

G423. Álgebra y Geometría. Prácticas MATLAB

Ruth Carballo Fidalgo (UC). Curso 21-22

Contents

P.1	Semana 14 de febrero	3
P.2	Semana 21 de febrero	10
P.3	Semana 28 de febrero	14
P.4	Semana 7 de marzo	20
P.5	Semana 14 de marzo	24
P.6	Semana 21 de marzo	26
P.7	Semana 28 de marzo	31
P.8	Semana 4 de abril	37
P.9	Semana 11 de abril	42
P.10	Semana 25 de abril	47
P.11	Semana 2 de mayo	48
P.12	Semana 9 de mayo	49
P.13	Semana 16 de mayo	54
P.14	Semana 23 de mayo	57

[Introducción a Matlab \(Versión 2017\)](#) preparada por R. Carballo. Incluida como referencia. Las instrucciones requeridas como contenidos de la asignatura estarán especificadas para cada práctica.

Descarga de MATLAB en tu ordenador utilizando la Licencia de la UC

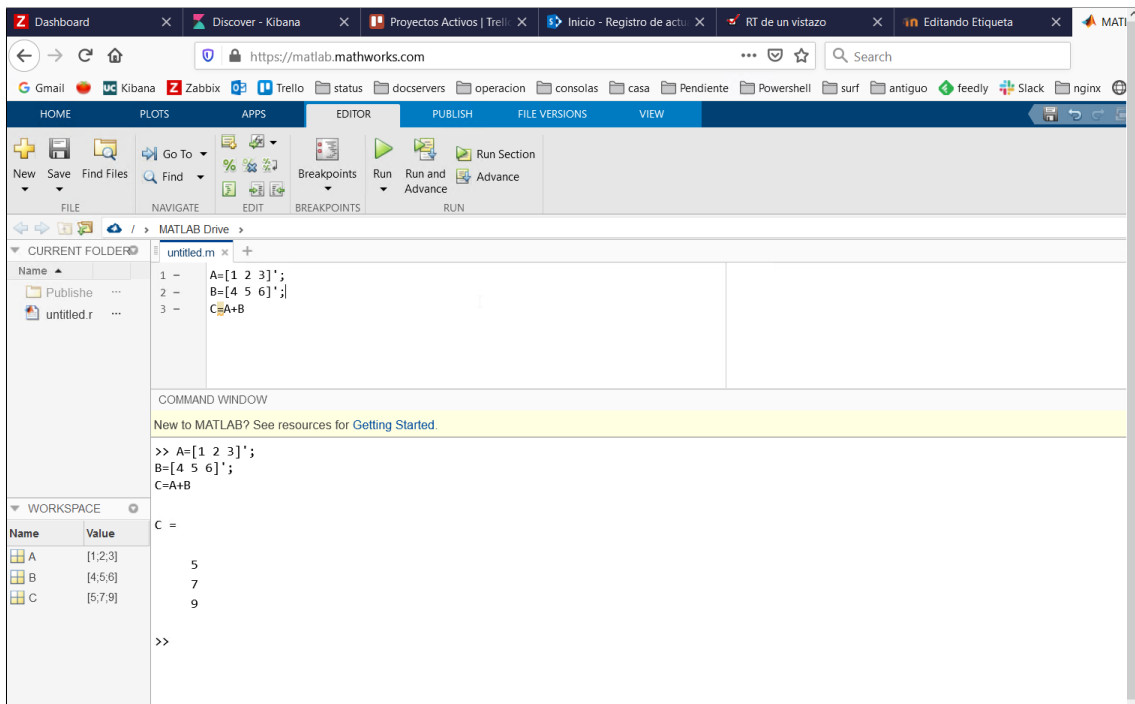
Para poder descargarte el software con la licencia MATLAB de la Universidad de Cantabria tienes que abrir previamente una cuenta en MathWorks www.mathworks.com con la dirección de correo de la universidad pero eligiendo una contraseña distinta de la de la Universidad.

Una vez dispongas de esa cuenta, debes conectarte a este [enlace](#) del Servicio de Informática de la UC, que te da las instrucciones. En la parte superior derecha del enlace, junto con el icono de MATLAB, te aparece el enlace [Portal Matlab de la UC](#) desde donde arranca el proceso de acreditación y descarga.

Uso de MATLAB online (por Valvanuz Fernández)

Es posible utilizar MATLAB desde cualquier ordenador sin necesidad de instalarlo. Basta con disponer de un navegador (chrome, edge, Mozilla,...) y de las credenciales utilizadas para descargarse MATLAB de la página de Mathworks.

Para poder utilizarlo hay que acceder al portal [MATLAB Online](https://matlab.mathworks.com) y validarse con las credenciales de Mathworks. Una vez dentro aparecerá una instancia de MATLAB muy similar a la que aparece cuando lo instalamos en nuestros ordenadores y que nos permitirá trabajar igual que lo hacemos en estos.



MATLAB online puede ser muy útil cuando tenemos un equipo en el que no está instalado MATLAB o cuando nuestro equipo es muy viejo y tarda mucho en ejecutar las órdenes. Una de las principales ventajas de MATLAB online es que las órdenes se "calculan" en la nube, no en nuestro equipo.

P.1 Semana 14 de febrero

Autores: Ruth Carballo, Valvanuz Fernández

Las instrucciones de MATLAB, incluidos los comentarios, están escritos en azul, y las salidas en rojo.

Ejercicio 1.1.5

Obtén la solución $\vec{x} = (x_1, x_2, x_3, x_4, x_5)$ de los sistemas dados ¹, utilizando el método de Gauss-Jordan. Presenta la solución en forma vectorial paramétrica, es decir, como $\vec{x} = \vec{p} + \alpha_1 \vec{v}_1 + \dots + \alpha_k \vec{v}_k$, siendo \vec{v}_i vectores numéricos específicos de \mathbb{R}^5 y α_i parámetros libres (cualquier elección de los reales $\alpha_1, \dots, \alpha_k$ produce una solución válida del sistema).

$$I: \begin{cases} x_2 - 2x_3 + 2x_4 & = -7 \\ x_1 + x_2 - 4x_3 + 5x_4 + 2x_5 & = -23 \\ -x_1 + 3x_2 - 4x_3 + 3x_4 & = 3 \end{cases}$$

$$II: \begin{cases} x_2 - 2x_3 + 2x_4 & = 0 \\ x_1 + x_2 - 4x_3 + 5x_4 + 2x_5 & = 0 \\ -x_1 + 3x_2 - 4x_3 + 3x_4 & = 0 \end{cases}$$

`% CLASIFICACION Y SOLUCION DEL SISTEMA LINEAL.`

`A=[0 1 -2 2 0; 1 1 -4 5 2; -1 3 -4 3 0]; % A Matriz de coeficientes`

`% b vector de terminos independientes (columna)`
`b=[-7 -23 3]';`

`% CLASIFICACION:`

`Am=[A b] % Matriz ampliada concatenando A con b a la derecha`

```
Am =  
    0     1    -2     2     0    -7  
    1     1    -4     5     2   -23  
   -1     3    -4     3     0     3
```

`% La existencia o no de solucion/es se determina mediante el Teorema`
`% de Rouche-Frobenius`

`rank(A) % rango = numero de cols. pivotaes = numero de vectores l.i.`
`rank(Am)`

`% El sistema tiene solucion si esos rangos son iguales`

`n=size(A,2) % n es el numero de cols de A = numero de incognitas del SL`

`% En caso de que el sistema sea compatible, sera indeterm. si n>rango(A)`

`% siendo n-rango(A) el grado de la indeterminacion.`

`% Sera determinado si n=rango(A)`

¹El SL II se dice que es el correspondiente homogéneo del SL I

```

ans =
    3
ans =
    3
n =
    5

% el Sistema es compatible indeterminado con 5-3=2 parametros libres

% SOLUCION MEDIANTE METODO GAUSS-JORDAN:

% A continuacion aplicamos Gauss-Jordan para obtener
% el sistema equivalente mas sencillo posible.
[U j]=rref(Am) % U proporciona la forma escalonada reducida,
               % que dada Am es unica.
               % U es la matriz ampliada del sistema equivalente
               % mas sencillo posible.
               % U viene de "upper", porque la matriz al estar escalonada
               % tiene ceros debajo de la diagonal principal.
               % rref viene de "reduced row echelon form"
               % j es un vector fila que da los indices de las cols pivotaes
               % por lo que length(j), que es el numero de entradas de j,
               % es el rango de la matriz que estamos estudiando, Am.

length(j)

U =
    1    0   -2    3    0   -24
    0    1   -2    2    0    -7
    0    0    0    0    1    4

j =
    1    2    5

ans =
    3

% Obtenemos obviamente el mismo rango de Am que antes, 3.
% Al ser las columnas pivotaes la primera, la segunda y la quinta tomaremos:
%   x1, x2, x5   variables basicas, tambien llamadas incognitas principales.
%   x3, x4     variables libres, tambien llamadas parametros libres.
%
% Procedimiento 1) A partir de U despejamos las incognitas principales.
% Antes declaramos los parametros libres

syms x3 x4 real           % declaramos

x1 = -24 + 2*x3 - 3*x4 ;   % despejamos
x2 = -7 + 2*x3 - 2*x4 ;   % despejamos
x3 = x3                    ;
x4 = x4                    ;
x5 = 4                     ;   % despejamos

```

```
sol=[x1 x2 x3 x4 x5]' % solucion en forma vectorial
pretty(sol)           % formato mas legible para datos simbolicos
```

```
sol =
```

```

2*x3 - 3*x4 - 24
 2*x3 - 2*x4 - 7
                x3
                x4
                4
```

```

/ 2 x3 - 3 x4 - 24 \
|                    |
| 2 x3 - 2 x4 - 7 |
|                    |
|          x3      |
|                    |
|          x4      |
|                    |
\          4        /
```

```
% La solucion en forma vectorial parametrica, separando el vector constante
% y cada vector que queda multiplicado por un parametro libre, es:
%
```

```
% solucion = p + x3*v1 + x4*v2 siendo
p = [-24 -7 0 0 4]'
v1 = [ 2  2 1 0 0]'
v2 = [-3 -2 0 1 0]'
```

```

p =
-24
 -7
  0
  0
  4
```

```

v1 =
 2
 2
 1
 0
 0
```

```

v2 =
-3
-2
 0
 1
 0
```

```
[p v1 v2] , rank(ans) % comprobamos que esos tres vectores son l.i.,
                        % es decir, rango 3.
```

```
ans =
    -24     2    -3
     -7     2    -2
      0     1     0
      0     0     1
      4     0     0
```

```
ans =
      3
```

```
% La solución es  $x = (-24, -7, 0, 0, 4) + x_3(2, 2, 1, 0, 0) + x_4(-3, -2, 0, 1, 0)$ 
```

```
% Al tener dos parámetros libres la solución la forman las infinitas
% combinaciones lineales de dos direcciones independientes, trasladadas
% mediante un vector p distinto de cero, y también independiente de los
% dos anteriores.
```

```
% Procedimiento 2). Una vez escogidos los parámetros libres,  $x_3$ ,  $x_4$ , podemos
% obtener la solución con la función solve()
```

```
x=[x1;x2;x3;x4;x5]
sol=solve(A*x-b,[x1,x2,x5]) % también se podría escribir como
                            % solve(A*x-b==0,[x1,x2,x5]) o como
                            % solve(A*x==b,[x1,x2,x5])
```

```
struct with fields:
    x1: 2*x3 - 3*x4 - 24
    x2: 2*x3 - 2*x4 - 7
    x5: 4
```

```
[sol.x1; sol.x2; x3; x4; sol.x5]
```

```
2*x3 - 3*x4 - 24
2*x3 - 2*x4 - 7
      x3
      x4
      4
```

```
% SOLUCIÓN DEL CORRESPONDIENTE HOMOGÉNEO
```

```
% Aplicando el procedimiento anterior con  $c = [0 \ 0 \ 0]'$  (es decir, con
% igual A, la misma función rref(), pero los términos independientes
% cero), se obtendría como solución:
```

```
% sol =  $x_3v_1 + x_4v_2$  con los mismos  $v_1$  y  $v_2$  que en el SL no homogéneo
```

```
% x = x3*(2,2,1,0,0) + x4*(-3,-2,0,1,0)
```

```
% Veamoslo, aunque sea obvio:
```

```
c=[0 0 0]'; Amh=[A c] , rref(Amh)
```

```
Amh =
```

```
    0    1   -2    2    0    0
    1    1   -4    5    2    0
   -1    3   -4    3    0    0
```

```
ans =
```

```
    1    0   -2    3    0    0
    0    1   -2    2    0    0
    0    0    0    0    1    0
```

```
% Aqui tendríamos que despejar como antes x1, x2 y x5 en función  
% de los parámetros, obteniendo
```

```
% x = x3*(2,2,1,0,0) + x4*(-3,-2,0,1,0)
```

```
% Lo podríamos hacer directamente con:
```

```
sol=solve(A*x,[x1,x2,x5])
```

```
struct with fields:
```

```
    x1: 2*x3 - 3*x4
    x2: 2*x3 - 2*x4
    x5: 0
```

```
[sol.x1; sol.x2; x3; x4; sol.x5]
```

```
2*x3 - 3*x4
```

```
2*x3 - 2*x4
```

```
    x3
```

```
    x4
```

```
    0
```

```
% METODO ALTERNATIVO PARA LA RESOLUCION DE SISTEMAS HOMOGENEOS:
```

```
% null(Matriz Coeficientes) El nombre de null viene de nulo, o cero,  
% ya que resolvemos [A | 0]
```

```
% Hemos visto que la solución general del SLH no es más que el conjunto de  
% todas las combinaciones lineales de dos soluciones independientes v1 y v2.
```

```
% La función null() nos dará directamente esas dos soluciones v1 y v2.
```

```
% Si hubiera más parámetros libres daría más soluciones
```

```

% (da 2 en este caso porque los parametros libres son dos).

% Obviamente el algoritmo, null(), ya sabe que resuelve un SL homogeneo,
% por tanto el unico argumento que acepta es la matriz de coeficientes A.
% En efecto si añadieramos, por error, la columna de ceros de los terminos
% independientes, null() consideraria esa columna como una incognita mas.

sol=null(A)

    -0.7107    0.3857
    -0.1867    0.6532
     0.4307    0.5941
     0.5240    0.2675
    -0.0000   -0.0000

u1=sol(:,1)
u2=sol(:,2)

u1 =
    -0.7107
    -0.1867
     0.4307
     0.5240
    -0.0000

u2 =
     0.3857
     0.6532
     0.5941
     0.2675
    -0.0000

% La sol. general es sol=alpha*u1+beta*u2, con alpha y beta param. libres.
% El lugar geometrico de las soluciones es el mismo que antes,
% simplemente se estan utilizando dos vectores diferentes de los anteriores
% para generarlo.
% Los vectores que nos da null son unitarios (norma 1) y ortogonales entre si.

% Los vectores v1 y v2 que calculamos con rref como si efectuaramos
% los calculos a mano se pueden obtener tambien como null,
% simplemente haciendolo trabajar en modo simbolico, usando en vez de A, sym(A).

sol= null(sym(A))

sol=
[2, -3]
[2, -2]
[1,  0]
[0,  1]
[0,  0]

```

```

% La solucion deducida asi es
x=alpha*(2,2,1,0,0)+beta*(-3,2,0,1,0),
% que es la misma que la obtenida con rref, y que nos evita ahora
% tener que despejar.

%%%%%%%%%%%%%% FIN DEL EJERCICIO %%%%%%%%%

¿Por qué syms lo acompañamos con real?

syms x real % syms crea la variable simbolica x
            % y en el uso de x se entendera que es un numero real.

2*x-5*x
conj(x)     % Probamos a calcular el conjugado de x y nos
            % contesta que conj(x) es x
            % porque el conjugado de un real es el mismo real
            % (no hay parte imaginaria que tenga que cambiar de signo).

-3*x
x

syms x      % syms crea la variable simbolica x sin especificar que tipo
            % de numero (las opciones son real, integer, positive,
            % rational pero no usamos ninguna)

2*x-5*x
conj(x)     % Nos va a decir que conj(x) es conj(x). No puede calcular
            % el conjugado de x porque x puede ser complejo y por el dato
            % no se sabe cual es la parte real y cual
            % la parte imaginaria de x

-3*x
conj(x)

% Esta diferencia entre variables simbolicas reales y complejas
% es importante para nosotros al usar la trasposición con el símbolo '
% Ya que ' significa para MATLAB traspuesta-conjugada.

% Veamoslo en el siguiente ejemplo:

syms x y
[x y]'

conj(x)
conj(y)

syms x y real
[x y]'

x
y

```

Observa que las funciones syms y sym() son distintas.

P.2 Semana 21 de febrero

- Ejercicios de clasificación y resolución de sistemas de ecuaciones lineales.
- Cuestionario Moodle.
- Si el último dígito de tu DNI/NIE es cero o par, resuelve el cuestionario modelo 2. Si es impar resuelve el cuestionario modelo 1.

Se proporciona la siguiente Guía para la resolución de este cuestionario concreto:

- `format rat`
- definir la matriz de coeficientes $A = [\dots ; \dots ; \dots]$
- definir el vector b como $b = [\dots]'$
- definir matriz ampliada como $A_m = [A \ b]$
- resolver los SLs mediante `rref(Am)`
- no usar `null()` aunque alguno sea homogéneo
- si el sistema es indeterminado no usar `syms` para declarar parámetros libres, sino simplemente despejar desde la escalonada reducida y usar el nombre del parámetro libre que se quiera
- obtener los rangos contando las columnas pivotaes visualmente o mediante la función `rank()`
- se proporciona dentro del cuestionario Moodle el valor de `sum(sum(Am))` para cada SL, de forma que podáis comprobar que habéis introducido bien los datos. Se avisa que si no coincide ha habido error en la entrada de datos, y que si coincide tenéis que revisar de todas formas.

Cuestionario

1. Considera el sistema lineal de ecuaciones:

$$\begin{cases} x_1 + 2x_2 + 3x_3 = 1 \\ x_1 - x_2 + x_3 = 0 \\ x_1 + 3x_2 - x_3 = -1 \\ 3x_1 + 4x_2 + 3x_3 = 0 \end{cases}$$

Se pide:

- (a) (0.50 pts) La forma escalonada reducida de la matriz ampliada.
- (b) (0.50 pts) El rango de la matriz de coeficientes y el rango de la matriz ampliada.
- (c) (1.00 pts)
 - La clasificación del sistema como compatible determinado, compatible indeterminado o incompatible.
 - La solución en forma vectorial, si existe.

$\text{sum}(\text{sum}(\text{Am}))$ tiene que ser 20

2. Considera el sistema lineal de ecuaciones:

$$\begin{cases} x_1 + x_2 + x_3 + 2x_4 = 0 \\ x_2 - x_3 = 5 \\ x_1 + 2x_3 + 2x_4 = 1 \\ x_1 + 2x_2 = 0 \end{cases}$$

Se pide:

- (a) (0.50 pts) La forma escalonada reducida de la matriz ampliada.
- (b) (0.50 pts) El rango de la matriz de coeficientes y el rango de la matriz ampliada.
- (c) (1.00 pts)
 - La clasificación del sistema como compatible determinado, compatible indeterminado o incompatible.
 - La solución en forma vectorial, si existe.

$\text{sum}(\text{sum}(\text{Am}))$ tiene que ser 19

3. Considera el sistema lineal de ecuaciones:

$$\begin{cases} x_1 - 3x_2 + 4x_3 = 0 \\ -x_1 - 2x_2 + 4x_3 = 0 \\ -2x_1 + x_2 = 0 \end{cases}$$

Se pide:

- (a) (0.50 pts) La forma escalonada reducida de la matriz ampliada.
- (b) (0.50 pts) El rango de la matriz de coeficientes y el rango de la matriz ampliada.
- (c) (1.00 pts)
 - La clasificación del sistema como compatible determinado, compatible indeterminado o incompatible.
 - La solución en forma vectorial, si existe.

$\text{sum}(\text{sum}(\text{Am}))$ tiene que ser 2

SOLUCIÓN

% Ejercicio 1

```
format rat , Am=[1 2 3 1; 1 -1 1 0; 1 3 -1 -1; 3 4 3 0]
```

```
%      1      2      3      1
```

```
%      1     -1      1      0
```

```
%      1      3     -1     -1
```

```
%      3      4      3      0
```

```
sum(sum(Am)) % 20
```

```
rref(Am)
```

```
%      1      0      0     -1/2
```

```
%      0      1      0      0
```

```
%      0      0      1      1/2
```

```
%      0      0      0      0
```

% La forma escal. reducida de la matriz ampliada es la matriz anterior

% rango A=3 rango Ampliada=3

% sistema compatible determinado sol (-1/2,0,1/2)

% Ejercicio 2

```
Am=[1 1 1 2 0; 0 1 -1 0 5; 1 0 2 2 1; 1 2 0 0 0]
```

```
%      1      1      1      2      0
```

```
%      0      1     -1      0      5
```

```
%      1      0      2      2      1
```

```
%      1      2      0      0      0
```

```
sum(sum(Am)) % 19
```

```
rref(Am)
```

```
%      1      0      2      0      0
```

```
%      0      1     -1      0      0
```

```
%      0      0      0      1      0
```

```
%      0      0      0      0      1
```

% rango A=3 rango Ampliada=4

% sistema incompatible no tiene solucion

% Ejercicio 3

```
Am=[1 -3 4 0; -1 -2 4 0; -2 1 0 0]
```

```
%      1     -3      4      0
```

```
%     -1     -2      4      0
```

```
%     -2      1      0      0
```

```
sum(sum(Am)) % 2
```

```
rref(Am)
```

```
%      1      0     -4/5      0
```

```
%      0      1     -8/5      0
```

```
%      0      0      0      0
```

% rango A=2 rango Ampliada=2

% sistema compatible indeterminado

% solucion $x=(4/5, 8/5, 1)*x_3 / x_3$ pertenece a R

```

% Otra version:
Am=[1 2 3 3; 1 -1 1 0; 1 3 -1 -3; 3 4 3 0]
sum(sum(Am))    % 20
rref(Am)

%      1      2      3      3
%      1     -1      1      0
%      1      3     -1     -3
%      3      4      3      0

%      1      0      0     -3/2
%      0      1      0      0
%      0      0      1      3/2
%      0      0      0      0

% rango A=3  rango Ampliada=3
% sistema compatible determinado  sol  (-3/2,0,3/2)

Am=[1 1 1 2 0; 0 1 -1 0 5; 1 0 2 2 1; 2 2 0 4 0]
sum(sum(Am))    % 24
rref(Am)

%      1      1      1      2      0
%      0      1     -1      0      5
%      1      0      2      2      1
%      2      2      0      4      0

%      1      0      0      2      0
%      0      1      0      0      0
%      0      0      1      0      0
%      0      0      0      0      1

% rango A=3  rango Ampliada=4
% sistema incompatible no tiene solucion

Am=[1 -3 5 0; -1 -2 5 0; -2 1 0 0]
sum(sum(Am))    % 4
rref(Am)

%      1     -3      5      0
%     -1     -2      5      0
%     -2      1      0      0

%      1      0     -1      0
%      0      1     -2      0
%      0      0      0      0

% rango A=2  rango Ampliada=2
% sistema compatible indeterminado
% solucion x=(1, 2, 1)*x3 / x3 pertenece a R

```

P.3 Semana 28 de febrero

Ejercicios de introducción a las aplicaciones lineales

- **Ejercicio 1.4.3** Considerada la aplicación lineal f de \mathbb{R}^2 en \mathbb{R}^2 con matriz estándar asociada $A = \begin{bmatrix} 1 & -1 \\ 2 & 3 \end{bmatrix}$ determina:

- a) La imagen de $\vec{x} = (1, 5)$.
- b) El antecedente de $\vec{y} = (-1, 18)$, si existe.

Ver enunciado también en página 27 de las diapositivas del Tema 1.

- Matriz estándar del giro en \mathbb{R}^2 de centro en el origen y ángulo θ .

Ver en página 32 de las diapositivas del Tema 1, **Ejercicio 1.4.12**.

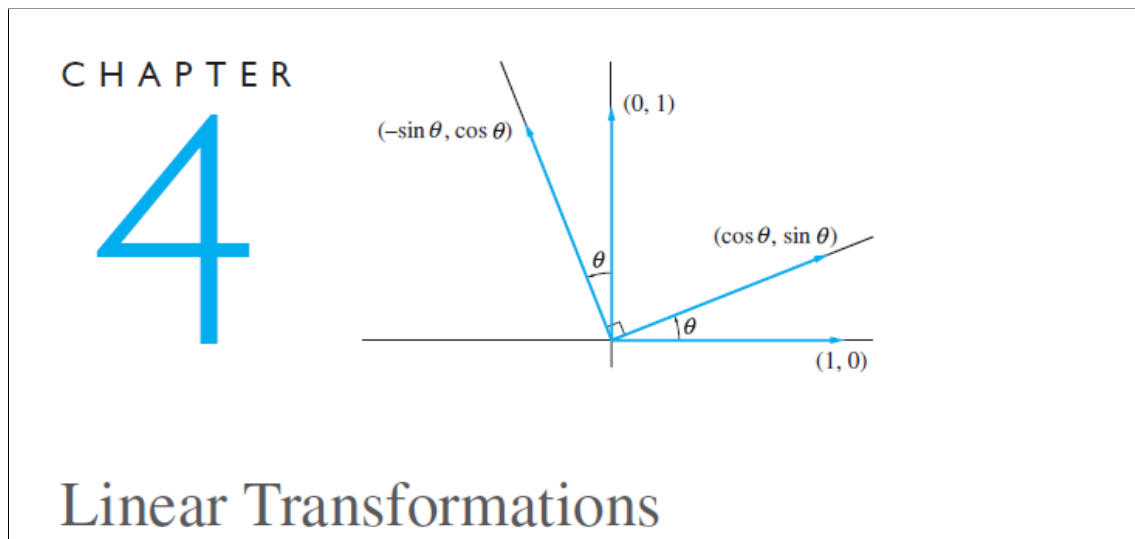


Figure 1: S.J. Leon. Linear Algebra with Applications. Edición 9^a. Pearson. 2015.

La matriz estándar del endomorfismo, o matriz respecto de la base canónica, es:

$$A = \begin{bmatrix} \cos\theta & -\text{sen}\theta \\ \text{sen}\theta & \cos\theta \end{bmatrix}$$

En efecto, siendo $\vec{e}_1 = (1, 0)$ y $\vec{e}_2 = (0, 1)$,

$$A = [f(\vec{e}_1) \ f(\vec{e}_2)] \quad y \quad \begin{cases} f(\vec{e}_1) = \cos\theta \vec{e}_1 + \text{sen}\theta \vec{e}_2 \\ f(\vec{e}_2) = -\text{sen}\theta \vec{e}_1 + \cos\theta \vec{e}_2 \end{cases}$$

$0 \leq \theta \leq \pi$ en el sentido antihorario o positivo

$-\pi < \theta < 0$ en el sentido horario o negativo

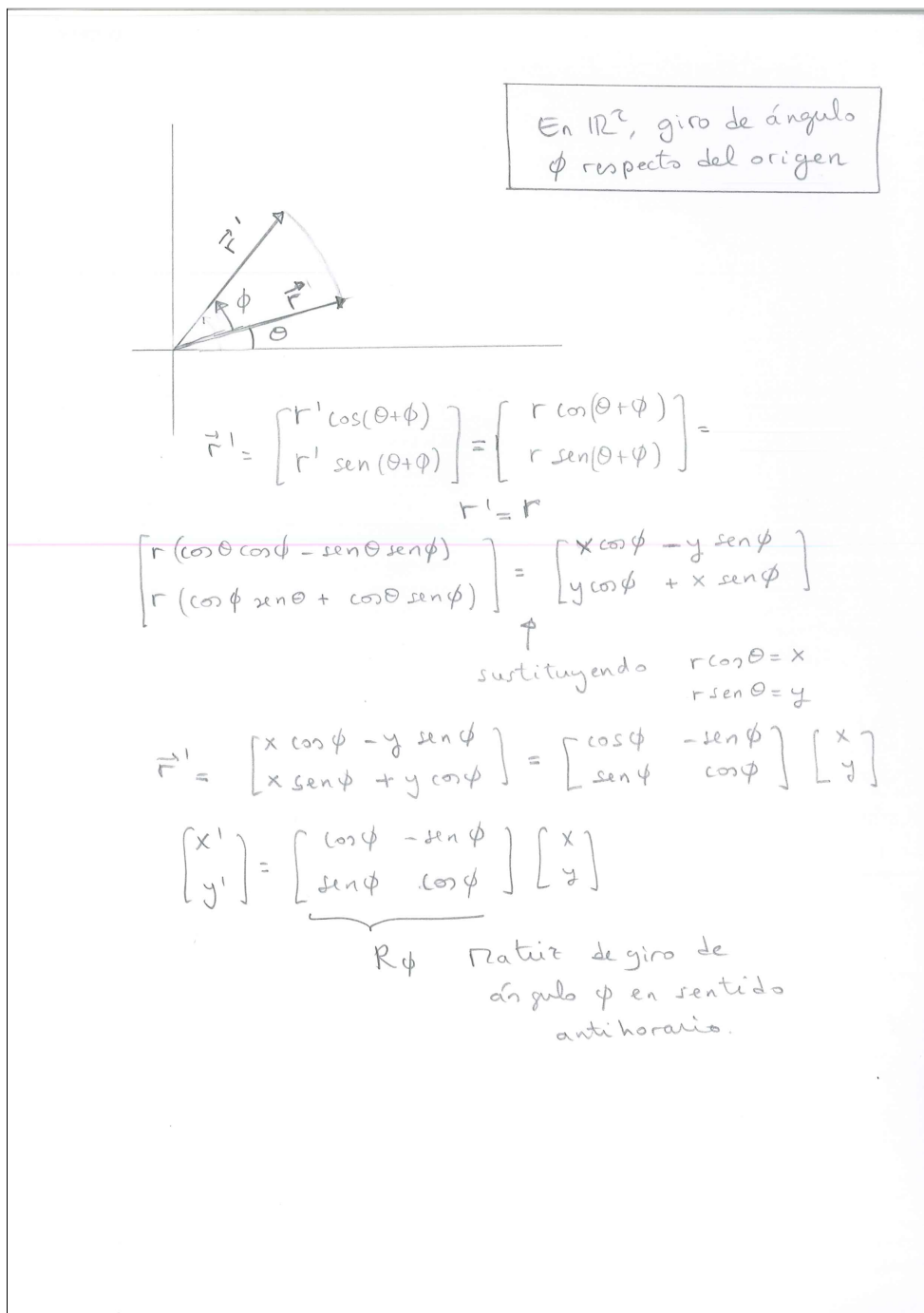


Figure 2: Otra deducción de la matriz de giro.

Ejercicio Adicional 1.4.13

En \mathbb{R}^2 , escribe la matriz estándar correspondiente al giro de 50 grados en sentido antihorario de centro en el origen.	$A =$
Obtén el transformado del vector $\vec{v} = (4, 3)$	$\vec{v}_{\text{girado1}} =$
Comprueba que las normas de los dos vectores son iguales, utilizando la función <code>norm()</code>	$\ \vec{v}\ =$ $\ \vec{v}_{\text{girado1}}\ =$
Obtén el transformado del vector $\vec{v} = (4, 3)$ para un giro de 50 grados en sentido horario.	$\vec{v}_{\text{girado2}} =$
Obtén el transformado del vector $\vec{v} = (4, 3)$ para un giro de 180 grados en sentido horario	$\vec{v}_{\text{girado3}} =$

Ejercicio Adicional 1.4.14 Escribe la matriz estándar A de la aplicación lineal de \mathbb{R}^5 en \mathbb{R}^3 que asigne a $(x_1, x_2, x_3, x_4, x_5)$ un vector (y_1, y_2, y_3) tal que:

$$\begin{cases} y_1 & \text{es la media de } x_1, x_2, x_3 \\ y_2 & \text{es la media de } x_1, x_2, x_3, x_4, x_5 \\ y_3 & \text{es la suma de } x_1, x_2, x_3, x_4, x_5 \end{cases}$$

Aplicación de esta matriz para el estudio de casos PCR+ en Cantabria, en los períodos del 1 al 5 y del 8 al 12 de marzo de 2021.

Fecha	Nuevos PCR+
1-3-2021	40
2-3-2021	48
3-3-2021	37
4-3-2021	43
5-3-2021	26
8-3-2021	27
9-3-2021	48
10-3-2021	44
11-3-2021	39
12-3-2021	45

Primera semana de marzo	media LMX	
	media LMXJV	
	casos totales LMXJV	
Segunda semana de marzo	media LMX	
	media LMXJV	
	casos totales LMXJV	

```

%% SOLUCION DEL EJERCICIO 1.4.3:

>> A=[1 -1; 2 3] % Matriz de la aplicacion lineal
    1  -1
    2   3

>> v=[1 5]' % Vector cuya imagen queremos determinar
    1
    5

>> fv= A*v % A multiplicado por v nos da la imagen de v
   -4
   17

>> % La siguiente pregunta es el antecedente de  $y=(-1,18)$ , si existe
>> % Estudiamos en primer lugar la compatibilidad del sistema lineal  $[A | y]$ 
>> % Primero rango de A

>> rank(A)
    2

>> % Por ser rango de la matriz de coeficientes 2, y tener A dos filas,
>> % el SL  $[A | y]$  es compatible cualquiera que sea el vector y
>> % Dos formas de resolverlo:
>> % I) Con la forma escalonada reducida:

>> y = [-1 18]'
   -1
   18

>> Am = [A y] % es la matriz ampliada
    1  -1  -1
    2   3  18

>> rref(Am) % leemos la solucion a la derecha
    1   0   3
    0   1   4

>> % La solucion es  $x=(3,4)$ 
>> % Comprobacion:

>> x=[3 4]'
    3
    4

>> A*x % aqui tenemos que obtener (-1,18)
   -1
   18

>> % Otra forma de comprobar es que  $A*x - y$  nos de (0,0)
>> A*x - y
    0
    0

>> % II) Introducimos una funcion para resolver sist. compat. determinados
>> % que es linsolve(A,b) siendo A la matriz de coeficientes y
>> % b el vector de terminos independientes

```

```

>> % Este sistema es compatible determinado para todo b porque tiene dos
>> % ecs. y dos incognitas y el rango de la matriz de coefs es 2
>> % Esta garantizado por tanto que podemos usar linsolve()
>> linsolve(A,y)
3
4
>> % Tambien podemos almacenar el resultado con el nombre sol, por ejemplo.
>> sol=linsolve(A,y)
3
4
>> % la comprobacion sera que A*sol-y quede (0,0)
>> A*sol-y
0
0

```

%%%%%%%% SOLUCION DEL EJERCICIO ADICIONAL 1.4.13

```

>> a=50
50

>> A= [cosd(a)  -sind(a) ;  sind(a)  cosd(a)]
0.6428  -0.7660
0.7660   0.6428

>> v=[4 3]'
4
3

>>vg1=A*v
0.2730
4.9925

>> norm(v)
5
>> norm(vg1)
5.0000

>> A = [cosd(-a)  -sind(-a) ;  sind(-a)  cosd(-a)]
0.6428   0.7660
-0.7660  0.6428

vg2 = A*v
4.8693
-1.1358

>> norm(vg2)
5

>> a = 180
180

```

```

>> A= [cosd(a)  -sind(a) ;  sind(a)  cosd(a)]
      -1      0
      0      -1

>> vsimetrico_respecto_origen = A*v
-4
 3
>> % Con el giro de 180 grados se obtiene el vector opuesto.
>> % Esta transformacion se conoce con dos nombres: giro de 180
>> % grados y simetría respecto del origen.

%%%%% SOLUCION DEL EJERCICIO ADICIONAL 1.4.14

>> % Construimos la matriz de forma que:
>> % y1 sea la media de los tres primeros dias, de ahi 1/3
>> % y2 es la media de los 5 dias, de ahi 1/5
>> % y3 es la suma de casos de los 5 dias

>> A=[1/3 1/3 1/3 0 0; 1/5 1/5 1/5 1/5 1/5 ; 1 1 1 1 1]
      0.3333    0.3333    0.3333         0         0
      0.2000    0.2000    0.2000    0.2000    0.2000
      1.0000    1.0000    1.0000    1.0000    1.0000

>> % metodo para resolver la semana 1

>> PCRplus_s1 = [40 48 37 43 26]'
      40
      48
      37
      43
      26

>> A * PCRplus_s1
      41.6667
      38.8000
      194.0000

>> % metodo para resolver las dos semanas a la vez
>> % primera columna es la primera semana
>> % segunda columna es la segunda semana

>> PCRplus = [40 48 37 43 26 ; 27 48 44 39 45]'
      40      27
      48      48
      37      44
      43      39
      26      45

>> A * PCRplus
      41.6667    39.6667
      38.8000    40.6000

```

194.0000 203.0000

%%%%%%%%%% FIN %%%%%%%%%%

P.4 Semana 7 de marzo

Operaciones con matrices, inversa y determinantes

OBSERVACIÓN PARA RESOLUCIÓN CON MATLAB: Si trabajas con matrices que incluyen variables simbólicas puedes utilizar la función **det()**, pero no de las funciones **rank()**, ni **rref()**, ni **inv()**, ya que estas tres en general darán resultados erróneos. La función **inv()** calcula la inversa de una matriz. Antes de calcular la inversa de una matriz debes asegurarte de que su determinante es distinto de cero, y de que su valor tampoco es compatible con cero dentro de la precisión numérica de MATLAB.

- **Ejercicio 2.9. Modificado.** Considera el SL siguiente:
$$\begin{cases} x_1 + x_2 - x_3 = 1 \\ x_1 + a x_2 + 3x_3 = 2 \\ 2 x_1 + 3 x_2 + a x_3 = 3 \end{cases}$$

1. Determina los valores de a para los que el sistema lineal es compatible determinado.
2. Para el resto de valores de a , clasifica el SL resultante y obtén la solución si existe.

- **Ejercicio 2.10. Modificado.** Considera el SL siguiente:
$$\begin{cases} -x_1 - 2x_2 + ax_3 = a \\ 2x_1 - ax_2 + 2x_3 = -2 \\ x_1 + 2x_2 + 4x_3 = 3 \end{cases}$$

1. Determina los valores de a para los que el sistema lineal es compatible determinado.
2. Para el resto de valores de a , clasifica el SL resultante y obtén la solución si existe.

- **Ejercicio 2.11.** Dadas las matrices $A = \begin{bmatrix} 2 & -3 & -5 \\ -1 & 4 & 1 \\ 1 & -3 & -4 \end{bmatrix}$ y $B = \begin{bmatrix} 2 & 1 & -4 \\ 0 & 1 & 4 \\ 1 & -2 & -3 \end{bmatrix}$, y sabiendo que

$D = ABC = \begin{bmatrix} 11 & 17 \\ 17 & 1 \\ 4 & 10 \end{bmatrix}$, calcula la matriz C . Si no existe, indícalo explícitamente.

- **Ejercicio 2.17** Haciendo uso del concepto de determinante, calcula el valor de m para que el siguiente sistema tenga soluciones no triviales:
$$\begin{cases} 2x - y + z = 0 \\ x + my - z = 0 \\ x + y + z = 0 \end{cases}$$

AÑADIDO: Determina la solución para el caso en que existan soluciones distintas de la trivial.

SOLUCIONES

- EJERCICIO 2.9 MODIFICADO

```
syms a real
A = [ 1 1 -1 ; 1 a 3 ; 2 3 a ]
% [1, 1, -1]
% [1, a, 3]
% [2, 3, a]

% 1)
det(A) % el SL, con tres ecuaciones y tres incognitas, es comp.
        % determinado si y solo si rgA=3
        % si y solo si det(A) distinto de cero.

        % es obvio que si rg(A)=3, entonces rg(A*) tambien es 3,
        % ya que hay tres filas.

% a^2 + a - 6

% vemos ahora dos formas de obtener las raices

factor(ans) % factoriza a^2+a-6 = (a+3)*(a-2)
% [a + 3, a - 2]

solve(det(A)) % forma mas directa
% -3
% 2

% Conclusion: el SL es comp. determ.
% si y solo si a distinto de 2 y distinto de -3.
% En este caso rg(A)=3=rg(A*)=numero de incognitas.

% 2) Analizamos el caso a=-3

Amenos3 = subs(A,a,-3)
% [1, 1, -1]
% [1, -3, 3]
% [2, 3, -3]

Amenos3_ampli=[Amenos3 [1 2 3]'] % concatenamos b a la derecha

% [1, 1, -1, 1]
% [1, -3, 3, 2]
% [2, 3, -3, 3]

[rank(Amenos3) rank(Amenos3_ampli)]
% 2 3

% para a=-3 el SL es incompatible, porque rg(A) distinto de rg(A*)

% Tambien podriamos verlo en la forma escalonada reducida de Amenos3_ampli
```

```

% caso a=2
Amas2 = subs(A,a,2)
Amas2_ampli = [Amas2 [1 2 3]']

[rank(Amas2) rank(Amas2_ampli)]
%      2      2

% para a=2 el SL es compatible indeterminado;
rg(A) = rg(A*) = 2 < numero de incognitas

rref(Amas2_ampli) % Para despejar la solucion obtenemos reducida
% [1, 0, -5, 0]
% [0, 1, 4, 1]
% [0, 0, 0, 0]

% despejamos:
% x=5z, y=-4z+1, z=z          ecs parametricas
% sol=(0,1,0) + (5,-4,1)*z    solucion en forma vectorial parametrica

```

• EJERCICIO 2.10. MODIFICADO

```

syms a real
A=[-1 -2 a; 2 -a 2; 1 2 4]
% [-1, -2, a]
% [ 2, -a, 2]
% [ 1,  2, 4]

det(A)
% a^2 + 8*a + 16

solve(ans) % en vez de hacer solve(det(A))
% hacemos solve(ans) ans=ultima expresion calculada
% - 4
% - 4

% El SL es comp. determ. si y solo si a distinto de -4.

Amenos4 = subs(A,a,-4) % analizamos el caso restante a = -4
% [-1, -2, -4]
% [ 2,  4,  2]
% [ 1,  2,  4]

% al concatenar b=(a, -2, 3) tenemos que
% escribir (-4, -2, 3), pues estamos considerando a=-4

Ampli = [ Amenos4 [-4 -2 3]']
% [-1, -2, -4, -4]
% [ 2,  4,  2, -2]
% [ 1,  2,  4,  3]

[rank(Amenos4) rank(Ampli)] % sistema incompatible

```

```
%      2      3
```

• EJERCICIO 2.11

```
A=[2 -3 -5; -1 4 1; 1 -3 -4]
%      2      -3      -5
%     -1      4      1
%      1      -3      -4
```

```
B=[2 1 -4; 0 1 4; 1 -2 -3]
%      2      1      -4
%      0      1      4
%      1      -2      -3
```

```
D=[ 11 17; 17 1; 4 10]
%      11      17
%      17      1
%      4      10
```

```
% El procedimiento mas sencillo es el siguiente:
% Partiendo de  $A*B*C=D$  analizamos si  $A*B$  es invertible
```

```
det(A*B)
%-216
```

```
% obtenemos determinante distinto de cero, por tanto si lo es
```

```
%  $\text{inv}(A*B) * A*B * C = \text{inv}(A*B)*D$  es como queda la ec. premultiplicando
% los dos miembros por esa inversa
%  $\Rightarrow C = \text{inv}(A*B)*D$ 
```

```
C=inv(A*B)*D
%      2.0000      1.0000
%      2.0000      2.0000
%      1.0000           0
```

```
% La comprobacion seria similar a la anterior
 $A*B*C-D$  % esl resultado es una matriz de ceros del tamaño de D
```

• EJERCICIO 2.17. MODIFICADO

```
syms m real
A=[2 -1 1; 1 m -1; 1 1 1]
% el sistema tiene soluciones distintas de la trivial (ademas de la
% trivial, (0,0,0)) si y solo si el rango de A es menor que 3,
% o lo que es lo mismo, si el determinante de A es 0.
```

```
det(A)
% m+5
```

```
% por tanto tiene soluciones distintas de las trivial para  $m=-5$ 
```

```

Asol=subs(A,m,-5) % La matriz de coeficientes para este caso

null(sym(Asol)) % funcion de MATLAB para obtener
                % la solucion de un SLHomogeneo

% -2/3
% -1/3
% 1

% La solución para este valor de m es sol=(-2/3,-1/3,1) * alpha, con alpha real

% Para m distinto de -5 la unica solucion es la trivial, sol=(0,0,0).

```

P.5 Semana 14 de marzo

Tema 2. Determinantes

Sección 2.8. Rango de una matriz como el orden del mayor menor no nulo.

Ejercicio 2.7 Determina el rango de $A = \begin{bmatrix} 2 & 2 & 3 & 3 \\ 1 & 7 & 2 & 4 \\ 0 & 0 & a & a(a-1) \end{bmatrix}$ en función del parámetro a , basándote en sus menores.

Ejercicio 2.19 a) Obtén el rango de $A = \begin{bmatrix} 2 & -1 & a \\ 1 & 10 & -6 \\ 1 & a & -1 \end{bmatrix}$ en función del parámetro a .

b) Obtén el rango de $B = \begin{bmatrix} 2 & -1 & a & 5 \\ 1 & 10 & -6 & 1 \\ 1 & a & -1 & 2 \end{bmatrix}$ en función del parámetro a . Para ello utiliza el método de orlar menores.

SOLUCION:

```

% EJERCICIO 2.7
% syms a real
A=[2 2 3 3; 1 7 2 4; 0 0 a a*(a-1)]
rank(A) % Al existir una variable en A, rank() no se puede utilizar
        % y en efecto da un resultado falso
rref(A) % Lo mismo sucede con rref()

% Podemos calcular el rango como el orden del mayor subdeterminante(menor)
% que no sea cero.

% Resultados y procedimientos
% 1) el subdeterminante formado por filas 1 y 2 y cols 1 y 2 es distinto
% de cero, por tanto el rango es como minimo 2.

% 2) toca explorar los subdeterminantes de orden 3.
%
% 3) aunque la matriz tiene 4 subdeterminantes de orden 3 (quitando col1

```

```

%   o col2 o col3 o col4), solo tenemos que comprobar los que incluyen
%   el subdeterminante no cero identificado en el paso 1)

m2=det(A(1:2,1:2))

m3_1=det(A(1:3,1:3))      % resultado: 12*a
m3_2=det(A(1:3,[1 2 4])) % resultado: 12*a*(a-1)

% 4) concluimos que rango=2 si y solo si a=0. Porque para este valor son
%   cero los dos menores. Para a=1 el rango es 3, porque aunque se anula
%   uno de los menores, no se anula el otro.

% EJERCICIO 2.19
A = [ 2 -1 a ; 1 10 -6 ; 1 a -1]
B = [ 2 -1 a 5 ; 1 10 -6 1 ; 1 a -1 2]

% [2, -1, a]
% [1, 10, -6]
% [1, a, -1]

% [2, -1, a, 5]
% [1, 10, -6, 1]
% [1, a, -1, 2]

m2=det(A(1:2,1:2)) % 21, A simple vista vemos que el rango(A) >= 2

solve(det(A)) % -5 y 3

% Para a=-5 y para a=3 el unico menor de orden 3 se anula, por tanto el rango
% de A no "crece", y es por tanto 2.

% Para a distinto de -5 y distinto de 3 el menor de orden 3 no es cero,
% por tanto el rango es 3.
% Puedes comprobar que por ejemplo para a=0 el rango es 3.

det(subs(A,a,0)) % -15 distinto de cero y por tanto rango 3.

% también puedes comprobar que para a=-5 o a=3 el determinante de orden 3
% es cero.

[det(subs(A,a,-5)) det(subs(A,a,3))] % puestos juntos para tener el resultado
% en una sola linea

[0 0]

% La matriz B difiere de A en que tiene una columna más.
% el rango de B es tres cuando a distinto de -5 y cuando a distinto de 3
% pues ya tenemos un menor distinto de cero (el formado por las tres
% primeras columnas de B).

% Solo tenemos que estudiar el otro menor de orden 3 que rodea a m2
% que es el formado por las columnas 1,2 y 4.

```

```

M3_2=B(:,[1 2 4]); m3_2=det(M3_2)    %3*a-9
solve(m3_2)    % 3
% si a es distinto de 3 este menor no se anula, por tanto rango 3.

% rango(B)= 2 si y solo si a=3 (para a=-5 acabamos de encontrar que el segundo
%                               menor no es 0).

% Conclusion:

% rg(A) = 3 para a distinto de -5 y distinto de 3. rg(A)=2 para a=-5 o a=3
% rg(B) = 3 para a distinto de 3 . rg(B)=2 para a=3

% En una tabla se expresaria asi:

% |-----|-----|
% | valores de a                | rg A |
% |-----|-----|
% | a distinto de -5 y distinto de 3 | 3 |
% | a=-5                          | 2 |
% | a=3                             | 2 |
% |-----|-----|

% |-----|-----|
% | valores de a                | rg B |
% |-----|-----|
% | a distinto de 3             | 3 |
% | a=3                          | 2 |
% |-----|-----|

```

P.6 Semana 21 de marzo

Espacios vectoriales: cambio de base

Ejercicio 3.9 . Sean las bases $B = \left\{ \begin{bmatrix} 5 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \end{bmatrix} \right\}$ y $B' = \left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \end{bmatrix} \right\}$

a) Obtén la matriz de cambio de base o coordenadas de la base B a la base B' .

b) Sabiendo que las coordenadas de un vector \vec{v} respecto de la base B son $[\vec{v}]_B = \begin{bmatrix} 4 \\ 3 \end{bmatrix}$, obtén sus coordenadas respecto de la base B' .

c) ¿Cuales son las coordenadas estándar del vector \vec{v} ?

```

% a) Obtencion de la matriz de cambio de base de B a Bprima
% Escribimos la matriz de cada base de forma que los vectores
% columna sean los vectores base en su orden

```

```

PB = [ 5 1; 2 3] '
PBp = [ 1 1; -1 1] '

```

```

5     2
1     3

1     -1
1     1

% La matriz P de cambio de base de B a Bprima tiene por columnas
% las coords. de los vectores de la base B respecto de la base Bprima.
% Primera columna = coordenadas de [5 1]' respecto de la base B'
% Segunda columna = coordenadas de [2 3]' respecto de la base B'.
% Obtenemos esas coordenadas resolviendo el SL con matriz ampliada
% [PBp | PB]

% Presentamos dos metodos.

% Con la inversa de PBp:

P = PBp^-1*PB
3 5/2
-2 1/2

% Resolviendo el SL [PBp | PB] mediante reducción (eliminacion de Gauss-Jordan),
% utilizando rref()

[PBp PB]
1 -1 5 1
1 1 1 3

reducida = rref([PBp PB])
1 0 3 5/2
0 1 -2 1/2

P = reducida( : , 3:4 )
3 5/2
-2 1/2

```

La matriz de cambio de base de la base B a la base B' es:

$$P = \begin{bmatrix} 3 & 5/2 \\ -2 & 1/2 \end{bmatrix}$$

```

% b) El dato son las coords. de v respecto de la base B: vB=[4 3]'
% Se piden las coordenadas de v en la base B'.
% P es la matriz que convierte coordenadas relativas a la base B
% en coordenadas relativas a la base B'. Por tanto vB'=P*vB

```

```
vB = [4 3]'
```

```
4
3
```

```
vBprima = P*vB
```

```
39/2
-13/2
```

Las coordenadas de \vec{v} respecto de la base B' son:

$$[\vec{v}]_{B'} = \begin{bmatrix} 39/2 \\ -13/2 \end{bmatrix}$$

```
% c) Se piden las coordenadas estandar de v
% La matriz PB transforma de coordenadas relativas a base B a
% coordenadas estandar.
```

```
v = PB*vB
```

```
ans =
    26
    13
```

Las coordenadas de \vec{v} respecto de la base canónica son:

$$\vec{v} = \begin{bmatrix} 26 \\ 13 \end{bmatrix}$$

Base y dimensión de $\text{Nul}(A)$, o lo que es lo mismo, base y dimensión a partir de las ecuaciones implícitas del subespacio

3.12 Determina una base y la dimensión del subespacio $\text{Nul}(A)$, para $A = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix}$.

3.13 Determina una base y la dimensión del subespacio $\text{Nul}(A)$, siendo $A = [1 \ 1 \ 1]$.
El enunciado también podía haber sido: “Determina una base y la dimensión del subespacio de \mathbb{R}^3 de ecuación implícita $x + y + z = 0$.”

Soluciones de los ejercicios 3.12 y 3.13.

3.12

```
A=[1 1 1 0; 2 1 0 1]
B=null(sym(A))
```

```
% A =
%      1      1      1      0
%      2      1      0      1
```

```
% B =
% [ 1, -1]
% [-2,  1]
% [ 1,  0]
% [ 0,  1]
```

```
% El resultado es una matriz y las columnas
% son la base del subespacio nulo de A, simbolizado como Nul(A), N(A) o Ker(A)
% y también denominado nucleo de A
```

```
% Base = { (1,-2,1,0), (-1,1,0,1) }
size(B,2) % el numero de columnas es la dimension del subespacio N(A)
```

```
% 2
```

```

% La dimension es 2

% A es la matriz de coeficientes de las ecs. implicitas del subespacio
% Los vectores base cumpliran dichas ecuaciones, lo que podemos comprobar
% obteniendo A*B, que debe resultar
% en una matriz cuyas columnas son, cada una, el vector cero.

A*B
% [0, 0]
% [0, 0]

```

3.13

```

A=[1 1 1]
B=null(sym(A))
% Ejercicio similar al anterior. El resultado es una matriz
% y las columnas son la base del subespacio nulo de A, N(A)

% A =
%      1      1      1

% B =
% [-1, -1]
% [ 1,  0]
% [ 0,  1]

% La base del subespacio es B = { (-1,1,0), (-1,0,1) }
size(B,2) % el numero de columnas es la dimension del subespacio

% 2

% La dimension es 2

```

Conversión de base del subespacio a ecuaciones del subespacio

Explicado en el aula 24-3-2022

3.16 Obtén la/s ec./ecs. implícita/s de $H = \langle (1, 1, 0, 0), (1, 2, -1, 1) \rangle$ utilizando los dos métodos que conoces.

3.16

```

PB=[1 1 0 0; 1 2 -1 1]'
% Sabemos que este conjunto generador de H es base
% porque los vectores no son uno multiplo de otro

% o por rank(PB) % 2

% Esperamos dos ecuaciones implicitas,
% al tratarse de un subespacio de dim=2 en R4.

```

```

% Metodo 1:

syms x1 x2 x3 x4 real
x=[x1 x2 x3 x4]'
[PB x] % x pertenece a H <=> el rango de esta matriz es 2

% aqui vemos la matriz:

% 1 1 x1
% 1 2 x2
% 0 -1 x3
% 0 1 x4

% Vemos que el menor de las filas 1,2 y cols 1,2 es distinto
% de cero. El rango de [PB x] es 2 si y solo si los dos
% menores que orlan ese menor son cero

A=[PB x] % le damos nombre

det(A(1:2,1:2)) % menor de orden 2 no cero

% ans= 1

det(A(1:3,:)) % la expresion resultante IGUALADA A CERO es la primera ecuacion.

% ans = x2 - x1 + x3

det(A([1 2 4],:)) % la expresion resultante igualada a cero es la segunda ecuacion

% ans = x1 - x2 + x4

% Las ecs. son por tanto  $x_2 - x_1 + x_3 = 0$  ,  $x_1 - x_2 + x_4 = 0$ 

% Metodo 2

A=(null(sym(PB')))'

% A =
% [-1, 1, 1, 0]
% [ 1, -1, 0, 1]

% Las ecs son  $-x_1 + x_2 + x_3 = 0$  ,  $x_1 - x_2 + x_4 = 0$ 
% Se obtiene el mismo resultado.

% Observacion: Si el resultado hubiera salido distinto, la respuesta estaría bien
% de todos modos, siempre que los dos sistemas lineales obtenidos sean equivalentes

```

P.7 Semana 28 de marzo

Ejemplo de suma de subespacios de \mathbb{R}^n

Ejercicio 3.22, con los datos del vector \vec{v} modificados.

Considera los siguientes subespacios de \mathbb{R}^3 :

* el plano Π : $3x + 6y - z = 0$

* la recta r : $\{(1, 2, 3)\lambda / \lambda \in \mathbb{R}\}$

- Obtén una base de Π .
- Justifica si la suma de Π y r es o no suma directa.
- Justifica si Π y r son o no complementarios.
- Argumenta si es posible expresar el vector $\vec{v} = (7, 11, 63)$ como suma de un vector \vec{v}_1 del plano Π y un vector \vec{v}_2 de la recta r .
- En caso de que sea posible:

Calcula \vec{v}_1 y \vec{v}_2 .

Comprueba explícitamente que el vector \vec{v}_1 obtenido se encuentra en el plano Π .

Comprueba explícitamente que el vector \vec{v}_2 obtenido se encuentra en la recta r .

Comprueba explícitamente que $\vec{v}_1 + \vec{v}_2 - \vec{v} = \vec{0}$.

Solución 3.22

% a) Obtenemos la base del plano resolviendo su ecuación $Ax=0$ con `null()`

```
B_Pi = null(sym([3 6 -1]))
```

```
[ -2, 1/3]
[  1,  0 ]
[  0,  1 ]
```

Las dos columnas que proporciona `null()`, que son dos soluciones independientes de la ecuación $3x + 6y - z = 0$, nos dan una base del plano Π :

$$B_{\Pi} = \{(-2, 1, 0), (1/3, 0, 1)\}$$

% b) Escribimos como vector columna la base de la recta

```
B_r = [1 2 3]' ;
```

% La dimensión del plano es el n° de vectores de cualquier base del plano,
% por tanto 2

% La dim. de la recta es el n° de vectores de cualquier base de la recta,
% por tanto 1

% La unión de las bases de plano y recta genera el subespacio plano+recta

% La dimensión de (plano+recta) es el rango de las dos bases unidas.

% La suma es directa si $\dim(\text{plano+recta}) = \dim(\text{plano}) + \dim(\text{recta}) = 2+1=3$

```
[ B_Pi B_r]
```

```
rank([ ans ] )
```

```

[ -2, 1/3, 1]
[  1,  0, 2]
[  0,  1, 3]

```

3

El rango es 3 y por tanto la dimensión de la suma es 3. La suma de ambos subespacios es entonces todo \mathbb{R}^3 . Esto revela que la recta no se encuentra en el plano.

La suma es directa (suma de las dimensiones igual a la dimensión de la suma). Dicho de otra forma, la union de las bases es base (no hay que quitar ningún vector).

c) Π y r son complementarios porque su suma es directa y porque la suma es igual a \mathbb{R}^3 .

d) Es posible expresar cualquier vector \vec{v} de \mathbb{R}^3 como suma de un vector de Π y un vector de r , ya que en la expresión de \vec{v} como combinación lineal de los vectores de la base de \mathbb{R}^3 $B = \{(-2, 1, 0), (1/3, 0, 1), (1, 2, 3)\}$, $\vec{v} = c_1(-2, 1, 0) + c_2(1/3, 0, 1) + c_3(1, 2, 3)$, con $\vec{v}_1 = c_1(-2, 1, 0) + c_2(1/3, 0, 1)$ en el plano y $\vec{v}_2 = c_3(1, 2, 3)$ sobre la recta. \vec{v}_1 y \vec{v}_2 son únicos (no existe otra descomposición posible), porque las coordenadas de un vector respecto de una base son únicas, es decir, c_1 , c_2 y c_3 son únicos.

Lo aplicamos al vector v del enunciado.

```
v = [7 11 63]';
```

```
% Escribimos la base B de R3, formada por las bases de plano y recta.
```

```
B = [ B_Pi B_r ]
```

```
% Resolvemos el sistema compatible determinado con linsolve(B,v).
```

```
% Llamamos c a la solucion, que son las coordenadas de v respecto
```

```
% de la base B
```

```
c = linsolve(B,v)    % matriz ampliada [ B | v ]
```

```
7
```

```
57
```

```
2
```

```
% La solucion es (7,57,2), por tanto la combinacion lineal es
```

```
% v = 7 b1 + 57 b2 + 2 b3
```

```
% con v1 = 7 b1 + 57 b2 (en el plano) y v2 = 2 b3 (en la recta)
```

```
v1 = c(1)*B(:,1) + c(2)*B(:,2)
```

```
v2 = c(3)*B(:,3)
```

```
v1=
```

```
5
```

```
7
```

```
57
```

```
v2=
```

```
2
```

```
4
```

```
6
```

$$\vec{v}_1 = (5, 7, 57) \quad \vec{v}_2 = (2, 4, 6)$$

Comprobaciones:

```
[3 6 -1]*v1      % resultado cero si v1 se encuentra en el plano
rank([PB_r v2]) % resultado 1 si v2 se encuentra en la recta r
v1 + v2 - v      % resultado el vector 0 si la descomp. esta bien hecha

ans =
0
ans =
1
ans =
0
0
0
```

Ejemplo de obtención de base de subespacio generado, ec/s implícitas, y relación/es de dependencia lineal

Ejercicio 3.19 Considera el conjunto $\{\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4, \vec{v}_5\} \subset \mathbb{R}^4$, siendo los \vec{v}_i los vectores

siguientes en su orden, $\begin{bmatrix} 1 \\ 0 \\ -3 \\ 2 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 2 \\ -3 \end{bmatrix}, \begin{bmatrix} -3 \\ -4 \\ 1 \\ 6 \end{bmatrix}, \begin{bmatrix} 1 \\ -3 \\ -8 \\ 7 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \\ -6 \\ 9 \end{bmatrix}$

- Encuentra una base B del subespacio H generado por el conjunto.
- A partir de la base B , obtén la/s ec/ecs. implícita/s de H , denotando a las variables de \mathbb{R}^4 como x_1, x_2, x_3, x_4 .
 - Obtén el o los valores de a tales que $(a, 4, -5, -10)$ pertenezca a H . Escribe “ninguno” o “para todo a ” si ese es el caso.
 - Obtén el o los valores de a tales que $(a, 8, -10, 10)$ pertenezca a H . Escribe “ninguno” o “para todo a ” si ese es el caso.
- Añadido: Obtén las relaciones de dependencia lineal en el conjunto $\{\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4, \vec{v}_5\}$, es decir, ecuaciones de la forma $c_1\vec{v}_1 + c_2\vec{v}_2 + c_3\vec{v}_3 + c_4\vec{v}_4 = \vec{0}$, donde no todos los reales c_i son cero, que permitan despejar los vectores eliminados en el apartado a) como combinación lineal del resto.

Solución 3.19

```
v1=[1 0 -3 2]'; v2=[0 1 2 -3]'; v3=[-3 -4 1 6]'; v4=[1 -3 -8 7]'; v5=[ 2 1 -6 9]';
S=[v1 v2 v3 v4 v5]
%   1   0   -3   1   2
%   0   1   -4  -3   1
%  -3   2   1  -8  -6
%   2  -3   6   7   9

rref(S)
%   1   0   -3   0   4
%   0   1   -4   0  -5
%   0   0   0   1  -2
%   0   0   0   0   0
```

```

% A ojo vemos que solo las columnas 1,2 4 con pivotaes.
% Por tanto la base de H la forman {v1,v2,v4}

syms x1 x2 x3 x4 real
x=[x1 x2 x3 x4]'
Bx=[v1 v2 v4 x]

%[ 1, 0, 1, x1]
%[ 0, 1, -3, x2]
%[ -3, 2, -8, x3]
%[ 2, -3, 7, x4]

% Las tres primeras columnas proporcionan rango 3, pues hemos visto que
% esos tres vectores son l.i.
% Bx (con el vector generico añadido a la derecha)
% tendra rango 3 si y solo si su determinante es 0.

det(Bx) % la expresion resultante igualada a cero es la ec. implicita.

% 10*x1 - 5*x2 + 4*x3 + x4

% Por tanto la ec es 10x1 - 5x2 + 4x3 + x4 = 0

% coefs=[10 -5 4 1]

% Otra forma:
A=(null( sym([v1 v2 v4] ')))'

% [10 -5 4 1]

syms a real , v=[a 4 -5 -10]'

A*v % si v pertenece a H el resultado tiene que ser 0

% 10*a - 50

solve(ans)

% 5 % a=5

% PARA OTRO MODELO DE EJERCICIO

v=[a 8 -10 10]'
solve( A*v)

% 7 % a = 7

% Recordemos que la solucion de un SL proporciona
% el conjunto de coeficientes por los que hay que
% multiplicar las columnas para que la combinacion

```

```

% lineal de estas produzca el vector de terminos independientes

% El apartado d) nos pide calcular los coeficientes
% que producen el vector 0.

% A partir de la forma escalonada reducida de la matriz S
% podemos despejar la solucion(=coeficientes) del SLH  $Sx = 0$ 

%      1      0      -3      0      4
%      0      1      -4      0      -5
%      0      0      0      1      -2
%      0      0      0      0      0

% El sistema, incluyendo los terminos independientes, todos
% cero, queda asi en la forma escal. reducida

%      1      0      -3      0      4      |      0
%      0      1      -4      0      -5      |      0
%      0      0      0      1      -2      |      0
%      0      0      0      0      0      |      0

% La solucion es:

%  $c_1 = 3c_3 - 4c_5$ 
%  $c_2 = 4c_3 + 5c_5$ 
%  $c_3 = c_3$ 
%  $c_4 = 2c_5$ 
%  $c_5 = c_5$ 

% Y en forma parametrica

(3,4,1,0,0) * alpha + (-4,5,0,2,1) * beta

% Estos mismos vectores se pueden obtener mediante sol=null(sym(S))

sol=null(sym(S))

% [3, -4]
% [4, 5]
% [1, 0]
% [0, 2]
% [0, 1]

% Por tanto tenemos dos relaciones de dependencia lineal
% independientes entre si:

%  $3v_1 + 4v_2 + v_3 = 0$ 

%  $-4v_1 + 5v_2 + 2v_4 + v_5 = 0$ 

% De la primera podemos despejar la expresion de  $v_3$  en

```

```
% funcion de la base B

% De la segunda podemos despejar v5 en
% funcion de la base B

%  $v_3 = -3v_1 - 4v_2$ 

%  $v_5 = 4v_1 - 5v_2 - 2v_4$ 

% Notese además como los coeficientes se pueden deducir
% de forma sencilla a partir de la escalonada reducida, sin necesidad
% de despejar  $(c_1, c_2, c_3, c_4, c_5)$ , ya que es obvio,
% por los digitos de las columnas en la escalonada reducida,
% que  $3v_1 + 4v_2 + v_3 = 0$ 
% y que  $-4v_1 + 5v_2 + 2v_4 + v_5 = 0$ 
```

P.8 Semana 4 de abril

Endomorfismos en \mathbb{R}^2 con interpretación geométrica sencilla

Ejercicio 4.5 Obtén la matriz estándar de las siguientes transformaciones lineales en \mathbb{R}^2 :

- simetría respecto de la recta $y = \frac{1}{3}x$
- proyección ortogonal sobre la recta $y = \frac{1}{3}x$

Considerada la figura con los vértices P , M , Q , R y T dados en la tabla, obtén sus imágenes para las transformaciones anteriores y rellena la tabla con los resultados.

(x, y)	simetrico (x', y')	proyectado (x', y')
$P = (6, 6)$	$P' = (\quad , \quad)$	$P' = (\quad , \quad)$
$M = (6, 7)$		
$Q = (7, 7)$		
$R = (8, 6)$		
$T = (7, 5)$		

Ampliación del ejercicio: Obtén las siguientes transformaciones lineales adicionales de la figura original:

- Dilatación de un factor 2.
- La siguiente composición de aplicaciones: dilatación de un factor 2 y a continuación simetría respecto de la recta $y = \frac{1}{3}x$.
- Giro centrado en el origen, de ángulo 20 grados en el sentido contrario al de las agujas del reloj.

Denota a las matrices obtenidas para las transformaciones a), b), c), d) y e) como A_s ('s' por simetría), A_p ('p' por proyección), A_d ('d' por dilatación), A_{compos} y A_g ('g' por giro), respectivamente.

Representación gráfica de los resultados obtenidos en las cinco transformaciones.

Las funciones de MATLAB de representación gráfica se usan para la mejor comprensión de los contenidos, pero no son materia exigida en la asignatura.

Solución para las transformaciones lineales a) y b):

```

% datos = [ 6 6 ; 7 7 ; 8 6 ; 7 5 ; 6 6]' % que estan en estandar.
% se repite el ultimo punto para cerrar la figura
% en la representacion grafica

% datos(1,:) % en fila 1 coordenadas x de los vectores
% datos(2,:) % en fila 2 coordenadas y de los vectores

% S =[ 1 0 ; 0 -1 ] % matriz de la simetria referida a su base natural
% Pr=[ 1 0 ; 0 0 ] % matriz de la proyeccion referida a su base natural

% P = [ 3 1 ; -1 3 ]' % base natural para la simetria y la proyeccion

% Marco teorico:
% A = P*F*inv(P) % Cambio de base: de matriz F referida a base natural
% a matriz A referida a base estandar

% Copia de la deduccion A=P*F*inv(P) de los apuntes:

% Ecuaciones de partida:
% P xB = x => xB = inv(P)*x
% P yB = y => yB = inv(P)*y
%
% F xB = yB A x = y
%
% A partir de ellas:
% F inv(P) x = inv(P) y sustituyendo xB , yB
% P F inv(P) x = y premultiplicando por P
% A = P F inv(P)

% Ejecucion:
datos = [ 6 6 ; 6 7 ; 7 7 ; 8 6 ; 7 5; 6 6 ]'
P = [ 3 1 ; -1 3 ]' % vector de la recta de simetria o de proy. en col 1
% vector perpendicular en col 2
S = [1 0; 0 -1] ; Pr = [1 0; 0 0]
As = P*S*inv(P) % es la matriz estandar pedida (simetria)
Ap = P*Pr*inv(P) % es la matriz estandar pedida (proyeccion)
datos_s = As*datos % los vectores simetricos (las imagenes pedidas)
datos_p = Ap*datos % los vectores proyectados (las imagenes pedidas)
figure1=figure % Ejecucion
plot([ -9 24] , [ -3 8] , 'k-' ) % recta desde -3*(3,1) hasta 8*(3,1)
% de (-9,-3) a (24,8)

hold on
plot([ -9 24] , [ 0 0] , 'k:' ) % eje x
plot([ 0 0] , [ -9 24] , 'k:' ) % eje y
axis equal

plot(datos(1,:), datos(2,:), 'ok-') % original en negro 'k'
plot(datos_s(1,:), datos_s(2,:), 'or-') % transformado en rojo 'r'
plot(datos_p(1,:), datos_p(2,:), 'or-') % transformado en rojo 'r'
saveas(figure1,'figura_ejerc_4_5_ab.jpg')

```

```

datos =
    6    6    7    8    7    6
    6    7    7    6    5    6

P =
    3    -1
    1     3

S =
    1     0
    0    -1

Pr =
    1     0
    0     0

As =
    0.8000    0.6000
    0.6000   -0.8000

Ap =
    0.9000    0.3000
    0.3000    0.1000

datos_s =
    8.4000    9.0000    9.8000   10.0000    8.6000    8.4000
   -1.2000   -2.0000   -1.4000    0.0000    0.2000   -1.2000

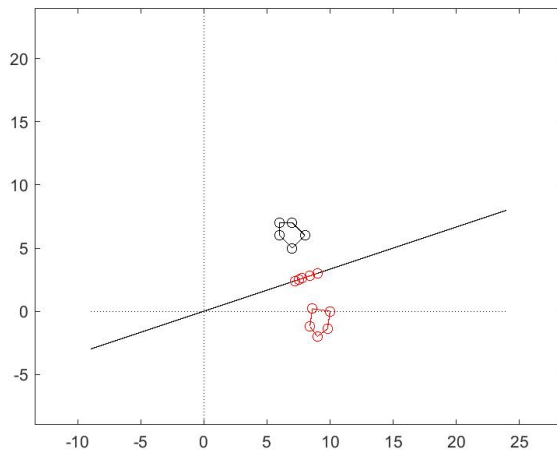
datos_p =
    7.2000    7.5000    8.4000    9.0000    7.8000    7.2000
    2.4000    2.5000    2.8000    3.0000    2.6000    2.4000

% datos_s' y datos_p' nos daran las coordenadas (x,y) de cada punto
% para cubrir, respectivamente, las columnas 2 y 3 de la tabla.

datos_s'
    8.4000   -1.2000
    9.0000   -2.0000
    9.8000   -1.4000
   10.0000    0.0000
    8.6000    0.2000
    8.4000   -1.2000

datos_p'
    7.2000    2.4000
    7.5000    2.5000
    8.4000    2.8000
    9.0000    3.0000
    7.8000    2.6000
    7.2000    2.4000

```



Añadimos al fichero .m las transformaciones lineales c), d) y e)

```
% Ad = 2*eye(2) % Matriz estandar de la dilatacion de factor 2.
                % eye(2) es la matriz identidad de orden 2.
                % Ad es la matriz asociada respecto de cualquier base.
                % Demostracion: Supuesto que en cierta base F=2*eye(2)
                % A = P * F * inv(P) = P * 2*eye(2) * inv(P) =
                % 2 * P*inv(P) = 2 * eye(2)

% Acompos = As * Ad % Matriz estandar: primero dilatacion y despues simetria
                % Notese que en este caso el orden es indiferente.

% a = 20 % angulo de 20 grados (degrees)

% Matriz estándar de giro 20 grados en sentido positivo,
% es decir, contrario al de las agujas del reloj

% Ag = [cosd(a) -sind(a); sind(a) cosd(a)] % 'd' para angulo en grados

                % Se aplicaría la misma formula, pero a=-20, si el giro pedido
                % fuera en el sentido de las agujas del reloj

figure1=figure % Ejecucion

Ad = 2*eye(2)
Acompos = As * Ad

a = 20 ; Ag = [cosd(a) -sind(a); sind(a) cosd(a)]

datos_d = Ad*datos % dilatacion aplicada a figura original
datos_compos = Acompos*datos % dilat. y despues simetria aplicada a fig. original
datos_g = Ag*datos % giro aplicado a figura original

plot([ -9 24] , [ -3 8] , 'k-' ) % recta desde -3*(3,1) hasta 8*(3,1)
                                % de (-9,-3) a (24,8)

hold on
plot([ -9 24] , [ 0 0] , 'k:' ) % eje x
plot([ 0 0] , [ -9 24] , 'k:' ) % eje y
axis equal
```

```

plot(datos(1,:), datos(2,:), 'ok-')           % original en negro 'k'
plot(datos_d(1,:), datos_d(2,:), 'ob-')       % color azul
plot(datos_compos(1,:), datos_compos(2,:), 'og-') % color verde
plot(datos_g(1,:), datos_g(2,:), 'om-')       % color magenta
saveas(figure1,'figura_ejerc_4_5_cde.jpg')

```

```

Ad =
     2     0
     0     2

```

```

Acompos =
     1.6000     1.2000
     1.2000    -1.6000

```

```

a =
    20

```

```

Ag =
     0.9397    -0.3420
     0.3420     0.9397

```

```

datos_d =
    12    12    14    16    14    12
    12    14    14    12    10    12

```

```

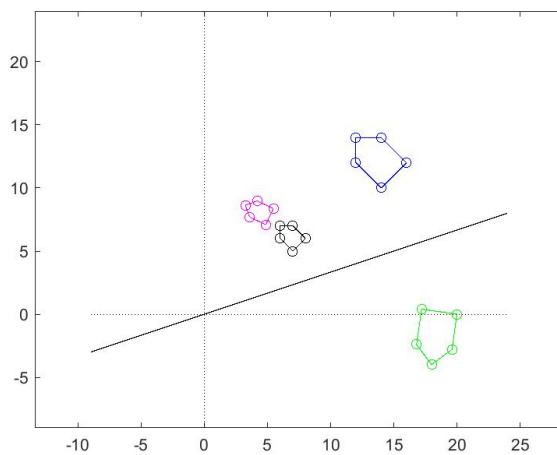
datos_compos =
    16.8000    18.0000    19.6000    20.0000    17.2000    16.8000
    -2.4000    -4.0000    -2.8000     0.0000     0.4000    -2.4000

```

```

datos_g =
     3.5860     3.2440     4.1837     5.4654     4.8677     3.5860
     7.6903     8.6300     8.9720     8.3743     7.0926     7.6903

```



P.9 Semana 11 de abril

Autovalores, autovectores y diagonalización

Ejemplo 10 Ampliado

a) Obtén los autovalores y base de cada subespacio propio para la matriz $A = \begin{bmatrix} 2 & -3 & 1 \\ 1 & -2 & 1 \\ 1 & -3 & 2 \end{bmatrix}$

b) Comprueba que las imágenes de los autovectores producen los múltiplos esperados.

c) Razona si existe base de \mathbb{R}^3 formada por autovectores de A y en caso afirmativo escribe esa base y la matriz estándar de la aplicación lineal relativa a ella.

Ejemplo 11

Estudia si son diagonalizables las siguientes matrices. En caso afirmativo obtén P invertible y D diagonal tales que $X = PDP^{-1}$, siendo X la matriz dada.

$$A = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 1 & 0 & 2 \end{bmatrix} \quad B = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 4 & 0 \\ -3 & 6 & 2 \end{bmatrix}$$

Metodología para la resolución:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Definicion de A
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% A= [ .... ; ... ; etc ]
n=3          % dimension de Rn

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Obtencion de los autovalores
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

syms s
% | A - s I | es el polinomio caracteristico de A, p(lambda)
solve(det(A-s*eye(3))) % Da las raices del pol. caracterist. con sus multiplicidades.
% Teniendo en cuenta multiplicidades seran n raices en total
% si hay raices complejas aparecera cada una con su conjugado
% las raices reales son los autovalores
% si solo hay un autovalor diferente, lo llamamos lambda
% si hay varios autovals diferentes: lambda1, lambda2, etc

% PROPIEDADES DE LAS RAICES DEL POLINOMIO CARACTERISTICO.
% Suma de todas (incluyendo multiplicidades) = traza de A
% Producto de todas (incluyendo multipl.) = determinante de A
% * Debemos comprobar que se cumplen estas propiedades. Si no se cumplen es
% que hemos determinado mal las raices de p(lambda) o sus multipl.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Obtencion de la base de cada subesp. propio
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% PROPIEDAD: 1 <= dim Vlambda <= multiplic. de lambda

s = valor % valor es el valor del autovalor cuyo subesp. propio queremos estudiar
% en vez de usar el nombre lambda tomamos s porque es más corto
```

```

B=null(sym(A-s*eye(n))) % las cols de la salida nos dan la base B de Vlambda.
                        % Es decir la base de las soluciones del SL  $Ax = sx$ 
                        % que es el mismo que  $(A - sI)x = 0$ 

                        % si, estando A bien copiada, el resultado es que no existe base,
                        % entonces el "valor" dado no corresponde a un autovalor, porque la
                        % dim de un subesp. propio siempre es como minimo 1.

null( A- s*eye(n) ) % tambien es correcto, pero en vez de dar la solucion como la
                    % obtendriamos a mano, nos da una base de vectores unitarios y
                    % ortogonales entre si.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Existencia o no de base de  $R^n$  formada por autovectores de A
% En caso afirmativo matriz asociada, es decir, diagonalizacion.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% PROPIEDAD. La suma de los subespacios propios es directa.
% La union de las bases de subesp. propios distintos es conjunto l.i.
% La base de la suma de estos subesp. es la union de sus bases.

% PROPIEDAD. Existe base de  $R^n$  formada por autovectores de A si y
% solo si se cumplen dos condiciones:
% * condicion 1: todas las raices de  $p(\lambda)$  son reales
% * condicion 2: todos los subesp. propios tienen dimension
% igual a la multiplicidad de su autovalor

% Si las n raices son reales y distintas (multip. 1), los n subesp. tienen dimension 1.
% Si las n raices son reales, pero las hay de multip. mayor que uno,
% hay que obtener la dimension de los subesp. de estas, para ver si se cumple la condicion 2.

% PROPIEDAD. Respecto de la base de  $R^n$  formada por autovectores la
% matriz asociada es diagonal, y sus elementos son los correspondientes autovalores
% en su orden.

% Por ejemplo si en  $R^3$  tuvieramos  $\lambda_1$  con mult. 2 y  $\lambda_2$  con mult. 1 y  $\dim B_1=2$ .
% Entonces las cols de  $B = [B_1 B_2]$  serian base de  $R^3$  formada por autovectores
% y  $D = \text{diag}(\lambda_1 \lambda_1 \lambda_2)$  la matriz asociada respecto de esa base
%  $A = B D B^{-1}$ 
% Comprobacion de la diagonalizacion sin necesidad de calcular la inversa de B:
%  $A*B$  ha de ser igual a  $B*D$ 

% Comprobacion separada de cada subespacio:
%  $A*\text{autovector de } B_i = \lambda_{i} * \text{autovector}$ 
%  $A * v - \lambda_{i} * v$  nos tiene que devolver una columna con 3 ceros.

```

Solución Ejemplo 10 ampliado

```

A=[2 -3 1; 1 -2 1; 1 -3 2]
%   2   -3   1
%   1   -2   1
%   1   -3   2

% a) Autovalores:
% -----

syms s ; solve(det(A-s*eye(3)))
% 0
% 1
% 1

% Los autovalores son  $\lambda=1$  doble y  $\lambda=0$  simple

```

```

% a) Base de cada subespacio propio:
% -----

s1=1; B1=null( sym( A-s1*eye(3) ) )
% [3, -1]
% [1, 0]
% [0, 1]

% La base del subespacio correspondiente a lambda = 1 es {(3,1,0), (-1,0,1)}

s2=0; B2=null( sym( A-s2*eye(3) ) )
% 1
% 1
% 1

% La base del subespacio correspondiente a lambda = 0 es {(1,1,1)}.
% Es la misma que la base de Nul(A).

% b) Comprob. de que las imagenes de los autovects producen los multiplos esperados:
% -----

A*B1      % devuelve la propia base B1, que es lo esperado para autovalor 1
% [3, -1]
% [1, 0]
% [0, 1]

A*B2      % devuelve 0*vector = vector0
% 0
% 0
% 0

% c) Razona si existe base B de R3 formada por autovectores de A.
% En caso afirmativo escribe B y la matriz estandar de la aplicacion
% relativa a esa base.
% -----

% dim B1 = 2 = multiplicidad de lambda=1 => existe base de R3 formada
% por autovectores de A

% B = {(3,1,0),(-1,0,1),(1,1,1) } base de R3 formada por autovectores de A

D=diag([1 1 0]) % matriz de la aplicacion relativa a la base B de autovectores

% D =
% 1 0 0
% 0 1 0
% 0 0 0

% No se pide comprobacion de que A = B D B^-1.
% Ni la comprobacion alternativa mas sencilla A*B = B*D.
% Pero realizamos aqui la segunda.

B=[B1 B2]

% B =

```

```
% [ 3, -1, 1]
% [ 1,  0, 1]
% [ 0,  1, 1]
```

```
A*B
% [ 3, -1, 0]
% [ 1,  0, 0]
% [ 0,  1, 0]
```

```
B*D
% [ 3, -1, 0]
% [ 1,  0, 0]
% [ 0,  1, 0]
```

Solución Ejemplo 11

```
% Pide determinar si la transformacion es diagonalizable.
```

```
% Primera transformacion lineal:
% -----
```

```
A=[2 0 0; 0 4 0; 1 0 2]
syms s
solve(det(A-s*eye(3)))
```

```
% A =
%      2      0      0
%      0      4      0
%      1      0      2
```

```
% 2
% 2
% 4
```

```
% Las tres raices son reales, y por tanto autovalores,
% por lo que se cumple la primera condición.
% Los autovalores son lambda=2 doble y lambda=4 simple.
```

```
% La segunda condicion es que para los autovalores multiples
% dim Vlambda = multiplicidad de lambda.
% Tenemos lambda=2 doble, por tanto dim Vlambda ha de ser 2.
```

```
s1=2; B1=null( sym( A-s1*eye(3) ) );
%      0
%      0
%      1
```

```
% Este subespacio no aporta dos vectores l.i., es decir dim Vlambda no es igual a 2 =>
% no existe base de R3 formada por autovectores A y A no es diagonalizable.
```

```
% Segunda transformacion lineal:
% -----
```

```
% Estudiamos ahora la matriz B del enunciado
A = [ 2 0 0 ; -1 4 0; -3 6 2] % volvemos a llamar A a la matriz
solve(det(A-s*eye(3)))
```

```
% A =
```

```

%      2      0      0
%     -1      4      0
%     -3      6      2

% 2
% 2
% 4

% De nuevo tenemos tres autovalores, y tenemos que comprobar la dimension del
% subespacio propio correspondiente a lambda=2, que tiene multiplic. 2.

s1=2; B1=null( sym( A-s1*eye(3) ) );
%      2      0
%      1      0
%      0      1

% Este subespacio aporta los dos autovectores l.i. necesarios =>
% existe base de R3 formada por autovectores de A, y por tanto A es diagonalizable.

% Se pide determinar P invertible y D diagonal tales que  $A = P D P^{-1}$ 

% P ha de tener como columnas una base de R3 de autovectores
% D los autovalores correspondientes en su orden.

s2=4; B2=null( sym( A-s2*eye(3) ) );
% 0
% 1/3
% 1

% B={ (2,1,0), (0,0,1), (0,1/3,1) } es la base de R3 formada por autovectores de A

P=[B1 B2]          % La matriz P pedida

D=diag([2 2 4])    % La matriz D pedida

% P =
% [ 2, 0, 0]
% [ 1, 0, 1/3]
% [ 0, 1, 1]

% D =
%      2      0      0
%      0      2      0
%      0      0      4

% comprobamos  $A*P = P*D$ , aunque no lo pide el ejercicio.

A*P
P*D

% Ambos productos dan el siguiente resultado:
% [ 4, 0, 0]
% [ 2, 0, 4/3]
% [ 0, 2, 4]

% En esta matriz comprobamos que:
%      el primer autovector se transforma en 2 veces el mismo.
%      el segundo en 2 veces el mismo.
%      el tercero en 4 veces el mismo.

```

P.10 Semana 25 de abril

Autovalores, autovectores y diagonalización

Ejemplo 11 con la función de MATLAB eig()

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
primera matriz a estudiar
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
A=[2 0 0; 0 4 0; 1 0 2];
[ P D ]=eig(sym(A))    % la segunda matriz, D, es una matriz diagonal
                        % con las raices de |A-lambda I| en la diagonal principal
                        % Si todas las raices son reales, entonces D es la matriz de
                        % autovalores.
                        % En ese caso la primera matriz, P, tiene como columnas
                        % los autovectores linealmente independientes obtenidos
                        % para los sucesivos autovalores, en el mismo orden en el
                        % que aparecen en D

% P =
% [0, 0]
% [1, 0]
% [0, 1]

% D =
% [4, 0, 0]
% [0, 2, 0]
% [0, 0, 2]

% La matriz no es diagonalizable. Todas las raíces del polinomio caracteristico
% son reales, pero el autovalor de multiplicidad 2 solo aporta un autovector
% en P.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
segunda matriz a estudiar
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
A = [ 2 0 0 ; -1 4 0; -3 6 2];
[ P D ]=eig(sym(A))

% P =
% [ 0, 2, 0]
% [1/3, 1, 0]
% [ 1, 0, 1]

% D =
% [4, 0, 0]
% [0, 2, 0]
% [0, 0, 2]

% La matriz es diagonalizable. Todas las raíces del polinomio caracteristico
% son reales, y el autovalor de multiplicidad 2 aporta 2 autovectores
% en P.
```

También se resolvió el Ejercicio 1 con la función de MATLAB eig()

P.11 Semana 2 de mayo

Autovalores, autovectores y diagonalización

Se resolvieron los siguientes ejercicios del Tema 5:

- Ejercicio 12 (con y sin MATLAB)
- Ejercicio 13 (con y sin MATLAB)
- Ejercicio 14 (sin MATLAB)

Ver los enunciados en Tutoría Presencial número 6 de [este enlace](#)

Se utilizó en MATLAB la función eig en el formato

```
[P D] = eig(sym(A))
```

y se añadió la función

```
diag([d1 d2 d3])
```

para definir la matriz diagonal de elementos d1, d2, d3

P.12 Semana 9 de mayo

Geometría elemental de vectores, rectas y planos en el espacio ordinario

Ejemplo 19. Considerados en \mathbb{R}^3 el vector $\vec{v} = (5, -1, 1)$ y el plano Π de ecuación $x - 2y + z = 0$.

- Obtén la proyección ortogonal de \vec{v} sobre Π .
- Obtén el simétrico de \vec{v} respecto de Π .
- Obtén la distancia de \vec{v} a Π .
- Obtén la norma del simétrico de \vec{v} .
- Obtén el ángulo en grados que forma el vector \vec{v} con su proyección sobre Π . Solución:

```
% a) Proyección ortogonal del vector v = (5,-1,1) sobre el plano Pi
% de ec. x - 2y + z = 0

% Calculamos en primer lugar, mediante productos escalares, la proy.
% ortogonal sobre la recta vectorial ortogonal al plano.

% En efecto v = proy_v_r + proy_v_pi, siendo
% proy_v_r la proyección ortogonal de v sobre la recta r ortogonal a pi
% proy_v_pi la proyección ortogonal sobre el plano, que es la pregunta.

% Para seguir la notación de los apuntes del Tema 7 llamamos z a proy_v_r

% Datos:
format rat
v = [5 -1 1]' ;
n = [1 -2 1]' ; % n es el vector normal al plano
                % Es el vector director de la recta r ortogonal a pi
                % Sus coords. son los coeficientes de la ec. implícita de pi

% Fórmulas:
% proy de v sobre <n> = v.n / (n.n) n , es el vector que denotamos z
% y seguidamente:
% proy de v sobre pi = v - proyeccion anterior

% El producto escalar u.v se puede obtener como u'*v o como dot(u,v)
% El nombre dot() proviene de que al producto escalar también se
% le conoce como producto punto, que se diferencia del producto vectorial
% que se denomina también producto cruz. En MATLAB el producto cruz
% de dos vectores a,b es cross(a,b)

z = dot(v,n) / dot(n,n) * n % vector proy sobre <n>
%      4/3
%     -8/3
%      4/3

proy_v_Pi = v - z
%      11/3
%       5/3
%     -1/3

n'*proy_v_Pi % resultado 0 es la comprobación de que la proyección obtenida
% está en el plano.
% Podemos entender la ec. de dos formas:
```

```

% * proy_v_Pi cumple la ecuación  $x - 2y + z = 0$ 
% * n y el vector proyección han de ser ortogonales, y
%   por tanto su producto escalar ha de ser 0

%       1/1125899906842624

% El valor anterior es cero, pues cero es compatible con ese valor, para la
% precisión numérica de MATLAB

% b) simétrico de v respecto del plano

% Fórmula:   vsim = v - 2*z       o       vsim = proy_v_pi- z

vsim = v - 2*z

%       7/3
%       13/3
%       -5/3

% c) distancia de v a Pi   % la norma de un vector se obtiene con norm()
dist=norm(z)

% 1277/391

format short, dist
% 3.2660

format rat

% d) norma de vsim

[norm(v) norm(vsim)] % La calculamos y a la vez comprobamos que coincide con
                    % la norma de v

% 1351/260      1351/260

format short, [norm(v) norm(vsim)]
% 5.1962      5.1962

% e) Obtén el ángulo en grados que forma v con su proyección sobre Pi.

format short, coseno = v'*proy_v_Pi/ norm(v) / norm(proy_v_Pi)

% 0.7778

% el coseno está entre 0 y 1, por tanto el ángulo es agudo

acosd(coseno)
% 38.9424

% También podríamos calcular el ángulo de la siguiente forma
% 1) Obtenemos el ángulo del vector con la recta r ortogonal a Pi,
% asegurándonos de que resulte agudo. Si no lo es usaríamos -n
% como vector normal en vez de utilizar n

n'*v
% 8

```

```

% al ser este producto escalar positivo ya sabemos que el
% vector n produce con v ángulo agudo, y no hay que
% cambiar de signo.

coseno= n'*v/ norm(n) / norm(v)
% 0.6285

angulo1=acosd(coseno)
% 51.0576

% 2) El ángulo pedido es 90-angulo1
% 38.9424

% RECORDATORIO DEL METODO DE OBTENCION DEL PROYECTADO SOBRE PI SEGUN TEMAS ANTERIORES.
% Z = V - PROYECTADO SOBRE PI, PARA OBTENER SIMETRICO

% METODO BASADO EN TEMAS 4 Y 5

% proy_v_pi = A * v
% Hay que calcular A

% Base de Pi:

Base_pi = null(sym(n'))

% [2, -1]
% [1, 0]
% [0, 1]

B = [Base_pi n] % es base de R3 formada por autovectores

% [2, -1, 1]
% [1, 0, -2]
% [0, 1, 1]

% lambda=1 para los dos primeros
% lambda=0 para n

% En efecto, los dos primeros vectores de B están en el plano
% y la proyección sobre el plano los ''transforma'' en ellos mismos.
% Tienen por tanto autovalor 1.

% El vector n se transforma en el vector (0,0,0),
% porque se proyecta sobre (0,0,0),
% por tanto n es autovector con autovalor 0.

% Respecto de esta base la matriz asociada es:

D = diag([1 1 0])

%      1      0      0
%      0      1      0
%      0      0      0

```

```

A = B * D * inv(B)

% [ 5/6, 1/3, -1/6]
% [ 1/3, 1/3, 1/3]
% [-1/6, 1/3, 5/6]

A*v    % el resultado es proy_v_pi, que ha de coincidir con el obtenido
        % con los métodos anteriores

% 11/3
% 5/3
% -1/3

% Recordamos el significado de autovectores y autovalores de la transformación:

% f ( 2, 1, 0) = ( 2, 1, 0) = 1 ( 2, 1, 0)
% f (-1, 0, 1) = (-1, 0, 1) = 1 (-1, 0, 1)
% f ( 1,-2, 1) = ( 0, 0, 0) = 0 ( 1,-2, 1)

% Otra forma de obtener A, sin llegar a escribir D, es la siguiente:

% A B = [ 2 1 0; -1 0 1; 0 0 0]'

% Podemos denotar como M la matriz de la derecha. Entonces A*B = M, y
% despejando A: A = M * B^-1

M=[2 1 0; -1 0 1; 0 0 0]'
A = M*inv(B)

% M
%  2   -1   0
%  1    0   0
%  0    1   0

% A
% [ 5/6, 1/3, -1/6]
% [ 1/3, 1/3, 1/3]
% [-1/6, 1/3, 5/6]

% Podemos comprobar en la matriz A
% que tiene la misma traza, el mismo determinante y el mismo rango que D.

[rank(A) det(A) trace(A)]
% 2 0 2

[rank(D) det(D) trace(D)]
% 2 0 2

A*v
% 11/3
% 5/3
%-1/3

% a partir de este resultado se van obteniendo los de los demás apartados

```

```

% METODO BASADO EN TEMA 3

% Descomposición única de v en proy + z a partir
% de la suma directa de dos subespacios.

% Base de Pi      junto con      n      forman base de R3
% -----
% plano          recta r
% -----

Ampliada = [B v]

[2, -1, 1, 5]
[1, 0, -2, -1]
[0, 1, 1, 1]

% es el SL que hay que resolver para obtener alpha, beta, gamma tales que
% v = alpha * (2,1,0) + beta * (-1,0,1) + gamma * (1,-2,1)

% En esta descomposición, los sumandos de la derecha resultan ser
% los vectores siguientes:

% proy_v_pi = alpha * (2,1,0) + beta*(-1,0,1)
%      z      = gamma*(1,-2,1)

rref(Ampliada)

[1, 0, 0, 5/3]
[0, 1, 0, -1/3]
[0, 0, 1, 4/3]

% alpha = 5/3, beta = -1/3, gamma = 4/3

linsolve(B,v) % Otra forma de resolver el SL Compatible Determinado
              % Da directamente la solución.

% 5/3
% -1/3
% 4/3

Base_pi*[5/3 -1/3]' % el resultado es proy_v_pi

% 11/3
% 5/3
% -1/3

n*4/3          % el resultado es z

```

P.13 Semana 16 de mayo

Espacio euclídeo canónico R^n

EJEMPLO 11 Supongamos dos clases de alimento, A y B, con las cantidades de vitamina C, calcio y magnesio dadas en la tabla. Las cantidades corresponden a miligramos por unidad de alimento.

	A	B
Vitamina C	1	1
Calcio	5	2
Magnesio	3	1

a) Demuestra que combinando las dos clases de alimentos no podemos obtener el siguiente aporte exacto:

$$\vec{v} = (17 \text{ mg de vitamina C}, 54 \text{ mg de calcio}, 31 \text{ mg de magnesio})$$

b) Determina el aporte más cercano al aporte exacto que se podría conseguir combinando los dos alimentos.

SOLUCIÓN:

```

a=[1 5 3]' ; b=[1 2 1]' ; v=[17 54 31]' ; % Datos
% a y b son l.i. pues vemos que uno no es múltiplo de otro

% Rouché-Frobenius para comprobar que v no se puede
% escribir como combinación lineal de {a,b}

baseH=[a b] % Estamos denotando como H al subespacio generado por las cols a y b,
           % es decir, H=<a,b>
[rank(baseH) rank([baseH v])] % 2 3 es el resultado
                               % ==> [a b] es l.i. y sistema incompatible

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% El vector que resulta ser el aporte más cercano posible %
% es el vector proyección de v sobre el subespacio H = <a,b>, %
% es decir, sobre el subespacio H generado por {a,b}. %
% v_proy= proy sobre <a,b> de v %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Método 1: Se determina una base ortogonal del subespacio H
% y se aplica la fórmula de proyección sobre dicho subespacio, que
% requiere la base ortogonal.
% Es decir, v_proy =
% Suma desde i=1 hasta i=d, siendo d la dimensión de H,
% de dot(v,bi)/dot(bi,bi)*bi, siendo {b1, ..., bd} base ortogonal de H.

% La base ortogonal de H la obtendremos con orth. orth() devuelve, de hecho, una base ortonormal

baseHorto=orth(baseH)
% -0.2042 0.9417
% -0.8446 -0.0314
% -0.4950 -0.3348

baseHorto'*baseHorto % Comprobación de que la base es ortonormal
                    % (U'*U = I en matrices U cuyas cols. son base ortonormal del
                    % subespacio generado por ellas)

% 1.0000 -0.0000
% -0.0000 1.0000

```

```

b1=baseHorto(:,1) ; b2=baseHorto(:,2)

v_proy=dot(v,b1)/dot(b1,b1)*b1+dot(v,b2)/dot(b2,b2)*b2

    16.8571
    54.2857
    30.5714

% La fórmula usada es válida para cualquier base ortogonal de H.
% Al haber usado base ortonormal no era necesario escribir los denominadores dot(bi,bi),
% ya que su valor es 1. Como se ha indicado más arriba, orth() devuelva directamente
% base ortonormal.

% Método 2:
% v_proy= v - z, con z proyección de v sobre el complemento ortogonal de <a,b>

baseH_co = null(sym(baseH')) % así obtenemos la matriz cuyas columnas
                             % son base del complemento ortogonal de H

% En este caso particular dim(H_co)=1 por eso es más sencillo
% proyectar antes sobre el complemento.

n=baseH_co; % por comodidad llamamos n a este vector

z=dot(v,n)/dot(n,n)*n , v_proy=v-z

% 1/3
% -2/3
% 1

% 1/7
% -2/7
% 3/7

% 118/7
% 380/7
% 214/7

% El resultado sale como fracción por haber usado sym() más arriba,
% originando que n sea simbólico, y en consecuencia las variables
% obtenidas a partir de n

double(v_proy) % para visualizarlo en formato decimal

% 16.8571
% 54.2857
% 30.5714

% En este método, al tratarse de un plano de R3, se podría haber calculado
% el vector ortogonal al plano con el producto vectorial n=cross(a,b)
% Este n no tiene por qué ser igual al anterior, aunque obviamente
% resultará ser proporcional.

% Método 3:
% Se calcula n, bien como base del complemento ortogonal o bien
% a través del producto vectorial en este caso.

```

```

% [a b n | v ] Resolviendo el SL compatible determinado con esta
% matriz ampliada, obtendremos sol=(alpha,beta,gamma)
sol=linsolve([a b n], v)

% 48/7
% 10
% 3/7

v_proy=sol(1)*a + sol(2)* b

% 118/7
% 380/7
% 214/7

% Nótese que si estuviéramos por ejemplo en R4, con dim(H)=2,
% tendríamos dos vectores en el complemento ortogonal n, o, y el
% SL a resolver sería [a b n o | v ], siendo de nuevo la proyección
% v_proy=sol(1)*a + sol(2)* b

% Método 4:
% v_proy = A*v
% A = P * pr * inv(P) siendo pr la matriz natural
% de la proyección sobre un plano en R3 pr=[1 0 0; 0 1 0; 0 0 0]
% y P la base natural con los dos primeros en el plano y el tercero
% ortogonal a él.

P=[a b n]; pr=diag([1 1 0]); A=P*pr*inv(P)

% [13/14, 1/7, -3/14]
% [ 1/7, 5/7, 3/7]
% [-3/14, 3/7, 5/14]

v_proy=A*v

% 118/7
% 380/7
% 214/7

```

P.14 Semana 23 de mayo

Espacio euclídeo canónico \mathbb{R}^n

EJERCICIO 1. Se considera el subespacio $H = \{(x, y, z, t) \in \mathbb{R}^4 / x - z - t = 0\}$

- Calcula una base ortonormal de H .
- Calcula una base ortonormal de H^\perp .
- Calcula la proyección ortogonal del vector $\vec{v} = (1, 0, 1, 1)$ sobre H .
- Calcula la distancia de \vec{v} a H .

```
hecs = [1 0 -1 -1]

%      1      0      -1      -1

% a) Base ortonormal de H
bh_orto = null(hecs) % se resuelve el SL homogéneo
% null sin sym devuelve una base que es base ortonormal
%      0      0.5774      0.5774
%      1.0000      0      0
%      0      0.7887      -0.2113
%      0      -0.2113      0.7887
%
% Las tres columnas forman la base ortonormal de H

% b) Base ortonormal del complemento ortogonal de H

bh_co = hecs' % no hay que comprobar que las
% ecs sean l.i. ya que sólo hay una.
% (x,y,z,t) pertenece a H si y solo si (x,y,z,t).(1, 0, -1, -1) = 0
% Eso es lo que nos dice la ec. de H x - z - t = 0
% 0 sea que la ec. nos dice que los vectores de H son los ortogonales
% a (1,0,-1,-1), o lo que es lo mismo, los ortogonales a H son los
% múltiplos de (1,0,-1,-1).

%      1
%      0
%      -1
%      -1

n=hecs'; % uso esta notación más sencilla, por tratarse de un único vector

n0=n/norm(n) % Por ser un solo vector directamente vale como base ortogonal.
% Dividimos por la norma para convertirlo en unitario

%      0.5774
%      0
%      -0.5774
%      -0.5774

% n0 es la base pedida

% c) proyección ortogonal de v sobre H

v=[1 0 1 1]'
```

```

z=dot(v,n)/dot(n,n)*n, vproy=v-z % si usara n0 en vez de n
                                % no tendria que dividir por dot(n0,n0) (=1)

%    1
%    0
%    1
%    1

%   -0.3333
%    0
%    0.3333
%    0.3333

%    1.3333
%    0
%    0.6667
%    0.6667

% Hemos optado por proyectar antes sobre el complemento, por ser más sencillo, al
% tratarse de un subespacio de dimensión 1.

norm(z) % distancia de v a H    0.5774

```

EJERCICIO 2. Considerado en \mathbb{R}^4 el subespacio vectorial $W = \langle (1, 0, -1, 0), (1, 1, 0, -1) \rangle$ y el vector $\vec{v} = (1, 0, 0, 13)$, se pide:

- Una base de W^\perp .
- El vector $\vec{v}_1 \in W$ más cercano a \vec{v} .
- El vector $\vec{v}_2 \in W^\perp$ más cercano a \vec{v} .
- La distancia de \vec{v} a W (la expresión más simplificada posible).

```

w    = [1 0 -1 0; 1 1 0 -1]'
% a) Base del complemento ortogonal
w_co = null(w') % Se resuelve el SL homogéneo formado por las condiciones
                % de ortogonalidad a la base de W
                % (x,y,z,t) pertenece al complemento ortogonal de W si y solo si:
                %      (1 0 -1 0) . (x,y,z,t) = 0
                %      (1 1 0 -1) . (x,y,z,t) = 0
                % Por usar null sin sym MATLAB devuelve base ortonormal

%    1    1
%    0    1
%   -1    0
%    0   -1

%    0.6205    0.1225
%   -0.4472    0.6325
%    0.6205    0.1225
%    0.1733    0.7550

% Estas dos columnas son la base pedida

% b) y c)

v=[1 0 0 13]' ;

```

```

U=w_co;          % Renombro como U que es notación frecuente en Álgebra para matrices
                  % cuyas columnas forman una base ortonormal (viene de ``unitary'')

v2=U*U'*v        % Empezamos por v2, ya que del complemento ya hemos calculado la
                  % base ortonormal

% Lo anterior también se podría obtener así:
u1=U(:,1); u2=U(:,2); v2=dot(v,u1)*u1+dot(v,u2)*u2

% no hacen falta las divisiones por dot(ui,ui) ya que son 1

v1=v-v2

% v2 =
%  3.0000
%  5.0000
%  3.0000
%  8.0000

% v1 =
% -2.0000
% -5.0000
% -3.0000
%  5.0000

% d)
dist=norm(v2)    % 10.3441

% Otra forma de resolver los apartados b y c, usando solo la base
% de W:

% La base dada en el enunciado no es ortogonal, por tanto
% no puedo usar la formula de los sumandos dot(v,bi)/dot(bi,bi)*bi

w_orto=orth(w)   % A mano se haría con Gram-Schmidt.
                  % Con orth() devuelve base ortonormal
% -0.7236    0.2764
% -0.4472   -0.4472
%  0.2764   -0.7236
%  0.4472    0.4472

% Cambio de nombre a U para hacer énfasis en que tengo base ortonormal

U=w_orto ;

v1=U*U'*v
% o
u1=U(:,1); u2=U(:,2);
v1=dot(v,u1)*u1+dot(v,u2)*u2 % no hacen falta los denominadores
                              % por ser base ortonormal

```

EJERCICIO 3. Obtén la solución de mínimos cuadrados del SL $A\vec{x} = \vec{b}$, con $A = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 1 \\ -1 & -1 \end{bmatrix}$ y $\vec{b} = (8, 7, 3, -4)$.

```
A = [1 1 -1 -1;1 -1 1 -1]'
b = [8 7 3 -4]'
```

```
rref([A b])           % se comprueba con la escalonada reducida que
                    % las cols de A son l.i. y que el SL es incompatible
                    % (la última columna añade pivote)
```

```
%   1   0   0
%   0   1   0
%   0   0   1
%   0   0   0
```

% También se podría hacer con rank:

```
rank([A b])   % 3 => sistema incompatible, pues A solo tiene 2 columnas
```

```
% Resolveríamos entonces [A | bproj], siendo bproj la proyección de b sobre
% el subespacio ColA
```

```
% bproj se podría calcular con las instrucciones
% U=orth(A) , bproj=U*U'*b
% En el enunciado del ejercicio en el Manual se justifica que la solución
% del SL de matriz ampliada [ A | bproj ] es la misma que la
% del SL de matriz ampliada [ A'*A | A'*b ]
```

```
rref([A'*A A'*b])
```

```
%   1   0   4
%   0   1   2
```

% La solución de mínimos cuadrados es por tanto (4,2)

```
% Nótese que la función de MATLAB linsolve(A,b) siendo A matriz de coeficientes con cols.
% l.i. y b vector de términos independientes, obtiene la solución de mínimos cuadrados
% en el caso de que el SL sea incompatible.
```

```
linsolve(A,b)
    4.0000
    2.0000
```

```
% No lo pide el ejercicio y con el método del Manual no hay que calcularlo,
% pero bproj=(6,2,-2,-6).
% rref([A bproj]) si que muestra un sistema compatible determinado, con solución
% (4,2)
```

```
%   1   0   4
%   0   1   2
%   0   0   0
%   0   0   0
```