Y}_{i\cdot} = \frac{1}{n_i} \sum_{l=1}^{n_i} Y_{il}$ . But what about estimates of  $\mu, \alpha_1, \dots, \alpha_I$ ? **Identifiability problem**: we started with  $I$  parameters  $\mu_1, \dots, \mu_I$  and now have  $I + 1$  new parameters

$\mu, \alpha_1, \dots, \alpha_I$ . To tackle this, we use **extra linear restriction(s)** (in R: **parametrization**).

# One-way ANOVA in R: estimation (1)

By default R uses **treatment parametrization**, i.e.,  $\alpha_1 = 0$ . In this case, R reports the estimates of  $\mu_1 = \mu + \alpha_1 = \mu$ ,  $\alpha_2 = \mu_2 - \mu_1$ ,  $\dots$ ,  $\alpha_4 = \mu_4 - \mu_1$ .

```
> summary(melonaov)
[ some output deleted ]
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	22.1133	2.0891	10.585	1.21e-09 ***
variety2	0.9717	2.9545	0.329	0.746
variety3	3.5233	2.9545	1.193	0.247
variety4	2.4183	2.9545	0.819	0.423

In the **treatment contrasts**, R takes the first level (here variety1, in alphabetical order) as a **base level** and compares the other levels to it. The estimates are  $\hat{\mu}_1 = 22.1133$ ,  $\hat{\alpha}_2 = 0.9717$ ,  $\hat{\alpha}_3 = 3.5233$ ,  $\hat{\alpha}_4 = 2.4183$ . The group means are found as  $\hat{\mu}_i = \hat{\mu} + \hat{\alpha}_i$ ,  $i = 1, \dots, 4$  (remember  $\hat{\alpha}_1 = 0$ ), they can also be obtained by command **fitted(melonaov)**.

Command **summary(model)** also provides the output for testing **individual**  $H_0 : \mu_1 = 0$ ,  $H_0 : \alpha_i = \mu_i - \mu_1 = 0$ ,  $i = 2, 3, 4$  (basically  $H_0 : \mu_i = \mu_1$  vs.  $H_1 : \mu_i \neq \mu_1$ ,  $i = 2, 3, 4$ ). The test statistic  $T_i = \frac{\hat{\alpha}_i}{s_{\hat{\alpha}_i}} \sim t_{n-1}$  under  $H_0$ . The estimates  $\hat{\alpha}_i$  (and  $\hat{\mu}_1$ ) are given in column Estimate, the standard errors  $s_{\hat{\alpha}_i}$  in Std. Error, the statistics values  $T_i$  in column t value, and the **p-values in Pr(>|t|)** (found as  $P(|T| \geq |t|)$  for  $T \sim t_{n-1}$ ).

## One-way ANOVA in R: estimation (2)

```
> confint(melonaov)
              2.5 %    97.5 %
(Intercept) 17.755509 26.471158
variety2     -5.191228  7.134561
variety3     -2.639561  9.686228
variety4     -3.744561  8.581228
```

Theory gives the following  $(1 - \alpha)$ -CI's:  $[\hat{\mu} \pm t_{n-l,\alpha/2} s_{\hat{\mu}}]$  and  $[\hat{\alpha}_i \pm t_{n-l,\alpha/2} s_{\hat{\alpha}_i}]$ .

In this case, the 95% confidence intervals are for  $\mu = \mu_1$ : [17.755509, 26.471158]; for  $\alpha_2 = \mu_2 - \mu_1$ : [-5.191228, 7.134561]; for  $\alpha_3 = \mu_3 - \mu_1$ : [-2.639561, 9.686228]; for  $\alpha_4 = \mu_4 - \mu_1$ : [-3.744561, 8.581228].

## One-way ANOVA in R: estimation (3)

An alternative to the (default) treatment parametrization is sum parametrization. This gives a decomposition of the population means into the overall mean  $\mu$  and factor effects  $\alpha_1, \alpha_2, \alpha_3, \alpha_4$  as

$$\mu_i = \mu + \alpha_i, \quad i = 1, \dots, I, \quad \text{with the restriction } \sum_{i=1}^I \alpha_i = 0.$$

$\alpha_i$ 's are expressing the deviations from the mean, and their average is zero.

```
> contrasts(melonframe$variety)=contr.sum #to specify sum-parametrization
> melonaov=lm(yield~variety,data=melonframe); summary(melonaov)
[ some output deleted ]
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	23. 8417	1.0446	22.825	8.55e-16 ***
variety1	-1.7283	1.8092	-0.955	0.351
variety2	-0.7567	1.8092	-0.418	0.680
variety3	1.7950	1.8092	0.992	0.333

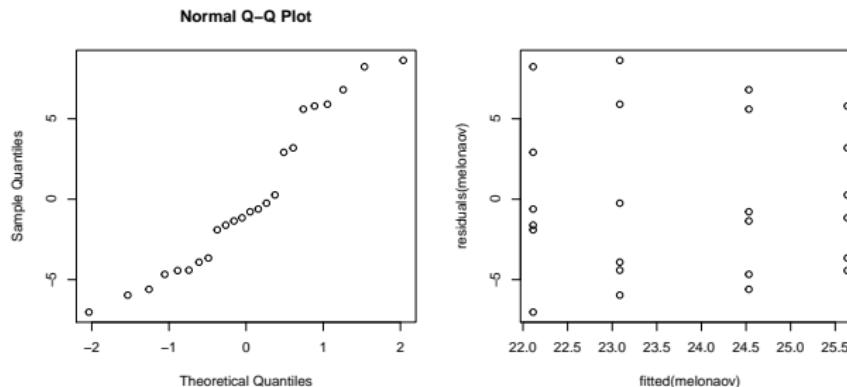
The 4 lines of the table give estimates of  $\mu, \alpha_1, \alpha_2, \alpha_3$ , now in sum-parametrization. The estimate for  $\alpha_4$  is omitted, but could be computed from  $\sum_{i=1}^4 \hat{\alpha}_i = 0$ . We can compute the estimates for the  $\mu_i$ 's:  $\hat{\mu}_i = \hat{\mu} + \hat{\alpha}_i$ ,  $i = 1, \dots, 4$  (they must be the same as before).

# One-way ANOVA in R: diagnostics

Use the data to check whether the **assumption of normality** is not totally untrue.

- The **residuals**  $\hat{e}_{ik} = Y_{ik} - \hat{\mu}_i$ ,  $k = 1, \dots, n_i$ ,  $i = 1, \dots, I$ , are in a sense “estimated errors”  $e_{ik}$  (by using the data), hence should **look normal**.
- Another important plot for checking normality is the plot of the **fitted values**  $\hat{Y}_{ik} = \hat{\mu}_i$  against the **residuals**  $\hat{e}_{ik}$ , it should show **no pattern**.

```
> par(mfrow=c(1,2)); qqnorm(residuals(melonaov))
> plot(fitted(melonaov),residuals(melonaov))
```



# If the assumptions fail?

In this course, when applying any linear model (including all ANOVA models), you need to check normality of errors by using (at least) the following **two tools**: **qqnorm plot of the residuals** and **the plot of fitted against residuals**.

- The design of the experiment ensures that the data are independent random samples from the populations.
- However, the populations might be nonnormal or have different variances.
- If the number of data points is large, then the  $p$ -value should still be accurate.
- Otherwise, you may consider
  - transforming the data (e.g. use  $\log Y$ );
  - using a different test;
  - omit some (outlying) data-points (careful!);
  - something else (there is no fix that always works).

1-way ANOVA

ooooooooooooooo

Kruskal-Wallis

●ooooooo

permutation test in 1-way ANOVA

oooooooo

## Kruskal-Wallis test (a nonparametric counterpart of ANOVA test)

# Kruskal-Wallis test: design

The **setting** and **design** are the same as in the 1-way ANOVA (consider  $n_i = N$ , the balanced design). What if the normality assumption fails?

The **Kruskal-Wallis test**

- **does not rely on the normality**, it is based on ranks;
- is a nonparametric alternative to one-way ANOVA,
- is a generalization of the Mann-Whitney test for 2 samples;
- computes the sum of the ranks of  $Y_{i1}, \dots, Y_{iN}$  for each  $i$  within the total data. Under  $H_0$  these  $N$  ranks should all lie randomly between 1 and  $N!$ .

## Data

sample 1:  $Y_{11}, Y_{12}, \dots, Y_{1N}$

sample 2:  $Y_{21}, Y_{22}, \dots, Y_{2N}$

:

sample  $I$ :  $Y_{I1}, Y_{I2}, \dots, Y_{IN}$

Assume that these are sampled independently from  $I$  populations  $F_1, \dots, F_I$  which are possibly different.

We **test**  $H_0 : F_1 = \dots = F_I$  versus  $H_1$ : at least two distributions are different.

# Kruskal-Wallis test: setting and analysis

**Setting:** measurements  $Y_{ik}$  for  $i = 1, \dots, I$  and  $k = 1, \dots, n_i$  from  $I$  different populations,  $Y_{ik}$  follows distribution  $F_i$  of population  $i$ .

**Hypotheses:**  $H_0 : F_1 = \dots = F_k$  versus  $H_1 : F_i \neq F_j$  for some  $i, j$ .

**Test statistic:**  $W = \frac{12}{n(n+1)} \sum_{i=1}^I n_i \bar{R}_i^2 - 3(n+1)$ , where  $N = n_1 + \dots + n_I$  and  $\bar{R}_i = \sum_{k=1}^{n_i} R_{ik} / n_i$  is the average pooled rank of the observations in sample  $i$ ,  $R_{ik}$  is the rank (among all observations) of observation  $k$  from group  $i$ .

**Distribution of  $W$  under  $H_0$ :**  $\chi^2_{I-1}$  (approximately), the test is one sided.

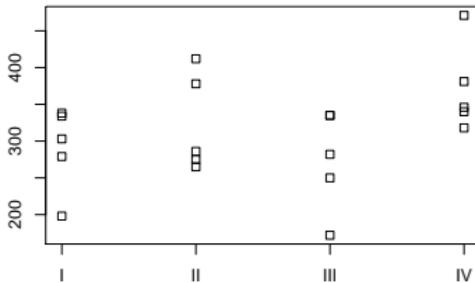
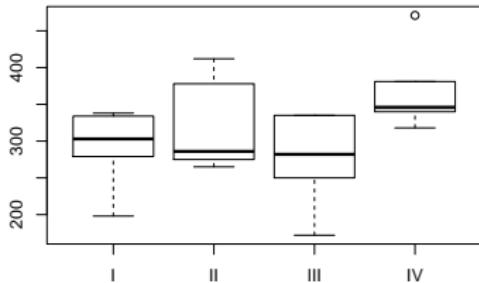
**Assumption:** all  $n_i > 5$ .

In R: `kruskal.test(y,f,data=...)`, where  $y$  is the outcome,  $f$  is the factor.

# Analysis in R: data input and graphics

The dataset `ratdata.txt` contains the number of worms in rats in 4 different treatment groups.

```
> ratdata=read.table("ratdata.txt",header=TRUE); ratdata
   I   II   III   IV
1 279  378  172  381
2 338  275  335  346
3 334  412  335  340
4 198  265  282  471
5 303  286  250  318
> boxplot(ratdata); stripchart(ratdata,vertical=TRUE)
```



# Analysis in R: data input

Create a data frame with the first columns containing all the outcomes  $Y_{i,n}$  and the second column that indicates the levels of the factor factor.

```
> ratframe=data.frame(worms=as.vector(as.matrix(ratdata)),
+                      group=as.factor(rep(1:4,each=5)))
> ratframe[1:6,]
   worms group
1     279     1
2     338     1
3     334     1
4     198     1
5     303     1
6     378     2
> is.factor(ratframe$group); is.numeric(ratframe$group)
[1] TRUE
[1] FALSE
```

# Analysis in R: testing (1)

Now we perform the Kruskal-Wallis test.

```
> attach(ratframe); kruskal.test(worms,group)

Kruskal-Wallis rank sum test

data: worms and group
Kruskal-Wallis chi-squared = 6.2047, df = 3, p-value = 0.1021
```

The command `kruskal.test` performs the Kruskal-Wallis test and yields a *p*-value. The *p*-value for testing  $H_0 : F_1 = F_2 = F_3 = F_4$  is 0.1021, hence  $H_0$  is not rejected.

## Analysis in R: testing (2)

Compare the result of Kruskal-Wallis test with the ANOVA test results:

```
> rataov=lm(worms~group); anova(rataov)
```

Analysis of Variance Table

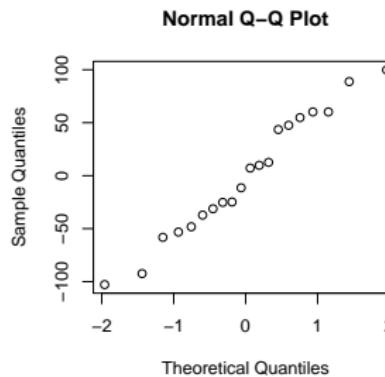
Response: worms

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
group	3	27234	9078.1	2.2712	0.1195
Residuals	16	63954	3997.1		

The one-way ANOVA also does not yield a significant difference.

```
> qqnorm(rataov$residuals)
```

The residuals do not seem to deviate significantly from normal, and both tests could be used here.



1-way ANOVA

ooooooooooooooo

Kruskal-Wallis

ooooooo

permutation test in 1-way ANOVA

●ooooooo

permutation tests in the setting of one-way ANOVA

# Setting and design

**Setting:** an experiment with

- a **numerical outcome**  $Y$ ,
- a **factor** that can be fixed at  $I$  levels ("label").

The same setting as for 1-way ANOVA, the sample sizes for labels may differ.

**EXAMPLE** Medical experiment with outcome **age at onset** of a certain disease and label **blood type**.

**EXAMPLE** Quality of a genetic algorithm to determine the minimal value of a criterion function with outcome **CPU time needed to find true minimum** and label **mutation probability** set to 0.01, 0.02, 0.03, 0.04 or 0.05.

**Design:**

- Select  $I$  different labels
- Select  $n_i$  experimental units randomly from the population of label  $i$ .
- Perform the experiment  $n_1 + n_2 + \dots + n_I$  times, independently.

# Analysis

## Data

sample 1:  $Y_{11}, Y_{12}, \dots, Y_{1n_1}$

sample 2:  $Y_{21}, Y_{22}, \dots, Y_{2n_2}$

:

sample  $I$ :  $Y_{I1}, Y_{I2}, \dots, Y_{In_I}$ .

Assume that these are sampled independently from  $I$  populations  $F_1, \dots, F_I$  which are possibly different.

We **test** the null hypothesis  $H_0 : F_1 = F_2 = \dots = F_I$  versus the alternative  $H_1 : F_i \neq F_j$  for some  $(i, j)$ .

The idea: we **choose a test statistic** that expresses the conjectured differences between the  $I$  levels, and **simulate** the distribution of this statistic under  $H_0$ .

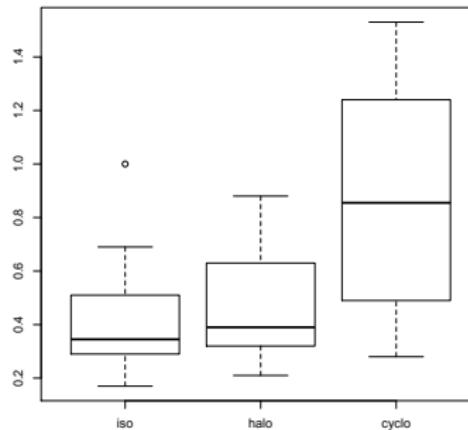
The same null hypothesis as in the Kruskal Wallis test, the difference between the Kruskal Wallis test and permutation tests is in the test statistic.

# Analysis in R: data input and graphics

The dataset `dogs.txt` concerns measures of plasma epinephrine in dogs for three different anesthesia drugs ("iso", "halo", "cyclo").

```
> dogs=read.table("dogs.txt",header=TRUE)
> treat=factor(rep(1:3,c(10,10,10)),labels=c("iso","halo","cyclo"))
> dogsdata=data.frame(plasma=as.vector(as.matrix(dogs)),treat)
```

```
> head(dogsdata)
  plasma treat
1   0.28   iso
2   0.51   iso
3   1.00   iso
4   0.39   iso
5   0.29   iso
6   0.36   iso
> boxplot(plasma~treat,data=dogsdata)
```



# Analysis in R: testing (1)

```
> attach(dogsdata)
> mystat=function(x) sum(residuals(x)^2)
> B=1000
> tstar=numeric(B)
> for (i in 1:B) {
+   treatstar=sample(treat)    ## permuting the labels
+   tstar[i]=mystat(lm(plasma~treatstar)) }
> myt=mystat(lm(plasma~treat))
```

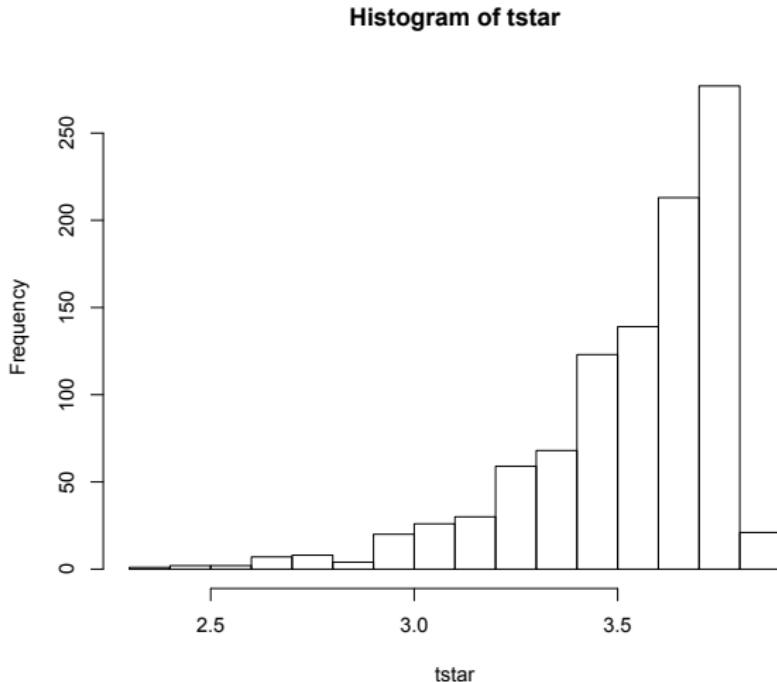
The above test statistic is the sum the squared residuals which measures the fit of one-way ANOVA model to the observed data:

$\sum_{i=1}^n \sum_{k=1}^{n_i} \hat{e}_{ik}^2 = \sum_{i=1}^n \sum_{k=1}^{n_i} (Y_{ik} - \hat{\mu}_i)^2$ , where  $\hat{\mu}_i$  is the average per label. In R, this can be programmed efficiently as `sum(residuals(lm(data~labels))^2)`. Note that we do **not use the p-values** of `lm`, we find p-values in a bootstrap fashion.

## Analysis in R: testing (2)

```
> hist(tstar)
> myt
[1] 2.72474
> pl=sum(tstar<myt)/B
> pr=sum(tstar>myt)/B
> 2*min(pl,pr)
[1] 0.022
```

The treatment is clearly significant. This is (hopefully) in line with your results using 1-way ANOVA and Kruskal-Wallis test in the corresponding assignment.



# Discussion

- A permutation test for independent samples can be performed with [any test statistic](#) that expresses difference between the samples.
- An alternative to the permutation test for independent samples is the Kruskal-Wallis test.
- Nearly all hypotheses concerning the dependence of some quantity on different levels of a "treatment" can be investigated using some sort of permutation.
- By permuting the categories of either the row or column factor in a [contingency table](#) (considered later on), one can test the null hypothesis of no dependence between these two factors.
- In fact a permutation test is a [bootstrap test](#), because the distribution of the test statistic is approximated by [simulation](#).

# To finish

Today we learned:

- one-way ANOVA
- Kruskal-Wallis test
- permutation tests in the setting of one-way ANOVA

Next time: 2-way ANOVA, factorial design, multiple comparisons.

# Experimental Design and Data Analysis, Lecture 5

Eduard Belitser

VU Amsterdam

2-way ANOVA

oooooooooooooooooooo

randomized block design

oooooooooooo

repeated measures

ooooooo

Friedman test

ooooooo

# Lecture overview

- ① two-way ANOVA
- ② randomized block design
- ③ repeated measures
- ④ Friedman test

2-way ANOVA

●ooooooooooooooo

randomized block design

○oooooooo

repeated measures

○○○○○

Friedman test

○○○○○

## two way ANOVA (completely randomized design)

# Two-way ANOVA: setting and data

An **experiment** with a **numerical outcome**  $Y$  and two **factors** (categorical variables) that can be fixed at  $I$  and  $J$  **levels** (**categories**), respectively.

**EXAMPLE** Outcome **time to develop mold** on bread and factors **temperature** and **humidity**.

**Data** consists of  $IJ$  independent **samples of sizes**  $n_{ij}$  from normal distributions with (possibly different) means  $\mu_{ij}$ , and with **equal variances**:

sample  $(i, j)$ :  $Y_{ij1}, Y_{ij2}, \dots, Y_{ijn_{ij}}$ ,  $i = 1, \dots, I$ ;  $j = 1, \dots, J$ .

Mathematically:  $Y_{ijk} \stackrel{ind}{\sim} N(\mu_{ij}, \sigma^2)$ , or  $Y_{ijk} = \mu_{ij} + \epsilon_{ijk}$  with  $\epsilon_{ijk} \stackrel{ind}{\sim} N(0, \sigma^2)$ .

Commonly, **balanced design**:  $n_{ij} = N$  for all subgroups  $(i, j)$ .

We want to **test** the following null hypotheses:

- no interaction between the two factors A and B,
- no main effect of the first factor A,
- no main effect of the second factor B.

The **overall** nullhypothesis  $H_0 : \mu_{ij} = \mu_{kl}$  for every  $i, j, k, l$  is of modest interest.

# Design

- Select  $N|J$  experimental units randomly from the population of interest.
- Assign combined levels  $(i,j)$  of the factors to a random set of  $N$  units.
- Independently perform the  $N|J$  experiments.

Randomization in R:

```
> I=4; J=2; N=3
> rbind(rep(1:I,each=N*J),rep(1:J,N*I),sample(1:(N*I*J)))
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
[1,]    1    1    1    1    1    1    2    2    2    2    2    2    2    3
[2,]    1    2    1    2    1    2    1    2    1    2    1    2    1    1
[3,]   20    1    3   14   17   24   19   12   22   13   16   15    4
     [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
[1,]    3    3    3    3    3    4    4    4    4    4    4    4    4
[2,]    2    1    2    1    2    1    2    1    2    1    2    1    2
[3,]   23    8   10    2    7   21    9    5    6   18   11
```

For unit 20 use levels (1,1) of (factor 1, factor 2); for unit 1 use levels (1,2);  
 . . .; for unit 11 use levels (4,2).

# Two-way ANOVA model: assumptions

The **two-way ANOVA** model is:

$$Y_{ijk} = \mu_{ij} + e_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + e_{ijk}, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad k = 1, \dots, n_{ij}.$$

**Assumption:** the indep. errors  $e_{ijk} \sim N(0, \sigma^2)$ , with unknown variance  $\sigma^2$ .

We decomposed the  $(i, j)$ -group means as  $\mu_{ij} = \mu + \alpha_i + \beta_j + \gamma_{ij}$ , where

- $\mu$  is the **overall mean**,
- $\alpha_i$  is the **main effect** of level  $i$  of the **first factor A**,
- $\beta_j$  is the **main effect** of level  $j$  of the **second factor B**,
- $\gamma_{ij}$  is the **interaction effect** of levels  $i$  and  $j$  of the first and second factors.

Now we can formalize the hypothesis to test:

- $H_{AB} : \gamma_{ij} = 0$  for every  $(i, j)$  (no interactions between factors A and B),
- $H_A : \alpha_i = 0$  for every  $i$  (no main effect of factor A),
- $H_B : \beta_j = 0$  for every  $j$  (no main effect of factor B).

For the parameters to be identifiable, we need to impose  $I + J + 1$  linear restrictions, (done by command **contrasts** in R). The default in R is the **treatment** parametrization:

$\alpha_1 = \beta_1 = \gamma_{1j} = \gamma_{i1} = 0$ ,  $j = 1, \dots, J$ ,  $i = 1, \dots, I$ . Often one uses the **sum** parametrization:  $\sum_i \alpha_i = 0$ ,  $\sum_j \beta_j = 0$ ,  $\sum_i \gamma_{ij} = 0$  for all  $j = 1, \dots, J$ , and  $\sum_j \gamma_{ij} = 0$  for all  $i = 1, \dots, I$ .

# Tests in two-way ANOVA

**Setting:** a two-way ANOVA model:  $Y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + e_{ijk}$ .

**Hypotheses:** we want to test  $H_{AB}$ ,  $H_A$ ,  $H_B$  against their negations.

**Test statistics:**  $F_{AB}$  for testing  $H_{AB}$ ,  $F_A$  for testing  $H_A$ , and  $F_B$  for testing  $H_B$ .

**Distribution of  $F$ 's under  $H_{AB}$ ,  $H_A$ ,  $H_B$ :**  $F_{AB} \sim F_{(I-1)(J-1), n-IJ}$ ,  $F_A \sim F_{I-1, n-IJ}$ ,  $F_B \sim F_{J-1, n-IJ}$ .  $F_{m,k}$  is the **F-distribution** with  $m$  and  $k$  degrees of freedom.

**Test:** larger values of  $F_{AB} = f_{AB}$  give **more evidence against  $H_{AB}$** , hence we reject  $H_{AB}$  if  $F_{AB}$  is large. The test is therefore **always right-sided**: compare the  $p$ -value  $p_{right} = P(F > f_{AB})$  with a significance level  $\alpha$ . Similarly for  $F_A$ ,  $F_B$ .

**In R:** the  $p$ -value is in `anova(lm(y~f1*f2))`, with `f1` and `f2` the two factors.

**Balanced design:** equal group size  $n_{ij} = N$  for each  $i$  and  $j$ , thus  $n = NIJ$ .

Formula `y~f1*f2` is the same as `y~f1+f2+f1:f2`, meaning that the model includes  $\mu$  ( $\mu$  is always included by default), and all  $\alpha_i$ 's,  $\beta_j$ 's and  $\gamma_{ij}$ 's.

If  $H_{AB}$  is not rejected (i.e., we concluded that all  $\gamma_{ij} = 0$ ), then it is **proper practice** to test for main effects A and B under the **additive model**  $\mu_{ij} = \mu + \alpha_i + \beta_j$  (in R: `y~f1+f2`). Otherwise, we can proceed to test for main effects using the full model.

# $F$ -statistics in two-way ANOVA

The idea of the  $F$ -statistics is  $F = \frac{\text{explained variance}}{\text{unexplained variance}} = \frac{\text{between-groups SS}}{\text{within-groups SS}}$ .

Denote the total mean  $\bar{Y}_{...} = \frac{1}{I} \sum_{i=1}^I \frac{1}{J} \sum_{j=1}^J \frac{1}{n_{ij}} \sum_{k=1}^{n_{ij}} Y_{ijk}$ , and

$$\bar{Y}_{ij\cdot} = \frac{1}{n_{ij}} \sum_{k=1}^{n_{ij}} Y_{ijk}, \quad \bar{Y}_{i..} = \frac{1}{J} \sum_{j=1}^J \frac{1}{n_{ij}} \sum_{k=1}^{n_{ij}} Y_{ijk}, \quad \bar{Y}_{.j\cdot} = \frac{1}{I} \sum_{i=1}^I \frac{1}{n_{ij}} \sum_{k=1}^{n_{ij}} Y_{ijk}.$$

The test statistics are

$$F_{AB} = \frac{\sum_{i=1}^I \sum_{j=1}^J n_{ij} (\bar{Y}_{ij\cdot} - \bar{Y}_{i..} - \bar{Y}_{.j\cdot} + \bar{Y}_{...})^2 / ((I-1)(J-1))}{\sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^{n_{ij}} (Y_{ijk} - \bar{Y}_{ij\cdot})^2 / (n - IJ)},$$

$$F_A = \frac{\sum_{i=1}^I \sum_{j=1}^J n_{ij} (\bar{Y}_{i..} - \bar{Y}_{...})^2 / (I-1)}{\sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^{n_{ij}} (Y_{ijk} - \bar{Y}_{ij\cdot})^2 / (n - IJ)},$$

similarly for  $F_B$ .

# General form of ANOVA tables

One-way ANOVA results are usually presented in an one-way [ANOVA table](#):

Source	Df	Sum Sq	Mean Sq	F value	p-value
Factor A	$I - 1$	$SS_A$	$SS_A/(I - 1)$	$F_A = \frac{SS_A/(I-1)}{RSS/(n-I)}$	$P_A(F_A > f)$
Residuals	$n - I$	$RSS$	$RSS/(n - I)$		
Total	$n - 1$	$SS_T$			

Two-way ANOVA results are usually presented in a two-way [ANOVA table](#):

Source	Df	Sum Sq	Mean Sq	F value	p-value
Factor A	$I - 1$	$SS_A$	$SS_A/(I - 1)$	$F_A = \frac{SS_A/(I-1)}{RSS/(n-I)}$	$P_A(F_A > f)$
Factor B	$J - 1$	$SS_B$	$SS_B/(J - 1)$	$F_B = \frac{SS_B/(J-1)}{RSS/(n-I)}$	$P_B(F_B > f)$
Interaction	$(I - 1)(J - 1)$	$SS_{AB}$	$SS_{AB}/(I - 1)(J - 1)$	$F_{AB} = \frac{SS_{AB}/[(I-1)(J-1)]}{RSS/(n-I)}$	$P_{AB}(F_{AB} > f)$
Residuals	$n - IJ$	$RSS$	$RSS/(n - IJ)$		
Total	$n - 1$	$SS_T$			

$$SS_T = SS_A + SS_B + SS_{AB} + RSS = \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^{n_{ij}} (y_{ijk} - \bar{y}_{...})^2.$$

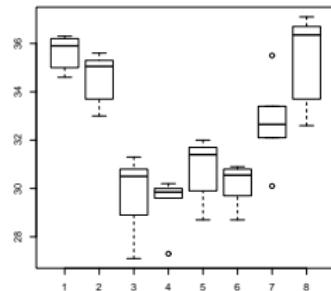
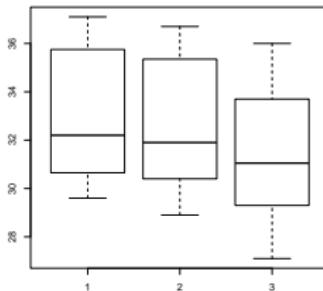
## Example pvc

The following data is from an experiment to study factors affecting the production of the plastic PVC, 3 operators used 8 different devices called resin railcars to produce PVC, two samples for each of the 24 combinations.

```
> pvc=read.table(file="pvc.txt",header=TRUE)
> pvc[1:4,]
  psizer operator resin
1   36.2        1      1
2   36.3        1      1
3   35.3        1      2
4   35.0        1      2
```

```
> attach(pvc)
> boxplot(psizer~operator)
> boxplot(psizer~resin)
```

These pictures give an idea of the main effects of the factors. Interactions are not visible.

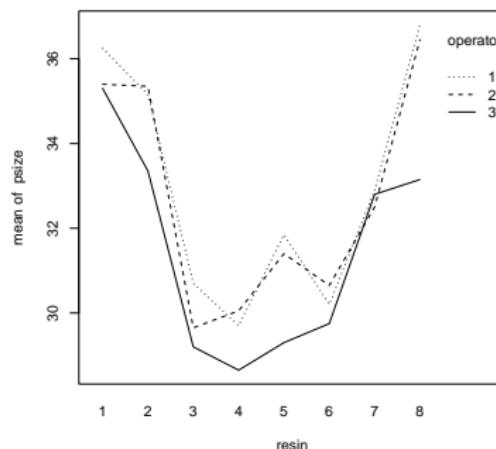
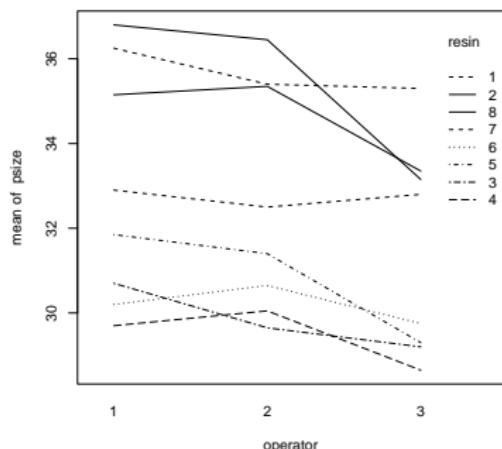


## Example pvc: interaction plots

An **interaction plot** fixes one factor and plots the average outcome (vertical axis) against the levels of the other factor (horizontal axis).

**Interaction shows up as nonparallel curves.**

```
> interaction.plot(operator,resin,psize)
> interaction.plot(resin,operator,psize)
```



Lines may be unparallel, because of interactions, but also because of noise in the data.

## Example pvc (anova test)

```
> pvc$operator=as.factor(pvc$operator); pvc$resin=as.factor(pvc$resin)
> pvcaov=lm(psize~operator*resin); anova(pvcaov)
[ some output deleted ]
Response: psize
          Df  Sum Sq Mean Sq F value    Pr(>F)
operator      2  20.718  10.359  7.0072  0.00401 ***
resin         7 283.946  40.564 27.4388 5.661e-10 ***
operator:resin 14  14.335   1.024  0.6926  0.75987
Residuals     24  35.480   1.478
```

The  $p$ -value for testing  $H_0 : \alpha_i = 0$  for all  $i$  is 0.00401; for  $H_0 : \beta_j = 0$  for all  $j$  is 5.661e-10; for  $H_0 : \gamma_{i,j} = 0$  for all  $(i,j)$  is 0.75987. So, there is no evidence for interaction (both factors seems to have a main effect but one should not draw conclusions about the factors at this stage).

The command `as.factor` (or `factor`) is necessary, because the 2nd and 3rd columns of the data matrix were read in as numerical variables (with values 1, 2, 3, 4), but should be treated as factors in the analysis.

## Example pvc: summary command

```
> summary(pvcaov) # estimates in the default treatment contrasts
[ some output deleted ]
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    36.2500   0.8598  42.164 < 2e-16 ***
operator2      -0.8500   1.2159  -0.699 0.491216
operator3      -0.9500   1.2159  -0.781 0.442245
resin2         -1.1000   1.2159  -0.905 0.374615
[ some output deleted ]
resin8          0.5500   1.2159   0.452 0.655078
operator2:resin2 1.0500   1.7195   0.611 0.547175
[ some output deleted ]
operator3:resin8 -2.7000   1.7195  -1.570 0.129454
```

The output of `summary(pvcaov)` shows estimates of  $\mu, \alpha_2, \alpha_3, \beta_2, \dots, \beta_8, \gamma_{22}, \dots, \gamma_{38}$  in the default `treatment` parametrization:  $\alpha_1 = \beta_1 = \gamma_{1j} = \gamma_{i1} = 0$ ,  $i = 1, 2, 3$ ,  $j = 1 \dots, 8$ . The corresponding estimates  $\hat{\alpha}_1 = \hat{\beta}_1 = \hat{\gamma}_{11} = \dots = \hat{\gamma}_{31} = 0$  are not shown. The  $p$ -values in column `Pr(>|t|)` are for testing the `individual` null hypothesis that the coefficient is 0. The test statistic, computed as  $t \text{ value} = \frac{\text{Estimate}}{\text{Std. Error}}$ , has  $t_{n-IJ}$ -distribution under  $H_0$ .

## Example pvc: (summary in sum-parametrization)

The command `contrasts` overrules the default `treatment` parametrization (e.g., to `sum` parameterization), `lm` and `anova` have to be run again.

```
> contrasts(pvc$operator)=contr.sum; contrasts(pvc$resin)=contr.sum
> pvcaov2=lm(psize~operator*resin,data=pvc); summary(pvcaov2)
[ some output deleted ]
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	32.35417	0.17550	184.359	< 2e-16 ***
operator1	0.58958	0.24819	2.376	0.025855 *
operator2	0.32708	0.24819	1.318	0.199983
resin1	3.29583	0.46432	7.098	2.45e-07 ***
[ some output deleted ]				
resin7	0.37917	0.46432	0.817	0.422183
operator1:resin1	0.01042	0.65664	0.016	0.987474
[ some output deleted ]				
operator2:resin7	-0.56042	0.65664	-0.853	0.401844

The output shows estimates of  $\mu, \alpha_1, \alpha_2, \beta_1, \dots, \beta_7, \gamma_{11}, \gamma_{12}, \dots$  in the `sum parametrization`. The estimates of  $\alpha_3$  (operator 3) and  $\beta_8$  (resin 8) are not shown.

These can be found from the restrictions  $\sum_{i=1}^3 \hat{\alpha}_i = 0$ ,  $\sum_{j=1}^8 \hat{\beta}_j = 0$ ; similarly for the interactions:  $\sum_{i=1}^3 \hat{\gamma}_{ij} = 0$  for  $j = 1, \dots, 8$  and  $\sum_{j=1}^8 \hat{\gamma}_{ij} = 0$  for  $i = 1, 2, 3$ . The p-values in  $\text{Pr}(>|t|)$  are for testing `individual` hypothesis  $H_0: \text{coefficient}=0$ .

## Example pvc: additive model

As we see, the previous analysis says there is no interaction. Now we remove interaction term from the model and fit the [additive model](#)

$$Y_{ijk} = \mu + \alpha_i + \beta_j + e_{ijk}, \quad i = 1, \dots, I, \quad j = 1, \dots, J, \quad k = 1, \dots, n_{ij},$$

i.e., now  $\mu_{ij} = \mu + \alpha_i + \beta_j$ .

```
> pvc$operator=as.factor(pvc$operator); pvc$resin=as.factor(pvc$resin)
> pvcaov=lm(psize~operator+resin,data=pvc)
> anova(pvcaov)
[ some output deleted ]
Response: psize
          Df  Sum Sq  Mean Sq   F value    Pr(>F)
operator      2  20.718  10.359     7.902    0.00135 ***
resin         7 283.946  40.564    30.943  8.111e-14 ***
Residuals    38  49.815   1.311

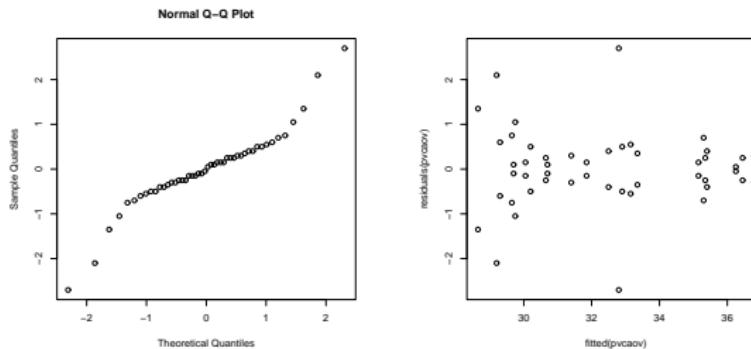
```

The  $p$ -value for testing  $H_A : \alpha_i = 0$  for all  $i$  is 0.00135; for  $H_B : \beta_j = 0$  for all  $j$  is  $8.111e - 14$ . So both factors have a main effect in the additive model.

## Example pvc: (checking model assumptions)

We check the normality and the assumption of equal variances. The **residuals**  $\hat{e}_{ijk} = Y_{ijk} - \hat{\mu} - \hat{\alpha}_i - \hat{\beta}_j - \hat{\gamma}_{ij}$  are the data corrected for the different population means and ought to look normal. The **fitted value**  $\hat{Y}_{ijk}$  for  $Y_{ijk}$  is the estimated mean  $\hat{Y}_{ijk} = \hat{\mu}_{ij} = \hat{\mu} + \hat{\alpha}_i + \hat{\beta}_j + \hat{\gamma}_{ij}$ . The spread in the residuals should not change systematically with any variable, in particular not with the fitted values.

```
> qqnorm(residuals(pvcaov2)); plot(fitted(pvcaov2),residuals(pvcaov2))
```



Left plot: normality is doubtful. Right plot: the spread in the residuals seems to be bigger for smaller fitted values. Some data-points also seem extreme. Perhaps transform the data or consider **outliers**.

# One observation per cell (1)

The following dataset contains the strength of a thermoplastic composite depending on power of a laser and speed of a tape.

```
> composite=read.table("composite.txt",head=T); composite
   strength  laser    tape
1      25.66   40W   slow
2      29.15   50W   slow
3      35.73   60W   slow
4      28.00   40W medium
5      35.09   50W medium
6      39.56   60W medium
7      20.65   40W  fast
8      29.79   50W  fast
9      35.66   60W  fast
```

Notice that we have only one observation per cell (i.e., per each combination of levels of the two factors `laser` and `tape`). But then there is a problem in the test statistics  $F$  for interaction: since  $n_{ij} = 1$ ,  $n = IJ$  and the denominator  $RSS/(n - IJ)$  is not well defined. To estimate and test interaction effects, it is necessary to have at least 2 observations per combination  $(i, j)$  of factor levels.

## One observation per cell (2)

R produces a warning message if the data is not sufficient to fit the model, in this case it is impossible to estimate interactions with one observation per cell:

```
> attach(composite); anova(lm(strength~laser*tape))
      Df  Sum Sq Mean Sq F value Pr(>F)
laser       2 224.184 112.092
tape        2  48.919  24.459
laser:tape  4  10.503   2.626
Residuals    0    0.000
Warning message:
In anova.lm(lm(strength ~ laser * tape, data = composite)) :
  ANOVA F-tests on an essentially perfect fit are unreliable
```

If it can be assumed a priori that all interactions are 0, then it is possible to test and estimate main effects. (Interaction plots may help to justify this assumption.)

```
> anova(lm(strength~laser+tape,data=composite))
      Df  Sum Sq Mean Sq F value    Pr(>F)
laser       2 224.184 112.092 42.6893 0.002003 ***
tape        2  48.919  24.459  9.3151 0.031242 *
Residuals    4  10.503   2.626
```

2-way ANOVA

oooooooooooooooooooo

randomized block design

●oooooooooo

repeated measures

ooooooo

Friedman test

ooooooo

## randomized block design

# Setting

An experiment with:

- a **numerical outcome**  $Y$  ("dependent variable"),
- a **factor** of interest that can be fixed at  $I$  levels ("treatment"),
- a **factor** that is *not* of interest that can be fixed at  $B$  levels ("block").

The purpose is to understand the dependence of  $Y$  on the **treatment factor**.

The **block variable** is thought (or known) to be of influence. It is used to create homogeneous groups of experimental units, in which the treatment effect is easier to see and not blurred by variation due to the block factor.

**EXAMPLE** Chemical production process with outcome **total yield**, treatment variable **temperature** fixed at levels low, medium and high and block **blend of raw material**.

**EXAMPLE** Study of web design with outcome **total time of a user on webpage**, treatment variable **type of design** and block **user skill**. Each user is tested with a single type of web design.

# Design

Independently, for  $b = 1, 2, \dots, B$ :

- select  $Nl$  experimental units randomly from the population of units with block level  $b$ ,
- assign level  $i$  of the factor to a random set of  $N$  units ( $i = 1, 2, \dots, l$ ),
- perform the experiment  $Nl$  times, independently.

Randomization in R.

```
> I=4; B=5; N=1
> for (i in 1:B) print(sample(1:(N*I)))
[1] 3 1 2 4
[1] 4 3 2 1
[1] 1 4 2 3
[1] 3 4 1 2
[1] 2 4 3 1
```

For block 1 assign unit 3 to treatment 1, unit 1 to treatment 2, etc., for block 2 assign unit 4 to treatment 1, unit 3 to treatment 2, etc.

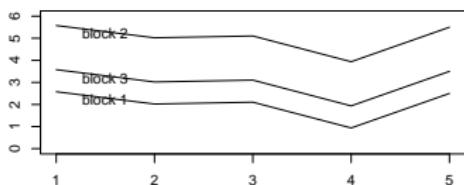
For many blocks, it is typical to use one replicate per treatment level per block:  $N = 1$ .

# Analysis

Data ( $Y_{ibk}$ ) are assumed to follow the model

$$Y_{ibk} = \mu + \alpha_i + \beta_b + e_{ibk}, \quad i = 1, \dots, I; \quad b = 1, \dots, B; \quad k = 1, \dots, N,$$

where the “errors” ( $e_{ibk}$ ) are a random sample from a **normal** population.



The pattern  $(\alpha_1, \alpha_2, \dots, \alpha_I)$  of treatment effects is assumed to be **the same within every block**.

We **test** the null hypothesis  $H_0 : \alpha_1 = \alpha_2 = \dots = \alpha_I = 0$ .

We also **estimate** the treatment effects  $\alpha_1, \alpha_2, \dots, \alpha_I$ .

The model is the same as in a two-way factorial experiment, with the block as a second factor, but with zero interactions.

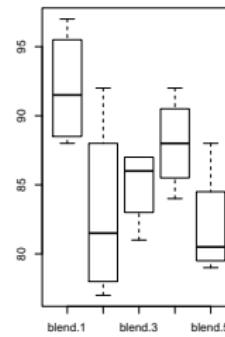
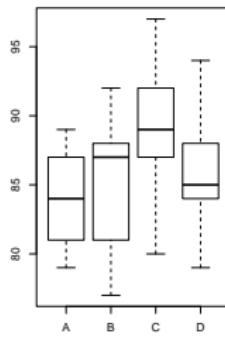
## Analysis in R: data input

The following data frame contains the data about penicillin made by production processes A, B, C, D (treatment); with 5 different blends of raw material (blocks), as in a two-way factorial experiment.

```
> penicillin
  treat   blend yield
1      A blend.1    89
2      B blend.1    88
3      C blend.1    97
4      D blend.1    94
5      A blend.2    84
[ some output deleted ]
20     D blend.5    88
> xtabs(yield~treat+blend,data=penicillin)
            blend
treat blend.1 blend.2 blend.3 blend.4 blend.5
  A      89      84      81      87      79
  B      88      77      87      92      81
  C      97      92      87      89      80
  D      94      79      85      84      88
```

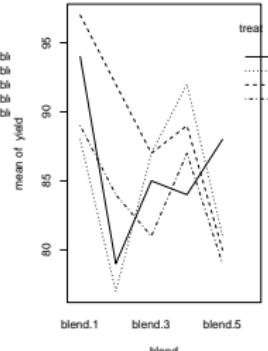
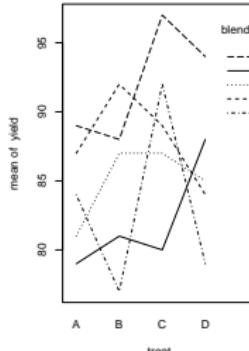
# Analysis in R: graphics

```
> attach(penicillin)
> par(mfrow=c(1,2))
> boxplot(yield~treat)
> boxplot(yield~blend)
```



```
> par(mfrow=c(1,2))
> interaction.plot(treat,blend,yield)
> interaction.plot(blend,treat,yield)
```

The left plot gives estimates of the treatment patterns per block.



# Analysis in R: testing and estimation

```
> aovpen=lm(yield~treat+blend)
> anova(aovpen)
Response: yield
      Df Sum Sq Mean Sq F value    Pr(>F)
treat     3    70   23.333  1.2389  0.33866
blend     4   264   66.000  3.5044  0.04075 *
Residuals 12   226   18.833
```

The treatment effects are not significantly different from 0. The blocks (blend) are, but this was not the research question.

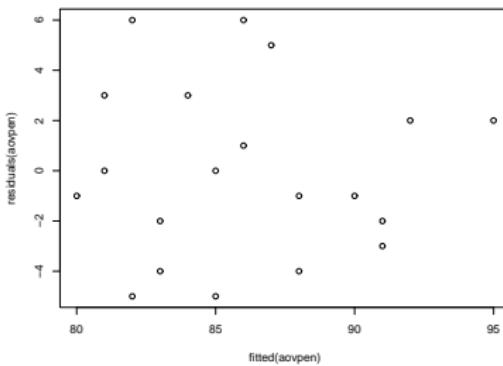
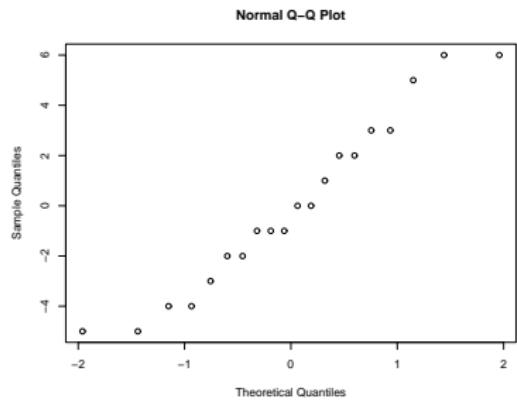
```
> summary(aovpen)
[ some output deleted ]
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	90.000	2.745	32.791	4.1e-13 ***
treatB	1.000	2.745	0.364	0.72194
treatC	5.000	2.745	1.822	0.09351 .
treatD	2.000	2.745	0.729	0.48018
blendblend.2	-9.000	3.069	-2.933	0.01254 *
blendblend.3	-7.000	3.069	-2.281	0.04159 *
blendblend.4	-4.000	3.069	-1.304	0.21686
blendblend.5	-10.000	3.069	-3.259	0.00684 **

The yield of treatment C is estimated 5 higher than that of treatment A, etc.

# Analysis in R: diagnostics

```
> qqnorm(residuals(aovpen))
> plot(fitted(aovpen),residuals(aovpen))
```



Perhaps a slight curve in the qq-plot. The interaction plots (see some slides back) can also be considered diagnostic.

# Discussion

- The **advantage** of the block design is that more precise conclusions can be obtained by removing variation, present due to block factor. The units must be **similar within the blocks**, and **dissimilar between the blocks**.
- Assuming that *the pattern of treatment effects is the same for each block* means assuming the **absence of interaction** between block and treatment. Without replications ( $N = 1$ ), this cannot be tested, with  $N > 1$  it can.
- If treatment and blocks do interact, the interpretation of the results of a factorial analysis is more subtle.
- **Multiple treatment factors:** a multi-way factorial experiment can be done within every block (rather than a one factor experiment).
- **Multiple block factors:** all combinations of levels of the block factors can be viewed as a new, single block factor, to which the block design applies.

2-way ANOVA

oooooooooooooooooooo

randomized block design

oooooooooooo

repeated measures

●ooooo

Friedman test

ooooooo

## repeated measures

# Setting and design

Setting: an experiment with

- a **numerical outcome**  $Y$  ("dependent variable"),
- a **factor** of interest that can be fixed at  $I$  levels, ("treatment").
- **experimental units** that are measured at **every** treatment level.

The purpose is to understand the dependence of  $Y$  on the **treatment factor**.

The same **experimental units** are used for every treatment, because this is thought to reduce "extraneous variation": the units serve as blocks.

For  $I = 2$  treatments, this is simply the **paired sample design**.

**EXAMPLE** Study of web design with outcome **total time on webpage**, treatment variable **type of design**. Each **user** is tested with every type of design.

**EXAMPLE** The **velocity** of a ball is measured for **different types of tennis rackets** for a number of **players**, where every player uses all types of rackets.

Design:

- Select  $B$  experimental units randomly from a population of units.
- Measure each unit at every treatment level, if possible in random order.

# Exchangeable case

Data vectors  $(Y_{1b}, Y_{2b}, \dots, Y_{lb})$  for  $B$  units are assumed to follow the model

$$Y_{ib} = \mu + \alpha_i + \beta_b + e_{ib}, \quad i = 1, \dots, I; \quad b = 1, \dots, B,$$

- the “error vectors”  $(e_{1b}, \dots, e_{lb})$  for the  $B$  units are a random sample from a (multivariate) normal distribution;
- the “errors”  $e_{1b}, \dots, e_{lb}$  within a single unit are **exchangeable**, i.e., the ordering (within a block) is irrelevant, in a way, generalizing the paired samples (in general, dependent within the same unit);
- the effects  $\beta_1, \dots, \beta_B$  of the units may be considered fixed or random.

The treatment pattern  $(\alpha_1, \dots, \alpha_I)$  is assumed to be **the same for each unit**.

We want to **test** the null hypothesis  $H_0 : \alpha_1 = \alpha_2 = \dots = \alpha_I = 0$ .

We also want to **estimate** the treatment effects  $\alpha_1, \alpha_2, \dots, \alpha_I$ .

The model is the same as in a randomized block experiment, with the units as blocks, except for the assumption on the errors. These are allowed to be **dependent** within the units, even though still “exchangeable”.

# Analysis in R: data input

Data input is as in a block design, with columns for outcome, treatment level, and block level (=identification of unit).

```
> ashinalong
  pain id order treatment
1 -167  1    pa      a
2 -102  1    pa      p
3 -127  2    pa      a
4  -39   2    pa      p
5  -58   3    pa      a
6   32   3    pa      p
7 -103  4    pa      a
8   28   4    pa      p
[ some output deleted ]
31  -72  16   ap      a
32  -36  16   ap      p
```

The data frame ashinalong contains the same data as ashina, but every individual is represented by two lines, one for the treatment with the active drug, the other for the placebo. The extra column id shows the pairing of the measurements.

# Analysis in R: exchangeable case

Analysis is as for a randomized block design, with every unit being a block.

```
> ashinalong$id=factor(ashinalong$id)
> aovashina=lm(pain~treatment+id,data=ashinalong); anova(aovashina)
Analysis of Variance Table
Response: pain
    Df Sum Sq Mean Sq F value    Pr(>F)
treatment   1 14706 14706.1 10.413 0.005644 **
id          15 51137  3409.2  2.414 0.049184 *
Residuals  15 21184  1412.3
```

Compare to the two sample *t*-test:

```
> t.test(ashina[,1],ashina[,2],paired=TRUE)
Paired t-test
data: ashina[, 1] and ashina[, 2]
t = -3.2269, df = 15, p-value = 0.005644
```

The p-value for treatment is identical to the one of the paired-sample *t*-test found previously (the order of the treatments was ignored). The p-value for id is not interesting. Note that R had to be told to treat id as labels, not as numbers.

# Discussion

- Repeated measures may **not** be exchangeable, then this model is **invalid**.
  - Time effect (in longitudinal studies): growth, increasing or decreasing variation.
  - Learning effect: subject becomes better or bored at tasks (cf. **crossover design**).
  - Dissimilar subjects: the pattern of response to treatment varies too much (too different reactions to treatments for different units).
- The discussed **repeated measures design** corrects for some dependencies.
- Taking repeated measures is attractive, because fewer experimental units are needed and “extraneous” variation between units is reduced.
- However, in many studies, in particular most “longitudinal studies”, where individuals are followed over time, the assumption of “exchangeability” fails. More complicated models are then necessary.
- Models with **random effects** (called **mixed effects models**) are a possibility.

2-way ANOVA

oooooooooooooooooooo

randomized block design

oooooooooooo

repeated measures

ooooooo

Friedman test

●oooooo

## Friedman test

# Setting and design

Setting and design for the **Friedman test** are either as in a randomized block design **with  $N = 1$**  or as in repeated measures. An experiment with:

- a numerical outcome  $Y$  ("dependent variable").
- a **factor** of interest that can be fixed at  $I$  levels. ("treatment").
- a number of **blocks** or **units** that are measured at **every** treatment level.

## Data

block1	block2	...	blockB
level 1: $Y_{11}, Y_{12}, \dots, Y_{1B}$			
level 2: $Y_{21}, Y_{22}, \dots, Y_{2B}$			
⋮			
level $I$ : $Y_{I1}, Y_{I2}, \dots, Y_{IB}$			

Data ( $Y_{ib}$ ) are **not** assumed to come from a normal distribution.

We want to test the null hypothesis of no treatment effect taking the blocks into account, by using ranks.

The underlying idea of this test: the **Friedman test** computes the ranks of the  $i$ -th measurement within each block. Under  $H_0$  the rank of  $Y_{ib}$  should lie randomly between 1 and  $I$  for each  $b$ . If the average rank of  $Y_{ib}$  (averaged over blocks) is lower/higher than expected, this indicates that  $H_0$  might not be true.

The sign test (two-sided) is equivalent to a Friedman test on two groups.

# Analysis in R: data input

The dataset `itch.tx` contains the numbers of hours subjects were itching after treatment with 7 different drugs (incl. No\_Drag and Placebo) against itching.

```
> itch=read.table("itch.txt",header=TRUE,sep=","); itch
```

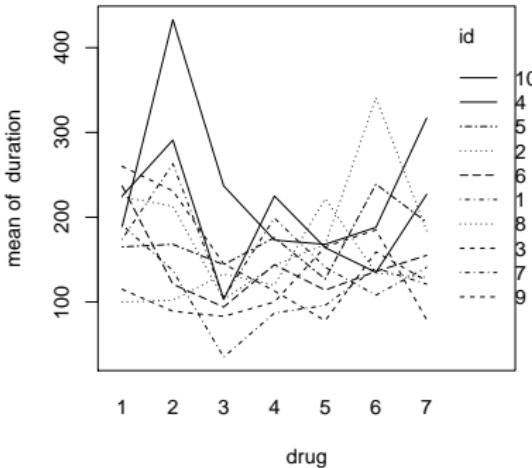
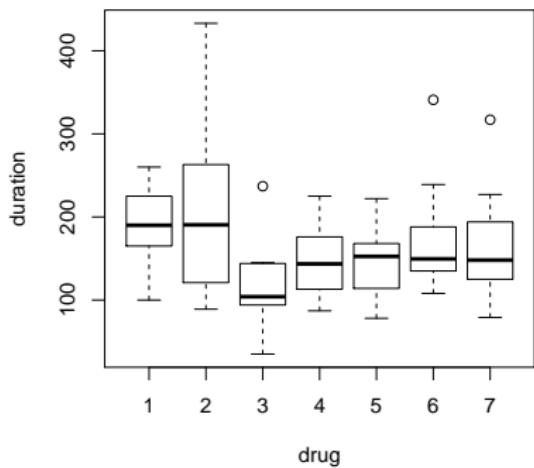
	Subject	No_Drug	Placebo	Papaverine	Morphine	Aminophylline	Pentobarbital	Triplennamine
1	BG	174	263	105	199	141	108	141
2	JF	224	213	103	143	168	341	184
3	BS	260	231	145	113	78	159	125
4	SI	225	291	103	225	164	135	227
5	BW	165	168	144	176	127	239	194
6	TS	237	121	94	144	114	136	155
7	GM	191	137	35	87	96	140	121
8	SS	100	102	133	120	222	134	129
9	MU	115	89	83	100	165	185	79
10	OS	189	433	237	173	168	188	317

Create a data frame with duration as 1st, id as 2d, and drug as 3d columns.

```
> duration=as.vector(as.matrix(itch[,2:8]))
> id=as.factor(rep(1:10,7)); drug=as.factor(rep(1:7,each=10))
> itchdata=data.frame(cbind(duration,id,drug)); itchdata[1:3,]
   duration id drug
1       174  1    1
2       224  2    1
3       260  3    1
```

# Analysis in R: graphics

```
> boxplot(duration~drug,xlab="drug",ylab="duration")
> interaction.plot(drug,id,duration)
```



Parallel lines in the interaction plot indicate that there is no significant interaction effect. But beware that we're dealing with  $N = 1$ .

# Analysis in R: testing (1)

```
> friedman.test(duration,drug,id,data=itchdata)

  Friedman rank sum test

data: duration, drug and subject
Friedman chi-squared = 14.2796, df = 6, p-value = 0.02666
```

Command `friedman.test(duration,drug,id,data=itchdata)` performs the Friedman test, testing the **relevance of factor drug** taking into account the **blocking factor id**. The *p*-value for testing ( $H_0$  : no treatment effect) is 0.02666, so  $H_0$  is rejected, there is a treatment effect.

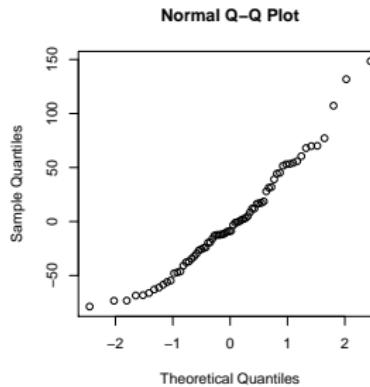
## Analysis in R: testing (2)

Compare the Friedman test results to results for the repeated measures design:

```
> itchaov=lm(duration~drug+subject); anova(itchaov)
Analysis of Variance Table
Response: duration
  Df Sum Sq Mean Sq F value    Pr(>F)
drug      6 51487  8581.2  2.7893 0.019494 *
subject    9 101253 11250.3  3.6569 0.001261 **
Residuals 54 166127  3076.4
```

```
> qqnorm(itchaov$residuals)
```

In a randomized block design we also find a significant treatment effect. The QQ-plot looks ok, perhaps slightly bowed.



2-way ANOVA

oooooooooooooooooooo

randomized block design

oooooooooooo

repeated measures

ooooooo

Friedman test

oooooo●

# To finish

Today we discussed:

- ① 2-way ANOVA
- ② randomized block design
- ③ repeated measures
- ④ Friedman test

Next time: general factorial and incomplete block designs, random effects, more block designs.

# Experimental Design and Data Analysis, Lecture 7

Eduard Belitser

VU Amsterdam

# Lecture overview

- ① contingency tables
  - ① chisquare test
  - ② Fisher test
- ② simple linear regression
- ③ multiple linear regression

## contingency tables

# Setting

An experiment with:

- a **count** of individuals or units in different categories of two **factors**.

Interest is in a possible dependence of the two factors.

**EXAMPLE** Study possible dependency between **blood group** and **disease** by counting the **number of patients** having a certain blood group (A, B or O) and a certain disease (stomach cancer, kidney cancer, no disease).

**EXAMPLE** Study possible dependency between **web layout** and **size of a company** by counting the **number of companies** of a certain size (small, moderate, large) using a certain web design (relative, fixed, elastic, liquid) .

**EXAMPLE** Consider the following (fictive) counts amongst 60 VU-students:

	exact	arts	total
men	23	17	40
women	7	13	20
total	30	30	60

Question: study and gender **independent?**

# Design

## Design A:

- Take a random sample of experimental units from the relevant population.
- Count for each cross-category the number of units falling into that cross-category.

## Design B:

- Take for each category of the first (row) factor a random sample of experimental units.
- Count for each category of the second factor the number of units falling into that cross-category.

## Design C:

- Take for each category of the second (column) factor a random sample of experimental units.
- Count for each category of the first factor the number of units falling into that cross-category.

# Analysis (1)

The general form of a **contingency table** is

$n_{11}$	$n_{12}$	$\cdots$	$n_{1J}$	$n_{1\cdot}$
$n_{21}$	$n_{22}$	$\cdots$	$n_{2J}$	$n_{2\cdot}$
$\vdots$		$\ddots$	$\vdots$	$\vdots$
$n_{J1}$	$n_{J2}$	$\cdots$	$n_{JJ}$	$n_{J\cdot}$
$n_{\cdot 1}$	$n_{\cdot 2}$	$\cdots$	$n_{\cdot J}$	$n_{\cdot \cdot}$

We want to test whether the two factors are **independent** (under design A):

$H_0$  : *row variable and column variable are independent.*

Or, we want to test whether the distributions are **homogeneous** over rows (design B) or columns (design C):

$H_0$  : *the distributions over row (column) factors are equal.*

## Analysis (2)

Let  $n = n_{..}$  be the total number of observations. Under the null hypothesis of no dependence (or homogeneity), the counts are expected to be in proportion:

$$E_{ij} = np_{ij} = np_i \cdot p_j = n \frac{n_i}{n} \frac{n_j}{n} = \frac{n_i \cdot n_j}{n}.$$

Expected counts in the example data set:

	exact	arts	total		exact	arts	total
men	?	?	40	men	$60 \cdot \frac{40}{60} \cdot \frac{30}{60}$	$60 \cdot \frac{40}{60} \cdot \frac{30}{60}$	40
women	?	?	20	women	$60 \cdot \frac{20}{60} \cdot \frac{30}{60}$	$60 \cdot \frac{20}{60} \cdot \frac{30}{60}$	20
total	30	30	60	total	30	30	60

The **test statistic** is based on the (appropriately normalized) **differences** between the **expected counts**  $E_{ij}$  under  $H_0$  and the **observed** counts  $n_{ij}$ :

$$T = \sum_{i=1}^I \sum_{j=1}^J \frac{(n_{ij} - E_{ij})^2}{E_{ij}} \sim \chi^2_{(I-1)(J-1)}, \quad (\text{approx. a } \text{chisquare distribution}).$$

The  $p$ -value is **always right-sided**:  $p_{right} = P(T > t)$ . Why?

**Condition:** For the test to be reliable, at least **80%** of the  $E_{ij}$ 's should be **at least 5**.

In R: `chisq.test(data)`

# Analysis in R: data input

First, we need to create a table of the counts in the form of a matrix.

The following data consists of grade counts in an elementary statistics class, classified by the students' majors.

```
> grades=matrix(c(8,15,13,14,19,15,15,4,7,3,1,4),byrow=TRUE,ncol=3,nrow=4,  
+ dimnames=list(c("A","B","C","D-F"),c("Psychology","Biology","Other")))  
> grades  
Psychology Biology Other  
A 8 15 13  
B 14 19 15  
C 15 4 7  
D-F 3 1 4
```

For the calculations on the next slide, *R* needs the data in a **matrix** object, rather than in a **dataframe** format.

# Analysis in R: testing (1)

```
> rowsums=apply(grades,1,sum); colsums=apply(grades,2,sum)
> total=sum(grades); expected=(rowsums%*%t(colsums))/total
> round(expected,0)
      Psychology Biology Other
[1,]        12     12     12
[2,]        16     16     16
[3,]         9      9      9
[4,]         3      3      3
> sum((grades-expected)^2/expected) # realization of statistics T
[1] 12.18346
> 1-pchisq(12.18346,6)    # p-value for the observed T=12.18346
[1] 0.05799897
```

Less than 80% of the expected counts are above 5. Hence, the approximation by a chi-square test is not reliable.

## Analysis in R: testing (2)

Of course, no need to perform all these computations, just use build-in R command: `chisq.test`, which executes the  $\chi^2$ -test.

```
> z=chisq.test(grades); z
                  Pearson's Chi-squared test
data:  grades
X-squared = 12.1835, df = 6, p-value = 0.058
```

Warning message:

In `chisq.test(grades)` : Chi-squared approximation may be incorrect

R gives a warning because the chi-squared approximation in this case is **not reliable**. In such a case one can use the setting `simulate.p.value=TRUE`, which computes a  $p$ -value in a bootstrap fashion. This may yield a very different  $p$ -value.

```
> chisq.test(grades,simulate.p.value=TRUE)
Pearson's Chi-squared test with simulated p-value (based on 2000 replicates)

data:  grades
X-squared = 12.1835, df = NA, p-value = 0.05647
```

## Analysis in R: testing (3)

You can extract information from `z=chisq.test(grades)`: `z$expected` gives the table of expected values, `z$observed` recovers the observed values. We can look at the (square root) contributions of each cell to the chi-squared statistics, by using `residuals(z)` (or `z$residuals`), to determine which observed values deviate most from the expected under  $H_0$ .

```
> residuals(z) # = (z$observed-z$expected)/sqrt(z$expected)
   Psychology      Biology      Other
A    -1.2032599  0.8992005  0.3193881
B    -0.5630451  0.7872412 -0.2170232
C     2.0838439 -1.5668929 -0.5434979
D-F   0.1749697 -1.0110751  0.8338764
```

- From this table we see that psychology students have relatively more C's,
- biology students have relatively less C's,
- psychology students have relatively less A's,

than expected under  $H_0$  (the differences are not significant though ( $p \approx 0.06$ )).

Alternatively, we can look at the **standardized residuals** using the command `z$stdres` ( $= (z$observed-z$expected)/sqrt(V)$ ), where  $V$  is the residual cell variance, see Agresti, 2007, section 2.4.5) and compare this to  $z_{\alpha/2} = qnorm(0.975) \approx 1.96$ .

# Fisher's exact test for 2x2-tables

For 2x2-tables it is possible to compute an [exact \*p\*-value](#), that does not use approximation or simulation. This is called [Fisher's exact test](#).

Data on right- and left-handed people, classified according to gender.

```
> handed=matrix(c(2780,3281,311,300),nrow=2,ncol=2,byrow=TRUE,  
+ dimnames=list(c("right-handed","other"),c("men","women")))  
> handed  
      men   women  
right-handed 2780   3281  
left-handed   311    300
```

We can compare this to picking without replacement 3091 balls from a vase which contains 6672 balls, 6061 white and 611 red. The number of white balls amongst the picked 3091 balls is  $n_{11} = 2780$ .

$n_{11}$	...	6061
...	...	611
3091	3581	6672

⇒

$n_{11}$	$6061 - n_{11}$
$3091 - n_{11}$	$3581 - (6061 - n_{11})$

The number  $n_{11}$  determines all other numbers. [Fisher's exact test](#) is based on this number. Under the null hypothesis of no dependence between the two factors it has a [hypergeometric distribution](#).

# Analysis in R: testing

```
> fisher.test(handed)
    Fisher's Exact Test for Count Data

data: handed
p-value = 0.01918
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
0.6894895 0.9688105
sample estimates:
odds ratio
0.8173619
> chisq.test(handed)
Pearson's Chi-squared test with Yates' continuity correction

data: handed
X-squared = 5.4542, df = 1, p-value = 0.01952
```

The chisquare approximation is also fine for these data. The odds ratio is computed as  $\frac{2780/311}{3281/300} = 0.8173619$  and can be interpreted as "for one right-handed women there is  $\approx 0.82$  right-handed men", there are relatively more left handed men than women.

## Recap: simple linear regression

# Setting, design and data

An experiment with a **numerical outcome**  $Y$  (dependent variable) and a **numerical explanatory variable**  $X$  (independent variable). The purpose is to explain  $Y$  by a **numerical function** of  $X$ .

**EXAMPLE** Chemical production process with outcome **total yield** and explanatory variable **temperature**.

## Design

- Fix a set of values  $X$  of the explanatory variable.
- Perform the corresponding experiments and measure the outcome  $Y$ .

It is natural to let the explanatory variable  $X$  vary over a grid of values.

**Data:**  $(X_1, Y_1), \dots, (X_n, Y_n)$ . The simple **linear regression model** assumes that

$$Y_i = \beta_0 + \beta_1 X_i + e_i, \quad i = 1, \dots, n, \quad e_1, \dots, e_n \sim N(0, \sigma^2).$$

We **test** the null hypothesis  $H_0 : \beta_1 = 0$  that the explanatory variable does *not* influence the outcome. We also want to **estimate** the parameters  $\beta_0, \beta_1$ .

The function  $x \mapsto \beta_0 + \beta_1 x$  is a line with **intercept** (value at  $x = 0$ )  $\beta_0$  and **slope** (change per unit)  $\beta_1$ . This is a simple function and may give a bad fit!

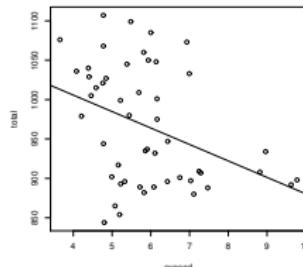
# Analysis in R: data input, graphics, estimation and testing

The column `total` of the dataset `sat.txt` is the average score on the *scolastic aptitude test* of pupils in US states in 1994/95; the column `expend` is the amount of dollars spent per pupil in the state.

```
> sat=read.table("sat.txt",header=TRUE); sat1=sat[,c(1,7)]; sat1[1:2,]  
      expend total  
Alabama    4.405 1029  
Alaska     8.963  934  
  
> sat1lm=lm(total~expend,data=sat1); summary(sat1lm)  
[ some output deleted ]  
Coefficients:  
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) 1089.294     44.390  24.539 < 2e-16 ***  
expend       -20.892      7.328  -2.851  0.00641 **
```

The parameters  $\beta_0$  and  $\beta_1$  are estimated to be 1089.294 and -20.892. The  $p$ -value for testing  $H_0 : \beta_1 = 0$  is 0.00641. The slope is significantly negative!

```
> plot(total~expend,data=sat1)  
> abline(sat1lm)
```



# Compare to Pearson's correlation test

Compare simple linear regression to Pearson's correlation test (treated earlier) which tests whether the response and explanatory variable (in our case columns total and expend) are uncorrelated.

```
> cor.test(sat1$total,sat1$expend)
```

Pearson's product-moment correlation

```
data: sat1$total and sat1$expend  
t = -2.8509, df = 48, p-value = 0.006408
```

Notice that the  $p$ -value of the correlation test between response and covariate is equal to the  $p$ -value for testing the zero slope in simple linear regression. In fact, this is the same test: **testing  $H_0 : \rho = 0$  is the same as testing  $H_0 : \beta_1 = 0$ .**

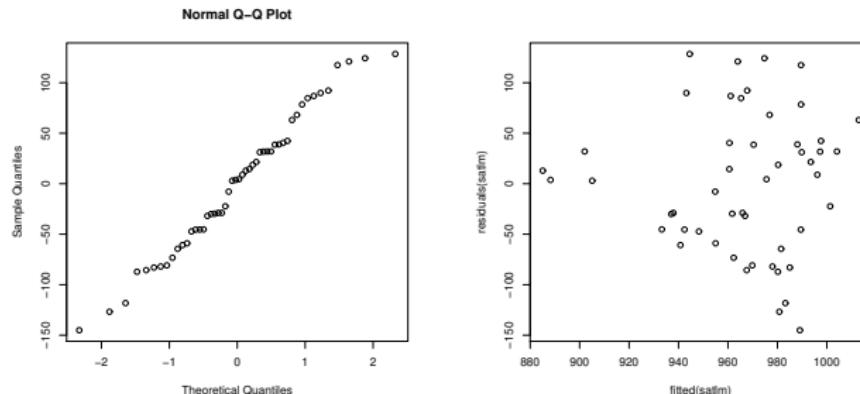
# Analysis in R: diagnostics

We can use the data to check whether the assumptions on the **errors**  $e_i = Y_i - \beta_0 - \beta_1 X_i$  are not totally untrue.

The **residuals** are  $\hat{e}_i = Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i$ ; the **fitted values**  $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i$ .

The residuals should look normal, and their spread should not vary with the fitted values.

```
> qqnorm(residuals(sat1lm))
> plot(fitted(sat1lm),residuals(sat1lm))
```



The two plots look ok.

## multiple linear regression

# Setting and design

**Setting:** an experiment with

- a **numerical outcome**  $Y$  ("dependent variable");
- $p$  **numerical explanatory variables**  $X_1, \dots, X_p$  ("independent variables", "predictors").

The purpose is to explain  $Y$  by a **numerical function** of  $X_1, \dots, X_p$ .

**EXAMPLE** Chemical production process with outcome **total yield** and explanatory variables **temperature** and **pressure**.

**EXAMPLE** Educational study with outcome **score on final exam** and explanatory variables **teacher salaries** and **number of pupils per teacher**.

**Design:**

- Fix a set of combinations  $(X_1, \dots, X_p)$  of explanatory variables.
- Perform the corresponding experiments and measure the outcome  $Y$ .

It is natural to let each explanatory variable vary over a grid and use all their possible combinations, but this may necessitate many experiments. (Regression analysis is also often used in non-experimental situations, with the explanatory variables not under control.)

# Analysis

Data  $Y_i, X_{i1}, X_{i2}, \dots, X_{ip}$ ,  $i = 1, \dots, n$ . The **linear regression model**:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \dots + \beta_p X_{ip} + e_i, \quad i = 1, \dots, n, \quad (\text{matrix notation } Y = X\beta + e)$$

where **errors**  $e_1, e_2, \dots, e_n$  are viewed as a random sample from  $N(0, \sigma^2)$ ,  
 $\beta_0, \dots, \beta_p$  are unknown population parameters.

We **test** the null hypotheses  $H_0 : \beta_j = 0$  that the  $j$ th explanatory variable does  
*not* influence the outcome for  $j = 1, \dots, p$ .

We also want to **estimate** the parameters  $\beta_j$ 's.

Possible **explanatory variables** (prediction variables):

- all  $x_j$  different  $Y = \beta_0 + \beta_1 x_1 + \dots + \beta_7 x_7 + e$ ,
- powers of  $x_j$ 's  $y = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2 + \beta_3 x_1^3 + e$ ,
- interactions between  $x_j$ 's  $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + e$ .

**Essential:** all models are linear in  $\beta_j$ 's, but not necessarily in  $x_j$ 's.

All ANOVA models can also be written in the matrix notation  $Y = X\beta + e$ , for some  
design matrix  $X$  (composed of "dummy variables"), where  $\beta$  is the vector of all the  
ANOVA coefficients involved. Thus the rest of this part also relates to all ANOVA  
models.

# Estimating parameters, SSE

To find the best parameters we minimize the sum of squared errors:

$$\min_{\beta_0, \dots, \beta_p} \sum_{i=1}^n (Y_i - \beta_0 - \beta_1 X_{i1} - \dots - \beta_p X_{ip})^2 = \sum_{i=1}^n (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i1} - \dots - \hat{\beta}_p X_{ip})^2 = RSS,$$

$\hat{\beta}_0, \dots, \hat{\beta}_p$  are the **least squares** estimates, RSS is the **Residual Sum of Squares**.

Notation:  $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \dots + \hat{\beta}_p x_{ik}$  is called **prediction/predicted response**.

The **Residual Sum of Squares** RSS (also called **Sum of Squared Errors**, SSE) and the **estimated variance** of the errors  $e_n$ :

$$RSS = SSE = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^n \hat{e}_i^2, \quad \hat{\sigma}^2 = s^2 = \frac{RSS}{n - p - 1} = \frac{SSE}{n - p - 1}.$$

$\hat{\sigma}^2$  is the **estimated variance** of the  $e_i$ 's,  $\hat{e}_i = Y_i - \hat{Y}_i$  is the  $i$ -th **residual** (the estimated error  $e_i$  of the  $i$ -th observation).

In R: `model=lm(y~x1+...+xp,data=...)`

# Coefficient of determination $R^2$

- The coefficient of determination (also called the proportion of explained variance)  $R^2$  compares the fits for the models

$$\omega: Y = \beta_0 + e \quad \text{and} \quad \Omega: Y = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p + e.$$

- For model  $\omega$ ,  $\hat{\beta}_0 = \bar{Y}$ , the fit is  $SS_y = \sum_{i=1}^n (Y_i - \bar{Y})^2$ , called total SS.
- For model  $\Omega$ , the fit is  $SSE = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$ , the residual SS.
- explained variation =  $\sum_{i=1}^n (Y_i - \bar{Y})^2 - \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$ .
- The coefficient of determination  $R^2$  is defined as

$$R^2 = \frac{SS_y - RSS}{SS_y} = \frac{\sum_{i=1}^n (Y_i - \bar{Y})^2 - \sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} = \frac{\text{explained variation}}{\text{total variation}}.$$

$0 \leq R^2 \leq 1$  because always  $SS_y \geq SSE \geq 0$ .

- $R^2$  yields a global check on the multiple linear regression model.  
The higher  $R^2$ , the more variation the model explains.
- If  $p = 1$ , then  $R^2 = r^2$  (the squared correlation between  $X_1$  and  $Y$ ).

$R^2 \approx 1$  means that the linear regression model can explain the measured response values  $Y$  very well using a linear function of the explanatory variables  $(X_1, \dots, X_p)$ .  
 $R^2 \approx 0$  means that the linear model does not explain much.

# Global model fit

- **Data:**  $X_{i1}, X_{i2}, \dots, X_{ip}, Y_i, i = 1, \dots, n$ .
- **Assumption:** the ind. errors follow a  $N(0, \sigma^2)$ -distribution.
- When is the linear regression model adequate as a whole? In linear regression we compare the models

$$\omega : Y = \beta_0 + e \quad \text{and} \quad \Omega : Y = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p + e.$$

- **Test** if  $X_1, \dots, X_p$  **together** have significant explanatory power in the model:  $H_0 : \beta_1 = \dots = \beta_p = 0$  versus  $H_1 : \text{at least one } \beta_i \neq 0$ .
- **The test statistic:**  $T = \frac{R^2/p}{(1-R^2)/(n-(p+1))} = \frac{(SS_Y - RSS)/p}{RSS/(n-(p+1))} \sim F_{p,n-(p+1)}$  under  $H_0$ . Notice that the case  $p = 1$  corresponds to Pearson's correlation test.
- The **larger  $R^2$**  (hence  $T$  is large), the more evidence against  $H_0$ , hence we reject  $H_0$  if  $T$  is large.
- The **right-sided** test: for  $T \sim F_{p,n-(p+1)}$ , reject  $H_0$  if  $p = P(T > t) < \alpha$ .
- In R: this  $p$ -value is in the last line of `summary(model)`.

# Relevance of individual coefficients

- Not all available explanatory variables may have **explanatory power**.
- From all explanatory variables, we need to find **relevant** ones by testing for **individual** coefficients.
- Test**  $H_0 : \beta_i = 0$  vs.  $H_1 : \beta_i \neq 0$  for individual  $\beta_i$ 's (usually two-sided).
- The setting and assumptions are the same as before.
- Test statistic:** under  $H_0$ ,

$$T_i = \frac{\hat{\beta}_i}{s_{\hat{\beta}_i}} \sim t_{n-(p+1)}, \quad \text{where } s_{\hat{\beta}_i}^2 = \hat{\sigma}^2 \nu_{ii}, [\nu_{ij}] = (X^T X)^{-1}, Y = X\beta + e.$$

- In R:** the estimates  $\hat{\beta}_i$ , standard errors  $s_{\hat{\beta}_i}$ , the statistics values  $T_i$  and the  $p$ -values are (in the column  $\text{Pr}(>|t|)$ ) all given in the output of `summary(model)`.
- In case  $p = 1$ , testing for  $\beta_1 = 0$  is the same as Pearson's correlation test. Thus, if  $p = 1$ , Pearson's correlation test = Global model fit test = test for  $H_0 : \beta_1 = 0$  vs.  $H_1 : \beta_1 \neq 0$ .

# Example: bodyfat data

Data of 20 individuals between 25 and 30 years old on amount of body fat, triceps skinfold thickness, thigh circumference and midarm circumference.  
Body fat is hard to measure, while the other 3 variables are easy to measure.

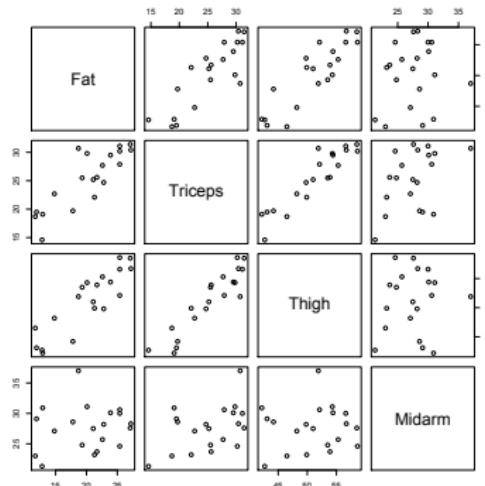
Question: can we predict Fat from the other 3 variables?

```
> bodyfat=read.table("bodyfat.txt",header=T)
> bodyfat
```

	Fat	Triceps	Thigh	Midarm
1	11.9	19.5	43.1	29.1
2	22.8	24.7	49.8	28.2
3	18.7	30.7	51.9	37.0
...				
19	14.8	22.7	48.2	27.1
20	21.1	25.2	51.0	27.5

Scatter plots of all pairs:

```
> pairs(bodyfat)
```



# Example: bodyfat data

```
> bodyfatlm=lm(Fat~Triceps+Thigh+Midarm,data=bodyfat); summary(bodyfatlm)
[some output is deleted]
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	117.085	99.782	1.173	0.258
Triceps	4.334	3.016	1.437	0.170
Thigh	-2.857	2.582	-1.106	0.285
Midarm	-2.186	1.595	-1.370	0.190

Residual standard error: 2.48 on 16 degrees of freedom

Multiple R-squared: 0.8014, Adjusted R-squared: 0.7641

F-statistic: 21.52 on 3 and 16 DF, p-value: 7.343e-06

Many things can be read from this output. The estimates  $\hat{\beta}_i$  are in the column Estimate,  $\hat{\sigma} = 2.48$  (so  $\hat{\sigma}^2 = 6.15$ ),  $s_{\hat{\beta}_i}$ 's are in the column Std. Error,  $T_i$ 's in the column t value, the p-values for individual tests  $\beta_i = 0$  are in column Pr(>|t|). The CI's for the  $\beta_i$ 's are  $\hat{\beta}_i \pm t_{\alpha/2, n-(p+1)} s_{\hat{\beta}_i}$ , obtained in R by confint(bodyfatlm). Next,  $R^2 = 0.8014$ ,  $R^2_{adj} = 0.7641$ . For testing the global model fit, statistics  $T = 21.52$ , the p-value=7.343e-06. From this output: **none** of the  $\beta_i$ 's is **individually significant**, but all **together they are significant** and explain 80%!

# Adjusted $R^2$

- We want a good fit (high  $R^2$ ) and a small number of explan. variables.
- Since more explanatory variables always explain more,  $R^2$  always increases with more variables.  $R^2$  can be found in the output of `summary(model)`.
- For  $p$  of explanatory variables in the model,  $R^2$  adjusted is

$$R_{adj}^2 = 1 - \frac{n-1}{n-(p+1)}(1 - R^2).$$

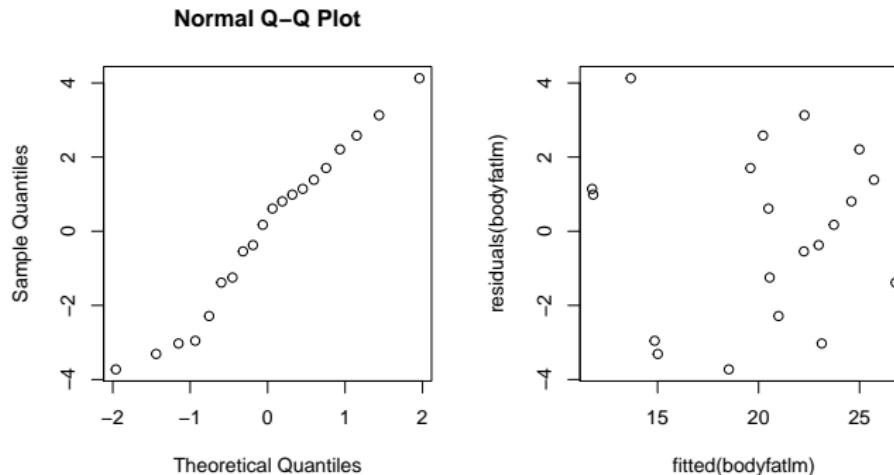
The more variables, the more conservative  $R_{adj}^2$  becomes (as compared to  $R^2$ ), it can be used to choose between models with different amounts of variables.  $R_{adj}^2$  can also be found in the output of `summary(model)`.

- The interpretation of  $R_{adj}^2$  is not fraction of explained variance anymore.

# Analysis in R: diagnostics

The **residuals**  $\hat{e}_i = Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i1} - \cdots - \hat{\beta}_p X_{ip}$  (in R: `residuals(model)`);  
the **fitted values**  $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_{i1} + \cdots + \hat{\beta}_p X_{ip}$  (in R: `fitted(model)`).

```
> qqnorm(residuals(bodyfatlm))
> plot(fitted(bodyfatlm), residuals(bodyfatlm))
```



Both plots look ok.

# If the assumptions fail?

One can consider:

- transforming the outcomes (e.g., use  $\log Y$ ,  $Y^3$ ).
- transforming the explanatory variables (e.g. use  $\log X$ ,  $X^2$ ).
- adding powers  $X_i^2, X_i^3, \dots$  of the regression variables.
- adding “interactions” like  $X_i X_j$ .
- performing nonparametric or additive regression.
- something else (there is no fix that always works).

# To finish

Today we discussed:

- contingency tables
  - chi-square test
  - Fisher test
- simple linear regression
- multiple linear regression

Next time: more on linear regression.

# Experimental Design and Data Analysis, Lecture 6

Eduard Belitser

VU Amsterdam

# Lecture overview

- 1 general factorial and incomplete block designs
- 2 random effects
- 3 crossover design
- 4 split-plot design
- 5 overview anova designs

general factorial and incomplete block designs

●○○○○

random effects

○○○

crossover design

○○○○○○○○

split-plot design

○○○○○○○○○○

overview

○○○○

## general factorial and incomplete block designs

# General factorial design

- Everything extends to an arbitrary number of factors.
- A practical difficulty is that the number of combinations of factors increases rapidly, so that many experiments are necessary.
- In the decomposition of the population means this becomes visible through many **interaction parameters**. E.g., given 3 factors there are 3 2nd order and 1 3rd order interactions:

$$\mu_{ijk} = \mu + \alpha_i + \beta_j + \gamma_k + (\alpha\beta)_{ij} + (\alpha\gamma)_{ik} + (\beta\gamma)_{jk} + (\alpha\beta\gamma)_{ijk}.$$

- It is often assumed that higher order interactions are zero. Then lower order interactions can be estimated using fewer experiments (**incomplete designs**).

# Incomplete block designs

- In a **regular block design** every treatment (the factor of interest) is applied **at least once within every block**.
- If there are many blocks (in particular if two or more block factors are crossed), then this requires many experiments.
- In an **incomplete block design** only a subset of the experiments is performed.
- It is advisable to choose this subset in a “**balanced way**”.
- Example of incomplete block design with 3 factors (1 treatment factor + 2 block factors): latin squares.

# Incomplete designs with 3 factors: latin squares

The setting for a **latin square design** for 2 block factors is an experiment with:

- a numerical outcome  $Y$  (dependent variable).
- a **factor** of interest that can be fixed at  $I$  levels (**treatment**).
- **two factors** that are *not* of interest, **both** with fixed *levels* (**blocks**).

Example of a **latin square design** for  
2 block factors, with levels 1, 2, 3, 4  
and I, II, III, IV, and a treatment  
with levels A, B, C, D

	I	II	III	IV
1	D	C	B	A
2	B	D	A	C
3	C	A	D	B
4	A	B	C	D

The outcome is measured (only) for blocks and treatment combinations  
(1,I,D), (1,II,C), (1,III,B), (1,IV,A), (2,I,B), etc.: 16 experiments in total.

**Every treatment is measured exactly once for every level of both blocks.**

The analysis assumes the **additive** model (interactions are assumed to be 0):

$$Y_{ikl} = \mu + \alpha_i + \beta_{1k} + \beta_{2l} + e_{ikl},$$

$\beta_{1k}$  and  $\beta_{2l}$  are the block effects at levels  $k \in \{1, 2, 3, 4\}$  and  $l \in \{I, II, III, IV\}$ .

```
> lm(y~block1+block2+treatment,data=...)
```

# Balanced incomplete block design

A **balanced incomplete block design** for a block factor with levels  $b_1, \dots, b_{10}$  and a treatment factor with levels A, B, C, D, E, F takes the form

	b1	b2	b3	b4	b5	b6	b7	b8	b9	b10
A		*		*			*	*		*
B	*	*				*	*			*
C	*	*	*	*	*					
D			*		*		*		*	*
E				*	*	*		*	*	
F	*		*			*		*		*

The outcome is measured (only) for the combinations marked by a “\*”: 30 experiments in total, 3 per block. **Every pair of treatments is compared within exactly 2 blocks.** The analysis is the same as for an ordinary block design.

Ideally a latin square is chosen at random from all possible latin squares, but this is computationally difficult. Instead one may apply a sequence of swaps of randomly chosen pairs of columns or rows.

Advantage of incomplete block designs: great save in experiments; disadvantage: even a rough graphical check on interactions between blocks and treatments is impossible.

general factorial and incomplete block designs  
○○○○○

random effects  
●○○

crossover design  
○○○○○○○○

split-plot design  
○○○○○○○○○○

overview  
○○○○

## random effects (mixed effects models)

# The idea of random effects

So far we have considered block effects as **fixed effects**. That is, we regard the blocks as predetermined, not as a random selection of all available blocks.

Alternatively, we can regard the blocks as a random selection of all possible blocks (the **block population**). In that case, the effects of the blocks occurring in our experiment are **random effects**.

**EXAMPLE** We want to investigate whether exam 1 is more difficult than exam 2. Because math professors may have different grading styles, resulting in different heights of the grades, we take “professor” as block factor. We randomly select 6 math professors from the math professor population. We apply a randomized block design by selecting 10 students for each professor. 5 randomly chosen students per professor make exam 1 (treatment 1) and the other 5 make exam 2 (treatment 2). The treatment effect (exam effect) is a fixed effect, whereas the block effect (professor effect) is a random effect. We are interested in the treatment effect.

# Analysis

Data ( $Y_{ibk}$ ) are assumed to follow the model

$$Y_{ibk} = \mu + \alpha_i + \tau_b + e_{ibk}, \quad i = 1, \dots, I; \quad b = 1, \dots, B; \quad k = 1, \dots, N,$$

where the treatment effect ( $\alpha_i$ ) is a **fixed effect**, and the block effect ( $\tau_b$ ) is a **random effect**. That means, we assume the block effects  $\tau_b$  form a random sample from a **centered normal distribution** (i.e., with mean 0).

As in 1-way ANOVA we **test**  $H_0 : \alpha_1 = \dots = \alpha_I = 0$ .

We also **estimate**  $\mu$  and the  $\alpha_i$ 's.

Since we have both fixed and random effects, this is called a **mixed effects model**.

general factorial and incomplete block designs  
ooooo

random effects  
ooo

crossover design  
●oooooooo

split-plot design  
ooooooooo

overview  
oooo

## mixed effects model: crossover design

# Setting and design

## Setting:

An experiment with two **numerical outcomes** per experimental unit, corresponding to two different treatments. Interest is in a possible difference between the two outcomes. An **order effect** of the outcomes is suspected.

(The crossover design can be extended to more than 2 outcomes.)

**EXAMPLE** Comparing **pain relief** by a dedicated drug or by a placebo. Both treatments are applied to every individual (with recovery time in between).

**EXAMPLE** Comparing **time needed** to complete a search task in a tree of webpages as function of the organization of the webpages. Every individual performs a search task with both types of organization.

## Design:

- Take a random sample of experimental units from the relevant population.
- Divide the units at random in two equal groups.
- Apply the treatments in one order to the units in the first group, and in the reversed order to the units in the second.

# Analysis

Data are  $2N$  measurements (on  $N$  individuals), which can be classified to belong to one of the 4 entries in the  $2 \times 2$  table.

		period	
		1	2
sequence	$T_1 T_2$	$T_1$	$T_2$
	$T_2 T_1$	$T_2$	$T_1$

The **crossover design** assumes that

$$Y_{ispbk} = \mu_{isp} + b_b + e_{ispbk},$$

where **errors** ( $e_{ispbk}$ ) and random **individual effects** ( $b_b$ ) are independent samples from centered normal distributions, and the mean values  $\mu_{isp}$  is parametrized as

		period	
		1	2
sequence	$T_1 T_2$	$\mu$	$\mu + \alpha + \beta$
	$T_2 T_1$	$\mu + \alpha + \gamma$	$\mu + \beta + \gamma$

$\alpha$  the treatment effect ( $T_2 - T_1$ ),  
 $\beta$  the learning (or period) effect,  
 $\gamma$  the sequence effect.

If the effect  $b_b$  is random, the model has the 4 mean values (over the 4 cells) and 4 parameters ( $\mu, \alpha, \beta, \gamma$ ), i.e., identifiable. For example, the parameter  $\alpha$  is found as the average  $(\mu + \alpha + \beta + \mu + \alpha + \gamma)/2$  of the two  $T_2$  treatments minus the average  $(\mu + \mu + \beta + \gamma)/2$  of the two  $T_1$  treatments in the table.

# Analysis in R: data input

The rows of the data frame `ashinal` correspond to 16 subjects and give measures of pain (for chronic headache) when treated with a drug (a) (that inhibits nitric oxide synthase) or a placebo (p). The bigger the outcome pain, the more the measured headache. One of the three columns sequence, treatment and period is redundant, but useful for the analysis.

```
> ashinal=read.table("ashinal.txt",header=TRUE); ashinal
   pain id sequence treatment period
1  -167  1      pa        a      2
2  -102  1      pa        p      1
3  -127  2      pa        a      2
[ some output deleted ]
30    3 15     ap        p      2
31   -72 16     ap        a      1
32   -36 16     ap        p      2
```

# Analysis in R: fixed effects (1)

```
> ashinal$id=factor(ashinal$id); ashinal$period=factor(ashinal$period)
> ashinalm=lm(pain~treatment+period+id,data=ashinal)
> anova(ashinalm)

Df Sum Sq Mean Sq F value    Pr(>F)
treatment   1  14706  14706.1 10.4624 0.005994 **
period      1    1505   1505.2  1.0709 0.318298
id          15  51137   3409.2  2.4254 0.052870 .
Residuals  14  19679   1405.6
```

If factor id enters the model, we have 5 parameters for 4 groups and the parameters become **unidentifiable**. The sequence effect is therefore left out, as it cannot then be estimated in a fixed effects model. In the mixed effects model this is possible.

In general (e.g., for unbalanced designs), changing the order of factors in the anova formula gives **different p-values**, as anova performs “sequential tests”. To obtain the correct p-value the factor of interest (treatment), put this factor **last in the formula**.

```
> anova(lm(pain~id+period+treatment,data=ashinal)) # treatment last!

Df Sum Sq Mean Sq F value    Pr(>F)
id          15  51137   3409.2  2.4254 0.05287 .
period      1    4608   4608.0  3.2783 0.09171 .
treatment   1   11603  11603.3  8.2550 0.01228 *
Residuals  14  19679   1405.6
```

## Analysis in R: fixed effects (2)

```
> summary(ashinalm)
[ some output deleted ]
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -147.08     28.63  -5.137 0.000151 ***
treatmentp    39.33     13.69   2.873 0.012276 *
period2      -14.17     13.69  -1.035 0.318298
id2           51.50     37.49   1.374 0.191150
id3           121.50    37.49   3.241 0.005921 **
[ some output deleted ]
id16          80.50     37.49   2.147 0.049781 *
```

The active drug gives 39.33 more pain relief (recall the treatment parameterization p is compared to a). There is no significant learning (=period) effect.

The “fixed effects” analysis given here is not the correct implementation of the model assumptions. The “**mixed effects**” ought to be used instead with id as **random factor**.

In this case however, the difference between the incorrect “fixed effects” analysis and the correct “mixed effects” analysis on the next slide is minor.

# Analysis in R: mixed effects (1)

```
> library(lme4); attach(ashinal)
> ashinalmer=lmer(pain~treatment+sequence+period+(1|id),
+ REML=FALSE,data=ashinal); summary(ashinalmer)
[ some output deleted ]
```

Random effects:

Groups	Name	Variance	Std.Dev.
id	(Intercept)	755.91	27.494
	Residual	1229.92	35.070

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	-28.50	18.19	-1.567
treatmentp	39.33	12.81	3.071
sequencepa	-31.13	19.12	-1.628
period2	-14.17	12.81	-1.106

The R-library lme4 implements the mixed effects models, another library is nlml. The function lmer gives the correct implementation of the crossover design, with the individuals as “random effects”. The number 755.91 under Random effects is the estimated variance of the normal population of the “individual effects” ( $b_n$ ). The estimated treatment and period effects under Fixed effects are identical to those in the previous slide. The model:  $Y_{ispbn} = \mu + \alpha_i + \beta_s + \gamma_p + b_b + e_{ispbn}$ .

## Analysis in R: mixed effects (2)

```
> ashinalmer1=lmer(pain~sequence+period+(1|id),data=ashinal,REML=FALSE)
> anova(ashinalmer1,ashinalmer) # test reduced model inside full model
Models:
ashinalmer1: pain ~ sequence + period + (1 | id)
ashinalmer: pain ~ treatment + sequence + period + (1 | id)
      npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
ashinalmer1     5 348.72 356.05 -169.36    338.72
ashinalmer     6 343.31 352.10 -165.65    331.31 7.4161   1  0.006464 **
```

The function `lmer` does not automatically produce *p*-values (and they cannot be extracted by `anova(ashinalmer)`), but these can be found by refitting the model without the effect of interest (in our case `treatment`), and applying `anova` with 2 arguments (to test the fit of the reduced model without `treatment` inside the full model). Factor `treatment` has a significant effect.

Notation: 1 in `(1|id)` means the random effect `id` is with respect to the intercept. Note that within this mixed effects model it is also possible to estimate the sequence effect.

general factorial and incomplete block designs  
○○○○○

random effects  
○○○

crossover design  
○○○○○○○○

split-plot design  
●○○○○○○○○

overview  
○○○○

## mixed effects model: split-plot design

# Setting and design

**Setting:** an experiment with a numerical outcome  $Y$ ,

- a treatment factor with  $I$  levels that is difficult to apply or randomize,
- a treatment factor with  $J$  levels that is easy to apply or randomize.
- possibly a block factor.

Interest is as in a two-way factorial experiment.

The experimental units are grouped as subplots of whole plots; the levels of the first, outer factor are randomized over the groups (whole plots), whereas the levels of the second, inner factor are randomized over the subplots. The experiment may be repeated within the levels of a block variable.

**Design:** for each of the  $B$  levels of the block factor

- Select  $I$  groups of  $NJ$  experimental units randomly from the population.
- Randomize the  $I$  levels of the ("difficult") outer factor over the  $I$  groups.
- Within every group randomize the  $J$  levels of the ("easy") inner factor over the  $NJ$  units in the group.
- Perform the experiment  $NIJ$  times independently.

Instead of "outer" one says "whole plot" and instead of "inner" one says "subplot".

# Examples

**EXAMPLE** To study the yield of 4 varieties of a crop under 3 varieties of fertilizer a large field is subdivided into 3 **whole plots**, which are subdivided into 8 **subplots**. The 3 levels of fertilizers are randomized over the 3 whole plots; in each whole plot the 4 varieties are randomized over the 8 subplots. The motivation is that it is hard to apply fertilizer to small, contiguous plots. The experiment is replicated on 2 other fields which serve as blocks. It is suspected that **fertilizer influences the yield**, i.e., the yields within the same whole plot share more similarity than the yields from different whole plots.

**EXAMPLE** An experiment to study reaction time to 3 types of stimuli is run in two different experimental set-ups (e.g. room lay-out, furnishings, electronic equipment). Because it is time-consuming to change the set-ups, the experiment is run 6 times, 3 times with both set-ups, in random order, and in each run 18 subjects are randomized to the 3 types of stimuli. It is suspected that measurements within one of the 6 runs share some uncontrolled variables (day of the week, the weather, the experimenter,etc.), more than measurements from different runs.

# Analysis

The **split-plot design** assumes that the measurement  $Y_{ijbk}$  at levels  $i$  and  $j$  of the outer and inner factors, of the  $k$ th replicate in the  $b$ th block, satisfies

$$Y_{ijbk} = \mu_{ij} + b_b + c_{ib} + e_{ijbk}, \quad i = 1, \dots, I; j = 1, \dots, J; b = 1, \dots, B; k = 1, \dots, N.$$

for **errors** ( $e_{ijbk}$ ), **block effects** ( $b_b$ ) and **block-whole plot interactions**  $c_{ib}$  that are independent random samples from centered **normal** distributions.

- The variables  $b_b$  model dependence between the measurements within blocks.
- The variables  $c_{ib}$  model (further) dependence within the groups of experimental units (= “whole plots”) within blocks that receive the same treatment of the outer factor.

As in a **two-way lay-out** the means  $\mu_{ij}$  can be decomposed in main and interaction effects as

$$\mu_{ij} = \mu + \alpha_i + \beta_j + \gamma_{ij}.$$

The same tests and estimates as in 2-way ANOVA are of interest.

# Analysis in R: data input

At two farms (= block) a field was subdivided in 3 parts (= whole plot) and the (outer) factor spray was independently randomized over the 3 whole plots. Next, each of the  $3 \times 2 = 6$  whole plots was subdivided in 2 subplots and within every whole plot the (inner) factor variety was randomized over the 2 subplots. Little of this description can be seen from the data matrix wheat.

```
> wheat
```

	farm	yield	spray	variety
1	f1	56	2	2
2	f1	64	2	1
3	f1	71	1	1
4	f1	66	1	2
5	f1	84	3	1
6	f1	82	3	2
7	f2	88	3	2
8	f2	97	3	1
9	f2	79	1	2
10	f2	83	1	1
11	f2	77	2	1
12	f2	73	2	2

# Analysis in R: fixed effects (1)

```
> wheat$spray=factor(wheat$spray); wheat$variety=factor(wheat$variety)
> wheatlm=lm(yield~spray*variety+farm+farm:spray ,data=wheat)
> anova(wheatlm)
      Df Sum Sq Mean Sq F value    Pr(>F)
spray        2  842.17   421.08 76.5606 0.002664 ***
variety      1   85.33    85.33 15.5152 0.029157 *
farm         1  456.33   456.33 82.9697 0.002796 ***
spray:variety 2    1.17     0.58  0.1061 0.902597
spray:farm    2   15.17     7.58  1.3788 0.376117
Residuals     3   16.50     5.50
```

Interest is in the main and interaction effects of the outer and inner factor. Main effects for spray and variety are significant, whereas interaction effects between these two are not. Here, the model is **three-way ANOVA** with some interactions included:  $Y_{ijbk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + b_b + c_{ib} + e_{ijbk}$ .

## Analysis in R: fixed effects (2)

```
> summary(wheatlm)
[ some output deleted ]
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    70.750     2.031  34.835 5.2e-05 ***
spray2        -7.750     2.872  -2.698  0.0739 .
spray3       15.000     2.872   5.222  0.0137 *
variety2      -4.500     2.345  -1.919  0.1508
farmf2        12.500     2.345   5.330  0.0129 *
spray2:variety2 -1.500     3.317  -0.452  0.6818
spray3:variety2 -1.000     3.317  -0.302  0.7827
spray2:farmf2    2.500     3.317   0.754  0.5057
spray3:farmf2   -3.000     3.317  -0.905  0.4324
```

This “fixed effects” analysis is nowadays considered old-fashioned, and preference is for the **mixed effects** analysis on the next slide.

# Analysis in R: mixed effects (1)

```
> wheatlmer=lmer(yield~spray*variety+(1|farm)+(1|farm:spray),  
+ data=wheat,REML=FALSE); summary(wheatlmer)  
[ some output deleted ]
```

Random effects:

Groups	Name	Variance	Std.Dev.
farm:spray	(Intercept)	0.52083	0.72169
farm	(Intercept)	37.39584	6.11521
Residual		2.75000	1.65831

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	77.000	4.509	17.076
spray2	-6.500	1.808	-3.594
spray3	13.500	1.808	7.465
variety2	-4.500	1.658	-2.714
spray2:variety2	-1.500	2.345	-0.640
spray3:variety2	-1.000	2.345	-0.426

Recall the model:  $Y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + b_b + c_{ib} + e_{ijk}$ . The estimates and  $p$ -values for the effects of spray and variety (under Fixed effects) are a bit different from the previous slide. The column Variance under random effects gives estimates 0.52, 37.39 and 2.75 for the variances of the (normal) populations of the  $c_{ib}$ ,  $b_b$  and  $e_{ijk}$  in the model.

## Analysis in R: mixed effects (2)

```
> wheatlmer1=lmer(yield~spray+(1|farm)+(1|farm:spray),data=wheat,REML=FALSE)
> anova(wheatlmer1,wheatlmer)
Models:
wheatlmer1: yield ~ spray + (1 | farm) + (1 | farm:spray)
wheatlmer: yield ~ spray * variety + (1 | farm) + (1 | farm:spray)
          npar    AIC    BIC  logLik deviance   Chisq Df Pr(>Chisq)
wheatlmer1     6 81.610 84.519 -34.805    69.610
wheatlmer      9 74.316 78.680 -28.158    56.316 13.294    3   0.004042 **
```

Recall that we cannot directly run `anova(wheatlmer)` to test for any factor of interest. We need to create a model without that factor and test that model inside the full one. For example, to test the effect of the factor variety we fit the mixed effects model again, now without this factor in `wheatlmer1` and test by `anova` its fit within the full model `wheatlmer`. The significance of the difference in the models is computed, which is the effect of the factor variety. It appears that the effect of variety is significant.

general factorial and incomplete block designs  
ooooo

random effects  
ooo

crossover design  
oooooooo

split-plot design  
ooooooooo

overview  
●ooo

## overview anova designs so far

# Overview designs so far (1)

- 1-way anova (completely randomized)
  - **Design:** select NI units simultaneously.
  - **Model:**  $Y_{ik} = \mu + \alpha_i + e_{ik}$ , fixed effects.
- 2-way anova (completely randomized)
  - **Design:** select NIJ units simultaneously.
  - **Model:**  $Y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + e_{ijk}$ , fixed effects.
- Randomized block design
  - **Design:** select NI units from each block population.
  - **Model:**  $Y_{ibk} = \mu + \alpha_i + \beta_b + e_{ibk}$ , fixed effects, **no interactions**.
- Repeated measures
  - **Design:** select B units (which serve as block).
  - **Model:**  $Y_{ib} = \mu + \alpha_i + \beta_b + e_{ib}$ , fixed effects, **no interactions**.

## Overview designs so far (2)

- General factorial and incomplete block designs
- Mixed effects: crossover design (2 fixed effects + 1 random ind. effect)
  - **Design:** select N units and divide in two "sequence" groups.
  - **Model:**  $Y_{ispbk} = \mu + \alpha_i + \beta_s + \gamma_p + b_b + e_{ispbk}$ , mixed effects with fixed treatment effect  $\alpha$ , fixed period effect  $\beta$ , fixed sequence effect  $\gamma$  (but effectively 2 fixed effects), and one random individual effect  $b_b$ .
- Mixed effects: split-plot design (2 treatments + one random block factor + random interaction with one treatment)
  - **Design:** for each block select I (outer) groups of size NJ units from the block population.
  - **Model:**  $Y_{ijbk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + b_b + c_{ib} + e_{ijbk}$ , mixed effects with random block effects  $b_b$  and random block-whole plot interactions ( $c_{ib}$ ) and fixed main effects ( $\alpha_i$ ,  $\beta_j$ ) and fixed interaction effects ( $\gamma_{ij}$ ).

# To finish

Today we discussed:

- general factorial and incomplete block designs
- random effects
- crossover design
- split-plot design
- overview anova designs

Next time: contingency tables, linear regression.

# Experimental Design and Data Analysis, Lecture 8

Eduard Belitser

VU Amsterdam

# Lecture overview

- ① strategies to choose the variables (lasso method in the next lecture)
  - step up
  - step down
- ② diagnostics in linear regression
- ③ problems in linear regression
  - outliers and influence points
  - collinearity

strategies to choose the variables

●oooooooo

prediction

ooooo

diagnostics in linear regression

oooooo

outliers and influence points

ooooo

collinearity

ooooooo

## strategies to choose the variables

# Strategies to choose the variables

An important issue in multiple linear regression is how to find a suitable model. That is, how to select explanatory variables  $X_1, \dots, X_p$  such that

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip} + e_i, \quad i = 1, \dots, n,$$

is a **good model** for the given data.

A good model should be as precise and as concise as possible. It should

- contain all explanatory variables  $X_j$  that are essential in explaining  $Y$
- not contain any variable  $X_j$  that does not contribute significantly.

Common strategies to build a model are:

- **step-up**
- **step-down**
- **lasso** (next lecture)

The **coefficient of determination**  $R^2 \in [0, 1]$  yields a global check on the linear regression model. The higher  $R^2$  the more variation the model explains.

# Two strategies for finding a good model

In practice we need a strategy for building a model. We consider **two strategies**.

The **step up** method:

1. start with the background model  $Y = \beta_0 + e$ ;
2. take the variable (that is not in the model) that yields the maximum increase in  $R^2$ ;
3. if this variable is significant ( $t$ -test) add it to the model and go to step 2, otherwise stop.

The **step down** method:

1. start with the full model  $Y = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p + e$ ;
2. test all variables by using the  $t$ -test;
3. if the largest  $p$ -value is larger than 0.05, remove the corresponding variable and go back to step 2.

# Step up (1)

We apply the **step up** strategy to the bodyfat data:

```
> summary(lm(Fat~Triceps))
    Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.4961     3.3192  -0.451   0.658
Triceps      0.8572     0.1288   6.656 3.02e-06 ***
```

Multiple R-squared: 0.7111

```
> summary(lm(Fat~Thigh))
    Estimate Std. Error t value Pr(>|t|)
(Intercept) -23.6345    5.6574  -4.178 0.000566 ***
Thigh        0.8565    0.1100   7.786 3.6e-07 ***
```

Multiple R-squared: 0.771

```
> summary(lm(Fat~Midarm))
    Estimate Std. Error t value Pr(>|t|)
(Intercept) 14.6868    9.0959   1.615   0.124
Midarm      0.1994    0.3266   0.611   0.549
```

Multiple R-squared: 0.02029

Thus, the **first variable to add** is Thigh.

## Step up (2)

The second step:

```
> summary(lm(Fat~Thigh+Triceps))  
             Estimate Std. Error t value Pr(>|t|)  
(Intercept) -19.1742     8.3606  -2.293   0.0348 *  
Thigh         0.6594     0.2912   2.265   0.0369 *  
Triceps       0.2224     0.3034   0.733   0.4737
```

Multiple R-squared: 0.7781

```
> summary(lm(Fat~Thigh+Midarm))  
             Estimate Std. Error t value Pr(>|t|)  
(Intercept) -25.99695    6.99732  -3.715  0.00172 **  
Thigh         0.85088    0.11245   7.567 7.72e-07 ***  
Midarm        0.09603    0.16139   0.595  0.55968
```

Multiple R-squared: 0.7757

Resulting model:  $\text{Fat} = -23.6345 + 0.8565 \cdot \text{Thigh} + \text{error}$ , with  $R^2 = 0.771$ .

# Step down (1)

We now apply the **step down** strategy to the bodyfat data:

```
> summary(lm(Fat~Triceps+Thigh+Midarm))  
Estimate Std. Error t value Pr(>|t|)  
(Intercept) 117.085 99.782 1.173 0.258  
Triceps 4.334 3.016 1.437 0.170  
Thigh -2.857 2.582 -1.106 0.285  
Midarm -2.186 1.595 -1.370 0.190
```

Multiple R-squared: 0.8014

We see that none of the variables is significant. The **first variable to remove** is Thigh, which has the highest  $p$ -value.

## Step down (2)

The second step:

```
> summary(lm(Fat~Triceps+Midarm))
    Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.7916     4.4883   1.513   0.1486
Triceps      1.0006     0.1282   7.803 5.12e-07 ***
Midarm     -0.4314     0.1766  -2.443   0.0258 *

```

Multiple R-squared: 0.7862

All remaining variables are significant.

Resulting model: **Fat = 6.7916 + 1.0006\*Triceps -0.4314\*Midarm + error** with  $R^2 = 0.7862$ .

# Step up or step down?

Now we are left with two different models.

Model 1 with  $R^2 = 0.771$  and  $\hat{\sigma} = 2.51$ :

$$\text{Fat} = -23.6345 + 0.8565 \cdot \text{Thigh} + \text{error}$$

Model 2 with  $R^2 = 0.7862$  and  $\hat{\sigma} = 2.496$ :

$$\text{Fat} = 6.7916 + 1.0006 \cdot \text{Triceps} - 0.4314 \cdot \text{Midarm} + \text{error}$$

**Question:** which one do we prefer, and why?

**Answer:** Model 1 is preferred, because it has **less variables** and a **comparable value of  $R^2$** . Also the term **-0.4314\*Midarm** in the second model is not well interpretable.

Remember that one needs to **check the model assumptions** for the resulting model.

strategies to choose the variables  
oooooooo

**prediction**  
●oooo

diagnostics in linear regression  
oooooo

outliers and influence points  
ooooo

collinearity  
ooooooo

## **prediction**

# The predicted value

For the  $x$ -values in the data set, the **fitted (predicted) values** are

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_{1i} + \dots + \hat{\beta}_p X_{pi}, \quad i = 1, \dots, n.$$

The  $\hat{Y}_i$ 's can be computed by the R-command `fitted(model)`. Notice that these are in general **different** from the observed  $Y_i$ 's.

```
> fitted(bodyfatlm)
      1       2       3       4       5
14.85499 20.21884 20.98668 23.12732 11.75761
...
      16      17      18      19      20
23.72747 22.97360 26.78590 18.52628 20.48791
```

Once all  $\hat{\beta}_i$ 's are computed, one can **predict** the  $y$ -value for a **new** measurement of the  $p$  explanatory variables:  $x_{new} = (x_{new,1}, \dots, x_{new,p})$  as

$$\hat{Y}_{new} = \hat{\beta}_0 + \hat{\beta}_1 x_{new,1} + \dots + \hat{\beta}_p x_{new,p}.$$

This is done by the command `predict(model, newxdata, ...)`.

# Confidence and prediction intervals

We can also construct two types of intervals for  $\hat{Y}_{new}$  for given  $x_{new}$ -values:

- confidence interval for  $Y_{new}$ : an interval for the mean  $Y_{new}$ -value for given  $x_{new}$ -values. (This is interval  $x_{new}^T \hat{\beta} \pm t_{\alpha/2, n-(p+1)} \sqrt{s^2(x_{new}^T(X^T X)^{-1} x_{new})}$ .)
- prediction interval for  $Y_{new}$ : an interval for an individual  $Y_{new}$ -observation for given  $x_{new}$ -values. **This interval is larger** as the error is also taken into account. (This is interval  $x_{new}^T \hat{\beta} \pm t_{\alpha/2, n-(p+1)} \sqrt{s^2(1 + x_{new}^T(X^T X)^{-1} x_{new})}$ .)

To summarize, **confidence** is for the population mean, **prediction** is for an individual observation.

In R: `predict(lm(y~x1+...+xk), newxdata, interval=..., level=...)`

# Example: bodyfat data

Prediction intervals for the body fat data for new data can be found by

- designing a `data.frame` with the new  $x$ -values
- applying `predict` to this `data.frame` and specify the type of interval.

```
> newxdata=data.frame(Triceps=24.5,Thigh=51.3,Midarm=28.7)
> predict(bodyfatlm,newxdata)
13.97372
> predict(bodyfatlm,newxdata,interval="prediction")
      fit      lwr      upr
13.97372 3.053481 24.89396
> predict(bodyfatlm,newxdata,interval="prediction",level=0.95)
      fit      lwr      upr
13.97372 3.053481 24.89396
> predict(bodyfatlm,newxdata,interval="confidence")
      fit      lwr      upr
13.97372 4.402296 23.54515
```

The prediction interval is indeed larger!

# Discussion

Finding different models by different strategies is exemplary for linear regression: **there is no golden strategy to resolve this.**

In such a case one should compare

- $R^2$  values of both models (higher is better),
- plots of fitted values versus residuals of both plots (should be no specific structure),
- the number of explanatory variables in both models (fewer is better),
- the character of the explanatory variables in both models (easy to measure?),
- interpretation of both models,
- ...

and choose the one that is most appropriate.

strategies to choose the variables  
oooooooo

prediction  
ooooo

diagnostics in linear regression  
●ooooo

outliers and influence points  
ooooo

collinearity  
oooooooo

## diagnostics in linear regression

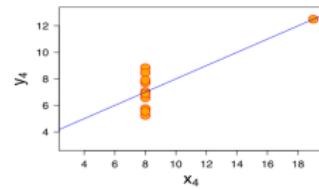
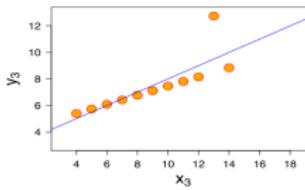
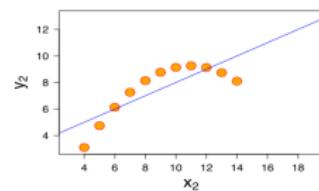
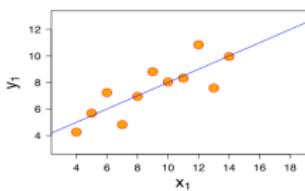
# Example

Checking the fit in the linear regression by looking at the (adjusted)  $R^2$  is not sufficient, we need to check the **model assumptions**: the **linearity of the relation** and the **normality** of the errors. We consider both **graphical** and **numerical** tools.

In the following 4 examples of artificial data, the fitted model is  
 $y = 3.0 + 0.5*x + \text{error}$ ,  $\hat{\sigma}^2 = 1.5$  and  $R^2 = 0.67$ .

The differences between the 4 situations illustrate the need for a **diagnostic tool**, apart from  $R^2$ ,  $\hat{\sigma}$ .

- 1 The first looks ok.
- 2 No lin. relation between  $X$ ,  $Y$ .
- 3 Outlying point in  $Y$ .
- 4 Only one  $X$  is different.



# Diagnostic plots

To check the model quality look at

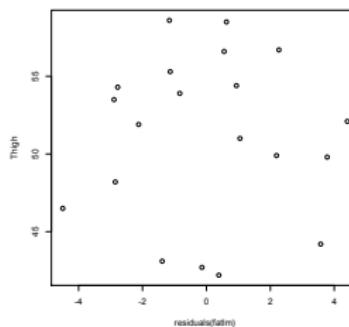
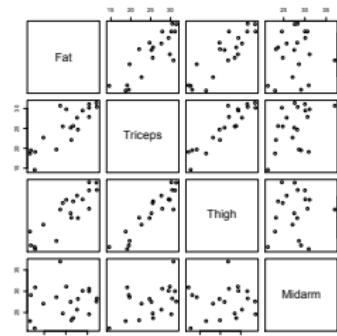
1. scatter plot: plot  $Y$  against each  $X_k$  separately (this yields overall picture, and shows outlying values)
2. scatter plot: plot residuals against each  $X_k$  in the model separately (look at pattern (curved?) and spread)
3. added variable plot (partial regression plot, see Velleman and Welsch (1981)): plot residuals of  $X_j$  against residuals of  $Y$  with omitted  $X_j$  (to show the effect of adding  $X_j$  to the model.) (Or, to show the relationship between  $Y$  and  $X_j$ , once all other predictors have been accounted for.)
4. scatter plot: plot residuals against each  $X_k$  not in the model separately (look at pattern — linear? then include!)
5. scatter plot: plot residuals against  $Y$  and  $\hat{Y}$  (look at spread)
6. normal QQ-plot of the residuals (check normality assumption)

# Example: bodyfat data (1)

Read in the data.

```
>bodyfat=read.table("bodyfat.txt",header=T)  
>attach(bodyfat)
```

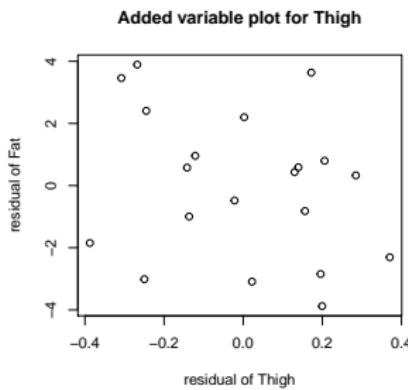
1. Scatter plot of  $Y$  against each  $X_k$  separately.  
> pairs(bodyfat)
2. Scatter plot of residuals against each  $X_k$  in the model separately.  
> bodyfatlm=lm(Fat~Thigh)  
> plot(residuals(bodyfatlm), Thigh)  
  
If a curved pattern is visible, include, e.g.,  $X_j^2$  or transform  $X_j$  (e.g.,  $\log(X_j)$ ,  $\sqrt{X_j}$ ).



## Example: bodyfat data (2)

3. Added variable plot of residuals of  $X_j$  against residuals of  $Y$  with omitted  $X_j$ .

```
> x=residuals(lm(Thigh~Midarm+Triceps))
> y=residuals(lm(Fat~Midarm+Triceps))
> plot(x,y,main="Added variable plot for
+ Thigh", xlab="residual of Thigh",
+ ylab="residual of Fat")
```

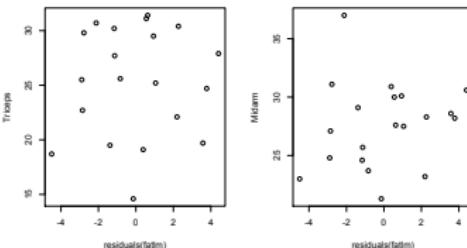


The slope in this plot is the regression coefficients  $\beta_j$  from the original multiple regression model, and the residuals in this plot are precisely the residuals from the original multiple regression. Outliers and heteroskedasticity (caused by  $X_j$ ) can be identified by looking at the plot of this simple rather than multiple regression model. All the added variable plots can be obtained from the full model mod by the command `avPlots(mod)` from the package car.

## Example: bodyfat data (3)

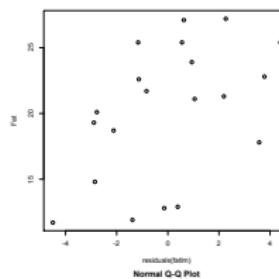
- Scatter plot of residuals against each  $X_k$  not in the model separately.

```
> plot(residuals(bodyfatlm),Triceps)  
> plot(residuals(bodyfatlm),Midarm)
```



- Scatter plot of residuals against  $Y$  (and  $\hat{Y}$ ).

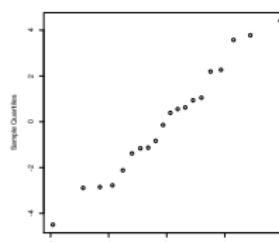
```
> plot(residuals(bodyfatlm),Fat)  
> plot(residuals(bodyfatlm),fitted(bodyfatlm))
```



- Normal Q-Q-plot of the residuals.

```
> qqnorm(residuals(bodyfatlm))
```

Also: `shapiro.test(residuals(bodyfatlm))`. If residuals are not normally distributed, go back to scatter plots and start with different model, possibly apply transforms.



strategies to choose the variables  
oooooooo

prediction  
ooooo

diagnostics in linear regression  
oooooo

outliers and influence points  
●oooo

collinearity  
ooooooo

## outliers and influence points

# Outliers, leverage points, influence points

- An **outlier** is an observation with an extremely high or low **response value**, compared to what is expected under the model.
- A **leverage (or potential) point** is an observation with an outlying value in the **explanatory variable**.
- To study the effect of a leverage point one can fit the model **with** and **without** that data point. If the estimated parameters change drastically by deleting the leverage point, the observation is called an **influence point**.
- The **Cook's distance**  $D_i$  quantifies the influence of observation  $i$  on the predictions:

$$D_i = \frac{1}{(p+1)\hat{\sigma}^2} \sum_{j=1}^n (\hat{Y}_{(i),j} - \hat{Y}_j)^2,$$

with  $\hat{Y}_{(i),j}$  the predicted  $j$ -th response based on the model **without** the  $i$ -th data point.

- **Rule of thumb:** if the Cook's distance for some data point is larger than 1, it is considered to be an influence point.

# Outlier: Forbes' data

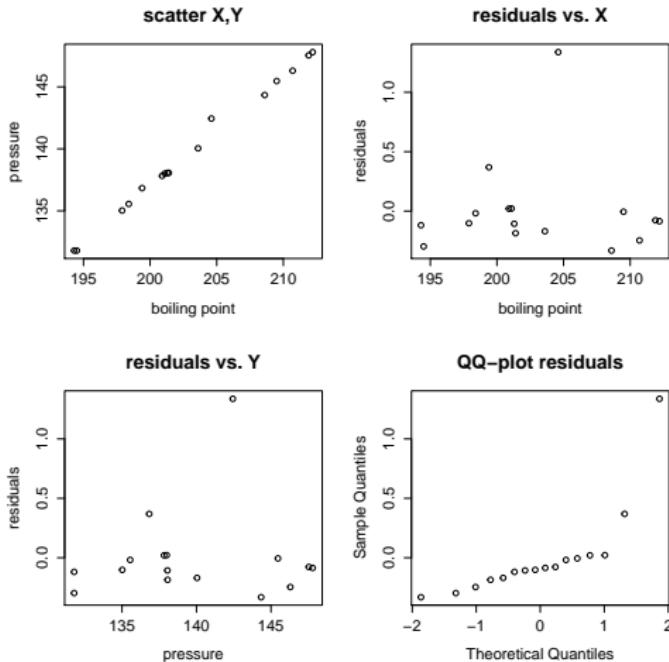
An **outlier** is an observation with an extremely high or low response value, compared to what is expected under the model.

Consider **Forbes' data** (in file `forbes.txt`) which describe the relation between boiling point of water and pressure.

```
> x=forbes[,2];y=forbes[,3]  
> forbeslm=lm(y~x)
```

One outlier point “spoils” all the plots, its value deviates too much from what it is expected under the model.

Residuals are for the simple linear regression model.



# Outlier: Forbes' data

```
> order(abs(residuals(forbeslm)))
[1] 12  4  6  7 15 16  3  9  2 10  8 14  1 13  5 11
```

The 11-th data point seems to be an outlier. The command `order(abs(residuals(model)))` gives the indices of the ordered absolute values of residuals from smallest to largest.

The **mean shift outlier model** can be applied to test whether the  $k$ -th point significantly deviates from the other points in a linear regression setting.

```
> u11=rep(0,16); u11[11]=1; u11
[1] 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
> forbeslm11=lm(y~x+u11); summary(forbeslm11)
...
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -40.787278   1.530216 -26.655 9.87e-13 ***
x             0.888534   0.007533 117.950 < 2e-16 ***
u11          1.433143   0.177565   8.071 2.03e-06 ***
...

```

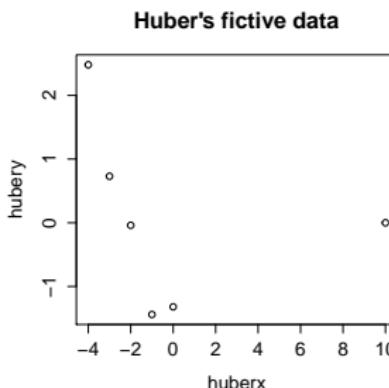
Since the coefficient for explanatory variable `u11` is significantly different from 0, the outlier is **significant**.

# Leverage/influence points: Huber's data

Consider Huber's fictive data.

**Question:** what is the influence of the observation with value  $x=10$  of the explanatory variable?

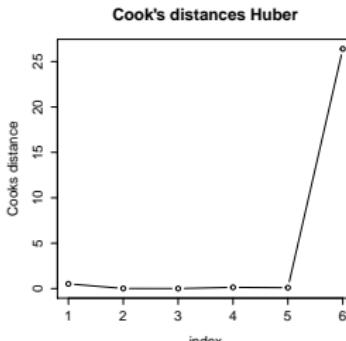
```
> xh = c(-4:0,10)
> yh = c(2.48,.73,-.04,-1.44,-1.32,0)
> huberlm = lm(yh ~ xh)
```



Compute and plot the Cook's distances.

```
> round(cooks.distance(huberlm),2)
    1     2     3     4     5     6
  0.52   0.01   0.00   0.13   0.10  26.40
> plot(1:6,cooks.distance(huberlm),type="b")
```

Here we see an influence point: the Cook's distance is **26.40** for the leverage point.



strategies to choose the variables  
oooooooo

prediction  
ooooo

diagnostics in linear regression  
oooooo

outliers and influence points  
ooooo

collinearity  
●oooooo

## collinearity

# Collinearity

**Collinearity** is the problem of **linear relations** between explanatory variables. A straight line in a scatter plot of two variables means they explain the same.

**Example.** Suppose we have a response variable  $Y$  and one explanatory variable  $X_1$ . Now we add a second explanatory variable  $X_2 = 2X_1$ . Can we do a meaningful analysis using the model  $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + e$ ? No, in this model we cannot uniquely estimate  $\beta_1$  and  $\beta_2$ , because

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + e = \beta_0 + (\beta_1 + 2\beta_2) X_1 + e$$

and only the sum  $\beta'_1 = \beta_1 + 2\beta_2$  is estimable. There are many choices  $\beta_1$  and  $\beta_2$  giving the same  $\beta'_1 = \beta_1 + 2\beta_2$  (e.g.,  $1 = \beta'_1 = 0 + 2 \cdot 0.5 = 1 + 2 \cdot 0$ ).

If  $X_1$  and  $X_2$  are close to **collinear** then  $\beta_1$  and  $\beta_2$  are difficult to estimate. This is reflected in **large variances** and **large confidence intervals** of  $\hat{\beta}_1$  and  $\hat{\beta}_2$ .

If the confidence interval of  $\hat{\beta}_j$  is large, the **estimate is not reliable**.

We can have collinearity amongst a set of more than two explanatory variables (multicollinearity).

# Ways to investigate and remove collinearity

Graphical ways to investigate collinearity:

- scatter plot of  $X_i$  against  $X_j$  for all  $i, j$  (only pairwise collinearities visible).

Numerical way to investigate collinearity:

- pairwise linear correlation of  $X_i$  and  $X_j$  for all combinations  $i, j$ .
- variance inflation factor of  $\beta_j$  for all  $j$  (check whether these are high).

There are more advanced numerical ways to investigate collinearity (special packages in R like `car`), e.g.: condition indices, variance decomposition.

When there is collinearity amongst the explan. variables  $X_1, \dots, X_p$  one should

- avoid having two collinear explanatory variables in the model
- choose a model with a small number of explanatory variables
- choose a model that intuitively/practically makes sense

Recognizing multicollinearity among a set of explanatory variables is not necessarily easy. For pairwise collinearity, we can simply examine the scatterplots or the correlations between the variables, but we may miss more subtle forms of multicollinearity.

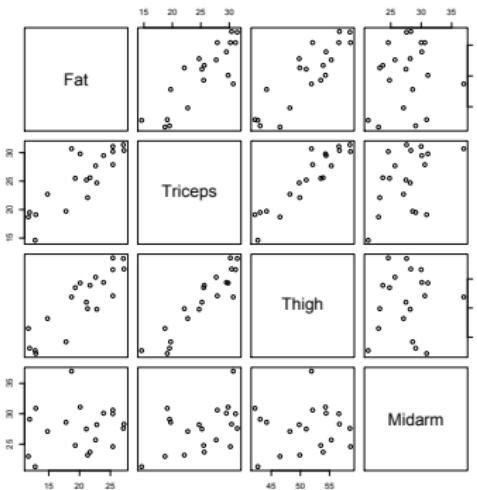
# Example: bodyfat data

Apply these checks to the bodyfat data:

```
> round(cor(bodyfat),2)
      Fat Triceps Thigh Midarm
Fat     1.00    0.84   0.88   0.14
Triceps 0.84    1.00   0.92   0.46
Thigh    0.88   0.92    1.00   0.08
Midarm   0.14   0.46   0.08    1.00
```

```
> pairs(bodyfat)
```

Clearly Triceps and Thigh are collinear, both from the plot and from the correlation value of 0.92.



# Variance inflation factor

A more useful approach is to examine the **variance inflation factors** (VIF) of the explanatory variables. The VIF for the  $j$ -th independent variable is given by

$$VIF_j = \frac{1}{1 - R_j^2}, \quad j = 1, \dots, k,$$

where  $R_j^2$  the determination coefficient  $R^2$  from the regression of the  $j$ -th explanatory  $X_j$  (as response) variable on the remaining explanatory variables.

The VIF of an explanatory variable indicates the strength of the linear relationship between the variable  $X_j$  and the remaining explanatory variables.

**Rule of thumb:**  $VIF_j$ 's larger than 5 (equivalent to  $R_j^2 > 0.8$ ) give some cause for concern.

**Remark:** these values do not give information about which variables are in the same collinear group of variables.

## Example: bodyfat data

We compute the *VIF*-values for the bodyfat data.

```
> bodyfatlm=lm(Fat~Thigh+Triceps+Midarm, data=bodyfat)
> library(car); vif(bodyfatlm)
    Thigh   Triceps   Midarm
564.3434 708.8429 104.6060
> bodyfatlm2=lm(Fat~Triceps+Midarm, data=bodyfat)
> vif(bodyfatlm2)
    Triceps   Midarm
1.265118 1.265118
> bodyfatlm3=lm(Fat~Thigh, data=bodyfat)
> vif(bodyfatlm3)
Error in vif.default(bodyfatlm3) : model contains fewer than 2 terms
```

If we fit the full model all 3 VIF's are large, so there is a collinearity problem (as we saw in the scatter plots). The other 2 models are ok with respect to collinearity problems.

# To finish

Today we discussed:

- strategies to choose the variables (step up, step down)
- diagnostics in linear regression
- problems in linear regression (outliers and influence points, collinearity)

Next time: Lasso, ANCOVA, multiple testing, FDR control.

# Experimental Design and Data Analysis, Lecture 9

Eduard Belitser

VU Amsterdam

# Lecture overview

- ① ANCOVA
- ② prediction and feature selection in linear regression:
  - lasso
  - ridge
  - elastic net
- ③ multiple testing procedures, FDR control

## analysis of covariance (ANCOVA)

# Setting and design

An experiment with:

- a **numerical outcome**  $Y$ ;
- a **factor** that can be fixed at  $I$  levels.
- a **numerical explanatory variable**  $X$ .

Often the dependence of  $Y$  on the numerical variable  $X$  is a-priori evident, and the variable is included to increase the precision of the analysis.

**EXAMPLE** We investigate the **strength** of a wire as response to the **type of material** used and its **thickness**. (Thickness could not be controlled.)

**EXAMPLE** A subject must press a green or red button if there is a car in the picture shown on the screen, with outcome **reaction time**, factors **presence or not of an auditory stimulus** and explanatory variable **age of the subject**.

## Design

- Select  $NI$  experimental units randomly from the population of interest.
- Measure the  $X$  of each unit.
- Assign level  $i$  of the factor randomly to  $N$  units.
- Perform the experiment  $NI$  times independently.

Randomization is as for one-factor experiments (1-way ANOVA).

# The model and hypothesis to test

Data:  $(Y_{i1}, X_{i1}), (Y_{i2}, X_{i2}), \dots, (Y_{iN}, X_{iN})$ ,  $i = 1, 2, \dots, I$ .

The linear ANCOVA model assumes that

$$Y_{ik} = \mu + \alpha_i + \beta X_{ik} + e_{ik}, \quad i = 1, \dots, I, \quad k = 1, \dots, N,$$

for errors ( $e_{ik}$ ) that can be viewed a random sample from a normal population.

We want test the null hypothesis  $H_0 : \alpha_i = 0$ ,  $i = 1, 2, \dots, I$ , and  $H_0 : \beta = 0$ .

We also want to estimate the parameters  $\alpha_1, \dots, \alpha_I$  and  $\beta$ .

Any ANCOVA/ANOVA can always be seen as linear regression  $\mathbf{Y} = \mathbf{Z}\boldsymbol{\gamma} + \mathbf{e}$  with the certain design matrix  $Z$  and parameter vector  $\gamma$ . For example, for  $I = 2$ ,  $N = 3$ ,

$$\mathbf{Y} = \begin{pmatrix} Y_{11} \\ Y_{12} \\ Y_{13} \\ Y_{21} \\ Y_{22} \\ Y_{23} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & X_{11} \\ 1 & 1 & 0 & X_{12} \\ 1 & 1 & 0 & X_{13} \\ 1 & 0 & 1 & X_{21} \\ 1 & 0 & 1 & X_{22} \\ 1 & 0 & 1 & X_{23} \end{pmatrix} \begin{pmatrix} \mu \\ \alpha_1 \\ \alpha_2 \\ \beta \end{pmatrix} + \begin{pmatrix} e_{11} \\ e_{12} \\ e_{13} \\ e_{21} \\ e_{22} \\ e_{23} \end{pmatrix} = \mathbf{Z}\boldsymbol{\gamma} + \mathbf{e}.$$

The first column of  $Z$  is related to the intercept  $\mu$ , the next two are “dummy” variables related to ANOVA part  $\alpha_1, \alpha_2$ , the last is related to the linear regression part  $\beta$ .

# Analysis in R: data input

The data frame contains the data about the strength of a fiber made on 3 different machines. Thickness cannot be controlled, but measured.

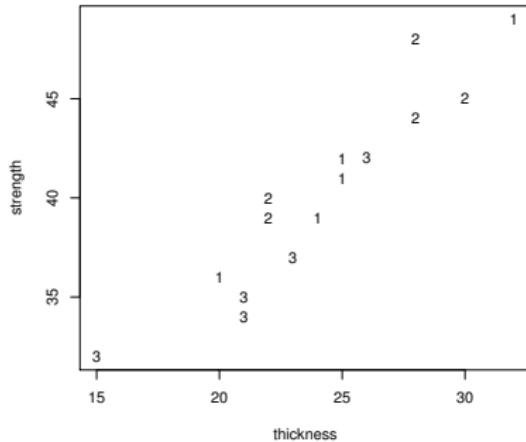
```
> fiber=read.table("fiber.txt",header=TRUE); fiber
```

	strength	thickness	type
--	----------	-----------	------

1	36	20	1
2	41	25	1
3	39	24	1
4	42	25	1
5	49	32	1
6	40	22	2
7	48	28	2
8	39	22	2
9	45	30	2
10	44	28	2
11	35	21	3
12	37	23	3
13	42	26	3
14	34	21	3
15	32	15	3

# Analysis in R: graphics

```
> plot(strength~thickness,pch=as.character(type))
```



Strength clearly increases with thickness. Its dependence on type is not so clear.

# Analysis in R: testing (1)

```
> fiber$type=as.factor(fiber$type)
> anova(lm(strength~type,data=fiber))
[ some output deleted ]
      Df Sum Sq Mean Sq F value Pr(>F)
type     2   140.4   70.200   4.0893 0.04423 *
```

Factor type is significant, but one-way ANOVA with only factor type is not correct!

```
> fiber1=lm(strength~thickness+type,data=fiber) # type second!
> anova(fiber1)      # only p-value for type is relevant
[ some output deleted ]
      Df Sum Sq Mean Sq F value Pr(>F)
thickness  1 305.130 305.130 119.9330 2.96e-07 ***
type      2   13.284    6.642   2.6106  0.1181
Residuals 11   27.986    2.544
```

Factor type is now insignificant. The output of ANCOVA depends on the order of the variables in the model formula. The correct p-value for type is obtained with `strength~thickness+type`, not with `strength~type+thickness`. Alternative: use `drop1` instead of `anova`, see next slide.

## Analysis in R: testing (2)

```
> drop1(fiber1,test="F")      # here all p-values are relevant
Single term deletions

Model:
strength ~ thickness + type
          Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>             27.986 17.355
thickness   1    178.014 206.000 45.297 69.9694 4.264e-06 ***
type       2     13.284  41.270 19.181  2.6106   0.1181
```

The command `drop1` is very handy: it performs the tests for the both models, `strength~thickness+type` and `strength~type+thickness` at once, whereas the *p*-values in the output of `anova` are sequential, as in a step-up strategy. This problem does not arise in (balanced) ANOVA or linear regression, but it does in an unbalanced ANOVA, ANCOVA and mixed models. Another (and the best) way to get correct *p*-values, e.g., for the factor `type`: `fiber2=lm(strength~thickness,data=fiber)`, then `anova(fiber2,fiber1)` will give the right *p*-values for the factor `type`.

# Analysis in R: estimation

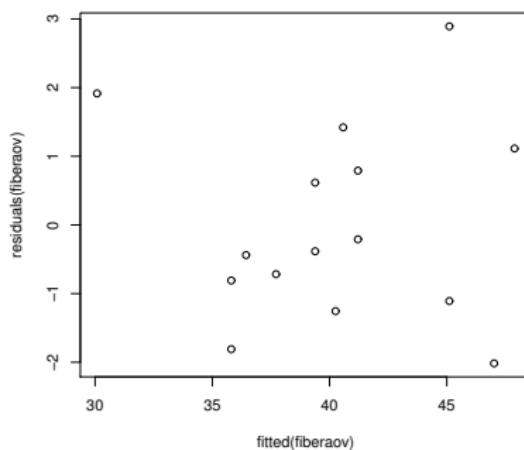
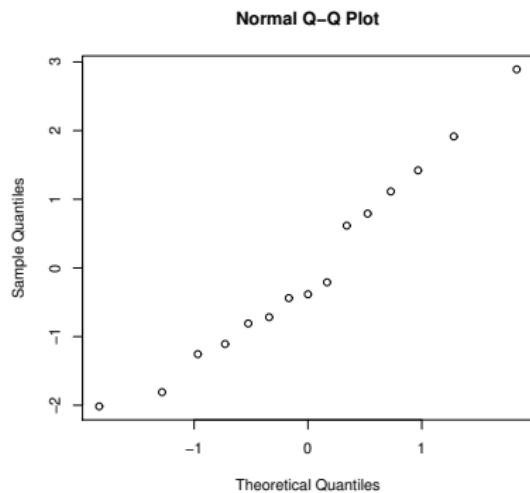
```
> fiber1=lm(strength~thickness+type,data=fiber); summary(fiber1)
[ some output deleted ]
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  17.360    2.961   5.862  0.000109 ***
thickness     0.954    0.114   8.365  4.26e-06 ***
type2        1.037    1.013   1.024  0.328012
type3       -1.584    1.107  -1.431  0.180292
```

This shows the coefficient estimates  $\hat{\mu}$ ,  $\hat{\beta}$ ,  $\hat{\alpha}_2$  and  $\hat{\alpha}_3$  ( $\hat{\alpha}_1 = 0$  as this is the default treatment parameterization). Their confidence intervals can be obtained by `confint(fiber1)`. As  $\hat{\beta} = 0.954 > 0$ , the thicker the fiber, the stronger it is, the strongest type of fiber is type2, although factor type is now insignificant. As, in case of anova and linear model, the rest concerns testing the individual hypothesis about the coefficients being zero. For example, the p-value for testing  $H_0 : \beta = 0$  (the coefficient for thickness variable is 4.26e-06, hence  $H_0 : \beta = 0$  is rejected).

# Analysis in R: diagnostics

The residuals and fitted values can (and **should**) be investigated as usual.

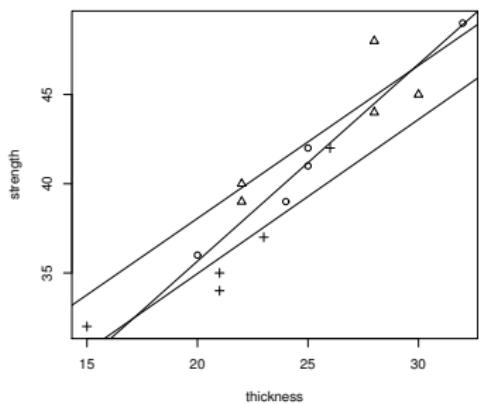
```
> qqnorm(residuals(fiber1))
> plot(fitted(fiber1),residuals(fiber1))
```



# Analysis in R: interaction between factor and predictor (1)

The model  $Y_{ik} = \mu + \alpha_i + \beta X_{ik} + e_{ik}$  says that within each level  $i$  of the factor the dependence of  $Y$  on  $X$  is a straight line with the same slope.

```
> plot(strength~thickness,pch=unclass(type))
> for (i in 1:3) abline(lm(strength~thickness,data=fiber[fiber$type==i,]))
```



Plot shows no indication that the true lines would not be parallel. We can test for that as follows: fit the model with different slopes  $\beta_1, \beta_2, \beta_3$  for each factor level  $Y_{in} = \mu + \alpha_i + \beta_i X_{in} + e_{in}$ , and then test  $H_0 : \beta_1 = \beta_2 = \beta_3$ . In other words, this is testing for the **interaction between factor type and predictor thickness**.

# Analysis in R: interaction between factor and predictor (2)

Testing for the **interaction between factor type and predictor thickness** is done by including the interaction term **type:thickness** in the model.

```
> fiber3=lm(strength~type*thickness,data=fiber); anova(fiber3)
[ some output deleted ]
          Df  Sum Sq Mean Sq F value    Pr(>F)
type           2 140.400  70.200 25.0231 0.0002107 ***
thickness       1 178.014 178.014 63.4538 2.291e-05 ***
type:thickness 2   2.737   1.369  0.4878 0.6292895
Residuals      9  25.249   2.805
```

The model formula **type\*thickness**, rather than **type+thickness**, describes the model  $Y_{ik} = \mu + \alpha_i + \beta_i X_{ik} + e_{ik}$ . Only the last p-value is relevant which always concerns interaction for models with interaction. We conclude from it that  $H_0 : \beta_1 = \beta_2 = \beta_3$  is not rejected, i.e., there is no interaction between factor type and predictor thickness (or, the slopes for all groups are the same).

# Analysis in R: interaction between factor and predictor (3)

```
> summary(fiber3)
[ some output deleted ]
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  13.5722    4.9375   2.749  0.022520 *
type2        7.3421    7.6684   0.957  0.363355
type3        4.1068    6.6631   0.616  0.552932
thickness     1.1043    0.1937   5.702  0.000294 ***
type2:thickness -0.2471   0.2960  -0.835  0.425337
type3:thickness -0.2401   0.2843  -0.845  0.420215
```

The estimates of type2:thickness and type3:thickness give the estimated differences  $\hat{\beta}_2 - \hat{\beta}_1$  and  $\hat{\beta}_3 - \hat{\beta}_1$ . The interaction term is not significant. So, no indication that the initial analysis is in trouble.

## Prediction and feature selection in linear regression

# Lasso, ridge and elastic net method (1)

- In this case we have only 4 variables to choose from, so we were able to identify the significant ones by a manual inspection of  $p$ -values.
- This will quickly become unfeasible if the number of predictors is big.
- An algorithm that could somehow automatically shrink the coefficients of the insignificant variables or (better!) set them to zero altogether?
- This is precisely what **lasso** and its close cousin, **ridge regression**, do.
- Lasso and ridge regularization** work by adding a penalty term  $\lambda P(\beta)$  to the mean residual sum of squares

$$\frac{1}{N} \sum_{n=1}^N (Y_n - (\beta_0 + \beta_1 X_{n,1} + \dots + \beta_p X_{n,p}))^2 = \frac{\|Y - X\beta\|^2}{N}$$

and minimizing the resulting sum  $\frac{1}{N} \|Y - X\beta\|^2 + \lambda P(\beta)$  ( $2N$  can be used instead of  $N$ ) with respect to  $\beta = (\beta_0, \beta_1, \dots, \beta_p) \in \mathbb{R}^{p+1}$ :

$$\frac{1}{N} \|Y - X\beta\|^2 + \lambda P(\beta) \rightarrow \min_{\beta}$$

## Lasso, ridge and elastic net methods (2)

- **Lasso method:**  $P(\beta) = \|\beta\|_1 = \sum_{k=0}^p |\beta_k|$ , i.e.,

$$\min_{\beta} \left\{ \frac{\|Y - X\beta\|^2}{N} + \lambda \|\beta\|_1 \right\} = \min_{\beta} \left\{ \frac{\|Y - X\beta\|^2}{N} + \lambda \sum_{k=0}^p |\beta_k| \right\}.$$

- **Ridge method:**  $P(\beta) = \|\beta\|_2^2 = \sum_{k=0}^p \beta_k^2$ , i.e.,

$$\min_{\beta} \left\{ \frac{\|Y - X\beta\|^2}{N} + \lambda \|\beta\|_2^2 \right\} = \min_{\beta} \left\{ \frac{\|Y - X\beta\|^2}{N} + \lambda \sum_{k=0}^p \beta_k^2 \right\}.$$

- **Elastic net method:**  $P(\beta) = \alpha \|\beta\|_1 + (1 - \alpha) \|\beta\|_2^2$  ( $0 \leq \alpha \leq 1$  controls the “mix” of ridge and lasso regularisation, with  $\alpha = 1$  being “pure” lasso and  $\alpha = 0$  being “pure” ridge), i.e.,

$$\min_{\beta} \left\{ \frac{\|Y - X\beta\|^2}{N} + \lambda (\alpha \|\beta\|_1 + (1 - \alpha) \|\beta\|_2^2) \right\}.$$

- Parameter  $\lambda \geq 0$  is a free parameter which is usually selected by using a method called **cross-validation**.

# Lasso, ridge and elastic net methods

- Ridge regression enforces the  $\beta$  coefficients to be lower, but it does not enforce them to be zero. That is, it will not get rid of irrelevant features but rather minimize their impact on the trained model.
- Lasso method overcomes the disadvantage of ridge regression by setting the coefficients  $\beta$  to zero if they are not relevant. One usually ends up with fewer features included in the model than you started with, which is an advantage.
- The R-package `glmnet` implements the elastic net method (for any  $0 \leq \alpha \leq 1$ ) by R-function `glmnet`, with particular cases `ridge` ( $\alpha = 0$ ) and `lasso` ( $\alpha = 1$ ).
- The choice of  $\lambda$  is done by the `cross-validation method`, implemented by the R-function `cv.glmnet`.

# Analysis in R: generic code for lasso (ridge and elastic net)

Suppose we have a data frame named `data`, with its first column being the response variable, and the remaining columns are the features to select from.

```
>library(glmnet)
>x=as.matrix(data[,-1]) #remove the response variable
>y=as.double(as.matrix(data[,1])) #only the response variable
>train=sample(1:nrow(x),0.67*nrow(x)) # train by using 2/3 of the data
>x.train=x[train,]; y.train=y[train] # data to train
>x.test=x[-train,]; y.test=y[-train] # data to test the prediction quality
>lasso.mod=glmnet(x.train,y.train,alpha=1)
>cv.lasso=cv.glmnet(x.train,y.train,alpha=1,type.measure='mse')
>plot(lasso.mod,label=T,xvar="lambda") #have a look at the lasso path
>plot(cv.lasso) # the best lambda by cross-validation
>plot(cv.lasso$glmnet.fit,xvar="lambda",label=T)
>lambda.min=lasso.cv$lambda.min; lambda.1se=lasso.cv$lambda.1se
>coef(lasso.model,s=lasso.cv$lambda.min) #beta's for the best lambda
>y.pred=predict(lasso.model,s=lambda.min,newx=x.test) #predict for test
>mse.lasso=mean((y.test-y.pred)^2) #mse for the predicted test rows
```

`lambda.min` is the value of  $\lambda$  that gives minimum mean cross-validated error. The other  $\lambda$  saved is `lambda.1se`, which gives the most regularized model such that error is within one standard error of the minimum.

multiple comparisons

# Multiple testing

- $H_0$  is falsely rejected (**type I error**) with probability at most  $\alpha_{ind}$  ( $= 0.05$ ).
- Given 2 null hypotheses there are 2 possibilities to make such an error.  
The probability of at least 1 error is then at most  $0.05 + 0.05 = 0.1$ .
- Suppose for each of  $m$  null hypotheses  $H_{0,1}, \dots, H_{0,m}$ , the probability of type I error is at most  $\alpha_{ind}$ , then the probability of at least 1 error is at most  $m\alpha_{ind}$ . Indeed,

$$P(\text{at least one } H_{0,i} \text{ is rejected}) \leq \sum_{i=1}^m P(H_{0,i} \text{ is rejected}) \leq m\alpha_{ind}.$$

- $P(\text{at least one } H_{0,i} \text{ is rejected})$  is called **family-wise error rate (FWER)**.
- To provide  $\text{FWER} \leq 0.05$ , we can impose  $\alpha_{ind} \leq \frac{0.05}{m}$  for all  $H_{0,i}$ . Then indeed

$$\text{FWER} \leq m\alpha_{ind} \leq m \frac{0.05}{m} = 0.05.$$

# Multiple testing: Bonferroni correction

- Thus, a simple way to control the family-wise error rate FWER  $\leq \alpha_{tot}$  for some overall level  $\alpha_{tot}$  is to carry out each individual test with  $\alpha_{ind} = \frac{\alpha_{tot}}{m}$ , known as the **Bonferroni correction**.
- This is the same as to compare the individual  $p$ -values  $p_{ind}$  to  $\alpha_{ind} = \frac{\alpha_{tot}}{m}$ .
- Adjusted  $p$ -values for simultaneous tests  $p_{adj}$  are such that if every  $H_{0,i}$  with  $p_{adj} \leq \alpha_{tot}$  is rejected, then FEWR  $\leq \alpha_{tot}$ .
- Adjusted  $p$ -value according to Bonferroni correction is  $p_{adj} = mp_{ind}$ .
- In R, the adjusted  $p$ -values are called **adjusted P-values for Multiple Comparisons**, and are computed by `p.adjust`.
- Bonferroni correction is very conservative. Indeed, for reasonable  $\alpha_{tot}$  (like 0.05) and relatively large  $n$  (like  $n = 100$ ), there will be very few **simultaneously rejected**  $H_{0,i}$ 's, because hardly ever we will have  $100p_{ind} \leq 0.05$ , or  $p_{ind} \leq 0.00005$ .

# Multiple testing procedures for controlling FWER

Multiple testing arises when:

- there are many parameters of interest.
- investigating all differences  $\alpha_i - \alpha_{i'}$  of a set of effects  $\alpha_i$  in ANOVA.

The latter is the so called "*a-posteriori testing*", performed following rejection of a composite hypothesis of the type  $H_0 : \alpha_i = 0, i = 1, \dots, I$ .

Bonferroni correction is not the only method to **control FWER**, alternatives:

- Sidak correction** (under indep. assump., slightly better than Bonferroni),
- Holm-Bonferroni method**, better than Bonferroni (making it obsolete)
- Hochberg's step-up procedure**
- Tukey's procedure** (`library(multcomp)`, only for pairwise comparisons).
- some extensions of the above mentioned
- Similarly, one designs **simultaneous confidence intervals** for a set of parameters that have overall confidence level of  $1 - \alpha_{tot}$ .

To implement these methods in R: fed with given individual  $p$ -values  $p_{ind}$  and a specified method, `p.adjust` gives the adjusted  $p$ -values  $p_{adj}$  (not for Sidak and Tukey's procedures) which should be compared to a specified significance level  $\alpha_{tot}$ . The corresponding method rejects those hypothesis for which  $p_{adj} \leq \alpha_{tot}$ .

# Individual $p$ -values obtained in ANOVA

Recall the data pvc on the production of the plastic PVC, where 3 operators used 8 different devices called resin to produce PVC of size psiz.

```
> pvc$operator=as.factor(pvc$operator); pvc$resin=as.factor(pvc$resin)
> pvcaov=lm(psize~operator*resin,data=pvc); summary(pvcaov)
[ some output deleted ]
Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 36.2500    0.8598 42.164 < 2e-16 ***
operator2   -0.8500    1.2159 -0.699 0.491216
operator3   -0.9500    1.2159 -0.781 0.442245
resin2      -1.1000    1.2159 -0.905 0.374615
resin3      -5.5500    1.2159 -4.565 0.000126 ***
[ some output deleted ]
resin8       0.5500    1.2159  0.452 0.655078
operator2:resin2 1.0500    1.7195  0.611 0.547175
[ some output deleted ]
operator3:resin8 -2.7000    1.7195 -1.570 0.129454
```

The  $p$ -values produced above are **not simultaneous**. The  $p$ -values in the lines resin2, resin3, ... are for testing the **individual** hypotheses  $H_0 : \beta_2 = \beta_1, H_0 : \beta_3 = \beta_1, \dots$

# Multiple testing in R by Tukey's method

```
> library(multcomp)
> pvcmult=glht(pvcaov,linfct=mcp(resin="Tukey"))
> summary(pvcmult)

Estimate Std. Error t value Pr(>|t|)
2 - 1 == 0   -1.100     1.216  -0.905  0.9827
3 - 1 == 0   -5.550     1.216  -4.565  <0.01  ***
4 - 1 == 0   -6.550     1.216  -5.387  <0.01  ***
5 - 1 == 0   -4.400     1.216  -3.619  0.0251 *
6 - 1 == 0   -6.050     1.216  -4.976  <0.01  ***
7 - 1 == 0   -3.350     1.216  -2.755  0.1538
8 - 1 == 0    0.550     1.216   0.452  0.9998
[ some output deleted ]
8 - 6 == 0    6.600     1.216   5.428  <0.01  ***
8 - 7 == 0    3.900     1.216   3.208  0.0625 .
```

Adjusted *p*-values for simultaneous testing the null hypotheses  $H_0 : \beta_2 = \beta_1$ ,  $H_0 : \beta_3 = \beta_1$ ,  $H_0 : \beta_4 = \beta_1$ , ...,  $H_0 : \beta_8 = \beta_7$ , where  $\beta_j$  is the main effect of the *j*th level of resin. The probability that one or more of these would be less than 0.05 while the corresponding null hypothesis were true is less than 0.05. Thus we can "safely" say that *all* differences with *p*-value < 0.05 are nonzero.

# False Discovery Rate (FDR)

- Procedures that control the FWER are considered too conservative for most cases of multiple testing (they lead to a substantial loss in power).
- Better to control (and **less stringent**) is the **False Discovery Rate (FDR)** introduced by Benjamini and Hochberg (1995), the expected proportion of falsely rejected null hypothesis among the rejected hypotheses.
- Testing  $m$  hypotheses simultaneously (of which  $m_0$  are true null hypotheses):

	$H_0$ is true	$H_1$ is true	Total
Procedure rejects $H_0$	$V$	$S$	$R$
Procedure does not reject $H_0$	$U$	$T$	$m - R$
Total	$m_0$	$m - m_0$	$m$

$V$  is the number of **false positives**;  $T$  is the number of **false negatives**.

- Random variable  $R$  is observed and the number of hypothesis  $m$  is known.
- Random variables  $V, S, U, T$  are unobserved and the number of true hypothesis  $m_0$  is unknown.
- $\text{FDR} = E\left(\frac{V}{R}\right)$ , where we define  $\text{FDR} = 0$  if  $R = 0$  (then also  $V = 0$ ).

# BH and BY procedures to control FDR

- The Benjamini-Hochberg procedure ensures that its FDR is at most  $\alpha$ :
  - Order the  $p$ -values  $p_{(1)} \leq p_{(2)} \leq \dots \leq p_{(m)}$  and the null hypotheses  $H_{0,(1)}, H_{0,(2)}, \dots, H_{0,(m)}$  correspondingly;
  - If  $k_{max} = \max_k (p_{(k)} \leq \frac{\alpha k}{m})$  exists, reject  $H_{0,(1)}, \dots, H_{0,(k_{max})}$ ; otherwise reject nothing.
- The BH procedure is valid when the  $m$  tests are independent.
- Notice that  $k_{max} = \max_k (p_{(k)} \leq \frac{\alpha k}{m}) = \max_k (\frac{mp_{(k)}}{k} \leq \alpha)$ .
- Command `p.adjust` gives the adjusted ordered  $p$ -values  $\frac{mp_{(k)}}{k}$ , which should be compared to  $\alpha$ , to control FDR up to level  $\alpha$ .
- Benjamini-Yekutieli procedure (BY) is the generalization of BH procedure (for arbitrary dependence assumptions): instead of  $m$  one takes  $mc(m)$  where  $c(m) = \sum_{i=1}^m \frac{1}{i}$ , so the BY procedure is a bit more conservative.
- `p.adjust` gives the adjusted ordered  $p$ -values also for the BY procedure.

# Multiple testing in R

```
> p.raw=summary(pvcaov)$coef[,4] # vector of individual (raw) p-values
> p.raw=p.raw[order(p.raw)] # order the p-values
> p.val=as.data.frame(p.raw)
> p.val$Bonferroni=p.adjust(p.val$p.raw,method="bonferroni")
> p.val$Holm=p.adjust(p.val$p.raw,method="holm")
> p.val$Hochberg=p.adjust(p.val$p.raw,method="hochberg")
> p.val$BH=p.adjust(p.val$p.raw,method="BH")
> p.val$BY=p.adjust(p.val$p.raw,method="BY"); round(p.val,3)
```

	p.raw	Bonferroni	Holm	Hochberg	BH	BY
(Intercept)	0.000	0.000	0.000	0.000	0.000	0.000
resin4	0.000	0.000	0.000	0.000	0.000	0.001
resin6	0.000	0.001	0.001	0.001	0.000	0.001
resin3	0.000	0.003	0.003	0.003	0.001	0.003
resin5	0.001	0.033	0.027	0.027	0.007	0.025
resin7	0.011	0.264	0.209	0.209	0.044	0.166
operator3:resin8	0.129	1.000	1.000	0.954	0.444	1.000
operator3:resin5	0.361	1.000	1.000	0.954	0.892	1.000
resin2	0.375	1.000	1.000	0.954	0.892	1.000
operator3	0.442	1.000	1.000	0.954	0.892	1.000
[ some output deleted ]						

# To wrap up

Today we learned:

- ANCOVA
- prediction and feature selection in linear regression
- multiple testing procedures, FDR control

Next time: Logistic regression, Poisson regression

# Experimental Design and Data Analysis, Lecture 10

Eduard Belitser

VU Amsterdam

# Lecture overview

## ① generalized linear models

- logistic regression
- Poisson regression

glm  
●○○

logistic regression  
○○○○○○○○○○○○○○○○○○○○○○○○

Poisson regression  
○○○○○○○

further designs  
○○○

# generalized linear models

# Setting

An experiment with:

- an **outcome  $Y$**  that has a **different nature** than in ANOVA or linear regression;
- one or more **numerical explanatory variables  $X_1, \dots, X_p$** .
- one or more **factor explanatory variables**. ("independent variable").

The purpose is to explain  $Y$  by a **linear function** of  $X$ .

**EXAMPLE** Educational study with outcome **passed the exam or not** and explanatory variable **number of pupils per teacher**.  $Y$  is **binary**.

**EXAMPLE** The **number of plant species** on a Galapagos Island, with explanatory variables **area**, **highest elevation**, **distance to nearest island**, **distance to Santa Cruz island** and **area of adjacent island**.  $Y$  is a **count**.

**EXAMPLE** Political study with outcome **party identification** with explanatory variables **age**, **education level** and **income**.  $Y$  is **multinomial (categorical)**.

# Different models

For each of the three examples a different model applies.

- For **binary** responses, the **logistic regression model** assumes:

$$\log \frac{P(Y = 1)}{P(Y = 0)} = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p.$$

- For **multinomial** responses, the **multinomial logit model** assumes:

$$\log \frac{P(Y = C_i)}{P(Y = C_1)} = \beta_0^i + \beta_1^i X_1 + \dots + \beta_p^i X_p,$$

where  $C_1$  is the reference class of the categorical responses.

- For **count** responses, the **Poisson regression model** assumes:

$$\log E(Y) = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p.$$

## logistic regression

# Setting

An experiment with:

- an **outcome**  $Y$  that is 0 or 1 ("binary dependent variable");
- one or more **numerical explanatory variables**  $X_1, \dots, X_p$ .
- one or more **factor explanatory variables**  $F_1, \dots, F_m$ .

The purpose is to explain  $Y$  by a **function** of  $X$ 's and  $F$ 's .

**EXAMPLE** A subject **participates or not** in an internet survey presented in 3 formats at 3 different days of the week.

**EXAMPLE** Educational study with outcome **passed the exam or not** and explanatory variable **number of pupils per teacher**.

**EXAMPLE** Medical study with outcome **patient died or not** with explanatory variables **type of treatment, sex** and **age**.

# Design

Logistic regression can be used for factorial experiments, in a regression setting, for ANCOVA, and for experiments with blocks.

- The design is the same as for the corresponding experiment.

Logistic regression is also used in a [case-control setting](#).

- Consider a population consisting of 2 subpopulations of units with outcome 0 and with outcome 1, respectively (“controls” and “cases”).
- Independently choose random samples of units from the two subpopulations.
- Measure the explanatory variables for these units.

The case-control design has the advantage that the numbers of cases and controls in the samples can be fixed in advance (and made approx. equal).

# Logistic regression model

- Response  $Y$  is **categorical** (0-1) → cannot use lin.regr./anova/ancova.
- In this case, we model  $\Pr(Y = 1)$  as a function of explanatory variables.
- The **logistic regression model** assumes that outcome  $Y_k \in \{0, 1\}$  satisfies

$$P(Y_k = 1) = \Psi(\mathbf{x}_k^T \boldsymbol{\theta}) = \frac{1}{1 + e^{-\mathbf{x}_k^T \boldsymbol{\theta}}}, \quad P(Y_k = 0) = 1 - P(Y_k = 1),$$

$\mathbf{x}_k^T \boldsymbol{\theta} = \mu + \alpha_{f(k)} + \dots + \beta_1 x_{k1} + \dots$ ,  $f(k) \in \{1, \dots, I\}$  is the factor level of observation  $Y_k$ ,  $\mathbf{x}_k = (1, \dots, 0, 1, 0, \dots, x_{k1}, \dots)^T$  is the  $k$ -th vector of predictor values,  $\boldsymbol{\theta} = (\mu, \alpha_1, \dots, \beta_1, \dots)^T$  is the parameter vector.

- $\Psi(x) = 1/(1 + e^{-x})$ ,  $\Psi : \mathbb{R} \mapsto [0, 1]$ , is called **logistic function**.
- The explanatory variables can be either numerical or categorical, or a mix.
- As in lin.regr./anova/ancova, we can test for factors/variables, their interactions, estimate the parameters, and predict future observations.
- In R: `glm(y~f1+...+x1+...,family=binomial,data=mydata)`

If the categorical response variable has more than 2 values, one extends the usual logistic regression to **multinomial logistic regression** (implem. in R by special packages).

## Example: logistic regression with one factor and one contin. predictor

- For example, for a single factor with  $I$  levels and a single numerical explanatory variable the **logistic regression model** assumes that the outcome  $Y_{ik}$  of a unit measured at level  $i$  of the factor and having explanatory variable  $X_{ik}$  satisfies

$$P(Y_{ik} = 1) = \Psi(\mu + \alpha_i + \beta X_{ik}), \quad i = 1, \dots, I, \quad k = 1, \dots, N,$$

- Want to **tests** the hypotheses  $H_0 : \alpha_1 = \dots = \alpha_I = 0$ , and  $H_0 : \beta = 0$ , i.e., the factor and/or explanatory variable do not influence the outcome.
- Also **estimate** the factor effects  $\alpha_1, \dots, \alpha_I$  and the regression parameter  $\beta$ .

The outcome  $Y$  is like a coin-toss; the probability  $P(Y = 1)$  of “heads” is modelled. The **linear predictor**  $\mu + \alpha_i + \beta X_{ik}$  can take any real value. The logistic function (monotonically increasing) maps this into a probability: a number between 0 and 1. A bigger linear predictor gives a probability of heads closer to 1.

# Logistic regression: odds

- The **odds** is  $o = \frac{P(Y=1)}{P(Y=0)}$ . This means that the probability of “success”  $P(Y = 1)$  is  $o$  times as big as the probability of “failure”  $P(Y = 0)$ .
- Logistic regression is a linear model for the **log odds**:  $\log o_k = \mathbf{x}_k^T \boldsymbol{\theta}$ .
- E.g., a logistic regression with one factor and one contin. predictor,

$$o_{ik} = \frac{P(Y_{ik} = 1)}{P(Y_{ik} = 0)} = e^{\mu + \alpha_i + \beta X_{ik}}, \quad \text{or} \quad \log o_{ik} = \mu + \alpha_i + \beta X_{ik}.$$

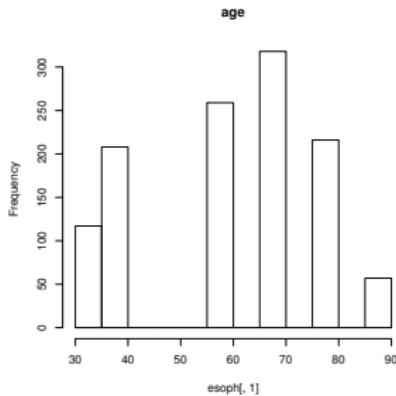
- A change  $\Delta$  in the **linear predictor**  $\mu + \alpha_i + \beta X_{ik}$  multiplies the odds by  $e^\Delta$ . For example,
  - an increase of predictor  $X$  by one unit multiplies the odds by  $e^\beta$ .
  - a change from level  $i$  to level  $i'$  multiplies the odds by  $e^{\alpha_{i'} - \alpha_i}$ .

# Analysis in R: data input, graphics

In the data set `esoph.txt`, the column `cancer` indicates whether the individual (1–1175) suffers from cancer of the esophagus (gullet). The first three columns give the age rounded to a multiple of 10, alcohol consumption, and tobacco use.

```
> hist(esoph[,1],main="age")
```

```
> esoph=read.table("esoph.txt",h=T)
> esoph
  age  alc tob cancer
1   30   20   5     0
2   30   20   5     0
3   30   20   5     0
[ a lot of output deleted ]
1173  90  140   5     0
1174  90  140  15     1
1175  90  140  15     0
```



The histogram shows the age distribution.

# Analysis in R: summary

```
> tot=xtabs(~alc+tob,data=esoph)
> tot
      tob
alc      5   15   25   35
  20  270  94  47  33
  60  213 102  77  38
 100  80  68  22  19
 140  40  30  19  23
```

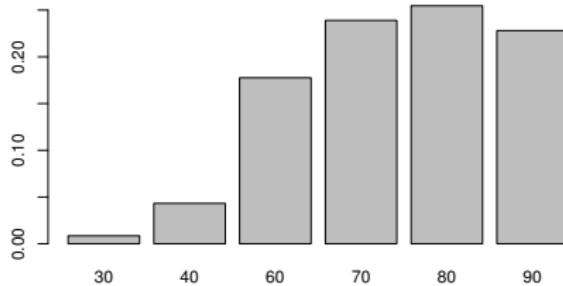
The table shows the total numbers of individuals for each combination of levels of alcohol and tobacco use.

```
> tot.c=xtabs(cancer~alc+tob,data=esoph)
> round (tot.c/tot,2)
      tob
alc      5   15   25   35
  20  0.03 0.11 0.11 0.15
  60  0.16 0.17 0.19 0.24
 100 0.24 0.28 0.27 0.37
 140 0.40 0.40 0.37 0.43
```

The table shows the percentage of individuals with cancer for every combination of levels of alcohol and tobacco use.

# Analysis in R: graphics

```
> totage=xtabs(~age,data=esoph) # counts of age groups  
> barplot(xtabs(cancer~age,data=esoph)/totage) # % of cancer per age group
```



The barplot shows the percentage per age-group. Since it doesn't look very linear, we will add  $age^2$  as an explanatory variable in the next slide.

Remark. This is just to demonstrate that one can create and include other variable(s) in the model, this is not necessarily good thing to do, for example the variable  $age^2$  is not well interpretable and, besides, it will turn out to be not useful.

# Analysis in R: estimation and testing

```
> esoph$age2=esoph$age^2
> esophglm=glm(cancer~age+age2+alc+tob,data=esoph,family=binomial)
> summary(esophglm)

            Estimate Std. Error z value Pr(>|z|)
(Intercept) -9.8072283  1.5850673 -6.187 6.12e-10 ***
age          0.1688542  0.0491991  3.432 0.000599 ***
age2         -0.0009608  0.0003776 -2.545 0.010934 *
alc          0.0162614  0.0021092  7.710 1.26e-14 ***
tob          0.0256080  0.0081412  3.145 0.001658 **
```

The R-function `glm` (generalized linear model) is used instead of `lm` to create the `glm` object. The option `family=binomial` overrules the default normal model (which gives `lm`). The 4 explanatory variables are inserted here as **numerical**. The estimated odds is  $\hat{o}_k = \frac{\widehat{P(Y_k=1)}}{\widehat{P(Y_k=0)}} \approx \exp\{-9.8 + 0.17\text{age}_k - 0.00096\text{age}^2_k + 0.016\text{alc}_k + 0.026\text{tob}_k\}$ . The positive signs of the parameter estimates mean that higher values of these variables give higher probability of cancer. For instance, raising tobacco by 1 increases the linear predictor by 0.0256080 and increases the odds of cancer by a factor  $e^{0.0256080} = 1.026$ . For age the dependence is parabolic: from 25 to 30 years the odds increase by  $\exp\{0.17 \cdot (30 - 25) - 0.00096 \cdot (30^2 - 25^2)\} = 1.786932$ .

# Analysis in R: `glm` instead of `lm`

- Once a `glm` object is created one can access the various components of the results in the same way as for any other linear model R-object, using functions such as `summary`, `anova`, `drop1`, `coef`, `residuals`, etc.
- For example, `mod=glm(y~x1+x2,data,family=binomial)`, and the command `summary(mod)` displays the (MLE) estimates of the model coefficients and individual tests that these coefficients are zero.
- Pay attention to the parametrization (in case of factors) and to the order of the variables in the model formula. Need to `specify the test` (for GLM's, "Chisq") in testing commands, e.g. `drop1(mod,test="Chisq")`.
- Instead of `anova` table, `anova(mod,test="Chisq")` yields the so called deviance tables, which are used to examine the progressive fit of the model as each covariate/factor is added to the model.
- The safest way (and to have the full control of what you test) is to use `anova(mod1,mod2,test="Chisq")` or `drop1(mod,test="Chisq")`.
- Diagnostics for GLM's is **not as straightforward as for linear models**, and will not be treated in this course. For example, there are at least **5 types of residuals** and **2 types of fitted values** for GLM's ( $\hat{\mu}_k$  and  $x_k^T \hat{\theta}$ ).

# Analysis in R: estimation and testing (1)

In the previous model, tob, alc and age were numeric, now they are categorical, treated as factors. The variable age2 is dropped.

```
> esoph$age=factor(esoph$age); esoph$alc=factor(esoph$alc)
> esoph$tob=factor(esoph$tob) # note: the variables are factors now
> glm2=glm(cancer~age+alc+tob,data=esoph,family=binomial); summary(glm2)
[ some output deleted ]
```

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-5.9108	1.0302	-5.738	9.59e-09	***
age40	1.6095	1.0675	1.508	0.131631	
age60	2.9752	1.0242	2.905	0.003673	**
age70	3.3584	1.0198	3.293	0.000991	***
age80	3.7270	1.0252	3.635	0.000278	***
age90	3.6818	1.0644	3.459	0.000542	***
alc60	1.1216	0.2384	4.704	2.55e-06	***
alc100	1.4471	0.2628	5.506	3.68e-08	***
alc140	2.1154	0.2876	7.356	1.90e-13	***
tob15	0.3407	0.2054	1.659	0.097159	.
tob25	0.3962	0.2456	1.613	0.106708	
tob35	0.8677	0.2765	3.138	0.001701	**

For example, the estimated odds for the group (age70, alc20, tob35) is  
 $\hat{o} \approx \exp\{-5.91 + 3.36 + 0 + 0.87\} = e^{-1.68}$ .

## Analysis in R: estimation and testing (2)

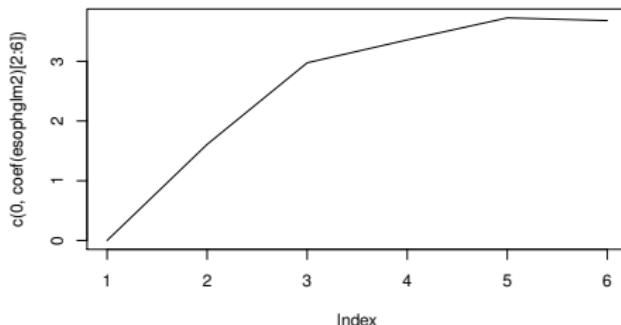
Recall that  $\widehat{P}(Y_k = 1) = \Psi(\mathbf{x}_k^T \hat{\theta})$ . For example, the estimate of the probability of cancer for the group (age70, alc20, tob35) is computed as

$$\Psi(\text{Intercept} + \text{age70} + \text{alc20} + \text{tob35}) = 0.1564698.$$

In R, all  $\widehat{P}(Y_k = 1)$  are obtained by `fitted(glm2)`. To predict the probability of cancer for newdata, use `predict(glm2,newdata,type="response")`, for example, for `newdata=data.frame(age="70",alc="20",tob="35")`.

Make a graph of the coefficients for the different age categories:

```
> plot(c(0,coef(glm2)[2:6]),type="l")
```



By inserting the variables as factors each level gets its own parameter, and we can look at the dependence on levels. Disadvantage: (too) many parameters.

# Analysis in R: estimation and testing (3)

```
> drop1(glm2,test="Chisq")
Single term deletions
      Df Deviance    AIC    LRT   Pr(Chi)
<none>  898.86  922.86
age     5    976.37 990.37 77.511 2.782e-15 ***
alc     3    964.91 982.91 66.054 2.984e-14 ***
tob     3    909.46 927.46 10.599   0.01411 *
```

As the variables are factors now, the `drop1` command reduces the list of the  $p$ -values to one  $p$ -value per variable **in the model formula**, for testing the null hypothesis that the factor has no effect. All three factors are significant. The `anova` command works too, but gives “sequential” tests, which are hard to interpret (**only the last p-value** can be well interpreted). Another (and the best) way to get correct  $p$ -values, for example, for the factor `alc`: `glm3=glm(cancer~age+tob,data=esoph,family=binomial)`, then `anova(glm3,glm2)` will give the right  $p$ -values for the factor `alc`.

# Aggregated data format (for logistic model)

- Measurements with the same values of all explanatory variables need not be represented by separate lines in the data matrix.
- Instead we can count for every combination of explanatory variables the total numbers of 0's and 1's.
- One line in dataset `esophshort.txt` contains the aggregated data of lines with equal values of the explanatory variables (factors) in the dataset.

```
> esophshort=read.table("esophshort.txt",header=TRUE)
> head(esophshort)
  age alc tob ncases ncontrols
1 30  20   5      0       40
2 30  20  15      0       10
3 30  20  25      0        6
4 30  20  35      0        5
5 30  60   5      0       27
6 30  60  15      0        7
> esophshort$age2=esophshort$age^2 # add the variable age^2
```

## Aggregated data format (2)

```
> shortglm=glm(cbind(ncases,ncontrols)~age+age2+alc+tob,  
+                 data=esophshort,family=binomial)  
> summary(shortglm)  
[ some output deleted ]  
              Estimate Std. Error z value Pr(>|z|)  
(Intercept) -9.8072283  1.5850903 -6.187 6.13e-10 ***  
age          0.1688542  0.0491997  3.432 0.000599 ***  
age2         -0.0009608  0.0003776 -2.545 0.010935 *  
alc           0.0162614  0.0021092  7.710 1.26e-14 ***  
tob           0.0256080  0.0081413  3.145 0.001658 **
```

The output is identical to that of the earlier analysis with the “long” data, using the explanatory variables as **numeric** variables.

This **aggregated format** in the form of pair (success,failure), the counts of successes and failures for each combination of levels of the factors (or values of numeric variables), is one of **3 possible ways** to specify the responses in R for the logistic model. This format **is not useful if there is a continuous predictor** in the model, taking different values for different individuals (e.g., diff. ages for diff. individuals).

# Testing interaction between factor and contin. predictor (1)

Consider a model with one factor alc and one contin. predictor age.

```
> esoph$age=as.numeric(esoph$age)
> glm3=glm(cancer~age+alc,data=esoph,family=binomial)
      Df Deviance    AIC    LRT  Pr(>Chi)
<none>   925.23  935.23
age       1   983.67  991.67 58.440 2.096e-14 ***
alc       3  1012.48 1016.48 87.244 < 2.2e-16 ***
```

Recall the model we are actually studying

$$P(Y_{in} = 1) = \Psi(\mu + \alpha_i + \beta X_{in}) = 1/(1 + e^{-(\mu + \alpha_i + \beta X_{in})}),$$

both the factor and contin. predictor are in the model (as in ancova).

However, the coefficient(s)  $\beta$  (reflecting the influence of the continuous predictor) may depend on the level of the factor, i.e.,

$$P(Y_{in} = 1) = \Psi(\mu + \alpha_i + \beta_i X_{in}) = 1/(1 + e^{-(\mu + \alpha_i + \beta_i X_{in})}).$$

In this case we say that the corresponding **factor and variable interact**.

## Testing interaction between factor and contin. predictor (2)

Testing for **no interaction** between the factor and predictor:  $H_0 : \beta_1 = \dots = \beta_l$ .

In R, to test for **interaction** (in **logistic model and ANCOVA**) between factor and contin. predictor, simply include the interaction term in the model formula, e.g.,  $y \sim f + x + f:x$  or  $y \sim f * x$ .

Testing for the **interaction** between factor alc and predictor age:

```
> glm4=glm(cancer~age*alc,data=esoph,family=binomial)
> anova(glm4,test="Chisq") # only the last p-value is relevant
[ some output deleted ]
```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
NULL		1174	1072.13		
age	1	59.647	1173	1012.48	1.135e-14 ***
alc	3	87.244	1170	925.23	< 2.2e-16 ***
age:alc	3	4.549	1167	920.68	0.208

Only the **last p-value is relevant** which always concerns interaction for models with interaction. We conclude that  $H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4$  is not rejected, i.e., there is **no interaction** between factor alc and predictor age. Testing for interaction between factors and contin. variables is the same as in **ANCOVA**. In this case drop1 will also give only one relevant p-value for the interaction.

# From logistic regression to machine learning prediction

- Fitting the observed data  $(X_1, Y_1), \dots, (X_N, Y_N)$  in logistic regression

$$P(Y_k = 1) = \frac{1}{1 + e^{-x_k^T \theta}}, \quad k = 1, \dots, N,$$

we obtain (by the maximum likelihood) an estimate  $\hat{\theta}$  of the parameter  $\theta$ .

- For a new predictor vector  $X_{new}$ , we can predict its success probability

$$\hat{P}_{new} = \frac{1}{1 + e^{-x_{new}^T \hat{\theta}}}.$$

- Now use  $\hat{P}_{new}$  to predict the new label  $\hat{Y}_{new}$  as

$$\hat{Y}_{new} = \begin{cases} 1, & \text{if } \hat{P}_{new} \geq p_0 \\ 0, & \text{if } \hat{P}_{new} < p_0 \end{cases} \quad \text{for some threshold } p_0 \in [0, 1].$$

- This yields one of the commonly used prediction methods in **machine learning**, which you may have had in one of your machine learning courses.

glm  
○○○

logistic regression  
○○○○○○○○○○○○○○○○○○○○○○

Poisson regression  
●○○○○○○

further designs  
○○○

## Poisson regression

# Setting and design

An experiment with:

- an **outcome**  $Y$  that is a count;
- one or more **numerical explanatory variables**  $X_1, \dots, X_p$ .
- one or more **factor explanatory variables**. ("independent variable").

The purpose is to explain  $Y$  by a **function** of  $X$ .

**EXAMPLE** The **number of plant species** on a Galapagos Island, with explanatory variables **area**, **highest elevation**, **distance to nearest island**, **distance to Santa Cruz island** and **area of adjacent island**.

**EXAMPLE** The **number of military coups** in some countries with explanatory variables **number of years country ruled by military oligarchy**, **number of political parties** and **population size**.

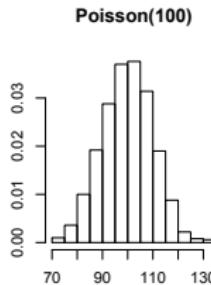
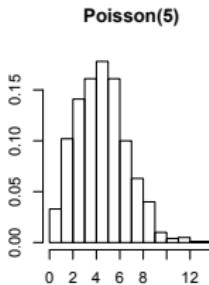
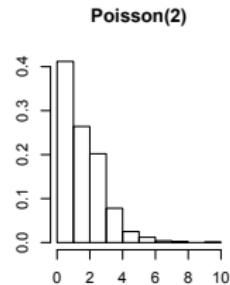
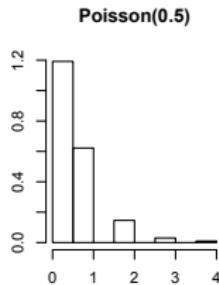
**Design.** Poisson regression can be used for factorial experiments, in a regression setting, for ANCOVA, and for experiments with blocks. The design is the same as for the corresponding experiment.

# The Poisson distribution

- A random variable  $Y$  is said to have the  $\text{Poisson}(\lambda)$ -distribution,  $\lambda > 0$ , if

$$P(Y = k) = \frac{\lambda^k}{k!} e^{-\lambda}, \quad k = 0, 1, 2, \dots$$

- If  $Y \sim \text{Poisson}(\lambda)$ , then  $E(Y) = \text{Var}(Y) = \lambda$ .
- Hence, the larger the parameter, the larger the values of  $Y$  on average and the larger the spread in the values of  $Y$ .
- For very large  $\lambda$ , the  $\text{Poisson}(\lambda)$ -distribution is approximately equal to a normal distribution with mean  $\mu = \lambda$  and variance  $\sigma^2 = \lambda$ .



# Analysis

- In Poisson-regression, the parameter  $\lambda$  is modelled as:

$$\log \lambda = \mu + \alpha_i + \dots + \beta_1 X_1 + \dots, \quad \text{or} \quad \lambda = e^{\mu + \alpha_i + \dots + \beta_1 X_1 + \dots},$$

on the right: the combination of (numerical and/or categorical) variables.

- For each  $Y_k$  the parameter  $\lambda_k$  is modelled differently, since the values of involved factors/predictors will differ for diff. observations:  $\lambda_k = e^{x_k^T \theta}$ .
- For example, for the Poisson regression with one factor (with  $I$  levels) and one continuous predictor  $X$ ,

$$Y_{ik} \sim \text{Poisson}(\lambda_{ik}), \quad \lambda_{ik} = e^{\mu + \alpha_i + \beta X_{ik}}, \quad i = 1, \dots, I, \quad k = 1, \dots, N.$$

- Hence, the variances are different as well. This means that the **response residuals**  $Y_{ik} - \hat{Y}_{ik} = Y_{ik} - e^{\hat{\mu} + \hat{\alpha}_i + \hat{\beta} X_{ik}}$  are not from one fixed distribution, hence a normal QQ-plot of these response residuals **is not relevant!**

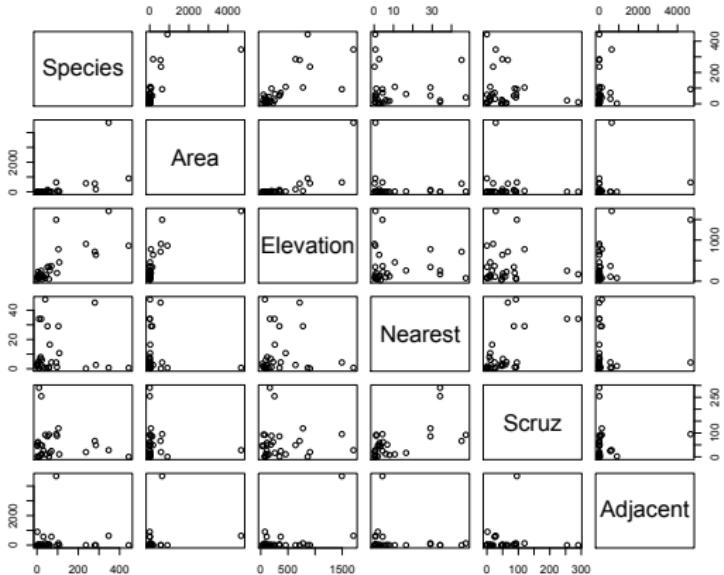
Instead, the **deviance residuals** are useful for diagnostic plots. **Deviance** is a measure of the discrepancy between the “full model” and the model under consideration. Deviance residuals are response residuals scaled by the deviance of that observation.

# Analysis in R: data input

The column Species of the data set gala.txt indicates the number of different plant species on the Galapagos island. The explanatory variables are Area (area of island), Elevation (highest elevation of island), Nearest (distance to nearest island), Scruz (distance to Santa Cruz) and Adjacent (area of adjacent island). All explanatory variables are numeric.

```
> gala=read.table("gala.txt",header=TRUE); gala
      Species   Area Elevation Nearest Scruz Adjacent
Baltra        58 25.09       346     0.6    0.6    1.84
Bartolome     31  1.24       109     0.6   26.3   572.33
Caldwell       3  0.21       114     2.8   58.7    0.78
Champion      25  0.10       46     1.9   47.4    0.18
Coamano        2  0.05       77     1.9    1.9   903.82
Daphne.Major   18  0.34       119     8.0    8.0    1.84
[ some output deleted ]
```

# Analysis in R: graphics



The problem of collinearity amongst explanatory variables is similar in nature as in the linear models case.

# Analysis in R: estimation and testing

```
> galaglm=glm(Species~Area+Elevation+Nearest+Scruz+Adjacent,  
+ family=poisson,data=gala)  
> summary(galaglm)  
[ some output deleted ]  
Coefficients:  
              Estimate Std. Error z value Pr(>|z|)  
(Intercept) 3.155e+00 5.175e-02 60.963 < 2e-16 ***  
Area        -5.799e-04 2.627e-05 -22.074 < 2e-16 ***  
Elevation   3.541e-03 8.741e-05 40.507 < 2e-16 ***  
Nearest      8.826e-03 1.821e-03   4.846 1.26e-06 ***  
Scruz       -5.709e-03 6.256e-04  -9.126 < 2e-16 ***  
Adjacent    -6.630e-04 2.933e-05 -22.608 < 2e-16 ***
```

The output of the function `glm` is an object to which functions as `anova`, `drop1`, `summary`, `coef`, `fitted`, `predict`, `confint`, etc. can be applied, in the **same way** as for the logistic and linear regressions. Remember that the interpretation of the **predicted responses** is of course **different**: for example, the predicted **responses** (i.e., estimate of  $EY_{in}$ ) for the Poisson regression with one factor (with  $I$  levels) and one contin. predictor  $X$  are  $\hat{Y}_{ik} = \hat{\lambda}_{ik} = e^{\hat{\mu} + \hat{\alpha}_i + \hat{\beta}X_{ik}}$ .

glm  
○○○

logistic regression  
○○○○○○○○○○○○○○○○

Poisson regression  
○○○○○○

further designs  
●○○

further designs

## Remarks en further designs

- Other GLM's for non-normal outcomes. Besides binomial and count data the `glm` function can also model multinomial, negative binomial, Gamma.
- The problems of identifiability, outliers, potential points, collinearity, checking model assumptions are inherent also for the GLM's.
- **Longitudinal analysis.** In longitudinal experiments one is interested in the development of individuals or other experimental units over time. This typically leads to multiple measurements per individual, taken at different time points (and often modeled with **mixed effects models**).
- **Mixed models.** Mixed models define outcomes in terms of **parameters**, **(random) errors** and additional **random effects**. This allows to model variation due to the selection of experimental units, fluctuations over time, extraneous variables that influence some measurements, etc.

# To finish

Today we discussed

- ① generalized linear models
  - ① logistic regression
  - ② Poisson regression