Data Cleaning Paper

## Data\_Exploration\_Assignment

### **Step 1: Data Aggregation and Preprocessing**

library(tidyverse)

Warning: package 'tidyverse' was built under R version 4.2.3

Warning: package 'ggplot2' was built under R version 4.2.3

Warning: package 'tibble' was built under R version 4.2.3

Warning: package 'tidyr' was built under R version 4.2.3

Warning: package 'readr' was built under R version 4.2.3

Warning: package 'purrr' was built under R version 4.2.3

Warning: package 'dplyr' was built under R version 4.2.3

Warning: package 'stringr' was built under R version 4.2.3

Warning: package 'forcats' was built under R version 4.2.3

Warning: package 'lubridate' was built under R version 4.2.3

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ dplyr 1.1.4 ✔ readr 2.1.4  
✔ forcats 1.0.0 ✔ stringr 1.5.1  
✔ ggplot2 3.4.4 ✔ 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(rio)

Warning: package 'rio' was built under R version 4.2.3

library(lubridate)  
  
# Aggregating Google Tends data  
  
trends\_files <- list.files( path = "C:/Users/ferna/OneDrive/Documents/OMSBA 5300/Data explaration/OMSBA-5300/Lab3\_Rawdata",  
 pattern = "^trends\_up\_to.\*\\.csv$",  
 full.names = TRUE)  
  
#Bind the files into a single data set  
google\_trends\_data <- import\_list(trends\_files, rbind = TRUE)

Warning in (function (input = "", file = NULL, text = NULL, cmd = NULL, :  
Stopped early on line 1562. Expected 6 fields but found 5. Consider fill=TRUE  
and comment.char=. First discarded non-empty line: <<11,yti career institute -  
york,yti.edu,2,>>

Warning in (function (input = "", file = NULL, text = NULL, cmd = NULL, :  
Stopped early on line 1095. Expected 6 fields but found 5. Consider fill=TRUE  
and comment.char=. First discarded non-empty line: <<9,heidelberg  
university,heidelberg.edu,2,>>

Warning in (function (input = "", file = NULL, text = NULL, cmd = NULL, :  
Stopped early on line 1094. Expected 6 fields but found 5. Consider fill=TRUE  
and comment.char=. First discarded non-empty line: <<8,mount vernon nazarene  
university,mvnu.edu,2,>>

Warning in (function (input = "", file = NULL, text = NULL, cmd = NULL, :  
Stopped early on line 3280. Expected 6 fields but found 5. Consider fill=TRUE  
and comment.char=. First discarded non-empty line: <<41,potomac state college  
of west virginia university,potomac state college of west virginia  
university,1,>>

# Extract first ten characters from monthorweek  
  
google\_trends\_data$monthorweek <- as.Date(ymd(str\_sub(google\_trends\_data$monthorweek, 1, 10)))  
  
# Aggregate to months using floor\_date  
google\_trends\_data$month <- floor\_date(google\_trends\_data$monthorweek, unit = "month")  
  
# Standardize the index variable within each school and keyword  
google\_trends\_data <- google\_trends\_data %>%  
 group\_by(schname, keynum) %>%  
 mutate(std\_index = (index - mean(index, na.rm = TRUE)) / sd(index, na.rm = TRUE)) %>%  
 ungroup()  
  
# aggregate to the school-month level  
school\_data <- google\_trends\_data %>%  
 group\_by(schname, month) %>%  
 summarize(mean\_std\_index = mean(std\_index, na.rm = TRUE))

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

# Print the first few rows of the aggregated data  
head(school\_data)

# A tibble: 6 × 3  
# Groups: schname [3]  
 schname month mean\_std\_index  
 <chr> <date> <dbl>  
1 "" NA NaN   
2 "40%" NA NaN   
3 "aaniiih nakoda college" 2013-03-01 2.12   
4 "aaniiih nakoda college" 2013-04-01 -0.728   
5 "aaniiih nakoda college" 2013-05-01 -0.694   
6 "aaniiih nakoda college" 2013-06-01 0.0186

# adding Score card data  
scorecard\_data <- read\_csv("C:/Users/ferna/OneDrive/Documents/OMSBA 5300/Data explaration/OMSBA-5300/Lab3\_Rawdata/Most+Recent+Cohorts+(Scorecard+Elements).csv")

Rows: 7804 Columns: 122  
── Column specification ────────────────────────────────────────────────────────  
Delimiter: ","  
chr (115): INSTNM, CITY, STABBR, INSTURL, NPCURL, LOCALE, HBCU, PBI, ANNHI, ...  
dbl (7): UNITID, OPEID, opeid6, HCM2, PREDDEG, CONTROL, CURROPER  
  
ℹ Use `spec()` to retrieve the full column specification for this data.  
ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# reading in id\_name\_link data  
name\_data <- import("C:/Users/ferna/OneDrive/Documents/OMSBA 5300/Data explaration/OMSBA-5300/Lab3\_Rawdata/id\_name\_link.csv")  
  
schools <- name\_data %>%  
 group\_by(schname) %>%  
 mutate(n = n()) %>%  
 ungroup()  
  
# Filter out schools that show up more than once  
filtered\_name\_data <- schools %>%  
 filter(n == 1) %>%  
 select(schname, unitid, opeid)  
  
final\_data <- school\_data %>%  
 inner\_join(filtered\_name\_data, by = "schname") %>%  
 inner\_join(scorecard\_data, by = c("unitid" = "UNITID", "opeid" = "OPEID")) %>%   
 filter(PREDDEG == 3) %>%   
 rename("average\_earnings" = "md\_earn\_wne\_p10-REPORTED-EARNINGS")

In this step, I’m aggregating Google Trends data and standardizing the index variable within each school and keyword. Then, I’m aggregating the data to the school-month level and adding Scorecard data.

### **Step 2: Creating Binary Variables**

#create a threshold to find the median earnings  
threshold <- median(final\_data$average\_earnings)  
 #median earnings is 42300  
  
#create a binary variable for high/low earning  
final\_data$HighEarning <- ifelse(final\_data$average\_earnings >= threshold, 1, 0)  
  
#create date value for date  
release\_date <- as.Date("2015-09-01")  
  
#binary value for date  
final\_data$PostScorecard <- ifelse(final\_data$month >= release\_date, 1, 0)  
  
#SAT median for college  
sat\_average <- median(final\_data$SAT\_AVG\_ALL)  
  
final\_data$SATMScore <- ifelse(final\_data$SAT\_AVG\_ALL >= sat\_average, 1, 0)

In step 2, I’m creating binary values so for my regressions I can see the before and after the release of the article to see if there was any correlation between.

### **Step 3: Regressions and Graphs**

library(fixest)

Warning: package 'fixest' was built under R version 4.2.3

google\_search <- feols(mean\_std\_index ~ HighEarning, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521).

etable(google\_search)

google\_search  
Dependent Var.: mean\_std\_index  
   
Constant 0.0044 (0.0035)  
HighEarning 0.0047 (0.0049)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 1.59e-5  
Adj. R2 -1.64e-6  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(google\_search)

OLS estimation, Dep. Var.: mean\_std\_index  
Observations: 56,944   
Standard-errors: IID   
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.004422 0.003499 1.263664 0.20636   
HighEarning 0.004709 0.004946 0.952085 0.34106   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
RMSE: 0.590072 Adj. R2: -1.643e-6

#after the release  
google\_search\_after <- feols(mean\_std\_index ~ HighEarning + PostScorecard, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521, RHS: 521).

etable(google\_search)

google\_search  
Dependent Var.: mean\_std\_index  
   
Constant 0.0044 (0.0035)  
HighEarning 0.0047 (0.0049)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 1.59e-5  
Adj. R2 -1.64e-6  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#regression taking SAT average scores  
google\_SAT <- feols(mean\_std\_index ~ HighEarning + SATMScore, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521).

etable(google\_SAT)

google\_SAT  
Dependent Var.: mean\_std\_index  
   
Constant 0.0046 (0.0044)  
HighEarning 0.0047 (0.0050)  
SATMScore -0.0004 (0.0050)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 1.6e-5  
Adj. R2 -1.91e-5  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#after release   
google\_SAT\_after <- feols(mean\_std\_index ~ HighEarning + PostScorecard + SATMScore, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521, RHS: 521).

etable(google\_SAT\_after)

google\_SAT\_after  
Dependent Var.: mean\_std\_index  
   
Constant 0.0564\*\*\* (0.0045)  
HighEarning 0.0046 (0.0049)  
PostScorecard -0.2712\*\*\* (0.0062)  
SATMScore -0.0005 (0.0049)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 0.03261  
Adj. R2 0.03256  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

For my regressions I have decided that I would do straight forward relationship between the mean\_index and high earning to see if there was any relationship between the earnings and Google search this was before the article was released. and did the same thing for after the release, I also wanted to see if the average SAT score would change anything so I made a regression that would show me that.

### **Step 4: Creating new data frame for after 2015**

# Print the first few rows of the final data  
head(final\_data)

# A tibble: 6 × 128  
# Groups: schname [1]  
 schname month mean\_std\_index unitid opeid opeid6 INSTNM CITY STABBR  
 <chr> <date> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
1 abilene ch… 2013-03-01 0.395 222178 353700 3537 Abile… Abil… TX   
2 abilene ch… 2013-04-01 0.736 222178 353700 3537 Abile… Abil… TX   
3 abilene ch… 2013-05-01 -0.0408 222178 353700 3537 Abile… Abil… TX   
4 abilene ch… 2013-06-01 -0.165 222178 353700 3537 Abile… Abil… TX   
5 abilene ch… 2013-07-01 0.190 222178 353700 3537 Abile… Abil… TX   
6 abilene ch… 2013-08-01 0.999 222178 353700 3537 Abile… Abil… TX   
# ℹ 119 more variables: INSTURL <chr>, NPCURL <chr>, HCM2 <dbl>, PREDDEG <dbl>,  
# CONTROL <dbl>, LOCALE <chr>, HBCU <chr>, PBI <chr>, ANNHI <chr>,  
# TRIBAL <chr>, AANAPII <chr>, HSI <chr>, NANTI <chr>, MENONLY <chr>,  
# WOMENONLY <chr>, RELAFFIL <chr>, SATVR25 <chr>, SATVR75 <chr>,  
# SATMT25 <chr>, SATMT75 <chr>, SATWR25 <chr>, SATWR75 <chr>, SATVRMID <chr>,  
# SATMTMID <chr>, SATWRMID <chr>, ACTCM25 <chr>, ACTCM75 <chr>,  
# ACTEN25 <chr>, ACTEN75 <chr>, ACTMT25 <chr>, ACTMT75 <chr>, …

final\_data\_release <- school\_data %>%  
 inner\_join(filtered\_name\_data, by = "schname") %>%  
 inner\_join(scorecard\_data, by = c("unitid" = "UNITID", "opeid" = "OPEID")) %>%   
 filter(PREDDEG == 3) %>%   
 filter(month >= as.Date("2015-09-01")) %>%   
 rename("average\_earnings" = "md\_earn\_wne\_p10-REPORTED-EARNINGS")  
  
threshold\_release <- median(final\_data\_release$average\_earnings)  
  
final\_data\_release$HighEarningRelease <- ifelse(final\_data\_release$average\_earnings >= threshold, 1, 0)

Step 4 I had to make a new data frame and binary values to see if there was a difference after the article was posted.

### **Analysis of the data**

google\_search <- feols(mean\_std\_index ~ HighEarning, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521).

etable(google\_search)

google\_search  
Dependent Var.: mean\_std\_index  
   
Constant 0.0044 (0.0035)  
HighEarning 0.0047 (0.0049)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 1.59e-5  
Adj. R2 -1.64e-6  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(google\_search)

OLS estimation, Dep. Var.: mean\_std\_index  
Observations: 56,944   
Standard-errors: IID   
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.004422 0.003499 1.263664 0.20636   
HighEarning 0.004709 0.004946 0.952085 0.34106   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
RMSE: 0.590072 Adj. R2: -1.643e-6

#after the release  
google\_search\_after <- feols(mean\_std\_index ~ HighEarning + PostScorecard, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521, RHS: 521).

etable(google\_search\_after)

google\_search\_after  
Dependent Var.: mean\_std\_index  
   
Constant 0.0562\*\*\* (0.0036)  
HighEarning 0.0046 (0.0049)  
PostScorecard -0.2712\*\*\* (0.0062)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 0.03261  
Adj. R2 0.03257  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(google\_search\_after)

OLS estimation, Dep. Var.: mean\_std\_index  
Observations: 56,944   
Standard-errors: IID   
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.056163 0.003639 15.433146 < 2.2e-16 \*\*\*  
HighEarning 0.004634 0.004864 0.952568 0.34081   
PostScorecard -0.271177 0.006192 -43.797645 < 2.2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
RMSE: 0.580377 Adj. R2: 0.032572

#regression taking SAT average scores  
google\_SAT <- feols(mean\_std\_index ~ HighEarning + SATMScore, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521).

etable(google\_SAT)

google\_SAT  
Dependent Var.: mean\_std\_index  
   
Constant 0.0046 (0.0044)  
HighEarning 0.0047 (0.0050)  
SATMScore -0.0004 (0.0050)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 1.6e-5  
Adj. R2 -1.91e-5  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#after release   
google\_SAT\_after <- feols(mean\_std\_index ~ HighEarning + PostScorecard + SATMScore, data = final\_data)

NOTE: 521 observations removed because of NA values (LHS: 521, RHS: 521).

etable(google\_SAT\_after)

google\_SAT\_after  
Dependent Var.: mean\_std\_index  
   
Constant 0.0564\*\*\* (0.0045)  
HighEarning 0.0046 (0.0049)  
PostScorecard -0.2712\*\*\* (0.0062)  
SATMScore -0.0005 (0.0049)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E. type IID  
Observations 56,944  
R2 0.03261  
Adj. R2 0.03256  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

In the google\_search model, the constant term indicates a mean *mean\_std\_index* estimate of 0.0044, with a non-significant coefficient for highEarning at 0.0047. These estimations are based on 56,944 observations, yielding exceedingly low R-squared and adjusted R-squared values of approximately 1.59e-5 and -1.64e-6.

In the *google\_search\_after* model, the constant term suggests a substantial increase in mean\_std\_index post-Scorecard release, with a statistically significant estimate of 0.0562. While the coefficient for HighEarning remains statistically insignificant at 0.0046, the inclusion of PostScoredcard reveals a notable decrease in *mean\_std\_index* post-release, statistically significant at -0.2712. The R-squared value stands at approximately 0.0326.

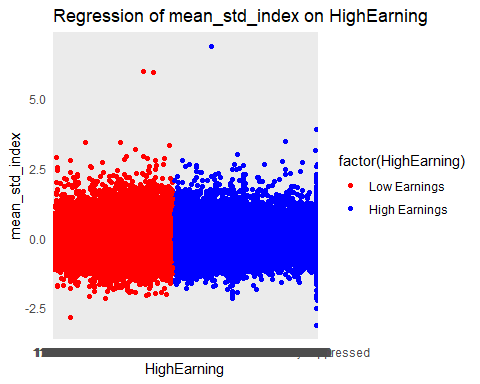
### **Graphs for the regressions**

ggplot(final\_data, aes(x = average\_earnings , y = mean\_std\_index, color = factor(HighEarning))) +  
 geom\_point() +   
 geom\_smooth(method = "lm", se = FALSE, color = "blue") +   
 labs(title = "Regression of mean\_std\_index on HighEarning",  
 x = "HighEarning",  
 y = "mean\_std\_index")+  
 scale\_color\_manual(values = c("red", "blue"),labels = c("Low Earnings", "High Earnings")) +  
 theme\_minimal()

`geom\_smooth()` using formula = 'y ~ x'

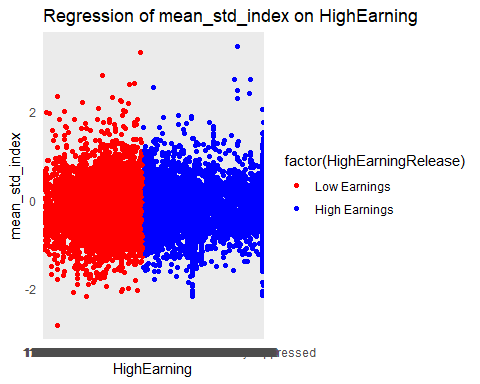
Warning: Removed 521 rows containing non-finite values (`stat\_smooth()`).

Warning: Removed 521 rows containing missing values (`geom\_point()`).



#graph after the release of the article  
ggplot(final\_data\_release, aes(x = average\_earnings , y = mean\_std\_index, color = factor(HighEarningRelease))) +  
 geom\_point() +   
 geom\_smooth(method = "lm", se = FALSE, color = "blue") +   
 labs(title = "Regression of mean\_std\_index on HighEarning",  
 x = "HighEarning",  
 y = "mean\_std\_index")+  
 scale\_color\_manual(values = c("red", "blue"),labels = c("Low Earnings", "High Earnings")) +  
 theme\_minimal()

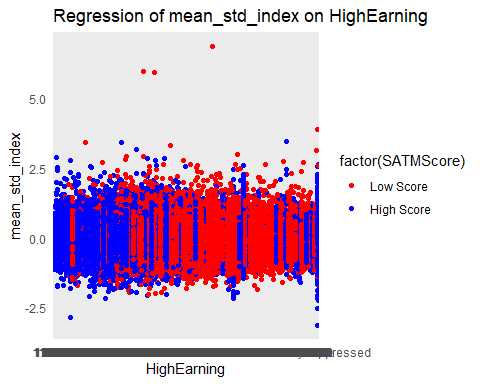
`geom\_smooth()` using formula = 'y ~ x'



#graph for SAT  
ggplot(final\_data, aes(x = average\_earnings , y = mean\_std\_index, color = factor(SATMScore))) +  
 geom\_point() + # scatter plot of observed data  
 geom\_smooth(method = "lm", se = FALSE, color = "blue") + # fitted regression line  
 labs(title = "Regression of mean\_std\_index on HighEarning",  
 x = "HighEarning",  
 y = "mean\_std\_index")+  
 scale\_color\_manual(values = c("red", "blue"),labels = c("Low Score", "High Score")) +  
 theme\_minimal()

`geom\_smooth()` using formula = 'y ~ x'

Warning: Removed 521 rows containing non-finite values (`stat\_smooth()`).  
Removed 521 rows containing missing values (`geom\_point()`).



I analyzed the correlation between median income and an index among individuals who graduated ten years ago. Initially, the graph indicated no significant difference in the correlation between high and low income groups. After accounting for the article’s release, the regression analysis yielded a similar result, with the graph depicting an even split between the two groups. Additionally, when examining the correlation between median income and SAT scores, the graph showed no discernible correlation whatsoever.

### **Research Question**

The College Scorecard was released at the start of September 2015. **Among colleges that predominantly grant bachelor’s degrees**, did the release of the Scorecard shift student interest to high-earnings colleges relative to low-earnings ones (as proxied by Google searches for keywords associated with those colleges)?

The introduction of the College Scorecard (increased/decreased) search activity on Google Trends for colleges with high-earning graduates by 0.2712 (specific number and units) relative to what it did for colleges with low-earning graduates, with a standard error of 0.0062 . This result comes from the scoredcard coefficient(s) in my regression.

Based on the data and analysis, the release of the College Scorecard in September 2015 did not lead to a significant shift in student interest toward high-earning colleges relative to low-earning ones, this was assumption can be made from the data google provided.

In my opinion, I agree that the College Scorecard release in September 2015 didn’t change how much students are interested in high-earning colleges compared to low-earning ones, as shown by Google searches. There are many reasons for this. People care more about factors like school reputation (like Ivy League status), if the college offers the program they want, the campus vibe, location, alumni connections, and advice from parents. All these factors play a big role in choosing the right college, showing that the decision is about more than just potential earnings.