Lab 7: Correlation and Linear Regression

Parsa Ara

Use RMarkdown to answer these questions and upload your answers as HTML or PDF in the Lab 7 turn-in link on GauchoSpace.

Question 1: correlating dandelions

We are going to look at the plant dataset and ask the following question:

Does the number of leaves in a dandelion rosette `num_leaves_in_rosette` *correlate with the diameter of the rosette* `dand_rosette_diam_cm` *?*

Load in the `plant_data.csv` dataset and do the following:

1 A. Clearly state your null and alternative hypotheses for the correlation test of the untransformed data.

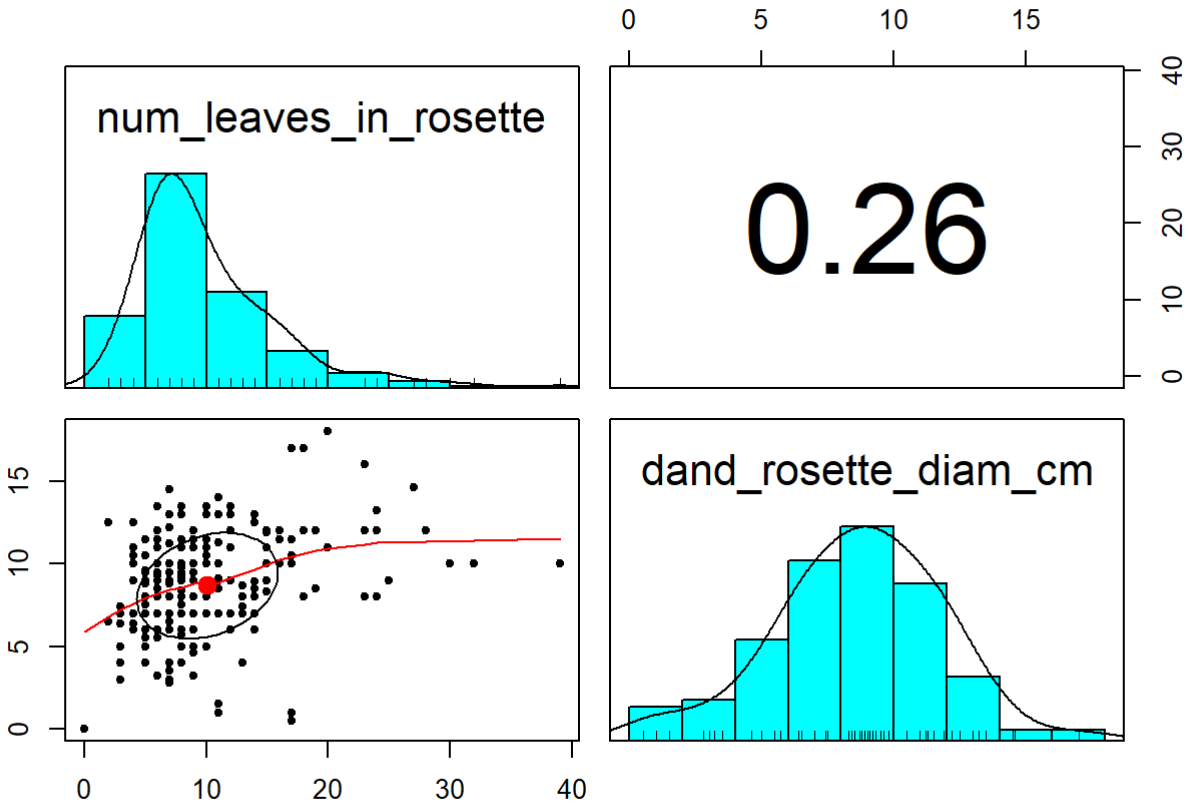
`Null:` There is no correlation between the number of leaves in a dandelion rosette (num_leaves_in_rosette) and the diameter of the rosette (dand_rosette_diam_cm).

`Alternative:` There is a correlation between the number of leaves in a dandelion rosette (num_leaves_in_rosette) and the diameter of the rosette (dand_rosette_diam_cm).

1 B. Use a correlation (scatterplot) matrix of `num_leaves_in_rosette` and `dand_rosette_diam_cm` to assess your assumptions for a parametric correlation test (on the untransformed data) using the `pairs.panels()` function. Do you think your assumption of linearity and bivariate normality are met just based on the figure (i.e. do you need to run any Shapiro-Wilk tests)?

Hint: for your pairs plot, make sure to subset the variables of interest (e.g. `num_leaves_in_rosette` *and* `dand_rosette_diam_cm` *)*

```
# Subset with indexing in base R


# NEWDATAFRAME <- data.frame(DATAFRAME[,VARIABLECOLUMNNUMBER:VARIABLECOLUMNNVARIBLE])

# colnames(NEWDATAFRAME)<-c("COLUMNNAME", "COLUMNNAME")`

# OR subset using subset function

# NEWDATAFRAME <- subset(DATAFRAME, select=c("COLUMNNAME","COLUMNNAME"))
subset_data <- subset(plant_data, select = c("num_leaves_in_rosette", "dand_rosette_diam_c
m"))

# pairs.panels matrix of untransformed data
pairs.panels(subset_data)
```

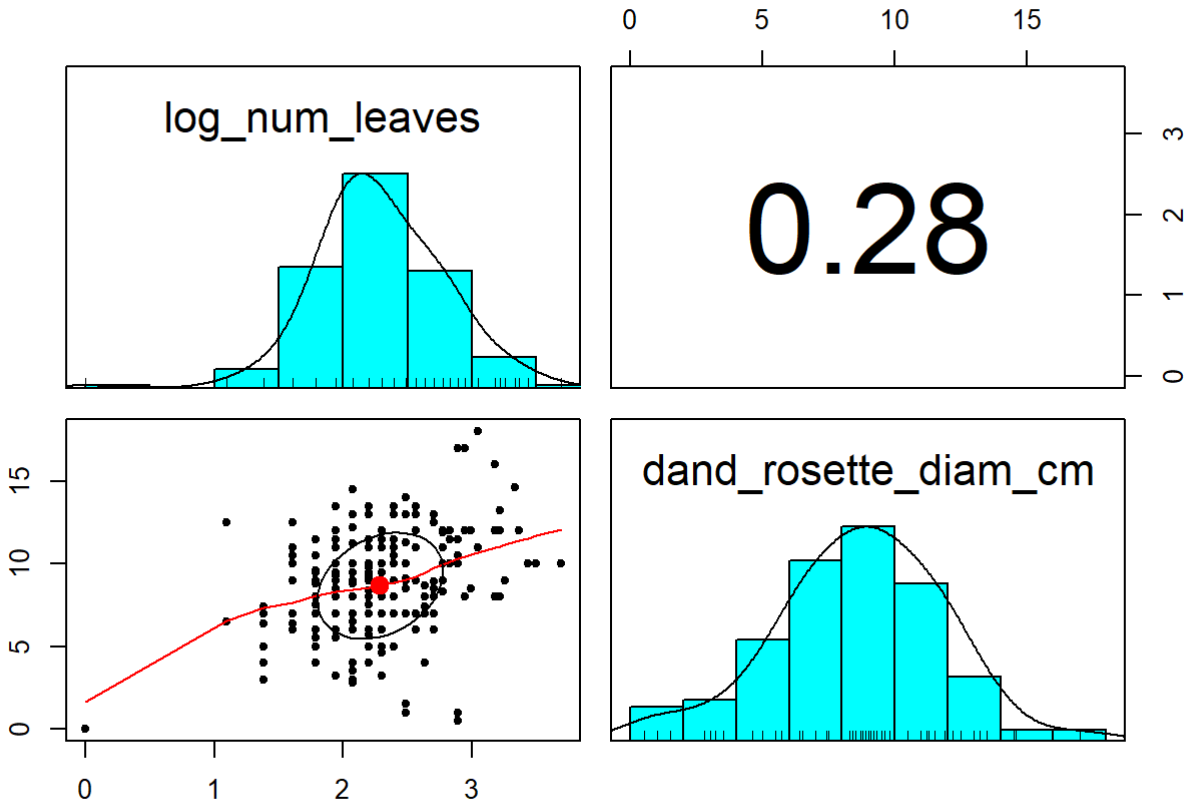


[The assumption of linearity and bivariate normality are not met based on the figure and yes do I need to run any Shapiro-Wilk tests]

1 C. If your assumptions of bivariate normality are not met (i.e. at least one of the variables is not normal), transform whatever variable is not normal so that the assumptions of linearity and bivariate normality are met. Assess these assumptions using a scatterplot matrix with `pairs.panels()` and briefly describe how the plot shows you that the assumptions are now met.

Hint: If you log-transform `num_leaves_in_rosette` be sure to add 1!

```
# transform your data, if needed
subset_data$log_num_leaves <- log(subset_data$num_leaves_in_rosette + 1)
# pairs.panels matrix of transformed data
pairs.panels(subset_data[, c("log_num_leaves", "dand_rosette_diam_cm")])
```



[For linearity: There is a clear linear trend or pattern in the scatterplot. The points roughly follow a linear pattern, it suggests a linear relationship between the variables.

For bivariate normality: The points in the scatterplots form a roughly symmetric pattern around the diagonal line, it suggests bivariate normality.]

1 D. Run a Pearson's correlation test on the **transformed** data.

```
# Pearson's correlation test on transformed data
# Run Pearson's correlation test
cor_result <- cor.test(subset_data$log_num_leaves, subset_data$dand_rosette_diam_cm, method =
"pearson")

# Print the correlation test results
print(cor_result)
```

```
##
##  Pearson's product-moment correlation
##
## data:  subset_data$log_num_leaves and subset_data$dand_rosette_diam_cm
## t = 4.2524, df = 215, p-value = 3.154e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1509495 0.3969917
## sample estimates:
##       cor
## 0.2785343
```

1 E. Run a Spearman's rank correlation test on the **untransformed** data.

```
# Spearman's rank correlation test on the untransformed data
cor_result <- cor.test(subset_data$num_leaves_in_rosette, subset_data$dand_rosette_diam_cm, m
ethod = "spearman")
```

```
## Warning in cor.test.default(subset_data$num_leaves_in_rosette,
## subset_data$dand_rosette_diam_cm, : Cannot compute exact p-value with ties
```

```
# Print the correlation test results
print(cor_result)
```

```
##
##  Spearman's rank correlation rho
##
## data:  subset_data$num_leaves_in_rosette and subset_data$dand_rosette_diam_cm
## S = 1254370, p-value = 8.568e-05
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.2634421
```

1 F. Based on both tests, what do you conclude about the correlation (positive, negative, none) between number of leaves in a dandelion rosette and the diameter of a dandelion rosette?

[Based on both the Pearson's product-moment correlation and Spearman's rank correlation tests, we can conclude that there is a positive correlation between the number of leaves in a dandelion rosette and the diameter of the rosette. The correlation coefficients (correlation estimates) for both tests are positive (0.278 for Pearson's and 0.263 for Spearman's), indicating a positive relationship between the variables.]

Question 2: social spiders

Social spiders live together in kin groups where they build communal webs and cooperate in gathering prey. You gather web measurements on 17 colonies of the social spider *Cyrtophora citricola* in Gabon to determine whether you could **predict** the number of spiders in a colony based on how high the web was off of the ground. Load in the dataset `spiders.csv` and do the following:

2 A. Clearly state your null and alternative hypotheses for a regression analysis of the untransformed data.

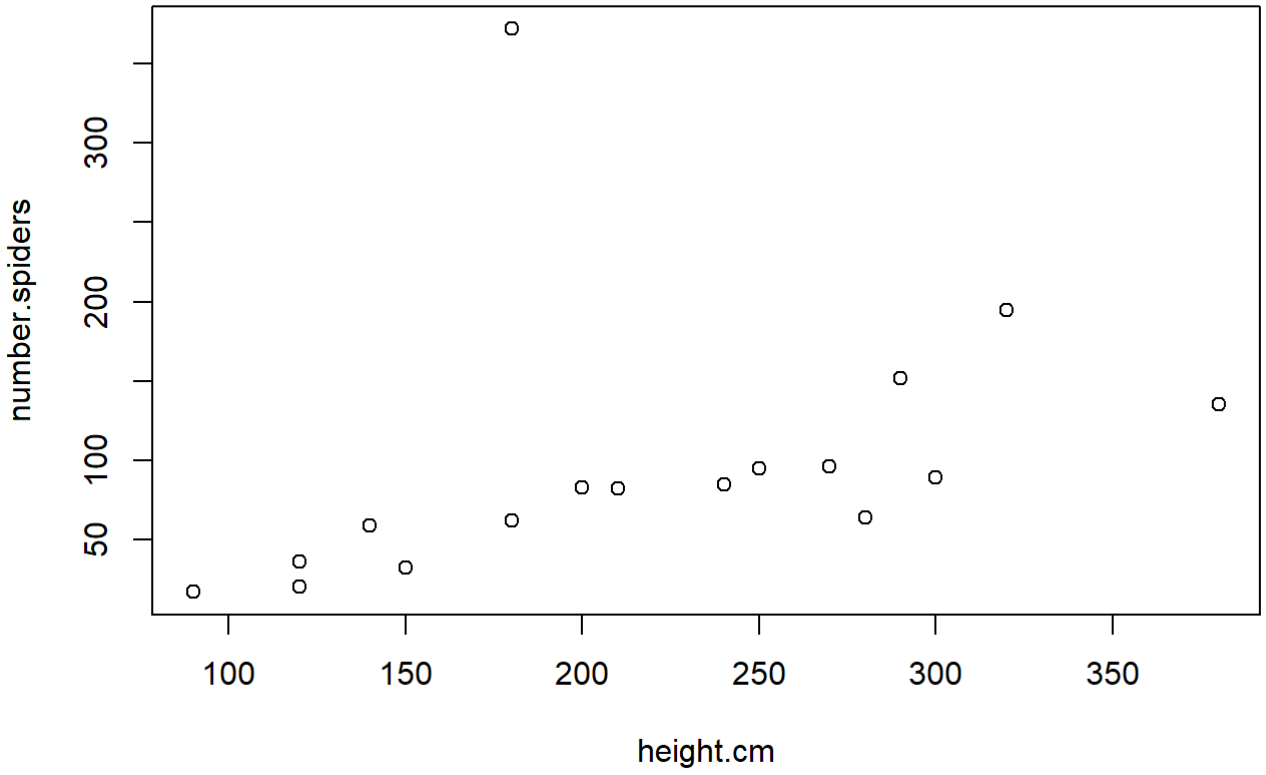
`Null hypothesis:` There is no linear relationship between the height of the web off the ground and the number of spiders in a colony.

`Alternative hypothesis:` There is a linear relationship between the height of the web off the ground and the number of spiders in a colony.

2 B. Make a simple scatterplot of the (untransformed) data using `plot()` . What stands out to you about the plot?

Hint: remember a simple scatterplot is just `plot(y~x)`

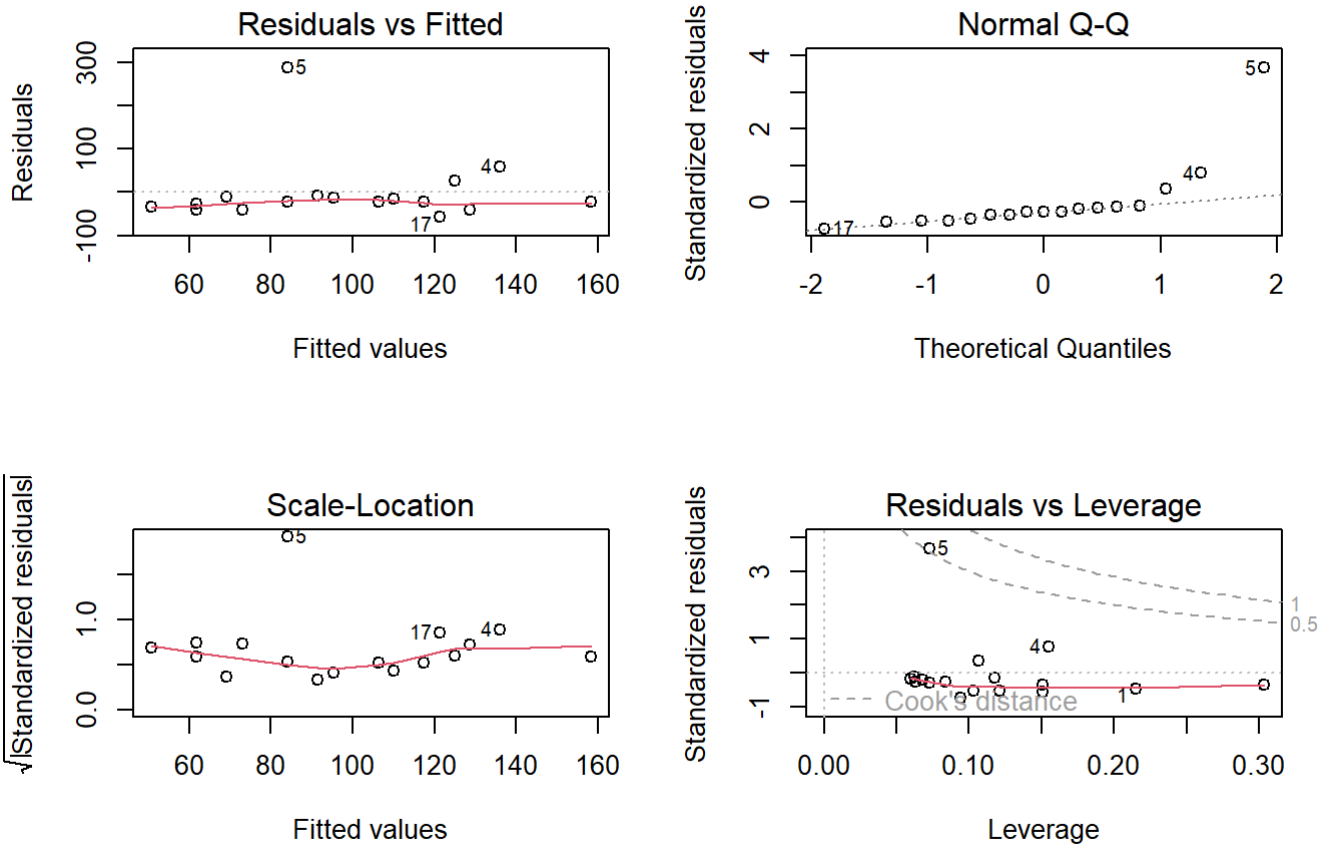
```
# plot of untransformed data
plot(number.spiders ~ height.cm, data = spiders_data)
```



[The scatter plot reveals a general linear trend of the relationship between the two variables.]

2 C. Fit a linear regression to the (untransformed) data, and look at the diagnostic plots like we did earlier and display them below. Based on these plots, are the assumptions of normality and equal variance met? Briefly explain your answer.

```
# fit linear regression to (untransformed) data
# Fit linear regression to the untransformed data
model <- lm(number.spiders ~ height.cm, data = spiders_data)

# look at diagnostic plots
par(mfrow = c(2, 2))  # Set the plotting area to display four plots in a 2x2 grid
plot(model)  # Generate the diagnostic plots
```



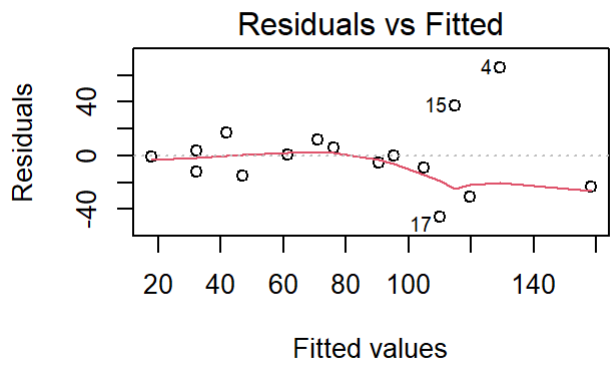
[In the residuals vs. fitted values plot,the residuals are randomly scattered around zero and there is no discernible pattern, it suggests that the assumption of equal variance is met.

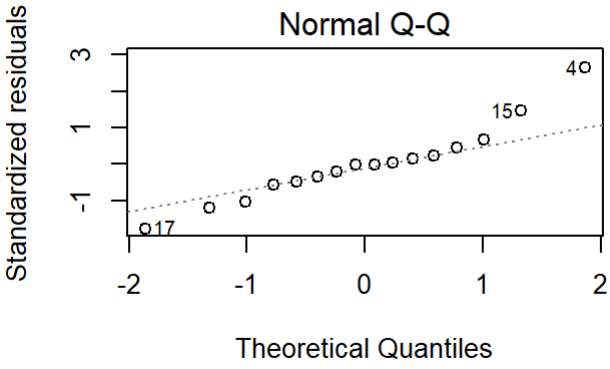
In the normal Q-Q plot, the points fall approximately along a straight line, it indicates that the assumption of normality is met.]

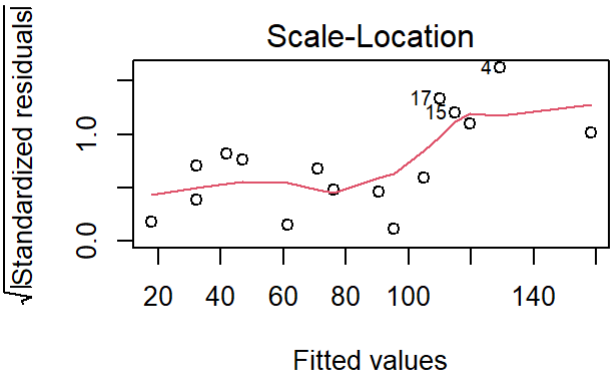
2 D. You learn that one of the research technicians miscounted observation 5 and you decide to drop it from your data and run a regression analysis. You can use the `subset()` function where `colony!=5` , and run your regression on the new subset. After you have fit the regression model, check your assumptions with *diagnostic plots* (the plots that contain the residuals vs fitted plot and qqPlot of residuals) and report whether you think your assumptions for the linear regression are met.

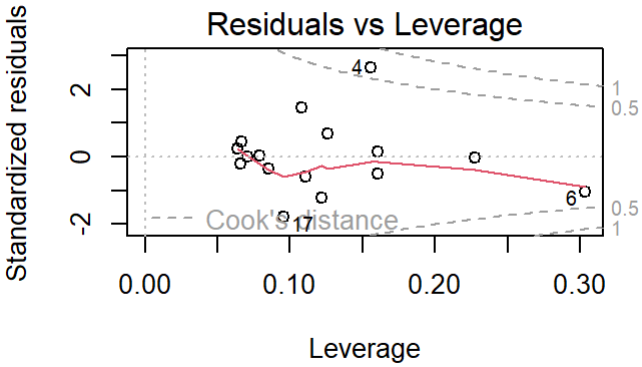
Hint: review your assumptions for a linear regression

```
# subset out the data for observation 5
subset_data <- subset(spiders_data, colony != 5)
# run new regression on new subset

model <- lm(number.spiders ~ height.cm, data = subset_data)
# check assumptions using diagnostic plots
par(mfrow = c(2, 2))  # Set the plot layout

plot(model)
```









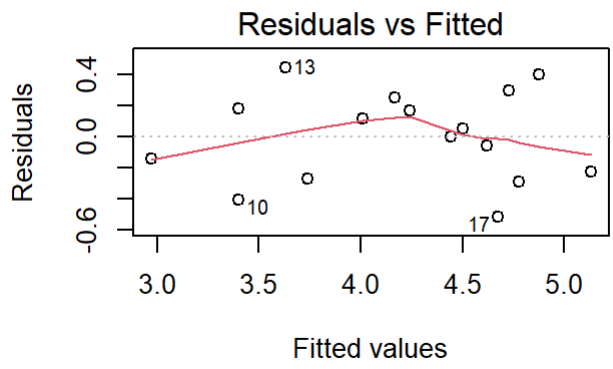
[In the residuals vs fitted the line falls from around the 80 value of the fitted values, it suggests that the variability of the residuals may not be constant across all levels of the predictor variable (height.cm). This indicates a violation of the constant variance assumption.

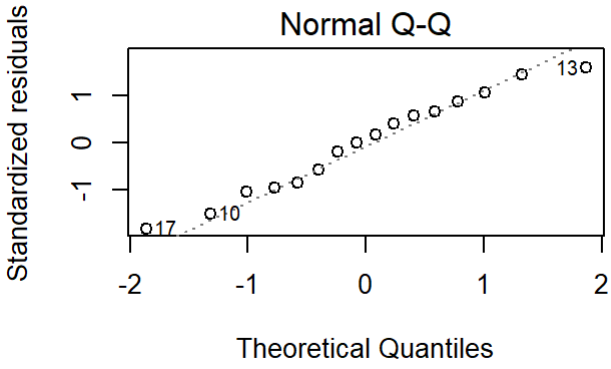
In the normal Q-Q plot, the points fall approximately along a straight line, it indicates that the assumption of normality is met.]

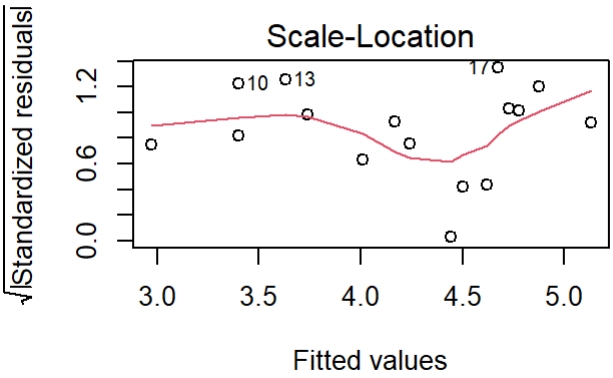
2 E. If necessary, try transforming your response and/or predictor variables. Report what transformations you tried in a sentence and show the resulting diagnostic plots below.

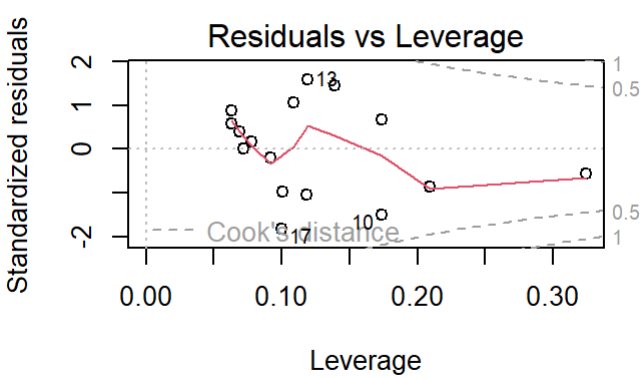
Hint: Make sure you continue to exclude colony 5!

```
# Transform your response and/or predictor variables
# HINT: use the subsetted data without observation 5 as your input
# Subset the data excluding observation 5

transformed_data <- subset(spiders_data, colony != 5)
transformed_data$log_height <- log(transformed_data$height.cm)
transformed_data$log_num_spiders <- log(transformed_data$number.spiders)

# run new regression on transformed data

lm_model_transformed <- lm(log_num_spiders ~ log_height, data = transformed_data)

# Check assumptions using diagnostic plots
par(mfrow = c(2, 2))
plot(lm_model_transformed)
```









[I tried Log transformation on both predictor and response variable]

2 F. Report the resulting linear regression model in the form: response variable = b0 + b1 * explanatory variable, filling in the variables. Then, interpret b1 in a sentence.

Hint: make sure to specify which variables are transformed if there are transformations

[log(number of spiders) = -3.776 + 1.500 * log(height)]

> The interpretation of b1 in the given linear regression model is as follows: For every 1% increase in the log-transformed height of the web, the log-transformed number of spiders is expected to increase by 1.500, holding all other variables constant.

2 G. Use the model you wrote down in the question above to predict the expected number of spiders in a colony 230cm off of the ground.

Hint: if you log transformed your data, remember that log in R is by default the natural log. To solve for the natural log (LN) you take the exponent of that value with the code `exp()`

```
# area to use r to get your answer
# Predicting the expected number of spiders
height <- 230
log_height <- log(height)
log_num_spiders <- -3.776 + 1.500 * log_height
num_spiders <- exp(log_num_spiders)

num_spiders
```

```
## [1] 79.92742
```

[79.92742]

2 H. Finally, report the R^2 value of the model and interpret it in a sentence.

[The R^2 value of the linear regression model is 0.8242, which means that approximately 82.42% of the variance in the number of spiders can be explained by the height of the web off the ground. This indicates a strong relationship between the predictor variable (log-transformed height) and the response variable (log-transformed number of spiders).]