“TRAINING OF YOUNG ACADEMICS PROCESSING, ANALYZING AND INTERPRETING WILDLIFE SURVEY DATA”

Nairoby

16.11. – 28.11.2025

INTRODUCTION TO LINEAR MODELS

LINEAR MODELS

all the things we investigate are influenced by many factors for instance,

- the abundance of a species will be influenced by the availability of food, the abundance of predators and competing species, habitat type and structure, climate and weather, and, and, and

- the behaviour and physiology of individuals will be influenced by diurnal rhythms, seasonal variation in reproduction, weather, food availability, the presence or abundance and behaviour of predators and conspecifics, the individuals’ age and sex, experimental conditions, and, and, and

any particular observation we make is most likely influenced by multiple processes/factors

LINEAR MODELS

with the models we fit to our data, we try to do some kind of ’reverse engineering’: we fit models to our observations (i.e., the response) with the aim of better understanding the processes that shaped them the models, hence, reflect our understanding of the processes we investigate and the hypotheses we have about them

=> each model reflects a hypothesis (and this could be a complex one) about life

since whatever we observe will very rarely be influenced by a single factor (even in simple experimental studies), our models necessarily will be of some complexity

but this complexity simply reflects the complexity of life

LINEAR MODELS

with the linear models considered in this course, we model a single response as a (mathematical) function of one or several predictor variables

the response is the variable that we study (e.g., the behaviour, development, physiology, or cognition of individuals, the abundance of species…)

the predictors are the variables of which we assume that they influence the response (e.g., species, age, sex, rank, individuality, gene, education, pollution, hunting pressure, environmental gradients, experimental condition…)

the models built/investigated have the general form of

response ~ predictors (read as 'response as a function of predictors')

LINEAR MODELS

linear models allow for investigating…

- many phenomena

- pretty complex phenomena

such investigations are conducted using a unified, flexible and general framework/approach

the mathematical machinery behind them is a little complex (but you don't need to bother much about it)

but the models' complexity expresses the complexity of life/the world

'thinking' in terms of linear models helps thinking about life/the 'world'

BUT WHY BOTHERING?

why should one bother about the complexity of life when it comes to modelling effects of predictors?

can't one just estimate/test an effect of one predictor while ignoring all the possible effects of others?

do we really need to bother about the complexity of life (and deal with complex models) when we are only interested in the effect of single predictor?

yes, because it pays

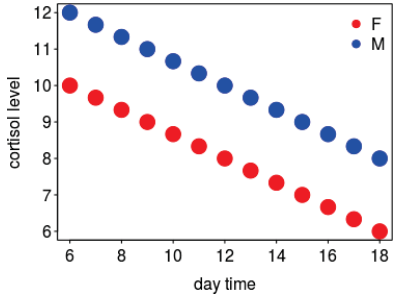
BUT WHY BOTHERING?

an example:

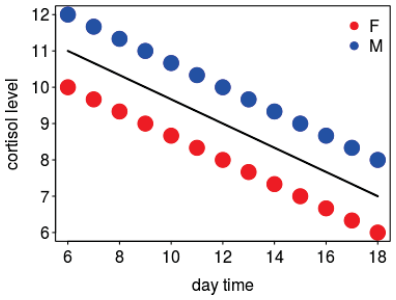
assume you want to estimate the effect of daytime on cortisol levels (and assume that these differ between females and males)

and assume a perfectly balanced sample

i.e., exactly the same sample size per combination of daytime and sex

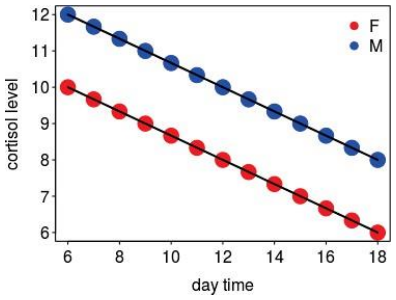


BUT WHY BOTHERING?



certainly one could estimate daytime effects ignoring sex differences

and this would perfectly estimate the decrease of cortisol levels over the course of the day

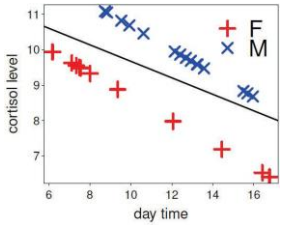


however, controlling for sex differences will equally certainly lead to a smaller residual variance, reduced uncertainty in the estimate, and larger power

(and the same applies to estimating the sex difference)

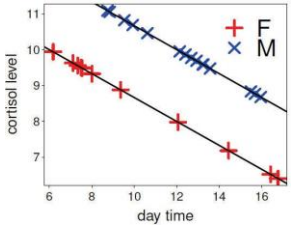
BUT WHY BOTHERING?

but when do we ever have such perfectly balanced data?



fitting a model not controlling for potential sex differences means to conflate day time effects with sex effects
such a model will not reveal a reliable estimate of day time effects

rather, what it reveals for the effect of day time is actually a mixture of the effects of day time and sex, because the two predictors are confounded



fitting a model controlling potential sex differences allows to estimate the day time effect with much higher precision
that's why we should bother!

MODULES TO BUILD A MODEL

each model applied combines certain modules particular to the question and data at hand

these are, e.g.,
- setting up the model structure
- setting up the data
- fitting the model
- dealing with assumptions and issues
- interpreting the results
- plotting the results

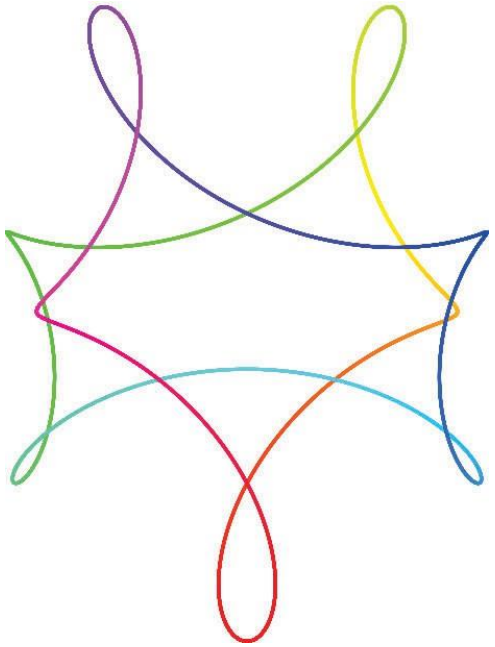
when it comes to your own models you may need to combine them in different ways

keep that in minds

MODEL

in the past, people chose the particular test that best matched their data design/data structure

today we have the freedom and the responsibility(!) to carefully design the model such that it appropriately reflects our data structure, adequately addresses the hypotheses/questions we have, and controls for potential confounders



TERMS, DEFINITIONS

throughout the course I distinguish between predictor and response variables

predictor (a.k.a. independent) variable:

can be set or selected by the experimenter / observer

response (a.k.a. dependent) variable:

is supposed to be influenced by the predictor variable(s)

throughout the course, we model the influence of one (or several) predictor(s) on a single response

PREDICTOR & RESPONSE VARIABLES

try to think of predictor and response variables in terms of cause and consequence or in terms of what comes first and what next

it is…

- food availability that might cause nutritional stress
- an experimental manipulation that might change an animal's behaviour
- hunting that impacts monkey abundance
- age that leads to an increase in experience
- species that determines cognitive capacity
- …

and not the other way round…

FACTORS & COVARIATES

predictors can be distinguished according to their scales

a predictor can be a…

- factor (i.e., qualitative or categorical)

  typical factors are species, sex, subject…

- covariate (i.e., quantitative)

  typical covariates are age, environmental gradients…

in the literature the term 'covariate' frequently encompasses covariates (as defined above) *and* factors

a factor has levels, i.e., the unique cases it has (e.g., 😐, ☺; bonobo, chimp, human); it can be two or more

SCALES OF THE RESPONSE

responses can be distinguished according to their scales

a response can be at…

- nominal scale (i.e., qualitative or categorical)
- ordinal scale (i.e., a rank)
- interval scale (i.e., numerical with arbitrary zero)
- ratio scale (i.e., numerical with natural zero)

in the course we will treat responses at interval/ratio scale (continuous and counts) as well as at nominal scale (binary response)

(the same categorization could be applied to predictors, but for this course a categorization into factors and covariates is sufficient)

DATA STRUCTURE & ORGANIZATION

fitting linear models means to deal with data

very generally, data need to come in the form of a single, cohesive and rectangular table with rows and columns

the colums represent the different 'variables'

the rows represent the different 'observations'

VARIABLES & DATA POINTS

variable

property known for a number of cases

'property': quantity or quality

can be set by the observer or simply be observed

is usually represented by one column in a spread sheet

different variables may be included in one spread sheet

case/data point

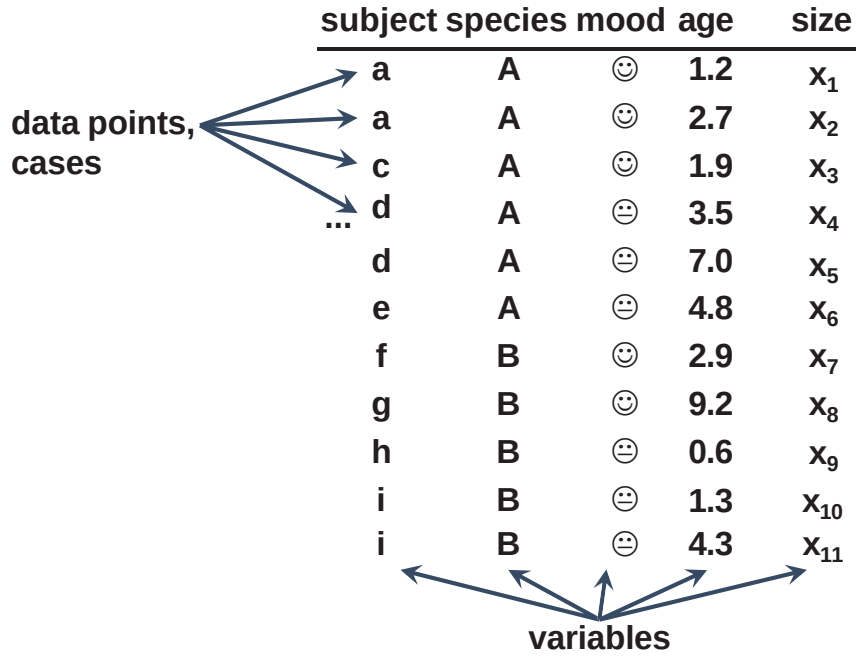
individual observation (of, e.g., one subject)

experimental or observational unit, e.g., subject, site, combination of subject and experimental treatment...

is represented by one row in a spreadsheet

one data point may comprise different variables

VARIABLES & DATA POINTS



GETTING STARTED

data (are in '02_water_cons.txt')

read them into R using

```
setwd("<…>")
xdata=read.table(file="02_water_cons.txt",
 header=T, sep="\t")
```

check `str(xdata)`

<u>motivation of the model</u>

a stats teacher drinks lots of water while teaching and the question is whether her/his water consumption is influenced by the number of participants

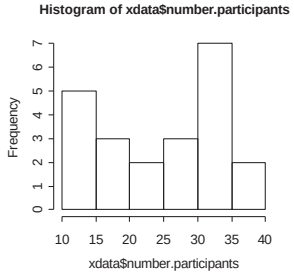
<u>data</u>

predictor: number.participants

response: liters.per.hour

total sample size: 22

INSPECT THE DATA



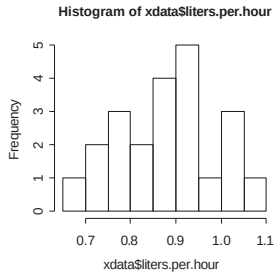
as a first step, inspect the distribution of the predictor and the response

`hist(xdata$number.participants)`

=> looks okay[(1)]

`hist(xdata$liters.per.hour)`

=> looks okay



[(1)] note that `plot(table(xdata$number.participants))` **would be another option for a variable comprising only integers**

THE FUNCTION `lm`

in R, simple and multiple regression can be fitted using the function `lm`

the function `lm`…

- builds general linear models[1] with any mixture of categorical and quantitative predictors
- runs regression, ANOVA and ANCOVA

general use:

indicate model formula in the form

response ~ predictor(s)

if variables are in a 'data frame' and you are lazy typing all variable names preceded by the name of the data frame, use argument `data`

[1] i.e., with Gaussian error structure and identity link (see lessons about GLM)

FIT THE MODEL

fit the model (for later use of the result, store it in an object named `res`):

```
res=lm(liters.per.hour~number.participants,
 data=xdata)
```

`res` now comprises the results of the call of the function `lm`

typing `res` indicates the estimated coefficients

but we need to know more (much more)…

first of all, model diagnostics which indicate whether the assumptions are fulfilled and whether the model is stable are important

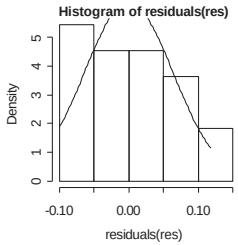
CHECKS OF ASSUMPTIONS

<u>normality of residuals</u>

get residuals using `residuals(res)`

check normality of residuals:

```
hist(residuals(res), probability=T)
```



add a line showing the respective normal distribution:

```
x=seq(from=min(residuals(res)),
 to=max(residuals(res)), length.out=100)
lines(x=x, y=dnorm(x, mean=0,
 sd=sd(residuals(res))))
```

are not really normal but neither very skewed nor with outliers (± okay)

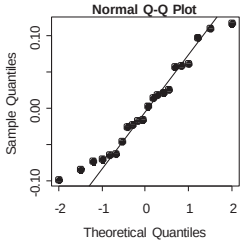
CHECKS OF ASSUMPTIONS

a qq-plot gives a safer view of the residual distribution

use

```
qqnorm(residuals(res)); qqline(residuals(res))
```

it suggests that small values are too large (compare also with histogram on previous page)



in an ideal case all points would fall on a straight line

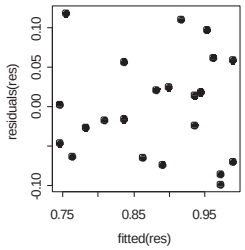
CHECKS OF ASSUMPTIONS

homogeneity of residuals

get fitted values using `fitted(res)`

plot residuals against the fitted values:

```
plot(x=fitted(res), y=residuals(res),
 pch=19)
```



no obvious 'pattern' should be visible

=> seems okay

see appendix for how to write a function revealing all the diagnostics plots at once

CHECKS OF ASSUMPTIONS

one may also correlate absolute residuals with fitted values

```
cor.test(fitted(res),  abs(residuals(res)))
```

reveals

```
...
t = 1.0632, df = 20, p-value = 0.3004
...
       cor
0.2312891
```

the correlation coefficient should be close to 0 and not significant…

=> seems okay

CHECKS OF ABSENCE OF INFLUENTIAL CASES

influence diagnostics, DFFit:

compare fitted values between model using all data and model with cases excluded one at a time

`dffits(res)`

reveals *standardized* DFFit-values

the argument (here `res`) is the result of a linear model

an absolute value > ~2 is a reason to worry

`max(abs(dffits(res)))` **reveals** `0.89`

get the distribution of the DFFits: `hist(dffits(res))`

in case you want DFFits on the scale of the response ask me

if you want to locate a single large or small value, use `which.max(dffits(res))`, **or** `which.min(dffits(res))`, **or** `which.max(abs(dffits(res)))`

`which.max` **returns the position in the vector where the maximum occurs**

CHECKS OF ABSENCE OF INFLUENTIAL CASES

influence diagnostics, DFBeta:

compare model coefficients between model using all data and model with cases excluded one at a time

`dfbeta(res)` **reveals *unstandardized* DFBeta values (difference between estimates derived from all data and with cases excluded one at a time)**

`head(dfbeta(res))` **reveals**

	(Intercept)	number.participants
1	-0.0029749	0.0000840
2	0.0009189	-0.0000820
3	-0.0024179	-0.0000443
4	-0.0048567	0.0000720
5	-0.0019383	0.0000464
6	0.0086773	-0.0005203

=> comprises one column for each estimated parameter and one row for each data point)

CHECKS OF ABSENCE OF INFLUENTIAL CASES

influence diagnostics, DFBeta:

to compare the estimated coefficients based on all data with the range of estimates derived from data dropping cases one at a time use

```
round(cbind(coefficients(res), coefficients(res)+
 t(apply(X=dfbeta(res), MARGIN=2, FUN=range))), 5)
```

reveals

	[,1]	[,2]	[,3]
(Intercept)	0.65519	0.63907	0.68802
number.participants	0.00901	0.00795	0.00953

columns are original, min, and max

=> looks good (little variation)

the function `dfbetas` reveals standardized DFBeta values (absolute values larger than 1 or 2 are a reason to worry)

CHECKS OF ABSENCE OF INFLUENTIAL CASES

influence diagnostics, Cook's distance:

`max(cooks.distance(res))` **reveals** `0.34`

several thresholds were recommended:

values > 1 are a reason to worry

values > 4/n (here 4/22=0.18) are reason to worry

values > $F_{k, n-k, 1-\alpha}$ are a reason to worry

can be determined using

```
qf(p=1-0.05, df1=2, df2=20, lower.tail=T)
```

which reveals `3.49`

CHECKS OF ABSENCE OF INFLUENTIAL CASES

influence diagnostics, leverage:

`max(as.vector(influence(res)$hat))` **reveals** `0.17`

values > 2*(k+1)/n or > 3*(k+1)/n are a reason to worry[1]

n = number of cases (here 22, or `length(residuals(res))`)

reveals `0.18` **or** `0.27`**, respectively**

=> is okay

[1] see the appendix for a function determining the threshold

CHECKS FOR ASSUMPTIONS AND INFLUENTIAL CASES

generally, also plots of the various diagnostics might be helpful

```
plot(residuals(res))
plot(dffits(res))
plot(dfbeta(res)[,1])
```

etc. (needs to be done by column)

```
plot(cooks.distance(res))
plot(as.vector(influence(res)$hat))
```

a potentially interesting function is `identify(<…>)`

if called after such a plot is produced, one can click (yeah!) on the points and it will indicate the position of the point in the vector

on Windows/MacOS press esc in the R GUI window, on Linux close the plotting window to turn back to normal working mode

SAVING THE WORKSPACE

for now we are fine with the model:

=> no obvious violations of its assumptions

=> no obviously influential cases

but what are the results?

some more theory is needed for that…

but before that:

save the workspace (we will later need to continue working with it):

```
getwd()
```

```
save.image("water_cons.RData")
```
(don't forget to add the extension!)

SUMMARY

before running a regression:

- inspect the distributions of the predictor(s)

  potentially transform those that are skewed (otherwise larger (or smaller) values might have much leverage)

- inspect the distribution of the response

  potentially transform it (but ultimately the distribution of the *residuals* matters, not the distribution of the response)

SUMMARY

after running a regression:

- inspect the distribution of the residuals (using a qq-plot and potentially a histogram)
- inspect the residuals plotted against the fitted values (no pattern should be discernable)

  potentially transform the response (and rerun the model)
- inspect influence diagnostics

  if influential cases do exist: don't just delete them! Try to figure out what caused them to be so different (typo? forgotten predictor?)

  potentially run an additional model with the influential cases removed from the data and compare the results (and hope that they don't differ much)

SUMMARY

take all the assumptions really serious:

violated assumptions may lead to grossly invalid results

before having checked them (and having ensured that they are fulfilled) it does not make sense to inspect the results of the model

EXERCISE

write a function revealing all three thresholds for Cook's distance

tip: you can return multiple values in a named vector

APPENDIX

a function that creates all diagnostics plots at once:

```
diagnostics.plot<-function(mod.res){
  old.par = par(no.readonly = TRUE)
  par(mfrow=c(2, 2))
  par(mar=c(3, 3, 1, 0.5))
  hist(residuals(mod.res), probability=T, xlab="",
    ylab="", main="")
  mtext(text="histogram of residuals", side=3, line=0)
  x=seq(min(residuals(mod.res)), max(residuals(mod.res)),
    length.out=100)
  lines(x, dnorm(x, mean=0, sd=sd(residuals(mod.res))))
  qqnorm(residuals(mod.res), main="", pch=19)
  qqline(residuals(mod.res))
  mtext(text="qq-plot of residuals", side=3, line=0)
  plot(fitted(mod.res), residuals(mod.res), pch=19)
  abline(h=0, lty=2)
  mtext(text="residuals against fitted values", side=3,
    line=0)
  par(old.par)
}
```

APPENDIX

a function that determines the threshold for leverage:

```
lev.thresh<-function(model.res){
 k=length(coefficients(model.res))
 n=length(residuals(model.res))
 return(2*(k+1)/n)
}
```

if you save these two functions in a file, you can make them accessible to R loading the file

e.g., if they are in 'my_fcns.r', use

```
setwd("<…>")
source("my_fcns.r")
```

then the functions can be used as any other R function by just calling them with a model results object