

Práctica 3: Gráficas y curvas de nivel

Índice

1. Gráficas en \mathbb{R}^2

Para representar una función $y = f(x)$, $x \in [a, b] \subset \mathbb{R}$ existen dos posibilidades:

- Construir un vector de puntos $(x_k)_{k=1,\dots,n}$ contenidos en el intervalo $[a, b]$ y dibujar segmentos que unan los puntos $(x_k, f(x_k))$, $k = 1, \dots, n$. La forma de definir el vector es $x=a:p:b$, siendo a el primer valor de x , b el último, y p el paso que permite avanzar de a hasta b .

La forma de actuación de la función sobre un vector no es siempre la misma. Hay funciones que al actuar sobre un vector lo hacen sobre cada una de las componentes (por ejemplo \sin , \cos , \exp) y otras que interpretan que se está haciendo una operación con matrices (por ejemplo las potencias y los productos). En este último caso, si queremos evaluar la función sobre cada componente del vector, debemos escribir un punto (.) antes del operador. Por ejemplo,

Ventana de comandos

editor

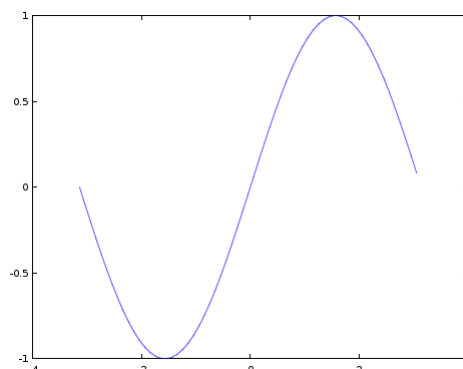
```
x=0:0.5:2
a=exp(x)
b=x^2
c=x.^2
```

```
x=0:0.5:2
x =
    0.00000    0.50000    1.00000    1.50000    2.00000
a=exp(x)
a =
    1.0000    1.6487    2.7183    4.4817    7.3891
b=x^2
error: for A^b, A must be a square matrix. Use .^for elementwise power.
c=x.^2
c =
    0.00000    0.25000    1.00000    2.25000    4.00000
```

Una vez definido el vector de originales, se define el vector de imágenes en la forma $y=f(x)$ y el dibujo se hace con el comando `plot(x,y)`. Por ejemplo, si queremos dibujar la gráfica de la función $\sin(x)$ en el intervalo $[-\pi, \pi]$ debemos escribir:

editor

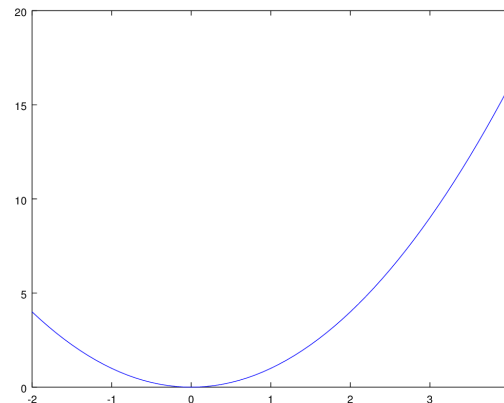
```
x=-pi:0.1:pi;
y=sin(x);
plot(x,y)
```



Y para dibujar la gráfica de $y = x^2$ debemos tener en cuenta la observación relativa al punto antes del operador potenciación.

editor

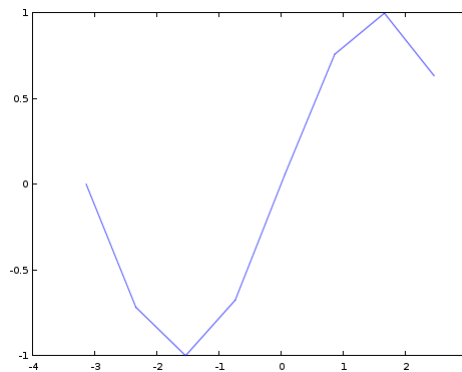
```
x=-2:0.1:4;
y=x.^2;
plot(x,y)
```



Si el número de componentes del vector x es elevado, el aspecto final será de curva. Pero si el número de puntos es escaso, se observarán los segmentos, como en el siguiente ejemplo.

editor

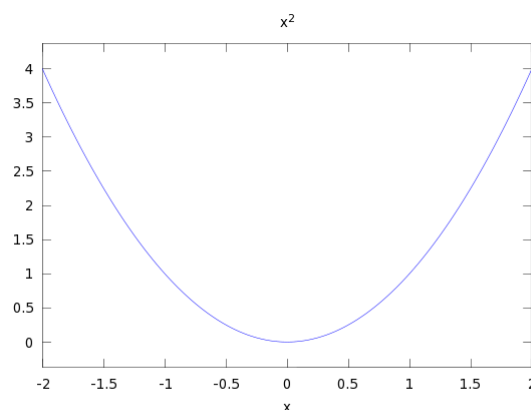
```
x=-pi:0.8:pi;
y=sin(x);
plot(x,y)
```



- El comando `ezplot('f',[a,b])` permite dibujar la gráfica eligiendo automáticamente el número de puntos en el intervalo. Por ejemplo, para representar la función $y = x^2$ en el intervalo $[-2, 2]$ se escribirá:

editor

```
ezplot('x.^2',[-2,2])
```



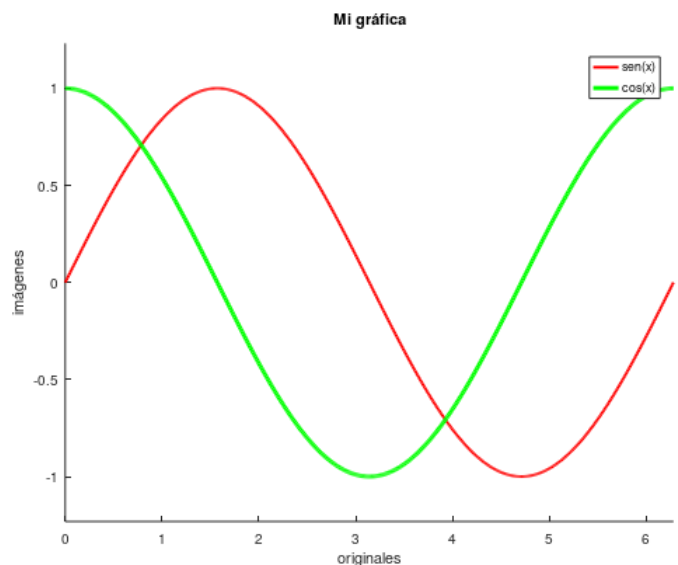
Cada vez que se dibuja una nueva gráfica, se debe borrar la anterior con el comando `clf`. Si queremos dibujar varias gráficas simultáneamente, se deben usar los siguientes comandos: `hold on` y `hold off`, situando en medio las funciones que se desea dibujar. Es conveniente, en el caso de dibujar varias gráficas, elegir colores y una leyenda que permita identificar cada función. Algunos de los comandos que se usan son:

	Código	Opciones
color de la línea	<code>set(..., 'Color','NOMBRE DEL COLOR',...)</code>	blue, black, cyan, green, magenta, red, white, yellow
grosor de la línea	<code>set(...,'linewidth', NÚMERO ENTERO,...)</code>	1,2,3,...
estilo de la línea	<code>set(..., 'linestyle','ESTILO',...)</code>	- continua - - (discontinua) : (dos puntos) . (guión y punto)
título de la gráfica	<code>title('...')</code>	
leyenda de la gráfica	<code>legend('...')</code>	north, south, east, west outside
etiquetas de los ejes	<code>xlabel('...'), ylabel('...')</code>	
líneas guía	<code>grid</code>	activa la inclusión de una cuadrícula en el dibujo
escalas de los ejes	<code>axis equal</code>	establece la misma escala para los dos ejes

A continuación se presenta un ejemplo en el que se dibujan simultáneamente las funciones $\sin(x)$ y $\cos(x)$ en el intervalo $[0, 2\pi]$.

editor

```
clf
hold on;
d1=ezplot('sin(x)',[0,2*pi]);
set(d1, 'Color','red','linewidth',2)
d2=ezplot('cos(x)',[0,2*pi]);
set(d2, 'Color','green','linewidth',3)
title('Mi gráfica')
legend('sen(x)','cos(x)')
xlabel('originales')
ylabel('imágenes')
hold off
```



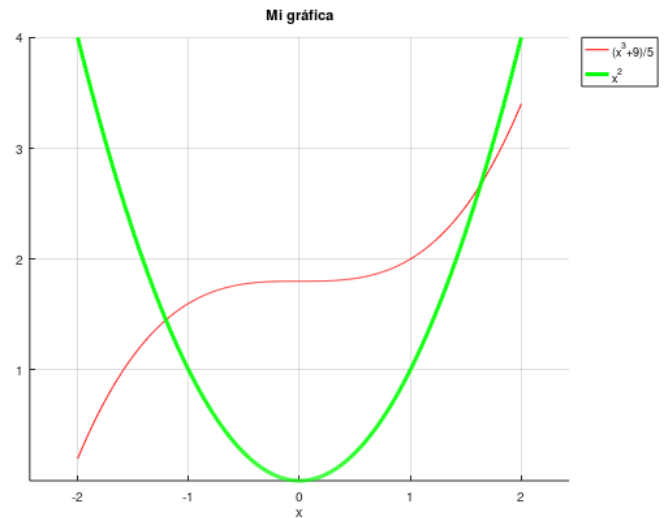
Existe la posibilidad de definir previamente las funciones, como sucede en el ejemplo que figura a continuación.

editor

```

clf
hold on;
f=@(x)(x.^3+9)/5;
d1=ezplot(f,[-2,2]);
set(d1, 'Color','red','linewidth',1)
g=@(x)x.^2;
d2=ezplot(g,[-2,2]);
set(d2, 'Color','green','linewidth',3)
title('Mi gráfica')
legend('(x.^3+9)/5','x.^2',
'location','NorthEastOutside')
grid
hold off
axis equal

```



También se pueden dibujar las funciones definidas implícitamente mediante una relación $f(x, y) = 0$, siendo f una función escalar. El comando que lo permite es `ezplot('función', [a,b,c d])`, siendo $[a, b]$ el intervalo de variación de la variable x y $[c, d]$ el intervalo de variación de la y . Por ejemplo,

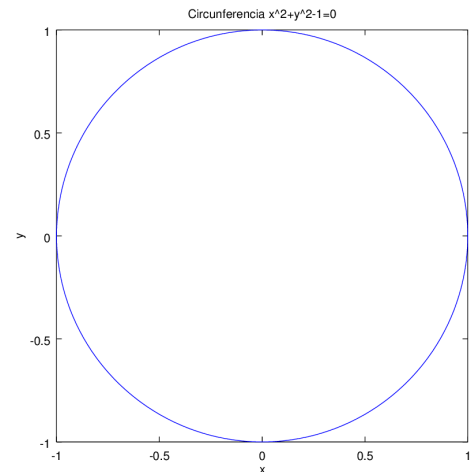
editor

```

ezplot('x^2+y^2-1',[-1,1,-1,1])
title('Circunferencia x^2+y^2-1=0')
axis equal

```

proporciona a circunferencia $x^2 + y^2 - 1 = 0$.

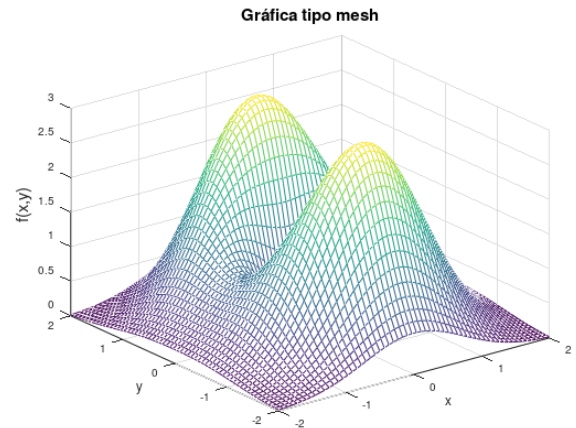


2. Gráficas de funciones escalares en \mathbb{R}^3

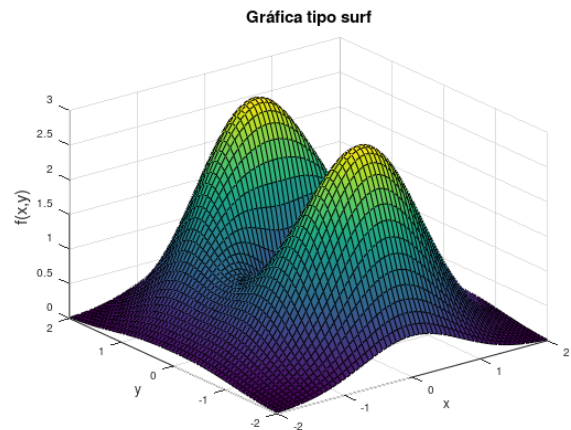
Dada una función $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, existen varios comandos que permiten representar la superficie $z = f(x, y)$ en un sistema de coordenadas xyz . Se debe especificar: la ecuación de la función, el tipo de representación (*mesh* o *surf*) y la variación de las variables x e y mediante la sintaxis $[a, b, c, d]$, siendo $[a, b]$ el intervalo de variación de x y $[c, d]$ el de y . Se pueden añadir etiquetas y un título que complementen el dibujo. Por ejemplo:

editor

```
ezmesh('(x.^2+3*y.^2)*exp(1-x.^2-y.^2)',
[-2,2,-2,2]);
% Añade título
title('Gráfica tipo mesh','fontsize',15);
% Añade etiquetas en los ejes
xlabel('x','fontsize',12);
ylabel('y','fontsize',12);
zlabel('f(x,y)','fontsize',15)
```

**editor**

```
ezsurf('(x.^2+3*y.^2)*exp(1-x.^2-y.^2)',
[-2,2,-2,2]);
% Añade título y etiquetas en los ejes
title('Gráfica tipo surf','fontsize',15);
xlabel('x','fontsize',12);
ylabel('y','fontsize',12);
zlabel('f(x,y)','fontsize',15)
```

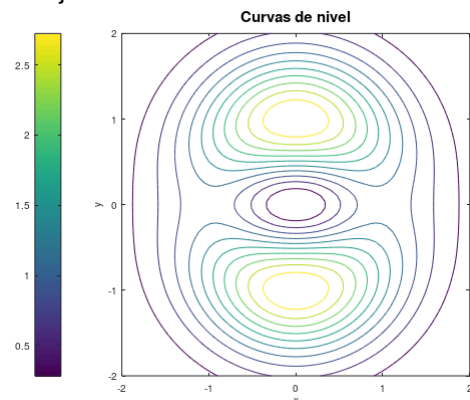


3. Curvas de nivel

Las curvas de nivel son las líneas que se obtienen al intersecar la gráfica de una función con los planos paralelos al XY, cuyas ecuaciones son $z = \text{constante}$. Para la correcta interpretación de las curvas, el gráfico debe estar acompañado de una escala que indique los valores de la función en las distintas curvas. El comando que permite eso es `colorbar`, cuyas opciones para situar a barra son: `WestOutside` a la izquierda, `EastOutside` a la derecha, `NorthOutside` arriba y `SouthOutside` abajo.

editor

```
ezcontour('(x.^2+3*y.^2)*exp(1-x.^2-y.^2)',
[-2,2,-2,2]);
title('Curvas de nivel','fontsize',15);
xlabel('x','fontsize',10);
ylabel('y','fontsize',10);
colorbar('WestOutside')
```

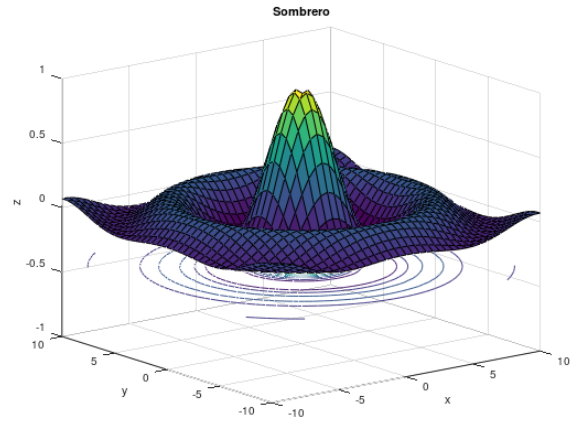


Las curvas de nivel se pueden dibujar al mismo tiempo que la gráfica de la función. Por ejemplo.

```

x = -10:0.5:10; y = -10:0.5:10;
[X,Y] = meshgrid(x,y);
% Define la función Z = f(X,Y)
Z = sin(sqrt(X.^2+Y.^2))./sqrt(X.^2+Y.^2);
% Crea una superficie do tipo contour
surf(X, Y, Z);
view(-38, 18);
% Añade título y etiquetas en los ejes
title('Sombrero');
xlabel('x'); ylabel('y'); zlabel('z');

```



Ejercicio 1 Hacer un dibujo que contenga las gráficas de las funciones $y = x^n$, $x \in [-1, 2]$, para $n = 1, 2, 3, 4$, usando diferentes colores, tipos y grosores de líneas.

Ejercicio 2 Dibujar la frontera del conjunto $\{(x, y); y \geq x^2, y \leq x + 2\}$.

Ejercicio 3 Dibujar la función escalar $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, $f(x, y) = x^2 - y^2$.

Ejercicio 4 Dibujar los conjuntos de nivel de las funciones del ejercicio 3.