Exercises ISLR – Ch.5

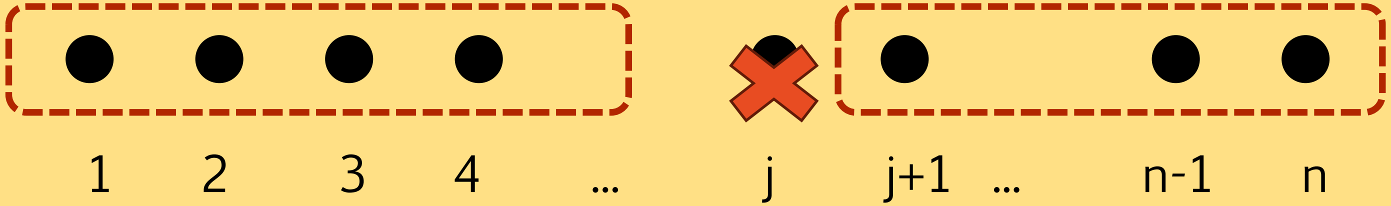
Marcelo Previato Simoes Nº 2367070
15/09/2025

Conceptual

Exercises: 2

Ex.2) Bootstrap Probability (a)

(a) What is the probability that the first bootstrap observation is *not* the jth observation from the original sample? Justify your answer.



$$P\,(1^{st}\text{ not } j) = (n-1)/n = (1-1/n)$$

Ex.2) Bootstrap Probability (b-c)

(b) What is the probability that the second bootstrap observation is *not* the jth observation from the original sample?

Since the bootstrap is run with replacement, the probability is the same as in (a)

P (2nd not j) = (n-1)/n = (1-1/n)

(c) Argue that the probability that the jth observation is *not* in the bootstrap sample is $(1-1/n)^n$.

P(j not in sample) = P(1st not j) * P(2nd not j) * … * P(nth not j)

= (1-1/n)^n

Ex.2) Bootstrap Probability (d-f)

(d) When $n = 5$, what is the probability that the jth observation is in the bootstrap sample?

P (j in sample) = 1- P(j not in sample) = 1 - (1-1/5)^5 ~ <u>0.67232</u>

(e) When $n = 100$, what is the probability that the jth observation is in the bootstrap sample?

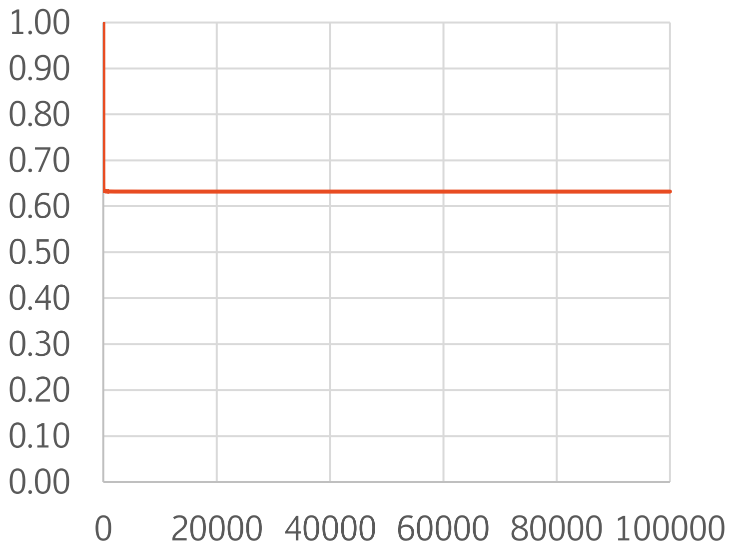
P (j in sample) = 1 - (1-1/100)^100 ~ <u>0.63397</u>

(f) When $n = 10,000$, what is the probability that the jth observation is in the bootstrap sample?

P (j in sample) = 1 - (1-1/10000)^10000 ~ <u>0.63214</u>

Ex.2) Bootstrap Probability (g)

(g) Create a plot that displays, for each integer value of n from 1 to 100,000, the probability that the jth observation is in the bootstrap sample. Comment on what you observe.



P (j in sample) = lim 1 - (1-1/n)^n

= 1 - lim(1+ (-1)/n)^n

= 1 - e^-1 =   1- 1/e = <u>0.632121</u>

The probability of being part of the bootstrap sample rapidly converge to 1-1/e (63.21%), which does not converge to 1.

Ex.2) Bootstrap Probability (h)

(h) We will now investigate numerically the probability that a bootstrap sample of size $n = 100$ contains the jth observation. Here $j = 4$. We first create an array `store` with values that will subsequently be overwritten using the function `np.empty()`. We then repeatedly create bootstrap samples, and each time we record whether or not the fifth observation is contained in the bootstrap sample.

```
rng = np.random.default_rng(10)
store = np.empty(10000)
for i in range(10000):
    store[i] = np.sum(rng.choice(100, replace=True) == 4)
            > 0
np.mean(store)
```

Comment on the results obtained.

N = 10.000 samples

N (j in sample) = 6357

→P (j in sample) = 0.6357

This value is very close to the expected value of large samples 0.6321 (0.5% relative difference).

Applied

Exercise 9 – Boston dataset

Ex.9) Mean of Residence Value (a-b)

(a) Based on this data set, provide an estimate for the population mean of `medv`. Call this estimate $\hat{\mu}$.

```
      medv
 Min.   : 5.00
 1st Qu.:17.02
 Median :21.20
 Mean   :22.53
 3rd Qu.:25.00
 Max.   :50.00
```

```
> print(mu)
[1] 22.53281
```

(b) Provide an estimate of the standard error of $\hat{\mu}$. Interpret this result.

Hint: We can compute the standard error of the sample mean by dividing the sample standard deviation by the square root of the number of observations.

mean(mu) ~ N(mu, sigma2/n) **CLT**

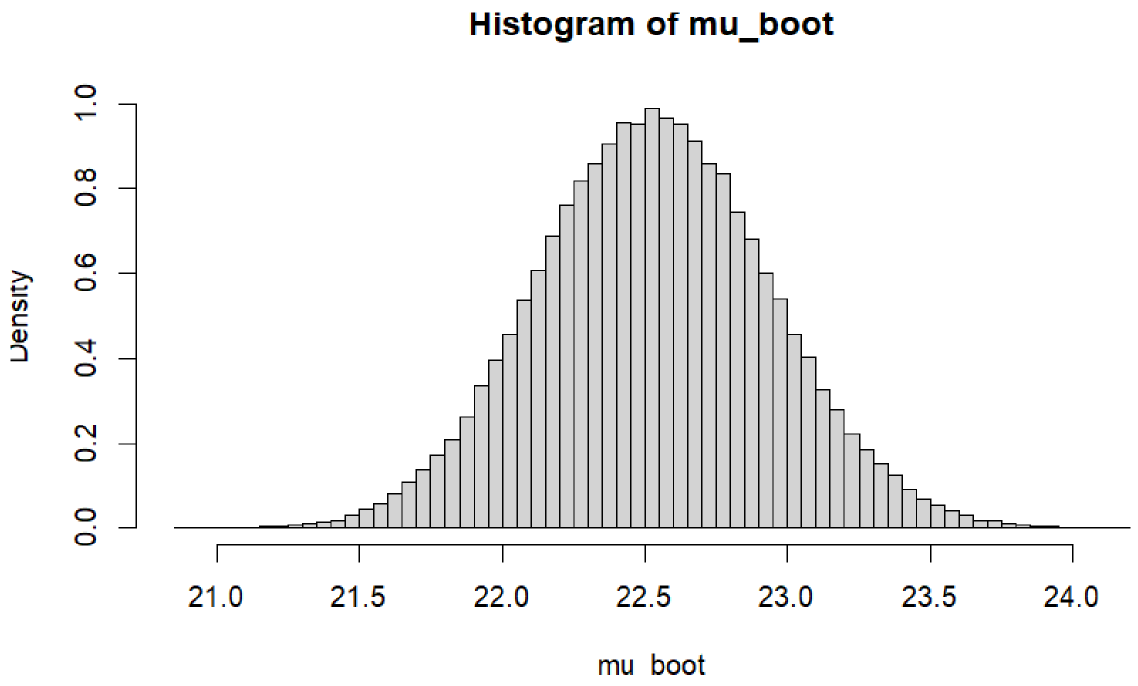
SE = mu_hat/sqrt (s2/n) **approx.**

```
> print(se_mu)
[1] 0.4088611
```

Ex.9) Mean of Residence Value (c)

(c) Now estimate the standard error of $\hat{\mu}$ using the bootstrap. How does this compare to your answer from (b)?

B = 100.000 bootstrap samples n = 506 datapoints



Mean

```
> print(mu_hat_boot)
[1] 22.53182
> print(mu)
[1] 22.53281
```

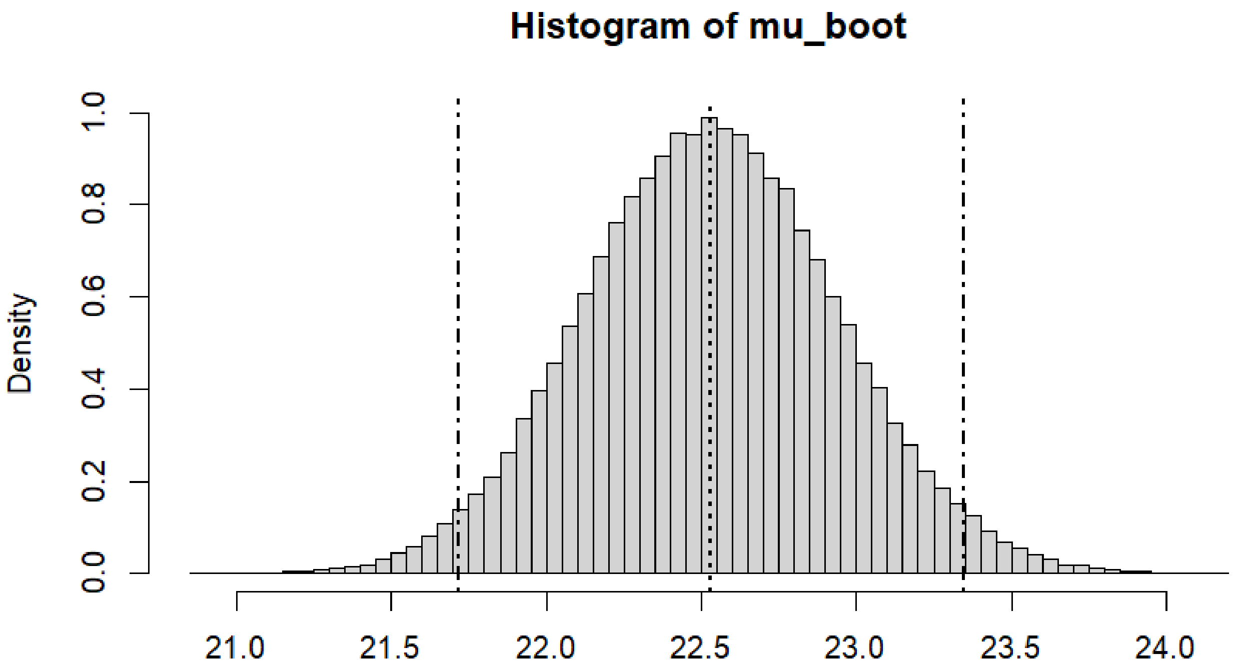
Standard Error

```
> print(se_mu_boot)
[1] 0.4070665
> print(se_mu)
[1] 0.4088611
```

Ex.9) Mean of Residence Value (d)

(d) Based on your bootstrap estimate from (c), provide a 95 % confidence interval for the mean of `medv`. Compare it to the results obtained by using `Boston['medv'].std()` and the two standard error rule (3.9).

Hint: You can approximate a 95 % confidence interval using the formula $[\hat{\mu} - 2\text{SE}(\hat{\mu}), \hat{\mu} + 2\text{SE}(\hat{\mu})]$.



Confidence Interval

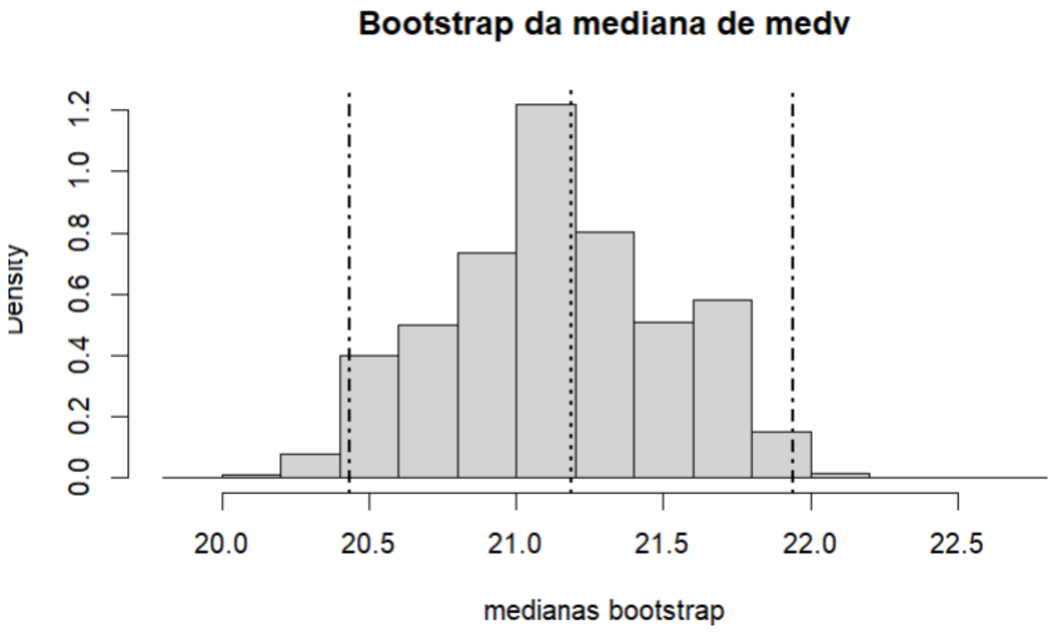
```
> print(ci_mu_hat)
[1] 21.71508 23.35053
```

```
> print(ci_mu_hat_boot)
[1] 21.71769 23.34596
```

Ex.9) Median of Residence Value (e-f)

(e) Based on this data set, provide an estimate, $\hat{\mu}_{med}$, for the median value of `medv` in the population.

(f) We now would like to estimate the standard error of $\hat{\mu}_{med}$. Unfortunately, there is no simple formula for computing the standard error of the median. Instead, estimate the standard error of the median using the bootstrap. Comment on your findings.



Median

```
> print(mu_med_hat)
[1] 21.2
> print(mu_med_hat_boot)
[1] 21.18717
```

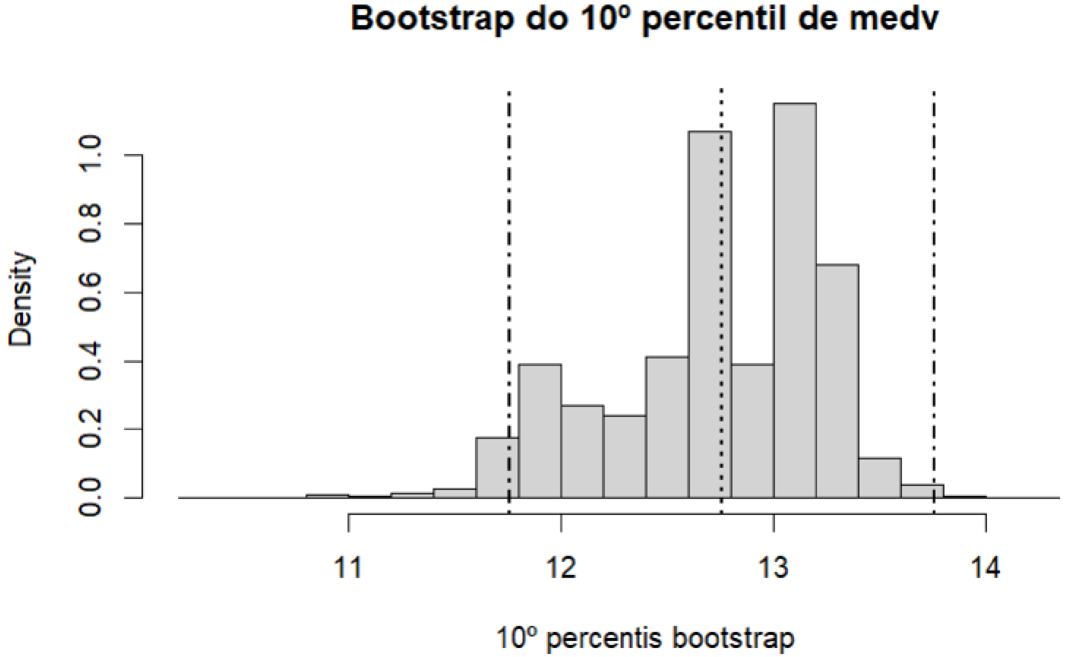
Standard Error

```
> print(se_mu_med_boot)
[1] 0.3774551
```

Ex.9) Percentile 10% of Value (g-h)

(g) Based on this data set, provide an estimate for the tenth percentile of `medv` in Boston census tracts. Call this quantity $\hat{\mu}_{0.1}$. (You can use the `np.percentile()` function.)

(h) Use the bootstrap to estimate the standard error of $\hat{\mu}_{0.1}$. Comment on your findings.



Percentile 10%

```
> print(mu_p10_hat)
[1] 12.75
```

```
> print(mu_p10_hat_boot)
[1] 12.7537
```

Standard Error

```
> print(se_mu_p10_boot)
[1] 0.5009613
```

Ex.9) Bootstrap Takeaways - Applied

- **Estimated values through bootstrap are very close to those obtained using sample statistics and their distribution** (ex. sample mean follows asymptotically a normal distribution)
- **Bootstrap allows to estimate parameters, which do not have a well-known formula or a known distribution** (ex. median and percentile 10% do not seem to follow a normal)