

Entrada/Salida

Fdi, Lenguajes y Sistemas Informáticos

Lenguajes y Sistemas Informático

2021/2022



GASTEIZKO
INGENIARITZAKO
UNIBERTSITATE ESKOLA
ESCUELA UNIVERSITARIA
DE INGENIERÍA
DE VITORIA-GASTEIZ

Contenidos

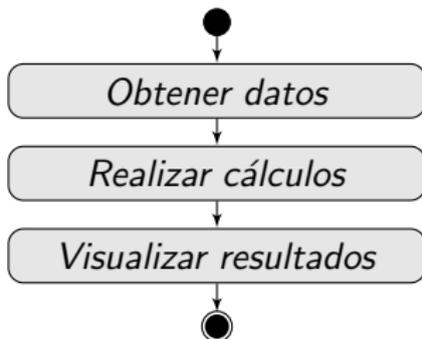
- 1 **Introducción**
- 2 **Carga de datos de ficheros**
- 3 **Gráficas**

Contenidos

- 1** **Introducción**
- 2 Carga de datos de ficheros
- 3 Gráficas

Estructura básica de los programas

Gran parte de los programas, sobre todo en Matlab/Octave, tienen esta estructura:



Contenidos

- 1 Introducción
- 2 Carga de datos de ficheros**
- 3 Gráficas

Leer datos de un fichero

Necesidad de usar ficheros

Muchas veces necesitamos procesar una elevada cantidad de datos. Los ficheros o archivos permiten almacenar y compartir datos. Vamos a usar los siguientes tipos de ficheros:

- Ficheros con formato CSV
- Hojas de cálculo (.xls, .xlsx, .ods)

Formato de los ficheros

Ficheros CSV

- Son ficheros con "*valores separados por comas*"
- Pueden tener en las primeras líneas una cabecera que describe los campos (significado del contenido de las columna)
- Cada línea representa una fila en la matriz
- Las columnas se separan por comas u otro carácter especial como los punto y coma (;) o tabuladores)

Hojas de cálculo

Una hoja de cálculo (*spreadsheet* o *worksheet* en inglés) es un fichero ofimático organizado en una tabla con filas y columnas para gestionar datos y calcular otros.

Las hojas de cálculo se llaman a veces libros porque manejan más de una hoja mediante pestañas, con el fin de guardar varias matrices relacionadas.

Lectura de datos de un CSV

Lectura de una matriz de un CSV

Para leer los datos de un CSV en Matlab vamos a usar la función **readmatrix**, que devuelve una matriz. Esta función lee el fichero, identifica cuál es el carácter delimitador de columnas y detecta y elimina la cabecera si hay.

Sintaxis

readmatrix (*fichero*)

Example

```
m=readmatrix('data.csv');
```

Lectura parcial de datos de un CSV

Lectura de una parte de la matriz

Se puede usar `readmatrix` para leer una submatriz desde un fichero si le indicamos las filas que queremos leer.

Sintaxis

```
readmatrix (file, 'Range', rango)
```

Ejemplo

```
m=readmatrix('data.csv', 'Range',  
[2 3 4 7]);
```

Nota

Los números de fila tienen en cuenta la cabecera que pudiera tener. Es decir, si el CSV tiene una línea de cabecera, la fila 1 es la cabecera.

Lectura de datos de una hoja de cálculo

Lectura de la matriz

La función `readmatrix` puede usarse para leer datos de una hoja de cálculo (`.xls`, `.xlsx`, `.ods`). En este caso, al no indicar de qué hoja, por defecto **devuelve los datos que contiene la primera hoja del fichero de hoja de cálculo.**

Sintaxis

`readmatrix` (*file*)

Example

```
m=readmatrix('data.xlsx');
```

Lectura parcial de una hoja de cálculo

Lectura parcial de la matriz

readmatrix se puede usar para leer una parte de una hoja de cálculo. Se puede usar un rango de filas o las coordenadas de inicio y de final de la parte de la hoja que queremos cargar.

Sintaxis

readmatrix (*file*, 'Range', *rango*)

Ejemplo

```
m=readmatrix('data.xlsx', 'Range',  
'C2:D4');
```

Nota

Las hojas de cálculo identifican las celdas uniendo la letra de la columna y el número de línea, por ejemplo C2. El rango se debe especificar en una cadena que contenga las celdas de inicio y final, separadas por dos puntos (:), por ejemplo 'C2:D4'

Lectura de otra hoja de la hoja de cálculo

Selección de hoja

`readmatrix` puede leer una hoja distinta de la primera del fichero si se le indica.

Sintaxis

`readmatrix` (*file*, 'Sheet', *nombre de la hoja*)

Ejemplo

```
m=readmatrix('data.xlsx', 'Sheet',  
             'Sheet2');
```

Nota

Se puede combinar la opción de seleccionar hoja con la de seleccionar rango para cargar una parte de la hoja.

Contenidos

- 1 Introducción
- 2 Carga de datos de ficheros
- 3 Gráficas**

Gráficas

La función `plot`

En MatLab se pueden crear gráficas con la función `plot`.

Syntax

`plot (Y)`

`plot (Y, format)`

`plot (X, Y)`

`plot (X, Y, format)`

`plot (X1, Y1 , Xn, Yn)`

`plot (X1, Y1, format1 , Xn, Yn, formatn)`

Example

```
plot(rand(1,10));  
plot(1:5, 'b-');  
plot(1:5, rand(1,5), 'g*-')  
plot(1:5, rand(1,5), 'rp', '  
MarkerSize', 20);
```

Nota

Si sólo se especifica un vector, se considera que son los datos de las ordenadas de la gráfica (eje Y). El vector de las abscisas (eje X) se genera con `1:length(Y)`.

Formato (l)

'b-'

Color

Tipo de marcador

Character	Description
y	Yellow
m	Magenta
c	Cyan
r	Red
g	Green
b	Blue
w	White
k	Black

Carácter	Descripción
-	Línea continua
- -	Línea intermitente
:	Línea de puntos
-.	Línea combinada punto-línea
o	Círculos
+	Cruces
*	Asteriscos
.	Puntos

Ver más opciones: https://es.mathworks.com/help/matlab/creating_plots/create-line-plot-with-markers.html

Formato (II)

Color $\left| \begin{matrix} r \\ p \end{matrix} \right|$ Tipo de marcador

Character	Description
y	Yellow
m	Magenta
c	Cyan
r	Red
g	Green
b	Blue
w	White
k	Black

Character	Description
o	Circle
.	Point
*	Asterisk
x	Cross
s	Square
d	Diamond
^	Upward-pointing triangle
v	Downward-pointing triangle
>	Right-pointing triangle
<	Left-pointing triangle
p	Pentagram
h	Hexagram

Título de la gráfica

Título

Para describir e identificar el significado de la gráfica es necesario usar la función `title` tras usar `plot` para generarla. Si se usa antes, es necesario usar `hold on` para que no desaparezca.

Sintaxis

title (*text*)

Ejemplo

```
title('Func. Seno');
```

Etiquetas de los ejes

Ejes

Con el objetivo de clarificar el significado de la gráfica, vamos a identificar los ejes con un texto (por ejemplo, con tipo de dato y unidades) usando estas funciones: `xlabel` y `ylabel`

Syntax

```
xlabel (text)
```

```
ylabel (text)
```

Adibidea

```
xlabel('Punto');
```

```
ylabel('Altura (m)');
```

Leyenda

Leyenda

Cuando se visualizan o grafican varias funciones o datos en la misma gráfica, es necesario identificar cada línea con su significado. Automáticamente relaciona colores con el orden de graficado.

Sintaxis

```
legend (text1 ... , textn)
```

Ejemplo

```
legend('Sine' , 'Cosine');
```

Límites en ejes

Límites en ejes

Podemos cambiar el rango de valores que se visualizan en cada eje con la función `axis` si no queremos que se muestre de forma automática toda la gráfica.

Sintaxis

`axis ([x_{min} x_{max} y_{min} y_{max}])`

Ejemplo

```
axis([1 10 -2 2]);
```

Etiquetas de los valores en los ejes

Las etiquetas de los valores en los ejes se generan automáticamente en función de los datos mostrados, pero los podemos modificar con estas instrucciones.

Sintaxis

```
xticks ([values])  
xticklabels ([labels])  
yticks ([values])  
yticklabels ([labels])
```

Ejemplo

```
xticks([1 2 3]);  
xticklabels(["A" "B" "C"]);
```

Note

Se deben especificar valores y etiquetas.

https://es.mathworks.com/help/matlab/creating_plots/change-tick-marks-and-tick-labels-of-graph-1.html

Guardar gráficas en fichero

Comando saveas

Guardar gráficas en fichero con el comando saveas

Ejemplo

```
%No se muestra la figura en pantalla  
fig = figure('visible', 'off');  
%Creamos la figura de forma oculta  
plot(time, height);  
%La guardamos en un fichero PNG  
saveas(fig, 'image.png');
```

Varias gráficas en una figura

subplot

Podemos crear una matriz de gráficas en una sola figura usando la función `subplot` de forma que se divide la figura en celdas. Con `subplot` elegimos la celda y con `plot` generamos la gráfica.

Sintaxis

`subplot (rowN, colN, pos)`

Ejemplo

```
%tabla de 1 fila, 2 columnas
subplot(1,2,1); %primera celda
plot(rand(1,10));
subplot(1,2,2); %segunda celda
plot(1:5, 'b-');
```

Nota

Una gráfica puede ocupar varias celdas de una figura con `subplot`.
`subplot(3,2,[1 3 5]);`

Varias funciones y datos en una gráfica

hold

Podemos visualizar varias funciones y datos en una gráfica con `hold`

Sintaxis

```
hold on  
hold off
```

Ejemplo

```
hold on;  
plot(rand(1,10));  
plot(1:5, 'b-');  
hold off;
```

Gráficas de barras

bar

Para crear gráficas de barras usaremos la función `bar` que, en su versión más sencilla, tiene esta forma.

Sintaxis

```
bar (Y)
```

```
bar (X, Y)
```

Ejemplo

```
bar(rand(1,10));
```

```
bar(years, profits);
```

Entrada/Salida

Fdl, Lenguajes y Sistemas Informáticos

Lenguajes y Sistemas Informático

2021/2022



GASTEIZKO
INGENIARITZAKO
UNIBERTSITATE ESKOLA
ESCUELA UNIVERSITARIA
DE INGENIERÍA
DE VITORIA-GASTEIZ