progress report

Rui Gao

11/30/2021

```r
knitr::opts_chunk$set(echo = TRUE)
set.seed(2021)
yrbss_2007 <- readRDS("yrbss_2007.rds")
yrbss_2017 <- readRDS("yrbss_2017.rds")

library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(ggplot2)
library(plyr)
```

```
## Warning: package 'plyr' was built under R version 4.1.2
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

```
##
## Attaching package: 'plyr'
```

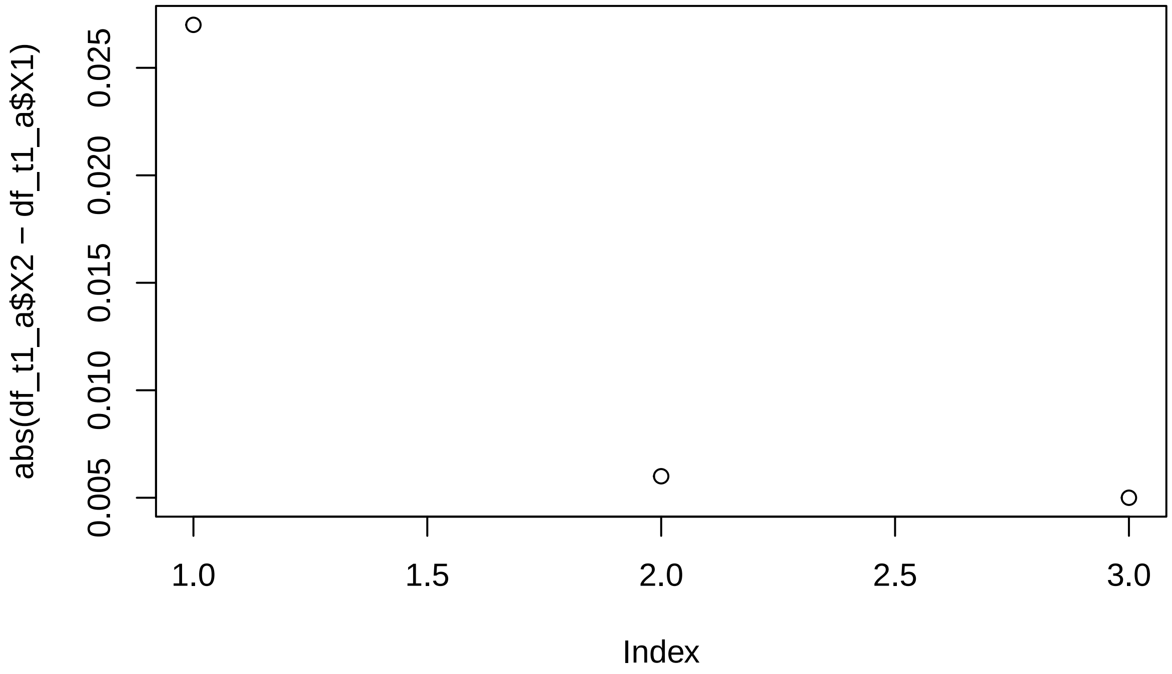
```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
## The following object is masked from ’package:purrr’:
##
##     compact
```

#Task 1 ## a. Using repeated samples of size n = 10, 100, and 1000 from the bmi variable, describe the sampling distribution of the sample mean of BMI in 2017. Include at least one plot to help describe your results. Report the means and standard deviations of the sampling distributions, and describe how they change with increasing sample size.

```r
#
sample_size_n = c(10,100,1000)
set.seed(2021)
n_sim <-1000
pop_sd <- sd(yrbss_2017$bmi)

newdf_generator <- function(n) {
  index <- sample(1:length(yrbss_2017$year),n)
  yrbss_2017[index,]
}

get_means <- function(n, n_sim) {
  replicate(n_sim, mean(newdf_generator(n)$bmi))
}



ns <- c(10, 100, 1000)
means <- lapply(ns, get_means, n_sim = n_sim)

spread_sampdist <- sapply(means, sd)

# true_se
true_se <- pop_sd/sqrt(ns)



rbind(round(spread_sampdist, 3),round(true_se, 3))
```

```
##       [,1]  [,2]  [,3]
## [1,] 1.618 0.526 0.160
## [2,] 1.645 0.520 0.165
```

```r
#plot
df_t1_a <- data.frame(cbind(round(spread_sampdist, 3),round(true_se, 3)))
plot(abs(df_t1_a$X2-df_t1_a$X1))
```



As the size of sample increase, the difference decreases. However, they basically the same value, as you can see in the plot.

b. Repeat the simulation in part (a), but this time use the 25th percentile as the sample statistic. In R, quantile(x, prob = 0.25) will give you the 25th percentile of the values in x.

```
#

get_quan <- function(n, n_sim) {
  replicate(n_sim, quantile(newdf_generator(n)$bmi,prob=0.25))
}



ns <- c(10, 100, 1000)
quans <- lapply(ns, get_quan, n_sim = n_sim)
cat("n=10",mean(quans[[1]]),"n=100",mean(quans[[2]]),"n=1000",mean(quans[[3]]))
```

```
## n=10 20.50207 n=100 20.1464 n=1000 20.10989
```

When, n = 10, the value I select for 25th percentile is 20.5,

When, n = 100, the value I select for 25th percentile is 20.15,

When, n = 1000, the value I select for 25th percentile is 20.11,

As the size of sample increase, the difference decreases.

c. Repeat the simulation in part (a), but this time use the sample minimum as the sample statistic.

```r
get_min <- function(n, n_sim) {
  replicate(n_sim, min(newdf_generator(n)$bmi))
}




ns <- c(10, 100, 1000)
mins <- lapply(ns, get_min, n_sim = n_sim)
cat("n=10",mean(mins[[1]]),"n=100",mean(mins[[2]]),"n=1000",mean(mins[[3]]))
```

```
## n=10 17.92916 n=100 15.6259 n=1000 13.67283
```

When, n = 10, the value I select for minimum percentile is 17.93,

When, n = 100, the value I select for minimum percentile is 15.63,

When, n = 1000, the value I select for minimum percentile is 13.68,

As the size of sample increase, the difference decreases.

d.Describe the sampling distribution of the difference in the sample median BMI between 2017 and 2007, by using repeated samples of size (n1 = 5,n2 = 5), (n1 = 10,n2 = 10) and (n1 = 100,n2 = 100). Report the means and standard deviations of the sampling distributions, and describe how they change with the different sample sizes.

```r
n=5
set.seed(2021)

index2017 <- sample(1:length(yrbss_2017$year),n)
df2017_5<-yrbss_2017[index2017,]$bmi
index2007 <- sample(1:length(yrbss_2007$year),n)
df2007_5<-yrbss_2007[index2007,]$bmi
global_med <- median(rbind(df2017_5,df2007_5))

x <- c(sum(df2007_5<global_med),sum(df2017_5<global_med))
n<-c(5,5)
prop.test(x, n, correct=F)
```

```
## Warning in prop.test(x, n, correct = F): Chi-squared approximation may be
## incorrect
```

```
##
##  2-sample test for equality of proportions without continuity
##  correction
##
## data:  x out of n
## X-squared = 3.6, df = 1, p-value = 0.05778
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -1.000000 -0.104164
## sample estimates:
## prop 1 prop 2
##    0.2    0.8
```

```
cat("2007mean:",mean(df2007_5),"2017mean:",
    mean(df2017_5),"2007sd",sd(df2007_5),"2017sd",sd(df2017_5))
```

```
## 2007mean: 25.94962 2017mean: 20.63696 2007sd 6.999025 2017sd 3.895473
```

H_0:median_2007 = median_2017 vs H_a:median_2007 != median_2017

When n = 5,2007mean: 25.94962 2017mean: 20.63696 2007sd 6.999025 2017sd 3.895473

###With 95% confidence to say there has difference between their medians

```
n=10
set.seed(2021)

index2017 <- sample(1:length(yrbss_2017$year),n)
df2017_10<-yrbss_2017[index2017,]$bmi
index2007 <- sample(1:length(yrbss_2007$year),n)
df2007_10<-yrbss_2007[index2007,]$bmi
global_med <- median(rbind(df2017_10,df2007_10))

x <- c(sum(df2007_10<global_med),sum(df2017_10<global_med))
n<-c(10,10)
prop.test(x, n, correct=F)
```

```
##
##  2-sample test for equality of proportions without continuity
##  correction
##
## data:  x out of n
## X-squared = 3.2, df = 1, p-value = 0.07364
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.801673089  0.001673089
## sample estimates:
## prop 1 prop 2
##    0.3    0.7
```

```
cat("2007mean:",mean(df2007_10),"2017mean:",
    mean(df2017_10),"2007sd",sd(df2007_10),"2017sd",sd(df2017_10))
```

```
## 2007mean: 27.18287 2017mean: 20.94665 2007sd 8.216503 2017sd 3.351787
```

H_0:median_2007 = median_2017 vs H_a:median_2007 != median_2017

###When n = 10,2007mean: 27.18287 2017mean: 20.94665 2007sd 8.216503 2017sd 3.351787 ###With 95% confidence to say there has difference between their medians

```
n=100
set.seed(2021)

index2017 <- sample(1:length(yrbss_2017$year),n)
df2017_100<-yrbss_2017[index2017,]$bmi
index2007 <- sample(1:length(yrbss_2007$year),n)
df2007_100<-yrbss_2007[index2007,]$bmi
global_med <- median(rbind(df2017_100,df2007_100))

x <- c(sum(df2007_100<global_med),sum(df2017_100<global_med))
n<-c(100,100)
prop.test(x, n, correct=F)
```

```
##
##  2-sample test for equality of proportions without continuity
##  correction
##
## data:  x out of n
## X-squared = 0.72029, df = 1, p-value = 0.396
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.19831292  0.07831292
## sample estimates:
## prop 1 prop 2
##   0.46   0.52
```

```
cat("2007mean:",mean(df2007_100),"2017mean:",
    mean(df2017_100),"2007sd",sd(df2007_100),"2017sd",sd(df2017_100))
```

```
## 2007mean: 24.33437 2017mean: 22.96323 2007sd 6.463946 2017sd 5.212375
```

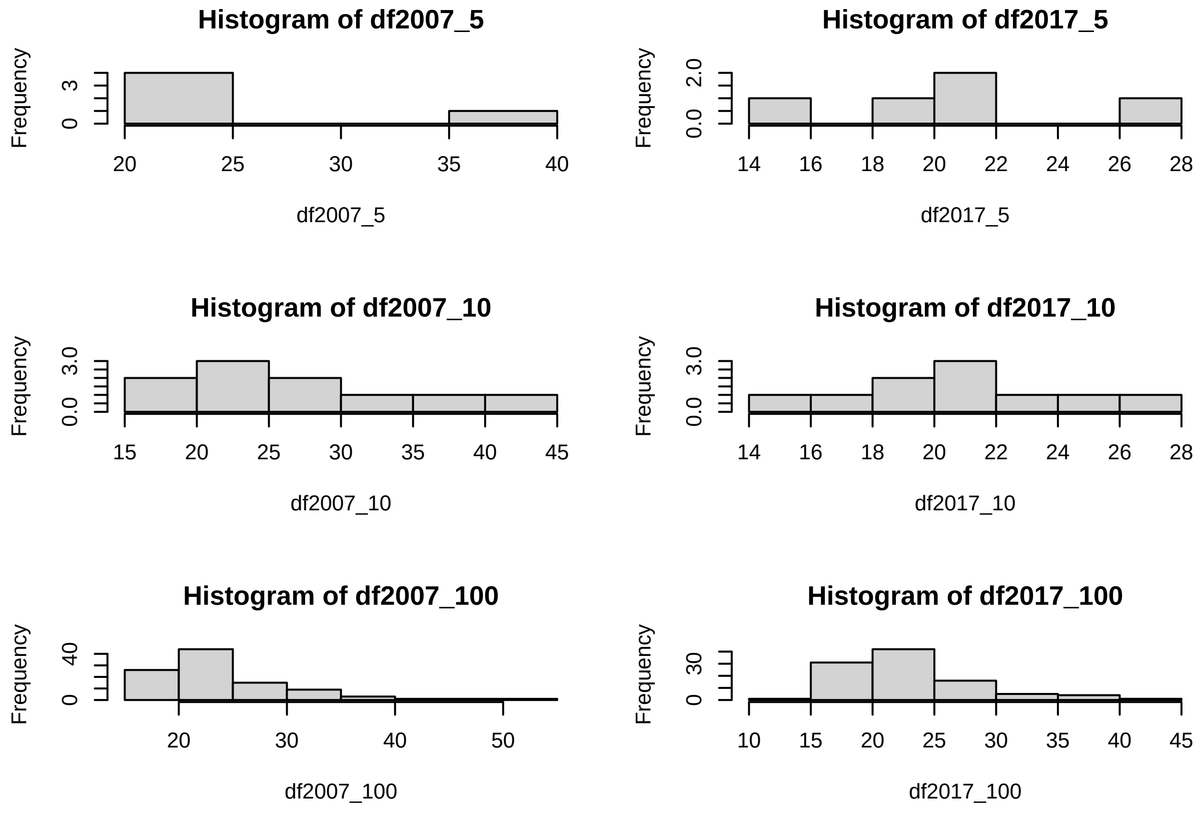
H_0:median_2007 = median_2017 vs H_a:median_2007 != median_2017

###When n = 100,2007mean: 24.33437 2017mean: 22.96323 2007sd 6.463946 2017sd 5.212375 ###With 95% confidence to say there has difference between their medians

###Conclusion: As the number of size increases, the p-value gets bigger. The sample means and standard deviations are unstable. ###But we know, as the size number of sample increases, the sample's parameters are more and more closer to the population's. Hence, this results means the real situation had been covered when the n was small, when n becomes bigger enough, the problem had been exposured.

e. Summarize your results. Make sure you comment on the center, spread and shape for the sampling distribution of each statistic as the sample size increases. You should also contrast the behaviour of the sample statistics.Here are some statements, that are sometimes true, you might find useful to reflect on in your summary: 1) The sampling distribution of an estimate is always centered around the true parameter value. 2) The precision of an estimate (the standard deviation of the sampling distibution) decreases according to the square root of the sample size. 3)As the sample size increases, the sampling distribution of an estimate always approaches a Normal distribution.

```r
par(mfrow=c(3,2))
hist(df2007_5)
hist(df2017_5)
hist(df2007_10)
hist(df2017_10)
hist(df2007_100)
hist(df2017_100)
```



As the number of size increases, the shape of the distribution become more and more like the standard distribution and the percision of an estmate gets higher. But the population in this task has left skewing.

Comparing to the means and SDs', we can tell the distribution is always centered around the true parameter value.

#Task 2 ## 1. How has the BMI of high-school students changed between 2007 and 2017? Are high-schoolers getting more overweight?

```
#H_0: mu_2007 - mu_2017  = 0
#H_a: mu_2007 - mu_2017  != 0


t.test(x = yrbss_2007$bmi, y = yrbss_2017$bmi)
```

```
##
##  Welch Two Sample t-test
##
## data:  yrbss_2007$bmi and yrbss_2017$bmi
## t = 2.4775, df = 26125, p-value = 0.01324
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.03249091 0.27864773
## sample estimates:
## mean of x mean of y
##  23.77572  23.62015
```

###Hence, 0.03< mu_2007 - mu_2017 <0.28. ###With 95% confidence the mean weight of 2007 students is heavier than those students in 2017.However, since the dietary habit hasn't changed too much during the decade and the food suppliers/brands are almost the same in the USA, we cannot say the data is totally independent.

2. In 2017, are 12th graders more or less likely than 9th graders to be "physically active at least 60 minutes per day on 5 or more days"?

```
#H_0: mu_2007 != mu_2017
#H_a: mu_2007 = mu_2017

y2017_9 <- yrbss_2017[yrbss_2017$grade=="9th",]
y2017_12 <- yrbss_2017[yrbss_2017$grade=="12th",]


#total912<-rbind(y2017_9,y2017_12)
#total912$qn79 <- revalue(as.factor(total912$qn79) ,c("FALSE"=0))
#total912$qn79 <- revalue(as.factor(total912$qn79) ,c("TRUE"=1))
#final_total912 <- total912[complete.cases(total912),]

#delete rows with NA for 9th
y2017_9$qn79 <- revalue(as.factor(y2017_9$qn79) ,c("FALSE"=0))
y2017_9$qn79 <- revalue(as.factor(y2017_9$qn79) ,c("TRUE"=1))
```

```
med_y2017_9 <- y2017_9[,-c(1:10)]

final_y2017_9 <- med_y2017_9[complete.cases(med_y2017_9),]
count(final_y2017_9$qn79)
```

```
##   x freq
## 1 0 1604
## 2 1 1680
```

```
#delete rows with NA for 12th
y2017_12$qn79 <- revalue(as.factor(y2017_12$qn79) ,c("FALSE"=0))
y2017_12$qn79 <- revalue(as.factor(y2017_12$qn79) ,c("TRUE"=1))

med_y2017_12 <- y2017_12[,-c(1:10)]

final_y2017_12 <- med_y2017_12[complete.cases(med_y2017_12),]
count(final_y2017_12$qn79)
```

```
##   x freq
## 1 0 1860
## 2 1 1149
```

```
X <- c(1680,1149)
n <- c((1680+1604),(1860+1149))
prop.test(X,n,correct=FALSE)
```

```
##
##  2-sample test for equality of proportions without continuity
##  correction
##
## data:  X out of n
## X-squared = 106.77, df = 1, p-value < 2.2e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.1053524 0.1540812
## sample estimates:
##    prop 1    prop 2
## 0.5115713 0.3818544
```

The data result in a 95% confidence interval for the distance between 9th and 12th grades' students' exercising time over 69 minutes per day on 5 or more days are (0.105,0.15) hours. there is convincing evidence to say that. As the leaving time decrease, the 12th graders are facing the more and more enrollment pressure, the result meet with our intuition for the students.

3. How much sleep do highschoolers get?

```
#sleep hours
s2007 <- yrbss_2007[,-c(1:9,11)]
```

```
s2017 <- yrbss_2017[,-c(1:9,11)]

med_sleep<-rbind(s2007,s2017)
total_sleep <- med_sleep[complete.cases(med_sleep),]

#transformation
total_sleep$q88 <- revalue(as.factor(total_sleep$q88) ,c("4 or less hours"=4))
total_sleep$q88 <- revalue(as.factor(total_sleep$q88) ,c("5 hours"=5))
total_sleep$q88 <- revalue(as.factor(total_sleep$q88) ,c("6 hours"=6))
total_sleep$q88 <- revalue(as.factor(total_sleep$q88) ,c("7 hours"=7))
total_sleep$q88 <- revalue(as.factor(total_sleep$q88) ,c("8 hours"=8))
total_sleep$q88 <- revalue(as.factor(total_sleep$q88) ,c("9 hours"=9))
total_sleep$q88 <- revalue(as.factor(total_sleep$q88) ,c("10 or more hours"=10))

# Assuming the average sleep hours is the recommanded number, 8 hours
Z <- (mean(as.numeric(total_sleep$q88)) - 8) /
  (sd(as.numeric(total_sleep$q88))/sqrt(length(as.numeric(total_sleep$q88))))
P <- 2 * pnorm(abs(Z), mean = 0, sd = 1, lower.tail = FALSE)
P <0.05
```

```
## [1] TRUE
```

```
list(Z=Z,P=P)
```

```
## $Z
## [1] -475.8039
##
## $P
## [1] 0
```

```
mean(as.numeric(total_sleep$q88))
```

```
## [1] 3.678194
```

```
mean(as.numeric(total_sleep$q88)) -
  qnorm(0.975) * sd(as.numeric(total_sleep$q88))/sqrt(length(as.numeric(total_sleep$q88)))
```

```
## [1] 3.660391
```

```
mean(as.numeric(total_sleep$q88)) +
  qnorm(0.975) * sd(as.numeric(total_sleep$q88))/sqrt(length(as.numeric(total_sleep$q88)))
```

```
## [1] 3.695997
```

###I don't know why does the mean value less than 4, I deleted all NA and all values i the table are larger than 4. ###Anyway, the data is discrete and unaccurate, so I set it to integer. The schools have basically the same schedule for students, it's hard to say the data is independent. ###However, since p is small, we can tell the sleep hours for highschoolers are not enough, they normally get 3.66 to 3.7 hours for sleeping. That's too short for a human.