RWorksheet_Ceniza#6

Zydrick Ceniza

2023-12-06

Worksheet-6 in R (Individual Activity)

RWorksheet_Ceniza#6

Basic Statistics

1. Create a data frame for the table below. Show your solution.

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

```
num1 <- data.frame(
  ID = c(1, 2, 3, 4, 5,6,7,8,9,10),
  Age = c(55,54,47,57,51,61,57,54,63,58),
  Salary = c(61,60,56,63,56,63,59,56,62,61)
)



num1
```

```
##    ID Age Salary
## 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
```

```
  install.packages("Hmisc")
```

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

```
  install.packages("pastecs")
```

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

```
library(Hmisc)
```

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

```
library(pastecs)


summary_hmisc <- Hmisc::describe(num1)
summary_hmisc
```

```
## num1
##
##  3  Variables      10  Observations
## --------------------------------------------------------------------------------
## ID
##        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
##
## 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
##
## For the frequency table, variable is rounded to the nearest 0
## --------------------------------------------------------------------------------
## Age
##        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
## --------------------------------------------------------------------------------
## Salary
##        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
## --------------------------------------------------------------------------------
```

```
summary_pastecs <- pastecs::stat.desc(num1)
summary_pastecs
```

```
##                      ID          Age       Salary
## 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.

```
data <- c(10, 10, 10, 20, 20, 50, 10, 20, 10, 50, 20, 50, 20, 10)

ordered_factor <- factor(data, levels = c(10, 20, 50), ordered = TRUE)
summary(ordered_factor)
```

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

```
ordered_factor
```

```
##  [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 subjects were “l”, “n”, “n”, “i”, “l” , “l”, “n”, “n”, “i”, “l” ; n=none, l=light, i=intense

a. What is the best way to represent this in R?

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

exercise_factor <- factor(exercise_levels, levels = c("n", "l", "i"), ordered = TRUE)
summary(exercise_factor)
```

```
## n l i
## 4 4 2
```

```
exercise_factor
```

```
##  [1] l n n i l l n n i l
## Levels: n < l < i
```

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 <- 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 <- c("tas", "sa", "qld", "nsw", "nsw", "nt", "wa", "wa", "qld","vic", "nsw", "vic", "qld", "qld",

statef <- factor(state)

statef
```

```
##  [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
```

5. From #4 - continuation: • Suppose we have the incomes of the same tax accountants in another vector (in suitably large units of money)

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)

a. Calculate the sample mean income for each state we can now use the special function tapply(): Example: giving a means vector with the components labelled by the levelsincmeans <- tapply(incomes, statef, mean)

```
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

levelsincmeans <- tapply(incomes, statef, mean)

levelsincmeans
```

```
##      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.

```
state <- c("tas", "sa", "qld", "nsw", "nsw", "nt", "wa", "wa", "qld","vic", "nsw", "vic", "qld", "qld",
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

levelsincmeans <- tapply(incomes, statef, mean)

levelsincmeans
```

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

```
summary(levelsincmeans)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   44.50   53.26   55.25   54.34   56.33   60.50
```

```
The "tas" has the highest mean of 60.50 and "act" that has the shortest mean of 44.50
```

6. Calculate the standard errors of the state income means (refer again to number 3) stdError <- function(x) sqrt(var(x)/length(x)) Note: After this assignment, the standard errors are calculated by: incster <- tapply(incomes, statef, stdError)

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

```
stdError <- function(x) sqrt(var(x) / length(x))

incster <- tapply(incomes, statef, stdError)

incster
```

```
##      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.

```
The functiion compute the desired computaions such as dividing var and length inside the
sqr.   A larger standard error indicates greater variability in the income means for that
state. Lower standard errors suggest more precision in estimating the true population
mean for each state.
```

7.Use the titanic dataset.

a. subset the titatic dataset of those who survived and not survived. Show the codes and its result.

```
titanic<-as.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
```

```
survived <- subset(titanic, Survived == 'Yes')
survived
```

```
##    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
```

```
not_survived <- subset(titanic, Survived == 'No')
not_survived
```

```
##    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 reports his clinical cases. The database therefore reflects this

a. describe what is the dataset all about.

```
The data set is all about is all about the breast cancer diagnosis
```

```r
library(readr)
breastcancer_wisconsin <- read_csv("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.
```

```r
breastcancer_wisconsin
```

```
## # A tibble: 699 x 11
##         id clump_thickness size_uniformity shape_uniformity marginal_adhesion
##      <dbl>           <dbl>           <dbl>            <dbl>             <dbl>
##  1 1000025               5               1                1                 1
##  2 1002945               5               4                4                 5
##  3 1015425               3               1                1                 1
##  4 1016277               6               8                8                 1
##  5 1017023               4               1                1                 3
##  6 1017122               8              10               10                 8
##  7 1018099               1               1                1                 1
##  8 1018561               2               1                2                 1
##  9 1033078               2               1                1                 1
## 10 1033078               4               2                1                 1
## # i 689 more rows
## # i 6 more variables: epithelial_size <dbl>, bare_nucleoli <chr>,
## #   bland_chromatin <dbl>, normal_nucleoli <dbl>, mitoses <dbl>, class <dbl>
```

```r
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>
```

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

```
stdError <- function(x) sqrt(var(x) / length(x))
er_clump_thickness<-stdError(breastcancer_wisconsin$clump_thickness)
er_clump_thickness
```

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

d.2 Coefficient of variability for Marginal Adhesion.

```
coe_marginal_adhesion <- sd(breastcancer_wisconsin$marginal_adhesion) / mean(breastcancer_wisconsin$marg
coe_marginal_adhesion
```

```
## [1] 1.017283
```

d.3 Number of null values of Bare Nuclei.

```
null<-sum(is.na(breastcancer_wisconsin$bare_nucleoli))
null
```

```
## [1] 15
```

d.4 Mean and standard deviation for Bland Chromatin

```
mean_bland_chromatin <- mean(breastcancer_wisconsin$bland_chromatin)
sd_bland_chromatin <- sd(breastcancer_wisconsin$bland_chromatin)
print(paste("Mean:", mean_bland_chromatin, " SD:", sd_bland_chromatin))
```

```
## [1] "Mean: 3.43776824034335  SD: 2.43836425232425"
```

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

```
library(stats)
ci_uniformity_cell_shape <- t.test(breastcancer_wisconsin$shape_uniformity)$conf.int
print(ci_uniformity_cell_shape)
```

```
## [1] 2.986741 3.428138
## attr(,"conf.level")
## [1] 0.95
```

d. How many attributes?

```
ncol(breastcancer_wisconsin)
```

```
## [1] 11
```

e. Find the percentage of respondents who are malignant. Interpret the results.

```
malignant_percentage <- (sum(breastcancer_wisconsin$class == 4) / nrow(breastcancer_wisconsin)) * 100

# Display the result
malignant_percentage
```

```
## [1] 34.47783
```

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

install.packages("AppliedPredictiveModeling") library("AppliedPredictiveModeling") View(abalone) head(abalone) summary(abalone)

save(abalone, file="abalone.csv")