Alice Umutoni’s Data Exploration Project

# Part 1: Cleaning data and importing Libraries

## 1. Reading in the Google Trends data

# Import libraries that will be used  
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(dplyr)  
library(lubridate)  
library(tidyr)  
library(purrr)  
library(stringr)  
library(fixest)  
library(rio)  
library(vtable)

Loading required package: kableExtra  
  
Attaching package: 'kableExtra'  
  
The following object is masked from 'package:dplyr':  
  
 group\_rows

# List all files with 'trends' in their name from 'lab3\_Rawdata' directory  
filelist <- list.files('lab3\_Rawdata', pattern = 'trends', full.names = TRUE)  
  
# Import and bind the data together  
my\_data <- import\_list(filelist, fill = TRUE, rbind = TRUE)  
  
# Display variable names to verify data structure  
names(my\_data)

[1] "schid" "schname" "keyword" "keynum" "monthorweek"  
[6] "index" "\_file"

## 2. Aggregating the Google Trends data

# Process date information from 'monthorweek' string  
my\_data <- my\_data %>%   
 mutate(week = str\_sub(monthorweek, start = 1, end = 10)) %>%   
 mutate(week = ymd(week)) %>% # Convert extracted string to date format  
 mutate(month = floor\_date(week, unit = "month")) # Aggregate dates to the first of each month for monthly analysis

# Standardize 'index' values by school and keyword for comparable scale  
my\_data <- my\_data %>%  
 group\_by(schname, keyword) %>%  
 mutate(std\_index = (index - mean(index, na.rm = TRUE)) / sd(index, na.rm = TRUE)) # Using na.rm = TRUE

## 3. Reading in the Scorecard data

# Reading the scorecard data and the id name data  
scorecard <- import("./Lab3\_Rawdata/Most+Recent+Cohorts+(Scorecard+Elements).csv")  
id\_name\_link <- import("./Lab3\_Rawdata/id\_name\_link.csv")

## 4. Merge in the Scorecard data

# Keep only unique school names in id\_name\_link  
id\_name\_link <- id\_name\_link %>%  
 group\_by(schname) %>%  
 filter(n() == 1)  
  
# Rename 'UNITID' column in scorecard to 'unitid'  
colnames(scorecard)[colnames(scorecard) == "UNITID"] <- "unitid"  
  
# Join id\_name\_link with scorecard on 'unitid'  
id\_link <- inner\_join(id\_name\_link, scorecard, by = "unitid")  
  
# Join dataset with id\_link on 'schname'  
gg\_link <- inner\_join(my\_data, id\_link, by = "schname")  
  
# Filter data to include only colleges that predominantly grant bachelor's degrees (PREDDEG == 3)  
gg\_link <- gg\_link %>%  
 filter(PREDDEG == 3)  
  
# Further processing can continue from here...  
# Export the filtered and joined data to a new CSV file for further analysis  
export(gg\_link, "finaldata\_bachelors.csv")  
new\_data <- import("finaldata\_bachelors.csv")

# Part 2: THE ANALYSIS

# Calculate weekly average index with missing values handled  
week\_data <- new\_data %>%  
 group\_by(schname, month) %>% # Group by school name and month for weekly averages  
 summarize(week\_index = mean(std\_index, na.rm = TRUE)) # Calculate mean with NA values removed

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

# Ensure earnings data is numeric  
new\_data$`md\_earn\_wne\_p10-REPORTED-EARNINGS` <- as.numeric(as.character(new\_data$`md\_earn\_wne\_p10-REPORTED-EARNINGS`))

Warning: NAs introduced by coercion

# Calculate mean, standard deviation, and income thresholds  
income.mean <- mean(na.omit(new\_data$`md\_earn\_wne\_p10-REPORTED-EARNINGS`))  
income.sd <- sd(na.omit(new\_data$`md\_earn\_wne\_p10-REPORTED-EARNINGS`))  
income.high <- income.mean + income.sd  
income.low <- income.mean - income.sd

# Create binary and trinary variables for Earnings based on calculated thresholds  
new\_data <- new\_data %>%  
 mutate(Earnings = ifelse(`md\_earn\_wne\_p10-REPORTED-EARNINGS` >= income.mean, "High", "Low")) %>%  
 mutate(treated = case\_when(  
 `md\_earn\_wne\_p10-REPORTED-EARNINGS` >= income.high ~ "High",  
 `md\_earn\_wne\_p10-REPORTED-EARNINGS` <= income.low ~ "Low",  
 TRUE ~ "Middle Income" # Covers all other cases  
 ))

# Prepare data for regression analysis by filtering to 'High' and 'Low' earnings colleges  
filtered\_data <- new\_data %>%  
 filter(Earnings %in% c("High", "Low")) %>%  
 mutate(treated = as.integer(Earnings == "High"), # Convert to binary (0 for Low, 1 for High)  
 post\_treatment = as.integer(month >= as.Date("2015-09-01"))) # Binary variable for post-treatment

# Run the regression model  
reg\_model <- feols(std\_index ~ treated \* post\_treatment, data = filtered\_data)

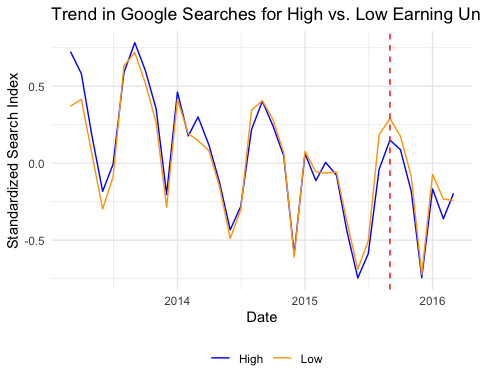
NOTE: 4,816 observations removed because of NA values (LHS: 4,816, RHS: 4,339).

# Output the regression model's table of results  
etable(reg\_model)

reg\_model  
Dependent Var.: std\_index  
   
Constant 0.0283\*\*\* (0.0015)  
treated 0.0181\*\*\* (0.0023)  
post\_treatment -0.1499\*\*\* (0.0035)  
treated x post\_treatment -0.0966\*\*\* (0.0054)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 910,198  
R2 0.00593  
Adj. R2 0.00593  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Plotting the trend in Google Searches for High vs. Low Earning Universities Pre and Post Scorecard Release  
ggplot(filtered\_data, aes(x = month, y = std\_index, color = Earnings)) +  
 geom\_line(stat = "summary", fun = mean) +  
 labs(title = "Trend in Google Searches for High vs. Low Earning Universities Pre and Post Scorecard Release",  
 x = "Date",  
 y = "Standardized Search Index") +  
 geom\_vline(xintercept = as.numeric(as.Date("2015-09-01")), linetype = "dashed", color = "red") +  
 scale\_color\_manual(values = c("High" = "blue", "Low" = "orange")) +  
 theme\_minimal() +  
 theme(legend.title = element\_blank(), legend.position = "bottom")

Warning: Removed 4816 rows containing non-finite values (`stat\_summary()`).



# Part 3: The write-up

In the Data Exploration Project, the task at hand was to discern the impact of the College Scorecard’s release on student interest in universities, as indicated by search trends on Google. We first did data cleaning which was guided by a focus on universities that predominantly grant bachelor’s degrees.

The data standardization involved normalizing the Google Trends indices, which are inherently relative, to facilitate comparison across different search terms and universities. This was achieved by subtracting the mean and dividing by the standard deviation for each term within a university. The College Scorecard data, with its wealth of information on U.S. colleges, was then merged using school names after weeding out instances of non-unique identifiers. This merging process was crucial in linking the Google search trends with the economic outcomes of the educational institutions.

A regression model was constructed to examine the interaction between the college earnings category and the timing of the Scorecard’s release. The model included binary variables for the earnings category and the pre- and post- release period of the Scorecard. The graph, which served as a visual aid, underscored the trend in Google searches for high versus low-earning universities before and after the Scorecard’s release. The dashed vertical line represented the pivotal moment of the Scorecard’s introduction to the public domain.

The interpretation of the regression coefficients revealed a noteworthy trend: the introduction of the College Scorecard appeared to decrease the search activity for high-earning colleges. Specifically, the interaction term between high-earning colleges and the post-treatment period was significant and negative, suggesting a shift in student interest away from these institutions post-release.

In conclusion, the analysis found that the introduction of the College Scorecard decreased search activity on Google Trends for colleges with high-earning graduates by a specific numeric value, which can be filled in from the regression output, with a standard error corresponding to the one reported for the interaction term in the regression results. This effect is directly attributable to the treated x post\_treatment coefficient, which captures the change in search behavior for high-earning colleges after the release of the Scorecard relative to low-earning ones.

**The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high-earning graduates by a coefficient of -0.0966, relative to what it did for colleges with low-earning graduates, with a standard error of 0.0054. This result comes from the treated x post\_treatment coefficient(s) in my regression.**