



## COMPUTACIONAL ESTADÍSTICA

### LABORATORIO 1,2

#### Vectores y Factores

1. Crea un vector con los nombres de 8 ciudades y determina si es un vector de caracteres o un factor.
2. Convierte ese vector a factor y ordénalo alfabéticamente.
3. Genera un factor ordenado con las categorías: “Bajo”, “Medio”, “Alto” para 6 observaciones aleatorias.
4. Crea un vector numérico con las edades de 10 personas y determina cuántas son mayores de 30.
5. Usa table() para contar la frecuencia de cada color en un vector de colores.
6. Crea un factor de tipo “Estudiante” con niveles “Pregrado” y “Posgrado” para 12 observaciones y muestra sus niveles.
7. Usa which() para obtener las posiciones de los elementos mayores a 50 en un vector de 20 números aleatorios.
8. Crea un factor con 5 niveles y comprueba si es ordenado.
9. Convierte un vector de enteros en factor con etiquetas personalizadas.
10. Genera un factor de 10 observaciones con 3 niveles y cambia el orden de los niveles.

#### Matrices y Arreglos

1. Crea una matriz de 4x4 con los números del 1 al 16.
2. Calcula la suma de cada fila y de cada columna de la matriz anterior.
3. Cambia el valor de la posición (3,2) de una matriz por 0.
4. Crea una matriz identidad de dimensión 5.
5. Calcula la transpuesta de una matriz 3x3.
6. Multiplica dos matrices 2x2 y verifica el resultado manualmente.
7. Genera un arreglo de dimensión (2,3,2) con valores del 1 al 12 y muéstralo.
8. Obtén el determinante de una matriz 3x3 aleatoria.
9. Resuelve un sistema de ecuaciones usando solve() en R.
10. Extrae la tercera columna de una matriz y guárdala como vector.



## Listas

1. Crea una lista con tu nombre, edad, carrera y año de estudios.
2. Genera una lista que contenga un vector, una matriz y un data frame.
3. Accede al segundo elemento de la lista creada en el ejercicio anterior.
4. Crea una lista que incluya otra lista dentro de ella.
5. Cambia el valor de un elemento específico dentro de la lista.
6. Usa `names()` para asignar nombres a los elementos de una lista.
7. Crea una lista con datos de 3 personas (nombre y edad) y extrae todas las edades.
8. Combina dos listas en una sola.
9. Accede a un elemento de lista usando `[[ ]]` y usando `$`.
10. Convierte una lista en vector usando `unlist()`.

## Data Frames

1. Crea un data frame con 4 columnas: Nombre, Edad, Sexo y Carrera, con 6 observaciones.
2. Usa `summary()` para describir el data frame creado.
3. Obtén solo la columna Edad usando `$`.
4. Filtra los registros donde la Edad sea mayor a 25.
5. Usa `subset()` para seleccionar las filas donde Sexo sea “Femenino” y Edad mayor a 20.
6. Agrega una nueva columna llamada “Promedio” con valores aleatorios.
7. Elimina la columna “Carrera” del data frame.
8. Guarda el data frame en un archivo .txt con `write.table()`.
9. Lee un archivo .txt y conviértelo en data frame.
10. Usa `names()` para mostrar los nombres de las columnas.

## Datos perdidos

1. Crea un vector con 10 valores numéricos y al menos 3 NA.
2. Usa `is.na()` para identificar los valores perdidos.
3. Calcula la media ignorando los NA.
4. Reemplaza todos los NA por 0.
5. Genera un vector con valores que contenga NaN y detecta su posición.
6. Usa `complete.cases()` para identificar las filas completas de un data frame.
7. Elimina las filas con valores perdidos usando `na.omit()`.
8. Cuenta cuántos NA hay en un vector.
9. Calcula la mediana ignorando los valores perdidos.
10. Genera un data frame con 3 columnas y valores NA y NaN mezclados.



## Familia Apply

1. Usa `apply()` para calcular la suma de cada fila de una matriz 4x4.
2. Usa `apply()` para calcular el valor máximo de cada columna.
3. Emplea `tapply()` para calcular la media de la Edad agrupada por Sexo.
4. Usa `apply()` para obtener el cuantil 0.25 y 0.75 de cada columna de una matriz.
5. Genera un data frame y usa `apply()` para obtener la desviación estándar de cada columna numérica.
6. Usa `lapply()` para aplicar la función `sqrt()` a cada elemento de una lista de vectores numéricos.
7. Usa `sapply()` para obtener el número de caracteres de cada elemento de un vector de cadenas.
8. Usa `mapply()` para sumar elemento a elemento dos vectores.
9. Con `tapply()`, calcula la suma de sueldos por tipo de contrato en un data frame.
10. Usa `apply()` para calcular el producto de cada fila en una matriz.

## Expresiones Aritméticas

1. Calcula  $(1 + 1/50)^{50}$  y compara con  $\exp(1)$ .
2. Obtén el resto y el cociente de dividir 45 entre 6.
3. Calcula el seno, coseno y tangente de  $\pi/4$ .
4. Redondea hacia abajo y hacia arriba el valor de  $\pi$ .
5. Asigna el valor 200 a una variable y calcula  $(1 + 1/x)^x$ .
6. Evalúa  $\sqrt{16} + \log(100)$ .
7. Calcula la raíz cúbica de 27 usando potencias.
8. Multiplica y divide dos números grandes y presenta el resultado en notación científica.
9. Calcula el área de un círculo de radio 5 usando  $\pi$ .
10. Genera una expresión con paréntesis anidados y evalúala.



## Sucesiones

1. Genera una secuencia del 1 al 20 de dos en dos.
2. Genera una secuencia descendente del 10 al 1.
3. Usa `rep()` para repetir el número 7 cinco veces.
4. Repite el vector `1:4` tres veces.
5. Genera una secuencia de números impares del 1 al 15.
6. Crea una secuencia de longitud 10 entre 0 y 5 usando `length.out`.
7. Genera un vector con el patrón `1,2,3` repetido cuatro veces.
8. Usa `seq()` para generar valores desde 0 hasta 1 en saltos de 0.1.
9. Genera una secuencia descendente de 50 a 10 de cinco en cinco.
10. Usa `rep()` para repetir cada número de `1:3` tres veces antes de pasar al siguiente.