ES 193DS Homework 3

Ella Stookey

2024-06-02

Preparations

Reading in packages

```r
# general use
library(tidyverse)
```

```
-- Attaching core tidyverse packages ---------------------- tidyverse 2.0.0 --
v dplyr     1.1.4      v readr     2.1.5
v forcats   1.0.0      v stringr   1.5.1
v ggplot2   3.5.0      v tibble    3.2.1
v lubridate 1.9.3      v tidyr     1.3.1
v purrr     1.0.2
-- 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 becon
```

```r
library(readxl)
library(here)
```

```
here() starts at /Users/ellastookey/Desktop/ENVS-193DS/git/stookey-ella_homework-03
```

```r
library(janitor)
```

1

```
Attaching package: 'janitor'

The following objects are masked from 'package:stats':

    chisq.test, fisher.test
```

```r
# visualizing pairs
library(GGally)
```

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

```r
# model selection
library(MuMIn)

# model predictions
library(ggeffects)

# model tables
library(gtsummary)
library(flextable)
```

```
Attaching package: 'flextable'

The following objects are masked from 'package:gtsummary':

    as_flextable, continuous_summary

The following object is masked from 'package:purrr':

    compose
```

```r
library(modelsummary)
```

```
`modelsummary` 2.0.0 now uses `tinytable` as its default table-drawing
  backend. Learn more at: https://vincentarelbundock.github.io/tinytable/
```

Revert to `kableExtra` for one session:

```
options(modelsummary_factory_default = 'kableExtra')
options(modelsummary_factory_latex = 'kableExtra')
options(modelsummary_factory_html = 'kableExtra')
```

Silence this message forever:

```
config_modelsummary(startup_message = FALSE)
```

```r
drought_exp <- read_xlsx(path = here("data",
                                     "Valliere_etal_EcoApps_Data.xlsx"),
                         sheet = "First Harvest")

# quick look at data
str(drought_exp)
```

```
tibble [70 x 13] (S3: tbl_df/tbl/data.frame)
 $ Species             : chr [1:70] "ENCCAL" "ENCCAL" "ENCCAL" "ENCCAL" ...
 $ Water               : chr [1:70] "WW" "WW" "WW" "WW" ...
 $ Rep #               : num [1:70] 1 2 3 4 5 1 2 3 4 5 ...
 $ Height (cm)         : num [1:70] 5.8 4.9 8.4 6.5 7.1 3.2 4.4 4.2 4.5 3.9 ...
 $ Leaf #              : num [1:70] 11 8 11 12 10 7 7 10 8 6 ...
 $ Leaf dry weight (g) : num [1:70] 0.0294 0.0185 0.0177 0.0178 0.0164 0.017 0.0193 0.0153 0.0...
 $ Leaf area (cm2)     : num [1:70] 5.01 3.98 3.69 3.84 3.63 3.06 3.1 2.94 2.73 2.61 ...
 $ SLA                 : num [1:70] 170 215 209 216 222 ...
 $ Total LA            : num [1:70] 55.1 31.8 40.6 46.1 36.3 ...
 $ Shoot (g)           : num [1:70] 0.253 0.164 0.241 0.213 0.232 ...
 $ Root (g)            : num [1:70] 0.202 0.165 0.209 0.146 0.12 ...
 $ Total (g)           : num [1:70] 0.455 0.329 0.45 0.359 0.352 ...
 $ R:S                 : num [1:70] 0.8 1 0.9 0.7 0.5 0.8 1.2 3.1 0.9 1.2 ...
```

```r
class(drought_exp)
```

```
[1] "tbl_df"     "tbl"        "data.frame"
```

Cleaning

```
# cleaning
drought_exp_clean <- drought_exp %>%
  clean_names() %>% # nicer column names
  mutate(species_name = case_when( # adding column with species scientific names
    species == "ENCCAL" ~ "Encelia californica", # bush sunflower
    species == "ESCCAL" ~ "Eschscholzia californica", # California poppy
    species == "PENCEN" ~ "Penstemon centranthifolius", # Scarlet bugler
    species == "GRICAM" ~ "Grindelia camporum", # great valley gumweed
    species == "SALLEU" ~ "Salvia leucophylla", # Purple sage
    species == "STIPUL" ~ "Nasella pulchra", # Purple needlegrass
    species == "LOTSCO" ~ "Acmispon glaber" # deerweed
  )) %>%
  relocate(species_name, .after = species) %>% # moving species_name column after species
  mutate(water_treatment = case_when( # adding column with full treatment names
    water == "WW" ~ "Well watered",
    water == "DS" ~ "Drought stressed"
  )) %>%
  relocate(water_treatment, .after = water) # moving water_treatment column after water
```

correlations
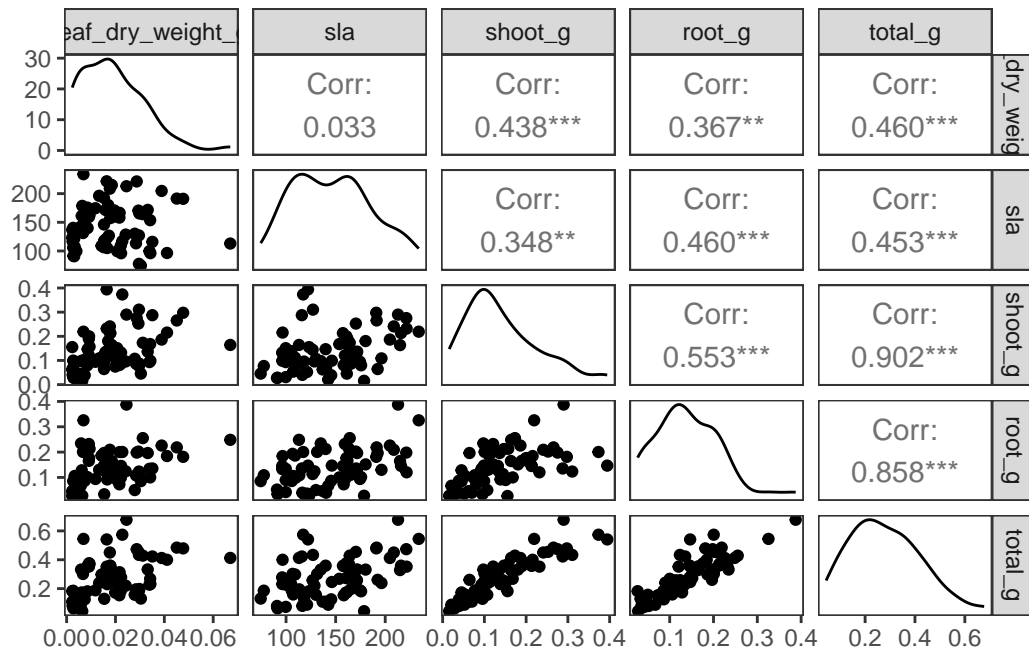
```
ggpairs(drought_exp_clean, # data frame
        columns = c("leaf_dry_weight_g", # columns to visualize
                    "sla",
                    "shoot_g",
                    "root_g",
                    "total_g"),
        upper = list(method = "pearson")) + # calculating Pearson correlation coefficient
  theme_bw() + # cleaner theme
  theme(panel.grid = element_blank()) # getting rid of gridlines
```

```
# bottom left scatterplots of listed variables -- Leaf dry weight on x axis, y axis is total
# diagnol shows ?
# upper right shows Pearson's correlation -- positively correlated
```

Problem 1. Multiple linear regression: model selection and construction
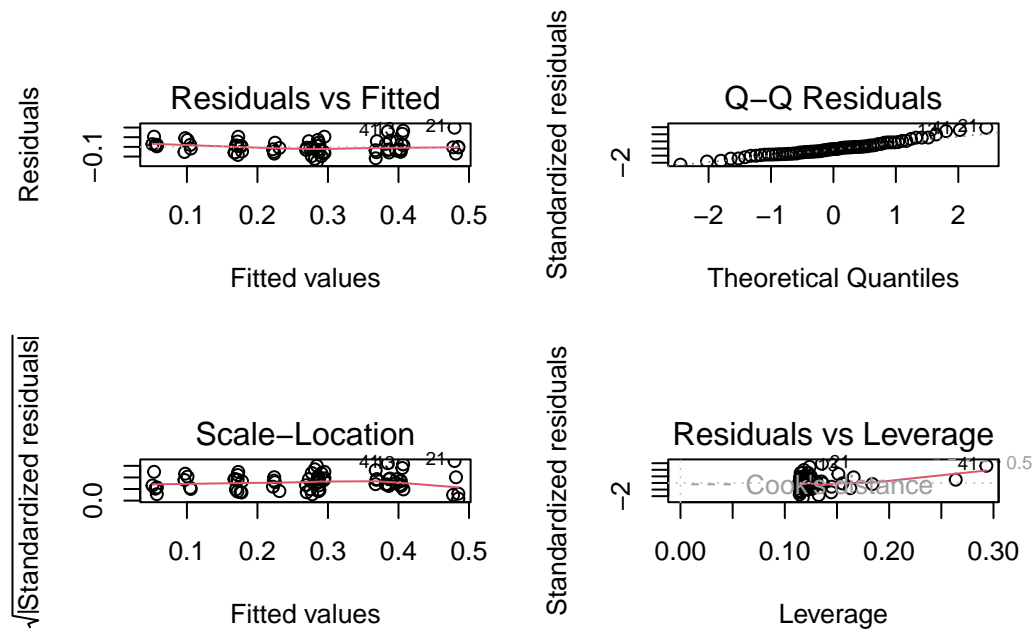
Part a

0. Null model

```
model0 <- lm(total_g ~ 1, # formula
             data = drought_exp_clean) # data frame
```

1. total biomass as a function of SLA, water treatment, and species

```
# saturated model
model1 <- lm(total_g ~ sla + water_treatment + species_name,
             data = drought_exp_clean)

par(mfrow = c(2, 2))
plot(model1)
```
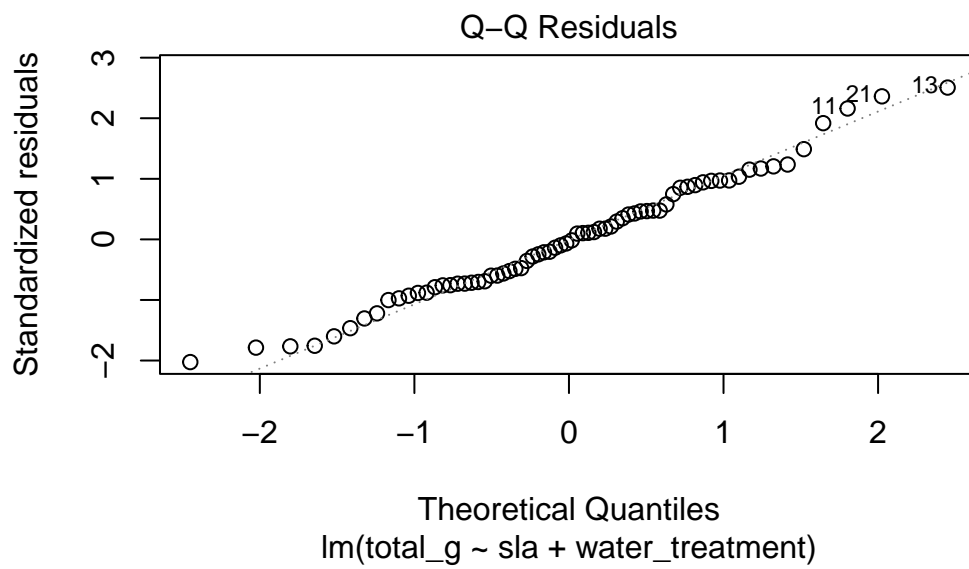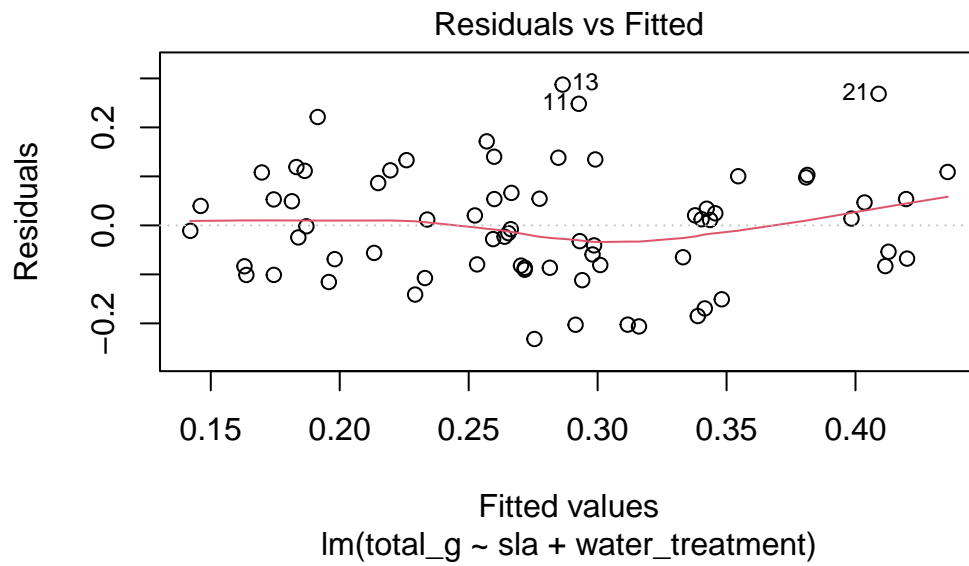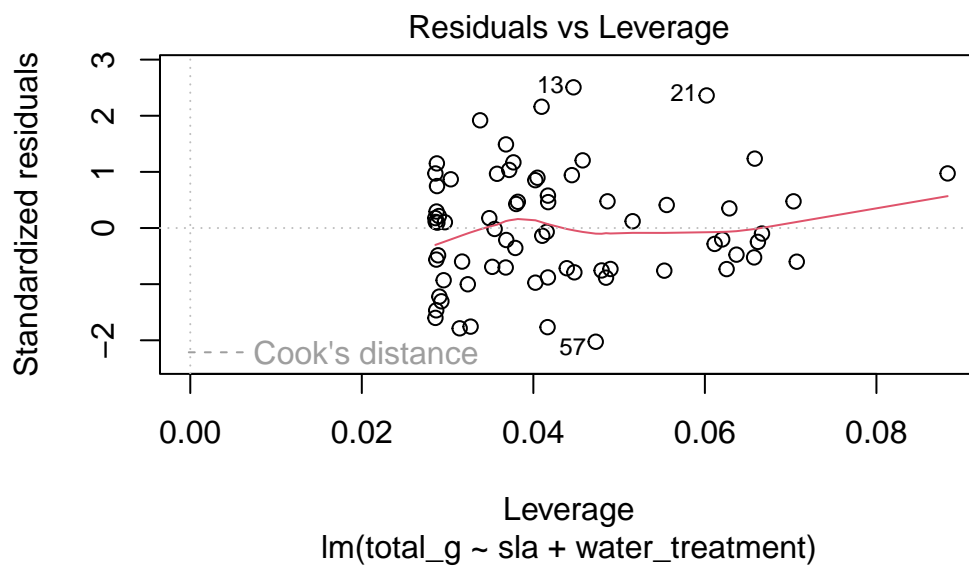

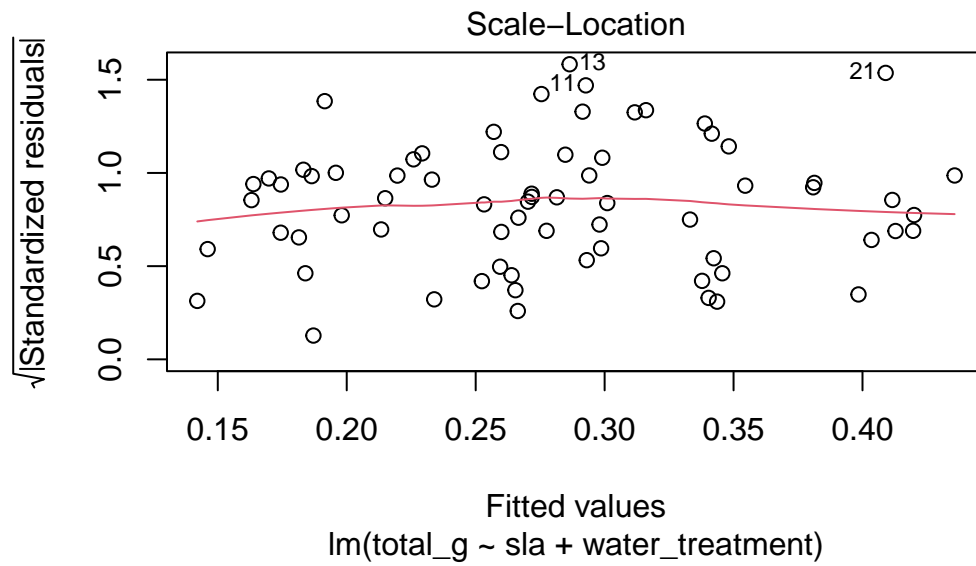**2. total biomass as a function of SLA and water treatment**

```
model2 <- lm(total_g ~ sla + water_treatment,
             data = drought_exp_clean)

plot(model2)
```

Residuals vs Fitted

13
11
21

Residuals

0.15 0.20 0.25 0.30 0.35 0.40

Fitted values
lm(total_g ~ sla + water_treatment)

Q–Q Residuals

11 21 13

Standardized residuals

−2 −1 0 1 2

Theoretical Quantiles
lm(total_g ~ sla + water_treatment)

Scale−Location

√|Standardized residuals|

Fitted values
lm(total_g ~ sla + water_treatment)

Residuals vs Leverage

Standardized residuals

Cook's distance

Leverage
lm(total_g ~ sla + water_treatment)
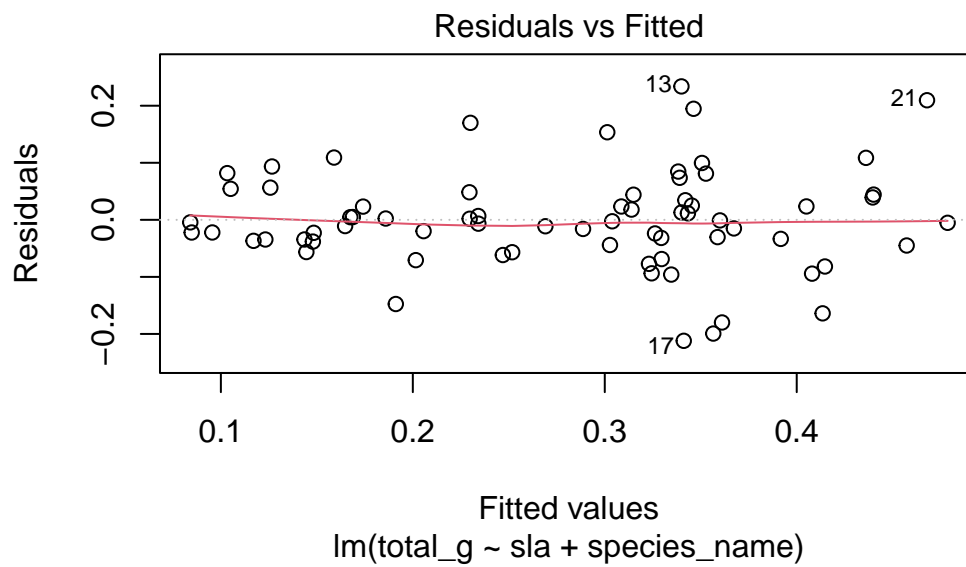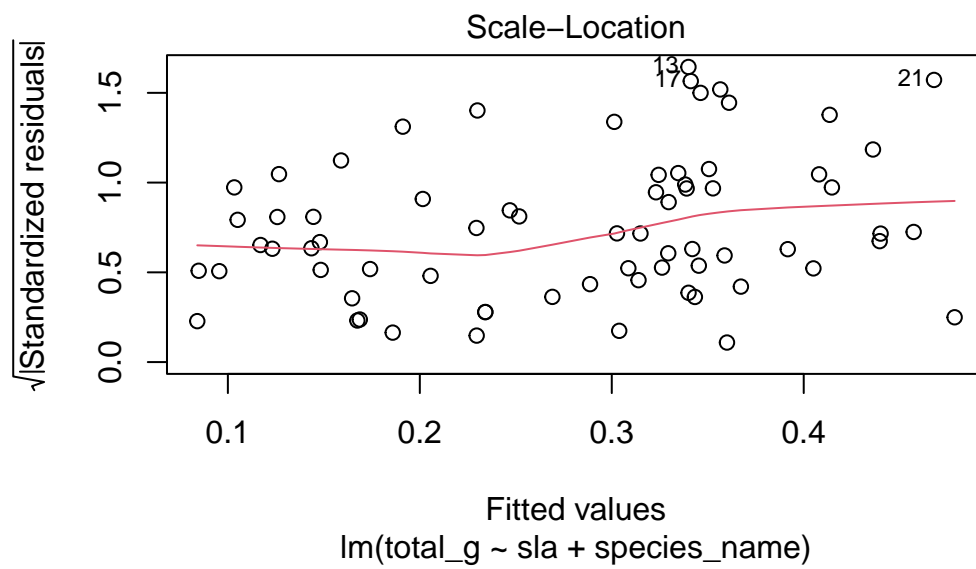
8

3. total biomass as a function of SLA and species

```
model3 <- lm(total_g ~ sla + species_name,
             data = drought_exp_clean)

plot(model3)
```


Residuals vs Fitted — lm(total_g ~ sla + species_name)

Q–Q Residuals

Standardized residuals

13○

21○

○17

Theoretical Quantiles
lm(total_g ~ sla + species_name)

Scale−Location

√|Standardized residuals|

13○
17○

21○

Fitted values
lm(total_g ~ sla + species_name)

10

Residuals vs Leverage

lm(total_g ~ sla + species_name)
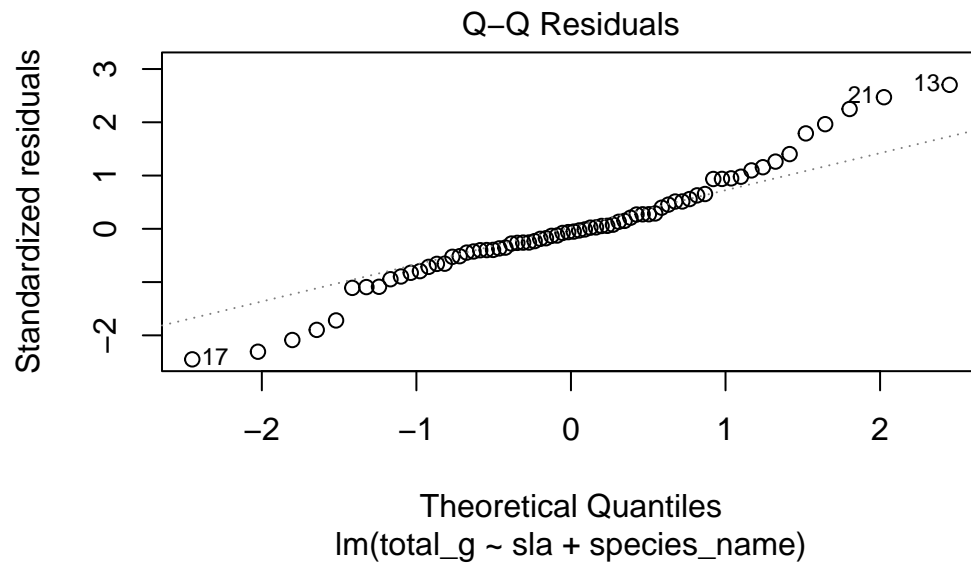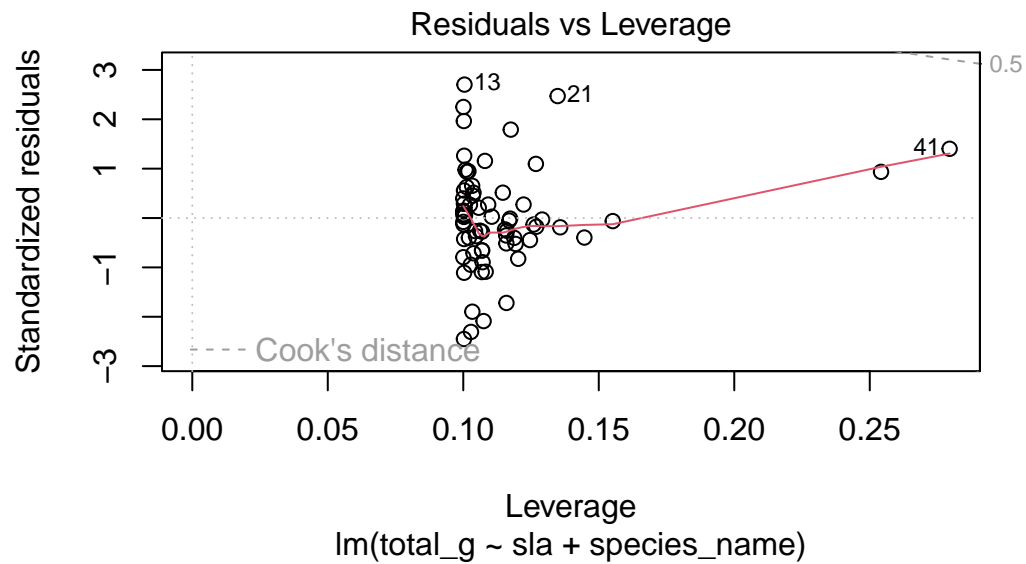
4. total biomass as a function of water treatment and species

```r
model4 <- lm(total_g ~ water_treatment + species_name,
             data = drought_exp_clean)

plot(model4)
```
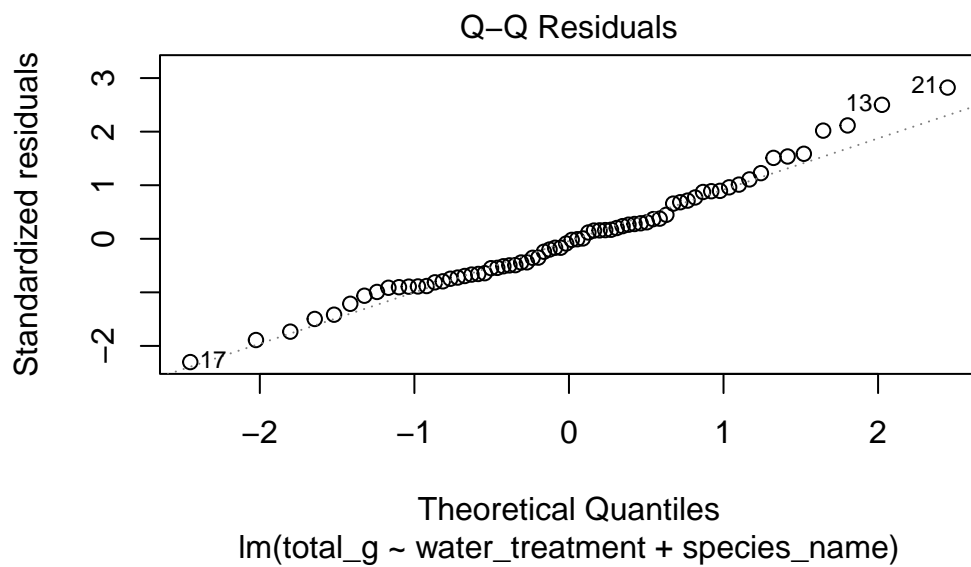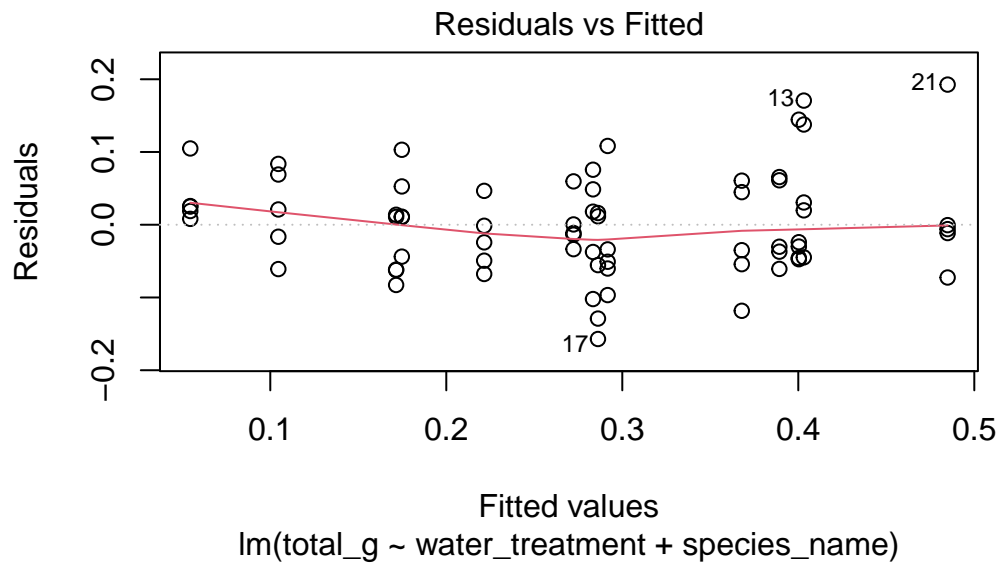
Residuals vs Fitted

Fitted values
lm(total_g ~ water_treatment + species_name)

Q−Q Residuals

Theoretical Quantiles
lm(total_g ~ water_treatment + species_name)

12

Scale−Location

√|Standardized residuals|

Fitted values
lm(total_g ~ water_treatment + species_name)

Constant Leverage:
Residuals vs Factor Levels

Standardized residuals

water_treatment :
Drought stressed   Well watered

Factor Level Combinations

```
modelselectiontable <- model.sel(model0,
        model1,
        model2,
        model3,
```

```
        model4)
# delta for the best AIC will always be 0
```

Table presentation

```
# one option for a single model
flextable::as_flextable(modelselectiontable)
```

(Intercept)	sla species_name	water_treatment	df	logLik	AICc
numeric	numeric factor	factor	integer	numeric	numeric
0.1	+	+	9	88.6	-156.2
0.1	-0.0 +	+	10	88.7	-153.8
-0.0	0.0 +		9	72.5	-124.1
0.0	0.0	+	4	52.2	-95.8
0.3			2	39.6	-75.0

n: 5

```
# comparing models
modelsummary::modelsummary( # this function takes a list of models
  list(
    "null" = model0, # "model name" = model object
    "model 1" = model1,
    "model 2" = model2,
    "model 3" = model3,
    "model 4" = model4
  )
)
```

	null	model 1	model 2	model 3	model 4
(Intercept)	0.279	0.080	0.047	−0.033	0.055
	(0.017)	(0.056)	(0.054)	(0.067)	(0.025)
sla		0.000	0.001	0.001	
		(0.000)	(0.000)	(0.001)	
water_treatmentWell watered		0.122	0.090		0.117
		(0.020)	(0.029)		(0.017)
species_nameEncelia californica		0.238		0.115	0.218
		(0.051)		(0.059)	(0.032)
species_nameEschscholzia californica		0.234		0.222	0.232
		(0.033)		(0.041)	(0.032)
species_nameGrindelia camporum		0.330		0.226	0.313
		(0.047)		(0.054)	(0.032)
species_nameNasella pulchra		0.241		0.168	0.229
		(0.040)		(0.048)	(0.032)
species_namePenstemon centranthifolius		0.061		−0.006	0.050
		(0.039)		(0.047)	(0.032)
species_nameSalvia leucophylla		0.117		0.139	0.120
		(0.033)		(0.041)	(0.032)
Num.Obs.	70	70	70	70	70
R2	0.000	0.755	0.303	0.610	0.754
R2 Adj.	0.000	0.722	0.282	0.566	0.726
AIC	−75.2	−157.5	−96.4	−127.1	−159.2
BIC	−70.7	−135.0	−87.4	−106.8	−139.0
Log.Lik.	39.580	88.741	52.220	72.538	88.598
RMSE	0.14	0.07	0.11	0.09	0.07

Part b

Part c

Model predictions

Note: only plot terms in the model you select - if your doesn't include one of these terms, take it out and adjust the plotting code accordingly!

```
model_preds <- ggpredict(model4,
                         terms = c(
                                    "water_treatment",
                                    "species_name"))

# use View(model_preds) to see the predictions as a data frame
# use model_preds to see the predictions formatted nicely
```

```
# creating new data frame of model predictions for plotting
model_preds_for_plotting <- model_preds %>%
  rename(water_treatment = x, # renaming columns to make this easier to use
         species_name = group)

# use View(model_preds_for_plotting)
# to compare this to the original model_preds data frame

ggplot() +
  # underlying data
  geom_point(data = drought_exp_clean,
             aes(x = water_treatment,
                 y = total_g,
                 color = water_treatment),
             alpha = 0.1) +
  # model prediction 95% CI ribbon
  geom_ribbon(data = model_preds_for_plotting,
              aes(x = water_treatment,
                  y = predicted,
                  ymin = conf.low,
                  ymax = conf.high,
                  fill = water_treatment),
              alpha = 0.2) +
  # model prediction lines
```

```
geom_line(data = model_preds_for_plotting,
          aes(x = water_treatment,
              y = predicted,
              color = water_treatment)) +
# cleaner theme
theme_classic() +
# creating different panels for species
facet_wrap(~species_name)
```

`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
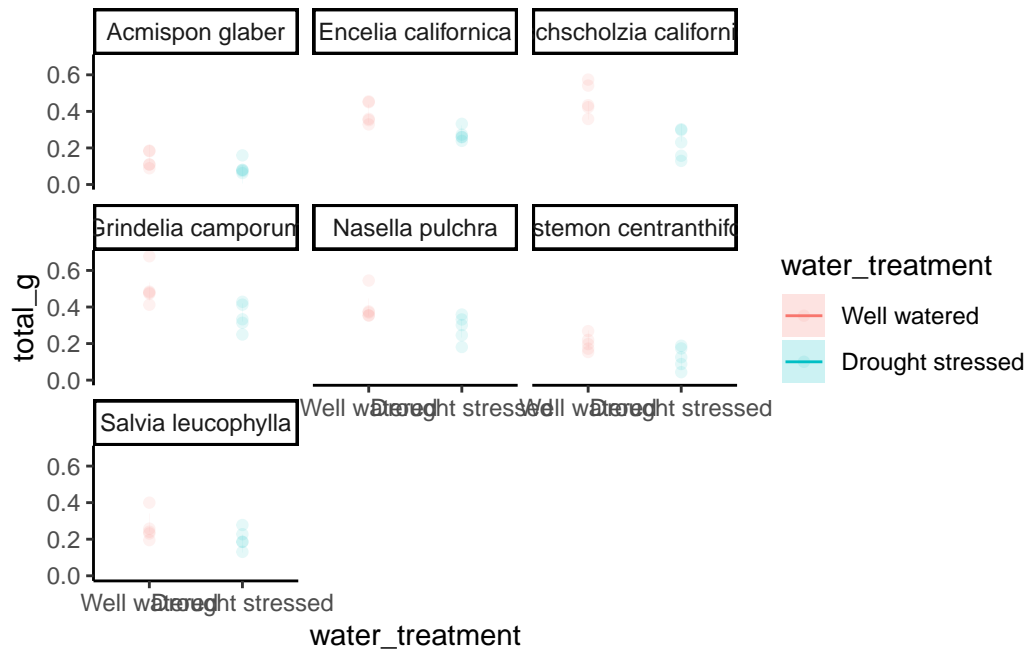
Part d

Part e

Problem 2. Affective visualization

Part a

For my personal data set, where I am examining the distance traveled each day, I could use a bar graph and outline the perimeter of each peak. In doing so, the graph will appear to be "hilly". Since my data is about driving, I will turn this into a scene with a car driving over hills (ie the bar graph).

Part b

Part c

Part d

For my visualization, I have created a scene of a car traveling on a hilly road. The hills represent the distance traveled (in miles) each day, some days peaking while others are flatter.

This work was done on a digital coloring platform called Notability. I began by importing a screenshot of my data (bar graph) and from there I traced and colored it. I finished it off by adding details, such as the road and cars. During this process I was unsure whether to keep each date and numerical value, however ultimately did decide to keep them because I thought it helped the viewer see the information clearer.

Problem 3. Statistical critique

Part a

To examine the long-term effects of a wildfire on soil nutrients and makeup, the researchers used a two-way ANOVA test and if significant differences were found ($p < 0.05$), a Tukey HSD post-hoc test was applied. The authors represented these statistical tests in three tables. Table 1 shows the results of the ANOVA test and Tables 2 & 3 show the descriptive statitics for certain nutrients. In addition to the tables, there were two figures. The first was a map that illustrated the study location and areas with varying fire severity. The second figure was an RDA for the relation between factors 1 and 2.

Part b

All three tables were very clear, with descriptive captions and column and row labels. Figure 1 was also simple to understand because it consisted of images and maps for context. However, figure 2 was significantly more confusing to understand because I have never looked at a redundancy analysis (RDA) plot before. There are no units on the x and y axis and at first glance the numbers seem quiet arbitrary. After researching how to read the plot, it made more sense and I could see how the variables' summary statistics (means and standard deviations) were being shown. No model predictions were in the matrix, but rather just the collected data.

Part c

The tables all hold a lot of information and data making them seem a bit visually cluttered. However, this was crucial information for the researchers to show so it was necessary to include it all. Figure 2, the RDA plot, had a very good data to ink ratio, only consisting of a few lines, two colors, and minimal lettering.

Part d

For the tables, I think they could have been made clearer if titles/ labels were bolded or a larger font. This would help differentiate the organizational aspects from the large amounts of data. Additionally having lines in the table would have helped section off different nutrients' information instead of it all blending together. As for the figure, I wish the caption included more information about what each percent on the sides of the plot and the axes represent. Although information about this is included in the text of the paper, I think including it on the figure is crucial so the reader can get a clear picture of the results from the experiment.