Lesson 9 Companion

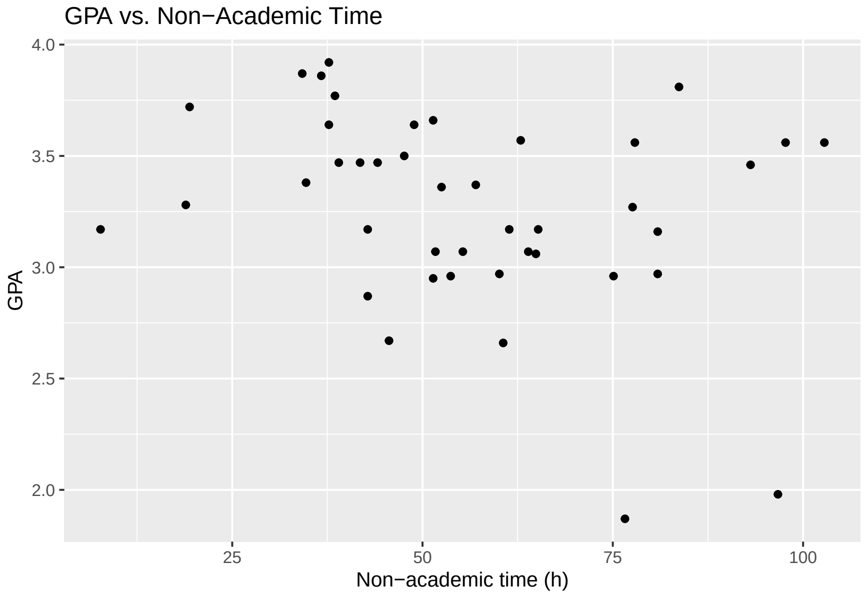
Data Exploration:

I will be utilizing the data from the GPA vs. non-academic time data set from section 10.4 for this example code.

```r
library(tidyverse)

GPA = read_table2("http://www.isi-stats.com/isi/data/chap10/GPA.txt")

head(GPA)
```

```
## # A tibble: 6 x 2
##   hours   gpa
##   <dbl> <dbl>
## 1   7.7  3.17
## 2  18.9  3.28
## 3  19.4  3.72
## 4  34.7  3.38
## 5  42.8  3.17
## 6  42.8  2.87
```

```r
GPA %>%
  ggplot(aes(x = hours, y = gpa)) +
  geom_point() +
  labs(x = "Non-academic time (h)", y = "GPA", title = "GPA vs. Non-Academic Time")
```



It appears as if there is a slightly negative association between these two variables. There are two unusual observations below a 2.0 GPA which are likely influential due to their large separation from the other observations.

Let’s establish our hypotheses for these tests:

H_0 : There is no association between GPA and time spent on non-academic activities ($\beta_1 = 0$)

H_a : There is an association ($\beta_1 \neq 0$)

Now let’s find the sample statistic we will use for our approaches. In this case, we will utilize the slope of the least squares regression line as our statistic. First a brief exploration of the **lm()** function in *R*. You will use **lm()** a lot in this course, you will learn to love it, and you will miss it when you leave the course. Like a lot of functions in *R*, it automates a lot of your work and provides more information than you know what to do with. Let’s start by building a **l**inear **m**odel for our research question.

```
model = lm(GPA$gpa ~ GPA$hours)

summary(model)
```

```
##
## Call:
## lm(formula = GPA$gpa ~ GPA$hours)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.27699 -0.20544  0.05476  0.33931  0.70479
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.597691   0.185733  19.370   <2e-16 ***
## GPA$hours   -0.005884   0.003070  -1.917   0.0624 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4303 on 40 degrees of freedom
## Multiple R-squared:  0.08411,    Adjusted R-squared:  0.06121
## F-statistic: 3.673 on 1 and 40 DF,  p-value: 0.06245
```

```
sample_stat = model$coefficients[[2]]
```

Starting with the first line, you should read these arguments inside the **lm()** function as “GPA as a function of hours.” Using this trick will help you decipher what model you are building when you move into more complicated models. Once we build this model we save it to the variable *model* for later use. The first use is to look at the summary of the model using the function… **summary()**. This summary provides a lot of important information about our model, much of which we will explore later. The first thing I want to do, however, is to simply establish how to write our the model from this summary output:

$\hat{y} = -0.005884x_1 + 3.597691$

where x_1 is non-academic time (hours)

You will recall that when we put a “hat” (ˆ) on something we are signifying it is an estimate. Therefore we can use this equation to find our estimate of y (GPA). This estimate will usually be different than the “actual” y values (the values in our data set). We can now identify our slope parameter (β_1) as -0.005884 and we will use this in the rest of the document. We can also see our intercept estimate (β_0) is 3.567691.

This means that, according to this model, for every one hour **increase** of non-academic time, we expect a 0.005884 **decrease** in GPA. Passes the common sense test for me.

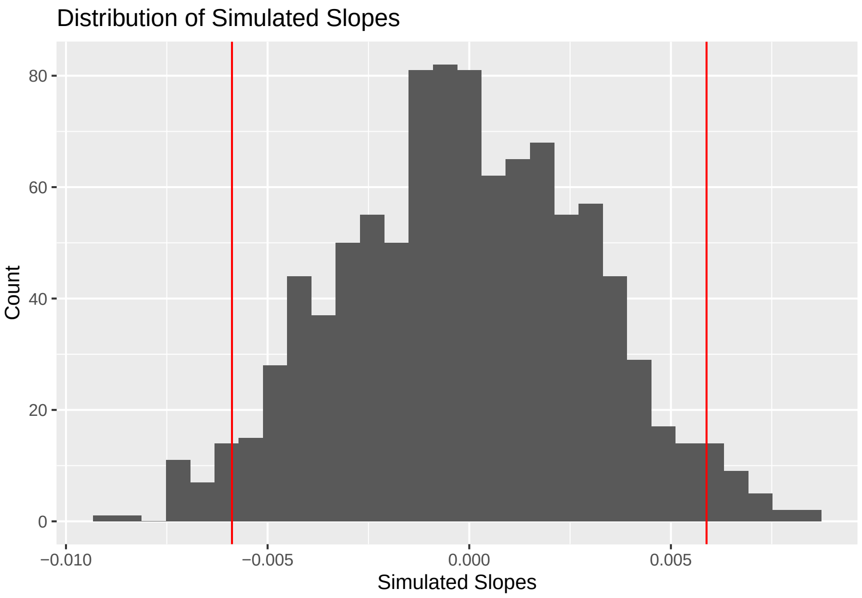
The last thing I want to point out on this summary output right now are the two r-squared values: *Multiple R-squared* and *Adjusted R-Squared*. For the purposes of this class (and your project) you will use multiple R-squared as it is the “normal” coefficient of determination discussed in your book. The adjusted R-squared is useful in predictive modeling contexts where you would like to “penalize” a model for over complication. If I’m trying to predict the GPA of a student and one model has three explanatory variables and an R^2 of 0.71 and another model has 10 explanatory variables and an R^2 of 0.72, I would prefer to use the less complicated model… even if accounts for less of the variation in the response variable. You would (usually) find that the adjusted R-squared would be lower for this more complicated second model given there is not much of a difference between the multiple r-squared values.

Now that we have identified our sample statistic (B_1) for our approaches, we will continue.

Simulation-based approach:

The first step in the simulation-based approach is always to build the null distribution. Once again, when we are building the null distribution we are assuming the truth of the null hypothesis. If there is no association between time and GPA than it doesn’t matter which time is associated with GPA. We can build our null distribution of simulated slopes by shuffling our explanatory and response pairs.

```r
replications_dataframe = NULL

num_reps = 1000

for (i in 1:num_reps){

  #Produce a new dataframe (scrambled_GPA) with a new column
  #that contains scrambled GPAs by sampling the original w/o
  #replacement.
  scrambled_GPA = GPA %>%
    mutate(new_GPA = sample(gpa, size = n(), replace = FALSE))

  #Produce model with this new_GPA but just pull out the slope
  trial_slope = lm(scrambled_GPA$new_GPA ~ scrambled_GPA$hours)$coefficients[[2]]

  #Add it to my list of simulated slopes
  replications_dataframe = rbind(replications_dataframe, data.frame(trial_slope))

}

replications_dataframe %>%
  ggplot(aes(x = trial_slope)) +
  geom_histogram() +
  labs(x = "Simulated Slopes", y = "Count",
       title = "Distribution of Simulated Slopes") +
  geom_vline(xintercept = -sample_stat, color = "red") +
  geom_vline(xintercept = sample_stat, color = "red")
```



```
replications_dataframe %>%
  summarise(pvalue = sum(abs(trial_slope) >= abs(sample_stat)) / n())
```

```
##   pvalue
## 1  0.058
```

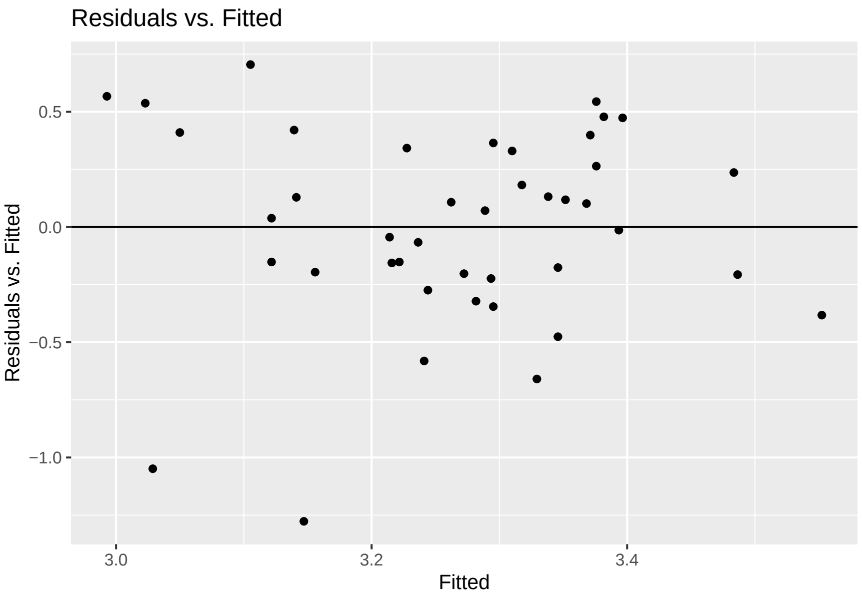
I know that was a big block of code but hopefully you recognize most of it from the previous simulation-based approach companion documents. It's important to understand that this null distribution represents the distribution of slopes we would expect to see if there was no association between non-academic time and GPA. This should help demonstrate that the slope isn't always zero between non-associated variables and (perhaps more importantly) that a slope that is determined from a sample (like the statistics we've discussed previously in this class) is not the definitive ground truth about the relationship between an explanatory and response variable. I believe that previous experience with linear regression makes it harder for students to accept this compared to the other sample statistics we discuss in this course. Your sample slope applies to that sample and may or may not apply beyond the sample... hence the need for inference.

Theory-based approach:

First we need to verify that we meet our validity conditions. Check out my document on these validity conditions on the MA256 GitHub for more information.

Linearity and **E**qual Variance:

```
model %>%
  ggplot(aes(x = .fitted, y = .resid)) +
  geom_point() +
  labs(x = "Fitted", y = "Residuals vs. Fitted",
       title = "Residuals vs. Fitted") +
  geom_hline(yintercept = 0)
```



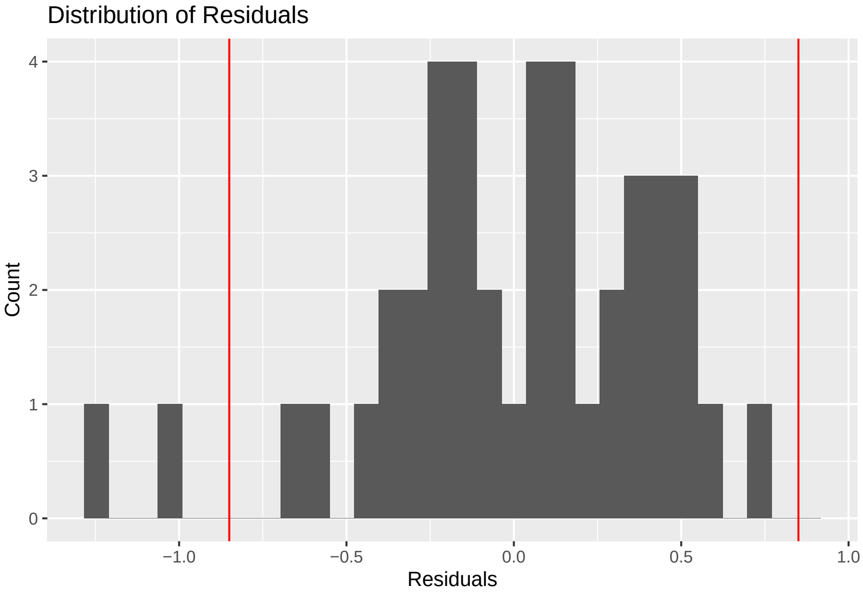
There does not appear to be a pattern to the residuals and there appears to be roughly equal spacing across the domain of the fitted values.

Independence:

It is reasonable to assume that the observations in our data set are independent from each other as the amount of time that one student spends on non-academic time and their GPA is logically independent from another.

Normally-distributed Residuals:

```
sd_resid = sd(model$residuals)

model %>%
  ggplot(aes(x = .resid)) +
  geom_histogram() +
  labs(x = "Residuals", y = "Count",
       title = "Distribution of Residuals") +
  geom_vline(xintercept = 2 * sd_resid, color = "red") +
  geom_vline(xintercept = -2 * sd_resid, color = "red")
```



While there are appear to be two outlying residuals (something we would have expected based on our data exploration) the vast majority of the residual values falls within two standard deviations of the mean (0). Furthermore, the histogram appears to be uni-model with the mode located in the center of the distribution.

With these validity conditions satisfied, we can continue with our theory-based approach.

Another thing that it is easy to lose track of in the course is what exactly is the *theory-based approach* for linear regression. Students often identify **lm()** as the theory-based approach because it conducts the theory-based test behind the scenes. As you may have identified, the theory-based approach for linear regression is another t-test. The code below goes a bit more in-depth than necessary for our purpose in this class but I didn't simply want to pull values from our model summary output.

```
#Sum of squared error
estimate = model$fitted.values
observed = GPA$gpa

sse = sum((observed - estimate)^2)

#Sum of squares
average_hours = mean(GPA$hours)

ss = sum((GPA$hours - average_hours)^2)

sample_size = nrow(GPA)

#Standard Error of Slope
standard_error = sqrt( ((1 / (sample_size - 2)) * sse) / ss)

tstat = (sample_stat - 0) /  standard_error

tstat
```

```
## [1] -1.91663
```

That was a lot of work to get a test statistic that matched what was already in the table but I wanted to demonstrate where the value comes from. Now that we have the t-statistic for our slope, we can use **pt()** to obtain our p-value. Keep in mind we are interested in the existence of an *association* between the explanatory and response variables which implies a two-sided test because it doesn't matter the direction of that association.

```r
2 * (1 - pt(abs(tstat), df = sample_size - 2))
```

```
## [1] 0.06244887
```

Once again, we can see that this p-value is already present in our model's summary output. You will notice that the degree of freedom parameter is n-2 this time instead of n-1. This is because we lose a degree of freedom for each predictor in our model. Read more about degrees of freedom here: https://statisticsbyjim.com/hypothesis-testing/degrees-freedom-statistics/

For the purposes of our class, you don't need to go through calculating your p-value by hand, trusting the summary output is enough. However, being able to identify the actual theory-based test and understanding the process is very important to linear regression and further concepts in the course.

Using the p-value:

We went through all this effort to build a model, interpret the coefficient, and calculate the p-value so what do we do now? Since we have met our validity conditions for utilizing the theory-based approach, I can safely use the p-value I calculated above. Going back the p-value guidelines in the *Introduction to Statistical Investigations* textbook, I see that a p-value of 0.0624 offers moderate evidence against the null hypothesis.

Recall that our null hypothesis is that there is no association between non-academic time and GPA. We found a slope estimate of -0.005884 for our sample and this p-value tells us that our sample provides moderate evidence against there being no association. There really isn't much else you can say beyond that if you haven't pre-established a significance level for your test.

This may feel a little disappointing but also not unexpected from our data exploration. Based on our scatter plot, we expected that there might be a slight negative association between the explanatory variable and the response variable in the data set. When we built our model we saw that our slope estimate matched this. Now that we completed our statistical test, we find that there is only moderate evidence to suggest we can make any inference from our sample to a true relationship between these two variables. If this was our life's work, it's probably worth further exploration to collect more data to get a more definitive answer.

What about interpreting the constant in our model?:

You probably remember that the b in your favorite equation ($y = mx + b$) is the y-intercept or constant in our linear model. So why didn't we spend any time interpreting it when we interpreted the other coefficient? Well, the primary reason is that I built this model because I was interested in the association between non-academic time and GPA. This question is answered using the coefficient and p-value for that variable.

Another reason is that interpreting the constant can be fraught with issues: Perhaps $x_n = 0$ falls outside your data set for all values of n? Perhaps $x_n = 0$ has no meaning such as for a weight or height variable? These are just a couple examples of why I don't waste too much time considering the intercept constant.

If you did want to consider it for this problem, we would say that the student who spends zero hours on non-academic activities will average a 3.597 GPA. That hardly seems worth it to me.

What about this ***F-statistic*** thing?:

You may have noticed that there is another test that is executed when you build a linear model using **lm()**: some test that uses the *F-statistic*. While we haven't covered this statistic (and distribution) yet, we will later in the course. For now it's sufficient to discuss the hypothesis associated with this test and what we can use it for. Unlike the *t-test* that only considers one regression coefficient at a time, the *F-test* can compare multiple coefficients.

H_0 : The fit of the intercept-only model and your model are equal.

H_a : The fit of the intercept-only model is significantly reduced compared to your model.

It's important to consider the result of the F-test as it informs you whether your explanatory variable is contributing to the fit of your model. When you are only considering a single explanatory variable it hardly seems like this extra p-value is worthwhile… but what if you had more than one explanatory variable? Overall, this *model utility test* is generally used in predictive modeling to judge whether our model is better than the intercept-only model.