

UNIDAD 1 - SEÑALES Y SISTEMAS (Señales Continuas y Discretas)

Introducción a Octave

Señales en tiempo continuo. Introducción a Octave

Octave

File Edit Debug Tools Window Help News

Current Directory: al\Análisis Numerico\Octave\Ejercicios Clases

File Browser

C:/Users/conherrera/OneDrive - Proofpoint, Inc/Documents/Personal/Análisis Numerico/Octave/Ejercicios Clases

Name

Prueba_Andres (1).m

Prueba_Andres (2).m

PruebaCoti.m

Workspace

filter

Name	Class	Dimension	Value	Attribute
a	double	1x1	0	
b	double	1x20	[2, -5.8795e-15, ...	
i	double	1x20	1:20	
j	double	1x30	1:30	
n	double	1x20	20:-1:1	
x	double	1x20	[2, -5.8795e-15, ...	
y	double	1x20	[40, 38, 36, 34, 3...	
z	double	1x30	[0, 0, 0, 0, 0, 0...	

Command History

filter

OctaveFunctionEscalonContinuo

OctaveFunctionEscalonContinuo

Octave 8.2.0, Fri Jul 07 15:07:01 2023 GMT <unknown@B2Q5163>

Octave 8.2.0, Wed Aug 23 10:48:49 2023 GMT <unknown@B2Q5163>

PruebaCoti

PruebaCoti

PruebaCoti

PruebaCoti

PruebaCoti

PruebaCoti

PruebaCoti

Editor

File Edit View Debug Run Help

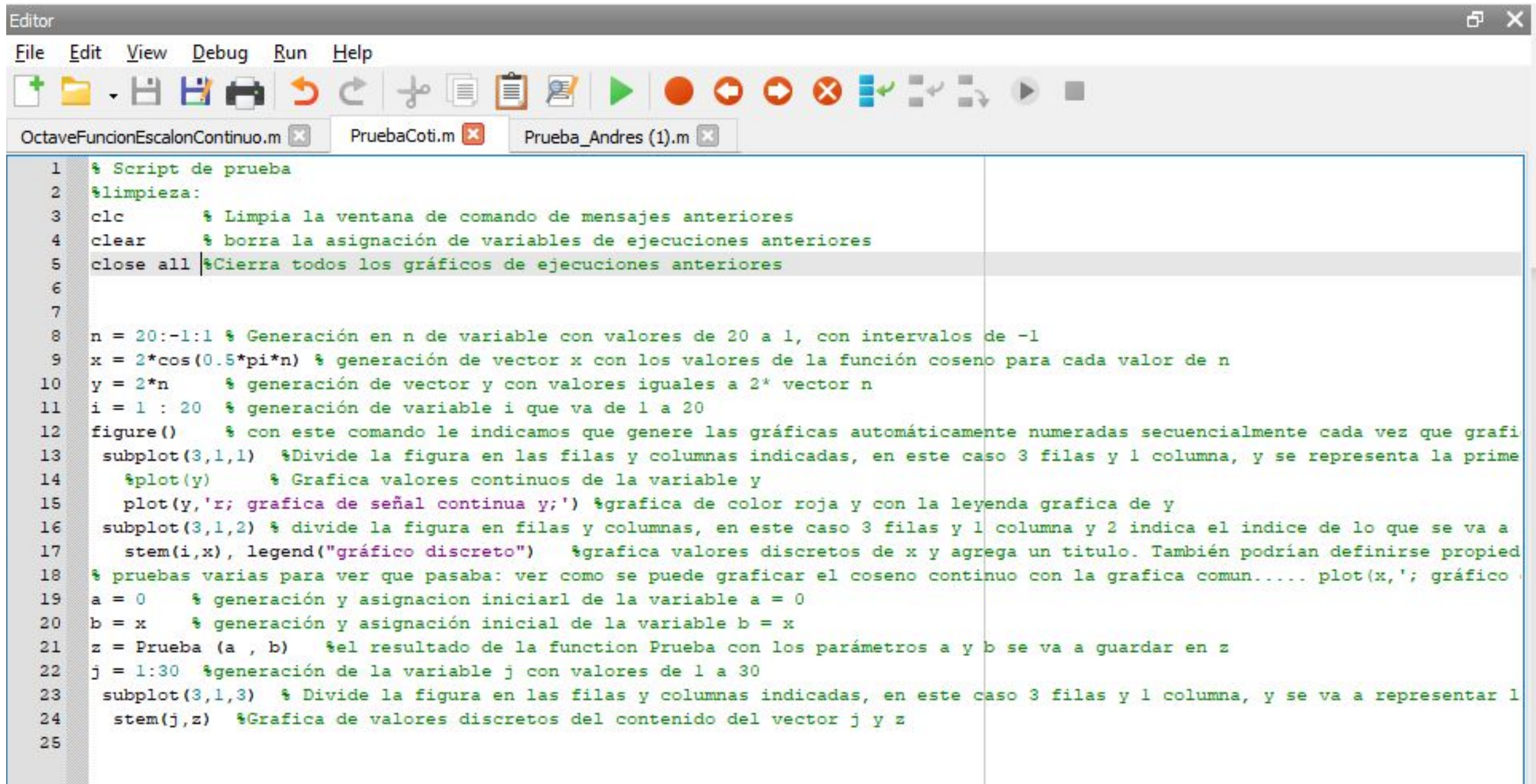
OctaveFunctionEscalonContinuo.m PruebaCoti.m Prueba_Andres (1).m

```
1 % Script de prueba
2 %limpieza:
3 clc % Limpia la ventana de comando de mensajes anteriores
4 clear % borra la asignación de variables de ejecuciones anteriores
5 close all %Cierra todos los gráficos de ejecuciones anteriores
6
7
8 n = 20:-1:1 % Generación en n de variable con valores de 20 a 1, con intervalos de -1
9 x = 2*cos(0.5*pi*n) % generación de vector x con los valores de la función coseno para cada valor de n
10 y = 2*n % generación de vector y con valores iguales a 2* vector n
11 i = 1 : 20 % generación de variable i que va de 1 a 20
12 figure() % con este comando le indicamos que genere las gráficas automáticamente numeradas secuencialmente cada vez que grafi
13 subplot(3,1,1) %Divide la figura en las filas y columnas indicadas, en este caso 3 filas y 1 columna, y se representa la prime
14 %plot(y) % Grafica valores continuos de la variable y
15 plot(y,'r','r; grafica de señal continua y;') %grafica de color roja y con la leyenda grafica de y
16 subplot(3,1,2) % divide la figura en filas y columnas, en este caso 3 filas y 1 columna y 2 indica el indice de lo que se va a
17 stem(i,x), legend("gráfico discreto") %grafica valores discretos de x y agrega un titulo. También podrían definirse propied
18 % pruebas varias para ver que pasaba: ver como se puede graficar el coseno continuo con la grafica comun..... plot(x,'r; gráfico
19 a = 0 % generación y asignación inicial de la variable a = 0
20 b = x % generación y asignación inicial de la variable b = x
21 z = Prueba(a , b) %el resultado de la function Prueba con los parámetros a y b se va a guardar en z
22 j = 1:30 %generación de la variable j con valores de 1 a 30
23 subplot(3,1,3) % Divide la figura en las filas y columnas indicadas, en este caso 3 filas y 1 columna, y se va a representar 1
24 stem(j,z) %Grafica de valores discretos del contenido del vector j y z
25
```

line: 5 col: 11 encoding: UTF-8 eol: CRLF

Command Window Documentation Variable Editor Editor

Señales en tiempo continuo. Introducción a Octave



The image shows the Octave Editor interface. At the top is a menu bar with 'File', 'Edit', 'View', 'Debug', 'Run', and 'Help'. Below the menu is a toolbar with various icons for file operations, editing, and execution. The editor window has three tabs: 'OctaveFuncionEscalonContinuo.m', 'PruebaCoti.m', and 'Prueba_Andres (1).m'. The active tab is 'Prueba_Andres (1).m', which contains the following MATLAB script:

```
1 % Script de prueba
2 %limpieza:
3 clc      % Limpia la ventana de comando de mensajes anteriores
4 clear    % borra la asignación de variables de ejecuciones anteriores
5 close all %Cierra todos los gráficos de ejecuciones anteriores
6
7
8 n = 20:-1:1 % Generación en n de variable con valores de 20 a 1, con intervalos de -1
9 x = 2*cos(0.5*pi*n) % generación de vector x con los valores de la función coseno para cada valor de n
10 y = 2*n      % generación de vector y con valores iguales a 2* vector n
11 i = 1 : 20    % generación de variable i que va de 1 a 20
12 figure()     % con este comando le indicamos que genere las gráficas automáticamente numeradas secuencialmente cada vez que grafi
13 subplot(3,1,1) %Divide la figura en las filas y columnas indicadas, en este caso 3 filas y 1 columna, y se representa la prime
14     %plot(y)      % Grafica valores continuos de la variable y
15     plot(y,'r; grafica de señal continua y;') %grafica de color roja y con la leyenda grafica de y
16 subplot(3,1,2) % divide la figura en filas y columnas, en este caso 3 filas y 1 columna y 2 indica el indice de lo que se va a
17     stem(i,x), legend("gráfico discreto") %grafica valores discretos de x y agrega un titulo. También podrían definirse propied
18 % pruebas varias para ver que pasaba: ver como se puede graficar el coseno continuo con la grafica comun..... plot(x,'; gráfico
19 a = 0      % generación y asignacion inicial de la variable a = 0
20 b = x      % generación y asignación inicial de la variable b = x
21 z = Prueba(a , b) %el resultado de la function Prueba con los parámetros a y b se va a guardar en z
22 j = 1:30   %generación de la variable j con valores de 1 a 30
23 subplot(3,1,3) % Divide la figura en las filas y columnas indicadas, en este caso 3 filas y 1 columna, y se va a representar l
24     stem(j,z) %Grafica de valores discretos del contenido del vector j y z
25
```

Señales en tiempo continuo. Introducción a Octave

Recomendaciones para trabajar de forma prolija y limpia

Script

- Son los documentos en los que se elaboran los Programas en Octave

Function

- Programa de uso común a varios scripts.
- Las funciones predefinidas del programa son function

CLEAR

- Borra la asignación de variables que se hicieron en anteriores ejecuciones.

CLOSE

- Cierra todos los gráficos de ejecuciones anteriores

Clc

- Limpia la ventana de comando, de los mensajes anteriores

;

- Al terminar un renglón evita que se muestre el resultado en la ventana de comandos

Señales en tiempo continuo. Introducción a Octave

CICLOS

FOR

- **for** i=1:5
 sentencias
- **endfor**

CONDICIONAL IF

- **If** (condición lógica)
 sentencias
- **else o else if**
- **endif**

También existen otros como **while** y **switch**

Señales en tiempo continuo. Introducción a Octave

VARIABLES Y VECTORES



`n = 0 : 0.1 : 1`

genera en n una variable, de 0 a 1 con intervalos de valor 0.1



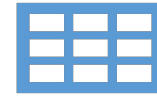
`x = 2 * cos(2 * pi * n)`

x es un vector con los valores de la función en cada valor de n definido en la variable de rango.



Para identificar elementos de un vector, podemos usar índices

`a = x (2:6)`
a contiene los valores de x entre 2 y 6

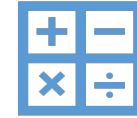


Ones y Zeros
Funciones que generan vectores (o matrices), con componentes nulas o unitarias

`a = ones (1,5)` Vector fila de 5 elementos¹unitarios.

`b = zeros (5,1)` Vector columna de 5 elementos nulos.

`c = zeros (2,6)` Matriz de 2x6 con elementos nulos.



Se pueden agrupar vectores para generar vectores compuestos

`b = [a , zeros(1,5), c]`

b es un vector fila, compuesto. Los vectores filas se colocan entre corchetes separados por comas

`c = [a ; b]`

c es una matriz, cuyas filas serían los vectores fila a y b. Las “,” separan filas, los “;” separan columnas en la definición del arreglo.

El 1s en los argumentos indica que son vectores.

Señales en tiempo continuo. Introducción a Octave

GRÁFICOS 2D: PLOT y STEM

- Los gráficos, se van generando en distintas figuras.
- Las figuras, podemos enumerarlas en el programa (script). Pero si cada vez que graficamos, anteponemos la sentencia **figure()**, el programa automáticamente las genera numeradas secuencialmente, con cada gráfico.

Señales en tiempo continuo. Introducción a Octave

GRÁFICOS 2D: PLOT (valores continuos)

- **Para graficar valores continuos**
 - **plot(x)** : Gráfica los valores de **x**. En la variable independiente coloca las posiciones **i** dentro del vector (**x(i) vs i**).
 - **plot(x,y)**: Grafica los valores del vector **y** en función de los del vector **x**.
 - **plot(-1:0.1:5,y)** . Los valores del eje **x**, pueden ser definidos en un vector o directamente como variables de rango. Siempre deben coincidir en la cantidad de elementos.
- **Se puede editar las propiedades/características gráficas**
 - Dando pares propiedad y valor: **plot(x,y,'linestyle','--','marker','o','markeredgecolor','g')**
 - Agregando cadena con 4 caracteres opcionales: **plot(x,y,'-- o g ; señal de entrada;')**
- **Se puede representar varias funciones en el mismo gráfico.**
 - Se colocan todos los pares **x,y** a continuación **plot(x1,y1,'-- o g ; señal de entrada; x2,y2,x1,y3)** (gráfico con 3 funciones). La condición es que sean de igual dimensión en cada par.
 - Si varias funciones se representan para la misma variable independiente, éstas se pueden colocar como filas o columnas de una matriz (primero prueba las columnas). **plot(x,[y1;y2]) , legend ('función y1', 'función y2')**: Representa las funciones **y1** e **y2 vs x**, con sus respectivas leyendas.

Señales en tiempo continuo. Introducción a Octave

GRÁFICOS 2D: STEM (valores discretos)

- Tiene características similares al **plot**, sólo que no grafica la curva continua, sino los valores discretos.
- Es posible editar propiedades/características gráficas:
 - **`stem(x,[y1;y2]','color','r','markeredgecolor','g','marker','^')`** , **`legend(' función y1',' función y2')`** , **`xlabel('tiempo ')`** , **`ylabel(' función v1','función v2')`**. Donde además de las propiedades, se agregan leyendas para los ejes y las funciones.
 - **`stem(x,[y1;y2]',' -- ^ r ')`**: aquí se le dan las propiedades de líneas verticales discontinuas, marcadores triangulares, todo en rojo.
- **Representación de varias funciones en el mismo gráfico diferente de **plot****
 - cuando la variable dependiente es una matriz, grafica el vector **x**, **vs** cada columna de la matriz, para comparar. El ejemplo de plot anterior con stem quedaría:
 - **`stem(x,[y1;y2]')`** , **`legend (' función y1',' función y2')`**: Representa las funciones **y1** e **y2 vs x**, con sus respectivas leyendas. Notar que la matriz ha sido **transpuesta**, para que cada función sea una columna de la misma

Señales en tiempo continuo. Introducción a Octave

GRÁFICOS 2D: SUBPLOT

Permite representar varios gráficos en la misma figura

- **Subplot (cant. filas, cant. columnas, índice que se quiere representar)**: divide la figura en filas y columnas (indicadas en los dos primeros índices), el último índice indica cuál se está representando en lo que se definirá.
- **Ej.:**
 - **Subplot(3,1,1)**: Divide la figura en tres filas de una columna, y en la primera fila le indicamos lo que vamos a querer representar, por ej: `stem(x,[y1;y2]','--^r')`
 - **Subplot(3,1,2)**: define la segunda fila de la figura, por ej: `stem(x,[a1;a2]','--^r')`
 - **Subplot(3,1,3)**: define la tercer fila de la figura, por ej: `stem(x,[b1;b2]','--^r')`

Señales en tiempo continuo. Introducción a Octave

FUNCTION

- **FUNCTION** genera subprogramas que se invocan desde otro programa principal o script
- **Ej.1)** Función que al ser invocada con los argumentos **x** y **h**, almacena en **y** el resultado de la convolución

```
function y = convolución (x,h)
```

```
    sentencias
```

```
endfunction
```

Desde un script se la llama como **y = convolución (x,h)**

- **Ej.2)** Función en Octave que es llamada dentro de un **script**

```
function z = Prueba (a, b)
```

```
    if (a == 1)
```

```
        z = [b,zeros(1,10)]
```

```
    else
```

```
        z = [zeros(1,10),b]
```

```
    endif
```

```
endfunction
```