

Representación gráfica

A visualización dos resultados pode ser unha ferramenta interesante a hora de interpreta-los resultados. MAPLE permite realizar gráficos en 2 e 3 dimensións. MAPLE dispón dunha gran variedade de comandos para a representación gráfica de expresións que se poden invocar en liña de comandos ou a través da interface gráfica interactiva indo ó menú "Tools", "Assistants", "Plot builder".

¿Qué se pode representar? Pequeno catálogo do máis básico:

- Curvas planas en forma explícita, paramétrica, implícita ou polar.
- Animacións dunha función $f(x)$ que varía co tempo ou con outra variábel.
- Sistemas de inecuacións lineares en 2 variábeis.
- Curvas en 3D dadas en forma paramétrica.
- Superficies en 3D dadas en forma explícita, paramétrica, implícita ou en coordenadas esféricas/cilíndricas.
- Animacións dunha función $f(x,y)$, correspondente a unha superficie en 3D, que varía co tempo.
- Conxuntos ou listas de puntos.

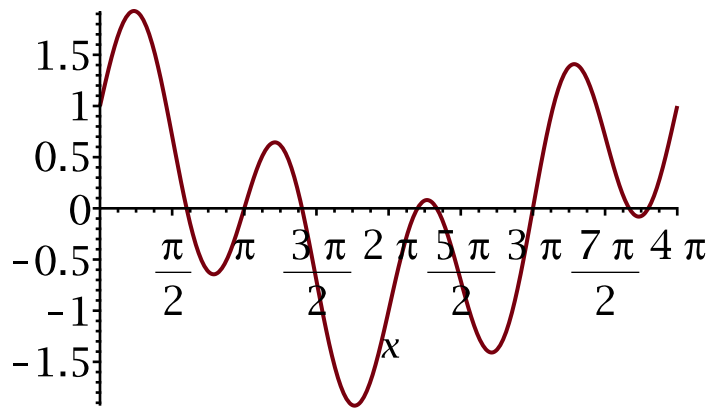
¿Qué opcións se poden variar no gráfico?

- Modifica-lo tamaño e cor de tódolos elementos do gráfico (eixos, títulos, gráficos, etc).
- Modifica-las escalas dos eixos.
- Fixa-lo intervalo de representación da curva, superficie, etc.
- Poñer lendas os eixos e título ó gráfico
- Nos gráficos en 2D: elixir se queremos ve-los puntos unidos por liñas, só liñas ou só puntos. Dependendo do caso seleccionado, pódese fixa-lo tipo e cor da liña e do punto.
- Graba-lo gráfico en formatos de gráficos habituais (JPEG, GIF e EPS).
- Copialo e pegalo a un documento noutra aplicación (Word)
- Etc.

A continuación veñen algúns exemplos en liña de comandos que presentan as posibilidades de representación gráfica de MAPLE. Os gráficos que construímos co programa quedan insertados no propio documento e, con posterioridade, poderíanse modifica-las opcións do gráfico. En primeiro lugar, cargaremos en memoria a libraría plots porque máis tarde ímola necesitar:

Gráficos en 2D

```
> with(plots):plot(cos((1/2)*x)+sin(2*x), x = 0 .. 4*Pi); #  
expresión especificando o rango no eixo x
```

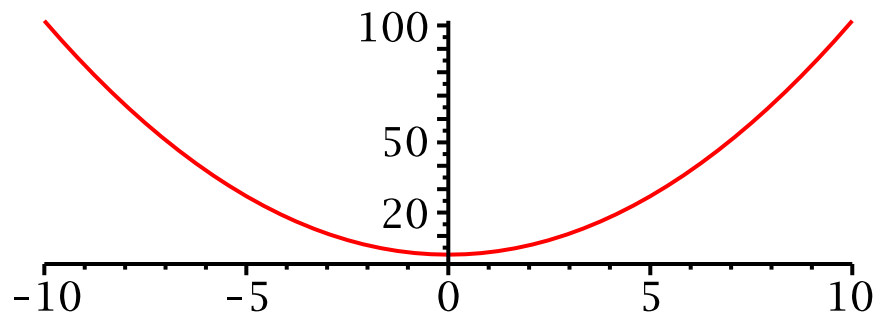


```
> f:=x->x^2+2; # defino unha función
```

$$f:=x \rightarrow x^2 + 2$$

(1.1.1)

```
> plot(f); # as escalas no eixo x e y establécense por defecto.
```



Cando se pulsa co rato no gráfico, actívanse os menús **Plot** e **Drawing**, que proporcionan botóns para configura-lo gráfico (grid, escalas do eixo, cores e tamanos de liñas, puntos e texto, títulos de gráfico e eixos, lendas, ...) e para engadir debuxos (menú **Drawing**).

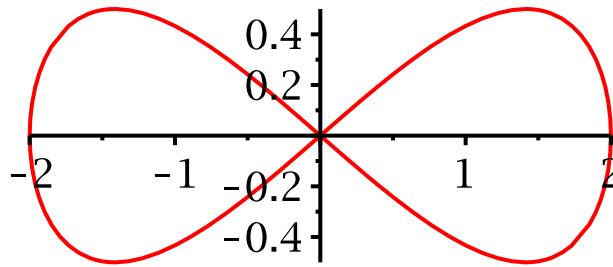
Unha **curva paramétrica** ven dada por ecuacións en función dun parámetro, por exemplo, nun movemento en 2D uniforme en X (con velocidade inicial v_{0x}) e uniformemente acelerado en Y (con velocidade inicial v_{0y} e aceleración a_y) a posición $(x(t), y(t))$ está dada por:

$$x(t) = x_0 + v_{0x}t$$

$$y(t) = y_0 + v_{0y}t + \frac{1}{2}a_y t^2$$

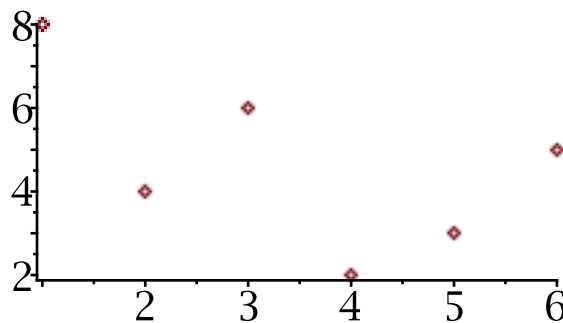
Neste caso, o parámetro é t (o tempo). Para representar unha curva paramétrica:

```
> plot([2*sin(t), sin(t)*cos(t), t=0..2*Pi]); # visualización dunha curva en forma paramétrica
```



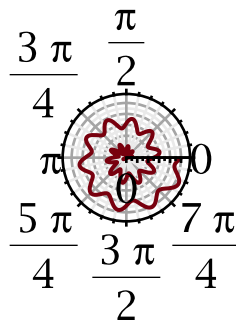
A curva paramétrica defínese como unha lista na que se especificas tanto as expresións como o rango de valores. Para non ver distorsionada a curva especifícase a opción *scaling=constrained* (que fixa o eixo x e y a mesma escala). Tamén se pode especificar premendo no botón dereito do rato (unha vez estea seleccionado o gráfico).

```
> plot([1, 2, 3, 4, 5, 6],[8, 4, 6, 2, 3, 5], style=point,
       symbolsize=20); #para visualizar un vector de puntos
       respecto a outro
```

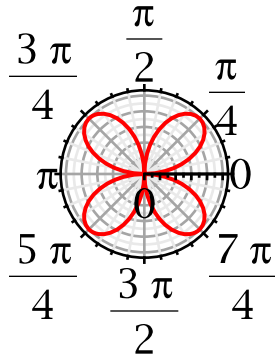


Unha **curva en coordenadas polares** está dada polas coordenadas ρ (radio) e θ (ángulo) de cada punto, de forma $\rho = \rho(\theta)$. As ecuacións de cambio de coordenadas cartesianas a polares son $\rho = \sqrt{x^2 + y^2}$, $\theta = \arctan\left(\frac{y}{x}\right)$. As ecuacións de cambio de polares a cartesianas son $x = \rho \cdot \cos(\theta)$, $y = \rho \cdot \sin(\theta)$. Unha curva en coordenadas polares está dada por unha ecuación da forma $\rho = \rho(\theta)$. Por exemplo $\rho(\theta) = 4\theta$ (espiral de Arquímedes)

```
> polarplot(4*theta+5*sin(10*theta), theta=0..4*Pi);
```

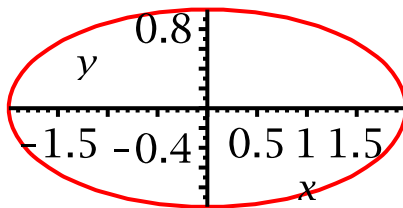


```
> polarplot(sin(2*theta));
```

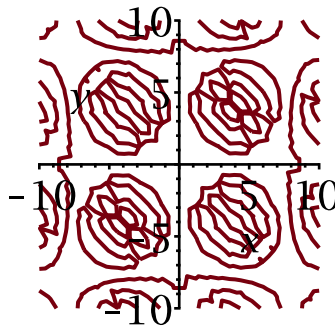


Unha **curva** (en 2D) **en forma implícita** ten a forma $f(x, y) = 0$. Por exemplo, $x^2 + y^2 - 1 = 0$ é a ecuación implícita dunha circunferencia de radio 1 e centro (0, 0). Para representar unha curva implícita:

```
> implicitplot((x^2)/4+y^2=1, x=-4..4, y=-1..1, scaling=
constrained); #debuxa curvas en dúas dimensións dadas de
forma implícita
```

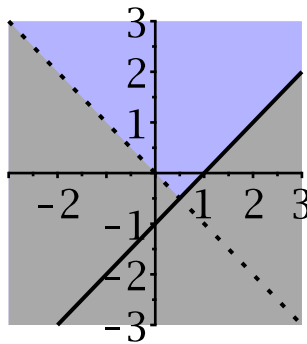


```
> implicitplot(x-y-sin(x-y) + 1, x=-10..10, y=-10..10)
```



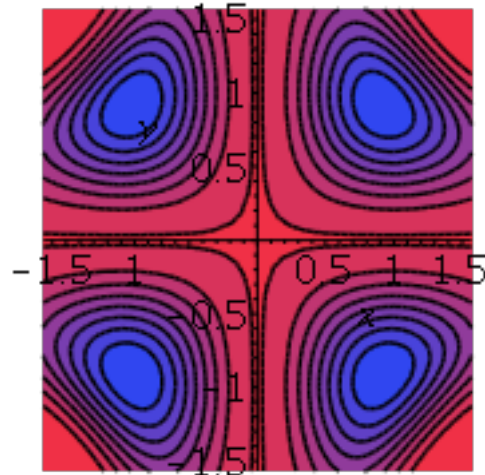
Unha **inecuación** é unha relación dada polos operadores $>$ e $<$ no canto de $=$. A función **inequal** ofrece a posibilidade de representar **sistemas de inecuacións lineais** en dúas variabeis. A sintaxe é: **inequal(ineqs, xspec, yspec, options)**, onde **ineqs** son as inecuacións, **xspec** e **yspec** son os rangos nos que se representa e **options** representa o modo de representación (ver axuda).

```
> inequal({x-y <= 1, 0 < x+y}, x = -3 .. 3, y = -3 .. 3);
```



Finalmente, o **mapa de calor**, coa función **contourplot**, representa cun código de cores (vermello= valores baixos, azul= valores altos) os valores dunha función de dúas variábeis no plano:

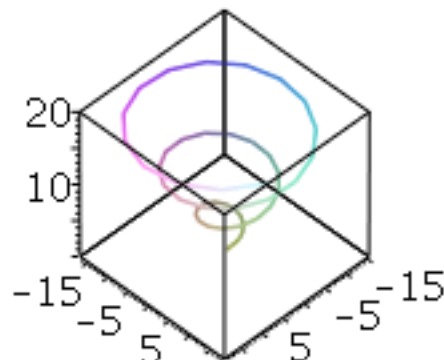
```
> contourplot(exp(-x^2-y^2)*sin(x^2*y^2),x=-Pi/2..Pi/2,y=-Pi/2..Pi/2,
filledregions=true);
```



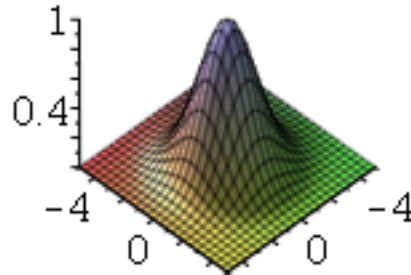
▼ Gráficos en 3D

Os gráficos tridimensionais poden ser de varios tipos. Unha **curva en 3D** exprésase usualmente en **ecuacións paramétricas** $x=x(t)$, $y=y(t)$, $z=z(t)$, en función dun parámetro t . Para representala, emprégase a función **spacecurve**($[x(t),y(t),z(t)],t=a..b$). Por exemplo, dada a curva $x(t)=t \cos t$; $y(t)=t \sin t$; $z=t$, para representala executaremos:

```
> with(plots):spacecurve([t*cos(t),t*sin(t),t],t=0..20);
```



Unha **superficie 3D** ten a forma $z = f(x, y)$: $z = \exp\left(-\frac{x^2 + y^2}{a^2}\right)$ é unha gaussiana de ancho a .

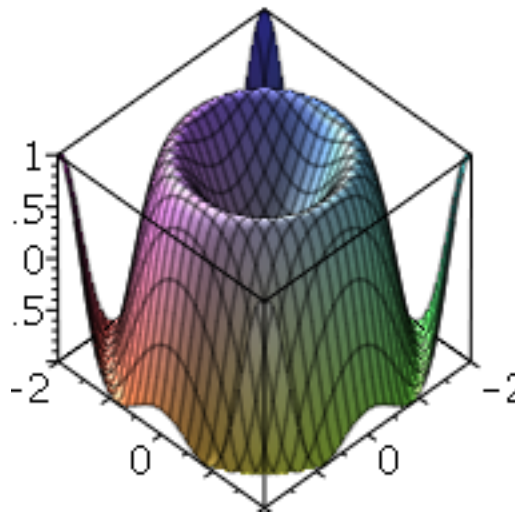


plot3d permite representar gráficos tridimensionais definidos por unha expresión de 2 variabéis definindo o rango no que se quere representar ou como unha función de 2 variabéis (poñemos os rangos sen incluír as variabéis).

```
> plot3d((x^2+y^2)*sin(x+y), x=-2..2, y=-2..2, axes=NORMAL);  
#utilizando unha expresión
```



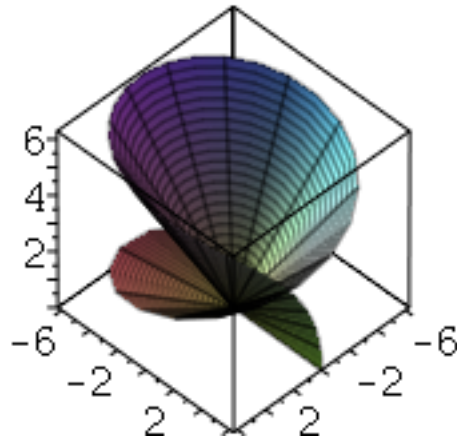
```
> f:=(x,y)->sin(x^2+y^2):plot3d(f, -2..2, -2..2); #utilizando  
unha función
```



Gráficos en 3 dimensións utilizando **funcións paramétricas** dependentes de 2 parámetros u e v , superficies dadas polas ecuacións: $x = x(u, v)$, $y = y(u, v)$, $z = z$

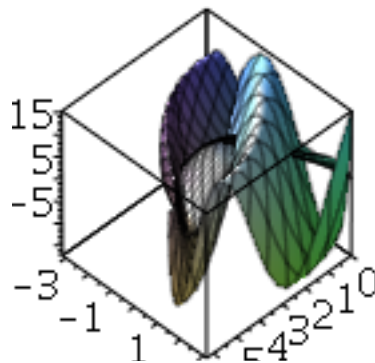
(u, v)

```
> plot3d([u*sin(v^2), u*cos(v^2), u*sin(v)], u=0..2*Pi, v=0..Pi); #con expresi3ns
```



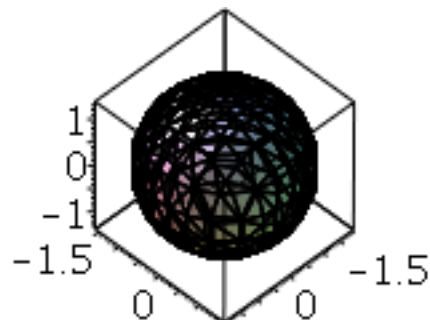
Se pinchas co rato no gr3fico activase o men3 Plot e podes facer cousas co gr3fico (rotalo, cambia-lo recheo, os eixos, entre outros).

```
> f:=(x,y)->x*sin(y); g:=(x,y)->y*cos(x); h:=(x,y)->x*y*sin(x*y);
plot3d([f,g,h], 0..2*Pi, 0..Pi);
f:=(x,y)->x sin(y)
g:=(x,y)->y cos(x)
h:=(x,y)->x y sin(x y)
```



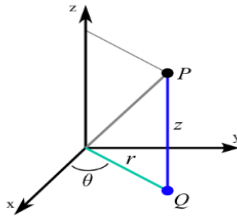
Superficies en 3D en forma implícita:

```
> restart:with(plots):implicitplot3d(2*x^2+3*y^2+4*z^2=8, x=-2..2, y=-1.7..1.7, z=-2..2);
```



Xa que $2 \cdot x^2 + 3 \cdot y^2 + 4 \cdot z^2 = 8$ é a ecuaci3n dun elipsoide en 3D

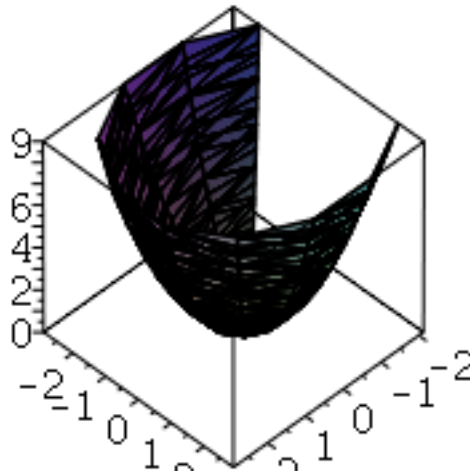
Superficies en coordenadas cilíndricas: As coordenadas cilíndricas (ρ, θ, z) teñen o significado mostrado nesta figura, e as ecuacións de transformación a coord. cartesianas móstranse na dereita:



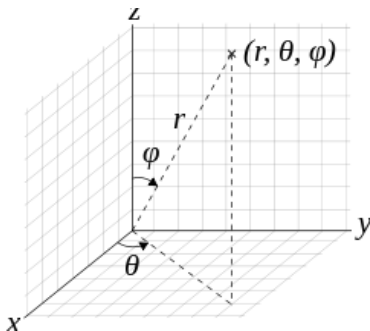
$$\begin{aligned}x &= \rho \cos \theta \\y &= \rho \sin \theta \\z &= z\end{aligned}$$

$$\begin{aligned}\rho &= \sqrt{x^2 + y^2} \\ \theta &= \arctan\left(\frac{y}{x}\right) \\ z &= z\end{aligned}$$

> `implicitplot3d(rho^2=z, rho=0..3, theta=-3*Pi/4..3*Pi/4, z=0..9, coords=cylindrical);`



Superficies en coordenadas esféricas (r, θ, ϕ) . As coordenadas esféricas teñen o significado mostrado nesta figura, e as ecuacións de transformación a coord. cartesianas móstranse na dereita:



$$\begin{aligned}x &= \rho \sin(\phi) \cos(\theta) \\y &= \rho \sin(\phi) \sin(\theta) \\z &= \rho \cos(\phi)\end{aligned}$$

$$\begin{aligned}\rho &= \sqrt{x^2 + y^2 + z^2} \\ \theta &= \arctan\left(\frac{\sqrt{x^2 + y^2}}{z}\right) \\ \phi &= \arctan\left(\frac{y}{x}\right)\end{aligned}$$

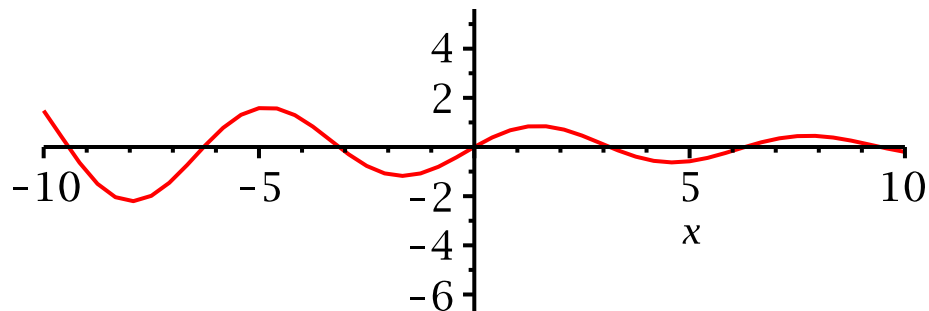
> `implicitplot3d(rho=sin(10*phi)*2^(theta/2), theta=-2*Pi..2*Pi, phi=0..Pi, rho=0..5, coords=spherical, axes=NORMAL);`



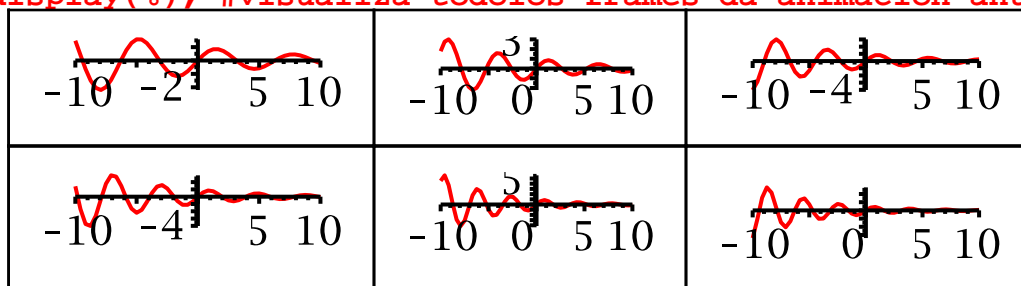
Animacións

MAPLE realiza animacións en 2D e 3D. Unha animación representa unha función que varía no tempo ou con algún parámetro. En 2D a sintaxe básica é **animate(F, rango_x, rango_t)** onde **F(x,t)** é a función a animar, **rango_x** o rango da variábel **x** e **rango_t** o rango do parámetro de animación **t** (o tempo). Posteriormente, pódese modifica-las características da animación interactivamente e ve-la animación como se fose un vídeo. Cando se pulsa co rato no gráfico, actívase o menú **Animation** (xunto con **Plot** e **Drawing**) e pódese reproducir-lo vídeo e configura-la reprodución (velocidade, etc.).

```
> animate(exp(-0.1*x*t)*sin(x*t), x=-10..10, t=1..2, frames=6)
;
```

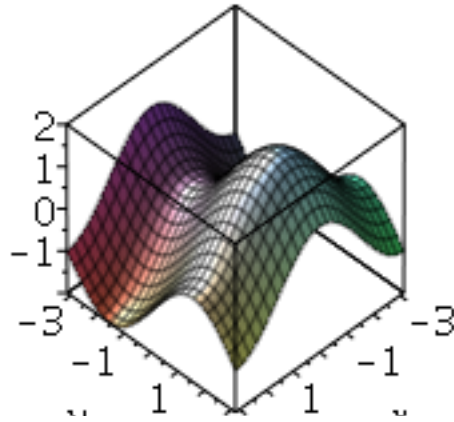


```
> display(%); #visualiza tódolos frames da animación anterior
```



No caso de 3D é análogo utilizando a función **animate3d(F, x, y, t)** pero especificando o rango de dúas variábeis **x** e **y**.

```
> animate3d(cos(t*x)+sin(t*y), x=-Pi..Pi, y=-Pi..Pi, t=1..2,
frames=4);
```



> display(%) # visualiza os 4 frames da animación anterior.

Táboa resumo dos gráficos 2D e 3D

Finalmente, aquí tedes esta gráfica para resumir os tipos de gráficos:

Tipo de Gráfico	Ecuación	Función en Maple
Gráficos en 2D		
Curva en forma explícita (función real de variábel real)	$y = f(x)$	<code>plot(f(x),x=a..b)</code>
Curva en forma paramétrica	$x = x(t), y = y(t)$	<code>plot([x(t),y(t),t=a..b])</code>
Vector de puntos	$(x_1, y_1) \dots (x_n, y_n)$	<code>plot(Vector([x₁, ..., x_n]), Vector([y₁, ..., y_n]))</code>
Curva en coordenadas polares (r, θ)	$\rho = \rho(\theta)$	<code>polarplot(r(q), theta=a..b)</code>
Curva en forma implícita	$f(x, y) = 0$	<code>implicitplot(f(x, y) = 0, x = a..b, y = a..b)</code>
Inecuacións	$f_1(x, y) > 0, \dots, f_n(x, y) > 0$	<code>inequal({f₁(x, y) > 0, ..., f_n(x, y) > 0}, x = a..b, y = a..b)</code>
Mapa de calor	$z = f(x, y)$	<code>contourplot(f(x,y),x=a..b,y=a..b, filledregions=true)</code>
Curva animada	$y = f(x, t)$	<code>animate(f(x, t), x = a..b, t = a..b, frames = n)</code>
Gráficos en 3D		

Superficie en forma explícita	$z = f(x, y)$	<code>plot3d(f(x, y), x = a..b, y = a..b)</code>
Superficie en forma paramétrica	$x = x(u, v), y = y(u, v), z = z(u, v)$	<code>plot3d([x(u, v), y(u, v), z(u, v)], u = a..b, v = a..b)</code>
Curva en forma paramétrica	$x = x(t), y = y(t), z = z(t)$	<code>spacecurve([x(t), y(t), z(t)], t = a..b, numpoints = n)</code>
Superficie en forma implícita	$f(x, y, z)=0$	<code>implicitplot3d(f(x, y, z) = 0, x = a..b, y = a..b, z = a..b)</code>
Superficie en coordenadas cilíndricas (r, q, z)	$f(r, q, z)=0$	<code>implicitplot3d(f(r,q,z), r=a..b,q=a..b, z=a..b,coords=cylindrical)</code>
Superficie en coordenadas esféricas (r, q, ϕ)	$f(r, q, \phi)=0$	<code>implicitplot3d(f(r,q,phi), r = a..b, q=a..b, phi = a..b, coords = spherical)</code>
Superficie animada	$z = f(x, y, t)$	<code>animate3d(f(x, y, t), x = a..b, y = a..b, t = a..b, frames = n)</code>