

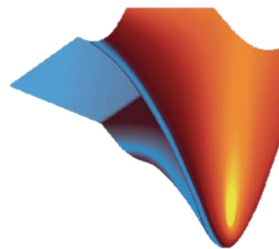
0

---

# PRACTICA 3

---

By Diana Laura de la Vega Sierra



28 DE MAYO DE 2018  
UNIVERSIDAD POLITÉCNICA DE SANTA ROSA  
Materia: Métodos numéricos.



## Tabla de contenido

<b>Introducción .....</b>	<b>2</b>
<b>Procedimiento de practica .....</b>	<b>3</b>
<b>Desarrollo de tarea</b>	
<b>a)códigos generados.....</b>	<b>4</b>
<b>b)ejecución de códigos .....</b>	<b>6</b>
<b>Conclusión individual.....</b>	<b>11</b>
<b>Bibliografía .....</b>	<b>12</b>



## Introducción

La presente practica es sobre las gráficas en 2 dimensiones elaboradas en Matlab, el uso de diferentes funciones para graficar, el como poner formato a estas graficas, como dar instrucciones para elaborarla.



## Procedimiento de práctica.

- Abra el ambiente de desarrollo MATLAB.
- Cree un archivo con el editor llamado Practica3.m y respetando el orden escriba las instrucciones necesarias para realizar las siguientes actividades:

1. Borrar todas las variables.
2. Limpiar el Command Window.
3. Usar el comando **plot** con las diversas opciones de líneas, colores y marcas, para la funciones:

$$w = 10e^{-60t}\cos(120\pi t) \quad y = 10e^{-60t}\sin(120\pi t) \quad z = 10e^{-60t}$$

Utilice  $t$  en el intervalo  $[0,0.06]$  y con incrementos de  $0.01 \times 10^{-3}$ .

4. Para las mismas expresiones graficar y colocar títulos, leyendas, etc.
5. Graficar varias figuras en una misma ventana gráfica.

$$v = \sin(x) \quad w = \cos(x) \quad y = e^{-x} \quad z = vw$$

Utilice  $x$  en el intervalo  $[0,2\pi]$  y con incrementos de 0.1.

6. Dividir la ventana gráfica en varias ventanas y colocar gráficos en cada una de ellas.
  - 6.1. Dividir la ventana en dos partes.
  - 6.2. Dividir la ventana en ocho partes.
7. Abrir varias ventanas gráficas simultáneamente y colocar gráficos en cada una de ellos.



## Desarrollo de tarea

### a) Código generado

```
clear all
close all
clc
%%
t=[0:0.01*10.^(-3):0.06]
w1=10.*exp(1).^(-60.*t).*cos(120.*pi.*t)
y=10.*exp(1).^(-60.*t).*sin(120.*pi.*t)
z=10.*exp(1).^(-60.*t)
figure(1)
plot(t,w1,'r','LineWidth',2)
xlabel('t')
ylabel('w1')
title('grafica variable w1 ')
legend('w1')
figure(2)
plot(t,y,'g')
xlabel('t')
ylabel('y')
title('grafica variable y ')
legend('y')
figure(3)
plot(t,z,'--y')
xlabel('t')
ylabel('z')
title('grafica variable z ')
legend('z')
%%
x=[0:0.1:2.*pi]
v=sin(x)
w=cos(x)
y1=exp(1).^(-x)
z1=v.*w
figure(4)
plot(x,v,':y',x,w,'--b',x,y1,'.-r',x,z1,'*g')
xlabel('x')
ylabel('v,w,y1,z1')
title('grafica variables en una misma ventana grafica')
legend('v','w','y1','z1')
%%
figure(5)
subplot(1,2,1)
plot(t,w1,'r','LineWidth',2)
subplot(1,2,2)
plot(x,v,':y',x,w,'--b',x,y1,'.-r',x,z1,'*g')
%%
figure(6)
subplot(4,3,1)
plot(t,w1,'r','LineWidth',2)
subplot(4,3,2)
plot(t,y,'g')
subplot(4,3,3)
plot(t,z,'--y')
subplot(4,3,4)
plot(x,w)
subplot(4,3,5)
plot(x,z1)
subplot(4,3,6)
```

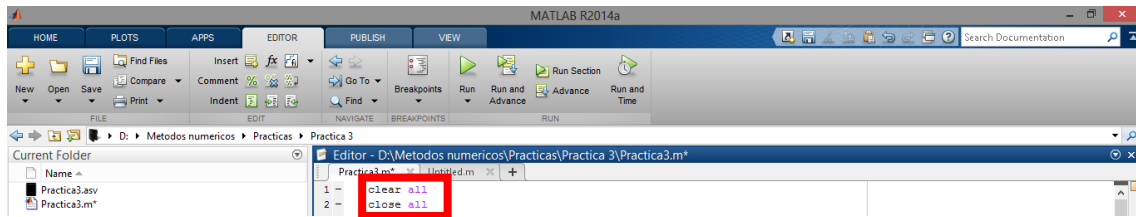


```
        plot(x,v)
        subplot(4,3,7)
plot(x,z1)
subplot(4,3,8)
plot(v,w)
%%
figure(7)
    subplot(4,3,1)
    subplot(4,3,2)
    subplot(4,3,3)
    subplot(4,3,4)
    subplot(4,3,5)
    subplot(4,3,6)
    subplot(4,3,7)
```

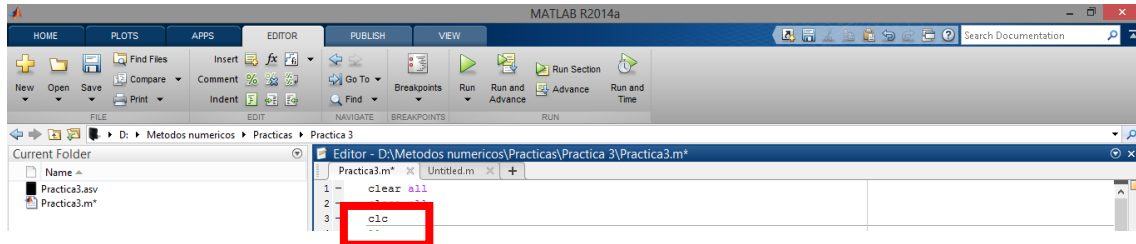


## b) ejecución de código

### 1. Borramos todas las variables.



### 2. Limpiar el Command Window.



### 3. Usar el comando **plot** con las diversas opciones de líneas, colores y marcas, para la funciones: (Utilizamos $t$ en el intervalo $[0, 0.06]$ y con incrementos de $0.01 \times 10^{-3}$ )

### 4. Para las mismas expresiones graficar y colocar títulos, leyendas, etc.

The screenshot shows the MATLAB R2014a Editor interface. The file 'Practica3.m' is open, and the code editor displays the following lines:

```
4 %%
5 t=[0:0.01*10.^(-3):0.06]
6 w1=10.*exp(1).^(-60.*t).*cos(120.*pi.*t)
7 y=10.*exp(1).^(-60.*t).*sin(120.*pi.*t)
8 z=10.*exp(1).^(-60.*t)
9
10 figure(1)
11 plot(t,w1,'r','LineWidth',2)
12 xlabel('t')
13 ylabel('w1')
14 title('grafica variable w1')
15 legend('w1')
16
17 figure(2)
18 plot(t,y,'g')
19 xlabel('t')
20 ylabel('y')
21 title('grafica variable y')
22 legend('y')
23
24 figure(3)
25 plot(t,z,'--y')
26 xlabel('t')
27 ylabel('z')
28 title('grafica variable z')
29 legend('z')
30
31 x=[0:0.1:2.*pi]
```

The workspace window shows the following variables:

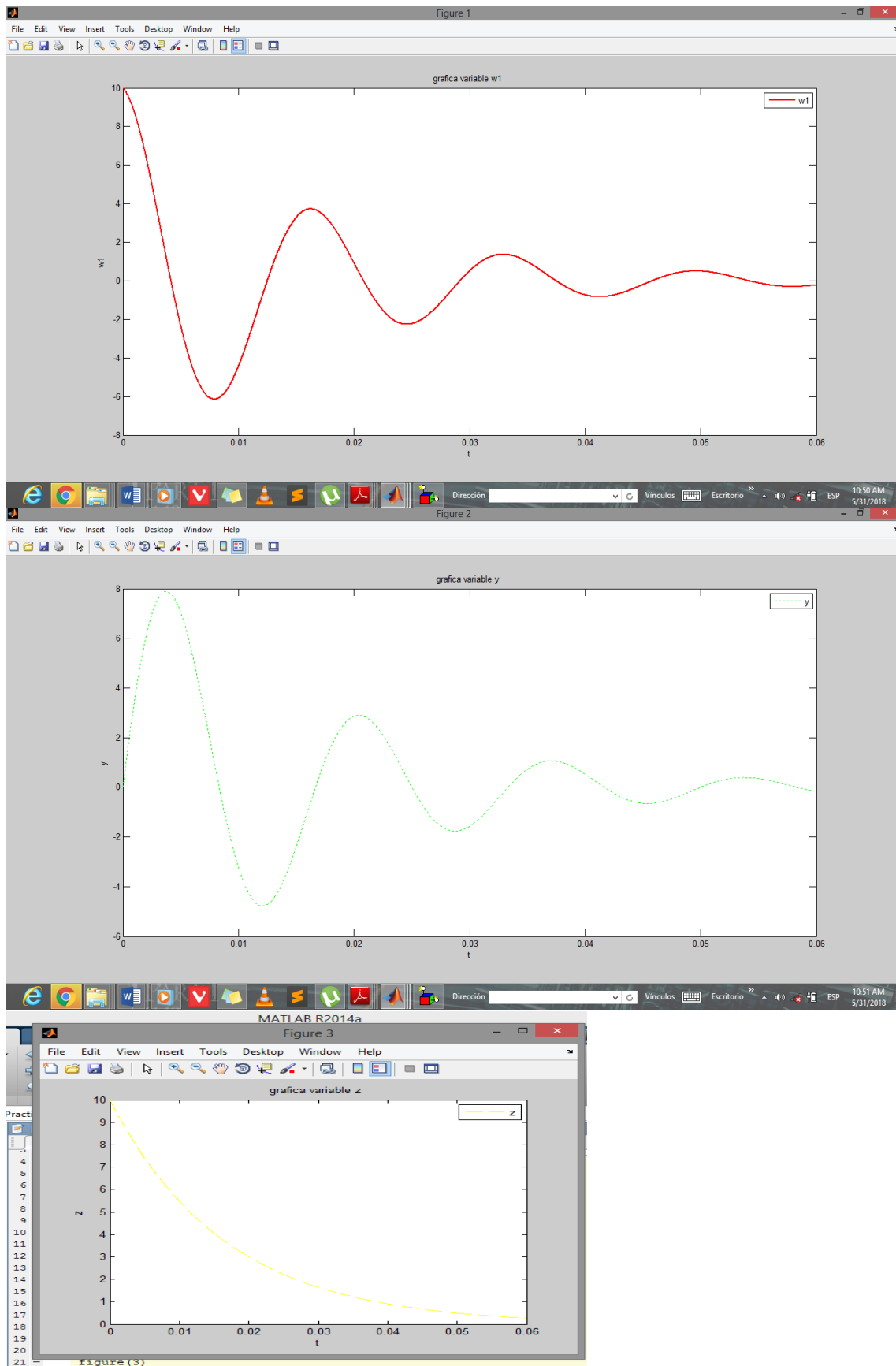
Name	Value	Min	Max
t	1x6001 double	0	0.0600
w1	1x63 double	-0.9...	0.9999
y	1x63 double	-6.1...	1
z	1x6001 double	-4.7...	7.8869

Three callouts provide instructions for the code:

1. Insertamos el intervalo con el cual estaremos trabajando
2. registramos las variables con las cuales trabajaremos
3. insertamos las instrucciones para crear las gráficas correspondientes, para identificarlas ponemos figura(n) donde n es el número de gráfica, así como también le dimos un formato, título y leyenda.



vistas de las gráficas:





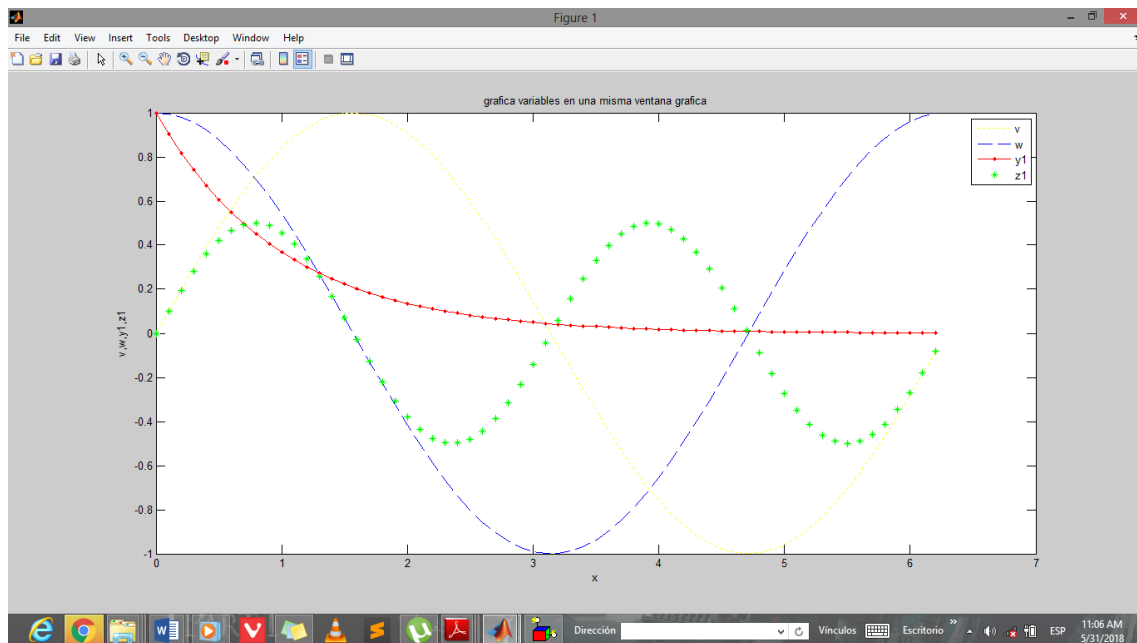


5. Graficar varias figuras en una misma ventana gráfica.  
(Utilice  $x$  en el intervalo  $[0, 2\pi]$  y con incrementos de 0.1.)

```
16 plot(t,y,'g')
17 xlabel('t')
18 ylabel('y')
19 title('grafica variable y ')
20 legend('y')
21 figure(3)
22 plot(t,z,'--y')
23 xlabel('t')
24 ylabel('z')
25 title('grafica variable z ')
26 legend('z')
27 %%
28 x=0:0.1:2.*pi
29 v=sin(x)
30 w=cos(x)
31 y1=exp(1).^(-x)
32 z1=w.*w
33 plot(x,v,'y',x,w,'--b',x,y1,'-r',x,z1,'*g')
34 xlabel('x')
35 ylabel('v,w,y1,z1')
36 title('grafica variables en una misma ventana grafica')
37 legend('v','w','y1','z1')
38 %%
39
```

Name	Value	Min	Max
t	1x6001 double	0	0.0600
v	1x63 double	-0.9...	0.9996
w	1x63 double	-0.9...	1
w1	1x6001 double	-61...	10
x	1x63 double	0	6.2000
y	1x63 double	0.0020	1
y1	1x63 double	0.0020	1
z	1x63 double	-0.5...	0.4998
z1	1x63 double	-0.5...	0.4998

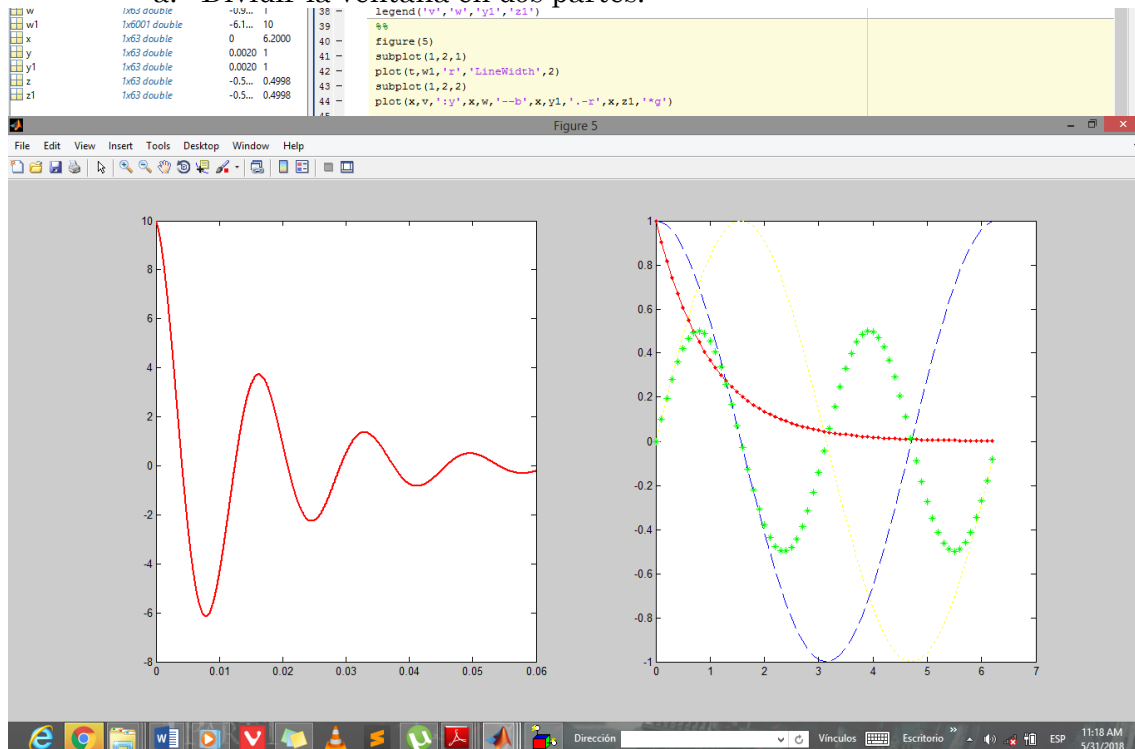
8



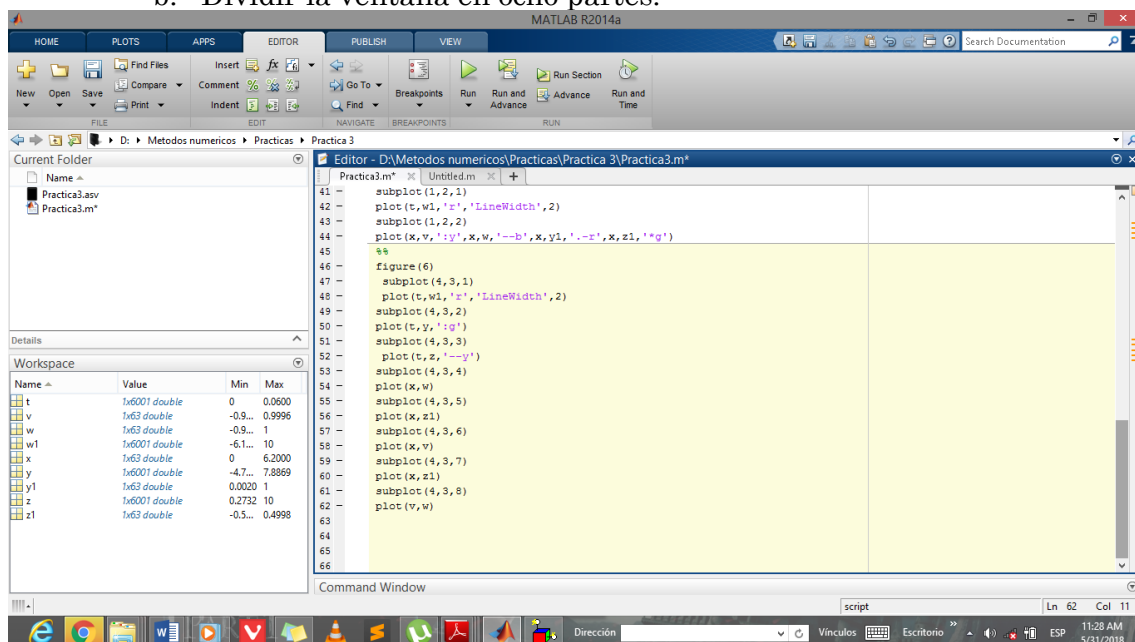


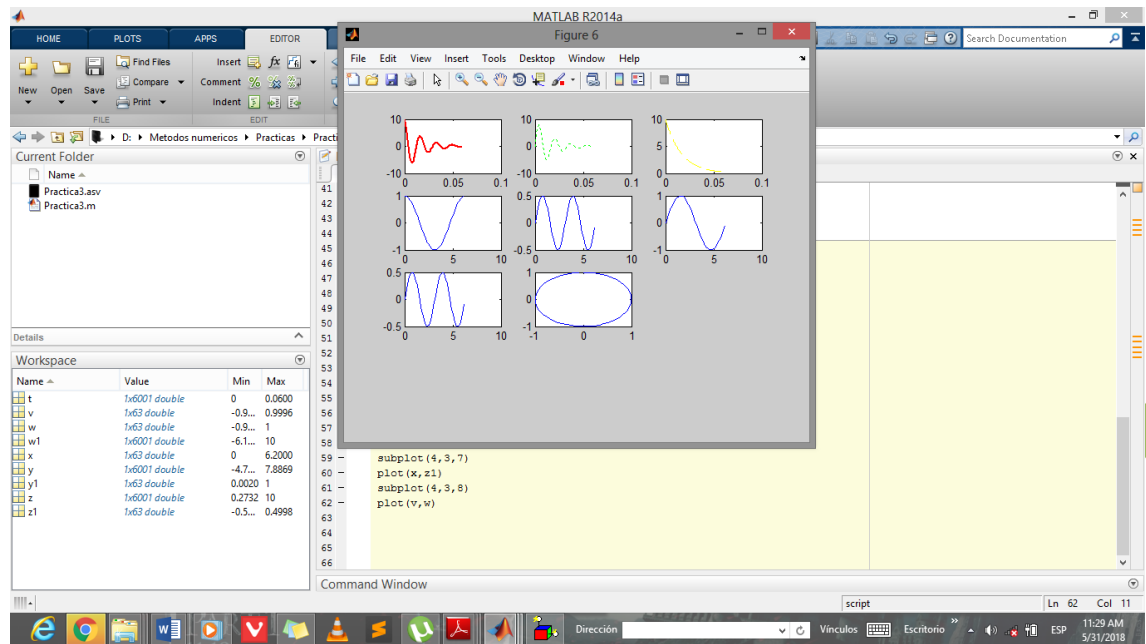
1. Dividir la ventana gráfica en varias ventanas y colocar gráficos en cada una de ellas.

a. Dividir la ventana en dos partes.



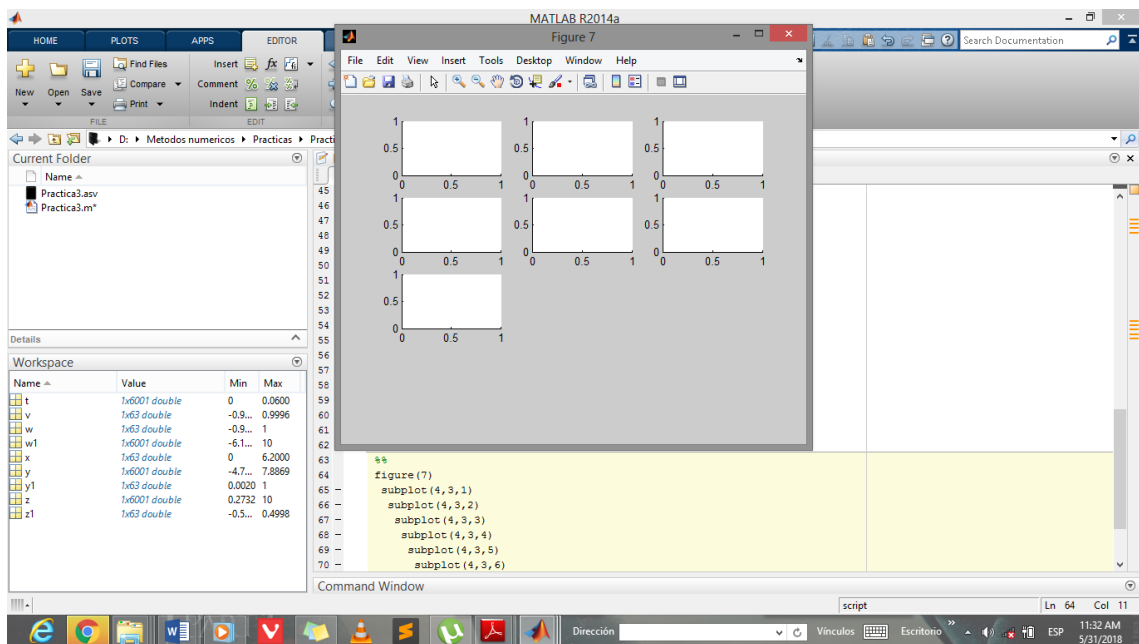
b. Dividir la ventana en ocho partes.





10

7. Abrir varias ventanas gráficas simultáneamente y colocar gráficos en cada una de ellos.





## Conclusión

Esta práctica fue un poco difícil para mí elaborarla, debido a que aún no manejo muy bien las herramientas para la elaboración de gráficas, pero aprendí mucho de ella el cómo es que se utiliza la función plot, también aprendí a como insertar el 'e' pues si lo insertamos así solo nos marcará error y la función correcta dentro de este programa es `exp(1)`, también aprendí a utilizar el subplot que sirve para poner muchos gráficos dentro de una ventana. De acuerdo a todos los conocimientos adquiridos en dicha herramienta he considerado que es una herramienta muy eficaz ya que si quieres observar  $n$  como un fenómeno 'x' avanza en 'y' de modo gráfico te ayudara bastante.



## **Bibliografía**

MATLAB para ingenieros