Gov 50: 22. More hypothesis testing

Matthew Blackwell

Harvard University

Roadmap

1. Two-sample tests
2. Two-sample permutation tests with infer
3. Issues with hypothesis testing

1/ Two-sample tests

Statistical hypothesis testing

- Statistical hypothesis testing is a **thought experiment**.
- What would the world look like **if we knew the truth**?
- Conducted with several steps:

    1. Specify your **null** and **alternative hypotheses**
    2. Choose an appropriate **test statistic** and level of test α
    3. Derive the **reference distribution** of the test statistic under the null.
    4. Use this distribution to calculate the **p-value**.
    5. Use p-value to decide whether to reject the null hypothesis or not

Social pressure experiment

- Experimental study where each household for 2006 MI primary was randomly assigned to one of 4 conditions:
    - Control: no mailer
    - Civic Duty: mailer saying voting is your civic duty.
    - Hawthorne: a “we’re watching you” message.
    - Neighbors: naming-and-shaming social pressure mailer.
- Outcome: whether household members voted or not.
- We’ll focus on Neighbors vs Control
- Randomized implies samples are **independent**

Neighbors mailer

Dear Registered Voter:

WHAT IF YOUR NEIGHBORS KNEW WHETHER YOU VOTED?

Why do so many people fail to vote? We've been talking about the problem for years, but it only seems to get worse. This year, we're taking a new approach. We're sending this mailing to you and your neighbors to publicize who does and does not vote.

The chart shows the names of some of your neighbors, showing which have voted in the past. After the August 8 election, we intend to mail an updated chart. You and your neighbors will all know who voted and who did not.

DO YOUR CIVIC DUTY — VOTE!

MAPLE DR	Aug 04	Nov 04	Aug 06
9995 JOSEPH JAMES SMITH	Voted	Voted	______
9995 JENNIFER KAY SMITH		Voted	______
9997 RICHARD B JACKSON		Voted	______
9999 KATHY MARIE JACKSON		Voted	______

Social pressure data

```
library(infer)
data(social, package = "qss")
social <- as_tibble(social)
social
```

```
## # A tibble: 305,866 x 6
##    sex   yearofbirth primary2004 messages primary2006 hhsize
##    <chr>       <int>       <int> <chr>          <int>  <int>
##  1 male         1941           0 Civic D~           0      2
##  2 fema~        1947           0 Civic D~           0      2
##  3 male         1951           0 Hawthor~           1      3
##  4 fema~        1950           0 Hawthor~           1      3
##  5 fema~        1982           0 Hawthor~           1      3
##  6 male         1981           0 Control            0      3
##  7 fema~        1959           0 Control            1      3
##  8 male         1956           0 Control            1      3
##  9 fema~        1968           0 Control            0      2
## 10 male         1967           0 Control            0      2
## # i 305,856 more rows
```

Two-sample hypotheses

- Parameter: **population ATE** $\mu_T - \mu_C$
  - μ_T: Turnout rate in the population if everyone received treatment.
  - μ_C: Turnout rate in the population if everyone received control.
- Goal: learn about the population difference in means
- Usual null hypothesis: no difference in population means (ATE = 0)
  - Null: $H_0 : \mu_T - \mu_C = 0$
  - Two-sided alternative: $H_1 : \mu_T - \mu_C \neq 0$
- In words: are the differences in sample means just due to chance?

Permutation test

How do we generate draws of the difference in means under the null?
$H_0 : \mu_T - \mu_C = 0$

If the voting distribution is the same in the treatment and control groups, we could randomly swap who is labelled as treated and who is labelled as control and it shouldn't matter.

Permutation test: generate the null distribution by permuting the group labels and see the resulting distribution of differences in proportions

Permuting the labels

```
social <- social |>
  filter(messages %in% c("Neighbors", "Control"))

social |>
  mutate(messages_permute = sample(messages)) |>
  select(primary2006, messages, messages_permute)
```

```
## # A tibble: 229,444 x 3
##    primary2006 messages messages_permute
##          <int> <chr>    <chr>
##  1           0 Control  Control
##  2           1 Control  Control
##  3           1 Control  Neighbors
##  4           0 Control  Control
##  5           0 Control  Control
##  6           1 Control  Neighbors
##  7           0 Control  Control
##  8           1 Control  Control
##  9           1 Control  Control
## 10           1 Control  Control
## # i 229,434 more rows
```

2/ Two-sample permutation tests with infer

Calculating the difference in proportion

infer functions with binary outcomes work best with factor variables:

```
social <- social |>
  mutate(turnout = if_else(primary2006 == 1, "Voted", "Didn't Vote"))

est_ate <- social |>
  specify(turnout ~ messages, success = "Voted") |>
  calculate(stat = "diff in props", order = c("Neighbors", "Control"))
est_ate
```

```
## Response: turnout (factor)
## Explanatory: messages (factor)
## # A tibble: 1 x 1
##     stat
##    <dbl>
## 1 0.0813
```

Specifying the relationship of interest

infer functions with binary outcomes work best with factor variables:

```r
social |>
  specify(turnout ~ messages, success = "Voted")
```

```
## Response: turnout (factor)
## Explanatory: messages (factor)
## # A tibble: 229,444 x 2
##    turnout     messages
##    <fct>       <fct>
##  1 Didn't Vote Control
##  2 Voted       Control
##  3 Voted       Control
##  4 Didn't Vote Control
##  5 Didn't Vote Control
##  6 Voted       Control
##  7 Didn't Vote Control
##  8 Voted       Control
##  9 Voted       Control
## 10 Voted       Control
## # i 229,434 more rows
```

Setting the hypotheses

The null for these two-sample tests is called `"independence"` for the `infer` package because the assumption is that the two variables are statistically independent.

```
social |>
  specify(turnout ~ messages, success = "Voted") |>
  hypothesize(null = "independence")
```

```
## Response: turnout (factor)
## Explanatory: messages (factor)
## Null Hypothesis: independence
## # A tibble: 229,444 x 2
##    turnout     messages
##    <fct>       <fct>
##  1 Didn't Vote Control
##  2 Voted       Control
##  3 Voted       Control
##  4 Didn't Vote Control
##  5 Didn't Vote Control
##  6 Voted       Control
##  7 Didn't Vote Control
##  8 Voted       Control
```

Generating the permutations

We can tell `infer` to do our permutation test by using the argument `type = "permute"` to `generate()`:

```
social |>
  specify(turnout ~ messages, success = "Voted") |>
  hypothesize(null = "independence") |>
  generate(reps = 1000, type = "permute")
```

```
## Response: turnout (factor)
## Explanatory: messages (factor)
## Null Hypothesis: independence
## # A tibble: 229,444,000 x 3
## # Groups:   replicate [1,000]
##    turnout     messages replicate
##    <fct>       <fct>        <int>
##  1 Voted       Control          1
##  2 Didn't Vote Control          1
##  3 Voted       Control          1
##  4 Didn't Vote Control          1
##  5 Didn't Vote Control          1
##  6 Voted       Control          1
##  7 Voted       Control          1
```

Calculating the diff in proportions in each sample

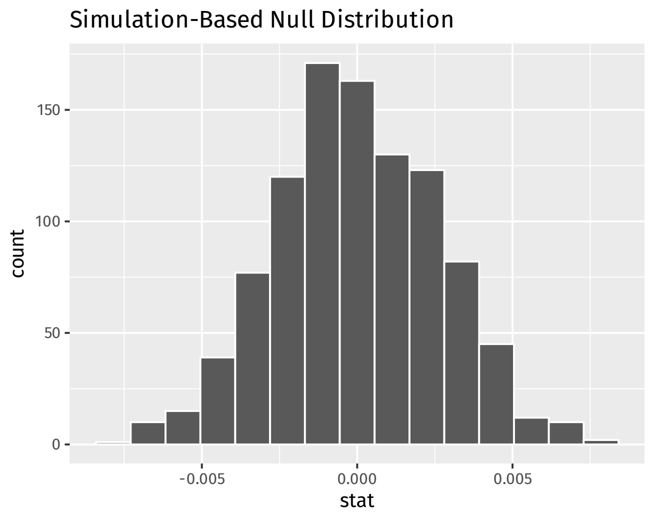
```
null_dist <- social |>
  specify(turnout ~ messages, success = "Voted") |>
  hypothesize(null = "independence") |>
  generate(reps = 1000, type = "permute") |>
  calculate(stat = "diff in props", order = c("Neighbors", "Control"))
```

```r
null_dist
```

```
## Response: turnout (factor)
## Explanatory: messages (factor)
## Null Hypothesis: independence
## # A tibble: 1,000 x 2
##    replicate      stat
##        <int>     <dbl>
##  1         1  0.00217
##  2         2 -0.00606
##  3         3  0.00286
##  4         4  0.00204
##  5         5 -0.000943
##  6         6 -0.00298
##  7         7  0.00311
##  8         8 -0.000315
##  9         9 -0.00126
## 10        10 -0.000912
## # i 990 more rows
```

Visualizing

```
null_dist |>
  visualize()
```



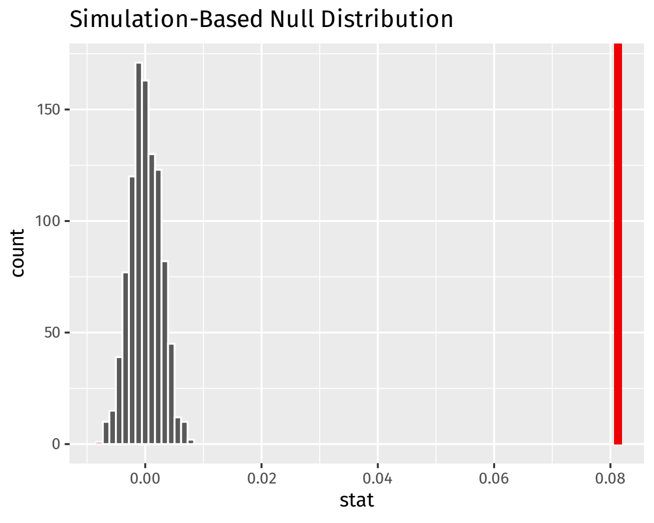
Calculating p-values

```r
ate_pval <- null_dist |>
  get_p_value(obs_stat = est_ate, direction = "both")
ate_pval
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```

Visualizing p-values

```
null_dist |>
  visualize() +
  shade_p_value(obs_stat = est_ate, direction = "both")
```



Tests and confidence intervals

- There is a deep connection between confidence intervals and tests.
- Any value outside of a $100 \times (1 - \alpha)\%$ confidence interval would have a p-value less than α if we tested it as the null hypothesis.
    - 95% CI for social pressure experiment: $[0.016, 0.124]$
    - $\rightsquigarrow$ p-value for $H_0 : \mu_T - \mu_C = 0$ less than 0.05.
- Confidence intervals are all of the null hypotheses we **can't reject** with a test.

CI in the trains example

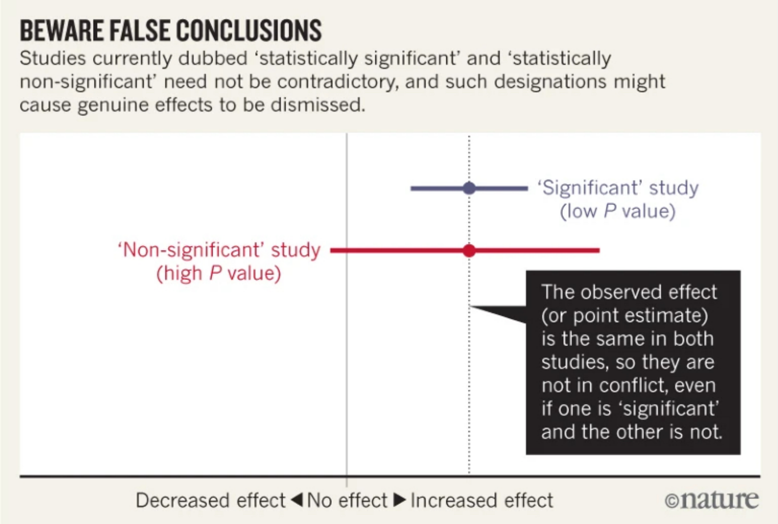
```
social |>
  specify(turnout ~ messages, success = "Voted") |>
  generate(reps = 1000, type = "bootstrap") |>
  calculate(stat = "diff in props",
            order = c("Neighbors", "Control"))  |>
  get_ci(level = 0.95)
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   0.0760   0.0867
```

3/ Issues with hypothesis testing

Significant vs not significant

The difference between statistically significant and not statistically significant is itself not statistically significant:

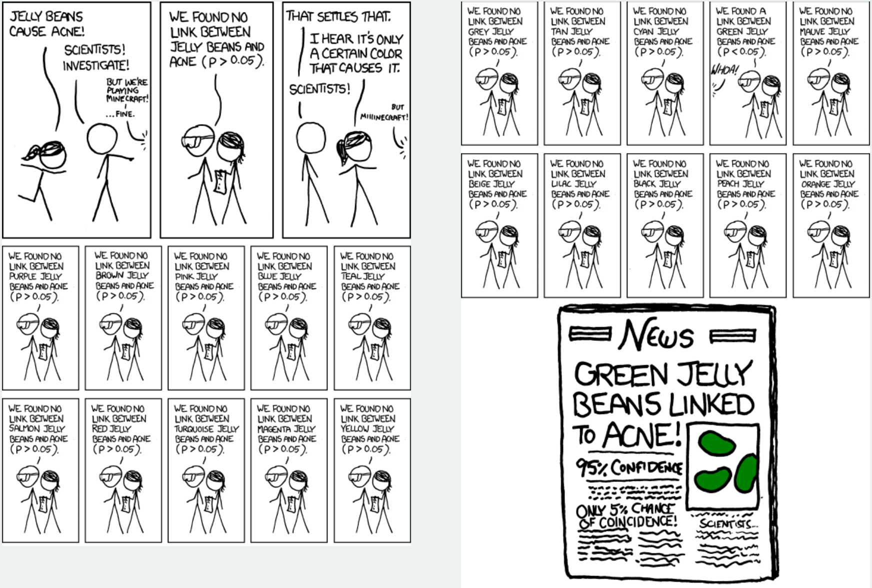


What kind of significance

There are different types of significance that don't all have to be true together:

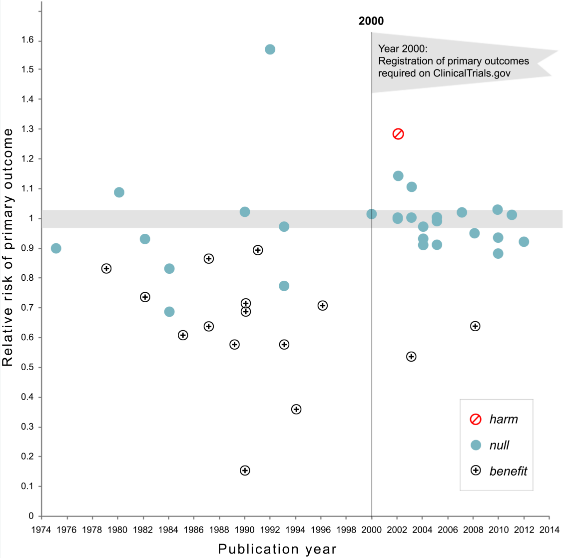
1. **Statistical significance:** we can reject the null of no effect.
2. **Causal significance**: we can interpret our estimated difference in means as a causal effect.
3. **Practical significance**: the estimated effect is meaningfully large.

p-hacking



JELLY BEANS CAUSE ACNE!
SCIENTISTS! INVESTIGATE!
BUT WE'RE PLAYING MINECRAFT!
... FINE.
WE FOUND NO LINK BETWEEN JELLY BEANS AND ACNE (P > 0.05).
THAT SETTLES THAT.
I HEAR IT'S ONLY A CERTAIN COLOR THAT CAUSES IT.
SCIENTISTS!
BUT MINECRAFT!
WE FOUND NO LINK BETWEEN PURPLE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BROWN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN PINK JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BLUE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TEAL JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN SALMON JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN RED JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TURQUOISE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN MAGENTA JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN YELLOW JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN GREY JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN TAN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN CYAN JELLY BEANS AND ACNE (P > 0.05).
WE FOUND A LINK BETWEEN GREEN JELLY BEANS AND ACNE (P < 0.05).
WHOA!
WE FOUND NO LINK BETWEEN MAUVE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BEIGE JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN LILAC JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN BLACK JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN PEACH JELLY BEANS AND ACNE (P > 0.05).
WE FOUND NO LINK BETWEEN ORANGE JELLY BEANS AND ACNE (P > 0.05).
NEWS
GREEN JELLY BEANS LINKED TO ACNE!
95% CONFIDENCE
ONLY 5% CHANCE OF COINCIDENCE!
SCIENTISTS...

p-hacking


2000
Year 2000:
Registration of primary outcomes
required on ClinicalTrials.gov
Relative risk of primary outcome
1.6
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1974 1976 1978 1980 1982 1984 1986 1988 1990 1992 1994 1996 1998 2000 2002 2004 2006 2008 2010 2012 2014
Publication year
harm
null
benefit