

# WORKSHEET\_7

MECHELLE SIMPRON

2022-12-22

```
Student <- seq(1:10)
PreTest <- c(55,54,47,57,51,61,57,54,63,58)
PostTest <- c(61,60,56,63,56,63,59,56,62,61)

datastudents <- data.frame(Student,PreTest,PostTest)
datastudents
```

```
##      Student PreTest PostTest
## 1         1      55       61
## 2         2      54       60
## 3         3      47       56
## 4         4      57       63
## 5         5      51       56
## 6         6      61       63
## 7         7      57       59
## 8         8      54       56
## 9         9      63       62
## 10        10      58       61
```

*#a. Compute the descriptive statistics using different packages (Hmisc and pastecs).  
#Write the codes and its result.*

```
library(Hmisc)
```

```
## Warning: package 'Hmisc' was built under R version 4.2.2
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```
##
```

```
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:base':
##
##   format.pval, units
```

```
library(pastecs)
```

```
## Warning: package 'pastecs' was built under R version 4.2.2
```

```
l <- describe(datastudents)
l
```

```
## datastudents
##
## 3 Variables      10 Observations
## -----
## Student
##      n missing distinct      Info      Mean      Gmd      .05      .10
##      10      0      10      1      5.5      3.667      1.45      1.90
##      .25      .50      .75      .90      .95
##      3.25      5.50      7.75      9.10      9.55
##
## lowest : 1 2 3 4 5, highest: 6 7 8 9 10
##
## Value      1 2 3 4 5 6 7 8 9 10
## Frequency  1 1 1 1 1 1 1 1 1 1
## Proportion 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1
## -----
## PreTest
##      n missing distinct      Info      Mean      Gmd
##      10      0      8      0.988      55.7      5.444
##
## lowest : 47 51 54 55 57, highest: 55 57 58 61 63
##
## Value      47 51 54 55 57 58 61 63
## Frequency  1 1 2 1 2 1 1 1
## Proportion 0.1 0.1 0.2 0.1 0.2 0.1 0.1 0.1
## -----
## PostTest
##      n missing distinct      Info      Mean      Gmd
##      10      0      6      0.964      59.7      3.311
##
## lowest : 56 59 60 61 62, highest: 59 60 61 62 63
##
## Value      56 59 60 61 62 63
## Frequency  3 1 1 2 1 2
## Proportion 0.3 0.1 0.1 0.2 0.1 0.2
## -----
```

```
m <- stat.desc(datastudents)
m
```

```
##           Student      PreTest      PostTest
```

```
## nbr.val      10.0000000 10.00000000 10.00000000
## nbr.null     0.0000000  0.00000000  0.00000000
## nbr.na       0.0000000  0.00000000  0.00000000
## min         1.0000000 47.00000000 56.00000000
## max        10.0000000 63.00000000 63.00000000
## range       9.0000000 16.00000000  7.00000000
## sum        55.0000000 557.00000000 597.00000000
## median      5.5000000 56.00000000 60.50000000
## mean       5.5000000 55.70000000 59.70000000
## SE.mean    0.9574271  1.46855938  0.89504811
## CI.mean.0.95 2.1658506  3.32211213  2.02473948
## var       9.1666667 21.56666667  8.01111111
## std.dev    3.0276504  4.64399254  2.83039063
## coef.var   0.5504819  0.08337509  0.04741023
```

*#2. The Department of Agriculture was studying the effects of several levels of a fertilizer on the growth of a plant. For some analyses, it might be useful to convert the fertilizer levels to an ordered factor.*

```
# The data were 10,10,10, 20,20,50,10,20,10,50,20,50,20,10.
d1 <- c(10,10,10,20,20,50,10,
        20,10,50,20,50,20,10)
d1
```

```
## [1] 10 10 10 20 20 50 10 20 10 50 20 50 20 10
```

```
#a. Write the codes and describe the result.
#ans- its levels differ from 10 to 20 to 50
factord1 <- factor(d1, ordered = TRUE)
factord1
```

```
## [1] 10 10 10 20 20 50 10 20 10 50 20 50 20 10
## Levels: 10 < 20 < 50
```

*#3. Abdul Hassan, president of Floor Coverings Unlimited, has asked you to study the exercise levels undertaken by 10 num3 were "l", "n", "n", "i", "l", "l", "n", "n", "i", "l", "l", "n", "n", "i", "l". ; n=none, l=light, i=intense*

```
d2 <- c("l","n","n","i","l","l","n","n","i","l")
d2
```

```
## [1] "l" "n" "n" "i" "l" "l" "n" "n" "i" "l"
```

```
#a. What is the best way to represent this in R?
#ans - presenting the data in Data frame
d2DF <- data.frame(d2)
d2DF
```

```
##      d2
## 1    l
## 2    n
## 3    n
## 4    i
```

```
## 5 1
## 6 1
## 7 n
## 8 n
## 9 i
## 10 1
```

*#4. Sample of 30 tax accountants from all the states and territories of Australia and their individual state of origin is specified by a character vector of state mnemonics #as:*

```
state_d <- c("tas", "sa", "qld", "nsw", "nsw", "nt", "wa", "wa", "qld",
             "vic", "nsw", "vic", "qld", "qld", "sa", "tas", "sa", "nt",
             "wa", "vic", "qld", "nsw", "nsw", "wa", "sa", "act", "nsw",
             "vic", "vic", "act")
state_d
```

```
## [1] "tas" "sa" "qld" "nsw" "nsw" "nt" "wa" "wa" "qld" "vic" "nsw" "vic"
## [13] "qld" "qld" "sa" "tas" "sa" "nt" "wa" "vic" "qld" "nsw" "nsw" "wa"
## [25] "sa" "act" "nsw" "vic" "vic" "act"
```

*#a. Apply the factor function and factor level. Describe the results.*

```
#factor function and factor level
fstate_d <- factor(state_d)
fstate_d
```

```
## [1] tas sa qld nsw nsw nt wa wa qld vic nsw vic qld qld sa tas sa nt wa
## [20] vic qld nsw nsw wa sa act nsw vic vic act
## Levels: act nsw nt qld sa tas vic wa
```

*#Getting factor level of states*

```
l_state <- levels(fstate_d)
l_state
```

```
## [1] "act" "nsw" "nt" "qld" "sa" "tas" "vic" "wa"
```

*#5. From #4 - continuation:*

*#• Suppose we have the incomes of the same tax accountants in another vector (insuitably large units of*

```
income <- c(60, 49, 40, 61, 64, 60, 59, 54,
            62, 69, 70, 42, 56, 61, 61, 61, 58, 51, 48,
            65, 49, 49, 41, 48, 52, 46, 59, 46, 58, 43)
income
```

```
## [1] 60 49 40 61 64 60 59 54 62 69 70 42 56 61 61 61 58 51 48 65 49 49 41 48 52
## [26] 46 59 46 58 43
```

*#a. Calculate the sample mean income for each state we can now use the special #function tapply():*

```
total_i <- tapply(income, state_d, mean)
total_i
```

```
##      act      nsw      nt      qld      sa      tas      vic      wa
## 44.50000 57.33333 55.50000 53.60000 55.00000 60.50000 56.00000 52.25000
```

*#b. Copy the results and interpret.*

```
total_i
```

```
##      act      nsw      nt      qld      sa      tas      vic      wa
## 44.50000 57.33333 55.50000 53.60000 55.00000 60.50000 56.00000 52.25000
```

*#ans- the state was factor out and was applied for the tapply function in order for the income #to be arranged corresponding in each state.*

*#6. Calculate the standard errors of the state income means (refer again to number 3)*

*#a. What is the standard error? Write the codes.*

```
length_d <- length(total_i)
length_d
```

```
## [1] 8
```

```
sd_d <- sd(total_i)
sd_d
```

```
## [1] 4.677966
```

```
d3 <- sd_d/sqrt(length_d)
d3
```

```
## [1] 1.653911
```

*#b. Interpret the result.*

*#the data was summarize in length as it shows it has a 8 observation and was also summarize by getting*

*#7. Use the titanic dataset.*

```
data(Titanic)
titanic <- data.frame(Titanic)
titanic
```

```
##      Class  Sex  Age Survived Freq
## 1    1st  Male Child      No     0
## 2    2nd  Male Child      No     0
## 3    3rd  Male Child      No    35
## 4   Crew  Male Child      No     0
## 5    1st Female Child      No     0
## 6    2nd Female Child      No     0
## 7    3rd Female Child      No    17
## 8   Crew Female Child      No     0
## 9    1st  Male Adult      No   118
## 10   2nd  Male Adult      No   154
## 11   3rd  Male Adult      No   387
```

```
## 12 Crew Male Adult No 670
## 13 1st Female Adult No 4
## 14 2nd Female Adult No 13
## 15 3rd Female Adult No 89
## 16 Crew Female Adult No 3
## 17 1st Male Child Yes 5
## 18 2nd Male Child Yes 11
## 19 3rd Male Child Yes 13
## 20 Crew Male Child Yes 0
## 21 1st Female Child Yes 1
## 22 2nd Female Child Yes 13
## 23 3rd Female Child Yes 14
## 24 Crew Female Child Yes 0
## 25 1st Male Adult Yes 57
## 26 2nd Male Adult Yes 14
## 27 3rd Male Adult Yes 75
## 28 Crew Male Adult Yes 192
## 29 1st Female Adult Yes 140
## 30 2nd Female Adult Yes 80
## 31 3rd Female Adult Yes 76
## 32 Crew Female Adult Yes 20
```

*#a. subset the titanic dataset of those who survived and not survived. Show the #codes and its result.*

```
survive_d <- subset(titanic, Survived == "Yes")
survive_d
```

```
## Class Sex Age Survived Freq
## 17 1st Male Child Yes 5
## 18 2nd Male Child Yes 11
## 19 3rd Male Child Yes 13
## 20 Crew Male Child Yes 0
## 21 1st Female Child Yes 1
## 22 2nd Female Child Yes 13
## 23 3rd Female Child Yes 14
## 24 Crew Female Child Yes 0
## 25 1st Male Adult Yes 57
## 26 2nd Male Adult Yes 14
## 27 3rd Male Adult Yes 75
## 28 Crew Male Adult Yes 192
## 29 1st Female Adult Yes 140
## 30 2nd Female Adult Yes 80
## 31 3rd Female Adult Yes 76
## 32 Crew Female Adult Yes 20
```

```
died_d <- subset(titanic, Survived == "No")
died_d
```

```
## Class Sex Age Survived Freq
## 1 1st Male Child No 0
## 2 2nd Male Child No 0
## 3 3rd Male Child No 35
```

```
## 4   Crew   Male Child      No    0
## 5    1st  Female Child      No    0
## 6    2nd  Female Child      No    0
## 7    3rd  Female Child      No   17
## 8   Crew  Female Child      No    0
## 9    1st   Male Adult      No  118
## 10   2nd   Male Adult      No  154
## 11   3rd   Male Adult      No  387
## 12  Crew   Male Adult      No  670
## 13   1st  Female Adult      No    4
## 14   2nd  Female Adult      No   13
## 15   3rd  Female Adult      No   89
## 16  Crew  Female Adult      No    3
```

*#8. The data sets are about the breast cancer Wisconsin. The samples arrive periodically as Dr. Wolberg  
#chronological grouping of the data. You can create this dataset in Microsoft Excel.*

*#a. describe what is the dataset all about.*

*# The data is for breast cancer patients, it was categorized in different types of breasts cancer and h*

*#b. Import the data from MS Excel. Copy the codes.*

```
library("readxl")
```

```
## Warning: package 'readxl' was built under R version 4.2.2
```

```
d4 <- read_excel("C:/Users/Naomi/Desktop/Breast_Cancer.xlsx")
```

```
d4
```

```
## # A tibble: 49 x 11
```

```
##       Id CL. thickne~1 Cell ~2 Cell ~3 Marg.~4 Epith~5 Bare.~6 Bl. C~7 Norma~8
##       <dbl>         <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>     <dbl>   <dbl>
##  1 1000025           5         1         1         1         2 1           3         1
##  2 1002945           5         4         4         5         7 10          3         2
##  3 1015425           3         1         1         1         2 2           3         1
##  4 1016277           6         8         8         1         3 4           3         7
##  5 1017023           4         1         1         3         2 1           3         1
##  6 1017122           8        10        10         8         7 10          9         7
##  7 1018099           1         1         1         1         2 10          3         1
##  8 1018561           2         1         2         1         2 1           3         1
##  9 1033078           2         1         1         1         2 1           1         1
## 10 1033078           4         2         1         1         2 1           2         1
## # ... with 39 more rows, 2 more variables: Mitoses <dbl>, Class <chr>, and
## # abbreviated variable names 1: 'CL. thickness', 2: 'Cell size',
## # 3: 'Cell Shape', 4: 'Marg. Adhesion', 5: 'Epith. C.size',
## # 6: 'Bare. Nuclei', 7: 'Bl. Cromatin', 8: 'Normal nucleoli'
```

*#c. Compute the descriptive statistics using different packages. Find the values of:  
#c.1 Standard error of the mean for clump thickness.*

```
clump_1 <- length(d4$`CL. thickness`)
clump_1
```

```
## [1] 49
```

```
clump_sd <- sd(d4$`CL. thickness`)
clump_sd
```

```
## [1] 2.865019
```

```
clump_se <- clump_sd/sqrt(d4$`CL. thickness`)
clump_se
```

```
## [1] 1.2812754 1.2812754 1.6541194 1.1696391 1.4325095 1.0129371 2.8650189
## [8] 2.0258743 2.0258743 1.4325095 2.8650189 2.0258743 1.2812754 2.8650189
## [15] 1.0129371 1.0828754 1.4325095 1.4325095 0.9059985 1.1696391 1.0828754
## [22] 0.9059985 1.6541194 1.0129371 2.8650189 1.2812754 1.6541194 1.2812754
## [29] 2.0258743 2.8650189 1.6541194 2.0258743 0.9059985 2.0258743 1.6541194
## [36] 2.0258743 0.9059985 1.1696391 1.2812754 2.0258743 1.1696391 0.9059985
## [43] 1.1696391 1.2812754 0.9059985 2.8650189 1.6541194 2.8650189 1.4325095
```

*#c.2 Coefficient of variability for Marginal Adhesion.*

```
d5 <- sd(d4$`Marg. Adhesion`) / mean(d4$`Marg. Adhesion`) * 100
d5
```

```
## [1] 97.67235
```

*#c.3 Number of null values of Bare Nuclei.*

```
nuclei <- subset(d4, `Bare. Nuclei` == "NA")
nuclei
```

```
## # A tibble: 2 x 11
##       Id CL. t~1 Cell ~2 Cell ~3 Marg.~4 Epith~5 Bare.~6 Bl. C~7 Norma~8 Mitoses
##       <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>      <dbl>   <dbl>   <dbl>
## 1 1.06e6     8     4     5     1     2 NA         7     3     1
## 2 1.10e6     6     6     6     9     6 NA         7     8     1
## # ... with 1 more variable: Class <chr>, and abbreviated variable names
## #   1: 'CL. thickness', 2: 'Cell size', 3: 'Cell Shape', 4: 'Marg. Adhesion',
## #   5: 'Epith. C.size', 6: 'Bare. Nuclei', 7: 'Bl. Cromatin',
## #   8: 'Normal nucleoli'
```

*#c.4 Mean and standard deviation for Bland Chromatin*

```
bc <- mean(d4$`Bl. Chromatin`)
bc
```

```
sdBC <- sd(d4$`Bl. Chromatin`)
sdBC
```

*#c.5 Confidence interval of the mean for Uniformity of Cell Shape*

*#Calculate the mean*

```
cs <- mean(d4$`Cell Shape`)
cs
```

*#Calculate the standard error of the mean*

```
length_se <- length(d4$`Cell Shape`)
```



```

lenght_se

sd_se <- sd(d4$`Cell Shape`)
sd_se

Cellshape <- sd_se/sqrt(lenght_se)
Cellshape

#Find the t-score that corresponds to the confidence level
t_score = 0.05
t_score

t_scores = lenght_se - 1
t_scores

ts = qt(p=t_score/2, df=t_scores,lower.tail=F)
ts

#Constructing the confidence interval
c <- ts * Cellshape
c

#Lower
c_diff <- cs - c
c_diff

#Upper
sumCI <- cs + c
sumCI

d6 <- c(c_diff, sumCI)
d6

```

```

#d. How many attributes?
attributess <- attributes(d4)
attributess

```

```

## $class
## [1] "tbl_df"      "tbl"        "data.frame"
##
## $row.names
## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
##
## $names
## [1] "Id"                "CL. thickness"    "Cell size"        "Cell Shape"
## [5] "Marg. Adhesion"    "Epith. C.size"    "Bare. Nuclei"      "Bl. Cromatin"
## [9] "Normal nucleoli"   "Mitoses"          "Class"

```

```

#e. Find the percentage of respondents who are malignant. Interpret the results.
res_percentage <- subset(d4, Class == "malignant")
res_percentage

```

```
## # A tibble: 18 x 11
##       Id CL. thickne~1 Cell ~2 Cell ~3 Marg.~4 Epith~5 Bare.~6 Bl. C~7 Norma~8
##       <dbl>         <dbl>    <dbl>    <dbl>    <dbl>    <dbl> <chr>      <dbl>    <dbl>
## 1 1017122           8         10         10         8         7 10          9         7
## 2 1041801           5          3          3          3         2 3           4         4
## 3 1044572           8          7          5         10         7 9           5         5
## 4 1047630           7          4          6          4         6 1           4         3
## 5 1050670          10          7          7          6         4 10          4         1
## 6 1054590           7          3          2         10         5 10          5         4
## 7 1054593          10          5          5          3         6 7           7        10
## 8 1057013           8          4          5          1         2 NA           7         3
## 9 1065726           5          2          3          4         2 7           3         6
## 10 1072179          10          7          7          3         8 5           7         4
## 11 1080185          10         10         10          8         6 1           8         9
## 12 1084584           5          4          4          9         2 10          5         6
## 13 1091262           2          5          3          3         6 7           7         5
## 14 1099510          10          4          3          1         3 3           6         5
## 15 1100524           6         10         10          2         8 10          7         3
## 16 1102573           5          6          5          6        10 1           3         1
## 17 1103608          10         10         10          4         8 1           8        10
## 18 1105257           3          7          7          4         4 9           4         8
## # ... with 2 more variables: Mitoses <dbl>, Class <chr>, and abbreviated
## #   variable names 1: 'CL. thickness', 2: 'Cell size', 3: 'Cell Shape',
## #   4: 'Marg. Adhesion', 5: 'Epith. C.size', 6: 'Bare. Nuclei',
## #   7: 'Bl. Cromatin', 8: 'Normal nucleoli'
```

*#18 respondents are malignant in total of 49 respondents. Getting the percentage  $18 / 49 * 100 / 49 * 1$*

*#There 36% of respondents who are malignant.*

*#9. Export the data abalone to the Microsoft excel file. Copy the codes.*

```
library("AppliedPredictiveModeling")
```

```
## Warning: package 'AppliedPredictiveModeling' was built under R version 4.2.2
```

```
data(abalone)
View(abalone)
head(abalone)
```

```
##   Type LongestShell Diameter Height WholeWeight ShuckedWeight VisceraWeight
## 1    M      0.455    0.365  0.095    0.5140      0.2245      0.1010
## 2    M      0.350    0.265  0.090    0.2255      0.0995      0.0485
## 3    F      0.530    0.420  0.135    0.6770      0.2565      0.1415
## 4    M      0.440    0.365  0.125    0.5160      0.2155      0.1140
## 5    I      0.330    0.255  0.080    0.2050      0.0895      0.0395
## 6    I      0.425    0.300  0.095    0.3515      0.1410      0.0775
##   ShellWeight Rings
## 1      0.150     15
## 2      0.070      7
## 3      0.210      9
## 4      0.155     10
## 5      0.055      7
## 6      0.120      8
```

```
summary(abalone)
```

```
## Type      LongestShell      Diameter      Height      WholeWeight
## F:1307   Min.    :0.075   Min.    :0.0550   Min.    :0.0000   Min.    :0.0020
## I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415
## M:1528   Median :0.545   Median :0.4250   Median :0.1400   Median :0.7995
##          Mean    :0.524   Mean    :0.4079   Mean    :0.1395   Mean    :0.8287
##          3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650   3rd Qu.:1.1530
##          Max.    :0.815   Max.    :0.6500   Max.    :1.1300   Max.    :2.8255
## ShuckedWeight VisceraWeight ShellWeight Rings
## Min.    :0.0010   Min.    :0.0005   Min.    :0.0015   Min.    : 1.000
## 1st Qu.:0.1860   1st Qu.:0.0935   1st Qu.:0.1300   1st Qu.: 8.000
## Median :0.3360   Median :0.1710   Median :0.2340   Median : 9.000
## Mean    :0.3594   Mean    :0.1806   Mean    :0.2388   Mean    : 9.934
## 3rd Qu.:0.5020   3rd Qu.:0.2530   3rd Qu.:0.3290   3rd Qu.:11.000
## Max.    :1.4880   Max.    :0.7600   Max.    :1.0050   Max.    :29.000
```

```
#Exporting the data abalone to the Microsoft excel file
library(xlsx)
```

```
## Warning: package 'xlsx' was built under R version 4.2.2
```

```
d8 <- write.xlsx("abalone","C:/Users/Naomi/Desktop/abalone.xlsx")
d8
```

```
## NULL
```