Pset1

Gabriel

October 9, 2025

1 Simulation

```r
# Step 1: Create a population
# Set seed for reproducibility
set.seed(123)

# Define population size
N <- 100000
# Define population with some traits
prop_party_Pop <- c(0.4, 0.4, 0.2) # define party distribution in population
prop_gender_Pop <- c(0.5, 0.5) # define gender distribution in population
prop_income_Pop <- c(0.4, 0.5, 0.1) # define income distribution in population

population <- data.frame(
  # income status: Low, Medium, High
  income = sample(c("Low", "Medium", "High"), N, replace = TRUE, prob = prop_party_
      Pop),

  # Gender: Male, Female
  gender = sample(c("Male", "Female"), N, replace = TRUE, prob = prop_gender_Pop),

  # Party: Democrat, Republican, Independent
  party = sample(c("Democrat", "Republican", "Independent"), N, replace = TRUE, prob
      = prop_income_Pop)
)

head(population)

# sample sizes to demonstrate "as n increases"
n_vals <- c(100, 200, 500, 1000, 5000, 8000)

# ---- simulate once for each n ----
library(tidyverse)
party_proportions <- data.frame() # define once outside the loop

for (n in n_vals){
  # Step 1: sample n observations
  sample_data <- sample_n(population, size = n, replace = TRUE)

  # Step 2: random assignment; 1 = treatment, 0 = control
  treatment <- rbinom(n, size = 1, prob = 0.5)

  # Step 3: make dataframe
  df <- data.frame(sample_data, treat = treatment)

  # Step 4: Calculate proportions
  prop_party_treat   <- prop.table(table(df$party[df$treat == 1]))
  prop_party_control <- prop.table(table(df$party[df$treat == 0]))
  prop_gender_treat  <- prop.table(table(df$gender[df$treat == 1]))
  prop_gender_control<- prop.table(table(df$gender[df$treat == 0]))
  prop_income_treat  <- prop.table(table(df$income[df$treat == 1]))
  prop_income_control<- prop.table(table(df$income[df$treat == 0]))

  # proportions for all
```

```r
  prop_party  <- prop.table(table(df$party))
  prop_gender <- prop.table(table(df$gender))
  prop_income <- prop.table(table(df$income))

  party_proportions <- bind_rows(
    party_proportions,
    data.frame(n = n, group = "All",     party = names(prop_party), prop = as.numeric(prop_party)),
    data.frame(n = n, group = "Treat",   party = names(prop_party), prop = as.numeric(prop_party_treat)),
    data.frame(n = n, group = "Control", party = names(prop_party), prop = as.numeric(prop_party_control))
  )
}

print(party_proportions)

# step 5: join sample proportions with population proportions
party_prop_full <- party_proportions %>%
  left_join(
    data.frame(
      party = names(prop_party),
      prop_party_Pop = as.numeric(prop_party_Pop)
    ),
    by = "party"
  ) %>%
  mutate(imbalance = prop - prop_party_Pop)

head(party_prop_full)

# simple wide table
party_wide <- party_prop_full %>%
  select(-imbalance) %>%
  pivot_wider(names_from = group, values_from = prop) %>%
  rename(prop_party_Pop = prop_party_Pop) %>%
  arrange(n, party)

head(party_wide)

# plot 1: groups approach population line as n grows
party_proportions_baseline <- tibble(
  Party=c("Democrat", "Republican", "Independent"),
  pop_prop=c(0.4, 0.4, 0.2)
)

ggplot(party_prop_full, aes(x = n, y = prop, color = group)) +
  geom_point() + geom_line() +
  geom_hline(data = party_proportions_baseline, aes(yintercept = pop_prop), linetype = 2) +
  facet_wrap(~ party, 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") +
  theme_minimal()

# plot 2: imbalance across parties shrinks with n
imbalance <- party_prop_full %>%
  filter(group %in% c("Treat","Control")) %>%
  group_by(n, party, group) %>%
  summarise(prop = mean(prop), .groups = "drop") %>%
  pivot_wider(names_from = group, values_from = prop) %>%
  mutate(abs_diff = abs(Treat - Control)) %>%
  group_by(n) %>%
  summarise(
    max_abs_diff = max(abs_diff),
    l1_sum_diff  = sum(abs_diff),
    .groups = "drop"
```

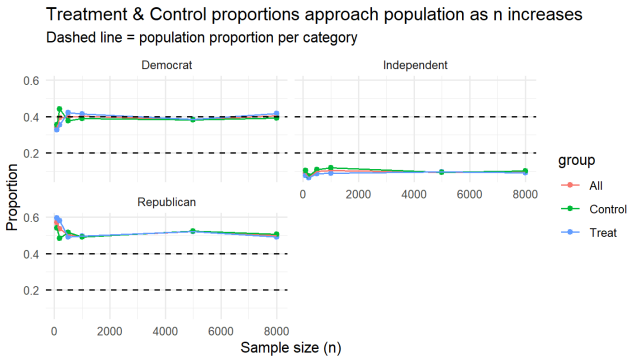

Figure 1: Treatment & Control proportions approach population as n increases

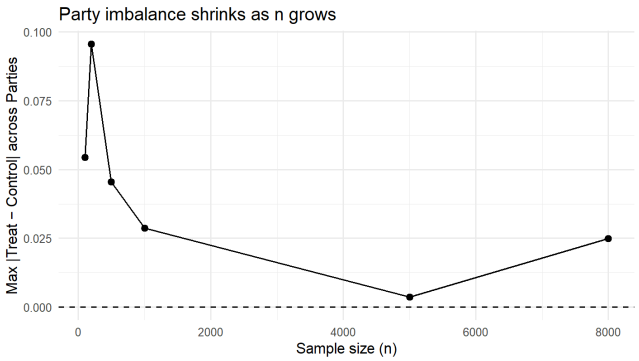

Figure 2: Party imbalance shrinks as n grows

```
  )

ggplot(imbalance, aes(x = n, y = max_abs_diff)) +
  geom_hline(yintercept = 0, linetype = 2) +
  geom_point(size = 2) + geom_line() +
  labs(x = "Sample size (n)", y = "Max |Treat - Control| across Parties",
       title = "Party imbalance shrinks as n grows") +
  theme_minimal()

table(party_prop_full$n)
```

2 Data Analysis

Question 1

```
# "message" is the treatment

voting <- read.csv("C:/WindowsD/TAM/25fall/Quant/Assignment/0930/voting.csv")
summary(voting$message)
# "message" is a continuous variable; its data type is "character"
```

Question 2

```
str(voting$message)
voting$treatment <- ifelse(voting$message == "no", 0 , 1 )
```

Question 3

```
library(dplyr)
mean(voting)
```

```r
voting %>%
  filter(treatment == 1) %>%
  summarise(mean_voted1 = mean(voted, na.rm = TRUE)) %>%
  print()
  #  mean_voted1: 0.3779482.
```

Among those who received the treatment, approximately 37.8% voted.

```r
voting %>%
  filter(treatment == 0) %>%
  summarise(mean_voted0 = mean(voted, na.rm = TRUE)) %>%
  print()
  #  mean_voted0:0.2966383
```

Among those who received no treatment, approximately 29.7% voted.

Question 4

```r
voting_treated <- voting %>%
  filter(treatment == 1)
voting_nottreated <- voting %>%
  filter(treatment == 0)
```

Question 5

```r
voting %>%
  filter(treatment == 0) %>%
  summarise(mean_birth0 = mean(birth, na.rm = TRUE)) %>%
  print()
```

The average birth year for the untreated group is 1956.186.

```r
voting %>%
  filter(treatment == 1) %>%
  summarise(mean_birth1 = mean(birth, na.rm = TRUE)) %>%
  print()
```

The average birth year for the treated group is 1956.147.

Question 6

```r
mean_voted1 <- 0.3779482
mean_voted0 <- 0.2966383
average_causal_effect <- mean_voted1 - mean_voted0
print(average_causal_effect)
#average_causal_effect: 0.0813099
```

The average causal effect is 0.0813099. The average causal effect of `message` on `voted` is the average change (0.0813099) in `voted` caused by a one-unit increase in `message` for a group of individuals.

Question 7

The more messages that pressured people to vote by promising to tell their neighbors if they voted in the upcoming election, the more likely people were to vote.