

# Imágenes y gráficas en MATLAB

Sistemas Conexionistas - Curso 07/08

---

## 1. Imágenes

Matlab dispone de comandos especiales para trabajar con imágenes de diversos formatos (pgm, ppm, gif, png, ...).

**imread** Lee una imagen en color o escala de grises.

```
IM = imread(filename)
```

La imagen es almacenada en la matriz **IM**. Las imágenes en escala de grises se almacenan en matrices bidimensionales del mismo tamaño que la imagen, mientras que en las imágenes en color, la matriz es tridimensional ya que cuenta con una submatriz del mismo tamaño que la imagen en cada canal RGB.

Si los colores de la imagen están indexados (por ejemplo las imágenes con formato gif), la lectura de la imagen se realiza de la siguiente forma:

```
[IM map] = imread(filename)
```

donde **map** es una matriz que almacena el mapa de colores de la imagen leída.

**imwrite** Escribe una imagen a fichero.

```
imwrite(IM, filename)
```

Si el fichero incluye la extensión no es necesario indicar el formato de la imagen.

Para mostrar una imagen por pantalla, Matlab dispone de las funciones **image** e **imagesc**. Su sintaxis es:

```
image(IM)
imagesc(IM)
```

donde IM se corresponde con una matriz que representa una imagen. Si no se especifica ningún mapa de colores, Matlab utilizará el mapa por defecto. Matlab dispone de varios mapas de colores aunque también es posible utilizar un mapa de colores leído en una imagen o definido por el usuario. Para especificar el mapa de colores se utiliza el comando `colormap`:

```
colormap(map)
```

## 2. Gráficas

### 2.1. Funciones en dos dimensiones

Para dibujar funciones de una variable, Matlab dispone de la función `plot`, con la siguiente sintaxis:

- `plot(Y)` Dibuja la gráfica que toma los índices de Y como valores en el *eje x* y las columnas de Y como valores en el *eje y*.
- `plot(x1,y1, ...)` Une con una línea todos los pares de la forma  $(x_n, y_n)$
- `plot(X,Y)` Si X e Y son vectores, dibuja las columnas/filas de X con respecto a las columnas/filas de Y.

`plot` dispone de argumentos opcionales que permiten especificar el tipo de línea que se dibuja, así como su color y el tipo de marcador para los puntos dibujados:

- `plot(X,Y, '--r')` Dibuja una línea discontinua en rojo
- `plot(X,Y, 's')` Dibuja sólo los puntos como cuadrados
- `plot(X,Y, '+r')` Dibuja sólo los puntos como cruces en rojo
- `plot(X,Y, 'g')` Dibuja una línea continua en verde

Es posible configurar ciertos aspectos de las gráficas, entre otros:

- `title(string)` Establece la cadena `string` como título de la gráfica
- `xlabel(string)` Establece la cadena `string` como etiqueta del eje x de la gráfica
- `ylabel(string)` Establece la cadena `string` como etiqueta del eje y de la gráfica
- `axis` Controla la apariencia y escala de los ejes. Presenta varias opciones

### Manipulación de ventanas de figura

Al utilizar la función `plot`, Matlab abre automáticamente una ventana de figura que se sobrescribe si se vuelve a llamar a otra función de dibujo. El comando `figure` permite abrir nuevas ventanas de figura. Su sintaxis es:

```
h = figure
```

`figure` devuelve un identificador que se utiliza para manipular la figura. Así, si tenemos varias ventanas de figura abiertas, con `figure(h2)` seleccionamos la figura relativa al identificador `h2` y con `close(h3)` cerramos la ventana `h3`. `close all` cierra todas las ventanas de figura abiertas.

### Subfiguras

El comando `subplot` permite dibujar varias gráficas en una misma ventana de figura. Su sintaxis es la siguiente:

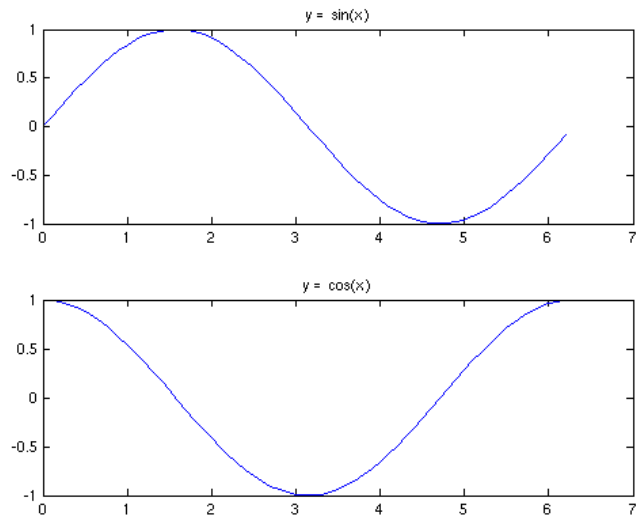
```
subplot(m,n,p)
```

Con este comando, la ventana de figura actual se descompone en una matriz de  $m \times n$  subfiguras y se selecciona la subfigura número  $p$  para ser utilizada.

```

x = 0:0.1:2*pi;
y1 = sin(x);
y2 = cos(x);
subplot(2,1,1)
plot(x,y1)
title('y = sin(x)');
subplot(2,1,2)
plot(x,y2)
title('y = cos(x)');

```



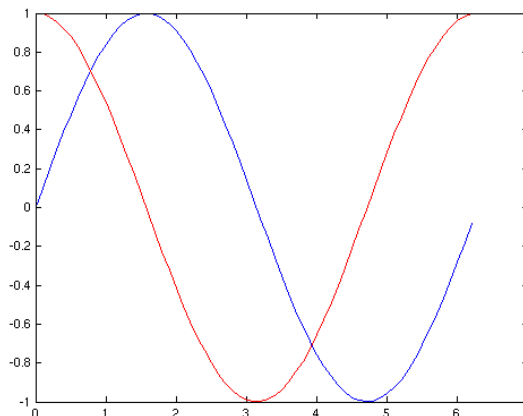
## Superposición de gráficas

Para dibujar varias gráficas en una misma figura/subfigura, se utiliza la función `hold on`. Para dejar de dibujar en la misma figura/subfigura, se utiliza `hold off`.

```

x = 0:0.1:2*pi;
y1 = sin(x);
y2 = cos(x);
plot(x,y1)
hold on;
plot(x,y2, 'r')
hold off;

```



## 2.2. Funciones en tres dimensiones

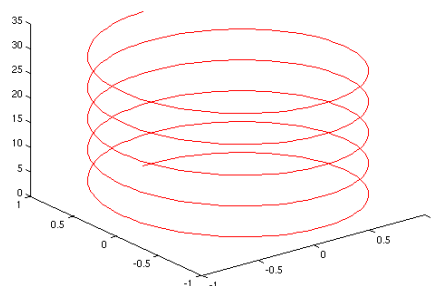
### Gráficas de líneas/puntos

La función `plot3` dibuja líneas y puntos en 3D. Su sintaxis es similar a la función `plot`:

- `plot3(x,y,z)` Dibuja una línea en el espacio 3D a través de los puntos cuyas coordenadas son los elementos de los vectores  $x,y,z$ .

- `plot3(x,y,z,linedef)` Dibuja una línea en el espacio 3D a través de los puntos cuyas coordenadas son los elementos de `x,y,z` teniendo en cuenta las características definidas en `linedef`.

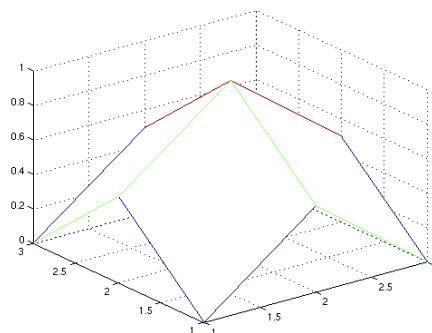
```
t = 0:pi/50:10*pi;
plot3(sin(t),cos(t),t, 'r');
```



## Gráficas de superficies

La función `mesh` dibuja superficies 3D a partir de matrices. El comando `mesh(Z)` dibuja los valores de la matriz `Z` sobre sus coordenadas. Los valores de `Z` se muestran como la altura sobre la rejilla `xy`.

```
Z = [0 0.5 0; 0.5 1 0.5; 0 0.5 0]
mesh(Z)
```



Con `mesh(X,Y,Z)` se muestra la matriz `Z` sobre las coordenadas de la rejilla especificada por las matrices `X` e `Y`. Para crear estas matrices se utiliza el comando `meshgrid`:

```
[X, Y] = meshgrid(x,y)
```

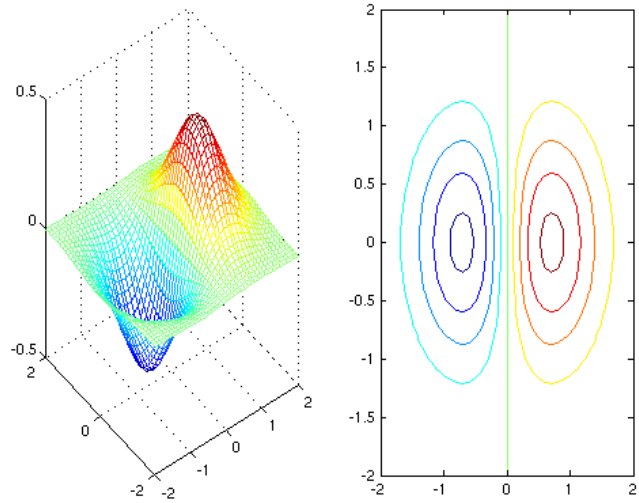
donde `x` e `y` son los vectores base a partir de los cuales se construye la rejilla. Las filas de la matriz de salida `X` son copias del vector `x` y las columnas de la matriz de salida `Y` son copias del vector `y`.

La función `contour` dibuja las curvas de nivel de la superficie pasada como parámetro. Se usa de forma similar a `mesh`.

```

x = -2:0.1:2;
y = -2:0.1:2;
[X,Y] = meshgrid(x,y);
Z = X.*exp(-X.^2 - Y.^2)
subplot(1,2,1);
mesh(X,Y,Z);
subplot(1,2,2);
contour(X,Y,Z);

```



### 3. Ejercicios

1. Dibujar la función  $y = x * e^{-x^2}$  con x variando entre -5 y 5 a intervalos de 0.05
2. Dibujar en una sola ventana con dos subventanas las funciones  $y = x^2 - 3x - 2$  y  $z = x^3 - 2x + 1$  con x variando entre -5 y 5 a intervalos de 0.02. La primera en líneas verdes continuas, la segunda en líneas azules discontinuas. Marcar con un '+' rojo el punto (2,1) en ambas gráficas.
3. Dibujar la superficie de la función  $z = \sin(x) + \cos(y)$  donde x e y varían entre -5 y 5 a intervalos de 0.1.