

Laboratori 0: Using MATLAB in Lineal Algebra

Grup 11- Estudiants:

- Pol Casacuberta Gil
- Marta Granero i Martí

Link: <https://drive.matlab.com/sharing/b8a7f88a-dcb6-4c09-a328-ba6c7e737807>

Sections:

- Variable declaration
- Matrix and array arithmetic
- Operations
- Indexing
- Special Matrices
- Systems of Linear Equations

Variable declaration

```
%scalar definitions
a = 3;
b = 0.5;
c = -2.7;

%vector definitions
u = [1:3]';
v = [-2;1.5;1];
s = [-1 -0.4 2 1];
w = [4 -2 6 3];

%arrays definitions
A = [1:4;
     -1 0 5 2;
     -1 2 8 1;
     0 5 3 1];
B = [0.28 -0.45 0.84 1.01;
     0.83 -0.3 -0.45 1.99;
     0.46 0.83 0.29 3.03;
     0 0 0 1];
C = [1:4;
     5:8];
```

Matrix and array arithmetic

```
%Esta sección no tiene ejercicios pero hemos leído la documentación
% y visto la sintaxis de las operaciones con matrices que hace MATLAB
```

Operations

A continuación, escribiremos los comandos para realizar las siguientes operaciones que se nos piden y comprobaremos y analizaremos el resultados de algunas operaciones:

Lista de operaciones:

$u+v$, $u-v$, conjugate transpose of vector u , $a*v$, $u*v$, $u.*v$, $u*v'$, dot product of vectors u and v , cross product of vectors u and v , dot product of vectors s and w , cross product of vectors w and w , transpose of matrix A , inverse of matrix A , $A+B$, $A-B$, matrix multiplication of A and B , array multiplication of A and B , matrix multiplication of A and C , matrix multiplication of C and A , array multiplication of A and C , array multiplication of C and A , $w*A$, $A*w$

```
u + v           % suma de vectores
```

```
ans = 3x1
-1.0000
 3.5000
 4.0000
```

```
u - v           % resta de vectores
```

```
ans = 3x1
 3.0000
 0.5000
 2.0000
```

```
u'              % conjugate transpose del vector u
```

```
ans = 1x3
     1     2     3
```

```
a * v           % multiplicacion de scalar a por v
```

```
ans = 3x1
-6.0000
 4.5000
 3.0000
```

```
%u * v          % producto interno de u*v (no se puede realizar
% esta operación ya que sus vectores no tienen una dimensionalidad correcta)
```

```
u .* v          % multiplicación de elementos de u por v
```

```
ans = 3x1
-2
 3
 3
```

```
u * v'          % producto exterior de vectores u por v'
```

```
ans = 3x3
-2.0000    1.5000    1.0000
-4.0000    3.0000    2.0000
-6.0000    4.5000    3.0000
```

```
dot(u, v)       % dot product de u y v
```

```
ans = 4
```

cross(u, v) % cross product de u y v

```
ans = 3x1
-2.5000
-7.0000
5.5000
```

dot(s, w) % dot product de s y w

```
ans = 11.8000
```

%cross(w, w) % cross product de w y w tenemos un error ya que
% como el cross product sigue la regla de la mano derecha el producto
% cruzado solo funciona para vectores tridimensionales(3 componentes) y,
% por lo tanto, no podemos encontrar el producto cruzado para cuatro
% dimensiones, que son el número de componentes que tiene w.

A' % transposada de A

```
ans = 4x4
1    -1    -1     0
2     0     2     5
3     5     8     3
4     2     1     1
```

inv(A) % inversa de A

```
ans = 4x4
0.5408   -1.1939    0.7347   -0.5102
-0.0612    0.0408   -0.1020    0.2653
0.0714   -0.2143    0.2857   -0.1429
0.0918    0.4388   -0.3469    0.1020
```

A + B % suma de matrices

```
ans = 4x4
1.2800    1.5500    3.8400    5.0100
-0.1700   -0.3000    4.5500    3.9900
-0.5400    2.8300    8.2900    4.0300
0         5.0000    3.0000    2.0000
```

A - B % resta de matrices

```
ans = 4x4
0.7200    2.4500    2.1600    2.9900
-1.8300    0.3000    5.4500    0.0100
-1.4600    1.1700    7.7100   -2.0300
0         5.0000    3.0000     0
```

A * B % multiplicación de matrices

```
ans = 4x4
3.3200    1.4400    0.8100   18.0800
2.0200    4.6000    0.6100   16.1400
5.0600    6.4900    0.5800   28.2100
5.5300    0.9900   -1.3800   20.0400
```

A .* B % multiplicación de elementos de A y B

```
ans = 4x4
    0.2800   -0.9000    2.5200    4.0400
   -0.8300         0   -2.2500    3.9800
   -0.4600    1.6600    2.3200    3.0300
         0         0         0    1.0000
```

```
%A * C           % multiplicación de matrices de A y C
C * A           % multiplicación de matrices de C y A
```

```
ans = 2x4
    -4     28     49     15
    -8     64    125     47
```

```
%A .* C           % multiplicación por elementos de A y C
%C .* A           % multiplicación por elementos de C y A
w * A             % multiplicación del vector w por A
```

```
ans = 1x4
     0     35     59     21
```

```
%A * w           % multiplicación de la matriz A por w
```

Vemos a continuación porqué no podemos realizar las 4 últimas operaciones que hemos comentado:

- **A * C** no funciona porque las dimensiones de A y C no son compatibles para la multiplicación de matrices.
- **A .* C** no funciona porque las dimensiones de A y C no son las mismas.
- **C .* A** no funciona tampoco porqué las dimensiones de A y C no son las mismas.
- **A * w** no funcionaría ya que no es una multiplicación de matrices válida porque las dimensiones de las dos matrices no son compatibles.

Indexing

```
A(2,4)           % Extrae el elemento en la fila 2, columna 4 de A
```

```
ans = 2
```

```
A(2,:)           % Extrae la fila 2 de A
```

```
ans = 1x4
    -1         0         5         2
```

```
B(:,4)           % Extrae la columna 4 de B
```

```
ans = 4x1
    1.0100
    1.9900
    3.0300
    1.0000
```

```
B(1:3,1:3) % Extrae la submatriz 3x3 de B de las primeras tres filas y columnas
```

```
ans = 3x3
    0.2800   -0.4500    0.8400
    0.8300   -0.3000   -0.4500
```

0.4600 0.8300 0.2900

Special Matrices

```
D = zeros(3,4) % Matriz de ceros: una matriz llena de ceros
```

```
D = 3x4
    0     0     0     0
    0     0     0     0
    0     0     0     0
```

```
E = ones(2,2) % Matriz de unos: una matriz llena de unos
```

```
E = 2x2
    1     1
    1     1
```

```
F = eye(3) % Matriz identidad: una matriz cuadrada con unos en la diagonal y ceros en el resto
```

```
F = 3x3
    1     0     0
    0     1     0
    0     0     1
```

```
G = rand(2,3) % Matriz aleatoria: una matriz con elementos muestreados aleatoriamente a partir de una distribución uniforme
```

```
G = 2x3
    0.1869    0.4456    0.7094
    0.4898    0.6463    0.7547
```

```
H = randn(4,4) % Matriz aleatoria distribuida normalmente: una matriz con elementos muestreados a partir de una distribución normal
```

```
H = 4x4
   -0.8637   -0.0068   -0.2256    0.5525
    0.0774    1.5326    1.1174    1.1006
   -1.2141   -0.7697   -1.0891    1.5442
   -1.1135    0.3714    0.0326    0.0859
```

```
I = magic(5) % Magic square: una matriz cuadrada en la que la suma de los elementos en cada fila y columna es la misma
```

```
I = 5x5
    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9
```

Systems of Linear Equations

```
%Definiciones de la matriz A y el vector columna B
```

```
A = [2, -1, 3; 1, 2, 2; -1, 5, -2]
```

```
A = 3x3
     2    -1     3
     1     2     2
    -1     5    -2
```

```
B = [11.5; 6; -9.5]
```

```
B = 3×1  
11.5000  
6.0000  
-9.5000
```

%Entonces podemos resolver para el vector columna desconocido X

```
X = A\B
```

```
X = 3×1  
1.0000  
-0.5000  
3.0000
```

La solución al sistema de ecuaciones lineales es: **x = 1.0, y = -0.5 y z = 3.0**