RWORKSHEET_6

RcCatedral

2023-12-09

```
                  #Basic Statistics
```

#1

```r
data <- read.csv("StudentScore.csv")
print(data)
```

```
##    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
```

```r
#install.packages("Hmisc")
#install.packages("pastecs")
library(Hmisc)
```

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

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

```r
library(pastecs)

data <- read.csv("StudentScore.csv")
numeric_columns <- sapply(data, is.numeric)
hdesc_stats <- describe(data[, numeric_columns])
pastecs_stats <- stat.desc(data[, numeric_columns])

cat("Descriptive Statistics using Hmisc:\n")
```

```
## Descriptive Statistics using Hmisc:
```

```r
print(hdesc_stats)
```

```
## data[, numeric_columns]
##
##  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
## --------------------------------------------------------------------------------
```

```
cat("\nDescriptive Statistics using pastecs:\n")
```

```
##
## Descriptive Statistics using pastecs:
```

```
print(pastecs_stats)
```

```
##                  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

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

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

```
#The code creates an ordered factor variable named fertilizer_factor from a numeric vector data represe
```

#3

```
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)
print(exercise_factor)
```

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

#4

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

state_factor
```

```
##  [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 output indicates that a factor variable called state_factor, which represents the state of origin
```

#5

```
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)
incmeans <- tapply(incomes, state_factor, mean)
print(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
```

```
#THE RESULT IS
#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 average incomes for tax accountants in different states are as follows: Australian Capital Territo
```

#6

```
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)

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

```r
print(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
```

```r
#The standard errors of the mean incomes for tax accountants in different states are as follows: ACT (A
```