RWorksheet_6

Tamayo

2023-12-23

1.

```
student <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
pre_test <- c(55, 54, 47, 57, 51, 61, 57, 54, 63, 58)
post_test <- c(61, 60, 56, 63, 56, 63, 59, 56, 62, 61)

myd <- data.frame(
  student = student,
  pre_test = pre_test,
  post_test = post_test
)
#print data frame
myd
```

```
##    student pre_test post_test
## 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
```

1b.

```
use_hmsic <- Hmisc::describe(myd)
use_pastecs <- pastecs::stat.desc(myd)

use_hmsic
```

```
## myd
##
##  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
##
## 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
## ---------------------------------------------------------------------------
## pre_test
##        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
## ---------------------------------------------------------------------------
## post_test
##        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
## ---------------------------------------------------------------------------
```

```
use_pastecs
```

```
##                 student     pre_test    post_test
## 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.

```
fertilizer <- c(10,10,10, 20,20,50,10,20,10,50,20,50,20,10)
factFert <- factor(fertilizer)
factFert2 <- as.ordered(factFert)
levels(factFert2)
```

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

```
#print output
fertilizer
```

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

```r
factFert2
```

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

```r
#the result show the levels of the vector.
```

#3.

```r
exerlev <- c("l", "n", "n", "i", "l", "l", "n", "n", "i", "l")

# Create a factor variable with custom levels
exerfact <- factor(exerlev)
levels(exerlev)
```

```
## NULL
```

```r
# Display the result
print(exerfact)
```

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

#4.

```r
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,levels = c("sa", "tas", "qld", "nsw", "wa", "vic", "act", "nt"))
levels(statefactor)
```

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

```r
#the result shows the levels of the vector. It summarizes what is the content of the vector.
```

#5a.

```r
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)
incomeans <- tapply(incomes, statefactor, mean)
incomeans
```

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

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

#it provides the average incomes for each state.
```

6.

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

stdErrors <- tapply(incomes, statefactor, stdError)
stdErrors
```

```
##       sa      tas      qld      nsw       wa      vic      act       nt
```

```
## 2.738613 0.500000 4.106093 4.310195 2.657536 5.244044 1.500000 4.500000
```

#7.Use the titanic dataset.

```
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")

survived <- subset(titanic_train, Survived == 1)

not_survived <- subset(titanic_train, Survived == 0)

head(survived)
```

```
##    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
```

```
head(not_survived)
```

```
##    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.

a. describe what is the dataset all about. The dataset employs a survey scale with a range of 1 to 10 and is centered on women who are coping with breast cancer. Cluster thickness, size uniformity, shape uniformity, and other properties of cell nuclei seen in breast cancer are evaluated using this scale. normal nucleoli, bland chromatin, bare nucleoli, epithelial size, marginal adhesion, and mitoses. The severity or abnormality of each attribute is reflected in the score on the scale. In order to obtain insight into the type of breast cancer that has affected the women polled, the dataset attempts to collect and examine these features.

```
#d. Compute the descriptive statistics using different packages. Find the values of:
# d.1 Standard error of the mean for clump thickness.
library(readr)

breast_wisconsin <- read_csv("/cloud/project/Worksheet#4/Worksheet#6/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.
```

```
head(breast_wisconsin)
```

```
## # A tibble: 6 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
## # i 6 more variables: epithelial_size <dbl>, bare_nucleoli <chr>,
## #   bland_chromatin <dbl>, normal_nucleoli <dbl>, mitoses <dbl>, class <dbl>
```

```
clump_column <- breast_wisconsin$clump_thickness
std_error <- sd(clump_column) / sqrt(length(clump_column))

print(std_error)
```

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

#d.2 Coefficient of variability for Marginal Adhesion.

```
marginal_column <- breast_wisconsin$marginal_adhesion
coefficient_variability <- sd(marginal_column) / mean(marginal_column) * 100

print(coefficient_variability)
```

```
## [1] 101.7283
```

#d.3 Number of null values of Bare Nuclei.

```
barenucleoli_column <- breast_wisconsin$bare_nucleoli
nullvalues_count <- sum(is.na(barenucleoli_column))
```

```
print(nullvalues_count)
```

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

#d.4 Mean and standard deviation for Bland Chromatin

```
mean_bland_chromatin <- mean(breast_wisconsin$bland_chromatin, )
sd_bland_chromatin <- sd(breast_wisconsin$bland_chromatin, )

print(paste("Mean:", mean_bland_chromatin))
```

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

```
print(paste("Standard deviation:", sd_bland_chromatin))
```

```
## [1] "Standard deviation: 2.43836425232425"
```

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

```
shape_uniformity <- breast_wisconsin$shape_uniformity

result <- t.test(shape_uniformity)

cat("Mean:", result$estimate, "\n")
```

```
## Mean: 3.207439
```

```
cat("95% confidence interval:", result$conf.int[1], result$conf.int[2], "\n")
```

```
## 95% confidence interval: 2.986741 3.428138
```

#d. How many attributes?

```
num_attributes <- length(names(breast_wisconsin))
print(num_attributes)
```

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

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

```
malignant_count <- sum(breast_wisconsin$class == "malignant")
total_count <- nrow(breast_wisconsin)

percentage_malignant <- (malignant_count / total_count) * 100
print(percentage_malignant)
```

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

9.

```
install.packages("AppliedPredictiveModeling")
```

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

```
library("AppliedPredictiveModeling")

data("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
```

```
library(openxlsx)

write.xlsx(abalone, file = "abalone.xlsx")
```