# Week_6_exercise

Katz

2/22/2021

```
x <- 3
```

**Problem 1** Interpretation of analysis for problem 1

## Problem 2

## For this exercise, you will be working with the student_happiness.csv file.

This file contains simulated data with several variables. The scenario is the following: You are interested in studying engineering student well being. One dimension of student health is their mental health. In particular, you have a measure of student happiness. This will be your outcome variable.

You have reason to believe that there are several factors that can contribute to a person's happiness. In particular, you believe that their class standing (e.g., undergrad, masters, phd), discipline (you only sampled mechanical, civil, and electrical engineering students), time spent outdoors, and time spent on zoom all play a role in a student's happiness. Sooo you go out and survey students, collecting each of these variables in addition to giving them a series of questions that let you calculate their happiness score. The composite score is what you now have in your `student_happiness.csv` file.

The objective here is to model the outcome as a function of the different predictors that you have. There will be a little less scaffolding than last week, but you can follow the class example in the Week_6_demo if you get stuck or can't think of what to do next…

First, it's probably a good idea to load in the data…

```
# pop_df <- read_csv("YOUR PATH HERE")
```

The `student_happiness.csv` file has an entire population of students in it. In reality, you will only be working with a sample of that total population. I have provided you with the full population so that you can test the effects of larger sample sizes on your model. I suggest you create a new dataframe that you create by sampling from the original dataframe with the `sample_n()` function from tidyverse. This sample is what you can use for the rest of your code. This will allow you to quickly change the sample size a few different times and look at how that affects model performance.

```
# sample_df <- pop_df %>% sample_n(size = 50)
```

Start with a sample of 50 students and change it after you create your model to see what happens if you have a sample size of 100 or 200 instead of 50

Next, it's never a bad idea to start summarizing what you have. You can try visualizing things with geom_bar() or geom_point() (as appropriate) or make summary tables with things like group_by() and summarize() or count().

```
# sample_df %>%
```

Now, it's probably time to start creating your model (or models...?).

Begin with just a simple model that has one predictor. You can choose which one.

```
# your_model <- lm(...)

#summary(your_model)
```

Next, try creating a more complicated model to test how multiple predictors affect the model

```
#your_fancier_model <- lm(...)

#summary(...)
```

Check your residuals for outliers and other model diagnostics. Try using the code in the book or in the `Week_6_demo.Rmd` file. You should try the Durbin Watson test for independent errors, looking at residuals, identifying potential influential cases, and multicollinearity (with the Variance Inflation Factor) or looking at a correlation matrix (be careful with this second option since you may have categorical predictors).

Try writing a few sentences in your markdown file about how you interpret the results of your model.