POLS 602 PS1

Julia Reyes

SIMULATION

Setup

```
set.seed(25)

library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.1     v stringr   1.5.2
## v ggplot2   3.5.2     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

Create population with traits and proportions

```
NFL_Teams <- c("Cowboys", "Packers", "Bills", "Ravens", "Lions")

p_prop <- c(0.1, 0.3, 0.3, 0.1, 0.2)

names(p_prop) <- NFL_Teams
```

Sample n observations from population randomly

```
sample1 <- sample(NFL_Teams, 100, replace = TRUE)

sample2 <- sample(NFL_Teams, 500, replace = TRUE)

sample3 <- sample(NFL_Teams, 1000, replace = TRUE)
```

Assign each observation randomly to the control or treatment group

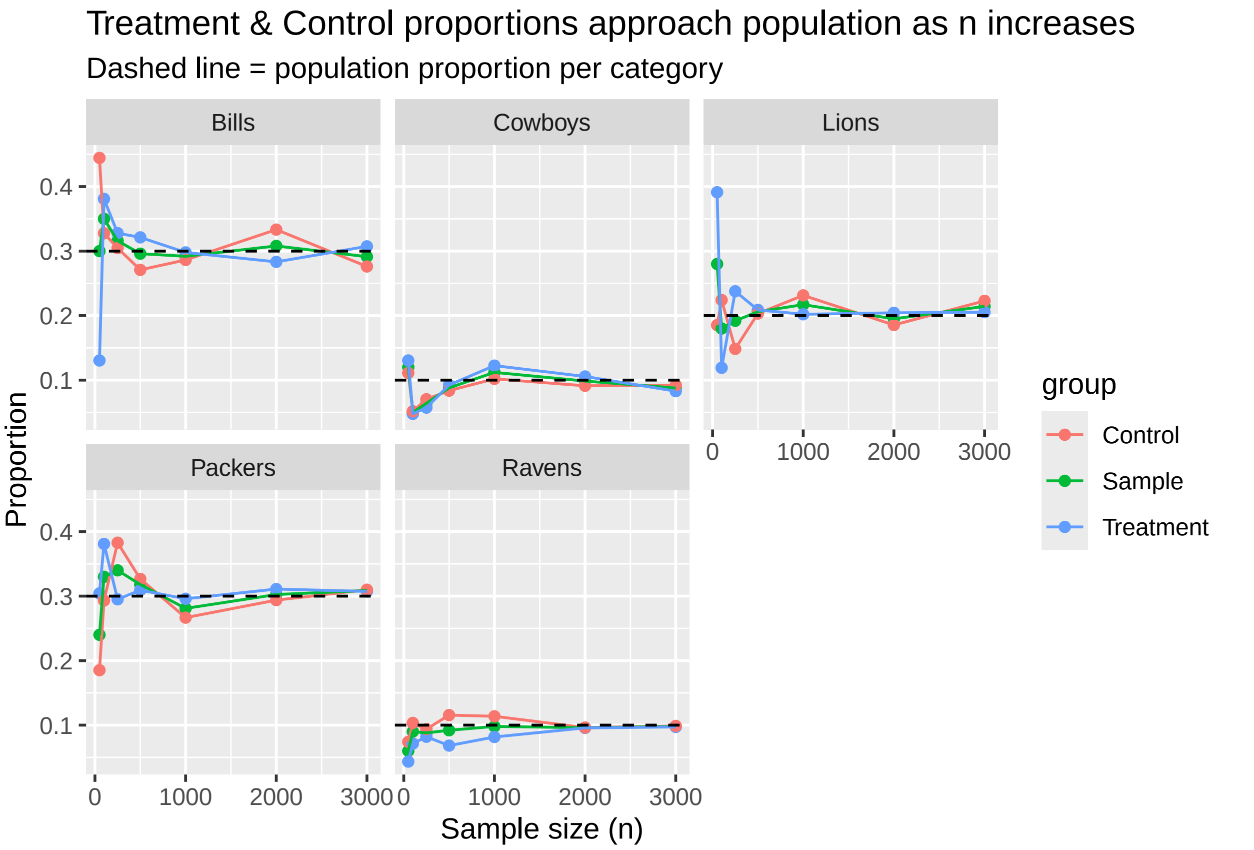
```
group <- sample(c("treatment", "control"), 1000, replace = TRUE)
```

Create container for results

```
results <- data.frame(n = integer(), group = character(), prop = numeric())
```

Repeat this process for different sample sizes

```
n_values <- c(50, 100, 250, 500, 1000, 2000, 3000)

for(n in n_values) {
 # drawing samples from population
  team <- sample(NFL_Teams, size = n, replace = TRUE, prob = p_prop)

  # randomly assign 1 for treatment group, and 0 for control group
  treatment <- rbinom(n, 1, 0.5)

  # proportions for sample of n values
  prop_sample <- as.numeric(table(factor(team, levels = NFL_Teams))) / n

  # Number of observations assigned to treatment group
  n_treatment <- sum(treatment == 1)

  # Number of observations assigned to the control group
  n_control <- n - n_treatment

  # proportions of observations in treatmeant and control groups
  prop_treatment <- as.numeric(table(factor(team[treatment==1], levels = NFL_Teams))) / n_treatment

  prop_control <- as.numeric(table(factor(team[treatment==0], levels = NFL_Teams))) / n_control

  results <- bind_rows(results,
                       data.frame(n = n, group = "Sample", team = NFL_Teams, prop = prop_sample),
                       data.frame(n = n, group = "Treatment", team = NFL_Teams, prop = prop_treatment),
                       data.frame(n = n, group = "Control", team = NFL_Teams, prop = prop_control))
}

# Add population proportions to table
  full_results <- results %>% left_join(data.frame(team = NFL_Teams, pop_prop = p_prop), by = "team") %
  mutate(difference = pop_prop-prop)

# Show that as n increases, the distribution of traits in the sample, treatment, and control has simila

  ggplot(full_results, aes(x = n, y = prop, color = group)) +
  geom_point() +
  geom_line() +
  geom_hline(aes(yintercept = pop_prop), linetype = 2) +
  facet_wrap(~ team, nrow = 2) +
  labs(x = "Sample size (n)", y = "Proportion",
       title = "Treatment & Control proportions approach population as n increases",
       subtitle = "Dashed line = population proportion per category")
```



In the above graphs, we see that the population proportions for treatment, control, and sample groups all converge on the population proportion line as the sample size increases.

The goal of the simulation is to show that random assignment of treatment within a representative sample is what makes the control and treatment groups comparable, which in turn allows us to more accurately estimate treatment effect. I showed this by creating traits (loosely interpreted as support for different NFL teams) and assigning them each a proportion of the overall population. By then generating random samples of the overall population, and assigning the treatment to approximately half of the sample, we see that the proportions of team support for the treatment, control, and sample are aligned with the actual population proportions that I designated at the beginning of the simulation. We can also see that, as the sample size increases, the proportions within each of the groups gets even closer to the actual population proportions, thus showing that random assignment of the treatment makes our groups comparable, especially as the representative sample size increases.

DATA ANALYSIS

Setup

```
set.seed(25)
```

Read in data from git repo

```
voting <- read.csv('https://raw.githubusercontent.com/MLBurnham/pols_602/refs/heads/main/data/voting.cs
```

What is the treatment variable? Is it a discrete or continuous variable? What is the variable's data type?

The treatment variable is "message" and it is discrete. The data type for the treatment variable is characters.

```
# confirming that data type is character
typeof(voting$message)
```

```
## [1] "character"
```

Create a new treatment variable in your dataframe that is a binary version of the existing treatment variable. Your new variable should equal 1 if the observation was treated, and 0 otherwise.

```
# Use ifelse to create binary treatment variable

voting$treated <- ifelse(voting$message == "yes", 1, 0)
```

Compute the average outcome for the treatment group and the average outcome for the control group. Interpret the results by writing 1-2 sentences about what these numbers mean substantively.

```
#Calculate mean for the voting column and the treated column of the dataset
mean(voting$voted)
```

```
## [1] 0.3101759
```

```
mean(voting$treated)
```

```
## [1] 0.1664938
```

The mean substantively means that about 31% of people voted, regardless of treatment. About 17% of individuals in the sample were given the treatment, i.e. they received the message.

Use brackets to subset the dataframe and create two new dataframes, one for the treatment group and one for the control group.

```
#Create subsets by passing it the desired row and column

treatment_df <- voting[voting$treated==1, ]
control_df <- voting[voting$treated==0, ]

#Check to make sure no rows were lost

nrow(treatment_df) + nrow(control_df)
```

```
## [1] 229444
```

```
nrow(voting)
```

```
## [1] 229444
```

What is the average birth year for the treatment and control groups?

```
#Calculate average birth years for new dataframes

mean(treatment_df$birth)
```

```
## [1] 1956.147
```

```
mean(control_df$birth)
```

```
## [1] 1956.186
```

What is the estimated average causal effect for this experiment? Provide the calculated average effect and a substantive interpretation.

```r
#First find the average voter turnout for treatment and control

mean(treatment_df$voted)
```

```
## [1] 0.3779482
```

```r
mean(control_df$voted)
```

```
## [1] 0.2966383
```

```r
#Then subtract to find average causal effect

mean(treatment_df$voted) - mean(control_df$voted)
```

```
## [1] 0.08130991
```

This means that the treatment (i.e. receiving the message) created about an 8 percentage point increase in voter turnout.

Suppose we wanted to claim that the estimated causal effect is an estimated effect for the entire U.S. population. What assumption would need to hold for us to make this claim?

We would have to assume that the sample used in the experiment is representative of the entire US population and that no groups were systematically excluded. Because they only sampled in Michigan and among homeowners, there were groups of the population which were systematically excluded, meaning that the causal effect is not generizable to the entire US population.