RWorksheet_Lumahan#5

2023-12-22

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

```
studentScores <- data.frame(
  Student = c(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)
)

studentScores
```

```
##    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
```

```
names(studentScores) <- c("Student", "PreTest", "PostTest")
```

1a.

```
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)

# Hmisc
describe(studentScores)
```

```
## studentScores
```

```
## 
##  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
## --------------------------------------------------------------------------------
## 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
## --------------------------------------------------------------------------------
```

```
# pastecs
stat.desc(studentScores)
```

```
##                  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 are 10,10,10, 20,20,50,10,20,10,50,20,50,20,10.

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

orderedLevels <- ordered(fertilizerLevels, levels = c(10, 20, 50))

orderedLevels
```

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

```
# The numbers inside the square brackets represent the observations or data points and below it are the
```

3. Abdul Hassan, president of Floor Coverings Unlimited, has asked you to study the ex- ercise 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?

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

factorExercise <- factor(exerciseLevels, levels = c("n", "l", "i"))

factorExercise
```

```
##  [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 

states <- 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")

factorState <- factor(states, levels = c("act", "nsw", "nt", "qld", "sa", "tas", "vic", "wa"))

factorState
```

```
##  [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
```

```
#the number inside the square brackets are the observations and below it are the levels. the levels rep
```

5. From #4 - continuation:

```
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

incMeans <- tapply(incomes, factorState, mean)
incMeans
```

```
##      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.

# we see that it calculates the means of every states.
```

6. Calculate the standard errors of the state income means (refer again to number 5)

```
stdError <- function(x) sqrt(var(x) / length(x))
incStdErr <- tapply(incomes, factorState, stdError)
incStdErr
```

```
##      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.
# in no.5 we see the means of every states while here, we calculate the standard error of each states.

# the standard errors provide a measure of the uncertainty associated with the sample mean incomes for
```

7.

```
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)
notSurvived <- 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(notSurvived)
```

```
##    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
```

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

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

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

```
library(AppliedPredictiveModeling)

data("abalone")

install.packages("openxlsx")
```

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

```
library(openxlsx)

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

View(abalone)
```

```
## Warning in View(abalone): unable to open display
```

```
## Error in .External2(C_dataviewer, x, title): unable to start data viewer
```

```
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
```