MTH 325, Lab #1, Spring 2023 Name _____

Instructions: Follow along with the tutorial portion of the lab. Replicate the code examples in R on your own, along with the demonstration. Then use those examples as a model to answer the questions/perform the tasks that follow. Copy and paste the results of your code to answer questions where directed. Submit your response file and the code used (both for the tutorial and part two). Your code file and your lab response file should each include your name inside.

Scatterplots and Ordinary Linear Regression (OLS)
Let's begin by reviewing scatterplots. We can make them with base R graphics or using ggplot. We'll look at both.

```
3  data(mtcars)
4  par(mfrow = c(1,1))
5  plot(mtcars$qsec,mtcars$mpg,main="Example Scatterplot")
```

Alternatively, we can make the same plot in ggplot.

```
7  ggplot(data=mtcars, aes(x=qsec, y=mpg))+geom_point()+labs(title='Example Scatterplot')
```

We can calculate the correlation of our variables using any of the correlation measures we discussed in lecture.

```
9  cor(mtcars$qsec, mtcars$mpg,  method = "pearson")
10 cor(mtcars$qsec, mtcars$mpg,  method = "spearman")
11 cor(mtcars$qsec, mtcars$mpg,  method = "kendall")
```

In this case, Pearson's correlation is 0.418684, Spearman's correlation is 0.4669358, and Kendall's tau is 0.3153652.

These are in what we could call the moderately weak range.

To create our regression models we can use the lm() function (linear model).

```
13 fit <- lm(mpg ~ qsec, data = mtcars)
14 summary(fit)
```

The y-variable (response variable) goes first in the function call, and the independent variable(s) after the ~.

The summary gives us the model test and coefficients.

```
Call:
lm(formula = mpg ~ qsec, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-9.8760 -3.4539 -0.7203  2.2774 11.6491

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -5.1140    10.0295  -0.510   0.6139
qsec          1.4121     0.5592   2.525   0.0171 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.564 on 30 degrees of freedom
Multiple R-squared:  0.1753,    Adjusted R-squared:  0.1478
F-statistic: 6.377 on 1 and 30 DF,  p-value: 0.01708
```

In base R, we have to add the equation to the graph using the coefficients from our model output. We can add the line to a ggplot graph without this information because it can be done internally.

For base R, replot the scatterplot, and then use abline() to add the regression line.

```
16  plot(mtcars$qsec,mtcars$mpg,main="Example Scatterplot with Regression Line")
17  abline(-5.1140, 1.4121)
```

We use the coefficients from our model, intercept first, to draw the line on the graph. In ggplot, we can add both the line and the confidence interval easily.

```
19  ggplot(data=mtcars, aes(x=qsec, y=mpg))+geom_point()+
20    geom_smooth(method='lm',formula=y~x,se=TRUE,level=0.95)+labs(title='Linear Regression Example')
21
```

In this case, it looks like the intercept is not persuasively different from zero. We can force the intercept to be set to zero and redo the model.

```
22  fit <- lm(mpg ~ qsec+0, data = mtcars)
23  summary(fit)
24
```

And now the intercept doesn't appear in our output.

```
Call:
lm(formula = mpg ~ qsec + 0, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-9.8879 -3.5449 -0.8198  1.9272 11.4456

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
qsec  1.12836    0.05418   20.83   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.497 on 31 degrees of freedom
Multiple R-squared:  0.9333,    Adjusted R-squared:  0.9311
F-statistic: 433.7 on 1 and 31 DF,  p-value: < 2.2e-16
```

Pay attention to the R^2 value between the two versions.

When trying to decide which variables to use in your model, it can be helpful to look at the correlation between more than just one pair of variables at a time.

```
25 res <- cor(mtcars)
26 round(res, 2)
```

The second command just rounds the output so we aren't looking at 6-8 decimals each.

```
       mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
mpg   1.00 -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.66  0.60  0.48 -0.55
cyl  -0.85  1.00  0.90  0.83 -0.70  0.78 -0.59 -0.81 -0.52 -0.49  0.53
disp -0.85  0.90  1.00  0.79 -0.71  0.89 -0.43 -0.71 -0.59 -0.56  0.39
hp   -0.78  0.83  0.79  1.00 -0.45  0.66 -0.71 -0.72 -0.24 -0.13  0.75
drat  0.68 -0.70 -0.71 -0.45  1.00 -0.71  0.09  0.44  0.71  0.70 -0.09
wt   -0.87  0.78  0.89  0.66 -0.71  1.00 -0.17 -0.55 -0.69 -0.58  0.43
qsec  0.42 -0.59 -0.43 -0.71  0.09 -0.17  1.00  0.74 -0.23 -0.21 -0.66
vs    0.66 -0.81 -0.71 -0.72  0.44 -0.55  0.74  1.00  0.17  0.21 -0.57
am    0.60 -0.52 -0.59 -0.24  0.71 -0.69 -0.23  0.17  1.00  0.79  0.06
gear  0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  0.21  0.79  1.00  0.27
carb -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66 -0.57  0.06  0.27  1.00
```

The diagonal entries are 1 since these are correlations of a variable with itself. If we look along the top row or first column, we can see the correlations with mpg. The largest one (in absolute value) appears to with wt (weight), and the next highest with cylinders and displacement. Both of these variables also have high correlations with each other, so as we'll see when we do multiple regression, these may present collinearity problems, meaning that we can't use them in the same model if we use more than one variable.

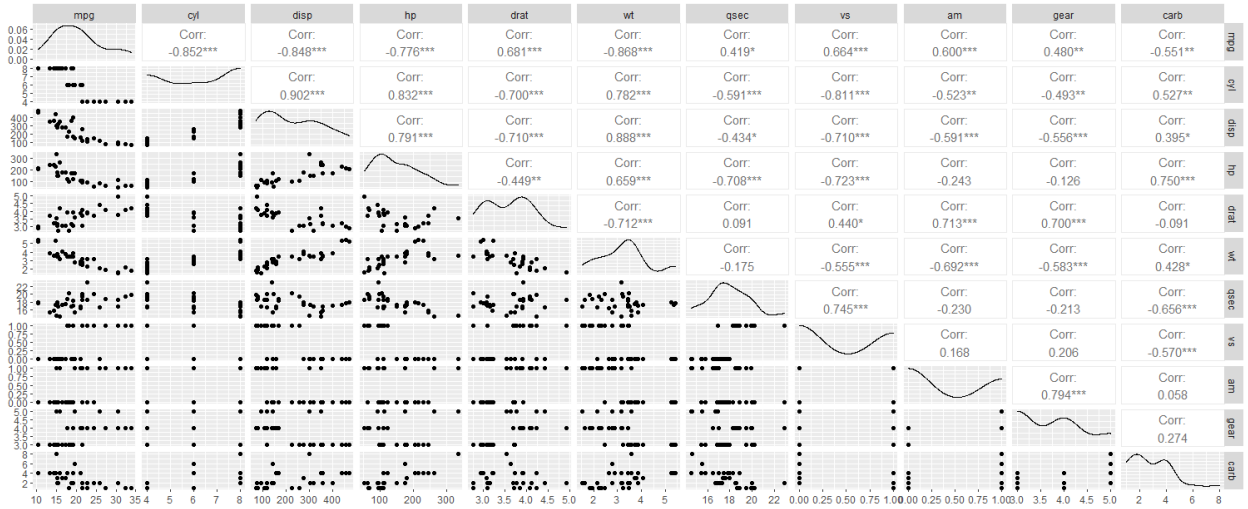
If a table of numbers is too hard to look at, we can also visualize this information.

```
28 library(corrplot)
29 corrplot(res, type = "upper", order = "hclust",  tl.col = "black", tl.srt = 45)
```

In the resulting graph, color indicates sign of the correlation, and the size of the dots represents the strength of the correlation.

```
34 ggpairs(mtcars)
35
```

You'll need to install the GGally package. The downside here is that since there are so many variables, the graph may have too much information in it to read easily. You may want to reduce the number of variables you are plotting first. But this plot includes density plots on the diagonal, scatterplots on the lower triangle, and correlation values and significance indicators on the upper diagonal.



Tasks

Review the lab directions for the write up for this semester, as they differ from the fall.

1. Use the built-in Orange data set (use the call data(Orange) to load the data). Create a scatterplot of the two variables. Use age to predict the second variable. Create a graph with the regression line on the graph. Conduct a hypothesis test on the model and the individual coefficients. If you reject variables, redo the model. Explain your process and interpret the final model. Include all graphs and output of the model.

2. Use the built-in trees data set. Create a correlation table and plot those correlations. Which variable, girth or height, appears to have the strongest correlation with volume? Plot the variable you choose against volume in a scatterplot. Assess the strength of the correlation visually and its linearity. Construct a linear model. Test your hypotheses about the model and the coefficients. If you reject any variables, redo the model. Explain your process and interpret the final model. What percentage of variability of volume can be explained by the linear relationship with your variable of choice? Include all graphs and output of the model(s).

References:

1. Discovering Statistics Using R. Andy Field, Jeremy Miles, Zoe Field. (2012)
2. http://www.stat.columbia.edu/~tzheng/files/Rcolor.pdf
3. https://book.stat420.org/applied_statistics.pdf
4. https://scholarworks.montana.edu/xmlui/bitstream/handle/1/2999/Greenwood_Book_Version3_CC_optimized.pdf?sequence=7&isAllowed=y
5. https://www.rstudio.com/resources/cheatsheets/
6. http://www.sthda.com/english/wiki/correlation-test-between-two-variables-in-r
7. https://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/
8. http://www.sthda.com/english/wiki/correlation-matrix-a-quick-start-guide-to-analyze-format-and-visualize-a-correlation-matrix-using-r-software
9. https://www.geeksforgeeks.org/how-to-create-and-interpret-pairs-plots-in-r/