Statistical Simulations Examples

STAT 404 Final Project

2024-11-07

Contents

```r
# Load required packages
library(ggplot2)
library(gridExtra)

# Source our functions
source("R/core_functions.R")
source("R/visualization_functions.R")
```

1. Theoretical Sampling Distribution

We'll demonstrate how the sampling distribution of the difference in proportions approaches normality as sample size increases.

```r
# Compare different sample sizes
plot_sampling_dist(
  n_values = c(10, 30, 50, 100),  # varying sample sizes
  p1 = 0.7,                        # probability for group 1
  p2 = 0.5,                        # probability for group 2
  reps = 1000                      # number of simulations
)
```
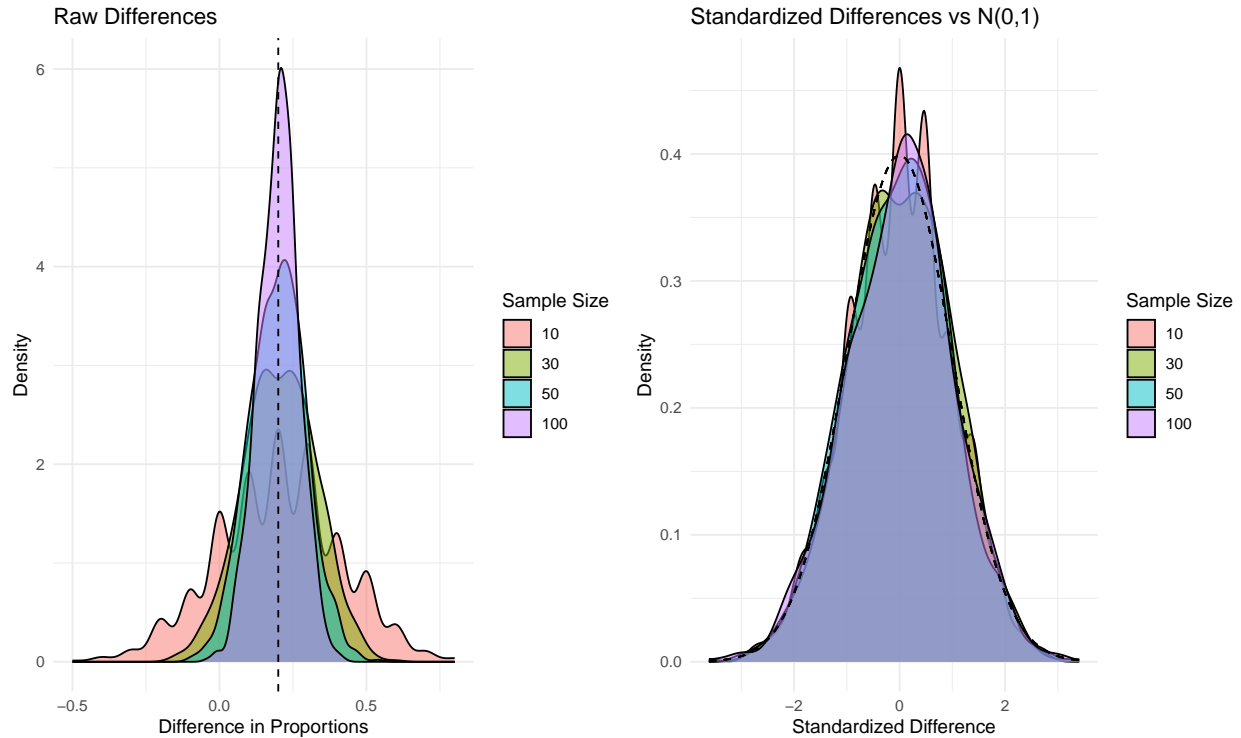
Observations: - Left plot shows the raw differences in proportions - Right plot shows standardized differences compared to N(0,1) - As sample size increases, the standardized distribution more closely matches the standard normal - The true difference (0.2) is shown by the dashed line in the left plot

2. Confidence Level Coverage

Let's examine how the empirical coverage of confidence intervals approaches the nominal level as we increase the number of simulations.

```
# Look at coverage probability for 95% confidence intervals
plot_confidence_coverage(
  p1 = 0.7,          # probability for group 1
  p2 = 0.5,          # probability for group 2
  n1 = 50,           # sample size group 1
  n2 = 50,           # sample size group 2
  alpha = 0.05,      # 95% confidence level
  max_reps = 500     # maximum number of repetitions
)
```

Confidence Interval Coverage
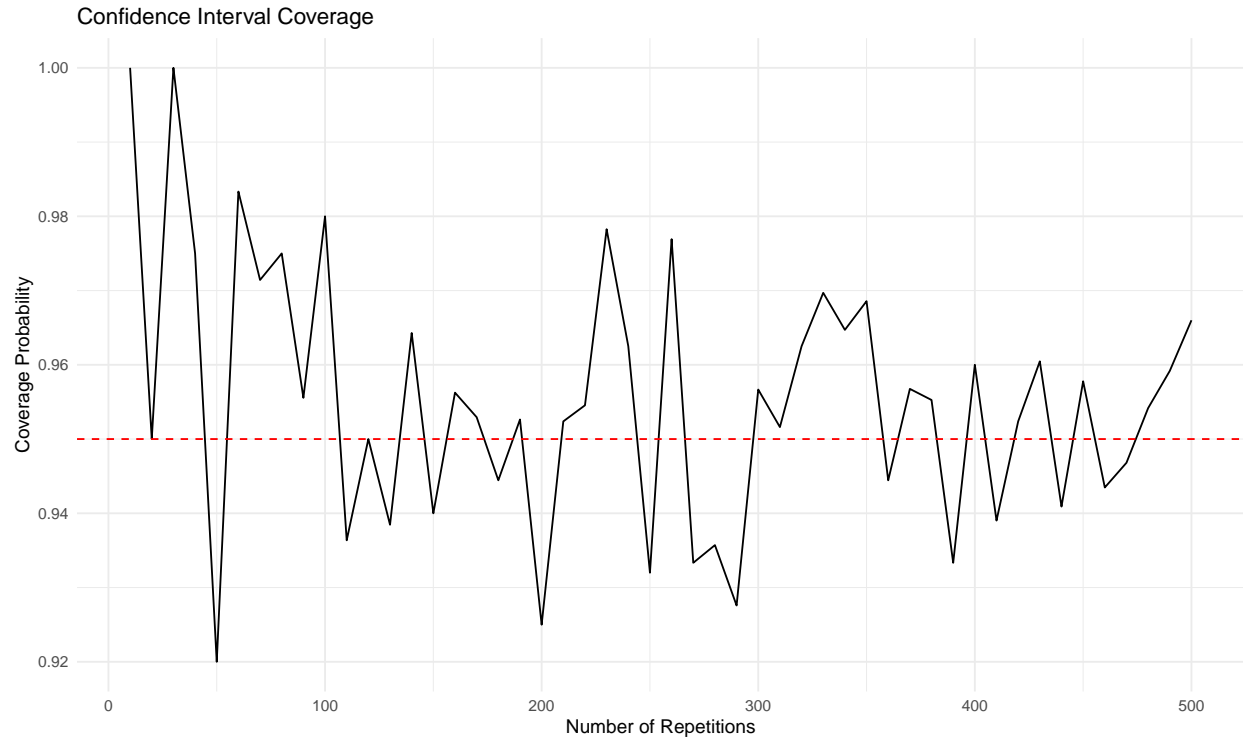
Observations: - The red dashed line shows the nominal 95% coverage level - Coverage probability stabilizes as the number of repetitions increases - Some random variation is expected but should converge to 0.95

3. Permutation Test

We'll perform a permutation test to assess whether the difference in proportions is statistically significant.

```r
# Generate data with a known difference
set.seed(123)  # for reproducibility
data <- sim_binary_data(0.7, 0.5, 50, 50)

# Perform and visualize permutation test
plot_permutation_test(data, reps = 1000)
```

Permutation Test Distribution
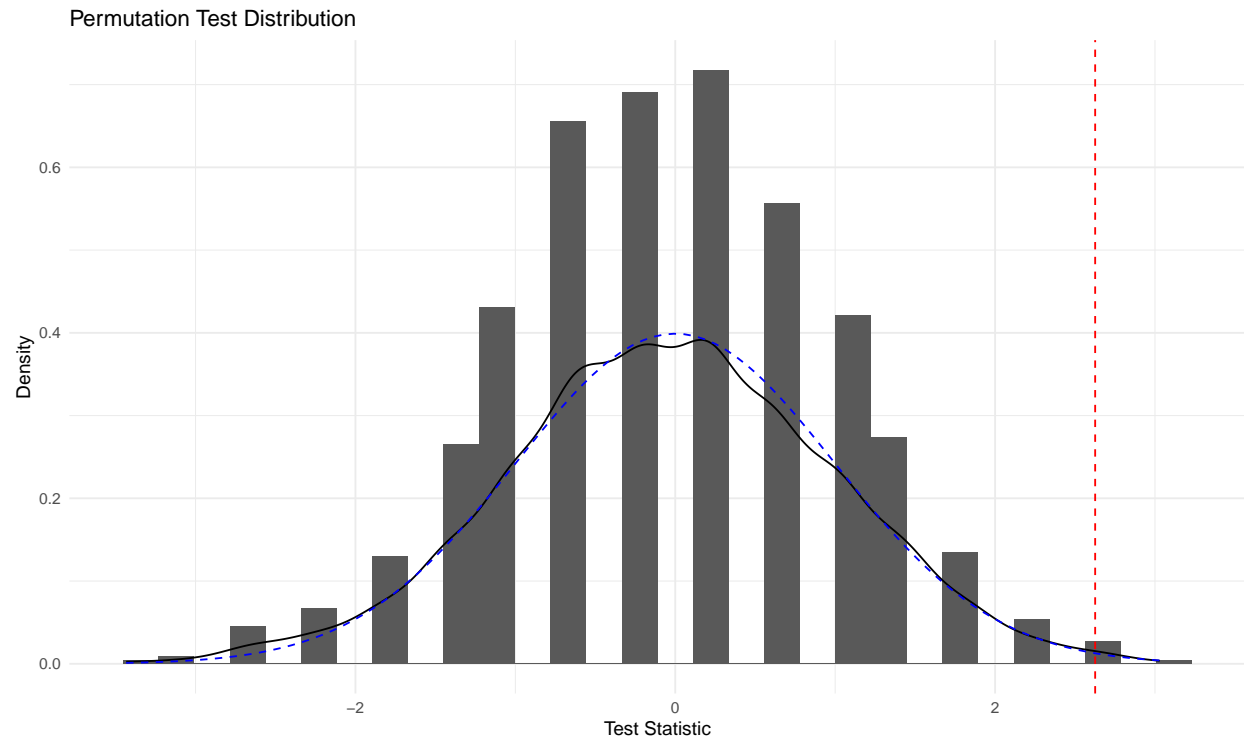
Observations: - The red dashed line shows the observed test statistic - The blue dashed line shows the theoretical normal distribution - The histogram shows the permutation distribution under the null hypothesis - If the red line is in the tails of the distribution, we have evidence against H0

4. Bootstrap Analysis

Compare bootstrap and theoretical methods for estimating uncertainty.

```
# Compare bootstrap and theoretical approaches
plot_bootstrap_comparison(data, reps = 1000, conf_level = 0.95)
```
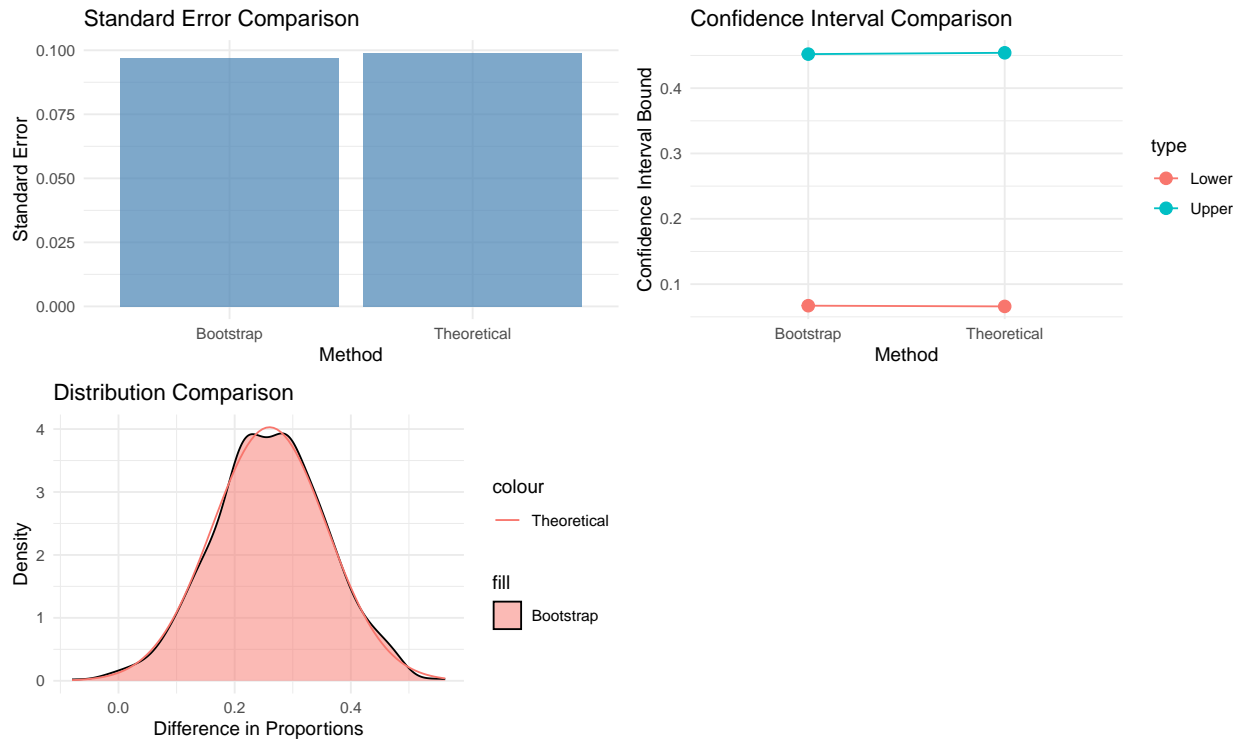
Observations: - Top left: Comparison of standard errors between methods - Top right: Comparison of confidence interval bounds - Bottom: Distribution comparison between bootstrap and theoretical

5. Complete Analysis Example

Let's put everything together in a complete analysis.

```r
# Set parameters
p1 <- 0.7  # probability for group 1
p2 <- 0.5  # probability for group 2
n <- 50    # sample size per group
reps <- 1000  # number of repetitions

# 1. Generate data
set.seed(456)  # for reproducibility
data <- sim_binary_data(p1, p2, n, n)

# 2. Calculate observed difference and SE
results <- calc_prop_diff(data)
cat("Observed difference:", round(results$diff, 3), "\n")
```

```
## Observed difference: 0.08
```

```r
cat("Standard error:", round(results$se, 3), "\n")
```

```
## Standard error: 0.099
```

```r
# 3. Perform permutation test
perm_results <- permutation_test(data, reps)
p_value <- mean(abs(perm_results$null_dist) >= abs(perm_results$obs_stat))
cat("Permutation test p-value:", round(p_value, 3), "\n")
```
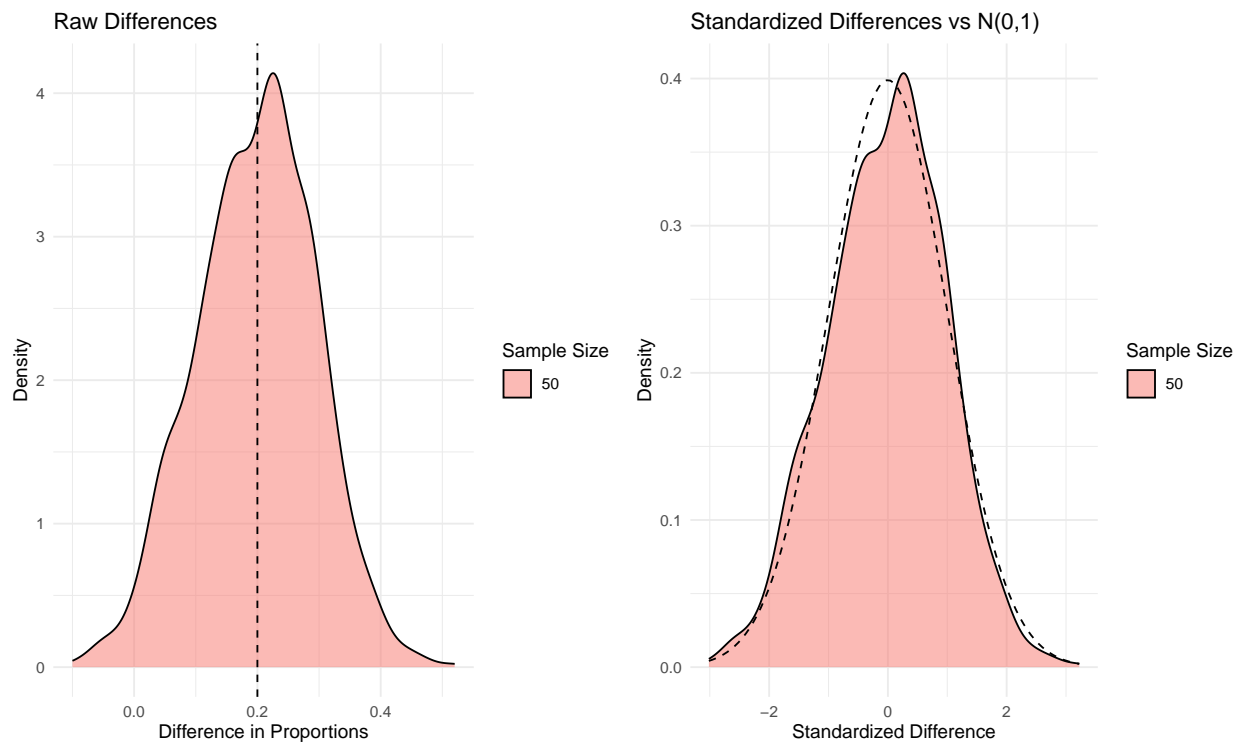
```
## Permutation test p-value: 0.559
```

```r
# 4. Calculate bootstrap confidence interval
boot_samples <- bootstrap_samples(data, reps)
boot_ci <- quantile(boot_samples, c(0.025, 0.975))
cat("Bootstrap 95% CI: (", round(boot_ci[1], 3), ",", round(boot_ci[2], 3), ")\n")
```
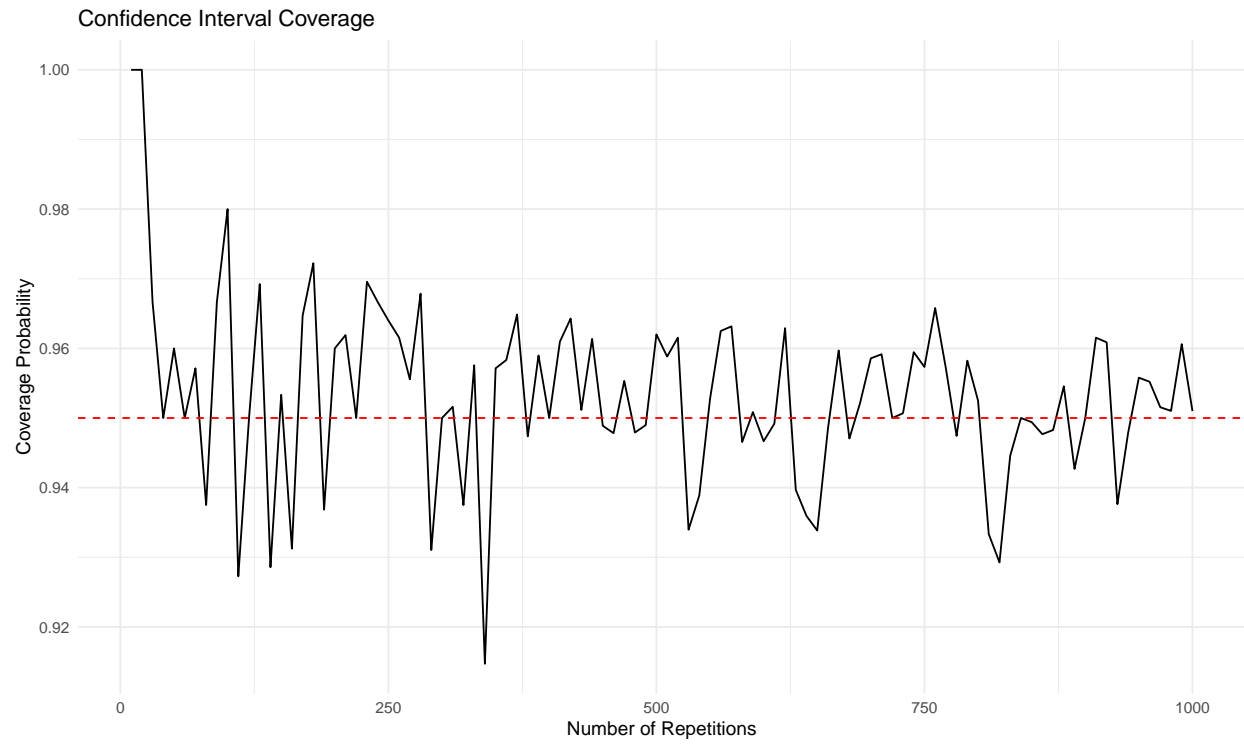
```
## Bootstrap 95% CI: ( -0.117 , 0.265 )
```

```r
# Visualize all aspects
par(mfrow = c(2,2))

# Sampling distribution
plot_sampling_dist(c(n), p1, p2, reps)
```
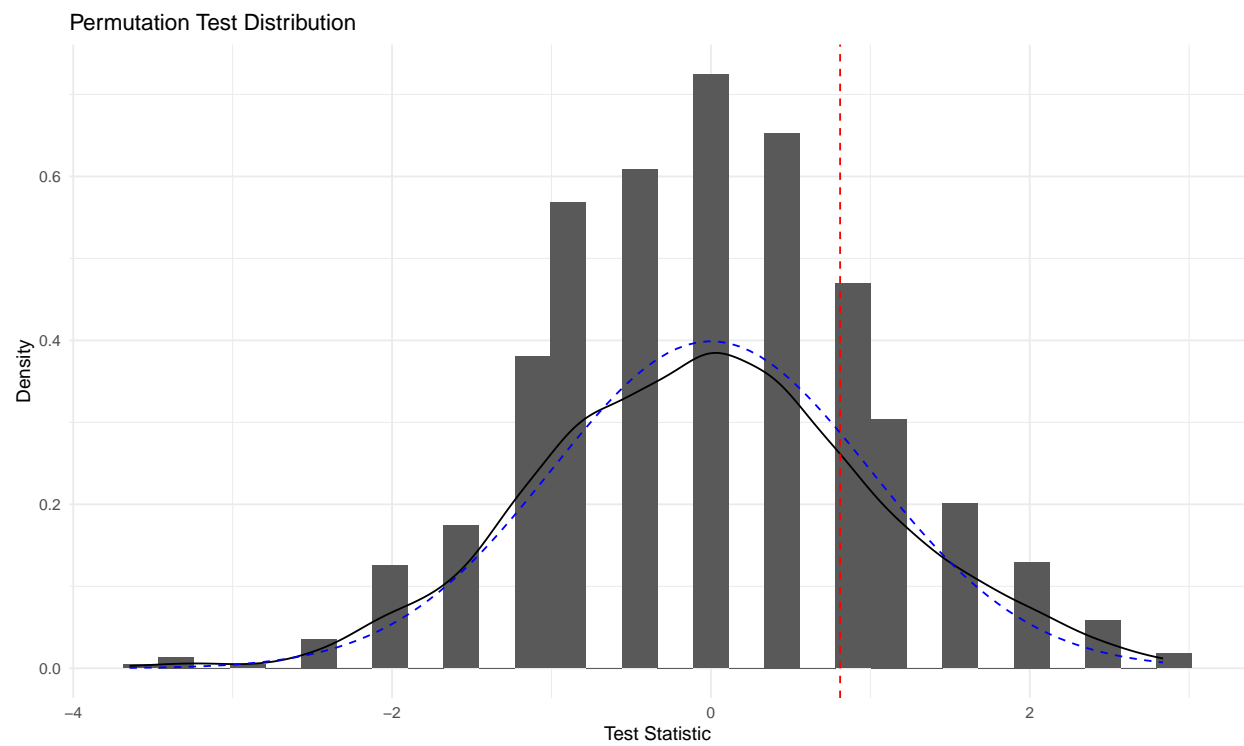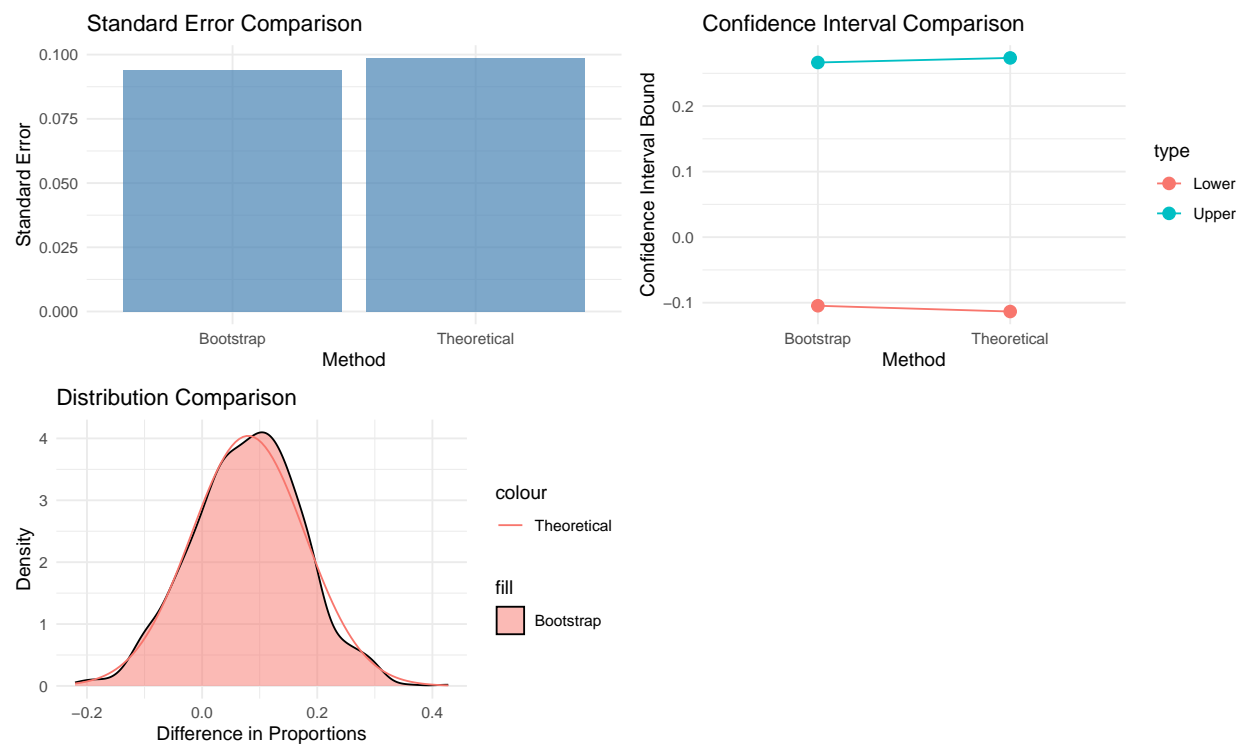


```r
# Confidence coverage
plot_confidence_coverage(p1, p2, n, n, max_reps = reps)
```

Confidence Interval Coverage


```
# Permutation test
plot_permutation_test(data, reps)
```
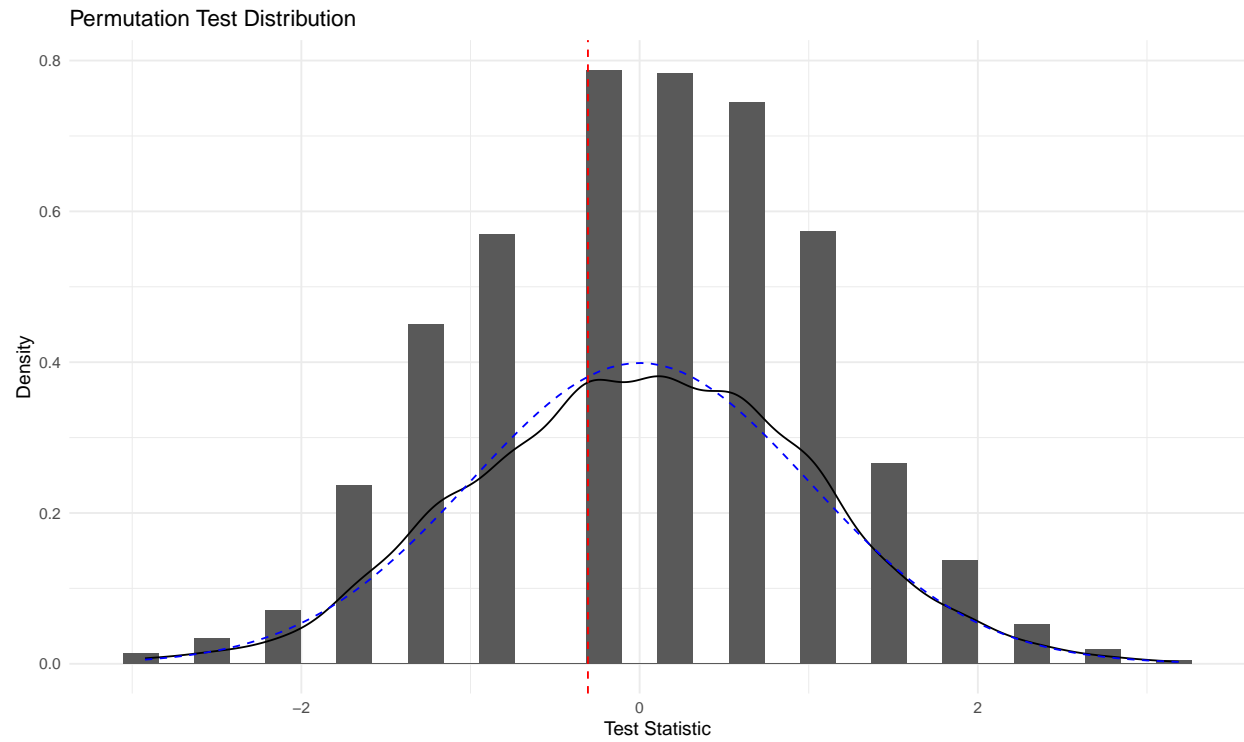
Permutation Test Distribution

```r
# Bootstrap comparison
plot_bootstrap_comparison(data, reps)
```


6. Additional Examples

Testing Different Sample Size Ratios

```r
# Compare cases with unequal sample sizes
data_unequal <- sim_binary_data(0.7, 0.5, 30, 70)
plot_permutation_test(data_unequal, reps = 1000)
```

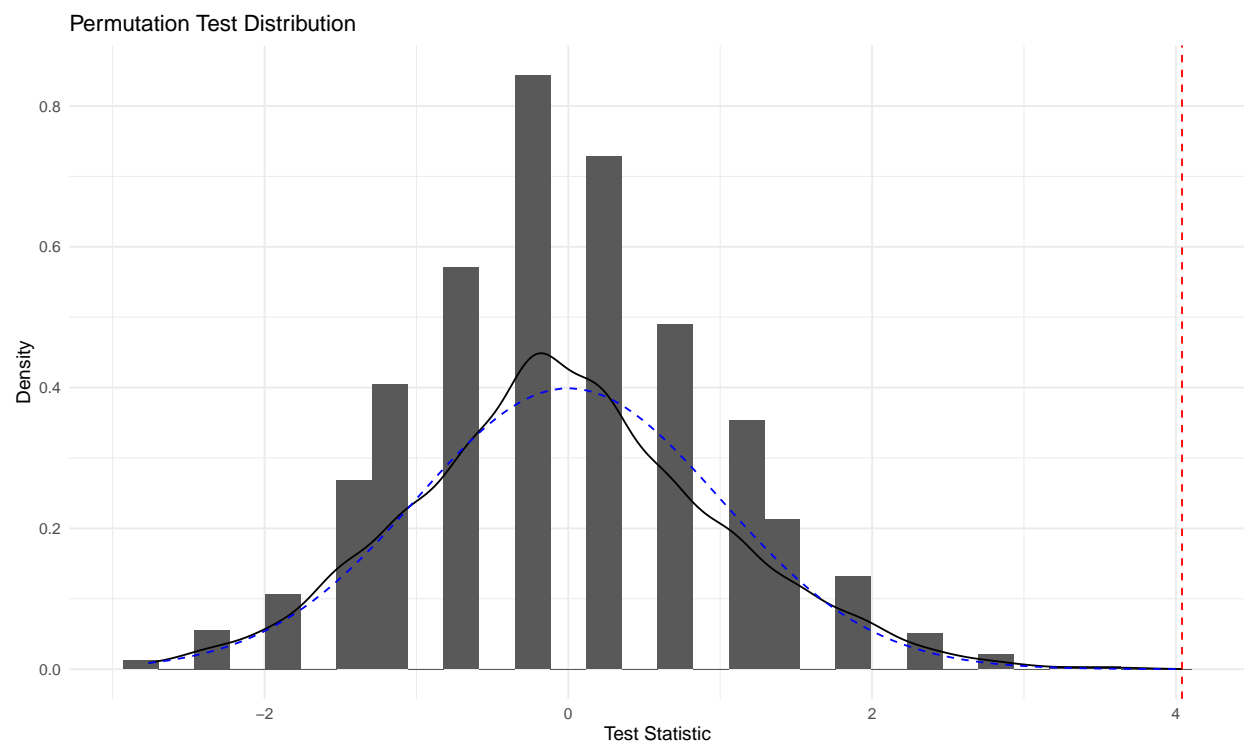Permutation Test Distribution

Testing Different Effect Sizes

```r
# Small effect
data_small <- sim_binary_data(0.52, 0.50, 50, 50)
plot_permutation_test(data_small, reps = 1000)
```

Permutation Test Distribution
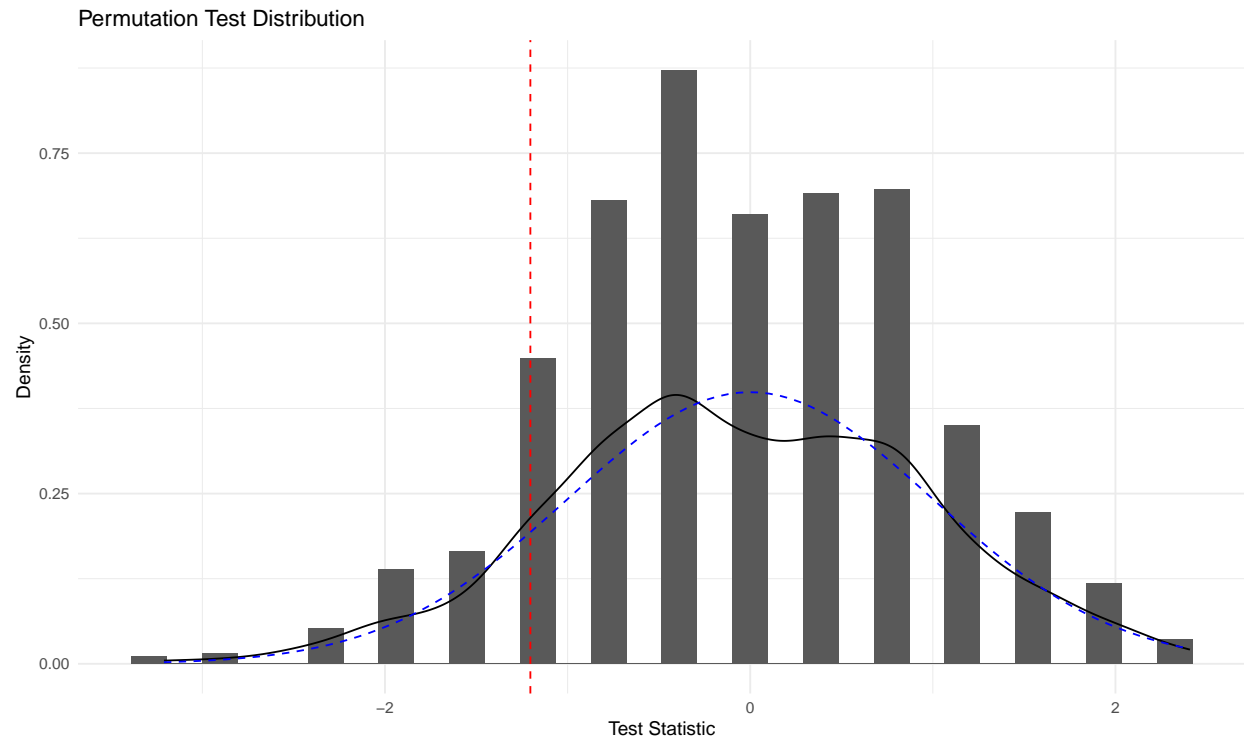

```
# Large effect
data_large <- sim_binary_data(0.8, 0.4, 50, 50)
plot_permutation_test(data_large, reps = 1000)
```

Permutation Test Distribution

Power Analysis Example

```r
# Function to calculate power
calc_power <- function(p1, p2, n, reps = 1000, alpha = 0.05) {
  results <- replicate(reps, {
    data <- sim_binary_data(p1, p2, n, n)
    perm_results <- permutation_test(data, 100)
    p_value <- mean(abs(perm_results$null_dist) >= abs(perm_results$obs_stat))
    p_value <= alpha
  })
  mean(results)
}

# Calculate power for different sample sizes
n_seq <- seq(20, 100, by = 20)
power_results <- sapply(n_seq, function(n) calc_power(0.7, 0.5, n))

# Plot power curve
ggplot(data.frame(n = n_seq, power = power_results), aes(x = n, y = power)) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept = 0.8, linetype = "dashed", color = "red") +
  labs(x = "Sample Size per Group",
       y = "Power",
       title = "Power Analysis for Detecting Difference between p1 = 0.7 and p2 = 0.5") +
  theme_minimal()
```


Power Analysis for Detecting Difference between p1 = 0.7 and p2 = 0.5
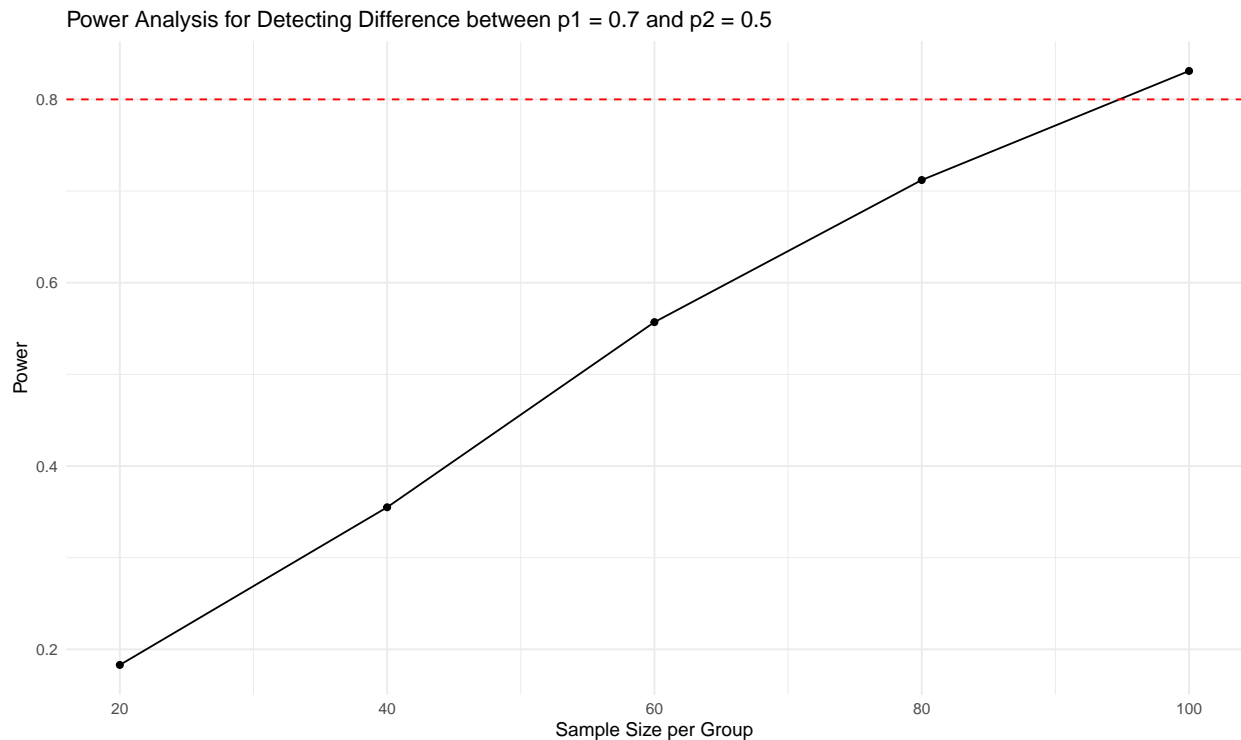
This completes our exploration of the statistical simulation functions. The examples demonstrate how to use each function and interpret the results, while also showing how they can be combined for more comprehensive

11

analyses.