🌬️🗳️ Assignment 2: Wind Turbines, Matching, and Difference-in-Differences

Replicate causal inference identification strategies in Stokes (2015)

AUTHOR
EDS 241 / ESM 244 (DUE: 2/4/26)

PUBLISHED
January 26, 2026

Assignment instructions

Working with classmates to troubleshoot code and concepts is encouraged. If you collaborate, list collaborators at the top of your submission.

All written responses must be written independently (in your own words).

Keep your work readable: Use clear headings and label plot elements thoughtfully.

Assignment submission (YOUR NAME): Peter Vitale

Introduction

In this assignment you will be doing political weather forecasting except the "storms" we care about are electoral swings that might follow local wind turbine development.

In Stokes (2015), the idea is that a policy with diffuse benefits (cleaner electricity) can create concentrated local costs (turbines nearby), and those local opponents may "send a signal" at the ballot box (i.e., NIMBYISM). Your job is to use two statistical tools:

- Matching: Can we create a more apples-to-apples comparison between precincts that did vs. did not end up near turbine proposals?
- Fixed effects + Difference-in-Differences: Can we use repeated elections to estimate how within-precinct changes in turbine exposure relate to changes in incumbent vote share?

Learning goal: Replicate the matching and fixed effects analyses from study:

> Stokes (2015): *"Electoral Backlash against Climate Policy: A Natural Experiment on Retrospective Voting and Local Resistance to Public Policy*.

- **Study:** Stokes (2015) - Article
- **Data source:** Dataverse-Stokes2015

> `NOTE:` Replication of study estimates will be approximate. An alternative matching procedure and fixed effects estimation package are utilized in this assignment for illustration purposes.

Setup: Load libraries

0. Load libraries (+ install if needed)

```r
library(tidyverse)
```

```
── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
✔ dplyr     1.1.4     ✔ readr     2.1.6
✔ forcats   1.0.0     ✔ stringr   1.6.0
✔ ggplot2   4.0.1     ✔ tibble    3.3.1
✔ lubridate 1.9.4     ✔ tidyr     1.3.2
✔ purrr     1.2.1
── 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
```

```r
library(here)
```

```
here() starts at /Users/petervitale/Desktop/MEDS/EDS-241/EDS-241-HW/HW2-Matching-DiD
```

```r
library(janitor)
```

```

Attaching package: 'janitor'

The following objects are masked from 'package:stats':

    chisq.test, fisher.test
```

```r
library(jtools)

library(gtsummary)
library(gt)

library(MatchIt) # matching
library(cobalt)  # balance + love plots
```

```
 cobalt (Version 4.6.1, Build Date: 2025-08-20)

Attaching package: 'cobalt'

The following object is masked from 'package:MatchIt':

    lalonde
```

```r
library(fixest) # fast fixed effects
library(scales) # plotting
```

```
Attaching package: 'scales'

The following object is masked from 'package:fixest':

    pvalue

The following object is masked from 'package:purrr':

    discard

The following object is masked from 'package:readr':

    col_factor
```

Part 1: Study Background

1A. Dive into the details of the study design and evaluation plan

> Goal: Get familiar with the study setting, environmental issue, and policy under evaluation.

> `NOTE:` Read over study to inform your response to the assignment questions. For this assignment we will skip-over sections that describe the *Instrumental Variables* identification strategy. We will cover instrumental variable designs weeks 6-7.

1A.Q1 Summarize the environmental policy issue, the outcome of interest, and the intervention being evaluated. Be sure to include a brief description of each of the following key elements of the study: unit of analysis, outcome, treatment, comparison group):

Response: The issue covered in this paper is the costs of environmental policies, specifically wind energy projects, and how they can manifest into political action. This political action is very spatially biased to the areas with wind turbines, and the effect diminishes the farther one lives from turbines. The analysis was undertaken using provincial electoral districts precincts (unit of analysis), and showed that the governing party when turbines were put in place saw a diminished vote in the following election. The treatment in this experiment were the turbines being built. The comparison groups were different precincts at different distances from turbines, as well as electoral districts as a whole.

1A.Q2 Why might turbine proposals be correlated with baseline political preferences or rural areas? Provide 2 plausible mechanisms, and explain why that creates confounding.

Response: Rural areas swing more conservative naturally, which already shifts the baseline against natural energies. Another plausible mechanism is that education is generally lower in rural areas, which may lead to lack of awareness of benefits of natural energy in relation to costs.

1B. Break down the causal inference strategy and identify threats to identification:

1B.Q1 What is the key identifying assumption for a fixed effects / Difference-in-Difference design? Explain how this assumption when satisfied provides evidence of causal effect:

Response: The key assumption of the fixed affects model is Parallel Trends Assumption (or Common Trends Assumption). This assumes that, in the absence of treatment, the average outcomes for both treatment and control groups would have followed the same, parallel trajectory over time. When satisfied, this assumption shows a distinct impact (or lacktherof) of the treatment. It essentially confirms that the change we are seeing is due to the treatment instead of inherent differences in the populations.

1B.Q2 What is the reason for using a fixed effects approach from a causal inference perspective? Summarize within the context of study (in your own words).

Response: From a causal inference perspective, a fixed effect model is great for dealing with issues that have many factors integrated within them. This is because you can account for the factors in your models. In our example, the study uses the model to add effects of income, education, and others.

1B.Q3 What part of the SUTVA assumption is most likely violated in the context of this study design (and why)?

Response: Spillover is most likely violated in the context of the study do to people inherently moving around. For example someone may live farther from a turbine, but work nearer to one spreading the treatment effect to what should be a control as they may start to disdain the turbine. Furthermore, the `lines` in this project aren't drawn super concretely. By using proximity to a turbine the lines become blurred and spillover becomes more common.

1B.Q4 Why does spillover matter when estimating an unbiased treatment effect?

Response: Spillover matters when comparing treatment to control becasue it can either unnaturally raise the control effect or dull the treatment effect. Both create bias and lead to an obfuscation of the results.

1B.Q5 How do the authors assess the risk of spillovers, and what analytic choice do they make to attempt to mitigate the risk that spillover biases the causal estimate?

Response: To examine how far the effect persists over space, I also examined whether vote share declines occur in precincts near turbines. Further, when estimating the effect for each group, the sample excludes units less than 6 km away from the turbines as control, to eliminate spillovers when estimating each group's treatment effect.

The authors assessed the risk by comparing whether party voting declined in precincts near to turbines, and set the control line 3 km further than the treatment as to reduce spillover.

Part 2: Matching

We will start by evaluating the 2007 survey (cross-sectional) data. Treatment is defined by whether a precinct is near a turbine proposal (within 3 km).

> Goal: Match precincts using pre-treatment covariates and then estimate the effect of proposed wind turbines on incumbent vote share.

2A. Load data for matching

1. Read in data file `stokes15_survey2007.csv`
2. Code `precinct_id` and `district_id` as factors
3. Take a look at the data

```
match_data <- read_csv(here('data', 'stokes15_survey2007.csv')) %>%
    mutate(precinct_id = factor(precinct_id),
           district_id = factor(district_id))
```

```
Rows: 5973 Columns: 16
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr  (1): precinct_id
dbl (15): district_id, change_liberal, proposed_turbine_3km, log_wind_power,...

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.
```

2A.Q1 Intuition check: **Why match?** Explain rationale for using this method.

Response: The reason for matching is to reduce selection bias and confounding variables, which is good in a dataset like this.

2B. Check imbalance (before matching)

- Create a covariate *balance table* comparing treated and control precincts
- Treatment indicator: `proposed_turbine_3km`
- Include pre-treatment covariates: `log_home_val_07`, `p_uni_degree`, `log_median_inc`, `log_pop_denc`
- Use the `tbl_summary()` function from the `{gtsummary}` package.

```
match_data %>%
    select('proposed_turbine_3km', 'log_home_val_07',
           'p_uni_degree', 'log_median_inc') %>%
    tbl_summary(
        by = proposed_turbine_3km,
        statistic = c(list(all_continuous() ~ "{mean} ({sd})"))) %>%
        modify_header(label ~ "**Covariate**") %>%
  modify_spanning_header(c("stat_1", "stat_2") ~ "**Proposed Turbine**")
```

	Proposed Turbine	
Covariate	**0** N = 5,619[1]	**1** N = 354[1]
log_home_val_07	12.26 (0.37)	12.29 (0.29)
p_uni_degree	0.17 (0.12)	0.13 (0.09)

[1] Mean (SD)

	Proposed Turbine	
Covariate	0 N = 5,619[1]	1 N = 354[1]
log_median_inc	10.32 (0.22)	10.31 (0.19)

[1] Mean (SD)

2B.Q1 Summarize the table output: Which covariates look balanced/imbalanced?

Response: The groups (proposed turbines within 3km) are the most uneven with ~5.5 thousand observations as a 0 (no) and only 354 as 1 (yes). The covariates are rather balanced, remaining pretty close throughout.

2B.Q2 Describe in your own words why these covariates might be expected to confound the treatment estimate:

Response (2-4 sentences): These covariates may be expected to confound the treatment estimate for a variety of reasons. Log home value, income, and education are all signifiers that tie to both social and political leanings. Furthermore, issues arise dealing with non-uniform deployal of turbines in low income communities (Lindvall 2023). This can lead to a major confounding in our treatment effect, so controlling for it will be necessary.

2B.Q3 Intuition check: What type of data do you need to conduct a matching analysis?

Response: For matching analysis one wants to have many observations of our control as well as many pre-treatment observations, both of which we have in our dataset.

Conduct matching estimation using the {`MatchIt`} package:

Documentation - MatchIt

Learning goals:

- Approximate the Mahalanobis matching method used in Stokes (2015)
- Implement another common matching approach called `propensity score matching`

> `NOTE`: In the replication code associated with Stokes (2015) the {`AER`} package is used for Mahalanobis matching. In this assignment we use the {`MatchIt`} package. The results are comparable but will not be exactly the same.

2C. Mahalanobis nearest-neighbor matching

- Conduct Mahalanobis matching
- Use nearest-neighbor match without replacement using Mahalanobis distance
- Use 1-to-1 matching (match one control unit to each treatment unit)
- Extract the matched data using `match.data()`

```
set.seed(2412026)
```

```r
match_model <- matchit(
      proposed_turbine_3km ~  log_home_val_07 + p_uni_degree +
          log_median_inc + log_pop_denc,
  data = match_data,
  method = "nearest",        # Nearest neighbor matching
  distance = "mahalanobis", # Mahalanobis distance
  ratio = 1,                # Match one control unit to one treatment unit (1:1 matching)
  replace = FALSE           # Control observations are not replaced
)

# Extract matched data
matched_data <- match.data(match_model)
```

```r
summary(match_model)
```

```
Call:
matchit(formula = proposed_turbine_3km ~ log_home_val_07 + p_uni_degree +
    log_median_inc + log_pop_denc, data = match_data, method = "nearest",
    distance = "mahalanobis", replace = FALSE, ratio = 1)

Summary of Balance for All Data:
                Means Treated Means Control Std. Mean Diff. Var. Ratio
log_home_val_07       12.2948       12.2620          0.1138     0.5941
p_uni_degree           0.1257        0.1688         -0.5032     0.4916
log_median_inc        10.3096       10.3219         -0.0636     0.7581
log_pop_denc           3.5398        5.1192         -0.8897     0.5474
                eCDF Mean eCDF Max
log_home_val_07    0.0382   0.0881
p_uni_degree       0.1032   0.1769
log_median_inc     0.0355   0.0750
log_pop_denc       0.2099   0.3713

Summary of Balance for Matched Data:
                Means Treated Means Control Std. Mean Diff. Var. Ratio
log_home_val_07       12.2948       12.2975         -0.0093     1.0063
p_uni_degree           0.1257        0.1262         -0.0060     1.0485
log_median_inc        10.3096       10.3096          0.0002     1.0403
log_pop_denc           3.5398        3.5982         -0.0329     0.9784
                eCDF Mean eCDF Max Std. Pair Dist.
log_home_val_07    0.0075   0.0282          0.1334
p_uni_degree       0.0088   0.0367          0.1642
log_median_inc     0.0073   0.0395          0.1225
log_pop_denc       0.0109   0.0508          0.1485

Sample Sizes:
          Control Treated
All          5619     354
Matched       354     354
Unmatched    5265       0
Discarded       0       0
```

2C.Q1 Using the `summary()` output: Which covariate had the largest and smallest `Std. Mean Diff.` before matching. Next, compare largest/smallest `Std. Mean Diff.` after matching.

Response: Before matching the smallest `Std. Mean Diff.` is `p_uni_degree` at -.05 and the largest is `log_home_val_07` at .11. This changes drastically when matched and actually becomes a -.0093. After treatment the largest `Std. Mean Diff.` is `log_median_inc` at .0002 and the smallest is `log_pop_denc`. But in general all the `Std. Mean Diff.`'s all become **tiny** which is great!

2D. Create a "love plot" using `love.plot()` ❤️

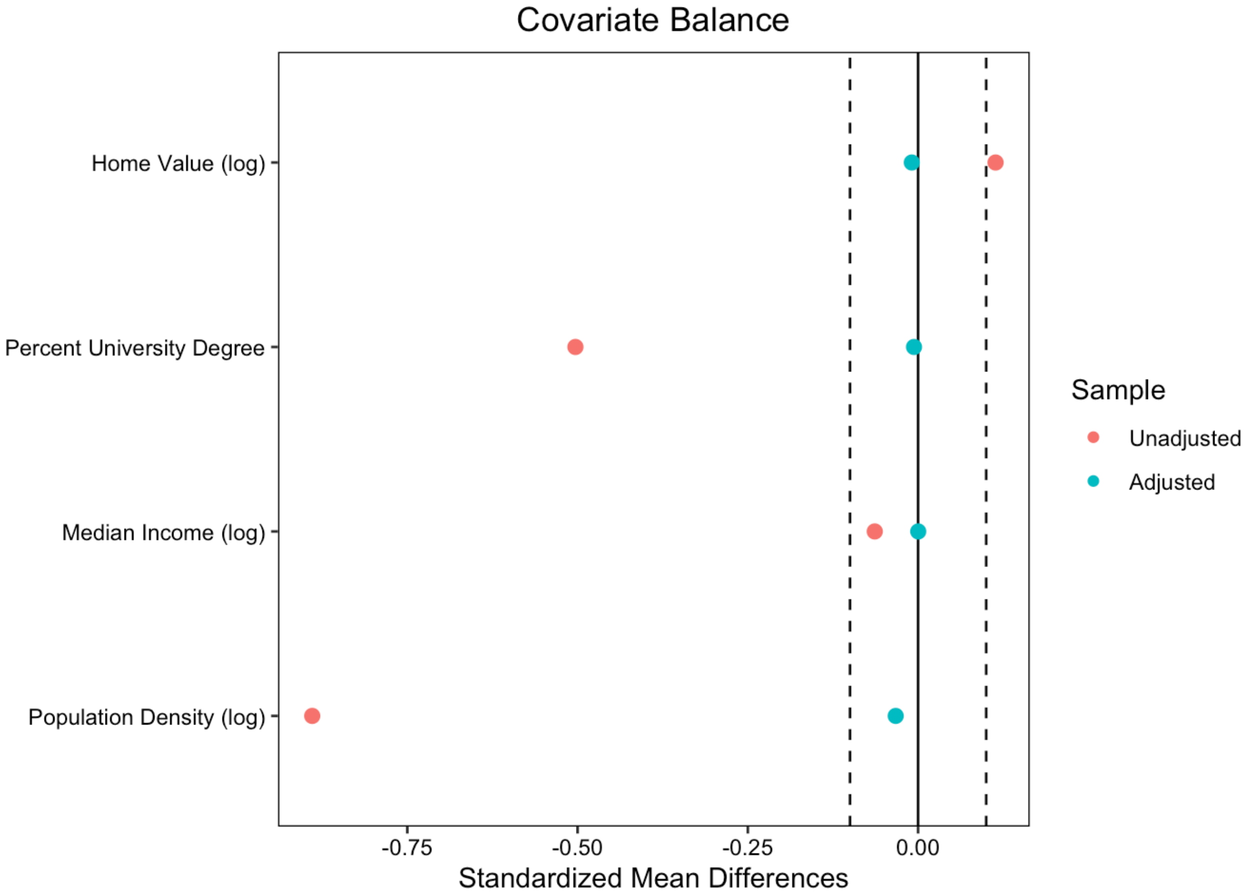
📜 [Documentation - cobalt]

- Plot mean differences for data before & after matching across all pre-treatment covariates
- This is an effective way to evaluate how effective matching was at achieving balance.

- Make a love plot of standardized mean differences (SMDs) before vs after matching.
- Include a threshold line at 0.1.
- In love plot display `mean.diffs`

```
new_names <- data.frame(
    old = c("log_home_val_07", "p_uni_degree", "log_median_inc", "log_pop_denc"),
    new = c("Home Value (log)", "Percent University Degree",
            "Median Income (log)", "Population Density (log)"))

# Love plot
love.plot(match_model, stats = "mean.diffs",
          thresholds = c(m = 0.1),
          var.names = new_names)
```



2D.Q1 Interpret the love plot in your own words:

Response: After matching our Covariate standard mean differences all become very close to 0.

Propensity score matching

2E. Propensity Score Matching (PSM)

- Estimate 1:1 nearest-neighbor Propensity Score Matching
- Same code as above except change `distance = "logit"`

```
set.seed(2412026)

propensity_scores <- matchit(
      proposed_turbine_3km ~  log_home_val_07 + p_uni_degree +
          log_median_inc + log_pop_denc,
  data = match_data,
  method = "nearest",       # Nearest neighbor matching
  distance = "logit", # logit distance
  ratio = 1,                # Match one control unit to one treatment unit (1:1 matching)
  replace = FALSE           # Control observations are not replaced
)
```

```
# Extract matched data
matched_propensity_data <- match.data(propensity_scores)
```

Create table displaying covariate balance using `cobalt::bal.tab()`

[Documentation - cobalt]

Use `bal.tab()` to report balance before and after matching.

```
bal.tab(propensity_scores,
        var.names = new_names)
```

```
Balance Measures
                    Type Diff.Adj
distance        Distance   0.0001
log_home_val_07  Contin.   0.0205
p_uni_degree     Contin.   0.0457
log_median_inc   Contin.  -0.0042
log_pop_denc     Contin.  -0.0365

Sample sizes
          Control Treated
All          5619     354
Matched       354     354
Unmatched    5265       0
```

```
bal.tab(match_model,
        var.names = new_names)
```

```
Balance Measures
                   Type Diff.Adj
log_home_val_07 Contin.  -0.0093
p_uni_degree    Contin.  -0.0060
log_median_inc  Contin.   0.0002
log_pop_denc    Contin.  -0.0329

Sample sizes
          Control Treated
All          5619     354
Matched       354     354
Unmatched    5265       0
```

2E.Q1 Compare Mahalanobis vs propensity score matching. Which method did a better job at achieving balance?

Response: The Mahalanobis method seemed to do a better job at achieving balance, as the adjusted differences between groups is smaller (closer to zero).Both methods produced 354 matches.

2F. Estimate an effect in the matched sample

Using the matched data (Mahalanobis method), estimate the effect of treatment on the change in incumbent vote share (`change_liberal`).

```
reg_match <- lm(change_liberal ~ proposed_turbine_3km, matched_data)

summ(reg_match, model.fit = FALSE)
```

Observations	708
Dependent variable	change_liberal
Type	OLS linear regression

	Est.	S.E.	t val.	p
(Intercept)	-0.07	0.01	-10.96	0.00
proposed_turbine_3km	-0.06	0.01	-7.25	0.00

Standard errors: OLS

2F.Q1 Have you identified a causal estimate using this approach: Why or why not?

Response: It seems we have identified a causal estimate, as the low p-value suggests the relationship between proposed turbine development and political change is (statistically significantly) unlikely to be random. This was done after the matching, which adds rigor to our estimate.

2F.Q2 When using a matching method, what is the main threat to causal identification?

Response: The matching method can only match the covariates you give it, therefore omitted variable bias is still posible and remains a threat to causal identification.

2F.Q3 Describe why the treatment estimate represents the `Average Treatment for the Treated (ATT)` and explain why this is the case relative to estimation of the `Average Treatment Effect (ATE)`.

Response: ATT represents the treatment effect of those treated, which were the 354 treated units (matches). The reason that we do not use ATE is if treatment is randomly assigned, but in our case we use the treatment effect for the treated as it is not random or equal.

Part 3: Panel Data, Fixed Effects, and Difference-in-Difference

Data source: [Dataverse-Stokes2015]

3A: Read in the panel data + code variables `precinct_id` and `year` as factors

```
panel_data <-  read_csv(here('data', 'stokes15_panel_data.csv')) %>%
    mutate(precinct_id = factor(precinct_id),
           year = factor(year))
```

```
Rows: 18558 Columns: 14
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
```

```
chr  (1): precinct_id
dbl (13): year, perc_lib, proposed_turbine, operational_turbine, log_pop, lo...

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.
```

```
# HINT: Try running `tabyl(panel_data$year)`. Review article to make sense of the row numbers
tabyl(panel_data$year)
```

```
 panel_data$year    n   percent
            2003 6186 0.3333333
            2007 6186 0.3333333
            2011 6186 0.3333333
```

3A.Q1: Why are there 18,558 rows in `panel_data`?

Response: There are 18.5k rows because each precinct has a value over 3 years (6186 precincts * 3 years)

```
# How many years are included in the panel?
print(paste0('There are ', n_distinct(panel_data$year), ' years included'))
```

```
[1] "There are 3 years included"
```

```
# How many precincts are there?
print(paste0('There are ', n_distinct(panel_data$precinct_id), ' precincts included'))
```

```
[1] "There are 6186 precincts included"
```

3A.Q2: How many unique precincts are *ever treated* (i.e., `proposed` & `operational`)?

```
print(paste0('There are ', n_distinct(panel_data %>%
               filter(proposed_turbine == 1 |
                      operational_turbine == 1)), ' precincts that are ever treated'))
```

```
[1] "There are 237 precincts that are ever treated"
```

Response: See above

```
panel_data %>%
  group_by(precinct_id) %>%
  summarise(
    ever_proposed    = any(proposed_turbine == 1, na.rm = TRUE),
    ever_operational = any(operational_turbine == 1, na.rm = TRUE),
    .groups = "drop") %>%
  summarise(
    n_ever_proposed    = sum(ever_proposed),
    n_ever_operational = sum(ever_operational))
```

```
# A tibble: 1 × 2
  n_ever_proposed n_ever_operational
```

```
           <int>              <int>
1           184                  52
```

3B. Plot and evaluate parallel trends: Replicate `Figure.2` (Stokes, 2015)

1. Create indicators for whether each precinct is ever treated by 2011 (`treat_p`, `treat_o`; separate indicator for proposals and operational turbines).
2. Plot mean incumbent vote share by year for treated vs control precincts (with 95% CIs).
3. Facet by turbine type (proposed & operational)

Step 1: Prepare data

```r
trends_data <- panel_data %>%
  group_by(precinct_id) %>%
  mutate(
    treat_p = as.integer(any(proposed_turbine == 1, na.rm = TRUE)),  # ever proposed (in any y
    treat_o = as.integer(any(operational_turbine == 1, na.rm = TRUE))) %>% # ever operational
  ungroup() %>%
  pivot_longer(c(treat_p, treat_o),
               names_to = "turbine_type", values_to = "treat") %>%
  mutate(
      turbine_type = factor(turbine_type,
                            levels = c("treat_p", "treat_o"),
                            labels = c("Proposed turbines", "Operational turbines")),
    status = if_else(treat == 1, "Treated", "Control"),
    year   = factor(year))
```

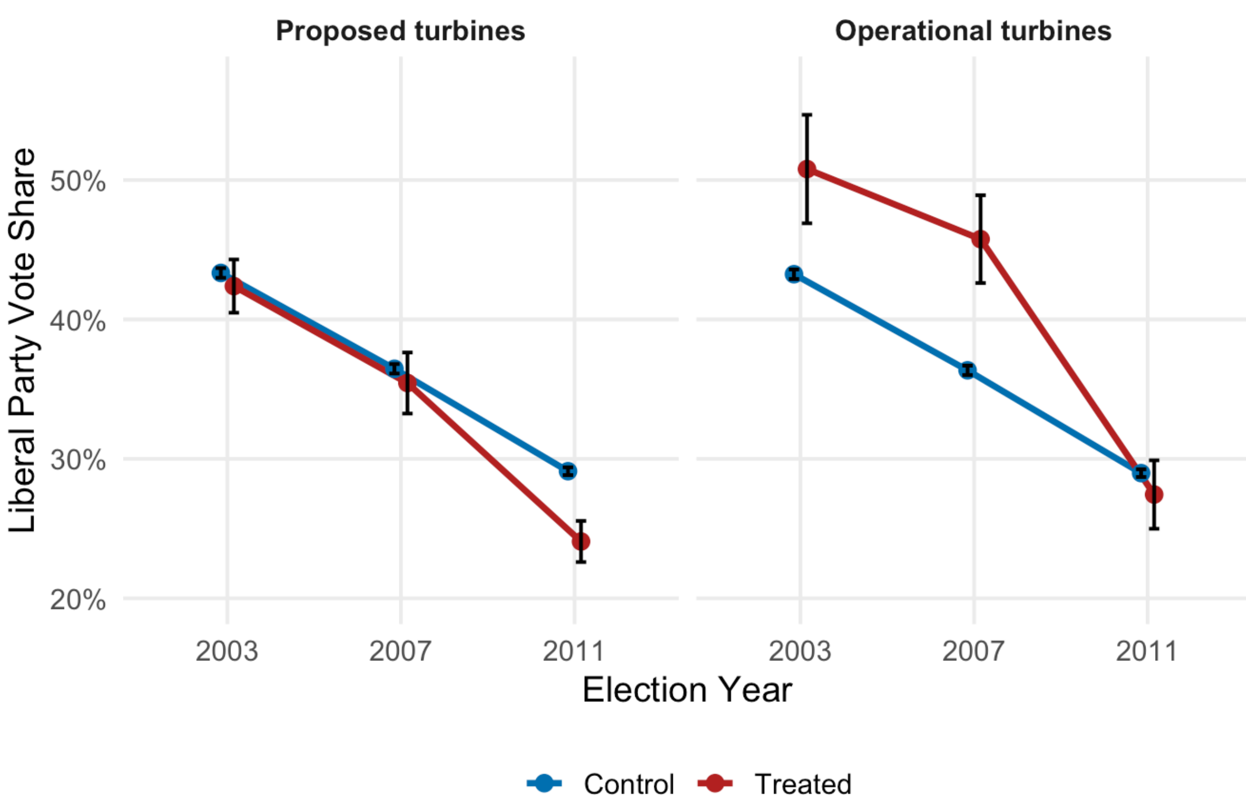
Step 2: Create trends plot

```r
pd <- position_dodge(width = 0.15)

trends_data %>%
  group_by(turbine_type, status, year) %>%
  summarise(
    mean = mean(perc_lib, na.rm = TRUE),
    n    = sum(!is.na(perc_lib)),
    se   = sd(perc_lib, na.rm = TRUE) / sqrt(n),
    ci   = qt(.975, df = pmax(n - 1, 1)) * se,
    .groups = "drop") %>%
ggplot(aes(year, mean, color = status, group = status)) +
  geom_line(position = pd, linewidth = 1.2) +
  geom_point(position = pd, size = 2.6) +
  geom_errorbar(
    aes(ymin = mean - ci, ymax = mean + ci),
    position = pd, width = .12, linewidth = .7, color = "black") +
  facet_wrap(~ turbine_type, nrow = 1) +
  scale_color_manual(values = c(Control = "#0072B2", Treated = "#B22222")) +
  scale_y_continuous(labels = percent_format(accuracy = 1)) +
  coord_cartesian(ylim = c(.20, .57)) +
  labs(
    title = "Figure 2. Trends in the Governing Party's Vote Share",
```

```
    x = "Election Year",
    y = "Liberal Party Vote Share",
    color = NULL) +
  theme_minimal(base_size = 14) +
  theme(
    panel.grid.minor = element_blank(),
    legend.position = "bottom",
    strip.text = element_text(face = "bold"))
```

Figure 2. Trends in the Governing Party’s Vote Share



3B.Q1: Write a short paragraph assessing the parallel trends assumption for each outcome.

Response (4-6 sentences): This plot shows some parallel trends in both proposed and operational turbines. This being said, it relies on very few points. This, to quote Annie, makes all time periods seem equal when they aren’t. Elections, war, and general discontent with the controlling parties could all be present in those large yearly jumps. These few data points with large jumps can cloud the results and create some doubt about parallel trends. Furthermore the lack of pre-proposal data makes truly assessing trends based on this data impossible.

Estimating Fixed Effects Models (DiD) for proposals

\text{Y}_{it} = \alpha_0 + \beta \cdot (\text{proposed_turbine}_{it}) + \gamma_i + \delta_t + \varepsilon_{it}

- Y_{it} is the vote share for the Liberal Party in precinct *i* in time *t*
- β is the treatment effect of a turbine being proposed within a precinct
- γ_i is the precinct fixed effect

- δ_t is the year fixed effect

Example 1: Randomly sample 40 precincts

- To illustrate the "dummy variable method" of estimating fixed effects using the the general `lm()` function we are going to randomly sample 40 precincts (20 "treated" precincts with proposed turbines).
- If we attempted to use this approach with the full sample estimating all 6185 (n-1) precinct-level coefficients is impractical (it would take a long time).

```
set.seed(40002026)

precinct_frame <- panel_data %>%
  group_by(precinct_id) %>%
  summarise(
    proposed_turbine_any = as.integer(any(proposed_turbine == 1, na.rm = TRUE)),
    .groups = "drop"
  )

ids_40 <- precinct_frame %>%
  group_by(proposed_turbine_any) %>%
  slice_sample(n = 20) %>%
  ungroup() %>%
  select(precinct_id)

sample_40_precincts <- panel_data %>%
  semi_join(ids_40, by = "precinct_id")
```

3C: Estimate a fixed effects model using `lm()` with fixed effects added for `precinct` and `year` using the sample of 40 precincts just created.

```
model1_ff <- lm(perc_lib ~ operational_turbine + year + precinct_id, data = sample_40_precinct

summ(model1_ff , model.fit = FALSE)
```

Observations	120
Dependent variable	perc_lib
Type	OLS linear regression

	Est.	S.E.	t val.	p
(Intercept)	0.28	0.05	5.65	0.00
operational_turbine	-0.11	0.05	-1.94	0.06
year2007	-0.05	0.02	-2.61	0.01
year2011	-0.15	0.02	-7.61	0.00
precinct_id10.115s.10.84.	0.17	0.07	2.41	0.02

Standard errors: OLS

	Est.	S.E.	t val.	p
precinct_id105.038.105.45.	0.05	0.07	0.77	0.45
precinct_id14.149.14.79.	0.19	0.07	2.74	0.01
precinct_id14.168.14.82.	0.17	0.07	2.51	0.01
precinct_id18.003.18.1.	0.10	0.07	1.48	0.14
precinct_id18.033.18.19.	-0.04	0.07	-0.59	0.56
precinct_id21.126.21.179.	0.20	0.07	2.96	0.00
precinct_id22.061.22.60.	0.23	0.07	3.21	0.00
precinct_id22.124.22.52.	0.14	0.07	1.98	0.05
precinct_id22.137.22.193.	0.18	0.07	2.59	0.01
precinct_id22.158.22.203.	0.18	0.07	2.54	0.01
precinct_id22.209.22.172.	0.20	0.07	2.87	0.01
precinct_id28.056.28.149.	-0.03	0.07	-0.44	0.66
precinct_id28.072.28.98.	0.10	0.07	1.41	0.16
precinct_id28.139.28.145.	0.00	0.07	0.02	0.98
precinct_id28.163.28.69.	0.12	0.07	1.81	0.07
precinct_id29.241.29.172.	0.05	0.07	0.69	0.49
precinct_id34.050.34.39.	0.34	0.07	4.96	0.00
precinct_id34.151.34.125.	0.08	0.07	1.14	0.26
precinct_id36.133.36.92.	0.36	0.07	5.19	0.00
precinct_id40.044.40.68.	0.20	0.07	2.87	0.01
precinct_id40.098.40.120.	0.18	0.07	2.55	0.01
precinct_id40.134.40.174.	0.22	0.07	3.13	0.00
precinct_id40.243.40.22.	0.31	0.07	4.43	0.00
precinct_id40.244.40.46.	0.19	0.07	2.72	0.01
precinct_id55.228.55.180.	0.13	0.07	1.89	0.06
precinct_id58.162.58.98.	0.28	0.07	4.03	0.00
precinct_id58.232.58.231.	0.20	0.07	2.95	0.00
precinct_id67.141.67.123.	0.00	0.07	0.04	0.97
precinct_id69.073.69.35.	0.07	0.07	1.02	0.31
precinct_id70.081.70.44.	0.06	0.07	0.90	0.37
precinct_id70.135.70.105.	0.44	0.07	6.40	0.00
precinct_id70.221.70.155.	0.25	0.07	3.58	0.00
precinct_id73.248.73.180.	0.17	0.07	2.43	0.02
precinct_id73.251.73.180.	0.19	0.07	2.69	0.01
precinct_id87.017.87.58.	0.13	0.07	1.84	0.07

Standard errors: OLS

	Est.	S.E.	t val.	p
precinct_id87.053.87.68.	0.21	0.07	3.11	0.00
precinct_id87.213.87.24.	0.01	0.07	0.20	0.84
precinct_id98.009.98.12.	0.13	0.07	1.87	0.07

Standard errors: OLS

```
summ(model1_ff , model.fit = FALSE, digits = 3,
     robust = TRUE)
```

Observations	120
Dependent variable	perc_lib
Type	OLS linear regression

	Est.	S.E.	t val.	p
(Intercept)	0.283	0.088	3.224	0.002
operational_turbine	-0.106	0.042	-2.541	0.013
year2007	-0.049	0.023	-2.103	0.039
year2011	-0.149	0.027	-5.629	0.000
precinct_id10.115s.10.84.	0.166	0.105	1.586	0.117
precinct_id105.038.105.45.	0.053	0.110	0.482	0.631
precinct_id14.149.14.79.	0.189	0.101	1.872	0.065
precinct_id14.168.14.82.	0.173	0.114	1.518	0.133
precinct_id18.003.18.1.	0.102	0.100	1.018	0.312
precinct_id18.033.18.19.	-0.041	0.093	-0.437	0.663
precinct_id21.126.21.179.	0.204	0.152	1.336	0.185
precinct_id22.061.22.60.	0.229	0.110	2.087	0.040
precinct_id22.124.22.52.	0.137	0.112	1.219	0.226
precinct_id22.137.22.193.	0.179	0.103	1.732	0.087
precinct_id22.158.22.203.	0.181	0.090	2.012	0.048
precinct_id22.209.22.172.	0.205	0.107	1.910	0.060
precinct_id28.056.28.149.	-0.030	0.114	-0.265	0.791
precinct_id28.072.28.98.	0.097	0.104	0.931	0.355
precinct_id28.139.28.145.	0.002	0.121	0.014	0.989
precinct_id28.163.28.69.	0.124	0.105	1.182	0.241
precinct_id29.241.29.172.	0.048	0.089	0.538	0.592
precinct_id34.050.34.39.	0.342	0.091	3.752	0.000
precinct_id34.151.34.125.	0.079	0.098	0.806	0.423

Standard errors: Robust, type = HC3

	Est.	S.E.	t val.	p
precinct_id36.133.36.92.	0.358	0.094	3.815	0.000
precinct_id40.044.40.68.	0.198	0.090	2.186	0.032
precinct_id40.098.40.120.	0.176	0.091	1.928	0.058
precinct_id40.134.40.174.	0.216	0.095	2.263	0.026
precinct_id40.243.40.22.	0.306	0.099	3.102	0.003
precinct_id40.244.40.46.	0.194	0.099	1.954	0.054
precinct_id55.228.55.180.	0.130	0.098	1.326	0.189
precinct_id58.162.58.98.	0.278	0.103	2.709	0.008
precinct_id58.232.58.231.	0.203	0.097	2.093	0.040
precinct_id67.141.67.123.	0.003	0.107	0.024	0.981
precinct_id69.073.69.35.	0.070	0.107	0.653	0.516
precinct_id70.081.70.44.	0.062	0.101	0.617	0.539
precinct_id70.135.70.105.	0.441	0.134	3.291	0.002
precinct_id70.221.70.155.	0.247	0.117	2.114	0.038
precinct_id73.248.73.180.	0.167	0.115	1.458	0.149
precinct_id73.251.73.180.	0.185	0.112	1.657	0.102
precinct_id87.017.87.58.	0.127	0.111	1.138	0.259
precinct_id87.053.87.68.	0.214	0.145	1.474	0.145
precinct_id87.213.87.24.	0.014	0.089	0.154	0.878
precinct_id98.009.98.12.	0.129	0.112	1.151	0.253

Standard errors: Robust, type = HC3

3C.Q1: Intuition check: Is the *signal-to-noise* ratio for the treatment estimate greater than *2-to-1*?

Response: The signal to noise ratio for the treatment is not greater than 2 to 1, however for many of the precincts it is much greater than 2 to 1.

> HINT: Add the argument `digits = 3` to the `summ()` function above

3C.Q2: Re-run the `summ()` function using the *heteroscedasticiy robust standard error adjustment* (`robust = TRUE`). Did the standard error (S.E.) estimates change? Explain why.

Response: The S.E. estimates became generally larger, this is because we adjusted to heteroscedasticity instead of homoscedasticity, which allows the standard errors to vary. This way the function doesnt oversimplify our errors.

3C.Q3: Compare results of the model above to the findings from the fixed effects analysis in the Stokes (2015) study. Why might the results be similar or different?

Response: The results may differ from Stokes (2015) because this model is estimated on a small random subsample of precincts rather than the full dataset, which reduces precision and may yield estimates that are less stable or representative of the full sample.

3C.Q4: In your own words, explain why it is advantageous from a causal inference perspective to include year and precinct fixed effects. Explain how between-level and within-level variance is relevant to the problem of omitted variable bias (OVB).

Response (2-4 sentences): Year and precinct are very apropriate fixed effects. The year can drastically change the inherent political landscape in a country – look at 2008 vs 2016 in the United States. Furthermore precincts can show community trends in political affiliation, which may influence a treatment effect. By including these variables as fixed effects we reduce between level variance, leaving the within-level variance we are testing, reducing omitted variable bias.

3D. Now using the full sample, estimate the treatment effect of wind turbine proposals on incumbent vote share. Use `feols()` from the `{fixest}` package to estimate the fixed effects.

See vignette here: [fixest walkthrough]

```r
model2_ff <- feols(perc_lib ~ proposed_turbine  | year + precinct_id,
                   data = panel_data)

summary(model2_ff, cluster = ~precinct_id)
```

```
OLS estimation, Dep. Var.: perc_lib
Observations: 18,558
Fixed-effects: year: 3,  precinct_id: 6,186
Standard-errors: Clustered (precinct_id)
                 Estimate Std. Error  t value   Pr(>|t|)
proposed_turbine -0.04157   0.007682 -5.41144 6.4869e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.070481     Adj. R2: 0.624835
                 Within R2: 0.00224
```

3D.Q1: Interpret the model results and translate findings to be clear to an audience that may not have a background in causal inference (Econometrics) methods.

In panel data settings, why is clustering by precinct important (i.e., `cluster = ~precinct_id`) ?"

Response (4-6 sentences): The effect of a proposed turbines on liberal voting is a drop of ~ 4%. This means precincts with proposed wind turbines saw their liberal vote share decrease by about 4 points compared to similar precincts without turbines. This number is very unlikely to be derived from random data, and thus is statistically significant. The reason that it is important to cluster by precinct is because - as stated earlier- precincts can share characteristics which create correlated errors over time. This can skew our results and increase the error (how wrong we are) within our model.

3E. Estimate the treatment effect of *operational wind turbines* on incumbent vote share. Use the same approach as the previous model.

```r
model3_ff <- feols(perc_lib ~ operational_turbine  | year + precinct_id,
                   data = panel_data)
```

```r
summary(model3_ff, cluster = ~precinct_id)
```

```
OLS estimation, Dep. Var.: perc_lib
Observations: 18,558
Fixed-effects: year: 3,  precinct_id: 6,186
Standard-errors: Clustered (precinct_id)
                     Estimate Std. Error  t value   Pr(>|t|)
operational_turbine -0.092762   0.011739 -7.90195 3.2271e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.070447     Adj. R2: 0.625198
                 Within R2: 0.003206
```

3E.Q1: Interpret the `model3_ff` results as clearly and **concisely** as you can.

Response: The implementation of an operational turbine reduces the liberal percentage of a community by 9%. This expectation is statistically significant, meaning it is likely not due to random chance.

3E.Q2: Why do you think the effect of proposed wind turbines is different from operational wind turbines. Develop your own theory about why incumbent vote share is affected in this way. Use the Stokes (2015) study to inform your response as needed.

Response: Proposed wind turbines are not realized, as in there is no direct effect to the proposal of a turbine. This may lead some to not even know it was proposed. Operational turbines, on the other hand, are hard to miss. The visibility, noise pollution, and construction issues make people more incensed to make a change in their voting behaviors.

References

Lindvall, Daniel. 2023. "Why Municipalities Reject Wind Power: A Study on Municipal Acceptance and Rejection of Wind Power Instalments in Sweden." *Energy Policy* 180 (September): 113664. https://doi.org/10.1016/j.enpol.2023.113664.