# STA 6351, Report.2.3

**Carson Slater** *Baylor University*

### 2.3

Suppose $X_1, \ldots, X_n \overset{\text{iid}}{\sim} N(\mu, \sigma^2)$, with $\sigma^2$ known. The usual $100(1-\alpha)\%$ confidence interval for $\mu$ is

$$C(\mathbf{X}) = \left[\bar{X} - z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}},\ \bar{X} + z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}\right].$$

**Verify that, under repeated sampling, this interval satisfies**

$$\Pr_{\mu}(\mu \in C(\mathbf{X})) = 1 - \alpha.$$

**That is, the coverage probability is exactly $1-\alpha$ when averaged over all possible samples.**

The coverage probability is defined as $\Pr_{\mu}(\mu \in C(\mathbf{X}))$. We substitute the definition of the confidence interval $C(\mathbf{X})$:

$$\Pr_{\mu}(\mu \in C(\mathbf{X})) = \Pr_{\mu}\left(\bar{X} - z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}} \le \mu \le \bar{X} + z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}\right)$$

We can rearrange the inequalities to isolate the sample mean $\bar{X}$:

$$\Pr_{\mu}(\mu \in C(\mathbf{X})) = \Pr_{\mu}\left(\mu - z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}} \le \bar{X} \le \mu + z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}\right)$$

Since $X_1, \ldots, X_n \overset{\text{iid}}{\sim} N(\mu, \sigma^2)$, the sample mean $\bar{X}$ also follows a normal distribution: $\bar{X} \sim N(\mu, \sigma^2/n)$. We standardize $\bar{X}$ by subtracting its mean $\mu$ and dividing by its standard deviation $\sigma/\sqrt{n}$. Let $Z = \frac{\bar{X}-\mu}{\sigma/\sqrt{n}}$, so $Z \sim N(0,1)$. Applying this transformation to the inequalities:

$$\begin{aligned}
\Pr_{\mu}(\mu \in C(\mathbf{X})) &= \Pr_{\mu}\left(\frac{\mu - z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}} - \mu}{\sigma/\sqrt{n}} \le \frac{\bar{X}-\mu}{\sigma/\sqrt{n}} \le \frac{\mu + z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}} - \mu}{\sigma/\sqrt{n}}\right) \\
&= \Pr_{\mu}\left(\frac{-z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}}{\sigma/\sqrt{n}} \le Z \le \frac{+z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}}{\sigma/\sqrt{n}}\right) \\
&= \Pr(-z_{1-\alpha/2} \le Z \le z_{1-\alpha/2})
\end{aligned}$$

Since $z_{1-\alpha/2}$ is the $(1-\alpha/2)$-th quantile of the standard normal distribution $\Phi$, we have $\Phi(z_{1-\alpha/2}) = 1-\alpha/2$. By symmetry, $-z_{1-\alpha/2}$ is the $\alpha/2$-th quantile, so $\Phi(-z_{1-\alpha/2}) = \alpha/2$. Therefore, the probability is:

$$\begin{aligned}
\Pr_{\mu}(\mu \in C(\mathbf{X})) &= \Phi(z_{1-\alpha/2}) - \Phi(-z_{1-\alpha/2}) \\
&= (1-\alpha/2) - \alpha/2 \\
&= 1-\alpha.
\end{aligned}$$

The coverage probability is exactly $1-\alpha$.

**Define the indicator variable**

$$A = I(\bar{X} > 0),$$

**which partitions the sample space into two disjoint regions. Show that $A$ is an ancillary statistic – its distribution does not depend on $\mu$.**

Recall: a statistic $T(\mathbf{X})$ is *ancillary* for parameter $\mu$ if its sampling distribution does not depend on $\mu$.

Compute the distribution of $A$. Since $\bar{X} \sim N(\mu, \sigma^2/n)$,

$$\Pr_\mu(A=1) = \Pr_\mu(\bar{X} > 0) = 1 - \Phi\Big(\frac{0-\mu}{\sigma/\sqrt{n}}\Big) = 1 - \Phi\Big(-\frac{\mu}{\sigma/\sqrt{n}}\Big).$$

Using $1 - \Phi(-t) = \Phi(t)$ we obtain

$$\Pr_\mu(A=1) = \Phi\Big(\frac{\mu}{\sigma/\sqrt{n}}\Big), \qquad \Pr_\mu(A=0) = 1 - \Phi\Big(\frac{\mu}{\sigma/\sqrt{n}}\Big).$$

These probabilities depend on $\mu$ (unless $\mu = 0$). Therefore the distribution of $A$ depends on $\mu$, so $A$ is *not* ancillary for $\mu$.

**Derive expressions for the conditional coverage probabilities.**

$$\Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 1) \quad \text{and} \quad \Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 0).$$

*Hint:* Express these in terms of the standard normal cumulative distribution function $\Phi(\cdot)$ and let $x = \mu/(\sigma/\sqrt{n})$.

The conditional coverage probability is defined as:

$$\Pr_\mu(\mu \in C(\mathbf{X}) \mid A) = \frac{\Pr_\mu(\mu \in C(\mathbf{X}) \cap A)}{\Pr_\mu(A)}$$

Let $z = z_{1-\alpha/2}$ and $x = \mu/(\sigma/\sqrt{n})$. The coverage event is equivalent to $-z \le Z \le z$, where $Z = \frac{\bar{X}-\mu}{\sigma/\sqrt{n}} \sim N(0,1)$.

*Conditional Coverage for* $A = 1 \iff \bar{X} > 0 \iff Z > -x$

The denominator is $\Pr_\mu(A=1) = \Pr(Z > -x) = 1 - \Phi(-x) = \Phi(x)$. The numerator is $\Pr_\mu(\mu \in C(\mathbf{X}) \cap A = 1) = \Pr(-z \le Z \le z \cap Z > -x) = \Pr(\max(-z, -x) < Z \le z)$.

$$\begin{aligned}\Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 1) &= \frac{\Phi(z) - \Phi(\max(-z, -x))}{\Phi(x)} \\ &= \frac{1 - \alpha/2 - \Phi(\max(-z_{1-\alpha/2}, -x))}{\Phi(x)}\end{aligned}$$

*Conditional Coverage for* $A = 0 \iff \bar{X} \leq 0 \iff Z \leq -x$

The denominator is $\Pr_\mu(A = 0) = \Pr(Z \leq -x) = \Phi(-x)$. The numerator is $\Pr_\mu(\mu \in C(\mathbf{X}) \cap A = 0) = \Pr(-z \leq Z \leq z \cap Z \leq -x) = \Pr(-z \leq Z \leq \min(z, -x))$.

$$\begin{aligned}\Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 0) &= \frac{\Phi(\min(z, -x)) - \Phi(-z)}{\Phi(-x)} \\ &= \frac{\Phi(\min(z_{1-\alpha/2}, -x)) - \alpha/2}{1 - \Phi(x)}\end{aligned}$$

**Show that the unconditional coverage probability is recovered by averaging the two conditional probabilities:**

$$(1 - \alpha) = \Phi(x) \Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 1) + \Phi(-x) \Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 0).$$

The unconditional coverage probability, $\Pr_\mu(\mu \in C(\mathbf{X}))$, can be expressed using the Law of Total Probability (LTP) over the partition defined by the indicator variable $A$:

$$\begin{aligned}\Pr_\mu(\mu \in C(\mathbf{X})) &= \Pr_\mu(\mu \in C(\mathbf{X}) \cap A = 1) + \Pr_\mu(\mu \in C(\mathbf{X}) \cap A = 0) \\ &= \Phi(x) \Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 1) + \Phi(-x) \Pr_\mu(\mu \in C(\mathbf{X}) \mid A = 0)\end{aligned}$$

The two events, $(\mu \in C(\mathbf{X}) \cap A = 1)$ and $(\mu \in C(\mathbf{X}) \cap A = 0)$, are disjoint, and their union is simply the event $\mu \in C(\mathbf{X})$ itself, since $(A = 1) \cup (A = 0)$ covers the entire sample space.

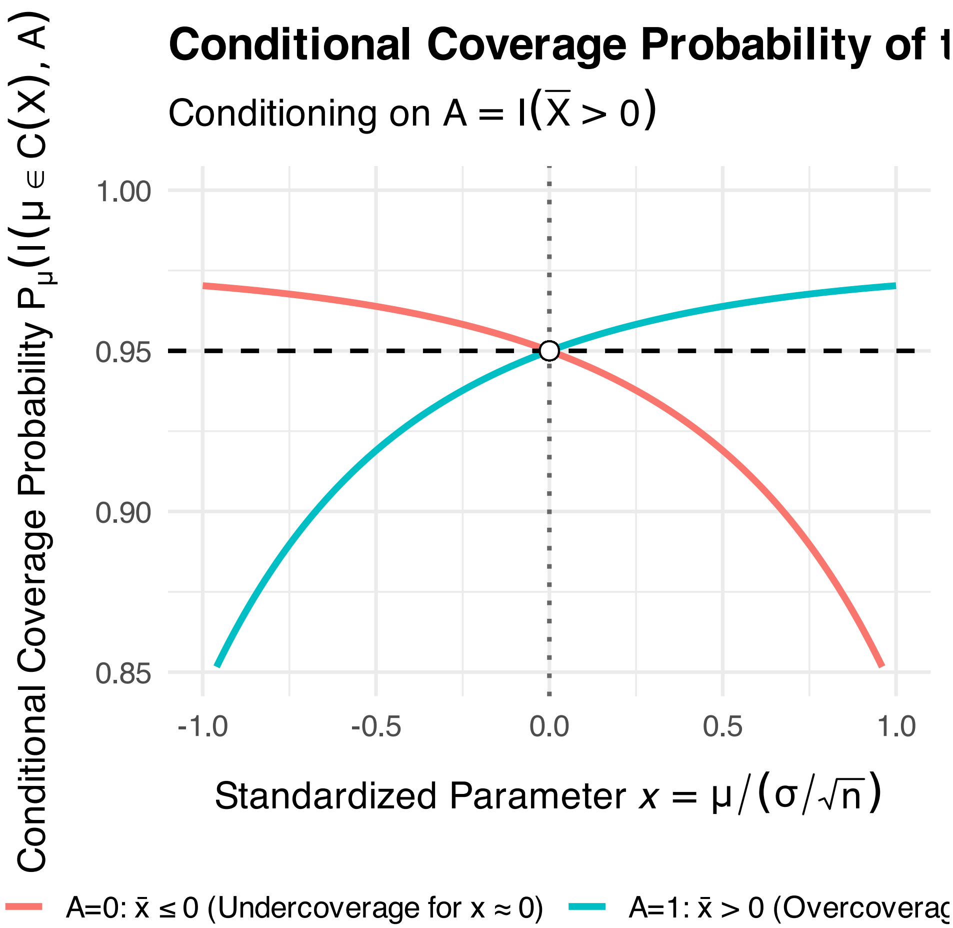
- $\mu \in C(\mathbf{X})$ is equivalent to $-z \leq Z \leq z$.
- $A = 1$ is equivalent to $Z > -x$.
- $A = 0$ is equivalent to $Z \leq -x$.

Therefore, summing the two joint probabilities recovers the probability of the unconditional event:

$$\begin{aligned}\Pr_\mu(\mu \in C(\mathbf{X})) &= \Pr(-z \leq Z \leq z \cap Z > -x) + \Pr(-z \leq Z \leq z \cap Z \leq -x) \\ &= \Pr(-z \leq Z \leq z \cap (Z > -x \cup Z \leq -x)) \\ &= \Pr(-z \leq Z \leq z) \\ &= \Phi(z) - \Phi(-z) \\ &= (1 - \alpha/2) - \alpha/2 \\ &= 1 - \alpha.\end{aligned}$$

The unconditional coverage probability $1 - \alpha$ is exactly recovered.

**Use numerical evaluation or a simple plot to confirm that, near $x = 0$, the conditional coverages differ from $1 - \alpha$; that is, the usual confidence interval **overcovers** when $\bar{X} > 0$ and **undercovers** when $\bar{X} \leq 0$.**




**Reflect on the implication: the standard interval procedure has good global repeated-sampling properties, yet it exhibits poor conditional behavior for particular outcomes. This illustrates Fisher's view that conditioning on ancillary information yields inferences that more faithfully represent the data actually observed.**

The statistic $A$ itself was *not* ancillary, as its distribution depended on $\mu$. Nonetheless, the exercise highlights the general principle that a procedure's good global properties (unconditional coverage) can mask severe defects when looking at specific subsets of the sample space defined by relevant auxiliary statistics. Ideally, one would condition on an ancillary statistic to ensure the conditional inference reflects the *precision* observed in the data.

**Appendix**

```
knitr::opts_chunk$set(
  dev = "cairo_pdf",
  fig.width = 5,
  fig.height = 5,
  fig.align = 'center',
  echo = FALSE,
  message = FALSE,
  warning = FALSE,
  error = FALSE,
  results = 'markup'
)

# Load required libraries
library("tidyverse")
library("patchwork")
library("glue")
library("scales", warn.conflicts = FALSE)
library("extrafont")
library("tinytex")
library("knitr")
library("tidyr")
library("latex2exp")
library("MASS")
library("kableExtra")

theme_set(theme_minimal(base_family = "Roboto Condensed"))

conflicted::conflicts_prefer(
  readr::col_factor(),
  purrr::discard(),
  dplyr::lag(),
  readr::parse_date(),
  kableExtra::group_rows(),
  dplyr::select
)
## --- Parameters ---
alpha <- 0.05
conf_level <- 1 - alpha
z_alpha_half <- qnorm(1 - alpha / 2)
# Let z be the shorthand for the critical value
z <- z_alpha_half

## --- Helper Functions for Conditional Coverage ---

#' Calculates the conditional coverage probability given A=1 (i.e., X_bar > 0
```

```
#' @param x The standardized parameter: x = mu / (sigma/sqrt(n))
#' @param z The critical value: z = z_{1-alpha/2}
conditional_coverage_A1 <- function(x, z) {
  # Denominator: Pr(A=1) = Pr(Z > -x) = Phi(x)
  denom <- pnorm(x)
  # Numerator: Pr(-z <= Z <= z AND Z > -x) = Pr(max(-z, -x) < Z <= z)
  numer <- pnorm(z) - pnorm(pmax(-z, -x))

  # Handle division by zero/near zero if x is very negative (Pr(A=1) -> 0)
  # In theory, this is the limit: Pr(A=1) -> 0, so conditional probability is
  # For practical plotting, we can return 0 or NA for extreme values.
  ifelse(denom > 1e-10, numer / denom, NA)
}

#' Calculates the conditional coverage probability given A=0 (i.e., X_bar <=
#' @param x The standardized parameter: x = mu / (sigma/sqrt(n))
#' @param z The critical value: z = z_{1-alpha/2}
conditional_coverage_A0 <- function(x, z) {
  # Denominator: Pr(A=0) = Pr(Z <= -x) = Phi(-x)
  denom <- pnorm(-x)
  # Numerator: Pr(-z <= Z <= z AND Z <= -x) = Pr(-z <= Z <= min(z, -x))
  numer <- pnorm(pmin(z, -x)) - pnorm(-z)

  # Handle division by zero/near zero if x is very positive (Pr(A=0) -> 0)
  ifelse(denom > 1e-10, numer / denom, NA)
}

## --- Generate Data for Plotting ---

# Create a sequence of x values (standardized mean parameter)
x_values <- seq(-3, 3, by = 0.01)

# Calculate conditional coverages for all x values
plot_data <- tibble(x = x_values) |>
  mutate(
    Coverage_A1 = conditional_coverage_A1(x, z),
    Coverage_A0 = conditional_coverage_A0(x, z)
  ) |>

  # Convert to long format for ggplot
  tidyr::pivot_longer(
    cols = starts_with("Coverage_"),
    names_to = "Condition",
    values_to = "Coverage"
  ) |>
  # Relabel conditions for plot legend
  mutate(
    Condition = case_when(
```

```
      Condition ==
        "Coverage_A1" ~ "A=1: x\u0304 > 0 (Overcoverage for x \u2248 0)",
      Condition ==
        "Coverage_A0" ~ "A=0: x\u0304 \u2264 0 (Undercoverage for x \u2248 0)"
      TRUE ~ Condition
    )
  )

## --- Create the Plot ---

conditional_coverage_plot <- plot_data |>
  ggplot(aes(x = x, y = Coverage, color = Condition)) +

  # Add the conditional coverage lines
  geom_line(linewidth = 1.2) +

  # Add the nominal coverage level
  geom_hline(
    yintercept = conf_level,
    linetype = "dashed",
    color = "black",
    linewidth = 0.8
  ) +

  # Add a vertical line at x=0
  geom_vline(
    xintercept = 0,
    linetype = "dotted",
    color = "gray40",
    linewidth = 0.8
  ) +

  # Highlight the region around x=0 where the conditional coverages differ
  geom_point(
    aes(y = conf_level),
    x = 0,
    shape = 21,
    fill = "white",
    color = "black",
    size = 3
  ) +

  # Labels and Titles
  labs(
    title = paste0(
      "Conditional Coverage Probability of the ",
      conf_level * 100,
      "% Z-Interval"
```

```
    ),
    subtitle = latex2exp::TeX("Conditioning on $A = I(\\bar{X} > 0)$"),
    x = expression(paste(
      "Standardized Parameter ",
      italic(x) ==
        mu /
          ({
            sigma
          } /
            sqrt(n))
    )),
    y = expression(paste(
      "Conditional Coverage Probability ",
      P[mu](mu %in% C(X) | A)
    ))
  ) +
  # Adjust y-axis limits to focus on deviation
  scale_y_continuous(
    limits = c(0.85, 1.05),
    breaks = c(seq(0.85, 1.05, by = 0.05), conf_level)
  ) +
  xlim(c(-1, 1)) +

  ylim(c(0.85, 1)) +

  # Theme and styling
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "bottom",
    legend.title = element_blank(),
    plot.title = element_text(face = "bold"),
    axis.title.y = element_text(margin = margin(r = 10)),
    axis.title.x = element_text(margin = margin(t = 10))
  )

# Display the plot
print(conditional_coverage_plot)
```