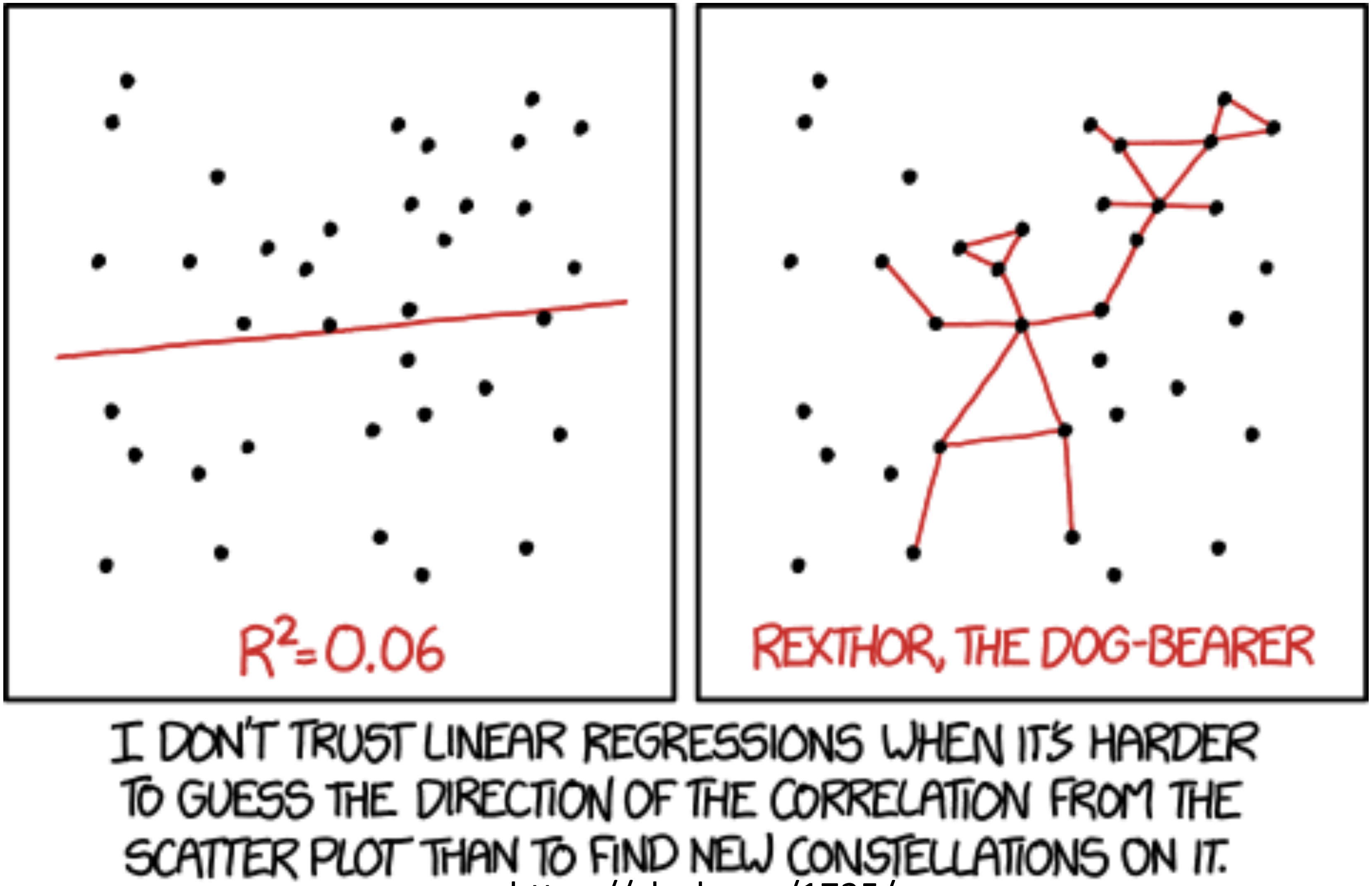
Peer-Feedback Survey
1 survey every Tuesday (constant link)

https://ubc.ca1.qualtrics.com/jfe/form/SV_bvLa6xMXdRk1j6u

Optional: download R script to follow along chickadee example on Canvas under lecture slides
R Script_Lecture 04_Linear Models_chickadee example of linear model.R

BIOL 501: Linear Models


R^2=0.06
REXTHOR, THE DOG-BEARER
I DON'T TRUST LINEAR REGRESSIONS WHEN IT'S HARDER TO GUESS THE DIRECTION OF THE CORRELATION FROM THE SCATTER PLOT THAN TO FIND NEW CONSTELLATIONS ON IT.

https://xkcd.com/1725/

Get me out, or make a new one

31 Jan. 2023

Dr. Beth Volpov, b.volpov@oceans.ubc.ca; UBC 2022 Term 2

Outline for today

- What is a linear model
- Example fitting and comparing a model
- Model comparison: full vs reduced
- Assessing model fits and assumptions

#1.scatter plot (examine data)

```
plot(y ~ x, data = mydata)
```

#2. Fit linear model

```
model1<- lm(y ~ x, data=mydata)
```

#3. Extract coefficients and information from the model

summary(model1) and model1$coefficients

#4.Add model line to scatter plot above

abline() or lines() or ggplot()
Plot CI with visreg()
predict()

#5. Test model fit with anova (test hypothesis)

```
anova(model1)
```

#6. Model comparison between a full and reduced model

```
anova(null,model1)
```

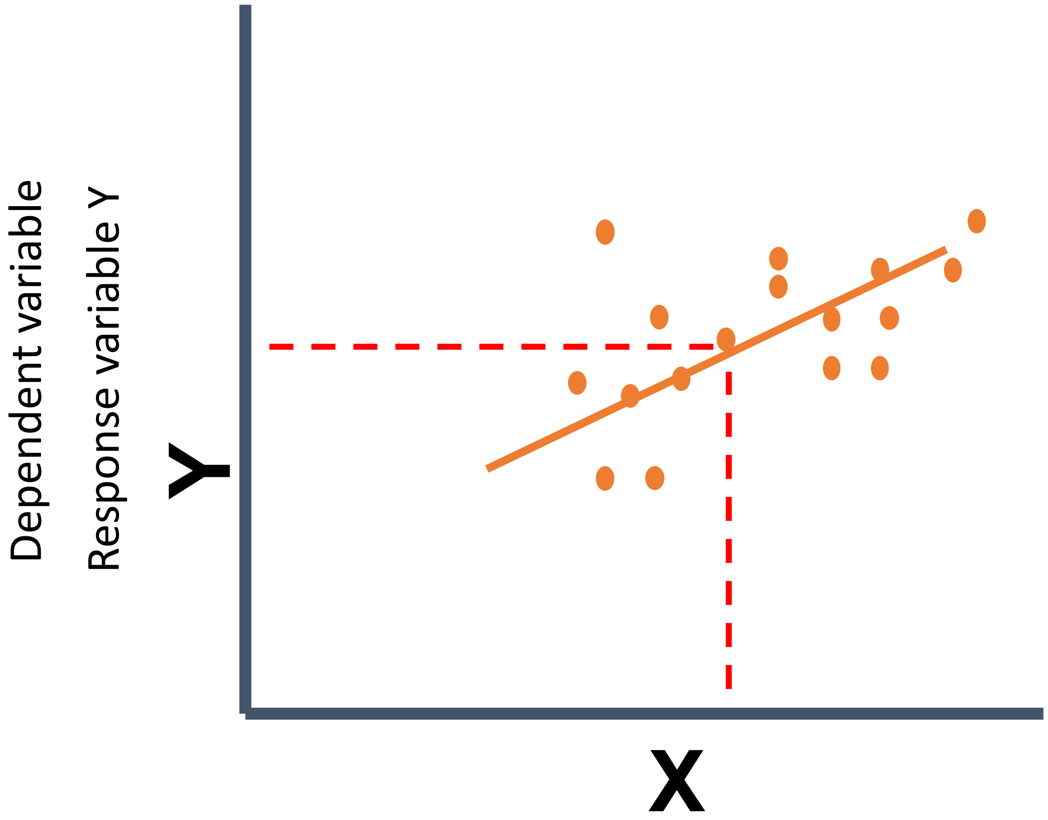
#7. Look at model assumptions on the best-fit model (diagnostics)

```
plot(model1)
```

#8. Predict() new data from model line (in workshop)

What is a linear model

Y= m*x+b
Y=slope* X+ Y-intercept

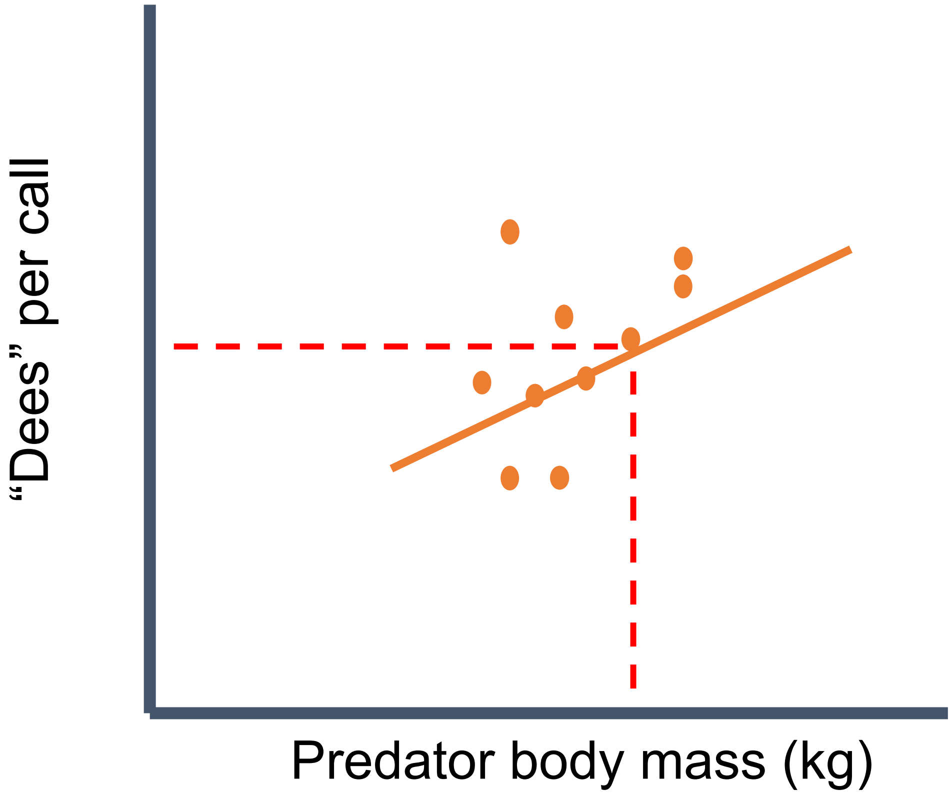


Explanatory variables (X_1, X_2, …..)

Independent variable

Normal random errors with equal variance $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots$ + error; where β_0, β_1, β_2, … are the *parameters* of the linear model

Example of linear model (both numerical X,Y)



Data: The average number of "dee" notes per alarm call by black-capped chickadees presented with a live, perched predator.

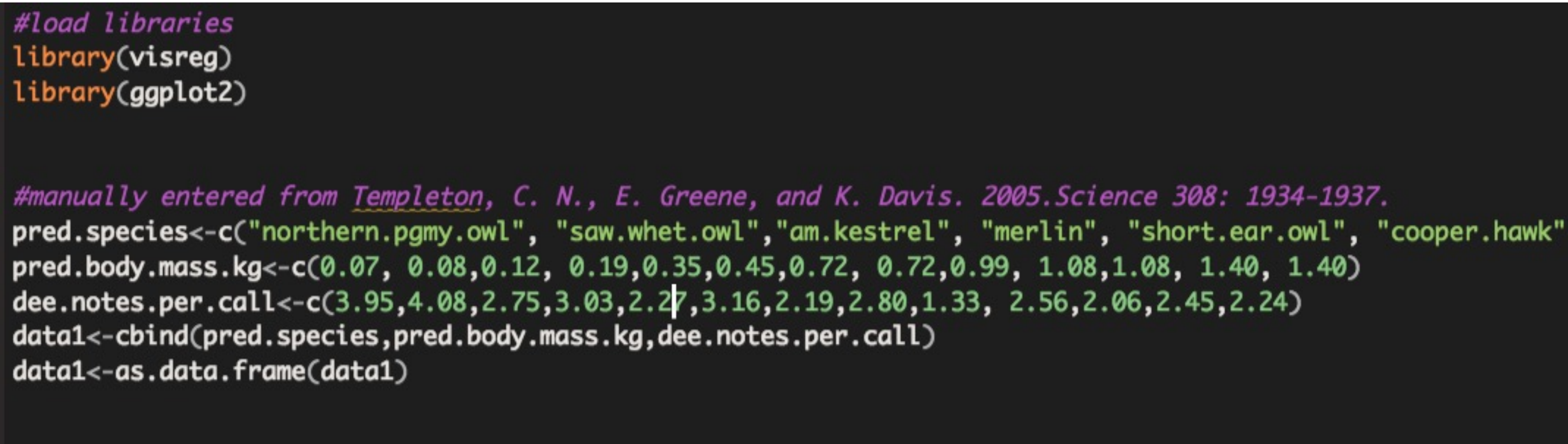
R code example

```
model1<-lm(y variable ~x variable,data=mydata)
```

Predator species	Predator body mass (kg)	“dee” notes per call
Northern pygmy-owl	0.07	3.95
Saw-whet owl	0.08	4.08
American kestrel	0.12	2.75
Merlin	0.19	3.03
Short-eared owl	0.35	2.27
Cooper’s hawk	0.45	3.16
Prairie falcon	0.72	2.19
Peregrine falcon	0.72	2.80
Rough-legged hawk	0.99	1.33
Red-tailed hawk	1.08	2.56
Great gray owl	1.08	2.06
Great horned owl	1.40	2.45
Gyrfalcon	1.40	2.24

Published data in Table

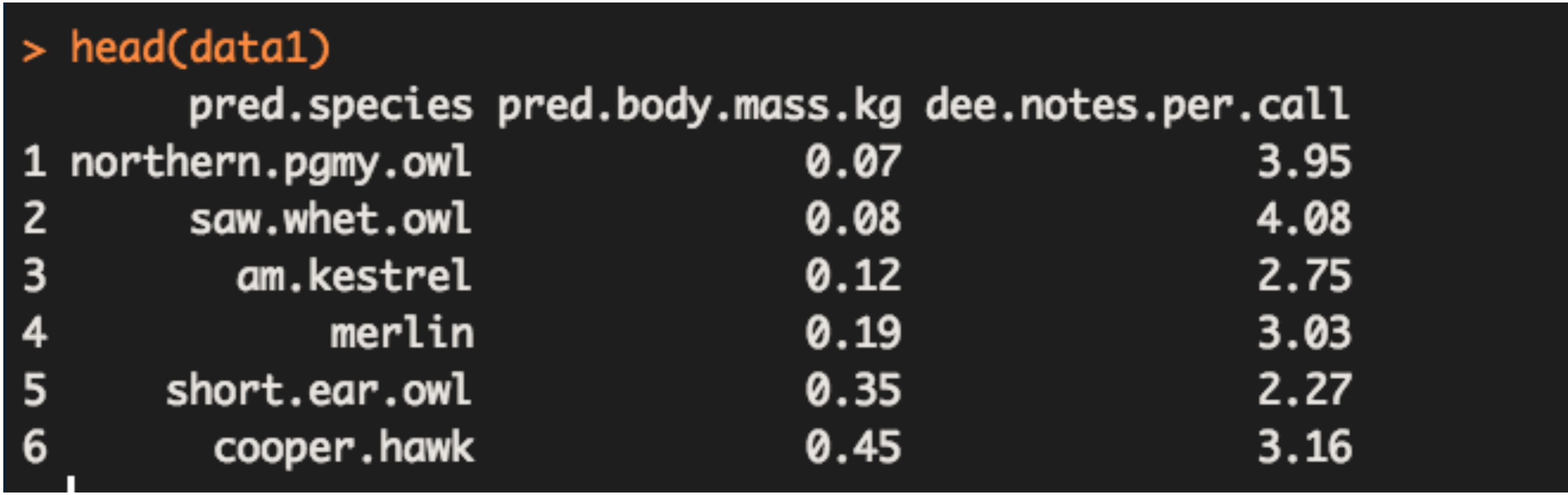
Manually entered in R and cbind()



```r
#load libraries
library(visreg)
library(ggplot2)


#manually entered from Templeton, C. N., E. Greene, and K. Davis. 2005.Science 308: 1934-1937.
pred.species<-c("northern.pgmy.owl", "saw.whet.owl","am.kestrel", "merlin", "short.ear.owl", "cooper.hawk"
pred.body.mass.kg<-c(0.07, 0.08,0.12, 0.19,0.35,0.45,0.72, 0.72,0.99, 1.08,1.08, 1.40, 1.40)
dee.notes.per.call<-c(3.95,4.08,2.75,3.03,2.27,3.16,2.19,2.80,1.33, 2.56,2.06,2.45,2.24)
data1<-cbind(pred.species,pred.body.mass.kg,dee.notes.per.call)
data1<-as.data.frame(data1)
```

Follow-along or go back and try the code later



```
> head(data1)
       pred.species pred.body.mass.kg dee.notes.per.call
1 northern.pgmy.owl              0.07               3.95
2      saw.whet.owl              0.08               4.08
3        am.kestrel              0.12               2.75
4            merlin              0.19               3.03
5     short.ear.owl              0.35               2.27
6       cooper.hawk              0.45               3.16
```

#1.scatter plot (examine data)

```
plot(y ~ x, data = mydata)
```

#2. Fit linear model

```
model1<- lm(y ~ x, data=mydata)
```

#3. Extract coefficients and information from the model

```
summary(model1) and model1$coefficients
```

#4.Add model line to scatter plot above

```
abline() or lines() or ggplot()
Plot CI with visreg()
predict()
```

#5. Test model fit with anova (test hypothesis)

```
anova(model1)
```

#6. Model comparison between a full and reduced model

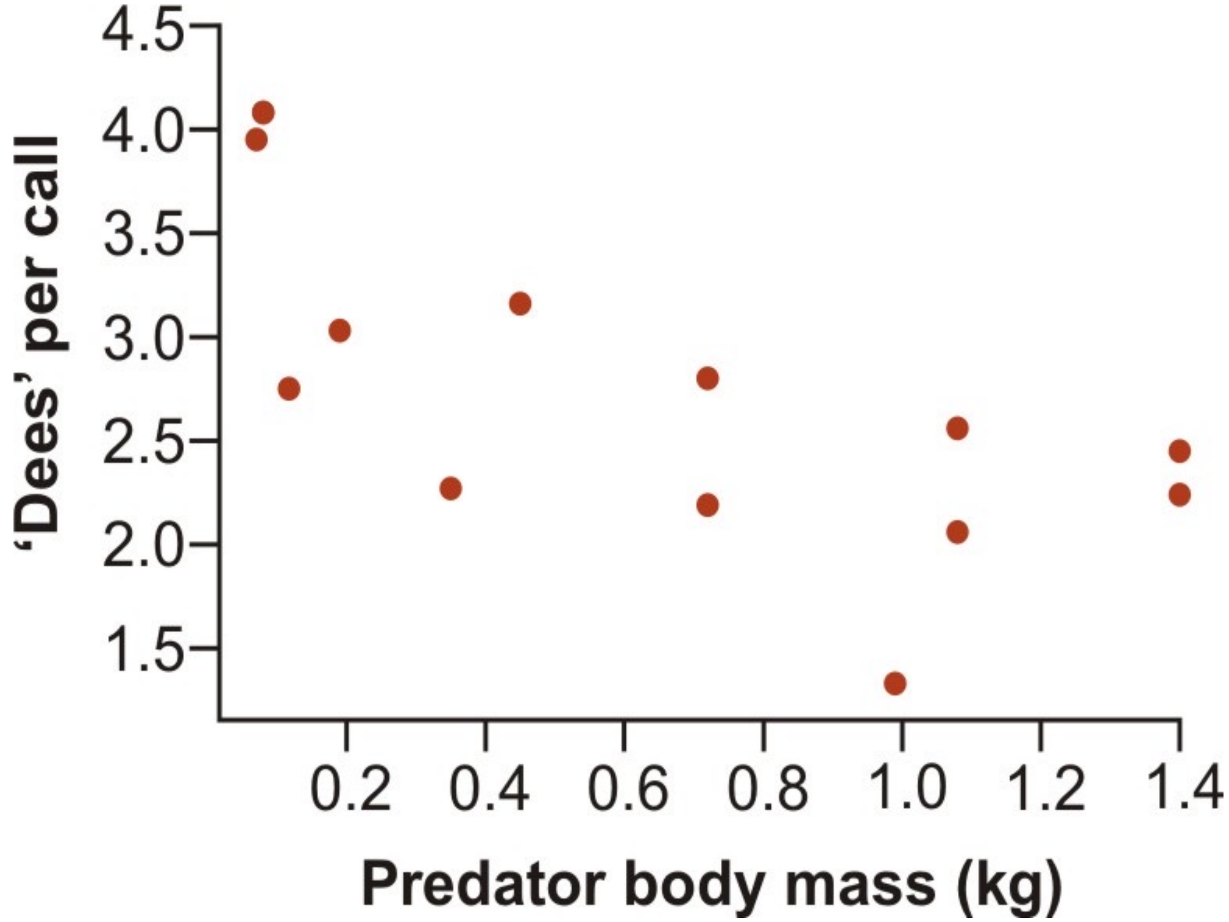
```
anova(null,model1)
```

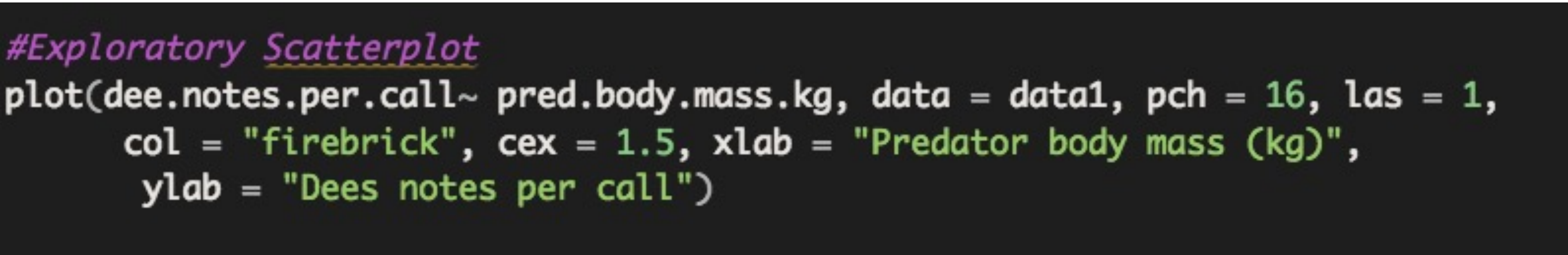
#7. Look at model assumptions on the best-fit model (diagnostics)

```
plot(model1)
```

#8. Predict() new data from model line (in workshop)

Plot and Examine Data



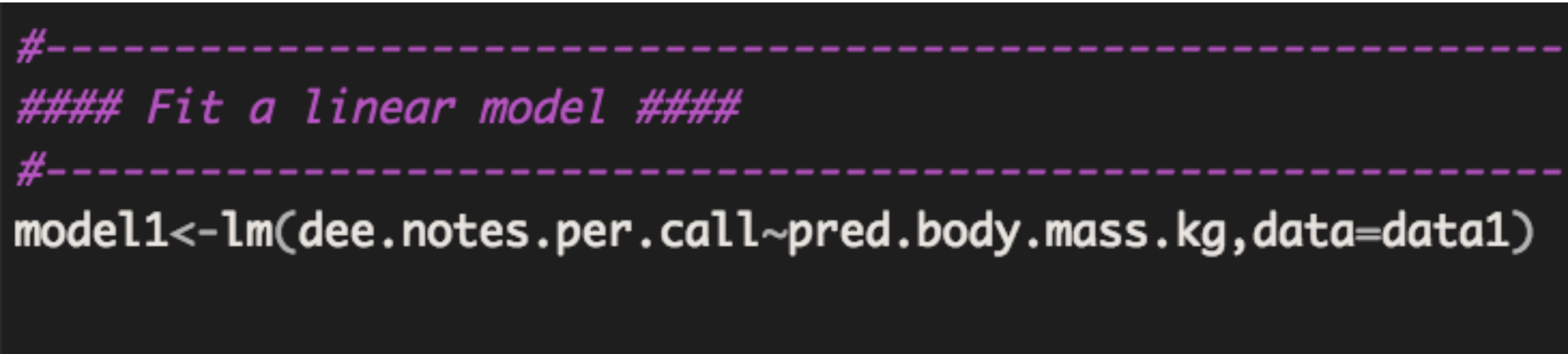


```
#Exploratory Scatterplot
plot(dee.notes.per.call~ pred.body.mass.kg, data = data1, pch = 16, las = 1,
      col = "firebrick", cex = 1.5, xlab = "Predator body mass (kg)",
       ylab = "Dees notes per call")
```

Fit a linear model with lm()

- In R the y-intercept is implicit and doesn’t need to be in the model formula
- We are modelling **only fixed** factors with lm()

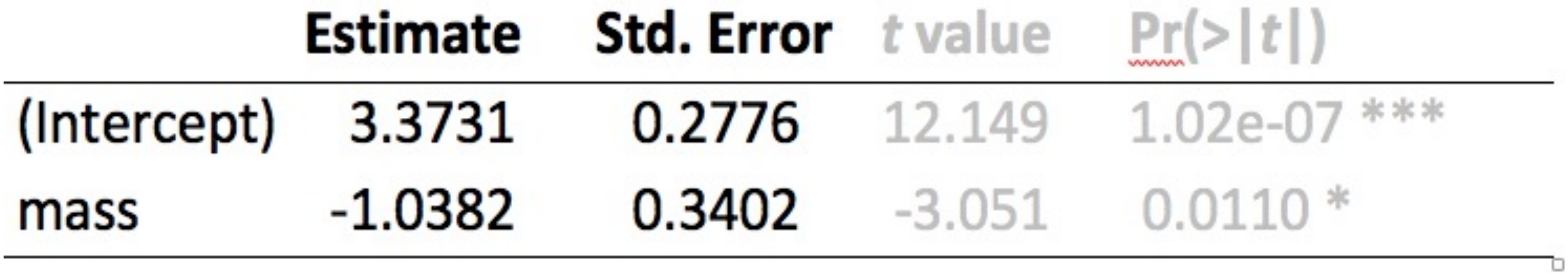
model1<- lm(y ~ x, data=mydata)



```
#-----------------------------------------------------------
#### Fit a linear model ####
#-----------------------------------------------------------
model1<-lm(dee.notes.per.call~pred.body.mass.kg,data=data1)
```

Use summary() to get parameter estimates

- Summary() produces a huge table that includes coefficients table, standard error, R2 and more

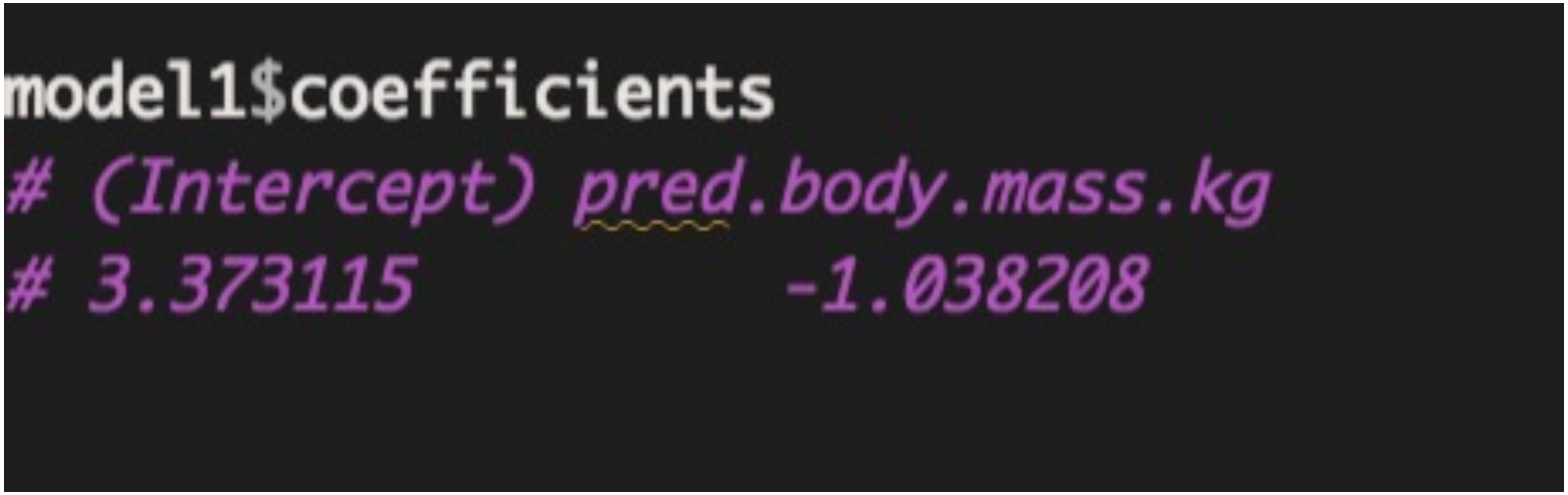


	Estimate	Std. Error	*t* value	Pr(>|*t*|)
(Intercept)	3.3731	0.2776	12.149	1.02e-07 ***
mass	-1.0382	0.3402	-3.051	0.0110 *

But ignore the tests for significance
(we will use anova later)

Faster way to extract only the coefficient values

- Coefficients in linear regression are the slope and intercept



```
model1$coefficients
# (Intercept) pred.body.mass.kg
# 3.373115          -1.038208
```

Plot the fitted model line over the scatter plot of data points

- abline(model1)
- lines()
- Ggplot→geom_points and geom_smooth
- Plot CI and model line with visreg()

You have lots of options in how to plot model lines

Add model line to plot in Base R with lines()

- Remember linear equation is y=m*x+b
- y=slope*X values+Y intercept

#extract coefficients from model summary with use indexing

model1$coefficients

```
#(Intercept)          pred.body.mass.kg
 # 3.373115             -1.038208
```

Slope

Intercept

[1]   [2]

Add model line to plot in Base R with lines()

y=m*x+b

#extract coefficients from fitted model

model1$coefficients

```
#(Intercept)              pred.body.mass.kg (Slope)
#  3.373115                    -1.038208
```

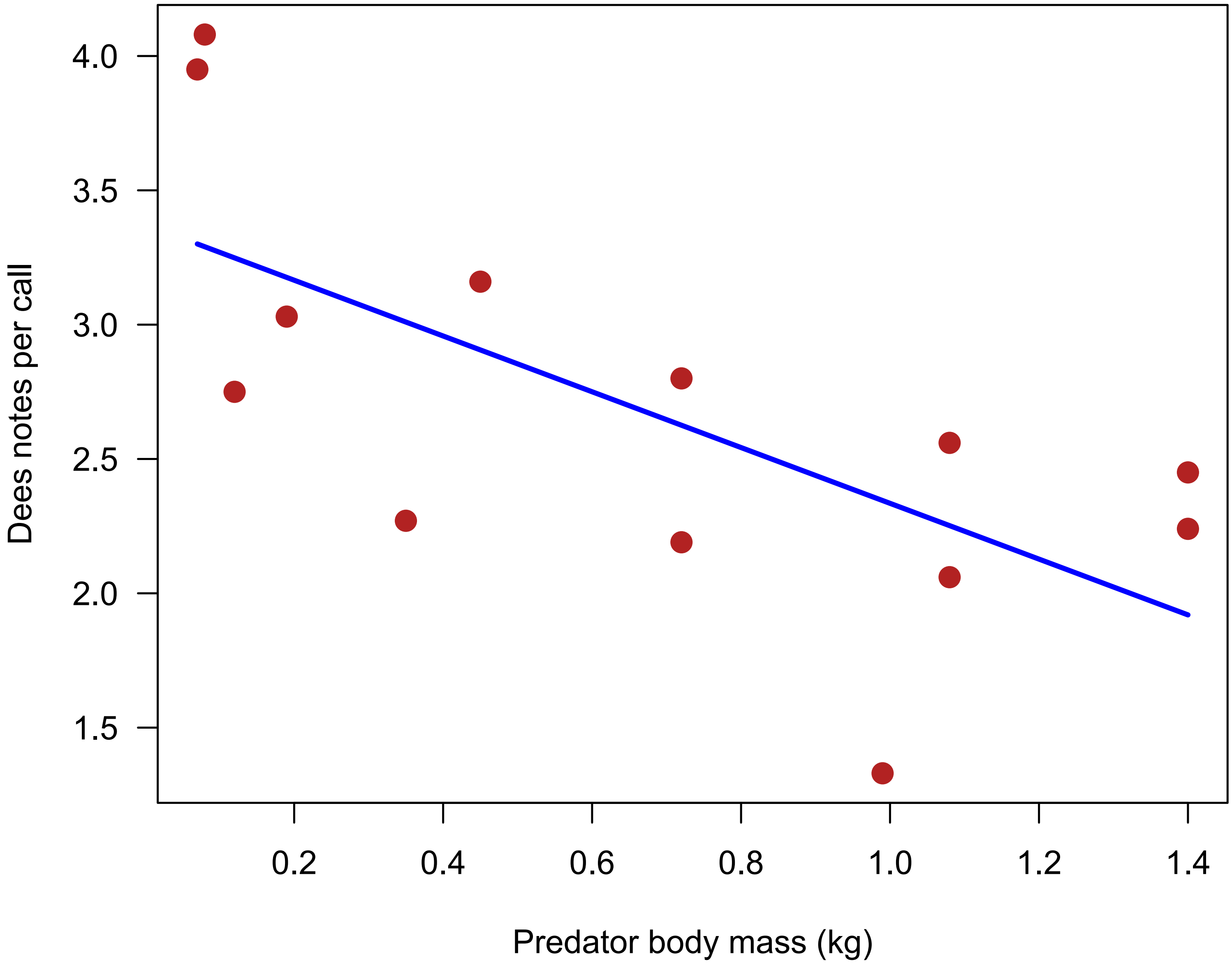
[1]   [2]

m * X + b

lines(mydata$x values,**model1$coefficients[2]***mydata$x values+**model1$coefficients[1]**)

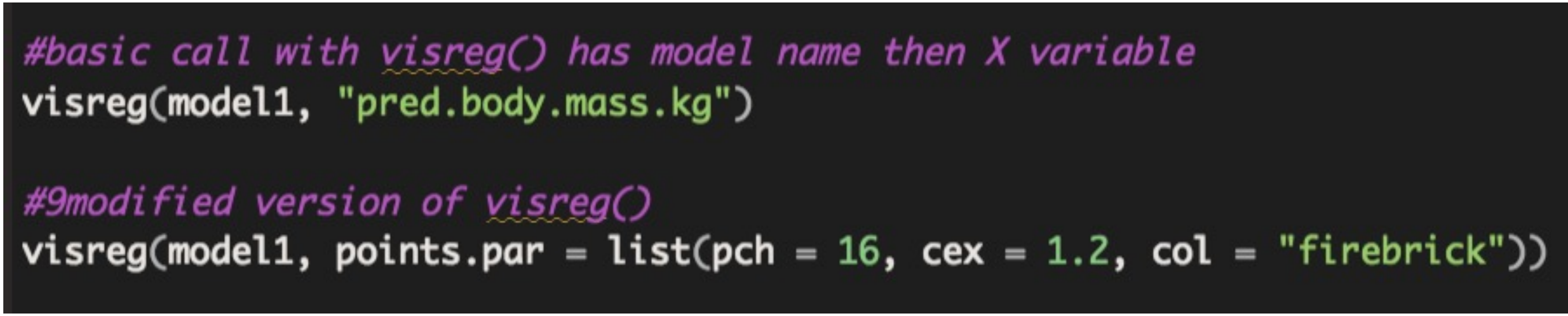
Y= -1.038208 * X+ 3.373115

Scatter plot of data and fitted model line
lines()



Example with visreg()

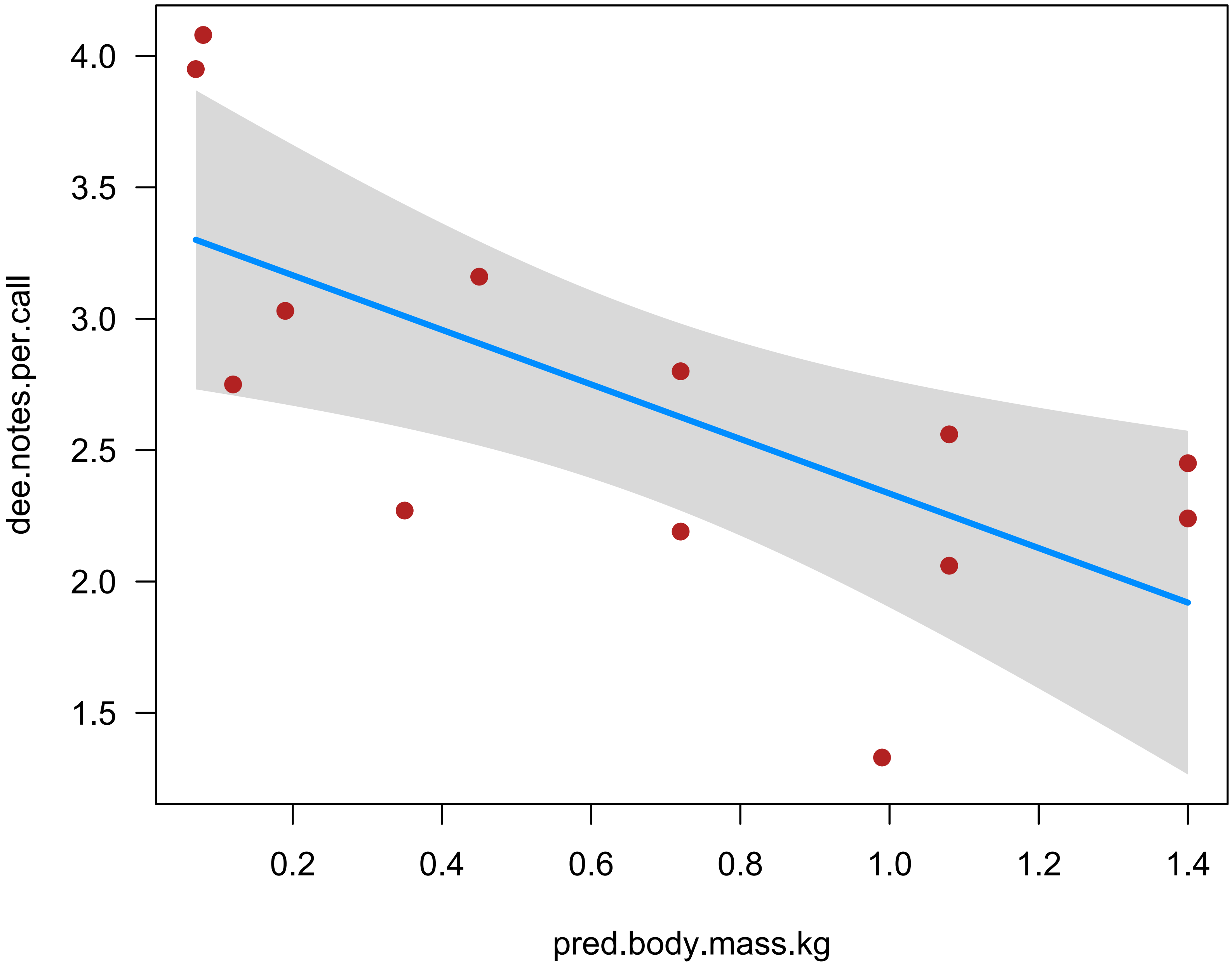
- Fast way to do scatter plot, model line, and 95%CI



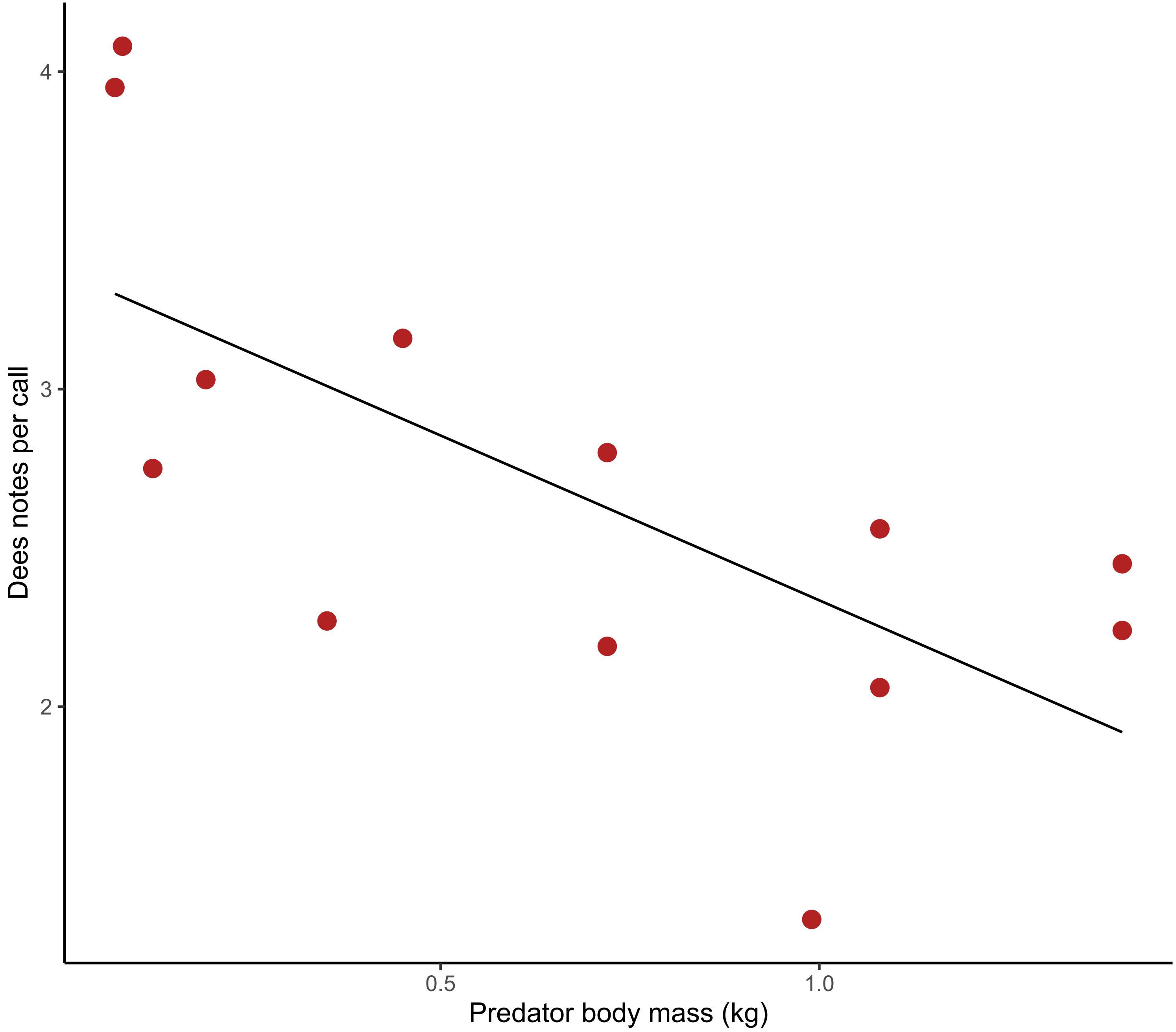
```
#basic call with visreg() has model name then X variable
visreg(model1, "pred.body.mass.kg")

#9modified version of visreg()
visreg(model1, points.par = list(pch = 16, cex = 1.2, col = "firebrick"))
```

Scatter plot of data and fitted model line, and 95% CI with visreg()



Default ggplot() model line



#5. Test model fit with anova (test hypothesis)

```
anova(model1)
```

#6. Model comparison between a full and reduced model

```
anova(null,model1)
```

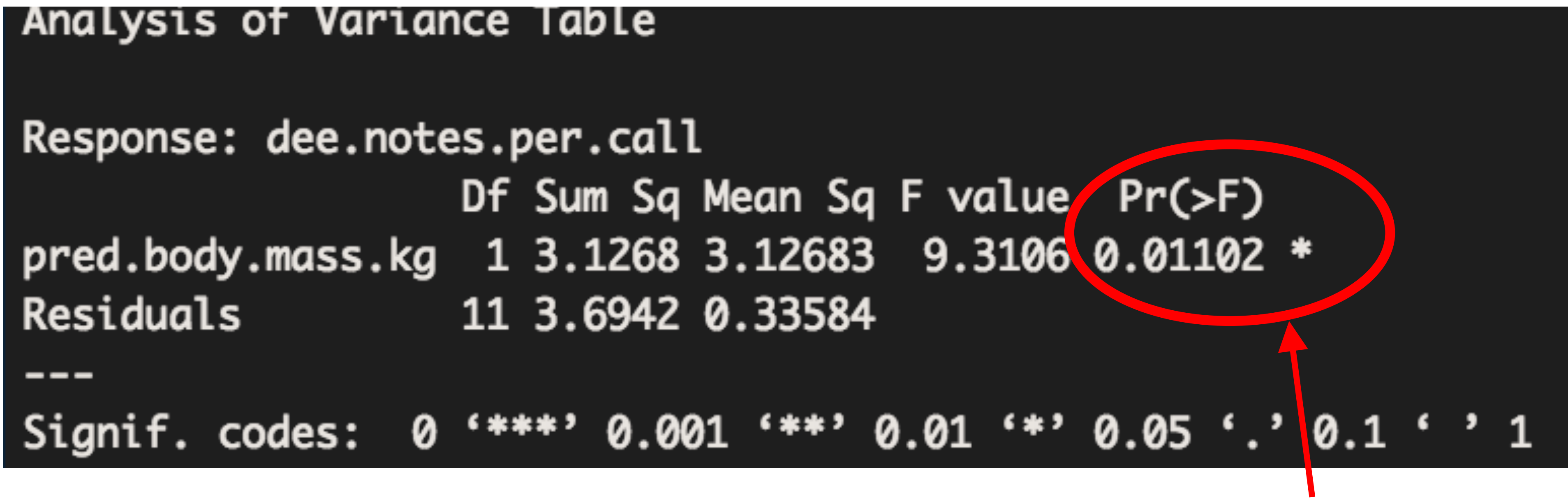
#7. Look at model assumptions on the best-fit model (diagnostics)

```
plot(model1)
```

#8. Predict() new data from model line (in workshop)

Test the hypothesis with anova(model1)

- Null hypothesis is that slope=0 (that there is no line)
- **anova(model1) asks, “Is this model linear?”**
- Yields an anova table



```
Analysis of Variance Table

Response: dee.notes.per.call
                  Df Sum Sq Mean Sq F value  Pr(>F)
pred.body.mass.kg  1 3.1268 3.12683  9.3106 0.01102 *
Residuals         11 3.6942 0.33584
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Test of null hypothesis that slope $\beta_1 = 0$

Model comparison between a full and reduced model

- Anova() on 2 models compares the model fits with an F-test
- ****must be comparing a reduced vs. full model otherwise test is invalid***
- The full model contains the term of interest and the reduced model leaves it out.
    - Reduced and full model **only differ by this 1 thing**
- Sometimes termed hierarchically nested

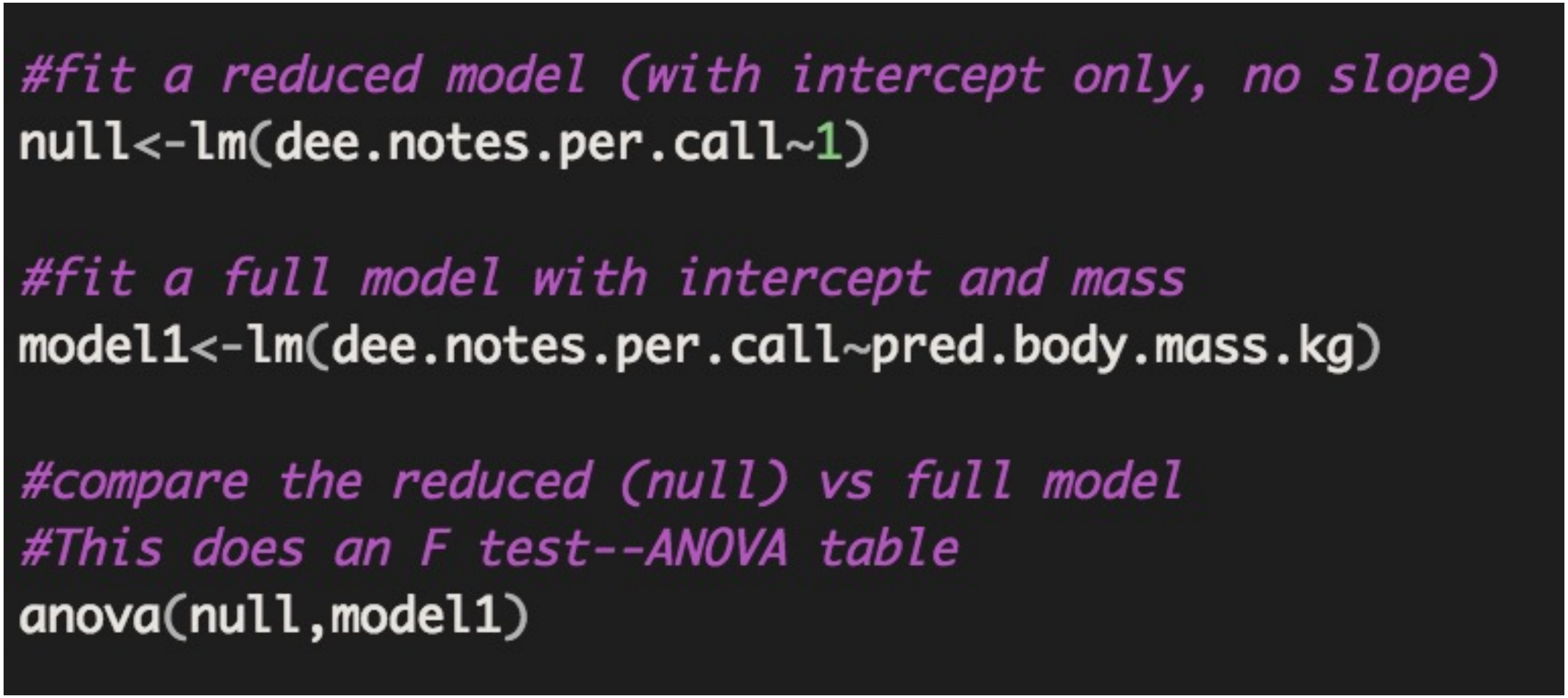
This concept will be very important when we compare mixed models also

Model comparison between a full and reduced model

- Behind the scenes, this is how R tests effect of predator body mas (x variable) on dees (y variable):

```
null<- lm(dees ~ 1)                      # fits reduced model (intercept only)
model1<- lm(dees ~ body.mass).   # fits full model intercept and mass
anova(null,model1)                       # compares fits with F test
```

Model comparison between a full and reduced model



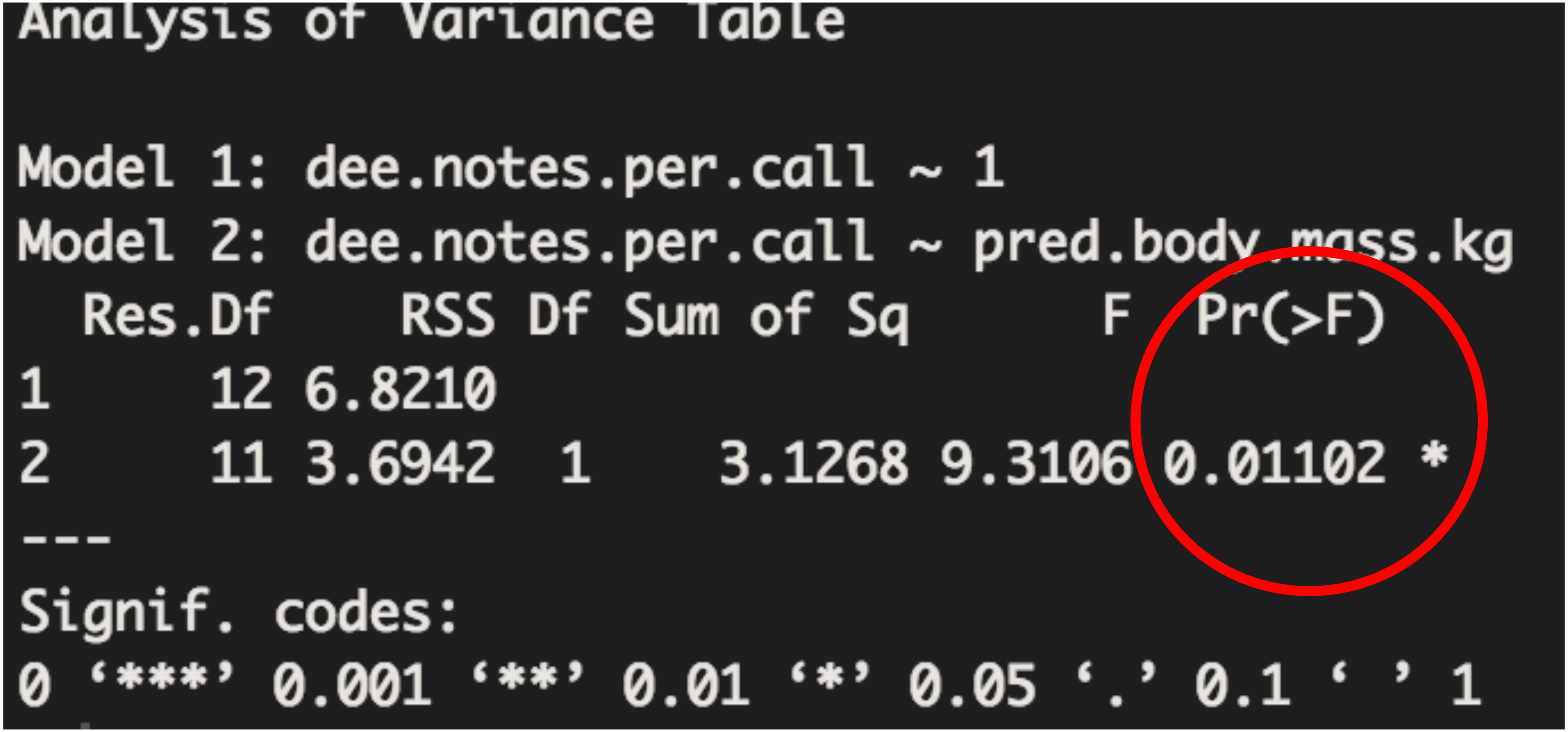
```r
#fit a reduced model (with intercept only, no slope)
null<-lm(dee.notes.per.call~1)

#fit a full model with intercept and mass
model1<-lm(dee.notes.per.call~pred.body.mass.kg)

#compare the reduced (null) vs full model
#This does an F test--ANOVA table
anova(null,model1)
```

Model comparison between a full and reduced model

Anova(null,model1) produces an F-test R output



```
Analysis of Variance Table

Model 1: dee.notes.per.call ~ 1
Model 2: dee.notes.per.call ~ pred.body.mass.kg
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)
1     12 6.8210
2     11 3.6942  1    3.1268 9.3106 0.01102 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Don’t mix up the anovas

anova (model1)→tests hypothesis
“Is it linear?”

anova(null,model1)→ compares full vs reduced models

“Is full model better than the reduced model” or is adding this factor better than not adding it?

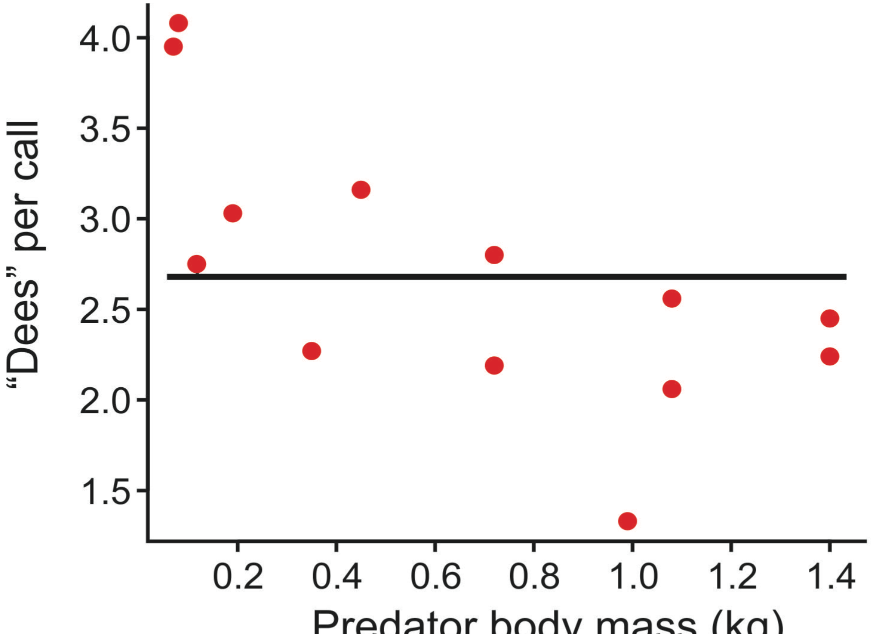
Visually, how R compares models

anova(null,model1)

The test of predator body mass involves a comparison of these two models

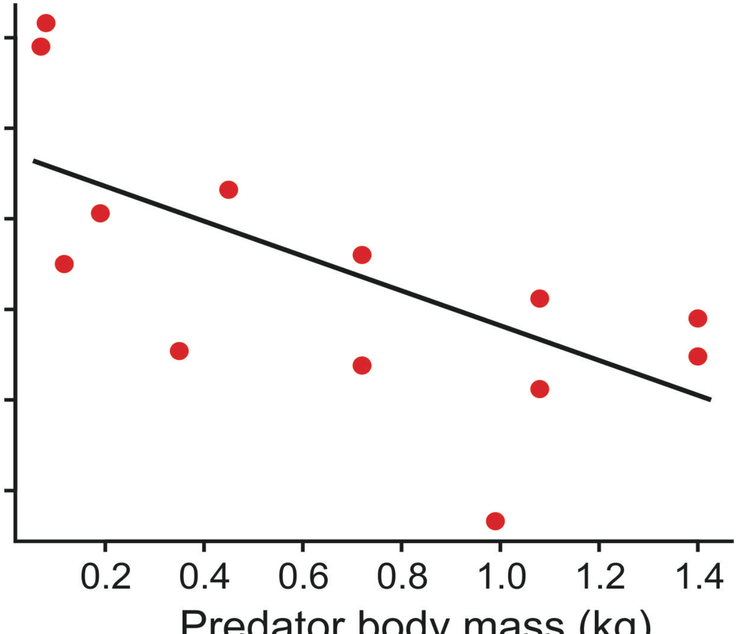
dees ~ 1

reduced model (fits only an intercept)



dees~ mass

full model (intercept and slope)



Outline for today

- What is a linear model
- Example fitting and comparing a model
- Model comparison: full vs reduced
- Assessing model fits and assumptions
- Sequential vs marginal testing of terms

Core Assumptions of linear models

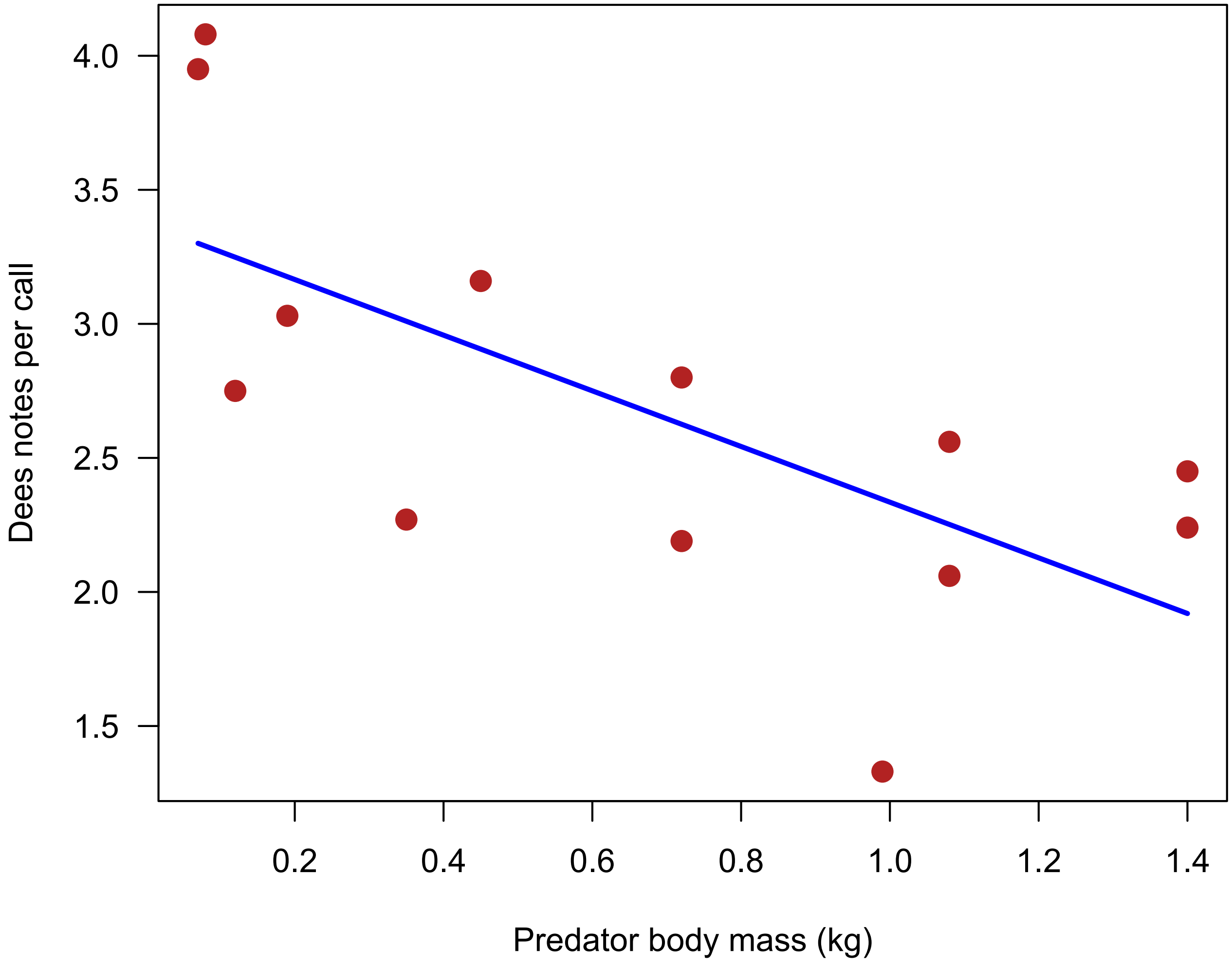
1. Normally-distributed errors
2. Equal variance of residuals in all groups
3. Independent errors (random sample; no pseudoreplication)
4. Continuous covariates have the same range of values in all groups
5. Sphericity: the variances of the differences between all pairs of factor levels are equal (more next week).

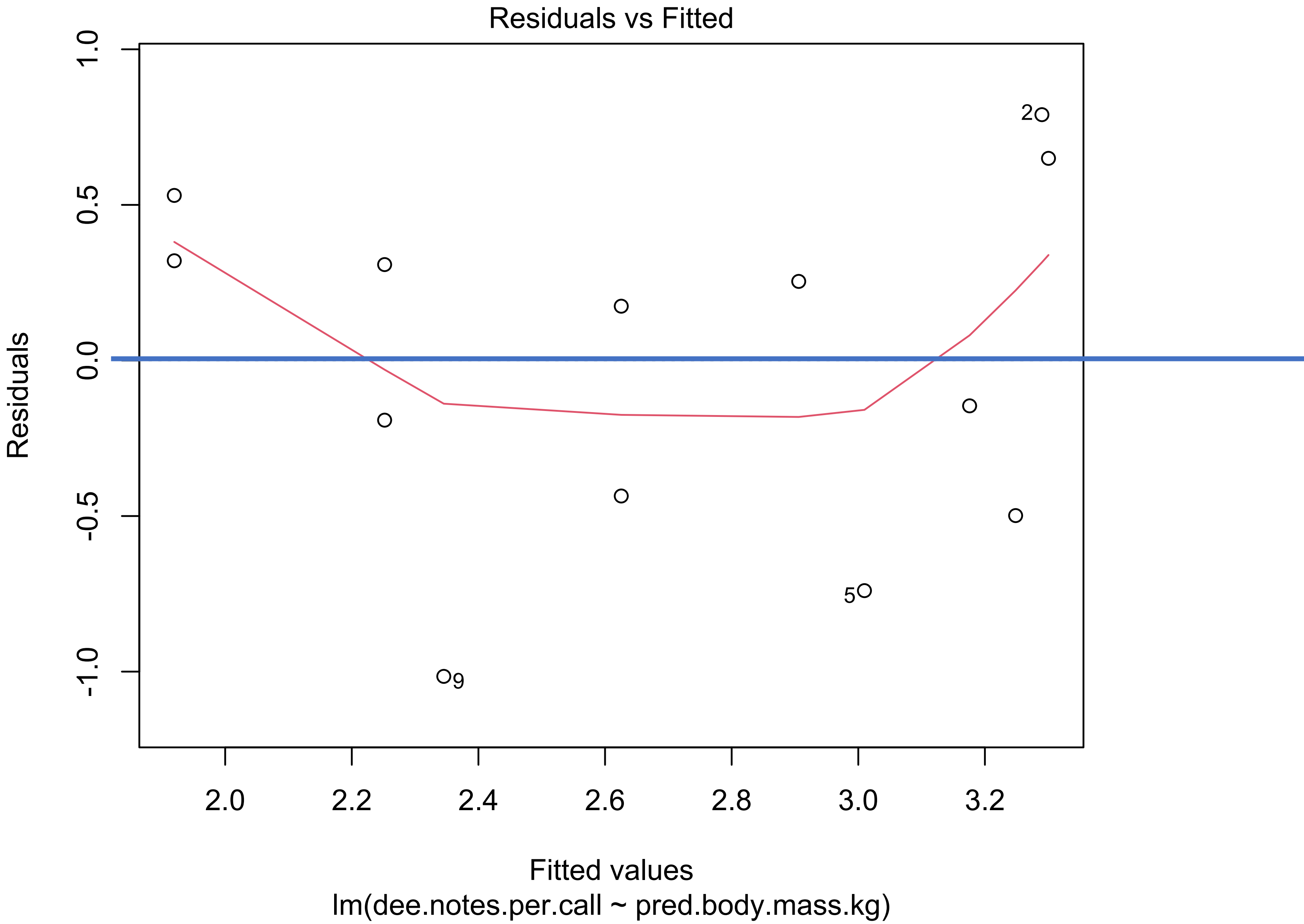
Linear models are reasonably robust to departures from assumptions 1 and 2, especially if sample size is large and balanced. However, outliers can cause problems.

Workshop this week: assess assumptions

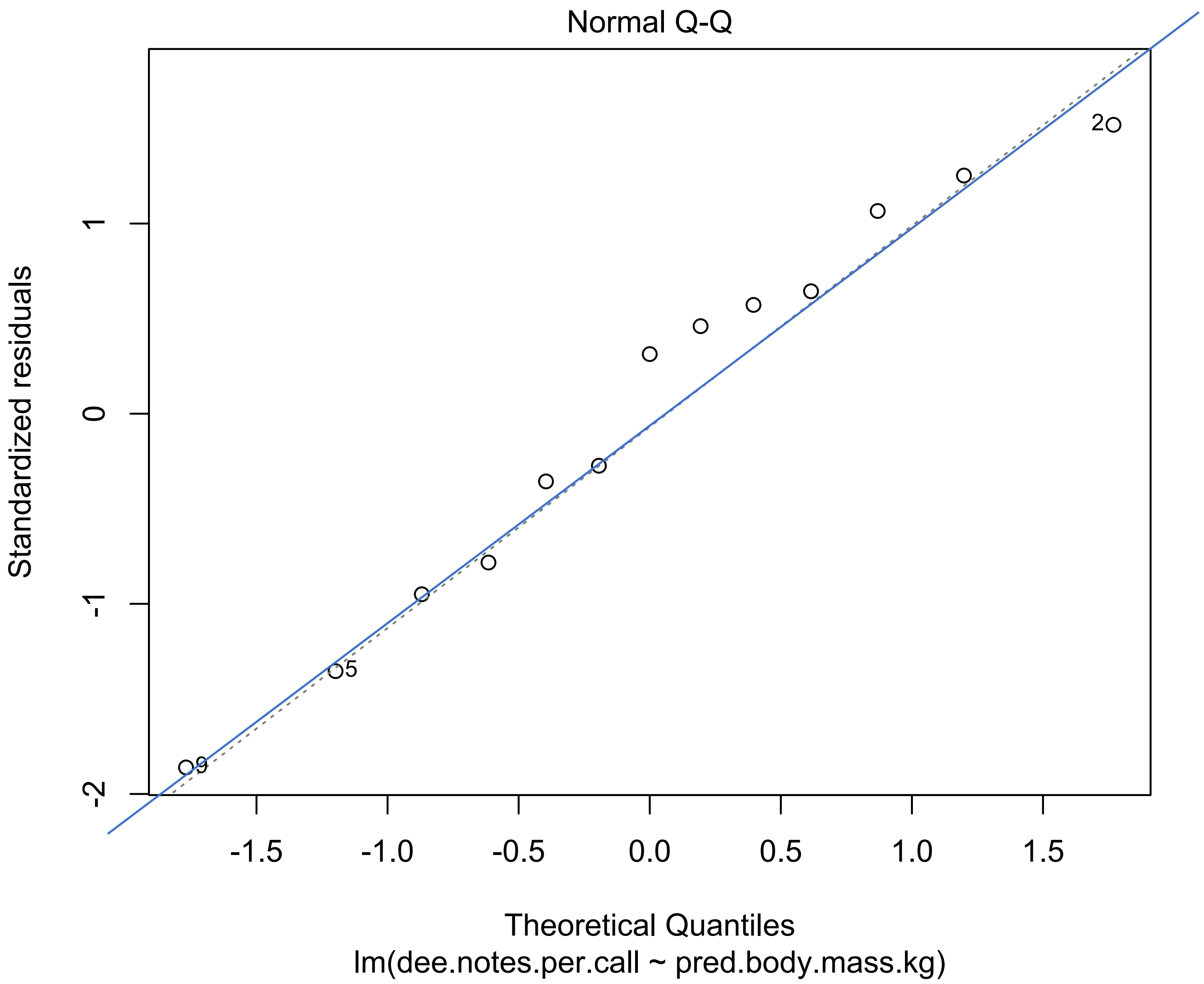
Assess the fit of our model on chickadees with diagnostic plots

Plot(model1)

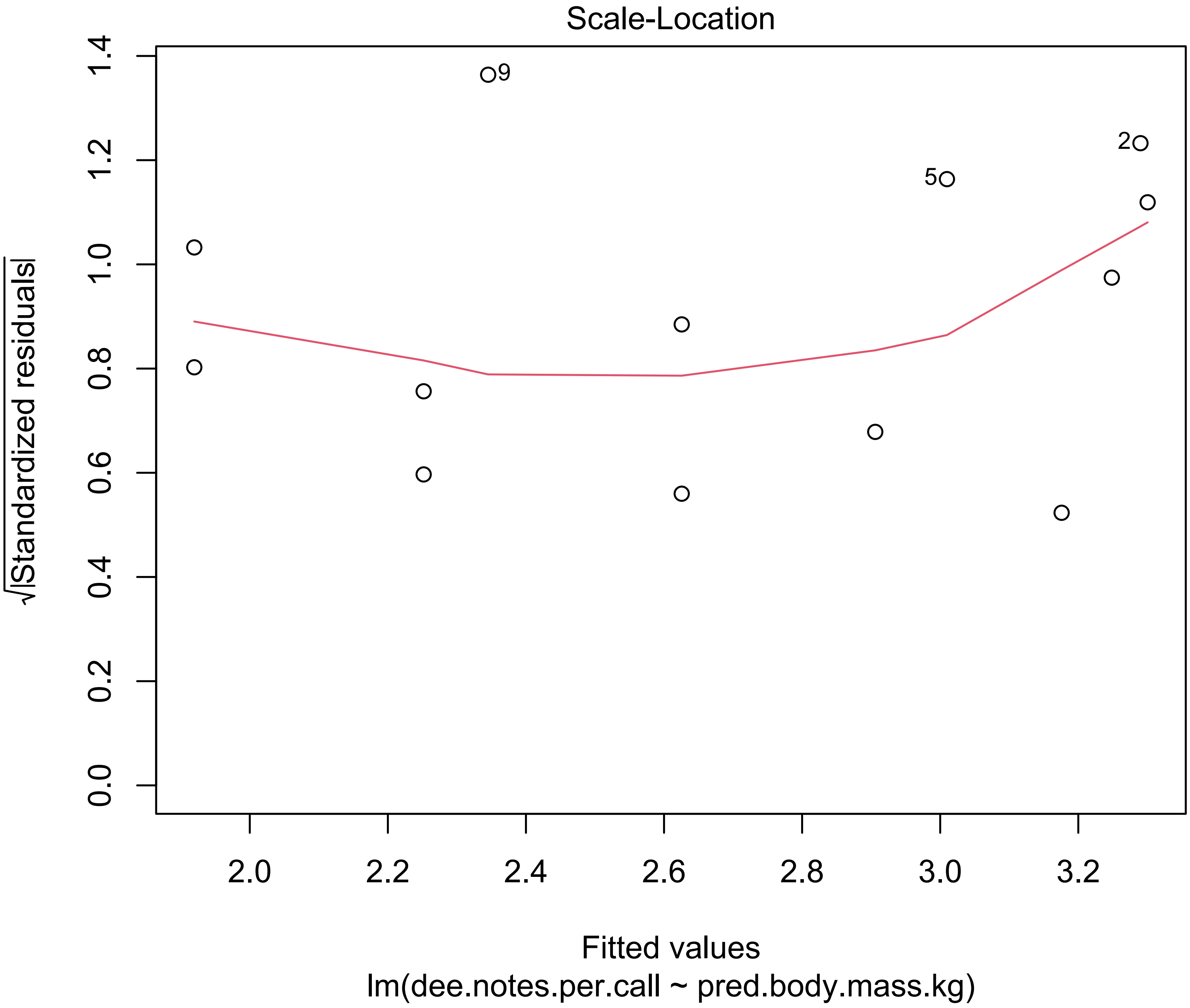




Residuals closer to 0 indicate a better fit



If data is normal, it will be *mostly* on the diagonal line


Scale-Location
√|Standardized residuals|
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
9
2
5
2.0
2.2
2.4
2.6
2.8
3.0
3.2
Fitted values
lm(dee.notes.per.call ~ pred.body.mass.kg)

Residual vs. Leverage Plot

“Leverage”

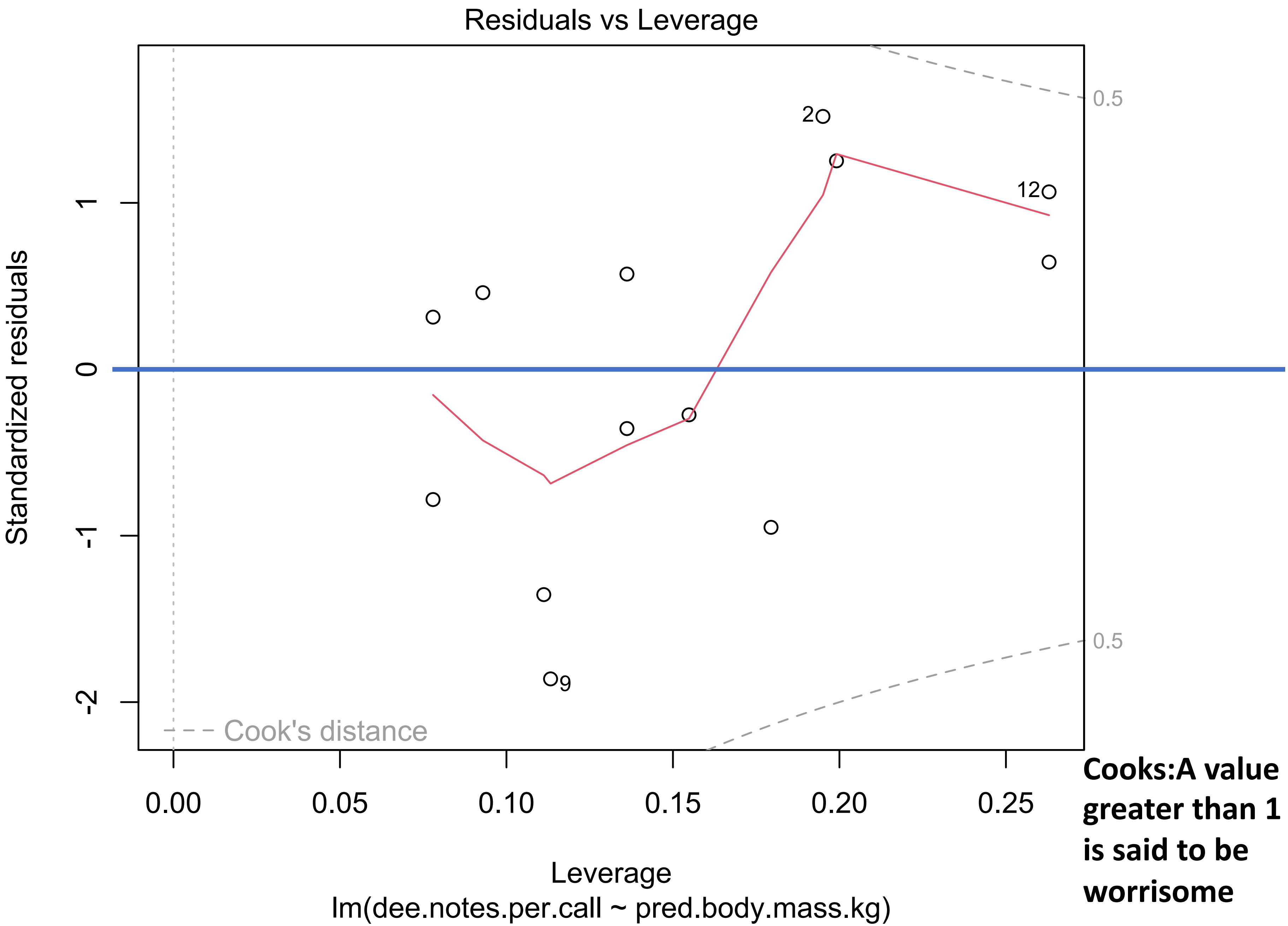
- Calculates the influence that each data point has on the estimated parameters.
- For example if the slope changes a great deal when a point is removed, that point is said to have high leverage.

“Cook’s distance”

- Effect of each data point on the predicted values for all the other data points. **A value greater than 1 is said to be worrisome**.
- Points with high leverage don’t necessarily have high Cook’s distance, and vice versa.

Ideally, you want both to be low or close to 0 and uniform

Ideally, you want both to be low or close to 0 and uniform



Cooks:A value greater than 1 is said to be worrisome

Workshop

Workshop Thurs: Linear Models

- Only looking at **fixed** effects→use lm()
- **See the “Fit model” and ”Graphs & Tables” R Tips pages**
- Fit a linear model with lm()
- Obtain coefficient estimates and standard errors
- R^2
- 95% CI for a linear model
- Visreg()
- Assess if the assumptions are met
- Prediction intervals with predict()
- Test if a categorical variable is a significant factor in a lm model

#1.scatter plot (examine data)

plot(y ~ x, data = mydata)

#2. Fit linear model

model1<- lm(y ~ x, data=mydata)

#3. Extract coefficients and information from the model

summary(model1) and model1$coefficients

#4.Add model line to scatter plot above

abline() or lines() or ggplot()

Plot CI with visreg()

predict()

#5. Test model fit with anova

anova(model1)

#6. Look at model assumptions (diagnostics)

plot(model1)

#7. Predict() new data from model line (in workshop)

Extensions to linear models

What if your residuals aren’t normal because of outliers? Nonparametric methods exist, but these don’t provide parameter estimates.

- Robust regression methods (rlm)

What if response data are binary or discrete?

- Generalized linear models (glm)

What if there are random effects?

- Linear mixed effects models (lme)

What if residuals are not independent because of autocorrelation or phylogeny?

- General least squares methods (gls)