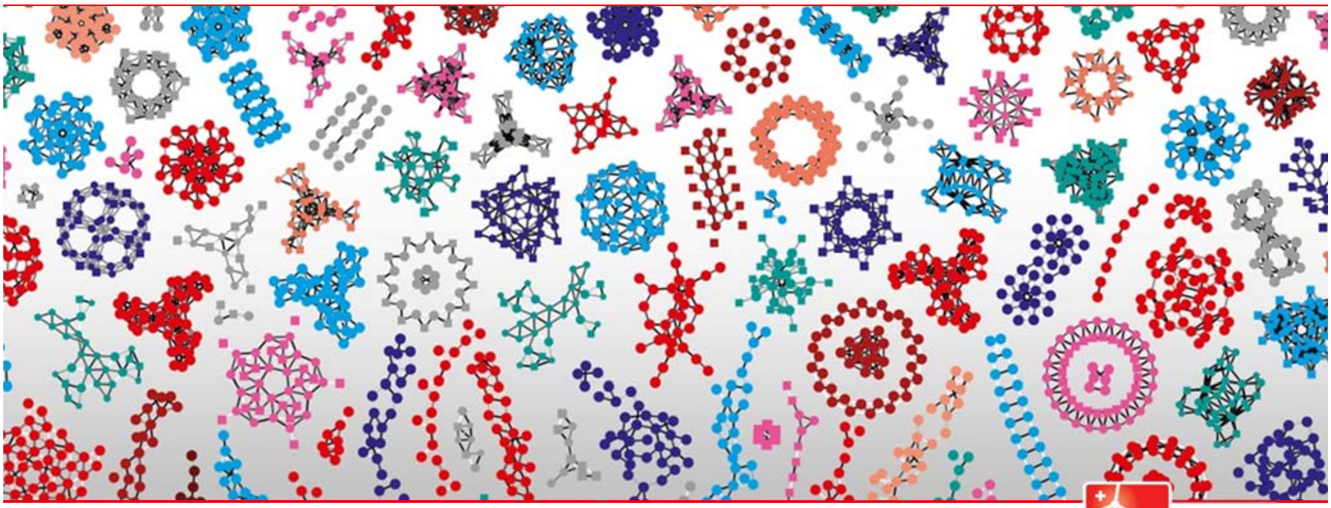


Swiss Institute of Bioinformatics

Advanced statistics: Statistical modeling

Linda Dib, Sina Nassari and Frédéric Schütz

www.sib.swiss

Statistical models

What is a statistical model ?

A **statistical model** is a set of equations involving random variables, with associated distributional assumptions, devised in the context of a **question** and a body of **data concerning some phenomenon**, with which **tentative answers** can be derived, along with **measures of uncertainty** concerning these answers.

questions + ***data*** $\xrightarrow[\textbf{\textit{model}}]{}$ ***answers*** + ***measures of uncertainty***

(from Terry Speed)

Modeling Overview

Want to capture important features of the *relationship between* a (set of) *variable(s)* and one or more *response(s)*

Many models are of the form

$$g(Y) = f(\mathbf{x}) + \text{error}$$

with *differences* in the form of g, f and distributional assumptions about the error term.

A word of caution !

Modelling is not about just finding the right type of equation to describe the data, and finding the right algorithm to estimate the parameters of this equation !

In other words, we should not consider that the modeling problem consists only of simple pairs of data points (e.g. response and explanatory variables).

Other information of interest include for example how the data was collected, how it is structured, what we expect from the model (description ? Prediction ?), and what other variables were *not* observed.

We will not discuss this in detail, but we will touch on it briefly in some places.

> *Essentially, all models are wrong, but some are useful.*
>
> *Georges Box*

Model formulas in R

A simple *model formula* in R looks something like:

```
yvar ~ xvar1 + xvar2 + xvar3
```

Can read **~** as “*described (or modeled) by*”.

We could write this model (algebraically) as

$$Y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3$$

Model formulas in R

By default, an intercept is included in the model – you don’t have to include a term in the model formula

If you want to leave the intercept out:

```
yvar ~ -1 + xvar1 + xvar2 + xvar3
```

More on model formulas

The generic form is **`response ~ predictors`**

The predictors can be **`numeric`** or **`factor`**

Other symbols to create formulas with *combinations of variables* (e.g. *interactions*)

`+` to *add* more variables

`-` to *leave out* variables

`:` to introduce *interactions* between two terms

`*` to include *both interactions and the terms*

(**`a*b`** is the same as **`a + b + a:b`**)

`^n` *adds all terms* including interactions up to order n

`I()` treats what’s in () as a *mathematical expression*

Linear models

Some references

Peter Dalgaard. *Introductory Statistics with R* (second edition). Springer, 2008.

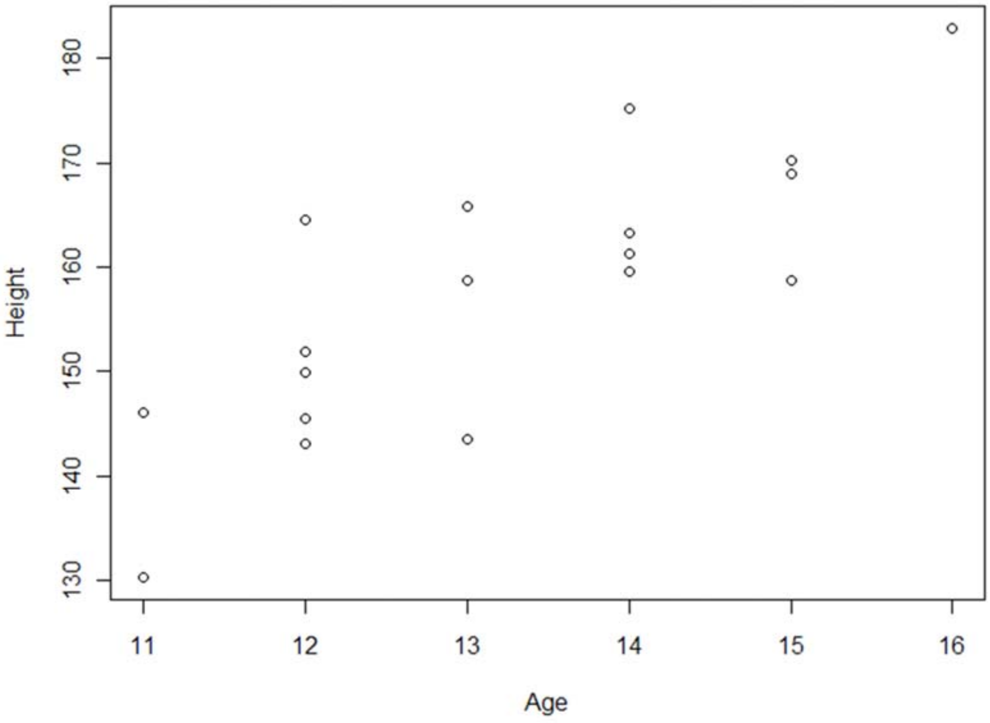
William N. Venables and Brian D. Ripley. *Modern Applied Statistics with S* (fourth edition). Springer, 2002.

John Fox. *Applied Regression, Generalized Linear Models, and Related Methods* (second Edition). Sage Publications, 2008.

John Fox. *An R and S-PLUS Companion to Applied Regression*. Sage Publications, 2002.

Can we predict the height of a teenager using his age ?

Example: scatterplot of age vs height in teenagers



(Simple) Linear Regression

Simple linear regression refers to drawing a (particular, special) line through a scatterplot

It is used for 2 broad purposes: **explanation** and **prediction**.

The equation for a line to predict y knowing x (in slope-intercept form) looks like

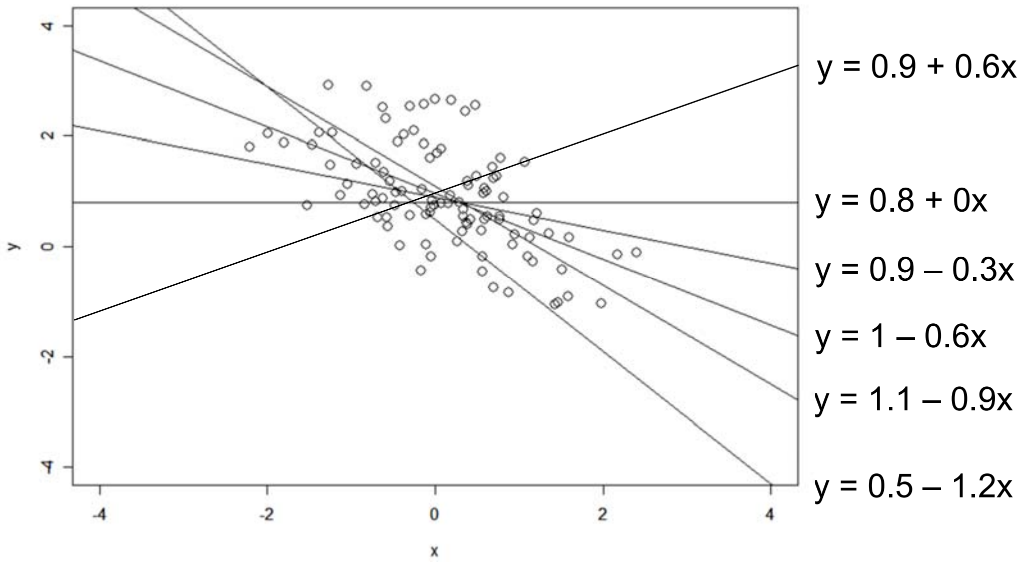
$$y = a + b\,x$$

where *a* is called the *intercept* and *b* is the *slope.*

Linear regression

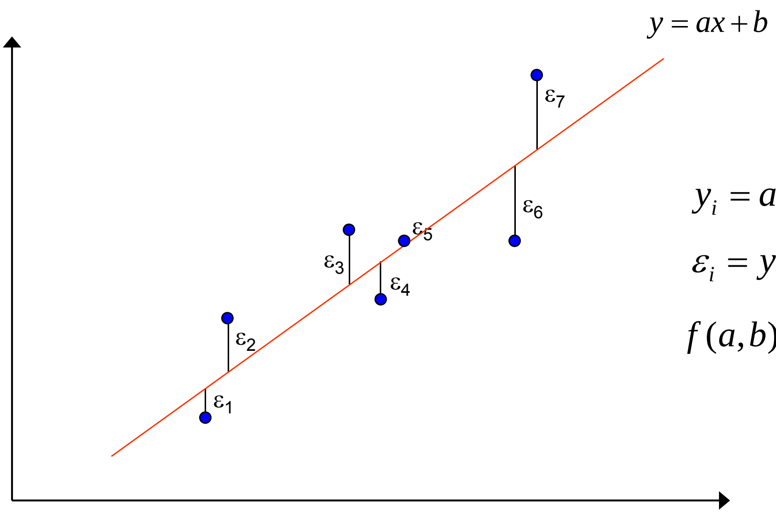
What is the “best” line which fits this data ?

Can we use it to summarise the relation between x and y ?



Linear regression: least-squares fitting

Least-square fitting



Regression line such that:

$$\sum_i \varepsilon_i^2 = \varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 + \ldots$$

minimum

$$y_i = ax_i + b + \varepsilon_i$$

$$\varepsilon_i = y_i - (ax_i + b)$$

$$f(a,b) = \sum_i \varepsilon_i^2 = \sum_i \left[y_i - (ax_i + b)\right]^2$$

$$\boxed{\begin{aligned} \partial f(a,b)/\partial a = 0 \\ \partial f(a,b)/\partial b = 0 \end{aligned}}$$

The least-squares procedure finds the straight line with the **smallest sum of squares of vertical errors**.

Linear models (simple case)

Formalization and extension of linear regression

$$\boxed{Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i}$$

$$i = 1, \cdots, n$$

Y represents **one** data point

Y_i : response (known)

β_0, β_1 : model parameters (estimated)

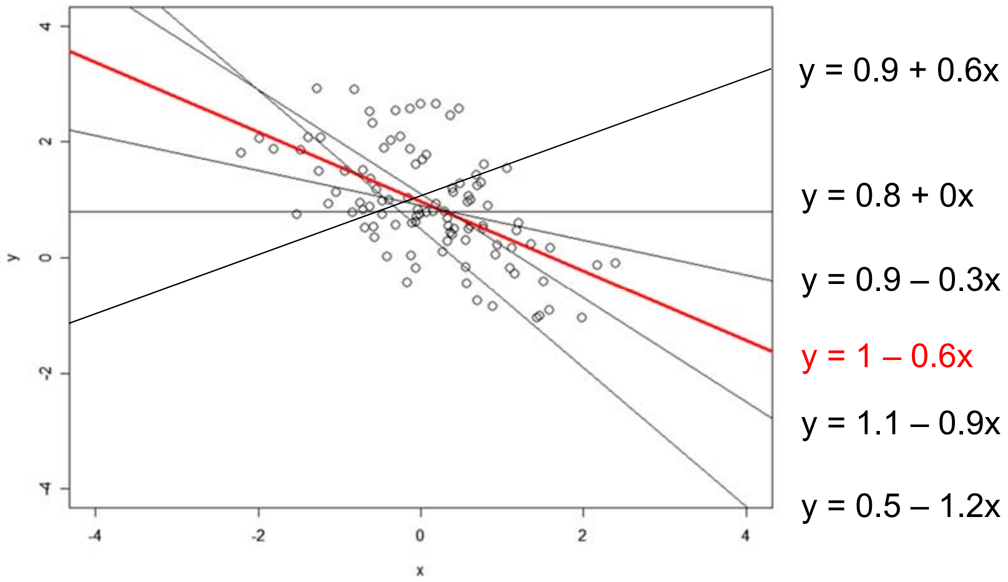
X_i : predictor (known)

ε_i : error term ~ $N(0, \sigma^2)$ (estimated)

Minimizing $\sum_i \varepsilon_i^2$ yields b_0 and b_1 estimators of β_0 and β_1

$$b_1 = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sum (X_i - \bar{X})^2} \qquad b_0 = \bar{Y} - b_1 \bar{X}$$

Over all possible straight lines, y= 1 - 0.6x is the “best” possible line according to this criterion.



Interpretation of parameters

The regression line has two parameters: the *slope* and the *intercept*

The regression *slope* is *the average change in Y when X increases by 1 unit*

The *intercept* is *the predicted value for Y when X = 0*

If the slope = 0, then *X* does not help in predicting *Y* (linearly)

Residuals

There is an *error* in making a regression prediction:

error = observed Y – predicted Y

= y – (a + b x)

These errors are called *residuals*

The regression equation is calculated so that the sum (and mean) of the residuals is 0 (« in average, the model is correct »).

Ideally, we want the regression to include all the predictable variance, so that the distribution of the residuals is random and does not depend on X or on the predicted X.

Linear models (general case)

p parameter linear model

$$\boxed{Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{ip-1} + \varepsilon_i} \qquad i = 1, \cdots, n$$

or $$Y_i = \sum_{k=0}^{p-1} \beta_k X_{ik} + \varepsilon_i \qquad \text{with} \qquad X_{i0} \equiv 1$$

Y_i response (e.g. expression of a gene)

X_{ik} predictor variables (e.g. dose of drug [continuous], or KO vs wt)

β_k model parameter (measurement of magnitude of effect associated to predictor variable)

ε_i error term (measurement of departure from ideal case)

Linear models (matrix form)

Matrix form of linear models

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{ip-1} + \varepsilon_i$$

is equivalent to

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{11} & X_{12} & \cdots & X_{1p-1} \\ 1 & X_{21} & X_{22} & \cdots & X_{2p-1} \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & X_{n2} & \cdots & X_{np-1} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

or $\boxed{\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}}$

Linear models (parameter estimation)

Least-square estimation of regression coefficients

$\{\beta_k\}$ such that

$$Q = \sum_i \varepsilon_i{}^2 = \sum_i (Y_i - \beta_0 - \beta_1 X_{i1} - \beta_2 X_{i2} - \cdots - \beta_{p-1} X_{ip-1})^2 \quad \text{minimum}$$

$\mathbf{b} = (b_0 \cdots b_{p-1})'$ estimator of $\boldsymbol{\beta}$ is computed as follows:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

$$\mathbf{X'Xb} = \mathbf{X'Y} \qquad E\{\boldsymbol{\varepsilon}\} = \mathbf{0}$$

$$\boxed{\mathbf{b} = (\mathbf{X'X})^{-1}\mathbf{X'Y}}$$

Linearity in linear models

Linearity is about the model parameters

$$\left.\begin{aligned}
Y_i &= \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{ip-1} + \varepsilon_i \\
Y_i &= \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \beta_3 X_i^3 + \varepsilon_i \\
Y_i &= \beta_0 + \beta_1 \log X_{i1} + \beta_2 X_{i2} + \varepsilon_i \\
Y_i &= \beta \sin X_i + \varepsilon_i
\end{aligned}\right\} \quad \text{Linear in } \beta\text{s}$$

$$\left.\begin{aligned}
Y_i &= \beta_0 + \log(\beta_1 X_{i1} + \beta_2 X_{i2}) + \beta_3 X_{i3} + \varepsilon_i \\
Y_i &= \beta_0 + \beta_1 \exp(\beta_2 X_i + \beta_3) + \varepsilon_i
\end{aligned}\right\} \quad \text{Not linear in } \beta\text{s}$$

A concrete example in R

Using the CLASS dataset, from the program SAS
(units have been modified from imperial to metric)

```
data <- read.table("http://lausanne.isb-sib.ch/~schutz/data/class.txt")
```

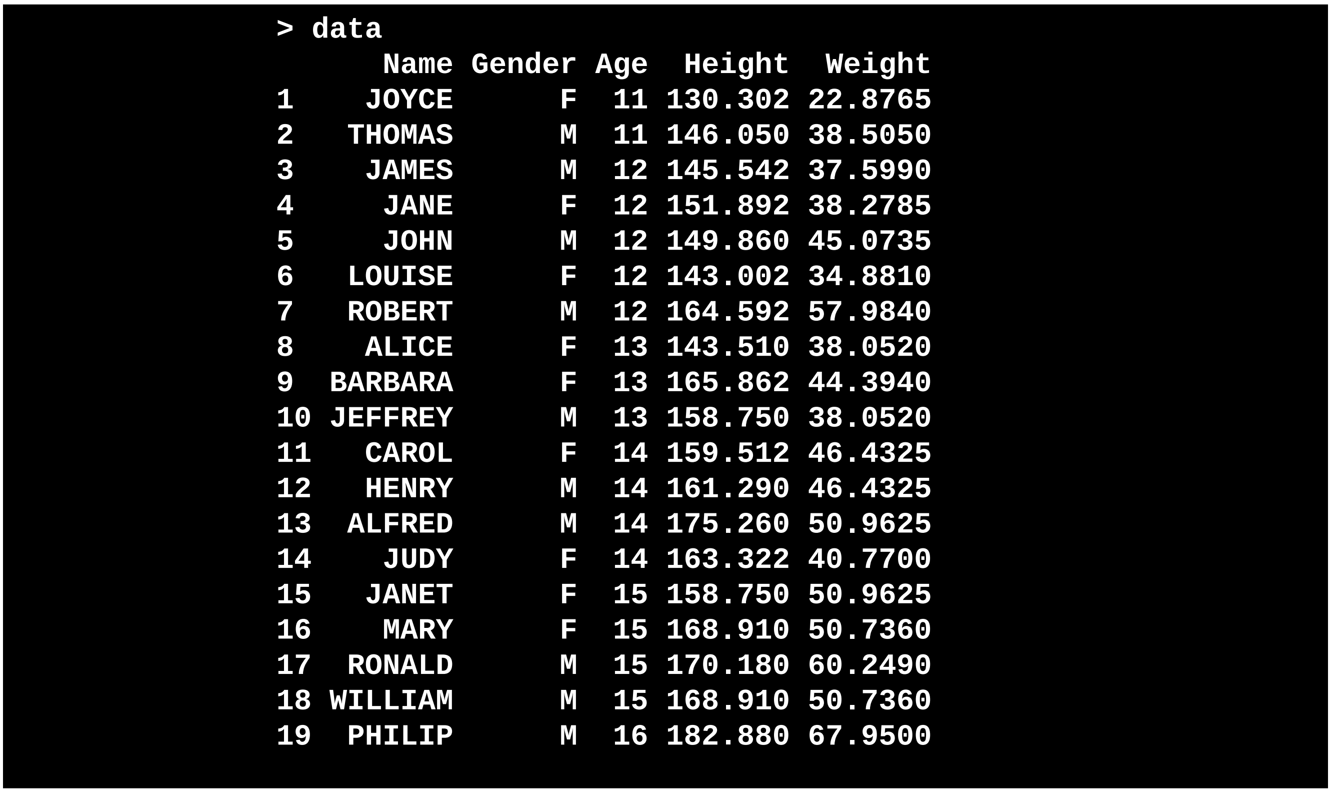
Exercise

Use statistical models to answer the question:

"Can we predict the height of a teenager, using his age, sex and weight ?"

```
data <- read.table("http://lausanne.isb-sib.ch/~schutz/data/class.txt")
```

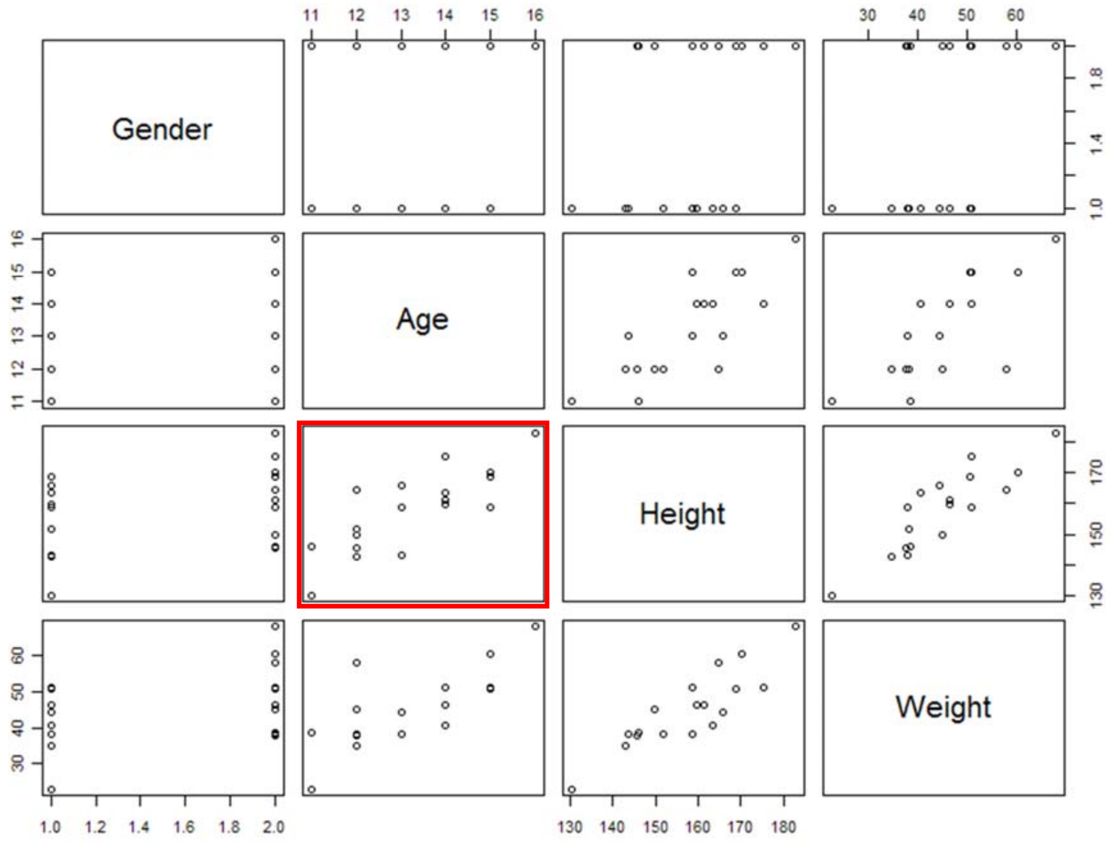
The CLASS dataset from SAS



```
> data
      Name Gender Age  Height  Weight
1    JOYCE      F  11 130.302 22.8765
2   THOMAS      M  11 146.050 38.5050
3    JAMES      M  12 145.542 37.5990
4     JANE      F  12 151.892 38.2785
5     JOHN      M  12 149.860 45.0735
6   LOUISE      F  12 143.002 34.8810
7   ROBERT      M  12 164.592 57.9840
8    ALICE      F  13 143.510 38.0520
9  BARBARA      F  13 165.862 44.3940
10 JEFFREY      M  13 158.750 38.0520
11   CAROL      F  14 159.512 46.4325
12   HENRY      M  14 161.290 46.4325
13  ALFRED      M  14 175.260 50.9625
14    JUDY      F  14 163.322 40.7700
15   JANET      F  15 158.750 50.9625
16    MARY      F  15 168.910 50.7360
17  RONALD      M  15 170.180 60.2490
18 WILLIAM      M  15 168.910 50.7360
19  PHILIP      M  16 182.880 67.9500
```

The CLASS dataset from SAS

```
> summary(data[,-1])

Gender       Age              Height           Weight
F: 9   Min.   :11.00   Min.   :130.3   Min.   :22.88
M:10   1st Qu.:12.00   1st Qu.:148.0   1st Qu.:38.17
       Median :13.00   Median :159.5   Median :45.07
       Mean   :13.32   Mean   :158.3   Mean   :45.31
       3rd Qu.:14.50   3rd Qu.:167.4   3rd Qu.:50.85
       Max.   :16.00   Max.   :182.9   Max.   :67.95
```

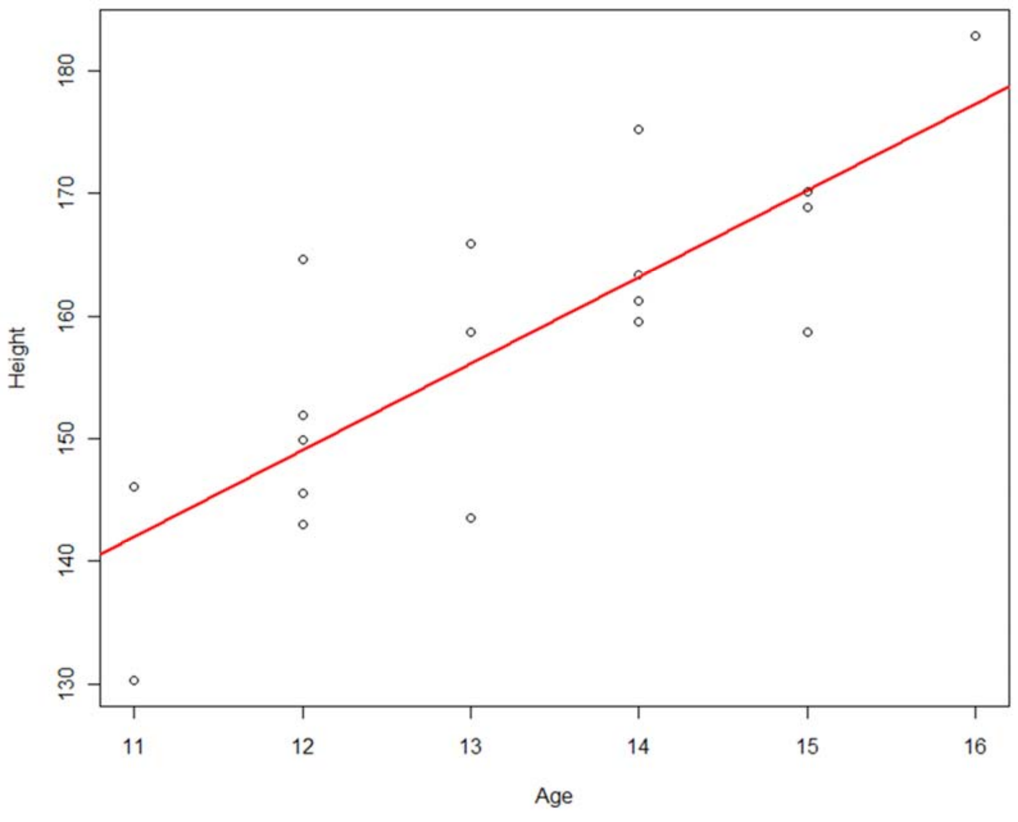


```
> pairs(data[,-1])
```

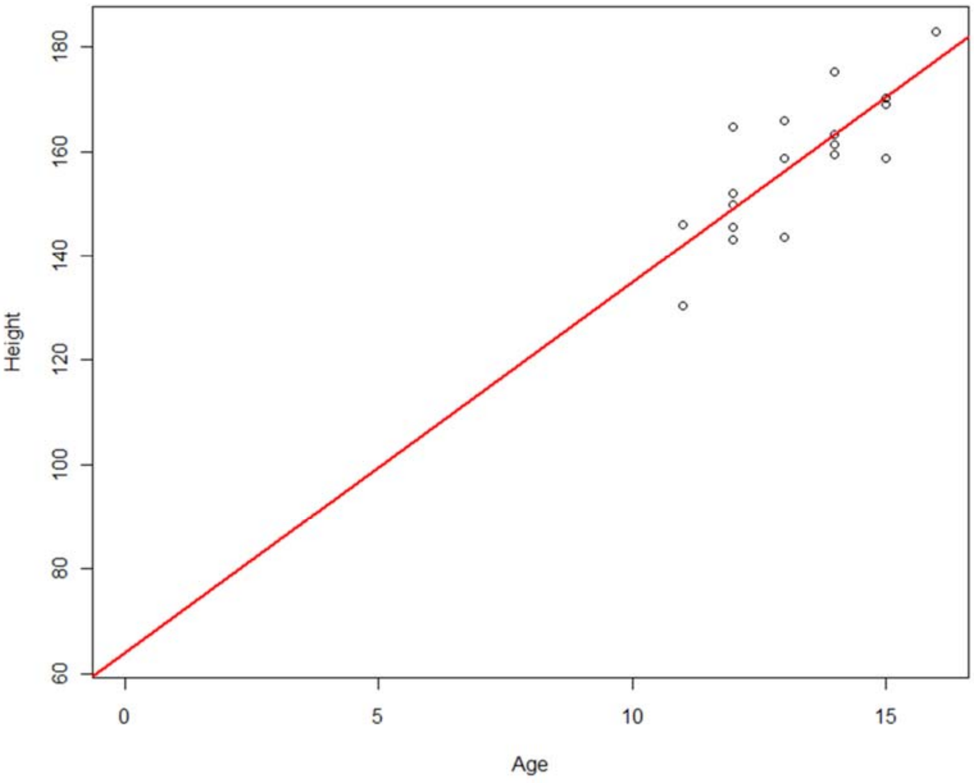
Fitting the linear model in R

```
> model <- lm( Height ~ Age )
> model

Call:
lm(formula = Height ~ Age)

Coefficients:
(Intercept)          Age
      64.07         7.08
```

Model: Height = 64.07 + 7.08 x Age



```
> plot( Age, Height )
> abline(model, col="red", lwd=2)
```



```
plot( Age, Height, xlim=range(0,Age), ylim=range(coef(model)[1], Height) )
abline(model, col="red", lwd=2)
```

Example of summary results of the `lm` command in R

```
> summary( lm( Height ~ Age) )

Call:
lm(formula = Height ~ Age)

Residuals:
      Min         1Q    Median         3Q       Max 
-12.59000  -3.57300  -0.07867   3.49000  15.57133 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   64.069     16.565   3.868  0.00124 ** 
Age            7.079      1.237   5.724 2.48e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 7.832 on 17 degrees of freedom
Multiple R-squared: 0.6584,     Adjusted R-squared: 0.6383 
F-statistic: 32.77 on 1 and 17 DF,  p-value: 2.48e-05
```

Function call

```
> model <- lm( Height ~ Age )
…
> summary( model )

Call:
lm(formula = Height ~ Age)
```

Distribution of the residuals

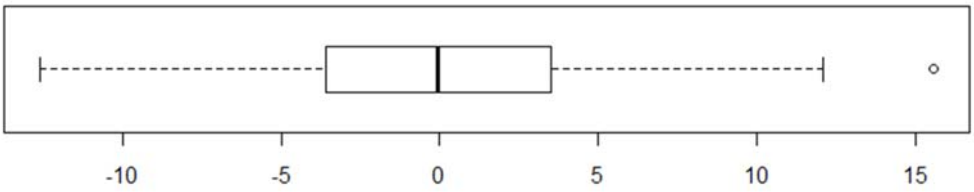
```
Residuals:
      Min        1Q    Median        3Q       Max
-12.59000  -3.57300  -0.07867   3.49000  15.57133
```

Five-number summary of the residuals (but no mean – why ?), equivalent to

```
> fivenum( residuals( model ) )
      8      11      17       4       7
-12.590  -3.573  -0.078   3.490  15.571
```

or, graphically, using a boxplot:

```
> boxplot( residuals ( model), horizontal=T)
```



Coefficients

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   64.069     16.565   3.868  0.00124 ** 
Age            7.079      1.237   5.724 2.48e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

These statistical tests tell us if the parameters are significantly different from 0. It is not interesting for the intercept, but usually interesting for the slope.

`Estimate` and `Std. Error` are obtained from the matrices of the model.

T-value = Estimate / Std. Error

This assumes that the residuals follow a normal distribution !

RSE (Residual Standard Error) and degrees of freedom

```
Residual standard error: 7.832 on 17 degrees of freedom
```

The *number of degrees* of freedom indicates the number of independant pieces of data that are available to estimate the error

While we have 19 residuals here, they are not all independent: for example, the last one is constrained because the sum of all residuals must be 0.

The number of DF is

total observations – number of parameters estimated

Two parameters are estimated (intercept + coefficient), so 19-2 = 17

RSE (Residual Standard Error) and degrees of freedom

```
Residual standard error: 7.832 on 17 degrees of freedom
```

The residual standard error is the standard deviation of the residuals (which we would usually like to be small)

It is not exactly equal to what the sd command would return:

```
> sd(residuals(model))
[1] 7.611075
> sqrt(sum(residuals(model)^2)/18)
[1] 7.611075
```

Here, we must divide by the number of degrees of freedom to get the same number:

```
> sqrt(sum(residuals(model)^2)/17)
[1] 7.831732
```

Multiple and adjusted R-squared

```
Multiple R-squared: 0.6584,     Adjusted R-squared: 0.6383
```

R^2 is the proportion of the total variance in the response data that is explained by the model (if R^2=1, the data fits perfectly on a straight line, and the model explains all the variance).

In the case of simple regression, it is equal to the square of the correlation coefficient between the two variables:

```
> summary(model)$r.squared
[1] 0.6584257
> cor(Age, Height)^2
[1] 0.6584257
```

The Adjusted R-squared is similar to R-squared, but it takes into account the number of variables in the model (we will come back to this later).

F-test for significance of regression

```
F-statistic: 32.77 on 1 and 17 DF,  p-value: 2.48e-05
```

The F-statistic allows us to test if the whole regression (adding all variables *vs* having only the intercept in) is significant.

With only one variable, it provides *exactly* the same result as the t-test for the significance of the coefficient of this variable.

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   64.069     16.565   3.868  0.00124 **
Age            7.079      1.237   5.724 2.48e-05 ***
```

Multiple regression: assessing the effect of several variables *together*

Two separate simple regressions

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   64.069     16.565   3.868  0.00124 ** 
Age            7.079      1.237   5.724 2.48e-05 ***
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 108.12816    6.80692  15.885 1.24e-11 ***
Weight        0.50194    0.06644   7.555 7.89e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Can we say anything about what would happen if both variables were included the same model ?

One multiple regression with two variables

```
Call:
lm(formula = Height ~ Age + Weight)

Residuals:
     Min       1Q   Median       3Q      Max 
-9.20695 -3.30604 -0.04478  2.11432 10.41880 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 81.77355   12.90896   6.335 9.92e-06 ***
Age          3.11575    1.34668   2.314  0.03431 *  
Weight       0.35064    0.08827   3.973  0.00109 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.728 on 16 degrees of freedom
Multiple R-squared: 0.828,      Adjusted R-squared: 0.8065 
F-statistic: 38.52 on 2 and 16 DF,  p-value: 7.646e-07
```

This model allows us to determine the respective contribution of each variable separately !

Coefficients

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 81.77355   12.90896   6.335 9.92e-06 ***
Age          3.11575    1.34668   2.314  0.03431 *  
Weight       0.35064    0.08827   3.973  0.00109 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

This is similar to the simple regression case.

Each test is conducted assuming that the tested parameter is the **last one entering the model**:

« If *weight* is already in the model, is the coefficient for *age* significantly different from 0 ? »

Two single regressions vs one multiple regression

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   64.069     16.565   3.868  0.00124 ** 
Age            7.079      1.237   5.724 2.48e-05 ***
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 108.12816    6.80692  15.885 1.24e-11 ***
Weight        0.50194    0.06644   7.555 7.89e-07 ***
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  81.77355   12.90896   6.335 9.92e-06 ***
Age           3.11575    1.34668   2.314  0.03431 *  
Weight        0.35064    0.08827   3.973  0.00109 ** 
```

While both age and weight seem significant by themselves, age is much less significant when weight is already included (see also the R^2)

It is likely that a lot of the information provided by the age is also provided by the weight, so that there may be little need to have both terms in the model.

Multiple and adjusted R-squared

```
Multiple R-squared: 0.828,      Adjusted R-squared: 0.8065
```

As before, R^2 is the proportion of the total variance in the response data that is explained by the model.

Adding a new variable in the model will always increase R^2, up to 1 when there the number of degrees of freedom is 0 (number of parameters to estimate = number of observations).

Multiple and adjusted R-squared

```
Multiple R-squared: 0.828,      Adjusted R-squared: 0.8065
```

The adjusted R-squared adjusts for the number of variables in the model, and does not necessarily increase when the number of variables increase; it can even be negative.

It is always equal or below R^2.

Example

```
y <- rnorm(10)
x1 <- rnorm(10); x2 <- rnorm(10); … ; x9 <- rnorm(10)
summary(lm(y ~ x1)); summary(lm(y ~ x1+x2)); …
```

```
1: Multiple R-squared: 0.1419,     Adjusted R-squared: 0.03464
2: Multiple R-squared: 0.5173,     Adjusted R-squared: 0.3794
3: Multiple R-squared: 0.557,      Adjusted R-squared: 0.3355
4: Multiple R-squared: 0.5577,     Adjusted R-squared: 0.2039
5: Multiple R-squared: 0.7953,     Adjusted R-squared: 0.5395
6: Multiple R-squared: 0.8321,     Adjusted R-squared: 0.4962
7: Multiple R-squared: 0.984,      Adjusted R-squared: 0.9281
8: Multiple R-squared: 0.9851,     Adjusted R-squared: 0.866
9: Multiple R-squared:     1,      Adjusted R-squared:   NaN
```

The last regression from the example

```
Call:
lm(formula = y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9)

Residuals:
ALL 10 residuals are 0: no residual degrees of freedom!

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.02693         NA      NA       NA
x1           0.53886         NA      NA       NA
x2          -0.52227         NA      NA       NA
x3           0.51881         NA      NA       NA
x4           0.74757         NA      NA       NA
x5           0.14394         NA      NA       NA
x6          -0.65387         NA      NA       NA
x7          -0.48271         NA      NA       NA
x8          -0.62487         NA      NA       NA
x9           0.23759         NA      NA       NA

Residual standard error: NaN on 0 degrees of freedom
Multiple R-squared:     1,      Adjusted R-squared:   NaN
F-statistic:   NaN on 9 and 0 DF,  p-value: NA
```

F-statistic for significance of regression

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 81.77355   12.90896   6.335 9.92e-06 ***
Age          3.11575    1.34668   2.314  0.03431 *
Weight       0.35064    0.08827   3.973  0.00109 **

F-statistic: 38.52 on 2 and 16 DF,  p-value: 7.646e-07
```

Again, the F-statistic allows us to test if the whole regression (adding all variables *vs* having only the intercept in) is significant.

If any of the tests for the individual variables is significant, the F-test will generally be significant as well.

However, even if no individual variable is significant (e.g. $p < 0.05$), the F-test can still be significant.

Categorical variables, dummy variables and contrasts

Categorical variables

We’d like to use categorical variables in a linear model, as in:

$$\text{Height} = b_0 + b_1 \text{ Age} + b_2 \text{ « Gender » + error}$$

Intuitively, we want to estimate a « Male » and a « Female » effect.

In practice, categorical variables (factors in R) are turned (by default, based on alphabetical order) into **dummy variables** of the form

$$\text{Gender} = \begin{cases} 0 \text{ if Female} \\ 1 \text{ if Male} \end{cases}$$

and the model can be interpreted as follows:

- b_0 is the baseline for height among women
- b_2 represent the increase/decrease of this baseline for men.

Example of summary results of the `lm` command in R

```
Call:
lm(formula = Height ~ Age + Gender)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.8462 -4.8523 -0.8102  3.3677 13.5058 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   62.291     14.957   4.165  0.00073 ***
Age            6.928      1.117   6.202 1.27e-05 ***
GenderM        7.204      3.251   2.216  0.04152 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 7.061 on 16 degrees of freedom
Multiple R-squared: 0.7387,     Adjusted R-squared: 0.706 
F-statistic: 22.61 on 2 and 16 DF,  p-value: 2.176e-05
```

Example of summary results of the `lm` command in R

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   62.291     14.957   4.165  0.00073 ***
Age            6.928      1.117   6.202 1.27e-05 ***
GenderM        7.204      3.251   2.216  0.04152 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

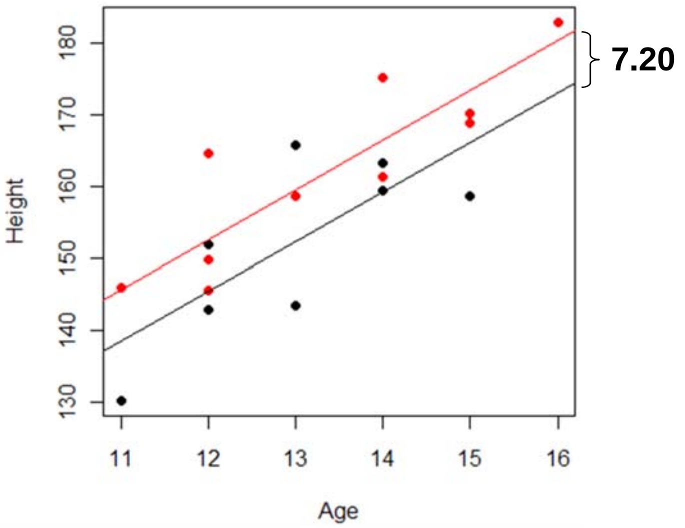
The model considers that the value for « Females » is the baseline.

The factor `GenderM` corresponds to the difference in baseline for Males compared to females.

Graphical interpretation

The model specifies 2 straight lines, with the same slope but different y-intercepts:

For women:	Height = 62.3 + 6.9 Age (in black)
For men:	Height = 69.4 + 6.9 Age (in red)



What if we don’t use a linear model ?

We could also compute the difference in means between males and females directly:

```
> means <- tapply( data$Height, data$Gender, FUN=mean )
> means
       F        M
153.8958 162.3314
> diff(means)
       M
8.435622
```

This result is slightly different from the 7.20 cm difference found with the linear model.

Where does the difference come from ?

Interactions

So far, we have assumed a difference between the lines, but the same slope; that is, for both men and women, the effect of age is the same.

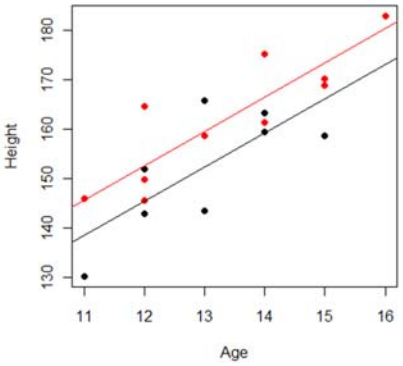
If this assumption is incorrect, it means that there is an ***interaction*** between the factors « age » and « gender », that is, the effect of age is different depending on the gender.

Interactions are modeled in R in the following way:

```
lm(formula = Height ~ Age + Gender + Age:Gender)
```

which is equivalent to

```
lm(formula = Height ~ Age * Gender)
```



Coefficients with an interaction

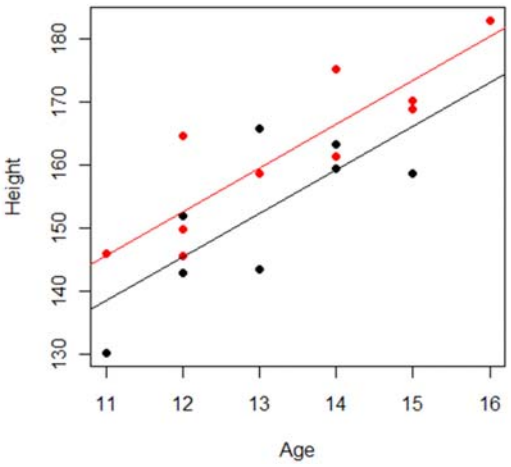
```
Call:
lm(formula = Height ~ Age * Gender)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  56.2610    24.4880   2.297  0.03640 *
Age           7.3841     1.8429   4.007  0.00114 **
GenderM      17.1304    31.5238   0.543  0.59483
Age:GenderM  -0.7468     2.3583  -0.317  0.75585
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The coefficients can be interpreted as follows:

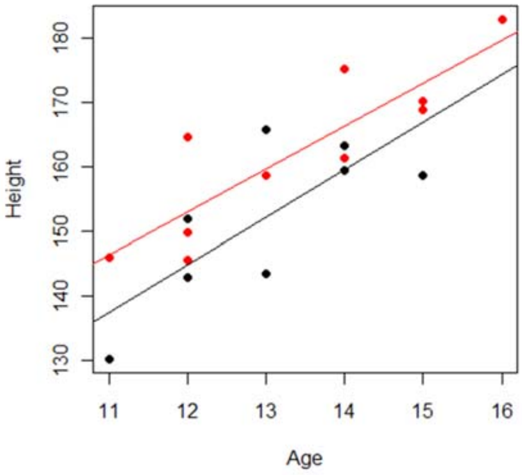
According to the model, the *height* is equal to

56.26 (the intercept)
plus 17.13, but only for males
plus 7.38 times the person’s age
minus 0.75 times the person’s age, but only for males.

Different slopes

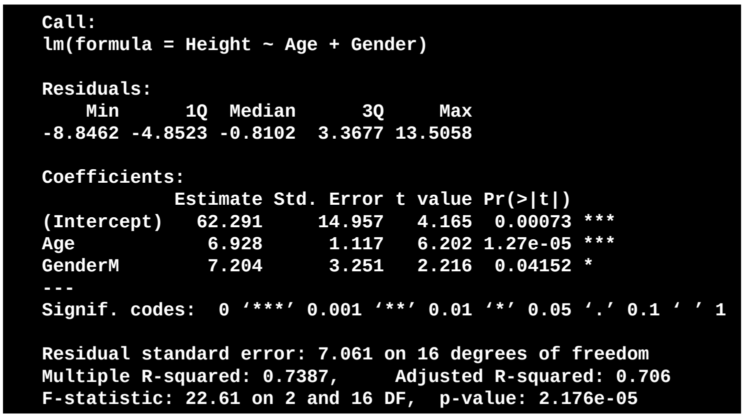


No interaction

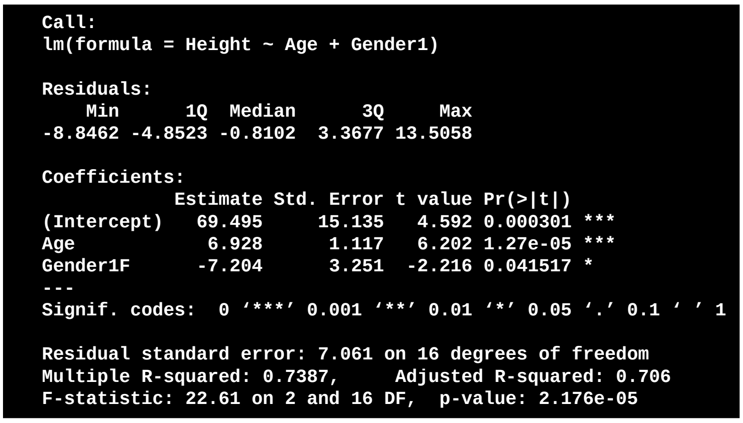


With interaction

What if Males were the baseline ?



```
Call:
lm(formula = Height ~ Age + Gender)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.8462 -4.8523 -0.8102  3.3677 13.5058 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   62.291     14.957   4.165  0.00073 ***
Age            6.928      1.117   6.202 1.27e-05 ***
GenderM        7.204      3.251   2.216  0.04152 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 7.061 on 16 degrees of freedom
Multiple R-squared: 0.7387,     Adjusted R-squared: 0.706 
F-statistic: 22.61 on 2 and 16 DF,  p-value: 2.176e-05
```



```
Call:
lm(formula = Height ~ Age + Gender1)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.8462 -4.8523 -0.8102  3.3677 13.5058 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   69.495     15.135   4.592 0.000301 ***
Age            6.928      1.117   6.202 1.27e-05 ***
Gender1F      -7.204      3.251  -2.216 0.041517 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 7.061 on 16 degrees of freedom
Multiple R-squared: 0.7387,     Adjusted R-squared: 0.706 
F-statistic: 22.61 on 2 and 16 DF,  p-value: 2.176e-05
```

The two models are exactly the same; only the way we look at the coefficient changes.

```
Gender1 <- relevel(Gender, ref="M")
```

What if my variable has more than 2 levels ?

The interpretation was straightforward with two levels: one was the baseline, and we estimated the difference between the second one and the baseline.

With more than two levels, there are different ways, termed contrasts, of looking at the coefficients. The most common one is called **treatment contrasts**, and corresponds to taking the first level as the baseline/intercept (as a control), and all the other coefficients correspond to differences of each level with the control (« treatments).

For more information on this, see e.g. Venables and Ripley, section 6.2.

Reminder: Linear models (matrix form)

Matrix form of linear models

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{ip-1} + \varepsilon_i$$

is equivalent to

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{11} & X_{12} & \cdots & X_{1p-1} \\ 1 & X_{21} & X_{22} & \cdots & X_{2p-1} \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & X_{n2} & \cdots & X_{np-1} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

or $\boxed{\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}}$

Continuous vs dummy predictors

X is the **design matrix**; a column of X_{ij} can be used to encode

Continuous quantities

- Drug dose
- Temperature
- Time

Discrete conditions (dummy predictor)

- KO (vs wt)
- Gender
- Treatment vs non-treatment

Discrete conditions require “zeros and ones” coding.

Reference condition coded as zero, alternative coded as one.
Discrete conditions with N levels require N-1 columns with 0/1.

Design matrix in R

```
> model.matrix( Height ~ Age + Gender )
   (Intercept) Age GenderM
1            1  11       0
2            1  11       1
3            1  12       1
4            1  12       0
5            1  12       1
6            1  12       0
7            1  12       1
8            1  13       0
9            1  13       0
10           1  13       1
11           1  14       0
12           1  14       1
13           1  14       1
14           1  14       0
15           1  15       0
16           1  15       0
17           1  15       1
18           1  15       1
19           1  16       1
attr(,"assign")
[1] 0 1 2
attr(,"contrasts")
attr(,"contrasts")$Gender
[1] "contr.treatment"
```

Diagnostic tools

Basic model checking

It is always possible to fit a linear model and find a slope and intercept

… but it does not mean that the model is meaningful !

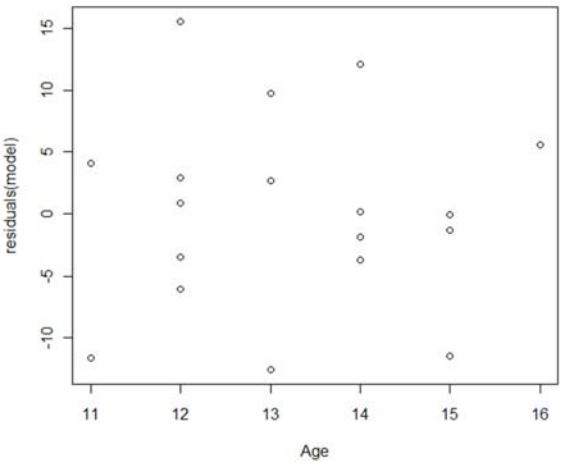
Examination of *residuals*: (which should show no obvious trend, since any systematic effect in the residuals should ideally be captured by the model):

- Normality
- Time effects
- Nonconstant variance
- Curvature

Detection of *influential observations*

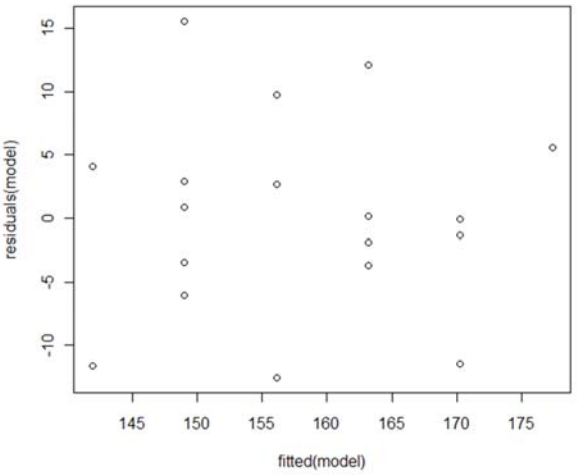
- *Hat matrix*

Residuals



`plot( Age, residuals(model) )`

Works only for simple regression (only one variable on x axis)



`plot( fitted(model), residuals(model) )`

Works also for multiple regression

Hat values

High leverage ('influential') points are far from the center, and have potentially greater influence

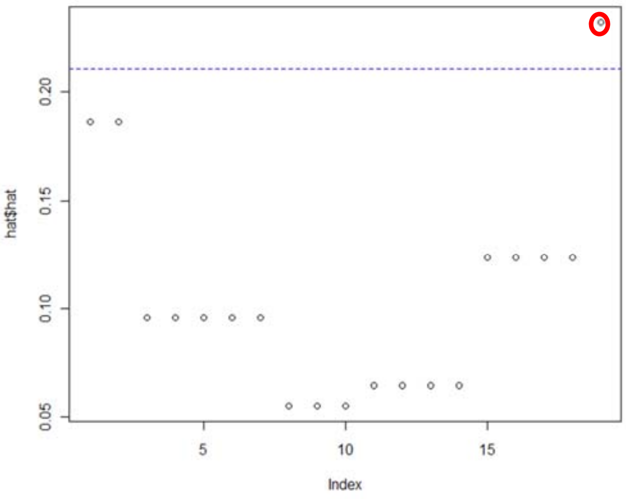
One way to assess points is through the *hat values* (obtained from the *hat matrix H*):

$$\hat{y} = Xb = X(X'X)^{-1}X'y = Hy$$

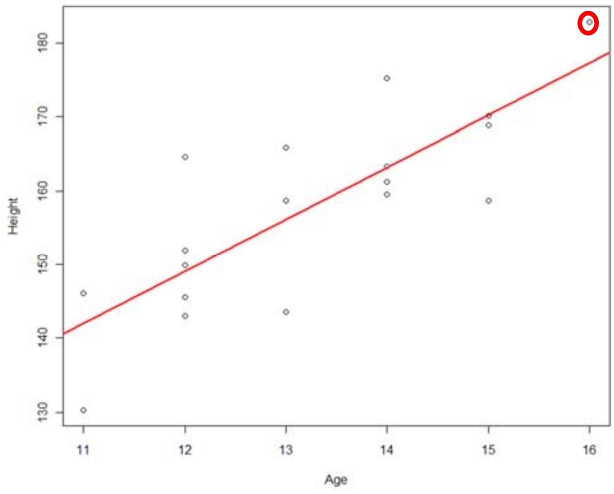
$$h_i = \Sigma_j h_{ij}^2$$

Average value of h = number of coefficients/n (including the intercept) = p/n

Cutoff typically 2p/n or 3p/n



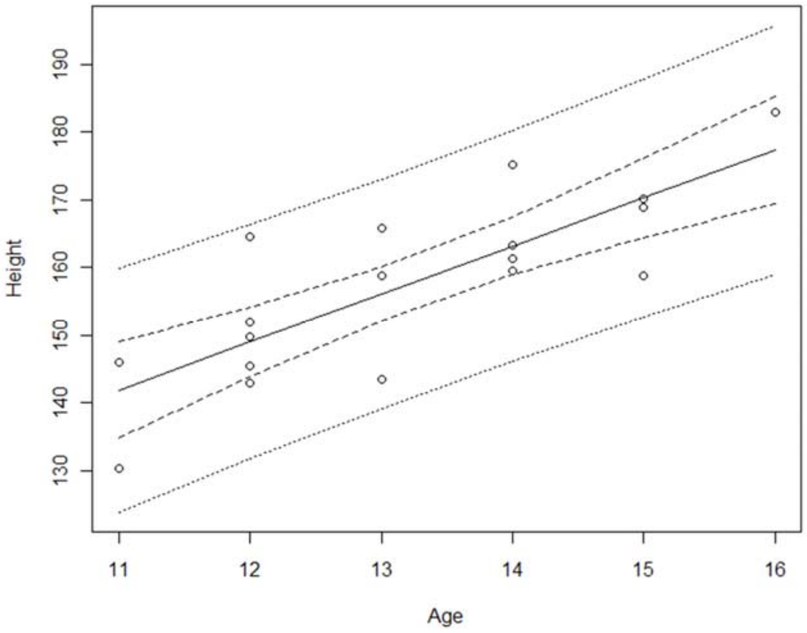
Hat values



Actual fit

```
hat <- lm.influence( model )
plot( hat$hat )
abline(h=c(c(2,3)*2/19),lty=c(2,3),col=c("blue","red") )
```

Confidence bands

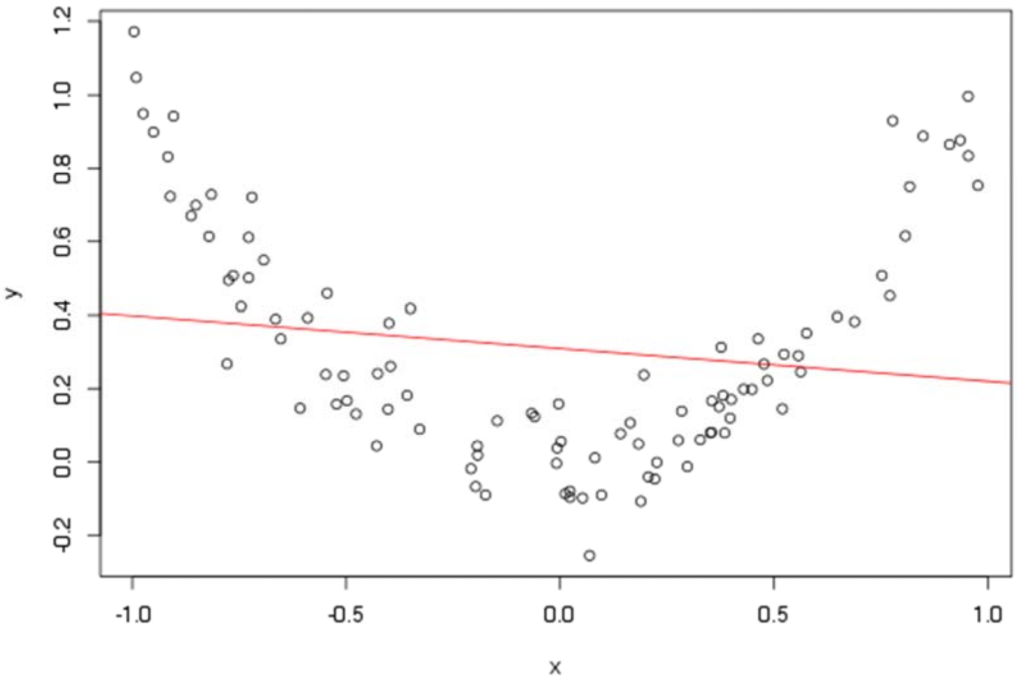


Narrow bands: describe the uncertainty about the regression line

Wide bands: describe where most (95% by default) predictions would fall, assuming normality and constant variance.

In R: `?predict.lm`

What if the data is not linear ?

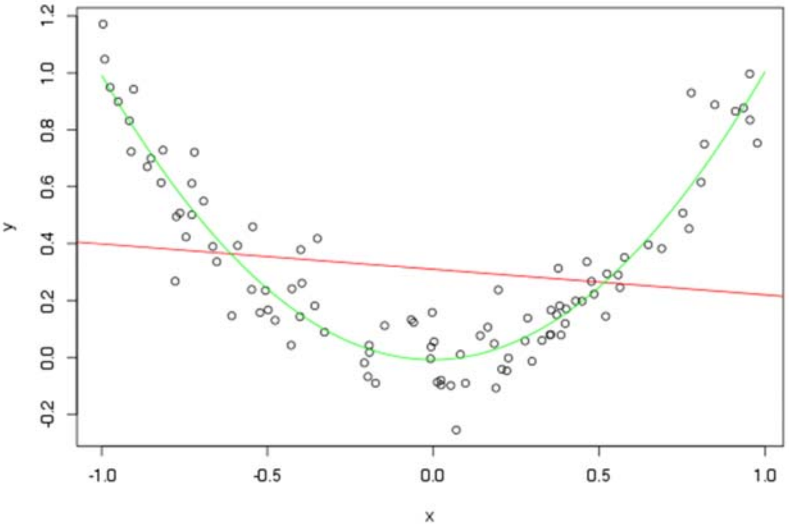


What if the data is not linear ?

Use a polynomial regression

$$y = b_0 + b_1 x + b_2 x^2$$

This is still linear for b_i; it is as if we had added a new variable.

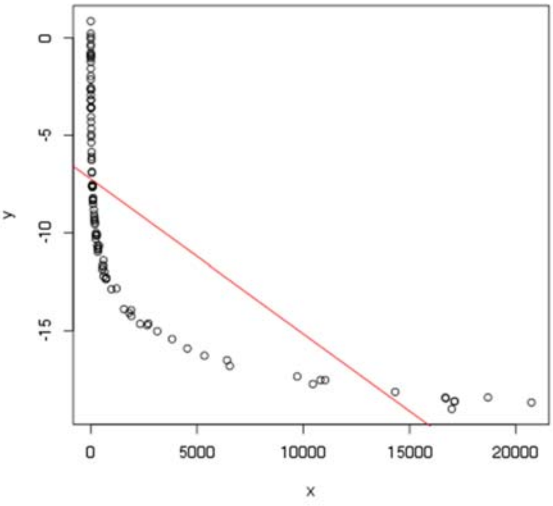


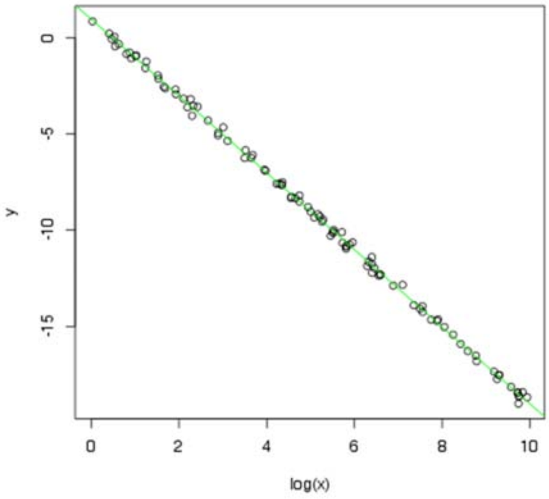
What if the data is not linear ?

Consider transforming the data (log)

$$\log(y) = a + b\,x$$

$$y = a + b \log(x)$$





Example: predicting cell concentration

The `hellung` dataset

" Diameter and concentration of *Tetrahymena* cells with and without glucose added to growth medium."

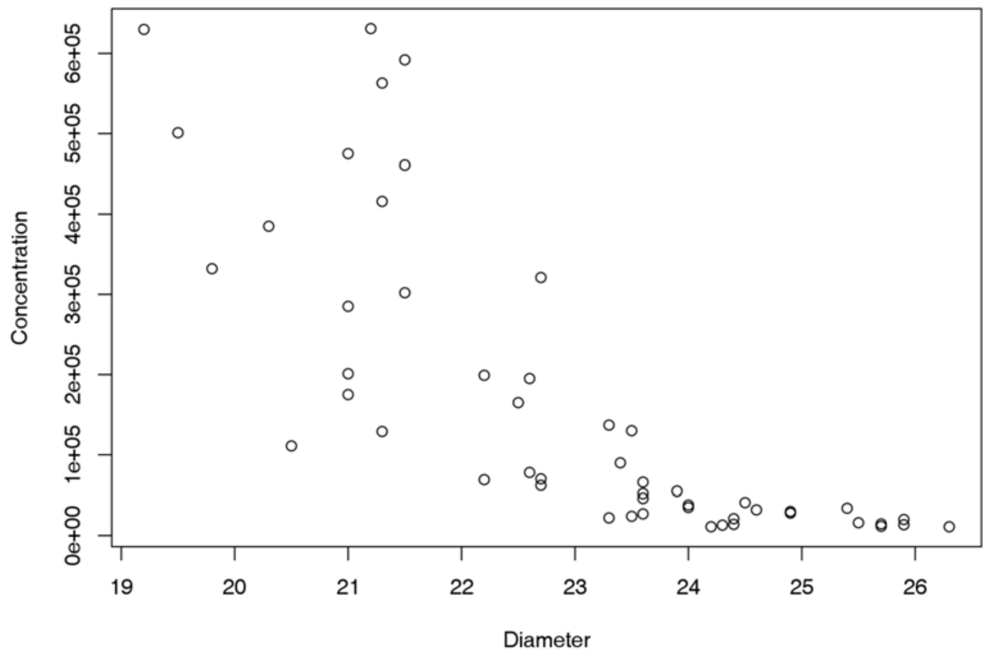
```
> library(ISwR); data(hellung)
```

Can we predict the concentration of cells using the diameter and the presence/absence of glucose ?

The Hellung data in R

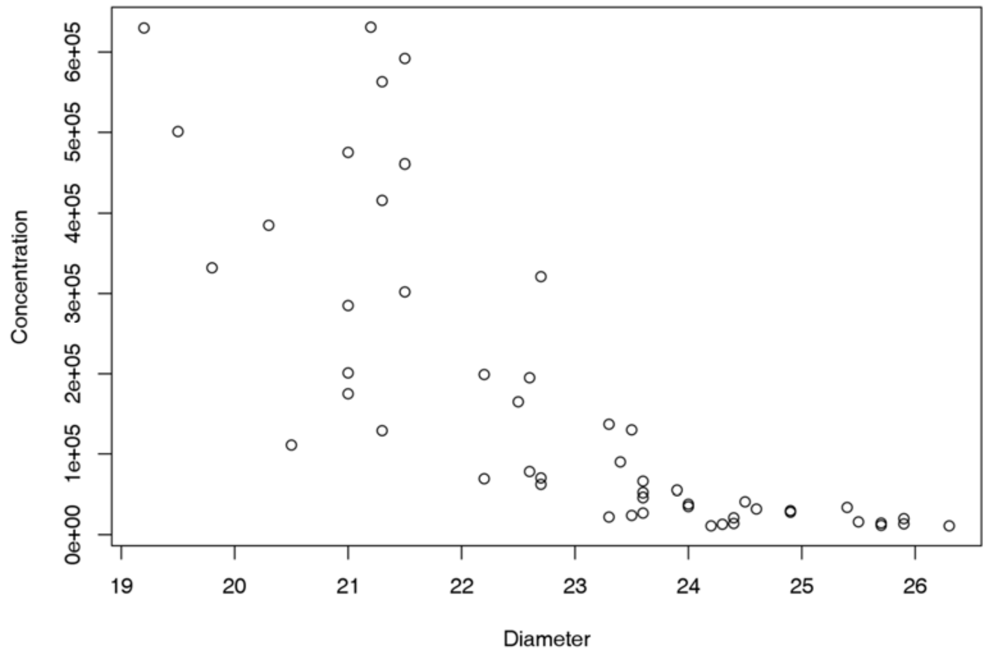
```
> hellung
   glucose   conc diameter
1        1 631000     21.2
2        1 592000     21.5
3        1 563000     21.3
4        1 475000     21.0
5        1 461000     21.5
[...]
33       2 630000     19.2
34       2 501000     19.5
35       2 332000     19.8
36       2 285000     21.0
37       2 201000     21.0
```

Hellung dataset: Diameter vs Concentration

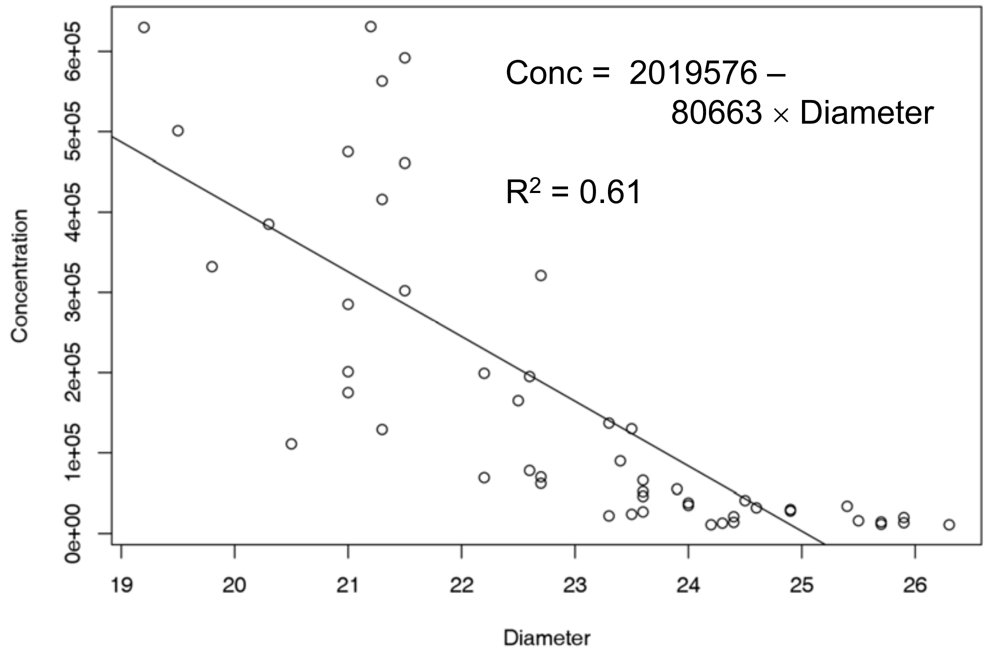


```
> plot(hellung$diameter, hellung$conc,
        xlab="Diameter", ylab="Concentration")
```

Can we predict the concentration given the diameter of the cells ?



Linear model predicting Concentration from Diameter

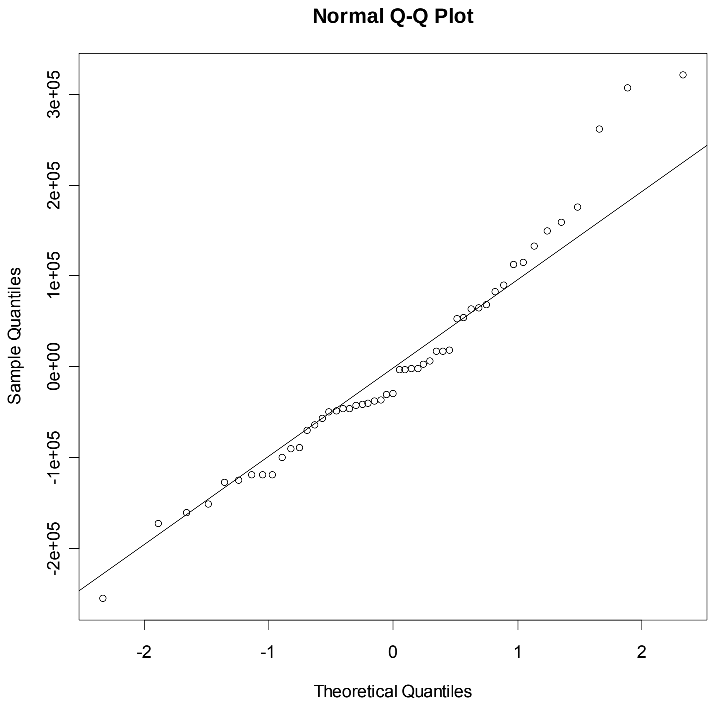


Conc = 2019576 – 80663 × Diameter

$R^2 = 0.61$

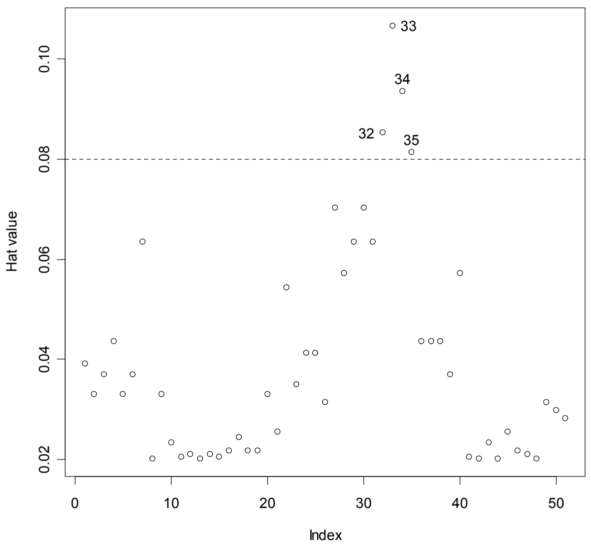
```
> model <- lm( conc ~ diameter, data=hellung )
> abline(model)
```

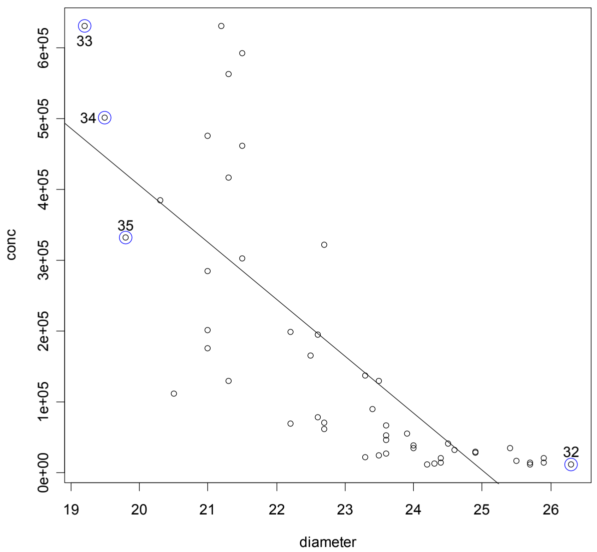
Do the residuals follow a normal distribution ?



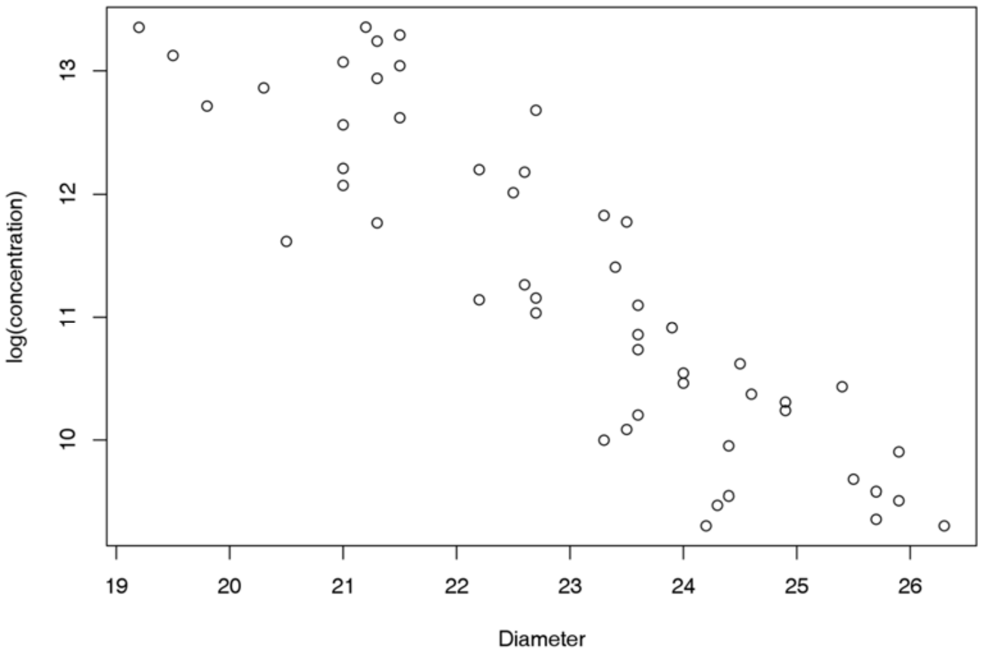
```
> qqnorm(residuals(model))
> qqline(residuals(model))
```

Hat values



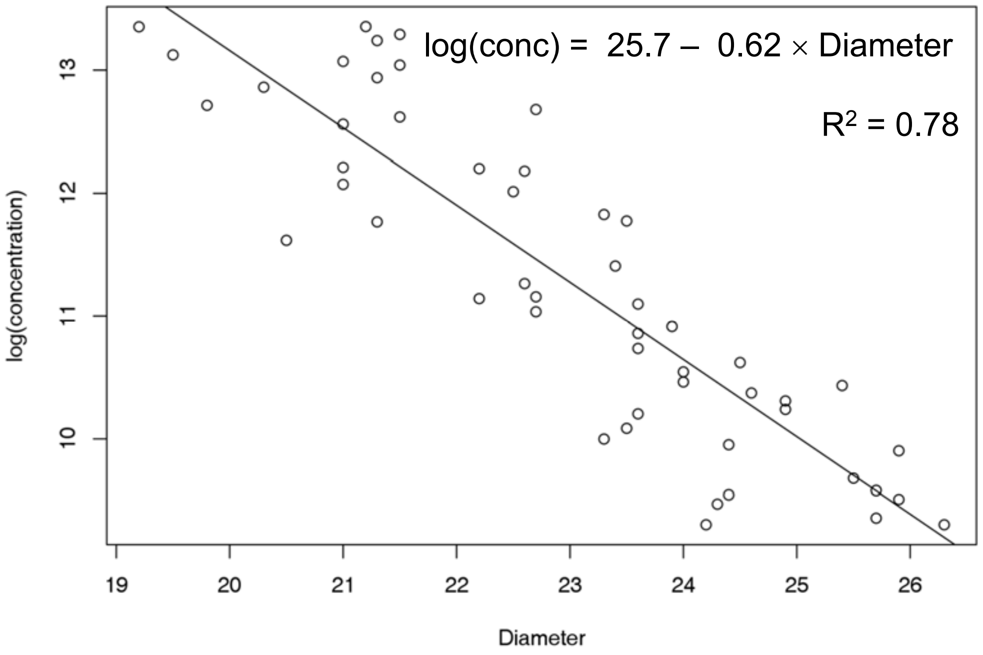


Transforming the data to improve the fit



```
logconc <- log(hellung$conc)
plot(hellung$diameter, logconc,
     xlab="Diameter", ylab="log(concentration)" )
```

Linear model predicting log(Concentration) from Diameter



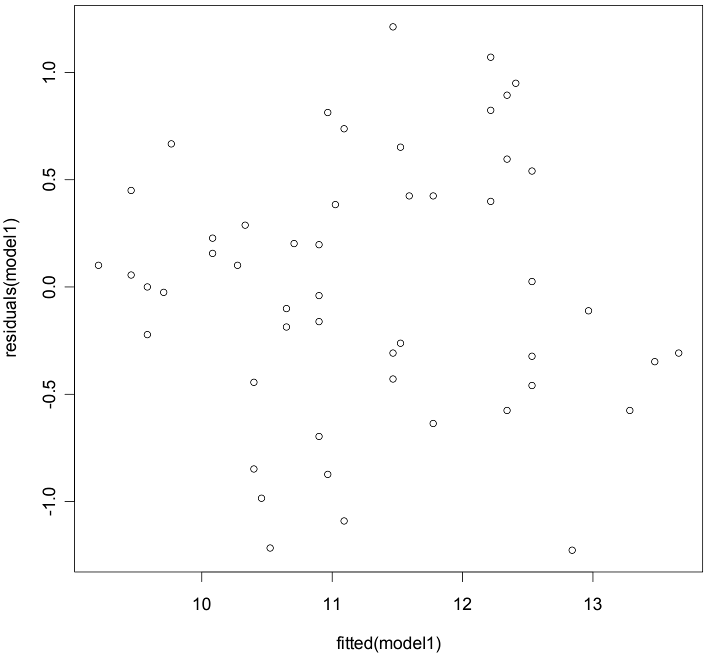
```
modellog <- lm(logconc ~ diameter, data=hellung)
abline(modellog)
```

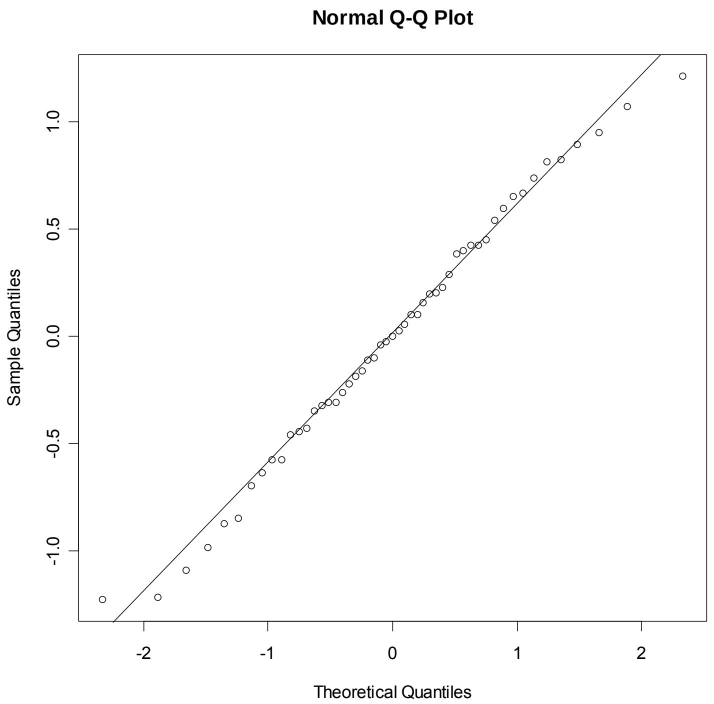
Details of the linear model

$$\log(\text{concentration}) = 25.7 - 0.63 \times \text{diameter}$$

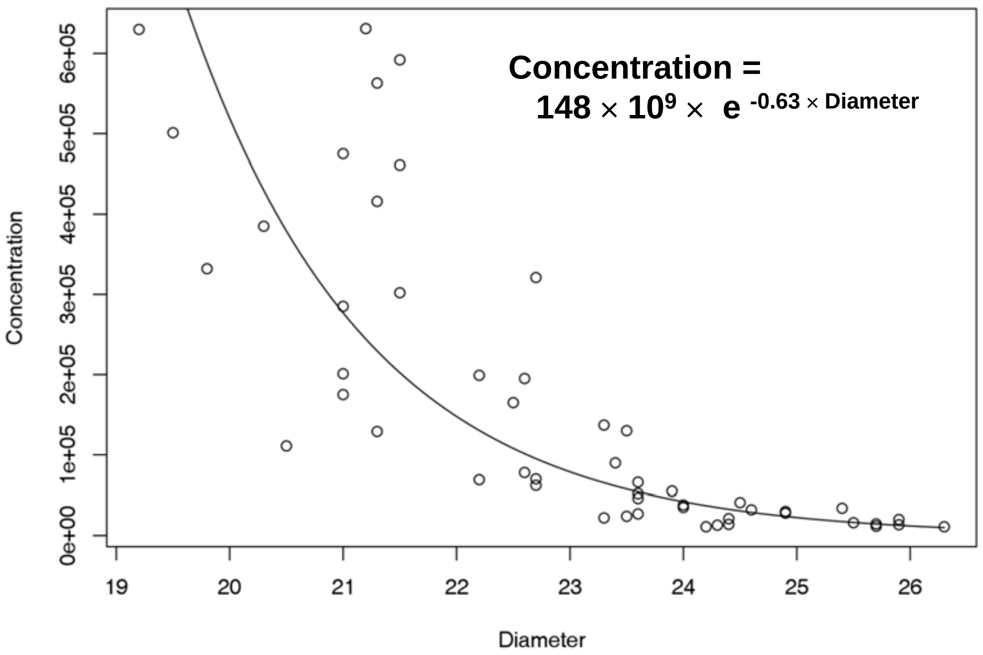
```
summary(modellog)

    Call:
    lm(formula = logconc ~ diameter)

    Residuals:
          Min        1Q    Median        3Q       Max 
    -1.227992 -0.388761  0.003015  0.424183  1.215852 

    Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
    (Intercept) 25.72239    1.09418   23.51   <2e-16 ***
    diameter    -0.62815    0.04743  -13.24   <2e-16 ***
    ---
    Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

    Residual standard error: 0.6105 on 49 degrees of freedom
    Multiple R-squared: 0.7817,     Adjusted R-squared: 0.7772 
    F-statistic: 175.4 on 1 and 49 DF,  p-value: < 2.2e-16
```

Residuals vs fitted values





Predicting Concentration from diameter



Concentration =

$$148 \times 10^{9} \times e^{-0.63 \times \text{Diameter}}$$

What we have done so far

We have a linear model for predicting the log of the concentration:

$$\log(\text{concentration}) = 25.7 - 0.63 \times \text{diameter}$$

We have a function that links this prediction to our value of interest (concentration):

log / exponential

This allows us to make predictions for the concentration:

$$\text{Concentration} = 148 \times 10^{9} \times e^{-0.63 \times \text{Diameter}}$$

The Hellung data in R

```
> hellung
   glucose    conc diameter
1        1 631000     21.2
2        1 592000     21.5
3        1 563000     21.3
4        1 475000     21.0
5        1 461000     21.5
[...]
33       2 630000     19.2
34       2 501000     19.5
35       2 332000     19.8
36       2 285000     21.0
37       2 201000     21.0
```

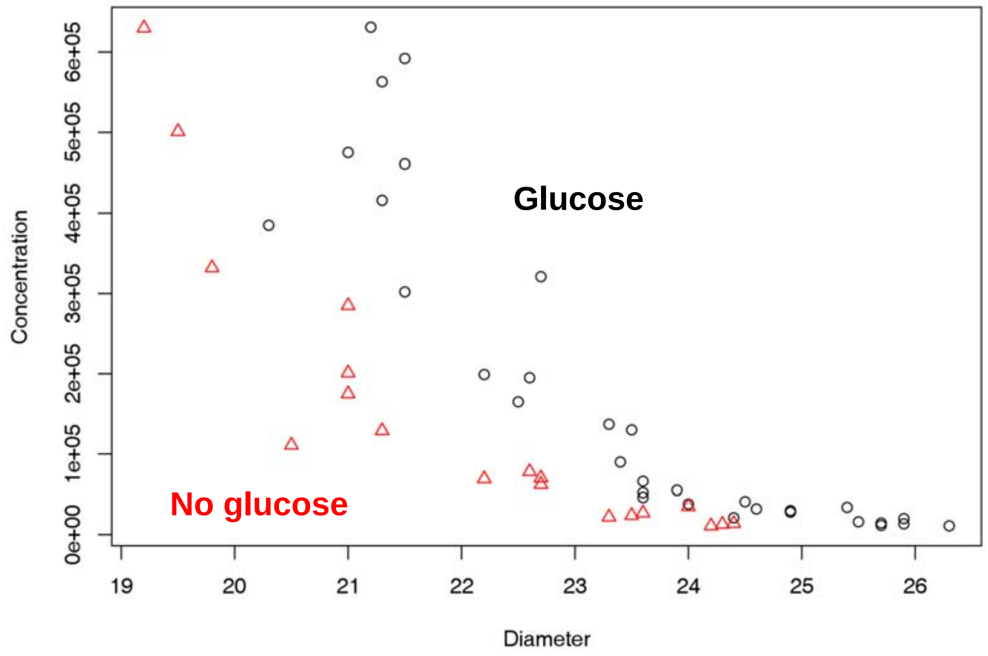
R help for the Hellung data

```
hellung                 package:ISwR                 R Documentation

Growth of Tetrahymena cells

Description:

     The ‘hellung’ data frame has 51 rows and 3 columns.  diameter and
     concentration of _Tetrahymena_ cells with and without glucose
     added to growth medium.

Format:

     This data frame contains the following columns:

     ‘glucose’ a numeric vector code, 1: yes, 2: no.

     ‘conc’ a numeric vector, cell concentration (counts/ml).

     ‘diameter’ a numeric vector, cell diameter (micrometre).

Source:

     D. Kronborg and L.T. Skovgaard (1990), _Regressionsanalyse_, Table
     1.1, FADLs Forlag (in Danish).
```

R help for the Hellung data

```
hellung                 package:ISwR                 R Documentation

Growth of Tetrahymena cells

Description:

     The ‘hellung’ data frame has 51 rows and 3 columns.  diameter and
     concentration of _Tetrahymena_ cells with and without glucose
     added to growth medium.

Format:

     This data frame contains the following columns:

     ‘glucose’ a numeric vector code, 1: yes, 2: no.

     ‘conc’ a numeric vector, cell concentration (counts/ml).

     ‘diameter’ a numeric vector, cell diameter (micrometre).

Source:

     D. Kronborg and L.T. Skovgaard (1990), _Regressionsanalyse_, Table
     1.1, FADLs Forlag (in Danish).
```

Concentration according to Diameter and Glucose



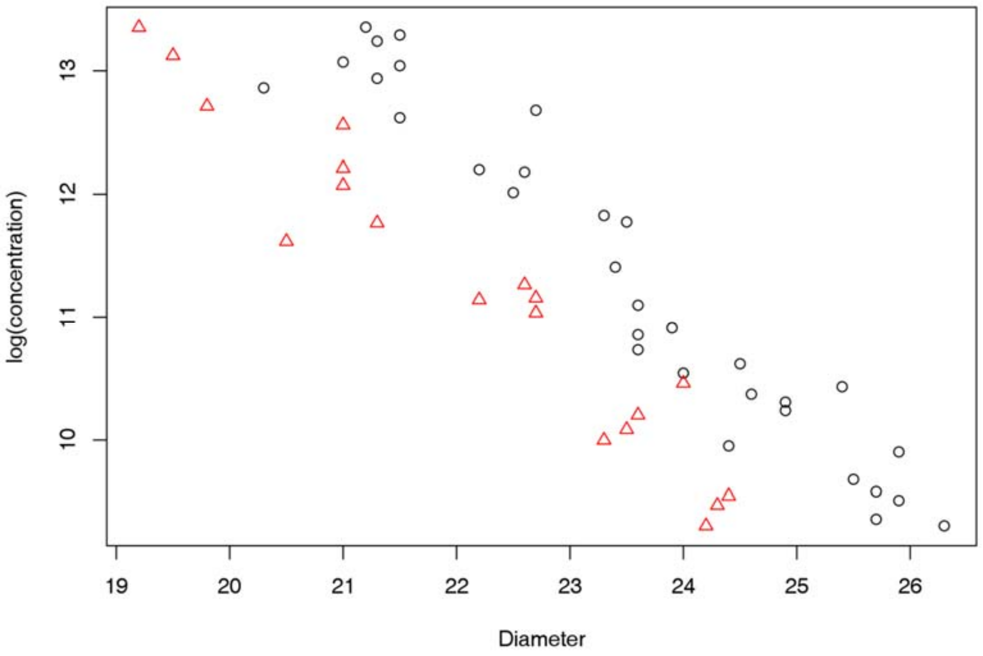
Reminder: using categorical variables as explanatory variables

We would like to use categorical variables in a linear model, as in:

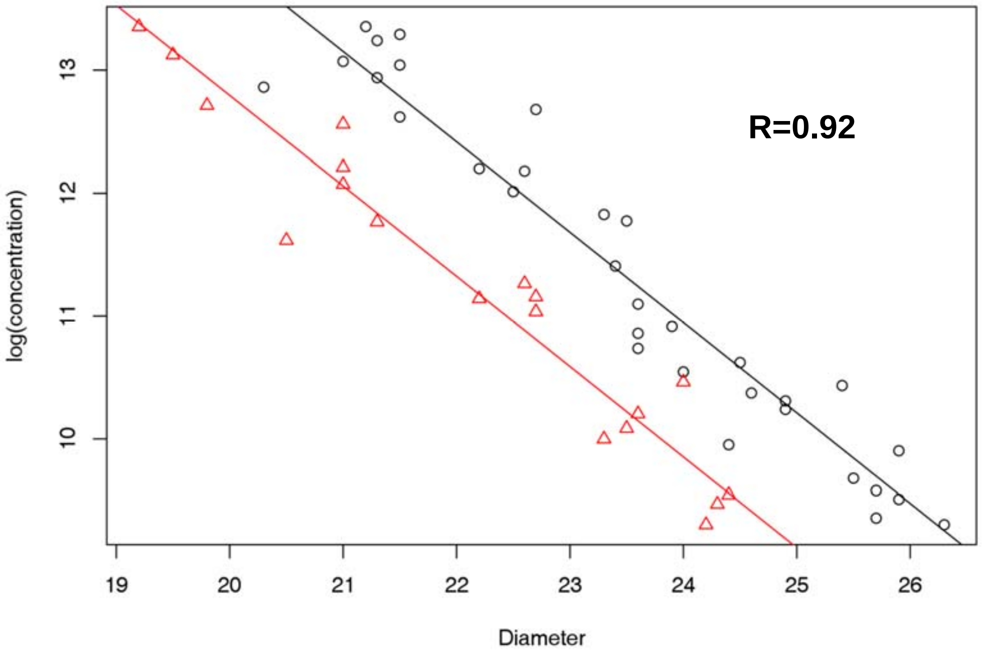
Concentration = b_0 + b_1 Diameter + b_2 « Glucose » + error

Intuitively, we want to estimate a « No glucose » and a « Glucose » effect.

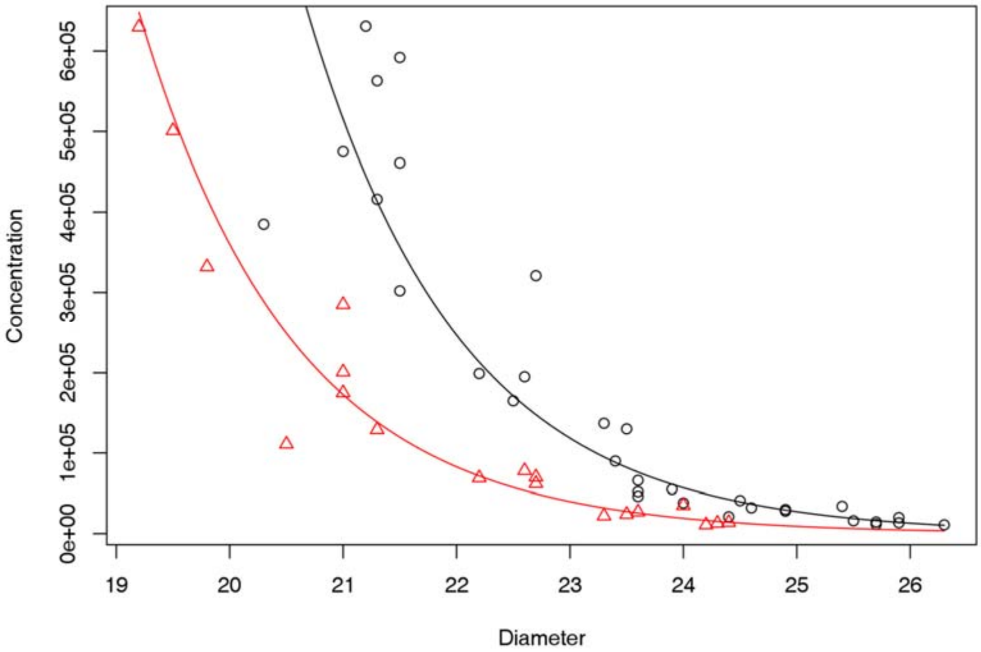
Log(concentration) according to diameter and glucose



Prediction of log Concentration according to Diameter and Glucose



Prediction of Concentration according to Diameter and Glucose

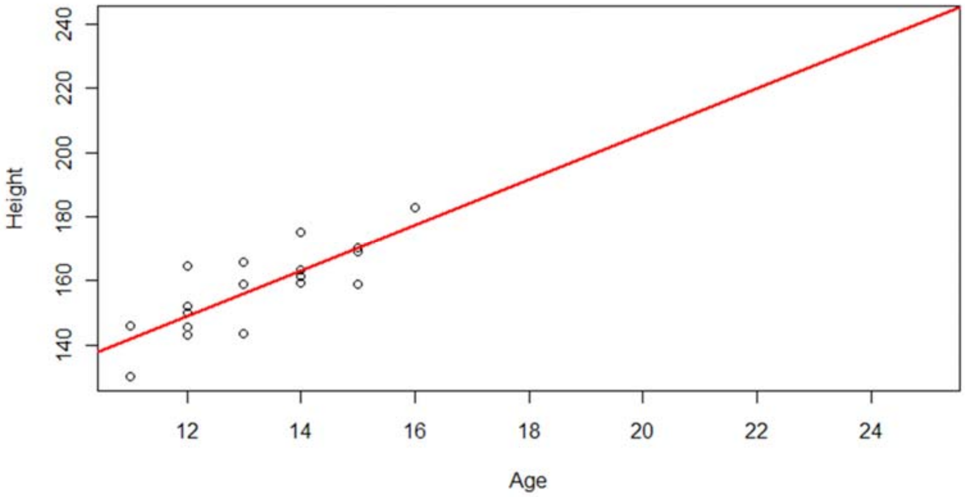


Pitfalls in regression

Pitfalls in regression: Extrapolation

We don’t know what the relationship between X and Y looks like outside the range of the data.

Extrapolating the model outside of this range is likely to give meaningless results.


Height
Age