ST2304 - Statistical Modelling for Biologists/Biotechnologists

Bob O'Hara

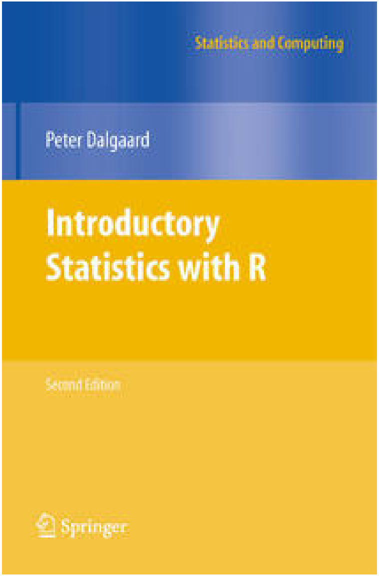
bob.ohara@ntnu.no

Administration Matters

- Reference Group
- Blackboard

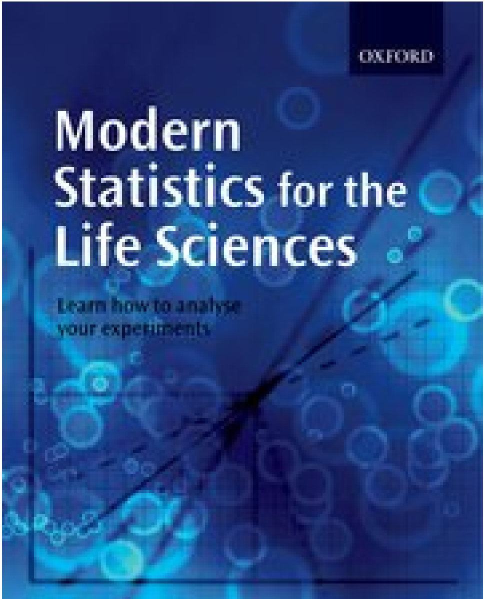
Text Books

Introductory statistics with R by Peter Dalgaard



Text Books

Grafen & Hails: Modern Statistics for the Life Sciences



Previous Knowledge

What do you know?

(so I don't assume too much!)

Mathematics

- logarithms/exponentials
- differeniation
- integration

Probability

- Conditional Probability
- Expectation E(X)
- Indepedence
- Probability distribution

Statistics

- Averages
- Variance, standard deviation
- Covariance, correlation
- t-tests
- regression (fitting a straight line)

Programming

- Languages
  - C, BASIC, Python, Java, Python
  - R
- Object Oriented Programming

Why do we need statistical modelling?

Because the world is complex!

Face width and Fitness

Men with wider faces are more aggressive

But does this affect their fitness?

- survival in the Winter War
- number of (legitimate) offspring

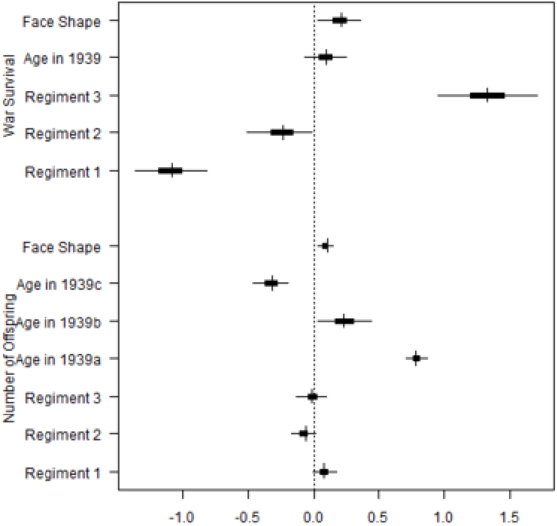
Face width and Fitness

But other things affect fitness - regiment - rank - survival affects number of offspring

We need a model to disentangle all these

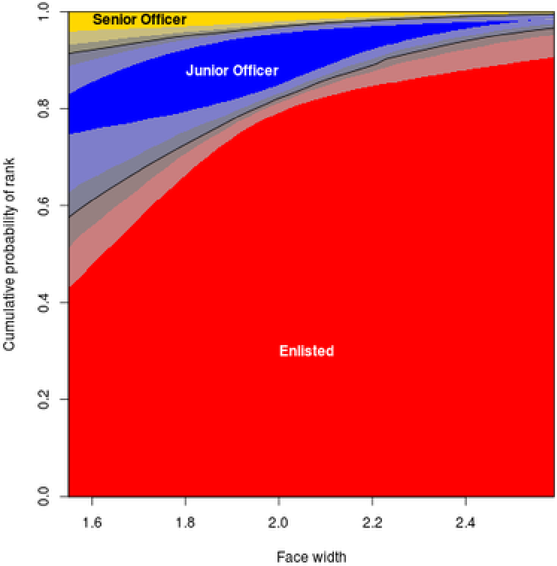
Face width and Fitness

Men with wider faces had more offspring, and might have slightly lower survival



Face width and Fitness

But thinner face -> higher rank



Statistical Modelling: Brain size in birds

- What sort of question can we ask?
- What is likely to influence brain size?
  - actual brain size
  - what we measure

Data Analysis

Different stages

Outline the full work flow

Modelling is only a part of this!

First, collect the data

Figure 5:

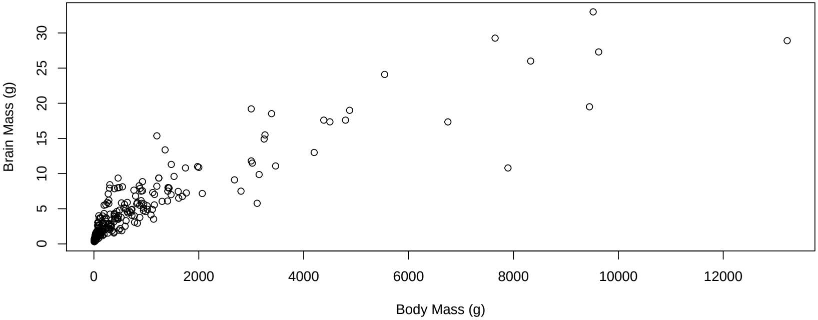
Format the data

- Get it into the right format so you can use it
- For big problems, use a data base
- Store the data in a convenient format, e.g. .csv

```
BirdBrains <- read.csv("ece32961-sup-0001-TableS1.csv")
names(BirdBrains) <- gsub("Brain.*", "Brain", names(BirdBra
names(BirdBrains)[grep("resou", names(BirdBrains))] <- "Boo
```

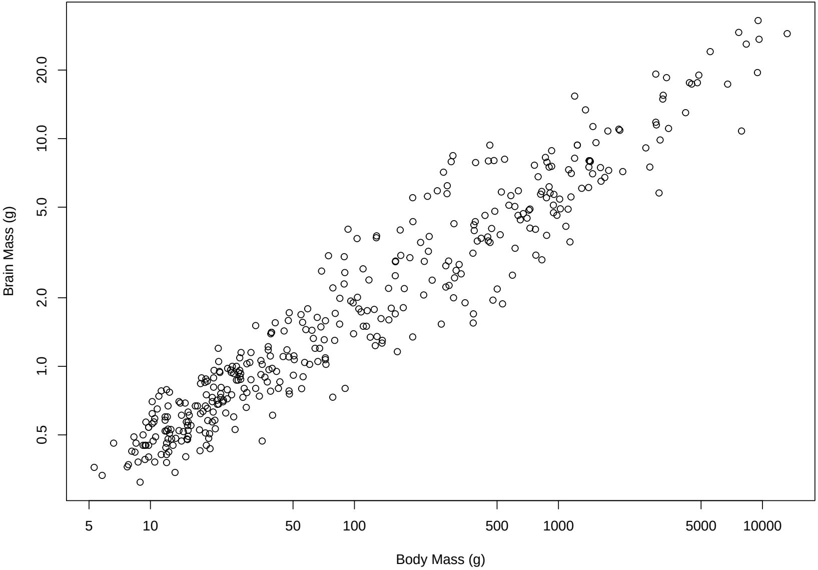
Look at the data

- Plot, calculate some summaries
- Spot any possible problems
  - Typos etc.



Look at the data

Nicer as a log-log plot?



Start the modelling

- What are the questions you are asking?
- What are problems in the data that you need to worry about?
- Do we need additional data?

Start the modelling

How does body size affect brain size in birds? - is is similar to mammals?

Does the relationship differ between different groups of bird?

Fit the model

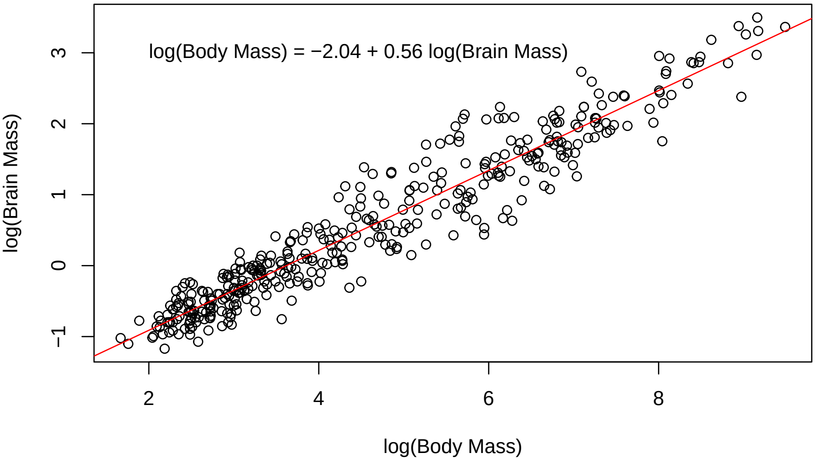
Get the computer to do it Look at the output

```
bird.mod <- lm(log(Brain) ~ log(Body), data=BirdBrains)
summary(bird.mod)
```

```
##
## Call:
## lm(formula = log(Brain) ~ log(Body), data = BirdBrains)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.87213 -0.20022 -0.01783  0.19395  0.94904
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.037838   0.043114  -47.27   <2e-16 ***
## log(Body)    0.563159   0.008625   65.29   <2e-16 ***
## ---
```

The Fitted Model

```
par(mfrow=c(1,1), mar=c(4.1,4.1,1,1))
plot(log(BirdBrains$Body), log(BirdBrains$Brain), type="p",
abline(bird.mod, col=2)
text(2, 3, paste0("log(Body Mass) = ", round(coef(bird.mod)
```

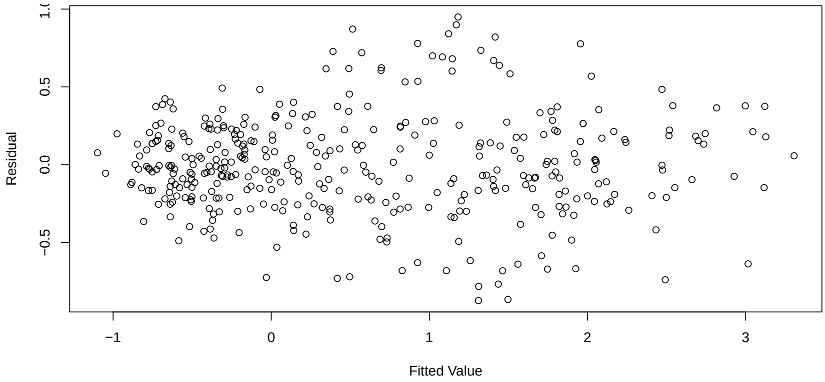


Check the Model

- Does it fit well?
- Outliers?
- Assumptions OK? e.g. linearity
- Can you understand the results?

Check the Model

```r
par(mar=c(4.1,4.1,0.1,1))
plot(fitted(bird.mod), resid(bird.mod), type="p", xlab="Fit
```



Interpret the model

Back to biology!

The slope is 0.56, with a 95% confidence interval of 0.55 to 0.58.

This is less than it is for mammal, where the slope is 0.75 (95% confidence interval: 0.69 to 0.81

Part of the reason for this is variation between bird orders. If we correct for order, the slope is 0.66, 95% confidence interval: 0.64 to 0.68.

A Quick Overview of R

A statistics programme

The language (S) designed for statistics

Object oriented

Interpreted language

Basic Syntax: Assignment

```
x <- 2
x
```

```
## [1] 2
```

```
(y <- 1:3)
```

```
## [1] 1 2 3
```

```
(z <- c(4.5,6.7,-3))
```

```
## [1]  4.5  6.7 -3.0
```

A Calculator

```
x + y
```

```
## [1] 3 4 5
```

```
exp(z)
```

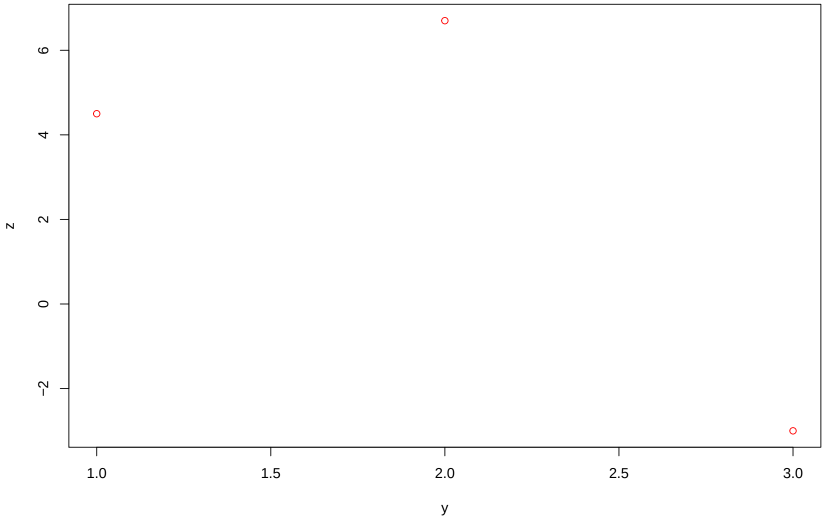
```
## [1]  90.01713130 812.40582517   0.04978707
```

```
z/y # = z[1]/y[1] z[2]/y[2] z[3]/y[3]
```

```
## [1]  4.50  3.35 -1.00
```

Simple Plotting

```
plot(y, z, col=2)
```



Functions & Objects

exp(z) & plot(y, z) are functions

y & z is an *arguments*: they are given to the function, which does something with them

The function returns an object: - exp() returns a vector - plot() returns NULL (i.e. nothing: the plot is a side-effect)

Statistical Models

lm() is a more complicated function

We give it 'log(Brain) $\sim$ log(Body)' and BirdBrains as arguments

```
bird.mod <- lm(log(Brain) ~ log(Body), data=BirdBrains)
```

It returns something horrid

A big object

```r
str(bird.mod)
```

```
## List of 12
##  $ coefficients : Named num [1:2] -2.038 0.563
##   ..- attr(*, "names")= chr [1:2] "(Intercept)" "log(Bod
##  $ residuals    : Named num [1:384] 0.2133 0.0616 0.1794
##   ..- attr(*, "names")= chr [1:384] "1" "2" "3" "4" ...
##  $ effects      : Named num [1:384] -11.196 -20.705 0.16
##   ..- attr(*, "names")= chr [1:384] "(Intercept)" "log(B
##  $ rank         : int 2
##  $ fitted.values: Named num [1:384] 1.81 1 3.13 2.68 2.2
##   ..- attr(*, "names")= chr [1:384] "1" "2" "3" "4" ...
##  $ assign       : int [1:2] 0 1
##  $ qr           :List of 5
##   ..$ qr   : num [1:384, 1:2] -19.596 0.051 0.051 0.051
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : chr [1:384] "1" "2" "3" "4" ...
##   .. .. ..$ : chr [1:2] "(Intercept)" "log(Body)"
```

The power of objects

We usually don't need to know what is in an object: there are functions to get us what we want

```
coef(bird.mod)
```

```
## (Intercept)   log(Body)
##  -2.0378376   0.5631593
```

```
summary(bird.mod)$coefficients
```

```
##               Estimate  Std. Error   t value      Pr(>|t
## (Intercept) -2.0378376 0.043113631 -47.26667 1.110796e-1
## log(Body)    0.5631593 0.008625092  65.29314 2.544019e-2
```

Data Types

Our basic data can be of diffeent types, e.g. integer, real number, text, factor, logical

```
(txt <- c("thing", "stuff", "thing")) # text
```

```
## [1] "thing" "stuff" "thing"
```

```
(f <- factor(c("thing", "stuff", "thing"))) # factor
```

```
## [1] thing stuff thing
## Levels: stuff thing
```

```
(lg <- c(TRUE, FALSE)) # logical
```

```
## [1]  TRUE FALSE
```

Using data types: indices

```r
(a1 <- -5:5)
```

```
##  [1] -5 -4 -3 -2 -1  0  1  2  3  4  5
```

```r
a1[1]
```

```
## [1] -5
```

```r
a1[c(3,5,6)]
```

```
## [1] -3 -1  0
```

Using data types: subsetting

```
(logical.a1 <- a1 > 0)
```

```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TI
```

```
(logical.a2 <- a1%%2==0)
```

```
##  [1] FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FAI
```

```
a1[logical.a1]
```

```
## [1] 1 2 3 4 5
```

```
a1[logical.a1 & logical.a2]
```

```
## [1] 2 4
```

Running an analysis

R has functions to read in data

```
MammalBrains <- read.table("mammals.dat", header = TRUE)
```

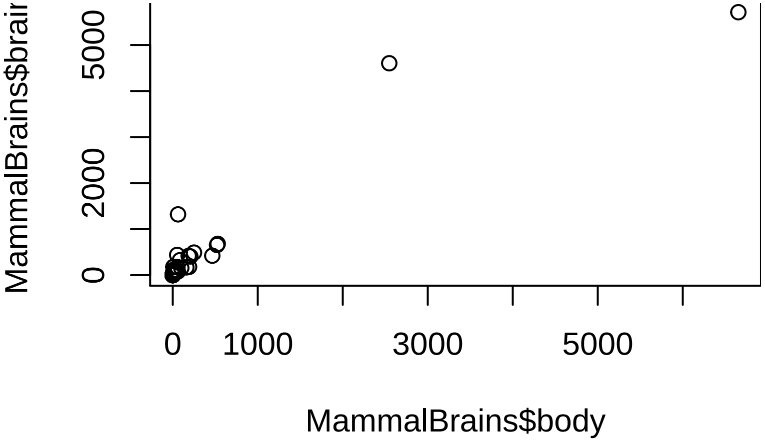
and write it out

```
write.csv(MammalBrains, "mammals.csv")
save(MammalBrains, x, file="mammals.RData")
```

Saving plots

We can save plots in files: pdf, png, jpg, etc etc

```
par(mar=c(4.1,4.1,0,0))
plot(MammalBrains$body, MammalBrains$brain)
```




```
png("MammalBrains.png")
```

Functions

We can write our own functions. . . .

```
HelloWorld <- function(str) {
  if(!is.character((str)))
    stop("str should be a character string you idiot")
  cat(paste0("Hello World, ", str, "!\n"))
}
HelloWorld("Orpheus")
```

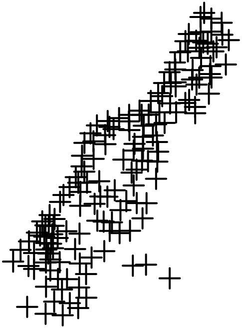
```
## Hello World, Orpheus!
```

Packages

Packages put together a lot of functions (and objects etc.)

A lot of these are on CRAN

```
library(sp)
data(meuse)
coordinates(meuse) <- c("x", "y")
par(mar=c(2,2,1,1))
plot(meuse)
```



Getting and Using R

You can download R from CRAN: `http://www.r-project.org/`

A lot of people use RStudio: `http://rstudio.org/`

Finally

If you want to start using R now:

`https://www.math.ntnu.no/emner/TMA4268/2018v/1Intro/Rbeginner.html`