PROBLEM_SET1

Kafayat Liadi

2025-10-21

Simulating effect of increasing sample size on trait distribution

---

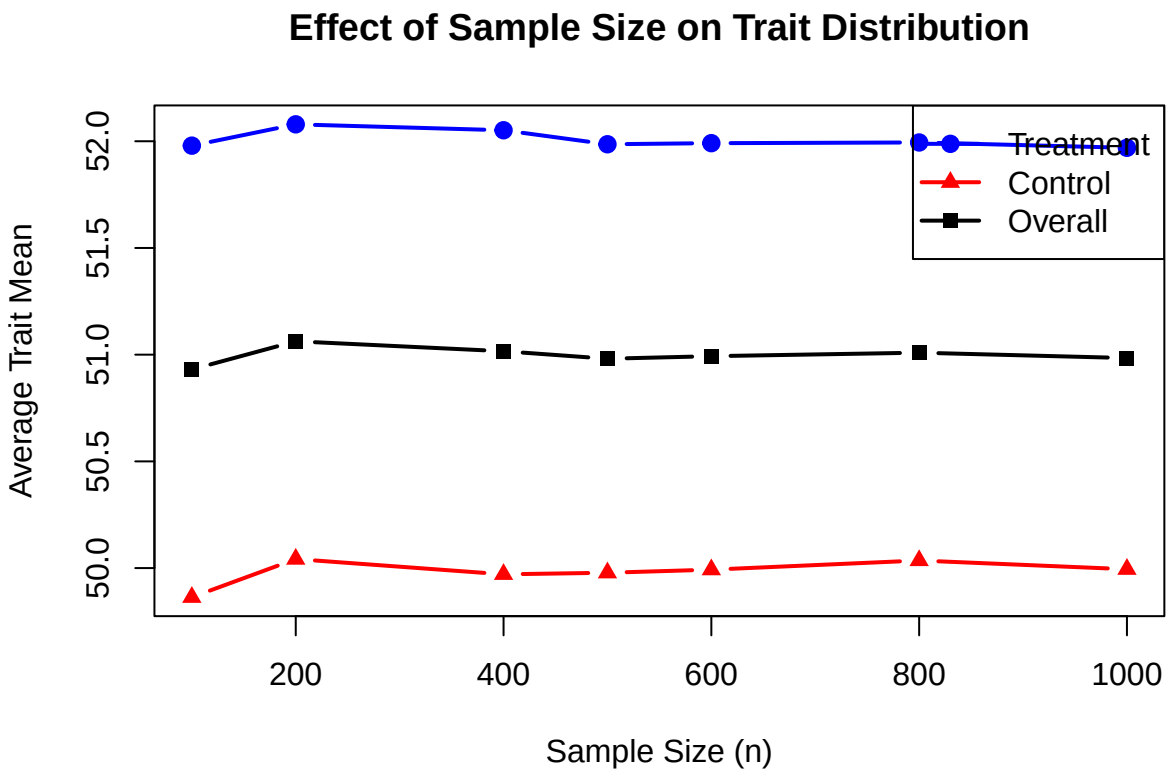
```r
# Define seven different sample sizes
sample_sizes <- c(100, 200, 400, 500, 600, 800, 1000)
# Set number of repetitions per sample size
n_iter <- 100

# Create a data frame to store all results
results <- data.frame(
  sample_size = numeric(),
  iteration = numeric(),
  mean_all = numeric(),
  mean_treatment = numeric(),
  mean_control = numeric()
)

# Set seed for reproducibility
set.seed(123)

# Loop over each sample size
for (n in sample_sizes) {

  # Repeat the process multiple times for each n
  for (i in 1:n_iter) {

    # Create dataset with unique IDs
    data <- data.frame(id = 1:n)

    # Randomly assign each observation to Treatment or Control
    data$group <- sample(c("Treatment", "Control"), size = n, replace = TRUE)

    # Simulate a trait value (e.g., yield or height)
    # The "Treatment" group has slightly higher mean trait values
    data$trait <- ifelse(data$group == "Treatment",
                         rnorm(n, mean = 52, sd = 5),
                         rnorm(n, mean = 50, sd = 5))
```

```
    # Calculate mean traits
    mean_all <- mean(data$trait)
    mean_treat <- mean(data$trait[data$group == "Treatment"])
    mean_ctrl <- mean(data$trait[data$group == "Control"])

    # Save results for this iteration
    results <- rbind(results, data.frame(
      sample_size = n,
      iteration = i,
      mean_all = mean_all,
      mean_treatment = mean_treat,
      mean_control = mean_ctrl
    ))
  }
}

# Summarize results by sample size

summary_stats <- aggregate(
  cbind(mean_all, mean_treatment, mean_control) ~ sample_size,
  data = results,
  FUN = mean
)
summary_stats
```

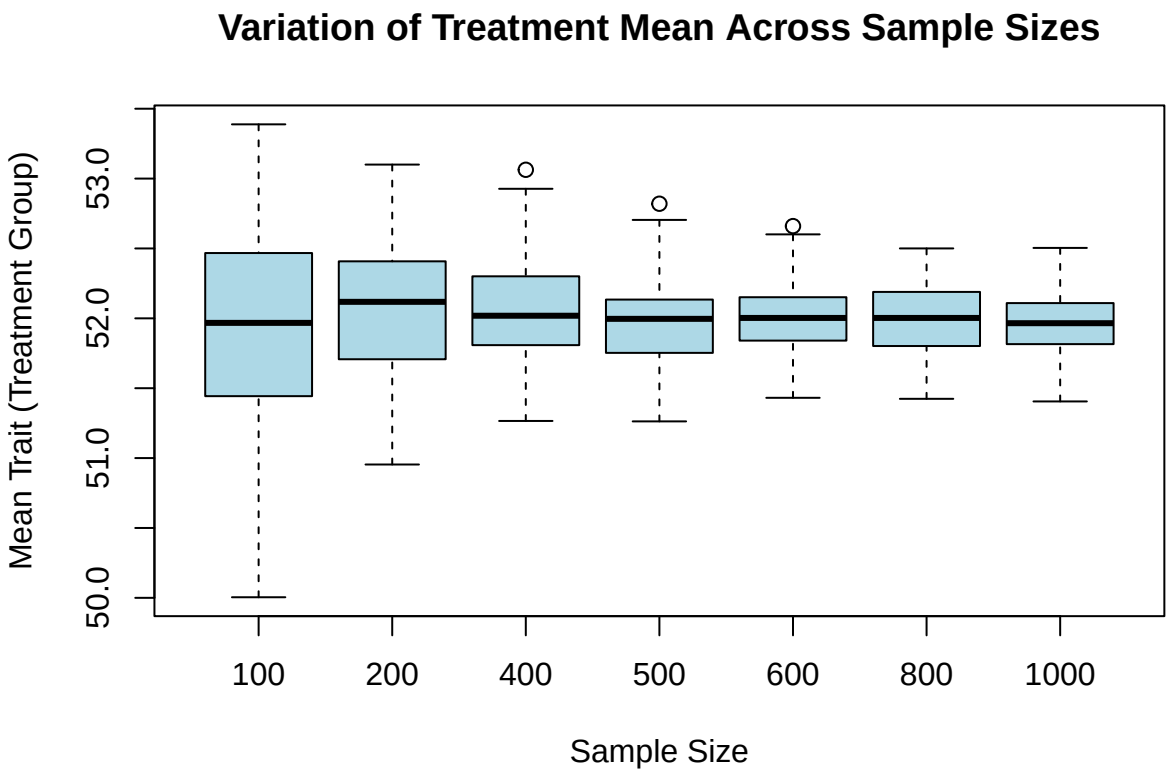
```
##   sample_size mean_all mean_treatment mean_control
## 1         100 50.93081       51.97943     49.86336
## 2         200 51.06266       52.07924     50.04218
## 3         400 51.01618       52.05159     49.97084
## 4         500 50.98090       51.98598     49.97807
## 5         600 50.99294       51.99107     49.99256
## 6         800 51.00937       51.99435     50.03551
## 7        1000 50.98429       51.96933     49.99469
```

Plot Variability across iterations for each sample size

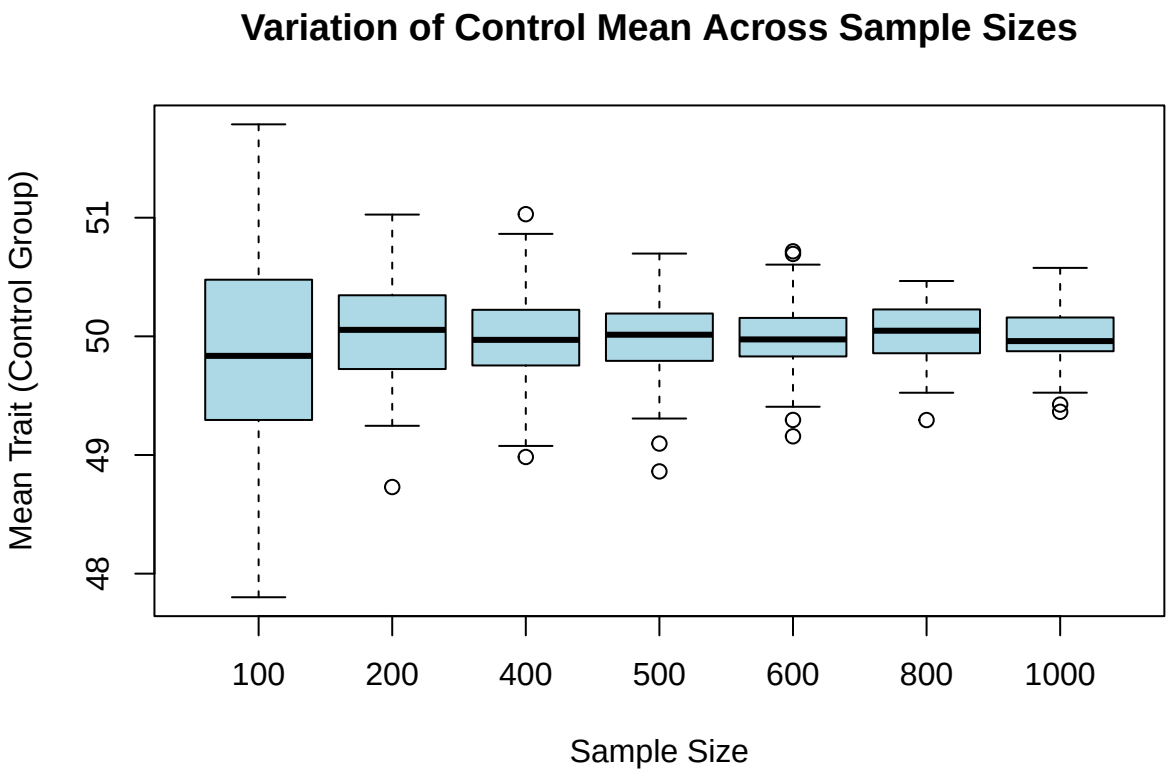
```
# Plot mean treatment and control vs. sample size
plot(summary_stats$sample_size, summary_stats$mean_treatment,
     type = "b", pch = 19, col = "blue", lwd = 2,
     ylim = range(summary_stats[, 2:4]),
     xlab = "Sample Size (n)",
     ylab = "Average Trait Mean",
     main = "Effect of Sample Size on Trait Distribution")
lines(summary_stats$sample_size, summary_stats$mean_control,
      type = "b", pch = 17, col = "red", lwd = 2)
lines(summary_stats$sample_size, summary_stats$mean_all,
      type = "b", pch = 15, col = "black", lwd = 2)
legend("topright",
       legend = c("Treatment", "Control", "Overall"),
       col = c("blue", "red", "black"),
       pch = c(19, 17, 15),
       lwd = 2)
```




```
### Using Boxplot: Plot Variability across iterations for each sample size

boxplot(mean_treatment ~ sample_size, data = results,
        col = "lightblue",
        main = "Variation of Treatment Mean Across Sample Sizes",
        xlab = "Sample Size",
        ylab = "Mean Trait (Treatment Group)")
```




```
boxplot(mean_control ~ sample_size, data = results,
        col = "lightblue",
        main = "Variation of Control Mean Across Sample Sizes",
        xlab = "Sample Size",
        ylab = "Mean Trait (Control Group)")
```



Data Analysis of Voting Dataset

```r
# Load the data

voting <- read.csv("voting.csv")
```

1. It's a discrete variable of nominal data type.

2. Create a binary treatment variable "message_binary"

```r
voting$message_binary <- ifelse(voting$message == "yes" | voting$message == 1, 1, 0)

# Check the first few rows
head(voting)
```

```
##   birth message voted message_binary
## 1  1981      no     0              0
```

```
## 2  1959      no     1              0
## 3  1956      no     1              0
## 4  1939     yes     1              1
## 5  1968      no     0              0
## 6  1967      no     0              0
```

3. Compute the average outcome for treatment and control groups

```
mean_treated <- mean(voting$voted[voting$message_binary == 1], na.rm = TRUE)
mean_control <- mean(voting$voted[voting$message_binary == 0], na.rm = TRUE)

cat("Average outcome for treatment group:", mean_treated, "\n")
```

```
## Average outcome for treatment group: 0.3779482
```

```
cat("Average outcome for control group:", mean_control, "\n")
```

```
## Average outcome for control group: 0.2966383
```

If mean_treated > mean_control, then being treated had a positive effect on the voting outcome.Otherwise, the treatment (social media pressure message received) may have had little or no effect.

4. Subset the data into two new dataframes

```
treated_data <- voting[voting$message_binary == 1, ]
control_data <- voting[voting$message_binary == 0, ]

head(treated_data)
```

```
##    birth message voted message_binary
## 4   1939     yes     1              1
## 19  1946     yes     0              1
## 20  1932     yes     0              1
## 26  1956     yes     1              1
## 27  1965     yes     1              1
## 28  1985     yes     1              1
```

```
head(control_data)
```

```
##   birth message voted message_binary
## 1  1981      no     0              0
## 2  1959      no     1              0
## 3  1956      no     1              0
## 5  1968      no     0              0
## 6  1967      no     0              0
## 7  1941      no     1              0
```

```r
# Check number of observations in each
nrow(treated_data)
```

```
## [1] 38201
```

```r
nrow(control_data)
```

```
## [1] 191243
```

5. Compute the average birth year for each group

```r
mean_birth_treated <- mean(treated_data$birth, na.rm = TRUE)
mean_birth_control <- mean(control_data$birth, na.rm = TRUE)

cat("Average birth year (treated):", mean_birth_treated, "\n")
```

```
## Average birth year (treated): 1956.147
```

```r
cat("Average birth year (control):", mean_birth_control, "\n")
```

```
## Average birth year (control): 1956.186
```

6. Estimate the Average Causal Effect

```r
CEffect <- mean_treated - mean_control

cat("Estimated Average Causal Effect (ACE):", CEffect, "\n")
```

```
## Estimated Average Causal Effect (ACE): 0.08130991
```

Interpretation of Casual Effect:

The Average Causal Effect represents the average difference in voting outcomes between those who were treated and those who were not. For example, if Average Causal Effect = 0.08, the treatment increases the outcome by 8% points.

7. Key assumption for generalizing to the population

To claim this estimated causal effect applies to the entire U.S. population, we must assume that the sample is representative of the population meaning that, treatment (social media pressure message received) assignment is random and not correlated with unobserved factors that affect the voting outcome.

‘‘‘