International Finance and Economics
Dept. of Economics and Law

Graphs of 1D and 2D functions

Prof. Elisabetta Michetti
Prof. Mauro Maria Baldi

GRAPH OF $z = f(x)$

Consider a function of one real variable $f: A \subseteq \mathbb{R} \to \mathbb{R}$.

We want to depict its graph $\{(x, f(x)) : x \in A\}$ (or plot).

To the scope there exist **two different ways** in MatLab:

1) PUNCTUAL DEFINITION

2) ANONYMOUS FUNCTION

1) Graph with punctual definition

The steps are the following

a. **Define a vector (say x) containing a reasonably high number of points in the interval of values to be considered for the independent variable x. This can be done through the a:step:b command or the linspace command**

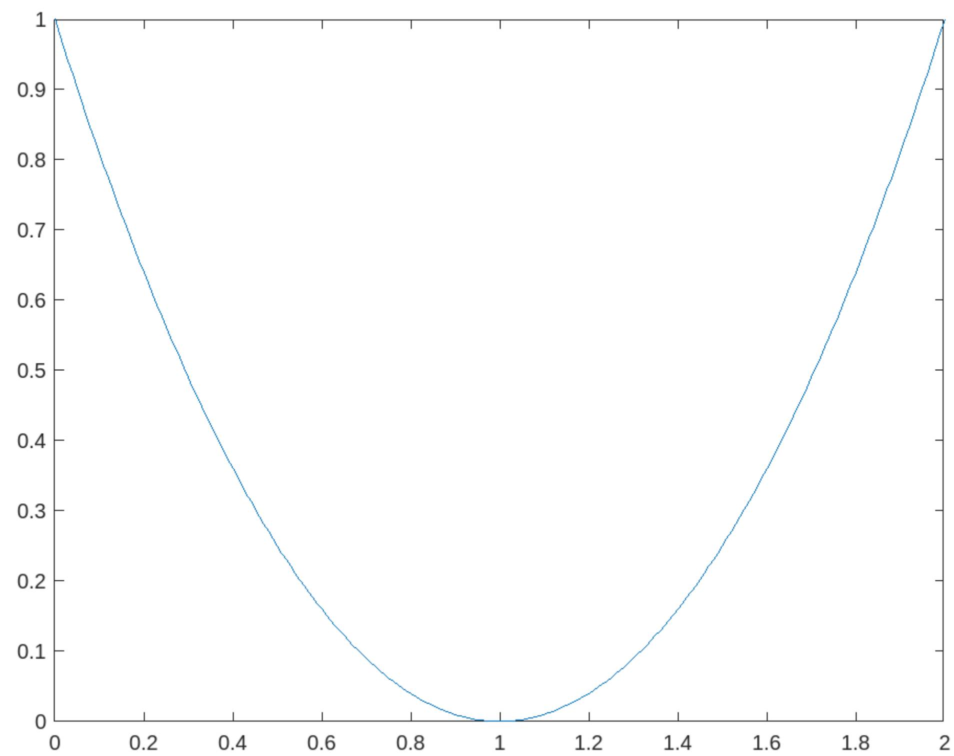
b. **Compute a corresponding vector (say y) obtained applying function f to the previous vector x**

c. **Use the command plot(x, y)**

EX 1

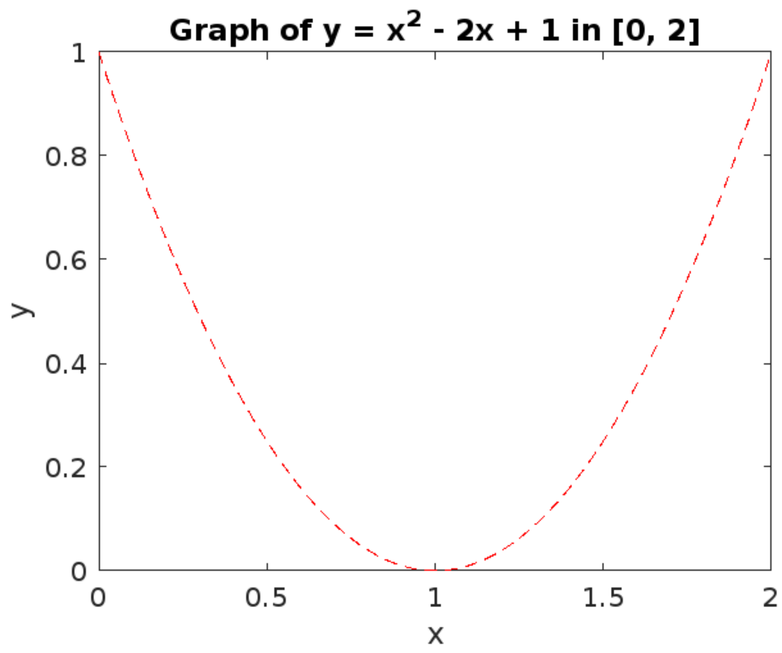
Depict the graph of $y = x^2 - 2x + 1$ in the interval [0, 2]

```
x = 0:.01:2;
y = x.^2 – 2*x + 1;
plot(x, y)
```



We can improve our graph putting labels on the axes and a title. This time we use the linspace command and we plot the graph in red using a dashed line.

```
x = linspace(0, 2, 1000);
y = x.^2 - 2*x + 1;
plot(x, y, "r--")
xlabel('x')
ylabel('y')
title('Graph of y = x^2 – 2x + 1 in [0, 2]')
```



An important remark

If we try the following piece of code in a script (or in the command line), we lose the initial figure.

```
x = 0:.01:2;
y = sin(x);
plot(x, y)

x = linspace(0, 2, 1000);
y = x.^2 - 2*x + 1;
plot(x, y, "r--")
xlabel('x')
ylabel('y')
title('Graph of y = x^2 – 2x + 1 in [0, 2]')
```

An important remark (continued)

In order to have both the figure available, we use the command **figure** for each plot. Moreover, we can also begin the script with some optional commands to clean the screen (**clc**) to delete all the previous variables (**clear** or **clear all**) and to close all the previous figures (**close all**)

```
clc
clear all
close all

figure
x = 0:.01:2;
y = sin(x);
plot(x, y)


figure
x = linspace(0, 2, 1000);
y = x.^2 - 2*x + 1;
plot(x, y, "r--")
xlabel('x')
ylabel('y')
title('Graph of y = x^2 – 2x + 1 in [0, 2]')
```

An important remark (continued)

We can also create a handle to the figure and use it, for instance, to save our figure.

```
clc
clear all
close all

h1 = figure;
x = 0:.01:2;
y = sin(x);
plot(x, y)
saveas(h1, "figure1.png")

h2 = figure;
x = linspace(0, 2, 1000);
y = x.^2 - 2*x + 1;
plot(x, y, "r--")
xlabel('x')
ylabel('y')
title('Graph of y = x^2 – 2x + 1 in [0, 2]')
saveas(h2, "figure2.png")
```

EX 2

Depict the graph of $y = x^2$ and $y = \sqrt{x^2 - 1}$, in the interval [-2, 2].

According to the previous example, we may try with the following code (we omit henceforth the clc, clear all and close all for reasons of space):

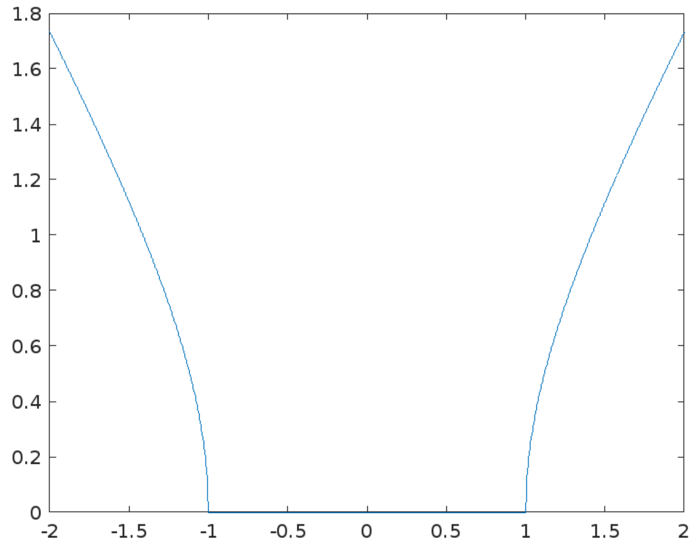
```
x = linspace(-2, 2, 1000);
y = x.^2;
plot(x, y)
y = sqrt(x.^2 - 1);
plot(x, y)
```

Unfortunately, we don’t get the desired results because:

- We can see only the last plot
- We get the following warning:

Warning: Imaginary parts of complex X and/or Y arguments ignored.
> In **script_2023_03_13** (line 10)

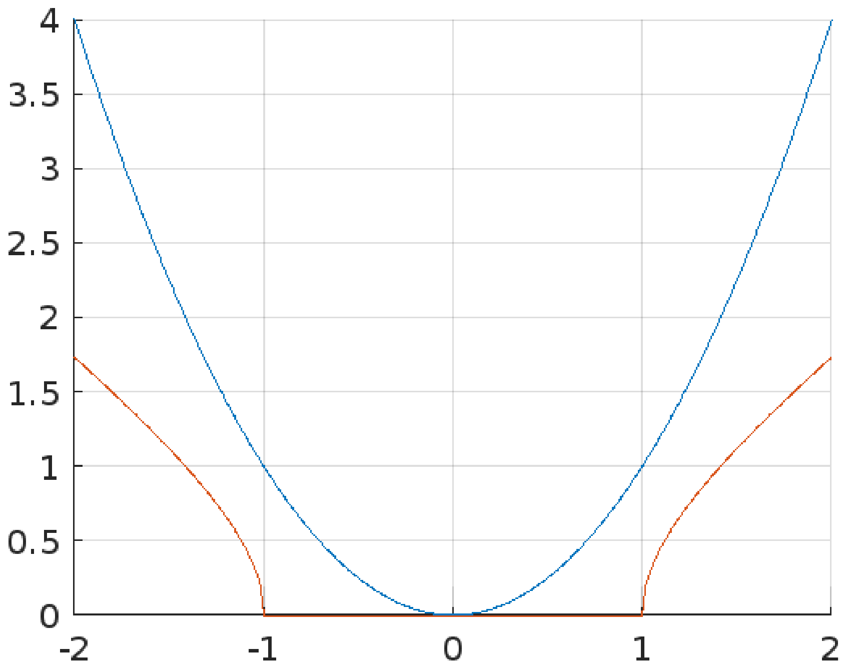
- The function is set to 0 between -1 and 1



We can solve the first problem in two ways: using the **hold** (or **hold on**) command or using the **plot** command with multiple arguments.

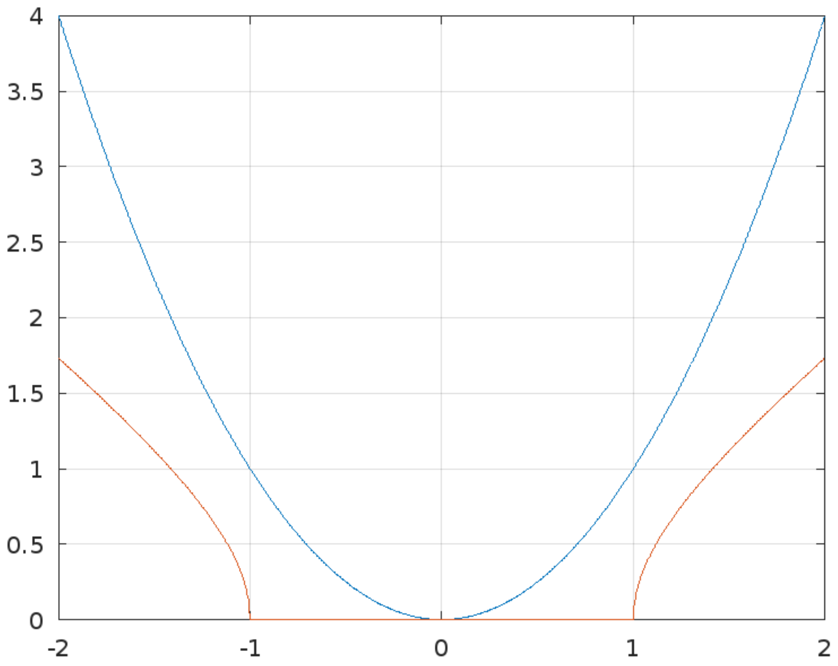
Let's see how the first solution works. We can also put a grid with the **grid** (or **grid on**) command.

```
figure
hold on
x = linspace(-2, 2, 1000);
y = x.^2;
plot(x, y)
y = sqrt(x.^2 - 1);
plot(x, y)
grid
```



We can achieve the same results using the **plot** command with multiple arguments.

```
figure
x = linspace(-2, 2, 1000);
y1 = x.^2;
y2 = sqrt(x.^2 - 1);
plot(x, y1, x, y2)
grid
% (optional) gcf: current figure handle
saveas(gcf, "figure5.png")
```



2) Graph with anonymous functions

We have fixed the problem of showing more than one graph in the same figure but we still have the problem of the warning and the function set to 0 in a range outside of the domain. In some contexts the previous solution is acceptable but not from a pure mathematical point of view.

The second approach addresses this problem but first we need to introduce the concept of anonymous functions. Let's consider again the function $f(x) = x^2$. We know that, for example,

$$f(0) = 0^2 = 0$$
$$f(-1) = (-1)^2 = 1$$

$$f(2) = 2^2 = 4$$

We can do the same in Matlab executing the following code in the command window:

```
f = @(x) x.^2;
f(0)

ans =

0


f(-1)

ans =

1


f(2)

ans =

4


f([-1 0 2])

ans =

1 0 4
```

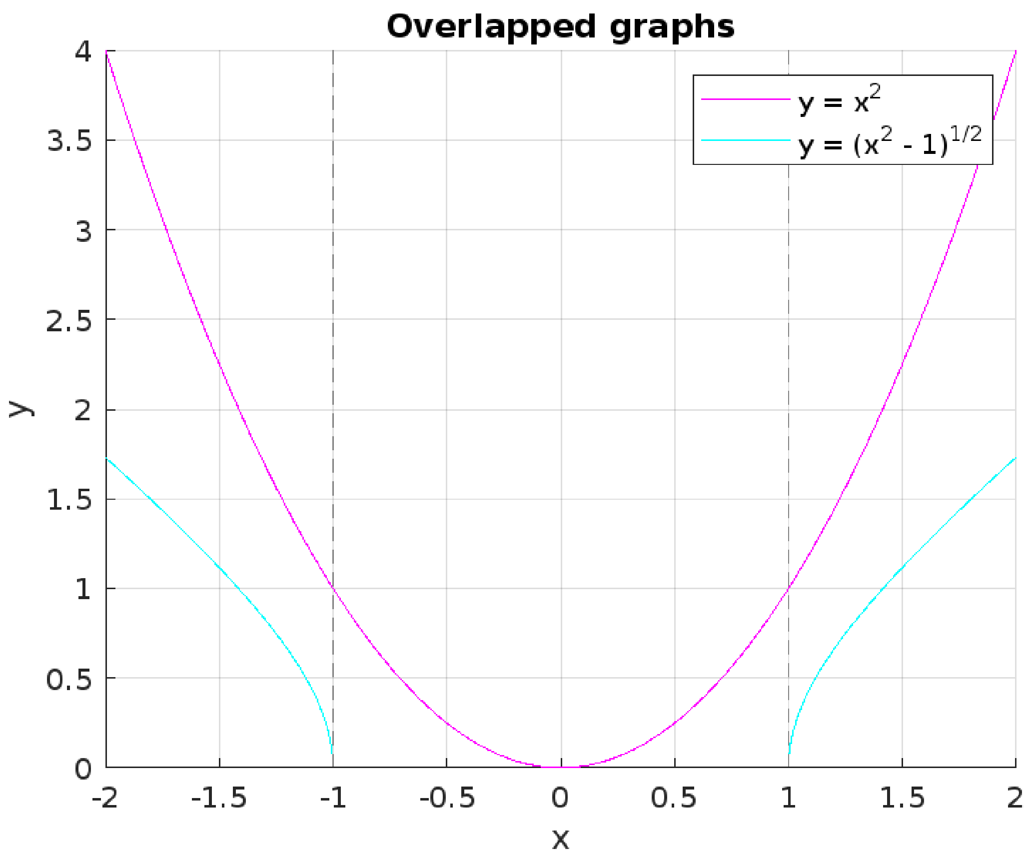
It is worth noting that we can also pass an array to the anonymous function and this is the key to create the graphs using the **fplot** command.

EX 3

We repeat what we did in example 2, this time using the **fplot** function.

```
figure
hold on
grid on
xlabel('x')
ylabel('y')
title("Overlapped graphs")
f = @(x) x.^2;
fplot(f, [-2 2], 'm')
% We can also define the anonymous
% function within the fplot command
fplot(@(x) sqrt(x.^2 - 1), [-2 2], 'c')
legend("y = x^2", "y = (x^2 - 1)^{1/2}")
```

We get the following result, this time without any warning.



SUBPLOTS

We want to create a figure with a 2×2 subfigures respectively containing the following graphs:

- ❖ $y = x^2 - 2x + 1$ in the interval [0, 2]
- ❖ $y = x\,\sin(x)$ in the interval [-5 5]
- ❖ $y = x|x| - 2|x| + 1$ in the interval [-5 5]
- ❖ $y = \frac{x^2-2x+1}{x^2+1}$ in the interval [-5 5]

We can proceed in the following way:

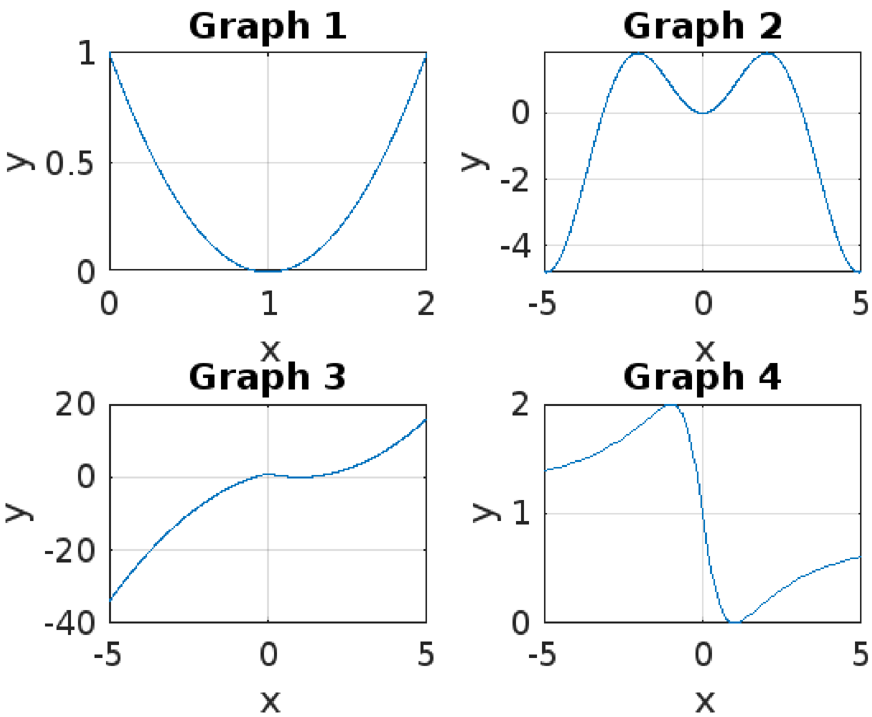
1) create a new figure with the command **figure**. We can skip this point if we are using just one figure
2) Use the command **subplot(m, n, p)** for each subplot, where m and n are integer numbers corresponding to the $m \times n$ subplot figure we want to depict and p is the number of the current subplot

For example, if $m = 3$ and $n = 2$, we have the following situation with the corresponding numeration of the subplots

Subplot 1 (p = 1)	Subplot 2 (p = 2)
Subplot 3 (p = 3)	Subplot 4 (p = 4)
Subplot 5 (p = 5)	Subplot 6 (p = 6)

With these premises, we can create our first subplot graph in the next slide.

```
figure
subplot(2, 2, 1)
x = linspace(0, 2, 1000);
y = x.^2 - 2*x + 1;
plot(x, y)
xlabel('x')
ylabel('y')
title("Graph 1")
grid
subplot(2, 2, 2)
x = -5:.01:5;
y = x.*sin(x);
plot(x, y)
xlabel('x')
ylabel('y')
title("Graph 2")
grid
subplot(2, 2, 3)
y = x.*abs(x) -2*abs(x) + 1
plot(x, y)
xlabel('x')
ylabel('y')
title("Graph 3")
grid
subplot(2, 2, 4)
fplot(@(x) (x.^2 - 2*x + 1)./(x.^2 + 1), [-5 5])
xlabel('x')
ylabel('y')
title("Graph 4")
grid
```



GRAPH OF z=f(x,y)

Consider a function of two real variables

$$f : A \subseteq \mathbb{R}^2 \rightarrow \mathbb{R}$$

We want to depict its **graph** (or **plot**)

To the scope there exist **two different ways** in MatLab:

1) PUNCTUAL DEFINITION

2) ANONYMOUS FUNCTION

1) Graph with punctual definition

The steps are the following

a. **Define the interval of values that must be considered for the two independent variables.** The interval must be defined as row vectors, x and y, having an high number of equally spaced elements. Thus the operator : or linspace can be used.

b. **Define a grid** on the plane (x,y) constituted by the set of couples having one element of the vector x and the second element taken form the vector y

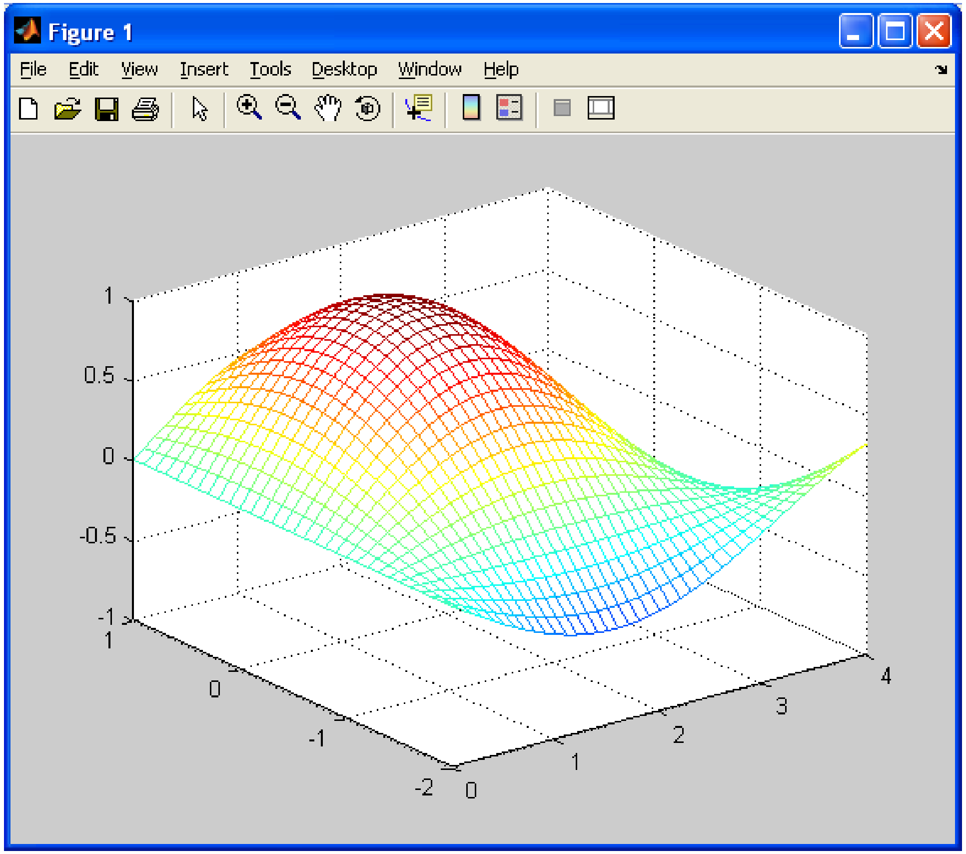
   the command [X Y]=meshgrid(x,y) creates matrices X and Y

c. **Calculate the function** z=f(X,Y) by applying f to the matrices X and Y. The punctual operators and the syntax of elementary functions must be considered. In such a way the value associated to each couple (x,y) is computed

d. **Depict the graph** of the function by using commands surf(X,Y,z) or mesh(X,Y,z) that plots the set of points (x,y,z) in $\mathbb{R}^3$

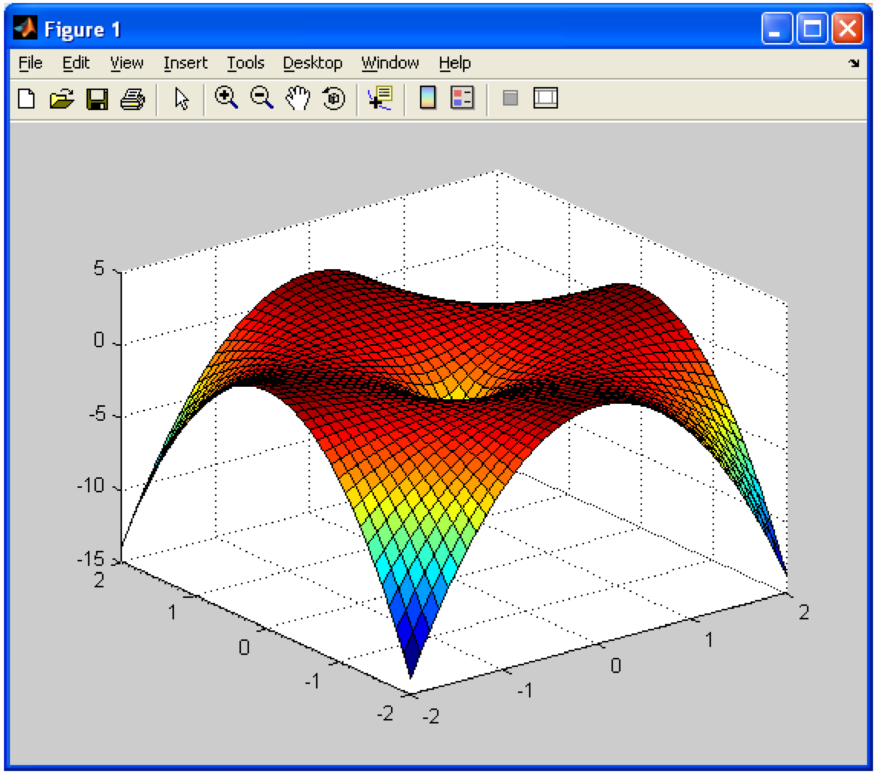
EX 4 (1) $z = \sin(x) \cdot \cos(y)$

```
>> x=0:0.1:4;
>> y=-2:0.1:1;
>> [X Y]=meshgrid(x,y);
>> z=sin(X).*cos(Y);
>> mesh(X,Y,z)
```



EX 5 (2) $z = \ln(x^2 + y^2) - x^2 y^2$

```
>> x=-2:0.1:2;
>> y=-2:0.1:2;
>> [X Y]=meshgrid(x,y);
>> z=log(X.^2+Y.^2)-(X.^2).*(Y.^2);
>> surf(X,Y,z)
```



EX 6

Plot the graphs of the following functions (punctual definition)

(1) $z = \ln(x) \cdot \ln(y)$

consider $x \in [1,4]$ and $y \in [1,4]$ and use mesh

(2) $z = x^2 + y^2 - \cos(x) - \cos(y)$

consider $x \in [-1,1]$ and $y \in [-1,1]$ and use surf

1) Graph with anonymous function

The steps are the following

a. **Define the anonymous function by using the following expression:**

   z=@(x,y) law_of_xy

   thus f(x,y) will be associated to z

Notice **It is then possible to calculate the value of z at a given point (x0,y0) by using the command z(x0,y0)**

b. **Depict the plot by using one of the following commands:**

   ezsurf(z,[x_min x_max],[y_min y_max]) **(deprecated), or**

   ezmesh(z ,[x_min x_max],[y_min y_max]) **(deprecated), or**

   fsurf(z, [x_min x_max y_min y_max])

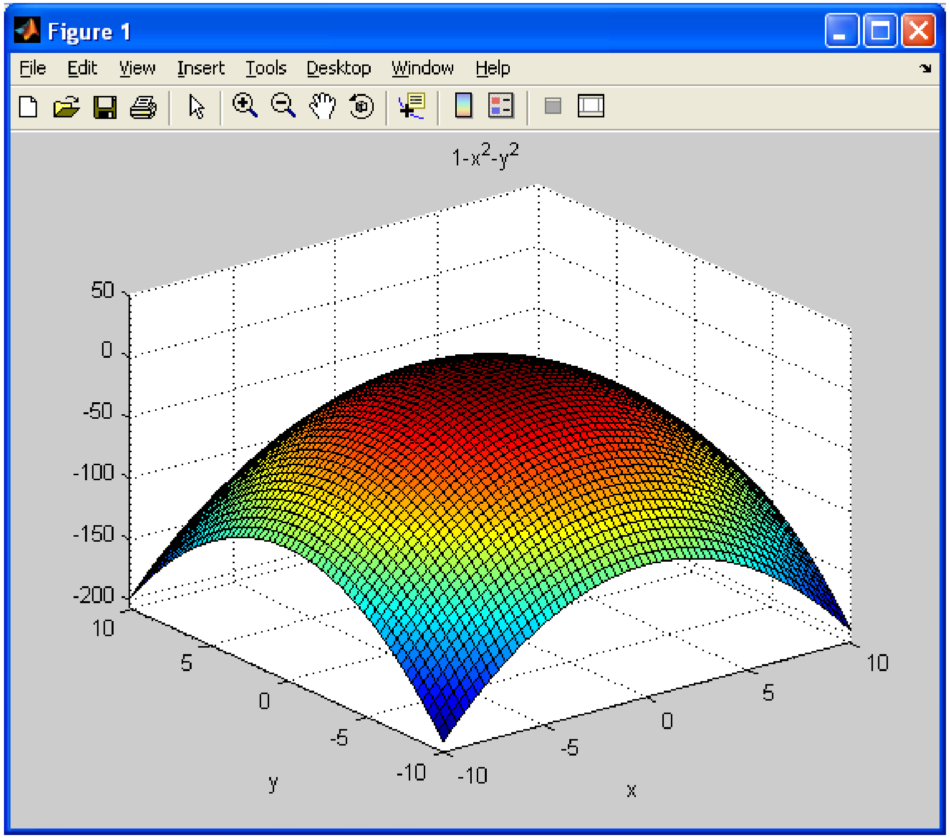
   and the graph will be represented for the independent variables belonging to the defined intervals

EX 7

(1) $z = 1 - x^2 - y^2$

```
>> z=@(x,y) 1-x.^2-y.^2;
>> ezsurf(z,[-10 10],[-10 10]);
```

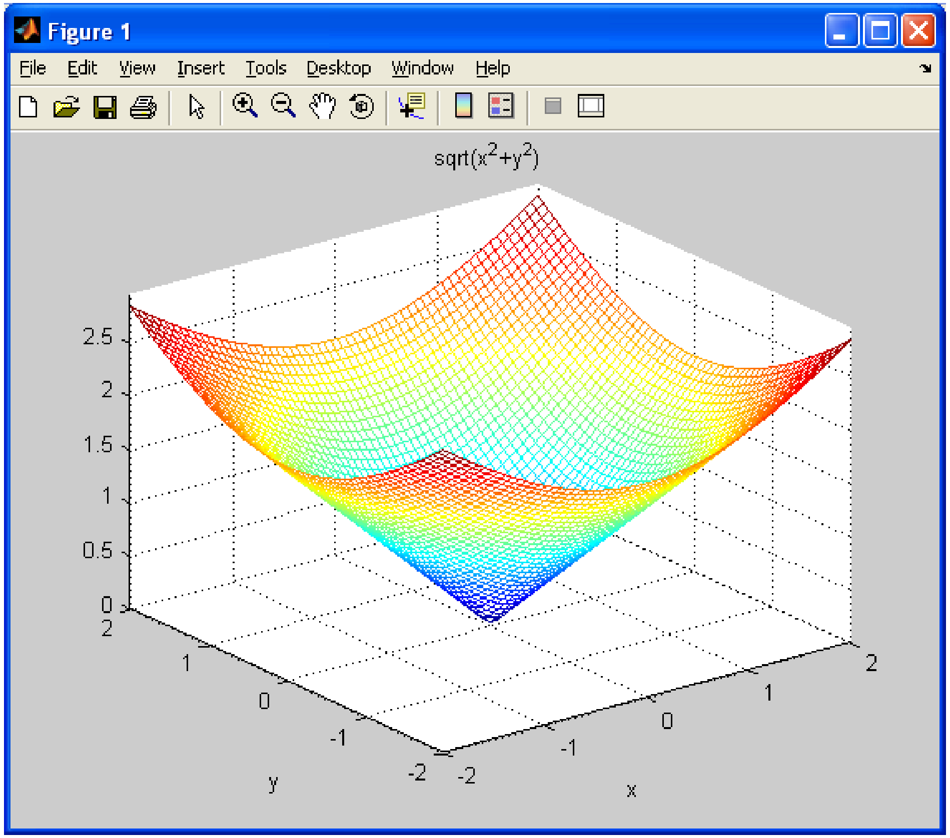
```
>> z(50,50)

ans =

     -4999
```



EX 8

(2) $z=\sqrt{x^2+y^2}$

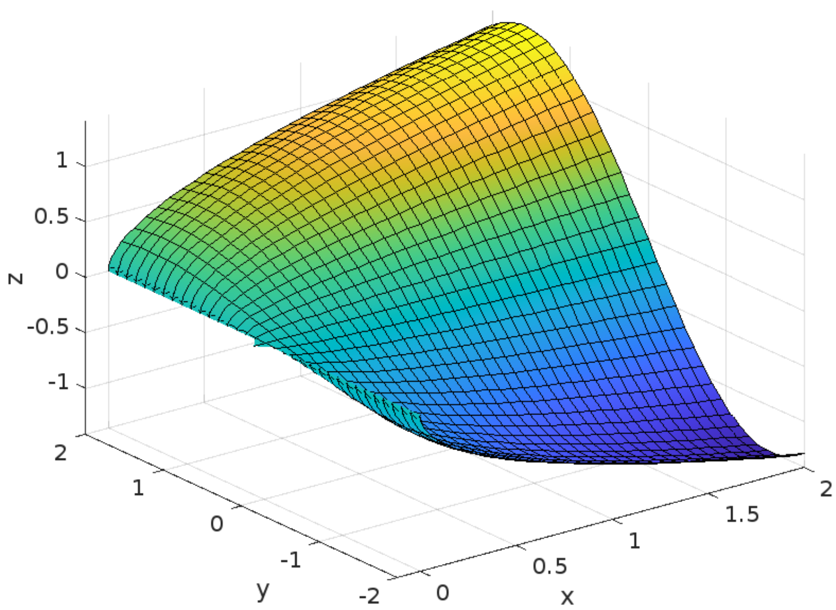
```
>> z=@(x,y) sqrt(x.^2+y.^2);
>> ezmesh(z,[-2 2],[-2 2]);
```



EX 9

$$z = \sqrt{x}\,\sin(y)$$

```
figure
fsurf(@(x, y) sqrt(x).*sin(y), [-2 2 -2 2])
xlabel('x')
ylabel('y')
zlabel('z')
```



Another important remark - I

The **mesh** or **surf** functions would have not worked in the previous example in the specified domain. In fact, if we try to execute the following piece of code

```
X = -2:.1:2;
Y = -2:.1:2;
[x, y] = meshgrid(X, Y);
z = sqrt(x).*sin(y);
mesh(x, y, z)
```

we would get the following error:

Error using **mesh**
X, Y, Z, and C cannot be complex.

Another important remark - II

This is because we are considering a domain where the square root of x is not defined.
In one-dimensional graphs, Matlab gave us a warning but this time is more restrictive.
Therefore, we have two options:

1) We create a domain with the **meshgrid** command that is a subset of the domain A of the function we want to visualize, or
2) We use the **fsurf** function.

Visualisation options

Once the graph is obtained **the options related to the visualisation of the graph can be activated** and the tools of the graph-window can be used

- first **show plot tools** by activating View -> Palette, Browser, Editor

- click on the **surface** to modify its characteristic

- click on the **space** to modify the graph properties (such as title, labels, ticks and so on)

EX 10

Plot the graphs of the following functions (anonymous definition)

$$(1) \quad z = \sqrt{|x|\,y^2} - |x| \text{ (use command ezsurf)}$$

$$(2) \quad z = (xy)e^{x^2 - y^2} \text{ (use command ezmesh)}$$

Select a **suitable interval** for variables x and y

Adjust the obtained graphs by using the plot tools

LEVEL CURVES

The level curves of function z=f(x,y) can be plotted with MatLab

1. **Punctual definition**

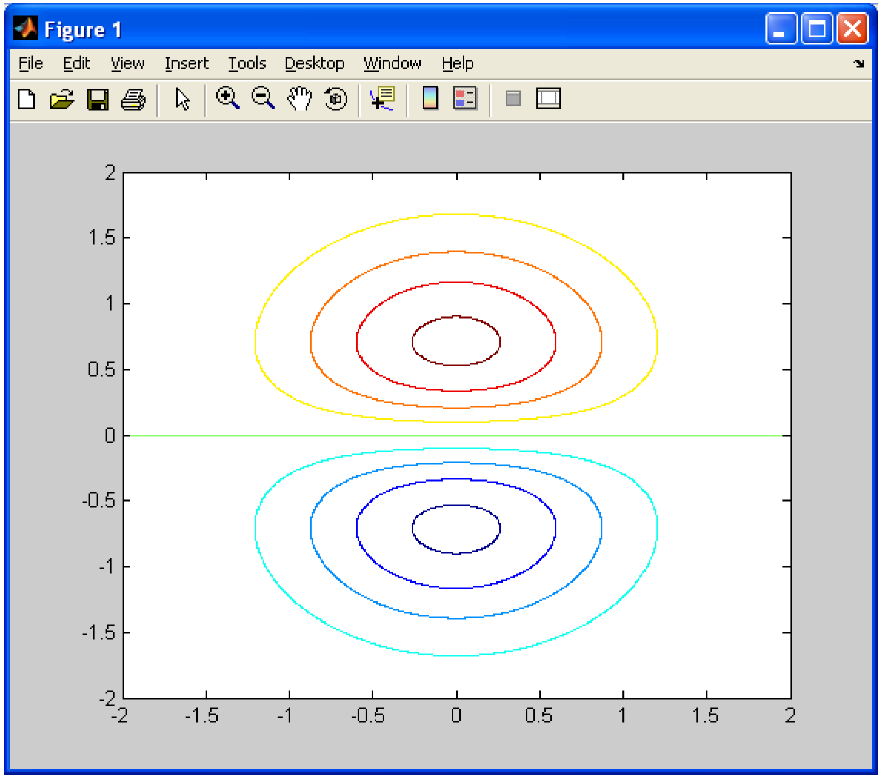
Define the function by discretization and then use the command

contour(x,y,z) (or contourf(x,y,z)) to obtain the level curves

EX 11

$$(1) \quad z = ye^{-x^2-y^2}$$

```
>> x=linspace(-2,2,1000);
>> y=linspace(-2,2,1000);
>> [X Y]=meshgrid(x,y);
>> z=Y.*exp(-X.^2-Y.^2);
>> contour(x,y,z);
```



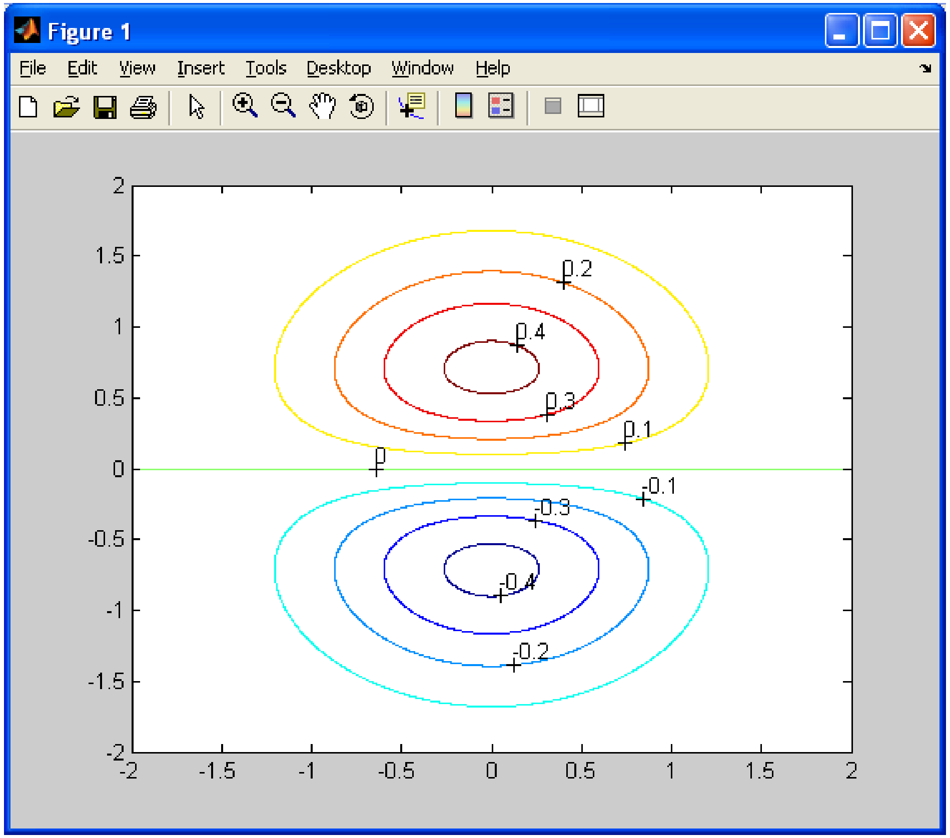
Notice: **It is possible to add the z-value to each level curve**

An output variable must be saved (for example c) while using the command contour:

[c]=contour(x,y,z)

With the instruction **clabel(c)** the z-value will be reported to each curve

```
>> [c]=contour(x,y,z);
>> clabel(c);
```



EX 12

Plot the level curves of the following functions (punctual definition); choose opportune intervals

(1) $z = \dfrac{1}{x^2 + y^2 + 1}$ (use command contour)

(2) $z = |\sin(x) + \cos(y)|$ (use command contourf)

LEVEL CURVES

2. Anonymous definition

Define the fuction as an anonymous function

The command ezcontour(z,[x_min x_max],[y_min y_max])

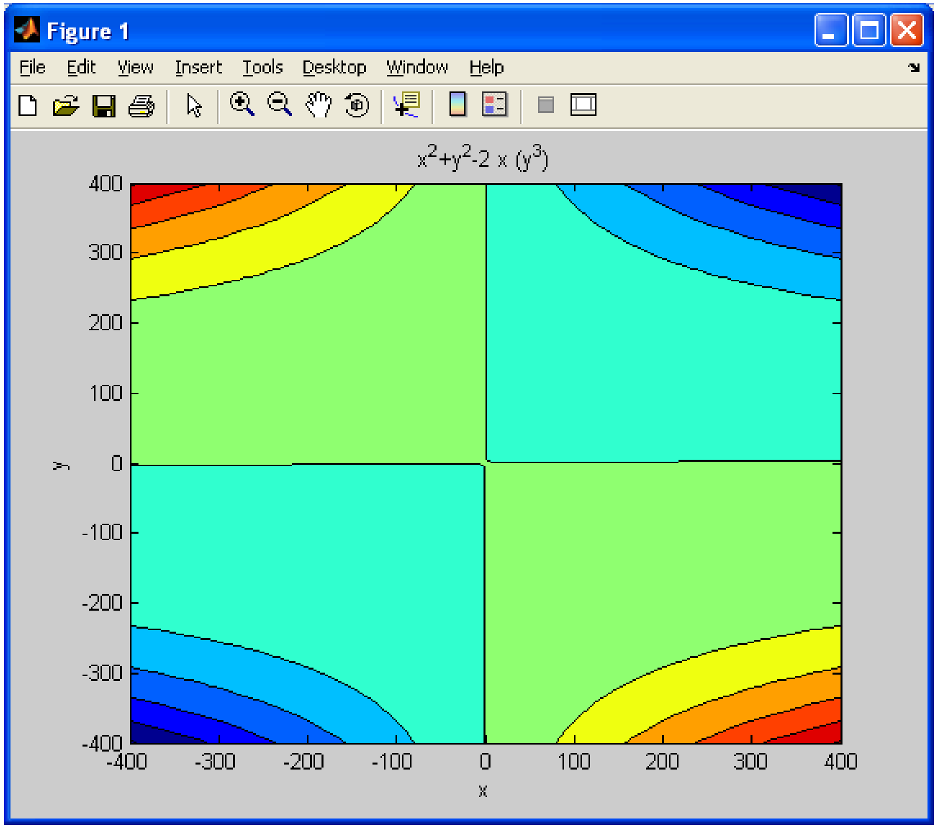
(or ezcontourf(z,[x_min x_max],[y_min y_max])) can be used to plot the level curves. However, these commands are deprecated.

Alternatively, the command fcontour(z, [x_min x_max y_min y_max]) can be used.

EX 13

(2) $z = x^2 + y^2 - 2xy^3$

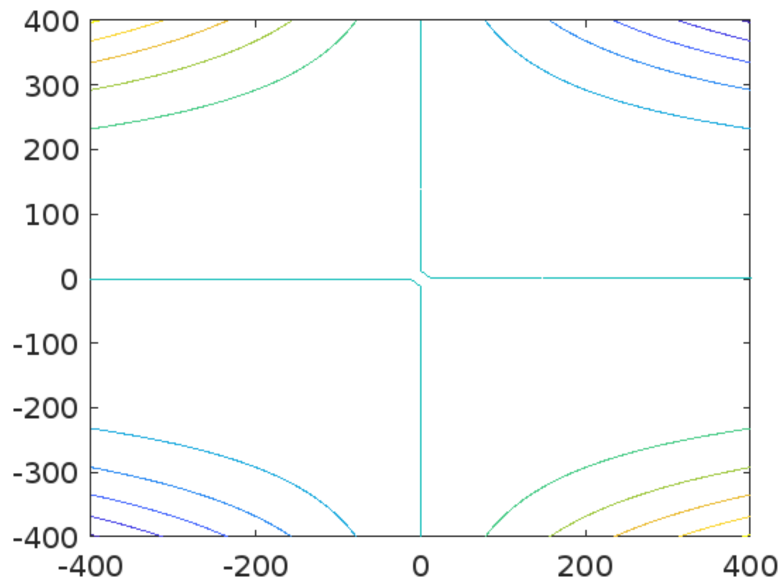
```
>> z=@(x,y) x.^2+y.^2-2*x.*(y.^3);
>> ezcontourf(z,[-400 400],[-400 400]);
```



EX 14

We repeat example 13 with the **fcontour** command.

```
figure
fcontour(@(x, y) x.^2 + y.^2 - 2*x*y.^3, [-400 400 -400 400])
saveas(gcf, "figure8.png") % optional
```



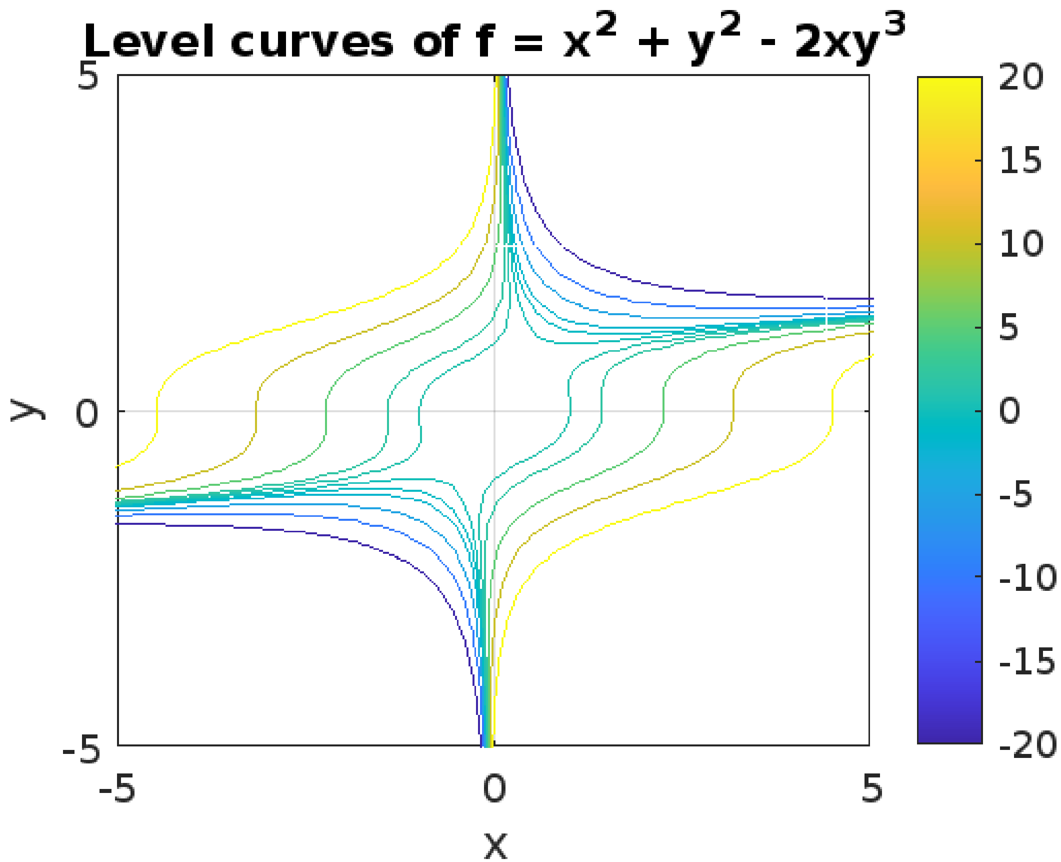
EX 14 (continued)

We can also add labels to the axes with **xlabel** and **ylabel** and select what level curves to draw using a figure handle.

```
figure
h = fcontour(@(x, y) x.^2 + y.^2 - 2*x*y.^3);
xlabel('x')
ylabel('y')
title("Level curves of f = x^2 + y^2 - 2xy^3")
h.LevelList = [-20 -10 -5 -2 -1 0 1 2 5 10 20];
colorbar
grid
saveas(h, "figure9.png") % optional
```

EX 14 (continued)

This is what we get:



EX 14

Plot the level curves of the following functions by using the anonymous definition

$$(1)\quad z=\ln\left(\left|xy\right|\right)+\sqrt{x^2+y^2}$$

$$(2)\quad z=x^2+y^2-1$$

It is also possible to plot both the surface and the level curves in the 3D space

1. **Punctual definition**

   Define the function and then use the commands surfc(x,y,z) (or meshc(x,y,z))

2. **Anonymous definition**

   Define the function and then use the commands

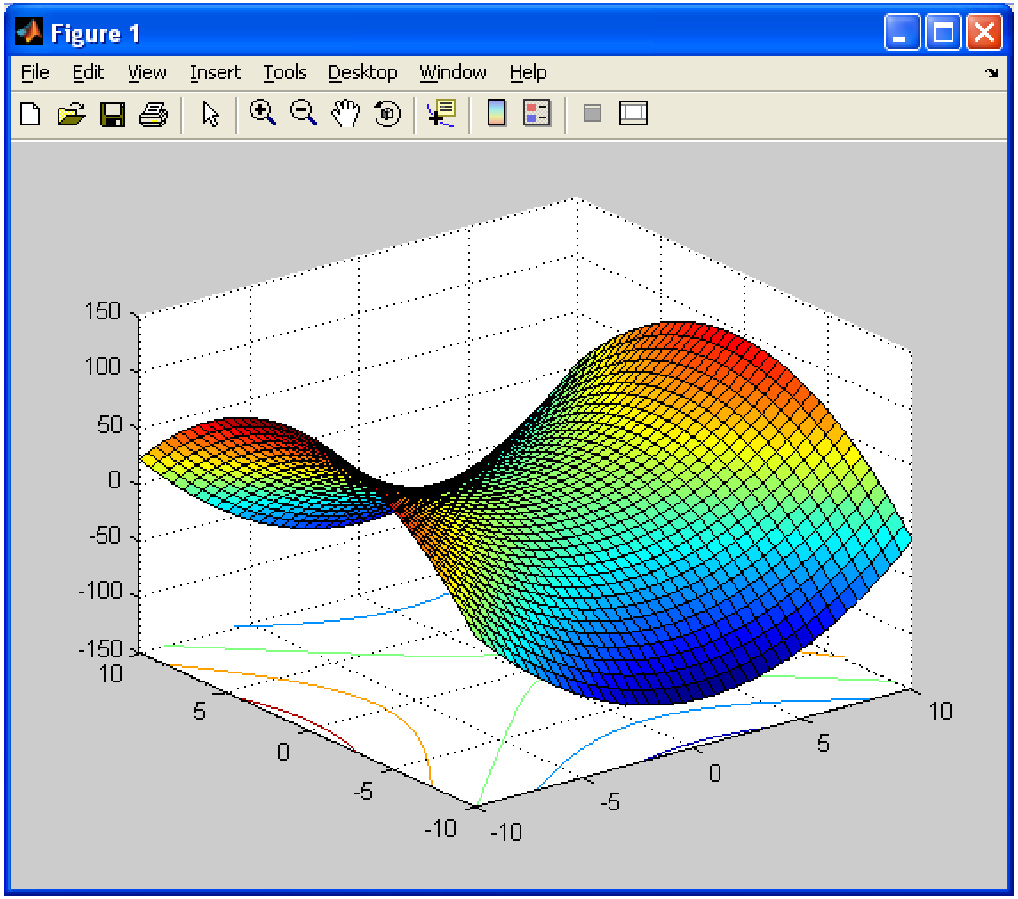
   ezsurfc(z,[x_min x_max],[y_min y_max])

   (or ezmeshc(z,[x_min x_max],[y_min y_max]))

EX 15

(1) $z = x^2 - y^2 - x + 2 + y$

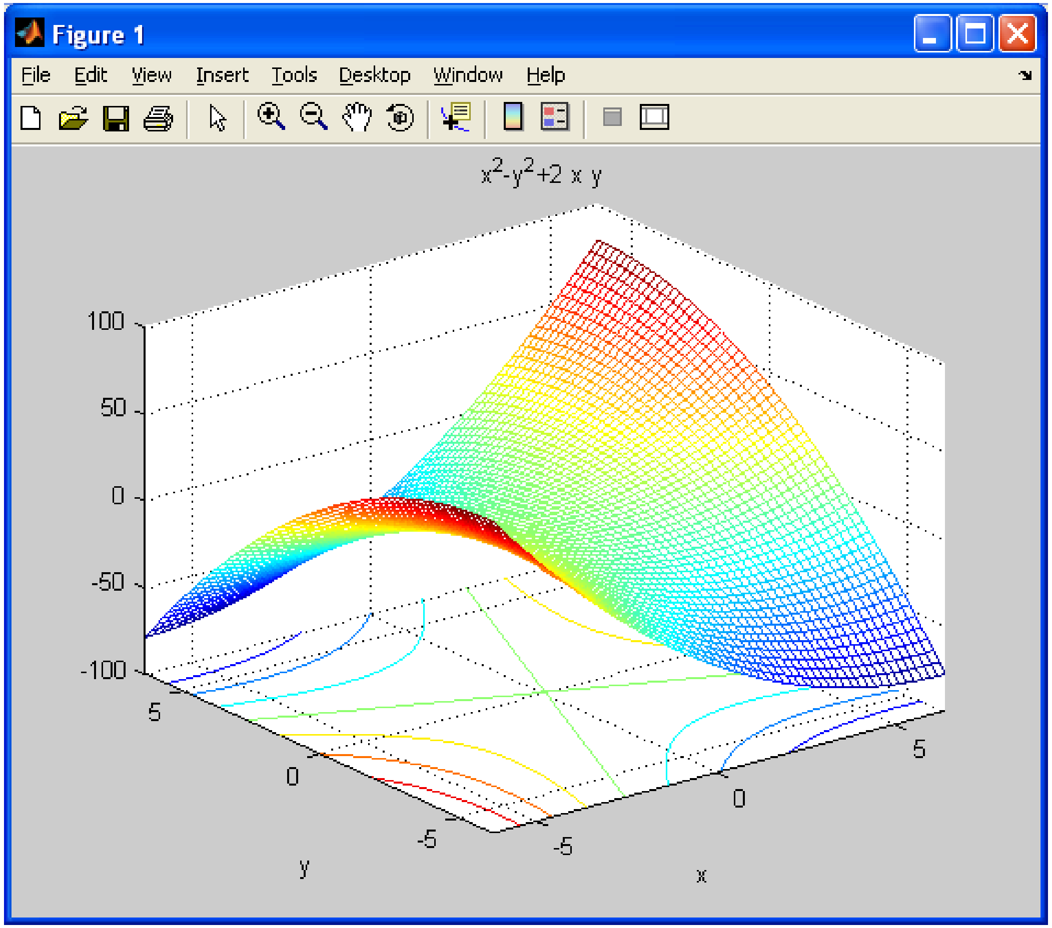
```
>> x=-10:0.5:10;
>> y=-10:0.5:10;
>> [X Y]=meshgrid(x,y);
>> z=X.^2-Y.^2-X+2+Y;
>> surfc(x,y,z);
```



EX 16

(2) $z = x^2 - y^2 + 2xy$

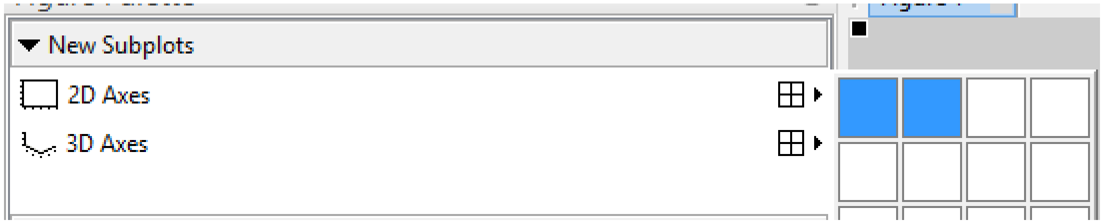
```
>> z=@(x,y) x.^2-y.^2+2*x.*y;
>> ezmeshc(z);
```



PLACE GRAPHS SIDE BY SIDE

It is also possible to plot two graphs side by side

Once a graph is obtained, by using the tool of the figure-window, it is possible to select the **new subplots** options



Then one of the plots can be selected: all the commands given in the command window will be applied to the selected plot. Change selection to apply command to a different plot

Notice: **the command axis square can be used to obtain a square plot area**

It is also possible to use the subplot command as we did for one-dimensional graphs.

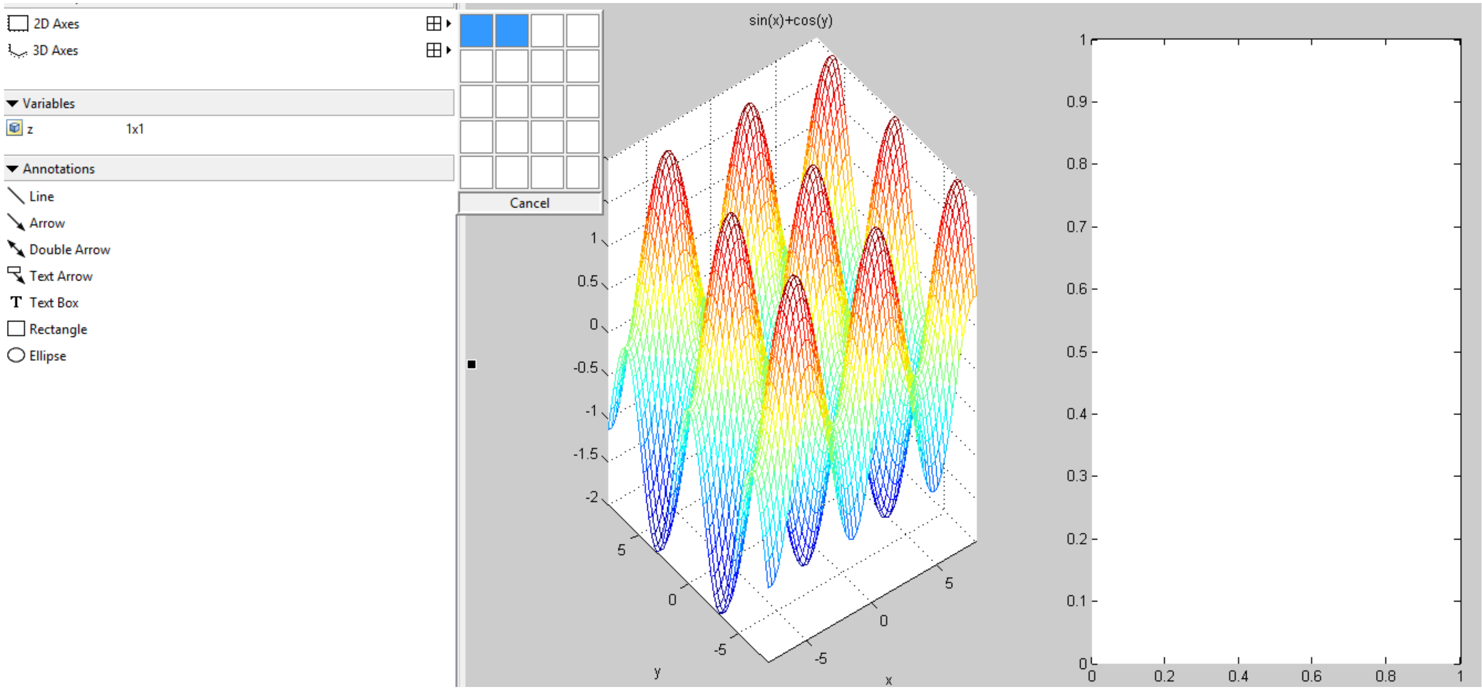
EX 17

Plot the graph of the following function and put the level curves on the right hand side

$$z = \sin x + \cos y$$

1) Fristly plot the graph and then select a second subplot

```
>> z=@(x,y) sin(x)+cos(y);
>> ezmesh(z,[-8 8],[-8 8])
```



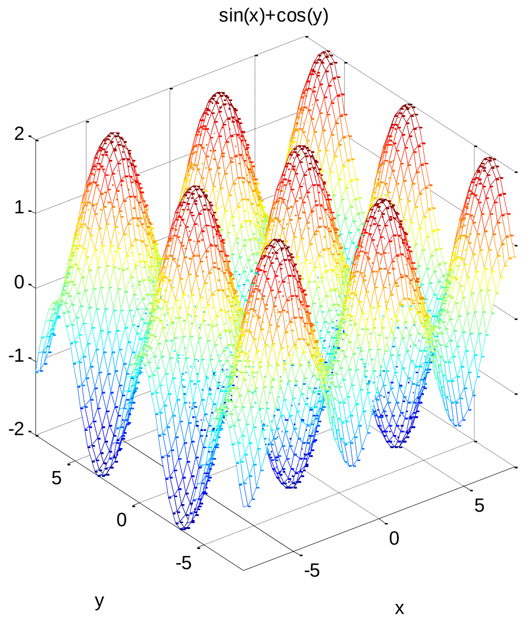
2) Select the second subplot and give instruction on the command window

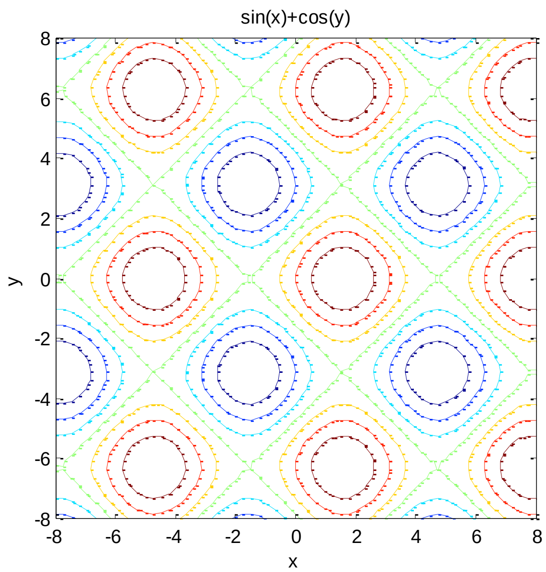
>> ezcontour(z,[-8 8],[-8 8])

Once obtained the second plot, select the first one and give command

>>axis square

And do the same with the second graph!





Notice: **the figure can be saved in several formats, for instance jpg**

EX 18

Plot the graphs and the level curves (side by side) of the following functions

(1) $z = \sqrt{|x^2 - y^2|} + ye^{x^2+y^2}$ (use the anonymous definition)

(2) $z = x^2 + y^2 - xy$ (use the puntual definition)

DOMAIN OF FUNCTIONS

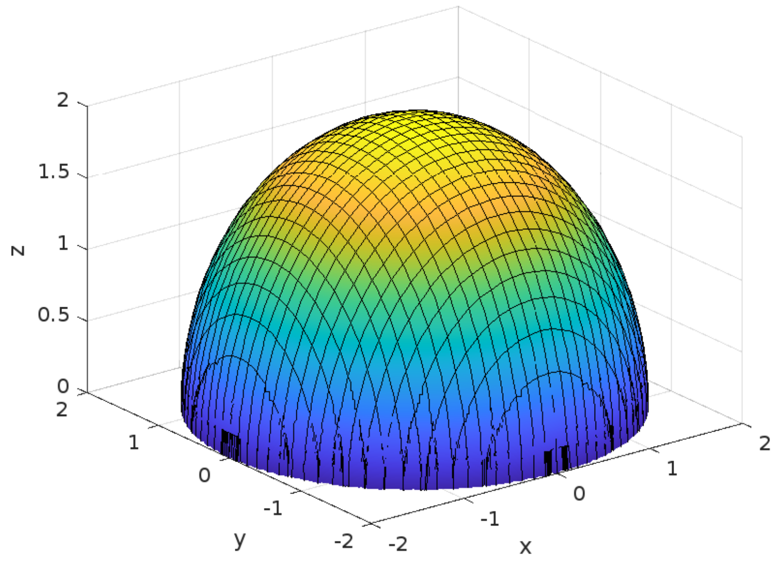
A naive approach to depict the domain of a function in Matlab is offered by the following approach. Consider, for instance, the function

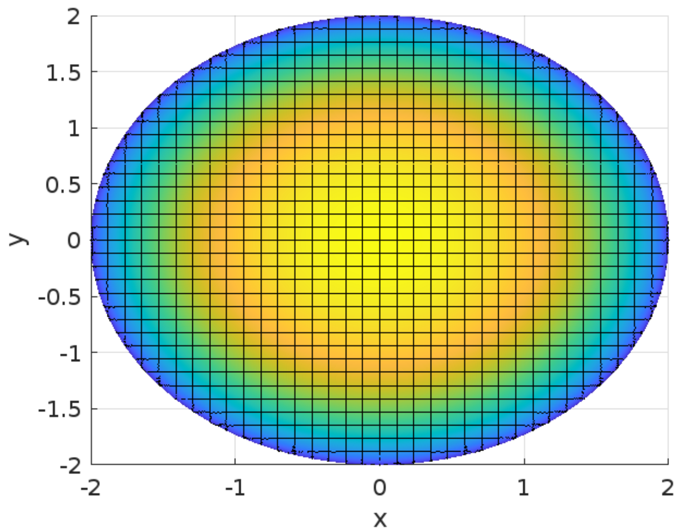
$$z = \sqrt{4 - x^2 - y^2}$$

The idea consists of the following steps:

1) plot the original function with the **fsurf** command and
2) project it on the xy plane with the **view(2)** command

```
z = @(x, y) sqrt(4 - x.^2 - y.^2);
h = figure;
fsurf(z, [-3 3 -3 3])
xlabel('x')
ylabel('y')
zlabel('z')
saveas(h, "figure11.png")
h = figure;
fsurf(z, [-3 3 -3 3])
xlabel('x')
ylabel('y')
zlabel('z')
view(2)
saveas(h, "figure12.png")
```





The proposed approach has some limitations:

- For more «complex» functions, it becomes difficult to get a precise figure
- Some points or segment or lines can miss from the domain and this lack may not be appreciated in the figure proced by Matlab

Therefore, this approach DOES NOT substitute a «by hand» identification of the domain of the function studied. However, it can be a useful tool as a confirmation of the computations made by hand.

Homeworks

1.9 Consider the following functions

$$
\begin{aligned}
&(1) \quad y = e^{-x}\tan(x) \\
&(2) \quad y = \sqrt{4 - x^2} \\
&(3) \quad y = \tan^{-1} x + 2\sin(x) \\
&(4) \quad y = \cos(x)\ln(1 + |x|)
\end{aligned}
$$

- Make a 4×4 subplot figure having one graph for each subplot
- Make a 2×1 subplot figure having functions (1) and (2) in subplot 1 and functions (3) and (4) in subplot 2
- Make a 1×2 subplot figure having functions (1) and (2) in subplot 1 and functions (3) and (4) in subplot 2

Possibly use a different color for each graph

Notice that: the command for the $\tan^{-1}(x)$ function is **atan(x)**

Homeworks

1.10 Consider the following two functions

$$(1)\quad z = x^2 - y^2 - 5xy$$

$$(2)\quad z = \sqrt{x^2 + y^2 - 3}$$

- Calculate the value of z for x=12 and y=-2 for both functions
- Plot the graph of function (1) together with its level curves and then put on the right hand side the graph of function (2)
- Adjust the graph by using the plot tools and save the final figure in jpg format

Notice that: it is necessary to use the anonymous definition!

Homeworks

1.11 Consider the following function

$$z = \log|x^2 y|$$

- Plot the graph and then put the level curves on the right hand side by specifying the z values

Notice that: it is necessary to use the punctual definition!

Homeworks

1.12 Consider the following linear utility function

$$y = 0.5x_1 + 0.2x_2$$

- Plot the graph and then put the indifference curves on the right hand side

Notice that: (1) the indifference curves are the level curves; (2) being an economic function only not-negative values of x and y must be considered!

Homeworks

1.13 Consider the following CES production function

$$z = 2(3x^{-0.5} + 0.5y^{-0.5})^{-0.2}$$

- Plot the graph and then put the isoquants on the right hand side

Notice that: being an economic function only not-negative values of x and y must be considered!

Homeworks

1.14 Consider the following functions (quadratic forms):

$$z = x^2 + y^2$$
$$z = -x^2 - y^2$$
$$z = x^2 - y^2$$
$$z = (x + y)^2$$
$$z = -(x + y)^2$$

- Plot each graph and then put the level curves on the right hand side by specifying the z values
- Make a subplot of each quadratic form with its level curves
- Make a subplot with all the graphs of the above quadratic forms.