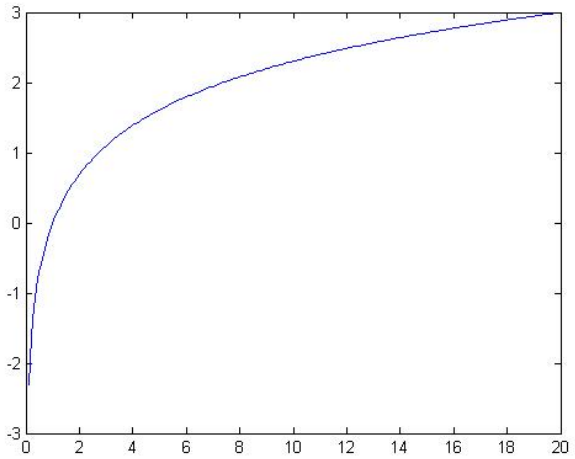
2D plotting commands (help graph2d)

plot(x,y) genera un grafico lineare dei valori di x (ascissa) e y (ordinata):

```
>>  x = 0.1:0.1:20;
>>  y = log(x);
>>  plot(x,y)
```



Plotting linestyles

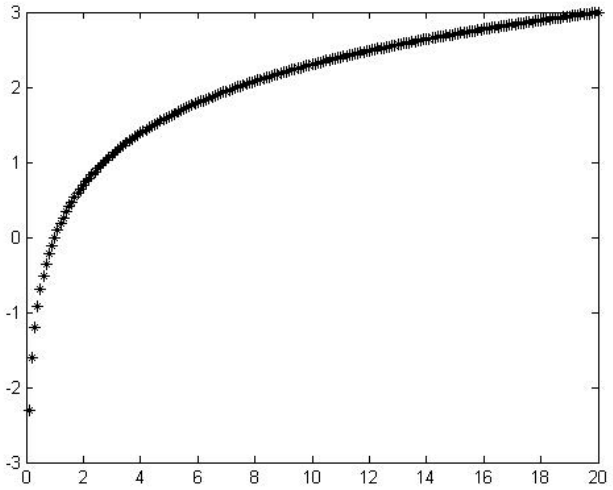
(help plot)

Plot(x,y,'color marker')

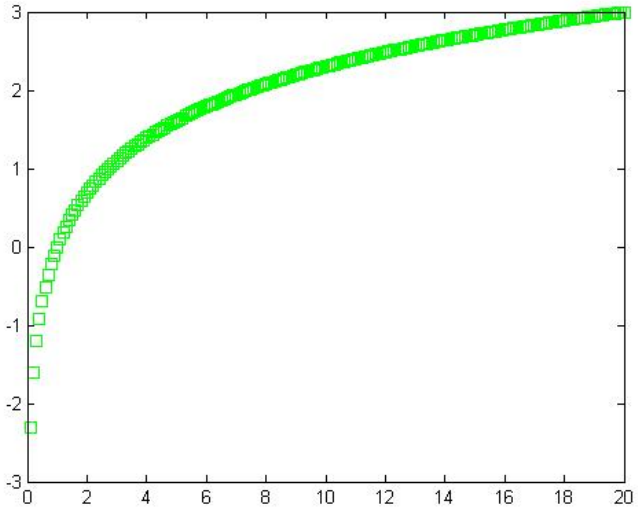
Symbol	Color	Symbol	Marker
y	yellow	.	●
m	magenta	o	○
c	cyan	x	×
r	red	+	+
g	green	*	*
b	blue	s	□
w	white	d	◇
k	black	v	▽
		^	△
		<	◁
		>	▷
		p	⋆
		h	hexagram

Examples

>> plot(x,y,'k*')



>> plot(x,y,'gs')



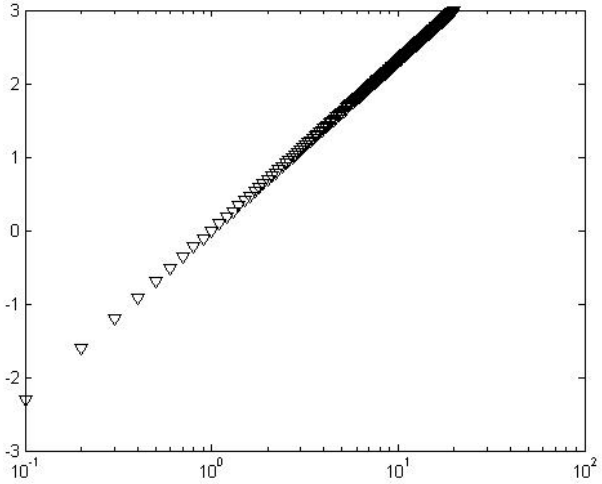
Linear and logarithmic scales

`plot(x,y)`	Generates a linear plot of the values of `x` (horizontal axis) and `y` (vertical axis).
`semilogx(x,y)`	Generates a plot of the values of `x` and `y` using a logarithmic scale for `x` and a linear scale for `y`.
`semilogy(x,y)`	Generates a plot of the values of `x` and `y` using a linear scale for `x` and a logarithmic scale for `y`.
`loglog(x,y)`	Generates a plot of the values of `x` and `y` using logarithmic scales for both `x` and `y`.

Note that the logarithm of a negative value or of zero does not exist and if the data contains negative or zero values, a warning message will be printed by MATLAB informing you that these data points have been omitted from the data plotted. Examples are shown in Figure 5.4 below.

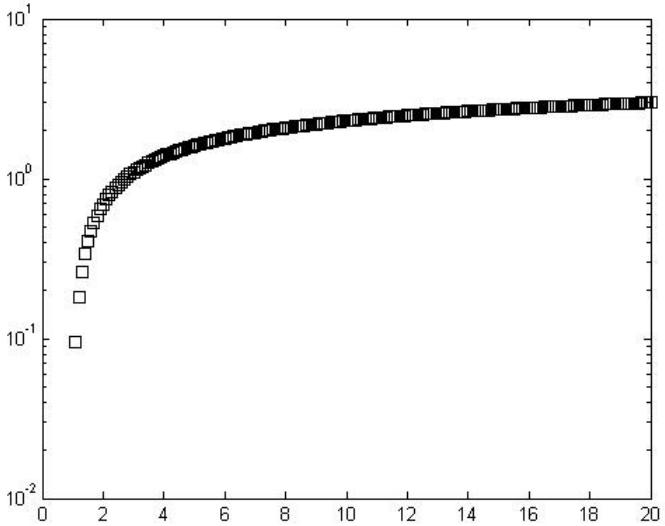
examples

>> semilogx(x,y,'kv')



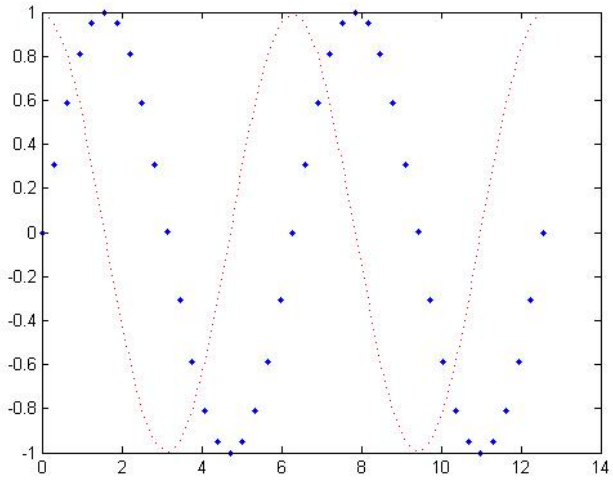
>> semilogy(x,y,'ks')

Warning: Negative data ignored

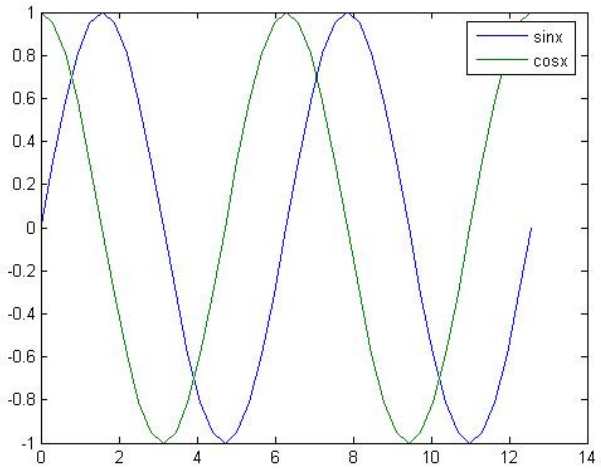


Multiple curves

```
>> x = (0:.1:4)*pi;
>>y1 = sin(x);
>>y2 = cos(x);
```



Method A: `>> plot(x,y1,'b.', x,y2,'r:')`



Method B:

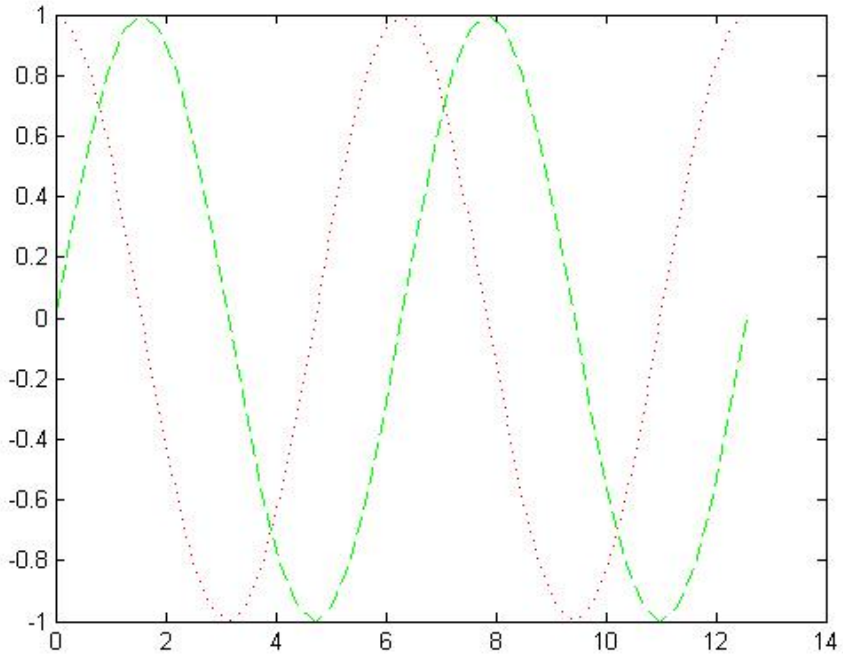
```
>> Y = [y1' y2']    % creo una matrice con gli elementi sulle
                    %colonne contenenti i valori delle ordinate dei grafici

>> plot(x,Y),...    % le curve non si possono differenziare con i simboli

Legend('sinx', 'cosx')
```

Method C:

```
>>  plot(x,y1,'g--')
>> hold on
>> plot(x,y2,'r:')
```



Customizing plot axes

The `axis` command provides control over the scaling and appearance of both the horizontal and vertical axes of a plot. This command has many features, so only the most useful will be discussed here. For more complete information, refer to on-line help. The primary features are given in the following table

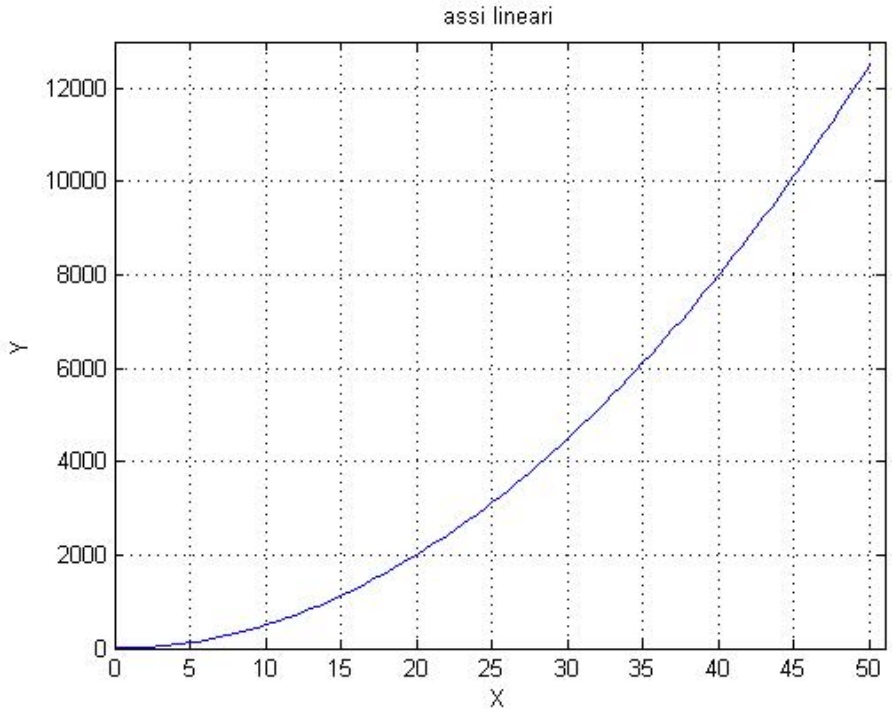
Command	Description
`axis([xmin xmax ymin ymax])`	Define minimum and maximum values of the axes
`axis square`	Produce a square plot instead of rectangular
`axis equal`	Equal scaling factors for both axes
`axis normal`	Turn off axis square, equal
`axis(auto)`	Return the axis to automatic defaults
`axis off`	Turn off axis background, labeling, grid, box, and tick marks. Leave the title and any labels placed by the `text` and `gtext` commands
`axis on`	Turn on axis background, labeling, tick marks, and, if they are enabled, box and grid

Plot grids, Axes box, Labels

Command	Description
`grid on`	Adds dashed grid lines at the tick marks
`grid off`	Removes grid lines (default)
`grid`	Toggles grid status (off to on, or on to off)
`box on`	Adds axes box, consisting of boundary lines and tick marks on top and right of plot
`box off`	Removes axes box (default)
`box`	Toggles box status
`title('text')`	Labels top of plot with text in quotes
`xlabel('text')`	Labels horizontal (x) axis with text in quotes
`ylabel('text')`	Labels vertical (y) axis with text in quotes
`text(x,y,'text')`	Adds text in quotes to location (x,y) on the current axes, where (x,y) is in units from the current plot
`gtext('text')`	Place text in quotes with mouse: displays the plot window, puts up a cross-hair to be positioned with the mouse, and write the text onto the plot at the selected position when the left mouse button or any keyboard key is pressed

example

```
% Esempio di grafico
X = 0:0.5:50;
Y = 5*X.^2;
figure, plot(X,Y), axis([0 51 0 13000]), title('assi lineari'),...
    xlabel('X'), ylabel('Y'), grid;
```



Multiple figures

- Quando si fa la chiamata di plot(…) la prima volta, si apre una nuova finestra: figure 1.
- Ad ogni chiamata successiva di plot, il nuovo grafico sovrascrive i precedenti, a meno che non si usa il comando 'hold on', sempre nella finestra con nome 'figure 1'.
- Per aprire una nuova finestra, occorre chiamare la funzione: figure(n), con 'n', numero della nuova finestra o, semplicemente 'figure' :

```
>> figure(2)     % aperta la figura numero 2, che diventa attiva
>> plot(x,y)     % il grafico è ora mostrato nella figura 2 (figura attiva)
```

Close multiple figures

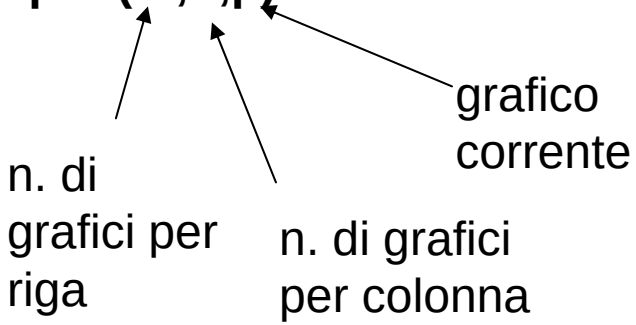
Le figure si possono chiudere:

- con il mouse , sulla X in alto a ds della finestra
- con il comando 'close(n)' dove 'n' è il numero della figura da chiudere
- con il comando 'close all' per chiudere TUTTE le finestre contenenti figure

Subplots

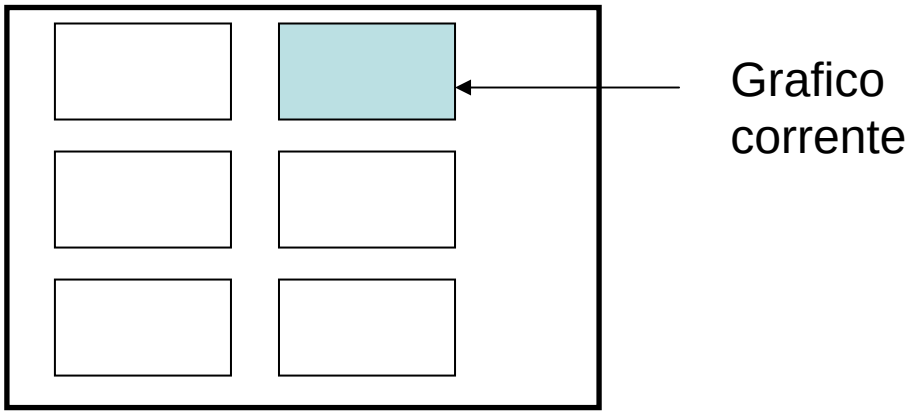
Più grafici nella stessa figura

subplot(m,n,p)



>> subplot(3,2,2)

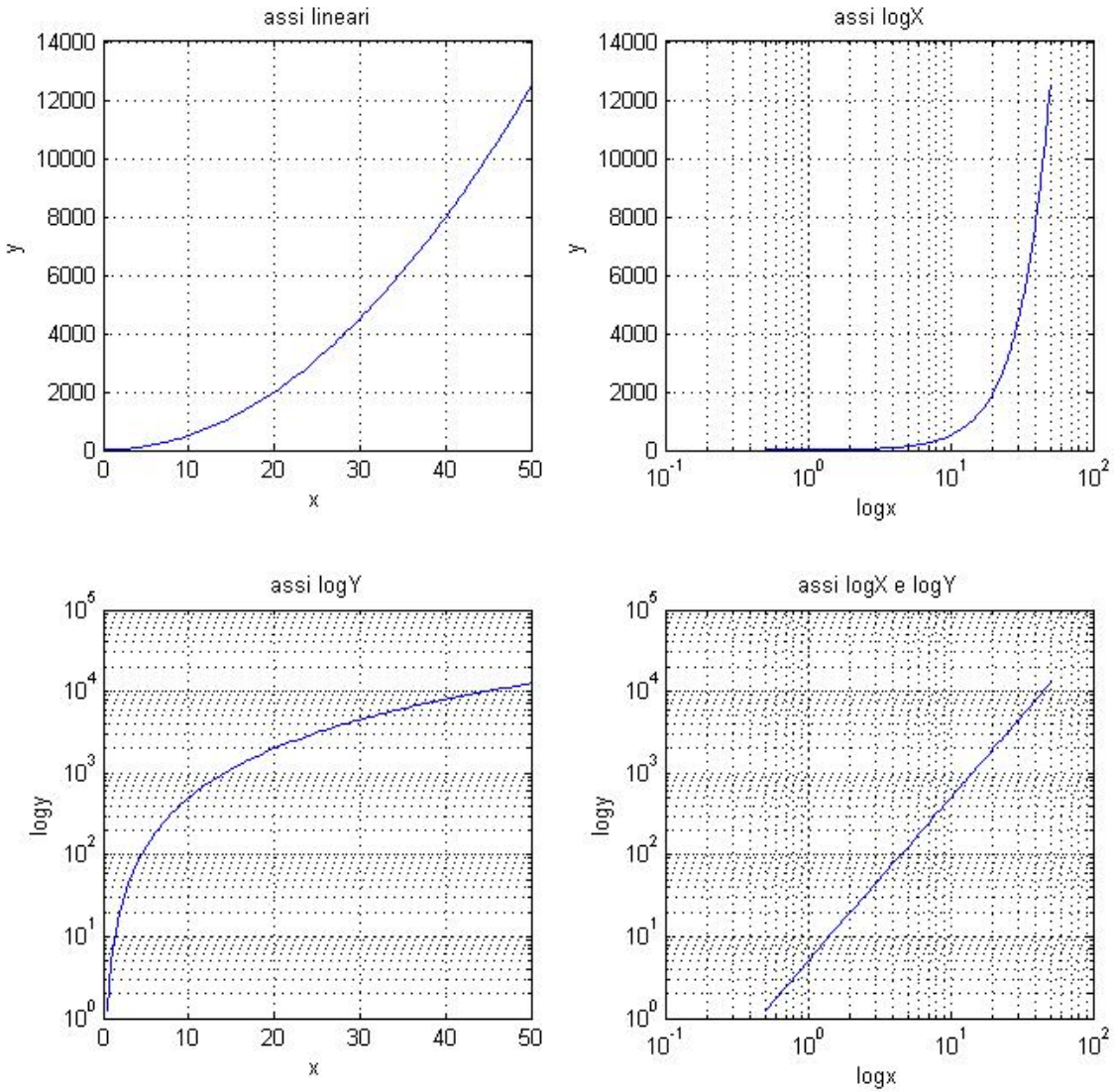
>> plot(x,y)



example

```
% Esempio di subplot e assi logaritmici
X = 0:0.5:50;
Y = 5*X.^2;
figure, subplot(2,2,1),plot(X,Y), title('assi lineari'),...
   xlabel('x'), ylabel('y'), grid,...
   subplot(2,2,2),semilogx(X,Y), title('assi logX'),...
   xlabel('logx'), ylabel('y'), grid,...
   subplot(2,2,3),semilogy(X,Y), title('assi logY'),...
   xlabel('x'), ylabel('logy'), grid,...
   subplot(2,2,4),loglog(X,Y), title('assi logX e logY'),...
   xlabel('logx'), ylabel('logy'), grid
```

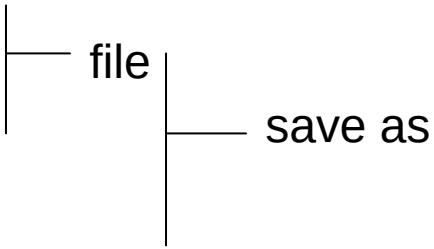
M-file :esempipograficolog


assi lineari
y
x
assi logX
y
logx
assi logY
logy
x
assi logX e logY
logy
logx

Saving and printing figures

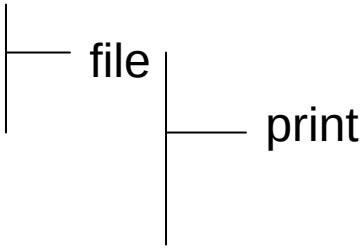
Salvare una figura:

nella finestra attiva:



stampare una figura:

nella finestra attiva:



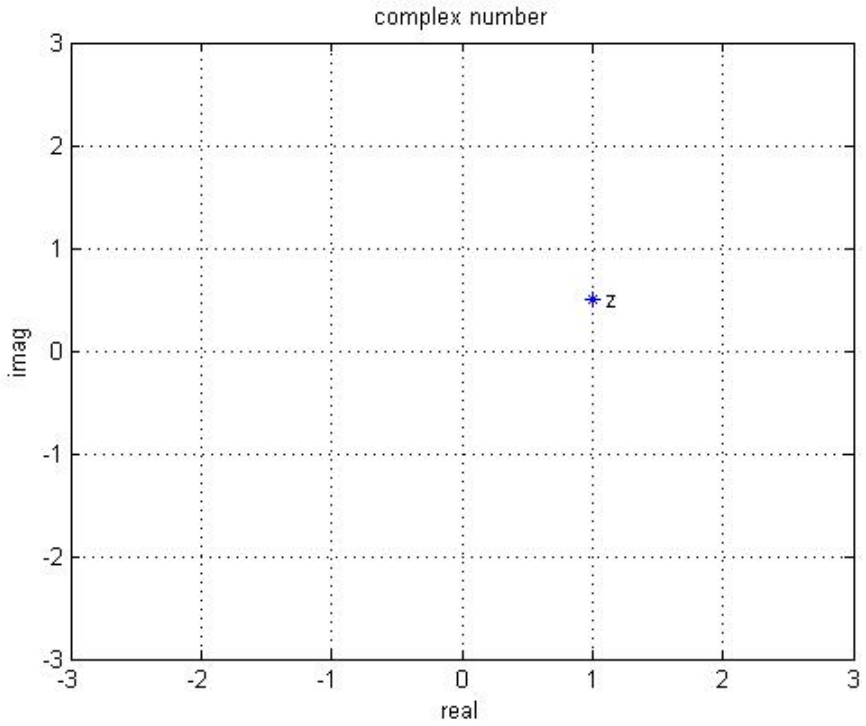
Plotting complex data

plot:

```
>> z = 1 + .5j
>> plot(z)
```

o, meglio:

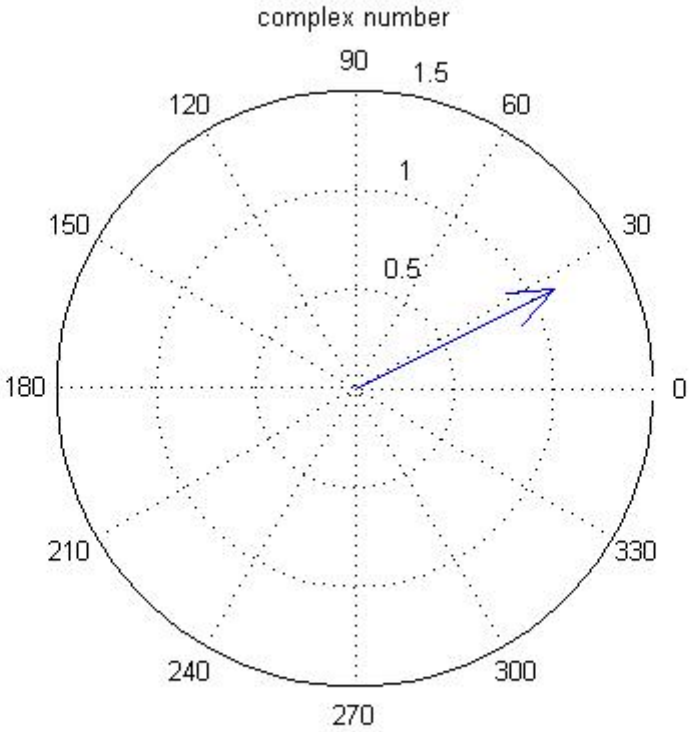
```
>> plot(z,'b*'), axis([-3 3 -3 3], grid on,…
text(real(z)+.1,imag(z),'z'),xlabel('real'),ylabel('imag'),…
title('complex number')
```



Plotting complex data

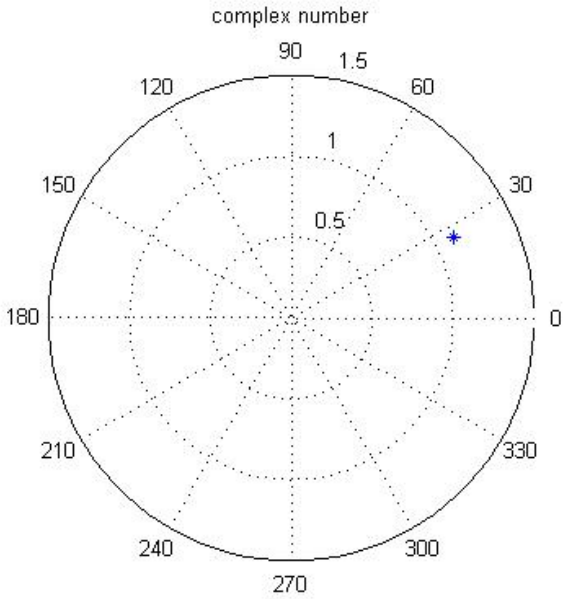
compass:

```
>> z = 1 + .5j
>>compass(z), title('complex number')
```



Plotting complex data **polar:**

```
>> z = 1 + .5j;
>> teta = angle(z);
>> r = abs(z);
>> polar(teta,r,'b*'), title('complex number')
```



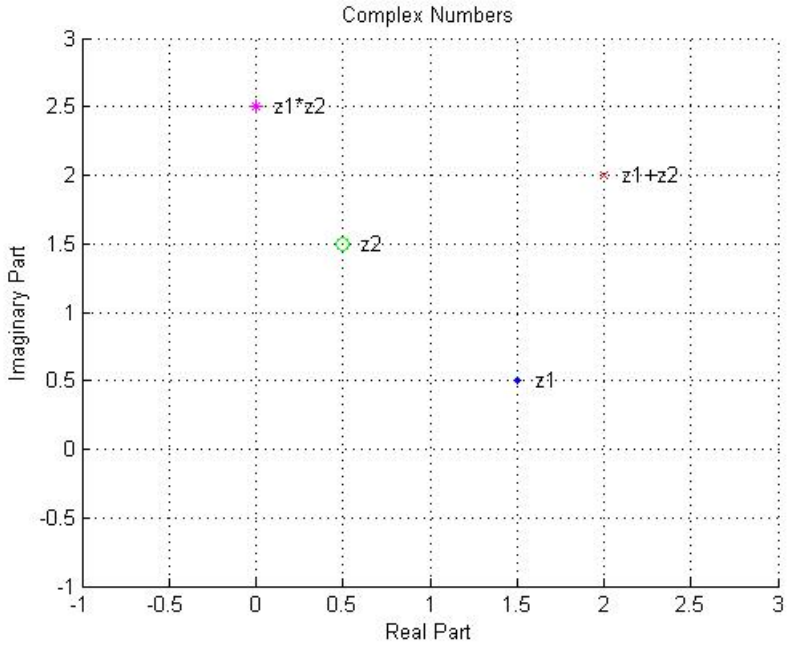
example

```
%esempio di grafici con numeri complessi

z1 = 1.5 + 0.5j;
z2 = 0.5 + 1.5j;
z3 = z1 + z2;
z4 = z1 * z2;

% grafico su assi rettangolari
figure,  axis([-1 3 -1 3]), grid on, hold on;
plot(z1,'b.'),plot(z2,'go'),plot(z3,'rx'),plot(z4,'m*');
text(real(z1)+0.1,imag(z1),'z1'),text(real(z2)+0.1,imag(z2),'z2'),...
    text(real(z3)+0.1,imag(z3),'z1+z2'),text(real(z4)+0.1,imag(z4),'z1*z2');
xlabel('Real Part'),ylabel('Imaginary Part'),title('Complex Numbers');
pause;

% grafico su assi polari
figure, compass(z1), compass(z2), compass(z3),...
    compass(z4);
```



% coordinate rettangolari

% coordinate polari