

Gov 50: 6. Causality

Matthew Blackwell

Harvard University

Roadmap

1. What is causality?
2. Data importing
3. Logicals

1/ What is causality?



Two roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
And looked down one as far as I could
To where it bent in the undergrowth;

What is a causal effect?

factual

vs.

counterfactual

- Does increasing the minimum wage increase the unemployment rate?
 - Unemployment rate went up after the minimum wage increased
 - Would it have gone up if the minimum wage increase not occurred?
- Does having girls affect a judge's rulings in court?
 - A judge with a daughter gave a pro-choice ruling.
 - Would they have done that if had a son instead?
- **Fundamental problem of causal inference:**
 - Can never observe counterfactuals, must be inferred.

Political canvassing study



POLITICAL SCIENCE

Durably reducing transphobia: A field experiment on door-to-door canvassing

David Broockman^{1*} and Joshua Kalla²

Existing research depicts intergroup prejudices as deeply ingrained, requiring intense intervention to lastingly reduce. Here, we show that a single approximately 10-minute conversation encouraging actively taking the perspective of others can markedly reduce prejudice for at least 3 months. We illustrate this potential with a door-to-door canvassing intervention in South Florida targeting antitransgender prejudice. Despite declines in homophobia, transphobia remains pervasive. For the intervention, 56 canvassers went door to door encouraging active perspective-taking with 501 voters at voters' doorsteps. A randomized trial found that these conversations substantially reduced transphobia, with decreases greater than Americans' average decrease in homophobia from 1998 to 2012. These effects persisted for 3 months, and both transgender and nontransgender canvassers were effective. The intervention also increased support for a nondiscrimination law, even after exposing voters to counterarguments.

- Can canvassers change minds about topics like transgender rights?
- Experimental setting:
 - Randomly assign canvassers to have a conversation about transgender right or a conversation about recycling.
 - Trans rights conversations focused on “perspective taking”
- Outcome of interest: support for trans rights policies.

A tale of two respondents

	Conversation Script	Support for Nondiscrimination Law
Respondent 1	Recycling	No
Respondent 2	Trans rights	Yes

Did the second respondent support the law **because** of the perspective-taking conversation?

Translating into math

Useful to have **compact** notation for referring to **treatment variable**:

$$T_i = \begin{cases} 1 & \text{if respondent } i \text{ had trans rights conversation} \\ 0 & \text{if respondent } i \text{ had recycling conversation} \end{cases}$$

Similar notation for the **outcome variable**:

$$Y_i = \begin{cases} 1 & \text{if respondent } i \text{ supports trans nondiscrimination laws} \\ 0 & \text{if respondent } i \text{ doesn't support nondiscrimination laws} \end{cases}$$

i is a placeholder to refer to a generic unit/respondent: Y_{42} is the outcome for the 42nd unit.

A tale of two respondents (redux)

	Conversation Script	Support for Nondiscrimination Law
Respondent 1	Recycling	No
Respondent 2	Trans rights	Yes

becomes...

i	T_i	Y_i
Respondent 1	0	0
Respondent 2	1	1

Causal effects & counterfactuals

- What does “ T_i causes Y_i ” mean? \rightsquigarrow **counterfactuals**, “what if”
- Would respondent change their support based on the conversation?
- Two **potential outcomes**:
 - $Y_i(1)$: would respondent i support ND laws if they had trans rights script?
 - $Y_i(0)$: would respondent i support ND laws if they had recycling script?
- **Causal effect**: $Y_i(1) - Y_i(0)$
 - $Y_i(1) - Y_i(0) = 0 \rightsquigarrow$ script has no effect on policy views
 - $Y_i(1) - Y_i(0) = -1 \rightsquigarrow$ trans rights script lower support for laws
 - $Y_i(1) - Y_i(0) = +1 \rightsquigarrow$ trans rights script increases support for laws

Potential outcomes

i	T_i	Y_i	$Y_i(1)$	$Y_i(0)$
Respondent 1	0	0	???	0
Respondent 2	1	1	1	???

- **Fundamental problem of causal inference:**
 - We only observe one of the two potential outcomes.
 - Observe $Y_i = Y_i(1)$ if $T_i = 1$ or $Y_i = Y_i(0)$ if $T_i = 0$
- To infer causal effect, we need to infer the missing counterfactuals!

Potential outcomes vs possible outcomes

- **Potential outcomes** are all about counterfactuals:
 - What outcome would we see if I received treatment?
- Different from the **possible values of the outcome**
 - the “vote” variable can take on a 0 or a 1.

How can we figure out counterfactuals?



- Find a similar unit! \rightsquigarrow **matching**
 - Mill's method of difference
- Does respondent support law because of the trans rights script?
 - \rightsquigarrow find a identical respondent who got the recycling script.
- NJ increased the minimum wage. Causal effect on unemployment?
 - \rightsquigarrow find a state similar to NJ that didn't increase minimum wage.

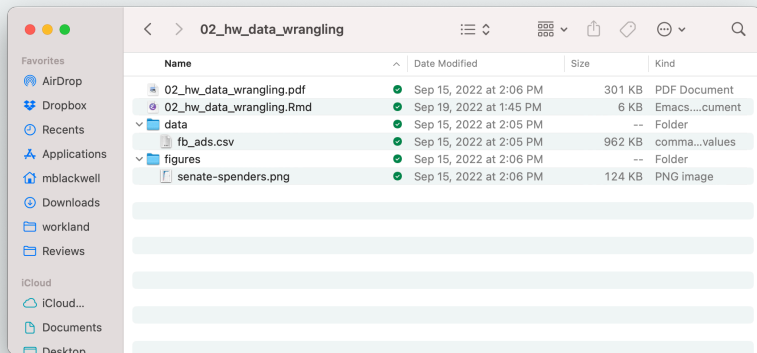
Imperfect matches



- The problem: imperfect matches!
- Say we match i (treated) and j (control)
- **Selection Bias:** $Y_i(1) \neq Y_j(1)$
- Those who take treatment may be different than those who take control.
- How can we correct for that?

2/ Data importing

Organizing your project



Keep your workspace clean. Directories help organize. Future you will thank present you.

read_csv to load CSV files

read_csv will import a csv file and create a tibble:

```
library(tidyverse)
resume <- read_csv("data/resume.csv")
resume
```

```
## # A tibble: 4,870 x 4
##   firstname sex    race    call
##   <chr>      <chr>  <chr>  <dbl>
## 1 Allison   female white    0
## 2 Kristen   female white    0
## 3 Lakisha    female black    0
## 4 Latonya    female black    0
## 5 Carrie     female white    0
## 6 Jay        male    white    0
## 7 Jill       female white    0
## 8 Kenya    female black    0
## 9 Latonya    female black    0
## 10 Tyrone     male    black    0
## # i 4,860 more rows
```

3/ Logicals

News data, redux

```
library(gov50data)
news <- na.omit(news)
news
```

```
## # A tibble: 2,560 x 10
##   callsign affiliation date       weekday ideology
##   <chr>      <chr>      <date>      <ord>      <dbl>
## 1 KECI      NBC        2017-06-07 Wed        0.0655
## 2 KPAX      CBS        2017-06-07 Wed        0.0853
## 3 KRBC      NBC        2017-06-07 Wed        0.0183
## 4 KTAB      CBS        2017-06-07 Wed        0.0850
## 5 KTMF      ABC        2017-06-07 Wed        0.0842
## 6 KTXS      ABC        2017-06-07 Wed       -0.000488
## 7 KAEF      ABC        2017-06-08 Thu        0.0426
## 8 KBVU      FOX        2017-06-08 Thu       -0.0860
## 9 KECI      NBC        2017-06-08 Thu        0.0902
## 10 KPAX     CBS        2017-06-08 Thu        0.0668
## # i 2,550 more rows
## # i 5 more variables: national_politics <dbl>,
## #   local_politics <dbl>, sinclair2017 <dbl>, post <dbl>,
## #   month <ord>
```

Creating logical vectors

You can create logical vectors using `mutate`. We can use the `.keep = "used"` here to only show the variables used in this `mutate` call:

```
news |>
  mutate(
    right_leaning = ideology > 0,
    fall = month == "Sep" | month == "Oct" | month == "Nov",
    .keep = "used"
  )
```

```
## # A tibble: 2,560 x 4
##   ideology month right_leaning fall
##   <dbl> <ord> <lgl>      <lgl>
## 1  0.0655 Jun    TRUE      FALSE
## 2  0.0853 Jun    TRUE      FALSE
## 3  0.0183 Jun    TRUE      FALSE
## 4  0.0850 Jun    TRUE      FALSE
## 5  0.0842 Jun    TRUE      FALSE
## 6 -0.000488 Jun    FALSE     FALSE
## 7  0.0426 Jun    TRUE      FALSE
## 8 -0.0860 Jun    FALSE     FALSE
## 9  0.0902 Jun    TRUE      FALSE
```

Using the logical variables to filter

```
news |>
  mutate(
    right_leaning = ideology > 0,
    fall = month == "Sep" | month == "Oct" | month == "Nov"
  ) |>
  filter(right_leaning & fall)
```

```
## # A tibble: 1,050 x 12
##   callsign affiliation date       weekday ideology
##   <chr>      <chr>      <date>      <ord>      <dbl>
## 1 KBZK      CBS        2017-09-01 Fri        0.121
## 2 KHSL      CBS        2017-09-01 Fri        0.0564
## 3 KNVN      NBC        2017-09-01 Fri        0.0564
## 4 KRCR      ABC        2017-09-01 Fri        0.324
## 5 WCTI      ABC        2017-09-01 Fri        0.0649
## 6 WCYB      NBC        2017-09-01 Fri        0.0613
## 7 WEMT      FOX        2017-09-01 Fri        0.187
## 8 WITN      NBC        2017-09-01 Fri        0.0297
## 9 WJHL      CBS        2017-09-01 Fri        0.151
## 10 WNCT     CBS        2017-09-01 Fri        0.186
## # i 1,040 more rows
## # i 7 more variables: national_politics <dbl>,
```

Using ! for not

To get the left-leaning fall broadcasts, negate the `right_leaning` logical:

```
news |>
  mutate(
    right_leaning = ideology > 0,
    fall = month == "Sep" | month == "Oct" | month == "Nov"
  ) |>
  filter(!right_leaning & fall)
```

```
## # A tibble: 167 x 12
##   callsign affiliation date       weekday ideology
##   <chr>      <chr>      <date>      <ord>      <dbl>
## 1 KRBC      NBC        2017-09-01 Fri       -0.0387
## 2 KTVM      NBC        2017-09-01 Fri       -0.302
## 3 WCTI      ABC        2017-09-04 Mon       -0.00694
## 4 WEMT      FOX        2017-09-04 Mon       -0.0140
## 5 KECI      NBC        2017-09-05 Tue       -0.0294
## 6 KRCR      ABC        2017-09-05 Tue       -0.0113
## 7 KTMF      ABC        2017-09-05 Tue       -0.105
## 8 KTXS      ABC        2017-09-05 Tue       -0.0286
## 9 KWYB      ABC        2017-09-05 Tue       -0.0462
## 10 WCTI     ABC        2017-09-05 Tue       -0.0313
```

Order of operations

Why doesn't this work:

```
news |>  
  filter(month == "Sep" | "Oct")
```

```
## Error in `filter()`:  
## i In argument: `month == "Sep" | "Oct"`.  
## Caused by error in `month == "Sep" | "Oct"`:  
## ! operations are possible only for numeric, logical or complex types
```

month == "Sep" evaluates first!

More subtle bugs

```
news |>
  mutate(
    month_num = as.numeric(month)
  ) |>
  filter(month_num == 9 | 10)
```

```
## # A tibble: 2,560 x 11
##   callsign affiliation date       weekday ideology
##   <chr>      <chr>      <date>      <ord>      <dbl>
## 1 KECI      NBC        2017-06-07 Wed        0.0655
## 2 KPAX      CBS        2017-06-07 Wed        0.0853
## 3 KRBC      NBC        2017-06-07 Wed        0.0183
## 4 KTAB      CBS        2017-06-07 Wed        0.0850
## 5 KTMF      ABC        2017-06-07 Wed        0.0842
## 6 KTXS      ABC        2017-06-07 Wed       -0.000488
## 7 KAEF      ABC        2017-06-08 Thu        0.0426
## 8 KBVU      FOX        2017-06-08 Thu       -0.0860
## 9 KECI      NBC        2017-06-08 Thu        0.0902
## 10 KPAX     CBS        2017-06-08 Thu        0.0668
## # i 2,550 more rows
## # i 6 more variables: national_politics <dbl>,
## #   local_politics <dbl>, sinclair2017 <dbl>, post <dbl>,
```


all and any

`all()` tests if a vector is all TRUE and `any()` tests if any entry in a vector is true.

```
all(c(TRUE, TRUE, TRUE))
```

```
## [1] TRUE
```

```
all(c(TRUE, FALSE, FALSE))
```

```
## [1] FALSE
```

```
any(c(TRUE, FALSE, FALSE))
```

```
## [1] TRUE
```

```
any(c(FALSE, FALSE, FALSE))
```

```
## [1] FALSE
```

Grouped summaries with all/any

Can use these to summarize groups:

```
news |>
  group_by(callsign) |>
  summarize(
    any_liberal = any(ideology < 0),
    all_local = all(national_politics < local_politics)
  )
```

```
## # A tibble: 22 x 3
##   callsign any_liberal all_local
##   <chr>      <lgl>      <lgl>
## 1 KAEF      TRUE      FALSE
## 2 KBVU      TRUE      FALSE
## 3 KBZK      TRUE      FALSE
## 4 KCVU      TRUE      FALSE
## 5 KECI      TRUE      FALSE
## 6 KHSL      TRUE      FALSE
## 7 KNVN      TRUE      FALSE
## 8 KPAX      TRUE      FALSE
## 9 KRBC      TRUE      FALSE
## 10 KRCR     TRUE      FALSE
```

Converting logicals

When passed to `sum()` or `mean()`, `TRUE` is converted to 1 and `FALSE` is converted to 0.

```
sum(c(TRUE, FALSE, TRUE, FALSE))
```

```
## [1] 2
```

```
mean(c(TRUE, FALSE, TRUE, FALSE))
```

```
## [1] 0.5
```

Grouped logical summaries with sum/means

```
news |>
  group_by(callsign) |>
  summarize(
    prop_liberal = mean(ideology < 0),
    num_local_bigger = sum(national_politics < local_politics)
  )
```

```
## # A tibble: 22 x 3
##   callsign prop_liberal num_local_bigger
##   <chr>         <dbl>         <int>
## 1 KAEF          0.138           111
## 2 KBVU          0.143           31
## 3 KBZK          0.0526          11
## 4 KCVU          0.185           38
## 5 KECI          0.137           44
## 6 KHSL          0.132          127
## 7 KNVN          0.115          130
## 8 KPAX          0.0833           74
## 9 KRBC          0.196          103
## 10 KRCR         0.0992           99
## # i 12 more rows
```