R Notebook

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*. 1. Introduction to Python for Machine Learning (a) Create an array of 10 random integers between 1 and 100, and then calculate the mean and standard deviation of the array

```r
# Generar una matriz de 10 enteros aleatorios entre 1 y 100
matriz <- sample(1:100, 10, replace = TRUE)

# Calcular la media de la matriz
media <- mean(matriz)

# Calcular la desviación estándar de la matriz
desviacion_standar <- sd(matriz)

# Imprimir los resultados
print(paste("Matriz:", matriz))
```

```
##  [1] "Matriz: 15" "Matriz: 13" "Matriz: 16" "Matriz: 79" "Matriz: 57"
##  [6] "Matriz: 53" "Matriz: 85" "Matriz: 11" "Matriz: 96" "Matriz: 93"
```

```r
print(paste("Media:", media))
```

```
## [1] "Media: 51.8"
```

```r
print(paste("Desviación estándar:", desviacion_standar))
```

```
## [1] "Desviación estándar: 35.477066144514"
```

(b) Create a 2-dimensional array of 3 rows and 4 columns with random integer values. Then, calculate the sum of each row and column.

```r
# Generar una matriz bidimensional de 3 filas y 4 columnas con valores enteros aleatorios
matriz <- matrix(sample(1:100, 12, replace = TRUE), nrow = 3, ncol = 4)

# Calcular la suma de cada fila
sumas_filas <- rowSums(matriz)

# Calcular la suma de cada columna
sumas_columnas <- colSums(matriz)

# Imprimir la matriz y las sumas
print("Matriz:")
```

```
## [1] "Matriz:"
```

```r
print(matriz)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]   78    8   17   25
## [2,]   73   96   72   73
## [3,]   37   66   72   37
```

```r
print("Sumas de filas:")
```

```
## [1] "Sumas de filas:"
```

```r
print(sumas_filas)
```

```
## [1] 128 314 212
```

```r
print("Sumas de columnas:")
```

```
## [1] "Sumas de columnas:"
```

```r
print(sumas_columnas)
```

```
## [1] 188 170 161 135
```

(c) Create a DataFrame with 3 columns: "Name", "Age", and "City". Add at least 5 rows of data to the DataFrame. Then, filter the DataFrame to only include rows where the person's age is greater than or equal to 30.

```r
# Cargar la biblioteca "dplyr" para trabajar con DataFrames
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Crear el DataFrame
df <- data.frame(
  Name = c("John", "Alicia", "Macarena", "Sarah", "David"),
  Age = c(25, 32, 41, 28, 35),
  City = c("New York", "London", "Paris", "Sydney", "Tokyo")
)

# Filtrar el DataFrame para incluir solo las filas donde la edad es mayor o igual a 30
df_filtered <- df %>%
  filter(Age >= 30)

# Imprimir el DataFrame filtrado
print(df_filtered)
```

```
##       Name Age   City
## 1   Alicia  32 London
## 2 Macarena  41  Paris
## 3    David  35  Tokyo
```

(d) Load a CSV file into a DataFrame and then calculate the mean, median, and mode of one of the columns in the DataFrame.

```
dframe <-read.csv("C:/Proyectos ML/Mineria/Hands_On_1/Codificado R/datos/dataset-29460.csv")
  dato_limpio <- na.omit(dframe)
  dato <- complete.cases(dframe)


 media <- mean(dato_limpio)
```

```
## Warning in mean.default(dato_limpio): argument is not numeric or logical:
## returning NA
```

```
 print(paste("",media))
```

```
## [1] " NA"
```