

Understanding Confidence Intervals

Gemini

Introduction

This notebook demonstrates the concept of “confidence” in confidence intervals. We will generate many samples from a known population, compute a 95% confidence interval for the population mean for each sample, and then visualize these intervals.

Simulation

The following R code performs the simulation.

```
library(tidyverse)

# --- 1. Set Parameters ---
set.seed(123)
population_mean <- 100
population_sd <- 15
sample_size <- 30
n_samples <- 100
confidence_level <- 0.95

# --- 2. Generate Samples and Compute Confidence Intervals ---
samples_data <- replicate(n_samples, rnorm(sample_size, mean = population_mean, sd = population_sd))

ci_data <- samples_data |>
  map_dfr(~{
    sample_mean <- mean(.x)
    se <- sd(.x) / sqrt(sample_size)
    margin_error <- qt(1 - (1 - confidence_level) / 2, df = sample_size - 1) * se
    tibble(
      lower = sample_mean - margin_error,
      upper = sample_mean + margin_error,
```

```

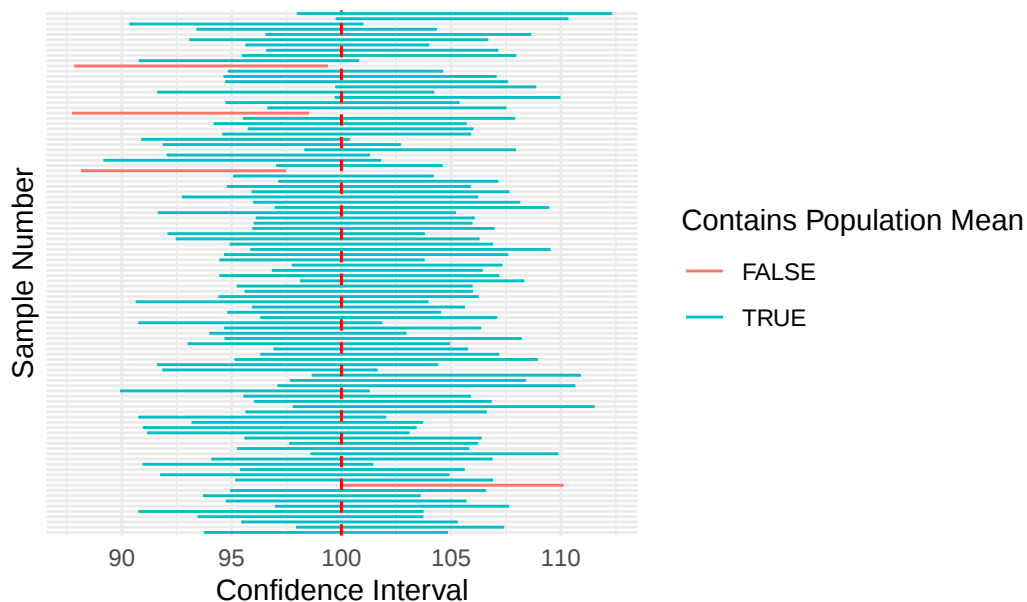
    sample_mean = sample_mean
  )
}) |>
mutate(
  sample_num = 1:n_samples,
  contains_mean = lower <= population_mean & upper >= population_mean
)

# --- 3. Plot the Confidence Intervals ---
ci_plot <- ggplot(ci_data, aes(x = factor(sample_num), ymin = lower, ymax = upper, color = c
  geom_errorbar(width = 0.5) +
  geom_hline(yintercept = population_mean, color = "red", linetype = "dashed") +
  coord_flip() +
  labs(
    title = "100 Confidence Intervals for the Population Mean",
    x = "Sample Number",
    y = "Confidence Interval",
    color = "Contains Population Mean"
  ) +
  theme_minimal() +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank())

print(ci_plot)

```

100 Confidence Intervals for the Population Mean



```
# --- 4. Count Intervals Containing the Mean ---
X <- sum(ci_data$contains_mean)
cat("Number of confidence intervals containing the population mean (X):", X, "\n")
```

Number of confidence intervals containing the population mean (X): 96

```
# --- 5. Estimate Observed Confidence Level ---
observed_confidence_level <- X / n_samples
cat("Observed level of confidence:", observed_confidence_level, "\n")
```

Observed level of confidence: 0.96

```
# --- 6. Compute 95% CI for the Confidence Level ---
confidence_interval_for_level <- binom.test(X, n_samples, p = confidence_level)
cat("95% confidence interval for the level of confidence:\n")
```

95% confidence interval for the level of confidence:

```
print(confidence_interval_for_level$conf.int)
```

```
[1] 0.9007428 0.9889955
attr("conf.level")
[1] 0.95
```

Law of Large Numbers for Confidence Levels

To see the Law of Large Numbers in action, we can run the simulation for a much larger number of trials. The following code runs the simulation 1,000 times and plots the convergence of the observed confidence level to the theoretical level of 95%. The shaded blue area represents the 95% confidence interval for the observed confidence level, which gets narrower as the number of simulations increases.

```
set.seed(789)
n_sims_lln <- 1000

# Run the simulation n_sims_lln times
lln_results <- map_lgl(1:n_sims_lln, ~{
  sample_data <- rnorm(sample_size, mean = population_mean, sd = population_sd)
```

```

sample_mean <- mean(sample_data)
se <- sd(sample_data) / sqrt(sample_size)
margin_error <- qt(1 - (1 - confidence_level) / 2, df = sample_size - 1) * se
lower_bound <- sample_mean - margin_error
upper_bound <- sample_mean + margin_error

# Check if the true mean is in the interval
lower_bound <= population_mean & upper_bound >= population_mean
})

# Calculate cumulative confidence level and its confidence interval
lln_convergence_data <- tibble(
  simulation_num = 1:n_sims_lln,
  success = cumsum(lln_results),
  cumulative_confidence_level = success / simulation_num
) |>
rowwise() |>
mutate(
  ci_lower = binom.test(success, simulation_num)$conf.int[1],
  ci_upper = binom.test(success, simulation_num)$conf.int[2]
) |>
ungroup()

# Plot the convergence
ggplot(lln_convergence_data, aes(x = simulation_num, y = cumulative_confidence_level)) +
  geom_line(color = "blue") +
  geom_ribbon(aes(ymin = ci_lower, ymax = ci_upper), alpha = 0.2, fill = "blue") +
  geom_hline(yintercept = confidence_level, color = "red", linetype = "dashed") +
  labs(
    title = "Convergence of Observed Confidence Level to 95%",
    subtitle = "Based on 1000 Simulations",
    x = "Number of Simulated Confidence Intervals",
    y = "Observed Confidence Level (Cumulative)"
  ) +
  ylim(0.85, 1.0) + # Zoom in on the convergence
  theme_minimal()

```

Warning: Removed 6 rows containing missing values or values outside the scale range (`geom\_line()`).

Warning: Removed 59 rows containing missing values or values outside the scale range

```
(`geom_ribbon()`).
```

