Problem Set 1

Fernanda Valdez

2025-10-22

Simulation

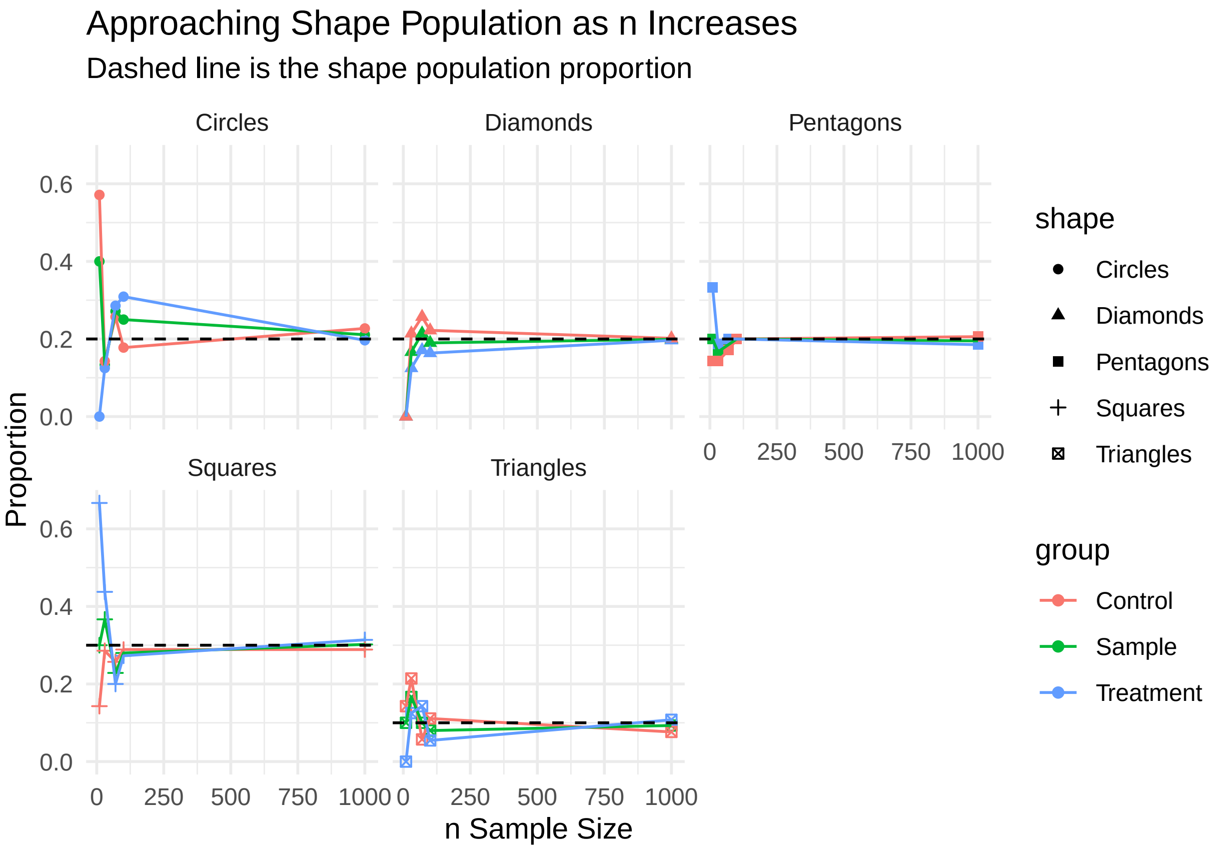
```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

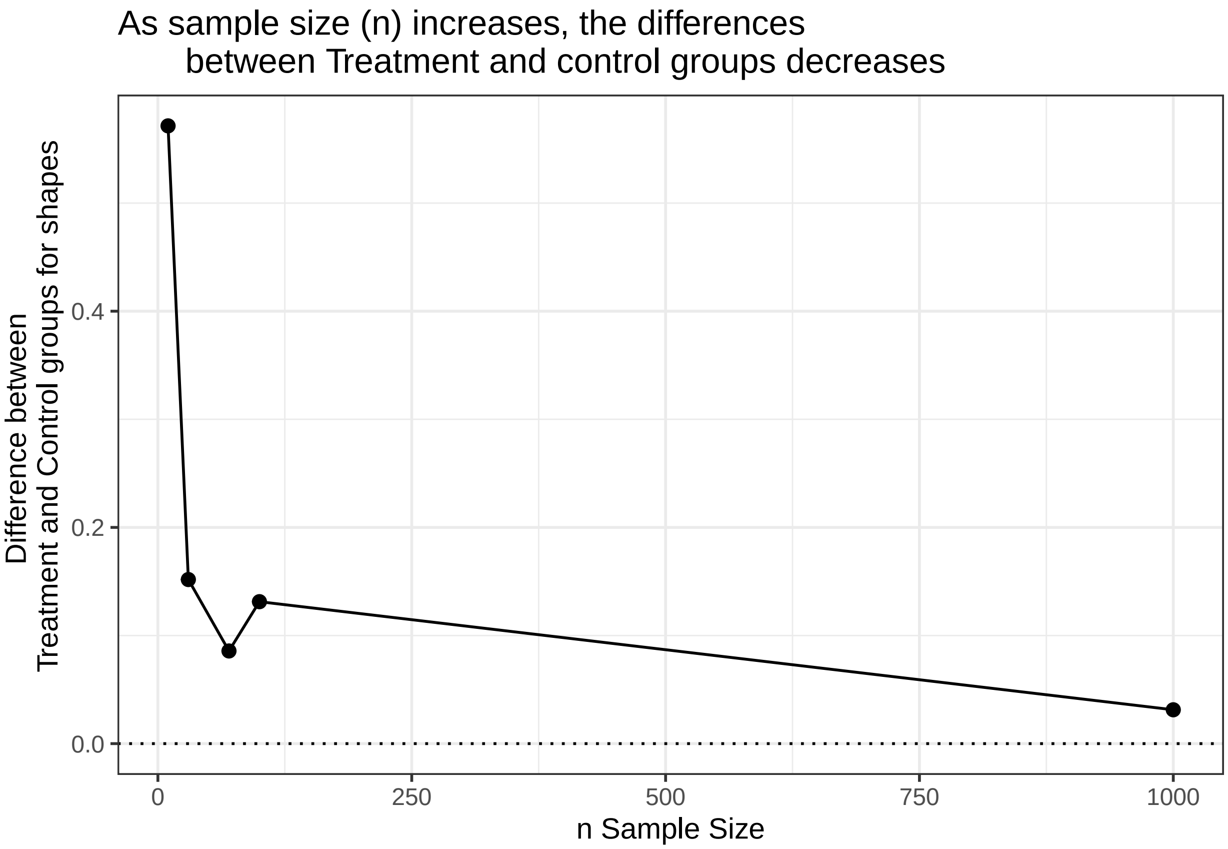
```
library(tidyr)
library(ggplot2)
require(ggplot2)
set.seed(0976)
```

```
# Population traits = shapes #
shapes <- c("Triangles", "Circles", "Squares", "Diamonds", "Pentagons")
# Proportion of shapes #
prop_shapes <- c(0.10, 0.20, 0.30, 0.20, 0.20)
# Assign proportions to each shape #
names(prop_shapes) <- shapes
# Sample sizes #
n <- c(10, 30, 70, 100, 1000)
# Draw the samples #
shape <- sample(shapes, size = 1, replace = TRUE, prob = prop_shapes)
# Randomly assign the treatment #
treatment <- rbinom(1, 1, 0.5)
# Container #
container <- data.frame(n = integer(), shape = character (), group = character(),
                        proportion = numeric ())
# Random sampling loop #
for(i in n){
  shape <- sample(shapes, size = i, replace = TRUE, prob = prop_shapes)
  treatment <- rbinom(i, 1, 0.5)

# Proportion of shapes for sample #
Proportion_sample <- as.numeric(table(factor(shape, levels = shapes)))/i
# Number of shapes with treatment #
n_treatment <- sum(treatment == 1)
```

```r
# Proportion of shapes with treatment #
proportion_treatment <- as.numeric(table(factor(shape[treatment==1], levels =
                                                  shapes)))/n_treatment
# Number of shapes without treatment #
n_control <- i - n_treatment
# Proportion of shapes without treatment #
proportion_control <- as.numeric(table(factor(shape[treatment==0], levels=
                                                shapes)))/n_control

# Combining data frame with loops #
container <- rbind(container,
                 data.frame(n=i, shape=shapes, group="Sample",
                            proportion= Proportion_sample),
                 data.frame(n=i, shape=shapes, group="Treatment",
                            proportion= proportion_treatment),
                 data.frame(n=i, shape=shapes, group="Control",
                            proportion= proportion_control))
}

# FIGURES #

# Population proportions #
container_full <- container %>% left_join(
  data.frame(shape = shapes, population_proportion = prop_shapes),
  by = "shape") %>%
  mutate(imbalance = population_proportion-proportion)

# Comparison #
container_wide <- container_full %>%
  select(-imbalance) %>%
  pivot_wider(names_from = shape, values_from = proportion) %>%
  rename(population = population_proportion) %>%
  arrange(n, group)


# As n grows, proportions get closer to the population #
population_reference <- tibble(shape = shapes, population_proportion = prop_shapes)

ggplot(container, aes(x = n, y = proportion, color = group, shape = shape)) +
  geom_point() + geom_line() +
  geom_hline(data = population_reference,
             aes(yintercept = population_proportion), linetype = 2) +
  facet_wrap(~ shape, nrow = 2) +
  labs(x = "n Sample Size", y = "Proportion",
       title = "Approaching Shape Population as n Increases",
       subtitle = "Dashed line is the shape population proportion") +
  theme_minimal()
```



```r
# Difference between treatment and control shrinks with n #
difference <- container %>%
  filter(group %in% c("Treatment","Control")) %>%
  select(n, shape, group, proportion) %>%
  pivot_wider(names_from = group, values_from = proportion) %>%
  mutate(abs_diff = abs(Treatment - Control)) %>%
  group_by(n) %>%
  summarise(
    max_abs_diff = max(abs_diff),
    l1_sum_diff  = sum(abs_diff),
    .groups = "drop"
  )
# Graph
ggplot(difference, aes(x = n, y = max_abs_diff)) +
  geom_hline(yintercept = 0, linetype = 3) +
  geom_point(size = 2) + geom_line() +
  labs(x = "n Sample Size", y = "Difference between
       Treatment and Control groups for shapes",
       title = "As sample size (n) increases, the differences
       between Treatment and control groups decreases") +
  theme_bw()
```



The goal of this simulation is to randomly assign the treatment in order to illustrate that treatment and control groups are comparable. To begin, I first assigned my population traits: shapes. Then, I chose my proportions (10%, 20%, 30%, 20%, and 20%) and assigned them to their respective shapes. I chose different increasing sample sizes (10, 30, 70, 100, 1000) and instructed 'R' how to draw from my samples. I followed this by randomly assigning my treatment and creating my data frame. From there, I created my random sampling loop and included my proportion of shapes for my sample, treatment, and control. Then, I combined my data frame with the loops.

After creating my simulation, I prepared my container so that I may compare my groups. In the first visualization, I show that for each shape, each group (control, sample, treatment) approaches their respective population proportion. For example, the population proportion for "Circles" is 20%. The graph illustrates that as the sample increases to 1000, all three groups for "Circles"–control, sample, and treatment–reach the population proportion (dashed line). Thus, this illustrates that as sample size increases for each shape, they begin to resemble their population proportion. Consequently, this allows me to claim that my sample is representative of the population. My second visualization illustrates that the differences between the two groups decrease as the sample size increases to 1000. This demonstrates that my treatment and control groups are comparable.

Data Analysis

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

The treatment variable is social pressure which is whether they received a message. This variable is a discrete variable and it's data type is categorical.

```
Vote <- read.csv("~/Desktop/voting.csv")
```

2. Create a new treatment variable in your data frame that is a binary version of the existing treatment

variable. Your new variable should equal 1 if the observation was treated, and 0 otherwise.

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

3. 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.

Of the whole population, 31% voted while from those who received the the message/treatment, 17% voted.

```
mean(Vote$voted)
```

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

```
mean(Vote$treated)
```

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

4. Use brackets to subset the data frame and create two new data frames, one for the treatment group and one for the control group.

```
treatment <- Vote[Vote$treated==1, ]
control <- Vote[Vote$treated==0, ]

nrow(treatment) + nrow(control)
```

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

```
nrow(Vote)
```

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

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

```
c("Average birth year for treatment group" = mean(treatment$birth),
  "Average birth year for control group" = mean(control$birth))
```

```
## Average birth year for treatment group   Average birth year for control group
##                               1956.147                               1956.186
```

```
mean(treatment$voted)
```

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

```
mean(control$voted)
```

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

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

After receiving the treatment/message, there is an 8.1 percentage point increase in voter turnout.

```
mean(treatment$voted) - mean(control$voted)
```

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

7. 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 need a more representative sample to claim external validity. That is, they would need to include more housing types in their sample to make sure all resident types are represented in their sample.