Bootstrapping

Leobardo Enriquez

2023-09-20

Bootstrapping method to estimate voters' preferences in US presidential election 2020.

Data

The data set used in this analysis includes poll results. There are in total 51 states, and every state has different polls indicating the percentage of supporters for the Democrat party and Republican party (GOP).

```
df<-read.csv('pres_polls.csv')
head(df)
```

```
##     Day   State Region EV Dem GOP   Date                  Pollster
## 1 275.5 Alabama  South  9  37  57  3-Oct Auburn U. at Montgomery-4
## 2 210.5 Alabama  South  9  36  58  2-Aug        Morning Consult-10
## 3 187.5 Alabama  South  9  41  55  9-Jul Auburn U. at Montgomery-8
## 4  36.0 Alabama  South  9  38  58  6-Feb             Mason-Dixon-3
## 5 273.5  Alaska   West  3  46  50  4-Oct Alaska Survey Research-10
## 6 205.5  Alaska   West  3  44  50 24-Jul                     PPP-2
```

We can see from the above table that there are 8 variables in this dataset. The Day variable represents the day of the year when the survey is done. The variable state represents the state in which the survey is done. Region variable describes the 4 regions in which each state falls. EV stands for electoral votes for that state. Dem and GOP represent the percentage of people who gave favorable responses for the Democratic party and the Republican party. The date column represents the date at which the survey is done, and finally, the Pollster is the name of the poll.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
poll=unique(pull(df,Pollster))
str(poll)
```

```
##  chr [1:233] "Auburn U. at Montgomery-4" "Morning Consult-10" ...
```

Let's look into the Pollster variable to see how many pool results we have in this sample. We can use the 'str' function to see how many unique polls are in the data set. From the output, there are 233 unique rows which tell us that the given sample has in total 233 polls. We will use the bootstrapping to generate 10,000 samples

which will have 233 pools' results in each of them. For 10,000 samples we will get 10,000 sample means which will make out bootstrap sampling distribution. As bootstrapping is a resampling technique with replacement, some samples may have the same pools several times.

Estimating Voters' Preference.

In this section, we will try to create the bootstrap distribution for the whole dataset regardless of region or state to get an estimate of the voters' preferences across the nation. After that, we will build a bootstrap distribution for each of the 4 regions of the United States of America. We will calculate the confidence interval of the mean from those bootstrap distribution to understand the level of preferences.

Confidence Intervals from the bootstrap distributions:

```
library(boot)
bootmean=function(x,i){mean(x[i])}
prefer_country=function(data){
  boot.object=boot(data,bootmean,R=10000)
  boot.ci(boot.object,conf = 0.95,type = 'bca')
}
Dem=round(prefer_country(df$Dem)$bca[,c(4,5)],4)
GOP=round(prefer_country(df$GOP)$bca[,c(4,5)],4)
c('Democratc party:',Dem)
```

```
##
## "Democratc party:"          "48.0083"          "48.8597"
```

```
c('Republican party:',GOP)
```

```
##
## "Republican party:"           "43.3045"           "44.2025"
```

In R, to calculate the bootstrap distribution:

1.- We first load the 'boot' library. We have to install the library before loading it if we are running it for the first time.

2.- To develop the bootstrap distribution, we first need to develop a function that tells R what statistic we want to calculate when we will generate multiple samples from our given sample. In our case, we want to calculate the average from every new sample we generate from the given sample. Therefore, we have created a function named 'bootmean' in the second line which mainly takes a vector and calculates the mean out of it.

2.- To develop the bootstrap distribution, we first need to develop a function that tells R what statistic we want to calculate when we will generate multiple samples from our given sample. In our case, we want to calculate the average from every new sample we generate from the given sample. Therefore, we have created a function named 'bootmean' in the second line which mainly takes a vector and calculates the mean out of it. The boot function is going to take the data that we will pass and calculates 10,000 new samples out of it as we mentioned R=10,000.

3.- Finally, it will calculate the statistic from each of the samples and store the results in the 'boot.object'. That is why in the second argument in the boot function we have passed the 'bootmean' function that we created in the second line of our code. There are different things stored in the boot.object. If we want to see what are the things stored in it we can use str(boot.object) to see the structure of the object. However, in our case, we are interested to get the confidence interval of the mean for the two parties. Therefore, in the next line, we have used the boot.ci function to calculate the interval of mean at a 95% level of confidence. There are several types of confidence intervals saved in the boot. object, but we want the bias-corrected and accelerated ('BCA') bootstrap interval. Therefore, we mentioned type='bca'.

4.- Finally, we have called the prefer_country function for both Democratic and Republican parties' data and saved the results in the 'Dem' and 'GOP' variable. The last two lines have printed out the results.

So, from our bootstrap analysis, we can tell at a 95% level of confidence, between 48.01% and 48.86% of Americans are preferring the Democratic candidate, and between 43.31% and 44.20% of Americans are preferring the Republican candidate in this presidential election.

Regional level

Let's look into the regional level to understand what are the confidence intervals for both parties. we will be using the same techniques but we need to calculate the bootstrap samples at every region rather than for the whole nation. Below is the R code for that.

```
lower=c()
upper=c()
region=c()
a=unique(pull(df,Region))
prefer_region=function(data){
  for (i in a){
    data_Dem=data[df$Region==i]
    boot.Dem=boot(data_Dem,bootmean,R=10000)
    p=boot.ci(boot.Dem,conf = 0.95)
    lower=c(lower,p$bca[,c(4)])
    upper=c(upper,p$bca[,c(5)])
    region=c(region,i)
  }
  preference=data.frame(region,lower,upper)
  preference}
DEM=prefer_region(df$Dem)%>%rename(Dem_lower=lower,Dem_upper=upper)
```

```
## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals

## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals

## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals

## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals
```

```
GOP=prefer_region(df$GOP)%>%rename(GOP_lower=lower,GOP_upper=upper)
```

```
## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals

## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals

## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals

## Warning in boot.ci(boot.Dem, conf = 0.95): bootstrap variances needed for
## studentized intervals
```

```
inner_join(DEM,GOP,by='region')
```

```
##       region Dem_lower Dem_upper GOP_lower GOP_upper
```

```
## 1      South  46.18410  47.56904  45.41841  46.82427
## 2       West  48.17213  50.82787  40.55738  43.13115
## 3 North East  51.07692  52.82051  39.18803  41.08547
## 4   Mid West  47.06771  48.14583  43.56250  44.75521
```

1.- We have started with establishing 3 empty vectors 'lower', 'upper', and 'region'. We will save the bootstrapping outputs in these vectors to develop a data frame.

2.- Next, we have saved the names of the distinct regions in vector 'a' which has four values South, North East, West, Mid West. We will use this vector in our for loop to filter our data for specific regions and then use the filtered data to develop bootstrap samples only for that region.

3.- Now, we have written another function named 'prefer_region' which takes a column from our df data frame as an input. Inside the function, we have written a loop that mainly takes the assigned column and filters it for a specific region. The loop is going to read the name of the regions from the vector 'a' and perform the bootstrap calculation in a similar fashion as we have done for the whole nation. Every time, we have calculated the intervals, we have saved the lower limit in the 'lower' vector and upper limit in the 'upper' vector. In addition, we have saved the region name in the 'region' vector. Once the calculations are done for all the regions, the for loop ends.

4.- Now, we have taken the three vectors 'lower', 'upper', and 'region' and created a data frame named 'preference'. This is the end of our function.

5.- In the last three lines, we have mainly called the functions for both democratic and republican parties and saved the results in one data frame which has been shown in the output.

If we look into the output, we can see that at a 95% level of confidence, the intervals of mean percentages are higher for the Democratic party in the West, North East, and Mid West region. However, in the south region, the intervals overlap each other. Between 46.19% and 47.56% of Americans in the south region are preferring the Democratic candidate, and between 45.41% and 46.81% of Americans in the south region are preferring the Republican candidate.

Conclusion

From the above discussion, we can see how to implement bootstrapping using the R programming language. We were able to generate 10,000 samples each having 233 observations from a given sample of 233 observations. One important thing here is that we try to find out the intervals from our bootstrapping sampling distribution at the national and regional levels. It doesn't necessarily tell us which party is going to win the election because the study is not done at the state level. The electoral votes vary from state to state. So, this study gives an overall understanding of the general political preferences of voters by applying the bootstrapping method.