Basic Linear Models in R

Clarke van Steenderen

2025-02

R tutorial 4: Running Linear Models

A linear model is used when one wants to investigate the relationship between predictor variables (x) (e.g. body mass, temperature, humidity) and a continuous response variable (y) (e.g. fecundity, longevity, height). It assumes a linear relationship, and can be used to make predictions (i.e. can x be used to predict y?). When the response variable takes the form of a binomial value (e.g. dead or alive, 1 or 0), or count, for example, then a generalised linear model (GLM) is appropriate as this method can handle data with non-normal error distributions and variances.

We will work with data from an experiment that aimed to find whether insect body mass and environmental temperature has an effect on reproductive output. There are four temperature levels: 15, 20, 25, and 30 degrees Celsius. The number of larvae produced were recorded per treatment. This tutorial was adapted from Guy Sutton's GitHub page.

```
if (!require("pacman"))
  install.packages("pacman")
```

```
## Warning: package 'pacman' was built under R version 4.3.3
```

```
pacman::p_load(xlsx, janitor, ggplot2, Rmisc, dplyr, tidyverse, effectsl)
```

```
## Warning: package 'effectsl' is not available for this version of R
##
## A version of this package for your version of R might be available elsewhere,
## see the ideas at
## https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
## Warning: unable to access index for repository http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contri
##   cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/4.3/PACKAGES'
```

```
## Warning in p_install(package, character.only = TRUE, ...):
```

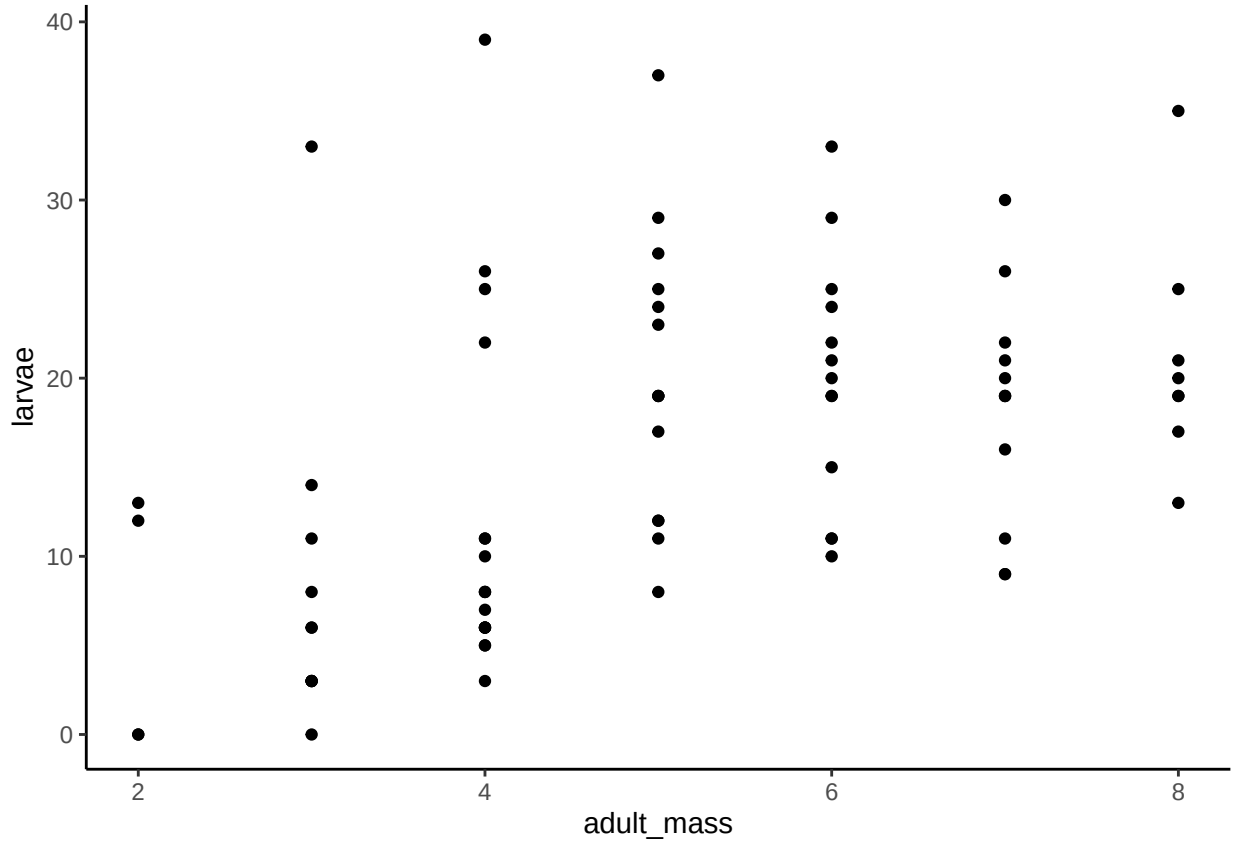
```
## Warning in library(package, lib.loc = lib.loc, character.only = TRUE,
## logical.return = TRUE, : there is no package called 'effectsl'
```

```
## Warning in pacman::p_load(xlsx, janitor, ggplot2, Rmisc, dplyr, tidyverse, : Failed to install/load:
## effectsl
```

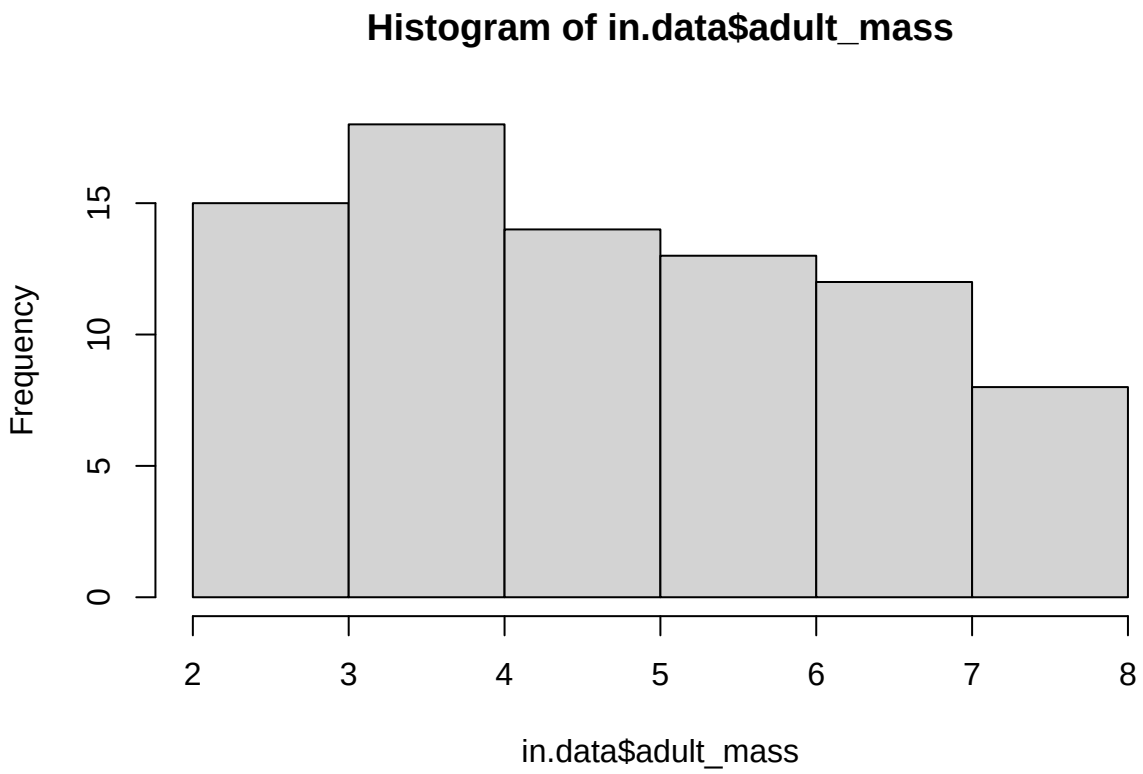
```r
# read in the data
in.data = readxl::read_excel("data/poisson_data.xlsx")

str(in.data)
```

```
## tibble [80 x 4] (S3: tbl_df/tbl/data.frame)
##  $ temp      : chr [1:80] "a" "a" "a" "a" ...
##  $ adult_mass: num [1:80] 3 5 4 6 3 4 7 5 2 3 ...
##  $ larvae    : num [1:80] 0 17 6 11 11 5 9 12 12 14 ...
##  $ adults    : num [1:80] 0 0 0 0 0 1 1 0 0 0 ...
```

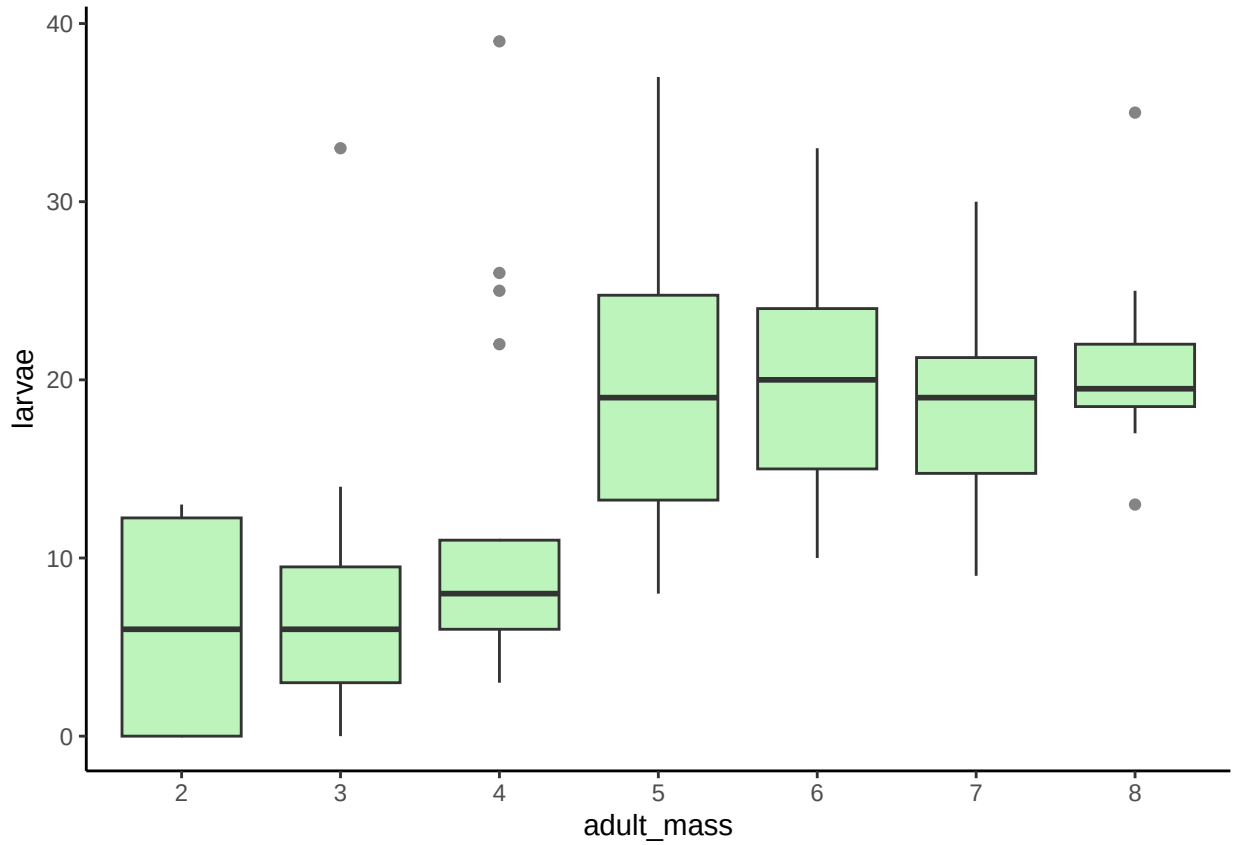
```r
# do some preliminary visualisation
ggplot2::ggplot(data = in.data, aes(x = adult_mass, y = larvae)) +
  geom_point() +
  theme_classic()
```




```r
# how is the data distributed?
hist(in.data$adult_mass)
```




```
# are there outliers?
raw_data <- in.data %>%
  dplyr::mutate(adult_mass = as.factor(adult_mass))

ggplot(data = raw_data, aes(x = adult_mass, y = larvae)) +
  geom_boxplot(fill = "lightgreen", alpha = 0.6) +
  theme_classic()
```




```
# Set plot theme
theme_set(theme_classic(base_size = 12) +
theme(panel.border = element_rect(colour = "black", fill = NA),
axis.text = element_text(colour = "black"),
axis.title.x = element_text(margin = unit(c(2, 0, 0, 0), "mm")),
axis.title.y = element_text(margin = unit(c(0, 4, 0, 0), "mm")),
legend.position = "none"))
```

We can now run a linear model to see whether adult mass has an effect on the number of larvae produced. We use the function **lm()**, and tell it that we want to investigate how our response variable (y) is affected by the predictor variable (x). This is written in the form y ~ x, where the tilde (~) means "according to" or "affected by". Below, the code says "how is the number of larvae affected by adult weight?". The general form of a linear model is:

model1 = lm(y ~ x, data = data)

model1 is the name we are assigning to our model, where it will be saved to
lm is the function "linear model" that is run in R
data is the name of the dataset we are interested in
y is our response variable, from **data**
x is our predictor variable, from **data**

Our linear regression equation can be written in the form $\mathbf{y = mx + c}$, such that:

$\mathbf{larvae} = \beta_0 + \beta_1\mathbf{(adult\ mass)} + \epsilon_i$

Where β_0 is the y-intercept (i.e. the number of larvae when mass = 0), and β_1 is the gradient/slope coefficient (i.e. the change in the number of larvae for every unit increase in adult body mass). The ϵ_i part is a random

error term, which indicates the difference between the actual number of larvae (y values), and the expected number of larvae based on the linear model. This gives an indication of how much the measured number of larvae was not due to the linear effect of adult body mass.

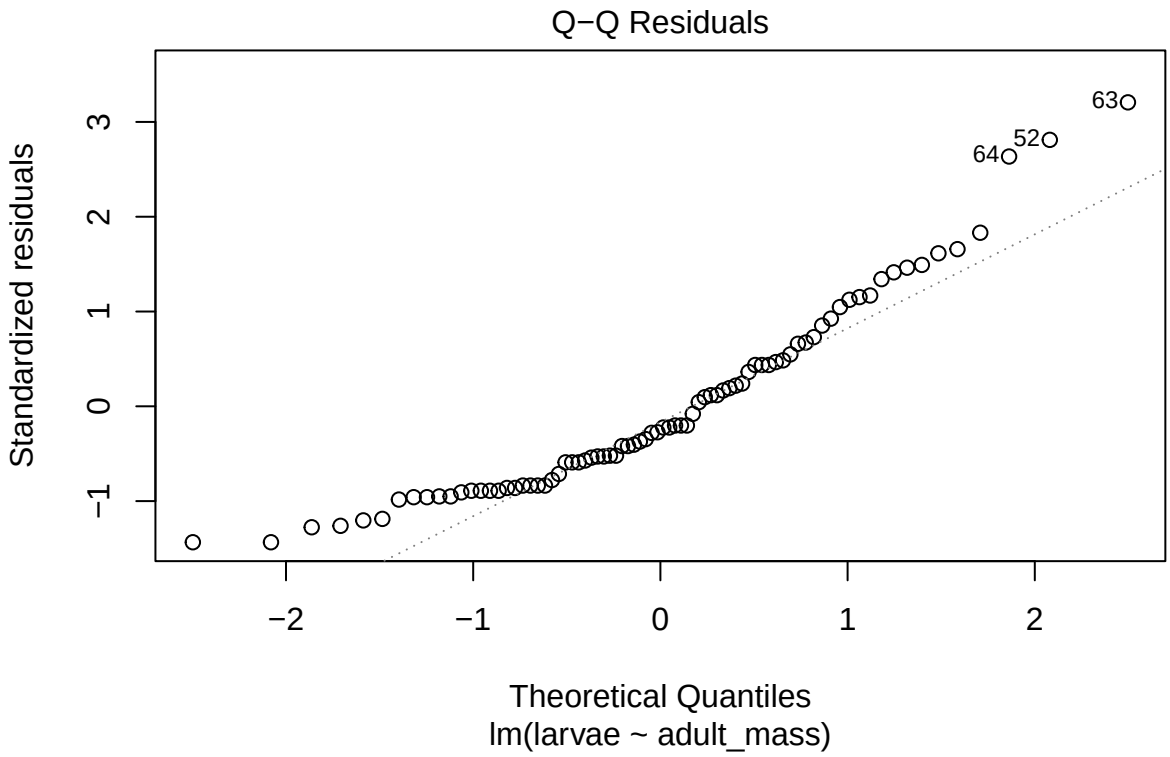
```
linmod1 = lm(larvae ~ adult_mass, data = in.data)
```

We need to make sure that running a linear model was statistically sound. The underlying assumptions include:

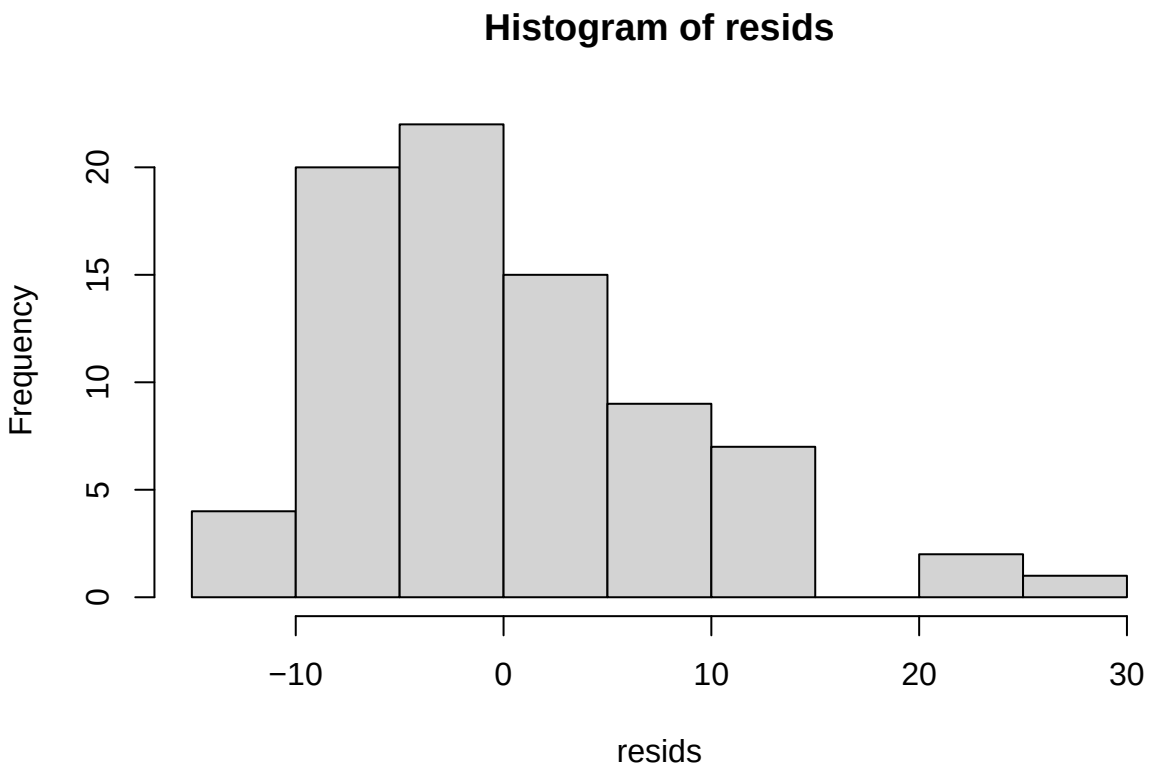
- **Linearity**: that there is a linear relationship between x and y. You can check this by having a look at a plot of residual vs fitted plot, where you expect to see residual values clustering around the y = 0 line, and no clustering/pattern in the points
- **Independence**: that data points are independent -> this depends on the experimental design. This means that the data collected from one plot/individual/area etc. is not affected by the data collected in others. For example, perhaps one wants to investigate behavioural patterns in a particular species in an enclosure. You might take measurements on multiple individuals in that enclosure, but not realise that the behaviour of one individual may affect that of others. This would be non-independent data. Non-independence can also come about due to spatial autocorrelation: measurements taken from areas that are geographically close together are more likely to share similar features that might be different from those that you are actually measuring. For example, maybe you want to look at the relationship between the abundance of a particular plant and a soil nutrient (e.g. nitrogen). Taking these measurements in plots of land that are close together (spatially autocorrelated) might lead one to conclude that a high abundance of the plant across plots may be linked to a high nitrogen concentration, when actually, they have similar abundances due to another reason that is linked to their spatial proximity (perhaps they are in an area with fewer herbivores, maybe there is greater access to water in that particular area, etc.).
- **Normality of residuals**: that a QQ (quantile-quantile) plot of observed versus expected residuals cluster around a y = x straight line (gradient = 1), with no patterning. A kolmogorv-Smirnoff (KS) test can tell whether residuals follow a normal distribution ($p > 0.05$) or not ($p < 0.05$).
- **Equal variance**: that the residual vs fitted value plot shows an equal distribution of y-values (sometimes referred to as a "shotgun" plot, or "homoscedasticity"); i.e. no patterning. DHARMa plots show three horizontal dotted lines -> at 0.25, 0.5, and 0.75. We should see horizontal bold lines falling on all three of these.

Let's make sure that our residuals are normally distributed. When we create a QQ plot, we are looking for a straight line - there should be no pattern in the data.

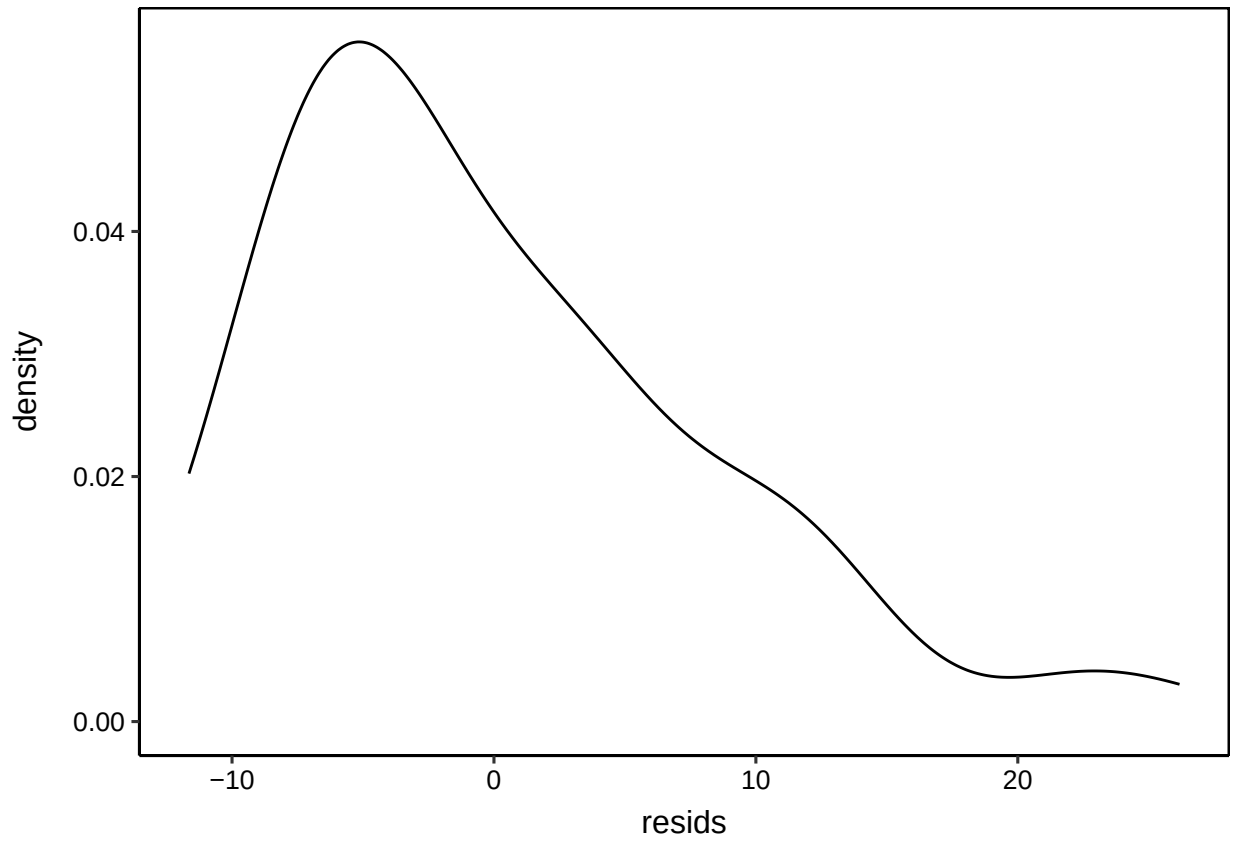
```
# QQ plot
plot(linmod1, which = 2)
```



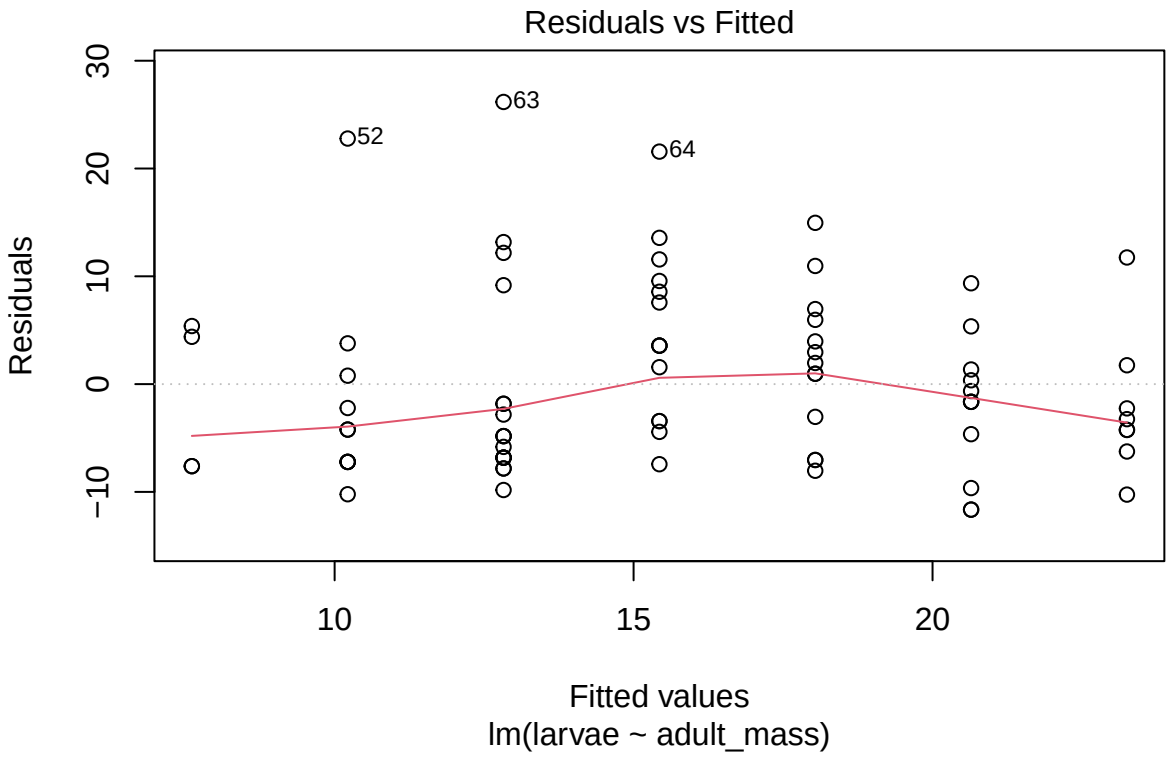

```
# residuals
resids = resid(linmod1)
resids.df = as.data.frame(resids)

# plot a histogram -> appears that the resids are not normally distributed
hist(resids)
```



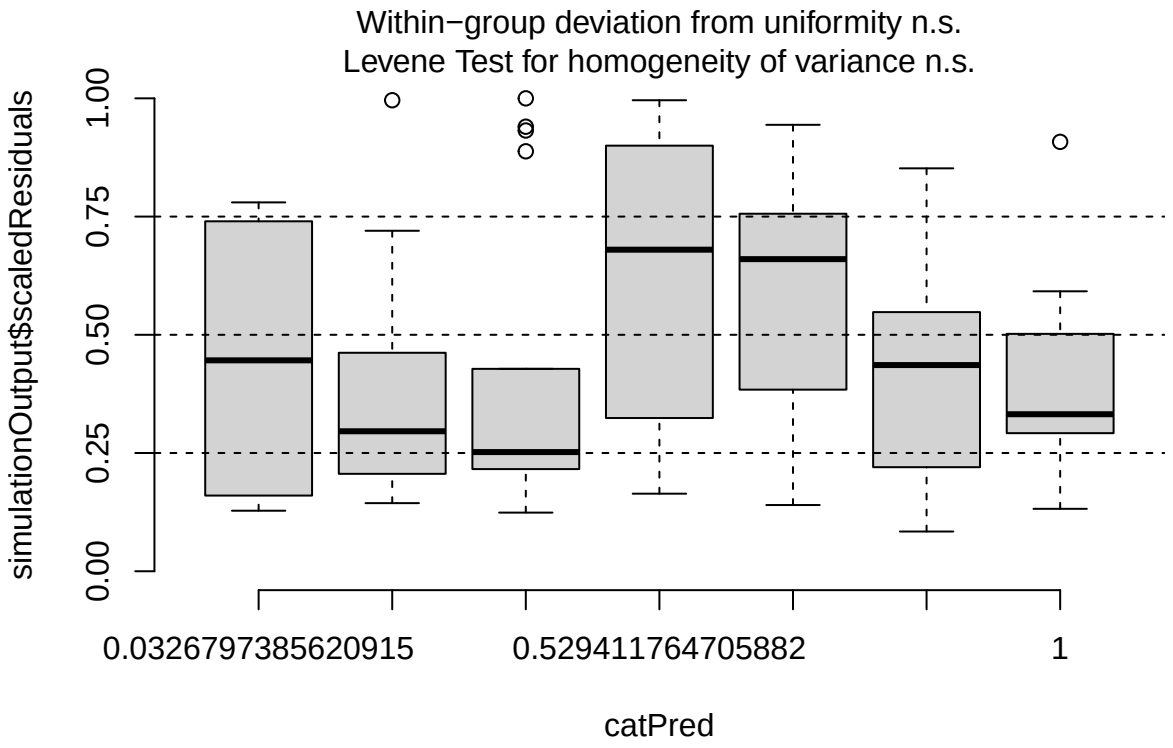

```
ggplot2::ggplot(data = resids.df, aes(y = resids)) +
  #geom_histogram() +
  geom_density() +
  coord_flip()
```



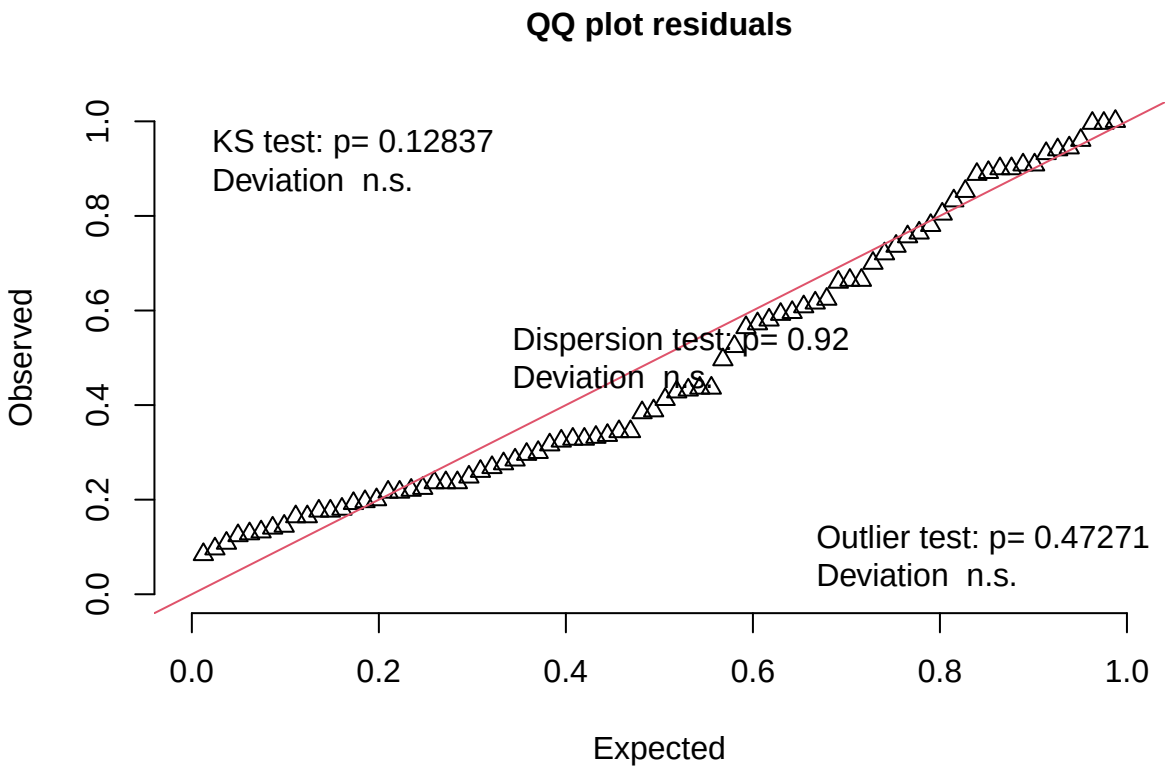

```
# have a look at the homogeneity of variance
# ideally, the red line should lie on the y = 0 axis line, and there should be
# no pattern in the points
plot(linmod1, which = 1)
```



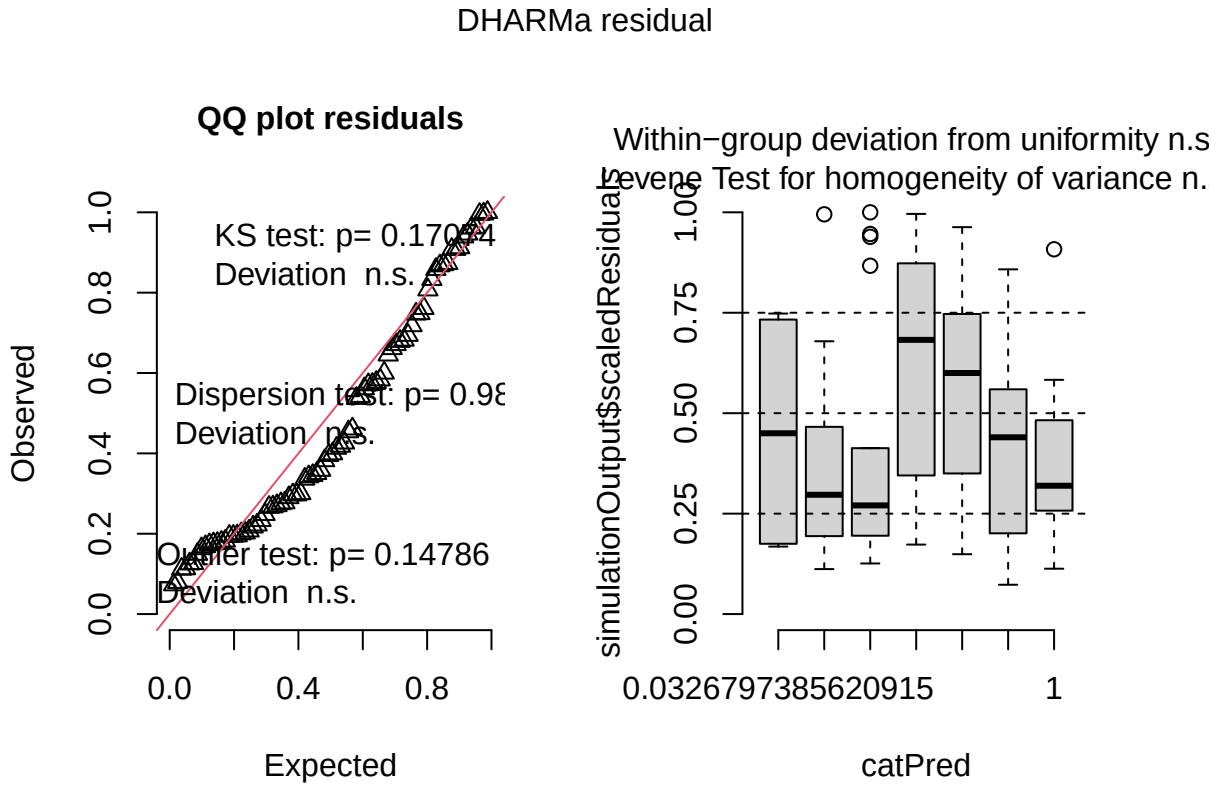

```
# you can also use the DHARMa package:
DHARMa::plotResiduals(linmod1)
```




```
DHARMa::plotQQunif(linmod1)
```




```
# you could also run it like this, setting it to 1000 simulations
DHARMa::simulateResiduals(linmod1, n = 1000, plot = TRUE)
```




```
## Object of Class DHARMa with simulated residuals based on 1000 simulations with refit = FALSE . See ?
##
## Scaled residual values: 0.112 0.572 0.179 0.2 0.537 0.163 0.079 0.336 0.718 0.679 0.202 0.125 0.345
```

There appears to be equality of variance, and the DHARMa QQ plot suggests normality of the residuals. We could explore a GLM at this point, but for the sake of running a LM, let's continue:

```
summary(linmod1)
```

```
##
## Call:
## lm(formula = larvae ~ adult_mass, data = in.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.645  -6.825  -2.022   4.068  26.175
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3990     2.9254   0.820    0.415
## adult_mass    2.6066     0.5431   4.799 7.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.236 on 78 degrees of freedom
```

```
## Multiple R-squared:  0.228,  Adjusted R-squared:  0.2181
## F-statistic: 23.03 on 1 and 78 DF,  p-value: 7.521e-06
```

```
# let's get an analysis of deviance table. This tells us that adult mass had
# a significant positive effect on the number of larvae produced
car::Anova(linmod1, test = "Chisq", type = "II")
```

```
## Anova Table (Type II tests)
##
## Response: larvae
##             Sum Sq Df F value    Pr(>F)
## adult_mass 1562.6  1  23.034 7.521e-06 ***
## Residuals  5291.4 78
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# same as
anova(linmod1, test = "F")
```

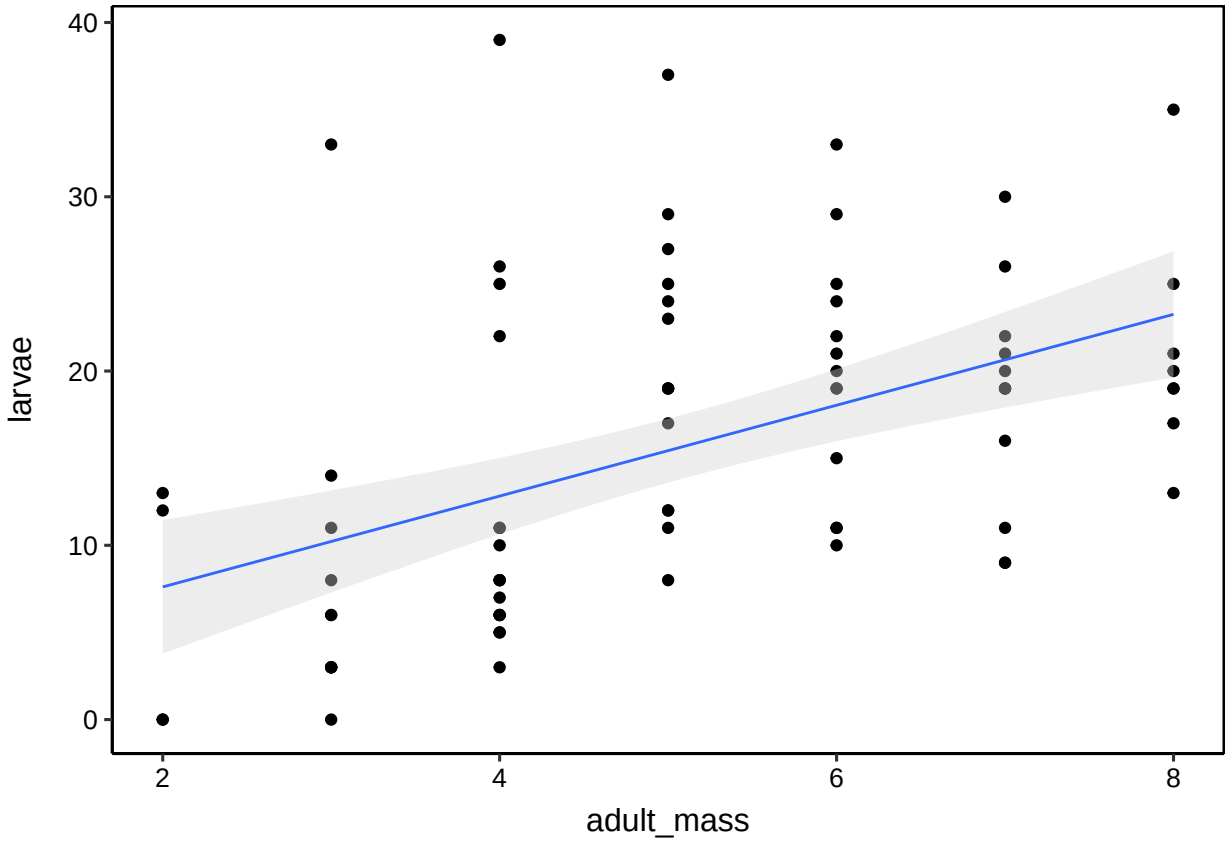
```
## Analysis of Variance Table
##
## Response: larvae
##             Df Sum Sq Mean Sq F value    Pr(>F)
## adult_mass   1 1562.6 1562.56  23.034 7.521e-06 ***
## Residuals   78 5291.4   67.84
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Get the beta 1 value
# < 0 means that the response decreases with greater predictor values
# > 0 means that the response increases with greater predictor values
coef(linmod1)[2]
```

```
## adult_mass
##   2.606555
```

Since $p < 0.05$, we can conclude that adult mass does have a significant effect on reproductive output. The β_1 estimate value of 2.61 means that there is a positive relationship between the predictor and response variable. This value also means that for every 1 unit increase in adult body mass, reproductive output increases by 2.6 individual larvae (absolute value).

```
ggplot2::ggplot(data = in.data, aes(x = adult_mass, y = larvae)) +
  geom_point() +
  geom_smooth(method = "lm", fill = "lightgrey", linewidth = 0.5, fullrange = TRUE)
```



Let's have a look at confidence intervals (CI) now. The 95% CI tells us how much uncertainty is in the model.

```
confint(linmod1)
```

```
##                 2.5 %   97.5 %
## (Intercept) -3.424936 8.222914
## adult_mass   1.525312 3.687797
```

This tells us that the 95% confidence interval for adult body mass is between 1.53 - 3.69. In other words, a 1 g increase in adult body mass will yield in the region of 1.53 - 3.69 more larvae. The 95% CI implies that if the experiment were to be repeated 100 times, our estimate value will fall within the 95% CI range (1.53 - 3.69) 95 times.

How much variation is explained by the model? We'll get an R-squared value to determine this, which tells you how much of the variation in the response variable (larvae) is explained by the predictor (adult mass).

```
summary(linmod1)$adj.r.squared
```

```
## [1] 0.2180824
```

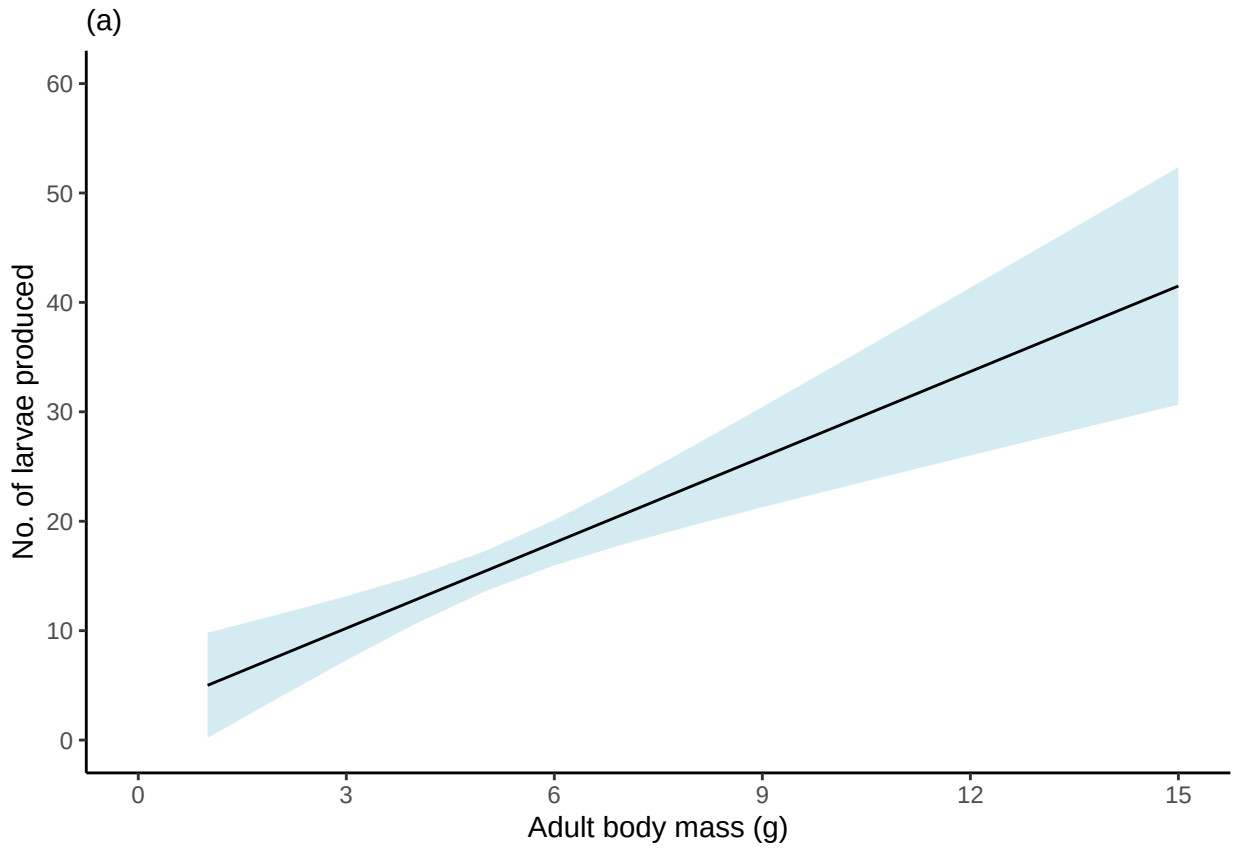
This means that 22% of the variation observed in reproductive output is explained by adult body mass.

Let's do some plotting:

```r
adult_mass = seq(1, 15, 1)

preds = predict(linmod1, list(temp = adult_mass), interval = "confidence") %>%
  as.data.frame()

preds <- dplyr::bind_cols(preds, as.data.frame(adult_mass))

head(preds)
```

```
##         fit        lwr       upr adult_mass
## 1  5.005544  0.1958364  9.815251          1
## 2  7.612098  3.7797832 11.444414          2
## 3 10.218653  7.2898024 13.147504          3
## 4 12.825208 10.6325234 15.017892          4
## 5 15.431763 13.5944452 17.269080          5
## 6 18.038317 15.9690719 20.107563          6
```

```r
# Plot your model predictions
ggplot() +
  # Add a ribbon of 95% confidence intervals
  geom_ribbon(data = preds, aes(x = adult_mass, ymin = lwr, ymax = upr),
              fill = "lightblue", alpha = 0.5) +
  # Add line of model prediction
  geom_line(data = preds, aes(x = adult_mass, y = fit)) +
  # Define y-axis limits
  scale_y_continuous(breaks = seq(0, 60, 10),
                     limits = c(0, 60)) +
  # Define x-axis limits
  scale_x_continuous(breaks = seq(0, 15, 3),
                     limits = c(0, 15)) +
  # Write x and y axis labels
  labs(x = "Adult body mass (g)",
       y = "No. of larvae produced",
       subtitle = "(a)") +
  theme_classic()
```



Write this up:

Larger females produced more larvae than smaller adults ($P < 0.05$). Approximately 21% of the variation in fecundity was explained by female body mass (Adj. R-squared = 0.21).For every 1g increase in adult body mass, adults produce approximately 2.61 more larvae (95% CI: 1.52 - 3.68).

Hypothesis testing

When we talk about hypothesis testing, we are referring to the NULL (H0) and ALTERNATIVE (H1) hypothesis. Here, the null hypothesis would be that adult body mass has no significant effect on the number of larvae produced. The alternative hypothesis is that mass does have a significant effect on larval output.

If we want to show that including body mass as a predictor in our model explains the data better, we can create two models representing our H0 and H1. Let's implement a GLM here, instead of the LM above. We'll apply the Gaussian family for normally-distributed data:

```
H0.model = glm(larvae ~ 1, data = in.data, family = gaussian)
H1.model = glm(larvae ~ 1 + adult_mass, data = in.data, family = gaussian)

summary(H0.model)
```

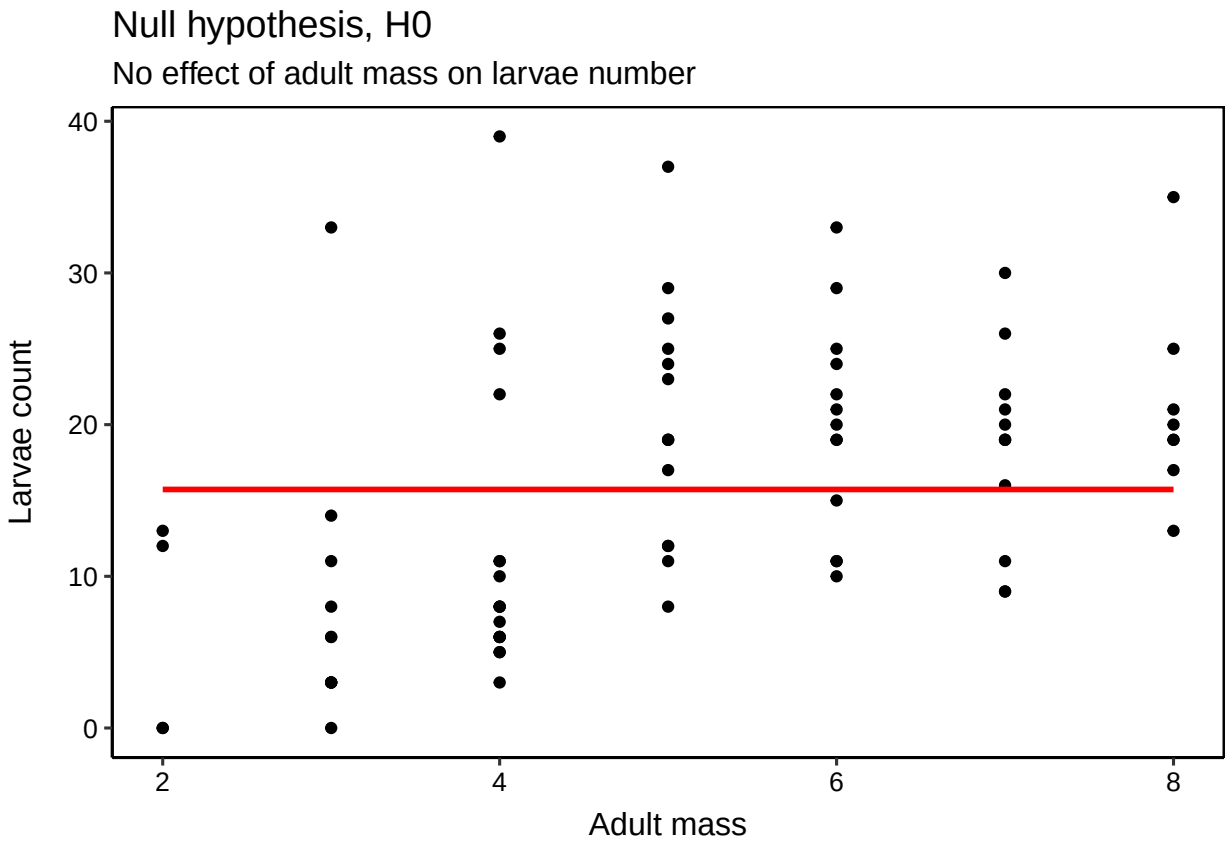
```
##
## Call:
## glm(formula = larvae ~ 1, family = gaussian, data = in.data)
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   15.725      1.041    15.1   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 86.75886)
##
##     Null deviance: 6853.9  on 79  degrees of freedom
## Residual deviance: 6853.9  on 79  degrees of freedom
## AIC: 587.07
##
## Number of Fisher Scoring iterations: 2
```

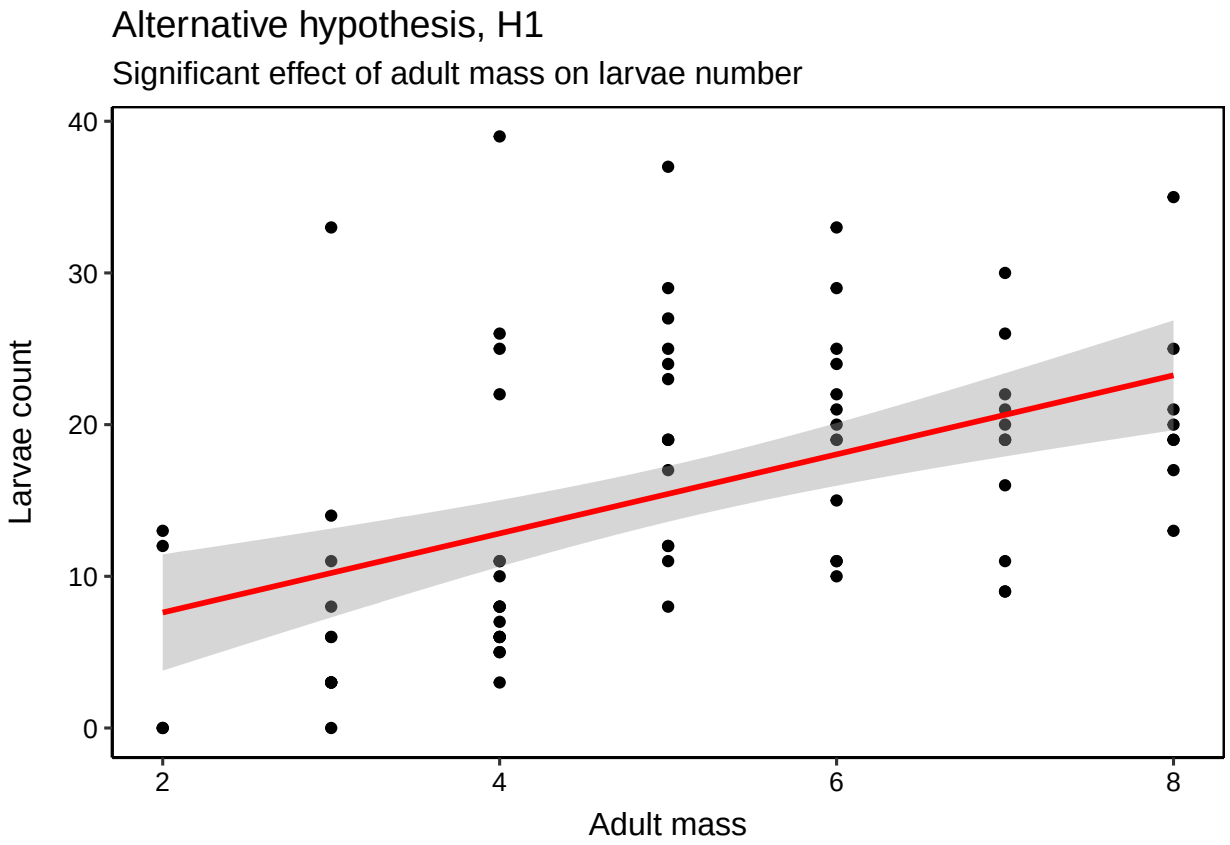
```
summary(H1.model)
```

```
##
## Call:
## glm(formula = larvae ~ 1 + adult_mass, family = gaussian, data = in.data)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3990     2.9254   0.820    0.415
## adult_mass    2.6066     0.5431   4.799 7.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 67.83828)
##
##     Null deviance: 6853.9  on 79  degrees of freedom
## Residual deviance: 5291.4  on 78  degrees of freedom
## AIC: 568.37
##
## Number of Fisher Scoring iterations: 2
```

```
# let's quickly plot this

# H0
ggplot(in.data, aes(x = adult_mass, y = larvae)) +
  geom_point() +
  geom_smooth(method = "lm", formula = y ~ 1, se = FALSE, color = "red") +
  labs(x = "Adult mass", y = "Larvae count") +
  ggtitle("Null hypothesis, H0", subtitle = "No effect of adult mass on larvae number")
```




```
# H1
ggplot(in.data, aes(x = adult_mass, y = larvae)) +
  geom_point() +
  geom_smooth(method = "lm", se = TRUE, color = "red") +
  labs(x = "Adult mass", y = "Larvae count") +
  ggtitle("Alternative hypothesis, H1", subtitle = "Significant effect of adult mass on larvae number")
```




```
# Perform a Likelihood Ratio Test (LRT) to assess goodness of fit
lmtest::lrtest(H0.model, H1.model)
```

```
## Likelihood ratio test
##
## Model 1: larvae ~ 1
## Model 2: larvae ~ 1 + adult_mass
##   #Df  LogLik Df Chisq Pr(>Chisq)
## 1   2 -291.54
## 2   3 -281.19  1  20.7  5.373e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

These results indicate that there is a significant effect of adult mass on larval output ($\chi 2 = 20.7$, df = 1, p < 0.001), and that the second model, H1 (alternative), is significantly better than the null hypothesis. The log likelihood is higher for H1 (-281.19), which signifies that it is the better model.

Like we did earlier, here's another way of making predictions and plotting the model:

```
# Extract expected relationship between X and Y
 preds.larvae <- ggeffects::ggeffect(
 model = H1.model,
 terms = c("adult_mass [0:15 by = 2]"),
 type = "fixed",
 interval = "confidence"
```

```
 ) %>%
 # Convert predictions into a data.frame
 as.data.frame() %>%
 # Rename columns for easier plotting
 dplyr::mutate(
 adult_mass = x
 )

# another way of plotting the model
ggplot2::ggplot(data = in.data, aes(x = adult_mass, y = larvae)) +
  geom_point() +
  geom_smooth(method = "glm", method.args = list(family = "gaussian"),
              se = TRUE, color = "black", fill = "lightgrey")
```