Stats with Sparrows - 10

Julia Schroeder

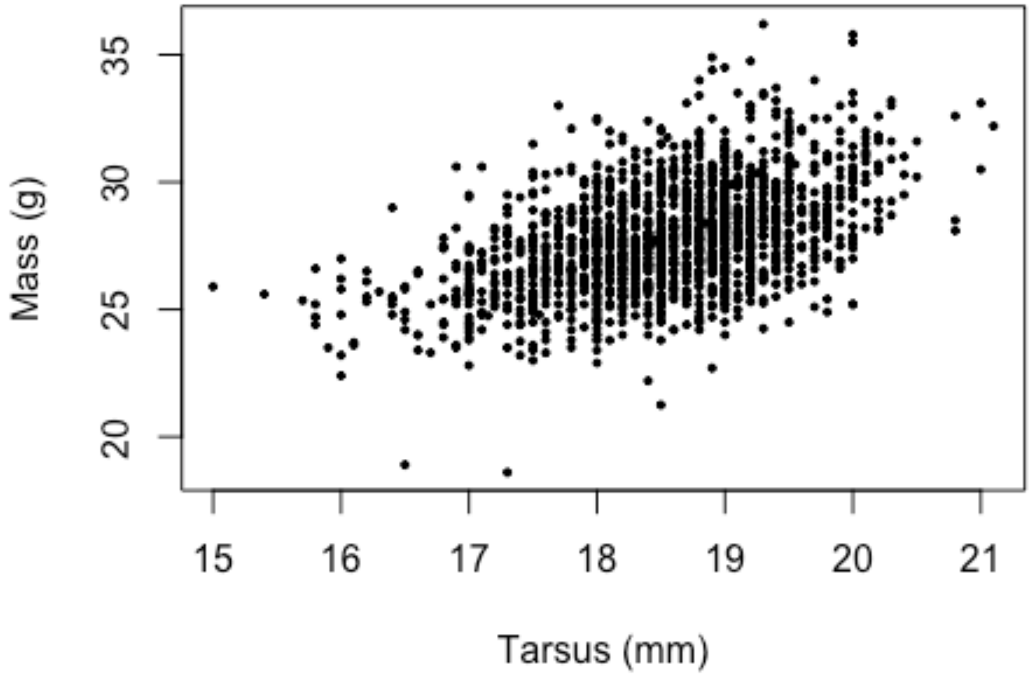
10

Again: housekeeping!

```
rm(list=ls())
setwd("H:/StatsWithSparrows")

d<-read.table("SparrowSize.txt", header=TRUE)
```

Now, we'll do a bold move. We move straight to linear models. Linear models include things like correlations. Correlations are associations between two variables. For instance, sparrows that are larger, may also be heavier:

```
plot(d$Mass~d$Tarsus, ylab="Mass (g)", xlab="Tarsus (mm)", pch=19, cex=0.4)
```



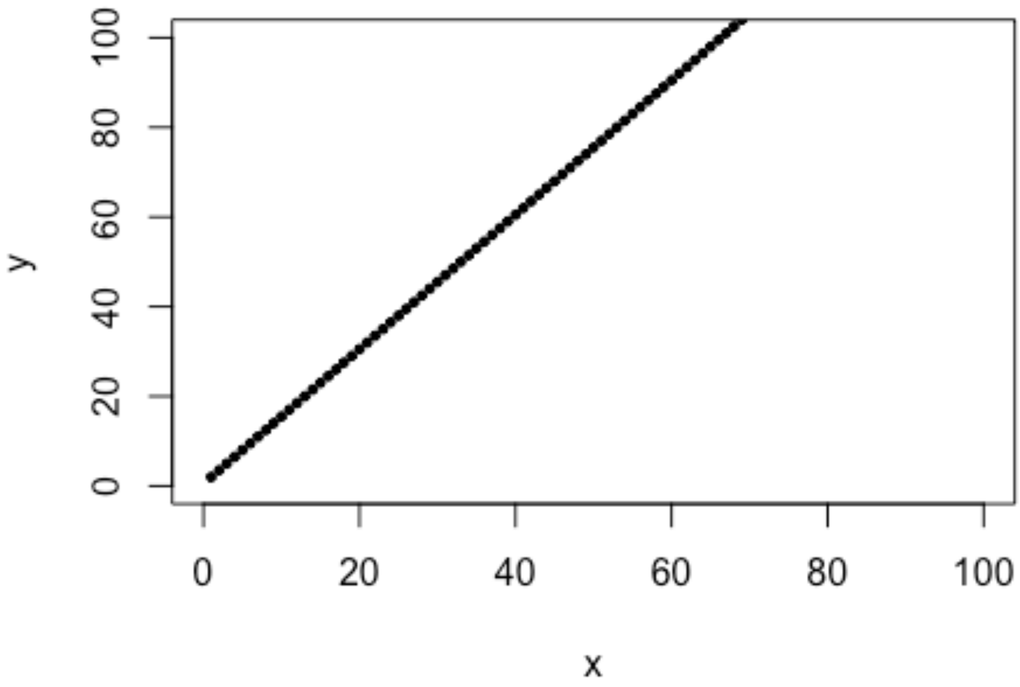
So, there clearly seems to be a linear relationship there. How can we describe a line in mathematical terms?

$$y = b + mx$$

This is what you've learned in school, hopefully. Do you remember what the respecctive parameters are?

Y is the y-coordinate, x the x-coordiates. b is the intercept, that's where the line crosses the y-axis. M is the slope, that defines how steep the line is, and whether negative (decrease) or positive (increase). The slope can be interpreted as increasing by m for each 1 x. You can then find all y and x that fit those criteria, and they will all fall on a line. I can demonstrate that easily in R:

```
x<-c(1:100)
b<-0.5
m<-1.5
y<-m*x+b
plot(x,y, xlim=c(0,100), ylim=c(0,100), pch=19, cex=0.5)
```



In statistics, we use such a line equation to describe the linear association between two variables. We use a slightly different equation, though:

$y_i = b_0 + b_1 x_i + \varepsilon_i$

First, we note the *i*'s at the right hand bottom of *y* and *x*. They denote that there is a population of data running from 1 to i observations.

In our case, y1 is

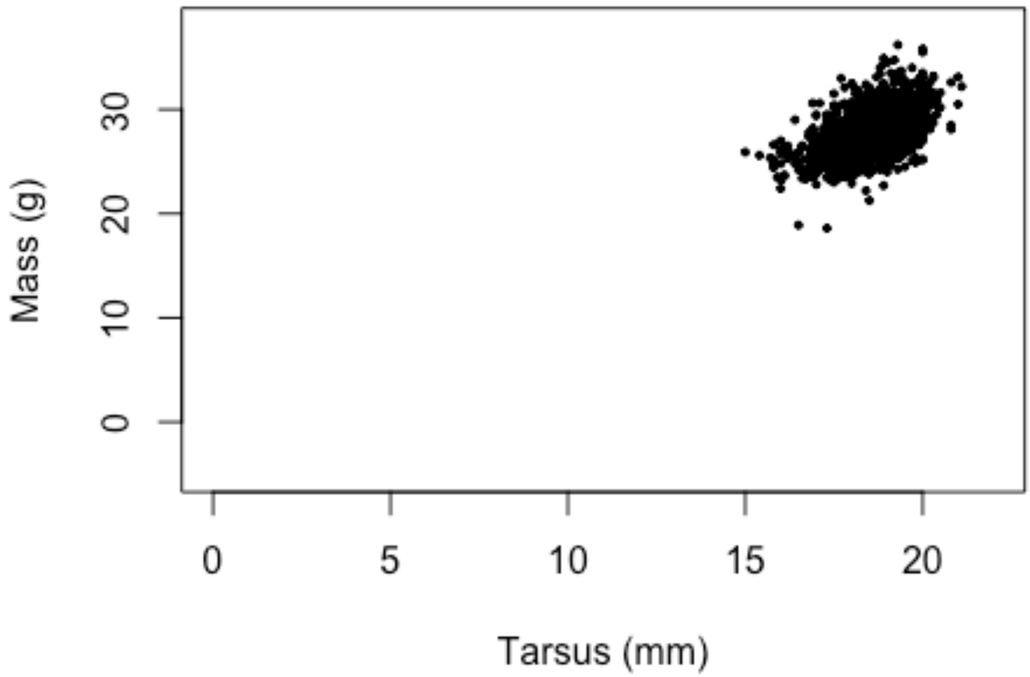
```
d$Mass[1]

## [1] 29.4
```

the maximum i, and respective yi are

```
length(d$Mass)

## [1] 1770

d$Mass[1770]

## [1] 33
```

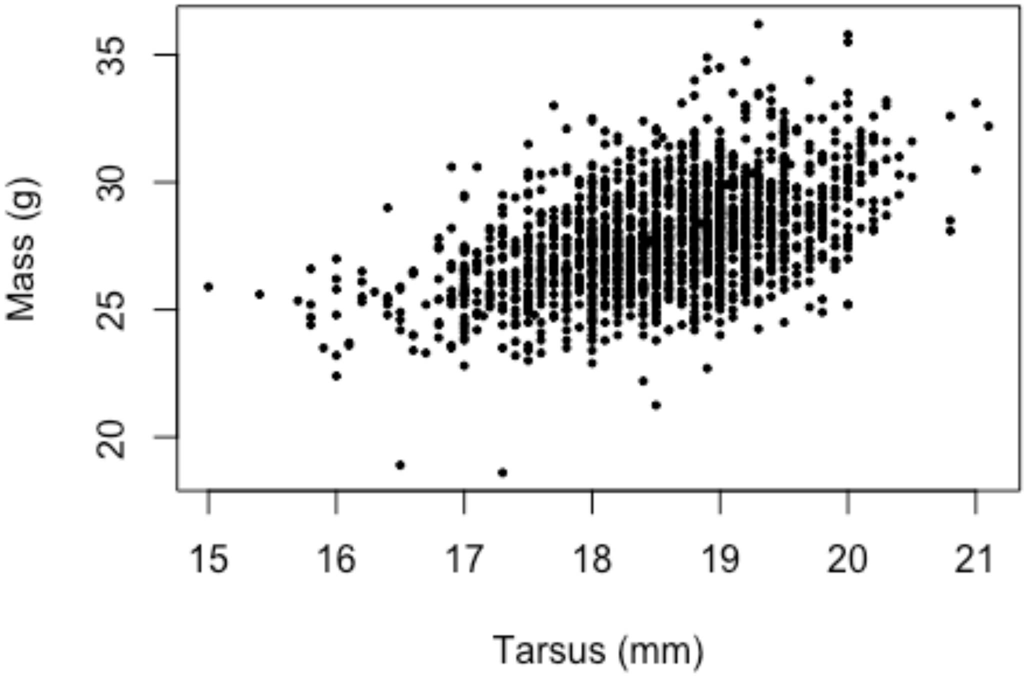
Then, there are b_0, which is the intercept, and b_1, which is the slope. These stay the same for all *i*s. These two are the parameters we want to estimate, to get a grip at the association between mass and tarsus. From looking at the plot, we can give it a guess (after we've plotted it such that we can see the 0 of the *x*axis, which is where the intercept is evaluated):

```
plot(d$Mass~d$Tarsus, ylab="Mass (g)", xlab="Tarsus (mm)", pch=19, cex=0.4,
ylim=c(-5,38), xlim=c(0,22))
```



Squinting a bit at it makes me think that the intercept is somewhere between -5 and 10. The slope is more difficult. We can look at the difference it makes for 5 mm - that's about the difference between 20 and 30 gramm, maybe a bit less. Very roughly. It might be easier to estimate on the original plot:

```
plot(d$Mass~d$Tarsus, ylab="Mass (g)", xlab="Tarsus (mm)", pch=19, cex=0.4)
```



Yes, so 10 gramm difference for each 5mm in Tarsus. Scaling down to 1mm Tarsus, that gives a change in mass of 10/5 = 2g, rather less than that, say, 1.6. So, we can predict that our equation should look like this, where $b_0 = 5$-ish and $b_{1}=2. We insert that into our equation:

$y_i = b_0 + b_1 x_i + \varepsilon_i$

$y_i = 5 + 1.6 x_i + \varepsilon_i$

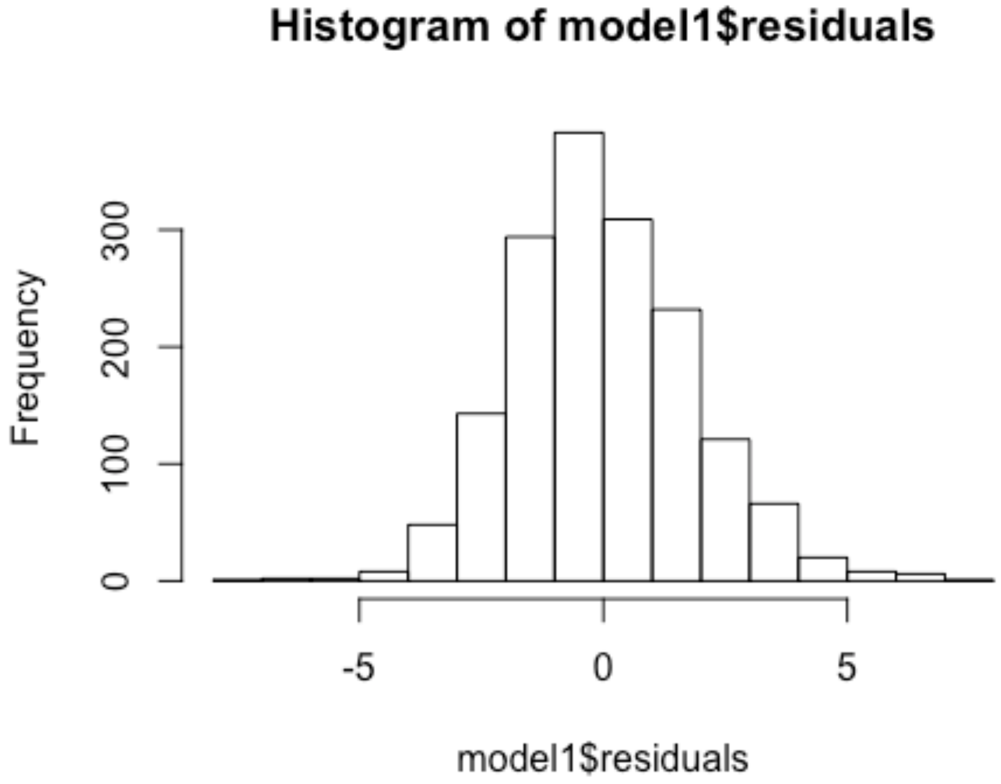
Umm, there's this odd epsilon that we've conveniently ignored until now. What's with this? Looking back at our plot, it is clear that our data does also do other things. We not only want to quantify the direction and steepness of the association, but also the spread. THis spread is the ERROR - the noise around the straight line. That's the epsilon - there is one error term, or *residual* for each observation. This residual describes how far the point is away from the actual line. The line is fitted such that these residuals take on the smalles possible value. To be completely honest, we don't take the simple dictance from the line to the point. No, we use the distance that is vertical (parralel to y), and then, we take the sum of the squares of them, - the sum of the squared residuals. Then, we wiggle the line so that the sum of squares (sounds familiar?), takes on the smalles value possible for these data points. We call this the *least square* method. $\sum(y_i$-$\hat{y}_i)$, where $\hat{y}_i)$ is the *i-th* y point that one calculates from the equation of the fitted line, with the respective x_i data point. This is all tremendously important. If you struggle understanding this, do ask, or use the Wednesday to ask your question and revisit this.

Now, we'll harness the powers of R to see how this looks like in real life. First, of course, we strip the dataset to one that does not contain missing values:

```
d1<-subset(d, d$Mass!="NA")
d2<-subset(d1, d1$Tarsus!="NA")
length(d2$Tarsus)

## [1] 1644

model1<-lm(Mass~Tarsus, data=d2)
summary(model1)

##
## Call:
## lm(formula = Mass ~ Tarsus, data = d2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7271 -1.2202 -0.1302  1.1592  7.5036
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.83246    0.98195    5.94 3.48e-09 ***
## Tarsus       1.18466    0.05295   22.37  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.841 on 1642 degrees of freedom
## Multiple R-squared:  0.2336, Adjusted R-squared:  0.2332
## F-statistic: 500.6 on 1 and 1642 DF,  p-value: < 2.2e-16
```

Let's examine this result. First, R tells us what we asked it to do, the call. That's easy. Then, it tells us about the residuals - you should know what this second report means. It tells us about the distribution of the residuals. We can actually access the residuals:

```
hist(model1$residuals)
```



```
head(model1$residuals)

##         1         2         3         4         5         6 
## 1.1774836 3.4959507 1.4405508 3.2590180 2.5405512 0.9436773
```

Remember this for later.

Now on the the coefficients of the equation. There are *Parameter estimate*, *Standard Errors*, *t-values*, and *p-values* of each of the *b*s - the intercept (b_0) and the slope for Tarsus ($b1$).

We've been somewhat wrong with our slope guesstimate, it was even smaller than we thought. But the intercept was well guessed. We can look at the standard errors, and t-values (do you know now where these come from?). The data for the intercept tells us it's statistically significantly different from zero. That's not really interesting, because we are not interested in the body mass of a bird with a tarsus of 0 length. Most of the time, when you do statistics, the intercept is not interesting. However, it pays if you recall what it really is. Later, we will do this analysis with z-standardized tarsus, and then the intercept is all of the sudden quite interesting. The slope is statistically significantly different from zero, with p<0.001. That's cool. It's in the ballpark of 1.2 - for 1 mm increase in tarsus, birds are about 1g heavier. Good. What's the rest of the stuff reported here?

There are the degrees of freedom. Let's check:

So, there are 1644 observations, and the *df*s are 1642. That means, R estimated 2 parameters, one for the intercept, and one for the tarsus. The standard error of the residuals is also estimated.

But, there is more. What's the R-squared stuff? We just look at the first one - that' s 0.23. That means, and here we finally get to the spread, that 23% of the variance in mass is explained by variation in tarsus. If this value is 1 (100%), we'd see a straight line. We can test this using the x/y data we generated before, with an intercept of 0.5, and a slope of 1.5:

```
model2<-lm(y~x)
summary(model2)

## Warning in summary.lm(model2): essentially perfect fit: summary may be
## unreliable

##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.372e-13 -1.237e-15  1.230e-15  3.407e-15  2.160e-14
##
## Coefficients:
##              Estimate Std. Error   t value Pr(>|t|)
## (Intercept) 5.000e-01  2.891e-15 1.729e+14   <2e-16 ***
## x           1.500e+00  4.971e-17 3.018e+16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.435e-14 on 98 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 9.106e+32 on 1 and 98 DF,  p-value: < 2.2e-16
```

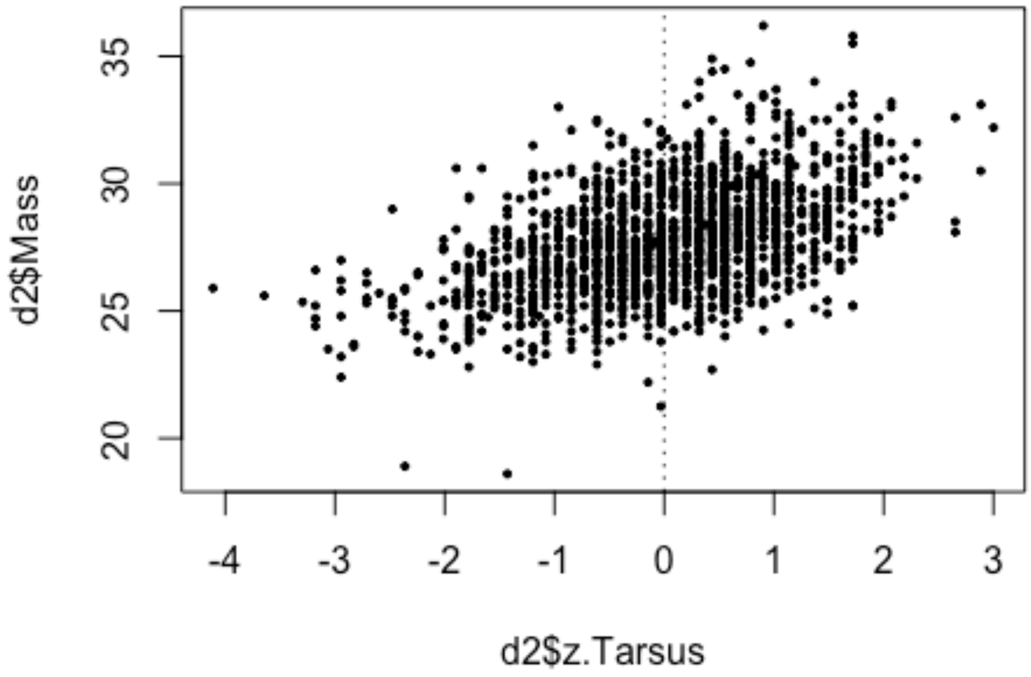
Back to the sparrow data. We learned before about z-scores, and standardization. And I've said it is commonplace in statistics to z-standardize the covariate - that is, the continous predictor variable. Why? Let's run the model with z-scores of Tarsus instead of Mass:

```
d2$z.Tarsus<-scale(d2$Tarsus)
model3<-lm(Mass~z.Tarsus, data=d2)
summary(model3)

##
## Call:
## lm(formula = Mass ~ z.Tarsus, data = d2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7271 -1.2202 -0.1302  1.1592  7.5036
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 27.77895    0.04539  611.94   <2e-16 ***
## z.Tarsus     1.01596    0.04541   22.37   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.841 on 1642 degrees of freedom
## Multiple R-squared:  0.2336, Adjusted R-squared:  0.2332
## F-statistic: 500.6 on 1 and 1642 DF,  p-value: < 2.2e-16
```

When we now look at the estimates, what does the Intercept reflect? Remember, it is where the regression line crosses 0 on the x-axis. It becomes more apparent when we plot it:

```
plot(d2$Mass~d2$z.Tarsus, pch=19, cex=0.4)
abline(v = 0, lty = "dotted")
```



And what does the slope reflect now?

Ok, let's see...

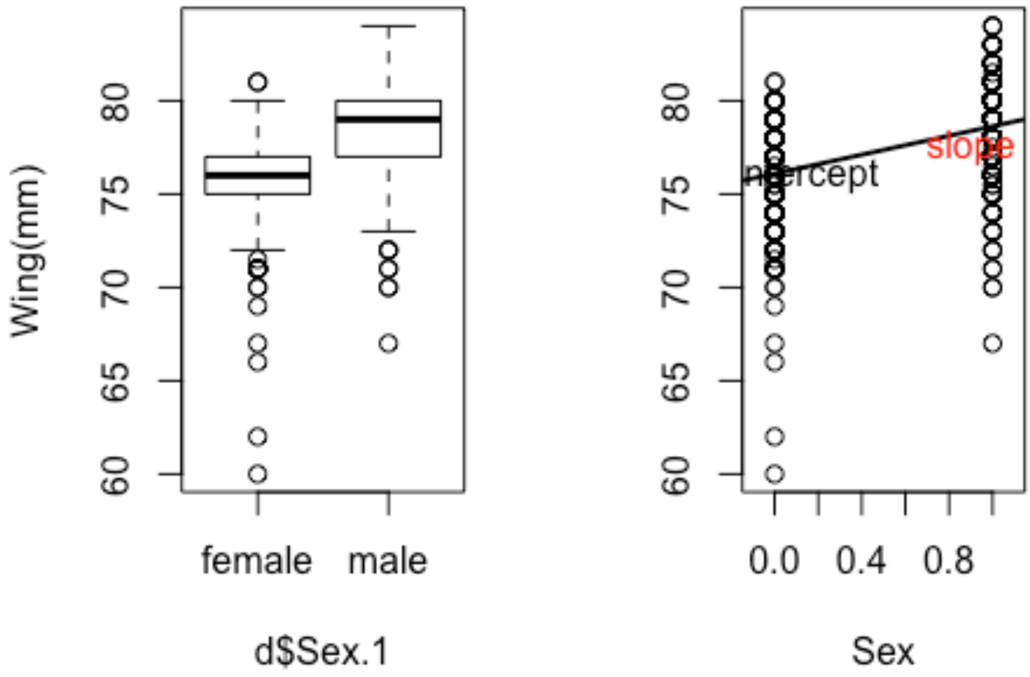
```
head(d)
```

```
##   BirdID Cohort CaptureDate CaptureTime Year Tarsus Bill Wing Mass Sex
## 1   4401   1991   21-Jun-00        <NA> 2000   18.9   NA   82 29.4   1
## 2   4401   1991   02-Oct-00        <NA> 2000   18.8   NA   79 31.6   1
## 3   4405   1994   20-Jun-00        <NA> 2000   19.1   NA   77 29.9   0
## 4   4405   1994   04-Oct-00        <NA> 2000   19.0   NA   78 31.6   0
## 5   4405   1994   07-Oct-00        <NA> 2000   19.1   NA   77 31.0   0
## 6   4409   1994   23-Mar-00        <NA> 2000   18.0   NA   76 28.1   1
##    Sex.1
## 1   male
## 2   male
## 3 female
## 4 female
## 5 female
## 6   male
```

```
str(d)
```

```
## 'data.frame':    1770 obs. of  11 variables:
##   $ BirdID     : int  4401 4401 4405 4405 4405 4409 4409 4409 4409 4409 ...
##   $ Cohort     : int  1991 1991 1994 1994 1994 1994 1994 1994 1994 1994 ...
##   $ CaptureDate: Factor w/ 414 levels "01-Aug-06","01-Dec-07",..: 272 18
254 41 88 303 174 18 159 164 ...
##   $ CaptureTime: Factor w/ 293 levels "04:00","04:30",..: NA NA NA NA NA NA
NA NA NA NA ...
##   $ Year       : int  2000 2000 2000 2000 2000 2000 2000 2000 2001 2001 ...
##   $ Tarsus     : num  18.9 18.8 19.1 19 19.1 ...
##   $ Bill       : num  NA NA NA NA NA NA NA NA NA NA ...
##   $ Wing       : num  82 79 77 78 77 76 76 73 79 77 ...
##   $ Mass       : num  29.4 31.6 29.9 31.6 31 ...
##   $ Sex        : int  1 1 0 0 0 1 1 1 1 1 ...
##   $ Sex.1      : Factor w/ 2 levels "female","male": 2 2 1 1 1 2 2 2 2 2
...
```

```
d$Sex<-as.numeric(d$Sex)
par(mfrow = c(1, 2))
plot(d$Wing ~ d$Sex.1, ylab="Wing(mm)")
plot(d$Wing ~ d$Sex, xlab="Sex", xlim=c(-0.1,1.1), ylab="")
abline(lm(d$Wing ~ d$Sex), lwd = 2)
text(0.15, 76, "intercept")
text(0.9, 77.5, "slope", col = "red")
```



Can you explain how the t-test, that as we know, tests for a statistically different from zero difference between two means, can be used to test for linear models?

```
d4<-subset(d, d$Wing!="NA")
m4<-lm(Wing~Sex, data=d4)
t4<-t.test(d4$Wing~d4$Sex, var.equal=TRUE)
summary(m4)

##
## Call:
## lm(formula = Wing ~ Sex, data = d4)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.0961  -1.0961  -0.0961   1.3683   5.3683
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 76.09611    0.07175 1060.50   <2e-16 ***
## Sex          2.53562    0.09998   25.36   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

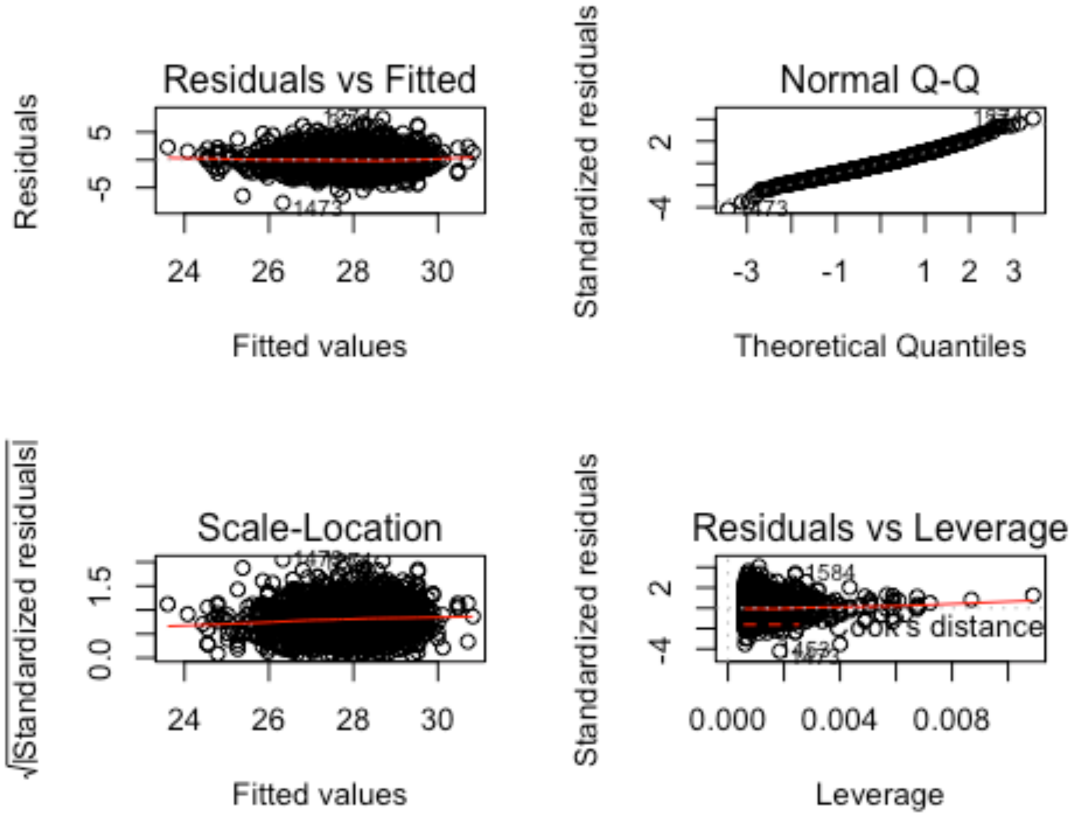
```
## Residual standard error: 2.057 on 1693 degrees of freedom
## Multiple R-squared:  0.2753, Adjusted R-squared:  0.2749
## F-statistic: 643.1 on 1 and 1693 DF,  p-value: < 2.2e-16
```

```
t4
```

```
##
##  Two Sample t-test
##
## data:  d4$Wing by d4$Sex
## t = -25.36, df = 1693, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.731727 -2.339518
## sample estimates:
## mean in group 0 mean in group 1
##        76.09611        78.63173
```

Before we end this, we haeve to do some linear model diagnostics. This is to make sure the assumtions that one has to make for a linear model to hold, are actually met. The most important assumption of linear models is that the residuals (!) are normally distributed. We can only test for that after we've fitted the model, because that's how we get the residuals.

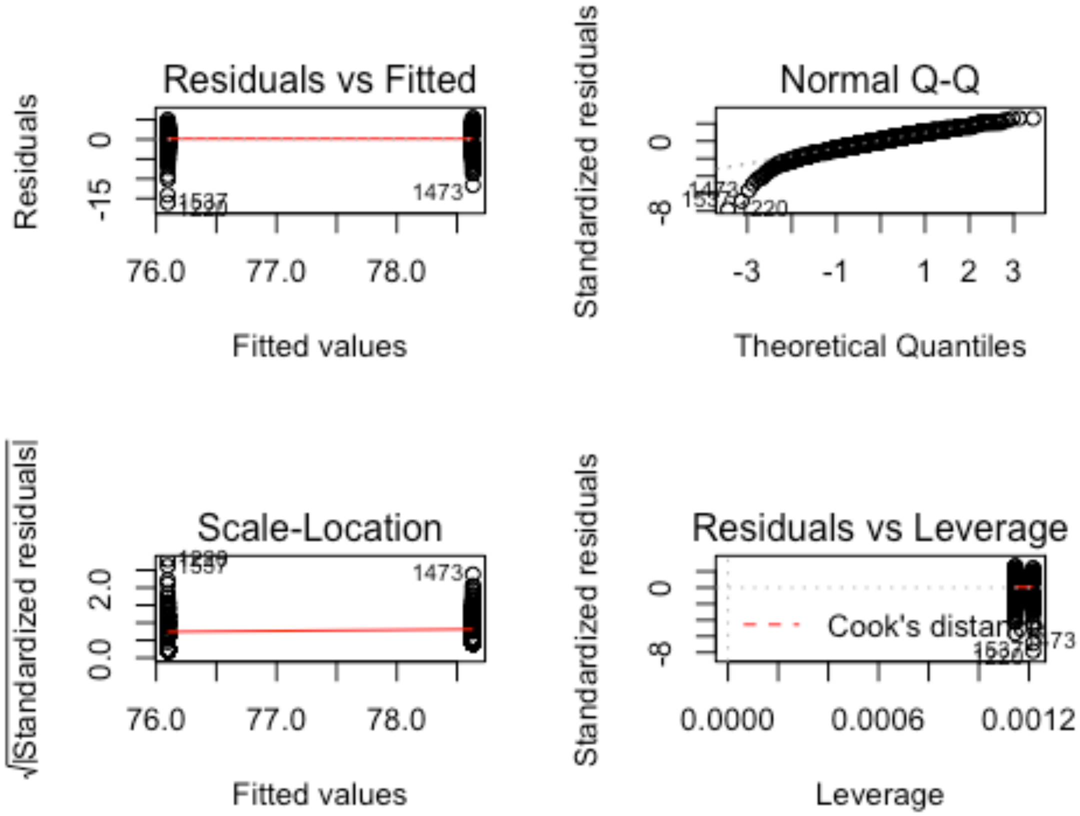
```
par(mfrow=c(2,2))
plot(model3)
```



Here, we want to see the top left plot. It shows the residuals (the distances between the values and the calculated regression line) on the y-axis, and the fitted values - those y's that you get when you calculate using the regression equation with b_0 and b_1. We want these to be distributed roughly random. Like "stars in the sky". This is an ok plot for this. We do not want patterns, that residuals get larger with increasing fitted values, or something like that. That's why R provides us with a nice red line, that shows some sort of deviations from y=0 (which is a super small residual).

The next plot right of the residual plot, the Q-Q plot, shows the standardised residuals (you should know by now what that means) plotted agains the quantiles they are supposed to fall into, assuming the residuals are normally distributed. You should know what this sentence means by now. This plot should look like a straight line, and we're happy with what we see here.

```r
par(mfrow=c(2,2))
plot(m4)
```



Can you explain why these two top plots test for the normality of residuals?

The bottom right plot shows the square root of the residuals - it's one you can look at, too. And the bottom left one is an interesting one - it shows the residuals in relationship to their leverage. How important some data points are in relationship to others.

Excercises:

Run diagnostics for a model with sex as explanatory variable. Interpret the plots.

Run a linear model, where you test the hypothesis that sparrows with bigger bills can eat more. The prediction is that the larger the bill, the heavier the sparrow.

Detail what your explanatory and what your response variable is. Write a short (1A4) report on methods and results. Before you go into the linear model, you should first describe your data, say how many sparrows, how many females and males, whether there is a difference in your response between the sexes. If that difference is meaningful, you should test the sexes separately. Write this section as you would write it for a scientific article.