Bootstrap

Prabidhik KC

2022-11-24

In this module I will be doing bootstrap

```
library(tidyverse)
```

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

## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.1
## v readr   2.1.2     v forcats 0.5.1

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

```
library(gov50data)
library(infer)
```

```
anes
```

```
## # A tibble: 5,162 x 8
##    state district  pid7 pres_vote sci_therm rural_therm favor_voter_id envir_d~1
##    <chr>    <dbl> <dbl> <chr>         <dbl>       <dbl>          <dbl>     <dbl>
##  1 ID           2     4 Other            70          60              1         0
##  2 VA           2     3 Biden           100          75              0         1
##  3 CO           4     4 Trump            60          90              1         0
##  4 TX           5     3 Biden            85          85              1         1
##  5 WI           6     6 Trump            85          70              1         1
##  6 CA          40     2 Biden            50          50              1         0
##  7 WI           5     2 Biden           100          70              1         0
##  8 OR           4     7 Trump            70          50              0         1
##  9 MA           5     3 Biden            80          70              0         1
## 10 NV           3     1 Biden            85          40              0         1
## # ... with 5,152 more rows, and abbreviated variable name 1: envir_doing_more
```

```
anes %>%
  summarize(sci_mean = mean(sci_therm))
```

```
## # A tibble: 1 x 1
##   sci_mean
##      <dbl>
## 1     80.6
```

```
## bootstrap: sampling from the population by resampling many times the sample itself
## boostrap: sampling from the population by resampling the sample itself many times

boot_1 <- anes %>%
  slice_sample(prop = 1, replace = TRUE)

boot_1 %>%
  summarize(sci_mean1 = mean(sci_therm))
```

```
## # A tibble: 1 x 1
##   sci_mean1
##       <dbl>
## 1      81.2
```

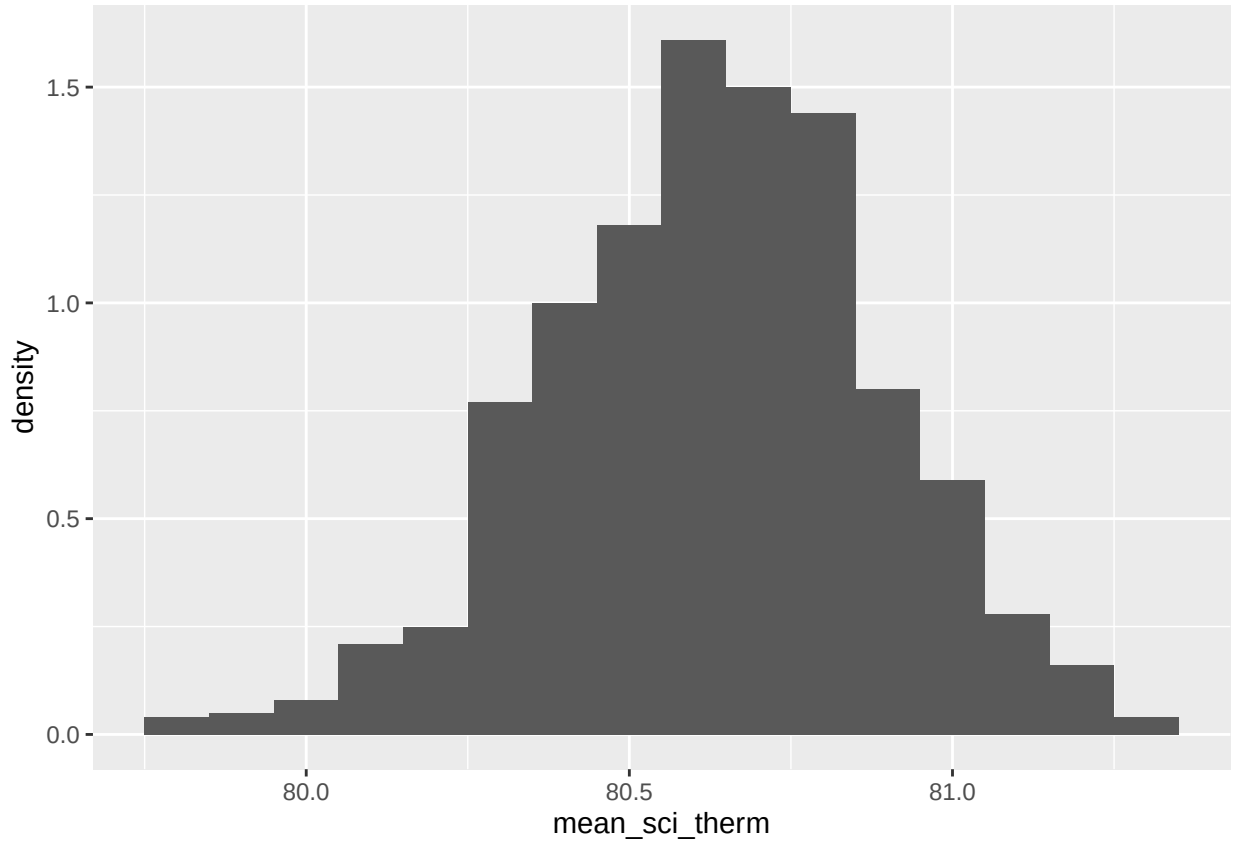
```
bootstrap_dist <- anes %>%
  rep_slice_sample(prop = 1, replace = TRUE, reps = 1000) %>%
  group_by(replicate)  %>%
  summarize(mean_sci_therm = mean(sci_therm))

bootstrap_dist
```

```
## # A tibble: 1,000 x 2
##    replicate mean_sci_therm
##        <int>          <dbl>
##  1         1           80.6
##  2         2           81.0
##  3         3           80.7
##  4         4           80.9
##  5         5           80.4
##  6         6           80.0
##  7         7           80.5
##  8         8           80.6
##  9         9           80.7
## 10        10           80.5
## # ... with 990 more rows
```

```
tail(bootstrap_dist)
```

```
## # A tibble: 6 x 2
##   replicate mean_sci_therm
##       <int>          <dbl>
## 1       995           80.7
## 2       996           80.3
## 3       997           80.4
## 4       998           81.1
## 5       999           80.2
## 6      1000           80.7
```

```
bootstrap_dist %>%
  ggplot(mapping = aes(x = mean_sci_therm)) +
  geom_histogram(mapping = aes(y = ..density..), binwidth = 0.1)
```




```
perc_ci99 <- quantile(bootstrap_dist$mean_sci_therm,
  probs = c(0.005, 0.995))

perc_ci99
```

```
##     0.5%    99.5%
## 79.90679 81.23112
```

```
perc_ci95 <- quantile(bootstrap_dist$mean_sci_therm,
                      probs = c(0.025, 0.975))
perc_ci95
```

```
##     2.5%    97.5%
## 80.08503 81.13233
```

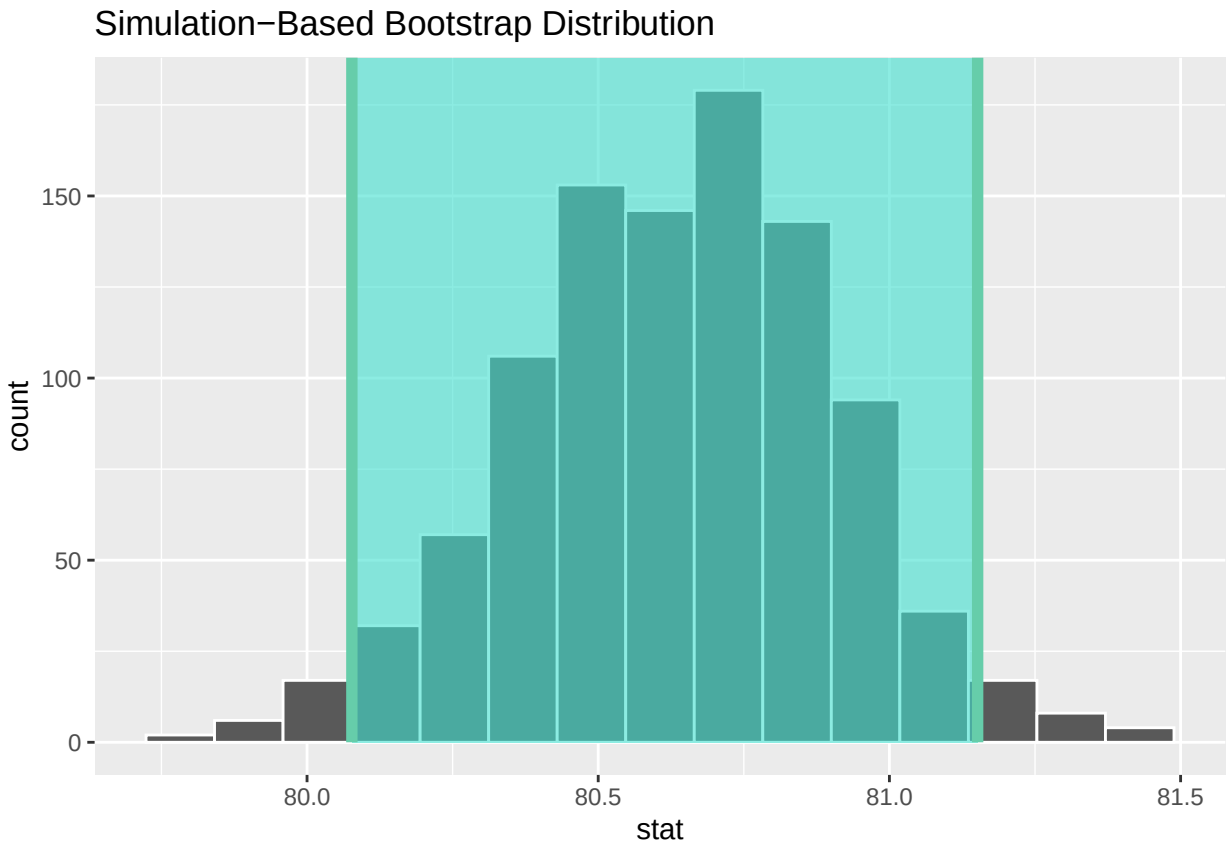
```
boot_dist_infer <- anes %>%
  specify(response = sci_therm) %>%
  generate(reps = 1000, type = "bootstrap") %>%
  calculate(stat = "mean")

boot_dist_infer
```

```
## Response: sci_therm (numeric)
## # A tibble: 1,000 x 2
##    replicate  stat
##        <int> <dbl>
##  1         1  80.5
##  2         2  80.7
##  3         3  80.6
##  4         4  80.7
##  5         5  80.9
##  6         6  80.9
##  7         7  79.9
##  8         8  80.8
##  9         9  80.6
## 10        10  80.6
## # ... with 990 more rows
```

```
perc_ci_95 <- boot_dist_infer %>%
  get_confidence_interval(level = 0.95, type = "percentile")
perc_ci_95
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1     80.1     81.2
```

```
visualize(boot_dist_infer) +
  shade_confidence_interval(endpoints = perc_ci_95)
```


Simulation−Based Bootstrap Distribution
count
150
100
50
0
80.0
80.5
81.0
81.5
stat