

```
# Las expresiones siempre deben ser LO MAS GENERALES POSIBLES.  
# Por ejemplo, si pido el último elemento de un vector VECT,  
debería escribir la expresión ULTIMO = VECT[length(VECT)] por más que conozcamos su longitud.
```

```
# VECTORES Y MATRICES
```

```
# Generar un vector V con 12 valores aleatorios entre 1 y 100.  
set.seed(99)
```

```
V
```

```
# EJEMPLO
```

```
# [1] 48 33 44 22 62 32 13 20 31 68 9 82
```

```
# Generar una matriz M de 3x4 con los valores contenidos en V colocados POR FILA en M.
```

```
M
```

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

```
# [1,] 48 33 44 22
```

```
# [2,] 62 32 13 20
```

```
# [3,] 31 68 9 82
```

```
# Con una unica EXPRESION GENERAL, QUE SIRVA PARA MATRICES DE CUALQUIER TAMAÑO, mostrar la  
ultima columna de M.
```

```
# [1] 22 20 82
```

```
# Obtener el máximo elemento impar de M.
```

```
# 33
```

```
# Calcular el promedio de cada fila de M.
```

```
# [1] 36.75 31.75 47.50
```

```
# Obtener el máximo entre las sumatorias de los elementos de cada columna.
```

```
# [1] 11 141
```

```
# Multiplicar por 2 los elementos de M, exepcto los elementos de la segunda columna de M.
```

```
M
```

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

```
# [1,] 96 33 88 44
```

```
# [2,] 124 32 26 40
```

```
# [3,] 62 68 18 164
```

```
# Armar un vector V2 con los elementos de la primera y segunda filas de M.
```

```
V2
```

```
# [1] 96 33 88 44 124 32 26 40
```

```
# Ver los elementos de V2 que son múltiplos de 3.
```

```
# 96 33
```

```
# Obtener el mayor número par de V2.
```

```
# 124
```

```
# Modificar a V2 reemplazando por 0 todos los elementos que no son múltiplos de 3.
```

```
V2
```

```
# [1] 96 0 88 44 124 32 0 40
```

```
# Contar cuántos elementos en V2 son múltiplos de 2 y mayores que 40.
```

```
# [1] 4
```

```
##### DATA FRAME
```

```
## Ejecutar las siguientes expresiones:
```

```
library(ggplot2)
```

```
data(msleep)
```

```
suenio = subset(data.frame(msleep), select = c(name, vore, sleep_total, bodywt))
```

```
suenio = na.omit(suenio)
```

```
eliminadas = na.action(suenio)
```

```
suenio$vore = as.factor(suenio$vore)
```

```
dim(suenio)
```

```
# [1] 76 4
```

```
# Se obtiene un sub-conjunto con 76 observaciones, y para cada una se informa el nombre de la especie, el tipo de alimentación, el tiempo total de horas que duerme y el peso en kilos.
```

```
# Al observar las primeras filas ejecutando head(suenio) debería aparecer la siguiente sub-tabla:
```

```
#           name  vore sleep_total bodywt
# 1       Cheetah  carni    12.1  50.000
# 2       Owl monkey  omni    17.0   0.480
# 3 Mountain beaver  herbi    14.4   1.350
# ...
```

```
# Generar un subconjunto a partir de "suenio" que contenga solo a los animales carnívoros, y que no tenga la columna "vore". Este nuevo conjunto se llamara "CARNIS".
```

```
head(CARNIS)
```

```
#           name sleep_total bodywt
# 1       Cheetah    12.1  50.00
# 7 Northern fur seal     8.7  20.49
# ...
```

```
# Escribir una expresión que permita calcular cuantos de los "CARNIS" pesan mas de 10 kg y duermen entre 5 y 10 horas.
```

```
# El resultado debería dar 4.
```

```
# Generar un subconjunto a partir de "suenio" que contenga solamente los animales que pesan más de 100 kilos, y duermen menos de 10 horas. Este nuevo conjunto se llamara "GRANDES".
```

```
GRANDES
```

```
#           name vore sleep_total  bodywt
```

```
# 5      Cow herbi      4.0 600.000
# 21     Asian elephant herbi      3.9 2547.000
# 23      Horse herbi      2.9 521.000
...
```

Escribir una expresión que permita calcular cuantos de los "GRANDES" son carnívoros. El resultado debería dar 2.

Ver el resumen de cuántos hay en "suenio" con cada tipo de alimentación.

```
# carni  herbi insecti  omni
# 19     32     5       20
```

Escribir una expresión para obtener el nombre del animal que menos duerme de todos los animales contenidos en el conjunto de datos suenio. Puede hacerlo en varios pasos.

El resultado debería ser "Giraffe".

Escribir una expresión para obtener el nombre y forma de alimentación del animal más pesado de todos los animales contenidos en "GRANDES". Puede hacerlo en varios pasos. El resultado debería ser:

```
# name     vore
# 36 African elephant herbi
```

Calcular el promedio de peso de los animales del conjunto original agrupados por su tipo de alimentación.

```
# # carni  herbi insecti  omni
# 90.75111 366.87725 12.92160 12.71800
```

Armar un gráfico de torta donde se vea la proporción de animales que tienen cada tipo de alimentación en el conjunto de datos "suenio".

Armar un subconjunto de "suenio", eliminando las filas correspondientes a "African elephant" y "Asian elephant"; llamarlo suenio2.

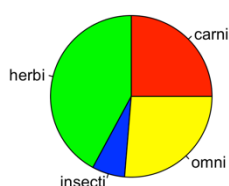
Armar un gráfico de dispersión donde se vea la relación entre el peso de los animales y la cantidad de horas que duermen. Los datos deben venir del conjunto de datos "suenio2".

Agregar a este último gráfico una línea roja horizontal, en el valor promedio de peso.

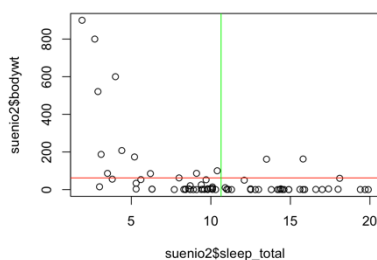
Agregar a este último gráfico una línea verde vertical, en el promedio de horas dormidas.

Armar un gráfico de cajas para ver cuántas horas duermen agrupados por el tipo de alimentación. ¿Quiénes son los más dormilones?

Alimentacion: proporciones



Peso vs. horas de sueño



¿Duermen según lo que comen?

