Statistical Thinking: Week 5 Lab

Due 12noon Wednesday 9 September 2020

Introduction

We continue the topic of assessing variability in means this week, with our focus turning to the Bootstrap method.

This document is organised in three main sections, Sections A, B and C, as shown below. Section B focuses on a comparison of the Bootstrap- and CLT-based confidence intervals when the data are simulated from a known distribution. Section C covers the application of the Bootstrap confidence interval techniques applied to the CBT data used in the Week 4 Lab. Section A provides some general instructions.

A. Instructions (including Lab 5 submission deadline).

- Be sure to download and use the **Lab05script.R** file available on Moodle!
- Also remember to load the libraries before using functions from a package.

B. Simulated examples:

- B.1 Illustration of the Bootstrap method when estimating the mean of a normal distribution, and
- B.2 Illustration of the Bootstrap method when estimating the mean of a Gamma(shape=3, scale=2) distribution.

C. Application of Bootstrap method with the **CBT** data.

A. Instructions (including Lab 4 submission deadline)

Before you begin **B. Lab Exercises**, review each of these detailed instructions.

- Create, name and save a new .Rmd file as **Lab05_nnnnnnnn.Rmd**, with your Monash student ID number replacing the segment **nnnnnnnn** in the file name. Including in the **YAML** section
  - a suitable title (e.g. "Statistical Thinking Lab 5")
  - put your name and student-id as the author information
  - set the date "Week 5, 2020"
  - ensure all code chunks[1] will show using the `knitr::opts_chunk$set(echo = TRUE)` command in the initial code chunk.
  - be sure to include all package library commands in the .Rmd
  - use your discretion with regard to further modification of the plots and tables for which **R** commands have been provided.

- Format your **R Markdown** files using (sub-)section heading titles corresponding to the Lab exercise section headings. Refer to Lab 3 for other relevant suggestions useful to complete prior to undertaking the Lab and for preparing files for submission.

[1] The initial code chunk containing the *kitr::opts_chunk* settings do not need to be displayed.

- Lab 5 submissions must be completed before **12noon on Wednesday 9 September 2020**.

Good luck and have fun!

B. Simulated examples

Exercise B1: Detailed steps numbered i. through vi. and **R** code are provided that produce a Bootstrap confidence interval for the population mean of a $N(\mu, \sigma^2)$ distribution, with the sample mean, $\bar{X}$, used as the point estimator for μ. Carefully review these steps, and consider how they relate to the commands in the **R** code chunk that follows. Then, reproduce the code chunk in your report - it's contents are available for you to copy in the **Lab05_ExB1.R** script file in Moodle. Once you have successfully reproduced the code and obtain the required output, consider the questions below. The question are predominantly for your discussion and reflection, however you are to briefly summarise **at least two insights** you have gained from this exercise in your Lab report.

- How do the two confidence intervals produced for μ compare in this illustration? That is, in what ways are they similar or different?
- Does the Bootstrap confidence interval appear to change very much when you increase B?
- What happens when you increase the sample size n?
- What do you think will happen if you change the population distribution to one that is skewed, or multi-modal? (Consider this *before* you try Exercise B2!)
- Do you think the Bootstrap confidence intervals do a good job of "characterising uncertainty" you have about the true value of μ? Why or why not?

Steps for Exercise B1

i. Simulate $n = 30$ independent observations from a $N(3, 4)$ distribution. Save these draws in a vector named **x**, and display a plot of the sample histogram overlayed with a kernal density plot.

ii. Pretending that you don't know the values of either $\mu = 3$ or $\sigma = 2$, report the CLT-based 95% confidence interval for μ. Save the value of the point estimate, $\bar{x}$.

iii. Generate $B = 5000$ Bootstrap samples from the sample **x**. For each Bootstrap sample, calculate and save the sample mean value, storing all Bootstrap sample means in a vector named **xbar_boot**.

iv. Report the approximate 95% Bootstrap confidence interval obtained for μ.

v. Use the new function named **bootplot.f** to plot the density estimate (overlayed histogram and kernal density plot) corresponding to the empirical Bootstrap distribution of $\bar{X}$, as well as the position of the 2.5% and 97.5% quantiles of this distribution shown using vertical dotted lines. In addition, indicate on the graph each of the following items:

- the position of the sample mean, $\bar{x}$, shown by a solid "red" vertical line,
- the numerical values, shown in "magenta" colour, of the lower 2.5% and 97.5% quantiles of the empirical Bootstrap distribution,
- the position and numerical values of the lower and upper end points of the CLT-based confidence interval (from part i), shown by "darkslategrey" vertical dotted lines and numbers,
- the population density, as a "blue" line plot, together with a solid, "blue" vertical line at the true value of the population mean (here mu.true=$\mu = 3$),
- an appropriate title, and
- appropriately modified axis labels.

vi. Save the final plot as the named object **p1**, and display it.

Code for Exercise B1

```
# Define functions at the start - so you will have it when
# you need it
############ the bootplot.f function #############
#
## This function 'bootplot.f' takes a vector of Bootstrap
## samples as the main argument ('stat_boot'), and produces a
## plot showing the histogram, with smooth density estimate
## overlay,and also provides a option (detail) for the number
## of *bins* used in the histogram. You will need to run
## through the function code once to save it as an object
## before you can use the function.
#
bootplot.f <- function(stat.boot, bins = 50) {
    df <- tibble(stat = stat.boot)
    CI <- round(quantile(stat.boot, c(0.025, 0.975)), 2)
    p <- df %>% ggplot(aes(x = stat, y = ..density..)) + geom_histogram(bins = bins,
        colour = "magenta", fill = "magenta", alpha = 0.2) +
        geom_density(fill = "magenta", colour = "magenta", alpha = 0.2) +
        geom_vline(xintercept = CI, colour = "magenta", linetype = 3) +
        theme_bw()
    p
}
######## end of bootplot.f function #######
#
######## STEPS ############
## step i.
set.seed(57892)
n <- 30
mu.true <- 3
sig.true <- 2
x <- rnorm(n = n, mean = mu.true, sd = sig.true)  # simulated values
dt_data <- tibble(x = x)
p1_simdata <- dt_data %>% ggplot(aes(x = x, y = ..density..)) +
    geom_histogram(colour = "steelblue", fill = "steelblue",
        alpha = 0.2) + geom_density(colour = "steelblue", fill = "steelblue",
    alpha = 0.2) + xlim(c(-4, 10)) + theme_bw()
p1_simdata
#
## step ii.
xbar.x <- mean(x)  # point estimate
SE.x <- sd(x)/sqrt(n)  # estimated SE
ttest.out <- t.test(x) %>% tidy()  #obtain the standard t-test output
CLT.CI <- c(ttest.out$conf.low, ttest.out$conf.high)
#
## step iii.
B <- 5000
xbar_boot <- rep(NA, B)
for (i in 1:B) {
    temp <- sample(x, size = n, replace = TRUE)
    xbar_boot[i] <- mean(temp)
}
```

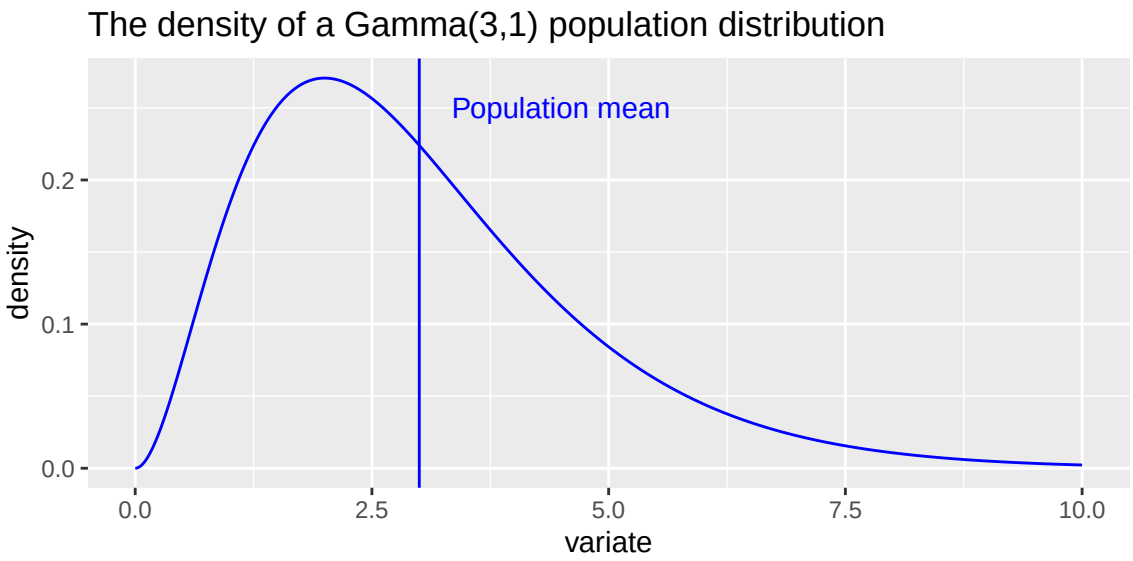
```
#
## step iv.
boot.CI <- quantile(xbar_boot, c(0.025, 0.975))
boot.CI
#
## step v.
# get the basic plot
p_xbarboot <- bootplot.f(xbar_boot, bins = 100)
## then add layers set up a range to view the entire
## population density
len <- (max(xbar_boot) - min(xbar_boot))/3
xxmax <- (max(xbar_boot) + sqrt(n) * len)
xxmin <- (min(xbar_boot) - sqrt(n) * len)
## define tibble to add layer for the population density
xx <- seq(xxmin, xxmax, length.out = 1000)
popn <- dnorm(xx, mean = mu.true, sd = sig.true)
dt <- tibble(xx = xx, popn = popn)
## layer the plot
p_xbarboot <- p_xbarboot + geom_vline(xintercept = xbar.x, colour = "red") +
    annotate("text", label = round(boot.CI[1], 2), x = (boot.CI[1] -
        0.5), y = 0.5, colour = "magenta") + annotate("text",
    label = round(boot.CI[2], 2), x = (boot.CI[2] + 0.5), y = 0.5,
    colour = "magenta") + geom_vline(xintercept = CLT.CI, colour = "darkslategrey",
    linetype = 2) + annotate("text", label = round(CLT.CI[1],
    2), x = (boot.CI[1] - 0.5), y = 0.75, colour = "darkslategrey") +
    annotate("text", label = round(CLT.CI[2], 2), x = (boot.CI[2] +
        0.5), y = 0.75, colour = "darkslategrey") + geom_line(data = dt,
    aes(x = xx, y = popn), colour = "blue") + geom_vline(xintercept = mu.true,
    colour = "blue")
## add titles and change axis labels, save plot as object p1
p1 <- p_xbarboot + xlab(expression(bar(x)))
p1 <- p1 + ggtitle(expression(paste("Bootstrap-based approximate sampling distribution of ",
    bar(X))), "N(3,4), n=30 and B=5000")
#
## step vi.
p1
```

Exercise B2: Copy the code provided from **Exercise B1** and modify it to produce the approximate confidence intervals for the population mean of a $Gamma(shape = 3,\ rate = 1)$ distribution, again with the sample mean, $\bar{X}$, used as the point estimator for the population mean. Report the CLT- and Bootstrap-based confidence intervals for the population mean, denoted again as μ and with $\mu = 3$, and describe the apparent similarities and differences bewteen the two approaches.

As background information, note that the $Gamma(shape = a, rate = b)$ distribution takes values only on the positive real line, and has a mean value equal to a/b. The code below produces the subsequent plot that illustrates the $Gamma(3, 1)$ distribution.

```
a <- 3
b <- 1
gmean <- a/b
xx_gamma <- seq(0.001, 10, length.out = 1000)
popn_gamma <- dgamma(xx_gamma, shape = a, rate = b)
dt_gamma <- tibble(xx = xx_gamma, popn = popn_gamma)
pp <- dt_gamma %>% ggplot(aes(x = xx, y = popn)) + geom_line(colour = "blue")
```

```
pp <- pp + ggtitle("The density of a Gamma(3,1) population distribution")
pp <- pp + xlab("variate") + ylab("density") + geom_vline(xintercept = gmean,
    colour = "blue") + annotate(geom = "text", label = "Population mean",
    colour = "blue", x = gmean + 1.5, y = 0.25)
pp
```



C. Application

Part C. Refer to **Lab 4 Section B2**, with regard to the **CBT** dataset, to complete the exercise **Exercise C1** and **Exercise C2** below.

Exercise C1: Consider the variable named *Diff* constructed from the difference in CBT assessment scores, with *Diff=score 2 - score1*. Use both the CLT- and Bootstrap-based approaches to obtain a 95% confidence interval for each of the unknown population mean $\delta = \mu_2 - \mu_1$. In addition, provide a plot of the approximate sampling distribution for the relevant point estimator, $\bar{X}$, with the point estimate and 95% confidence bounds indicated for each method. Provide a description of each output in your Lab report. Comment on any apparent similarities or differences between the two intervals or the appropriateness of the methods for this application.

This will require some slight modifications to the code used in Part B, as reflected in the modified steps highlighted in **bold** below:

Steps for Exercise C1

i. **Import the CBT data, calculate the Diff variable and set x=Diff.** Then display a plot of the sample histogram overlayed with a kernal density plot.

ii. **Now we don't need to pretend...as we don't know** the values of either μ or the population variance, report the CLT-based 95% confidence interval for μ. Save the value of the point estimate, in this case $\bar{x}$.

iii. Generate $B = 5000$ Bootstrap samples from the sample **x**. For each Bootstrap sample, calculate and save the sample mean value, storing them all in a vector named **xbar_boot**.

iv. Report the 95% Bootstrap confidence interval for μ.

v. Use the new function named **bootplot.f** to plot the density estimate (overlayed histogram and kernal density plot) corresponding to the empirical Bootstrap distribution of $\bar{X}$ and the position of the 2.5% and 97.5% quantiles of this distribution. In addition, indicate on the graph each of the following items:

- the position of the sample mean, $\bar{x}$, shown by a solid "red" vertical line,
- the numerical values, shown in "magenta" colour, of the lower 2.5% and 97.5% quantiles of the empirical Bootstrap distribution,
- the position and numerical values of the lower and upper end points of the CLT-based confidence interval (from part i), shown by "darkslategrey" vertical dotted lines and numbers,
- **there is no known population density, so do not try to include one, or vertical line for its mean value**,
- an appropriate title, and
- appropriately modified axis labels.

vi. Save the final plot as the named object **p3a**, and display it.

Exercise C2: Consider applying the Bootstrap method to obtain an approximate 95% confidence interval for the unknown standard deviation of the population, σ. Detail the changes that would need to be made to apply the Bootstrap in this case. (Note, you are not required to complete the code, though you might be interested to try it anyway!) Also explain why an alternative approach, other than the available CLT-based method, is required to obtain a confidence interval for σ.