

PRÁCTICA 5. MATRICES. SISTEMAS DE ECUACIONES LINEALES. APLICACIONES LINEALES. DIAGONALIZACIÓN

1 INTRODUCCIÓN.

El alumno debe ir realizando todos los EJEMPLOS que aparecen en esta práctica e ir anotando los resultados en la hoja que se le ha entregado. Además de estos ejemplos, hay planteados unos EJERCICIOS, que, a diferencia de lo que ocurre con los EJEMPLOS, no vienen desarrollados en el lenguaje Maxima, sino que tendrá que ser el alumno el que tenga que introducirlos para obtener el resultado.

2 DEFINICIÓN DE UNA MATRIZ.

Las matrices en Maxima son expresiones que comprenden el operador "matrix" y cuyos argumentos son listas. Esto indica que Maxima tiene establecida una diferencia entre una matriz y una lista de listas. Así

```
matrix([a11, a12, . . . , a1n], . . . , [am1, am2, . . . , amn]);
```

nos dará una matriz de orden $m \times n$, mientras que

```
[[a11, a12, . . . , a1n], . . . , [am1, am2, . . . , amn]];
```

nos dará una lista de vectores.

```
--> /*EJEMPLO 1*/
matrix([3,1,0],[2,4,1],[1,7,2]);
```

```
--> /*EJEMPLO 2*/
[[3,1,0],[2,4,1],[1,7,2]];
```

Con "apply", una lista de vectores se puede convertir en una matriz:

```
--> /*EJEMPLO 3*/
apply(matrix,[[3,1,0],[2,4,1],[1,7,2]]);
```

NOTA: En toda esta Práctica, y al igual que ocurre en las demás, podemos tener una gran ayuda si en lugar de usar un determinado comando (por ejemplo para escribir una matriz) usamos el menú que tiene Maxima en su parte superior. En esta Práctica puede ser útil usar el menú wxMaxima->Álgebra.

El comando "genmatrix(f,m,n)" devuelve una matriz $m \times n$ generada a partir de f. Por ejemplo, para generar una matriz $(a_{ij})_{3 \times 3}$, tal que $a_{ij}=i-2j$:

```
--> /*EJEMPLO 4*/
a[i,j]:=i-2*j $
genmatrix(a,3,3);
```

Maxima tiene comandos específicos para definir matrices concretas:

- "ident(n)" devuelve la matriz identidad de orden n.
- "zeromatrix(m,n)" devuelve una matriz rectangular m x n con todos sus elementos iguales a cero.
- "diagmatrix(n,elem)", devuelve una matriz diagonal de orden n con los elementos de la diagonal todos ellos iguales a elem.
- "diag_matrix(d1,d2,...,dn)" devuelve una matriz diagonal con los elementos de la diagonal iguales a d1,d2,...,dn.

```
--> /*EJEMPLO 5*/
      ident(5);
      zeromatrix(3,4);
      diagmatrix(4,-1);
      diag_matrix(1,2,-3);
```

Podemos elegir elementos específicos en una matriz, con los comandos:

- "M[i]" o "part(M,i)" nos da la fila i-sima de la matriz M.
- "M[i,j]" o "part(M,i,j)" nos da el elemento (i,j) de la matriz M.

Por ejemplo, si de matriz generada anteriormente queremos escoger la fila 3 y el elemento (1,3):

```
--> /*EJEMPLO 6*/
      a[i,j]:=i-2*j $
      A:genmatrix(a,3,3);
      A[3];
      A[1,3];
```

3 OPERACIONES CON MATRICES.

Los principales comandos para operar con matrices son:

- "A.B" nos da el producto de A y B.
- "invert(M)" devuelve la inversa de la matriz M.
- "determinant(M)" devuelve el determinante de la matriz M.
- "transpose(M)" devuelve la transpuesta de la matriz M.
- "minor(M,i,j)" devuelve el menor (i,j) de la matriz M.
- "adjoint(M)" devuelve la matriz adjunta de la matriz M.
- "col(M,i)" devuelve la i-ésima columna de la matriz M.
El resultado es una matriz de una sola columna.
- "row(M,i)" devuelve la fila i-ésima la de la matriz M.
El resultado es una matriz de una fila.
- "rank(M,i,j)" calcula el rango de la matriz M.
- "triangularize(M)" devuelve la forma triangular superior de la matriz M, obtenida por eliminación gaussiana.

(Nota: podemos ver muchos más comandos en cualquier manual de Maxima, por ejemplo en <http://www.unp.edu.pe/pers/ripanaque/download/manual.pdf>)

```
--> /*EJEMPLO 7*/
      /*Definimos una matriz cualquiera 2x2 y calcularemos
      su traspuesta, su inversa y su determinante*/
      A:matrix([a,b],[c,d]);
      transpose(A);
      invert(A);
      determinant(A);
```

Sabemos que el dominio de $g(x)$ es todo \mathbb{R} , puesto que el denominador no se anula nunca, aunque si hubiese duda resolvemos la ecuación que aparece en su denominador

```
--> /*EJEMPLO 8*/
/*Definimos una matriz B 3x3 y calcularemos
su inversa C y comprobamos que efectivamente B y C son
inversas*/
B:genmatrix(lambda([i,j],1/(i+j-1)),3,3);
C:invert(B);
B.C;
C.B;
```

Nota: El producto de matrices viene indicado con ".", y NO con "*" como suele ser habitual para el producto de números o letras. Observemos en el siguiente ejemplo la diferencia entre ambas:

```
--> /*EJEMPLO 8.1*/
/*Definimos dos matrices cualesquiera 2x2 y calcularemos
A.B y A*B para ver la diferencia*/
A:matrix([a,b],[c,d]);
B:matrix([e,f],[g,h]);
A.B;
A*B;
```

Nota: Igual ocurre cuando pretendemos calcular la potencia de una matriz: En este caso no se usará A^2 (como ocurre con los números o variables), sino que usaremos $A^{^2}$. Lo vemos a continuación:

```
--> /*EJEMPLO 8.2*/
/*Para la matriz A anterior calculamos A^3 y A^{^3} para ver
la diferencia*/
A^3;
A^{^3}
```

EJERCICIO 1: Hallar el rango de la siguiente matriz A. Hacerlo directamente y consiguiendo su matriz triangular superior:

```
A=(1,8,-6,-2)
(2,-2,3,2)
(1,2,-1,0)
```

EJERCICIO 2: Idem para la matriz

```
B=(0,1,-4,3)
(2,5,0,-2)
(1,3,-2,0)
(1,2,2,-2)
```

EJERCICIO 3: Hallar una base para el subespacio de \mathbb{R}^4 generado por los vectores:

```
(-1,3,2,4), (1,-3,-2,-4), (1,0,-2,-1), (0,3,0,5).
```

EJERCICIO 4: Idem para el generado por los vectores:

```
(0,0,0,4), (0,0,-2,-4), (1,0,-2,1), (4,0,2,3), (1,0,0,5).
```

4 SISTEMAS DE ECUACIONES LINEALES.

4.1 MATRICES ASOCIADAS A UN SISTEMA DE ECUACIONES LINEALES:

Son de destacar los dos siguientes comandos:

- "coefmatrix([eq1,...,eqm],[x1,...,xn])", devuelve la matriz de coeficientes para las variables x_1, \dots, x_n del sistema de ecuaciones lineales dado por eq_1, \dots, eq_m .
- "augcoefmatrix([eq1,...,eqm],[x1,...,xn])", devuelve la matriz aumentada de coeficientes para las variables x_1, \dots, x_n del sistema de ecuaciones lineales eq_1, \dots, eq_m .

Con la matriz de coeficientes y la matriz ampliada podremos saber como es un sistema lineal (SCD, SCI o SI) sin más que calcular sus respectivos rangos. Lo vemos en el ejemplo siguiente:

```
--> /*EJEMPLO 9: Para el sistema lineal dado por las ecuaciones
2x-3y=1; x+y=3, hallamos su matriz del sistema y la matriz
ampliada*/
/*Despues calculamos sus rangos y vemos que se trata de un
SCD*/
A:coefmatrix([2*x-3*y=1,x+y=3],[x,y]);
AM:augcoefmatrix([2*x-3*y=1,x+y=3],[x,y]);
rank(A);
rank(AM);
```

4.2 RESOLUCIÓN DE UN SISTEMA DE ECUACIONES LINEAL

Los sistemas de ecuaciones lineales podemos resolverlos usando "linsolve([eq1,...,eqm],[x1,...,xn])"

Nota: También podemos introducir y resolver un sistema lineal usando el menú superior de wxMaxima->Ecuaciones->Resolver sistema lineal.

```
--> /*EJEMPLO 9.1: Podemos resolver el sistema anterior*/
linsolve([2*x-3*y=1,x+y=3],[x,y]);
```

```
--> /*EJEMPLO 9.2: Podemos resolver sistemas indeterminados*/
linsolve([2*x-3*y+z=1,x+y-z=3],[x,y]);
```

```
--> /*EJEMPLO 9.3: Y si intentamos resolver un sistema incompatible
el programa nos avisa*/
linsolve([2*x-3*y=1,2*x-3*y=3],[x,y]);
```

EJERCICIO 5: Discutir y resolver los siguientes sistemas lineales:

- | | | |
|--------------|-------------------|-------------------|
| a) $x-y+z=2$ | b) $x+2y-z+t+u=0$ | c) $x+2y+z+3t=45$ |
| $3x+2y-2z=1$ | $3x-y+t-u=6$ | $x+y+z+4t=48$ |
| $-x+3y-z=2$ | $6x+y+t+u=1$ | $2x+y+4z+10t=101$ |
| | $x-2y+2z-2t=-5$ | $-x-3y+7z+5t=-4$ |

Saber resolver sistemas lineales nos permite hacer algunas cosas más relacionadas con vectores. Así por ejemplo, para saber si un vector se pone o no como combinación lineal de otros, y, en su caso, hallar los escalares que permiten hacer esto, lo que tenemos que hacer es precisamente resolver un sistema lineal. Un caso particular de esta situación se presenta cuando queremos hallar las coordenadas de un vector en una base, o, incluso, cuando queremos hallar la matriz cambio de base de dos bases de un espacio vectorial.

EJERCICIO 6: Responder a las siguientes preguntas:

1. Comprobar si el vector $(1, -1, 0, 2)$ es combinación lineal de los vectores $(0, -2, 3, 1)$ y $(1, 0, 0, 1)$, y en caso afirmativo, hallar los escalares que permiten dicha combinación lineal.

2. Hallar las coordenadas del vector $(1, -1, 0)$ en la base $\{(1, 2, 3), (1, 3, 0), (3, 2, 2)\}$ de \mathbb{R}^3 .

3. Dadas las bases de \mathbb{R}^3

$B_1 = \{(1, 0, 1), (1, 1, 0), (0, 1, 1)\}$ y $B_2 = \{(1, 2, 3), (1, 3, 0), (3, 2, 3)\}$ hallar las matrices cambio de base de B_1 a B_2 y de B_2 a B_1 .

El alumno debe ir realizando todos los EJEMPLOS que aparecen en esta práctica e ir anotando los resultados en la hoja que se le ha entregado. Además de estos ejemplos, hay planteados unos EJERCICIOS, que, a diferencia de lo que ocurre con los EJEMPLOS, no vienen desarrollados en el lenguaje Maxima, sino que tendrá que ser el alumno el que tenga que introducirlos para obtener el resultado.

5 APLICACIONES LINEALES.

5.1 Matriz asociada a una aplicación lineal.

Una vez que sabemos introducir matrices y resolver sistemas de ecuaciones lineales, podemos resolver la mayoría de ejercicios relacionados con aplicaciones lineales. Comenzamos introduciendo la matriz asociada a una aplicación lineal:

```
--> /*EJEMPLO 10: Introducir la matriz asociada a la a. lineal
f:R^4->R^3 dada por f(x,y,z,t)=(x-z,0,y-2z+t)*/
f(x,y,z,t):=(x-z,0,y-2*z+t);
B:apply(matrix,[f(1,0,0,0),f(0,1,0,0),f(0,0,1,0),f(0,0,0,1)]);
A:transpose(B);
```

NOTA: Seguramente será más rápido introducir directamente la matriz vista la expresión de la aplicación lineal:

```
--> /*EJEMPLO 10 BIS: Introducir la matriz asociada a la a. lineal
f:R^4->R^3 dada por f(x,y,z,t)=(x-z,0,y-2z+t)*/
A:matrix([1,0,-1,0],[0,0,0,0],[0,1,-2,1]);
```

5.2 Imagen y Núcleo de una aplicación lineal.

Sabemos que la $\text{Im}(f)$ está generada por las columnas de la matriz de f , por lo que si escogemos las que son linealmente independientes, obtendremos una base para el subespacio vectorial $\text{Im}(f)$. Sin embargo, en wxMaxima se puede obtener más fácilmente una base para $\text{Im}(f)$ sin más que usar directamente el comando "columnspace(A)" junto con "args(%)":

```
--> /*EJEMPLO 11: Vamos a hallar el subespacio Im(f), siendo
f la apl. lineal anterior*/
columnspace(A);
args(%);
```

De esta forma observamos que $\dim(\text{Im}(f))=2$, por lo que f no puede ser suprayectiva (de paso, también sabemos que $\dim(\text{ker}(f))=2$). De igual forma podemos hallar $\text{ker}(f)$ (y sin necesidad de resolver un sistema de ecuaciones $A.X=0$), usando el comando "nullspace(A)" junto con "args(%)":

```
--> /*EJEMPLO 12: Vamos a hallar el subespacio ker(f), siendo
      f la apl. lineal anterior*/
      nullspace(A);
      args(%)
```

5.3 Cambio de base en una aplicación lineal.

Aunque ésto ya se vió en el ejercicio 6.3 de la práctica anterior, volvemos a incluirlo por su gran utilidad para los ejercicios planteados en clase y en exámenes. Recordamos que para efectuar un cambio de base en una a. lineal, realizamos un esquema

$$\begin{array}{ccccccc} V & \xrightarrow{B1} & V & \xrightarrow{C} & W & \xrightarrow{B2} & W \\ & & C & & C & & B2 \end{array}$$

y que la nueva matriz en las bases $B1, B2$ viene dada por $MB1, B2(f) = MC, B2(\text{id}) \times MC, C(f) \times MB1, C(\text{id})$

```
--> /*EJEMPLO 13: Vamos a hallar la nueva matriz asociada a f
      cuando se considera en R^4 la base canónica y en R^3 la
      nueva base dada por B={ (1,1,0), (1,1,1), (1,0,0) }*/
      A:matrix([1,0,-1,0],[0,0,0,0],[0,1,-2,1]);
      MBC_id:matrix([1,1,1],[1,1,0],[0,1,0]);
      MBC_id.A;
```

En el ejemplo anterior la nueva matriz se ha calculado realizando el producto matricial $MC, B(f) = MB, C(\text{id}) \times A$ (notemos que, en este ejemplo, en el conjunto inicial R^4 seguimos teniendo la base canónica, por lo que solo se cambia de base en el conjunto final R^3).

EJERCICIO 7: Dada la aplicación lineal $f: R^3 \rightarrow R^2$ definida por $f(x, y, z) = (x-y-z, x-y-z)$

- Escribir la matriz asociada a f en las bases canónicas.
- Hallar bases para $\text{Im}(f)$ y $\text{ker}(f)$. Clasificar f .
- Hallar la nueva matriz asociada a f cuando se consideran las nuevas bases $B1$ y $B2$ dadas por $B1 = \{(1, 1, 0), (1, 1, 1), (1, 0, 0)\}$ y $B2 = \{(2, -1), (3, 1)\}$

6 VALORES Y VECTORES PROPIOS. DIAGONALIZACIÓN.

Para calcular los valores y vectores propios asociados a una matriz cuadrada (lo que nos llevará a saber si dicha matriz es o no diagonalizable), usaremos los siguientes comandos:

- "charpoly(M,x)", que calcula el polinomio característico vinculado a la matriz M , en términos de la variable x .
- "eigenvalues(M)", que devuelve una lista con dos sublistas, la primera de éstas conteniendo los valores propios de la matriz M y la segunda, conteniendo sus correspondientes multiplicidades.
- "eigenvectors(M)", que devuelve una lista con dos elementos; el primero está formado por los valores propios de M , y el segundo es una lista de listas de vectores propios, una por cada valor propio, pudiendo haber más de un vector propio en cada lista.

```
--> /*EJEMPLO 14: Para la siguiente matriz D, calculamos su polinomio
característico*/
D:matrix([3,4,5],[4,5,6],[5,6,7]);
charpoly(D,x);
```

```
--> /*EJEMPLO 14 (cont): Para la anterior matriz D, calculamos sus
valores propios y observamos que tiene 3 valores propios reales
y diferentes (cada uno con multiplicidad 1)*/
eigenvalues(D);
```

```
--> /*EJEMPLO 14 (cont): Para la anterior matriz D, calculamos sus
vectores propios asociados a sus respectivos valores propios*/
eigenvectors(D);
```

Notemos que en este último caso hemos obtenido 4 vectores: el primero de ellos son los valores propios de D, con sus multiplicidades; los 3 restantes son los vectores propios asociados, respectivamente, a cada valor propio.

EJERCICIO 8: Definir la siguiente matriz E. Calcular sus valores y vectores propios, y deducir porqué la matriz no es diagonalizable:

```
E=(1,0,0,0,0,0)
(1,1,0,0,0,0)
(0,1,1,0,0,0)
(0,0,0,2,0,0)
(0,0,0,1,2,0)
(0,0,0,0,0,-1)
```

(Solución: Observar que aunque todos los valores propios son reales (-1 tiene mult 1; el 1 tiene mult. 3 y el 2 tiene mult. 2), cada uno de ellos solo tiene un único vector propio asociado).

EJERCICIO 9: Para cada una de las siguientes matrices calcular sus polinomios característicos, sus valores y vectores propios y deducir si son o no diagonalizables.

Para las que sean diagonalizables deducir cual es su matriz diagonal asociada D y cual es la matriz de paso P.

En este último caso, comprobar que se verifica que $\text{invert}(P) \cdot A \cdot P = D$:

```
A=( 0,1,-1)      B=(2,0,-1)      C=(1, 0,-1)
(-1,3, 1)      (0,3, 0)      (1,-1, 0)
( 1,2, 0)      (5,0,-4)      (1,-1,-1)
```