

# Computing Stat Exercises

Khairy Mohamed  
20191480897

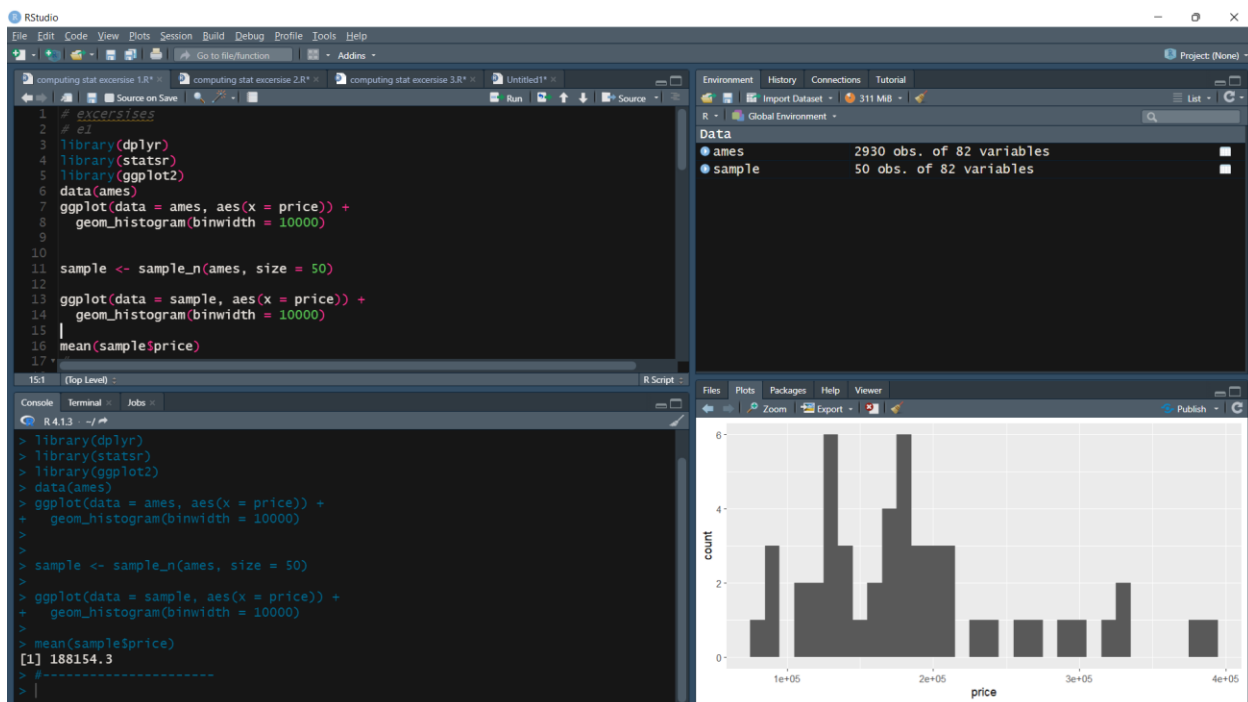
## Lab 1, E1

E1: Take a random sample of size 50 from `price`. Using this sample, what is your best point estimate of the population mean?

## Solution

Best point estimate of population mean with 50 samples = 188154.3

## Code



# Computing Stat Exercises

Khairy Mohamed  
20191480897

## Lab1, E2

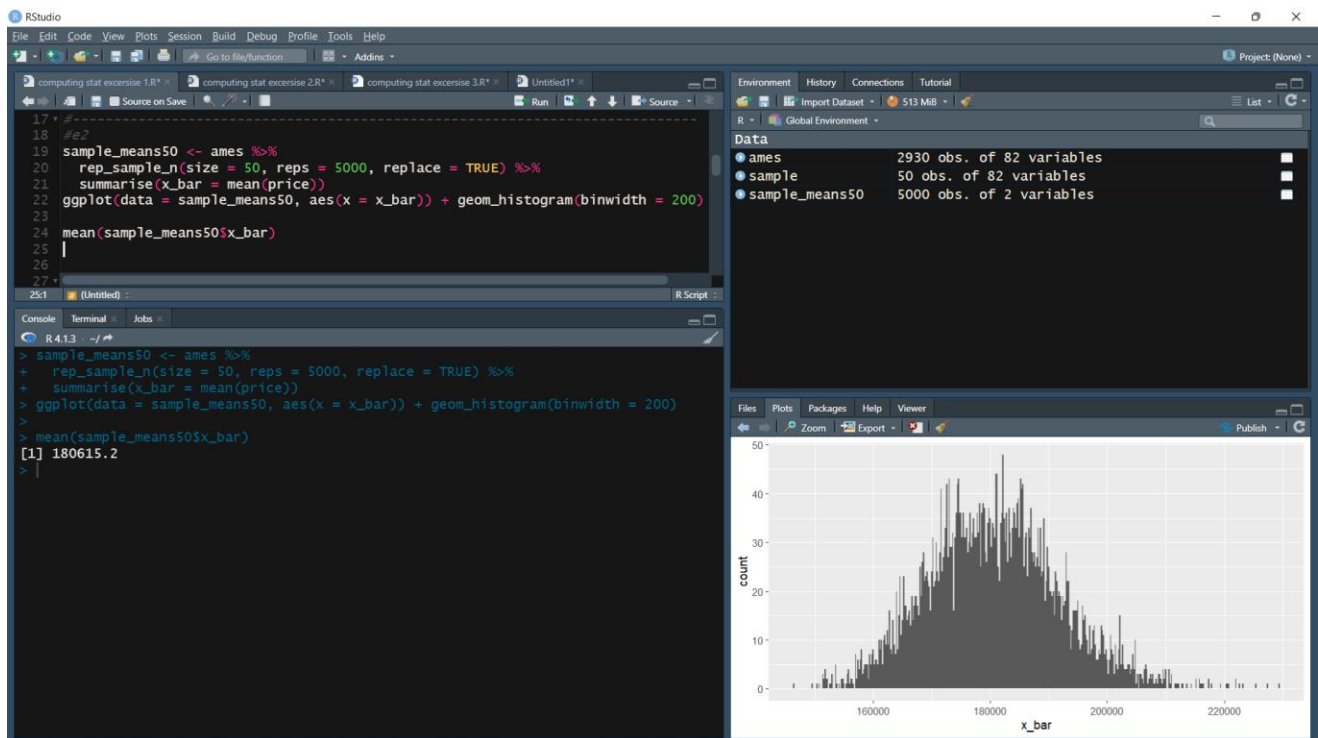
E2: Since you have access to the population, simulate the sampling distribution for  $\bar{x}_{price}$  by taking 5000 samples from the population of size 50 and computing 5000 sample means. Store these means in a vector called `sample_means50`. Plot the data, then describe the shape of this sampling distribution. Based on this sampling distribution, what would you guess the mean home price of the population to be?

## Solution

The sampling distribution is like to be normaly distributed

Estimated mean home price of population = 180615.2

## Code



# Computing Stat Exercises

Khairy Mohamed  
20191480897

## Lab1, E3

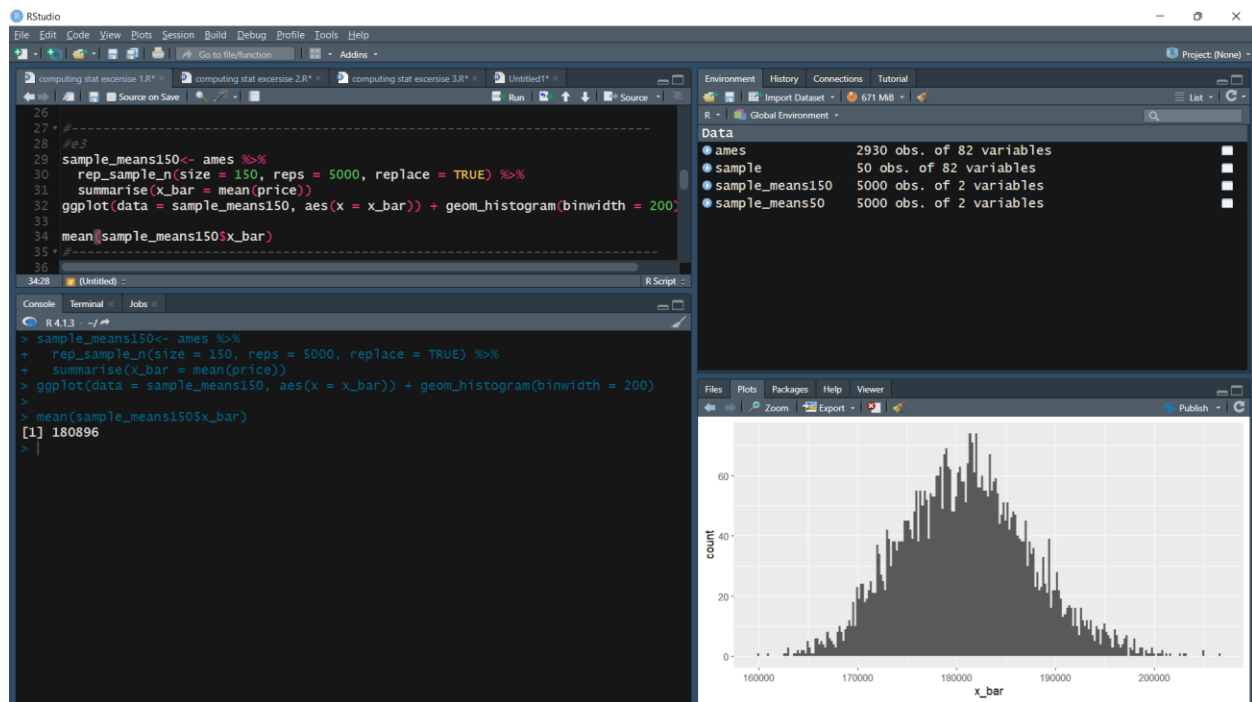
E3: Change your sample size from 50 to 150, then compute the sampling distribution using the same method as above, and store these means in a new vector called `sample_means150`. Describe the shape of this sampling distribution and compare it to the sampling distribution for a sample size of 50. Based on this sampling distribution, what would you guess to be the mean sale price of homes in Ames?

## Solution

The sampling distribution is like to be normaly distributed and more perfect than that one with sample size 50

Estimated mean home price of population = 180896

## Code



# Computing Stat Exercises

Khairy Mohamed  
20191480897

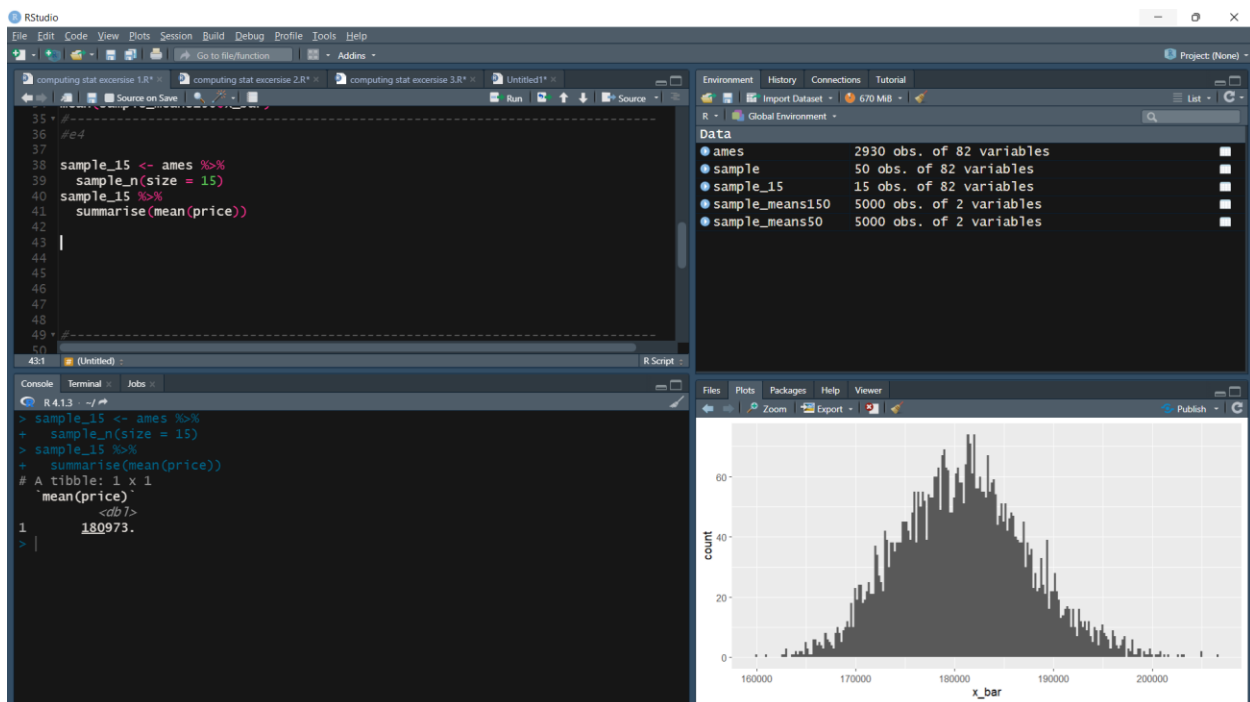
## Lab1, E4

E4: Take a sample of size 15 from the population and calculate the mean `price` of the homes in this sample. Using this sample, what is your best point estimate of the population mean of prices of homes?

## Solution

My best point estimation is 180973

## Code



# Computing Stat Exercises

Khairy Mohamed  
20191480897

## Lab1, E5

E5: Since you have access to the population, simulate the sampling distribution for  $\bar{x}_{price}$  by taking 2000 samples from the population of size 15 and computing 2000 sample means. Store these means in a vector called `sample_means15`. Plot the data, then describe the shape of this sampling distribution. Based on this sampling distribution, what would you guess the mean home price of the population to be? Finally, calculate and report the population mean

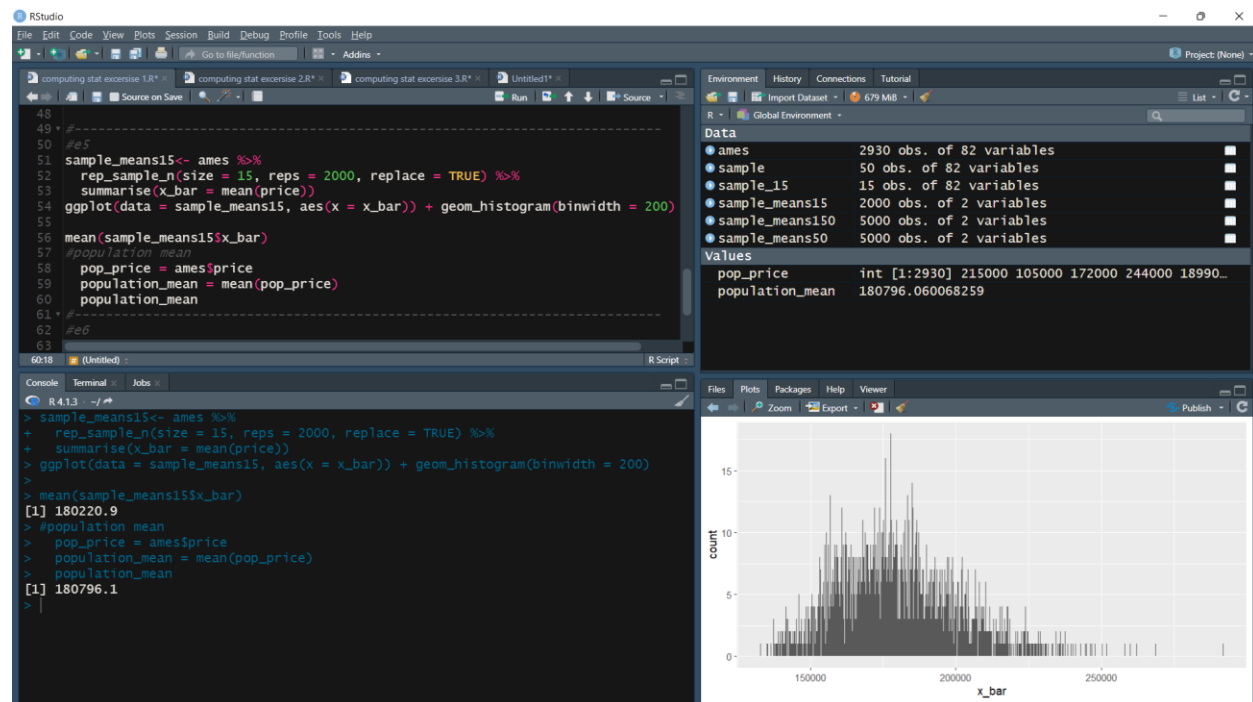
## Solution

The sampling distribution is shown to be normal

Estimated population mean = 180220.9

Population mean = 180796.1

## Code



# Computing Stat Exercises

Khairy Mohamed  
20191480897

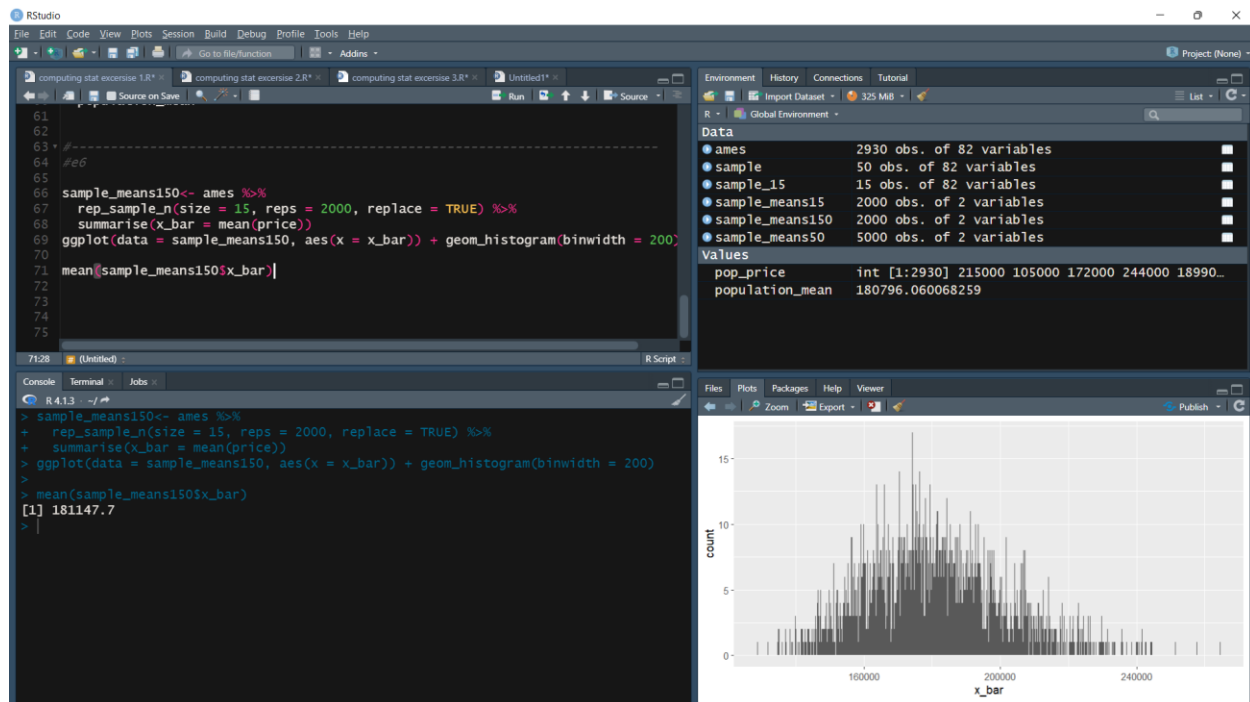
## Lab1, E6

E6: Change your sample size from 15 to 150, then compute the sampling distribution using the same method as above, and store these means in a new vector called `sample_means150`. Describe the shape of this sampling distribution and compare it to the sampling distribution for a sample size of 15. Based on this sampling distribution, what would you guess to be the mean sale price of homes in Ames?

## Solution

Mean of sale price = 181147.7

## Code



## Lab2, ECDF of BMI

- Body mass index (BMI) is a measure of body fat based on height and weight that applies to adult men and women.

- First find the formula for the BMI and calculate it for the data set (heightweight)
- Draw the ecdf for the BMI
- According to the following BMI category and the drawn ecdf, can you determine the percentage of people that are Underweight and Overweight.

### BMI Categories:

Underweight =  $< 18.5$

Normal weight =  $18.5 - 24.9$

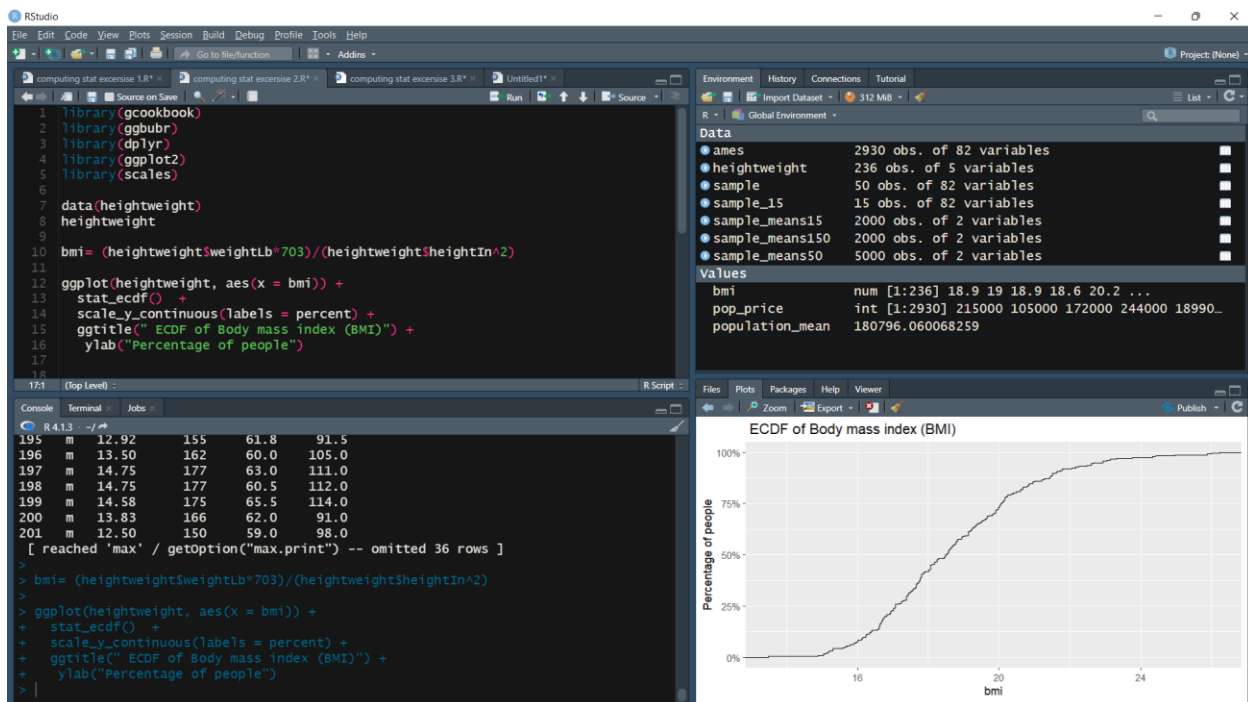
Overweight =  $25 - 29.9$

Obesity = BMI of 30 or greater

## Solution

### ECDF of BMI

## Code



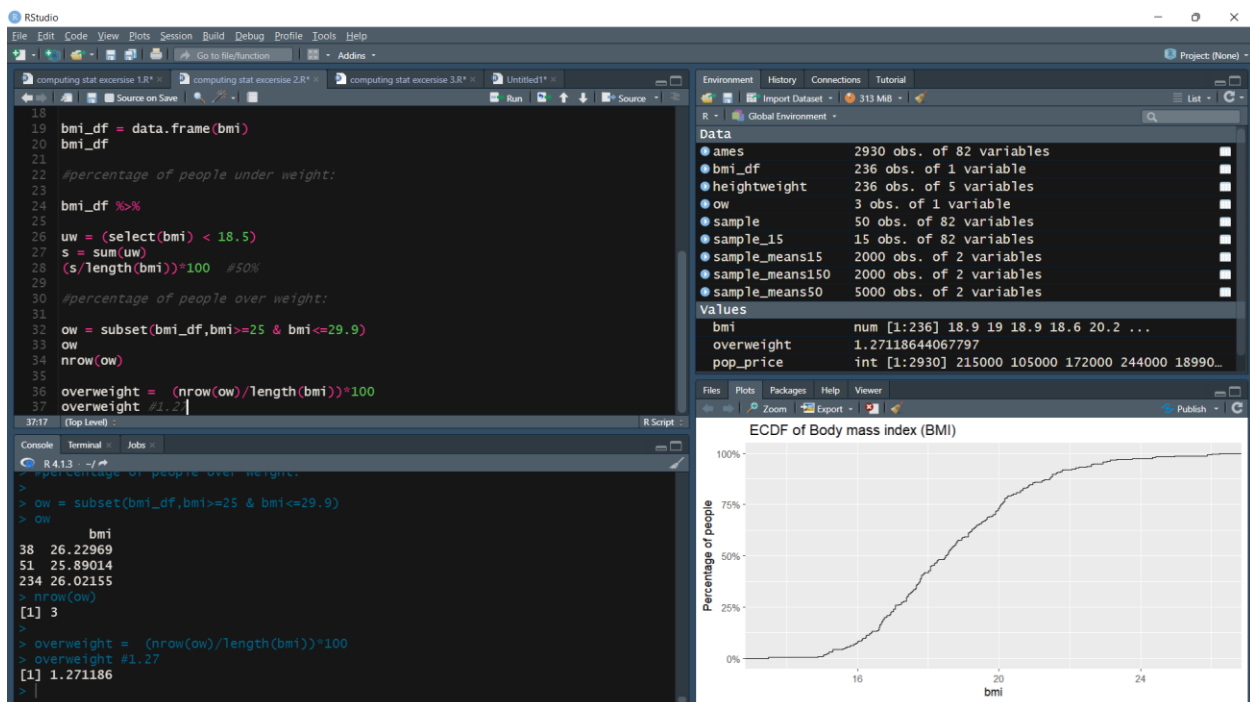
Computing Stat Exercises  
Khairy Mohamed  
20191480897

Lab2, Percentage of people

Solution

- 1) Percentage of people underweight = 50%
- 2) Percentage of people overweight = 1.27%

Code





# Computing Stat Exercises

Khairy Mohamed  
20191480897

## Lab3, E1

### Exercises 1

- Try to calculate the confidence interval in case of unknown population and using t-CI

## Solution

lower      upper

60.36408    62.63592

## Code

### Input

```
1 library(dplyr)
2 library(ggplot2)
3 library(detzrcr)
4 library(gcookbook)
5 #Ex 1
6 data(heightweight)
7 heightweight
8
9 population = data.frame(heightweight)
10
11 pop_var = var(heightweight$heightIn)
12 pop_var
13
14 pop_mean = mean(heightweight$heightIn)
15 pop_mean
16
17 sample_50 = sample_n(population, size = 50)
18
19 mean_sample = mean(sample_50$heightIn)
20 mean_sample
21
22 sample_sd = sd(sample_50$heightIn)
23 sample_sd
24
25 t_star_95 = qt(0.975, df = 49)
26
27 lower_t = mean_sample - t_star_95 * sample_sd/sqrt(50)
28 lower_t
29
30 upper_t = mean_sample + t_star_95 * sample_sd/sqrt(50)
31 upper_t
32
33 CI = c(lower = lower_t, upper = upper_t)
34 print(CI)
35
36 #-----
```

### Output

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
computing stat exercise 1.R* computing stat exercise 2.R* computing stat exercise 3.R* Untitled1*.R
Source
R 4.1.3
>
> population = data.frame(heightweight)
>
> pop_var = var(heightweight$heightIn)
> pop_var
[1] 15.47157
>
> pop_mean = mean(heightweight$heightIn)
> pop_mean
[1] 61.33856
>
> sample_50 = sample_n(population, size = 50)
>
> mean_sample = mean(sample_50$heightIn)
> mean_sample
[1] 60.85
>
> sample_sd = sd(sample_50$heightIn)
> sample_sd
[1] 3.779172
>
> t_star_95 = qt(0.975, df = 49)
>
> lower_t = mean_sample - t_star_95 * sample_sd/sqrt(50)
> lower_t
[1] 59.77597
>
> upper_t = mean_sample + t_star_95 * sample_sd/sqrt(50)
> upper_t
[1] 61.92403
>
> CI = c(lower = lower_t, upper = upper_t)
> print(CI)
      lower      upper
59.77597 61.92403
>
```

## Lab3, E2

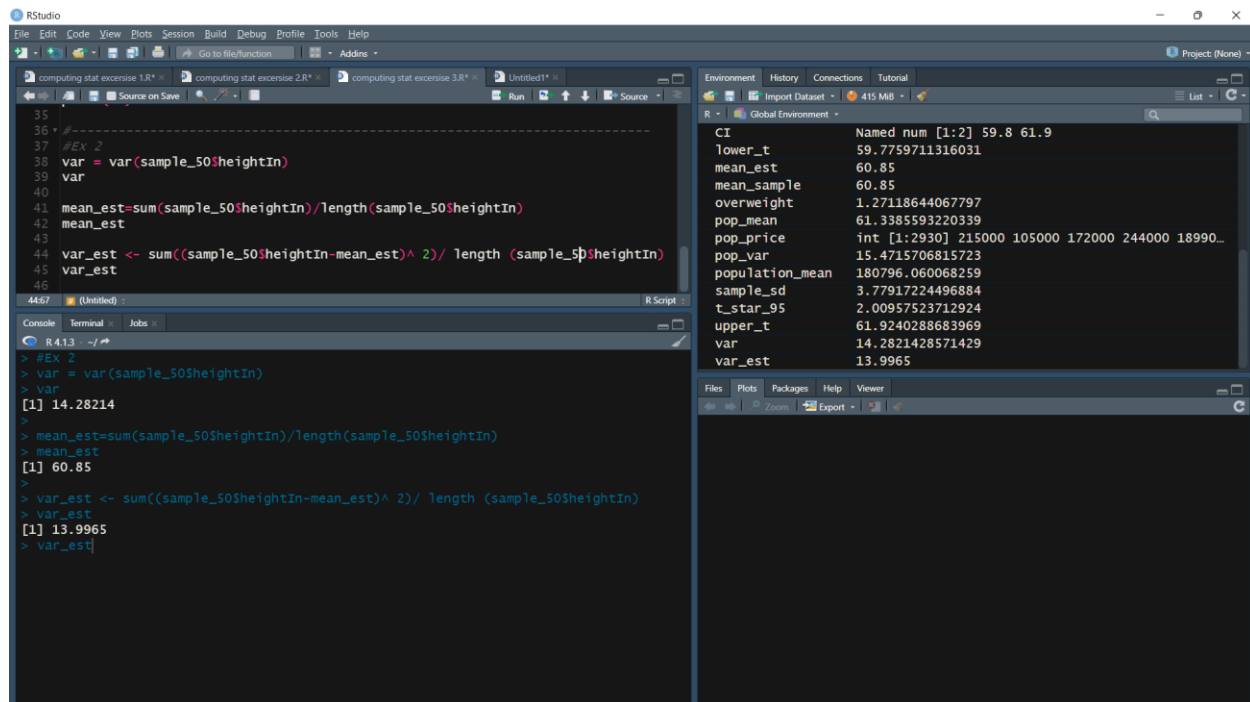
### Exercises 2

1) An alternative way of obtaining the variance will be through the R function `var`, discover the difference between that approach and the estimated value from plug-in estimator.

## Solution

The built in function `var` in R is more accurate than one we calculate manually because the plug-in estimator function is dependent on ECDF, so the error here is double, first when we calculate ECDF, second when we calculate plug-in variance estimator.

## Code



The screenshot shows the RStudio interface. The script editor on the left contains the following R code:

```
35 -----
36 #Ex 2
37 var = var(sample_50$heightIn)
38 var
39
40 mean_est=sum(sample_50$heightIn)/length(sample_50$heightIn)
41 mean_est
42
43 var_est <- sum((sample_50$heightIn-mean_est)^ 2)/ length (sample_50$heightIn)
44 var_est
45
```

The console on the bottom left shows the execution of the code:

```
> #Ex 2
> var = var(sample_50$heightIn)
> var
[1] 14.28214
>
> mean_est=sum(sample_50$heightIn)/length(sample_50$heightIn)
> mean_est
[1] 60.85
>
> var_est <- sum((sample_50$heightIn-mean_est)^ 2)/ length (sample_50$heightIn)
> var_est
[1] 13.9965
> var_est
```

The environment pane on the right shows the following variables:

Variable	Value
CI	Named num [1:2] 59.8 61.9
lower_t	59.7759711316031
mean_est	60.85
mean_sample	60.85
overweight	1.27118644067797
pop_mean	61.3385593220339
pop_price	int [1:2930] 215000 105000 172000 244000 18990...
pop_var	15.4715706815723
population_mean	180796.060068259
sample_sd	3.77917224496884
t_star_95	2.00957523712924
upper_t	61.9240288683969
var	14.2821428571429
var_est	13.9965

Thank You