



# Práctica con R: *Matrices*

Matemática para Ciencia de Datos

Prof. Esteban Ballester

# Ejecución de Instrucciones y comentarios en R

- ▶ Comentarios: Usar # al inicio de la instrucción

Ejemplo: # esto es un comentario

- ▶ Ejecución:

- ▶ Botón Run
- ▶ Seleccione el bloque de instrucciones, luego Ctrl + R
- ▶ Ejecución línea por línea: Ctrl + Enter
- ▶ Insertar un chunk: Ctrl + i
- ▶ Insertar símbolo de asignación (<-): Alt + -

- ▶ Solicitud de ayuda: ?

- ▶ Ejemplo: ?matrix

# Cálculos Básicos

Operación	Instrucción
Potencias	<code>^</code>
División, división entera	<code>/</code> , <code>/%</code>
Módulo	<code>%%</code>
Multiplicación usual, suma usual	<code>*</code> , <code>+</code>
Raíz cuadrada	<code>sqrt()</code>
Valores	<code>pi</code>
Valor absoluto	<code>abs()</code>
Algunas funciones básicas	<code>sin()</code> , <code>factorial()</code> , <code>log()</code>

# Instalando paquetes

- ▶ `install.packages("nombre_paquete",dependencies=TRUE)`
- ▶ NOTA: R es sensible a las mayúsculas y las minúsculas

# Tipos de datos y objetos de R

En el lenguaje de R, los elementos u objetos que se vayan definiendo, bien por nosotros mismos, bien como resultado del programa, pueden y deben ser distinguidos para su uso correcto. Seguidamente se enlistan los tipos de objetos en R:

- ▶ Vectores
- ▶ Matrices
- ▶ Factores
- ▶ Hojas de datos (data frames)
- ▶ Listas

# Matrices

Una matriz se define mediante la función `matrix()` a la que hay que especificarle los elementos y la dimensión. Por ejemplo, para definir la matriz use `matrix(c(entradas), N° filas, N° columnas)`

Ejemplo: cree la siguiente matriz

```
matriz<-matrix(c(1,2,3,4,5,6,7,8,9,0,0,-1),4,3)
```

```
matriz
```

# Ejercicios:

Para la matriz creada anteriormente, realice en RStudio las siguientes instrucciones y vaya anotando lo que observa:

```
dim(matriz) # dimensión de la matriz  
summary(matriz)  
str(matriz)  
is.matrix(matriz)  
length(matriz)
```

Consultas sobre entradas de la matriz:

```
matriz[2,3]  
matriz[1:2,2:3]  
matriz[,c(1,3)]
```

# Operaciones con Matrices

Por otra parte, tanto para vectores como para matrices, funcionan las operaciones suma y diferencia sin más complicaciones. En el caso del producto, sin embargo, hay que clarificar que por ejemplo, `matriz*matriz` devuelve la multiplicación elemento a elemento y no el producto matricial.

```
matriz*matriz
```

Mientras que `matriz%%matriz` sí devuelve el producto matricial

```
matriz%%matriz
```



# Operaciones con matrices

Cree una nueva matriz como sigue:

```
matriz2<-matrix(c(1,2,3,4,5,6),3,2)
```

```
dim(matriz2)
```

```
nrow(matriz2)
```

```
ncol(matriz2)
```

Muestre la matriz para asegurarse que se creó correctamente

Calcule:

```
matriz2%*%matriz
```

```
matriz%*%matriz2
```

# Creación de matrices:

Cree una nueva matriz como sigue:

```
X <- matrix(data=c(1,2,3,4), nrow=2, ncol=2)
```

```
x
```

```
X2 <- matrix(data=c(1,2,3,4), 2, 2)
```

```
X2
```

```
sqrt(x)
```

```
X^2
```

Las funciones anteriores, hacen un llenado de los datos por columnas (por defecto). Si interese hacer el llenado por filas, realice lo siguiente:

```
X2 <- matrix(data=c(1,2,3,4), 2, 2, byrow = TRUE)
```

# Creación de matrices:

A partir de un vector, si le añadimos el atributo dimensión podemos obtener una matriz:

```
m <- 1:12
```

```
m
```

```
dim(m) <- c(4, 3)
```

```
M
```

# Creación de matrices

En el siguiente ejemplo hacemos uso del argumento `dimnames` para dar nombre a las filas y columnas:

```
automoviles <- matrix(1:12, nrow = 4, ncol = TRUE, dimnames =  
list(c("Blanco", "Rojo", "Negro", "Gris"), c("Toyota", "Audi",  
"Nissan")))
```

`automoviles`

Para eliminar filas utilizaremos la notación `[-i, ]`, de forma similar para eliminar columnas utilizaremos la notación `[, -j]`

```
automoviles[-5, ]
```

```
automoviles[, -4]
```

# Combinando matrices

Las funciones `cbind()` y `rbind()` se pueden usar para unir dos o más vectores o matrices, por columnas o por filas, respectivamente.

```
m1 <- matrix(c(45, 23, 66, 77, 33, 44, 56, 12, 78, 23), 2, 5) m1
```

```
cbind(c(4, 76), m1[, 4])
```

```
m2 <- matrix(rep(10, 20), 4, 5) #se recomienda que la cantidad de  
elementos a repetir, coincida con el número de entradas de la matriz
```

```
m2
```

```
m3 <- rbind(m1[1, ], m2[3, ])
```

```
m3
```

# Otras operaciones con matrices

## ► Operaciones con matrices

- $A \%*\% B$  : producto de matrices
- $t(A)$  : transpuesta de la matriz  $A$
- $\text{solve}(A,b)$  : solución del sistema de ecuaciones  $Ax=b$ .
- $\text{solve}(A)$  : inversa de la matriz  $A$
- $\text{diag}(b)$  : matriz diagonal ( $b$  es un vector)
- $\text{diag}(A)$  : matriz diagonal ( $A$  es una matriz)
- $\text{rref}(A|B)$  : matriz en forma reducida por fila (escalonada reducida). La expresión  $A|B$  es la matriz aumentada del sistema. Requiere instalar y cargar el paquete `pracma`.
- $\text{fractions}(A)$  : transforma las entradas de la matriz de decimal a fracciones. Para usar esta función debe instalar y cargar el paquete `MASS`
- $\text{det}(A)$  : calcula el determinante de una matriz

# Práctica: Inversas y sistemas

Resuelva el siguiente sistema de ecuaciones lineales

$$1. \begin{cases} x + 2y - z = 3 \\ -5y + 3z = -4 \\ 3x + y + 2z = -1 \end{cases}$$

$$2. \begin{cases} x + 2y - z = 3 \\ -5y + 3z = -4 \\ 3x + y + 2z = -1 \end{cases}$$

$$3. \begin{cases} x - 5z = 15 \\ y + 8z = -26 \\ 3x + 2y + z = 5 \end{cases}$$

$$4. \begin{cases} x - z + 2w = -6 \\ y + z - w = 11 \\ z - 3w = 8 \end{cases}$$

$$5. \begin{cases} x + 2y - 3z = 3 \\ y - 2z = 2 \\ -4y + 8z = -8 \end{cases}$$