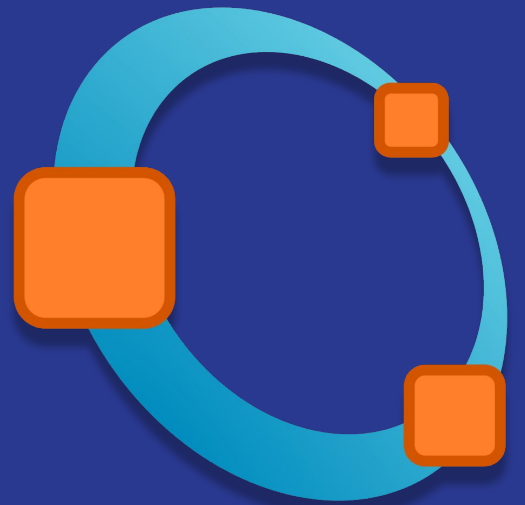


Manejo de Datos

- Tratamiento de matrices y vectores
- Plots



Creación de Vectores y Matrices

Asignación directa de datos:

```
# Vector
V = [1 2 4 5];
# Matriz
M = [1 2 3 4; 8 9 10 11];
```

Atajo:

```
# Vector
V = [1:5];
# Matriz
M = [1:4; 8:11];
```

Generación de vectores y matrices:

```
# Vector
serieA = zeros(1,6);
# Matriz
Mdatos = zeros(7,3);
```

```
# Vector
serieB = ones(1,6)*3;
# Matriz
Datos = ones(7,3)*100;
```

Subselección de datos | Ejemplo 1

Supongamos que tenemos una matriz llamada `datosTLBrujas` que recopila las temperaturas mínimas, medias y máximas para una semana dada en la estación Las Brujas. Ordenadas en fecha creciente (función `flip()`).

Necesitamos separar esta información en tres vectores individuales

Temperatura mínima, media y máxima

datosTLBrujas

18.5	21.7	27.2
17.5	25.1	33.4
20.4	27.6	34.4
20.1	24.6	33.8
15.1	20.2	23.8
11.4	20.1	27.1
14	21.8	29

Subselección de datos

Para hacer esto necesitamos hacer una subselección dentro de la matriz de datos.

```
# Vectores Temperaturas por semana
Tmin = datosTLBrujas(:,1);
Tmed = datosTLBrujas(:,2);
Tmax = datosTLBrujas(:,3);
```

¿Y si quisiéramos generar vectores que recopilen la temperatura mínima media y máxima para cada día?

```
# Vectores Temperaturas por semana
TJuev = datosTLBrujas(4,:);
```

datosTLBrujas

18.5	21.7	27.2
17.5	25.1	33.4
20.4	27.6	34.4
20.1	24.6	33.8
15.1	20.2	23.8
11.4	20.1	27.1
14	21.8	29

TJuev

Tmin

Editar Filas o Columnas Completas | Ejemplo 2

La segunda columna de la matriz representa la temperatura media diaria, si queremos **reemplazar** toda esta columna por los valores de temperatura promedio $(T_{\max} + T_{\min}) / 2$. Un camino es el siguiente:

```
# Vectores Temperaturas por semana
Tmin = datosTLBrujas(:,1);
Tmax = datosTLBrujas(:,3);

# Se calcula y asigna el promedio
datosTLBrujas(:,1) = (Tmax + Tmin)/2
```

datosTLBrujas

18.5	21.7	27.2
17.5	25.1	33.4
20.4	27.6	34.4
20.1	24.6	33.8
15.1	20.2	23.8
11.4	20.1	27.1
14	21.8	29

Editar Filas o Columnas Completas | Ejemplo 2

Si no quisiéramos perder la matriz original, podemos guardar el resultado en una nueva matriz

```
# Se calcula y asigna el promedio
# en la misma matriz
datosTLBrujas(:,1) = (Tmax + Tmin)/2

# Se calcula y asigna el promedio
# en otra matriz
datosTLBrujas2 = datosTLBrujas
datosTLBrujas2(:,2) = (Tmax + Tmin)/2
```

datosTLBrujas		
18.5	21.7	27.2
17.5	25.1	33.4
20.4	27.6	34.4
20.1	24.6	33.8
15.1	20.2	23.8
11.4	20.1	27.1
14	21.8	29

datosTLBrujas2		
18.5	22.9	27.2
17.5	25.5	33.4
20.4	27.4	34.4
20.1	27.0	33.8
15.1	19.5	23.8
11.4	19.3	27.1
14	21.5	29

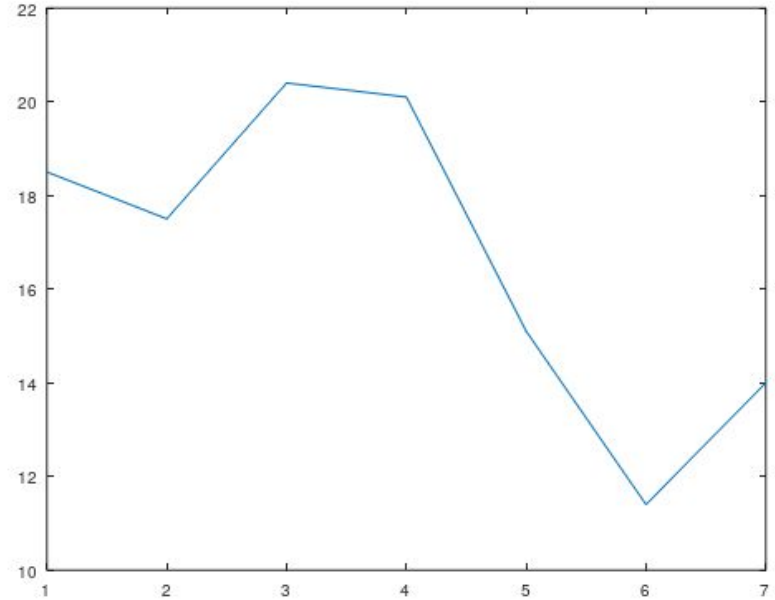
Gráficas

Existen diferentes tipos de salidas gráficas soportadas. Las primeras a conocer son: de línea, puntos y barras.

Para gráficas de líneas y puntos se utiliza la función `plot()`.

Para graficar los datos de temperatura media, el segmento de código más simple que podemos usar es el siguiente:

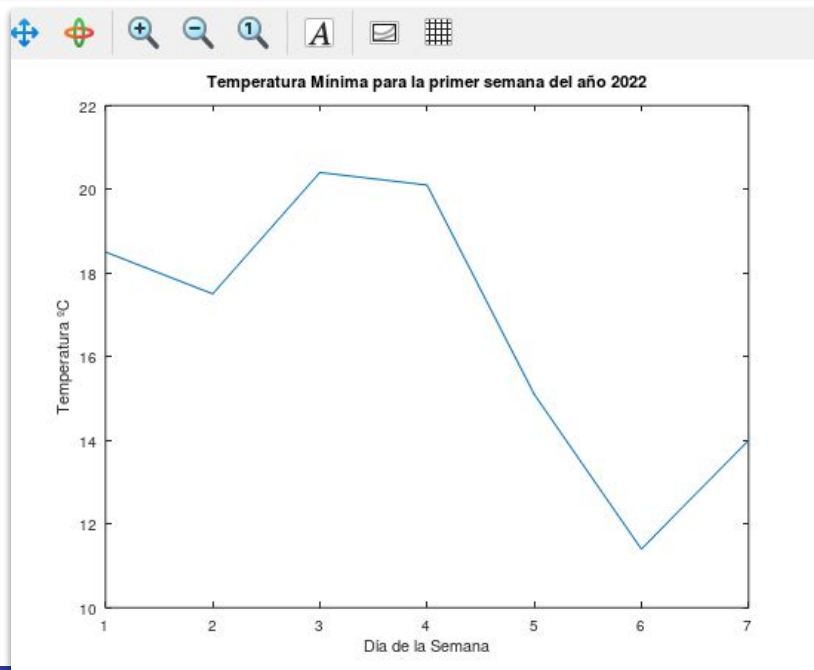
```
plot(Tmed)
```



Gráficas

Si a la función `plot` simplemente le pasamos el vector de datos `Tmed`, Octave dibuja una gráfica donde el eje de abscisas asigna el ordinal de cada elemento del vector `Tmed` y en el eje de ordenadas el respectivo valor de temperatura para cada ordinal.

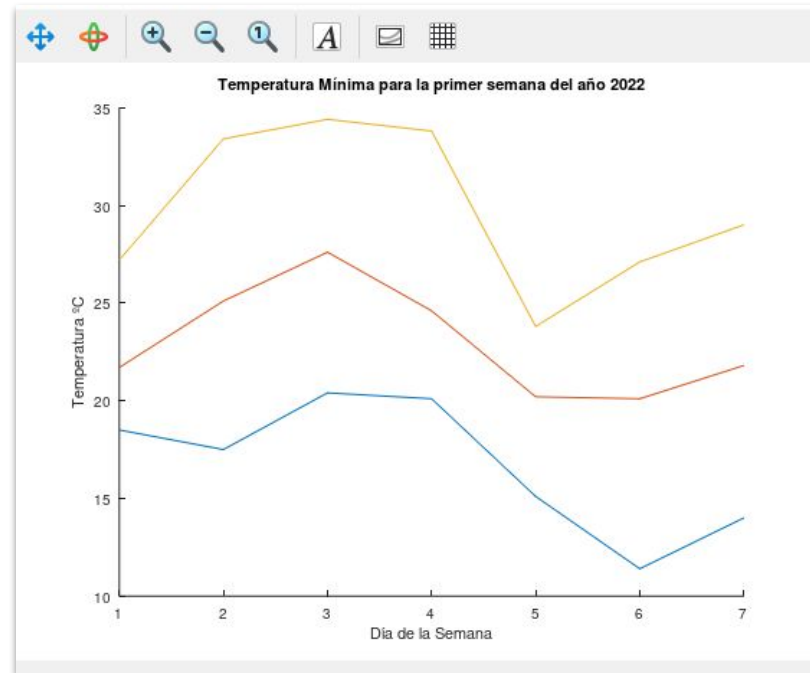
```
plot(Tmin);  
title('Tmin para la primer semana del año  
2022')  
xlabel('Día de la Semana')  
ylabel('Temperatura °C')
```



Superposición de gráficas

Para superponer varias gráficas debemos utilizar el comando `hold on`. Este comando permite superponer varias gráficas en una misma figura.

Si nuestro trabajo pretende mostrar varias gráficas es necesario además indicar una identidad para cada figura realizada.



```

# Una gráfica sola
figure(1)
plot(Tmin);
title('Tmin para la primer semana del año 2022' )
xlabel('Día de la Semana')
ylabel('Temperatura °C')

# Superposiición de varias gráficas
figure(2);
hold on;
plot(Tmin);
plot(Tmed);
plot(Tmax);
title('Tmin, Tmed y Tmax. 1er semana del año
2022')
xlabel('Día de la Semana')
ylabel('Temperatura °C')
legend('Tmin', 'Tmed', 'Tmax')

```

