Causality and observational studies

Seung-Ho An, University of Arizona

Agenda

- Review
- Observational studies (cont)
- Descriptive statistics
- Missing data
- Proportion tables
- Measurement

- Factual?
- Counterfactual?
- Why does RCT have a gold standard for social science research?
- Cross-sectional research

4. Observational Studies

Do newspaper endorsements matter

- Can newspaper endorsements change voters' minds?

Why not compare vote choice of readers of different papers?

- Problem: readers choose papers based on their previous beliefs
- Liberals $\approx$ New York Times, conservatives $\approx$ Wall Street Journal

Our case: British newspapers switching their endorsements

- Some news papers endorsing Tories in 1992 switched to Labour in 1997
- **Treated group:**: readers of Tory -> Labour papers
- **Control group:**: readers of papers who didn't switch

Variable	Description
to_labour	Read a newspaper that switched endorsement to Labour between 1992 and 1997 (1=yes, 0=no)
vote_lab_92	Did respondent vote for Labour in 1992 election (1=yes, 0=no)?
vote_lab_97	Did respondent vote for Labour in 1997 election (1=yes, 0=no)?
age	Age of respondent
male	Does the respondent identify as Male (1=yes, 0=no)
parent_labour	Does the respondent' identify as 's parents vote for Labour (1=yes, 0=no)
work_class	Does the responedent identify as working class (1=yes, 0=no)?

```r
library(tidyverse)
library(TPDdata)
newspapers
```

```
## # A tibble: 1,593 x 7
##    to_labour vote_lab_92 vote_lab_97        age  male parent_labour work_class
##        <dbl>       <dbl>       <dbl> <hvn_lbll> <dbl>         <dbl>      <dbl>
##  1         0           1           1         33     0             1          1
##  2         0           1           0         51     0             1          0
##  3         0           0           0         46     0             1          1
##  4         0           1           1         45     1             1          1
##  5         0           1           1         29     0             1          1
##  6         0           1           1         47     1             1          1
##  7         0           1           1         34     1             0          1
##  8         0           1           1         31     0             1          1
##  9         0           1           1         24     1             1          1
## 10         1           1           1         48     0             1          1
## # ... with 1,583 more rows
```

Before-and-after comparison

Compare readers of party-switching newspapers before & after switch

Advantage: all person-specific features held fixed

- comparing within a person over time

Before-and-after estimate:

$$\overline{Y}^{after}_{treated} - \overline{Y}^{before}_{treated}$$

Threat to inference: **time-varying counfounders**

- time trend: Labour just did better overall in 1997 compared to 1992

```
newspapers |>
  mutate(
    vote_change = vote_lab_97 - vote_lab_92
  ) |>
  summarize(avg_change = mean(vote_change))
```

```
## # A tibble: 1 x 1
##   avg_change
##        <dbl>
## 1      0.119
```

Key idea: use the before-and-after difference of **control group** to infer what would have happened to **treatment group** without treatment

DiD estimate:

$$\underbrace{(\overline{Y}^{after}_{treated} - \overline{Y}^{before}_{treated})}_{\text{trend in treated group}} - \underbrace{(\overline{Y}^{after}_{control} - \overline{Y}^{before}_{control})}_{\text{trend in control group}}$$

Change in treated group above and beyond the change in control group

Parallel time trend assumption

- Changes in vote of readers of non-switching papers roughly the same as changes that readers of switching papers would have been if they read non-switching papers
- Threat to inference: non-parallel trends

Difference-in-differences in R

```r
newspapers |>
  mutate(
    vote_change = vote_lab_97 - vote_lab_92,
    to_labour = if_else(to_labour ==1, "switched", "unswitched")
  ) |>
  group_by(to_labour) |>
  summarize(avg_change = mean(vote_change)) |>
  pivot_wider(
    names_from = to_labour,
    values_from = avg_change
  ) |>
  mutate(DID = switched - unswitched)
```

```
## # A tibble: 1 x 3
##   switched unswitched    DID
##      <dbl>      <dbl>  <dbl>
## 1    0.190      0.110 0.0796
```

Summarizing approaches

1. **Cross-sectional comparison**
   - Compare treated units with control unites after treatment
   - Assumption: treated and control units are comparable
   - Possible confounding
2. **Before-and-after comparison**
   - Compare the same units before and after treatment
   - Assumption: no time-varying confounding
3. **Differences-in-differences**
   - Assumption: parallel trends assumptions
   - Under this assumption, it accounts for unit-specific and time-varying confounding

All rely on assumptions that can't be verified to handle confounding

RCTs handle confounding by design (gold standard)

1. Descriptive Statistics

Lots of data

```r
library(tidyverse)
library(gapminder)
gapminder
```

```
## # A tibble: 1,704 x 6
##    country     continent  year lifeExp      pop gdpPercap
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779.
##  2 Afghanistan Asia       1957    30.3  9240934      821.
##  3 Afghanistan Asia       1962    32.0 10267083      853.
##  4 Afghanistan Asia       1967    34.0 11537966      836.
##  5 Afghanistan Asia       1972    36.1 13079460      740.
##  6 Afghanistan Asia       1977    38.4 14880372      786.
##  7 Afghanistan Asia       1982    39.9 12881816      978.
##  8 Afghanistan Asia       1987    40.8 13867957      852.
##  9 Afghanistan Asia       1992    41.7 16317921      649.
## 10 Afghanistan Asia       1997    41.8 22227415      635.
## # ... with 1,694 more rows
```

Lots and lots of data

```
head(gapminder$gdpPercap, n = 200)
```

```
##   [1]   779.4453   820.8530   853.1007   836.1971   739.9811   786.1134
##   [7]   978.0114   852.3959   649.3414   635.3414   726.7341   974.5803
##  [13]  1601.0561  1942.2842  2312.8890  2760.1969  3313.4222  3533.0039
##  [19]  3630.8807  3738.9327  2497.4379  3193.0546  4604.2117  5937.0295
##  [25]  2449.0082  3013.9760  2550.8169  3246.9918  4182.6638  4910.4168
##  [31]  5745.1602  5681.3585  5023.2166  4797.2951  5288.0404  6223.3675
##  [37]  3520.6103  3827.9405  4269.2767  5522.7764  5473.2880  3008.6474
##  [43]  2756.9537  2430.2083  2627.8457  2277.1409  2773.2873  4797.2313
##  [49]  5911.3151  6856.8562  7133.1660  8052.9530  9443.0385 10079.0267
##  [55]  8997.8974  9139.6714  9308.4187 10967.2820  8797.6407 12779.3796
##  [61] 10039.5956 10949.6496 12217.2269 14526.1246 16788.6295 18334.1975
##  [67] 19477.0093 21888.8890 23424.7668 26997.9366 30687.7547 34435.3674
##  [73]  6137.0765  8842.5980 10750.7211 12834.6024 16661.6256 19749.4223
##  [79] 21597.0836 23687.8261 27042.0187 29095.9207 32417.6077 36126.4927
##  [85]  9867.0848 11635.7995 12753.2751 14804.6727 18268.6584 19340.1020
##  [91] 19211.1473 18524.0241 19035.5792 20292.0168 23403.5593 29796.0483
##  [97]   684.2442   661.6375   686.3416   721.1861   630.2336   659.8772
## [103]   676.9819   751.9794   837.8102   972.7700  1136.3904  1391.2538
## [109]  8343.1051  9714.9606 10991.2068 13149.0412 16672.1436 19117.9745
## [115] 20979.8459 22525.5631 25575.5707 27561.1966 30485.8838 33692.6051
## [121]  1062.7522   959.6011   949.4991  1035.8314  1085.7969  1029.1613
## [127]  1277.8976  1225.8560  1191.2077  1232.9753  1372.8779  1441.2849
## [133]  2677.3263  2127.6863  2180.9725  2586.8861  2980.3313  3548.0978
## [139]  3156.5105  2753.6915  2961.6997  3326.1432  3413.2627  3822.1371
```

- How should we summarize the wages data? Many possibilities!
  - Up to now: focus on **averages** or means of variables
- Two salient features of a variable that we want to know:
  - **Central tendency**: where is the middle/typical/average value
  - **Spread** around the center: are all values to the center or spread out?

- "Center" of the data: typical/average value
- **Mean**: Sum of the values divided by the number of observations

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

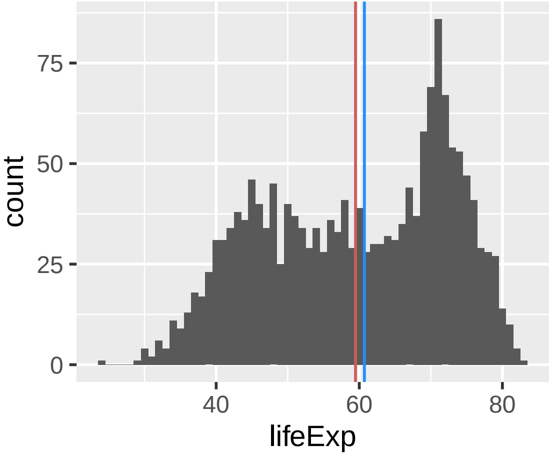
- **Median**:

$$\text{median} = \begin{cases} \text{middle value} & \text{if number of entries is odd} \\ \frac{\text{sum of two middle values}}{2} & \text{if number of entries is even} \end{cases}$$

- In **R**: mean() and median()

- Median more robust to **outliers**:
    - Example 1: data $= \{0, 1, 2, 3, 5\}$. Mean? Median?
    - Example 2 data$= \{0, 1, 2, 3, 100\}$. Mean? Median?
- What does Mark Zuckerberg do the mean vs. median income?

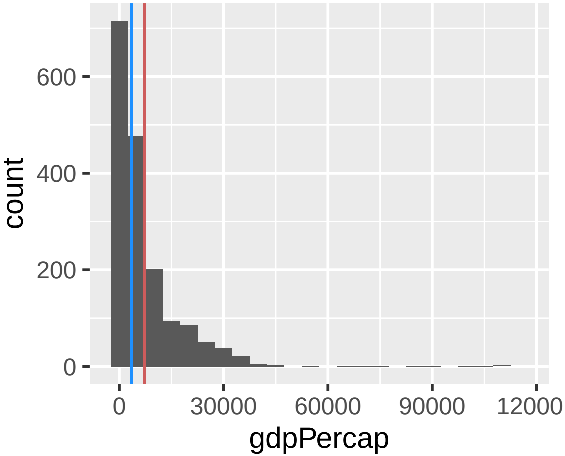
```r
ggplot(gapminder, aes(x = lifeExp)) +
  geom_histogram(binwidth = 1) +
  geom_vline(aes(xintercept = mean(lifeExp)), color = "indianred") +
  geom_vline(aes(xintercept = median(lifeExp)), color = "dodgerblue")
```




```r
summary(gapminder$lifeExp)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   23.60   48.20   60.71   59.47   70.85   82.60
```

```r
ggplot(gapminder, aes(x = gdpPercap)) +
  geom_histogram(binwidth = 5000) +
  geom_vline(aes(xintercept = mean(gdpPercap)), color = "indianred") +
  geom_vline(aes(xintercept = median(gdpPercap)), color = "dodgerblue")
```

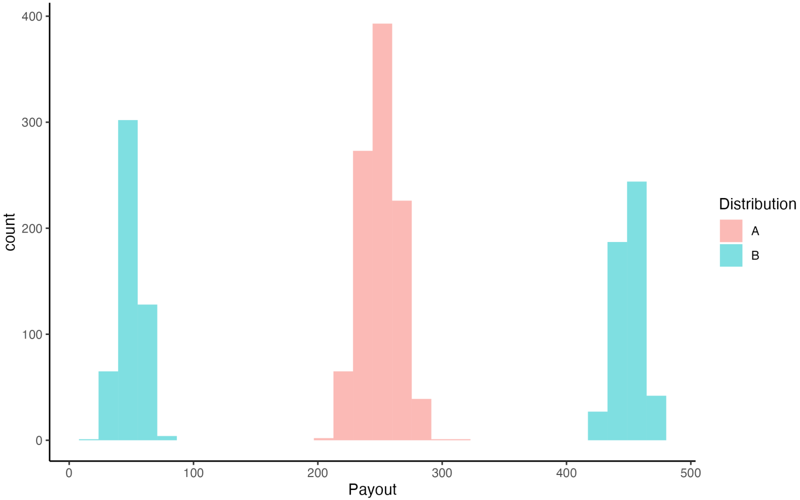



```r
summary(gapminder$gdpPercap)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##    241.2   1202.1   3531.8   7215.3   9325.5 113523.1
```

Which distribution would you prefer?

Lottery where we randomly draw one value from A or B:



They have the same mean, so why do we care about the difference? **Spread!!**

- Are the values of the variable close to the center?
- **Range**: [min(X), max(X)]
- **Quantile** (quartile, percentile, etc.): divide data into equal sized groups
  - 25th percentile = lower quartile (25% of the data below this value)
  - 50th percentile = median (50% of the data below this value)
  - 75th percentile = upper quartile (75% of the data below this value)
- **Interquartile range** (IQR): a measure of variability
  - How spread out is the middle half of the data?
  - Is most of the data really close to the median or are the values spread out?
- **R** function: range(), summary(), and IQR()

- **Standard deviation**: On average, how far away are data points from the mean?

$$\text{standard deviation} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

- Steps:
  1. Subtract each data point by the mean
  2. Square each resulting difference
  3. Take the sum of these values
  4. Divide by n-1 (or n, does not matter much)
  5. Take the square root
- **Variance** $=$ standard deviation^2
- Why not just take the average deviations from mean without squaring?

2. Missing data

Missing data

- **Nonresponse**: respondent can't or won't answer question
  - Sensitive questions $\approx$ **social desirability bias**
  - Some countries lack official statistics like unemployment
  - Leads to missing data
- Missing data in **R**: a special value NA
- Have already seen how to use `na.rm= TRUE`

```r
library(TPDdata)
cces_2020
```

```
## # A tibble: 51,551 x 6
##    gender race  educ                 pid3        turnout_self pres_vote
##    <fct>  <fct> <fct>                <fct>              <dbl> <fct>
##  1 Male   White 2-year               Republican             1 Donald J. Trump (~
##  2 Female White Post-grad            Democrat              NA <NA>
##  3 Female White 4-year               Independent            1 Joe Biden (Democr~
##  4 Female White 4-year               Democrat               1 Joe Biden (Democr~
##  5 Male   White 4-year               Independent            1 Other
##  6 Male   White Some college         Republican             1 Donald J. Trump (~
##  7 Male   Black Some college         Not sure              NA <NA>
##  8 Female White Some college         Independent            1 Donald J. Trump (~
##  9 Female White High school graduate Republican             1 Donald J. Trump (~
## 10 Female White 4-year               Democrat               1 Joe Biden (Democr~
## # ... with 51,541 more rows
```

drop_na() to remove rows with missing values

```r
cces_2020 |>
  drop_na()
```

```
## # A tibble: 45,651 x 6
##    gender race  educ                 pid3        turnout_self pres_vote
##    <fct>  <fct> <fct>                <fct>              <dbl> <fct>
##  1 Male   White 2-year               Republican             1 Donald J. Trump (~
##  2 Female White 4-year               Independent            1 Joe Biden (Democr~
##  3 Female White 4-year               Democrat               1 Joe Biden (Democr~
##  4 Male   White 4-year               Independent            1 Other
##  5 Male   White Some college         Republican             1 Donald J. Trump (~
##  6 Female White Some college         Independent            1 Donald J. Trump (~
##  7 Female White High school graduate Republican             1 Donald J. Trump (~
##  8 Female White 4-year               Democrat               1 Joe Biden (Democr~
##  9 Female White 4-year               Democrat               1 Joe Biden (Democr~
## 10 Female White 4-year               Democrat               1 Joe Biden (Democr~
## # ... with 45,641 more rows
```

Drop rows based on certain variables

```
cces_2020 |>
  dim_desc()
```

```
## [1] "[51,551 x 6]"
```

```
cces_2020 |>
  drop_na() |>
  dim_desc()
```

```
## [1] "[45,651 x 6]"
```

```
cces_2020 |>
  drop_na(turnout_self) |>
  dim_desc()
```

```
## [1] "[48,462 x 6]"
```

Available-case vs. complete-case analysis

Available-case analysis: use the data you have for that variable:

```
cces_2020 |>
  summarize(mean(turnout_self, na.rm=TRUE)) |>
  pull()
```

```
## [1] 0.9421815
```

Complete-case analysis: only use units that have data on all variables

```
cces_2020 |>
  drop_na() |>
  summarize(mean(turnout_self)) |>
  pull()
```

```
## [1] 0.9994524
```

(also called **listwise deletion**)

Trying to detect missingness with == doesn't work:

```
c(5, 6, NA, 0) == NA
```

```
## [1] NA NA NA NA
```

use `is.na()` instead:

```
is.na(c(5, 6, NA, 0))
```

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

Can use `sum()` or `mean()` on this to get number/proportion missing:

```
sum(is.na(c(5, 6, NA, 0)))
```

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

Nonresponse can create bias if lower turnout -> more non-response:

```
cces_2020 |>
  group_by(pid3) |>
  summarize(
    mean_turnout = mean(turnout_self, na.rm = TRUE),
    missing_turnout = mean(is.na(turnout_self))
  )
```

```
## # A tibble: 5 x 3
##   pid3        mean_turnout missing_turnout
##   <fct>              <dbl>           <dbl>
## 1 Democrat           0.963          0.0280
## 2 Republican         0.953          0.0403
## 3 Independent        0.924          0.0718
## 4 Other              0.957          0.0709
## 5 Not sure           0.630          0.431
```

3. Proportion tables

First, let's review how to get counts:

```
cces_2020 |>
  group_by(pres_vote) |>
  summarize(n = n())
```

```
## # A tibble: 7 x 2
##   pres_vote                      n
##   <fct>                      <int>
## 1 Joe Biden (Democrat)       26188
## 2 Donald J. Trump (Republican) 17702
## 3 Other                       1458
## 4 I did not vote in this race  100
## 5 I did not vote                13
## 6 Not sure                     190
## 7 <NA>                        5900
```

```
cces_2020 |>
  group_by(pres_vote) |>
  summarize(prop = n() / sum(n()))
```

```
## # A tibble: 7 x 2
##   pres_vote                          prop
##   <fct>                             <dbl>
## 1 Joe Biden (Democrat)                  1
## 2 Donald J. Trump (Republican)          1
## 3 Other                                 1
## 4 I did not vote in this race           1
## 5 I did not vote                        1
## 6 Not sure                              1
## 7 <NA>                                  1
```

Inside `summarize()` all operations are done within groups!

```
cces_2020 |>
  group_by(pres_vote) |>
  summarize(n = n()) |>
  mutate (prop = n / sum(n))
```

```
## # A tibble: 7 x 3
##   pres_vote                       n     prop
##   <fct>                       <int>    <dbl>
## 1 Joe Biden (Democrat)        26188 0.508
## 2 Donald J. Trump (Republican) 17702 0.343
## 3 Other                        1458 0.0283
## 4 I did not vote in this race   100 0.00194
## 5 I did not vote                 13 0.000252
## 6 Not sure                      190 0.00369
## 7 <NA>                         5900 0.114
```

Grouping is silently dropped after `summarize()`

Another approach

```
cces_2020 |>
  group_by(pres_vote) |>
  summarize(prop = n() / nrow(cces_2020))
```

```
## # A tibble: 7 x 2
##   pres_vote                          prop
##   <fct>                             <dbl>
## 1 Joe Biden (Democrat)           0.508
## 2 Donald J. Trump (Republican) 0.343
## 3 Other                          0.0283
## 4 I did not vote in this race  0.00194
## 5 I did not vote               0.000252
## 6 Not sure                       0.00369
## 7 <NA>                           0.114
```

Doesn't work if you have filtered the data in any way during the pipe

What happens with multiple grouping variables?

```
vote_by_party <- cces_2020 |>
  filter(pres_vote %in% c("Joe Biden (Democrat)",
                          "Donald J. Trump (Republican)")) |>
  mutate(pres_vote = if_else(pres_vote == "Joe Biden (Democrat)",
                             "Biden", "Trump")) |>
  group_by(pid3, pres_vote) |>
  summarize (n = n()) |>
  mutate(prop = n / sum(n)) |>
  select(-n)

vote_by_party
```

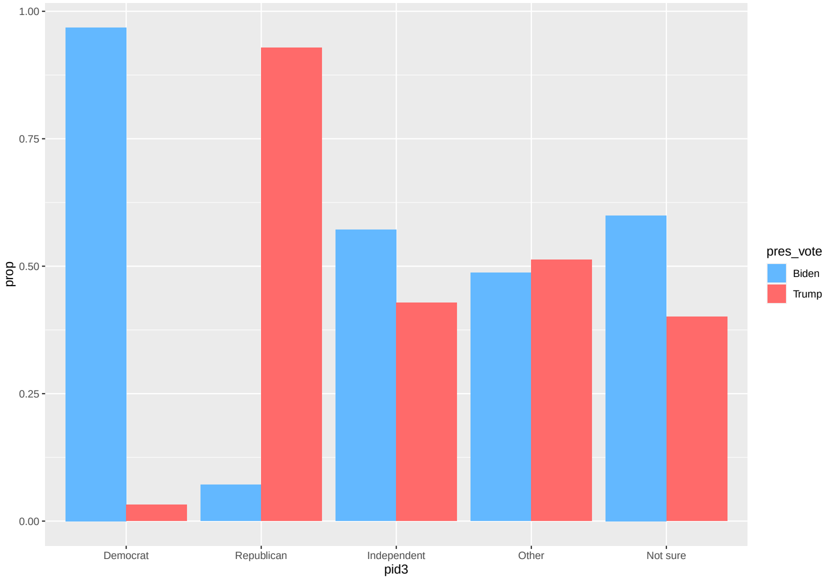
```
## `summarise()` has grouped output by 'pid3'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 10 x 3
## # Groups:   pid3 [5]
##    pid3        pres_vote   prop
##    <fct>       <chr>      <dbl>
##  1 Democrat    Biden     0.968
##  2 Democrat    Trump     0.0319
##  3 Republican  Biden     0.0712
##  4 Republican  Trump     0.929
##  5 Independent Biden     0.571
##  6 Independent Trump     0.429
##  7 Other       Biden     0.487
##  8 Other       Trump     0.513
##  9 Not sure    Biden     0.599
## 10 Not sure    Trump     0.401
```

With multiple grouping variables, `summarize()` drops the last one

We can visualize this using the `fill` aesthetic and `position="dodge"`:

```
ggplot(vote_by_party,
       aes( x = pid3, y = prop, fill = pres_vote)) +
  geom_col(position = "dodge") +
  scale_fill_manual(values = c(Biden = "steelblue1", Trump = "indianred1"))
```


1.00
0.75
0.50
0.25
0.00
prop
Democrat
Republican
Independent
Other
Not sure
pid3
pres_vote
Biden
Trump

Pivoting to create cross-tab

```r
vote_by_party <- cces_2020 |>
  filter(pres_vote %in% c("Joe Biden (Democrat)",
                          "Donald J. Trump (Republican)")) |>
  mutate(pres_vote = if_else(pres_vote == "Joe Biden (Democrat)",
                             "Biden", "Trump")) |>
  group_by(pid3, pres_vote) |>
  summarize (n = n()) |>
  mutate(prop = n / sum(n)) |>
  select(-n) |>
  pivot_wider(
    names_from = pid3,
    values_from = prop
  )

vote_by_party
```

```
## `summarise()` has grouped output by 'pid3'. You can override using the
## `.groups` argument.

## # A tibble: 2 x 6
##   pres_vote Democrat Republican Independent Other `Not sure`
##   <chr>        <dbl>      <dbl>       <dbl> <dbl>      <dbl>
## 1 Biden       0.968      0.0712       0.571 0.487      0.599
## 2 Trump       0.0319     0.929        0.429 0.513      0.401
```

What if we want row proportions?

```
vote_by_party <- cces_2020 |>
  filter(pres_vote %in% c("Joe Biden (Democrat)",
                          "Donald J. Trump (Republican)")) |>
  mutate(pres_vote = if_else(pres_vote == "Joe Biden (Democrat)",
                             "Biden", "Trump")) |>
  group_by(pres_vote, pid3) |>
  summarize (n = n()) |>
  mutate(prop = n / sum(n)) |>
  select(-n) |>
  pivot_wider(
    names_from = pid3,
    values_from = prop
  )
```

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

```
## # A tibble: 2 x 6
## # Groups:   pres_vote [2]
##   pres_vote Democrat Republican Independent  Other `Not sure`
##   <chr>        <dbl>      <dbl>       <dbl>  <dbl>      <dbl>
## 1 Biden       0.674      0.0327       0.252 0.0281     0.0133
## 2 Trump       0.0328     0.631        0.280 0.0437     0.0131
```

Proportion of all observations

If we want the proportion of all rows, drop all groups

```
vote_by_party <- cces_2020 |>
  filter(pres_vote %in% c("Joe Biden (Democrat)",
                          "Donald J. Trump (Republican)")) |>
  mutate(pres_vote = if_else(pres_vote == "Joe Biden (Democrat)",
                             "Biden", "Trump")) |>
  group_by(pid3, pres_vote) |>
  summarize (n = n(), .groups="drop") |>
  mutate(prop = n / sum(n)) |>
  select(-n) |>
  pivot_wider(
    names_from = pid3,
    values_from = prop
  )
```

```
## # A tibble: 2 x 6
##   pres_vote Democrat Republican Independent  Other `Not sure`
##   <chr>        <dbl>      <dbl>       <dbl>  <dbl>      <dbl>
## 1 Biden       0.402      0.0195       0.150 0.0167    0.00791
## 2 Trump       0.0132     0.254        0.113 0.0176    0.00529
```

4. Measurement

Where does data come from?

- Social science is about developing and testing **causal theories:**
  - Does minimum wage change levels of employment?
  - Does outgroup contact influence views on immigration?
- Theories are made up of **concepts**:
  - Minimum wage, level of employment, outgroup contact, views on immigration
  - We took these for granted when talking about causality
- Need **operational definition** to concretely measure these concepts

Concepts vary in how observable they are

Kinds of measurement arranged by how direct we can measure them:

Observable in the world

- Minimum wage laws
- Election results
- Crime rates

Observable by survey

- Age of a person
- Employment status
- Presidential approval

Not directly observable

- A person's ideology
- Levels of democracy

- Concept: presidential approval
- Conceptual definition:
  - Extent to which US adults support the actions and policies of the current US president
- Operational definition:
  - "On a scale from 1 to 5, where 1 is least supportive and 5 is more supportive, how much would you say you support the job that Joe Biden is doing as president?"

Measurement error

Table 1: Response to Citizenship Question Across Two-Waves of CCES Panel			
Response in 2010	Response in 2012	Number of Respondents	Percentage
Citizen	Citizen	18,737	99.25
Citizen	Non-Citizen	20	0.11
Non-Citizen	Citizen	36	0.19
Non-Citizen	Non-Citizen	85	0.45

- **Measurement error**: chance variation in our measurements
  - individual measurement = exact value + chance error
  - chance errors tend to cancel out when we take averages
  - why do these occur? often data entry errors or faulty memories

Bias


TRUMP PENCE
TRUMP MAKE AMERICA GREAT AGAIN COMMITTEE
45

Official Presidential Job Performance Survey

1. How would you rate President Trump's job performance so far?

- Great
- Good
- Okay
- Other

2. (Optional) Please explain why you selected your response.

Bias: systematic errors for all units in the same direction

- Individual measurement = exact value + bias + chance error
- "What did you eat yesterday?' ' $\rightsquigarrow$ underreporting

Literary Digest predicted elections using mail-in polls
Source of addresses: automobile registrations, phone books, etc

In 1936, sent out 10 million ballots, over 2.3 million returned

George Gappup used only 50,000 respondents

Polling organization	FDR's vote share
Literary Digest	43
George Gallup	56

Polling organization	FDR %
Literary Digest	43
George Gallup	56
Actual Outcome	62

Selection bias: ballots skewed toward the wealthy (with cars, phones)

- Only 1 in 4 households had a phone in 1936

Nonresponse bias: respondents differ from nonrespondents

- $\rightsquigarrow$ when selection procedure is biased, adding more units won't help


Chicago Daily Tribune
DEWEY DEFEATS TRUMAN
G.O.P. Sweep Indicated in State; Boyle Leads in City

Polling organization	Truman	Dewey	Thrumond	Wallace
Crossley	45	50	2	3
Gallup	44	50	2	4
Roper	38	53	5	4
Actual outcome	50	45	3	2

Quota sampling: fixed quota of certain respondents for each interviewer

- If black women make up 5% of the population, stop interviewing them once they make up 5% of your sample

Sample resembles the population on these characteristics

Potential unobserved confounding $\rightsquigarrow$ **selection bias**

Republicans easier to find within quotas (phones, listed addresses)

Probability sampling to ensure representatives

- Definition: every unit in the population has a know, non-zero probability of being selected into sample

Simple random sampling: every unit has an **equal** selection probability

Random digit dialing:

- Take a particular area code + exchange: 520-290-XXXX
- Randomly choose each digit in XXXX to call a particular phone
- Every phone in America has an equal chance of being included in sample

Target population: set of people we want to learn about

- Ex: people who will vote in the next elections

Sampling frame: list of people from which we will actually sample

- Frame bias: list of registered voters (frame) might include nonvoters!

Sample: set of people contacted

Respondents: subset of sample that actually responds to the survey

- Unit non-response: sample $\neq$ respondents
- Not everyone picks up their phone

Completed items: subset of questions that respondents answer

- Item non-response: refusing to disclose their vote preference

Problems of telephone survey

- Cell phone (double counting for the wealthy)
- Caller ID screening (unit non-response)
- Response rates down to 9%

An alternative: Internet surveys

- Opt-in panels, respondent-driven sampling $\rightsquigarrow$ **non-probability sampling**
- Cheaper, but non-representative
- Digital divide: rich vs. poor, young vs. old
- Correct for potential sampling bias via statistical methods