

Ejercicios Prácticos: Elementos Básicos de R

Ciencia de Datos para Negocios

Marzo 2025

Introducción

Este documento contiene 20 ejercicios prácticos que cubren los conceptos fundamentales de R vistos en clase:

- Elementos básicos y operaciones
- Vectores
- Matrices
- Dataframes
- Tibbles

Para cada ejercicio, se proporciona un espacio para escribir el código. Intente resolver cada ejercicio por su cuenta antes de revisar las soluciones.

Ejercicios de Elementos Básicos

Ejercicio 1: Operaciones Aritméticas

Calcule las siguientes operaciones y asigne el resultado a una variable con un nombre descriptivo:

- La raíz cuadrada de 144
- El resultado de elevar 3 al cubo
- El residuo de dividir 157 entre 10
- El resultado de la expresión: $(15 * 3) / (2 + 1)^2$

```
# Tu código aquí
raiz_cuadrada <- sqrt(144)
cubo <- 3^3
residuo <- 157 %% 10
expresion <- (15 * 3) / (2 + 1)^2

# Mostrar resultados
raiz_cuadrada
```

```
## [1] 12
```

```
cubo
```

```
## [1] 27
```

```
residuo
```

```
## [1] 7
```

```
expresion
```

```
## [1] 5
```

Ejercicio 2: Variables y Tipos de Datos

Cree variables para almacenar la siguiente información sobre un producto:

- Nombre del producto (character)
- Precio (numeric)
- Disponibilidad en stock (logical)
- Categoría como factor con niveles: “Electrónica”, “Ropa”, “Alimentos”

Luego verifique el tipo de cada variable usando la función `class()`.

Ejercicio 3: Conversión de Tipos de Datos

Dadas las siguientes variables:

```
a <- "25.5"  
b <- 30  
c <- TRUE
```

Realice las siguientes conversiones:

- Convierta ‘a’ a numérico
- Convierta ‘b’ a character
- Convierta ‘c’ a numérico
- Convierta el resultado numérico de ‘a’ a entero usando `as.integer()`

Ejercicio 4: Operaciones Lógicas

Cree variables para almacenar:

- Un vector de precios: `c(150, 230, 540, 90, 410)`
- Un vector de unidades en stock: `c(10, 0, 15, 3, 0)`

Luego, cree vectores lógicos que identifiquen:

- Productos con precio mayor a 200
- Productos sin stock (0 unidades)
- Productos con precio mayor a 200 Y con stock disponible

Ejercicios de Vectores

Ejercicio 5: Creación de Vectores

Cree los siguientes vectores:

- Un vector con los primeros 10 números pares
- Un vector con las letras de su nombre (cada letra como un elemento)
- Un vector que contenga 5 valores TRUE y 5 valores FALSE alternados
- Un vector con la secuencia de números de 0 a 1 en intervalos de 0.1

Ejercicio 6: Acceso a Elementos de Vectores

Utilizando el siguiente vector:

```
numeros <- c(13, 48, 93, 12, 58, 14, 37, 51, 29, 63)
```

Acceda a:

- El tercer elemento
- Los elementos en posiciones pares
- Los elementos mayores a 30
- Los tres últimos elementos
- Todos los elementos excepto el primero y el último

Ejercicio 7: Operaciones con Vectores

Dados los siguientes vectores:

```
v1 <- c(4, 8, 15, 16, 23, 42)
v2 <- c(2, 4, 6, 8, 10, 12)
```

Realice las siguientes operaciones:

- Sume ambos vectores
- Multiplique v1 por 2
- Divida v1 entre v2
- Calcule la suma total de elementos en v1
- Calcule la media, mediana y desviación estándar de v2
- Encuentre el valor máximo y mínimo en v1

Ejercicio 8: Vectorización

Escriba código vectorizado (sin utilizar bucles) para realizar las siguientes tareas: - Calcule el cuadrado de cada número del 1 al 20 - Determine si cada número del 1 al 100 es primo (pista: puede usar la función `isPrime()` del paquete `numbers` o crear su propia función) - Calcule el logaritmo en base 10 de los números de 1 a 10 - Genere un vector de 50 elementos donde cada elemento sea la suma de su índice y el cuadrado de su índice

Ejercicios de Matrices

Ejercicio 9: Creación de Matrices

Cree las siguientes matrices:

- Una matriz 3x4 con los números del 1 al 12, llenada por filas
- Una matriz 4x4 con todos los elementos iguales a 5
- Una matriz 3x3 con la diagonal principal compuesta por 1's y el resto de elementos 0
- Una matriz 2x3 con valores aleatorios entre 0 y 100 (use `runif()`)

Ejercicio 10: Acceso a Elementos de Matrices

Dada la siguiente matriz:

```
mat <- matrix(1:20, nrow=4, ncol=5)
```

Acceda a:

- El elemento en la fila 2, columna 3
- Toda la tercera fila
- Las columnas 1 y 3
- Los elementos de la submatriz formada por las filas 2 a 4 y las columnas 3 a 5
- La diagonal de la submatriz formada por las filas 1 a 3 y las columnas 1 a 3

Ejercicio 11: Operaciones con Matrices

Dadas las siguientes matrices:

```
A <- matrix(1:9, nrow=3)
B <- matrix(9:1, nrow=3)
```

Realice las siguientes operaciones:

- Sume las matrices A y B
- Multiplique elemento a elemento A por B
- Realice la multiplicación matricial de A y B
- Calcule la transpuesta de A
- Calcule la suma de cada fila de B
- Calcule el producto de cada columna de A

Ejercicio 12: Aplicación de Matrices

Suponga que tiene datos de ventas de 3 productos en 4 tiendas diferentes:

```
ventas <- matrix(c(120, 85, 190, 110,
                  95, 105, 140, 120,
                  45, 35, 70, 50),
                nrow=3, byrow=TRUE)
rownames(ventas) <- c("Producto A", "Producto B", "Producto C")
colnames(ventas) <- c("Tienda 1", "Tienda 2", "Tienda 3", "Tienda 4")
```

Calcule:

- Las ventas totales por producto
- Las ventas totales por tienda
- El producto más vendido en cada tienda
- La tienda con mayores ventas totales
- La proporción de ventas de cada producto respecto al total

Ejercicios de Dataframes

Ejercicio 13: Creación de Dataframes

Cree un dataframe llamado `empleados` con la siguiente información:

- 5 empleados con nombre, edad, departamento, salario y años de experiencia
- Los departamentos deben ser “Ventas”, “Marketing” o “IT”
- Edad entre 25 y 60 años
- Salario entre 30000 y 80000
- Años de experiencia entre 1 y 30

Ejercicio 14: Acceso a Datos en Dataframes

Utilizando el dataframe `empleados` creado en el ejercicio anterior, realice lo siguiente:

- Acceda a la columna de salarios usando `$`
- Acceda a la fila del tercer empleado
- Seleccione solo las columnas nombre y departamento
- Obtenga una tabla resumen de las estadísticas básicas del dataframe

Ejercicio 15: Filtrado de Dataframes

Utilizando el dataframe `empleados`:

- Filtre los empleados del departamento de “IT”
- Filtre los empleados con salario mayor a 50000
- Filtre los empleados con menos de 5 años de experiencia Y más de 40 años de edad
- Ordene el dataframe por salario en orden descendente

Ejercicio 16: Manipulación de Dataframes

Partiendo del dataframe `empleados`:

- Agregue una nueva columna llamada “bono” que sea el 10% del salario
- Agregue una columna “salario_total” que sea la suma del salario y el bono
- Agregue una columna “eficiencia” que sea el salario dividido por los años de experiencia
- Reemplace los valores de la columna departamento “IT” por “Tecnología”

Ejercicios de Tibbles

Ejercicio 17: Conversión a Tibbles

Instale el paquete `tibble` si no lo tiene y convierta el dataframe `empleados` en un tibble. Luego:

- Verifique que es un tibble usando `is_tibble()`
- Observe las diferencias en cómo se muestra en comparación con un dataframe normal
- Agregue una columna de identificación única con `rowid_to_column()`

Ejercicio 18: Datos Reales con Tibbles

Usando el paquete `tidyverse` y la función `read_csv()`, cargue el dataset “mtcars” convertido a CSV (incluido en R). Si no puede usar `read_csv()`, use el dataset directamente con `data("mtcars")` y conviértalo a tibble.

```
# Tu código aquí
# Opción 1: Usando tibble y data()
# library(tibble)
# data("mtcars")
# mtcars_tibble <- as_tibble(mtcars, rownames = "car_name")

# Opción 2: Si quiere trabajar con read_csv
# write.csv(mtcars, "mtcars.csv")
# library(readr)
# mtcars_tibble <- read_csv("mtcars.csv")
```

Después de cargar los datos, responda:

- ¿Cuántas filas y columnas tiene el dataset?
- ¿Qué tipos de datos contiene cada columna?
- ¿Cuáles son los coches con más de 6 cilindros?
- ¿Cuál es el consumo promedio de combustible (mpg) para coches con diferentes tipos de transmisión?

Ejercicios Integradores

Ejercicio 19: Análisis de Datos de Estudiantes

Cree un dataframe con información de 10 estudiantes que incluya:

- Nombre
- Edad
- Género (como factor)
- Notas en 3 asignaturas diferentes (Matemáticas, Ciencias, Literatura)
- Actividad extracurricular (como factor con niveles: “Deportes”, “Música”, “Arte”, “Ninguna”)

Después de crear el dataframe, realice las siguientes operaciones:

- Calcule la nota media de cada estudiante

- Identifique al estudiante con la nota más alta en Matemáticas
- Calcule la nota media por asignatura
- Determine si hay diferencia en las notas medias según el género
- Cree una columna que clasifique a los estudiantes según su nota media: “Sobresaliente” (≥ 9), “Notable” (≥ 7), “Aprobado” (≥ 5), “Suspendido” (< 5)
- Cuente cuántos estudiantes hay en cada categoría de calificación

Ejercicio 20: Proyecto Mini-Análisis

Cree un conjunto de datos que simule ventas mensuales de productos de una tienda durante un año. El conjunto debe incluir:

- Mes (1-12)
- Producto (al menos 5 productos diferentes)
- Unidades vendidas
- Precio unitario
- Costo unitario

Realice un análisis completo que incluya:

- Cálculo de ingresos totales por mes
- Cálculo del margen de beneficio por producto
- Identificación del producto más rentable
- Análisis de tendencia de ventas a lo largo del año
- Creación de una matriz que muestre las ventas de cada producto por mes
- Conversión de los datos a un tibble para mejor visualización
- Cálculo de estadísticas descriptivas para unidades vendidas