



## Sesión 1 de Prácticas

### Introducción a Matlab. Matrices y operaciones Básicas.

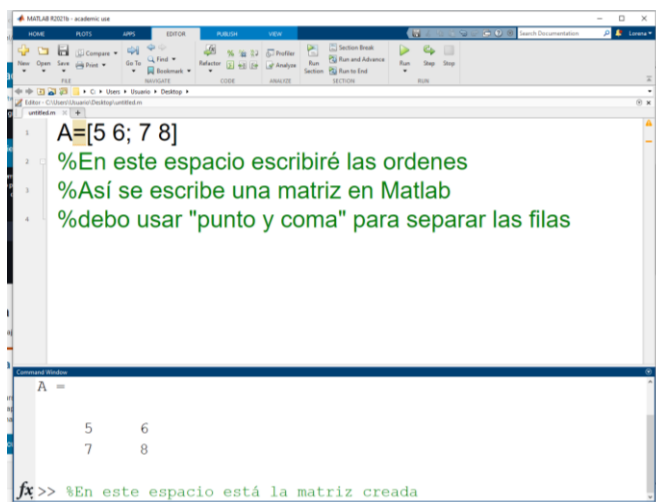
MATLAB, acrónimo de “Matrix LABoratory”, es un programa de gran utilidad como herramienta de cálculo numérico en aplicaciones simuladas por ordenador. Consiste en un entorno de programación especialmente diseñado para operar con matrices, ideal para cualquier tipo de aplicación que requiera el análisis de señales y sistemas. En esta primera sesión los objetivos son:

- Presentar el entorno MATLAB y su espacio de trabajo.
- Suministrar diferentes modos de generar matrices.
- Realizar operaciones básicas con matrices.
- Identificar y solucionar errores típicos en el tratamiento de matrices.

La Universidad de Huelva proporciona Licencia Individual para MATLAB, Simulink y más de 80 toolboxes complementarios de manera gratuita a los alumnos, profesores y personal de la UHU para el trabajo de clase y la investigación académica. Cada usuario tiene instalaciones ilimitadas y sólo es necesaria la conexión a Internet para la descarga y activación de la licencia. La descarga desde la página web de la UHU es a través de <http://uhu.es/sic/>.

#### 1. CONSIDERACIONES PREVIAS.

1. Utilizaremos la *ventana de edición* para escribir el pseudocódigo que dará las ordenes que se quiere que ejecute MATLAB. Y visualizaremos el resultado de ejecutar dichas ordenes en la *ventana de comandos*.
  - a) Para abrir la venta de edición debemos seguir esta ruta: Menú principal→HOME→ New Script→Save.
  - b) Para ejecutar el pseudocódigo debemos seguir esta ruta: Menú principal→EDITOR→ Run
2. Recuerda que se puede añadir comentarios o notas en MATLAB, para ello es necesario que antepongas el símbolo % de la nota que desees recordar.
3. Para realizar operaciones básicas en MATLAB se usan los operadores usuales (+, -, \*, ^).





## 2. TIPOS DE MATRICES

### 2.1 MATRICES BÁSICAS

**Ejercicio 1:** Haciendo uso de corchetes y punto y coma generaremos la siguiente matriz.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 2 & 8 \\ -5 & 7 & -1 \end{bmatrix}$$

Escribiremos  $A = [1 \ 2 \ 3; 4 \ 2 \ 8; -5 \ 7 \ -1]$

**Ejercicio 2:** En ocasiones necesitaremos operar una matriz con un vector. Así que aprenderemos a escribir un vector fila y vector columna. La diferencia de escritura de un vector u otro está en el empleo de “punto y coma”. Construye:

$$\text{a) } v1 = [1 \ 2 \ 3] \qquad \text{b) } v2 = \begin{bmatrix} 1 \\ 4 \\ -5 \end{bmatrix}$$

**Ejercicio 3:** En ocasiones necesitaremos operar una matriz con un vector. Para introducir un escalar basta con escribir una *letra=el número que deseamos*. Introduce el escalar 5 usando la letra e.

**Ejercicio 4:** Usaremos la función `>>zeros(m,n)` para producir una matriz  $m \times n$  de ceros. Además, usaremos la función `>>zeros(n)` para producir una matriz  $n \times n$  de ceros. Construye:

$$\text{a) } \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad \text{b) } \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad \text{c) una matriz cero de } 30 \times 52$$

**Ejercicio 5:** Usaremos la función `>>ones(m,n)` para producir una matriz  $m \times n$  de ceros. Además, usaremos la función `>>ones(n)` para producir una matriz  $n \times n$  de ceros. Construye una matriz de unos de dimensión  $3 \times 2$  y otra de  $6 \times 6$ .

**Ejercicio 6:** Usaremos la función `>>eye(m,n)` para producir una matriz identidad. Construye:

- a) Matriz identidad de  $3 \times 2$
- b) Matriz identidad de  $4 \times 4$
- c) Matriz identidad de  $25 \times 25$

Nota: Como se ha visto en los ejercicios, no es necesario que la matriz sea cuadrada, las matrices identidad creadas con *eye* tendrán unos en su diagonal principal



## 2.2 GENERACIÓN DE MATRICES AUTOMÁTICAS

**Ejercicio 7:** La función `>>rand(m,n)` genera una matriz  $m \times n$  de números aleatorios uniformemente distribuidos en el intervalo  $[0,1]$ . Construye:

- Una matriz  $5 \times 3$  con números aleatorios.
- Una matriz  $3 \times 3$  con números aleatorios.

**Ejercicio 8:** Es posible construir matrices a partir de otras matrices, siempre que se respeten las dimensiones. Tenemos la matriz  $D = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ , Podemos a partir de ella construir una matriz  $F = [D \text{ zeros}(2); \text{ones}(2) \text{ eye}(2)]$

## 3. CARACTERÍSTICAS DE UNA MATRIZ.

**Ejercicio 9:** Para poder determinar el tamaño de una matriz se utilizan las funciones

`>>size(A)` devuelve las dimensiones de la matriz  $F$

`>>length(A)` devuelve la mayor de las dimensiones de la matriz  $F$ .

Aplica estas funciones a las matrices creadas en el punto anterior.

## 4. OPERACIONES BÁSICAS CON MATRICES

**Ejercicio 10:** Para transponer una matriz es necesario usar un punto y una comilla (`.`). Determina la transpuesta de  $Y$ . Hacedlo primero manualmente. Luego, confirma tu respuesta usando Matlab.

$$Y = \begin{bmatrix} 20000 & 80000 & 160000 \\ 50000 & 10000 & 400000 \end{bmatrix}$$

**Ejercicio 11:** Suma, resta, multiplica y divide la matriz del ejercicio 1 con el escalar del ejercicio 3. Hacedlo primero manualmente. Luego, confirma tu respuesta usando Matlab.

**Ejercicio 12:** Construye las matrices  $P$  y  $Q$ . Luego, determina  $P+Q$ ,  $P-Q$ ,  $P*Q$ ,  $Q*P$ . Hacedlo primero manualmente. Luego, confirma tu respuesta usando Matlab.

$$P = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad Q = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$