RWorksheet_Aguas#6

2023-12-21

1.

a.

```
Score <- data.frame(Student = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 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))

Score
```

```
##    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
```

```
library(Hmisc)
```

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

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

```
library(pastecs)

Hmisc <- describe(Score[, c("PreTest", "PostTest")])
Hmisc
```

```
## Score[, c("PreTest", "PostTest")]
##
##  2  Variables      10  Observations
## --------------------------------------------------------------------------------
## PreTest
##        n  missing distinct     Info     Mean      Gmd
##       10        0        8    0.988     55.7    5.444
##
## 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
##
## For the frequency table, variable is rounded to the nearest 0
## --------------------------------------------------------------------------------
## PostTest
##        n  missing distinct     Info     Mean      Gmd
##       10        0        6    0.964     59.7    3.311
##
## 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
##
## For the frequency table, variable is rounded to the nearest 0
## --------------------------------------------------------------------------------
```

```r
pastecs <- stat.desc(Score[, c('PreTest', 'PostTest')])
pastecs
```

```
##                   PreTest     PostTest
## nbr.val       10.00000000  10.00000000
## nbr.null       0.00000000   0.00000000
## nbr.na         0.00000000   0.00000000
## min           47.00000000  56.00000000
## max           63.00000000  63.00000000
## range         16.00000000   7.00000000
## sum          557.00000000 597.00000000
## median        56.00000000  60.50000000
## mean          55.70000000  59.70000000
## SE.mean        1.46855938   0.89504811
## CI.mean.0.95   3.32211213   2.02473948
## var           21.56666667   8.01111111
## std.dev        4.64399254   2.83039063
## coef.var       0.08337509   0.04741023
```

2.

```r
library(dplyr)
```

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

```
## The following objects are masked from 'package:pastecs':
##
##     first, last
```

```
## The following objects are masked from 'package:Hmisc':
##
##     src, summarize
```

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

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

```r
FertiLevel <- c(10,10,10, 20,20,50,10,20,10,50,20,50,20,10)

OrderedFactor <- factor(FertiLevel, levels = unique(FertiLevel))
```

```
basicStats <- summary(OrderedFactor)
basicStats
```

```
## 10 20 50
##  6  5  3
```

3.

a.

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

ExerciseFactor <- factor(excerciseLevels, levels = c("n","l","i"))


BasicStats <- summary(ExerciseFactor)
BasicStats
```

```
## n l i
## 5 4 1
```

4.

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

```
state <- 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")
stateFactor <- factor(state)
stateFactor
```

```
##  [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
```

```
summaryState <- summary(stateFactor)


#The output will show the levels (unique values) in the factor (act, nsw, nt, qld, sa, tas, vic, wa) an
```

5.

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

```
incomes <- 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)

meanIncome <- tapply(incomes, stateFactor, mean)
meanIncome
```

```
##      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.

```
#  act      nsw       nt      qld       sa      tas      vic       wa
#44.50000 57.33333 55.50000 53.60000 55.00000 60.50000 56.00000 52.25000
#The code attempts to calculate the mean income for different states using the tapply function, but it
```

6.

a.

```
stdError <- function(x) sqrt(var(x)/length(x))
incster <- tapply(incomes, state, stdError)
standardError <- tapply(incomes, stateFactor, stdError)
standardError
```

```
##      act      nsw       nt      qld       sa      tas      vic       wa
## 1.500000 4.310195 4.500000 4.106093 2.738613 0.500000 5.244044 2.657536
```

b.interpret the result.

```
#These values indicate the precision of the estimated mean for each region. Higher standard errors gene
```

7.

a.

```
install.packages("titanic")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library(titanic)

data("titanic_train")
titanic_data <- titanic_train

survived_data <- subset(titanic_data, Survived == 1)

not_survived_data <- subset(titanic_data, Survived == 0)

head(survived_data)
```

```
##    PassengerId Survived Pclass
## 2            2        1      1
## 3            3        1      3
## 4            4        1      1
## 9            9        1      3
## 10          10        1      2
## 11          11        1      3
##                                                   Name    Sex Age SibSp Parch
## 2  Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                               Heikkinen, Miss. Laina female  26     0     0
## 4         Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 9    Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg) female  27     0     2
## 10                 Nasser, Mrs. Nicholas (Adele Achem) female  14     1     0
## 11                     Sandstrom, Miss. Marguerite Rut female   4     1     1
##              Ticket    Fare Cabin Embarked
## 2          PC 17599 71.2833   C85        C
## 3  STON/O2. 3101282  7.9250              S
## 4            113803 53.1000  C123        S
## 9            347742 11.1333              S
## 10           237736 30.0708              C
## 11          PP 9549 16.7000    G6        S
```

```r
head(not_survived_data)
```

```
##    PassengerId Survived Pclass                           Name  Sex Age SibSp
## 1            1        0      3        Braund, Mr. Owen Harris male  22     1
## 5            5        0      3       Allen, Mr. William Henry male  35     0
## 6            6        0      3               Moran, Mr. James male  NA     0
## 7            7        0      1        McCarthy, Mr. Timothy J male  54     0
## 8            8        0      3 Palsson, Master. Gosta Leonard male   2     3
## 13          13        0      3 Saundercock, Mr. William Henry male  20     0
##    Parch    Ticket    Fare Cabin Embarked
## 1      0 A/5 21171  7.2500              S
## 5      0    373450  8.0500              S
## 6      0    330877  8.4583              Q
## 7      0     17463 51.8625   E46        S
## 8      1    349909 21.0750              S
## 13     0 A/5. 2151  8.0500              S
```

```r
survived_data <- titanic_data[titanic_data$Survived == 1, ]

not_survived_data <- titanic_data[titanic_data$Survived == 0, ]


head(survived_data)
```

```
##    PassengerId Survived Pclass
## 2            2        1      1
## 3            3        1      3
## 4            4        1      1
## 9            9        1      3
## 10          10        1      2
## 11          11        1      3
##                                                    Name    Sex Age SibSp Parch
## 2  Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
## 3                               Heikkinen, Miss. Laina female  26     0     0
## 4         Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
## 9    Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg) female  27     0     2
## 10                 Nasser, Mrs. Nicholas (Adele Achem) female  14     1     0
## 11                     Sandstrom, Miss. Marguerite Rut female   4     1     1
##              Ticket    Fare Cabin Embarked
## 2          PC 17599 71.2833   C85        C
## 3  STON/O2. 3101282  7.9250              S
## 4            113803 53.1000  C123        S
## 9            347742 11.1333              S
## 10           237736 30.0708              C
## 11          PP 9549 16.7000    G6        S
```

```r
head(not_survived_data)
```

```
##    PassengerId Survived Pclass                           Name  Sex Age SibSp
## 1            1        0      3        Braund, Mr. Owen Harris male  22     1
## 5            5        0      3       Allen, Mr. William Henry male  35     0
## 6            6        0      3               Moran, Mr. James male  NA     0
## 7            7        0      1        McCarthy, Mr. Timothy J male  54     0
## 8            8        0      3 Palsson, Master. Gosta Leonard male   2     3
## 13          13        0      3 Saundercock, Mr. William Henry male  20     0
##    Parch    Ticket    Fare Cabin Embarked
## 1      0 A/5 21171  7.2500              S
## 5      0    373450  8.0500              S
## 6      0    330877  8.4583              Q
## 7      0     17463 51.8625   E46        S
## 8      1    349909 21.0750              S
## 13     0 A/5. 2151  8.0500              S
```

8.

chronologihttps://drive.google.com/file/d/16MFLoehCgx2MJuNSAuB2CsBy6eDIIr- u/view?usp=drive_link)

a. describe what is the dataset all about.

```
#The dataset consists of cytological features of breast cancer cell samples, such as clump thickness, s
```

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

```
library(readr)


breastcancer_wisconsin <- read_csv("/cloud/project/breastcancer_wisconsin.csv")
```

```
## Rows: 699 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): bare_nucleoli
## dbl (10): id, clump_thickness, size_uniformity, shape_uniformity, marginal_a...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
str(breastcancer_wisconsin)
```

```
## spc_tbl_ [699 x 11] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ id               : num [1:699] 1000025 1002945 1015425 1016277 1017023 ...
##  $ clump_thickness  : num [1:699] 5 5 3 6 4 8 1 2 2 4 ...
##  $ size_uniformity  : num [1:699] 1 4 1 8 1 10 1 1 1 2 ...
##  $ shape_uniformity : num [1:699] 1 4 1 8 1 10 1 2 1 1 ...
##  $ marginal_adhesion: num [1:699] 1 5 1 1 3 8 1 1 1 1 ...
##  $ epithelial_size  : num [1:699] 2 7 2 3 2 7 2 2 2 2 ...
##  $ bare_nucleoli    : chr [1:699] "1" "10" "2" "4" ...
##  $ bland_chromatin  : num [1:699] 3 3 3 3 3 9 3 3 1 2 ...
##  $ normal_nucleoli  : num [1:699] 1 2 1 7 1 7 1 1 1 1 ...
##  $ mitoses          : num [1:699] 1 1 1 1 1 1 1 1 5 1 ...
##  $ class            : num [1:699] 2 2 2 2 2 4 2 2 2 2 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   id = col_double(),
##   ..   clump_thickness = col_double(),
##   ..   size_uniformity = col_double(),
##   ..   shape_uniformity = col_double(),
##   ..   marginal_adhesion = col_double(),
##   ..   epithelial_size = col_double(),
##   ..   bare_nucleoli = col_character(),
##   ..   bland_chromatin = col_double(),
##   ..   normal_nucleoli = col_double(),
##   ..   mitoses = col_double(),
##   ..   class = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
summary(breastcancer_wisconsin)
```

```
##        id           clump_thickness  size_uniformity  shape_uniformity
##  Min.   :   61634   Min.   : 1.000   Min.   : 1.000   Min.   : 1.000
##  1st Qu.:  870688   1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 1.000
##  Median : 1171710   Median : 4.000   Median : 1.000   Median : 1.000
##  Mean   : 1071704   Mean   : 4.418   Mean   : 3.134   Mean   : 3.207
##  3rd Qu.: 1238298   3rd Qu.: 6.000   3rd Qu.: 5.000   3rd Qu.: 5.000
##  Max.   :13454352   Max.   :10.000   Max.   :10.000   Max.   :10.000
##  marginal_adhesion epithelial_size  bare_nucleoli      bland_chromatin
##  Min.   : 1.000    Min.   : 1.000   Length:699         Min.   : 1.000
##  1st Qu.: 1.000    1st Qu.: 2.000   Class :character   1st Qu.: 2.000
##  Median : 1.000    Median : 2.000   Mode  :character   Median : 3.000
##  Mean   : 2.807    Mean   : 3.216                      Mean   : 3.438
##  3rd Qu.: 4.000    3rd Qu.: 4.000                      3rd Qu.: 5.000
##  Max.   :10.000    Max.   :10.000                      Max.   :10.000
##  normal_nucleoli     mitoses           class
##  Min.   : 1.000   Min.   : 1.000   Min.   :2.00
##  1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.:2.00
##  Median : 1.000   Median : 1.000   Median :2.00
##  Mean   : 2.867   Mean   : 1.589   Mean   :2.69
##  3rd Qu.: 4.000   3rd Qu.: 1.000   3rd Qu.:4.00
##  Max.   :10.000   Max.   :10.000   Max.   :4.00
```

d.2 Coefficient of variability for Marginal Adhesion.

```
colnames(breastcancer_wisconsin)
```

```
##  [1] "id"                "clump_thickness"   "size_uniformity"
##  [4] "shape_uniformity"  "marginal_adhesion" "epithelial_size"
##  [7] "bare_nucleoli"     "bland_chromatin"   "normal_nucleoli"
## [10] "mitoses"           "class"
```

```
marginal_adhesion_cv <- sd(breastcancer_wisconsin$`Marginal Adhesion`) / mean(breastcancer_wisconsin$`Ma
```

```
## Warning: Unknown or uninitialised column: `Marginal Adhesion`.
## Unknown or uninitialised column: `Marginal Adhesion`.

## Warning in mean.default(breastcancer_wisconsin$`Marginal Adhesion`, na.rm =
## TRUE): argument is not numeric or logical: returning NA
```

```
marginal_adhesion_cv
```

```
## [1] NA
```

d.3 Number of null values of Bare Nuclei.

```
colnames(breastcancer_wisconsin)
```

```
##  [1] "id"                "clump_thickness"   "size_uniformity"
##  [4] "shape_uniformity"  "marginal_adhesion" "epithelial_size"
##  [7] "bare_nucleoli"     "bland_chromatin"   "normal_nucleoli"
## [10] "mitoses"           "class"
```

```
colnames(breastcancer_wisconsin) <- make.names(colnames(breastcancer_wisconsin))

bare_nuclei_null_count <- sum(is.na(breastcancer_wisconsin$`Bare Nuclei`))
```

```
## Warning: Unknown or uninitialised column: `Bare Nuclei`.
```

```
bare_nuclei_null_count
```

```
## [1] 0
```

d.4 Mean and standard deviation for Bland Chromatin

```
clump_thickness_mean <- mean(breastcancer_wisconsin$clump_thickness)
clump_thickness_sd <- sd(breastcancer_wisconsin$clump_thickness)
clump_thickness_sem <- clump_thickness_sd / sqrt(length(breastcancer_wisconsin$clump_thickness))

clump_thickness_mean
```

```
## [1] 4.41774
```

```
clump_thickness_sd
```

```
## [1] 2.815741
```

```
clump_thickness_sem
```

```
## [1] 0.1065011
```

d.5 Confidence interval of the mean for Uniformity of Cell Shape

```
library(readr)

# Read the CSV file
data <- read_csv("/cloud/project/breastcancer_wisconsin.csv")
```

```
## Rows: 699 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): bare_nucleoli
## dbl (10): id, clump_thickness, size_uniformity, shape_uniformity, marginal_a...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Extract the column of interest
column_of_interest <- data$`Uniformity of Cell Shape`
```

```
## Warning: Unknown or uninitialised column: `Uniformity of Cell Shape`.
```

```
# Remove rows with missing values
column_of_interest_clean <- na.omit(column_of_interest)

# Calculate sample mean, sample size, and sample standard deviation using the cleaned data
sample_mean <- mean(column_of_interest_clean)
```

```
## Warning in mean.default(column_of_interest_clean): argument is not numeric or
## logical: returning NA
```

```
sample_size <- length(column_of_interest_clean)
sample_sd <- sd(column_of_interest_clean)
```

```
# Set the confidence level
confidence_level <- 0.95

# Calculate the margin of error using the t-distribution
margin_of_error <- qt((1 + confidence_level) / 2, df = sample_size - 1) * (sample_sd / sqrt(sample_size
```

```
## Warning in qt((1 + confidence_level)/2, df = sample_size - 1): NaNs produced
```

```
# Calculate the confidence interval
confidence_interval <- c(sample_mean - margin_of_error, sample_mean + margin_of_error)

# Print the results
cat("Sample Mean:", sample_mean, "\n")
```

```
## Sample Mean: NA
```

```
cat("Confidence Interval:", confidence_interval[1], "to", confidence_interval[2], "\n")
```

```
## Confidence Interval: NA to NA
```

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

```
#install.packages("openxlsx")
#library(openxlsx)

#library(MASS)
#data(abalone)


#openxlsx::write.xlsx(abalone, "/cloud/project/RWorksheet_Aguas#4.xlsx", sheetName = "AbaloneData",)
```