POL501 – Preliminaries: Problem Set 3

Introduction to Statistics

2024-11-04

Table of Contents

[Preliminary Setup. 1](#_Toc181649422)

[Demonstration of the function confidence\_interval to compute confidence intervals 1](#_Toc181649423)

[Demonstration of the command z.test to run hypothesis tests 5](#_Toc181649424)

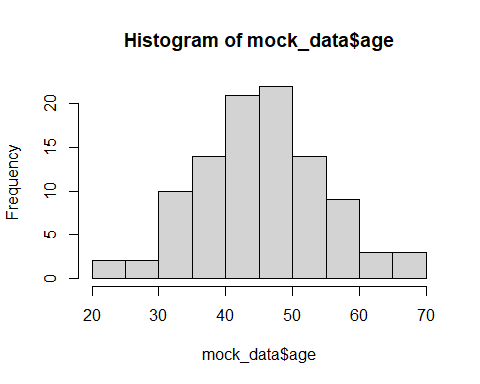
# Preliminary Setup.

## Demonstration of the function confidence\_interval to compute confidence intervals

First, we will start by defining a user defined function that will help us compute confidence intervals.

Now we will simulate some data and we will use this function as a demonstration:

# Example usage with mock data  
# Create a mock data frame with a column of ages  
# We will assume Age follows a normal distribution with mean 45 and SD 10.  
set.seed(123)  
mock\_data <- data.frame(age = abs(round(rnorm(100, mean = 45, sd = 10), 0)) )  
  
# Let's visualize the simulated data  
hist(mock\_data$age)



Using this simulated data, we will illustrate the use of the new function to compute confidence intervals. Since, we simulated the data under the assumption of a normal distribution, we know that the sampling distribution of the sample mean for this population will follow a normal distribution. Hence, we will use the z critical values to compute the margin of error (MOE).

# Calculate the confidence interval for the 'age' column with 95% confidence using the z-distribution  
result\_CI\_z\_95 <- confidence\_interval(data = mock\_data, var\_name = 'age', confidence\_level = 0.95, method = "z")  
  
# Print  
print(result\_CI\_z\_95)

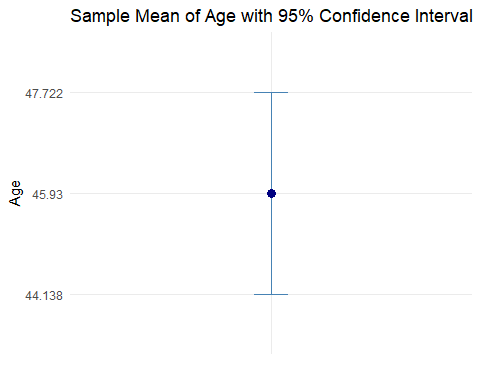
## Sample Mean of Estimate MOE Lower CI Bound Upper CI Bound   
## "age" "45.93" "1.792" "44.138" "47.722"

# Display using a table in the word doc - using `knitr::kable()` and using data.frame()  
kable(data.frame(result\_CI\_z\_95))

|  | result\_CI\_z\_95 |
| --- | --- |
| Sample Mean of | age |
| Estimate | 45.93 |
| MOE | 1.792 |
| Lower CI Bound | 44.138 |
| Upper CI Bound | 47.722 |

We can now plot this result, for better presentation. We can use ggplot to plot the result using a graph bar with whiskers representing the interval lower and upper bounds

# Extract the values from the result to use in the plot and Create a data frame for ggplot  
plot\_data <- data.frame(  
 Estimate = result\_CI\_z\_95["Estimate"],  
 Lower = result\_CI\_z\_95["Lower CI Bound"],  
 Upper = result\_CI\_z\_95["Upper CI Bound"]  
)  
  
# Plot using ggplot2 with a dot and whiskers for confidence interval  
ggplot(plot\_data, aes(x = "Age", y = Estimate)) +  
 geom\_errorbar(aes(ymin = Lower, ymax = Upper), width = 0.1, color = "steelblue") + # Whiskers for CI  
 geom\_point(size = 3, color = "navyblue") + # Dot for the sample mean  
 labs(  
 title = "Sample Mean of Age with 95% Confidence Interval",  
 x = "",  
 y = "Age"  
 ) +  
 theme\_minimal() +  
 theme(axis.text.x = element\_blank(), axis.ticks.x = element\_blank())



Note that if we want a confidence interval with 99% confidence we can just change the appropriate input of the function:

# Calculate the confidence interval for the 'age' column with 99% confidence using the z-distribution  
result\_CI\_z\_99 <- confidence\_interval(data=mock\_data, var\_name='age', confidence\_level = 0.99, method = "z")  
  
# Print  
print(result\_CI\_z\_99)

## Sample Mean of Estimate MOE Lower CI Bound Upper CI Bound   
## "age" "45.93" "2.355" "43.575" "48.285"

# Display using a table in the word doc - using `knitr::kable()` and using data.frame()  
kable(data.frame(result\_CI\_z\_99))

|  | result\_CI\_z\_99 |
| --- | --- |
| Sample Mean of | age |
| Estimate | 45.93 |
| MOE | 2.355 |
| Lower CI Bound | 43.575 |
| Upper CI Bound | 48.285 |

If we compute multiple confidence intervals of different confidence levels, or different variables, we can join the results using cbind

# Demonstration of `cbind`  
merge\_columns <- cbind(result\_CI\_z\_95, result\_CI\_z\_99)  
print(merge\_columns)

## result\_CI\_z\_95 result\_CI\_z\_99  
## Sample Mean of "age" "age"   
## Estimate "45.93" "45.93"   
## MOE "1.792" "2.355"   
## Lower CI Bound "44.138" "43.575"   
## Upper CI Bound "47.722" "48.285"

# Display using a table in the word doc - using `kable` and `data.frame`  
kable(data.frame(merge\_columns))

|  | result\_CI\_z\_95 | result\_CI\_z\_99 |
| --- | --- | --- |
| Sample Mean of | age | age |
| Estimate | 45.93 | 45.93 |
| MOE | 1.792 | 2.355 |
| Lower CI Bound | 44.138 | 43.575 |
| Upper CI Bound | 47.722 | 48.285 |

## Demonstration of the command z.test to run hypothesis tests

The z.test function is used to test the null hypothesis that the mean of a sample is equal to a hypothesized population mean. This test is appropriate when the sample size is large (typically n > 30) or the population standard deviation is known and the underlying random variable follows a normal distribution. We can run the test to determine if there is evidence to suggest that the sample mean differs from a specified value.

To run the z-test, we will first install the package BSDA which contains the z.test function. This package includes useful statistical methods for data analysis.

# Install BSDA package if not already installed  
if (!requireNamespace("BSDA", quietly = TRUE)) {  
 install.packages("BSDA")  
}

Once the package is installed, we can run the z.test on our mock data to determine if the mean age differs from 45 years.

# Load BSDA library  
library(BSDA)  
  
# Run the z-test on the age data to test if the mean is significantly different from 45  
z\_test\_result <- z.test(  
 x = mock\_data$age,  
 mu = 45, # Hypothesized population mean  
 sigma.x = sd(mock\_data$age), # Population standard deviation (we are assuming known)  
 conf.level = 0.95  
)  
  
# Print the z-test result  
print(z\_test\_result)

##   
## One-sample z-Test  
##   
## data: mock\_data$age  
## z = 1.017, p-value = 0.3091  
## alternative hypothesis: true mean is not equal to 45  
## 95 percent confidence interval:  
## 44.13772 47.72228  
## sample estimates:  
## mean of x   
## 45.93

In the output, we get the z-value, the p-value, and the confidence interval of the sample mean. The p-value helps us determine whether we can reject the null hypothesis. If the p-value is less than the significance level (typically 0.05), we reject the null hypothesis and conclude that there is sufficient evidence to suggest that the sample mean differs from the hypothesized population mean.

The confidence interval also gives us an idea of the range in which the true population mean lies, with a given level of confidence.

Here we see that the p-value is higher than 0.10 therefore, we cannot reject the null hypothesis that the true mean is equal to 45.