Lecture 6: More Regression

Bob O'Hara

bob.ohara@ntnu.no

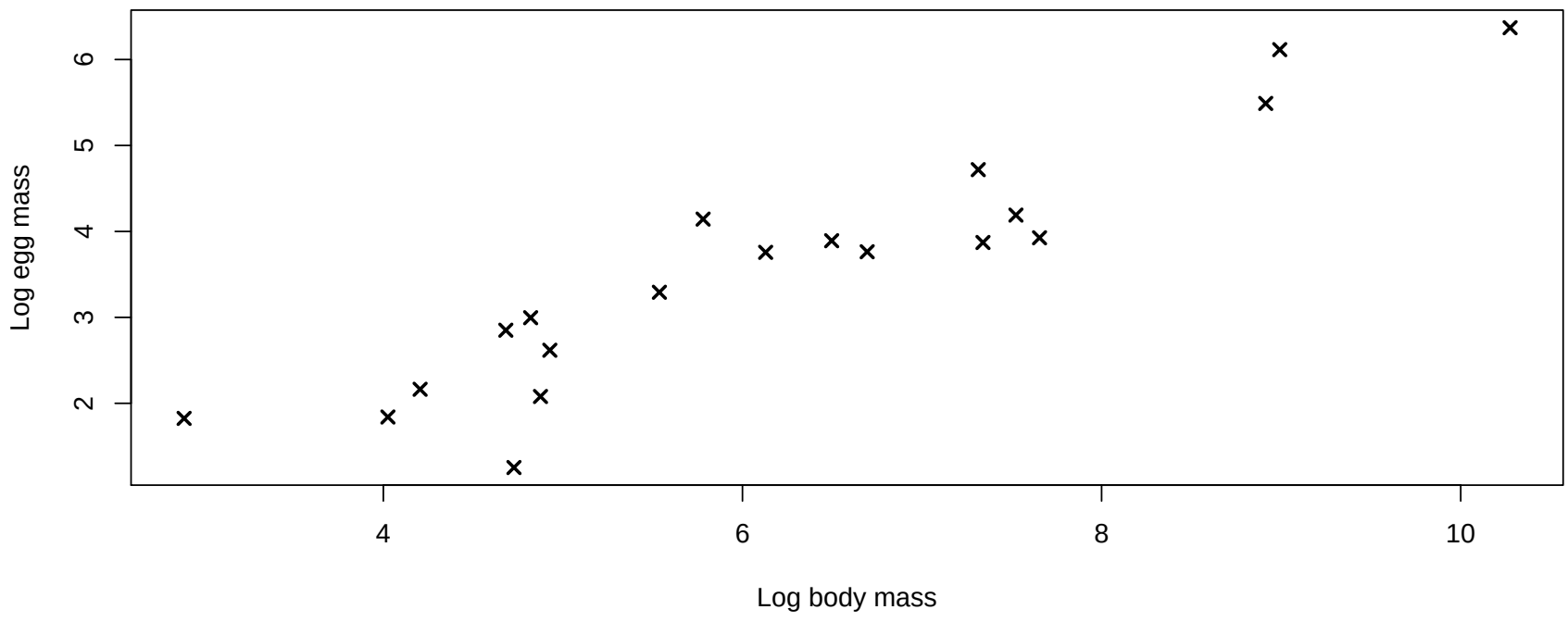
Regression & R

Read in the data (and sub-sample)

```
BirdEggs <- read.csv(file="../Data/BirdEggs.csv",
                     stringsAsFactors = FALSE)
# sub-sample 30 observations, just to make things clearer
BirdEggs <- BirdEggs[sample.int(nrow(BirdEggs), size=20),]
```

Plot the data

```
par(mar=c(4.1,4.1,1,1))
plot(BirdEggs$logFemaleMass, BirdEggs$logEggMass,
     xlab="Log body mass", ylab="Log egg mass",
     main="", lwd=2, col="black", pch=4)
```



The Model

The model is a straight line: we are regressing y against x

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

where x_i is the log body mass, and y_i is the log egg mass, and

$$\varepsilon_i \sim N(0, \sigma^2)$$

Fitting the Model in R

The code to fit the model is simple:

```
mod <- lm(logEggMass ~ logFemaleMass, data=BirdEggs)
```

- logEggMass ~ logFemaleMass is the formula that describes the model: Y~X means we regress Y against X
- data=BirdEggs just gives the data frame where the data are

The Parameter Estimates

```
round(coef(mod), 2)
```

```
##   (Intercept) logFemaleMass
##         -0.74           0.69
```

```
round(sigma(mod), 2)
```

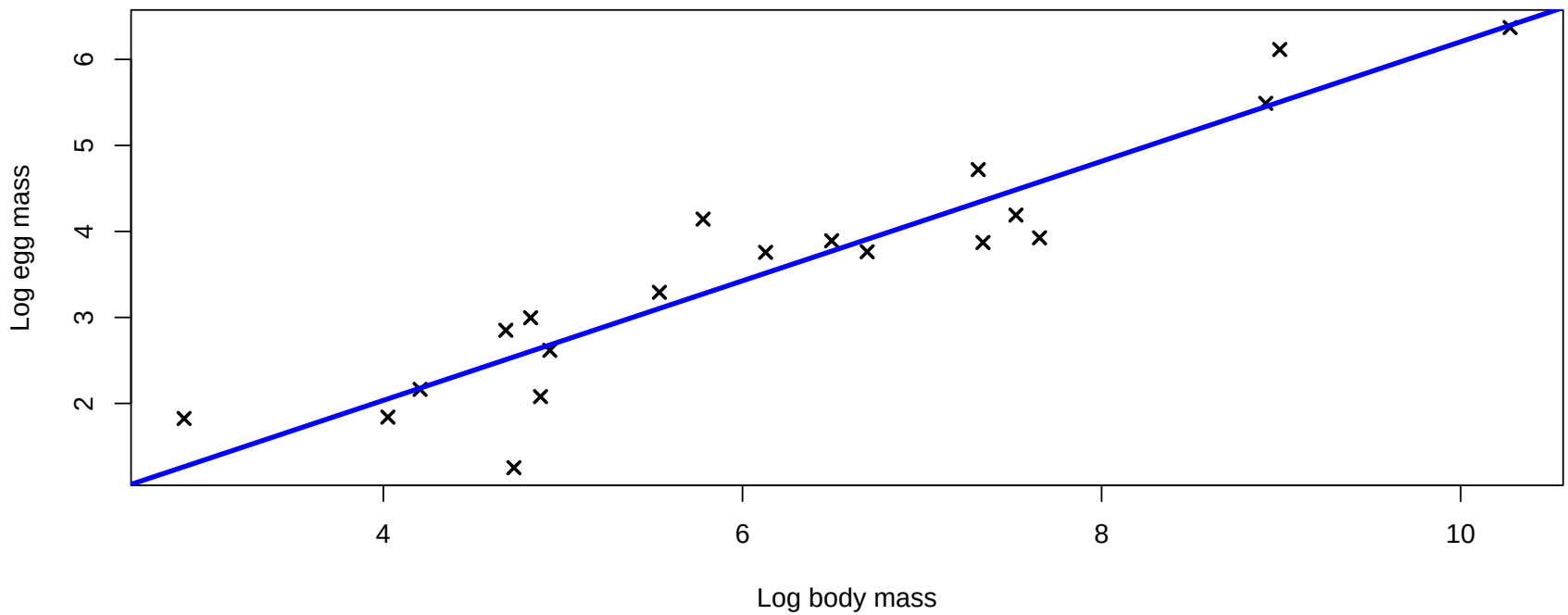
```
## [1] 0.51
```

So the model is

$$y_i = -0.74 + 0.69x_i + \varepsilon_i$$
$$\varepsilon_i \sim N(0, 0.51^2)$$

Plotting the Model

```
par(mar=c(4.1,4.1,1,1))
plot(BirdEggs$logFemaleMass, BirdEggs$logEggMass,
     xlab="Log body mass", ylab="Log egg mass",
     main="", lwd=2, col="black", pch=4)
abline(mod, col="blue", lwd=3)
```



Uncertainty in the parameters

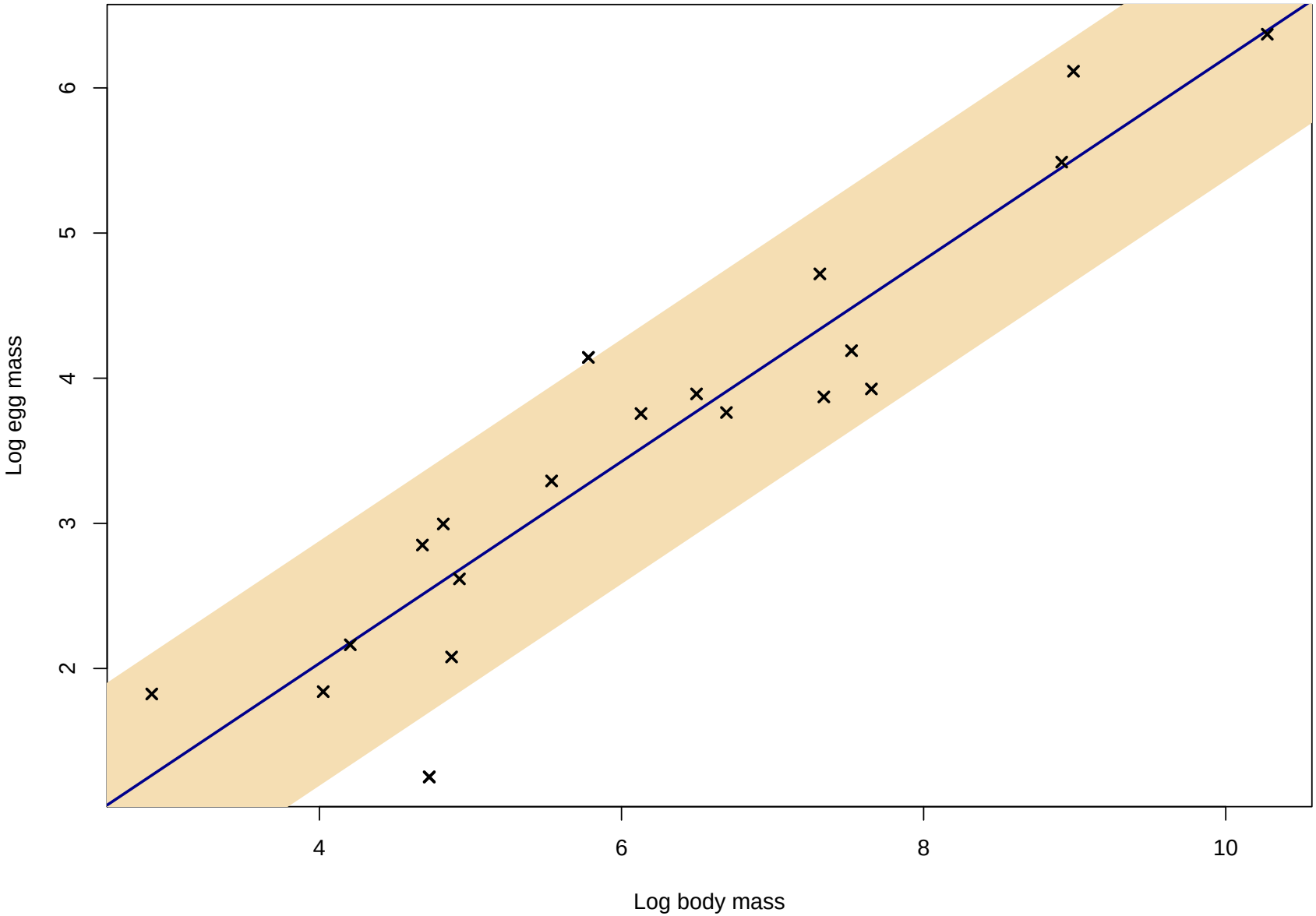
```
round(confint(mod), 2)
```

```
##                 2.5 % 97.5 %
## (Intercept)     -1.59   0.10
## logFemaleMass    0.56   0.83
```

Looking at uncertainty in the intercept

If we fix the slope, then we can look at variation in the intercept.

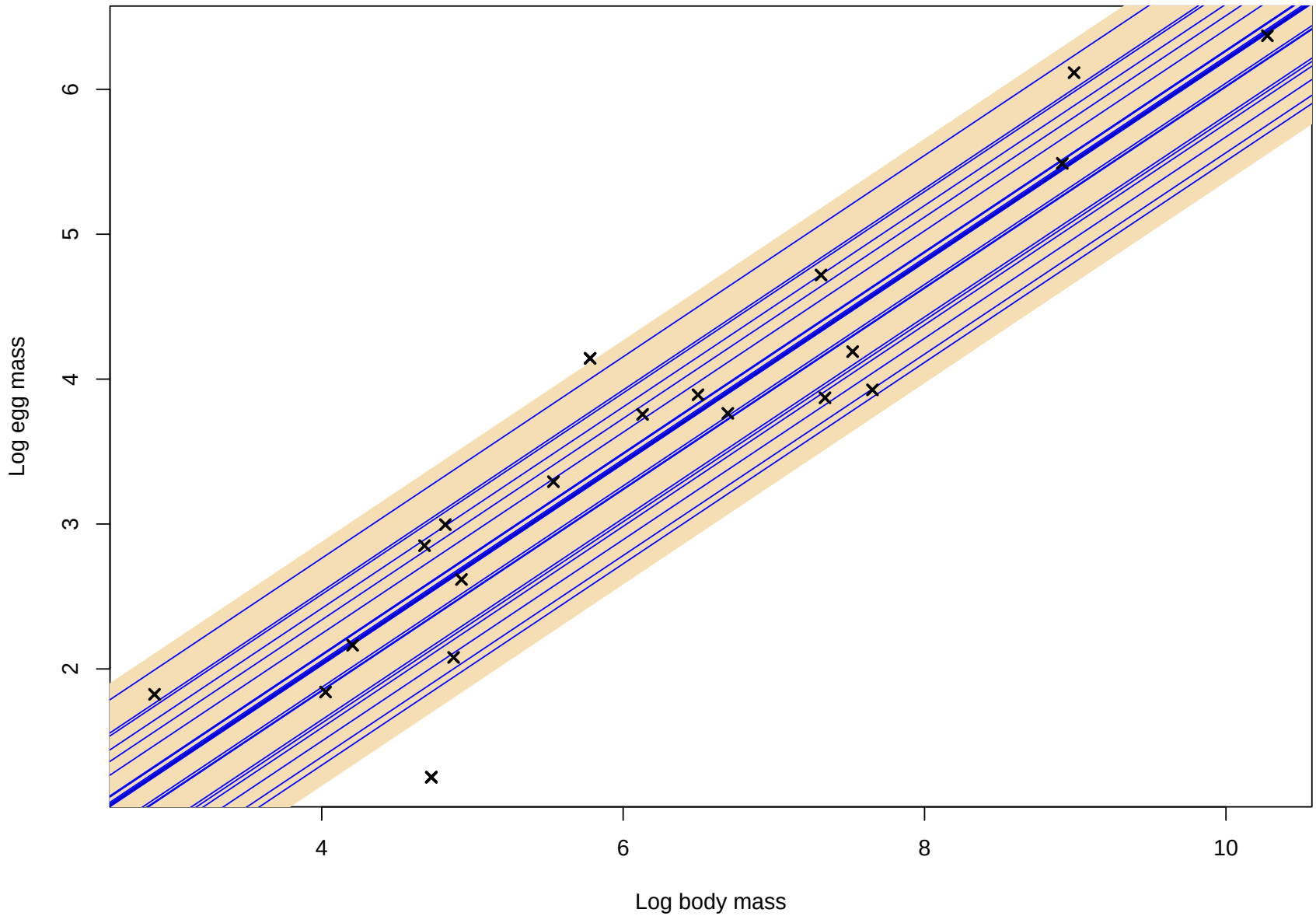
These are the upper & lower 95% confidence limits



Looking at uncertainty in the intercept

If we fix the slope, then we can look at variation in the intercept.

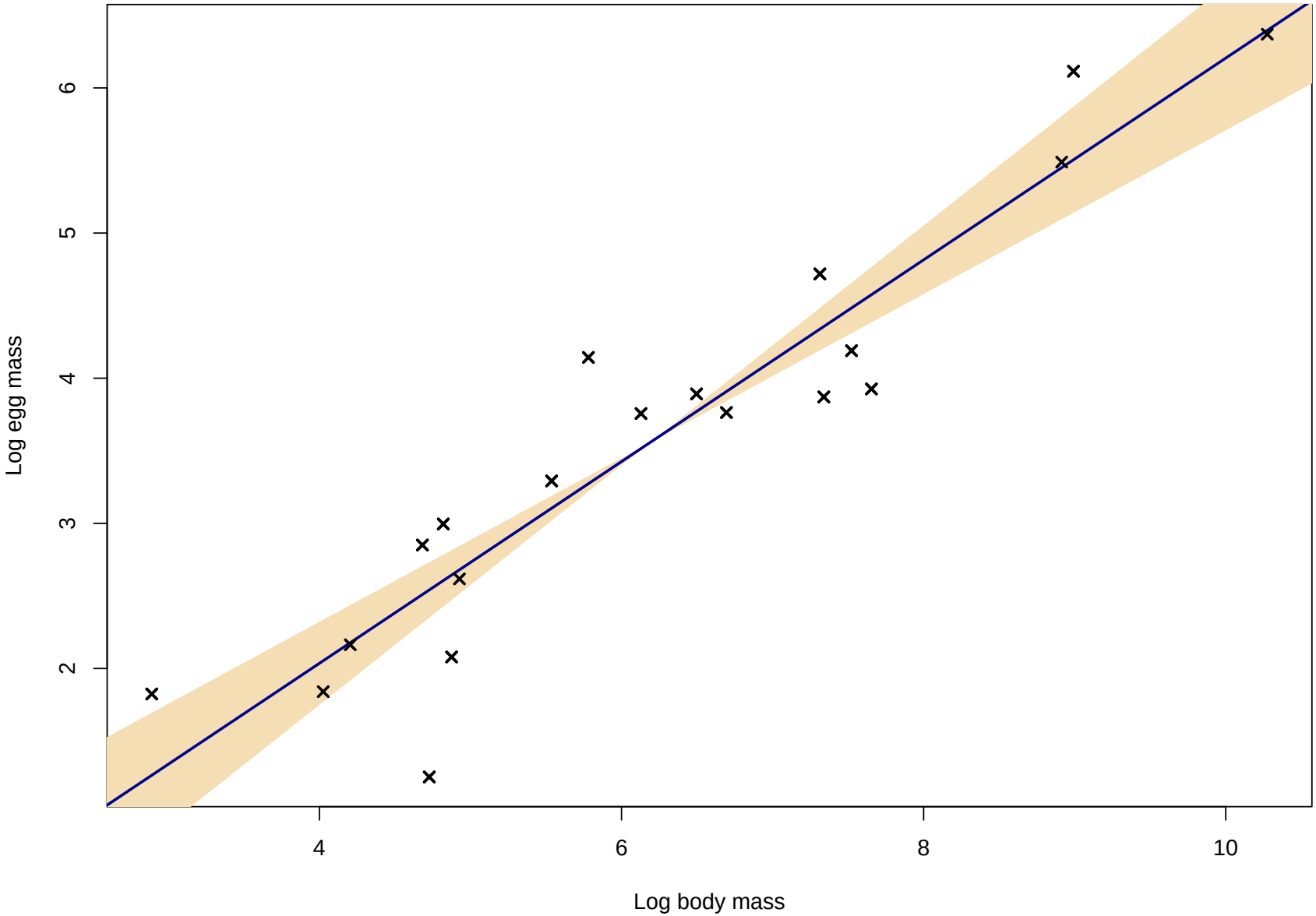
These are the upper & lower 95% confidence limits



Uncertainty in the slope

If we fix the intercept, then we can look at variation in the slope.

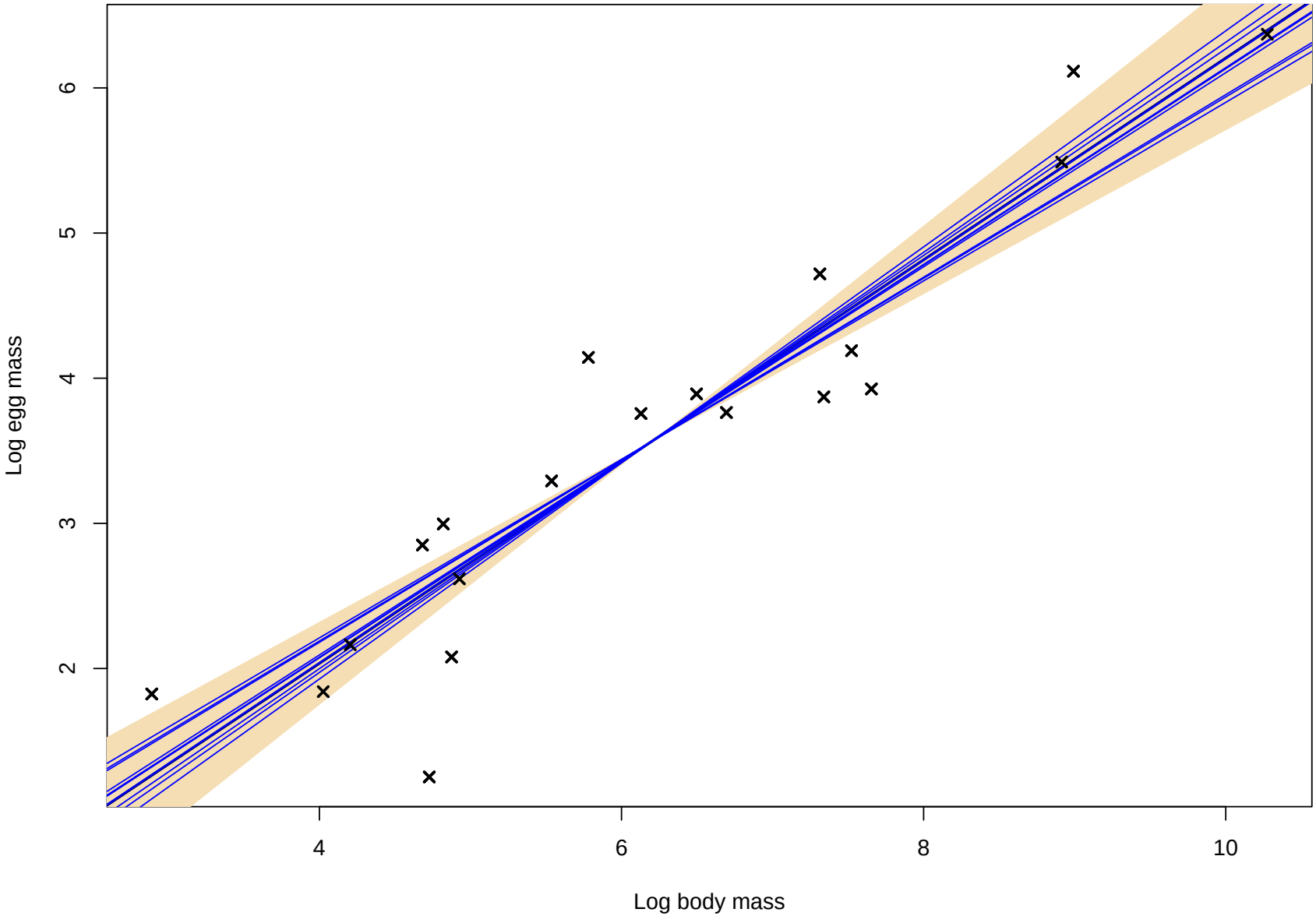
These are the uppr & lower 95% confidence limits



Uncertainty in the slope

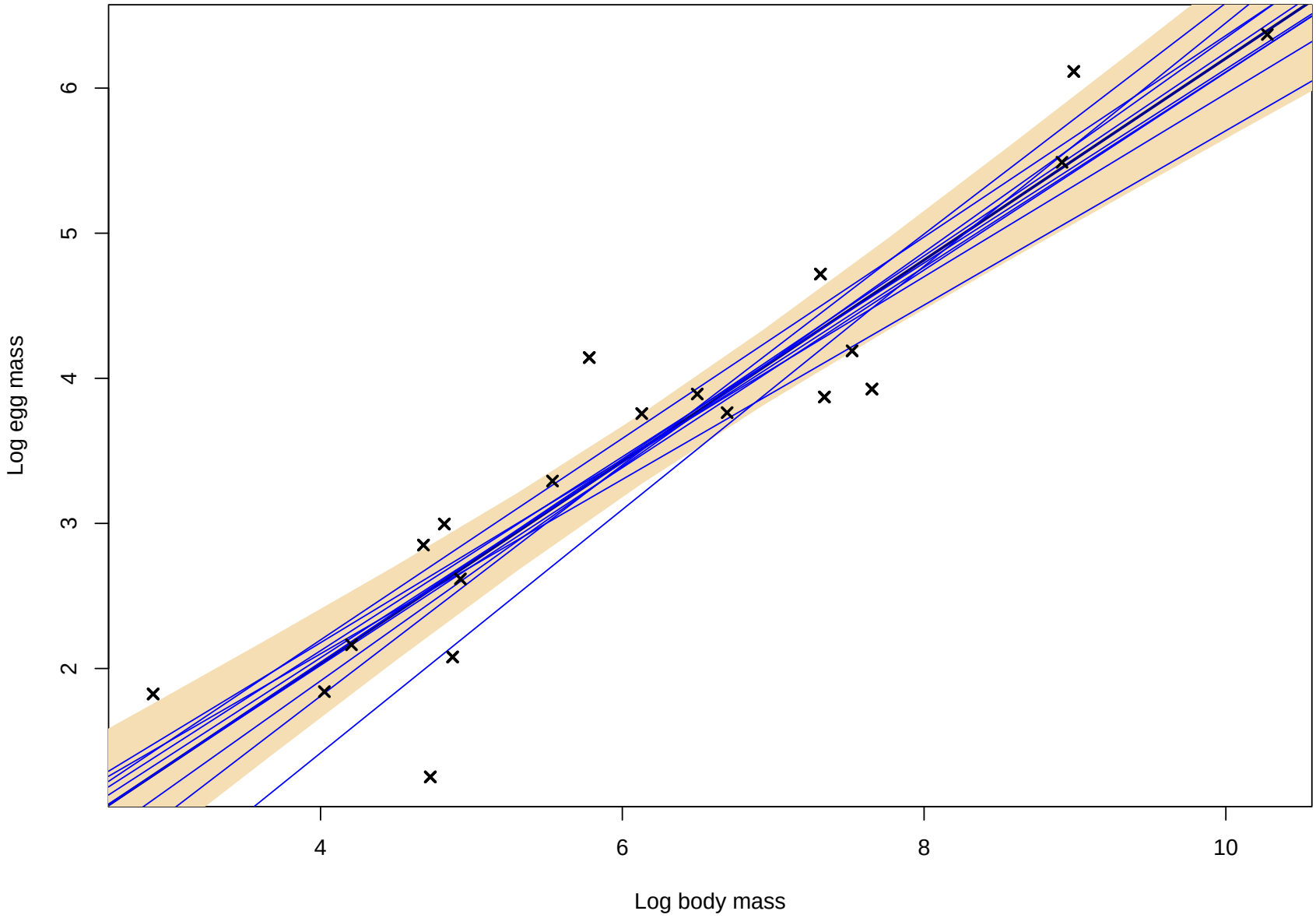
If we fix the intercept, then we can look at variation in the slope.

These are the uppr & lower 95% confidence limits



Uncertainty

In reality both slope & intercept are uncertain



Summaries

```
summary(mod)
```

```
Call:
lm(formula = logEggMass ~ logFemaleMass, data = BirdEggs)

Residuals:
     Min       1Q   Median       3Q      Max
-1.28823 -0.23295  0.01199  0.35099  0.87011

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   -0.74429    0.40126  -1.855   0.0801 .
logFemaleMass  0.69495    0.06211  11.190 1.54e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '

Residual standard error: 0.5142 on 18 degrees of freedom
```

Summaries: Coefficients

```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
 (Intercept)     -0.744      0.401    -1.9     0.08 .
 logFemaleMass    0.695      0.062    11.2    2e-09 ***
 ---
 Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Summaries: Last stuff

```
Residual standard error: 0.51 on 18 degrees of freedom
 Multiple R-squared:  0.87, Adjusted R-squared:  0.87
 F-statistic: 1.3e+02 on 1 and 18 DF,  p-value: 1.5e-09
```

Summaries: Last stuff

```
Residual standard error: 0.51 on 18 degrees of freedom
```

- Estimate of σ
- Degrees of freedom: how many "spare" data points we have

```
Multiple R-squared:  0.87,  Adjusted R-squared:  0.87
```

- Multiple R-squared: R^2 (see last lecture). What proportion of the variation are we explaining?
- Adjusted R-squared: ignore (at least for now)

```
F-statistic: 1.3e+02 on 1 and 18 DF,  p-value: 1.5e-09
```

- Test of if the data explains anything. Usually very silly to test this.

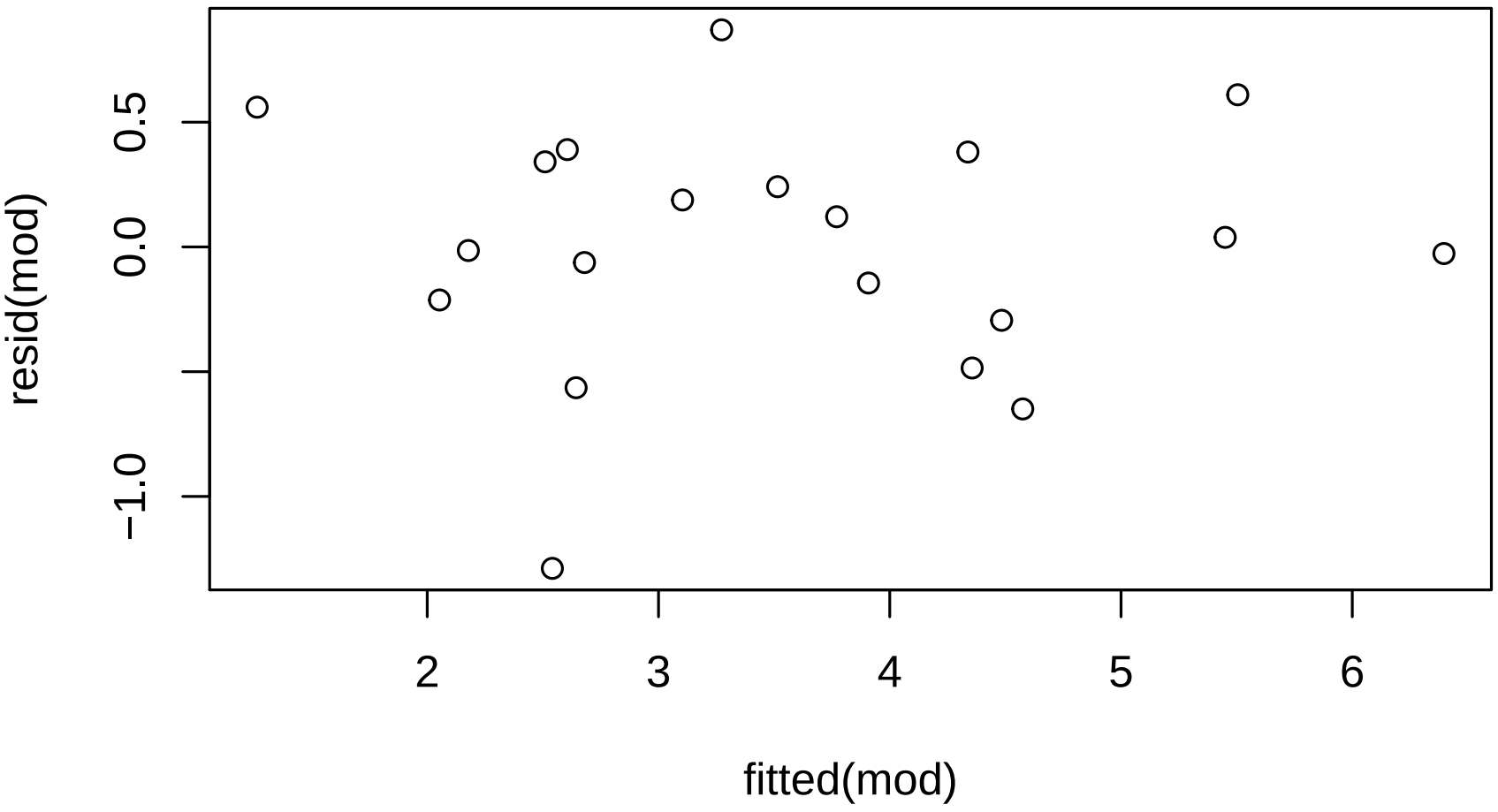
Pause

Model Checking

Lots more plots!

Building a Plot

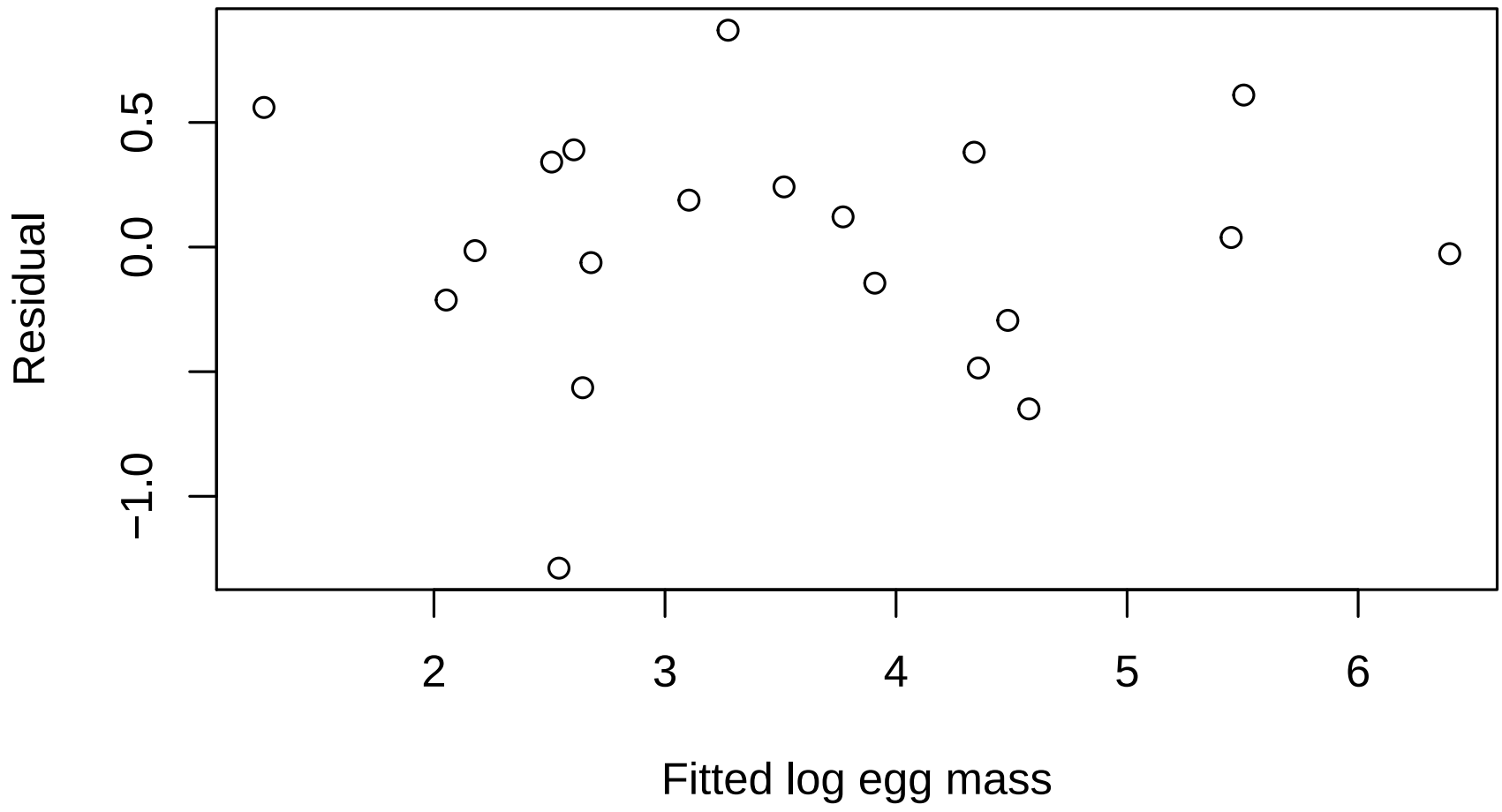
```
plot(fitted(mod), resid(mod))
```



(for your own plots, do what you feel comfortable with)

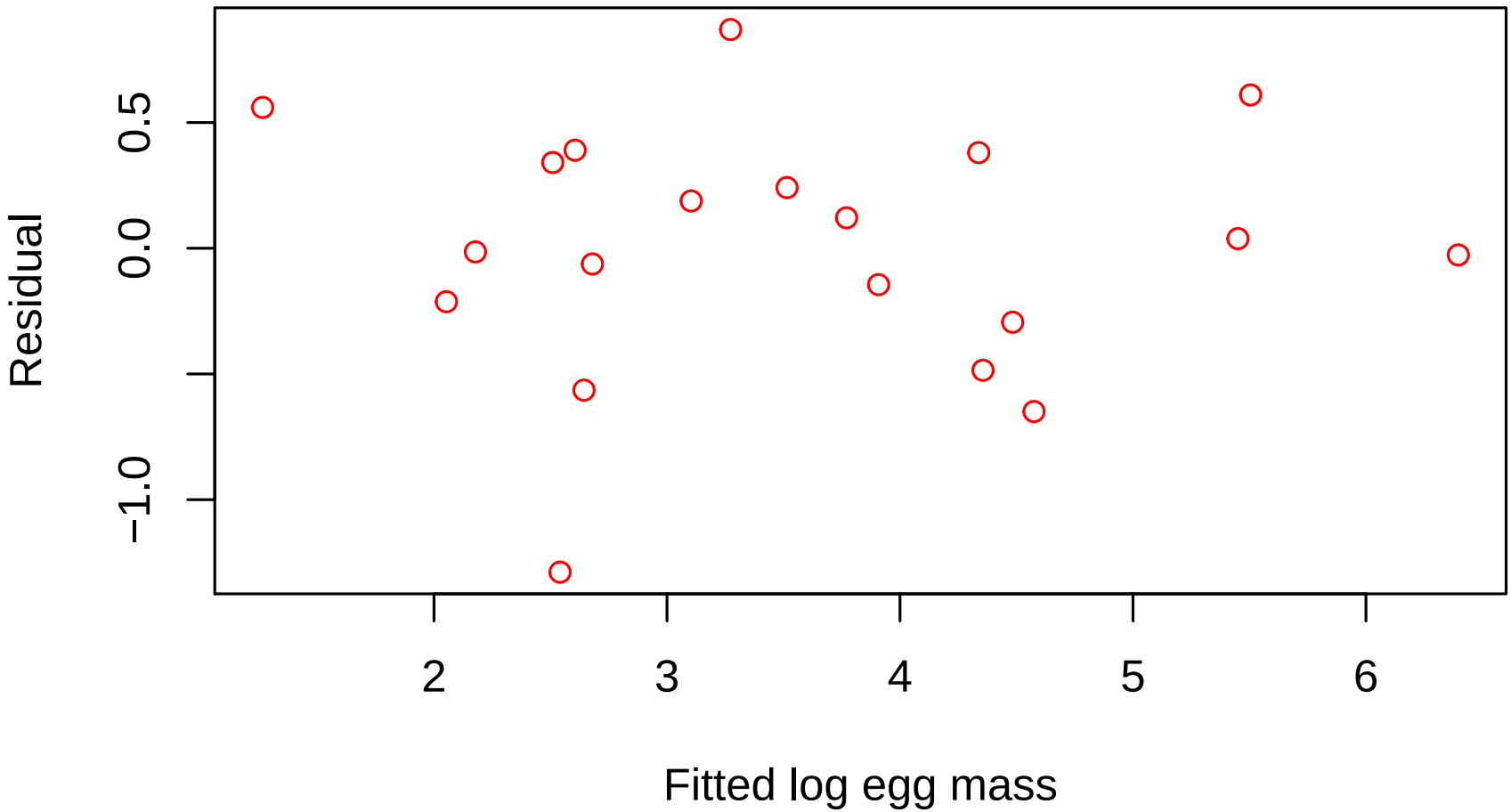
Add axis labels

```
plot(fitted(mod), resid(mod), xlab="Fitted log egg mass",
     ylab="Residual")
```



Change the plot colour

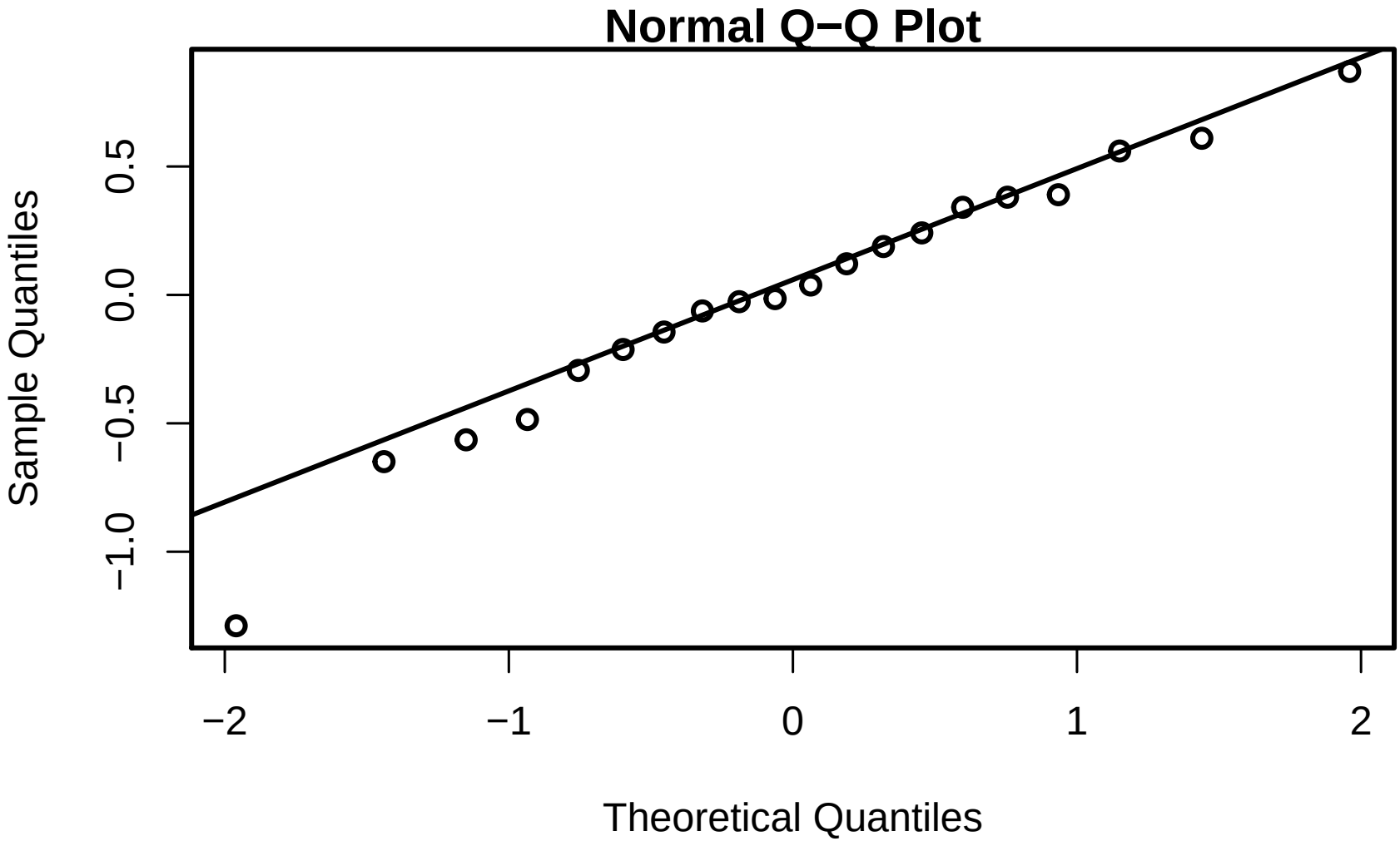
```
plot(fitted(mod), resid(mod), xlab="Fitted log egg mass",
     ylab="Residual", col="red")
```



(col= is documented in ?par, as are a lot of other options)

Quantile Plots

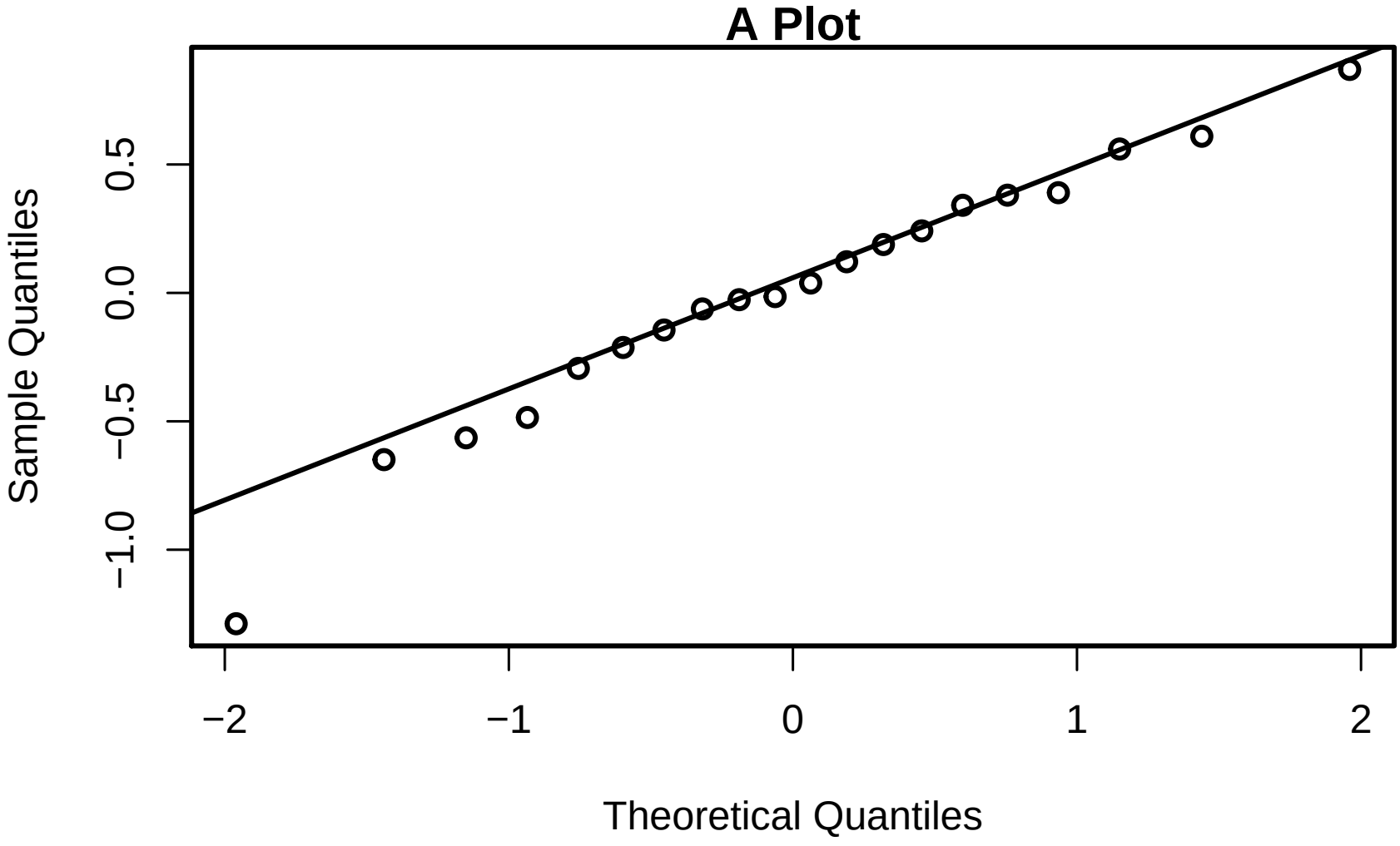
Special function!



(mar= sets margin sizes)

Quantile Plots

Special function!



(main= gives title)

Now Into More Dimensions

Multiple Regression

We have learned how to draw straight lines

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

But this is limited as a model: we cannot draw more complicated curves, and we cannot explain or predict the effects of more than one covariate.

The Model

This is out basic model

$$y_i = \alpha + \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i$$

$$y_i = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

So we replace βx_j with $\sum_{j=1}^{p} \beta_j x_{ij}$.

- we have p covariates, labelled from $j = 1$ to p
- we have p covariate effects
- the jth covariate values for the ith individual is x_{ij}

Design Matrices

We can write this more compactly. First, we turn the intercept into a covariate by using a covariate with a value of 1 for every data point. Then we write all of the covariates in a matrix, X:

$$X = \begin{pmatrix} 1 & 2.3 & 3.0 \\ 1 & 4.9 & -5.3 \\ 1 & 1.6 & -0.7 \\ \vdots & \vdots & \vdots \\ 1 & 8.4 & 1.2 \end{pmatrix}$$

So, the first column is the intercept, the second is the first covariate, and the third is the second covariate.

This is called the *Design Matrix*: it is helpful for writing down the model

Writing the Model

Using marix algebra, the regression model becomes

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

where $\mathbf{Y}$, β and ε are now all vectors of length n, where there are n data points. X is am $n \times p$ matrix.

We will not look at the mathematics in any detail: the point here is that the model for the effect of covariates can be written in the design matrix.

Writing the Model

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

is

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & 2.3 & 3.0 \\ 1 & 4.9 & -5.3 \\ 1 & 1.6 & -0.7 \\ \vdots & \vdots & \vdots \\ 1 & 8.4 & 1.2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_n \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

The Solution (just so you can see it)

After a bit of matric algebra, one can find the ML solution:

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

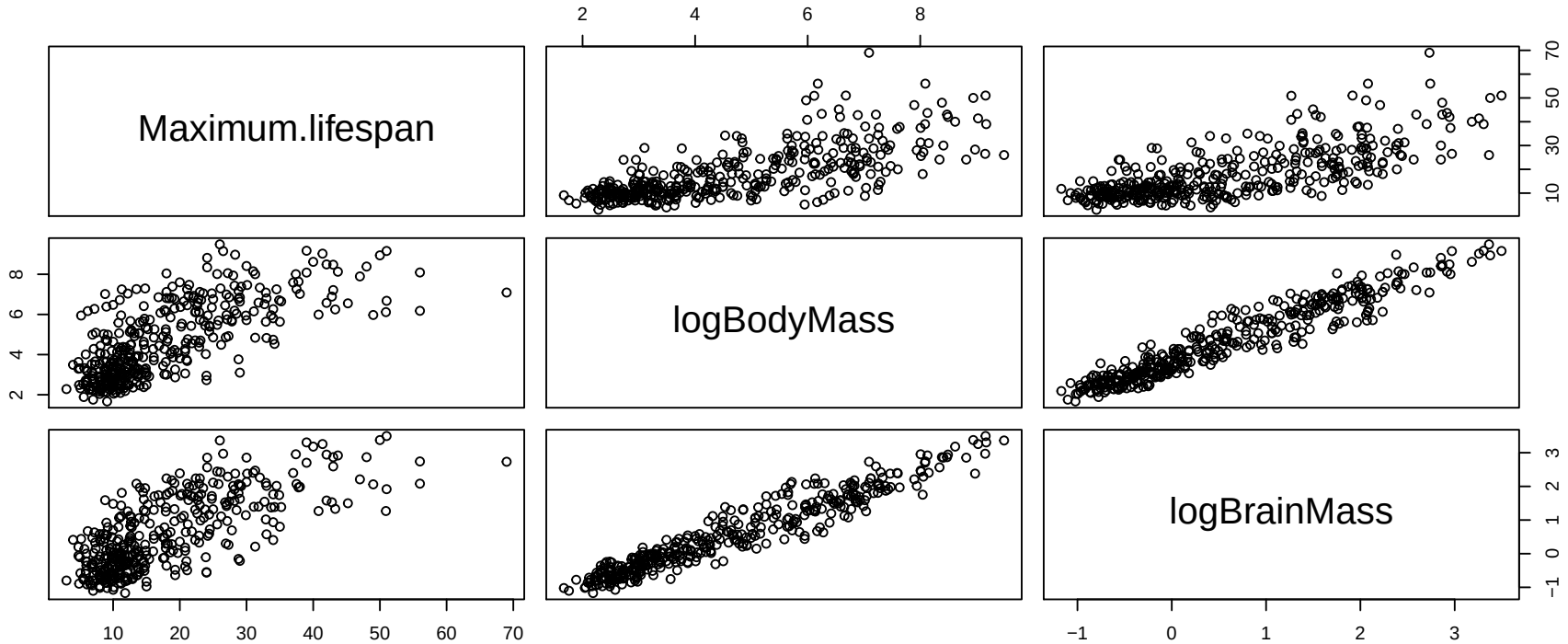
where $\mathbf{b}$ is the MLE for β.

In practice, (a) you won't have to calculate this: the computer does it, and (b) the computer actually doesn't use this

An Example: Bird Brains

The data wehave been using was collected to look at the effects of longevity on brain size.

But size is a counfounder, i.e. it also has an effect



Fitting the Model

We wil ltry to explain (log) brain mass with:

- Maximum lifespan
- Age at first reprodction
- logBodyMass

We can write the model as

logBrainMass $\sim$ Maximum lifespan $+$ Age at first reprodction $+$ logBodyMass

Fitting the Model

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.025      0.042 -47.942        0
## Maximum.lifespan     0.009      0.002   4.368        0
## logBodyMass          0.525      0.012  43.395        0
```

The Model

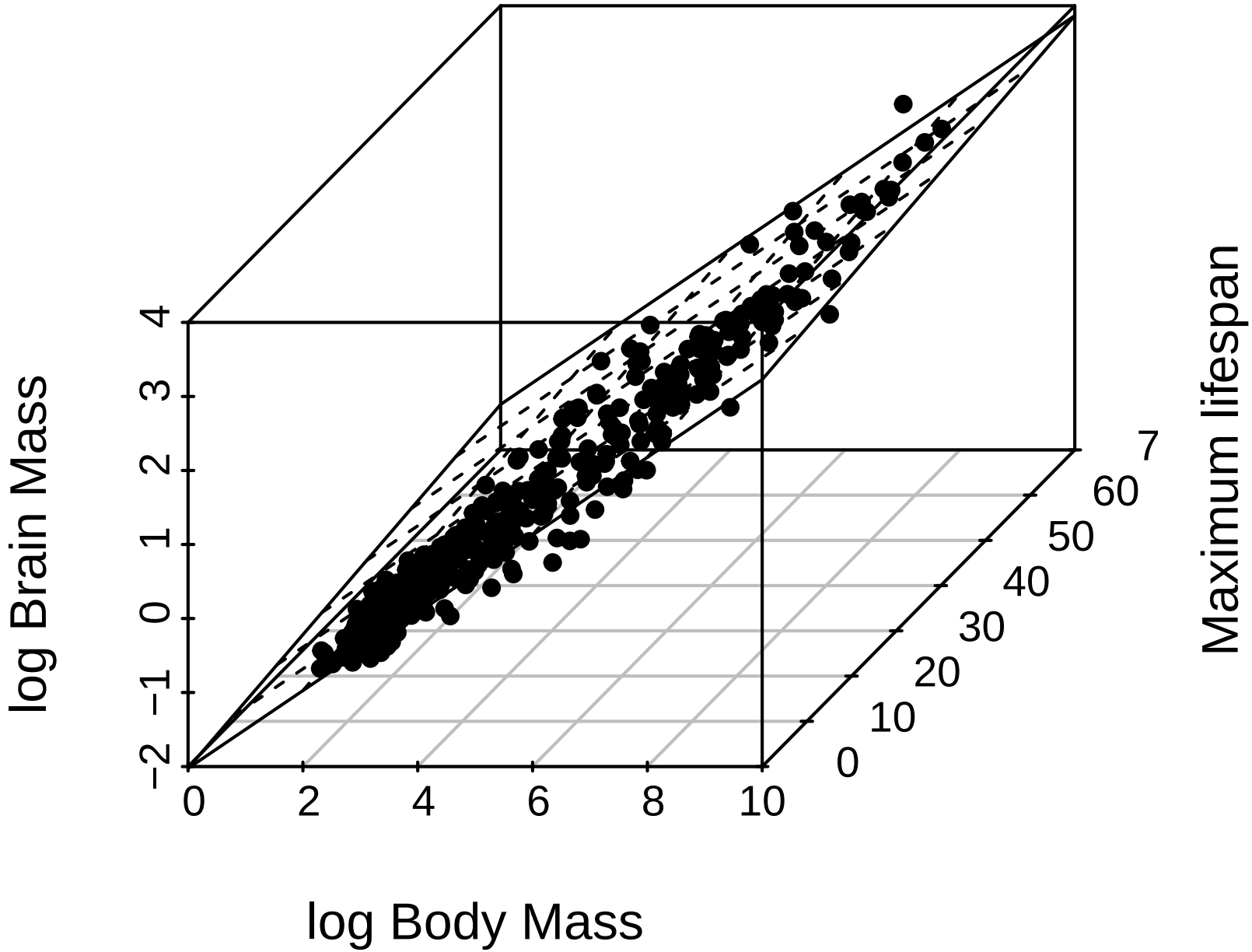
The model is

$$y_i = -2.02 + 0.01x_{i1} + 0.01x_{i2} + \varepsilon_i$$

Where the X's are maximum lifespan and logBodyMass

The Model

With one covariate we have a straight line, with two we have a plane.



Model Interpretation

The parameters say what happens if we increase a parameter by 1 if we hold the other covariates constant

e.g. if we increase maximum lifespan by 1 year, log brain mass increases by 0.009.

But this is scale dependent: if we measure lifespan in decades, the coefficient changes to 0.091

Makes it difficult to compare between different covariates: how do we compare 1 year to 1kg?

Standardised Coefficients

We can look at the standardised coefficients

- standardise by the standard deviation
- also mean-centre

Fit the Standardised Model

```
BirdBrains$Max.lifespan.std <-
  scale(BirdBrains$Maximum.lifespan)
BirdBrains$logBodyMass.std <-
  scale(BirdBrains$logBodyMass)
Mod.std <- lm(logBrainMass ~ logBodyMass.std +
                Max.lifespan.std, data=BirdBrains)
round(summary(Mod.std)$coefficients, 3)
```

```
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.571      0.016  36.134        0
## logBodyMass.std     0.987      0.023  43.395        0
## Max.lifespan.std    0.099      0.023   4.368        0
```

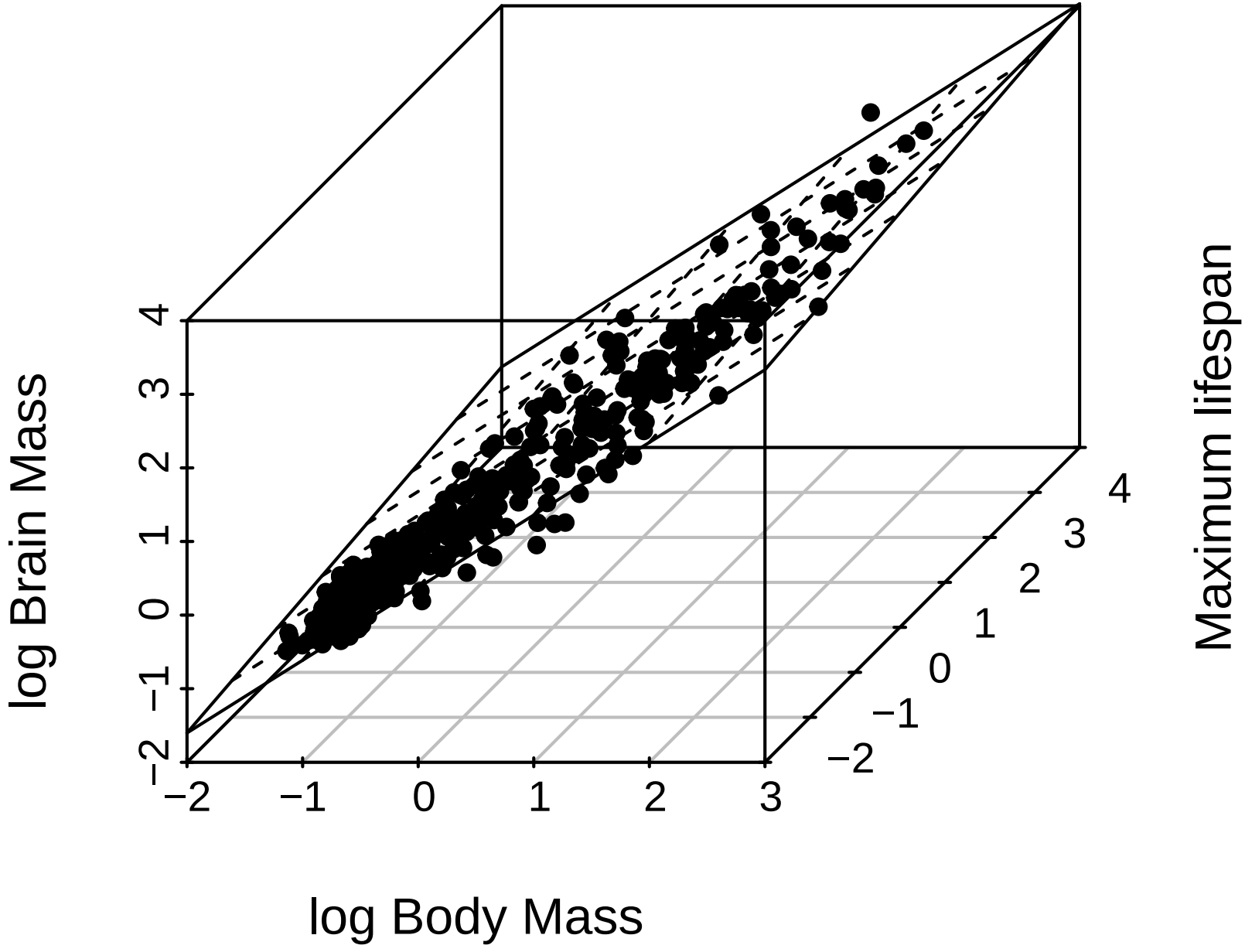
Interpret the Standardised Model

```
round(summary(Mod.std)$coefficients, 3)
```

```
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.571      0.016  36.134        0
## logBodyMass.std     0.987      0.023  43.395        0
## Max.lifespan.std    0.099      0.023   4.368        0
```

- The intercept is now at the mean of Max. lifespan & log Body Mass
- the estimates are the effects of changing covriates by 1 standard deviation.
  - comparable as relative changes in the data
  - effect of log body mass about 10 times bigger than lifespan

Plot the Standardised Model



Another use of multiple regression

Approximating curves

We can approximate curves with a Taylor series:

$$f(x) \approx \beta_0 + \beta_1(x - \bar{x}) + \beta_2(x - \bar{x})^2 + \beta_3(x - \bar{x})^3 + \cdots + \beta_p(x - \bar{x})^p$$

So we can fit an approximate curve by regressing Y against X, X^2, x^3 etc.

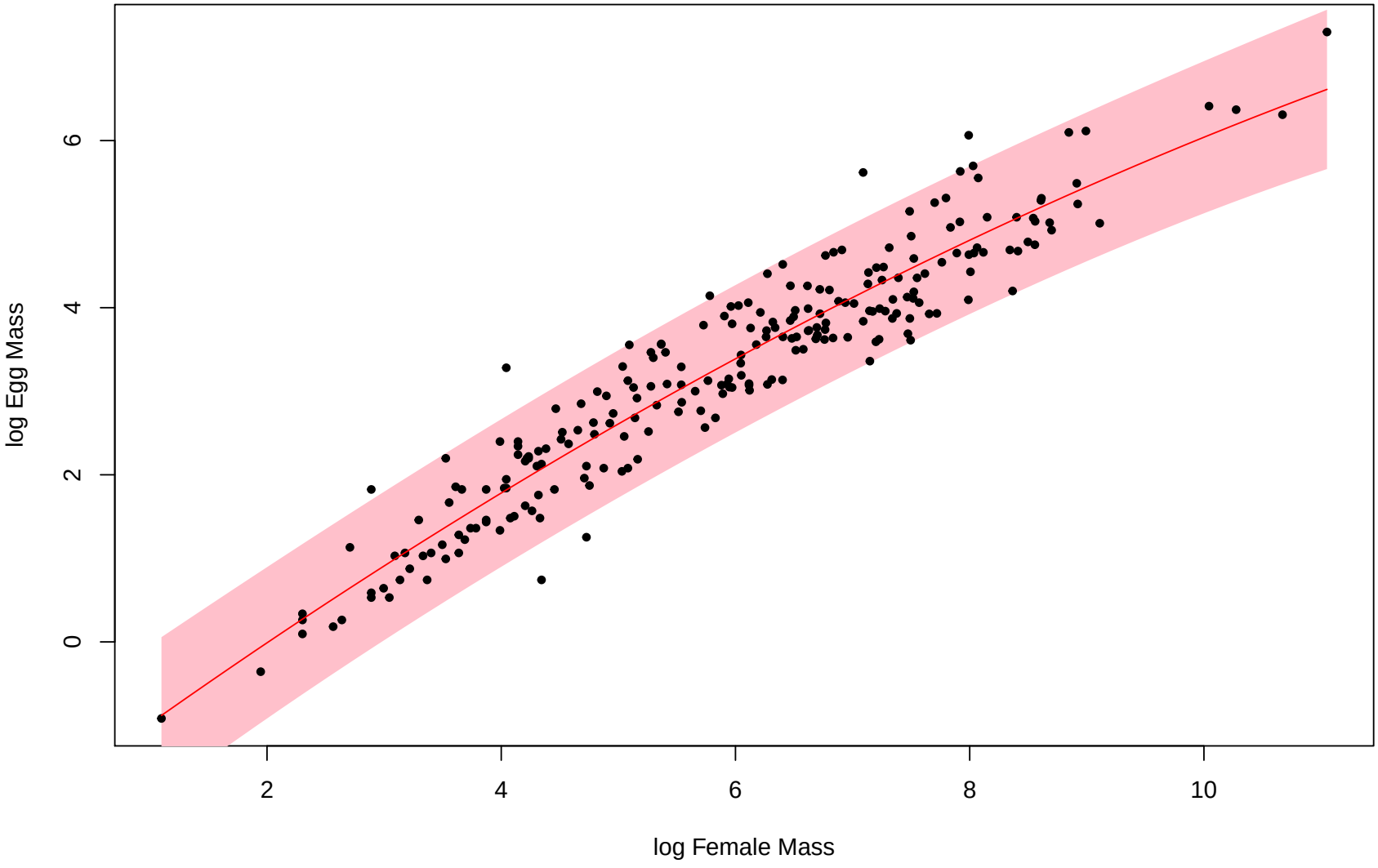
Bird Eggs

For the bird eggs data, there might be a curve, so we can fit that

```
BirdEggs <- read.csv(file="../Data/BirdEggs.csv",
                     stringsAsFactors = FALSE)
BirdEggs$lgFM.std <- scale(BirdEggs$logFemaleMass)
Mod.quad <- lm(logEggMass ~ lgFM.std + I(lgFM.std^2),
               data=BirdEggs)
round(summary(Mod.quad)$coefficients, 2)
```

```
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)         3.32       0.04   89.78        0
## lgFM.std            1.37       0.03   47.16        0
## I(lgFM.std^2)      -0.08       0.02   -3.28        0
```

Plot Bird Eggs



Summary

We can fit models with more than 1 covariate

- comparison of the coefficients is a bit tricky
- with more than 2 covariates, plotting the model is a pain

We can use this to fit more complicated curves

Next Week

How well does the model fit?

Do we need all of these parameters?