Module 3 R Practice

Angel Waters

2022-06-14

## Load and clean up the files

The packages needed for this analysis:

library(tidyverse)  
library(ggplot2)  
library(Hmisc)  
library(skimr)  
library(psych)  
library(pastecs)  
library(doBy)  
library(ggpubr)

First step is to collect the data and clean up the data. We are assuming this data is the population of a particular group.

#Reading in the table for this script to analyze  
lung <- read\_csv("LungCapDataCSV.csv")

## Rows: 725 Columns: 6  
## -- Column specification --------------------------------------------------------  
## Delimiter: ","  
## chr (3): Smoke, Gender, Caesarean  
## dbl (3): LungCap, Age, Height  
##   
## i Use `spec()` to retrieve the full column specification for this data.  
## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

lung <- mutate(lung, Smoke = as.logical(ifelse(Smoke=="no", FALSE, TRUE)),  
 Caesarean = as.logical(ifelse(Caesarean=="no", FALSE, TRUE)),  
 Gender = ifelse(Gender=="male", "Male", "Female"))  
names(lung)

## [1] "LungCap" "Age" "Height" "Smoke" "Gender" "Caesarean"

str(lung)

## tibble [725 x 6] (S3: tbl\_df/tbl/data.frame)  
## $ LungCap : num [1:725] 6.47 10.12 9.55 11.12 4.8 ...  
## $ Age : num [1:725] 6 18 16 14 5 11 8 11 15 11 ...  
## $ Height : num [1:725] 62.1 74.7 69.7 71 56.9 58.7 63.3 70.4 70.5 59.2 ...  
## $ Smoke : logi [1:725] FALSE TRUE FALSE FALSE FALSE FALSE ...  
## $ Gender : chr [1:725] "Male" "Female" "Female" "Male" ...  
## $ Caesarean: logi [1:725] FALSE FALSE TRUE FALSE FALSE FALSE ...

view(lung)

## Creating a sample set

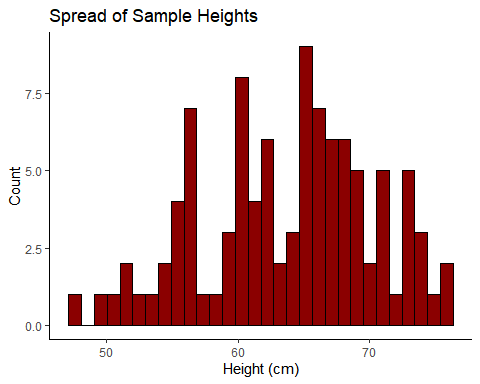
Creating a random sample from the dataset to have 100 samples of the population. The seed was set to an integer that would be kept constant during the analysis, and therefore only randomize the sample once for consistent analysis between sessions.

set.seed(23)  
sample1 <- sample(nrow(lung), 100)  
sample1\_set <- lung[sample1,]  
view(sample1\_set)

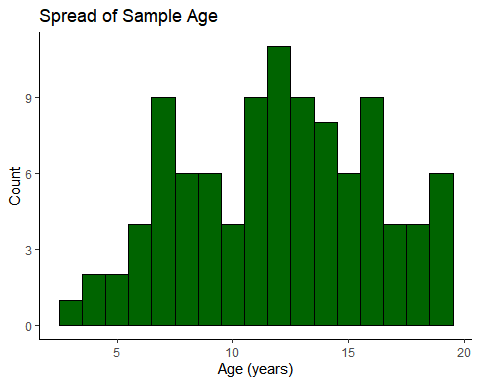
The sample n is over 30, however visualizing the spread to ensure the sample is still relatively normal.

#viewing the spread of the continuous variables  
sample1\_height <- ggplot(sample1\_set)+  
 geom\_histogram(mapping=aes(Height), fill="dark red", color="black")+  
 theme\_classic()+  
 labs(title="Spread of Sample Heights", y="Count", x="Height (cm)")  
sample1\_height

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

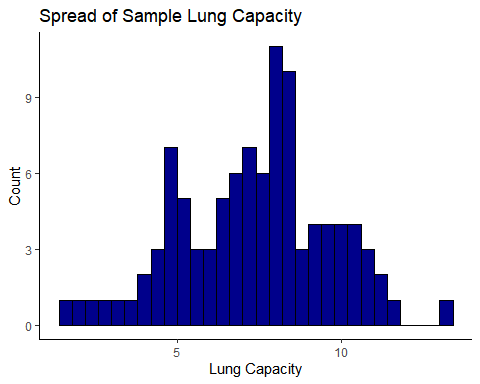


sample1\_age <- ggplot(sample1\_set)+  
 geom\_histogram(mapping=aes(Age), fill="dark green", color="black", binwidth = 1)+  
 theme\_classic()+  
 labs(title="Spread of Sample Age", y="Count", x="Age (years)")  
sample1\_age

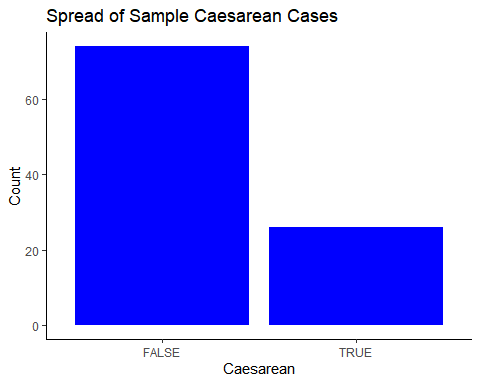


sample1\_lc <- ggplot(sample1\_set)+  
 geom\_histogram(mapping=aes(LungCap), fill="dark blue", color="black")+  
 theme\_classic()+  
 labs(title="Spread of Sample Lung Capacity", y="Count", x="Lung Capacity")  
sample1\_lc

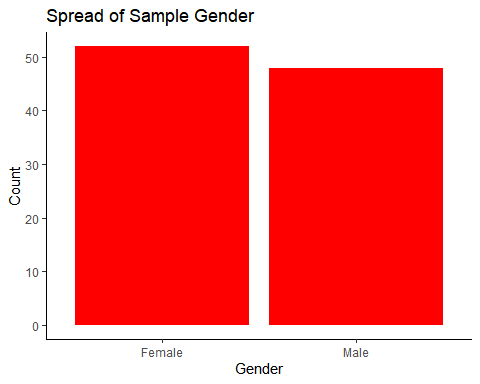
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



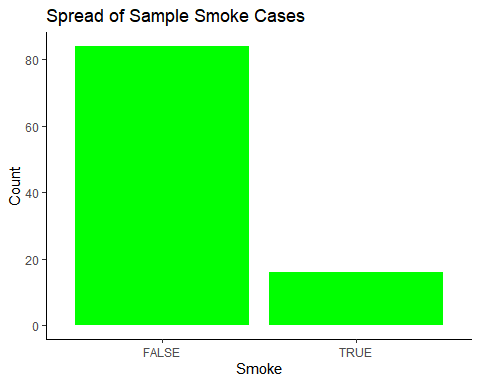
#viewing categorical variables  
sample1\_caes <- ggplot(sample1\_set)+  
 geom\_bar(mapping=aes(Caesarean), fill="blue")+  
 theme\_classic()+  
 labs(title="Spread of Sample Caesarean Cases", y="Count")  
sample1\_caes



sample1\_gender <- ggplot(sample1\_set)+  
 geom\_bar(mapping=aes(Gender), fill="red")+  
 theme\_classic()+  
 labs(title="Spread of Sample Gender", y="Count")  
sample1\_gender



sample1\_smoke <- ggplot(sample1\_set)+  
 geom\_bar(mapping=aes(Smoke), fill="green")+  
 theme\_classic()+  
 labs(title="Spread of Sample Smoke Cases", y="Count")  
sample1\_smoke



Making a function to capture generic sample statistics specific for t test calculations.

sumstats\_ttest <- function(x){  
 xbar <- mean(x)  
 sd <- sd(x)  
 n <- length(x)  
 df <- n-1  
 return(c(xbar=xbar, std.dev=sd, n=n, df=df))  
}

Using the function on the sample set.

LC\_sampstats <- sumstats\_ttest(sample1\_set$LungCap)  
LC\_sampstats

## xbar std.dev n df   
## 7.346250 2.295668 100.000000 99.000000

## t Test

A t Test will be used to evaluate the claim that the population has a mean lung capacity of 7.86 (calculated below).

mu <- mean(lung$LungCap)  
mu

## [1] 7.863148

For this analysis it will be tested using a 95% confidence level.

alpha <- 0.05

Step 1: State the Null and Alternative Hypothesis and state the claim.

null <- "Population mean is equal to 7.86"  
alt <- "Population mean is not equal to 7.86"  
claim <- "The mean lung capacity is 7.86"

Step 2: Compute the sample critical value. Note: this is a two sided test because the alternative is a do not equal.

CV <- qt(p=alpha, df=LC\_sampstats[4], lower.tail=TRUE)  
CV

## [1] -1.660391

CV2 <- qt(p=alpha, df=LC\_sampstats[4], lower.tail=FALSE)  
CV2

## [1] 1.660391

Step 3: Compute the test value.

samplet.test <- t.test(sample1\_set$LungCap, mu=mean(lung$LungCap), alternative = "two.sided")  
samplet.test

##   
## One Sample t-test  
##   
## data: sample1\_set$LungCap  
## t = -2.2516, df = 99, p-value = 0.02656  
## alternative hypothesis: true mean is not equal to 7.863148  
## 95 percent confidence interval:  
## 6.89074 7.80176  
## sample estimates:  
## mean of x   
## 7.34625

Step 4: Make a decision around the hypothesis.

conclusion <- if(CV2>abs(samplet.test$statistic)){  
 ("Do not reject the null hypothesis")  
 } else {"Reject Null hypothesis"  
}  
conclusion

## [1] "Reject Null hypothesis"

Step 5: Summarize the results.

summary <- if(conclusion=="Reject Null hypothesis"){  
 "There is not enough evidence to support the claim"  
} else {  
 "There is sufficient evidence to support the claim"  
}  
summary

## [1] "There is not enough evidence to support the claim"

claim

## [1] "The mean lung capacity is 7.86"

The sample set did not have the data to conclude the population mean. The claim was that with 95% confidence, the population lung capacity average is 7.86.

## Proportion test

The same sample was used for this evaluation.

The claim is that the number of patients born Caesarean is greater than 22.6%. Using a 95% confidence level, the proportion for sample set was calculated for patients born via caesarean.

#population proportion  
csec <- lung %>% filter(Caesarean==TRUE) %>% nrow()  
total <- nrow(lung)  
p <- csec/total  
pt

## function (q, df, ncp, lower.tail = TRUE, log.p = FALSE)   
## {  
## if (missing(ncp))   
## .Call(C\_pt, q, df, lower.tail, log.p)  
## else .Call(C\_pnt, q, df, ncp, lower.tail, log.p)  
## }  
## <bytecode: 0x000000002bb29fb8>  
## <environment: namespace:stats>

q <- 1-p  
  
#sample proportion  
csec\_samp <- sample1\_set%>% filter(Caesarean==TRUE) %>% nrow()  
n <- nrow(sample1\_set)  
phat <- csec\_samp/n

Quick check to see if np and nq are greater than 5 to continue.

np <- total\*p  
np

## [1] 164

nq <- total\*q  
nq

## [1] 561

Step 1: State the Null and Alternative Hypothesis and state the claim.

null <- "Population proportion is equal to 22.6%"  
alt <- "Population proportion is greater than 22.6%"  
claim <- "The proportion of people born caesarean is greater than 22.6%"

Step 2: Compute the sample critical value.

CV <- qnorm(p, lower.tail = FALSE)  
CV

## [1] 0.751397

Step 3: Compute the test value.

z <- (phat - p)/ sqrt((p\*q)/n)

Step 4: Make a decision around the hypothesis.

conclusion <- if(CV>z){  
 ("Do not reject the null hypothesis")  
} else {"Reject Null hypothesis"  
}  
conclusion

## [1] "Reject Null hypothesis"

Step 5: Summarize the results.

summary <- if(conclusion=="Do not reject the null hypothesis"){  
 "There is not enough evidence to support the claim"  
} else {  
 "There is sufficient evidence to support the claim"  
}  
summary

## [1] "There is sufficient evidence to support the claim"

claim

## [1] "The proportion of people born caesarean is greater than 22.6%"

There is not enough evidence to support the null hypothesis. The claim that the population proportion of patients born via caesareans is greater than 22.6% still stands and is supported by the sample data set.