

# G423. Álgebra y Geometría. Prácticas MATLAB

Ruth Carballo Fidalgo (UC). Curso 22-23

## Contents

P.1	Semana 13 de febrero	4
P.2	Semana 20 de febrero	12
P.3	Semana 27 de febrero	18
P.4	Semana 6 de marzo	25
P.5	Semana 13 de marzo	29
P.6	Semana 20 de marzo	32
P.7	Semana 27 de marzo	37
P.8	Semana 3 de abril	44
P.9	Semana 17 de abril	48
P.10	Semana 24 de abril	51
P.11	Semana 1 de mayo	53
P.12	Semana 8 de mayo	58
P.13	Semana 15 de mayo	62
P.14	Semana 22 de mayo	67

[Introducción a Matlab \(Versión 2020\)](#) preparada por R. Carballo. Incluida como referencia. Las instrucciones requeridas como contenidos de la asignatura estarán especificadas para cada práctica.

Para realizar las prácticas en las aulas de informática usaremos MATLAB a través de [Porticada](#). MATLAB 2020A → Ejecutar.

## 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](http://www.mathworks.com) con la dirección de correo de la universidad pero eligiendo una contraseña distinta de la de la Universidad.

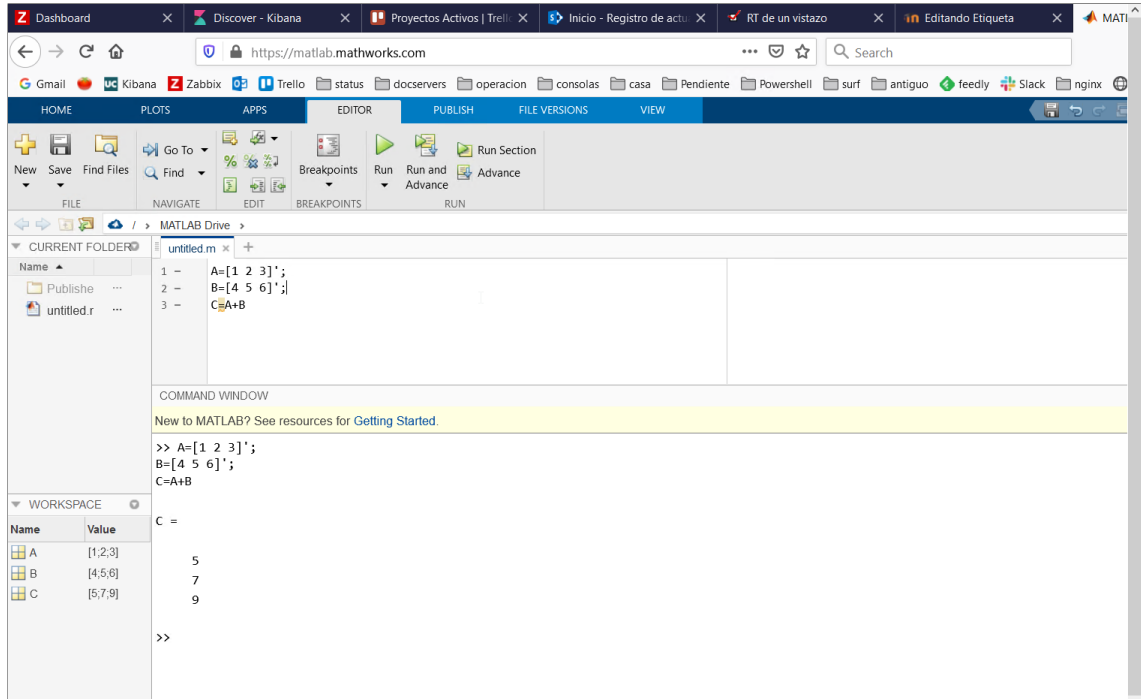
Una vez dispongas de esa cuenta, debes conectarte a este [enlace](#) del Servicio de Informática de la UC. En la nueva ventana, parte superior derecha, junto con el icono de MATLAB te aparece el enlace [Portal Matlab de la UC](#) que debes clicar a continuación. Desde la ventana emergente arrancas el proceso de acreditación, y de descarga de MATLAB y

Simulink. Fíjate en el cuadro negro y clicas en el botón azul “Inicia sesión para empezar a utilizarlos”. La ventana emergente nos pide la acreditación como miembros de la UC, y seguidamente se nos pide entrar a nuestra cuenta the Mathworks y a continuación nos lleva a la ventana de Descargas. Nos quedamos con la versión por defecto, R2022b, y clicamos en el botón azul “Descargar para Windows” (hay opciones de descarga para otros dos sistemas operativos).

## Uso de MATLAB online (por Valvanuz Fernández Quiruelas, curso 21-22)

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 13 de febrero

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 <sup>1</sup>, 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  $\alpha_i$  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 Rouché-Frobenius
```

```
rank(A) % rango = numero de cols. pivotaes
```

```
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 indeterminado si n>rango(A)
```

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

```
% Sera determinado si n=rango(A). (Ya sabemos por adelantado que no puede
```

```
% suceder, pues n=5 y A solo tiene tres filas)
```

```
ans =
```

```
 3
```

---

<sup>1</sup>El SL II se dice que es el correspondiente homogéneo del SL I

```

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

% Observamos las tres columnas pivotaes, que son la primera, la segunda y la quinta.
% Esas columnas corresponden a las incognitas x1, x2, x5 ,
% que son las variables basicas, o incognitas principales.
% x3, x4 son las 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 ; % esta linea y la siguiente se incluyen por
x4 = x4 ; % completitud pero se puede prescindir de ellas
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 solucion es x = (-24,-7,0,0,4) + x3*(2,2,1,0,0) + x4*(-3,-2,0,1,0)

% Al tener dos parametros libres la solucion la forman las infinitas
% combinaciones lineales de dos direcciones independientes, trasladadas
% mediante un vector p distinto de cero, y tambien independiente de los
% dos anteriores.

% Procedimiento 2). Una vez escogidos los parametros libres, x_3, x_4, podemos
% obtener la solucion con la funcion solve()

syms x1 x2 x3 x4 x5 real % declaramos (con esta instruccion, si alguna variable ya
                        % existiera y tuviera un valor asignado se reinicia,
                        % quedando sin valor)

x=[x1;x2;x3;x4;x5]
sol=solve(A*x-b,[x1,x2,x5]) % tambien se podria 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=[sol.x1; sol.x2; x3; x4; sol.x5]

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

% SOLUCION DEL CORRESPONDIENTE HOMOGENEO

% Aplicando el procedimiento anterior con c = [0 0 0]' (es decir, con
% igual A, la misma funcion rref()), pero los terminos independientes
% cero), se obtendria como solucion:

% sol = x3*v1 + x4*v2 con los mismos v1 y v2 que en el SL no homogeneo

% 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 funcion
% de los parametros, obteniendo
% x = x3*(2,2,1,0,0) + x4*(-3,-2,0,1,0)

% Lo podríamos hacer directamente con:
syms x1 x2 x3 x4 x5 real
x=[x1 x2 x3 x4 x5]'          % Para escribir el vector podemos utilizar los
                              % separadores ";" o trasponer (')
sol=solve(A*x,[x1,x2,x5])
    struct with fields:

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

sol = [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 solucion general del SLH no es mas que el conjunto de
% todas las combinaciones lineales de dos soluciones independientes v1 y v2.

% La funcion null() nos dara directamente esas dos soluciones v1 y v2.
% Si hubiera mas parametros libres daria mas 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 la matriz que le demos como argumento se
% interpreta como 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
% (su producto escalar es cero).

% 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 % En esta expresion 'syms' crea la variable simbolica x
            % y obliga a que sea real.
            % o lo que es lo mismo, la parte real de x es x
            % y la parte imaginaria de x es 0

% Observa los siguientes resultados:

2*x-5*x
conj(x) % Probamos a calcular el conjugado de x y la respuesta es
        % 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 especificamos ninguna, con lo que queda,
      % por defecto, que se trata de un numero complejo)

2*x-5*x % Aquí todo sigue siendo correcto

conj(x) % Nos va a decir que conj(x) es conj(x). No puede calcular
        % el conjugado de x porque x se trata como un complejo y no se
        % ha dado informacion de cual es la parte real de x y cual
        % la parte imaginaria.

-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
```

La función `transpose()`, aplicada a una matriz, columna o fila, devuelve la matriz traspuesta (sin conjugar).

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

`B=sym(A)` convierte la matriz numérica  $A$  en la matriz  $B$ , que es igual a  $A$  en sus elementos, pero siendo ahora considerados en modo simbólico.

`% Ejemplo:`

```
A = [ 1/3 2/3 ; -7/3 1/3 ]
```

```
A =  
    0.3333    0.6667  
   -2.3333    0.3333
```

```
det(A)
```

```
1.6667
```

```
det(sym(A)) % obligamos a que la funcion det() trabaje sobre expresiones simbolicas,  
            % y en consecuencia obtenemos el resultado exacto.
```

```
5/3
```

## P.2 Semana 20 de febrero

### Ejercicios de sistemas lineales y de operaciones con matrices

Guía:

- format rat
- definir la matriz de coeficientes  $A = [ \dots ; \dots ; \dots ]$
- definir el vector  $b$  como  $b = [ \dots ]'$
- definir matriz ampliada como  $A_m = [ A \ b ]$
- `rref(Am)` para obtener la forma escalonada reducida (no puede utilizarse si la matriz contiene variables, pues da resultados erróneos).
- `null(sym(A))` permite resolver SL homogéneo con matriz de coeficientes  $A$
- 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, al pasar el resultado a papel.
- obtener los rangos contando las columnas pivotaes visualmente en la forma escalonada reducida, mediante la función `rank()`, o determinando los índices de las columnas pivotaes utilizando las dos salidas posibles de `rref()` (por ejemplo `[matriz_red j]=rref(matriz)`)
- `rank()` no puede utilizarse si la matriz contiene variables, pues da resultados erróneos.
- los sistemas de tipo compatible determinado pueden resolverse con la función `linsolve(A,b)`, siendo  $A$  la matriz de coeficientes y  $b$  el vector de términos independientes.

**Ejercicio 1.2.11.** Dado el sistema de ecuaciones lineales 
$$\begin{cases} x + 2y + 3z - 5t = 1 \\ 3y + 3z + 3t = 3 \\ x + 3y + 4z - 4t = 2 \end{cases},$$

halla:

- La solución general en forma vectorial paramétrica.
- La solución particular, en forma vectorial, con valores  $z = 1$  y  $t = -2$ .
- La solución general en forma vectorial paramétrica del correspondiente sistema homogéneo.

**Ejercicio 1.2.13.** Considera el siguiente sistema constituido por 4 masas puntuales:

Punto	Masa
$\vec{v}_1 = (5, -4, 3)$	$m_1 = 2 \text{ g}$
$\vec{v}_2 = (4, 3, -2)$	$m_2 = 5 \text{ g}$
$\vec{v}_3 = (-4, -3, -1)$	$m_3 = 2 \text{ g}$
$\vec{v}_4 = (-9, 8, 6)$	$m_4 = 1 \text{ g}$

- Calcula el centro geométrico del sistema.  $\vec{v}_c = \frac{\vec{v}_1 + \dots + \vec{v}_k}{k}$

- Calcula el centro de masa,  $\vec{v}_{\text{cdm}}$ , del sistema, sabiendo que:

$$\vec{v}_{\text{cdm}} = \frac{m_1 \vec{v}_1 + \dots + m_k \vec{v}_k}{m_1 + \dots + m_k}$$

**Ejercicio 1.3.7.** 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

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

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

### Ejercicios de introducción a las aplicaciones lineales

**Ejercicio 1.4.4.** Escribe la matriz estándar  $A$  de la aplicación lineal de  $\mathbb{R}^5$  en  $\mathbb{R}^3$  que asigna a  $(x_1, x_2, x_3, x_4, x_5)$  un vector  $(y_1, y_2, y_3)$  tal que:

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

(Añadido) 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	

Soluciones:

Sólo se muestra parte de lo que sale por pantalla.

### 1.2.11

```
Am = [1 2 3 -5 1; 0 3 3 3 3; 1 3 4 -4 2]
A = Am(:,1:4)

% Resolvemos en primer lugar el apartado c)
% -----

B = null(sym(A)) % las cols de la salida son los vectores que acompañan
                  % a los parametros en la solucion del SLH
% [-1, 7]
% [-1, -1]
% [ 1, 0]
% [ 0, 1]

Amred = rref(Am) % Estamos utilizando aqui la matriz ampliada del no homogneo,
                 % pero no importa, porque lo que queremos es distinguir cuales son
                 % las incognitas principales y cuales los parametros libres

%      1      0      1      -7      -1
%      0      1      1      1      1
%      0      0      0      0      0

% Observamos (cols 3 y 4) que las incognitas z, t son los parametros libres

% Extraemos de B, los vectores que acompañan a las incógnitas

b1=B(:,1)
% -1
% -1
% 1
% 0

b2=B(:,2)
% 7
% -1
% 0
% 1

% Respuesta a solucion general del homogneo:
% x = alpha b1 + beta b2
% o x = z b1 + t b2

% Apartado a)
% -----

% El resultado Amred anterior muestra que el SL original, o
% no homogneo, es compatible.
```

```
% Despejando x, y en Amred deducimos la parte constante para x e y
% z, t son parametros libres, z=z t=t. Su parte constante es cero.
```

```
p = [-1 1 0 0]'
```

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

```
% Respuesta a la solucion general del original
```

```
% x = p + alpha b1 + beta b2
```

```
% o x = p + z b1 + t b2
```

```
% Apartado b)
```

```
% -----
```

```
xpar= p + 1*b1 + (-2)*b2 % solucion particular pedida, del SL original
```

```
% -16
% 2
% 1
% -2
```

```
% Comprobaciones
```

```
% -----
```

```
% Por ejemplo:
```

```
% A * xpar tiene que producir el vector (1,3,2)
```

```
% A * p tiene que producir el vector (1,3,2) (es la
```

```
% solu. particular con z=0, t=0
```

```
% A * b1 tiene que producir (0,0,0)
```

```
% A * b2 tiene que producir (0,0,0)
```

### 1.2.13

```
v1 = [ 5 -4 3 ]' ;
```

```
v2 = [ 4 3 -2 ]' ;
```

```
v3 = [-4 -3 -1 ]' ;
```

```
v4 = [-9 8 6 ]' ;
```

```
m=[2 5 2 1]' ; % creamos vector m cuyas entradas son las masas
```

```
M=sum(m) ; % la funcion sum() devuelve la suma de las entradas de m
```

```
c=m/M ; % vector que da la fraccion de masa
```

```
% Cada coeficiente o peso en la combinacion lineal es c(i)
```

```
% con i=1,2,3,4
```

```
cmasas = c(1) * v1 + c(2) * v2 + c(3) * v3 + c(4) * v4
```

```

% 1.3
% 0.9
% 0

% Para el centro geometrico las cuatro masas cuentan igual,
% cada una un 25%, o fracción 1/4.

cgeometrico = 1/4*v1 + 1/4*v2 + 1/4*v3 + 1/4*v4

% -1.0000
% 1.0000
% 1.5000

% Otro procedimiento consiste en escribir la combinacion lineal como un
% producto matriz-vector. La matriz tiene por columnas los vectores de
% posicion.

% Para el centro de masas el vector de los coeficientes es c
% Para el centro geometrico el vector de los coeficientes es [1/4 1/4 1/4 1/4]'

% cmasas = [v1 v2 v3 v4] * c;

% cgeo = [v1 v2 v3 v4] * [1 1 1 1]' * 1/4;

cmasas' % pido solucion como fila para que ocupe menos
% 1.3000 0.9000 0.0000

cgeo' % pido solucion como fila para que ocupe menos
% -1.0000 1.0000 1.5000

1.3.7

Am=[1 5 3; 1 2 1; 17 54 31]'
rref(Am) % aqui vemos que el sistema es incompatible.

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

% otro metodo
A=Am(:,1:2)
[rank(A) rank(Am)]

% 2 3

% => sist. incompatible

```

#### 1.4.4

```
% 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, de ahi coeficientes 1

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
% -----

% Preparamos la matriz de datos de forma que:
% primera columna son los datos de la primera semana
% segunda columna son los datos de 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
```

## P.3 Semana 27 de febrero

### La ecuación $A X = B$

#### Ejercicio 1.5.1

a) Resuelve los SLs  $A\vec{x} = \vec{b}_1$  y  $A\vec{x} = \vec{b}_2$ , con  $A = \begin{bmatrix} 1 & 3 & 2 \\ 1 & 4 & 4 \\ 1 & 7 & 8 \end{bmatrix}$ ,  $\vec{b}_1 = \begin{bmatrix} 4 \\ 7 \\ 8 \end{bmatrix}$  y  $\vec{b}_2 = \begin{bmatrix} 5 \\ 7 \\ 9 \end{bmatrix}$ .

b) Resuelve la ec. matricial  $A\vec{x} = B$ , con  $B = \begin{bmatrix} 4 & 5 \\ 7 & 7 \\ 8 & 9 \end{bmatrix}$

### Aplicaciones lineales: Giro en $\mathbb{R}^2$ alrededor del origen

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

Ver en página 33 de las diapositivas del Tema 1, **Ejercicio 1.4.12**.

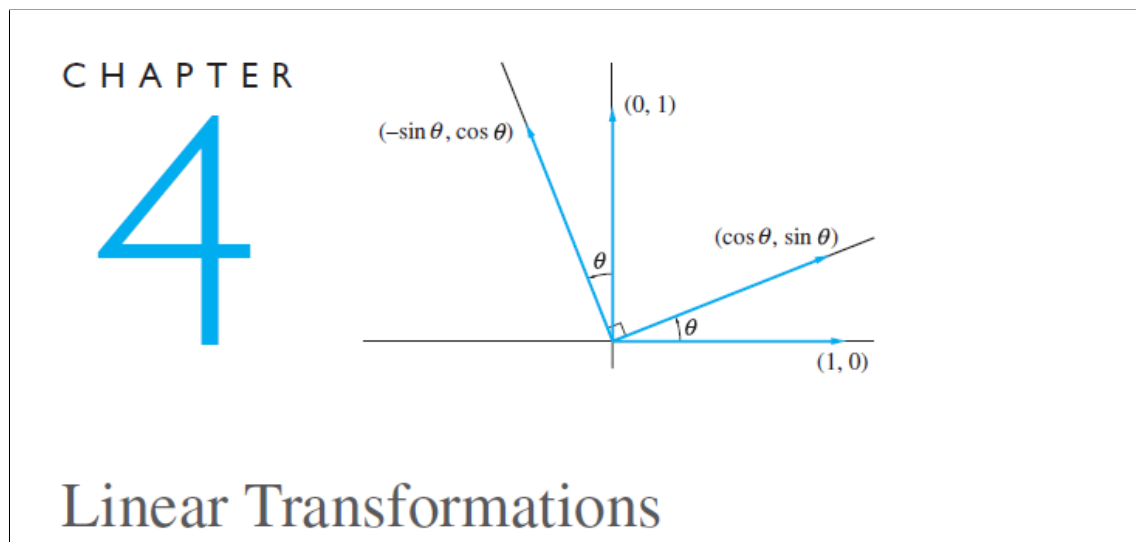


Figure 1: S.J. Leon. Linear Algebra with Applications. Edición 9<sup>a</sup>. Pearson. 2015.

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  $\vec{e}_2 = (0, 1)$ ,

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

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

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

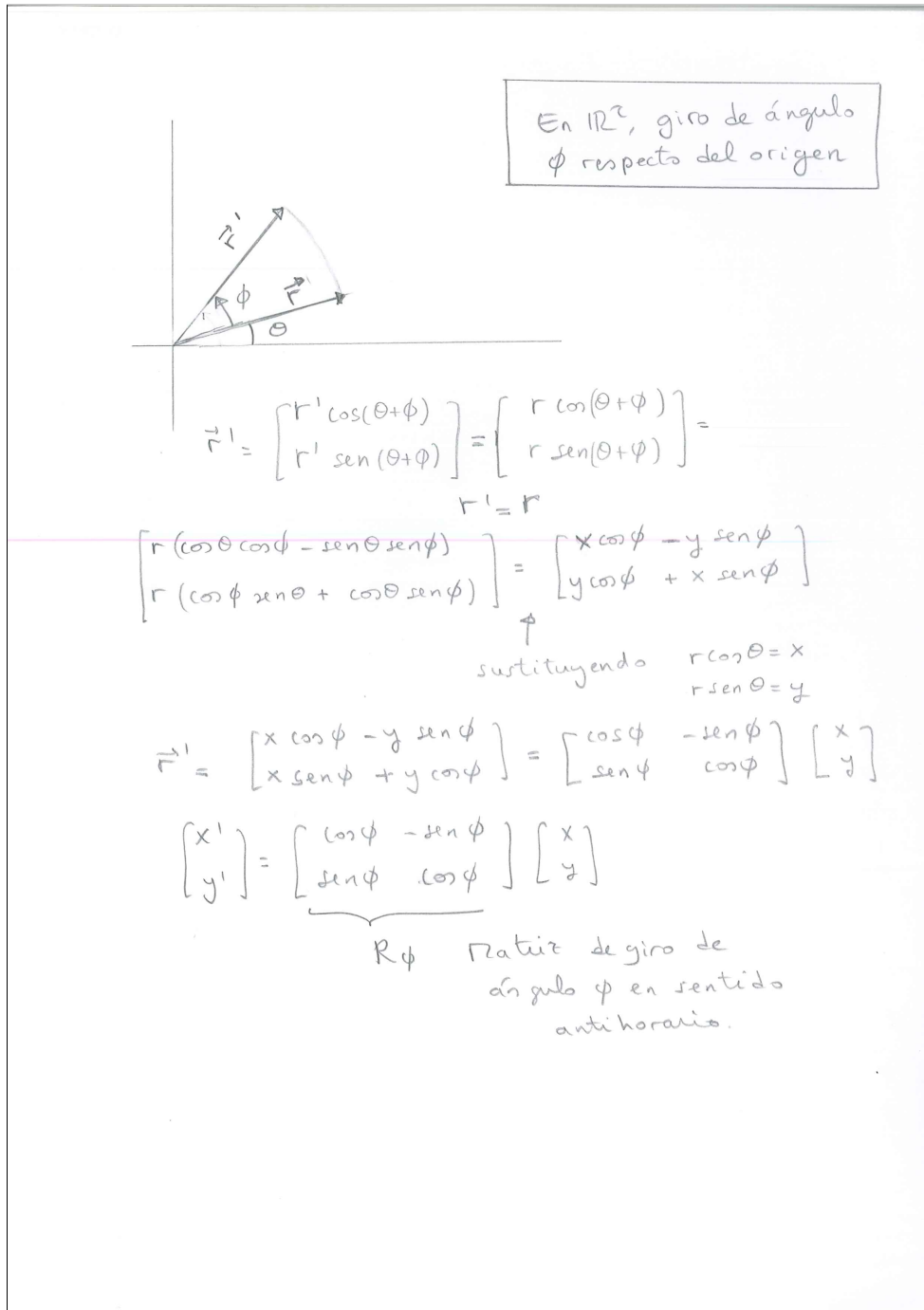


Figure 2: Otra deducción de la matriz de giro.

### 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	$\vec{v}_{\text{girado3}} =$

### Aplicaciones lineales

[HVZ12\\_Ejercicio5\\_pg38 ampliado](#) (pg. 33 de las diapositivas).

- a) Escribe la matriz de la siguiente aplicación:  $f(x_1, x_2, x_3) = (x_1 + x_2, 2x_1 - x_3)$ .  
b) Obtén la imagen del vector  $(1, 2, 3)$ .

### Sección 1.6. Inversa de un endomorfismo e inversa de una matriz cuadrada

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.

[HVZ12\\_EjemploE\\_pg44](#) (pg. 39 de las diapositivas). Encuentra la inversa de la aplicación lineal dada por  $f(x_1, x_2) = (3x_1 + 2x_2, x_1 + 2x_2)$ .

## Determinantes

OBSERVACIÓN PARA RESOLUCIÓN CON MATLAB: Si trabajas con matrices que incluyen variables simbólicas puedes utilizar la función **det()**, pero no las funciones **rank()**, ni **rref()**, ni **inv()**, ya que estas tres en general darán resultados erróneos.

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.

```
%% SOLUCION DEL EJERCICIO 1.5.1 %%

% Procedimiento 1
% -----
% Se obtiene la escalonada reducida de la matriz ampliada [A | B]
% mediante la funcion rref()
% Observamos que la parte de A se transforma en la identidad,
% por tanto los SLs Ax=b1 y Ax=b2 son compatibles determinados.
% La cuarta columna de la forma escalonada reducida es la solución del primer sistema.
% La quinta col. es la sol. del segundo sistema.

% Respecto del apartado b), la solución X es la matriz formada por las
% cols 3 y 4 de la escalonada reducida.

% Procedimiento 2
% -----
% Efectuamos rank(A), el resultado es 3, y por tanto sabemos que el SL es CD.
% Podemos utilizar la funcion linsolve(A,B), que permite resolver sistemas CD.
% La solución resultante X es la del apartado b).
% Las columnas 1 y 2 de X son las soluciones para b1 y b2 respectivamente.

% Procedimiento 3
% -----
% Efectuamos rank(A), el resultado es 3, y por tanto sabemos que A es invertible.
% Obtenemos la solución X del apartado b) mediante inv(A)*B
% De nuevo, las columnas 1 y 2 de X son las soluciones para b1 y b2.

%% 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 HVZ12_Ejercicio5_pg38  %%

>> A = [1 1 0; 2 0 -1]

%      1          1          0
%      2          0         -1

>> A*[1 2 3]'

%      3
%     -1

```

```

%% Solucion HVZ12_EjemploE_pg44  %%

>> A = [3 2 ; 1 2]
>> B=inv(A)

>> %    0.5000    -0.5000
>> %   -0.2500     0.7500

>> format rat
>> inv(A)

        1/2          -1/2
       -1/4           3/4

>> % g(x1,x2) = (1/2 x1 -1/2 x2, -1/4 x1 + 3/4 x2)

%% Solucion EJERCICIO 2.9  %%

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

```

## P.4 Semana 6 de marzo

### Operaciones con matrices, inversa y determinantes

Ejercicio 2.8. Modificado y ampliado.

Considera  $A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & a & 1 & 1 \\ 1 & 1 & a & 1 \\ 1 & 1 & 1 & a \end{bmatrix}$ .

1. Obtén el determinante de  $A$ , en función del parámetro  $a$ .
2. ¿Para qué valores de  $a$  es  $A$  invertible?
3. ¿Para qué valores de  $a$  es el rango de  $A$  igual a 4?
4. Escoge un valor de  $a$  que sea mayor o igual que 3 y para el que la matriz sea invertible y llama a esa nueva matriz  $B$ , calcula  $B^{-1}$  con la función `inv()`. Llama  $C$  a esa matriz y comprueba que  $BC = CB = I$ .
5. Obtén el determinante de  $B$ , el determinante de  $C$ , el determinante de  $2B$  y el determinante de  $\frac{1}{3}C$ . Razona los resultados obtenidos.

Puedes comprobar con esta matriz  $A$  que `rank(A)` produce el resultado 4, y que `rref(A)` da como resultado la identidad. Estos resultados son válidos si  $a \neq 1$ , pero no si  $a = 1$ .

Ejercicio 2.11 modificado. 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

$D = ABC = \begin{bmatrix} 11 & 17 & 108 \\ 17 & 1 & 108 \\ 4 & 10 & 108 \end{bmatrix}$ , calcula la matriz  $C$ . Si no existe, indícalo explícitamente.

Ejercicio 2.17 ampliado. Haciendo uso del concepto de determinante, calcula el valor de

$m$  para que el siguiente sistema tenga infinitas soluciones: 
$$\begin{cases} 2x - y + z = 0 \\ x + my - z = 0 \\ x + y + z = 0 \end{cases}$$

AÑADIDO: Determina la solución para ese valor de  $m$ .

### EJERCICIO 2.8 MODIFICADO Y AMPLIADO. SOLUCIÓN

```
syms a real
```

```
A = [ 1 1 1 1; 1 a 1 1; 1 1 a 1; 1 1 1 a]
```

```
% 1) Determinante en funcion del parametro a
```

```
det(A)
```

```
% (a - 1)^3
```

```
% 2) Para que valores de a es A invertible
```

```

% A es invertible si y solo si detA distinto de cero
% si y solo si a distinto de 1

factor(ans) % esta instruccion se usa para factorizar, aunque
            % vemos que det(A) ya nos daba el polinomio factorizado

% [a - 1, a - 1, a - 1] % (a-1) aparece tres veces porque a = 1
                        % es raiz triple del polinomio de grado 3

% otra forma de obtener las raices, si no se hubiesen
% visto claramente en el resultado de det(A)

det(A)
solve( det(A) ) % da las soluciones de la ecuacion det(A)=0
                % solve (expresion con una variable) da las soluciones
                % de igualar la expresion a cero.
% (a - 1)^3

% Lo siguiente es el resultado de solve(det(A))
% que son las tres raíces 1.
% 1
% 1
% 1

% 3) A tiene rango 4 si y solo si detA distinto de cero
% si y solo si a distinto de 1.

% 4)
B=subs(A,a,3) % Hacemos una "copia" de A, a la que llamamos B,
              % en la que a toma el valor 3. A no cambia.

% [1, 1, 1, 1]
% [1, 3, 1, 1]
% [1, 1, 3, 1]
% [1, 1, 1, 3]

% Las filas de la matriz aparecen con corchetes a los lados debido a
% que hemos partido de una matriz simbolica, al contener la variable
% simbolica a. Todas las matrices que construyamos a partir de A
% seran simbolicas.

C = inv(B)
% [ 5/2, -1/2, -1/2, -1/2]
% [-1/2,  1/2,  0,    0]
% [-1/2,  0,   1/2,  0]
% [-1/2,  0,   0,   1/2]

B*C      % comprobacion de matriz * inversa = identidad
% [1, 0, 0, 0]
% [0, 1, 0, 0]
% [0, 0, 1, 0]
% [0, 0, 0, 1]

```

```

C*B      % la misma comprobacion en sentido inversa*matriz
% [1, 0, 0, 0]
% [0, 1, 0, 0]
% [0, 0, 1, 0]
% [0, 0, 0, 1]

5) det(B)
% 8

det(C)   % el determinante de la inversa es el inverso del determ.
% 1/8

det(2*B)
% 128
% B es de orden 4, por tanto det(2B)=2^4*det(B)=16*8=128

2^4*det(B)
% 128

det(1/3*C)
% 1/648

% 1/3^4 * 1/8 = 1/81 * 1/8

% vemos que rref() y rank() dan resultados erroneos en este caso de
% matriz con parametros.

rref(A)
% [1, 0, 0, 0]
% [0, 1, 0, 0]
% [0, 0, 1, 0]
% [0, 0, 0, 1]

rank(A)
% 4

% En efecto, los valores que salen solo son validos para a distinto de 1.

EJERCICIO 2.11 MODIFICADO. SOLUCIÓN

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 17 108 ; 17 1 108; 4 10 108]

```

```

% 11 17 108
% 17 1 108
% 4 10 108

```

```

% 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

```

```

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

```

```

C=inv(A*B)*D
% 2.0000 1.0000 -29
% 2.0000 2.0000 11
% 1.0000 0 4

```

```

% La comprobacion seria similar a la anterior
 $A*B*C-D$  % es 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 infinitas soluciones 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 infinitas soluciones 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 solucion 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 13 de marzo

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

#### Ejerc. 3.19 modificado

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.

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

a) Encuentra una base  $B$  del subespacio  $H$  generado por el conjunto y la dimensión de  $H$ . Razona la respuesta.

b) A partir de la base  $B$ , obtén la/s ec/ecs. implícita/s de  $H$ , denotando a las variables de  $\mathbb{R}^4$  como  $x_1, x_2, x_3, x_4$ .

c1) 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.

c2) 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.

d) Añadido: Obtén las relaciones de dependencia lineal del conjunto  $\{\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4, \vec{v}_5\}$  es decir, ecuaciones de la forma  $c_1\vec{v}_1 + c_2\vec{v}_2 + c_3\vec{v}_3 + c_4\vec{v}_4 + c_5\vec{v}_5 = \vec{0}$ , 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

Se usa:

% para indicar los datos que salen por pantalla  
%% para los comentarios en el script

APARTADO a

-----

```
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
```

```
%% Vemos que las columnas pivotaes son 1,2 y 4.
```

```

%% Por tanto la base B de H es B={v1,v2,v4}
%% La dimensión de H es 3, porque la base tiene tres vectores

APARTADO b
-----

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.

%% El formato de salida de Bx, con corchetes a los lados y comas
%% separando los elementos, es debido a que Bx es matriz simbolica,
%% al construirse con variables simbolicas.

%% (x1,x2,x3,x4) pertenece a H si y solo si
%% Bx (con el vector generico añadido a la derecha) no aumenta el rango,
%% sino que sigue teniendo rango 3, y eso ocurre si y solo si su determinante es 0
%% (aprovechamos que Bx queda cuadrada para usar el determinante).

det(Bx) %% la expresion resultante igualada a cero es la ec. implicita.
% 10*x1 - 5*x2 + 4*x3 + x4
%% Por tanto la ec implicita es: 10x1 - 5x2 + 4x3 + x4 = 0

%% A=[10 -5 4 1], por tanto la ec. implicita tb la podemos escribir
%% como A x = 0 (x vector y 0 también vector)

%% Otra forma:
A=(null( sym([v1 v2 v4]')))'
% [10 -5 4 1]

APARTADOS c1 y c2
-----

syms a real , v=[a 4 -5 -10]'
A*v %% v pertenece a H si y solo si cumple la ecuación implicita A * v = 0
%% por tanto A*v ha de ser 0
% 10*a - 50
solve(ans)
% 5 %% a=5

%% PARA OTRO MODELO DE EJERCICIO
v=[a 8 -10 10]'
A*v
% 10*a - 70
solve(ans)

```

```
% 7          %% a = 7
```

```
APARTADO d
```

```
-----
```

```
%% Recordemos que la solución de un SL proporciona  
%% el conjunto de coeficientes por los que hay que  
%% multiplicar las columnas para que la combinación  
%% lineal de estas produzca el vector de términos independientes.  
%% El apartado d) nos pide calcular los coeficientes  
%% que producen el vector 0.
```

```
%% null(sym([v1 v2 v3 v4 v5])) nos dará las dos soluciones  
%% independientes, es decir las dos relaciones de dependencia  
%% lineal, independientes entre sí.  
%% Sabemos de antemano que tenemos dos soluciones independientes  
%% porque en la escalonada reducida teníamos dos parámetros libres.
```

```
>> null(sym([v1 v2 v3 v4 v5]))  
% [3, -4]  
% [4, 5]  
% [1, 0]  
% [0, 2]  
% [0, 1]
```

```
%% Las relaciones de dependencia lineal son:  
%% 3 v1 + 4 v2 + v3 = 0  
%% -4 v1 + 5 v2 + 2 v4 + v5 = 0
```

```
%% De la primera y segunda podemos despejar, respectivamente, los vectores dependientes  
%% v3 y v5 como combinación lineal de los vectores de B
```

```
%% v3 = - 3 v1 - 4 v2  
%% v5 = 4 v1 - 5 v2 - 2 v4
```

## P.6 Semana 20 de marzo

### Espacios vectoriales

#### Ejerc. 3.15

Sean  $\vec{v}_1 = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$ ,  $\vec{v}_2 = \begin{bmatrix} 3 \\ 6 \\ 2 \end{bmatrix}$ ,  $\vec{x} = \begin{bmatrix} 3 \\ 12 \\ 7 \end{bmatrix}$  y  $B = \{\vec{v}_1, \vec{v}_2\}$  base de  $H = \langle \vec{v}_1, \vec{v}_2 \rangle$ .

- Determina si  $\vec{x} \in H$ , y en caso afirmativo encuentra las coordenadas de  $\vec{x}$  relativas a  $B$ .
- Obtén la ec/ecs. implícita/s del sub. vec.  $H$ .
- A partir del resultado anterior obtén una nueva base  $B'$  de  $H$ .

#### Ejerc. 3.23

Considera en  $\mathbb{R}^4$  el subespacio  $F = \{(x, y, z, t) \in \mathbb{R}^4 / x - y + 2z + t = 0\}$ .

- Halla una base y la dimensión de  $F$ .
- Halla la dimensión y una base de un subespacio complementario de  $F$ , denotándolo  $F'$ .
- Descompón  $\vec{v} = (2, 1, 3, 0) \in \mathbb{R}^4$  como suma de un vector  $\vec{v}_1$  de  $F$  y un vector  $\vec{v}_2$  de  $F'$ .

**Ejerc. 3.21** Escribe el vector  $\vec{v} = (5, 6)$  como suma de dos vectores, uno sobre la recta  $\{(x, y) / y = 2x\}$  y otro sobre la recta  $\{(x, y) / y = x/2\}$  (Tomado de Lay, Lay y McDonald, "Linear Algebra and its Applications". Quinta edición. 2016. Pearson pg. 90).

### Solución 3.15

Se usa:

```
% para indicar los datos que salen por pantalla
%% para los comentarios en el script
```

```
B=[-1 0 1; 3 6 2]'
x=[3 12 7]'
```

```
%% apartado a)
%% -----
```

```
rank([B x]) %% si es 2 pertenece
             %% y si es 3 no pertenece
```

```
% 2
             %% Obtenemos que si pertenece.
```

```
xB=linsolve(B,x) % coordenadas de x en base B
```

```
%      3
%      2
```

```
%% Las coordenadas son xB=(3,2)
```

```
%% apartado b)
%% -----
```

```
syms x1 x2 x3 real
x=[x1 x2 x3]'
```

```
det([B x]) %% x=(x1,x2,x3) generico pertenece a H si y solo si
            %% la expresion anterior es cero
            %% (<=> el rango 2 de B no aumenta al añadir la columna con el vector x).
```

```
% 5*x2 - 6*x1 - 6*x3
```

```
%% La expresion anterior IGUALADA A CERO es la ec. implicita de H
%% 5*x2 - 6*x1 - 6*x3 = 0
```

```
%% apartado c)
%% -----
```

```
%% Para obtener una nueva base, resolvemos la ec. implicita obtenida:
```

```
A=[-6 5 -6] %% aqui escribimos los coeficientes de la ec.
```

```
Bprima=null(sym(A)) %% Las columnas de la matriz obtenida seran los vectores
                    %% de la nueva base B'.
                    %% Obviamente el numero de vectores sera el mismo que en la base B
                    %% del enunciado, dos, que es la dimension de H.
```

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

```

% [ 1, 0]
% [ 0, 1]

%% B' ={(5/6,1,0), (-1,0,1)}
%% quitando fracciones, podríamos tomar esta otra: {(5,6,0), (-1,0,1)}

%% Notese que el segundo vector es el mismo que en la base B

%% con rank([B Bprima]) podemos comprobar que los dos vectores
%% de Bprima estan en H, ya que no aumenta el rango,
%% que sigue tomando el valor 2 de B.

rank([B Bprima])
% 2

```

### Solución 3.23

a) Obtenemos la base de  $F$  resolviendo su ec. implícita

```

ecF= [1 -1 2 1]
B = null(sym(ecF))
B=
[1, -2, -1]
[1, 0, 0]
[0, 1, 0]
[0, 0, 1]

```

Las tres columnas forman base de  $F$ . La dimensión de  $F$  es 3. En efecto, al estar en  $\mathbb{R}^4$  tenemos cuatro variables, y con una sola ecuación quedan 3 parámetros libres, y por tanto la dimensión es 3.

$$B = \{(1, 1, 0, 0), (-2, 0, 1, 0), (-1, 0, 0, 1)\}$$

b) Subespacio complementario de  $F$  es un subespacio  $F'$  tal que  $F \oplus F' = \mathbb{R}^4$ .

$F'$  tiene que tener dimensión 1, que es la diferencia entre la dimensión de  $\mathbb{R}^4$  y la dimensión de  $F$ , y por tanto estará generado por un sólo vector.

A la vista de las columnas es obvio que el vector  $\vec{e}_1 = (1, 0, 0, 0)$  forma con  $F = [\vec{b}_1 \vec{b}_2 \vec{b}_3]$  un conjunto l.i., sin más que escribir los vectores con este orden:  $[\vec{e}_1 \vec{b}_1 \vec{b}_2 \vec{b}_3]$ .

```

e1 = [ 1 0 0 0 ]' ;
B4 = [ e1 B ]

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

```

Es obvio que las cuatro columnas son pivotales, o que el determinante,  $1 \times 1 \times 1 \times 1$ , es distinto de cero.

La base de  $F'$  que escogemos es  $B = \{(1, 0, 0, 0)\}$ , y la dimensión de  $F'$  es 1.

c) Con la base anterior de  $\mathbb{R}^4$  podemos pasar al siguiente apartado de descomponer un vector  $\vec{v}$  como suma de  $\vec{v}_1 \in F$  y  $\vec{v}_2 \in F'$ .

```

v = [ 2 1 3 0 ]' ;
c = linsolve( B4, v)      % el SL SCD tienen matriz ampliada [ B4 | v ]
7
1
3
0

% La descomposicion es la siguiente:

v2 = c(1)*B4(:,1)      % es el sumando en F'
v1 = c(2)*B4(:,2) + c(3)*B4(:,3) + c(4)*B4(:,4)      % es el sumando de F

v2 =
7
0
0
0

v1 =
-5
1
3
0

```

La respuesta es  $\vec{v}_2 = (7, 0, 0, 0)$  y  $\vec{v}_1 = (-5, 1, 3, 0)$ .  
 Más directo es obtener  $\vec{v}_2$  como se ha indicado, y a partir de este resultado obtener  $\vec{v}_1$  cómo

```
v1=v-v2
```

```

v1 =
-5
1
3
0

```

### Solución 3.21

```

format rat
B=[1 2; 1 1/2]'

%   1   1
%   2   1/2

v=[5 6]'

```

```
sol=linsolve(B,v)
```

```
%      7/3  
%      8/3
```

```
v1=sol(1)*B(:,1)
```

```
%      7/3  
%     14/3
```

```
v2=sol(2)*B(:,2)
```

```
%      8/3  
%      4/3
```

Presentamos además como se resolvería el SL a mano mediante eliminación de Gauss-Jordan, dando las sucesivas matrices ampliadas equivalentes, hasta llegar a la escalonada por filas reducida.

```
1   1   5  
2  1/2   6
```

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

```
1   1   5  
0  -1  -8/3
```

```
1   0   7/3  
0  -1  -8/3
```

```
1   0   7/3  
0   1   8/3
```

## P.7 Semana 27 de marzo

### Aplicaciones lineales

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 Ejercicio 4.5:**

```

% 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

% 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

% B = [ 3 1 ; -1 3 ]' % base natural para la simetria y la proyeccion
                    % vector de la recta de simetria o de proy. en col 1
                    % vector perpendicular en col 2

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

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

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

                %   B F inv(B) x = y   premultiplicando por B
                %   A = B F inv(B)

% Ejecucion:
datos = [ 6 6 ; 6 7 ; 7 7 ; 8 6 ; 7 5; 6 6 ]'
B = [ 3 1 ; -1 3 ]'
S = [1 0; 0 -1] ; Pr = [1 0; 0 0]
As = B*S*inv(B) % es la matriz estandar pedida (simetria)
Ap = B*Pr*inv(B) % 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)

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

hold on

axis equal % incrementos de x y de y son iguales
% datos(1,:) % en fila 1 coordenadas x de los vectores
% datos(2,:) % en fila 2 coordenadas y de los vectores

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(figure,'figura_ejerc_4_5_ab.jpg')
```

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

```
B=
    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'
```

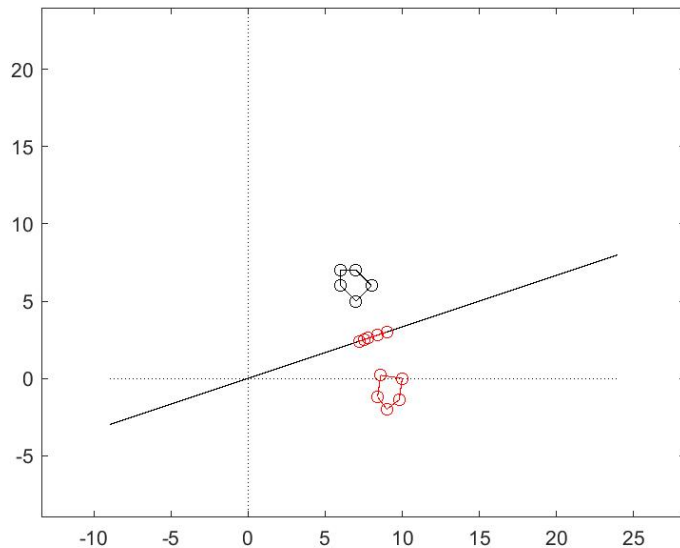
```
    8.4000   -1.2000
    9.0000   -2.0000
    9.8000   -1.4000
   10.0000    0.0000
    8.6000    0.2000
    8.4000   -1.2000
```

```
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 = B * F * inv(B) = B * 2*eye(2) * inv(B) =
                % 2 * B*inv(B) = 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 o antihorario

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

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

% 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(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(figure,'figura_ejerc_4_5_abcde.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

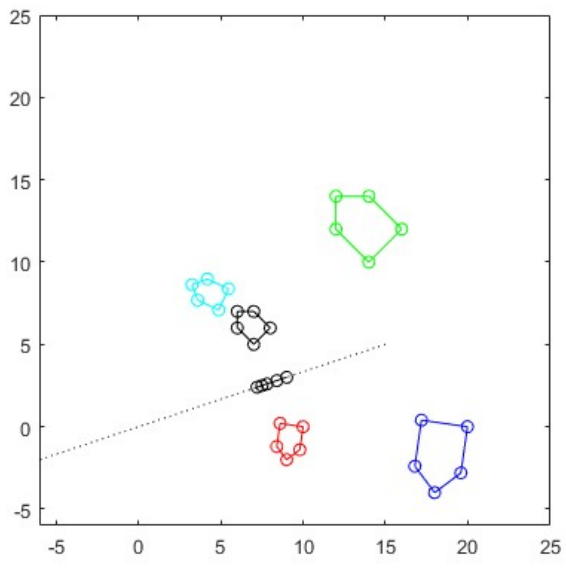
```

```
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

```



**Ejercicio 4.1, apartados a) y b).**

Dada la aplicación lineal con matriz estándar  $A = \begin{bmatrix} 1 & 4 & 5 \\ 2 & 5 & 7 \\ 3 & 6 & 9 \end{bmatrix}$

- a) Obtén una base de  $\text{Im}f$ .
- b) Obtén una base de  $\text{Ker}f$ .

**Solución Ejercicio 4.1, apartados a) y b)**

```
% A=[1 4 5; 2 5 7; 3 6 9];

%% Base de Imf
%% -----

% [ Ared j ]=rref(A);
% Bim=A(:,j)

% 1 4
% 2 5
% 3 6

%% Base de Kerf
%% -----
% Bnul=null(sym(A))
%
% -1
% -1
% 1

%% -----
%% Se cumple la relacion de dimensiones:
%% dim Espacio inicial = 3
%% dim Imf + dim Nulf = 2 + 1 = 3
%% Los dos resultados coinciden
%% -----
```

## P.8 Semana 3 de abril

### Aplicaciones lineales

**Ejercicio 4.4** Dada la aplicación lineal  $f$  con matriz asociada  $A = \begin{bmatrix} 1 & 0 & 3 & -1 \\ 0 & 1 & 2 & -1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 3 & 1 \end{bmatrix}$ ,

determina:

- Una base y la dimensión de  $\text{Im}f$ . *Es decir, base y dimensión de Cola*
- Una base y la dimensión de  $\text{Ker}f$ . *Es decir, base y dimensión de Nula*
- El núcleo de  $f$  te permite obtener las relaciones de dependencia lineal entre las columnas de  $A$ . Si el núcleo no resulta ser el vector  $(0, 0, 0, 0)$ , escribe una relación de dependencia lineal entre dichas columnas, denotándolas como  $\vec{a}_1, \vec{a}_2, \vec{a}_3$  y  $\vec{a}_4$  (por ese orden en  $A$ ).

**Añadido:** d) Obtén las ecuaciones implícitas de  $\text{Im}f$ .

**Ejercicio 4.9** Dada la aplicación lineal  $f$  en  $\mathbb{R}^3$  con matriz estándar asociada  $A = \begin{bmatrix} 2 & 2 & 0 \\ 1 & 1 & 2 \\ 1 & 1 & 2 \end{bmatrix}$ , obtén la matriz asociada a  $f$  respecto de la base  $B = \{(1, -1, 0), (-2, 1, 1), (1, 1, 1)\}$ .

*S.J. Leon. Linear Algebra with Applications. Edición 9. Pearson 2015. Pg. 212. Ejemplo 2.*

**Añadido:** Comprueba los tres invariantes: determinante, rango y traza.

Solución 4.4

```
A=[1 0 3 -1; 0 1 2 -1; 0 0 0 0; 1 0 3 1]
[Ared j]= rref(A)
```

```
ColA=A(:,j)
```

```
A =
     1     0     3    -1
     0     1     2    -1
     0     0     0     0
     1     0     3     1
```

```
Ared =
     1     0     3     0
     0     1     2     0
     0     0     0     1
     0     0     0     0
```

```
j =
     1     2     4
```

```
ColA =
     1     0    -1
     0     1    -1
     0     0     0
     1     0     1
```

a) La base de Imf son las columnas de esta matriz, ColA, y la dimensión es 3.

```
NulA=null(sym(A))
```

```
NulA =
    -3
    -2
     1
     0
```

b) La base de Kerf es  $B = \{(-3, -2, 1, 0)\}$  y la dimensión es 1.

```
% Relación de dependencia lineal:
```

```
% Denotando las columnas de a como a1, a2, a3, a4
```

```
% la relacion de dependencia lineal es
```

```
% a1*NulA(1)+a2*NulA(2)+a3*NulA(3)+a4*NulA(4) = 0
```

c)  $-3\vec{a}_1 - 2\vec{a}_2 + \vec{a}_3 = \vec{0}$

```
% Comprobacion:
```

```
a1=A(:,1)
a2=A(:,2)
a3=A(:,3)
a4=A(:,4)
a1*NulA(1)+a2*NulA(2)+a3*NulA(3)+a4*NulA(4)
```

```
a1 =
```

```
1
0
0
1
```

```
a2 =
```

```
0
1
0
0
```

```
a3 =
```

```
3
2
0
3
```

```
a4 =
```

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

```
ans =
```

```
0
0
0
0
```

```
ec = ( null(sym(ColA')) )'
```

```
ec =
```

```
[0, 0, 1, 0]
```

d) Denotando las variables de  $\mathbb{R}^4$  como  $(x_1, x_2, x_3, x_4)$  La ecuación implícita de  $\text{Im}f$  es  $x_3 = 0$ .

Solución 4.9

```
A=[2 2 0; 1 1 2; 1 1 2]
```

```
% La relacion entre la matriz A asociada a la base canónica y  
% la matriz F asociada a la base B, viene dada por la relacion:  
% A = B*F*B^-1  
% Podemos despejar F como, F=B^-1*A*B, y ese es el resultado
```

```
B=[1 -1 0; -2 1 1; 1 1 1]'  
F = B^-1*A*B
```

```
A =  
    2    2    0  
    1    1    2  
    1    1    2  
  
B =  
    1   -2    1  
   -1    1    1  
    0    1    1  
  
F =  
    0    0    0  
    0    1.0000   -0.0000  
    0    0    4.0000
```

El resultado pedido es esta matriz  $F$ .

Conservación de los invariantes

```
[ det(A)  det(F) ]  
[ rank(A) rank(F) ]  
[ trace(A) trace(F) ]
```

```
ans =  
    0    0
```

```
ans =  
    2    2
```

```
ans =  
    5    5
```

Observamos que determinante, rango y traza de las matrices relativas a las dos bases toman los mismos valores.

## P.9 Semana 17 de abril

### Autovalores, autovectores y diagonalización

#### Solución de Ejemplo 1 Ampliado

```
A= [ 10 -18; 6 -11] ; n=2

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

syms s

%% | A - s I | es el polinomio caracteristico de A
%% Queremos determinar todas las raices del polinomio, es decir
%% tanto las reales como las complejas,
%% por lo que no añadimos la opcion
%% 'real' en la expresión 'syms s'

s=solve(det(A-s*eye(2))) %% La salida contiene todas las raices del pol. caracteristico.

%% Serán n raíces en total
%% si hay raíces complejas aparecerá cada una con su conjugado
%% las raíces reales son los autovalores
%% si sólo hay un autovalor diferente, lo llamamos lambda
%% si hay varios autovals diferentes: lambda1, lambda2, etc
%% en MATLAB preferimos llamarlos s1, s2, etc

% -2
% 1

%% Los autovalores son -2 y 1, ambos de multiplicidad 1

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Obtencion de una base de cada subespacio propio
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Hay dos subespacios propios, el asociado a -2, y el asociado a 1

%% Creamos las variables s1 y s2 para almacenar sus valores.

s1=s(1) %% s1 es -2
s2=s(2) %% s2 es 1

Bmenos2=null(sym(A-s1*eye(n))) %% Las cols de la salida nos dan una base B de V(-2).
%% Es decir una base de las soluciones del SL  $A x = -2 x$ 
%% que es el mismo SL que  $(A - (-2) I) x = 0$ 

%% si, estando A bien copiada, el resultado es que no existe base,
%% entonces el "valor s1" dado no corresponde a un autovalor, porque la
%% dim de un subesp. propio siempre es como mínimo 1.

% 3/2
% 1

%% Podríamos tomar como base de este subespacio propio  $\{(3,2)\}$  si queremos evitar denominadores.

B1=null(sym(A-s2*eye(n))) %% las cols de la salida nos dan una base B de V(1).
%% Es decir una base de las soluciones del SL  $A x = 1 x$ 
```

```

%% que es el mismo SL que (A - 1 I) x = 0

% 2
% 1

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Existencia o no de base de R2 formada por autovectores de A
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Es un resultado general que la suma de los subespacios propios es directa, de forma que
%% la unión de las bases de los distintos subespacios propios es
%% un conjunto linealmente independiente.

%% Al tener dos autovectores l.i. tenemos base de R2 formada por autovectores de A

B=[Bmenos2 B1]

% [3/2, 2]
% [ 1, 1]

%% Las cols de B forman la base de R2 formada por autovectores de la aplicación lineal.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Diagonalizacion
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% La matriz de la aplicación lineal respecto a la base B de R2, formada por autovectores,
%% es la matriz D cuyos elementos son los autovalores en el mismo orden
%% en el que aparecen los correspondientes autovectores en la base B.

D=diag([s1 s2])

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

%% Comprobacion de A=BDB^-1 en la forma mas sencilla AB=BD

A*B

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

B*D

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

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% comprobación de los invariantes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[trace(A) trace(D)]
[det(A) det(D)]
[rank(A) rank(D)]

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

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Otra forma de comprobación de autovectores, cada uno con su autovalor
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

A*Bmenos2 - (-2)*Bmenos2   %% comprob. de que A*v = -2 v   para los v de V(-2)

A*B1 - 1*B1                 %% comprob. de que A*v = 1 v    para los v de V(1)

% 0
% 0

% 0
% 0

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Interpretación geométrica del endomorfismo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% La transformación lineal deja invariante la recta <(3/2,1)>, ya que los vectores de la recta
%% permanecen en la recta. Los vectores no permanecen fijos en ella, sino que cada uno se
%% transforma en -2 veces él mismo.

%% La recta <(2,1)>, es una recta de vectores fijos. Los vectores se transforman en ellos mismos.

%% Todos los vectores que están fuera de alguna de las dos rectas sufren un cambio de orientación
%% en la transformación, ya que no son vectores propios.

%% Utilizando como sistema de referencia la base B={(3/2,1),(2,1)}, la matriz asociada al endomorfismo
%% queda muy sencilla, D=[-2 0; 0 1].
%% Las coordenadas xB=(alpha,beta) del vector original simplemente se transforman en
%% yB=(alpha,-2beta) en el vector transformado.

```

## P.10 Semana 24 de abril

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

**Ejemplo 8.** a) Encuentra la ecuación implícita del plano de  $E_3$  que pasa por los puntos  $P = (1, 2, 1)$ ,  $Q = (-2, 3, -1)$  y  $R = (1, 0, 4)$ . b) Obtén el centro geométrico de los tres puntos y comprueba que se encuentra sobre el plano. *El centro geométrico o centroide de un triángulo se encuentra en la intersección de sus medianas.*

#### Solución de Ejemplo 8

```
P=[1 2 1]'; Q=[-2 3 -1]'; R=[1 0 4]';

PQ=Q-P; PR=R-P;

syms x y z real; %% Declaramos las variables porque las necesitamos
                %% para la expresión de la ec. implícita

x=[x;y;z]

%% -----
%% Ec. implícita del plano a partir de la condición de ortogonalidad
%% -----

n=cross(PQ,PR) %% cross(a,b) obtiene el producto vectorial a x b

% -1
% 9
% 6

dot(x- P ,n) %% dot(a,b) obtiene el producto escalar a.b

% -x + 9 y + 6 z - 23

%% El producto escalar anterior igualado a cero es la ec. implícita
%% -x + 9 y + 6 z - 23 = 0 o -x + 9 y + 6 z = 23

%% -----
%% Ec. implícita del plano utilizando el determinante
%% -----

det([x- P , PQ, PR])

%% El determinante anterior igualado a cero es la ec. implícita
%% -x + 9 y + 6 z - 23 = 0 o -x + 9 y + 6 z = 23

%% -----
%% Centro geométrico o punto medio o centroide de P, Q, R
%% Este punto es el baricentro del triángulo,
```

```

%% que es el punto donde se cortan las medianas
%% -----

B=1/3*(P+Q+R)

%      0
%      1.6667
%      1.3333

%% -----
%% Comprobación de que P,Q,R y B cumplen la ec. implícita.
%% La comprobación de que tres de los puntos la cumplen confirma
%% que la ec. implícita obtenida es correcta.
%% -----

%% La ec. implícita es (nx,ny,nz).(x,y,z) = 23

%% Una posible forma de confirmación es obtener n' * [P Q R B]
%% El resultado tiene que ser [23 23 23 23]

n' * [P Q R B]

% 23    23    23    23

```

## P.11 Semana 1 de mayo

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

Para el triángulo de vértices  $A = (1, 1)$ ,  $B = (5, 2.5)$  y  $C = (-3, 6.5)$ , obtén:

- Los ángulos interiores en  $A$ ,  $B$  y  $C$ .
- El perímetro.
- El área.
- El centro geométrico, centroide o baricentro.
- El cuarto vértice del paralelogramo de vértices consecutivos  $CAB$ .
- Un vector bisector en el vértice  $A$ .

Este ejercicio es muy similar al Ejercicio 1 de los apuntes de este tema. La diferencia es la inclusión aquí del apartado f), y el cambio de datos. ( $A = (1, 1)$ ,  $B = (5, 3)$ ,  $C = (-3, 7)$ ) en los apuntes).

Sol.:

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

a) Ángulos.

```
% Introducimos los datos que son las coordenadas de los tres vertices A, B, C y
% seguidamente determinamos los vectores AB, AC, BC
```

```
A=[1 1]'; B=[5 2.5]'; C=[-3 6.5]'; % introducidos como vectores columna
AB=B-A , AC=C-A , BC=C-B
```

```
AB =
  4.0000
  1.5000
AC =
 -4.0000
  5.5000
BC =
 -8
  4
```

```
% tomamos AB y AC para calcular el ángulo en el vértice A
```

```
angA=acosd(dot(AB,AC)/norm(AB)/norm(AC));
```

```
% dot( , ) es el producto escalar de los vectores que aparecen como argumentos
% norm( ) es la norma o longitud del vector
% acosd( ) es el arcocoseno en grados
```

```
% tomamos BA y BC para obtener con el mismo procedimiento el ángulo en B
% (BA=-AB)
```

```
angB=acosd(dot(-AB,BC)/norm(AB)/norm(BC));
```

```
% tomamos CA y CB para obtener el ángulo en vértice C (CA=-AC , CB=-BC)
```

```
angC=acosd(dot(-AC,-BC)/norm(AC)/norm(BC));
```

```
[ angA angB angC] % Pedimos que los valores de los ángulos se muestren como
% una fila para ahorrar espacio.
```

```
105.4713 47.1211 27.4076
```

```

angA + angB + angC - 180 % Este resultado deber salir 0, como comprobación
                          % de que la suma de los ángulos es 180

```

```
0
```

b) Perímetro.

```
peri=norm(AB)+norm(BC)+norm(AC)
```

```
20.0170
```

c) Área.

```

% Situamos el triángulo en el plano XY de R3, añadiendo a los vertices
% la coordenada z=0
% Hallamos el área a partir de la norma del producto vectorial de los vectores
% AB3, AC3, siendo AB3, AC3 los vectores AB, AC, pero considerados no sobre R2
% sino sobre el plano XY de R3.
% Es necesario hacerlo así ya que el producto vectorial es una operación en R3

```

```
AB3 = [AB ; 0] % extendemos el vector col. AB con una tercera coord. igual a 0
```

```
AC3 = [AC ; 0] % extendemos el vector col. AC con una tercera coord. igual a 0
```

```
area=1/2*norm(cross(AB3,AC3))
```

```
% cross( , ) producto vectorial de los vectores que aparecen como argumentos
```

```
AB3 =
```

```
4.0000
```

```
1.5000
```

```
0
```

```
AC3 =
```

```
-4.0000
```

```
5.5000
```

```
0
```

```
area =
```

```
14
```

d) Centroide

```
cen= 1/3*(A+B+C)
```

```
1.0000
```

```
3.3333
```

e) El cuarto vértice,  $D$ , del paralelogramo de vértices consecutivos  $CAB$ .

```
AD = AB+AC % AB+AC es el vector AD.
```

```
D = A + AD % D es igual a A + el vector AD
```

```
% También se puede entender así: OD = OA + AD
```

```
AD =
```

```
0
```

```
7
```

```
D =
```

```
1
```

```
8
```

f) Un vector bisector en el vértice A.

```

% El ángulo en A se obtuvo en un apartado anterior.
% En este apartado obtendremos un vector con origen en A que divide el ángulo
% anterior en dos partes iguales, o lo que es lo mismo, que forma el mismo ángulo
% con AB y con AC

v=AB/norm(AB)+AC/norm(AC) % es necesario tomar los vectores en las orientaciones
% AB y AC con la misma longitud, porque solo así
% la suma de los vectores da como resultante un vector
% que divide el ángulo en A en partes iguales.

v =
    0.3482
    1.1599

```

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

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

Solución:

Los apartados a) y b) se pueden resolver mediante conceptos de temas anteriores.

- Concretamente el vector proyección ortogonal y el vector simétrico se pueden obtener determinando la matriz asociada a cada transformación y aplicándola seguidamente a  $\vec{v}$ .
- El vector proyección ortogonal también puede obtenerse haciendo uso de la suma directa de subespacios, sin más que descomponer  $\vec{v}$  como suma de un vector en el plano y otro en la recta ortogonal. El sumando en el plano es la proyección ortogonal.

La solución presentada a continuación para el apartado a) parte del cálculo de la proyección de un vector sobre una recta mediante el producto escalar. Concretamente sobre la recta ortogonal al plano.

```

% -----
% 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 (el tema siguiente a éste)
% llamamos z a proy_v_r

% Datos:
format rat

```

```

v = [5 -1 1]' ;
n = [1 -2 1]' ; % n es el vector normal al plano. Lo decucimos de su ec. implícita:
                % Sus coords. son los coeficientes de la ec. implícita de pi.
                % n es el vector director de la recta r ortogonal a 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' es [1 -2 1] )
% * 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
% -----

dist=norm(z) % la norma de un vector se obtiene con norm()

```

```

% 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

```

## P.12 Semana 8 de mayo

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

**Ejemplo 19.** Utilizando métodos de temas anteriores para algunos apartados.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Proyección usando la matriz de proyección ( apartado a )
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% 1) Obtenemos la matriz estándar de proyección A
% 2)   proy_v_pi = A * v

n=[1 -2 1] '

% 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 coincide con el obtenido en la práctica
        % anterior, basándonos en el producto escalar.

% 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]'
% La expresión anterior significa que
% A [b1 b2 n] = [b1 b2 0], siendo b1 y b2 las dos primeras columnas de B

% 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

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Simetría usando la matriz de simetría ( apartado b )
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% El método es igual al anterior excepto en que el autovalor asociado a n es
% -1. Por tanto D es ahora D = diag([1 1 -1])

% La matriz M en este caso es M=[2 1 0; -1 0 1; -1 2 -1]'

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Proyección usando la descomposición única de v en  proy + z  a partir
% de la suma directa de dos subespacios ( apartado a )
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% 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

```

### Espacio euclídeo canónico $R^n$

**Ejemplo 9.** Considera en  $\mathbb{R}^2$  los vectores  $\vec{y} = (7,6)$  y  $\vec{b} = (2,1)$ . Encuentra la proyección ortogonal de  $\vec{y}$  sobre  $\langle \vec{b} \rangle$  y la distancia de  $\vec{y}$  a la recta  $\langle \vec{b} \rangle$ .

% Obtención de la proyección ortogonal mediante producto escalar.

```
y=[7 6]'  
b=[2 1]'
```

```
proy = y'*b/(b'*b) * b
```

```
8  
4
```

% Otro método. Obtención mediante descomposición ortogonal:

```
n=[1 -2]'
```

```
Ampli= [b n y]  
sol=linsolve([b n], y)
```

```
4  
-1
```

```
proy1=sol(1)*b
```

```
8  
4
```

```
% distancia  
norm(y-proy)
```

```
2.2361
```

% Recordatorio de la proyección mediante su forma diagonal:

```
B=[b n]  
Pr=[1 0; 0 0]
```

```
A=B*Pr*inv(B)
```

```
0.8000    0.4000  
0.4000    0.2000
```

```
A-A' % para comprobar que queda matriz simétrica (resultado = matriz cero)
```

```
0    0  
0    0
```

```
proy2=A*y
```

```
8  
4
```

## P.13 Semana 15 de mayo

### Espacio euclídeo canónico $\mathbb{R}^n$

**Ejemplo 13.** Obtén la matriz de la proyección ortogonal en  $\mathbb{R}^3$  sobre el plano  $W$  de ecuación implícita  $x + y + z = 0$ .

Método 1. Solución mediante la expresión  $A = UU^t$  siendo  $U$  una base ortonormal de  $W$ .

```
ecW=[1 1 1]

B=null(sym(ecW))
a1=B(:,1)
a2=B(:,2)
dot(a1,a2)    % si el resultado es distinto de cero la base no es ortogonal

b2=a2-dot(a2,a1)/dot(a1,a1)*a1    % aplicamos Gram-Schmidt

dot(a1,b2)    % comprobamos que {a1,b2} es base ortogonal.

U=[a1/norm(a1) b2/norm(b2)]

U'*U    % Comprobamos que U'U es la identidad

% La matriz de proyección es:
A=U*U'

% Lista de comprobaciones:
% A*ecW'   tiene que salir el vector 0 (el vector normal a W se proyecta sobre el origen)
% A*U      tiene que salir U (los vectores del plano W se proyectan sobre ellos mismos)
% Con esto ya está comprobado

% Pero podemos ver además que se cumple que A es simétrica:
%   A - A'   ha de ser cero
% Y que A es idempotente
%   A*A-A    ha de ser cero

% Salidas:

ecW =
     1     1     1

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

a1 =
-1
 1
 0

a2 =
-1
 0
 1
```

```

ans =
1

b2 =
-1/2
-1/2
1

ans =
0

U =
[-2^(1/2)/2, -(2^(1/2)*3^(1/2))/6]
[ 2^(1/2)/2, -(2^(1/2)*3^(1/2))/6]
[ 0, (2^(1/2)*3^(1/2))/3]

ans =
[1, 0]
[0, 1]

A =
[ 2/3, -1/3, -1/3]
[-1/3, 2/3, -1/3]
[-1/3, -1/3, 2/3]

% También podríamos haber calculado una matriz U directamente con la instrucción
U = null(ecW) % en efecto null sin sym devuelve una base que ya es ortonormal.

A=U*U' % La salida muestra la misma matriz A

% Por último, también podríamos haber utilizado la expresión de la matriz de proyección
% A = B*inv(B'*B)*B', con B base del subespacio sobre el que proyectamos.
% La expresión evita el cálculo de una base ortonormal, pero requiere de una inversa
% y de más productos de matrices.

% B la tenemos calculada desde el principio de la resolución, por tanto:

A=B*inv(B'*B)*B'
[ 2/3, -1/3, -1/3]
[-1/3, 2/3, -1/3]
[-1/3, -1/3, 2/3]

Método 2. Solución mediante diagonalización.

ecW=[1 1 1]

V1=null((ecW)) % Denominamos a la base V1, porque los vectores de W
% son autovectores del autovalor 1
% Aunque estamos tomando base ortonormal, nos valdría cualquier base.

V0=ecW' % La base del complemento ortogonal son los coeficientes de las
% ecuaciones. Y por otra parte el complemento ortogonal es V0,
% puesto que los vectores del complemento son ortogonales a W
% y por tanto se proyectan en el vector 0.
% Es decir, A*v = 0*v para los v del complemento.
% Por tanto los vectores de ese complemento son autovectores
% del autovalor 0.

```

```

                % W + Worto tienen dimension 2+1

D=diag([1 1 0])

A=[V1 V0]*D*[V1 V0]^-1    % 0    A=[V1 V0]*D*inv([V1 V0])

% Salidas:

ecW =
    1    1    1

V1 =
   -0.5774   -0.5774
    0.7887   -0.2113
   -0.2113    0.7887

V0 =
    1
    1
    1

D =
    1    0    0
    0    1    0
    0    0    0

A =
    0.6667   -0.3333   -0.3333
   -0.3333    0.6667   -0.3333
   -0.3333   -0.3333    0.6667

format rat    % Cambiamos de formato para ver A en formato racional
A
    2/3        -1/3        -1/3
   -1/3         2/3        -1/3
   -1/3        -1/3         2/3

```

**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  $H$ .

```
ecH = [1 0 -1 -1]
```

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

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% a) Base ortonormal de H
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
BH_orto = null(ecH) % 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 = ecH' % 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=ecH'; % 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 y
```

```
% d) distancia de v a H
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% En vez de calcular la proyección mediante la matriz de proyección,
```

```
% obtendremos primero z, y seguidamente vproy como vproy=v-z, ya que z lo necesitamos
% también para la distancia.
```

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

```
z=dot(v,n)/dot(n,n)*n, vproy=v-z % si usara n0 en vez de n
% no tendría 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
```

```
% Vemos ahora el resultado en formato racional:
```

```
format rat; vproy
```

```
% 4/3
```

```
% 0
```

```
% 2/3
```

```
% 2/3
```

```
format short
```

```
norm(z) % distancia de v a H
```

```
0.5774
```

## P.14 Semana 22 de mayo

### Tema 7. Espacio euclídeo canónico $\mathbb{R}^n$

#### Función eig

```
% Consideramos la transformación lineal A correspondiente
% a la simetría ortogonal respecto al plano x+y+z=0 en R3.

n = [1 1 1]'          % vector normal al plano o base
                        % del subespacio propio V(-1)
Bpi = null(sym([1 1 1])) % la solución de la ec. da la base del plano,
                        % o base del subespacio propio V(1)

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

B = [Bpi n] ;        % Base de R3 formada por autovectores de A

D=diag([1 1 -1])

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

A= B*D*inv(B)        % Acabamos de obtener la matriz referida a la base canónica.

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

format rat

A=double(A)          % para quitarle la forma simbólica

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

% Vamos a utilizar esta matriz para analizar la funcion eig()

A=[ 1 -2 -2; -2 1 -2; -2 -2 1]*1/3

eig(A)                % Devuelve las raíces del polinomio característico.
                        % En este caso todas son reales, por tanto autovalores

% -1
%  1
%  1
```

```

[B D]=eig(sym(A)) % D es la matriz diagonal con los autovalores en la diagonal
                  % principal.
                  % B tiene por columnas los autovectores asociados al autovalor
                  % que ocupa la misma columna en D.

                  % Por ser A diagonalizable, B es una matriz cuadrada e invertible.
                  % Podemos comprobar que su determinante es distinto de cero.
                  % La base de cada subespacio propio es igual a la que se calcula
                  % a mano, o con MATLAB usando null(sym(A-lambda*I)).
                  % B es base de R3 formada por la unión de esas bases.

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

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

det(B)           % En nuestro caso ya sabíamos que era diagonalizable.
% 3

% Repetimos el proceso pero sin usar sym() dentro de eig()

format short

[B D]=eig(A)     % Devuelve una base B de R3 en la que la base aportada por cada
                  % subespacio propio es base ortonormal.

                  % Como en este caso A es simétrica, los subespacios
                  % propios son además ortogonales entre sí, por tanto B es base
                  % ortonormal de R3.

% B
%   0.5774    0.5228    0.6272
%   0.5774    0.2817   -0.7664
%   0.5774   -0.8045    0.1392

% D =
%  -1.0000         0         0
%         0    1.0000         0
%         0         0    1.0000

det(B)
% 1

```

```

B*B'          % comprobación de que B es una matriz ortogonal:

%   1.0000    0.0000    0.0000
%   0.0000    1.0000    0.0000
%   0.0000    0.0000    1.0000

% Vemos a continuación un ejemplo con una matriz no diagonalizable:

A=[3 3 4; 0 3 1; 0 0 1]

%   3   3   4
%   0   3   1
%   0   0   1

eig(A)
%   3
%   3
%   1

[B D]=eig(sym(A)) % Vemos solo 2 columnas en B, porque el subespacio propio de
                  % lambda=3 tiene dimensión 1.

% B =
% [-5/4, 1]
% [-1/2, 0]
% [ 1, 0]

% D =
% [1, 0, 0]
% [0, 3, 0]
% [0, 0, 3]

% Ahora sin utilizar sym()

[B D]=eig(A) % vemos que los dos primeros autovectores son proporcionales.
              % SIEMPRE QUE USEMOS eig en este modo debemos calcular seguidamente
              % det(B), para saber si realmente tenemos una base de R3 o no.

% B =
%   1.0000   -1.0000   -0.7454
%           0     0.0000   -0.2981
%           0           0     0.5963

% D =
%   3   0   0
%   0   3   0
%   0   0   1

det(B)
%   1.3240e-16

```

## Función orth

```
% Dada una matriz A, la función orth() calcula una base ortonormal del subespacio
% generado por las columnas de A, o ColA.

% Ejemplo, en el que preparamos una matriz cuya columna tercera es combinación lineal
% de las dos primeras

a1=[1 2 3 4]'; a2= [0 1 2 3]'; a3=2*a2+3*a1; a4=[0 0 0 2]';
A=[a1 a2 a3 a4]
orth(A)

%      1      0      3      0
%      2      1      8      0
%      3      2     13      0
%      4      3     18      2

%    -0.1256   -0.2651    0.8645
%    -0.3350   -0.4220    0.2076
%    -0.5444   -0.5788   -0.4494
%    -0.7587    0.6455    0.0877

% La transformación lineal de R4 en R4 tiene como imagen un subespacio de
% dim=3, que es ColA.
```

**Ejercicio 16 modificado.** En vez de rotación de ángulo 90 grados, ángulo de 50 grados.

- En  $\mathbb{R}^3$ , obtén la matriz estándar  $A$  correspondiente a una rotación de 50 grados alrededor del eje dirigido según el vector  $\vec{v} = (1, 1, 0)$ .
- Comprueba que  $A$  es ortogonal.
- Obtén el transformado del vector  $\vec{v} = (1, -1, 0)$ .

```
% Apartado a)
% -----
```

```
n=[1 1 0]'; % Base del complemento ortogonal del plano de rotación
```

```
B=null(n'); % Base del plano de rotación. Eje de rotación y
% plano de rotación son uno complemento ortogonal del otro,
% por tanto las componentes de n (base del eje de rotación) son
% los coeficientes de la implícita del plano de rotación.
```

```
% Al usar null() obtenemos base ortonormal.
% En efecto necesitamos una base de este tipo para poder usar
% la expresión matricial del giro que conocemos de temas anteriores.
```

```

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

% -0.7071
% 0.7071
% 0

% 0
% 0
% 1

cross(u1,u2)

% 0.7071
% 0.7071
% 0

% Observamos que u1 x u2 tiene el mismo sentido que n, por tanto
% la base [u1,u2,n] tiene la orientación requerida,
% y la ortonormalización de los vectores del plano requerida
% (u1 y u2 unitarios y ortogonales entre sí)
% para que la matriz de rotación respecto de esta base sea:

% R = [cosd(a) -sind(a) 0; sind(a) cosd(a) 0; 0 0 1],
% con a = angulo dado en grados.

% Si u1 x u2 tuviera sentido contrario a n tomaríamos como base [b1 b2 n],
% con b1=u2 y b2=u1,
% ya que así, b1 x b2 tiene el sentido de n, y la base queda bien orientada.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Aclaremos la explicación de la orientación de la base
% comparando con una rotación más sencilla.
% Si tomamos la base canónica [e1 e2 e3], con [e1 e2] plano de giro
% y e3 dirección y sentido de giro, se considera sentido positivo de rotación
% el que lleva e1 hacia e2 por el camino más corto, y esto corresponde a que
% e1 x e2 tenga el sentido de e3.
% De este criterio del giro positivo de e1 hacia e2, también adoptado en R2
% (denotado como sentido antihorario), obtuvimos
% que la parte XY de la matriz de rotación tiene la forma:

% cos(a) -sin(a)
% sin(a) cos(a)

% Podemos usar la misma matriz para nuestro problema,
% con el giro según otro vector n, distinto de e3,
% siempre que [u1 u2 n] replique [e1 e2 e3] en el sentido de que:
% {u1,u2} sean base ortonormal del plano
% y que n tenga el mismo sentido que el del producto vectorial u1 x u2
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

B=[u1 u2 n] ; ang = 50 ;

% La matriz de rotación respecto de la base natural dada es:

R = [cosd(ang) -sind(ang) 0; sind(ang) cosd(ang) 0; 0 0 1]

% 0.6428   -0.7660     0
% 0.7660    0.6428     0
% 0         0          1.0000

% Ahora hay que realizar el cambio de base a la canónica,
% pues el apartado a) pide la matriz estándar:

A = B*R*inv(B)

% 0.8214    0.1786    0.5417
% 0.1786    0.8214   -0.5417
% -0.5417   0.5417    0.6428

% Apartado b)
% -----

A*A'
% 1.0000    0.0000     0
% 0.0000    1.0000   -0.0000
% 0         -0.0000    1.0000

% Apartado c)
% -----

v=[1 -1 0]'
A*v

% 0.6428
% -0.6428
% -1.0834

```