Midterm Project - Data Analysis

Shreeya Sampat

# Load necessary libraries  
library(rio)  
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(stringr)  
library(ggplot2)  
library(fixest)  
library(broom)  
library(forcats)

# Load cleaned data from the first QMD output  
trends\_agg <- rio::import("clean\_trends\_month\_school.rds")%>%  
 dplyr::mutate(schname = stringr::str\_squish(stringr::str\_to\_lower(schname)))

Warning: Missing `trust` will be set to FALSE by default for RDS in 2.0.0.

# Reading in the Scorecard data  
  
# You can just use import() to read in the Scorecard data ('Most+Recent+Cohorts+(Scorecard+Elements)'). The dictionary file you can also read in if you like, although you're probably just better off opening that in Excel to read it - you're going to use that to figure out what all the variables are but you're not going to use the dictionary in analysis.  
# You can also use import() to read in the id\_name\_link file.  
  
# Read in the Scorecard Data  
scorecard <- import("Lab3\_Rawdata/Most+Recent+Cohorts+(Scorecard+Elements).csv")  
  
# Read in the id\_name\_link.csv File  
id\_link <- import("Lab3\_Rawdata/id\_name\_link.csv")  
  
glimpse(id\_link)

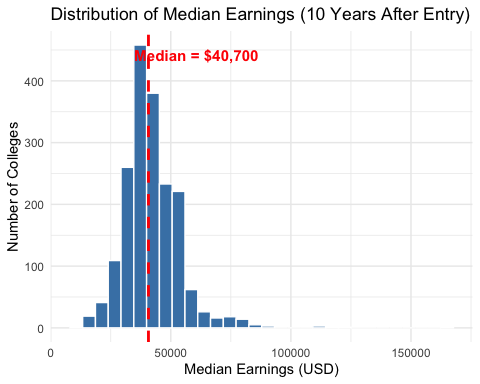
Rows: 3,595  
Columns: 3  
$ unitid <int> 180203, 222178, 138558, 172866, 412173, 108232, 475635, 126182…  
$ opeid <int> 2517500, 353700, 154100, 2050300, 3346300, 753100, 4185500, 13…  
$ schname <chr> "aaniiih nakoda college", "abilene christian university", "abr…

# Merge in the Scorecard data  
  
# Normalize keys on the Trends side  
trends\_agg <- trends\_agg %>%  
 dplyr::mutate(schname = stringr::str\_squish(stringr::str\_to\_lower(schname)))  
  
# First, use group\_by() and mutate(n = n()) to count how many times each school name pops up in id\_name\_link,  
# and then filter to get rid of any school names that show up more than once  
id\_link\_clean <- id\_link %>%  
 dplyr::mutate(schname = stringr::str\_squish(stringr::str\_to\_lower(schname))) %>%  
 dplyr::group\_by(schname) %>%  
 dplyr::filter(dplyr::n\_distinct(unitid) == 1) %>%  
 dplyr::distinct(schname, unitid, .keep\_all = FALSE) %>%  
 dplyr::ungroup()  
  
# Prep Scorecard   
# Keep bachelor focused and only needed columns  
# If the earnings column name differs, use grep to locate it  
earn\_col <- grep("md\_earn.\*p10|earn.\*p10", names(scorecard),  
 ignore.case = TRUE, value = TRUE)[1]  
  
scorecard\_clean <- scorecard %>%  
 dplyr::mutate(  
 UNITID = as.integer(UNITID),  
 PREDDEG = suppressWarnings(as.numeric(PREDDEG))  
 ) %>%  
 dplyr::filter(PREDDEG == 3) %>% # BA-focused  
 dplyr::distinct(UNITID, .keep\_all = TRUE) %>%  
 dplyr::select(UNITID, INSTNM, PREDDEG, dplyr::all\_of(earn\_col)) %>%  
 dplyr::mutate(earnings\_num = suppressWarnings(as.numeric(.data[[earn\_col]])))  
  
# Global median cutoff (this is what defines high vs. low)  
median\_cutoff <- median(scorecard\_clean$earnings\_num, na.rm = TRUE)  
cat("Global median 10-year earnings cutoff: $", round(median\_cutoff, 0), "\n", sep = "")

Global median 10-year earnings cutoff: $40700

# Histogram with median label moved to whitespace  
ggplot(scorecard\_clean, aes(x = earnings\_num)) +  
 geom\_histogram(fill = "steelblue", color = "white", bins = 30) +  
 geom\_vline(xintercept = median\_cutoff, linetype = "dashed", color = "red", linewidth = 1) +  
 annotate("text",  
 x = median\_cutoff + 20000,  
 y = max(table(cut(scorecard\_clean$earnings\_num, breaks = 30))) + 10,  
 label = paste0("Median = $", format(round(median\_cutoff, 0), big.mark = ",")),  
 color = "red", size = 4, fontface = "bold") +  
 labs(title = "Distribution of Median Earnings (10 Years After Entry)",  
 x = "Median Earnings (USD)", y = "Number of Colleges") +  
 theme\_minimal()

Warning: Removed 260 rows containing non-finite outside the scale range  
(`stat\_bin()`).



# Join Trends to id link, then to Scorecard  
# (inner\_join will drop any rows that don't find a match, which is fine for this project. The other \_join functions behave differently about non-matches)  
trends\_with\_unitid <- trends\_agg %>%  
 dplyr::inner\_join(id\_link\_clean, by = "schname")  
  
full\_data <- trends\_with\_unitid %>%  
 dplyr::inner\_join(scorecard\_clean, by = c("unitid" = "UNITID")) %>%  
 dplyr::mutate(earnings = suppressWarnings(as.numeric(.data[[earn\_col]]))) %>%  
 dplyr::filter(!is.na(earnings)) %>%  
 dplyr::mutate(  
 high\_earning = dplyr::if\_else(earnings >= median\_cutoff, 1L, 0L),  
 post\_scorecard = dplyr::if\_else(month >= as.Date("2015-09-01"), 1L, 0L)  
 )  
  
# Quick check  
cat("Rows after join to id link:", nrow(trends\_with\_unitid), "\n")

Rows after join to id link: 90082

cat("Rows after join to scorecard:", nrow(full\_data), "\n")

Rows after join to scorecard: 51565

glimpse(full\_data)

Rows: 51,565  
Columns: 11  
$ schname <chr> "abilene christian university", "a…  
$ month <date> 2013-03-01, 2013-04-01, 2013-05-0…  
$ avg\_std\_index <dbl> 0.215508169, 0.389866232, -0.02332…  
$ unitid <int> 222178, 222178, 222178, 222178, 22…  
$ INSTNM <chr> "Abilene Christian University", "A…  
$ PREDDEG <dbl> 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3…  
$ `md\_earn\_wne\_p10-REPORTED-EARNINGS` <chr> "40200", "40200", "40200", "40200"…  
$ earnings\_num <dbl> 40200, 40200, 40200, 40200, 40200,…  
$ earnings <dbl> 40200, 40200, 40200, 40200, 40200,…  
$ high\_earning <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…  
$ post\_scorecard <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…

# Build clean analysis set  
clean\_data <- full\_data %>%  
 filter(!is.na(avg\_std\_index)) %>%  
 group\_by(schname) %>%  
 mutate(  
 pre\_score = sum(post\_scorecard == 0, na.rm = TRUE),  
 post\_score = sum(post\_scorecard == 1, na.rm = TRUE)  
 ) %>%  
 # keep schools with at least six months pre score and six months post score  
 filter(pre\_score >= 6, post\_score >= 6) %>%  
 ungroup() %>%  
 select(schname, month, avg\_std\_index, high\_earning, post\_scorecard)

# Two way fixed effects DID  
did\_mod <- feols(  
 avg\_std\_index ~ post\_scorecard \* high\_earning | schname + month,  
 cluster = ~ schname,  
 data = clean\_data  
)

The variables 'post\_scorecard' and 'high\_earning' have been removed because of collinearity (see $collin.var).

summary(did\_mod)

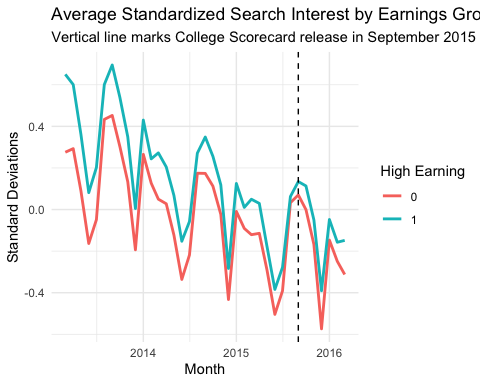
OLS estimation, Dep. Var.: avg\_std\_index  
Observations: 51,565  
Fixed-effects: schname: 1,404, month: 37  
Standard-errors: Clustered (schname)   
 Estimate Std. Error t value Pr(>|t|)   
post\_scorecard:high\_earning -0.057356 0.018864 -3.0406 0.0024048 \*\*   
... 2 variables were removed because of collinearity (post\_scorecard and high\_earning)  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
RMSE: 0.345572 Adj. R2: 0.729342  
 Within R2: 0.001059

att <- tidy(did\_mod) %>%  
 filter(term == "post\_scorecard:high\_earning") %>%  
 transmute(  
 direction = ifelse(estimate > 0, "increased", "decreased"),  
 estimate = round(estimate, 3),  
 se = round(std.error, 3)  
 )  
  
cat("The introduction of the College Scorecard", att$direction,"search activity on Google Trends for colleges with high-earning graduates by",att$estimate, "standard deviations relative to what it did for colleges with low-earning graduates,","with a standard error of", att$se,". This result comes from the post\_scorecard × high\_earning coefficient in my regression.")

The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high-earning graduates by -0.057 standard deviations relative to what it did for colleges with low-earning graduates, with a standard error of 0.019 . This result comes from the post\_scorecard × high\_earning coefficient in my regression.

# Visualizations  
# Visualize Search Interest Over Time by Earning Group  
ggplot(clean\_data, aes(x = month, y = avg\_std\_index, color = factor(high\_earning))) +  
 stat\_summary(fun = mean, geom = "line", size = 1) +  
 geom\_vline(xintercept = as.Date("2015-09-01"), linetype = "dashed") +  
 labs(  
 title = "Average Standardized Search Interest by Earnings Group",  
 subtitle = "Vertical line marks College Scorecard release in September 2015",  
 x = "Month",  
 y = "Standard Deviations",  
 color = "High Earning"  
 ) +  
 theme\_minimal()

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
ℹ Please use `linewidth` instead.



# Continuous earnings interaction  
earnings\_scaled <- scale(full\_data$earnings)  
cd\_cont <- clean\_data %>%  
 left\_join(  
 full\_data %>% select(schname, month, earnings) %>% mutate(ez = as.numeric(scale(earnings))),  
 by = c("schname","month"))  
  
did\_cont <- feols(  
 avg\_std\_index ~ post\_scorecard \* ez | schname + month,  
 cluster = ~ schname,  
 data = cd\_cont  
)

The variables 'post\_scorecard' and 'ez' have been removed because of collinearity (see $collin.var).

summary(did\_cont)

OLS estimation, Dep. Var.: avg\_std\_index  
Observations: 51,565  
Fixed-effects: schname: 1,404, month: 37  
Standard-errors: Clustered (schname)   
 Estimate Std. Error t value Pr(>|t|)   
post\_scorecard:ez -0.031287 0.00923 -3.38959 0.0007194 \*\*\*  
... 2 variables were removed because of collinearity (post\_scorecard and ez)  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
RMSE: 0.345537 Adj. R2: 0.729397  
 Within R2: 0.001264

# Quartiles of earnings instead of median split  
q <- quantile(full\_data$earnings, probs = c(.25, .5, .75), na.rm = TRUE)  
cd\_q <- full\_data %>%  
 mutate(  
 earn\_q = cut(earnings, breaks = c(-Inf, q[1], q[2], q[3], Inf), labels = c("Q1","Q2","Q3","Q4"))  
 ) %>%  
 semi\_join(clean\_data %>% distinct(schname, month), by = c("schname","month"))  
  
did\_q <- feols(  
 avg\_std\_index ~ post\_scorecard \* i(earn\_q, ref = "Q1") | schname + month,  
 cluster = ~ schname,  
 data = cd\_q  
)

The variables 'post\_scorecard', 'earn\_q::Q2', 'earn\_q::Q3' and 'earn\_q::Q4' have been removed because of collinearity (see $collin.var).

etable(did\_q)

did\_q  
Dependent Var.: avg\_std\_index  
   
post\_scorecard x earn\_q = Q2 0.0560\* (0.0235)  
post\_scorecard x earn\_q = Q3 0.0742\*\*\* (0.0219)  
post\_scorecard x earn\_q = Q4 -0.1585\*\*\* (0.0316)  
Fixed-Effects: -------------------  
schname Yes  
month Yes  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E.: Clustered by: schname  
Observations 51,565  
R2 0.73947  
Within R2 0.01080  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

clean\_data <- clean\_data %>%  
 mutate(  
 month\_num = month(month),   
 time\_index = as.integer(interval(min(month), month) %/% months(1))  
 )  
  
# Model A: Basic DID  
model\_a <- feols(  
 avg\_std\_index ~ post\_scorecard \* high\_earning,  
 data = clean\_data,  
 cluster = ~ schname  
)  
  
# Model B: DID + Seasonality  
# i(month\_num) adds 11 month indicators and leaves one out as reference  
model\_b <- feols(  
 avg\_std\_index ~ post\_scorecard \* high\_earning + i(month\_num),  
 data = clean\_data,  
 cluster = ~ schname  
)  
  
# Model C: DID + Seasonality + School Fixed Effects  
model\_c <- feols(  
 avg\_std\_index ~ post\_scorecard \* high\_earning + i(month\_num) | schname,  
 data = clean\_data,  
 cluster = ~ schname  
)

The variable 'high\_earning' has been removed because of collinearity (see $collin.var).

# Model D: Model C + Time Trend  
# This is the canonical two way fixed effects model for DID  
# Do not also include time\_index with month FE to avoid collinearity  
model\_d <- feols(  
 avg\_std\_index ~ post\_scorecard \* high\_earning | schname + month,  
 data = clean\_data,  
 cluster = ~ schname  
)

The variables 'post\_scorecard' and 'high\_earning' have been removed because of collinearity (see $collin.var).

# Model E: Linear Time Trend  
model\_e <- feols(  
 avg\_std\_index ~ post\_scorecard \* high\_earning + time\_index | schname,  
 data = clean\_data,  
 cluster = ~ schname  
)

The variable 'high\_earning' has been removed because of collinearity (see $collin.var).

# Comparison table  
etable(  
 list(  
 "Basic DID" = model\_a,  
 "Seasonality" = model\_b,  
 "FE School plus Seasonality" = model\_c,  
 "Two Way FE" = model\_d,  
 "Linear Time Trend" = model\_e  
 ),  
 dict = c(  
 "post\_scorecard" = "Post Scorecard",  
 "high\_earning" = "High Earning",  
 "post\_scorecard:high\_earning" = "DID Interaction",  
 "i(month\_num, ..1)" = "Month Dummies"  
 ),  
 se = "cluster",  
 cluster = ~ schname,  
 fitstat = ~ r2 + ar2 + n  
)

Basic DID Seasonality FE School plus Se..  
Dependent Var.: avg\_std\_index avg\_std\_index avg\_std\_index  
   
Constant -0.0068 (0.0208) 0.1113\*\*\* (0.0219)   
Post Scorecard -0.1903\*\*\* (0.0121) -0.2647\*\*\* (0.0117) -0.2645\*\*\* (0.0117)  
High Earning 0.1779\*\*\* (0.0280) 0.1777\*\*\* (0.0280)   
DID Interaction -0.0588\*\* (0.0189) -0.0586\*\* (0.0189) -0.0583\*\* (0.0189)  
month\_num = 2 -0.1228\*\*\* (0.0056) -0.1228\*\*\* (0.0056)  
month\_num = 3 -0.0568\*\*\* (0.0075) -0.0585\*\*\* (0.0074)  
month\_num = 4 -0.0260\*\*\* (0.0068) -0.0259\*\*\* (0.0068)  
month\_num = 5 -0.2110\*\*\* (0.0079) -0.2108\*\*\* (0.0079)  
month\_num = 6 -0.4433\*\*\* (0.0089) -0.4432\*\*\* (0.0089)  
month\_num = 7 -0.3322\*\*\* (0.0085) -0.3321\*\*\* (0.0085)  
month\_num = 8 0.0606\*\*\* (0.0073) 0.0607\*\*\* (0.0073)  
month\_num = 9 0.2103\*\*\* (0.0079) 0.2103\*\*\* (0.0079)  
month\_num = 10 0.1185\*\*\* (0.0071) 0.1185\*\*\* (0.0071)  
month\_num = 11 -0.0420\*\*\* (0.0069) -0.0420\*\*\* (0.0069)  
month\_num = 12 -0.4133\*\*\* (0.0071) -0.4133\*\*\* (0.0071)  
time\_index   
Fixed-Effects: ------------------- ------------------- -------------------  
schname No No Yes  
month No No No  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E.: Clustered by: schname by: schname by: schname  
R2 0.03218 0.11553 0.68683  
Adj. R2 0.03212 0.11529 0.67799  
Observations 51,565 51,565 51,565  
  
 Two Way FE Linear Time Trend  
Dependent Var.: avg\_std\_index avg\_std\_index  
   
Constant   
Post Scorecard 0.1888\*\*\* (0.0129)  
High Earning   
DID Interaction -0.0574\*\* (0.0189) -0.0576\*\* (0.0189)  
month\_num = 2   
month\_num = 3   
month\_num = 4   
month\_num = 5   
month\_num = 6   
month\_num = 7   
month\_num = 8   
month\_num = 9   
month\_num = 10   
month\_num = 11   
month\_num = 12   
time\_index -0.0207\*\*\* (0.0006)  
Fixed-Effects: ------------------ -------------------  
schname Yes Yes  
month Yes No  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
S.E.: Clustered by: schname by: schname  
R2 0.73690 0.66032  
Adj. R2 0.72934 0.65080  
Observations 51,565 51,565  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Tidy pull of the DID interaction from each model  
grab\_att <- function(mod) {  
 out <- broom::tidy(mod)  
 row <- dplyr::filter(out,  
 term %in% c("post\_scorecard:high\_earning",  
 "post\_scorecard x high\_earning"))  
 if (nrow(row) == 0) {  
 tibble::tibble(estimate = NA\_real\_, se = NA\_real\_)  
 } else {  
 tibble::tibble(estimate = row$estimate[1], se = row$std.error[1])  
 }  
}  
  
att\_tbl <- tibble::tibble(  
 model = c("A) Basic",  
 "B) Seasonality",  
 "C) FE School plus Seasonality",  
 "D) Two Way FE",  
 "E) Linear Time Trend")  
) |>  
 dplyr::bind\_cols(dplyr::bind\_rows(  
 grab\_att(model\_a),  
 grab\_att(model\_b),  
 grab\_att(model\_c),  
 grab\_att(model\_d),  
 grab\_att(model\_e)  
 )) |>  
 dplyr::mutate(  
 estimate = round(estimate, 3),  
 se = round(se, 3)  
 )  
  
print(att\_tbl)

# A tibble: 5 × 3  
 model estimate se  
 <chr> <dbl> <dbl>  
1 A) Basic -0.059 0.019  
2 B) Seasonality -0.059 0.019  
3 C) FE School plus Seasonality -0.058 0.019  
4 D) Two Way FE -0.057 0.019  
5 E) Linear Time Trend -0.058 0.019

model\_sentence <- function(name, est, se) {  
 direction <- ifelse(est > 0, "increased", "decreased")  
 est\_abs <- abs(est)  
 paste0(  
 name, ": The introduction of the College Scorecard ", direction,  
 " search activity on Google Trends for colleges with high earning graduates by ",  
 formatC(est\_abs, digits = 3, format = "f"), " standard deviations relative to what it did for colleges with low earning graduates, ",  
 "with a standard error of ", formatC(se, digits = 3, format = "f"),  
 ". This result comes from the post\_scorecard times high\_earning coefficient in my regression."  
 )  
}  
  
cat(  
 paste0(  
 apply(att\_tbl, 1, function(r) model\_sentence(  
 r[["model"]], as.numeric(r[["estimate"]]), as.numeric(r[["se"]])  
 )),  
 collapse = "\n\n"  
 ),  
 "\n\n"  
)

A) Basic: The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high earning graduates by 0.059 standard deviations relative to what it did for colleges with low earning graduates, with a standard error of 0.019. This result comes from the post\_scorecard times high\_earning coefficient in my regression.  
  
B) Seasonality: The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high earning graduates by 0.059 standard deviations relative to what it did for colleges with low earning graduates, with a standard error of 0.019. This result comes from the post\_scorecard times high\_earning coefficient in my regression.  
  
C) FE School plus Seasonality: The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high earning graduates by 0.058 standard deviations relative to what it did for colleges with low earning graduates, with a standard error of 0.019. This result comes from the post\_scorecard times high\_earning coefficient in my regression.  
  
D) Two Way FE: The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high earning graduates by 0.057 standard deviations relative to what it did for colleges with low earning graduates, with a standard error of 0.019. This result comes from the post\_scorecard times high\_earning coefficient in my regression.  
  
E) Linear Time Trend: The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high earning graduates by 0.058 standard deviations relative to what it did for colleges with low earning graduates, with a standard error of 0.019. This result comes from the post\_scorecard times high\_earning coefficient in my regression.

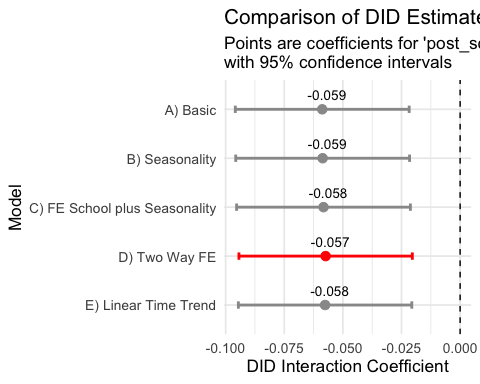
cat(  
 "Note: Linear Time Trend in Model E specification replaces month fixed effects with a linear time trend while keeping school fixed effects. The DID interaction is nearly identical to the two way fixed effects estimate, which supports validity of alternative time controls.\n\n"  
)

Note: Linear Time Trend in Model E specification replaces month fixed effects with a linear time trend while keeping school fixed effects. The DID interaction is nearly identical to the two way fixed effects estimate, which supports validity of alternative time controls.

att\_d <- dplyr::filter(att\_tbl, model == "D) Two Way FE")  
cat(  
 "Preferred specification: The introduction of the College Scorecard ",  
 ifelse(att\_d$estimate > 0, "increased", "decreased"), " search activity on Google Trends for colleges with high earning graduates by ",  
 formatC(abs(att\_d$estimate), digits = 3, format = "f"),  
 " standard deviations relative to low earning colleges, with a standard error of ",  
 formatC(att\_d$se, digits = 3, format = "f"),  
 ". This result comes from the post\_scorecard times high\_earning coefficient in the two way fixed effects model with school and month fixed effects.\n",  
 sep = ""  
)

Preferred specification: The introduction of the College Scorecard decreased search activity on Google Trends for colleges with high earning graduates by 0.057 standard deviations relative to low earning colleges, with a standard error of 0.019. This result comes from the post\_scorecard times high\_earning coefficient in the two way fixed effects model with school and month fixed effects.

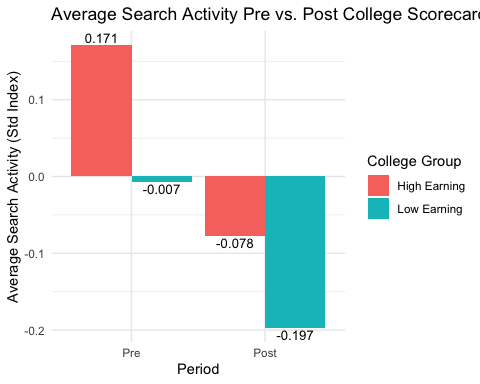
# Build the comparison data  
get\_att <- function(mod, label) {  
 tidy(mod) |>  
 filter(term %in% c("post\_scorecard:high\_earning",  
 "post\_scorecard x high\_earning")) |>  
 transmute(model = label, estimate, se = std.error)  
}  
  
att\_compare <- bind\_rows(  
 get\_att(model\_a, "A) Basic"),  
 get\_att(model\_b, "B) Seasonality"),  
 get\_att(model\_c, "C) FE School plus Seasonality"),  
 get\_att(model\_d, "D) Two Way FE"),  
 get\_att(model\_e, "E) Linear Time Trend")  
) |>  
 mutate(  
 lo = estimate - 1.96 \* se,  
 hi = estimate + 1.96 \* se,  
 label = sprintf("%.3f", estimate),  
 model = fct\_rev(factor(  
 model,  
 levels = c("A) Basic","B) Seasonality","C) FE School plus Seasonality",  
 "D) Two Way FE","E) Linear Time Trend")  
 )),  
 highlight = (model == "D) Two Way FE")  
 )  
  
# Plot with numbers, no zoom, highlight Model D  
model\_compare <- ggplot(att\_compare, aes(x = estimate, y = model)) +  
 geom\_errorbarh(aes(xmin = lo, xmax = hi, color = highlight), height = 0.15, size = 1) +  
 geom\_point(aes(color = highlight), size = 3) +  
 geom\_text(aes(label = label), vjust = -1, nudge\_x = 0.002, size = 3.6, color = "black") +  
 geom\_vline(xintercept = 0, linetype = "dashed") +  
 scale\_color\_manual(values = c(`TRUE` = "red", `FALSE` = "grey60"), guide = "none") +  
 labs(  
 title = "Comparison of DID Estimates Across Models",  
 subtitle = "Points are coefficients for 'post\_scorecard × high\_earning' \nwith 95% confidence intervals",  
 x = "DID Interaction Coefficient",  
 y = "Model"  
 ) +  
 theme\_minimal(base\_size = 13)  
  
model\_compare



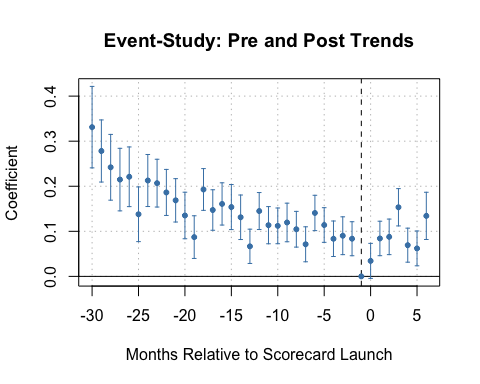
avg\_pre\_post <- clean\_data %>%  
 mutate(period = ifelse(post\_scorecard == 1, "Post", "Pre"),  
 period = factor(period, levels = c("Pre", "Post"))) %>%  
 group\_by(high\_earning, period) %>%  
 summarise(mean\_index = mean(avg\_std\_index, na.rm = TRUE)) %>%  
 mutate(high\_earning = ifelse(high\_earning == 1, "High Earning", "Low Earning"))

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

ggplot(avg\_pre\_post, aes(x = period, y = mean\_index, fill = high\_earning)) +  
 geom\_col(position = position\_dodge(width = 0.9)) +  
 geom\_text(  
 aes(label = round(mean\_index, 3)),  
 position = position\_dodge(width = 0.9),  
 vjust = ifelse(avg\_pre\_post$mean\_index >= 0, -0.3, 1.2),  
 size = 3.5  
 ) +  
 labs(  
 title = "Average Search Activity Pre vs. Post College Scorecard",  
 y = "Average Search Activity (Std Index)",  
 x = "Period",  
 fill = "College Group"  
 ) +  
 theme\_minimal()



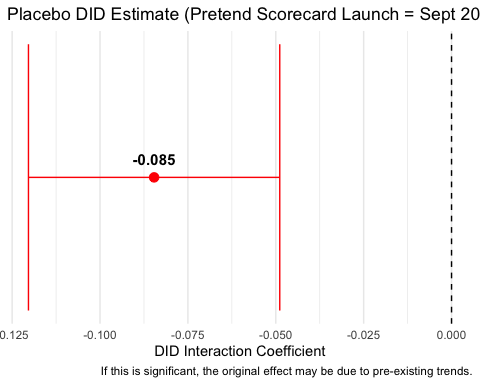
# Create relative time variable in months from Scorecard launch  
clean\_data <- clean\_data %>%  
 mutate(  
 rel\_month = as.integer(difftime(month, as.Date("2015-09-01"), units = "days") / 30)  
 )  
  
# Create event-time dummies (exclude -1 as the reference period)  
event\_study <- feols(  
 avg\_std\_index ~ i(rel\_month, high\_earning, ref = -1) | schname + month,  
 data = clean\_data,  
 cluster = ~schname  
)  
  
# Plot event-study coefficients  
iplot(event\_study, main = "Event-Study: Pre and Post Trends",  
 xlab = "Months Relative to Scorecard Launch", ylab = "Coefficient",  
 col = "steelblue")



# Create placebo treatment indicator (pretend launch = Sept 2014)  
full\_data <- full\_data %>%  
 mutate(  
 post\_scorecard\_placebo = if\_else(month >= as.Date("2014-09-01"), 1L, 0L)  
 )  
  
# Run placebo DID using your preferred model (Two-Way FE as example)  
placebo\_model <- feols(  
 avg\_std\_index ~ post\_scorecard\_placebo \* high\_earning | schname + month,  
 data = full\_data  
)

The variables 'post\_scorecard\_placebo' and 'high\_earning' have been removed because of collinearity (see $collin.var).

# Extract coefficient and CI  
placebo\_tidy <- tidy(placebo\_model, conf.int = TRUE) %>%  
 filter(term == "post\_scorecard\_placebo:high\_earning")  
  
# Plot placebo effect  
ggplot(placebo\_tidy, aes(x = estimate, y = 1)) +  
 geom\_point(color = "red", size = 3) +  
 geom\_errorbarh(aes(xmin = conf.low, xmax = conf.high), height = 0.2, color = "red") +  
 geom\_text(aes(label = sprintf("%.3f", estimate)),   
 vjust = -1.2, color = "black", size = 4, fontface = "bold") +  
 geom\_vline(xintercept = 0, linetype = "dashed") +  
 scale\_y\_continuous(breaks = NULL) +  
 labs(  
 title = "Placebo DID Estimate (Pretend Scorecard Launch = Sept 2014)",  
 x = "DID Interaction Coefficient",  
 y = NULL,  
 caption = "If this is significant, the original effect may be due to pre-existing trends."  
 ) +  
 theme\_minimal()



# Compare Model D (real Scorecard date) vs Placebo (Sept 2014)  
  
# Pull the interaction term + 95% CI from each model  
did\_d <- broom::tidy(model\_d, conf.int = TRUE) %>%  
 dplyr::filter(term %in% c("post\_scorecard:high\_earning",  
 "post\_scorecard x high\_earning")) %>%  
 dplyr::transmute(  
 what = "Model D: Two-Way FE",  
 estimate, conf.low, conf.high  
 )  
  
did\_p <- broom::tidy(placebo\_model, conf.int = TRUE) %>%  
 dplyr::filter(term %in% c("post\_scorecard\_placebo:high\_earning",  
 "post\_scorecard\_placebo x high\_earning")) %>%  
 dplyr::transmute(  
 what = "Placebo (Launch = Sept 2014)",  
 estimate, conf.low, conf.high  
 )  
  
compare\_df <- dplyr::bind\_rows(did\_d, did\_p) %>%  
 dplyr::mutate(  
 what = factor(what, levels = c("Model D: Two-Way FE",  
 "Placebo (Launch = Sept 2014)")),  
 label = sprintf("%.3f", estimate)  
 )  
  
# Plot  
ggplot(compare\_df, aes(x = estimate, y = what, color = what)) +  
 geom\_errorbarh(aes(xmin = conf.low, xmax = conf.high),  
 height = 0.15, size = 1) +  
 geom\_point(size = 3) +  
 geom\_text(aes(label = label), vjust = -1, nudge\_x = 0.001, size = 3.8,  
 color = "black") +  
 geom\_vline(xintercept = 0, linetype = "dashed") +  
 scale\_color\_manual(values = c("Model D: Two-Way FE" = "#1f78b4",  
 "Placebo (Launch = Sept 2014)" = "#e31a1c"),  
 guide = "none") +  
 labs(  
 title = "DID vs. Placebo DID",  
 subtitle = "Interaction coefficients for high × post\nwith 95% confidence intervals",  
 x = "DID interaction coefficient",  
 y = NULL,  
 caption = "Model D uses school and month fixed effects. Placebo pretends the launch was Sept 2014."  
 ) +  
 theme\_minimal(base\_size = 13)

