RWorksheet_de la Cruz-Hanz#6

2023-12-06

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
studentScoreDF <- data.frame(
      Student = 1:10,
      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))

studentScoreDF
```

```
##    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
```

```
library(Hmisc)
```

```
## Warning: package 'Hmisc' was built under R version 4.3.2
```

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

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

```
HmiscStudentDF <- describe(studentScoreDF)
HmiscStudentDF
```

```
## studentScoreDF
##
##  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
## --------------------------------------------------------------------------------
```

```
library(pastecs)
```

```
## Warning: package 'pastecs' was built under R version 4.3.2
```

```
pastecsStudentDF <- stat.desc(studentScoreDF)
pastecsStudentDF
```

```
##                 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
```

```
fertilizerLevel <- c(10, 10, 10, 20, 20, 50, 10, 20, 10, 50, 20, 50, 20, 10)
orderedFertilizer <- factor(fertilizerLevel, ordered = TRUE)
orderedFertilizer
```

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

```
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
```

```
subjects <- c("Subject1", "Subject2", "Subject3", "Subject4", "Subject5",
              "Subject6", "Subject7", "Subject8", "Subject9", "Subject10")

exerciseLabels <- c("light", "none", "none", "intense", "light",
                     "light", "none", "none", "intense", "light")
exerciseDF <- data.frame(Subject = subjects, Exercise_Level = exerciseLevels, Exercise_Label = exercise
exerciseDF
```

```
##      Subject Exercise_Level Exercise_Label
## 1   Subject1              l          light
## 2   Subject2              n           none
## 3   Subject3              n           none
## 4   Subject4              i        intense
## 5   Subject5              l          light
## 6   Subject6              l          light
## 7   Subject7              n           none
## 8   Subject8              n           none
## 9   Subject9              i        intense
## 10 Subject10              l          light
```

```
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("act", "nsw", "nt", "qld", "sa", "tas", "vic", "wa"))

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
```

```
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)

meanIncomes <- tapply(incomes,stateFactor, mean)

meanIncomes
```

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

```
sort(meanIncomes)
```

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

```
cat("\ntas has the most average income and act has the least. nt and sa share the same mean and are both
```

```
##
## tas has the most average income and act has the least. nt and sa share the same mean and are both in
```

```
statef <- factor(state, levels = c("act", "nsw", "nt", "qld", "sa", "tas", "vic", "wa"))

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
```

```
sort(incster)
```

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

```
cat("tas has the least standard error while vic has the most standard error for the income means")
```

```
## tas has the least standard error while vic has the most standard error for the income means
```

```
data('Titanic')
```

```
titanic_df <- as.data.frame(Titanic)

survivors <- subset(titanic_df, Survived == "Yes")

didNotSurvive <- subset(titanic_df, Survived == "No")

survivors
```

```
##    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
```

```
didNotSurvive
```

```
##    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
```

```
breastcancerDF <- read.csv("breastcancer_wisconsin.csv")
str(breastcancerDF)
```

```
## ’data.frame’:    699 obs. of  11 variables:
##  $ id               : int  1000025 1002945 1015425 1016277 1017023 1017122 1018099 1018561 1033078 10
##  $ clump_thickness  : int  5 5 3 6 4 8 1 2 2 4 ...
##  $ size_uniformity  : int  1 4 1 8 1 10 1 1 1 2 ...
##  $ shape_uniformity : int  1 4 1 8 1 10 1 2 1 1 ...
##  $ marginal_adhesion: int  1 5 1 1 3 8 1 1 1 1 ...
##  $ epithelial_size  : int  2 7 2 3 2 7 2 2 2 2 ...
##  $ bare_nucleoli    : chr  "1" "10" "2" "4" ...
##  $ bland_chromatin  : int  3 3 3 3 3 9 3 3 1 2 ...
##  $ normal_nucleoli  : int  1 2 1 7 1 7 1 1 1 1 ...
##  $ mitoses          : int  1 1 1 1 1 1 1 1 5 1 ...
##  $ class            : int  2 2 2 2 2 4 2 2 2 2 ...
```

```r
summary(breastcancerDF)
```

```
##        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
```

```r
cat("this csv file contains important information about the breast cancer and its qualities")
```

```
## this csv file contains important information about the breast cancer and its qualities
```

```r
View(breastcancerDF)
```

```r
# d.1 Standard error of the mean for clump thickness
clumpThickness <- sd(breastcancerDF$clump_thickness) / sqrt(length(breastcancerDF$clump_thickness))
cat("Standard error for clump thickness:", clumpThickness, "\n")
```

```
## Standard error for clump thickness: 0.1065011
```

```r
# d.2 Coefficient of variability for Marginal Adhesion
marginalAdhesionCV <- sd(breastcancerDF$marginal_adhesion) / mean(breastcancerDF$marginal_adhesion) * 1
cat("Coefficient of variability for Marginal Adhesion:", marginalAdhesionCV, "\n")
```

```
## Coefficient of variability for Marginal Adhesion: 101.7283
```

```r
# d.3 Number of null values of Bare Nuclei
nullValuesBareNuclei <- sum(is.na(breastcancerDF$bare_nucleoli))
cat("Number of null values for Bare Nuclei:", nullValuesBareNuclei, "\n")
```

```
## Number of null values for Bare Nuclei: 15
```

```r
# d.4 Mean and standard deviation for Bland Chromatin
meanBlandChromatin <- mean(breastcancerDF$bland_chromatin)
sdBlandChromatin <- sd(breastcancerDF$bland_chromatin)
cat("Mean of Bland Chromatin:", meanBlandChromatin, "\n")
```

```
## Mean of Bland Chromatin: 3.437768
```

```r
cat("Standard Deviation of Bland Chromatin:", sdBlandChromatin, "\n")
```

```
## Standard Deviation of Bland Chromatin: 2.438364
```

```r
# d.5 Confidence interval of the mean for Uniformity of Cell Shape
uniformityOfCellShapeCI <- t.test(breastcancerDF$shape_uniformity)$conf.int
cat("Confidence interval of the mean for Uniformity of Cell Shape:", uniformityOfCellShapeCI, "\n")
```

```
## Confidence interval of the mean for Uniformity of Cell Shape: 2.986741 3.428138
```

```r
# Number of attributes
breastCancerDFattributes <- ncol(breastcancerDF)
cat("Number of attributes for breast cancer dataframe:", breastCancerDFattributes, "\n")
```

```
## Number of attributes for breast cancer dataframe: 11
```

```r
# e. Percentage of respondents who are malignant
malignantPercentage <- round((sum(breastcancerDF$class == 4) / nrow(breastcancerDF)) * 100, 2)
cat("Percentage of respondents who are malignant:", malignantPercentage, "%\n")
```

```
## Percentage of respondents who are malignant: 34.48 %
```

```r
#install.packages("AppliedPredictiveModeling")
#install.packages("openxlsx")
```

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

```
## Warning: package ’AppliedPredictiveModeling’ was built under R version 4.3.2
```

```r
library(openxlsx)
```

```
## Warning: package ’openxlsx’ was built under R version 4.3.2
```

```r
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  
```

```
write.xlsx(abalone, "abalone.xlsx", row.names = FALSE)
```

```
## Warning: Please use ’rowNames’ instead of ’row.names’
```