Untitled

William Yip

2023-05-24

Contents

```
# should haves (from last week)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.1     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(here)
```

```
## here() starts at /Users/williamyip/github/ENVS-193DS_homework-05
```

```
library(janitor)
```

```
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(ggeffects)
library(performance)
library(naniar) # or equivalent
library(flextable) # or equivalent
```

```
##
## Attaching package: 'flextable'
##
## The following object is masked from 'package:purrr':
##
##     compose
```

```
library(car)
```

```
## Loading required package: carData
##
## Attaching package: 'car'
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
## The following object is masked from 'package:purrr':
##
##     some
```

```
library(broom)
# would be nice to have
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(AICcmodavg)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
here("data", "knb-lter-hfr.109.18")
```

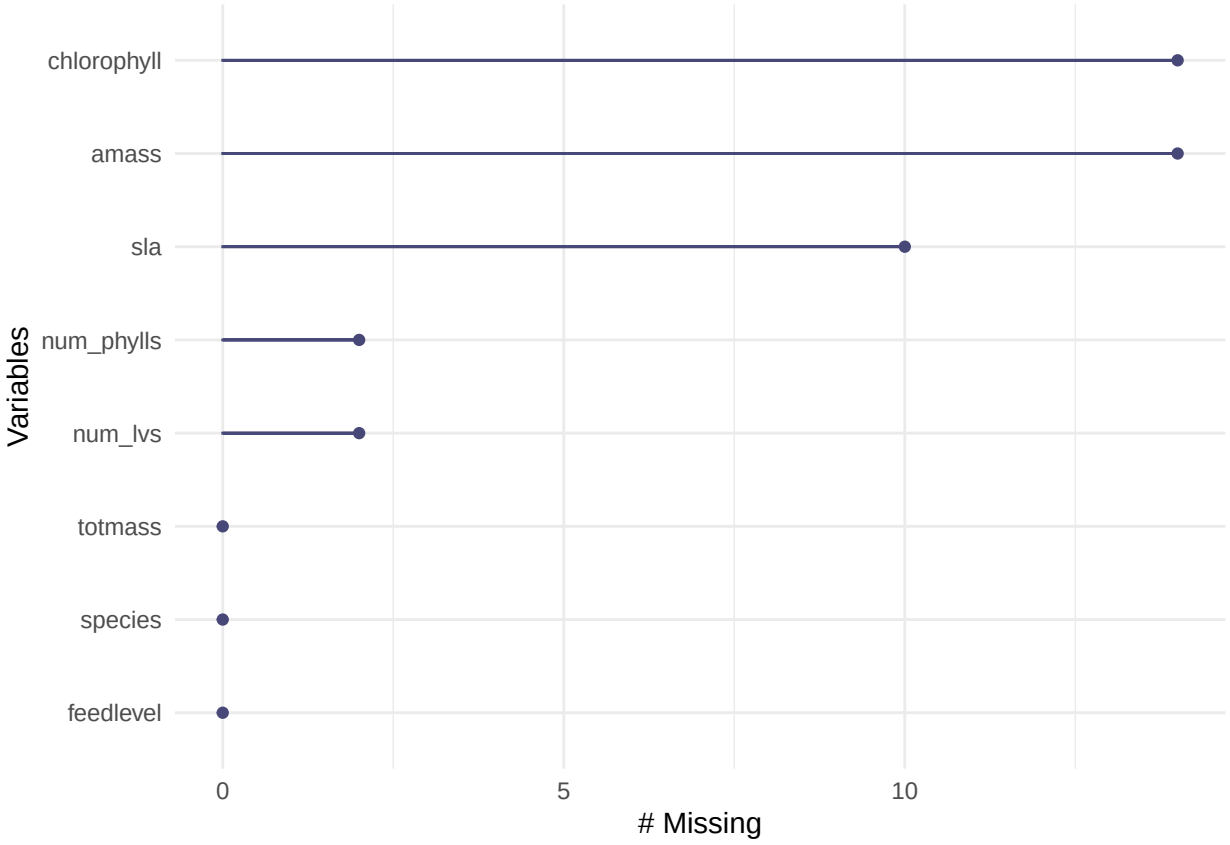
```
## [1] "/Users/williamyip/github/ENVS-193DS_homework-05/data/knb-lter-hfr.109.18"
```

```
plant <- read_csv("~/github/ENVS-193DS_homework-05/data/knb-lter-hfr.109.18/hf109-01-sarracenia.csv") %>
  #make column names cleaner
  clean_names() %>%
  #selecting columns of interest
  select(totmass, species, feedlevel, sla, chlorophyll, amass, num_lvs, num_phylls)
```

```
## Rows: 120 Columns: 32
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): species
## dbl (31): feedlevel, plant.num, fv.fm.lf1, fv.fm.lf2, totmass, rt.sht, mass....
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Visualizing the missing data:

```
gg_miss_var(plant)
```




```
#missing observations will be excluded
```

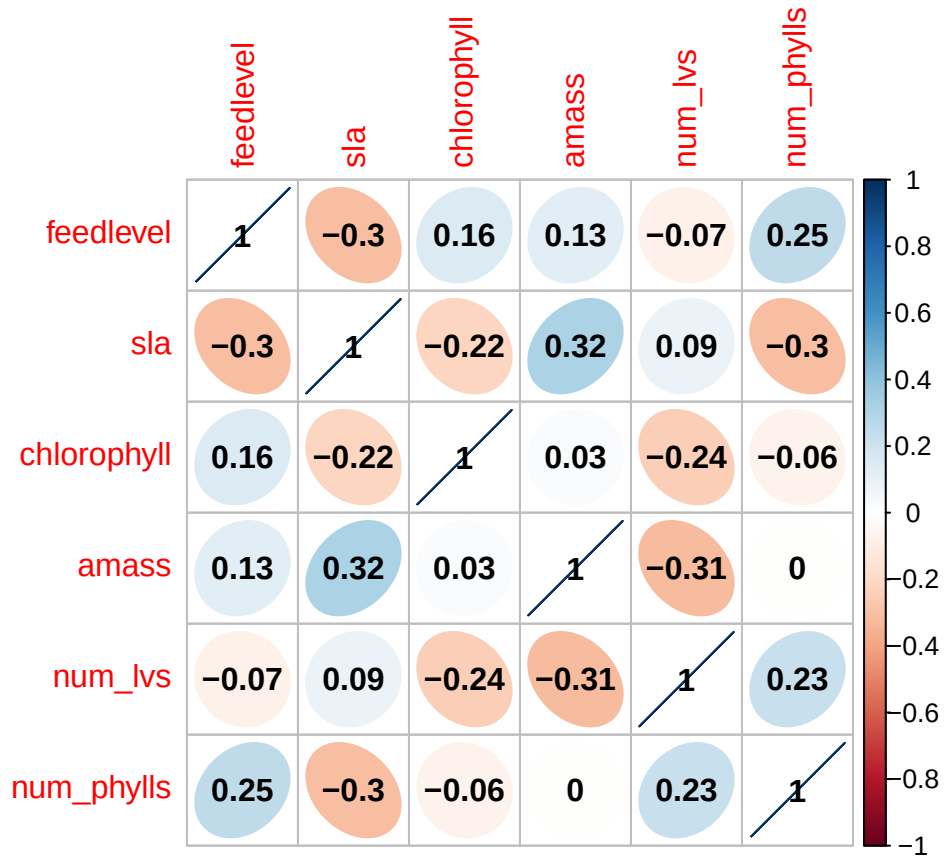
subsetting the data by dropping NAs:

```
plant_subset <- plant %>%
  drop_na(sla, chlorophyll, num_lvs, num_phylls, amass)
```

Create a correlation plot:

Example writing: To determine the relationships between numerical values in our dataset, we calculated Pearsons r and visually represented correlation using a correlation plot.

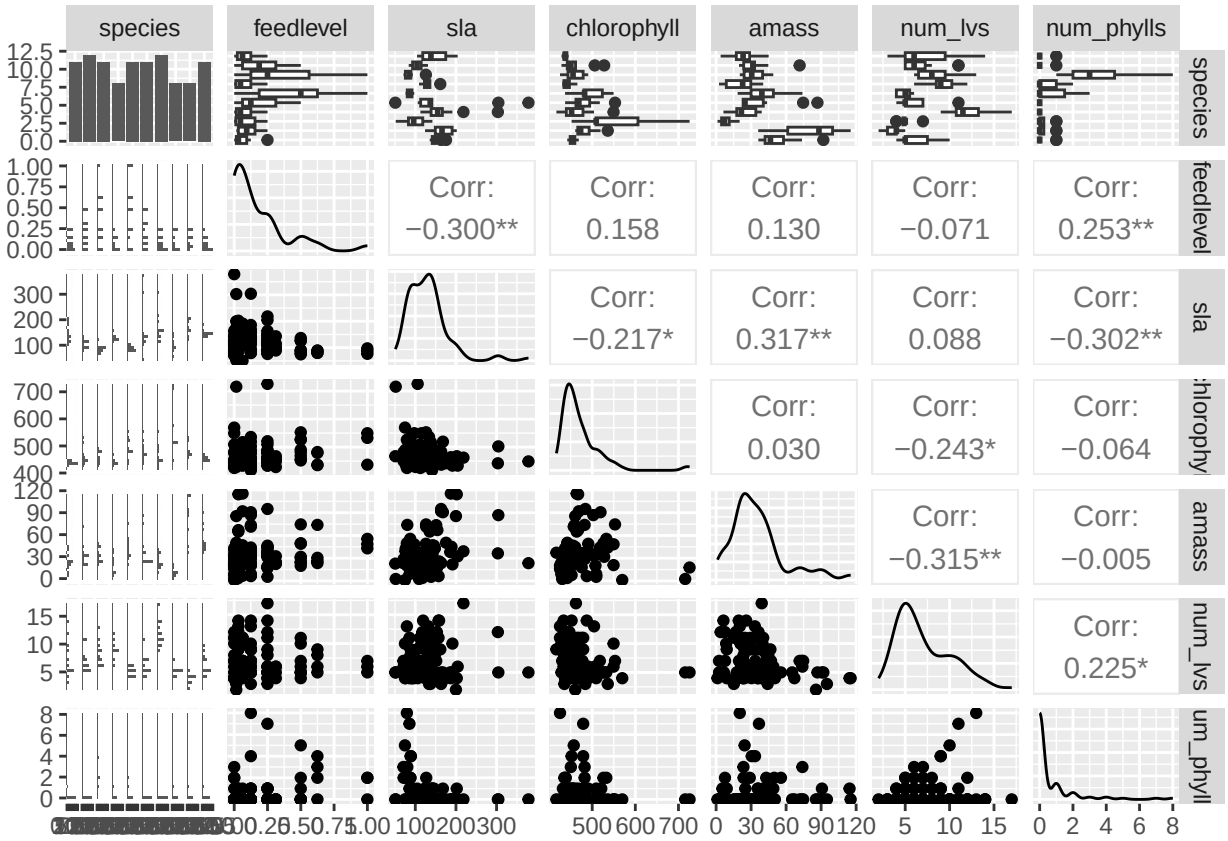
```
plant_cor <- plant_subset %>%
  select(feedlevel:num_phylls) %>%
  cor(method = "pearson")

#create correlation plot
corrplot(plant_cor,
         method = "ellipse",
         addCoef.col = "black"
         )
```



Create a plot of each variable compared against the others

```
plant_subset %>%
  select(species:num_phylls) %>%
  ggpairs()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



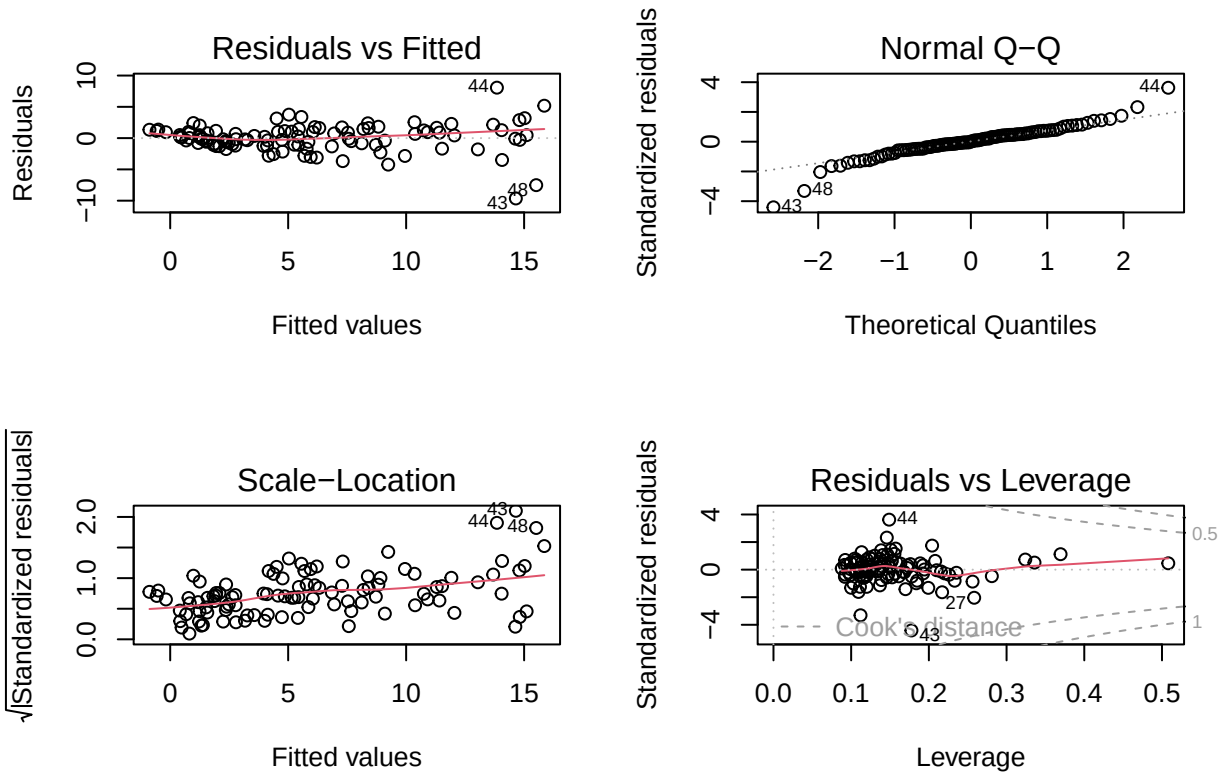
Starting regression here:

Example: To determine how species and physiological characteristics predict biomass, we fit multiple linear models.

```
null <- lm(totmass ~ 1, data = plant_subset) #start with nothing in there
full <- lm(totmass ~ species + feedlevel + sla + chlorophyll + amass + num_lvs + num_phylls, data = plan
```

We visually assess normality and homeoelasticity of residuals using diagnostic plot for the full model

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



We also tested for normality and heteroscedasticity using the Shapiro-Wilk test (null hypothesis: variable of interest (i.e the residuals) are normally distributed).

We tested for heteroskedaslicity using the Breush-Pagan test (null hypothesis: variable of interst has constant variance)

```
check_normality(full)
```

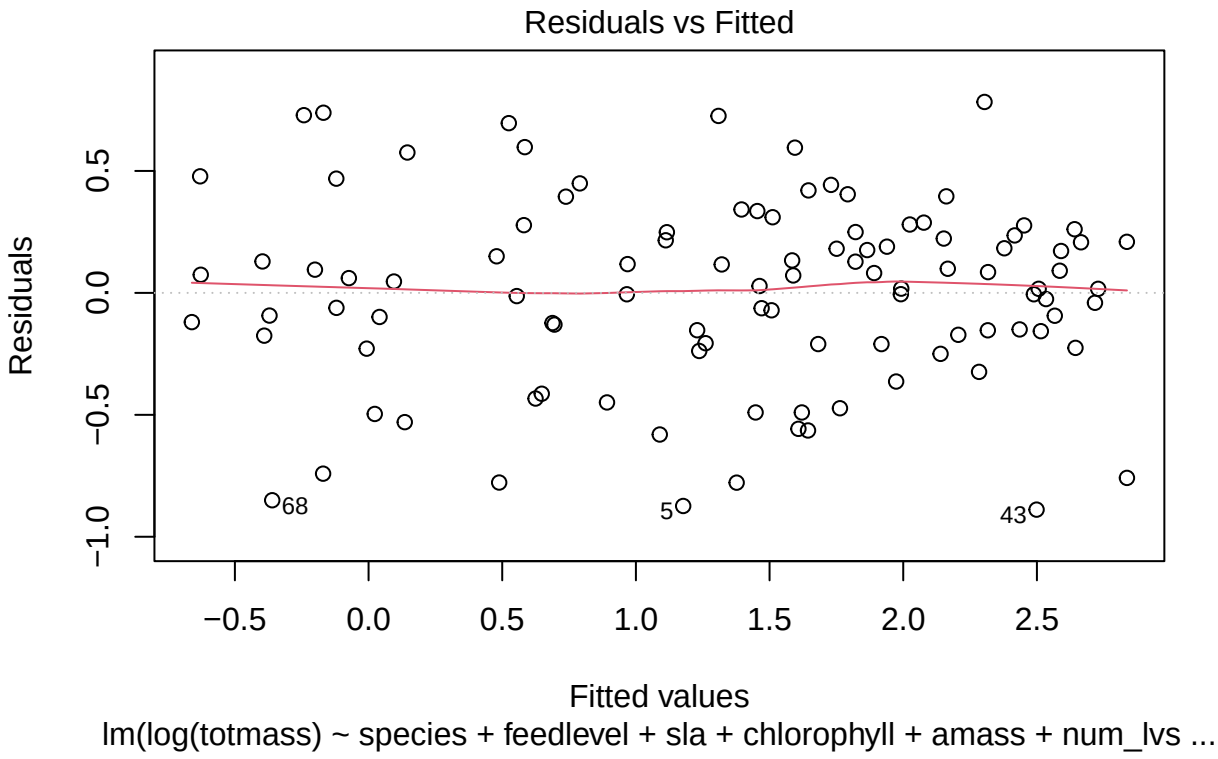
```
## Warning: Non-normality of residuals detected (p < .001).
```

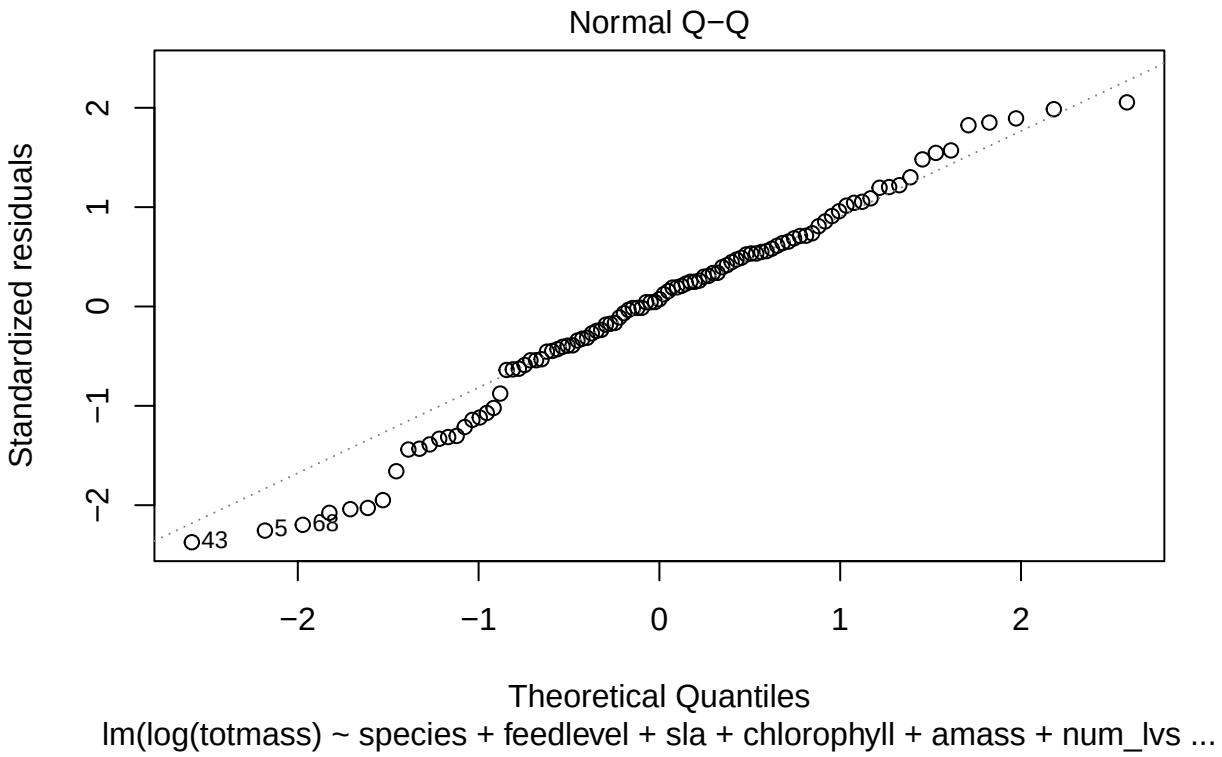
```
check_heteroscedasticity(full)
```

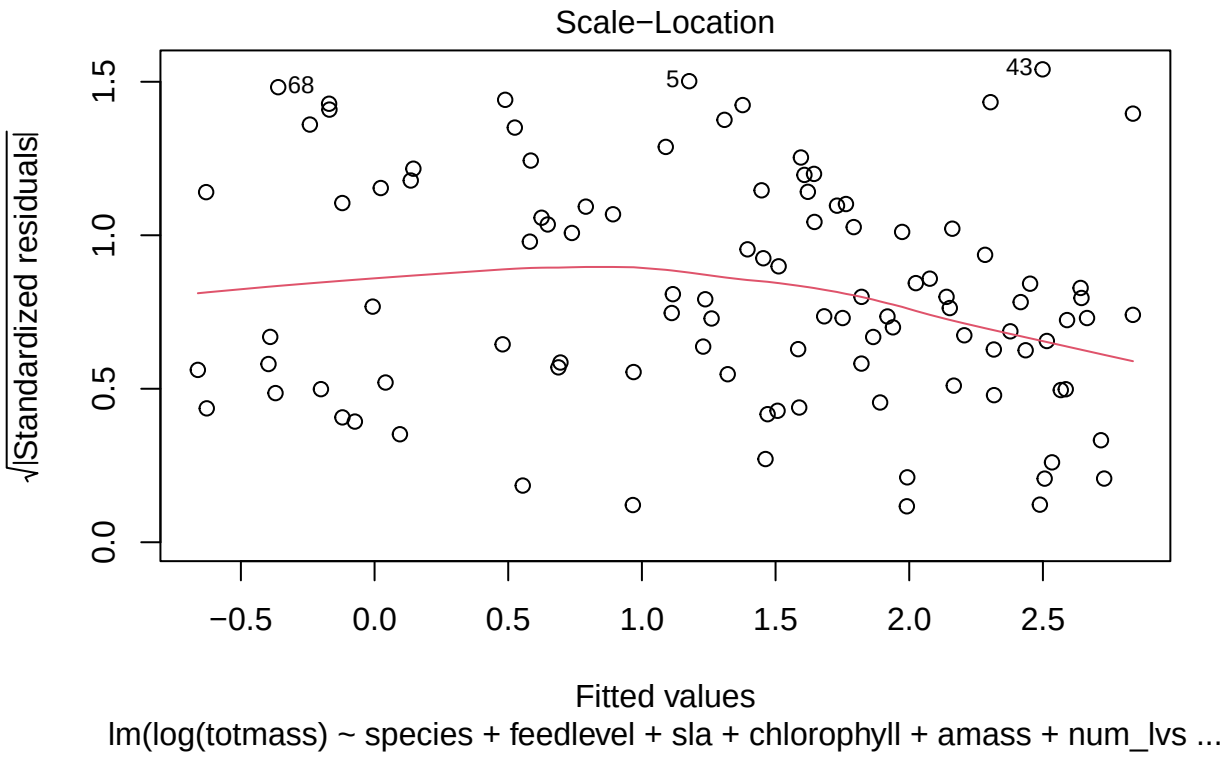
```
## Warning: Heteroscedasticity (non-constant error variance) detected (p < .001).
```

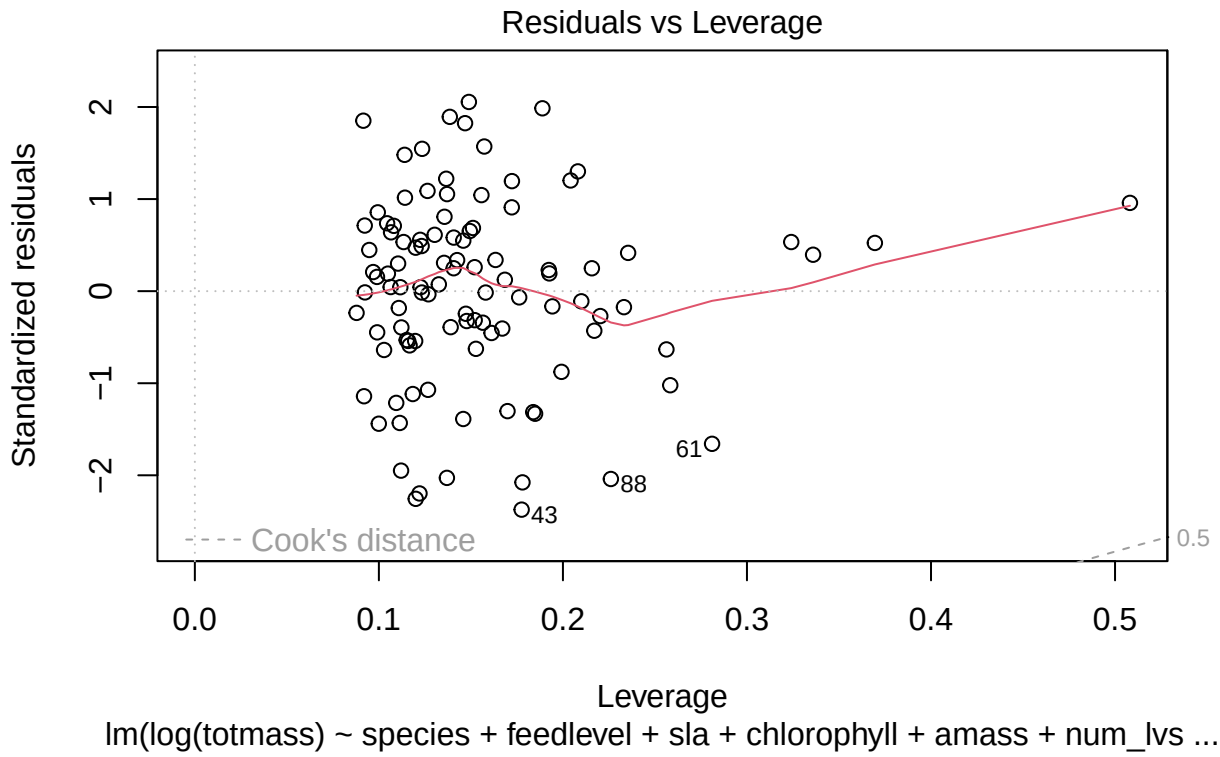
This dataset does not conform to the assumptions of linear regression (normal for bio datasets) Use a log10 of each observation to transform the resposne variable to transform residuals to normal

```
full_log <- lm(log(totmass) ~ species + feedlevel + sla + chlorophyll + amass + num_lvs + num_phylls, da

plot(full_log)
```


Residuals vs Fitted
Residuals
0.5
0.0
−0.5
−1.0
68
5
43
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
Fitted values
lm(log(totmass) ~ species + feedlevel + sla + chlorophyll + amass + num_lvs ...


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(log(totmass) ~ species + feedlevel + sla + chlorophyll + amass + num_lvs ...
43
5
68
−2
−1
0
1
2


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
68
5
43
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
Fitted values
lm(log(totmass) ~ species + feedlevel + sla + chlorophyll + amass + num_lvs ...




```
check_normality(full_log)
```

```
## OK: residuals appear as normally distributed (p = 0.107).
```

```
check_heteroskedasticity(full_log)
```

```
## OK: Error variance appears to be homoscedastic (p = 0.071).
```