Aalto University
School of Science

Descriptive statistics and plotting the data

Experimental and Statistical Methods in Biological Sciences I

Heini Saarimäki, BECS
25/09/2014

Aalto University
School of Science

Outline for today

Now…

- 1. Starters
- 2. Descriptive statistics
  - Categorical variables
  - Numeric variables
- 3. Plotting

Then exercises.

Practical tips # 1:

- `attach/detach(data)` can simplify things: e.g. instead of `summary(ageweight$WEIGHT)`, you can use `summary(WEIGHT)`
  - Useful also when running similar models on multiple datasets, e.g., before and after cleaning a dataset
  - Careful when modifying variables: if you modify a variable in the original data frame, remember to attach it again. Better to attach the data frame after you have prepared it (i.e., factors are factors, missing values coded as NA's).

Practical tips # 2:

- No need to overwrite data:
  - Save changes to a NEW dataset, e.g. data2, data3, data4, …
  - This way you can then review and see what impact your changes had

What we know so far

1. Preparing your data

When preparing your data, check that…

- Factors are factors
- Missing values are coded as NAs

What we know after today

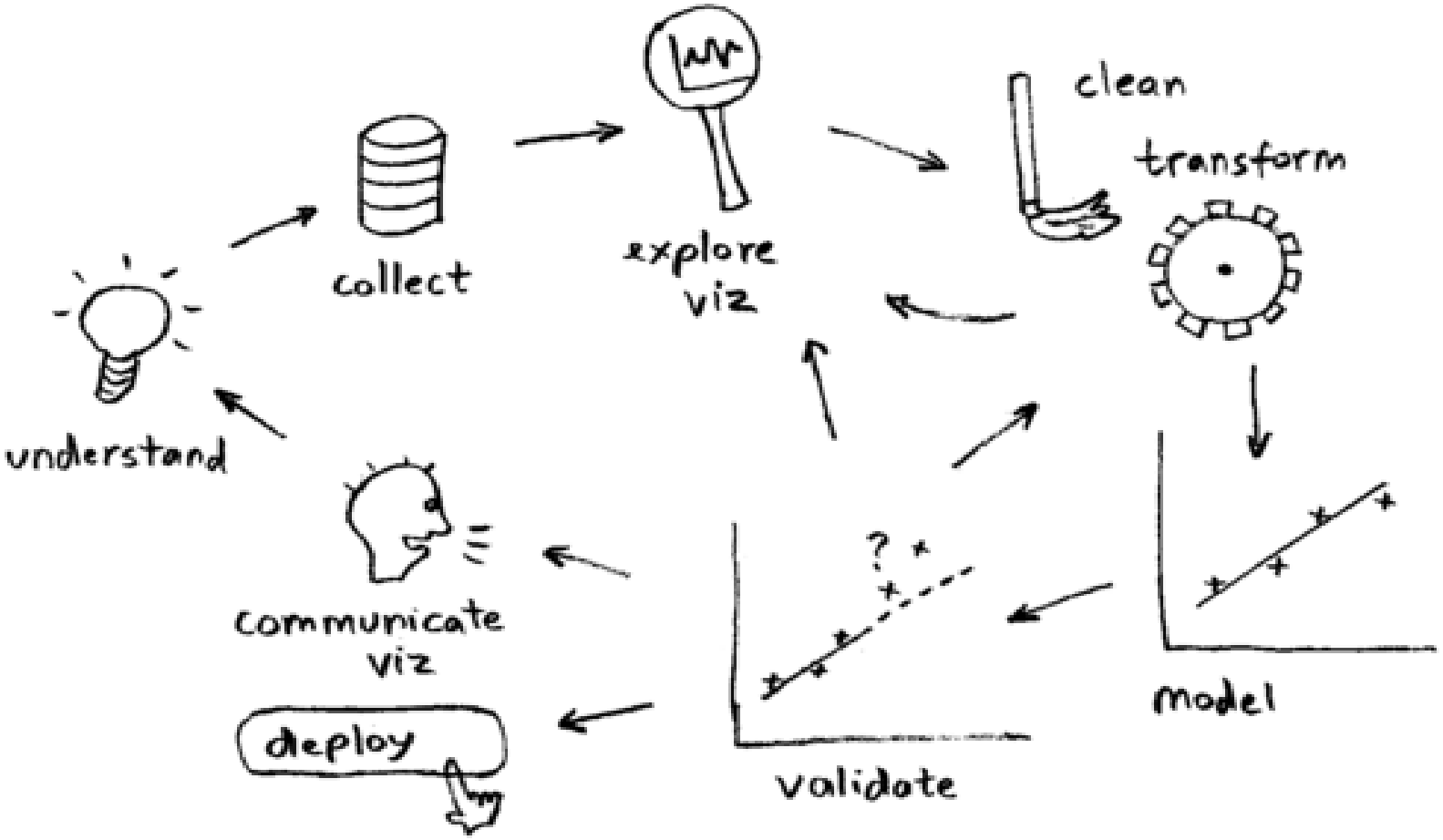
1. Preparing your data

When preparing your data, check that…

- Factors are factors
- Missing values are coded as NAs

2. Descriptive statistics & plotting

- How to examine, describe and present your data using plots and descriptive statistics



From datascience.la

2. Descriptive statistics

- Get an idea how your data looks like!
- Some descriptive statistics also necessary when reporting your data
  e.g., mean age of participants

2. Descriptive statistics

- Categorical variables: frequencies
  - Contingency tables

```
> table(MALE, SMOKE1)
    SMOKE1
MALE no yes
   0 23  29
   1 29  19
```

```
> round(prop.table(table(MALE,SMOKE1), margin=1)*100,1)
    SMOKE1
MALE   no  yes
   0 44.2 55.8
   1 60.4 39.6
```

2. Descriptive statistics

- Numeric variables: statistics to present central tendency and dispersion in your data
  - `summary()` shows us some details…

```
> summary(ageweight)
       X                MALE             AGE             WEIGHT          SMOKE1    SMOKE2       HEIGHT     
 Min.   :  1.00   Min.   :0.00   Min.   :20.00   Min.   : 27.53   no :52   no :43   Min.   :1.500  
 1st Qu.: 25.75   1st Qu.:0.00   1st Qu.:28.75   1st Qu.: 55.24   yes:48   yes:57   1st Qu.:1.652  
 Median : 50.50   Median :0.00   Median :42.50   Median : 78.23                     Median :1.731  
 Mean   : 50.50   Mean   :0.48   Mean   :42.44   Mean   : 76.00                     Mean   :1.727  
 3rd Qu.: 75.25   3rd Qu.:1.00   3rd Qu.:55.00   3rd Qu.: 93.96                     3rd Qu.:1.804  
 Max.   :100.00   Max.   :1.00   Max.   :65.00   Max.   :121.70                     Max.   :1.999  
                                                 NA's   :3
```

2. Descriptive statistics

- Numeric variables: statistics to present central tendency and dispersion in your data
  - … `describe()` from 'psych' package is more comprehensive

```
> describe(ageweight)
         var   n  mean    sd median trimmed   mad   min   max range  skew kurtosis   se
X          1 100 50.50 29.01  50.50   50.50 37.06  1.00 100.0 99.00  0.00    -1.24 2.90
MALE       2 100  0.48  0.50   0.00    0.48  0.00  0.00   1.0  1.00  0.08    -2.01 0.05
AGE        3 100 42.44 13.74  42.50   42.46 19.27 20.00  65.0 45.00 -0.03    -1.36 1.37
WEIGHT     4  97 76.00 23.99  78.23   76.15 29.16 27.53 121.7 94.17 -0.06    -0.95 2.44
SMOKE1*    5 100  1.48  0.50   1.00    1.48  0.00  1.00   2.0  1.00  0.08    -2.01 0.05
SMOKE2*    6 100  1.57  0.50   2.00    1.59  0.00  1.00   2.0  1.00 -0.28    -1.94 0.05
HEIGHT     7 100  1.73  0.10   1.73    1.73  0.11  1.50   2.0  0.50  0.14    -0.44 0.01
```

2. Descriptive statistics

- Numeric variables:
  - central tendency

```
> describe(ageweight)
         var   n  mean    sd median trimmed   mad   min   max range  skew kurtosis   se
X          1 100 50.50 29.01  50.50   50.50 37.06  1.00 100.0 99.00  0.00    -1.24 2.90
MALE       2 100  0.48  0.50   0.00    0.48  0.00  0.00   1.0  1.00  0.08    -2.01 0.05
AGE        3 100 42.44 13.74  42.50   42.46 19.27 20.00  65.0 45.00 -0.03    -1.36 1.37
WEIGHT     4  97 76.00 23.99  78.23   76.15 29.16 27.53 121.7 94.17 -0.06    -0.95 2.44
SMOKE1*    5 100  1.48  0.50   1.00    1.48  0.00  1.00   2.0  1.00  0.08    -2.01 0.05
SMOKE2*    6 100  1.57  0.50   2.00    1.59  0.00  1.00   2.0  1.00 -0.28    -1.94 0.05
HEIGHT     7 100  1.73  0.10   1.73    1.73  0.11  1.50   2.0  0.50  0.14    -0.44 0.01
```

2. Descriptive statistics

- Numeric variables:
  - dispersion

```
> describe(ageweight)
         var   n  mean    sd median trimmed   mad   min   max range  skew kurtosis   se
X          1 100 50.50 29.01  50.50   50.50 37.06  1.00 100.0 99.00  0.00    -1.24 2.90
MALE       2 100  0.48  0.50   0.00    0.48  0.00  0.00   1.0  1.00  0.08    -2.01 0.05
AGE        3 100 42.44 13.74  42.50   42.46 19.27 20.00  65.0 45.00 -0.03    -1.36 1.37
WEIGHT     4  97 76.00 23.99  78.23   76.15 29.16 27.53 121.7 94.17 -0.06    -0.95 2.44
SMOKE1*    5 100  1.48  0.50   1.00    1.48  0.00  1.00   2.0  1.00  0.08    -2.01 0.05
SMOKE2*    6 100  1.57  0.50   2.00    1.59  0.00  1.00   2.0  1.00 -0.28    -1.94 0.05
HEIGHT     7 100  1.73  0.10   1.73    1.73  0.11  1.50   2.0  0.50  0.14    -0.44 0.01
```

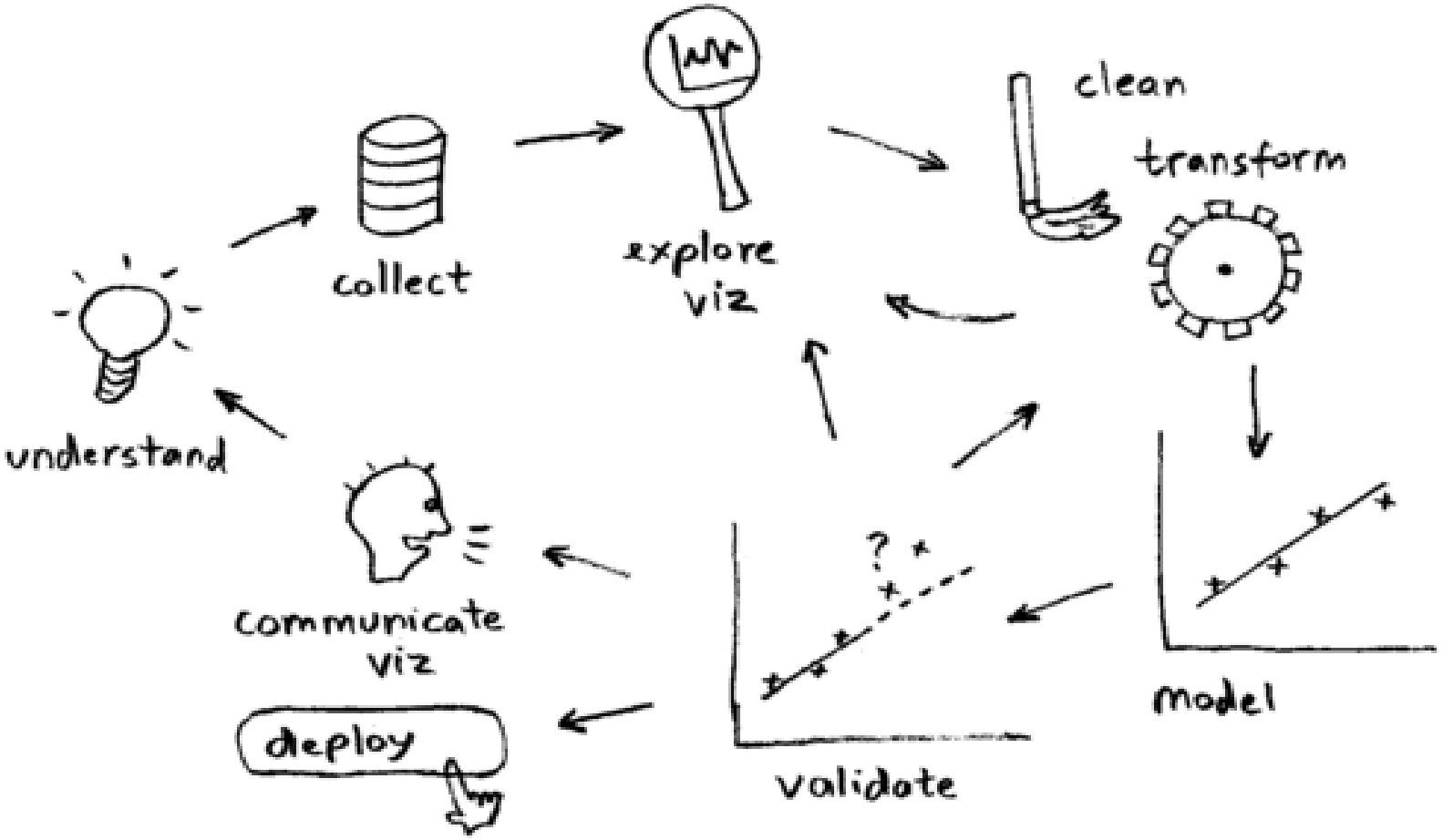
2. Descriptive statistics

- Numeric variables:
  - Use `tapply()` to get descriptive statistics separately for categories
  - E.g. mean and standard deviation:

```
> tapply(WEIGHT, list(MALE, SMOKE1), mean, na.rm=T)
        no      yes
0 64.12043 62.01714
1 87.39000 95.23833
> tapply(WEIGHT, list(MALE, SMOKE1), sd, na.rm=T)
        no      yes
0 19.67280 20.39246
1 21.00115 16.42853
```

2. Descriptive statistics

- Exporting your tables:
  - To be modified outside R:
    `write.csv()`
  - Export directly to LaTeX:
    stargazer package



From datascience.la

3. Plotting

The power of data visualization:

If you are not familiar with Hans Rosling yet, check out
www.ted.com/talks/hans_rosling_shows_the_best_stats_you_ve_ever_seen.html

3. Plotting

- If you want to generate all the plots mentioned in the lectures, navigate to Frank McCown's R page:
  - www.harding.edu/fmccown/r

- We will practice the most important ones:
  - Bar chart
  - Box plot
  - Histogram

Introduction to graphics

- With R, it is possible to produce publication-quality graphics with ease.
- Looking at your data is as important as analyzing it. R has many tools for doing just that
- `?plot, ?par, ?hist` etc.
- Try adding commands to the graphics bit-by-bit and see what changes.

Introduction to graphics

Remember the ’R way’: create objects, and then explore them? Also graphics are object-driven.

Introduction to graphics

Remember the 'R way': create objects, and then explore them? Also graphics are object-driven.

1. Define your basic format:

```
> plot(variable1, variable2)
> hist(variable1)
```

Introduction to graphics

Remember the 'R way': create objects, and then explore them? Also graphics are object-driven.

1. Define your basic format:

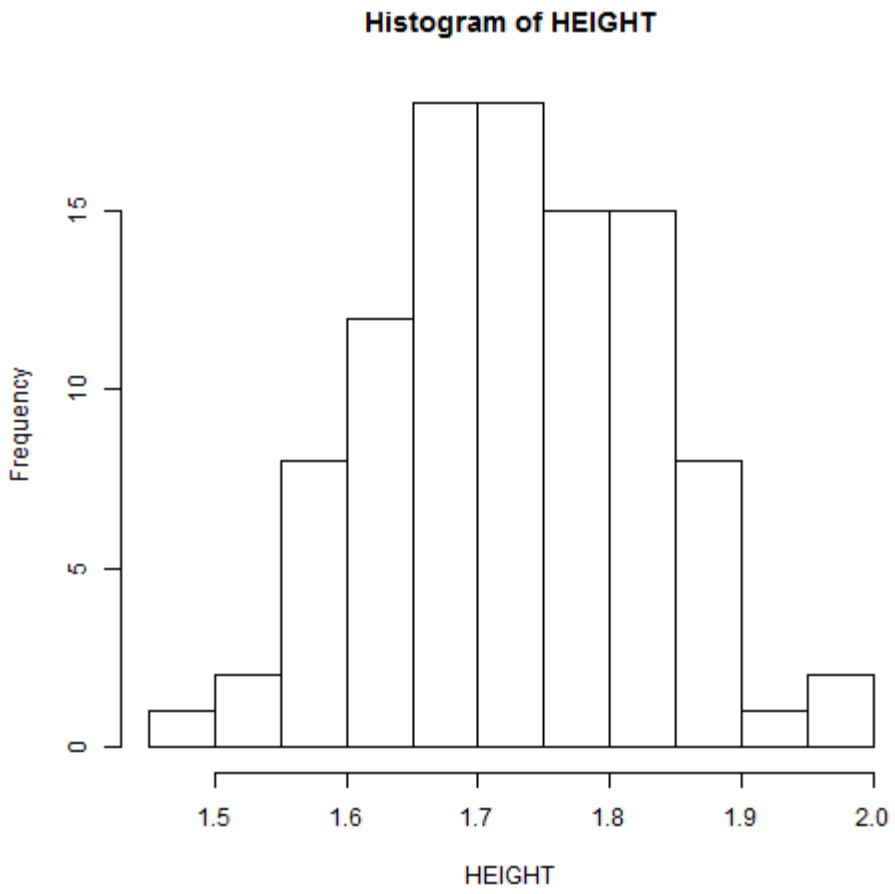
```
> plot(variable1, variable2)
> hist(variable1)
```

2. Then build it to a nice graph with **named arguments**:

```
> plot(variable1, variable2, col='red', main="My graph",
    xlab='variable 1', ylab = 'variable 2', lty='dashed')
```

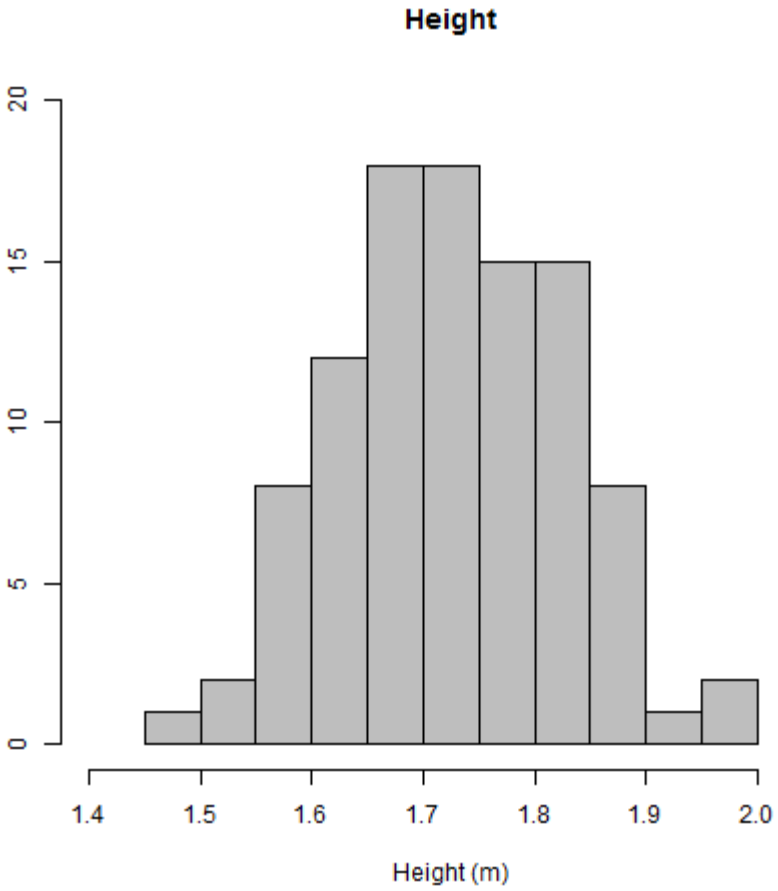
And so forth. Explore!

Examples: histogram



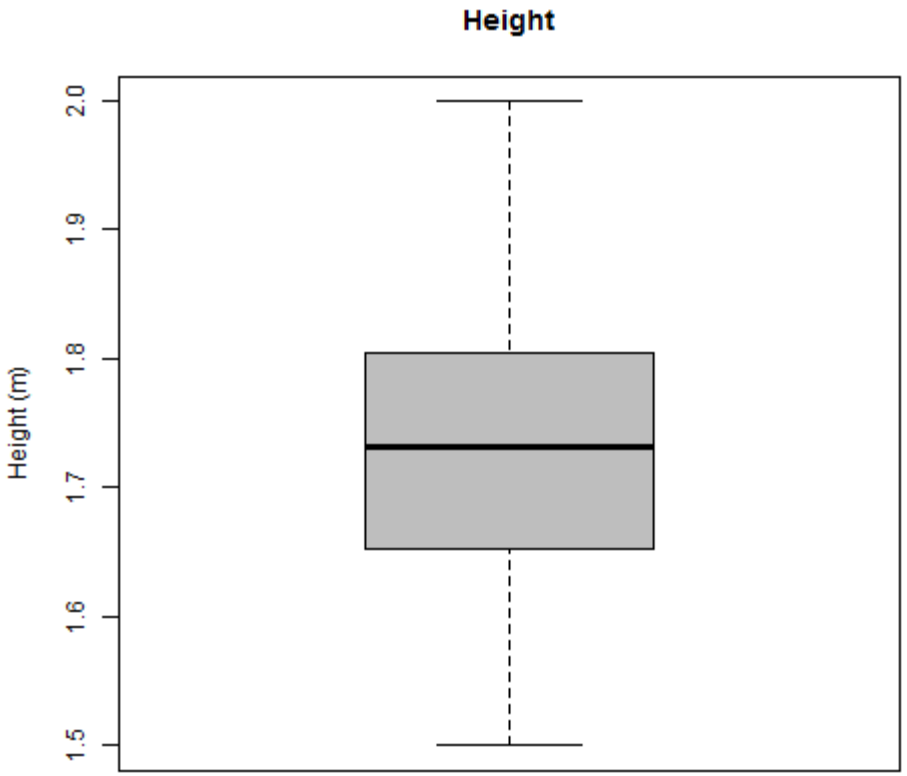

```
> hist(ageweight$HEIGHT)
```

Examples: boosted histogram



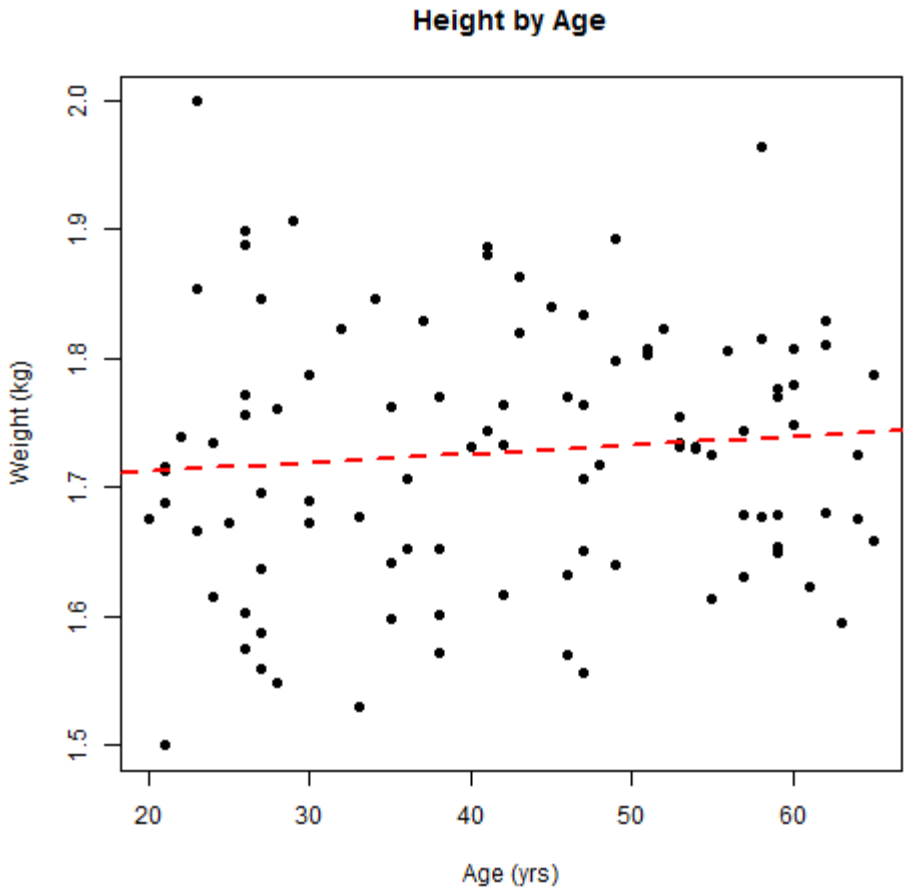

```
> hist(ageweight$HEIGHT,
    main="Height",
    xlab="Height (m)",
    ylab="",
    breaks=15,
    xlim=c(1.4,2.05),
    ylim=c(0,20),
    col="grey"
    )
```

Examples: boxplot




```
> boxplot(ageweight$HEIGHT,
    main="Height",
    xlab="",
    ylab="Height (m)")
```

Examples: scatterplot




```
> plot(HEIGHT~AGE,
    data=ageweight,
    main="Height by Age",
    xlab="Age (yrs)",
    ylab="Height (m)",
    pch=16
    )
> abline(lm(WEIGHT~AGE),
    data=ageweight,
    lty="dashed",
    lwd=2,
    col="red"
    )
```

Examples: plot by ggplot2

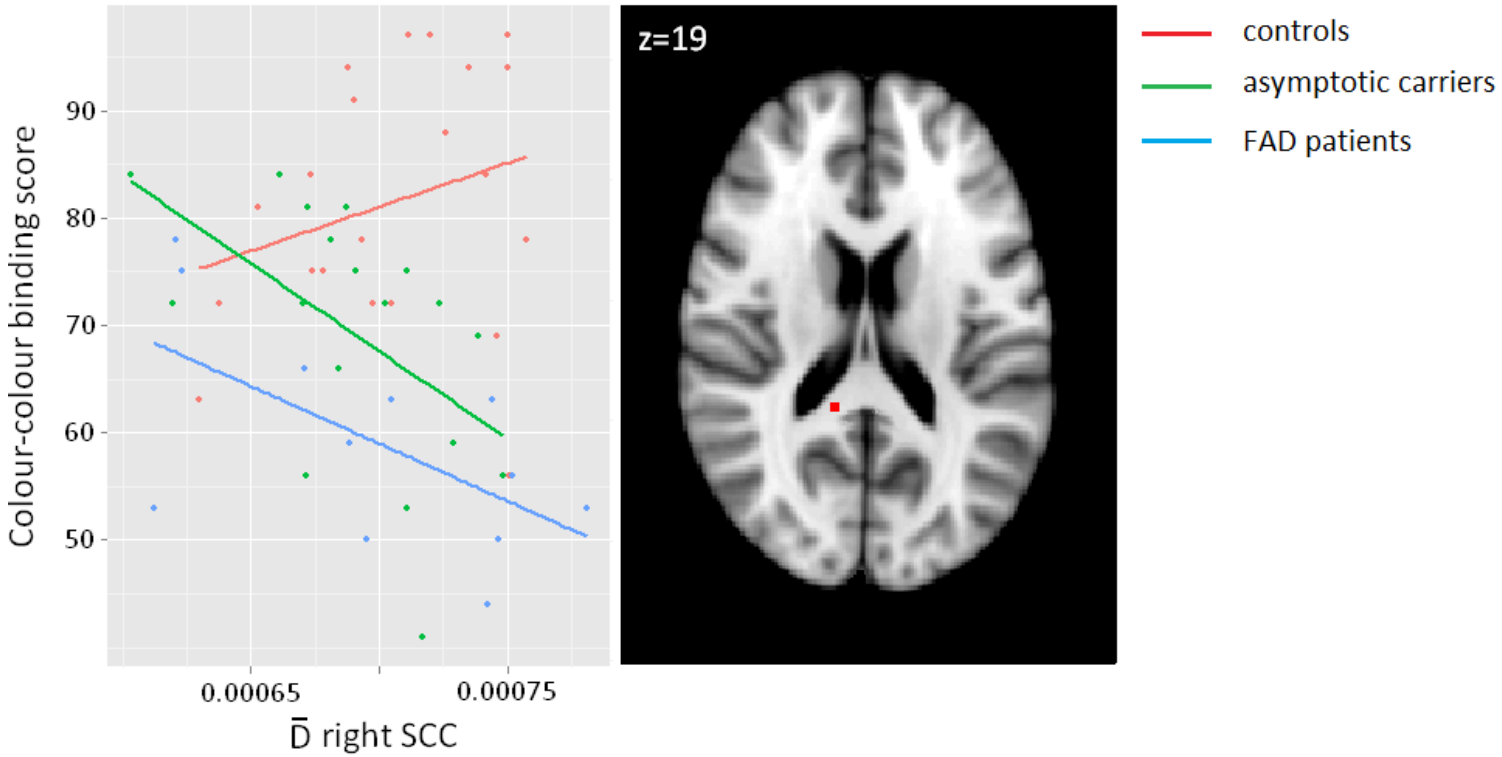


Figure 1. Significant linear regression models for the colour-colour binding task. Left: fitted regression lines. Right: the corresponding ROI in standard space.

Other graphics tools

- ggplot2 package
  - Great tool for publication-quality graphs!
  - Documentation: http://docs.ggplot2.org

- Ggvis (http://ggvis.rstudio.com/)
  - Documentation: http://docs.ggplot2.org

- Remember to look for packages…
  - E.g., violin plots (vioplot package)

Saving your graphics

When R creates graphics it writes to a 'device' (by default, the screen). To get R to save your graphics into a file, you need to

1. Switch the device to a file (the command you use depends on the graphics format you want: pdf is good, or you can find a whole list with ?Devices);

```
pdf('myplot.pdf')
```

2. Run the commands which create the graphics;

```
plot(x,y)
```

3. Switch the file device off (which switches back to the screen by default)

```
dev.off()
```

Next: exercises

- Basics in plotting
- Descriptive statistics & plotting
  - Categorical variables
  - Numeric variables
- Basic tools for testing for normality