ENVS 193DS Homework 4

Lauren Stiles

link to github here

Problem 1. How does fish length predict fish weight for trout perch (across all sample years)? 1. The null hypothesis is that the predictor variable does not predict the response variable, or that the fish length does not predict fish weight for trout perch. The alternative hypothesis is that the the predictor variable does predict the response variable, or that the fish length does predict fish weight for trout perch.

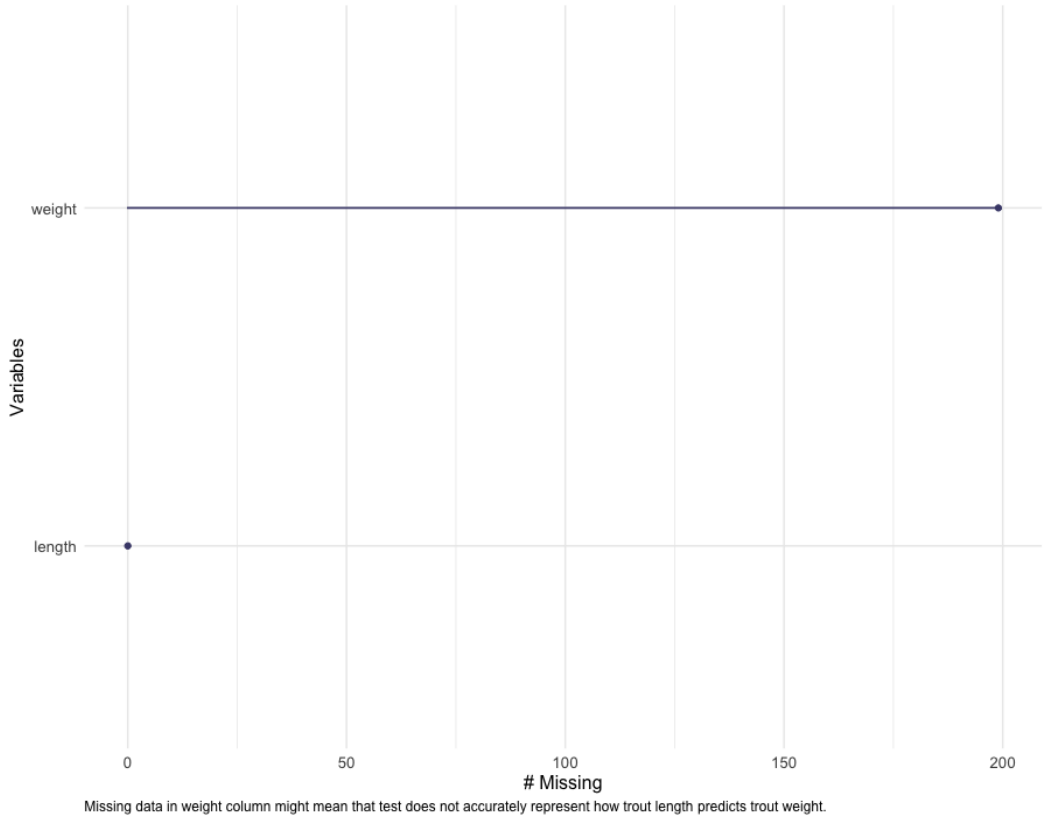
#Read in data and filter for desired variables

```
library(tidyverse)
library(here)
library(naniar)
library(performance)
library(broom) #puts all outputs from model into a table
library(flextable) #whole manual online, allows you to create tables that
render nicely
library(ggeffects) #get predictions from models and plot them...
library(car) #pull out ANOVA tables specifically for linear models



#read in data
fish_dat <- read_csv(here("data/ntl6_v12.csv"))

#filter out trout perch, select length and weight
trout_dat <- fish_dat |>
             mutate_all(tolower)|>
             filter(spname == "troutperch") |>
             dplyr::select(length, weight) |>
             mutate_at(1:2, as.numeric)
```

2. Show missing data

```
#create visualization of missing data
missing_data <- gg_miss_var(trout_dat) +
        #add caption
        labs(caption = "Missing data in weight column might mean that test
does not accurately represent how trout length predicts trout weight.") +
        theme(#change size of plot title
        plot.title = element_text(size = 15),
        #change size and location of plot caption
        plot.caption = element_text(size = 9, hjust = 0),
        #choose margins of plot
        plot.margin = unit(c(1,1,1,1), "cm"),
```

```
        #change axis title size
        axis.title = element_text(size = 12),
        #change axis text size
        axis.text = element_text(size = 10),
        #remove axis ticks
        axis.ticks = element_blank())



missing_data
```



Missing data in weight column might mean that test does not accurately represent how trout length predicts trout weight.

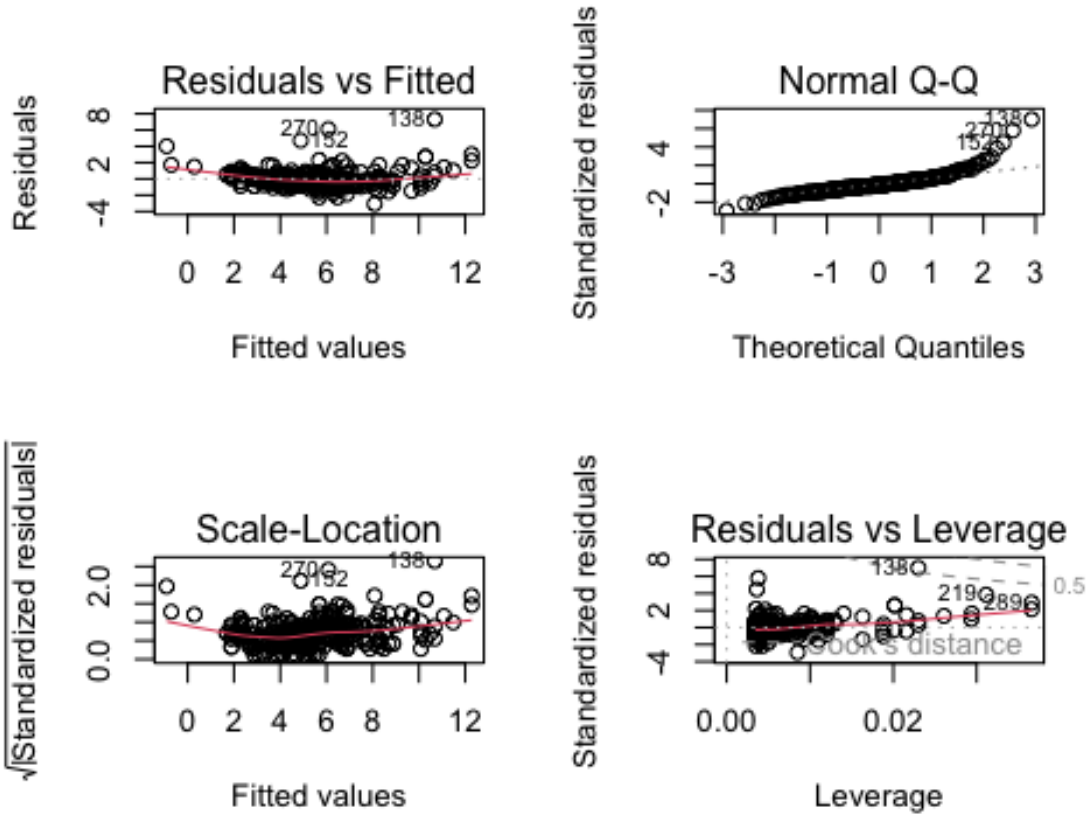
3. Running test

```
#get rid of NAs in weight column to make the model work better...
trout_subset <- trout_dat|>
                drop_na(weight)

#run linear model looking into whether trout length predicts weight
trout_model <- lm(weight ~ length, data = trout_subset)
trout_model


Call:
lm(formula = weight ~ length, data = trout_subset)
```

```
Coefficients:
(Intercept)       length
   -11.7025       0.1999
```

4. Visually check test assumptions

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



5. The residuals versus fitted plot shows a straight line and distribution of residuals, which indicate homoskedasticity, or constant variance of residuals. This plot does not seem to represent homoskedasticity since the residuals are clumped around the line. The normal q-q plot shows whether the residuals are normally distributed. Since the points appear to be in a mostly straight line, I would say that the residuals have a normal distribution, except for some on either end. The scale-location plot also shows homoskedasticity of variance, but using the square root of the standardized residuals. They are in a somewhat straight line but the points are again clustered about it, so they are heteroskedastic.The residuals vs leverage, or the cook's distance plot shows whether outliers are influencing the model estimate. There are some that are labeled as outliers, but only one outside the dotted line range, so it does not appear that there are outliers significantly affecting model predictions.

```
#run a summary of the model object
summary(trout_model)


Call:
lm(formula = weight ~ length, data = trout_subset)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0828 -0.4862 -0.1830  0.4128  7.3191

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -11.702476   0.481564  -24.30   <2e-16 ***
length        0.199852   0.005584   35.79   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.057 on 288 degrees of freedom
Multiple R-squared:  0.8164,    Adjusted R-squared:  0.8158
F-statistic:  1281 on 1 and 288 DF,  p-value: < 2.2e-16

#make table showing ANOVA table summary
trout_model_squares <- anova(trout_model)


trout_model_squares_table <- tidy(trout_model_squares) |>
                      mutate(p.value = case_when(p.value <0.001 ~ "<
0.001")) |>
                      flextable() |>  #easiest way to make this into a table
                      set_header_labels(df = "Degrees of Freedom", sumsq =
"Sum of Squares",                       meansq = "Mean squares", statistic =
"F-statistic")

trout_model_squares_table
```

term	Degrees of Freedom	Sum of Squares	Mean squares	F-statistic	p.value
length	1	1,432.2877	1,432.287687	1,280.844	< 0.001
Residuals	288	322.0525	1.118238		

8. The ANOVA test is built on the same parametric base as linear regression models, so they have similar math and outputs.The ANOVA highlights some of the summary

elements from the model object, giving more details about the test such as sum of squares, mean squares, and the f statistic.

9. We can reject the null hypothesis that trout length does not influence trout weight since the linear regression model summary resulted in a p-value of <0.001 (significance level = 0.05). Based on our observations, we can expect a 0.2g increase in fish weight as fish length increases by each mm, as shown by the linear model summary. The R2 value of 0.8164 indicates that this model does a decent job of approximating the actual data, but it would be better if it was closer to 1(a perfect fit).

10. Prediction Visualization

```
#pulling out predictions
#terms corresponds to whatever the predictor was in the model
predictions <- ggpredict(trout_model, terms = "length")

#plot predictions
plot_predictions <- ggplot(data = trout_dat, aes(length, y = weight)) +
  #first plot the underlying data
  geom_point() +
  #plotting model predictions from the predictions object from ggeffects
  geom_line(data = predictions, aes(x = x, y = predicted), color = "blue",
linewidth = 1) +
  #plot the confidence interval around model estimates
  geom_ribbon(data = predictions, aes(x = x, y = predicted, ymin = conf.low,
ymax = conf.high), alpha = 0.2) +
#do not use geom_smooth because it does not tell you where the model comes
from, what the equation is, standard intervals, ect...
   labs(x = "Length",
        y = "Weight",
        title = "Trout Perch Weight Predicted by Length",
        caption = "Figure 2. Dots show predicted values for fish weight based
on size and the blue line shows the equation from the linear regression run
on to test this prediction.") +
  theme_bw() +
  #change text font
  theme(#change size of plot title
        plot.title = element_text(size = 15),
        #change size and location of plot caption
        plot.caption = element_text(size = 9, hjust = 0),
        #choose margins of plot
        plot.margin = unit(c(1,1,1,1), "cm"),
        #change axis title size
        axis.title = element_text(size = 12),
        #change axis text size
        axis.text = element_text(size = 10),
        #remove axis ticks
        axis.ticks = element_blank())
```

```
plot_predictions
```

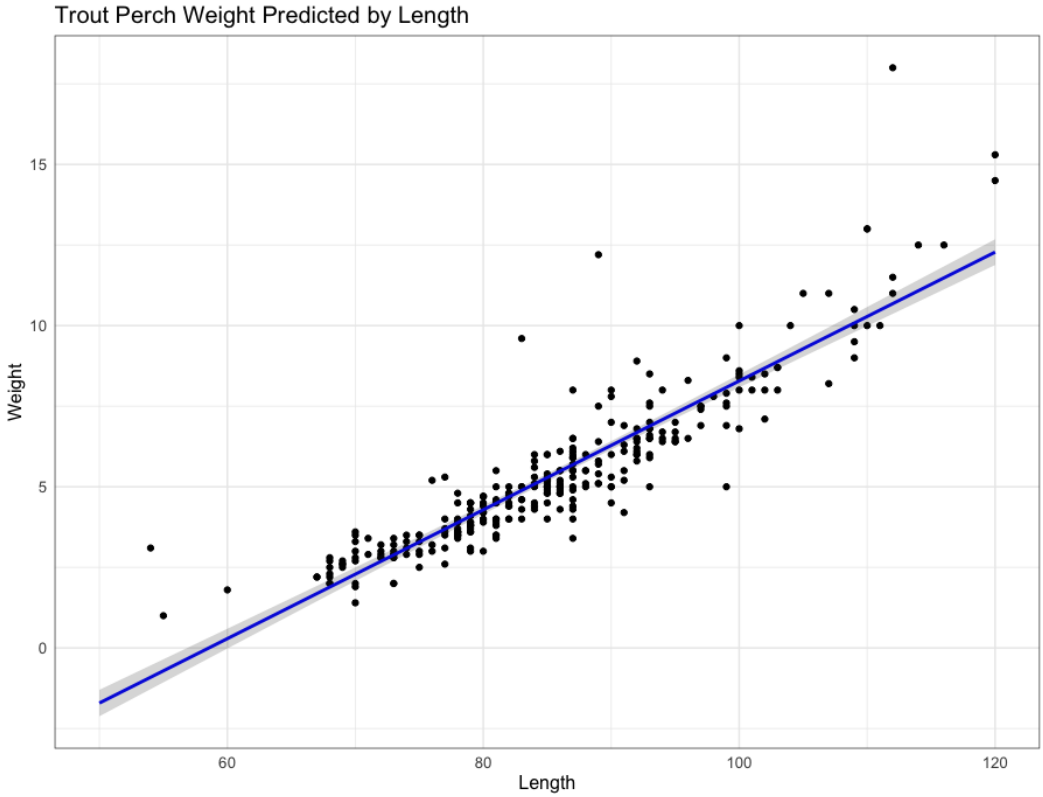


Figure 2. Dots show predicted values for fish weight based on size and the blue line shows the equation from the linear regression run on to test this prediction.