



Université Abdelmalek Essaadi
École Nationale des Sciences Appliquées AI
Hoceima



Cours d' *Informatique3*: **MATLAB**

MATLAB POUR L'INGÉNIEUR

Chapitre 3

Représentations Graphiques

Partie 1

Pr. Amina GHADBAN

Représentation 1D (2/21)

➔ Quelques spécifications et/ou attributs pour la commande plot

'y' : jaune	'.' : point
'm' : magenta	'o' : cercle
'c' : cyan	'x' : marque x
'r' : rouge	'+' : plus
'g' : vert	'*' : étoile
'b' : bleu	's' : carré
'w' : blanc	'd' : losange
'k' : noir	'p' : pentagone
'^' : triangle (haut)	'h' : hexagone
'v' : triangle (bas)	'-' : trait plein
'<' : triangle (gauche)	'.' : pointillé court
'>' : triangle (droit)	'-' : pointillé long
'-.' : pointillé mixte (tiret point)	

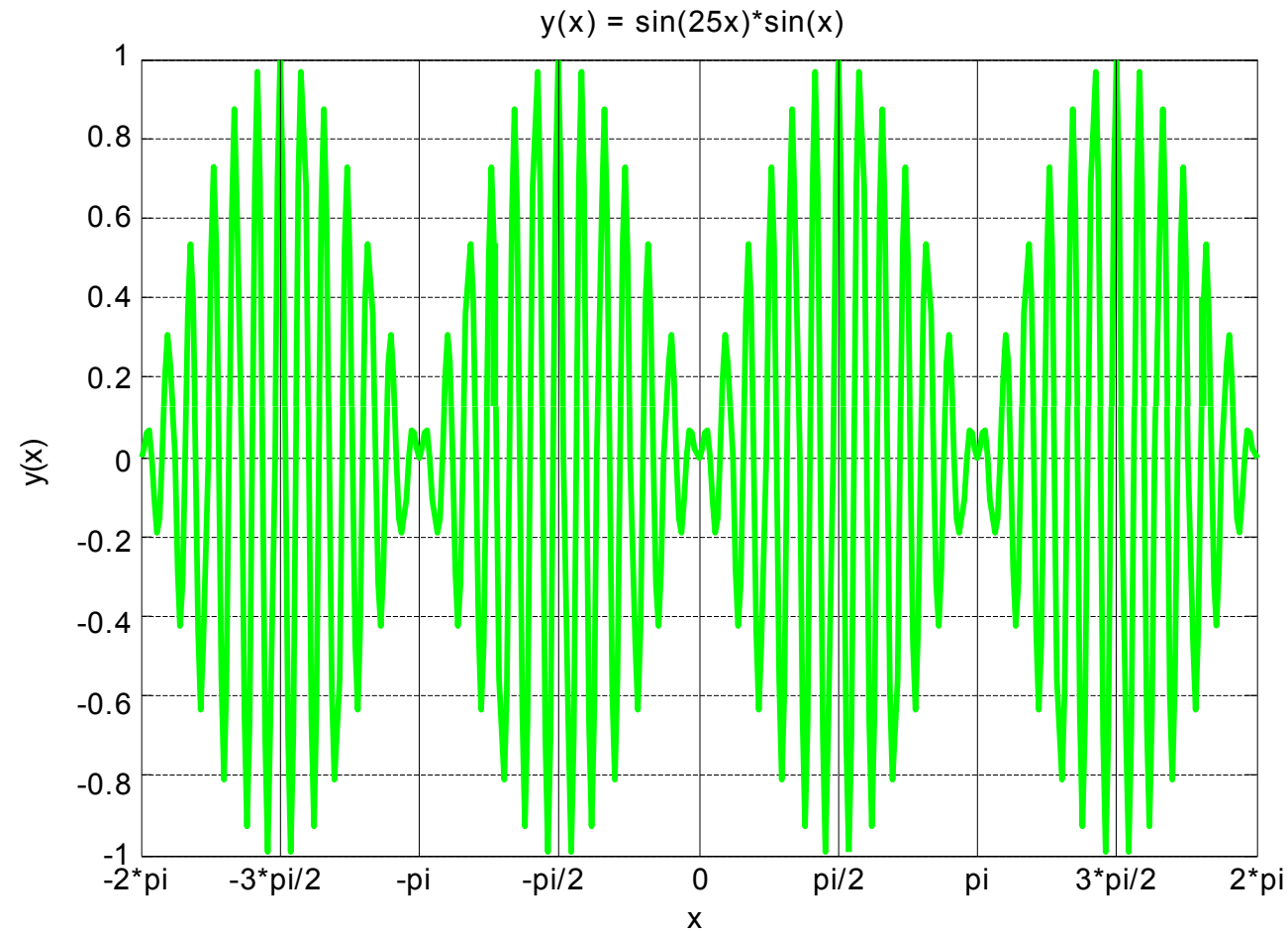
Représentation 1D (7/21)

- ➔ L'instruction **axis** est utilisée pour préciser les plages de la représentation graphique. Sa syntaxe est **axis([x_{min} x_{max} y_{min} y_{max}])**.
- ➔ Il est parfois souhaitable d'utiliser l'instruction **set** pour changer l'écriture des abscisses.

```
x = -2*pi : pi/100 : 2*pi;
y = sin(25*x).*sin(x);
plot(x,y,'g','linewidth',2);
axis([-2*pi 2*pi -1 1])
set(gca,'XTick',-2*pi:pi/2:2*pi);
set(gca,'XTickLabel',{'-2*pi','-3*pi/2','-pi','-pi/2','0','pi/2','pi','3*pi/2','2*pi'});
grid on
xlabel('x');
ylabel('y(x)');
title('y(x) = sin(25x)*sin(x)');
```

Représentation 1D (8/21)

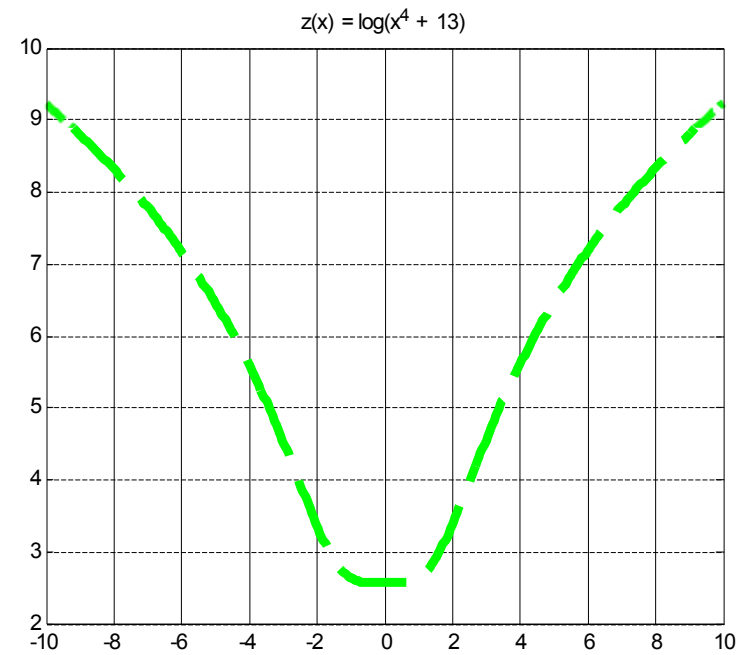
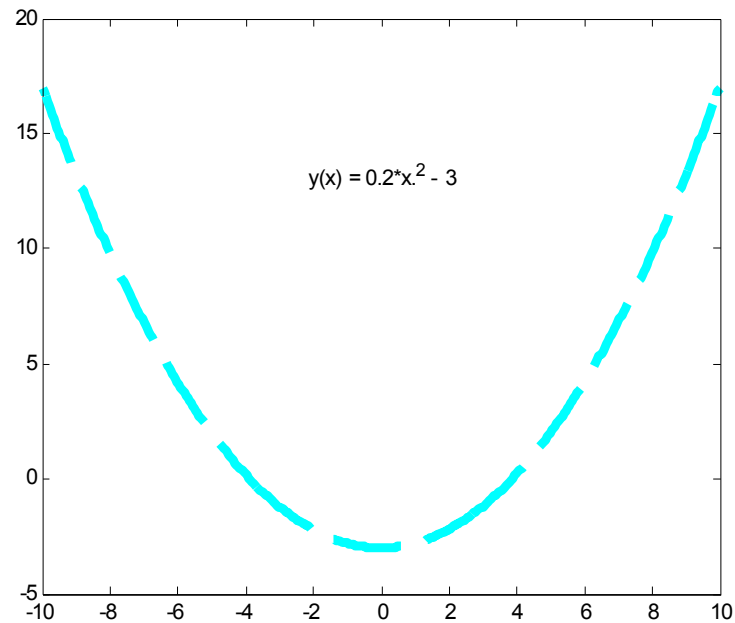
➔ Exemple : Exécution du code



Représentation 1D (9/21)

➔ Exemple

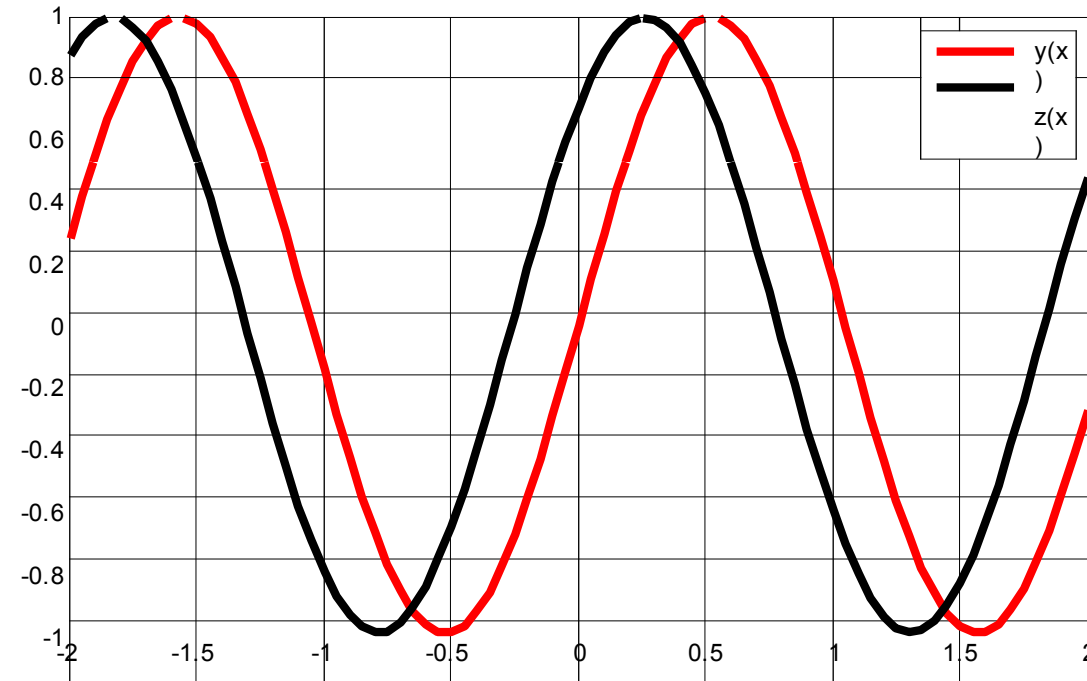
```
x = -10 : 0.05 : 10;  
y = 0.2*x.^2 - 3;  
z = log(x.^4 + 13);  
figure (1);  
plot(x,y,'-c','linewidth',4);  
gtext('y(x) = 0.2*x.^2 - 3');  
figure (2);  
plot(x,z,'-.g','linewidth',4);  
grid on  
title('z(x) = log(x^4 + 13)');
```



Représentation 1D (10/21)

➔ Exemple

```
x = -2 : 0.05 : 2;  
y = sin(3*x);  
z = sin(3*x + pi/4);  
plot(x,y,'r',x,z,'k');  
grid on  
legend('y(x)', 'z(x)');
```



Représentation 1D (11/21)

➔ L'instruction **subplot** permet de décomposer une fenêtre en sous-fenêtres et d'afficher sur chacune de ces sous-fenêtres une figure différente. Cette commande a la syntaxe suivante :

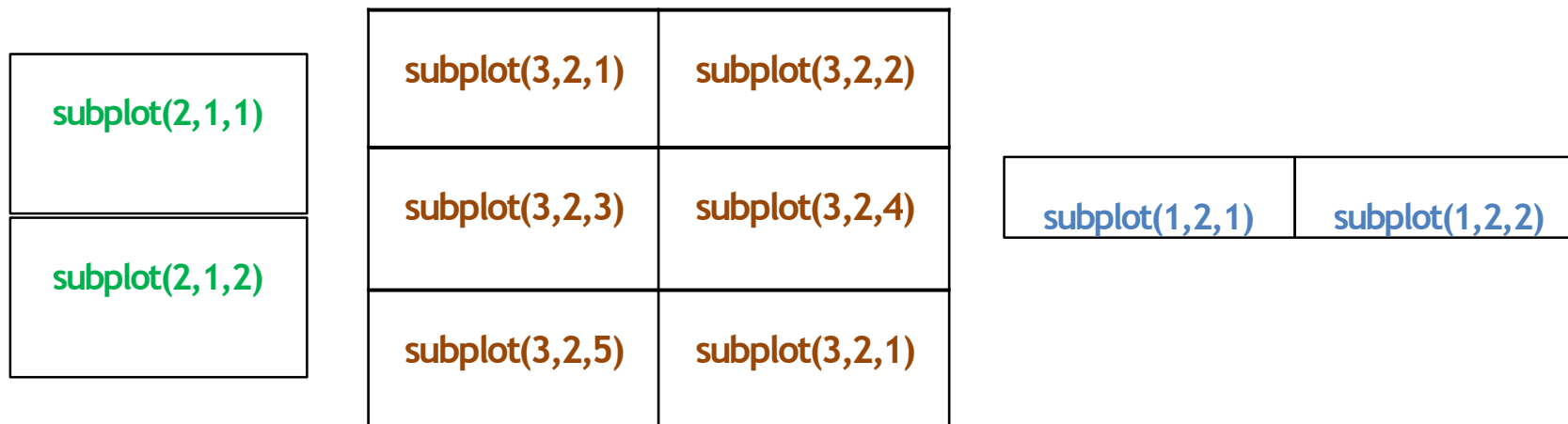
subplot(*m,n,i*)

où

m : est le nombre de lignes (encarts verticaux);

n : est le nombre de colonnes (encarts horizontaux);

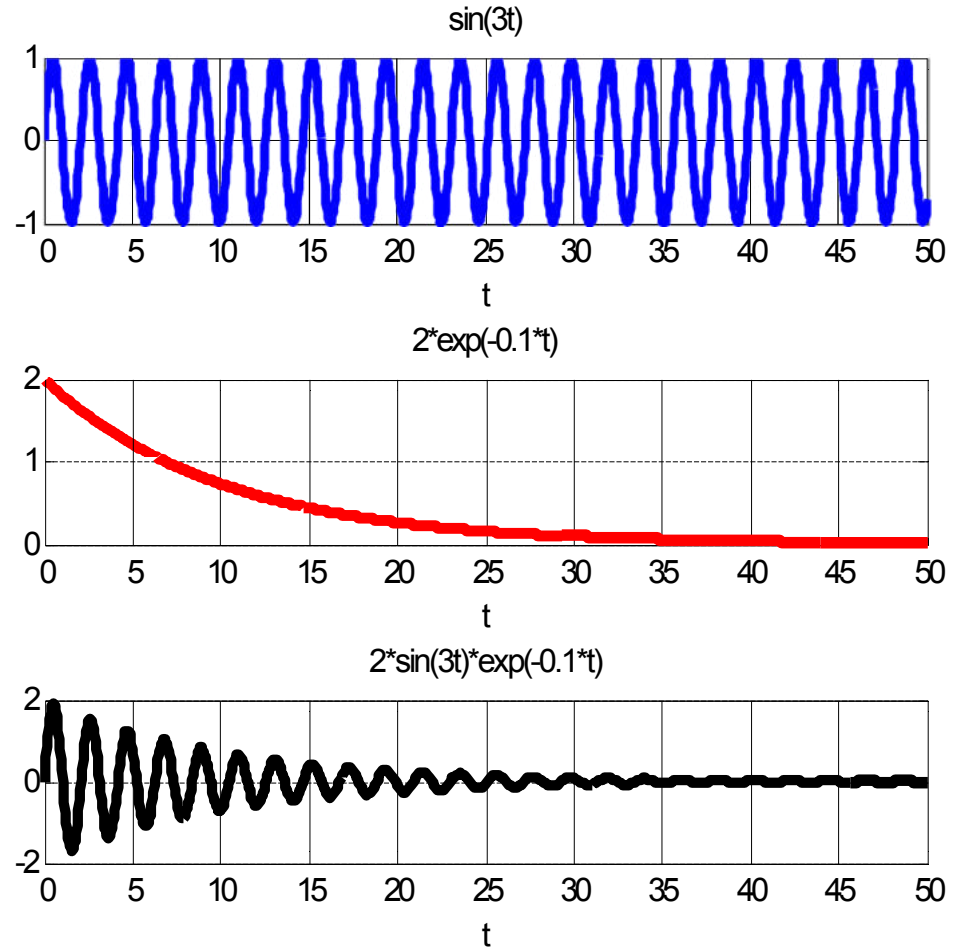
i : pour spécifier dans quelle sous-fenêtre doit s'effectuer l'affichage. L'ordre de numérotation des sous-figures est de gauche à droite et de haut en bas.



Représentation 1D (12/21)

➔ Exemple

```
t = 0 : 0.05 : 50;  
x = sin(3*t);  
y = 2*exp(-0.1*t);  
subplot(3,1,1)  
plot(t,x,'linewidth',3);  
xlabel('t');  
title('sin(3t)');  
grid on  
subplot(3,1,2)  
plot(t,y,'r','linewidth',3);  
xlabel('t');  
title('2*exp(-0.1*t)');  
grid on  
subplot(3,1,3)  
plot(t,x.*y,'k','linewidth',3);  
xlabel('t');  
title('2*sin(3t)*exp(-0.1*t)');  
grid on
```



Représentation 1D (13/1)

➔ Exemple

```
fe=1000;
te=1/fe;
t=-1:te:1;
x=cos(2*pi*10*t);
subplot(2,2,1);
plot(t,x);
xlabel('temps');
ylabel('x(t)');
title('cos(2.pi.f.t)');

% Transformée de Fourier
subplot(2,2,3);
f=linspace(-fe/2,fe/2,length(t));
Xf=fftshift(fft(x)/fe);
stem(f,abs(Xf));
axis([-20,20,0,1]);
grid on;
xlabel('fréquence');
ylabel('X(f)');
title('Transformée de Fourier de cos(2*pi*10*t)');
```

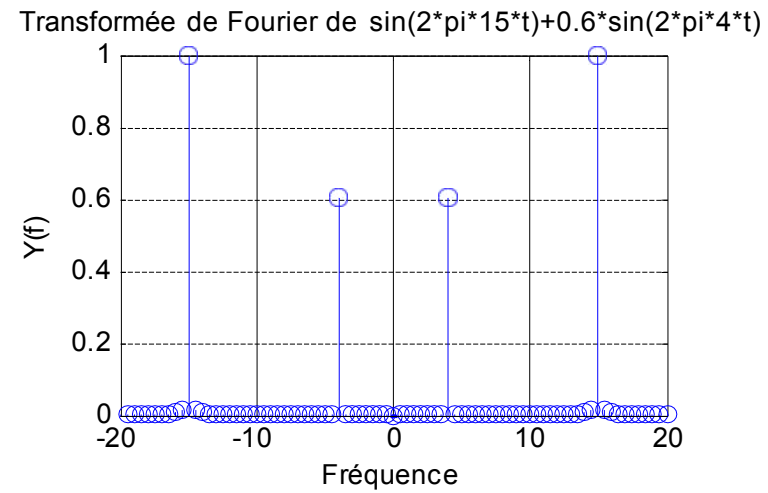
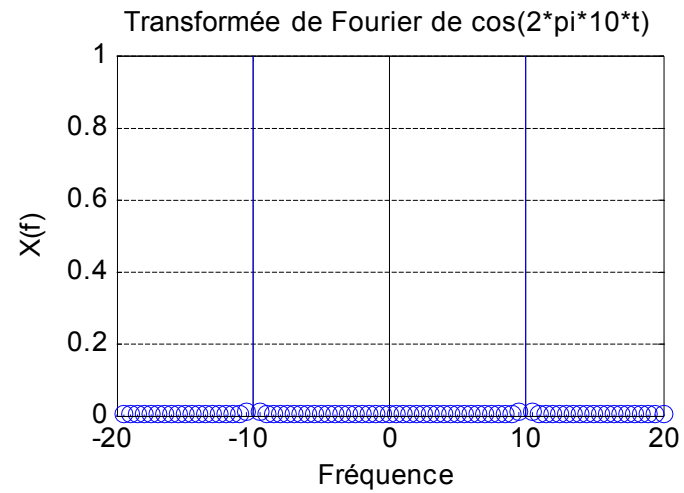
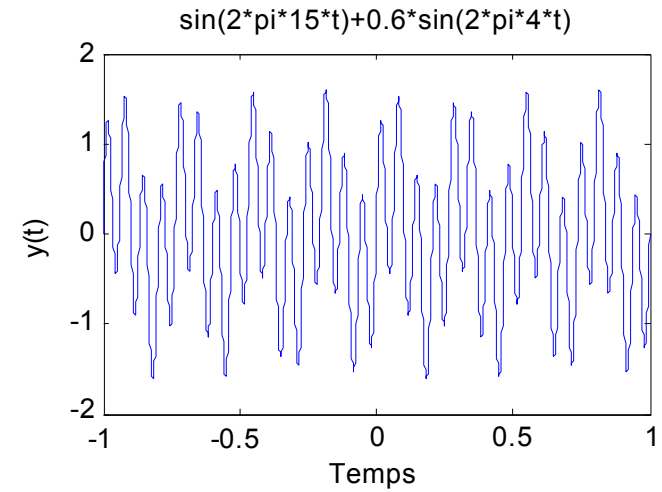
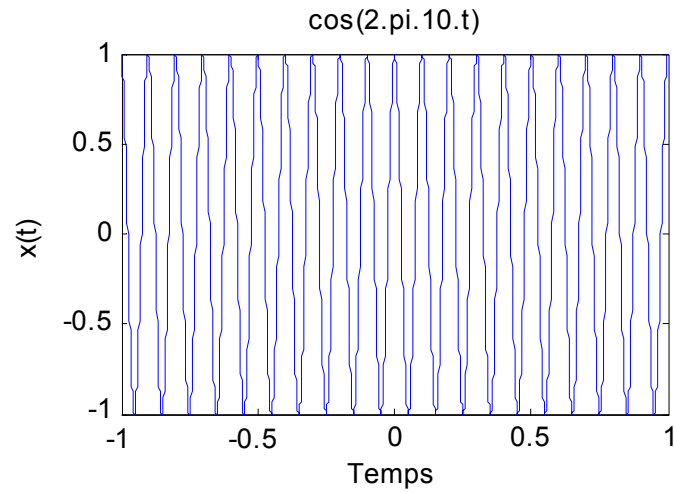
Représentation 1D (14/21)

```
y=sin(2*pi*15*t)+0.6*sin(2*pi*4*t);  
subplot(2,2,2);  
plot(t,y);  
xlabel('temps');  
ylabel('x(t)');  
title('sin(2*pi*15*t)+0.6*sin(2*pi*4*t)');
```

% Transformée de Fourier

```
subplot(2,2,4);  
f=linspace(-fe/2,fe/2,length(t));  
Yf=fftshift(fft(y)/fe);  
stem(f,abs(Yf));  
axis([-16,16,0,1]);  
grid on;  
xlabel('fréquence');  
ylabel('X(f)');  
title('Transformée de Fourier de sin(2*pi*15*t)+0.6*sin(2*pi*4*t)');
```

Représentation 1D (15/21)



Représentation 1D (16/21)

➡ La commande **fplot** permet elle aussi de tracer le graphe d'une fonction sur un intervalle donné. Sa syntaxe est :

fplot('nomf', [xmin , xmax])

où :

***nomf** est soit le nom d'une fonction MATLAB incorporée, soit une expression définissant une fonction de la variable x, soit le nom d'une macro (voir plus loin).

* $[x_{min}, x_{max}]$ est l'intervalle pour lequel est tracé le graphe de la fonction.

➡ Il est possible de tracer plusieurs fonctions sur la même figure. Il faut pour cela utiliser la commande **fplot** de la manière suivante:

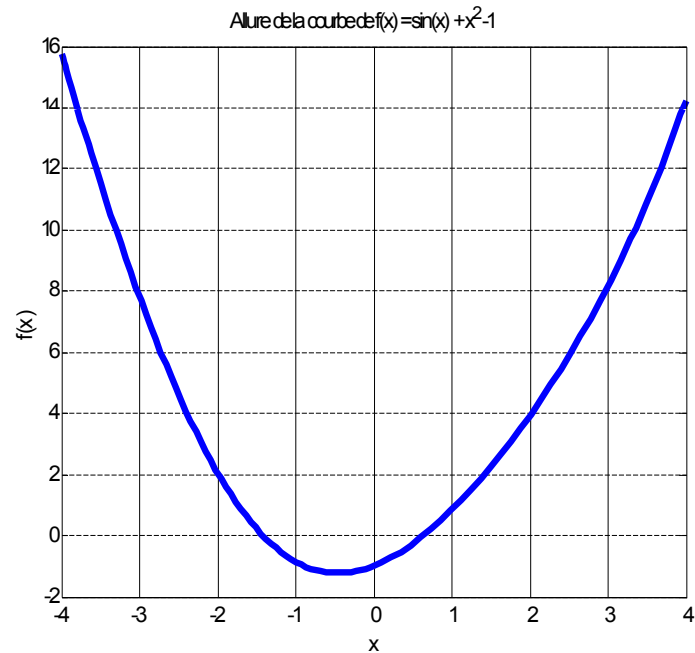
fplot('[nomf_1 , nomf_2 , nomf_3]', [xmin , xmax])

où **nomf_1**, **nomf_2**, **nomf_3** sont soit les noms des fonctions MATLAB incorporées, soit des expressions qui définissent une fonction de la variable x, soit des macros.

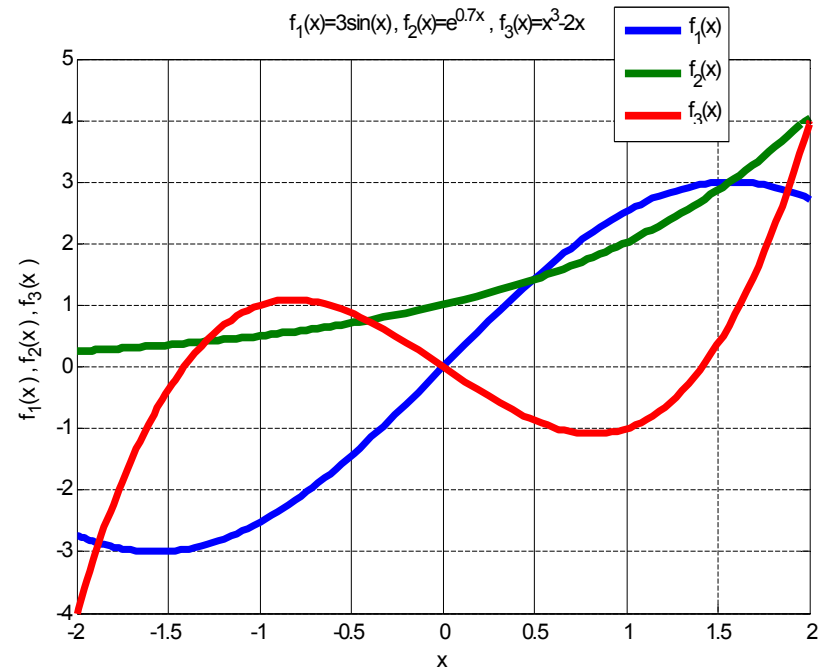
Représentation 1D (17/21)

→ Exemples

```
fplot('sin(x) + x^2-1',[-4 4])  
xlabel('x');  
ylabel('f(x)');  
title ('Allure de la courbe de f(x) = sin(x) + x^2-1');  
grid on
```



```
fplot('[3*sin(x) , exp(0.7*x) , x^3-2*x]', [-2 , 2])  
xlabel('x');  
ylabel('f_1(x) , f_2(x) , f_3(x)');  
title ('f_1(x)=3sin(x) , f_2(x)=e^{0.7x} , f_3(x)=x^3-2x');  
grid on  
legend('f_1(x)', 'f_2(x)', 'f_3(x)');
```



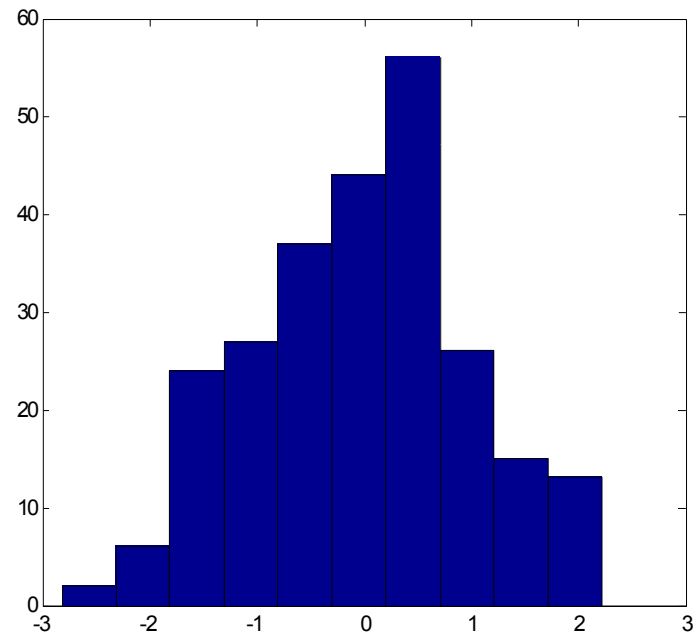
Représentation 1D (18/21)

- ➔ Autres types de graphique plus adaptés aux représentations des données de types listes (statistiques) : **hist**, **pie**, **bar**, **stem**, **boxplot**, ...
 - ➔ La fonction **hist** permet de tracer un histogramme d'un vecteur x , en répartissant ces valeurs (éléments du vecteur) en n classes. Sa syntaxe est **hist(x,n)**.
 - ➔ La commande **[N,X] = hist(x,n)** retourne dans N l'effectif de chacune des classes et dans X l'abscisse du centre de chaque classe.
 - ➔ La fonction **pie(x)** permet de dessiner un diagramme en cercle de valeurs de x normalisées.
- L'instruction **bar(x,y)** dessine un diagramme sous forme de barres des valeurs de y en fonction de celles de x .

Représentation 1D (19/21)

➔ Exemples

```
A = randn(1,250);  
figure (1)  
hist(A);  
[N,X] = hist(A);  
figure (2)  
pie(N)
```

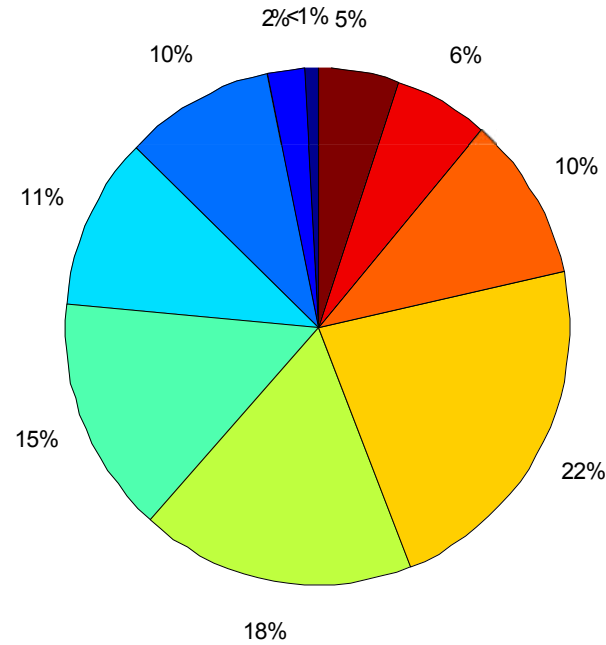


Après exécution, dans l'espace de commande :

```
>> N =  
     2     6    24    27    37    44    56    26    15    13
```

```
>> sum(N)
```

```
ans =  
    250
```

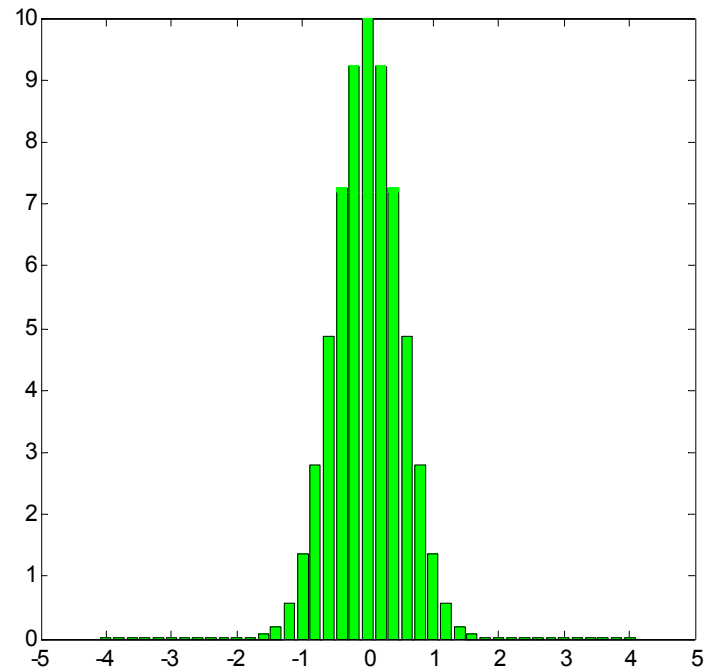


Représentation 1D (20/21)

→ Exemples

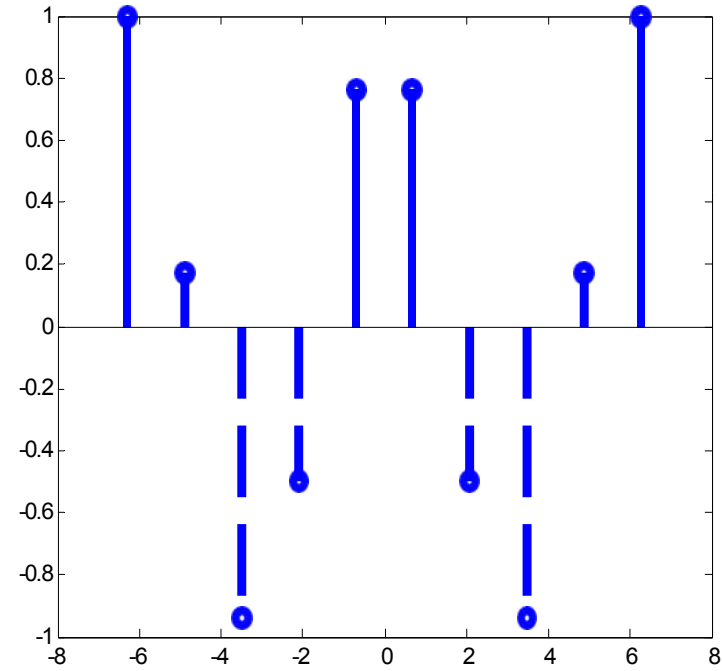
```
x = -4:0.2:4;
```

```
bar(x,10*exp(-2*x.^2),'g')
```



```
t = linspace(-2*pi,2*pi,10);
```

```
h = stem(t,cos(t),'--', 'linewidth',4);
```



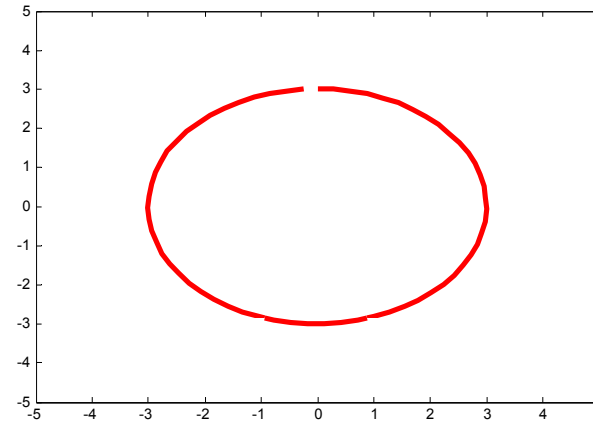
Représentation 2D & 3D (1/4)

➔ Cette partie ne sera pas trop détailler vu les chapitres qui restent à aborder (parties importantes du cours). On va se contenter de quelques exemples. Sachant que Matlab propose beaucoup de choses concernant les graphismes **2D**, **3D** et **l'animation**.

➔ Exemples

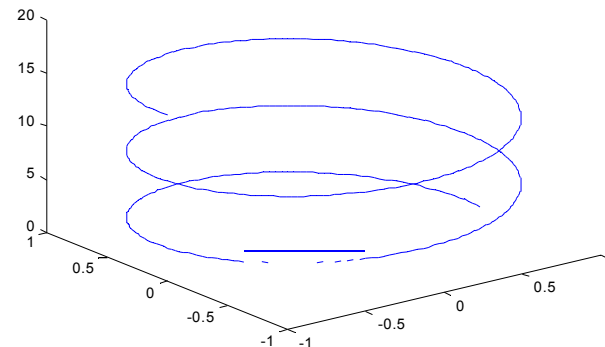
%Courbe paramétrique : le cercle

```
t = [0 :0.1: 2*pi];  
rayon = 3;  
X = rayon * sin(t);  
Y = rayon * cos(t);  
plot (X , Y , 'r','linewidth',3);  
axis([-5 5 -5 5]);
```



%Courbe paramétrique : le cercle

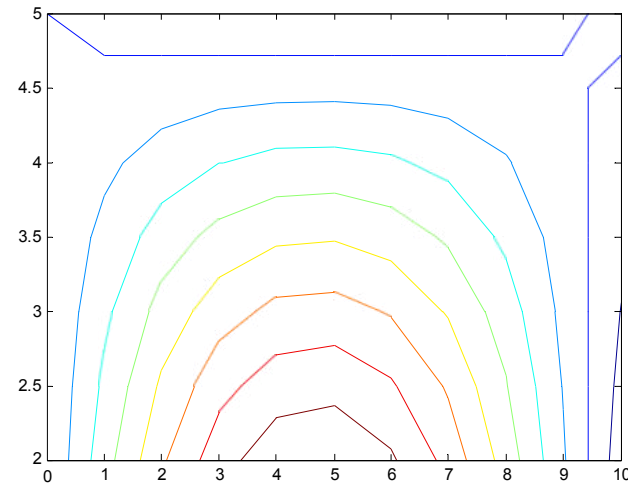
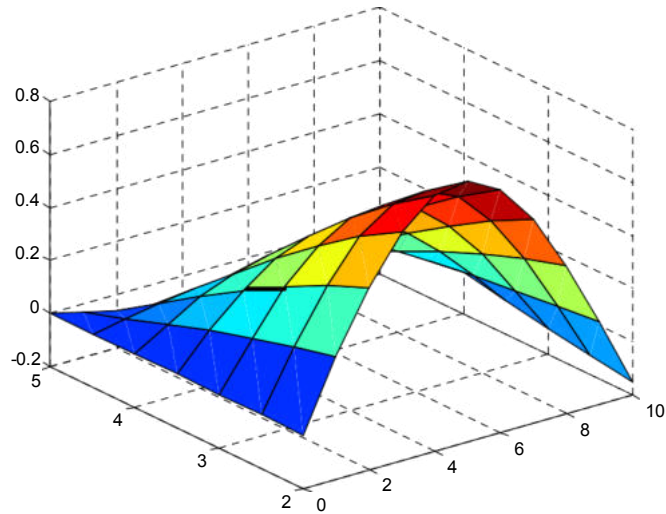
```
t = [0 :0.1: 2*pi];  
x = cos (t);  
y = sin (t);  
z = t;  
t = 0 : pi/100 : 5*pi;  
plot3 (cos(t),sin(t),t);
```



Représentation 2D & 3D (2/4)

➔ Exemple

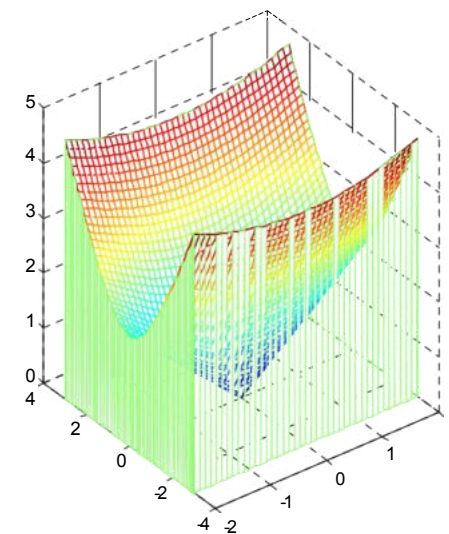
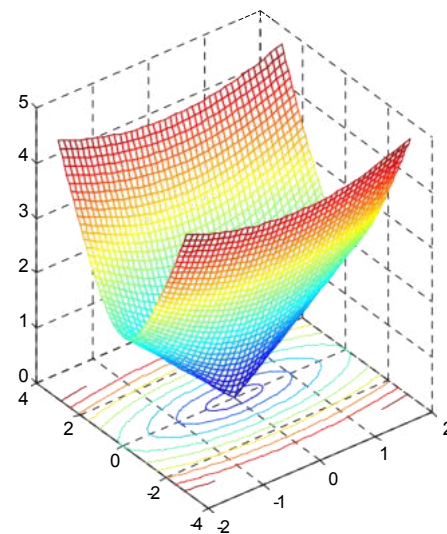
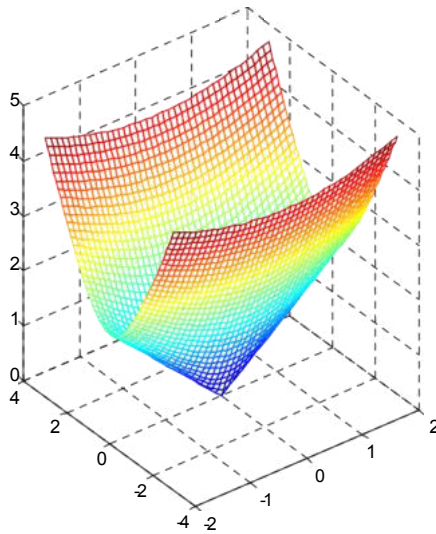
```
x=0:1:10;  
y=2:0.5:5;  
for k=1:length(x) % parcours de la grille, colonne après colonne  
    for l=1:length(y) % parcours de la grille, ligne après ligne  
        z1(l,k)= sin(x(k)/3)*cos(y(l)/3); % calcul de la matrice z, élément après élément  
    end  
end  
figure (1);  
surf(x,y,z1); % visualisation de la surface  
figure (2);  
contour(x,y,z1);
```



Représentation 2D & 3D (3/4)

➔ Exemple

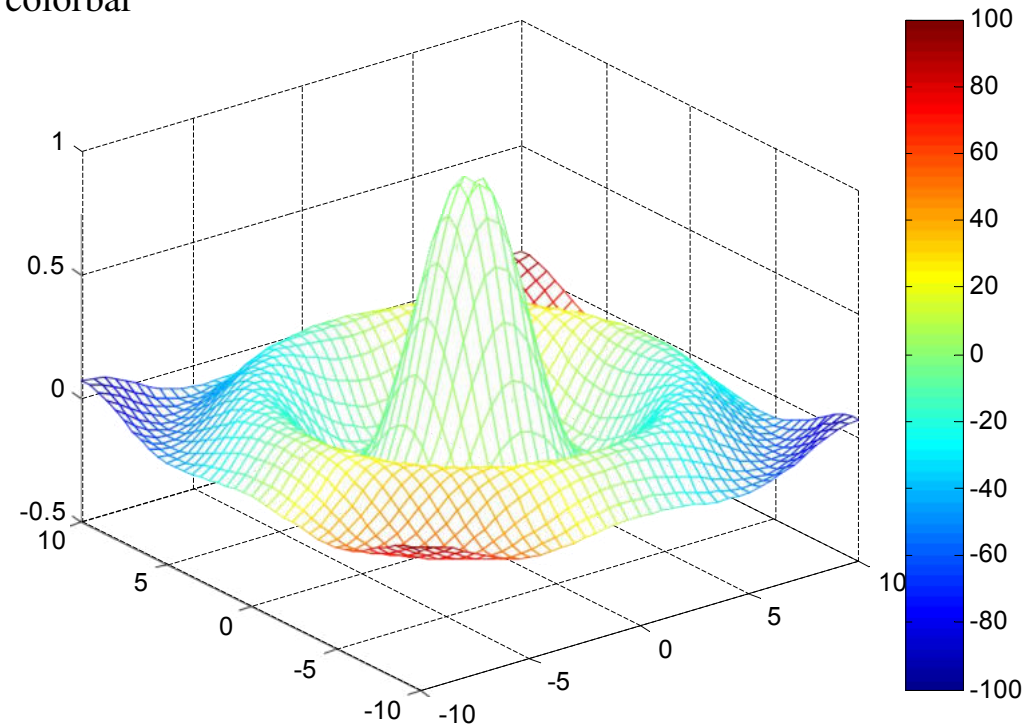
```
x = -2 : 0.1 : 2;  
y = -3 : 0.1 : 3;  
[X , Y] = meshgrid(x , y) ;  
Z = sqrt (X.^2 + 2.*Y.^2) ;  
figure (1)  
mesh (X,Y,Z) ;  
figure (2)  
meshc (X,Y,Z);  
figure (3)  
meshz (X,Y,Z);
```



Représentation 2D & 3D (4/4)

➔ Exemple

```
[X,Y] = meshgrid(-10:0.5:10);  
R = sqrt(X.^2 + Y.^2);  
Z = sin(R)./R;  
C = X.*Y;  
mesh(X,Y,Z,C)  
colorbar
```





MATLAB POUR L'INGÉNIEUR

CP-2

ENSAH-2020-2021

Partie 1

Pr. *Amina GHADBAN*

MATLAB POUR L'INGÉNIEUR

Chapitre 1

Manipulations des Polynômes

Partie 1

Prof. Amina GHADBAN

Polynômes (1/24)

➔ Un polynôme peut être représenté par un vecteur , selon les puissances décroissantes, dont le nombre d'éléments est égale au degré du polynôme + 1.

➔ $P(x) = x^5 + 2x^4 - 7x^3 + 12x^2 - 4x + 31$

>> $P = [1 \ 2 \ -7 \ 12 \ -4 \ 31]$

P =

$$1 \ 2 \ -7 \ 12 \ -4 \ 31$$

➔ $Q(x) = -15x^3 + 7x + 3$

>> $Q = [-15, 0, 7, 3]$

Q =

$$-15 \ 0 \ 7 \ 3$$

Polynômes (2/24)

➔ Évaluation d'un polynôme

polyval(P,x)

➔ Racines d'un polynôme

roots(P)

➔ Polynôme à partir des racines

poly([racine1, racine2,...])

➔ Dérivation d'un polynôme

polyder(P)

➔ Intégration d'un polynôme

polyint(P)

Polynômes (3/24)

➔ Addition de polynômes

$$P + Q$$

➔ Soustraction de polynômes

$$P - Q$$

➔ Produit de polynômes

$$\text{conv}(P, Q)$$

➔ Division de polynômes

$$[q, r] = \text{deconv}(P, Q)$$

➔ Décomposition en éléments simples

$$[r, p, k] = \text{residue}(P, Q)$$

Polynômes (4/24)

```
>> P = [1 -2 -3];
```

```
>> Q = [1 -3];
```

```
>> polyval(P,1) ans =
```

```
-4
```

```
% Cette fonction remplace le x  
du polynôme avec la valeur 1
```

```
>> polyval(Q,-3)
```

```
ans =
```

```
-6
```

```
>> P1 = polyder(P)
```

```
P1 =
```

```
2 -2
```

```
>> P2 = polyder(P1)
```

```
P2 =
```

```
2
```

```
>> polyval(P1,2)
```

```
ans =
```

```
2
```

```
>> polyval(P2,5)
```

```
ans =
```

```
2
```

```
>> polyval(P2,sqrt(3))
```

```
ans =
```

```
2
```

```
>> polyval(P1,0)
```

```
ans =
```

```
-2
```

```
>> polyval(P1,127)
```

```
ans =
```

```
252
```

Polynômes (5/24)

```
>> P = [1 -3 2];
```

```
>> Pd1 = polyder(P)
```

```
Pd1 =
```

```
2 -3
```

```
>> Pd2 = polyder(Pd1)
```

```
Pd2 =
```

```
2
```

```
>> Pi1 = polyint(P)
```

```
Pi1 =
```

```
0.3333 -1.5000 2.0000 0
```

```
>> polyval(Pi1,0)
```

```
ans =
```

```
0
```

```
>> polyval(Pi1,1)
```

```
ans =
```

```
0.8333
```

```
>> polyval(Pi1,-2)
```

```
ans =
```

```
-12.6667
```

```
>> Q = [2 3 -4];
```

```
>> Qi1 = polyint(Q)
```

```
Qi1 =
```

```
0.6667 1.5000 -4.0000 0
```

```
>> polyder(Qi1)
```

```
ans =
```

```
2 3 -4
```

```
>> polyder(polyder(Qi1))
```

```
ans =
```

```
4 3
```

Polynômes (6/24)

```
>> roots(P)
```

```
ans =
```

```
3.0000
```

```
-1.0000
```

```
>> roots(Q)
```

```
ans =
```

```
3
```

```
>> R = poly([-4 7])
```

```
R =
```

```
1 -3 -28
```

```
>> P + Q
```

??? Error using ==> plus
Matrix dimensions must agree.

```
>> P = [1 -2 -3];
```

```
>> Q = [0 1 -3];
```

```
>> P + Q
```

```
ans =
```

```
1 -1 -6
```

```
>> P - Q
```

```
ans =
```

```
1 -3 0
```

```
>> P - 2*R
```

```
ans =
```

```
-1 4 53
```

➔ Représentation graphique d'un polynôme

```
P = [1 -3 -1 -5 -1 8];
```

```
x = -2:0.1:4;
```

```
y = polyval(P,x);
```

```
plot(x,y,'k','linewidth',4);
```

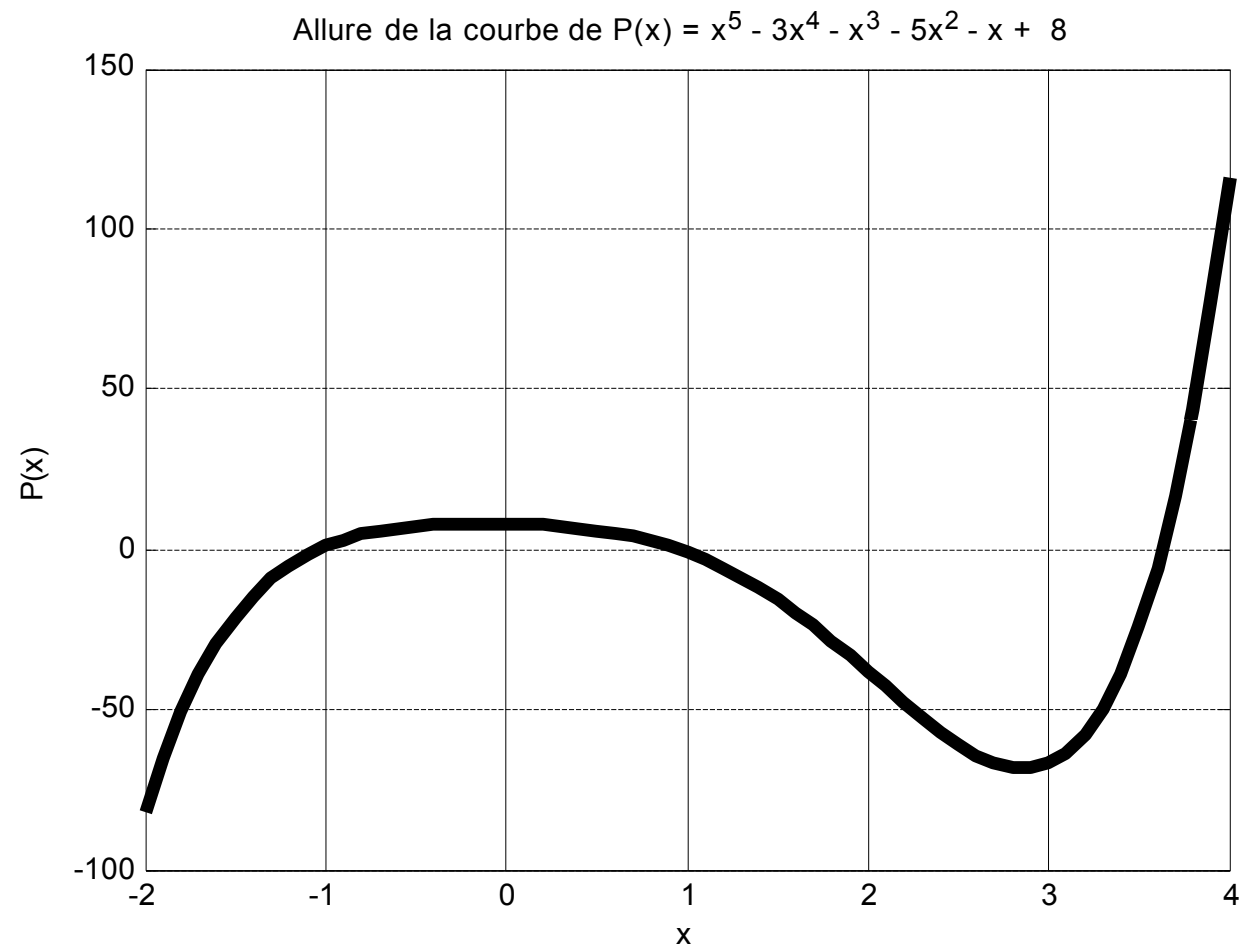
```
grid on
```

```
xlabel('x');
```

```
ylabel('P(x)');
```

```
title('Allure de la courbe de P(x) = x^5 -3x^4 - x^3 -5x^2 - x +8');
```

➔ Représentation graphique d'un polynôme

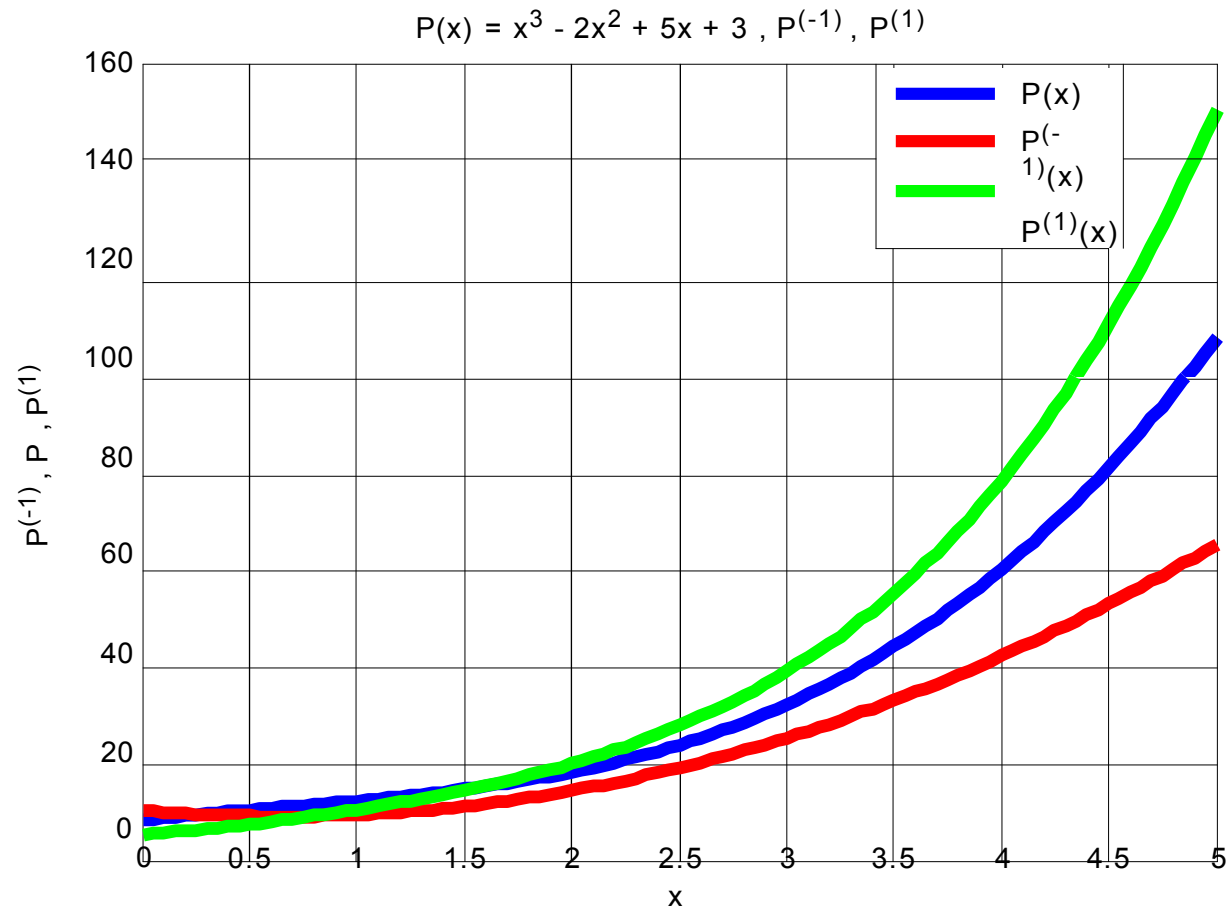


➔ Représentation graphique d'un polynôme

```
P = [1 -2 5 3];
Pd1 = polyder(P);
Pi1 = polyint(P); x = 0:0.05:5;
U = polyval(P,x);
V = polyval(Pd1,x);
W = polyval(Pi1,x);
plot(x,U,'linewidth',4); % cette fonction trace le polynôme P en
fonction de chaque valeur x
hold on
plot(x,V,'r','linewidth',4);
hold on
plot(x,W,'g','linewidth',4);
xlabel('x');
grid on
legend('P(x)', 'P^{(-1)}(x)', 'P^{(1)}(x)')
ylabel('P^{(-1)} , P , P^{(1)}');
title('P(x) = x^3 - 2x^2 + 5x + 3 , P^{(-1)} , P^{(1)}');
```

Polynômes (11/24)

➔ Représentation graphique d'un polynôme





Université Abdelmalek Essaadi
École Nationale des Sciences Appliquées Al
Hoceima



Cours d' *Informatique3*: **MATLAB**

MATLAB POUR L'INGÉNIEUR

CP-2

ENSAH -2020-2021

Fin de la Partie 1

Pr. Amina GHADBAN