

Universidad Autónoma de Baja California
Facultad de Ciencias Químicas e Ingeniería



SISTEMAS DE CONTROL

Introducción a Matlab

Docente: I.E. Araiza Medrano Lizette

Alumno(s):

Gómez Cárdenas Emmanuel Alberto 02161509

Rodriguez Contreras Raul Arturo 01261510

Objetivo:

El alumno se familiarizará con el ambiente de programación de Matlab y será capaz de utilizar comandos básicos.

Material:

- Lápiz y papel (en caso de hacer anotaciones o cálculos)
- Equipo utilizado Equipo de cómputo con software Matlab

Introducción:

MATLAB es una herramienta de cómputo numérico para operaciones con matrices y vectores. Su capacidad incluye graficado. El analista numérico llamado Cleve Moler desarrollo la primera versión de Matlab en los años 70's. Desde entonces ha evolucionado con varias paqueterías para convertirse en un software comercial muy utilizado.

Matlab puede ser utilizado de diversas formas; como calculador avanzado en el modo de cálculo (prompt inicial), en modo de programación de lenguaje de alto nivel para generar las propias funciones y como subrutina para ser llamada desde un programa de lenguaje C. Obtener información sobre los comandos es posible de cualquiera de las siguientes formas: - Desde la línea de comandos utilizando el comando `>>help`. - Desde la ventana de ayuda, bajo el comando "Help Menu". - Desde el CD-ROM de Matlab para help desk. Matlab cuenta también con demostraciones ("Demos") los cuales son accesibles escribiendo el comando `>>demo`, Sin embargo, la mejor manera de aprender MATLAB es realizando ejemplos.

Desarrollo:

- Realice las operaciones abajo listadas. Grabe su archivo con sus actividades *.m y realice un documento en formato Word o PDF. Necesitará tanto el archivo como la impresión para anexarlos al reporte de su práctica.
1. Realice las siguientes operaciones aritméticas en la ventana de comandos. Utilice 4 lugares decimales de precisión y posteriormente en fracciones (***format short, format rat***) y llene la siguiente tabla: Nota, se debe de activar el formato short, o el formato rat según se desee utilizar
Investigar ¿Qué utilidad o función tiene el formato short, rat y long en Matlab?
 - **Short:** Formatea la salida como un valor corto de punto fijo con 4 dígitos después del punto.
 - **Rat:** Formatea la salida como una aproximación racional (muestra la salida como una fracción).
 - **Long:** Formatea la salida como un valor largo de punto fijo con 15 dígitos después del punto decimal para "double values" y 7 dígitos después del punto decimal para "single values".

Formula	Expresión En Matlab	Resultado En Formato Short	Resultado En Formato Rat
$x = \frac{16 + (345^2) * 6}{72}$	>> x = (16 +(345^2)*6)/72	9.9190e+03	357083/36
$Y = [2.5 \times 10^{-3} + 1.5 \times 10^{-2}]^4$	>> Y = (2.5*10^(-3)+1.5*10^(-2))^4	9.3789e-08	1/10662224
$w = 2$ $z = -5$ $r = \frac{(3.5 * z - w^3)}{(z - w)}$	$w = 2$ $z = -5$ $r = (3.5*z-w^3)/(z-w)$	$w = 2$ $z = -5$ $r = 3.6429$	$w = 2$ $z = -5$ $r = 51/14$
$A = \text{seno}\left(\frac{\pi}{3}\right)$	>> A = sin(pi/3)	0.8660	1170/1351
$x = 0.1$ $t = e^{5x^2}$	$x = 0.1$ $t = \exp(5*x^2)$	$x = 0.1$ $t = 1.0513$	$x = 1/10$ $t = 2481/2360$
$\text{Log}_{10}(1000)*5$	log10(1000)*5	15	15

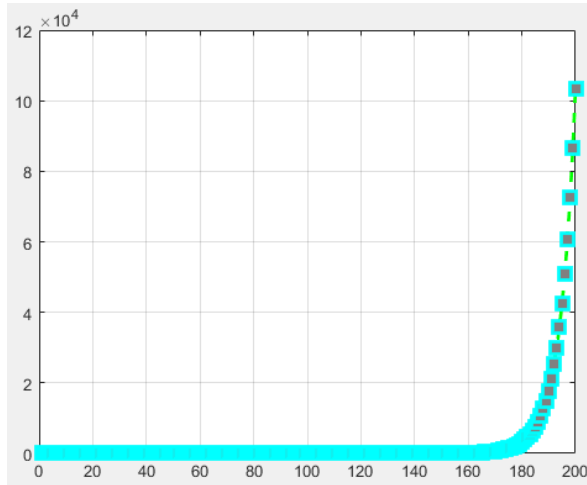
2. Inicialice Matlab anteponga el comando ***“more on”*** y de enter, posteriormente busque la ayuda para la función ***“plot”***, utilizando el comando ***“help”***. Esto ira desplegando la información poco a poco con el pulse de una tecla. *(no es compatible more on en la versión en línea).*

- ***¿Qué diferencia hay al desplegar la ayuda entre utilizar more on y al no utilizarla? ¿Cómo se desactiva la función more on?***
 - *More on controla la salida en la ventana de comandos por “paginas” de forma que sea legible el texto, la función es desactivada con el comando more OFF.*
- ***Qué diferencia hay entre la instrucción x=linspace(-2,2), x=linspace(-2,2,50) y x=-2:0.5:2***
 - *x=linspace(-2,2) genera un vector de 100 puntos lineal e igualmente espaciados entre -2 y 2.*
 - *x=linspace(-2,2,50) genera un vector de 50 puntos, en vez de 100.*
 - *x=-2:0.05:2 genera un vector entre -2 y 2 con puntos cada 0.05 espacios.*
- ***¿Cuáles son las características que se pueden variar en las gráficas al utilizar el comando PLOT?***
 - *El tipo de línea, los símbolos y colores de la gráfica.*
- ***¿Cuáles es la sintaxis para cambiar el color, el tipo de línea y el tipo de marcador en PLOT?***
 - *plot(x,y,s) en el cual “s” puede ser una cadena con un elemento de las siguientes tres columnas:*

b	blue	.	point	-	solid
g	green	o	circle	:	dotted
r	red	x	x-mark	-.	dashdot
c	cyan	+	plus	--	dashed
m	magenta	*	star	(none)	no line
y	yellow	s	square		
k	black	d	diamond		
w	white	v	triangle (down)		
		^	triangle (up)		
		<	triangle (left)		
		>	triangle (right)		
		p	pentagram		
		h	hexagram		

3. Grafique las siguientes funciones, con valores de x desde -2π hasta 2π (con un mínimo de 150 datos), puede apoyarse del comando “*linspace*” y “*grid on*”. Anexe las instrucciones utilizadas y las gráficas generadas para cada una de las funciones. Utilice diferentes colores, tipos de líneas y marcadores para darle formato sus gráficas.

NOTA: Anexar las líneas de código utilizadas en cada gráfica.

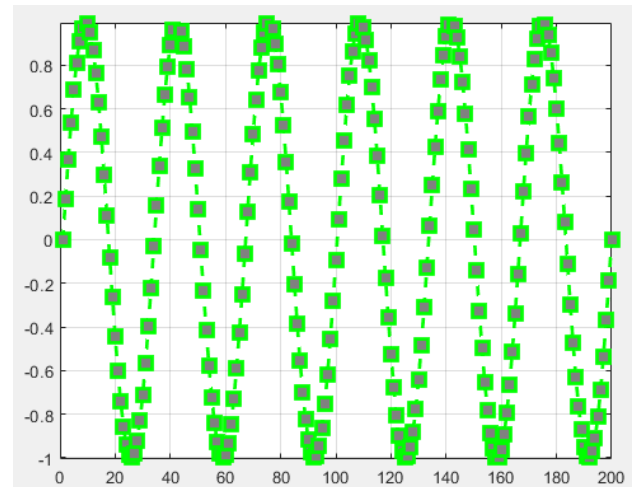


a) $Y = |x^2 - 4|$

```
%Grafica 3.1
x = linspace(-2*pi,2*pi,200);
Y = abs(power(x,x)-4);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',10,...
'MarkerEdgeColor','c',...
'MarkerFaceColor',[0.5,0.5,0.5])
```

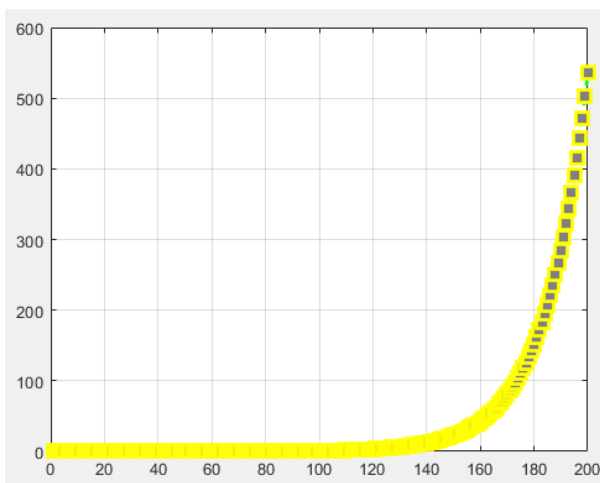
b) $Y = \sin(3x)$

```
%Grafica 3.2
Y = sin(3*x);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',10,...
'MarkerEdgeColor','g',...
'MarkerFaceColor',[0.5,0.5,0.5])
figure;
```



c) $Y = e^x$

```
%Grafica 3.2
Y = exp(x);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',10,...
'MarkerEdgeColor','y',...
'MarkerFaceColor',[0.5,0.5,0.5])
grid on;
figure;
```



4. Ahora Grafique las siguientes funciones en una misma ventana utilizando el comando ***“subplot”***. Visualizar utilizando una matriz de 3x1. Anexe las pantallas generadas y el código utilizado.

$$A) Y = x^2$$

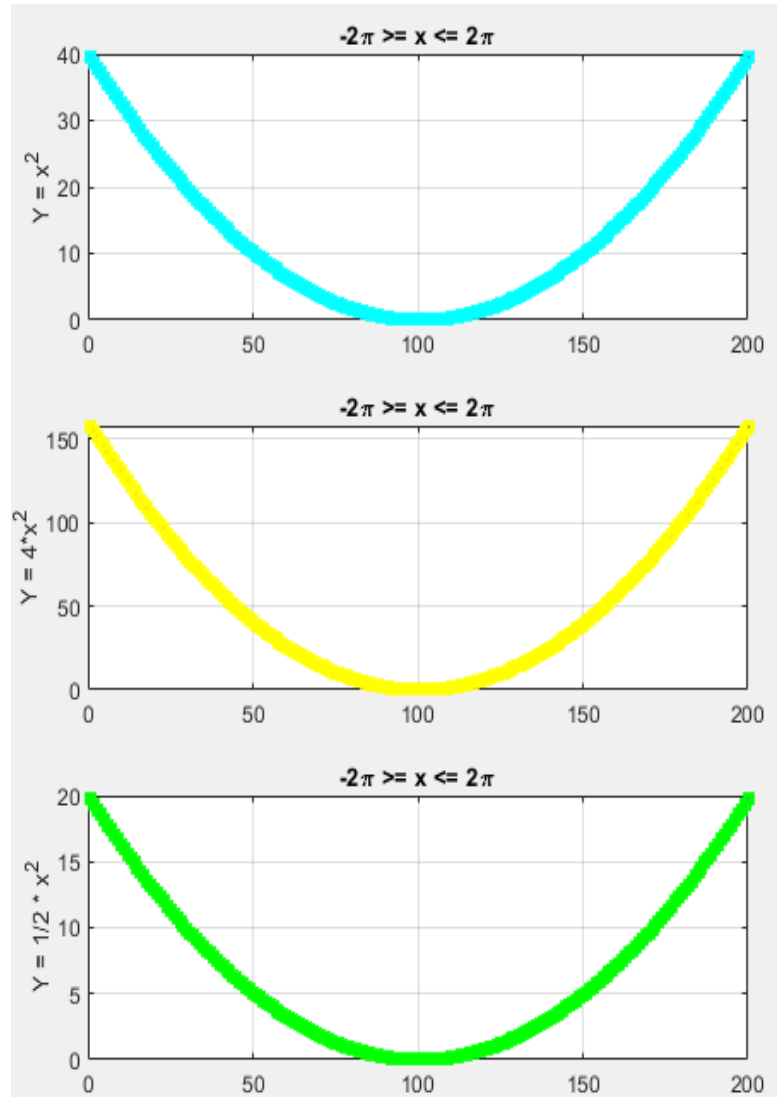
$$B) Y = 4x^2$$

$$C) Y = \frac{1}{2}x^2$$

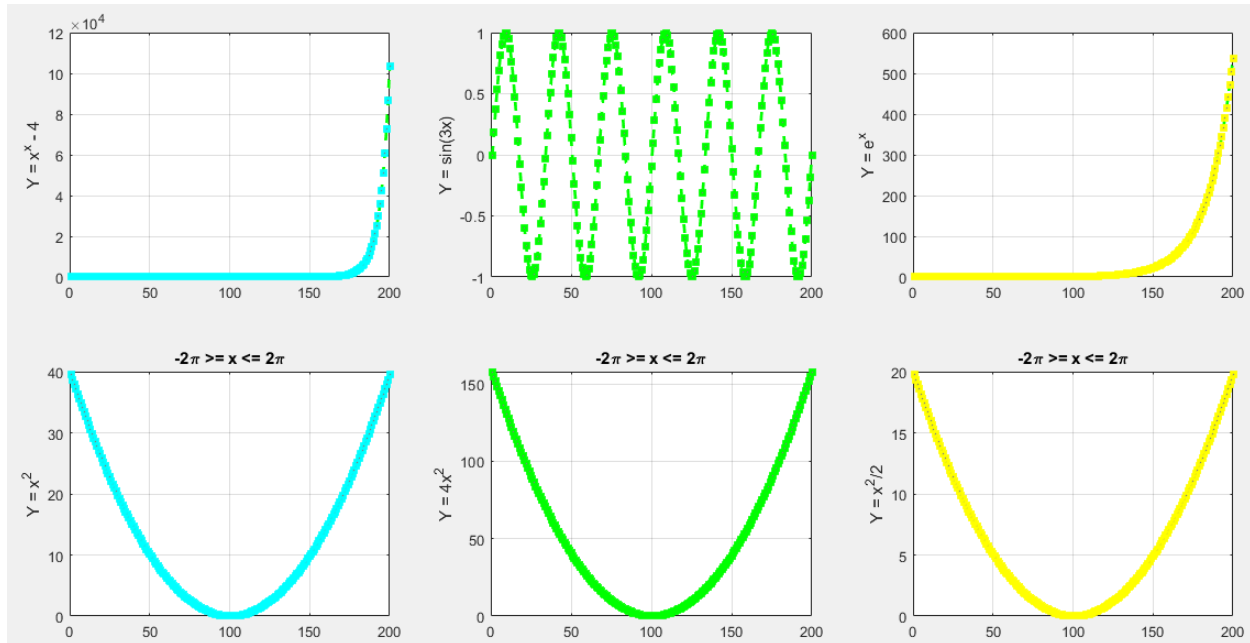
```
%Grafica 4.1
subplot (3,1,1);
Y = power(x,2);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','c',...
'MarkerFaceColor',[0.5,0.5,0.5])
title('-2\pi >= x <= 2\pi');
ylabel('Y = x^2');
grid on;
```

```
%Grafica 4.2
subplot (3,1,2);
Y = 4*Y;
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','y',...
'MarkerFaceColor',[0.5,0.5,0.5])
ylabel('Y = 4*x^2');
title('-2\pi >= x <= 2\pi');
grid on;
```

```
%Grafica 4.3
subplot (3,1,3);
Y = Y/8;
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','g',...
'MarkerFaceColor',[0.5,0.5,0.5])
ylabel('Y = 1/2 * x^2');
title('-2\pi >= x <= 2\pi');
grid on;
```



5. Genere un matriz de 2x3 y una de 3x2 utilizando las gráficas generadas en el paso 3 y en el paso 4. Utilice diferentes colores y líneas para cada gráfica. Anexe las instrucciones y las imágenes generadas.



```
%Grafica 5.1.1
subplot (2,3,1);
Y = abs(power(x,x)-4);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','c',...
'MarkerFaceColor',[0.5,0.5,
0.5])
ylabel('Y = x^x');
grid on;
```

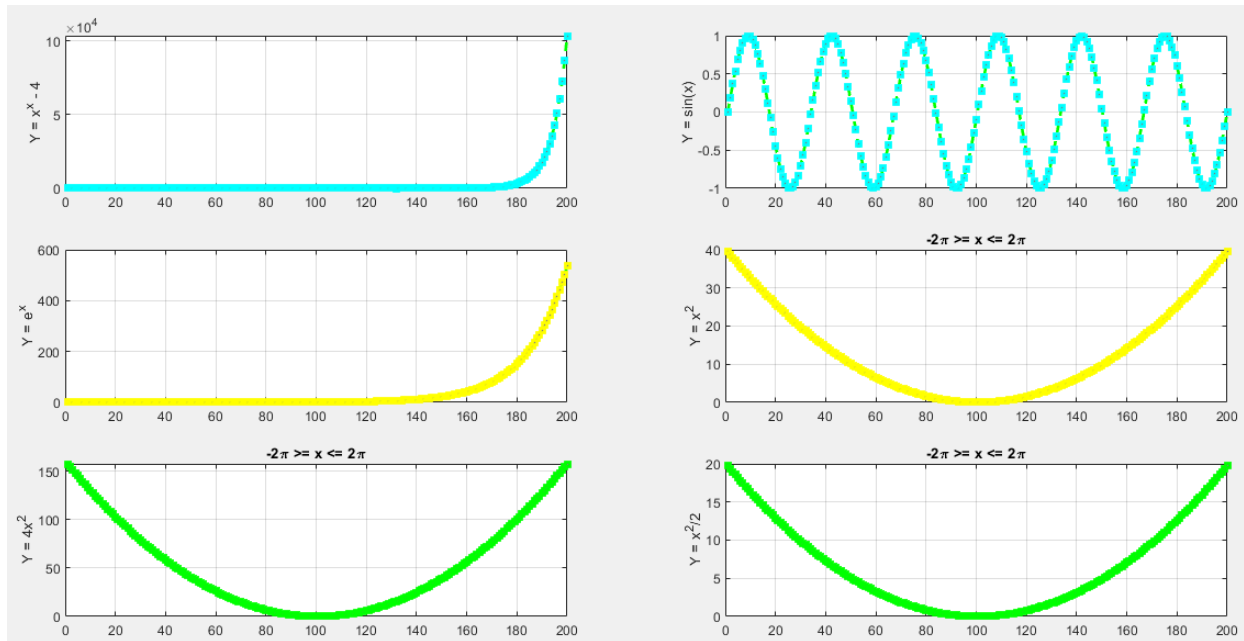
```
%Grafica 5.1.2
subplot (2,3,2);
Y = sin(3*x);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','g',...
'MarkerFaceColor',[0.5,0
.5,0.5])
ylabel('Y = sin(3x)');
grid on;
```

```
%Grafica 5.1.3
subplot (2,3,3);
Y = exp(x);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','y',...
'MarkerFaceColor',[0.5,0.5,0.
5])
ylabel('Y = e^x');
grid on;
```

```
%Grafica 5.1.4
subplot (2,3,4);
Y = power(x,2);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','c',...
'MarkerFaceColor',[0.5,0.5,
0.5])
title('-2\pi >= x <=
2\pi');
ylabel('Y = x^2');
grid on;
```

```
%Grafica 5.1.5
subplot (2,3,5);
Y = 4*Y;
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','y',...
'MarkerFaceColor',[0.5,0
.5,0.5])
ylabel('Y = 4x^2');
grid on;
```

```
%Grafica 5.1.6
subplot (2,3,6);
Y = Y/8;
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','g',...
'MarkerFaceColor',[0.5,0
.5,0.5])
ylabel('Y = x^2/2');
grid on;
```



```
%Grafica 5.2.1
subplot (3,2,1);
Y = abs(power(x,x)-4);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','c',...
'MarkerFaceColor',[0.5,0.5,0.5])
ylabel('Y = x^x-4');
grid on;
```

```
%Grafica 5.2.2
subplot (3,2,2);
Y = sin(3*x);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','g',...
'MarkerFaceColor',[0.5,0.5,0.5])
ylabel('Y = sin(3x)');
grid on;
```

```
%Grafica 5.2.3
subplot (3,2,3);
Y = exp(x);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','y',...
'MarkerFaceColor',[0.5,0.5,0.5])
ylabel('Y = e^x');
grid on;
```

```
%Grafica 5.2.4
subplot (3,2,4);
Y = power(x,2);
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','c',...
'MarkerFaceColor',[0.5,0.5,0.5])
title('-2\pi \ge x \le 2\pi');
ylabel('Y = x^2');
grid on;
```

```
%Grafica 5.2.5
subplot (3,2,5);
Y = 4*Y;
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','y',...
'MarkerFaceColor',[0.5,0.5,0.5])
ylabel('Y = 4x^2');
title('-2\pi \ge x \le 2\pi');
grid on;
```

```
%Grafica 5.2.6
subplot (3,2,6);
Y = Y/8;
plot(Y,'--gs',...
'LineWidth',2,...
'MarkerSize',5,...
'MarkerEdgeColor','g',...
'MarkerFaceColor',[0.5,0.5,0.5])
ylabel('Y = x^2/2');
title('-2\pi \ge x \le 2\pi');
grid on;
```


Conclusión Individual:

- **Gomez Cárdenas Emmanuel Alberto:**

- Matlab es una herramienta bastante útil a la hora de hacer cualquier tipo de operación matemática compleja, ya sea analizar datos, crear algoritmos, modelos y hasta para tareas más sencillas como lo es graficar, con la ventaja de ser también una ayuda grafica con una gran cantidad de opciones para tomar una idea y transformarla en algo mejor. Aprender a interpretar los resultados es una parte esencial por eso es importante aprender a mostrarlos gráficamente, para que quede más claro o sencillo de entender el resultado.

- **Rodriguez Contreras Raul Arturo:**

- La práctica es algo muy sencillo pero necesario, pues cambiar el formato en que se presentan resultados en Matlab es una habilidad indispensable para cualquier cálculo que se requiera hacer con esta herramienta. Además, presentar los resultados de manera gráfica en diferentes plots o en el mismo ayuda mucho de manera visual a interpretar los datos que se están manipulando, pues los vectores numéricos no siempre dan la imagen completa del comportamiento, hasta verlo en una gráfica se puede apreciar un comportamiento general. Por esto considero que la práctica a pesar de ser sencilla es un pilar fundamental para el uso de Matlab como herramienta.

References

Centro de Ayuda. (2022). Retrieved 21 August 2022, from <https://la.mathworks.com/help/matlab/>