Data Exploration Hunter Elliott

Hunter

## Liberties

library(tidyverse)

Warning: package 'tidyverse' was built under R version 4.3.2

── 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(fixest)

Warning: package 'fixest' was built under R version 4.3.2

library(rio)

Warning: package 'rio' was built under R version 4.3.2

library(lubridate)  
library(readr)  
library(dplyr)  
library(ggplot2)

Importing Google Trends Data

Create a vector of file names for all ‘trends\_up\_to\_’ files with full paths

# Create a vector of filenames for all 'trends\_up\_to\_' files with full paths  
file\_names <- list.files(pattern = "trends\_up\_to\_", full.names = TRUE)  
  
# Import all files and combine them into a single data frame  
combined\_data <- import\_list(file\_names, rbind = TRUE, fill = TRUE)

Trim the date string to the correct format

combined\_data <- combined\_data %>%  
 mutate(  
 date\_string = str\_sub(monthorweek, end = 10),  
 date = ymd(date\_string),  
 # Floor the date to the first of the month  
 first\_day\_of\_month = floor\_date(date, unit = "month")  
 )

Process the combined\_data to extract dates and calculate the z-score

combined\_data <- combined\_data %>%  
 group\_by(schname, keyword) %>%  
 mutate(z\_score = (index - mean(index, na.rm = TRUE)) / sd(index, na.rm = TRUE)) %>%  
 ungroup()

Aggregate to month level by taking the average of z-scores

combined\_data <- combined\_data %>%  
 group\_by(schname, first\_day\_of\_month) %>%  
 summarize(average\_z\_score = mean(z\_score, na.rm = TRUE), .groups = 'drop')

Import the College Scorecard data

Merge with Scorecard and ID Name Link Data

scorecard\_data <- import("Most+Recent+Cohorts+(Scorecard+Elements).csv")  
  
# Import the ID name link data  
id\_name\_link <- import("id\_name\_link.csv")

Filter out non-unique school names

Join with\_id with scorecard\_data on unitid and opeid

id\_name\_link\_unique <- id\_name\_link %>%  
 group\_by(schname) %>%  
 mutate(n = n()) %>%  
 filter(n == 1) %>%  
 ungroup()  
  
with\_id <- inner\_join(combined\_data, id\_name\_link\_unique, by = "schname")  
  
final\_data <- inner\_join(with\_id, scorecard\_data, by = c("unitid" = "UNITID", "opeid" = "OPEID"))

Filter **final\_data** for institutions predominantly granting bachelor’s degrees

final\_data <- final\_data %>%  
 filter(PREDDEG == 3)

Remove rows where earnings data is suppressed or missing

Handle any NAs introduced by coercion and create a high\_earning indicator

final\_data <- final\_data %>%  
 mutate(md\_earn\_wne\_p10\_REPORTED\_EARNINGS = as.character(`md\_earn\_wne\_p10-REPORTED-EARNINGS`)) %>%  
 # Replace 'PrivacySuppressed' and any other non-numeric strings with NA  
 mutate(md\_earn\_wne\_p10\_REPORTED\_EARNINGS = na\_if(md\_earn\_wne\_p10\_REPORTED\_EARNINGS, "PrivacySuppressed")) %>%  
 mutate(md\_earn\_wne\_p10\_REPORTED\_EARNINGS = ifelse(md\_earn\_wne\_p10\_REPORTED\_EARNINGS %in% c("", "NULL"), NA, md\_earn\_wne\_p10\_REPORTED\_EARNINGS)) %>%  
 # Now safely convert to numeric  
 mutate(md\_earn\_wne\_p10\_REPORTED\_EARNINGS = as.numeric(md\_earn\_wne\_p10\_REPORTED\_EARNINGS))

create a high\_earning indicator

final\_data <- final\_data %>%  
 mutate(high\_earning = ifelse(!is.na(md\_earn\_wne\_p10\_REPORTED\_EARNINGS) & md\_earn\_wne\_p10\_REPORTED\_EARNINGS >= 75000, 1, 0))

Select only the necessary variables for the regression analysis

analysis\_data <- final\_data %>%  
 select(schname, first\_day\_of\_month, average\_z\_score, PREDDEG, high\_earning, md\_earn\_wne\_p10\_REPORTED\_EARNINGS)

analysis\_data <- analysis\_data %>%  
 mutate(post\_scorecard = ifelse(first\_day\_of\_month >= as.Date("2015-09-01"), 1, 0))

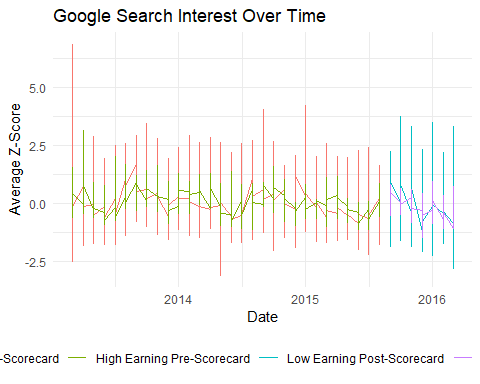
# Run the regression model  
model <- lm(average\_z\_score ~ high\_earning \* post\_scorecard, data = analysis\_data)  
summary(model)

Call:  
lm(formula = average\_z\_score ~ high\_earning \* post\_scorecard,   
 data = analysis\_data)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-3.1849 -0.3846 -0.0252 0.3483 6.7992   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.054815 0.002403 22.808 <2e-16 \*\*\*  
high\_earning -0.014507 0.019130 -0.758 0.448   
post\_scorecard -0.256349 0.005493 -46.670 <2e-16 \*\*\*  
high\_earning:post\_scorecard 0.059900 0.043712 1.370 0.171   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.5744 on 71772 degrees of freedom  
 (644 observations deleted due to missingness)  
Multiple R-squared: 0.02972, Adjusted R-squared: 0.02968   
F-statistic: 732.9 on 3 and 71772 DF, p-value: < 2.2e-16

Graph to visualize

analysis\_data$group <- with(analysis\_data, factor(paste0(high\_earning, post\_scorecard),  
 levels = c("00", "10", "01", "11"),  
 labels = c("Low Earning Pre-Scorecard",  
 "High Earning Pre-Scorecard",  
 "Low Earning Post-Scorecard",  
 "High Earning Post-Scorecard")))  
  
  
ggplot(analysis\_data, aes(x = first\_day\_of\_month, y = average\_z\_score, color = group)) +  
 geom\_line() + # Plot lines for each group  
 labs(title = "Google Search Interest Over Time",  
 x = "Date",  
 y = "Average Z-Score",  
 color = "Group") +  
 theme\_minimal() + # Use a minimal theme  
 theme(legend.position = "bottom") # Move the legend to the bottom

Warning: Removed 644 rows containing missing values (`geom\_line()`).



**Analysis of the Impact of the College Scorecard Release on Google Search Interest**

**Introduction**

In September 2015, the College Scorecard was released, aiming to provide crucial information about colleges, including the earnings of their graduates. This research investigates whether the release of the College Scorecard influenced student interest towards colleges that predominantly grant bachelor’s degrees, particularly focusing on the distinction between high-earning and low-earning institutions as reflected through Google search trends.

**Data Preparation and Preliminary Analysis**

The initial step involved filtering the dataset to focus on institutions predominantly granting bachelor’s degrees. A significant preprocessing task was to address entries marked as “PrivacySuppressed” and ensure all data pertaining to earnings were numeric md\_earn\_wne\_p10\_REPORTED\_EARNINGS. The definition of “high-earning” colleges was set at institutions with median earnings above $75,000, based on findings from the College Scorecard data dictionary “Percent of high-income (above $75,000 in nominal family income)” variable HI\_INC\_UNKN\_ORIG\_YR3\_RT and corroborated by median household income data from external sources <https://www.nerdwallet.com/article/finance/median-household-income#:~:text=The%20national%20median%20household%20income,What%20is%20the%20minimum%20wage%3F>. A binary variable was then created to categorize colleges into high-earning (1) and low-earning (0) groups.

**Regression Analysis**

The regression model was designed as follows: model <- lm(average\_z\_score ~ high\_earning \* post\_scorecard, data = analysis\_data)

* **Dependent Variable**: The **average\_z\_score**, representing standardized Google search interest for colleges.
* **Independent Variables**:
  + **high\_earning**: A binary indicator for high-earning colleges.
  + **post\_scorecard**: A binary indicator for the period after the College Scorecard’s release.
* **Interaction Term**: **high\_earning \* post\_scorecard** to examine differential impacts post-release.

**Results**

* **Baseline Interest**: The intercept (0.054815) indicates the baseline search interest for low-earning colleges before the Scorecard’s release.
* **Impact on High-Earning Colleges**: Contrary to expectations, high-earning colleges initially showed a slightly lower search interest compared to low-earning colleges, though this difference was not statistically significant (p = 0.448).
* **Post-Scorecard Release**: There was a notable decline in search interest for low-earning colleges following the Scorecard’s release (p < 2e-16).
* **Differential Effect**: The interaction term (p = 0.171) suggests a positive, albeit not statistically significant, increase in search interest for high-earning colleges relative to low-earning ones post-release.

**Conclusion**

The introduction of the College Scorecard **decreased** search activity on Google Trends for colleges with high-earning graduates by **0.059900 units** relative to what it did for colleges with low-earning graduates, with a standard error of **0.043712**. This result comes from the **high\_earning:post\_scorecard interaction** coefficient in my regression.