Lecture 7: Multiple Regression

Bob O'Hara

bob.ohara@ntnu.no

Before we start. . .

Exercises to be handed in by 17:00

- if we get a folder set up

Multiple Regression

We have learned how to draw straight lines

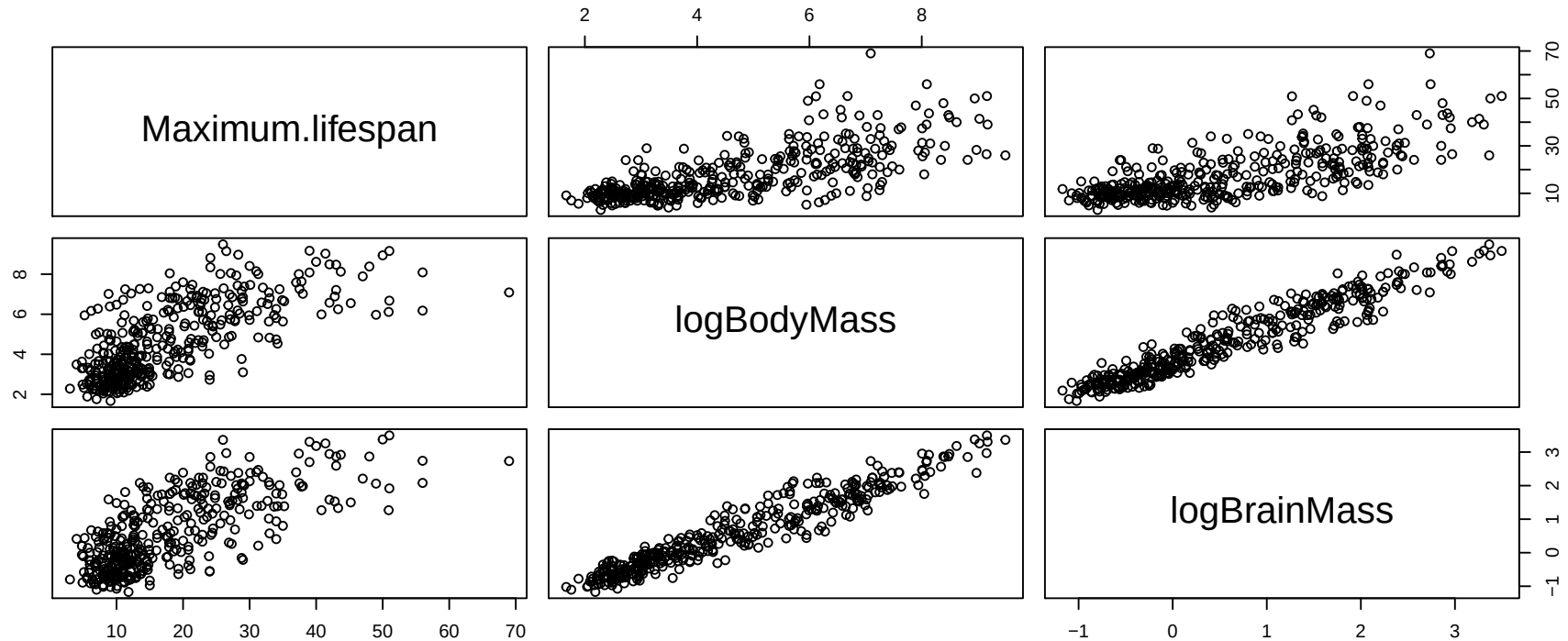
$$y_i = \alpha + \beta x_i + \varepsilon_i$$

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.

An Example: Bird Brains

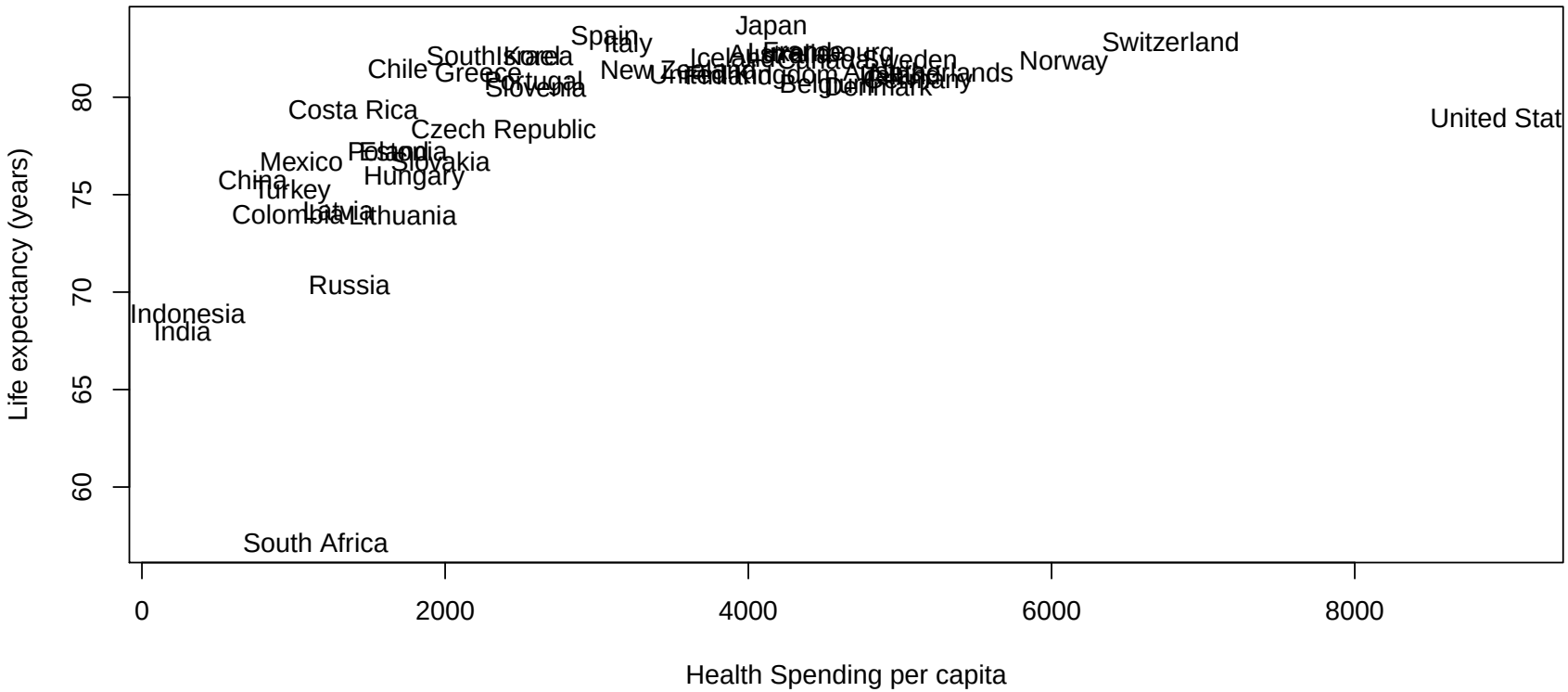
The data wehave been using was collected to look at the effects of longevity on brain size.

- but size is a counfounder, i.e. it also has an effect, so it had to be included



Another Example: Health care

The health care data isn't linear. We can try transforming, or we can fit a polynomial (i.e. include x^2, x^3 etc. terms)



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

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & 2.3 & 3.0 \\ 1 & 4.9 & -5.3 \\ 1 & 1.6 & -0.7 \\ \vdots & \vdots & \vdots \\ 1 & 8.4 & 1.2 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

- β_0 is the intercept

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:

- you won't have to calculate this: the computer does it, and
- the computer actually doesn't use this

Fitting the Model

We will try to explain (log) brain mass with:

- Maximum lifespan
- logBodyMass

We can write the model as

$$\text{logBrainMass} \sim \text{Maximum lifespan} + \text{logBodyMass}$$

Fitting the Model

Parameter Estimates

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

```
##                  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

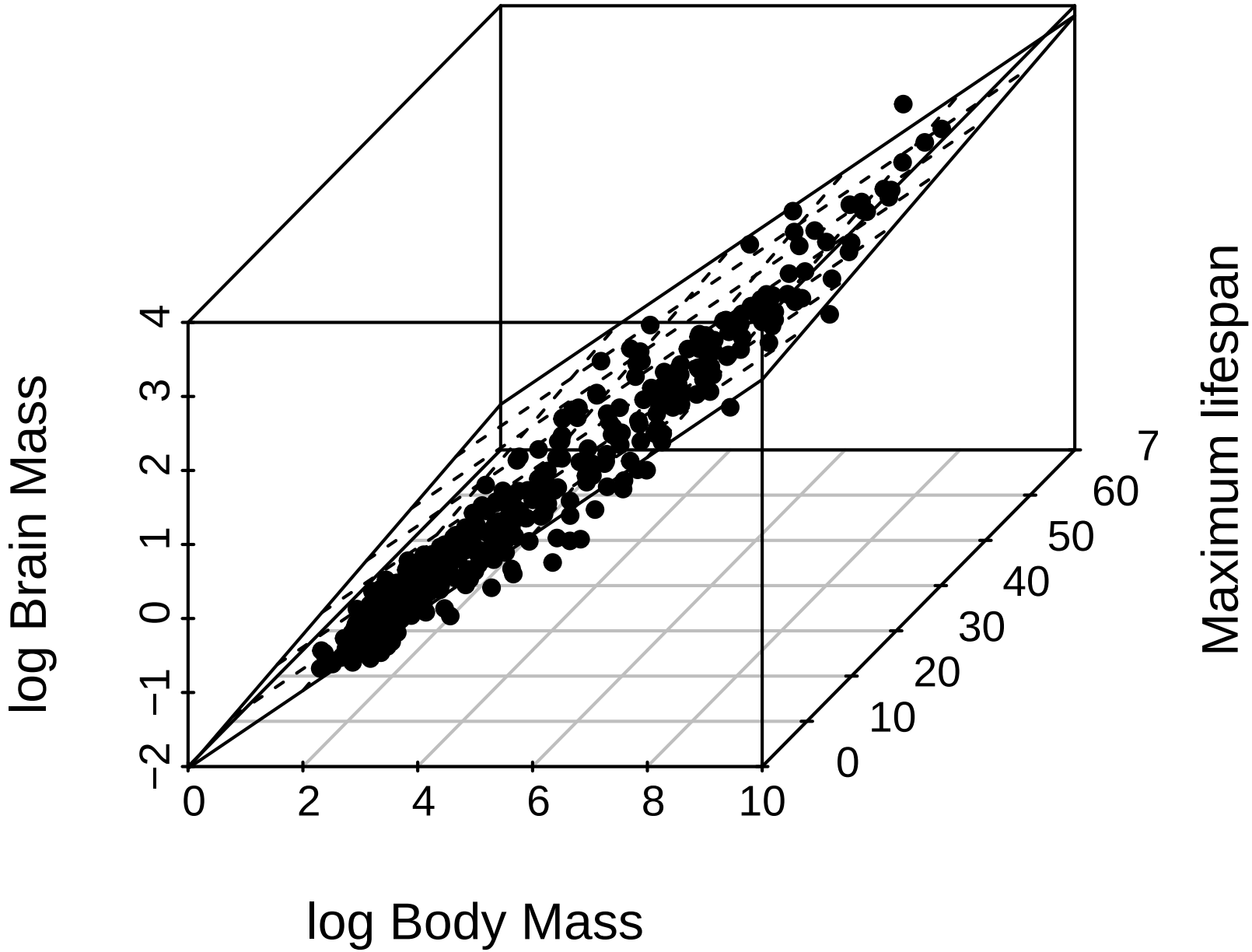
$$y_i = -2.02 + 0.01x_{i1} + 0.53x_{i2} + \varepsilon_i$$

Where the X's are maximum lifespan and logBodyMass

```
##                   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

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

```
BirdBrains$Max.lifespan.std <-
  scale(BirdBrains$Maximum.lifespan)
BirdBrains$logBodyMass.std <-
  scale(BirdBrains$logBodyMass)
```

Fit the Standardised Model

```
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
```

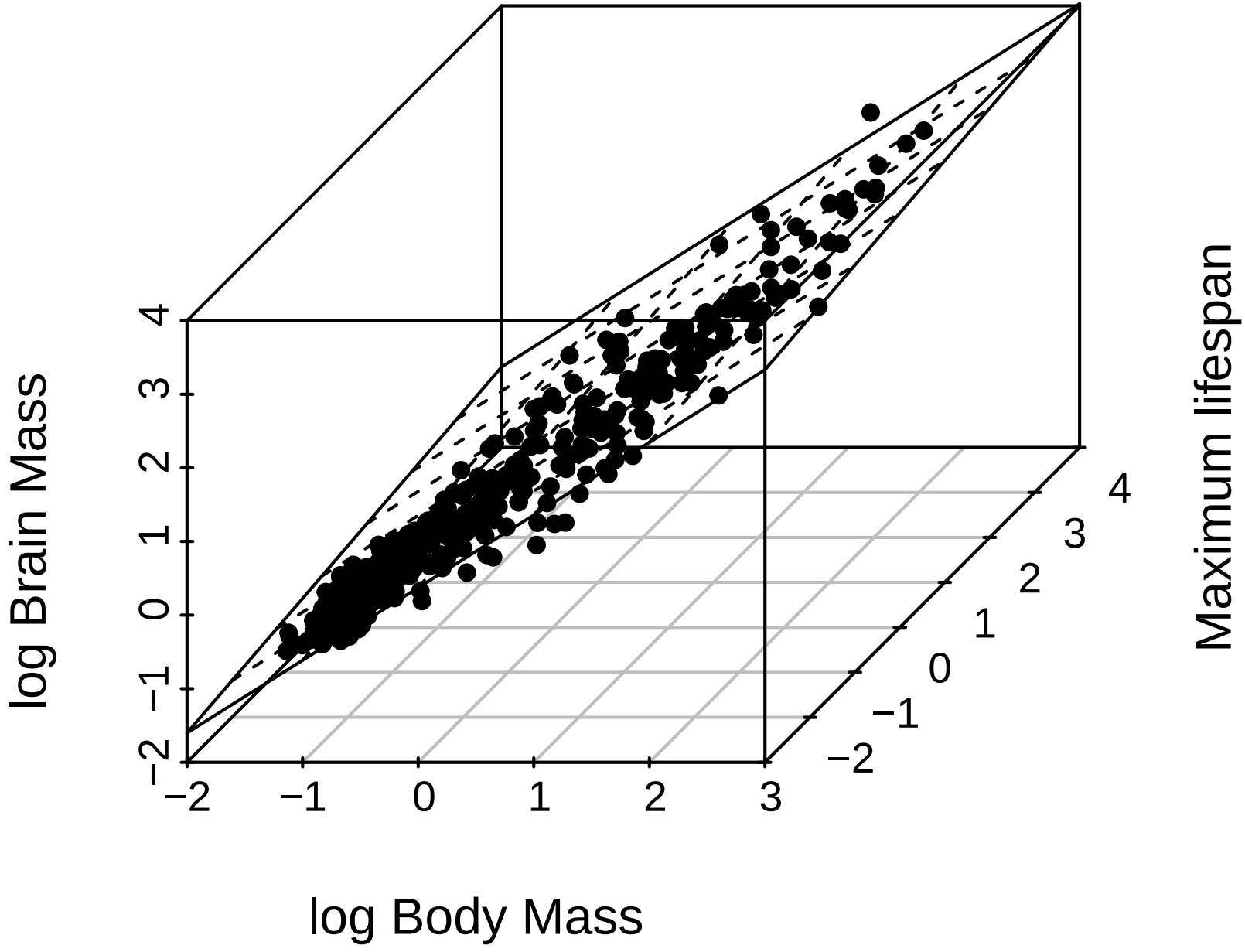
Interpret the Standardised Model

```
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 covariates 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$lgFM.std <- scale(BirdEggs$logFemaleMass) # stand
Mod.quad <- lm(logEggMass ~ lgFM.std + I(lgFM.std^2),
               data=BirdEggs)
```

The model formula is

logEggMass $\sim$ lgFM.std $+$ I(lgFM.std^2)

We need the I() to tell R to use the quadratic

Bird Eggs Summary

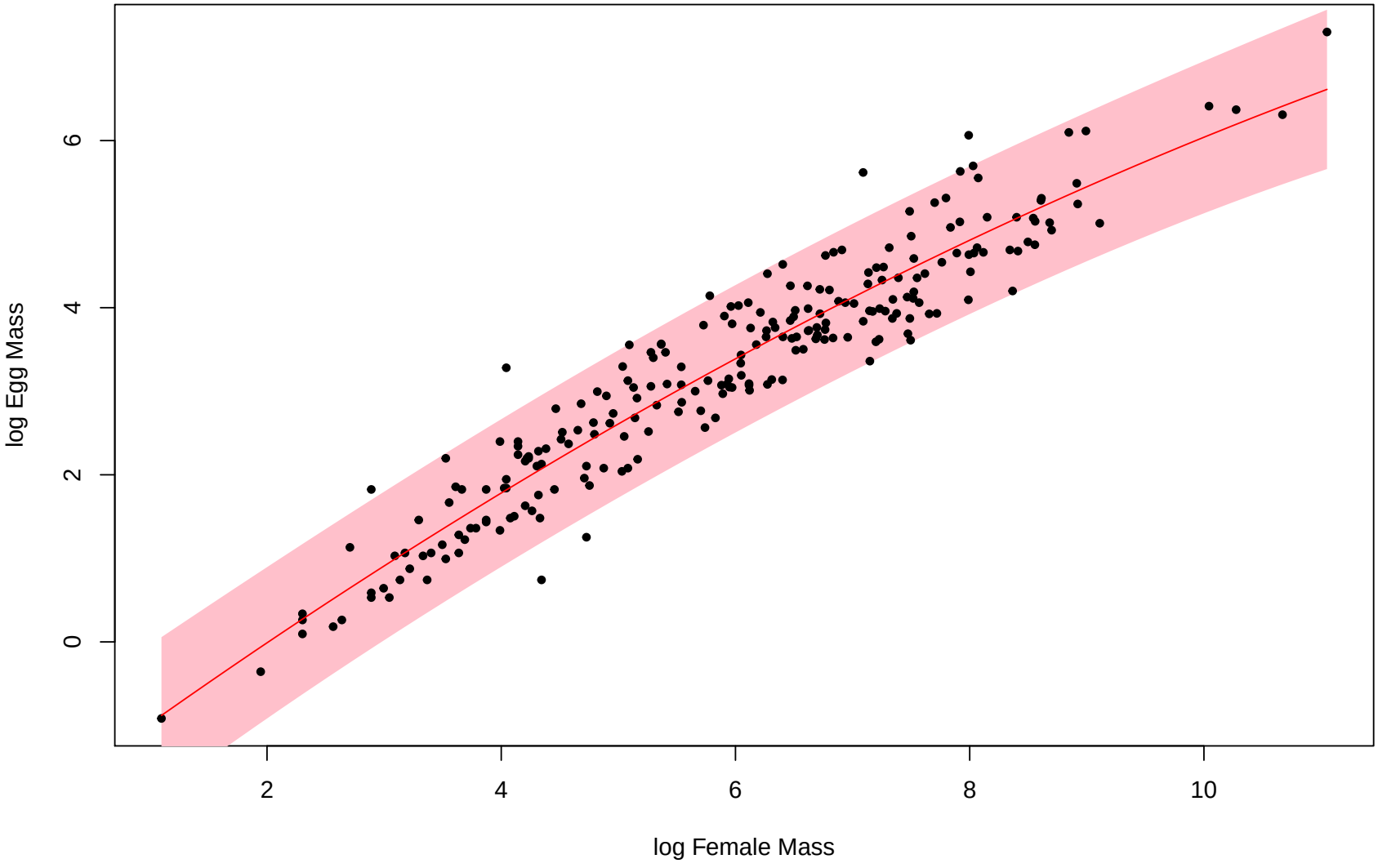
```
round(summary(Mod.quad)$coefficients, 2)
```

```
##                 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
```

We still see a positive linear term, but the quadratic is negative

- so, what does the curve look like?

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

Categorical Variables

Regression & R

Read in the data (and sub-sample)

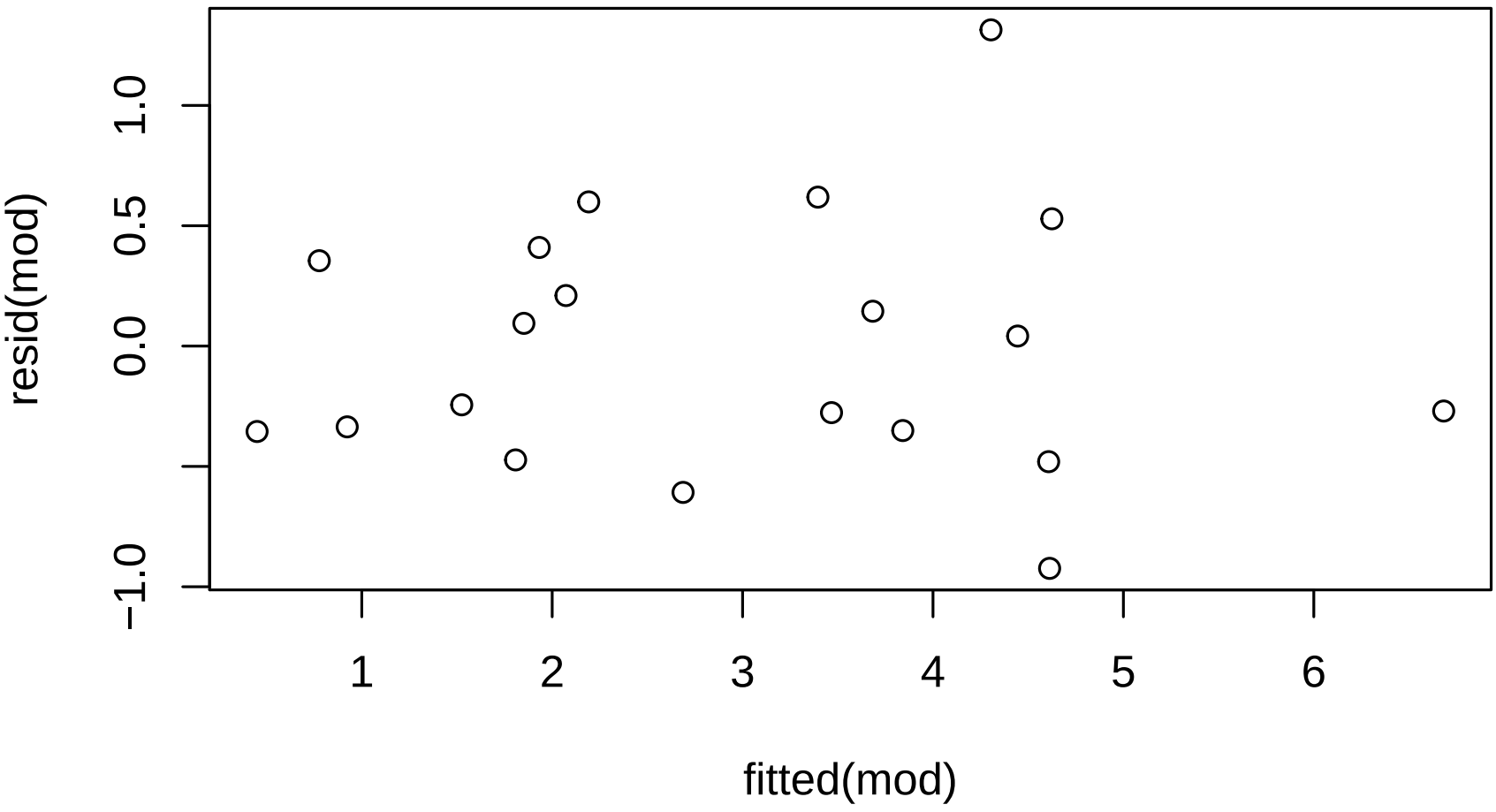
```
BirdEggs <- read.csv(file="../Data/BirdEggs.csv",
                     stringsAsFactors = FALSE)
# sub-sample 30 observations, just to make things clearer
BirdEggs <- BirdEggs[sample.int(nrow(BirdEggs), size=20),]
mod <- lm(logEggMass ~ logFemaleMass, data=BirdEggs)
```

Model Checking

Lots more plots!

Building a Plot

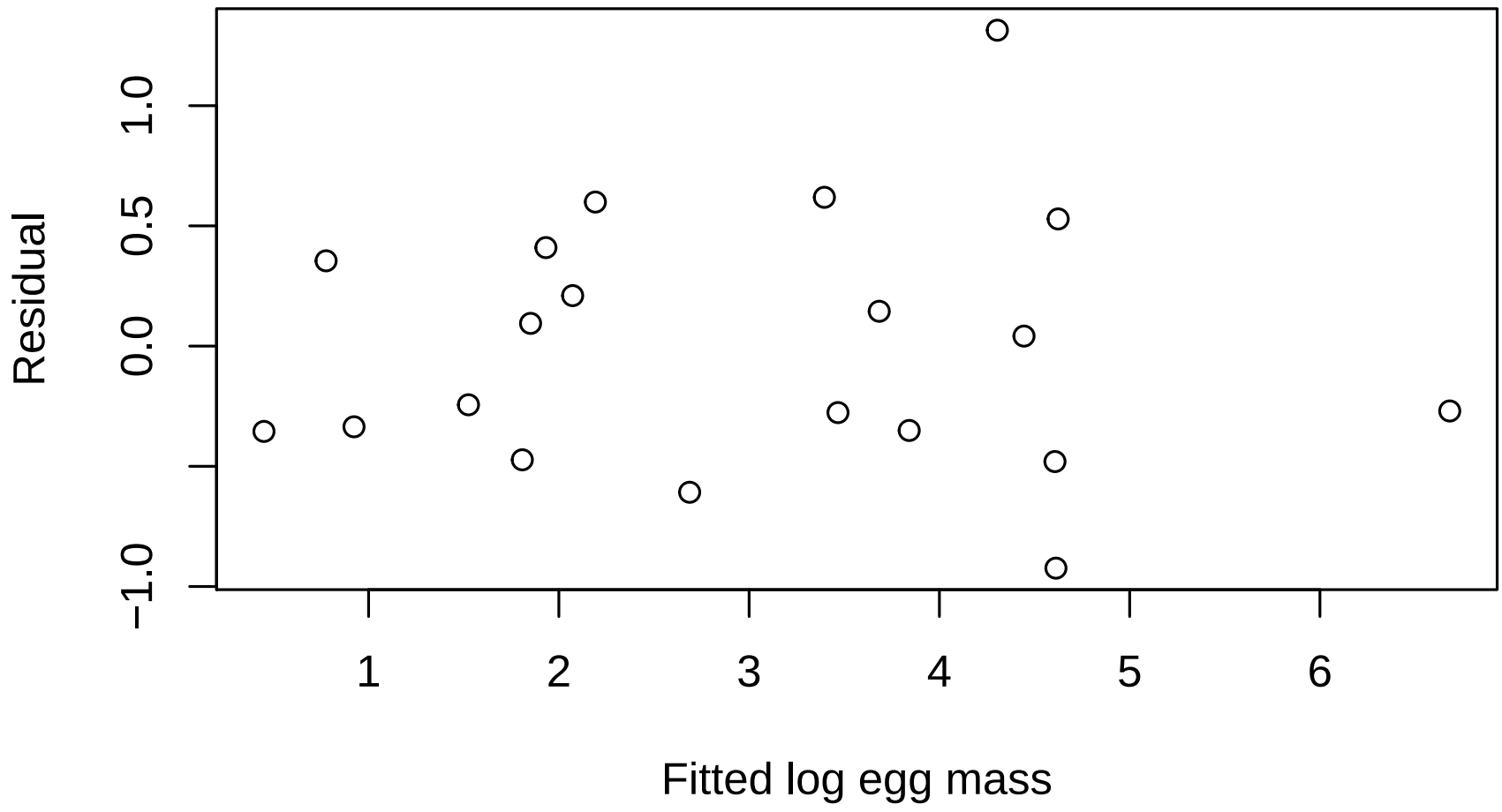
```
plot(fitted(mod), resid(mod))
```



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

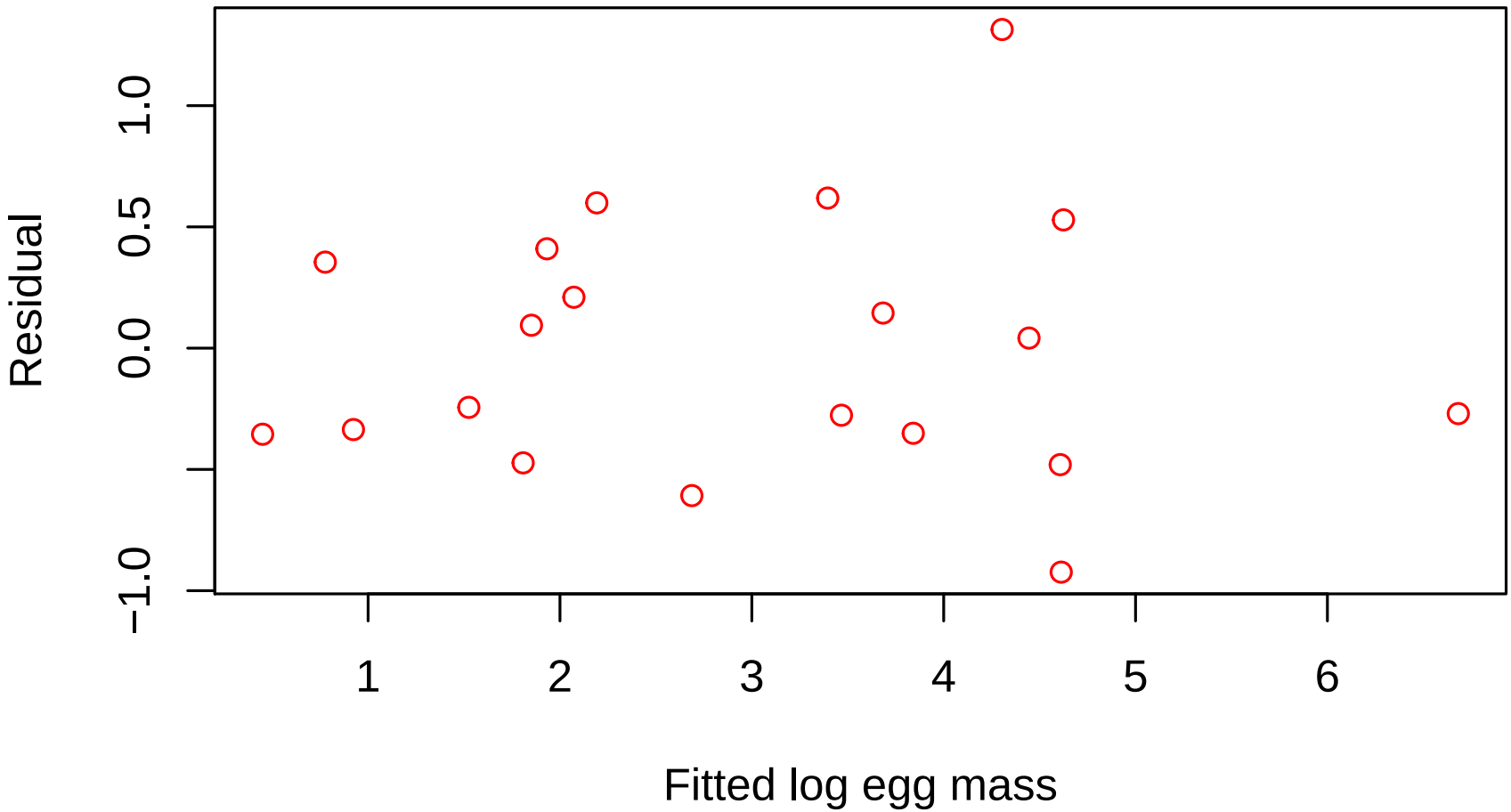
Add axis labels

```
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)

Add a Title

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