Data Exploration Project

Kyle

## Load Libraries

library(rio)  
library(tidyverse)

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ dplyr 1.1.3 ✔ readr 2.1.4  
✔ forcats 1.0.0 ✔ stringr 1.5.0  
✔ ggplot2 3.4.3 ✔ tibble 3.2.1  
✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
✔ purrr 1.0.2   
── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
✖ dplyr::filter() masks stats::filter()  
✖ dplyr::lag() masks stats::lag()  
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(lubridate)  
library(dplyr)

## Import Data

filelist <- list.files('Lab3\_Rawdata', pattern = 'trends', full.names = TRUE)  
  
my\_data <- import\_list(filelist, fill = TRUE, rbind = TRUE)

## Data Cleaning

#Turning monthorweek into a single date that's usable  
clean\_data <- my\_data %>% mutate(actual\_date = ymd(str\_sub(monthorweek, start = 0, end = 10)))  
  
cleaned\_data\_month <- clean\_data %>%  
 group\_by(month = floor\_date(actual\_date, "month"))

## Aggregating

#Aggregate index variable by subtracting mean of index and dividing the result by the standard deviation of index, calculating both of those within school name and keyword   
aggregated\_data <- cleaned\_data\_month %>%  
 group\_by(schname, keyword) %>%  
 mutate(aggregate\_index = (index - mean(index)) / sd(index), na.rm = TRUE)

## Reading in the Scorecard Data

#Importing in Scorecard Data, a subfolder from Lab3\_Rawdata  
  
scorecard\_data <- import("Lab3\_Rawdata/Most+Recent+Cohorts+(Scorecard+Elements).csv")  
  
id\_name\_link <- import('Lab3\_Rawdata/id\_name\_link.csv')

## Merge in the Scorecard Data

#Merging in Scorecard data by opeid and OPEID  
id\_name\_link\_merge <- id\_name\_link %>%  
 group\_by(schname) %>%  
 mutate(n = n()) %>%  
 filter(n() == 1)  
  
schnamemerge <- inner\_join(aggregated\_data, id\_name\_link\_merge, by = "schname")  
  
merged\_data <- inner\_join(schnamemerge, scorecard\_data, by = c("opeid" = "OPEID"))

Warning in inner\_join(schnamemerge, scorecard\_data, by = c(opeid = "OPEID")): Detected an unexpected many-to-many relationship between `x` and `y`.  
ℹ Row 47229 of `x` matches multiple rows in `y`.  
ℹ Row 1073 of `y` matches multiple rows in `x`.  
ℹ If a many-to-many relationship is expected, set `relationship =  
 "many-to-many"` to silence this warning.

## Limiting Data To Colleges That Predominantly Grant Bachelor’s Degrees and Separating Out Variables We Need

#PREDEG in scorecard data needs to only be 3 to have predominantly the bachelors degrees  
#Sort out variables that we need   
  
merged\_data\_important <- merged\_data %>%  
 filter(PREDDEG == 3, na.rm = TRUE) %>%  
 group\_by(schname, unitid, opeid, month, `md\_earn\_wne\_p10-REPORTED-EARNINGS`)%>%  
 summarise(index = mean(index, na.rm = TRUE), aggregate\_index = mean(aggregate\_index, na.rm = TRUE))

`summarise()` has grouped output by 'schname', 'unitid', 'opeid', 'month'. You  
can override using the `.groups` argument.

## Defining High vs Low Earnings Based on Median

#Defining high vs low earnings based on median of earnings  
  
threshold <- median(merged\_data\_important$`md\_earn\_wne\_p10-REPORTED-EARNINGS`)

Warning in mean.default(sort(x, partial = half + 0L:1L)[half + 0L:1L]):  
argument is not numeric or logical: returning NA

#median income is 42000  
  
merged\_data\_important$highvslow <- ifelse(merged\_data\_important$`md\_earn\_wne\_p10-REPORTED-EARNINGS` >= threshold, "High", "Low")

## Defining our Cutoff (aka Scorecard Data Implementation)

#Defining and implementing our cutoff on Pre September 2015 and Post September 2015 for DID analysis and cutoff for High and Low Earnings  
  
together <- merged\_data\_important%>%  
 group\_by(schname) %>%  
 mutate(aggregate\_index = (index - mean(index)) / sd(index), na.rm = TRUE) %>%  
 group\_by(month, opeid, `md\_earn\_wne\_p10-REPORTED-EARNINGS`) %>%  
 summarise(aggregate\_index = mean(index, na.rm = TRUE))

`summarise()` has grouped output by 'month', 'opeid'. You can override using  
the `.groups` argument.

defined\_data <- together %>%  
 mutate(highearners = `md\_earn\_wne\_p10-REPORTED-EARNINGS` >= 42000, after = month >= as.Date("2015-09-01"))

## The Regression

#Running a difference in differences regression to see the effects of the Google trends index on our high and low earning colleges before and after the Scorecard introduction  
library(fixest)  
  
regression <- feols(aggregate\_index ~ highearners\*after, data = defined\_data)

NOTE: 642 observations removed because of NA values (LHS: 642, RHS: 642).

etable(regression)

regression  
Dependent Var.: aggregate\_index  
   
Constant 47.04\*\*\* (0.0798)  
highearnersTRUE 2.323\*\*\* (0.1127)  
afterTRUE -3.577\*\*\* (0.1822)  
highearnersTRUE x afterTRUE -1.474\*\*\* (0.2575)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 71,556  
R2 0.02137  
Adj. R2 0.02133  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Write Up

The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high-earning graduates by 1.474 units relative to what it did for colleges with low-earning graduates, with a standard error of 0.2575. This result comes from the “aggregate\_index” , “highearners”, and “after” coefficient(s) in my regression.

I chose the variables to include in my analysis because I wanted to run difference in differences (DID) analysis to compare the results of the Google Trends search activity on colleges with high earning graduates relative to colleges with low earning graduates, before and after the Scorecard data was introduced. DID in my opinion, was the best option to illustrate this. Therefore I needed to make a variable for after the implementation of the Scorecard. Since high and low income was not defined, I decided to create a cutoff in income using the median income as a threshold. I found this number to be $42000. Anything above that was considered high income and anything below that was considered low income. Further, I needed to filter out the colleges to those that predominantly grant bachelor’s degrees using PREDEG = 3. I also needed to turn the monthorweek variable into a single date that’s usable, as it was a range of dates. I then simplified it down further to just the months as that was simpler and still allowed us to see the results over time. I also aggregated the index variable by subtracting the mean of index and dividing the result by the standard deviation of index.

My analysis addresses the research question because it shows the effects of the Scorecard data on Google trends searches on high and low income colleges, before and after September 2015, when the Scorecard data was introduced. In the graph below, the Google Index results are on the Y-axis and the Time in Years is on the X -axis. The blue line shows colleges with high-earning graduates and the red line shows colleges with low-earning graduates. The graph overall shows how overtime, the Google trends searches were already decreasing in waves, and after the Scorecard data was introduced in September 2015, it continued to decrease in waves.

In conclusion, our DID analysis shows some significance at the 95% level, however, with a decrease of -1.474 units, it is not convincing enough to conclude that the introduction of the Scorecard data had much of an effect on high-earning graduates relative to what it did for low-earning graduates.

## **Visualization**

#The colors show the colleges with high earning graduates (Blue) and colleges with low earning graduates (Red) and the graph shows the Google search index before September 2015 and after September 2015.   
  
library(ggplot2)  
  
#clean up: groupby and summarize "month" and "after" and take the mean index of the group  
for\_viz <- defined\_data %>%  
 group\_by(month, highearners) %>%  
 summarise(new\_index = mean(aggregate\_index, na.rm = TRUE))

`summarise()` has grouped output by 'month'. You can override using the  
`.groups` argument.

ggplot(data = for\_viz, aes (x = month, y = new\_index, color = highearners)) + geom\_line() + geom\_vline(xintercept = as.Date(ymd("2015-09-01"))) + theme\_minimal() + labs(x = "Year", y = "Google Index",title = "Google Index Score Per Month For High and Low Earners, Before and After Introduction of ScoreCard Data") + theme(plot.title = element\_text(size = 9)) + labs(color = "High Earner")

Warning: Removed 2 rows containing missing values (`geom\_line()`).

