

Métodos Matriciales para ingenieros con MATLAB

Juan Carlos Herrera Sánchez, Ph.D.



Pontificia Universidad
JAVERIANA
Cali

Facultad de Ingeniería



Pontificia Universidad
JAVERIANA
Cali

Rector: Jorge Humberto Peláez Piedrahita, S.J.
Vicerrector Académico: Antonio de Roux Rengifo
Vicerrector del Medio Universitario: Luis Fernando Granados Ospina, S.J.

Facultad de Ingeniería
Decano Académico: Mauricio Jaramillo Ayerbe
Decano del Medio Universitario: Alberto Benavides Herrán
Director Depto de Ciencias e Ingeniería de la Producción: Alvaro Figueroa Cabrera

Título: Métodos Matriciales para ingenieros con MATLAB
Autor: Juan Carlos Herrera Sánchez, Ph.D.

ISBN: 978-958-8347-52-3

Coordinador Editorial: Ignacio Murgueitio Restrepo
e-mail: mignacio@javerianacali.edu.co

© Derechos Reservados
© Sello Editorial Javeriano

Correspondencia, suscripciones y solicitudes de canje:
Calle 18 # 118-250
Santiago de Cali, Valle del Cauca
Pontificia Universidad Javeriana Cali
Facultad de Ciencias de la Salud
Teléfono 3218200 ext. 493 - 533
www.javerianacali.edu.co

Formato: 17 x 25 cms

Concepto Gráfico: Edith Valencia F.

Edición: agosto de 2011

Métodos Matriciales
para ingenieros
con MATLAB

Juan Carlos Herrera Sánchez, Ph.D.

Herrera Sánchez, Ph.D., Juan Carlos
Métodos Matriciales para ingenieros con MATLAB / Juan Carlos Herrera
Sánchez, Ph.D. -- Santiago de Cali: Pontificia Universidad Javeriana, Sello
Editorial Javeriano, 2011.
p. 154 : il.; 17 x 25 cm.

Incluye referencias bibliográficas.

ISBN: 978-958-8347-52-3

1. Matrices (Matemáticas) 2. MATLAB 3. Determinantes 4. Ecuaciones
lineales I. Pontificia Universidad Javeriana (Cali). Facultad de Ingeniería.

SCDD 512.9434 ed.21

BPUJC arm/11

Prefacio

El presente texto está orientado hacia los cursos de pregrado de Análisis de Estructuras, Análisis Matricial y Dinámico de Estructuras y Análisis Numérico, ofrecidos para Ingeniería Civil. También será de referencia en cursos de postgrado tales como Método de Elementos Finitos. No obstante, será útil como texto de referencia para estudiantes de otras áreas de la ingeniería ofrecidas por la Facultad. En el Capítulo 1 se presentan los conceptos básicos del álgebra de matrices así como al manejo de vectores y matrices con MATLAB. En éste capítulo y a lo largo del texto se presentan numerosos ejemplos usando el software citado, para que sirvan de complemento a los aspectos teóricos presentados. En el Capítulo 2 se tratan las operaciones fundamentales con matrices. El Capítulo 3 está dedicado al tema de inversión de matrices y al cálculo de determinantes. En el Capítulo 4 se presentan los métodos tradicionales para la solución de sistemas de ecuaciones lineales. Finalmente, en el Anexo, se presenta una introducción a al tópico sobre integración y diferenciación de matrices usando MATLAB. El texto es de carácter introductorio, y por tanto será de utilidad tanto a estudiantes de pregrado de ingeniería, como a profesionales de ingeniería.

El Autor

Tabla de contenido

Capítulo 1. Tipos de matrices.....	9
1.1 Definiciones.....	9
1.2 Manipulación de vectores y matrices en MATLAB.....	10
1.3 Clases de matrices.....	20
1.4 Referencias bibliográficas.....	41
1.5 Problemas.....	41
Capítulo 2. Operaciones con matrices.....	43
2.1 Producto de un número real por una matriz.....	43
2.2 Suma de matrices.....	44
2.3 Multiplicación de matrices.....	52
2.4 Partición de matrices.....	58
2.5 Referencias bibliográficas.....	63
2.6 Problemas.....	63
Capítulo 3. Determinantes e inversión de matrices.....	65
3.1 Determinante de una matriz.....	65
3.2 Expansión de Laplace.....	67
3.3 Determinante por Condensación Pivotal.....	77
3.4 Inversión usando la matriz adjunta.....	79
3.5 Método de Gauss-Jordan.....	82
3.6 Inversa de una matriz por medio de partición.....	91
3.7 Referencias bibliográficas.....	98
3.8 Problemas	98

Capítulo 4. Solución de sistemas de ecuaciones lineales.....	101
4.1 Forma matricial de las ecuaciones.....	101
4.2 Solución por inversión de matrices.....	102
4.3 Regla de Cramer.....	106
4.4 Método de eliminación de Gauss.....	109
4.5 Método de Gauss-Jordan.....	127
4.6 Método de Cholesky.....	137
4.7 Factorización LU.....	142
4.8 Referencias bibliográficas	146
4.9 Problemas.....	146
 Anexo. Diferenciación e integración de matrices con MATLAB.....	 149

Capítulo 1

Tipos de matrices

1.1 Definiciones

Una matriz se define como un conjunto de elementos ordenados en un número de filas m y un número de columnas n . En este caso la matriz se dice que es de orden $(m \times n)$. La matriz $[A]_{m \times n}$ se define por:

$$A = [a_{ij}]_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & a_{ij} & a_{in} \\ a_{m1} & a_{m2} & a_{mj} & a_{mn} \end{bmatrix}$$

La i -ésima fila de $[A]$ tiene n elementos:

$$[a_{i1} \quad a_{i2} \quad \cdots \quad a_{in}]_{1 \times n}$$

La j -ésima columna de $[A]$ tiene m elementos:

$$\begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix}$$

En la matriz anterior cada a_{ij} indica el elemento ubicado en la fila “ i ” y la columna “ j ”. Por ejemplo: el elemento a_{32} es el elemento de la fila 3 y columna 2.

Una matriz de orden $(1 \times n)$, tiene 1 sola fila, es un *vector fila*. Se puede escribir:

$$[c] = [c_1 \quad c_2 \quad \cdots \quad c_n]$$

Una matriz de orden $(m \times 1)$, tiene 1 sola columna y m filas, se denomina vector columna.

1.2 Manipulación de vectores y matrices en MATLAB

Para crear un vector fila en MATLAB, se escribe el conjunto de elementos entre corchetes. Se separan por espacios o una coma (,) para delimitar los números.

```
>> va=[-1 0 1]
va =
    -1     0     1
```

Si se usan comas:

```
>> va=[-1, 0, 1]
va =
    -1     0     1
```

Vector columna: Para crear un vector columna, se escribe el conjunto de números entre corchetes y se separan por punto y coma (;):

```
>> vc=[1;2;4;16]
```

TIPOS DE MATRICES

```
vc =  
    1  
    2  
    4  
   16
```

Transpuesta de un vector:

Un vector fila se puede convertir a un vector columna calculando su transpuesto. Se usa el comando `transpose (V)` o la comilla (`'`).

Ejemplo:

Obtener un vector columna del vector $va=[-1 \ 0 \ 1]$.

```
>> vc=va'  
vc =  
   -1  
    0  
    1
```

```
>> vc=transpose(va)  
vc =  
   -1  
    0  
    1
```

Calculemos un vector fila, usando el vector columna [vc]:

```
>> vc=[1;2;4;16]
```

```
vc =  
    1  
    2  
    4  
   16
```

```
>> vf=vc'
```

```
vf =  
    1    2    4   16
```

Para definir un *vector igualmente espaciado* se da el comando:

```
x=[vi : i: vf]
```

Donde :

vi: valor inicial

vf: valor final

i: incremento

Para generar el vector $a=[0 \ 1 \ 2 \ \dots \ 9]$ se escribe en MATLAB:

```
>> a=[0:1:9]
a =
      0      1      2      3      4      5      6      7
      8      9
```

Creemos el vector $x=[0, 0.25, 0.5, \dots, 1.75, 2]^T$, con incrementos de 0.25,

```
>> x=[0:0.25:2]'
```

```
x =
      0
    0.2500
    0.5000
    0.7500
    1.0000
    1.2500
    1.5000
    1.7500
    2.0000
```

Otra forma es usar el comando `linspace (x1, x2, N)`. Este comando genera N puntos entre los valores X_1 and X_2 .

Para crear el vector $x=[0, 0.25, 0.5, \dots, 1.75, 2]^T$ se escribe en MATLAB:

```
>> B=linspace(0,2,9)'
```

TIPOS DE MATRICES

B =

```
      0
0.2500
0.5000
0.7500
1.0000
1.2500
1.5000
1.7500
2.0000
```

Para definir una matriz se escriben las filas separadas por “;”.
Para definir la matriz [B] de 2 filas y 3 columnas, se escribe:

```
>> B=[1 2 3;4 5 6]
```

B =

```
      1      2      3
      4      5      6
```

La otra forma es introducir la primera fila y luego dar **enter**
(↵).

```
>> B=[1 2 3↵
      4 5 6]
```

B =

```
      1      2      3
      4      5      6
```

Para definir una matriz también se puede separar cada fila por
“,...”

```
>>ml=[2      0      0      6      0      0;...
      0    156     22      0     54    -13;...
      0     22      4      0     13     -3;...
      6      0      0     12      0      0;...
      0     54     13      0     56    -22;...
      0    -13     -3      0    -22     4];
```

```
ml =
```

```
      2      0      0      6      0      0
      0    156     22      0     54    -13
      0     22      4      0     13     -3
      6      0      0     12      0      0
      0     54     13      0     56    -22
      0    -13     -3      0    -22     4
```

Para indicar el elemento b_{12} escribimos:

```
>> b12=B(1,2)
```

```
b12 =      2
```

De igual forma para indicar el elemento b_{23} escribimos:

```
>> b23=B(2,3)
```

```
b23 =      6
```

El orden de [B] es (2×3) . En MATLAB para obtener el orden de una matriz es el comando `size`.

```
>> size(B)
```

```
ans =
```

```
      2      3
```

TIPOS DE MATRICES

El comando `size` indica el numero de filas y columnas de la matriz. Otra forma usar este comando es:

```
>> [m,n]=size(B)
```

```
m =      2
```

```
n =      3
```

La siguiente instrucción retorna el vector $\{Vb\}$ con el número de filas y columnas de la matriz $[B]$.

```
>> Vb=size(B)
```

```
Vb =
```

```
      2      3
```

Ejemplo:

Para definir la matriz fila $[A]$ de 1 fila y 3 columnas, se escribe:

```
>> A=[1 2 3]
```

```
A =
```

```
      1      2      3
```

```
>> [m,n]=size(A)
```

```
m =
```

```
      1
```

```
n =
```

```
      3
```

Para obtener el número de filas de un vector columna, o el número de columnas de un vector fila en MATLAB, se usa el comando `length(V)`

```
>> length(A)
```

```
ans =
```

```
3
```

Ejemplo:

Para definir un vector columna $[A]$ de 3 filas, se escribe los elementos separados por “;”:

```
>> A=[2;4;6]
```

```
A =
```

```
2
```

```
4
```

```
6
```

```
>> length(A)
```

```
ans =
```

```
3
```

```
>> [m,n]=size(A)
```

```
m =
```

```
3
```

```
n =
```

```
1
```