Tipos y Estructuras de Datos

D. S. Fernández del Viso

2024-12-15

## Tipos y Estructuras de Datos en R

En R, los datos se pueden clasificar en diferentes tipos y estructuras. En este documento, veremos ejemplos de tipos de datos básicos y estructuras de datos más complejas, como vectores, matrices, data frames, listas y factores.

### **1. Tipos de Datos en R**

#### **Ejemplo: Tipos básicos**

# Números (numeric)  
numero <- 42.5  
numero  
class(numero) # "numeric"  
  
# Enteros (integer)  
entero <- as.integer(10)  
entero  
class(entero) # "integer"  
  
# Caracteres (character)  
texto <- "Hola, mundo"  
texto  
class(texto) # "character"  
  
# Lógicos (logical)  
logico <- TRUE  
logico  
class(logico) # "logical"  
  
# Complejos (complex)  
complejo <- 3 + 2i  
complejo  
class(complejo) # "complex"  
  
# Fechas (Date)  
fecha <- as.Date("2021-10-01")  
fecha  
class(fecha) # "Date"  
  
# Hora solamente (POSIXct)  
hora <- as.POSIXct("12:00:00", format = "%H:%M:%S")  
hora  
class(hora) # "POSIXct"  
  
# Fecha con hora (POSIXct)  
fecha\_hora <- as.POSIXct("2021-10-01 12:00:00")  
fecha\_hora  
class(fecha\_hora) # "POSIXct"

#### **Ejercicio: Identificar tipos de datos**

1. Crea las siguientes variables y verifica su tipo con class():
   * Una variable llamada edad con el valor 25.
   * Una variable llamada nombre con el texto "Ana".
   * Una variable llamada es\_estudiante con el valor FALSE.

### **2. Estructuras de Datos en R**

#### **a. Vectores**

Un vector es una colección de elementos del mismo tipo.

# Crear un vector numérico  
numeros <- c(1, 2, 3, 4, 5)  
  
# Crear un vector de caracteres  
nombres <- c("Juan", "Ana", "Luis")  
  
# Operaciones con vectores  
suma <- numeros + 2 # Suma 2 a cada elemento

**Ejercicio:**  
Crea un vector llamado temperaturas con los valores: 23, 25, 21, 19, 22.  
1. Calcula el promedio de las temperaturas.  
2. Encuentra los valores mayores a 22.

#### **b. Matrices**

Las matrices son arreglos bidimensionales con elementos del mismo tipo.

# Crear una matriz  
matriz <- matrix(1:9, nrow = 3, ncol = 3)  
  
# Acceder a elementos  
elemento <- matriz[2, 3] # Fila 2, columna 3  
  
# Operaciones con matrices  
matriz\_transpuesta <- t(matriz) # Transponer la matriz

**Ejercicio:**  
1. Crea una matriz de 3 filas y 4 columnas con los números del 1 al 12.  
2. Extrae la segunda fila completa.

#### **c. Data Frames**

Un data frame es una tabla donde cada columna puede tener un tipo diferente de datos.

# Crear un data frame  
pacientes <- data.frame(  
 Nombre = c("Juan", "Ana", "Luis"),  
 Edad = c(25, 30, 35),  
 Diabetes = c(TRUE, FALSE, TRUE)  
)  
  
# Acceder a columnas  
edades <- pacientes$Edad  
  
# Filtrar datos  
diabeticos <- subset(pacientes, Diabetes == TRUE)

**Ejercicio:**  
1. Crea un data frame con los datos de 3 estudiantes: Nombre, Nota (número) y Aprobado (TRUE/FALSE).  
2. Filtra los estudiantes que aprobaron.

#### **d. Listas**

Las listas pueden contener elementos de diferentes tipos.

# Crear una lista  
mi\_lista <- list(  
 vector = c(1, 2, 3),  
 texto = "Hola",  
 matriz = matrix(1:4, nrow = 2)  
)  
  
# Acceder a elementos  
mi\_lista$texto  
mi\_lista$matriz[1, 2]

**Ejercicio:**  
1. Crea una lista llamada info que contenga: - Un vector con los números del 1 al 5. - Un texto con tu nombre. - Una matriz de 2x2 con los valores del 1 al 4.

#### **e. Factores**

Los factores representan datos categóricos.

# Crear un factor  
genero <- factor(c("Masculino", "Femenino", "Femenino", "Masculino"))  
  
# Ver niveles  
levels(genero)

**Ejercicio:**  
1. Crea un factor llamado estado\_civil con los valores: Soltero, Casado, Divorciado.  
2. Cambia el orden de los niveles para que Casado sea el primero.

### **3. Operaciones con los Datos**

#### **Ejercicio 1: Operaciones con Datos Numéricos**

Crear las variables var1 y var2 con valores numéricos y realizar las siguientes operaciones: - Suma: var1 + var2 - Resta: var1 - var2 - Multiplicación: var1 \* var2 - División: var1 / var2 - Potencia: var1 ^ var2 - Raíz cuadrada de var1 y var2 Usar sqrt() - Logaritmo natural de var1 y var2 Usar log() - Valor absoluto de var1 / var2 Usar abs()

#### **Ejercicio 2: Operaciones con Datos de Texto**

Crear las variables texto1 y texto2 con valores de texto (frases cortas) y realizar las siguientes operaciones: - Concatenar: paste(texto1, texto2) - Número de caracteres: nchar(texto1) - Convertir a mayúsculas: toupper(texto1) - Convertir a minúsculas: tolower(texto2) - Extraer una parte del texto: substr(texto1, 1, 3) - Reemplazar parte del texto: sub("a", "o", texto1) - Dividir el texto en palabras: strsplit(texto2, " ")

#### **Ejercicio 3: Operaciones con Datos Lógicos**

Usando las variables anteriores, realizar las siguientes operaciones lógicas: - Comparar si var1 es mayor que var2 - Comparar si texto1 es igual a texto2 - Comparar si var1 es igual a var2 - Comparar si var1 es diferente de var2 Usar != - Comparar si texto1 contiene la palabra “hola” Usar grepl("texto", objeto )

---  
  
#### \*\*Ejercicio 2: Exploración de Datos\*\*  
Usa el siguiente código para crear un data frame de ejemplo:  
```R  
data <- data.frame(  
 Producto = c("A", "B", "C", "D"),  
 Precio = c(10, 15, 8, 12),  
 Ventas = c(100, 200, 150, 50)  
)

Responde: 1. Calcula el ingreso total por producto (Precio \* Ventas) y agrega una nueva columna al data frame.  
2. Encuentra el producto con mayores ventas.