Introduccón al manejo de datos con R

Descripción General

Este documento proporciona una introducción al manejo de datos con R. A lo largo del tutorial, exploraremos diferentes aspectos del manejo de datos utilizando ejemplos prácticos.

Los siguientes datos indican el contenido de nitrógeno obtenido en 4 lagunas pampeanas el mes pasado (en µg/L):

5051.2, 6193.6, 3684.8 4928

1. Construya el vector con los datos, asignándole el nombre nitro

```
nitro <- c(5051.2, 6193.6, 3684.8, 4928)
```

2. Verifique que el tipo de datos ingresados sea numérico.

```
class(nitro)
```

```
## [1] "numeric"
```

3. Verifique que la cantidad de datos ingresados sea la correcta

```
length(nitro)
```

```
## [1] 4
```

4. Visualice el vector

```
nitro
```

```
## [1] 5051.2 6193.6 3684.8 4928.0
```

5. Se desea expresar el contenido de nitrógeno en mg/L. Genere el vector correspondiente y denomínelo Nmg, para esto dividimos el vector por 1000 con el siguiente script

```
Nmg <- nitro/1000
Nmg
```

```
## [1] 5.0512 6.1936 3.6848 4.9280
```

6. Se quiere identificar cada laguna con un número secuencial (de 1 a 4). Genere el vector correspondiente, nombrándolo Id

```r
id <- c(1:4)
id
```

```
## [1] 1 2 3 4
```

7. Los datos correspondieron a las lagunas Chascomús, Chis-Chis, El Burro y Adela respectivamente. Genere el vector laguna

```r
laguna <- c("Chascomus", "Chris-Chris","El Burro","Adela")
laguna
```

```
## [1] "Chascomus"   "Chris-Chris" "El Burro"    "Adela"
```

8. Construya el dataframe bd con el Id, el nombre de la laguna y el contenido de N en mg/L uniendo los vectores.

```r
bd <- data.frame(id,laguna,Nmg)
bd
```

```
##   id      laguna    Nmg
## 1  1   Chascomus 5.0512
## 2  2 Chris-Chris 6.1936
## 3  3    El Burro 3.6848
## 4  4       Adela 4.9280
```

9. Calcule el promedio del contenido de nitrógeno

```r
promedio <- mean(bd$Nmg)
promedio
```

```
## [1] 4.9644
```

Utilizando el dataframe bd

10. Seleccione el contenido de N correspondiente a la laguna Chis-Chis y Adela

```r
chris = subset(bd, laguna == "Chris-Chris" )
chris
```

```
##   id      laguna    Nmg
## 2  2 Chris-Chris 6.1936
```

11. Seleccione las lagunas con contenido de N superior a 5 mg/L

```r
n_5 = subset(bd, Nmg > 5)
n_5
```

```
##   id      laguna    Nmg
## 1  1   Chascomus 5.0512
## 2  2 Chris-Chris 6.1936
```

12. Seleccione los datos correspondientes a la laguna Adela

```r
Adela = subset(bd, laguna == "Adela" )
Adela
```

```
##   id laguna   Nmg
## 4  4  Adela 4.928
```

Instale el paquete faraway

```r
#install.packages(faraway)
library(faraway)
```

```
## Warning: package 'faraway' was built under R version 4.2.3
```

13. Explore la base de datos pima contenida en dicho paquete.

```r
# Cargamos la base de datos 'pima' contenida en el paquete 'faraway'.
pima <- faraway::pima
head(pima)
```

```
##   pregnant glucose diastolic triceps insulin  bmi diabetes age test
## 1        6     148        72      35       0 33.6    0.627  50    1
## 2        1      85        66      29       0 26.6    0.351  31    0
## 3        8     183        64       0       0 23.3    0.672  32    1
## 4        1      89        66      23      94 28.1    0.167  21    0
## 5        0     137        40      35     168 43.1    2.288  33    1
## 6        5     116        74       0       0 25.6    0.201  30    0
```

- ¿Cuáles y cuántas variables posee?

```r
ncol(pima) # Número de variables
```

```
## [1] 9
```

```r
names(pima) # Nombres de las columnas
```

```
## [1] "pregnant"  "glucose"   "diastolic" "triceps"   "insulin"   "bmi"
## [7] "diabetes"  "age"       "test"
```

- ¿De qué tipo son?

```r
sapply(pima, class)
```

```
##  pregnant   glucose diastolic   triceps   insulin       bmi  diabetes       age
## "integer" "integer" "integer" "integer" "integer" "numeric" "numeric" "integer"
##      test
## "integer"
```

- ¿Cuántos casos?

```r
nrow(pima) # Número de casos
```

```
## [1] 768
```

*¿Hay datos faltantes?

```r
colSums(is.na(pima)) #suma los datos faltantes de cada columna
```

```
##  pregnant   glucose diastolic   triceps   insulin       bmi  diabetes       age
##         0         0         0         0         0         0         0         0
##      test
##         0
```

14. ¿Para qué se utiliza la función summary(bd)? Aplíquela

La función summary() en R se utiliza para obtener un resumen estadístico de un objeto, como un dataframe, una matriz o un vector. El resumen proporciona información útil sobre las estadísticas descriptivas de los datos en el objeto. El propósito principal de summary() es proporcionar un vistazo rápido a los datos, especialmente cuando trabajas con conjuntos de datos grandes o complejos

```r
summary(pima)
```

```
##     pregnant         glucose        diastolic         triceps
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00
##  Median : 3.000   Median :117.0   Median : 72.00   Median :23.00
##  Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54
##  3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00
##     insulin           bmi           diabetes           age
##  Min.   :  0.0   Min.   : 0.00   Min.   :0.0780   Min.   :21.00
##  1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437   1st Qu.:24.00
##  Median : 30.5   Median :32.00   Median :0.3725   Median :29.00
##  Mean   : 79.8   Mean   :31.99   Mean   :0.4719   Mean   :33.24
##  3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00
##  Max.   :846.0   Max.   :67.10   Max.   :2.4200   Max.   :81.00
##       test
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.349
##  3rd Qu.:1.000
##  Max.   :1.000
```

15. Analice el rango de las variables. ¿Detecta alguna inconsistencia?

Las variables glucose, diastolic, triceps, insulin y bmi tienen casos con el valor 0, sabemo que eso no es posible, vamos a reemplazar esos valores por NA.

```r
#Cambio los ceros por NA

pima$glucose[pima$glucose == 0] <- NA
pima$diastolic[pima$diastolic == 0] <- NA
pima$triceps[pima$triceps == 0] <- NA
pima$insulin[pima$insulin == 0] <- NA
pima$bmi[pima$bmi == 0] <- NA
```

16. El índice de masa corporal (bmi) es el peso de una persona (en kg) dividido por el cuadrado de la altura (en m). Si bmi está entre 25 a <30 se considera con sobrepeso. Si su IMC es 30.0 o superior, obesidad. Genere la variable cat_peso. Clasifique a las pacientes según su peso en normales, con sobrepeso o con obesidad. Ayuda: explore la función ifelse

```r
pima$cat_peso <- ifelse(pima$bm < 25, "Normal",
                 ifelse(pima$bm >= 25 & pima$bm < 30, "Con sobrepeso",
                 "Obesidad"))
```

17. Genere la variable diabetes a partir de test (sí = 1, no=0)

```r
pima$result <- ifelse(pima$test == 1, "si","no")

colnames(pima)[7]<-"diabetes_gen"

head(pima)
```

```
##   pregnant glucose diastolic triceps insulin  bmi diabetes_gen age test
## 1        6     148        72      35      NA 33.6        0.627  50    1
## 2        1      85        66      29      NA 26.6        0.351  31    0
## 3        8     183        64      NA      NA 23.3        0.672  32    1
## 4        1      89        66      23      94 28.1        0.167  21    0
## 5        0     137        40      35     168 43.1        2.288  33    1
## 6        5     116        74      NA      NA 25.6        0.201  30    0
##        cat_peso result
## 1      Obesidad     si
## 2 Con sobrepeso     no
## 3        Normal     si
## 4 Con sobrepeso     no
## 5      Obesidad     si
## 6 Con sobrepeso     no
```