

CAPÍTULO 3

VECTORES Y MATRICES

1.- ESTRUCTURA DE DATOS. LISTAS

2.- DEFINICIÓN DE VECTORES Y DE MATRICES

3.- OPERACIONES CON VECTORES Y MATRICES

4.- RANGO DE UNA MATRIZ

5.- RESOLUCIÓN DE SISTEMAS DE ECUACIONES LINEALES

6.- EJEMPLO

1.- ESTRUCTURA DE DATOS. LISTAS

Una lista es una estructura de datos cuyos elementos aparecen representados entre llaves y separados por comas.

En una estructura del tipo lista cada elemento ocupa una posición determinada y el programa efectuará cuantas operaciones sean necesarias para simplificar los elementos de la lista.

```

In[1]:= {1, 2, 3, 4, 5}
Out[1]= {1, 2, 3, 4, 5}

In[2]:= {a, 1 - a, 1 - a2}
Out[2]= {a, 1 - a, 1 - a2}

In[3]:= {Sin[π], Cos[2 π], Sin[1 -  $\frac{\pi}{2}$ ]}
Out[3]= {0, 1, -Cos[1]}

```

A partir de los elementos de una lista pueden realizarse directamente operaciones matemáticas elementales.

```

In[4]:= {1, 2, 3, 4, 5} + {a, 1 - a, 1 - a2, 0, 1}
Out[4]= {1 + a, 3 - a, 4 - a2, 4, 6}

In[5]:= 5 {a, 1 - a, 1 - a2, 0, 1}
Out[5]= {5 a, 5 (1 - a), 5 (1 - a2), 0, 5}

```

Cuando los elementos de una lista se generan a través de una expresión matemática o siguen una secuencia, se utilizará la función **Table**, cuya sintaxis es:

Table[f,{var,valores}]

```

In[6]:= Table[n, {n, 8}]
Out[6]= {1, 2, 3, 4, 5, 6, 7, 8}

In[7]:= Table[n2, {n, 10}]
Out[7]= {1, 4, 9, 16, 25, 36, 49, 64, 81, 100}

In[8]:= Table[{n, n2}, {n, 10}]
Out[8]= {{1, 1}, {2, 4}, {3, 9}, {4, 16}, {5, 25},
          {6, 36}, {7, 49}, {8, 64}, {9, 81}, {10, 100}}

```

Esta función admite diversos formatos para determinar el rango de los elementos que se generan. Algunos de ellos son:

Table[f,{i}]
Table[f,{i,i_{max}}]
Table[f,{i, i_{min}, i_{max}}]
Table[f,{i, i_{min}, i_{max}, d}]

```

In[9]:= Table[x, {3}]
Out[9]= {x, x, x}

In[10]:= Table[2x, {x, 3}]
Out[10]= {2, 4, 8}

In[11]:= Table[2x, {x, 3, 6}]
Out[11]= {8, 16, 32, 64}

In[12]:= Table[2x, {x, 3, 9, 2}]
Out[12]= {8, 32, 128, 512}

```

De manera análoga, se generan listas de dimensión superior a uno.

Para ello, utilizamos la función **Table** con la sintaxis:

Table[f,{i, i_{max} }, {j, j_{max}}]

donde f es una función que depende de las variables i y j.

```

In[1]:= Table[i - j2, {i, 3}, {j, 4}]
Out[1]= {{0, -3, -8, -15}, {1, -2, -7, -14}, {2, -1, -6, -13}}

```

Cada elemento de una lista ocupa una posición determinada que podrá determinarse mediante la expresión:

lista[[i]]

que devolverá el elemento que ocupa la posición i en la lista.

```

In[3]:= lista = Table[x - x3, {x, 5}]
Out[3]= {0, -6, -24, -60, -120}

In[4]:= lista[[3]]
Out[4]= -24

```

2.- DEFINICIÓN DE VECTORES Y DE MATRICES

Los vectores y las matrices se consideran estructuras del tipo lista, por lo que su definición se realiza a través de su sintaxis.

Para introducir un vector, escribiremos directamente sus elementos, encerrados entre llaves y separados por comas.

De manera análoga, se define una matriz de dimensión $m \times n$.

```
In[5]:= {1, -2, 4, 0, -1}
Out[5]= {1, -2, 4, 0, -1}

In[6]:= {{1, 3, 4, -1}, {0, -2, 1, 5}, {2, 7, -3, 5}}
Out[6]= {{1, 3, 4, -1}, {0, -2, 1, 5}, {2, 7, -3, 5}}
```

Para expresar la lista anterior en formato matricial, utilizaremos las funciones **TableForm** o **MatrixForm**.

```
In[6]:= {{1, 3, 4, -1}, {0, -2, 1, 5}, {2, 7, -3, 5}}
Out[6]= {{1, 3, 4, -1}, {0, -2, 1, 5}, {2, 7, -3, 5}}

In[7]:= TableForm[%]
Out[7]//TableForm=
  1    3    4    -1
  0   -2    1    5
  2    7   -3    5

In[8]:= MatrixForm[%]
Out[8]//MatrixForm=

$$\begin{pmatrix} 1 & 3 & 4 & -1 \\ 0 & -2 & 1 & 5 \\ 2 & 7 & -3 & 5 \end{pmatrix}$$

```

En la paleta **BasicInput** disponemos del símbolo para introducir matrices cuadradas de orden 2.

```
In[9]:=  $\begin{pmatrix} 2 & 0 \\ -1 & 5 \end{pmatrix}$ 
Out[9]= {{2, 0}, {-1, 5}}

In[10]:= MatrixForm[%]
Out[10]//MatrixForm=

$$\begin{pmatrix} 2 & 0 \\ -1 & 5 \end{pmatrix}$$

```

Una vez situados en cualquiera de los elementos del símbolo de matrices cuadradas de orden 2 de la paleta, se pueden añadir tantas filas y

columnas como precisemos para la obtención de matrices de cualquier orden.

Ctrl - , Añade una columna

Ctrl – Enter Añade una fila

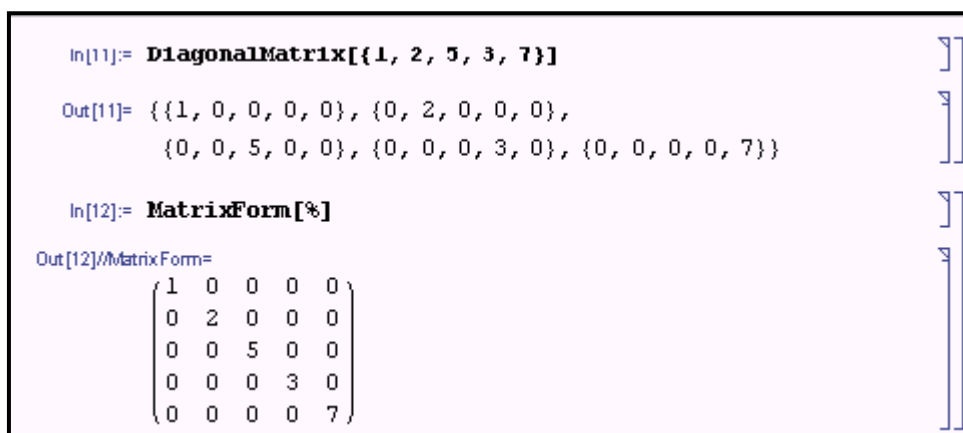


Matriz diagonal

Para definir una matriz diagonal utilizaremos la función **DiagonalMatrix**, cuya sintaxis es:

DiagonalMatrix[lista]

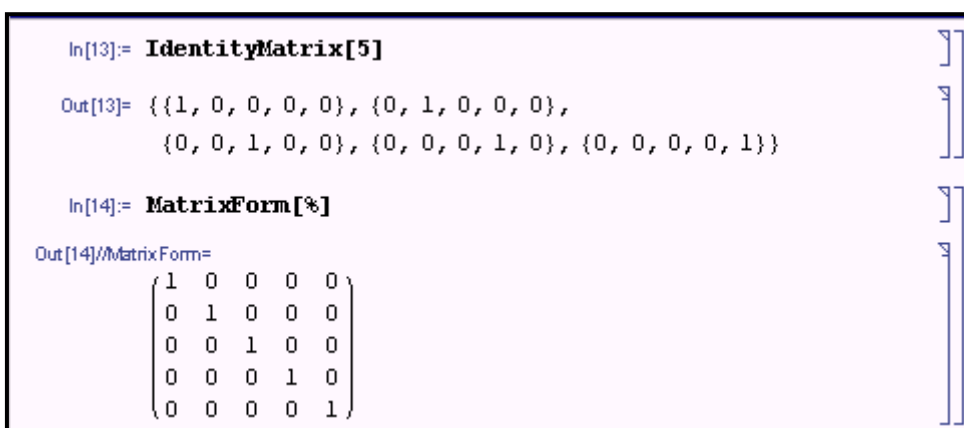
en lista se encuentran los elementos que componen la diagonal de la matriz.



Matriz identidad

La matriz identidad se define a través de la función **IdentityMatrix**, cuyo argumento es el valor de la dimensión.

IdentityMatrix[n]



Matriz nula y matrices triangulares

Utilizando la función **Table**, definiremos cualquier otra matriz, cuyos elementos se generen a través de una determinada ley de formación.

Algunas matrices generadas a partir de esta función son:

- Matriz nula de dimensión nxn : **Table[0,{n},{n}]**
- Matriz triangular superior de dimensión nxn : **Table[If[i>j,f,o],{i,n},{j,n}]**
- Matriz triangular inferior de dimensión nxn : **Table[If[i<j,f,o],{i,n},{j,n}]**

```
In[16]:= MatrixForm[Table[If[i < j, 0, i + j], {i, 4}, {j, 4}]]
Out[16]/MatrixForm=

$$\begin{pmatrix} 2 & 0 & 0 & 0 \\ 3 & 4 & 0 & 0 \\ 4 & 5 & 6 & 0 \\ 5 & 6 & 7 & 8 \end{pmatrix}$$

In[17]:= MatrixForm[Table[If[i > j, 0, i + j], {i, 4}, {j, 4}]]
Out[17]/MatrixForm=

$$\begin{pmatrix} 2 & 3 & 4 & 5 \\ 0 & 4 & 5 & 6 \\ 0 & 0 & 6 & 7 \\ 0 & 0 & 0 & 8 \end{pmatrix}$$

```

3.- OPERACIONES CON VECTORES Y MATRICES

Suma de vectores o de matrices

Si A y B son dos vectores o matrices de la misma dimensión, el vector o la matriz suma se obtiene a partir del operador +.

```
In[18]:=  $\begin{pmatrix} 1 & 3 & 4 & 1 \\ 0 & 2 & 5 & 0 \end{pmatrix} + \begin{pmatrix} 0 & -3 & 0 & 1 \\ 0 & 2 & 1 & 0 \end{pmatrix}$ 
Out[18]= {{1, 0, 4, 2}, {0, 4, 6, 0}}
In[19]:= MatrixForm[%]
Out[19]/MatrixForm=

$$\begin{pmatrix} 1 & 0 & 4 & 2 \\ 0 & 4 & 6 & 0 \end{pmatrix}$$

In[20]:= {{1, 3, 4, 1}, {0, 2, 5, 0}} + {{0, -3, 0, 1}, {0, 2, 1, 0}}
Out[20]= {{1, 0, 4, 2}, {0, 4, 6, 0}}
In[21]:= MatrixForm[%]
Out[21]/MatrixForm=

$$\begin{pmatrix} 1 & 0 & 4 & 2 \\ 0 & 4 & 6 & 0 \end{pmatrix}$$

```

Producto de un vector o de una matriz por un escalar

El producto de un vector o de una matriz por un escalar se realiza por medio del operador multiplicación (*) o insertando un espacio en blanco.

```

In[22]:= 6 *  $\begin{pmatrix} 1 & 3 & 4 & 1 \\ 0 & 2 & 5 & 0 \end{pmatrix}$ 
Out[22]= {{6, 18, 24, 6}, {0, 12, 30, 0}}

In[23]:= MatrixForm[%]
Out[23]/MatrixForm=
 $\begin{pmatrix} 6 & 18 & 24 & 6 \\ 0 & 12 & 30 & 0 \end{pmatrix}$ 

In[26]:= a = {{1, 3, 4, 1}, {0, 2, 5, 0}}; 6 a
Out[26]= {{6, 18, 24, 6}, {0, 12, 30, 0}}

In[27]:= MatrixForm[%]
Out[27]/MatrixForm=
 $\begin{pmatrix} 6 & 18 & 24 & 6 \\ 0 & 12 & 30 & 0 \end{pmatrix}$ 

```

Producto de matrices

El producto de dos matrices se representa mediante el símbolo • (un punto).

Recordaremos que para efectuar el producto de dos matrices, el número de columnas de la primera matriz debe coincidir con el número de filas de la segunda, en caso contrario, aparecerán los correspondientes mensajes de error.

```

In[28]:=  $\begin{pmatrix} 1 & 3 & 4 & 1 \\ 0 & 2 & 5 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & -1 \\ 5 & 1 & 2 \\ 3 & 0 & 0 \\ -1 & 0 & -2 \end{pmatrix}$ 
Out[28]= {{27, 3, 3}, {25, 2, 4}}

In[29]:= MatrixForm[%]
Out[29]/MatrixForm=
 $\begin{pmatrix} 27 & 3 & 3 \\ 25 & 2 & 4 \end{pmatrix}$ 

In[30]:= a = {{1, 3, 4, 1}, {0, 2, 5, 0}};
b = {{1, 0, -1}, {5, 1, 2}, {3, 0, 0}, {-1, 0, -2}};
a.b
Out[31]= {{27, 3, 3}, {25, 2, 4}}

In[32]:= MatrixForm[%]
Out[32]/MatrixForm=
 $\begin{pmatrix} 27 & 3 & 3 \\ 25 & 2 & 4 \end{pmatrix}$ 

```



```

In[33]:=  $\begin{pmatrix} 1 & 3 & 4 & 1 \\ 0 & 2 & 5 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 4 \\ 0 & 3 \end{pmatrix}$ 

Dot::dotsh : Tensors {{1, 3, 4, 1}, {0, 2, 5, 0}}
and {{1, 4}, {0, 3}} have incompatible shapes. More...

Out[33]= {{1, 3, 4, 1}, {0, 2, 5, 0}}.{{1, 4}, {0, 3}}
```

Traspuesta de una matriz

La función **Transpose** calcula la matriz traspuesta de una matriz dada.

```

In[2]:=  $\mathbf{a} = \begin{pmatrix} 1 & 0 & 3 \\ -1 & 2 & 1 \\ 0 & -2 & 3 \\ 5 & 6 & 4 \end{pmatrix}$ 

Out[2]= {{1, 0, 3}, {-1, 2, 1}, {0, -2, 3}, {5, 6, 4}}

In[3]:= Transpose[a]

Out[3]= {{1, -1, 0, 5}, {0, 2, -2, 6}, {3, 1, 3, 4}}

In[4]:= MatrixForm[%]

Out[4]/MatrixForm=
 $\begin{pmatrix} 1 & -1 & 0 & 5 \\ 0 & 2 & -2 & 6 \\ 3 & 1 & 3 & 4 \end{pmatrix}$ 
```

Determinante de una matriz

La función **Det** calcula el determinante de una matriz cuadrada.

```

In[5]:=  $\mathbf{b} = \begin{pmatrix} 1 & 0 & 3 & 1 \\ -1 & 2 & 1 & 2 \\ 0 & -2 & 3 & 1 \\ 5 & 6 & 4 & 3 \end{pmatrix};$ 

In[6]:= Det[b]

Out[6]= 30
```

Inversa de una matriz

La matriz inversa de una matriz cuadrada y regular, se calcula por medio de la función **Inverse**.

```

In[7]:= Inverse[b]

Out[7]= {{-1, -1/3, 2/3, 1/3}, {1, 1/6, -5/6, -1/6},
          {7/5, 1/15, -11/15, -4/15}, {-11/5, 2/15, 23/15, 7/15}}
```

```

In[8]:= MatrixForm[%]

Out[8]/MatrixForm=

$$\begin{pmatrix} -1 & -\frac{1}{3} & \frac{2}{3} & \frac{1}{3} \\ 1 & \frac{1}{6} & -\frac{5}{6} & -\frac{1}{6} \\ \frac{7}{5} & \frac{1}{15} & -\frac{11}{15} & -\frac{4}{15} \\ -\frac{11}{5} & \frac{2}{15} & \frac{22}{15} & \frac{7}{15} \end{pmatrix}$$


In[9]:= MatrixForm[% . b]

Out[9]/MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$


```

Si la matriz es una matriz singular, al intentar calcular la matriz inversa, aparecerá el mensaje de error

```

In[10]:= c =  $\begin{pmatrix} 1 & 3 & 4 \\ 0 & 2 & 2 \\ 2 & 1 & 3 \end{pmatrix}$ ;

In[11]:= Inverse[c]

Inverse::sing :
Matrix {{1, 3, 4}, {0, 2, 2}, {2, 1, 3}} is singular. More...

Out[11]= Inverse[{{1, 3, 4}, {0, 2, 2}, {2, 1, 3}}]

```

Traza de una matriz

La traza de una matriz se obtiene con la función **Tr**

```

In[12]:= Tr[c]

Out[12]= 6

```

Potencia entera de una matriz

La función **MatrixPower** , cuya sintaxis es:

$$\mathbf{MatrixPower[a,n]}$$

calcula la matriz potencia $n \in \mathbb{Z}$ de la matriz "a".

```

In[13]:= d =  $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 2 \\ 2 & 1 & 0 \end{pmatrix}$ ;

In[14]:= MatrixForm[MatrixPower[d, 3]]

Out[14]/MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ 12 & 16 & 12 \\ 6 & 6 & 4 \end{pmatrix}$$


```

```
In[15]:= MatrixForm[MatrixPower[d, -3]]
Out[15]/MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ -3 & -\frac{1}{2} & \frac{3}{2} \\ 3 & \frac{3}{4} & -2 \end{pmatrix}$$

```

4.- RANGO DE UNA MATRIZ

La función **MatrixRank** devuelve el rango de una matriz y la función **RowReduce** transforma la matriz en otra que no contenga filas o columnas linealmente dependientes, haciendo ceros en las filas y columnas.

```
In[17]:= e = 
$$\begin{pmatrix} 1 & 3 & 0 & 3 & 1 \\ 2 & 0 & 1 & 0 & 2 \\ -1 & 4 & 2 & 1 & 2 \\ 0 & 1 & -2 & 3 & 1 \\ 5 & 2 & 6 & 4 & 3 \end{pmatrix};$$

In[18]:= MatrixRank[e]
Out[18]= 5
In[19]:= MatrixForm[RowReduce[e]]
Out[19]/MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

```

5.- RESOLUCIÓN DE SISTEMAS DE ECUACIONES LINEALES

La función **LinearSolve**, cuya sintaxis es **LinearSolve[A,B]**, resuelve el sistema de ecuaciones lineales expresado matricialmente por $AX = B$.

```
In[20]:= A = 
$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 2 \\ 2 & 1 & 0 \end{pmatrix};$$
 B = 
$$\begin{pmatrix} 4 \\ 2 \\ 1 \end{pmatrix};$$

In[21]:= LinearSolve[A, B]
Out[21]= {{4}, {-7}, {8}}
```

Por tanto, la solución del sistema de ecuaciones es: $X = 4$, $Y = -7$, $Z = 8$.

Al usar esta función se debe considerar que, para un sistema compatible indeterminado, sólo devolverá una solución particular del sistema.

Por tanto, siempre que se aplique la función **LinearSolve**, se tendrá que averiguar si el resultado corresponde a la solución del sistema (SCD) o a una solución particular (SCI).

Para ello, utilizaremos la función **NullSpace**, sobre la matriz de los coeficientes, para determinar el conjunto de vectores que verifican la expresión $AX=0$; resultado que nos proporcionará el rango de dicha matriz de los coeficientes y nos permitirá determinar si el sistema es determinado o indeterminado.

```
In[1]:= A =  $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 2 \\ 2 & 1 & 0 \end{pmatrix}$ ; B =  $\begin{pmatrix} 4 \\ 2 \\ 1 \end{pmatrix}$ ;

In[2]:= LinearSolve[A, B]

Out[2]:= {{4}, {-7}, {8}}

In[3]:= NullSpace[A]

Out[3]:= {}
```

La única solución para el sistema homogéneo es la trivial, por lo que la solución $X = 4, Y = -7, Z = 8$ es única.

Utilizando este razonamiento para este otro ejemplo, obtenemos los resultados que aparecen a continuación, que nos indica que el sistema es incompatible.

```
In[7]:= A =  $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 2 \\ 2 & 1 & 3 \end{pmatrix}$ ; B =  $\begin{pmatrix} 4 \\ 2 \\ 1 \end{pmatrix}$ ;

In[8]:= LinearSolve[A, B]

LinearSolve::nosol :
Linear equation encountered which has no solution. More...

Out[8]:= LinearSolve[
{{1, 0, 1}, {0, 2, 2}, {2, 1, 3}}, {{4}, {2}, {1}}]
```

Realizando un procedimiento similar al efectuado en los ejemplos anteriores, deducimos que la solución corresponde a una solución particular; por lo que el sistema es compatible indeterminado.

```
In[16]:= A =  $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 2 \\ 1 & 2 & 3 \end{pmatrix}$ ; B =  $\begin{pmatrix} 4 \\ 2 \\ 6 \end{pmatrix}$ ;

In[17]:= LinearSolve[A, B]

Out[17]:= {{4}, {1}, {0}}

In[18]:= NullSpace[A]

Out[18]:= {{-1, -1, 1}}
```

Para obtener la expresión de las soluciones del sistema, utilizaremos el comando **Solve**.

```
In[19]:= Solve[{x + z == 4, 2 y + 2 z == 2, x + 2 y + 3 z == 6}]
Solve::svars : Equations may not
give solutions for all "solve" variables. More...
Out[19]= {{x -> 3 + y, z -> 1 - y}}
```

6.- EJEMPLO

EJEMPLO 1

Sea la matriz real:

$$A = \begin{pmatrix} q & 0 & 1 & q \\ 1 & 1 & -p & q \\ p+q & p & 0 & 2q \end{pmatrix}$$

Hallar el rango de la matriz A según los distintos valores de los parámetros p y q .

```
In[1]:= a = {{q, 0, 1, q}, {1, 1, -p, q}, {p + q, p, 0, 2 q}};
In[2]:= RowReduce[a]
Out[2]:= {{1, 0, 0, (2 + p)/(1 + p)}, {0, 1, 0, (-2 - p + q)/(1 + p)}, {0, 0, 1, -q/(1 + p)}}
In[3]:= MatrixRank[a]
Out[3]:= 3
```

Podemos observar como al depender los elementos de la matriz A de parámetros, no se puede utilizar las funciones **RowReduce** o **MatrixRank**.

Localizaremos un menor de orden 2 no nulo e iremos "orlando".

```
In[4]:= m2 = Det[a[{{1, 2}, {2, 3}}]]
Out[4]= -1
Menor M2 no nulo.

In[5]:= m3 = Det[a[{{1, 2, 3}, {1, 2, 3}}]]
Out[5]= -q + p2 q
In[6]:= Factor[%]
Out[6]= (-1 + p) (1 + p) q

In[7]:= m31 = Det[a[{{1, 2, 3}, {2, 3, 4}}]]
Out[7]= -2 q + p q + p2 q
In[8]:= Factor[%]
Out[8]= (-1 + p) (2 + p) q
```

Estos son los menores orlados de M_2 . Ahora debemos comprobar cuando se anulan simultáneamente.

```
In[9]:= Solve[m3 == 0 && m31 == 0, {p, q}]  
  
Solve::svars : Equations may not give solutions for all "solve" variables. More...  
  
Out[9]:= {{p -> 1}, {q -> 0}}
```

Conclusión:

- Si $p = 1 \vee q = 0$: $m3 = m31 = 0 \Rightarrow r(A) = 2$
- Si $p \neq 1 \wedge q \neq 0$: $m3$ o $m31$ no nulos $\Rightarrow r(A) = 3$