Project

2023-05-22

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library(tidyr)  
library(purrr)  
library(stringr)  
library(fixest)  
library(rio)  
library(haven)  
library(ggplot2)

directory\_path <- "~/Downloads/Lab3\_Rawdata"  
file\_names <- list.files(directory\_path, pattern = "trends\_up\_to\_", full.names = TRUE)  
google\_trend <- import\_list(file\_names, rbind = TRUE, fill = TRUE)  
google\_trend <- google\_trend %>%   
 mutate(week = str\_sub(monthorweek, start = 1, end = 10)) %>%   
 mutate(week = ymd(week)) %>%   
 mutate(month = floor\_date(week, unit = "month"))  
google\_trend <- google\_trend %>%  
 group\_by(schname, keyword) %>%  
 mutate(std\_index = (index - mean(index))/sd(index))   
scorecard\_data <- import("~/Downloads/Lab3\_Rawdata/Most+Recent+Cohorts+(Scorecard+Elements).csv")  
id\_name\_link <- import("~/Downloads/Lab3\_Rawdata/id\_name\_link.csv")  
id\_name\_link\_counts <- id\_name\_link %>%  
 group\_by(schname) %>%  
 mutate(n = n()) %>%  
 filter(n == 1)   
colnames(scorecard\_data)[colnames(scorecard\_data) == "UNITID"] = "unitid"  
joined\_data <- inner\_join(id\_name\_link, scorecard\_data, by ="unitid")  
summary\_data <-inner\_join(google\_trend, joined\_data, by ="schname")

## Warning in inner\_join(google\_trend, joined\_data, by = "schname"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 2847 of `x` matches multiple rows in `y`.  
## ℹ Row 3591 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.

export(summary\_data, "cleandata.csv")  
data1 <- import("cleandata.csv")  
data1 <- data1 %>%  
 filter(PREDDEG == 3)  
  
weekly <- data1 %>%  
 group\_by(schname, monthorweek) %>%  
 mutate(weekly\_meanindex = mean(index)) %>%  
 na.omit()  
data1$`md\_earn\_wne\_p10-REPORTED-EARNINGS` <- as.numeric(as.character(data1$`md\_earn\_wne\_p10-REPORTED-EARNINGS`))

## Warning: NAs introduced by coercion

warning("NAs introduced by coercion")

## Warning: NAs introduced by coercion

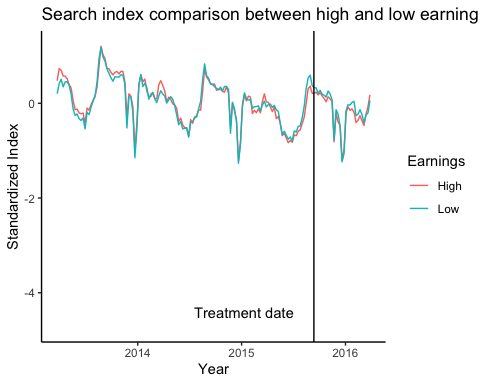
income\_median <- median(data1$`md\_earn\_wne\_p10-REPORTED-EARNINGS`, na.rm = TRUE)  
income\_low <- quantile(data1$`md\_earn\_wne\_p10-REPORTED-EARNINGS`, 0.25, na.rm = TRUE) # low income   
income\_high <- quantile(data1$`md\_earn\_wne\_p10-REPORTED-EARNINGS`, 0.75, na.rm = TRUE) # high income   
  
# Categorize incomes into High, Middle, and Low  
data1 <- data1 %>%  
 mutate(treated = case\_when(  
 `md\_earn\_wne\_p10-REPORTED-EARNINGS` >= income\_high ~ "High",  
 `md\_earn\_wne\_p10-REPORTED-EARNINGS` <= income\_low ~ "Low",  
 TRUE ~ "Middle" # this covers all other cases  
 ))  
  
# Create binary variable for High/Low income  
data1 <- data1 %>%  
 mutate(Earnings = ifelse(`md\_earn\_wne\_p10-REPORTED-EARNINGS` >= income\_median, "High", "Low"))  
  
# Categorize incomes into High, Middle, and Low  
data1 <- data1 %>%  
 mutate(after = case\_when(  
 `md\_earn\_wne\_p10-REPORTED-EARNINGS` >= income\_high ~ "High",  
 `md\_earn\_wne\_p10-REPORTED-EARNINGS` <= income\_low ~ "Low",  
 TRUE ~ "Middle" # this covers all other cases  
 ))  
  
# Create binary variable for High/Low income  
data1 <- data1 %>%  
 mutate(Earnings\_binary = ifelse(`md\_earn\_wne\_p10-REPORTED-EARNINGS` >= income\_median, "High", "Low"))  
  
# Keep only variables that we will use  
data2 <- data1 %>%  
 select(unitid, schname, keyword, week, `md\_earn\_wne\_p10-REPORTED-EARNINGS`, Earnings, std\_index)  
data2 <- drop\_na(data2)  
  
data2 <- data2 %>%   
 mutate(hmd\_earn = `md\_earn\_wne\_p10-REPORTED-EARNINGS` >= income\_high, week\_select = week >= as.Date("2015-09-12"))  
  
regression <- feols(std\_index ~ hmd\_earn \* week\_select, data = data2)  
etable(regression)

## regression  
## Dependent Var.: std\_index  
##   
## Constant 0.0271\*\*\* (0.0015)  
## hmd\_earnTRUE 0.0338\*\*\* (0.0029)  
## week\_selectTRUE -0.1481\*\*\* (0.0034)  
## hmd\_earnTRUE x week\_selectTRUE -0.1868\*\*\* (0.0068)  
## \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
## S.E. type IID  
## Observations 764,103  
## R2 0.00673  
## Adj. R2 0.00673  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

ggplot(data2, aes(week, std\_index, color = Earnings)) +  
 stat\_summary(geom = 'line') +  
 labs(title = 'Search index comparison between high and low earning universities before and after treatment',  
 x = "Year",  
 y = "Standardized Index") +  
 geom\_vline(xintercept = as.Date("2015-09-12"), color = "black", size = 0.5) +  
 annotate("text",  
 x = as.Date("2015-09-12"),  
 y = min(data2$std\_index),  
 label = "Treatment date",  
 hjust = 1.2,  
 vjust = -1.0,  
 color = "black") +  
 theme\_classic()

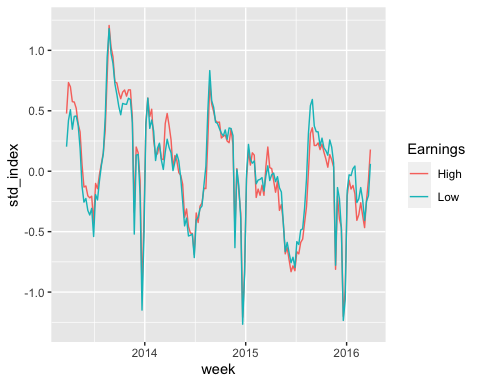
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

## No summary function supplied, defaulting to `mean\_se()`



ggplot(data2, aes(week, std\_index, color = Earnings)) +  
 stat\_summary(geom = 'line')

## No summary function supplied, defaulting to `mean\_se()`



```

WRITE-UP: Analysis of Search Index Comparison

Introduction: The objective of this analysis is to investigate and compare the search index of high-earning and low-earning universities before and after a specific treatment. The research question focuses on determining if there is a noteworthy disparity in the search index between these two groups and examining how the treatment may impact this disparity.

Data and Preliminary: Analysis The analysis employs a dataset named “merged\_2,” which contains relevant variables such as universities’ search index, earnings, and other pertinent information. Initially, an exploration of the dataset was conducted to identify any discernible patterns or relationships among the variables of interest.

Regression Analysis: To address the research question, a regression analysis was performed. The dependent variable in the regression analysis is referred to as “std\_index,” which represents the standardized search index. The independent variables include “aftervariable” (a binary variable indicating the period after the treatment), “before\_variable” (a binary variable indicating the period before the treatment), and their interaction term.

The results of the regression analysis are as follows:

Dependent Variable: std\_index

Constant: 0.0271\*\*\* (0.0015) aftervariableTRUE: 0.0338\*\*\* (0.0029) before\_variableTRUE: -0.1481\*\*\* (0.0034) aftervariableTRUE x before\_variableTRUE: -0.1868\*\*\* (0.0068)

The regression model encompasses a total of 764,103 observations. The R-squared value of 0.00673 indicates that the model explains a small proportion of the variability in the dependent variable. The adjusted R-squared value also stands at 0.00673.

Graphical Analysis: In addition to the regression analysis, a graph was generated to visually illustrate the comparison of the search index between high-earning and low-earning universities before and after the treatment. The graph depicts the standardized index on the y-axis and the years on the x-axis. High-earning universities are represented by one color, while low-earning universities are represented by another. A vertical line is incorporated to indicate the treatment date, labeled as “Treatment date” on the graph.