

MATLAB

Notas preparadas para el curso de Análisis Numérico
Ing. Luis Rodríguez Ojeda

Introducción

Matlab es un programa interactivo para aplicaciones matemáticas y de ingeniería.

Aspectos que incluye MATLAB:

Aplicaciones

- Cálculo numérico
- Manejo simbólico
- Graficación
- Programación estructurada
- Programación orientada a objetos
- Herramientas básicas para diseño de interfaz gráfica

Operativos

- Simple y eficiente
- Interactivo
- Sistema de ayuda en línea
- Interacción con otros entornos

Uso de Matlab

La interacción se realiza mediante órdenes o comandos y también mediante programas. Los objetos básicos con los cuales se opera son matrices.

El uso de los comandos mas útiles se explicará mediante ejemplos.

Procedimiento

Ingrese al programa MATLAB

Matlab está listo a recibir comandos mostrando el símbolo: **>>**

Escriba cada ejemplo presionando al final la tecla de ingreso.

Al final de cada ejemplo he escrito una breve explicación del mismo.

Por supuesto, usted no debe escribir esa explicación en el computador..

Una descripción detallada puede obtenerla con el sistema de ayuda de MATLAB.

Ejemplo

>> x=exp(2)/3;	el resultado se asigna a una variable
>> x	escriba la variable para conocer su contenido
>> x=exp(2)/3	si omite ; se muestra el resultado inmediatamente

Con el cursor $\uparrow \downarrow$ se pueden re-utilizar comandos

El sistema de ayuda de MATLAB

>> help despliega temas de ayuda

>> help ops	despliega comandos de un tema. ej. operadores
>> help exp	uso de un comando específico ej. función exponencial

Ejemplos para entender el poder de MATLAB

1) Resuelva el sistema

$$2x + 3y = 4$$

$$5x - 2y = 6$$

>> a=[2 3; 5 -2];	ingreso de la matriz de coeficientes
>> b=[4; 6];	ingreso del vector de constantes
>> x=inv(a)*b;	obtener la solución
>> x	mostrar el vector solución

2) Grafique la función tangente en un intervalo dado:

>> ezplot('tan(x)', -2, 2);	graficar la función tan(x) en el intervalo [-2,2]
>> grid;	mostrar cuadrículas
>> title('función tangente')	poner título

3) Resuelva la ecuación cuadrática $2x^2 + 3x + 5 = 0$;

>> a=[2 3 -5];	definición de los coeficientes
>> x=roots(a);	cálculo de raíces
>> x	mostrar solución

4) Resuelva y grafique la ecuación diferencial ordinaria: $y' = x + y$, $y(0) = 1$

```
>> y=dsolve('Dy=x+y','y(0)=1','x');
>> ezplot(y, 0, 2);
>> grid
```

Símbolos especiales

[]	para definir vectores y matrices
()	define precedencia en expresiones y subíndices
,	separa elementos de una lista, o espacios en blanco
;	separa declaraciones
%	inicio de comentarios

CALCULO NUMERICO

Formatos de exhibición de números

>> format long	14 decimales
>> x=2/3	un ejemplo para visualizar
>> format bank	2 decimales
>> x	
>> format rat	fracciones
>> x	
>> format short e	notación científica
>> x	

```
>> format long e
>> format +      visualizar signos +, -,
>> x
>> format short   4 decimales (por omisión)
```

Operadores aritméticos

+ - * / \ ^ ()

Funciones matemáticas

sin, cos,

```
>> help elfun      lista las funciones matemáticas elementales
```

Práctica con expresiones aritméticas

```
>> x=sin(3)+sqrt(2)
```

Operadores relacionales y lógicos

< <= > >= == ~= & | ~

Práctica con operadores relacionales y lógicos

```
>> x=3<4 & 4>2      el resultado es el valor lógico falso: 0
>> help ops          lista los operadores Matlab
```

Símbolos numéricos especiales

```
>> 2/0
      Inf          es el símbolo  $\infty$ 
>> 0/0
      NaN          significa "Not A Number" (indeterminación)
>> pi             contiene la constante  $\pi$ 
>> eps            es la precisión del tipo real en MATLAB
>> realmin        el menor número real
>> realmax        el mayor número real
>> i              representa al componente complejo  $\sqrt{-1}$ 
```

Práctica con números complejos

```
>> x=(3+2i)*(3-5i);
>> y=exp(x)        el resultado también es complejo
>> y=log(-2)       el referencial de Matlab son los complejos
>> y=sin(x)
```

Funciones adicionales para números complejos

conj, real, imag, abs, angle

```
>> z=3+2i;
>> t=conj(z)       obtener el conjugado
```

Variables

- No requieren ser declaradas
- Su tipo depende del valor asignado
- Pueden ser redefinidas
- Sensible al tipo de letra (mayúsculas o minúsculas)

- **ans** es la variable por omisión provista por MATLAB

>> x=3	x es de tipo real
>> x='mensaje'	x es de tipo literal
>> x=2+3i	x es de tipo complejo
>> x	muestra el contenido actual de la variable
>> whos x	muestra el tipo actual de la variable
>> disp (x)	muestra solamente el contenido
>> x= input ('dato?');	ingresa un valor desde el teclado
>> x	
>> exp(x)/3	
>> ans	la variable ans contiene el último resultado
>> y=2* ans ;	se la puede usar
>> y	

Algunos comandos del Sistema Operativo

>> help general	Lista de comandos
>> who	lista variables en uso
>> whos	lista variables en uso y su descripción
>> clear	borra variables
>> pwd	muestra cual es el directorio actual
>> cd	cambia la ruta del directorio actual
>> dir	lista el directorio actual
>> save	almacena variables en un archivo .mat
>> load	carga variables y su contenido
>> delete	elimina archivo
>> quit	termina la sesión con Matlab

Comandos especiales

>> date	fecha
>> clock	fecha hora, vea su uso con help.
>> format rat	visualízelo con mas claridad

Cadenas de caracteres

>> x='ejemplo';	asignación de una cadena
>> x(4)	manejo de un carácter de la cadena
>> t=x(2:5);	una subcadena
>> length(x)	longitud de la cadena
>> strcat(x,t)	concatenación de cadenas
>> help strfun	lista de funciones para cadenas

Vectores y Matrices

>> x=[3, -1, 4, 7, -2];	asignación directa de un vector fila
>> x=[3 -1 4 7 -2];	pueden separarse con , o espacios
>> x(2)=5;	manejo de uno de sus componente
>> y=x(2:4);	asignar parte de un vector
>> t=[3; -1; 4];	asignación directa de un vector columna
>> t=x';	o mediante la transpuesta de un vector fila
>> y=[3, x, -6];	asignación con otro vector
>> y=2:10;	asignación con valores con incremento 1

```
>> y=2:0.1:5;
>> y=[2, 5, 4, ...
>> 7, -3];
>> x=[3,5,2,0];
>> y=2*x
>> y=sin(x)
>> a=[2,-3;5,1];
>> a(2,1)
>> a=[2,-3;5,1;0,7]
>> a=[x;x];
>> a=[1,2;3,4];
>> b=[5,6];
>> c=[a;b];
>> d=[a,b'];
>> x=c(1,:);
>> x=c(:,1)
```

con un incremento diferente de 1

pueden abarcar mas de una línea

vector mediante una operación escalar
o mediante una función

asignación directa de una matriz 2x2

manejo de uno de sus componentes

una matriz 3x2

una matriz 2x4

una matriz 2x2

vector fila de 2 columnas

c es una matriz 3x2

c es una matriz aumentada 2x3

asigna al vector x la primera fila de c

asigna al vector x la primera columna de c

Matrices especiales

```
>> a=ones(5)
>> a=ones(3,5)
>> a=zeros(4,5)
>> a=eye(5)
```

matriz 5x5 iniciada con 1's

matriz 3x5 iniciada con 1's

matriz 4x5 iniciada con ceros

matriz identidad 5x5

Operaciones con matrices

```
>> a=[1,2;3,4];
>> b=[5,6;7,8];
>> c=2*a
>> c=a+b
>> c=a*b
>> c=a.*b
>> c=a^2
>> c=a.^2
```

escalar por matriz

suma de matrices

producto de matrices

multiplicación elemento a elemento

matriz al cuadrado

cada elemento al cuadrado

Funciones para matrices

```
>> x=[2,3,-6,5];
>> a=[1,2,3;4,5,6;7,8,9];
>> size(x)
>> size(a)
>> dot(x,x)
>> rank(a)
>> trace(a)
>> det(a)
>> inv(a)
>> [l,u]=lu(a)
>> l
>> u
>> l*u
>> a'
>> eig(a)
>> poly(a)
```

tamaño de x: 1x4, el resultado es un vector

tamaño de a: 3x3, el resultado es un vector

producto punto entre dos vectores

rango de a

traza de a

determinante de a

inversa de a

descomposición triangular de a

l y u son matrices tales que $l*u = a$

se obtiene la matriz a

transpuesta de a

valores propios de a

polinomio característico de a

>> roots (ans)	valores propios de a
>> a=[]	matriz nula
>> help matfun	funciones adicionales para matrices

Funciones adicionales para manejo de datos con vectores y matrices

>> x=[2,5,4];	
>> a=[5,-1;3,4;2,7];	
>> max (x)	el mayor valor
>> max(a)	el mayor valor por columnas
>> sum (x)	suma de componentes
>> sum(a)	suma de componentes por columnas
>> prod (x)	producto escalar
>> prod(a)	producto escalar por columnas
>> cumsum (x)	suma acumulada
>> cumsum(a)	suma acumulada por columnas
>> cumprod (x)	producto acumulado
>> cumprod(a)	
>> mean (x)	media aritmética
>> mean(a)	
>> median (x)	mediana
>> median(a)	
>> std (x)	desviación estándar
>> std(a)	
>> sort (x)	ordenamiento ascendente
>> sort(a)	
>> dsort (x)	ordenamiento descendente
>> bar (x)	diagrama de barras
>> bar(a)	
>> hist (x)	histograma
>> pie (x)	gráfico tipo pastel
>> pie3 (x)	pastel en relieve
>> v=[0,1,0]	vector para extraer sectores del pastel
>> pie3(x,v)	gráfico tipo pastel con sector separado

Polinomios

>> a=[2,-3,0,5];	define el polinomio $2x^3 - 3x^2 + 5$
>> y= polyval (a,4)	evaluación con un valor
>> x= roots (a)	raíces
>> polyval (a,x(1))	verificar una raíz
>> poly (x)	producto de todas las raíces
>> b=[3,4,-2];	define el polinomio $3x^2 + 4x - 2$
>> conv (a,b)	producto de polinomios
>> [c,r]= deconv (a,b);	división de polinomios
>> c	cociente
>> r	residuo
>> x=[2,3,5,7,8];	abscisas de puntos (x,y)
>> y=[3.2,4.1,5.8,6.4,6.3];	ordenadas de los puntos
>> z=3.2;	valor para interpolar, z puede ser un vector
>> u= interp1 (x,y,z,'linear')	interpolación lineal
>> u= interp1 (x,y,z,'spline')	interpolación con un 'trazador cúbico'

>> a=polyfit(x,y,2)

ajuste de mínimos cuadrados de grado 2

MANEJO SIMBÓLICO

>> syms x;

definición de variable tipo simbólico

>> 2*x+3*x

suma algebraica

>> f=3*x^2+5*x;

definición simbólica de una función

>> factor(f)

factorización

>> expand(f)

expandir la expresión

>> taylor(exp(x))

expansión con la serie de Taylor

>> f='2*x+1';

definición de una función en forma literal

>> x=3;

>> y=eval(f)

evaluación de la función

Funciones especiales para análisis numérico

>> f='exp(x)-3*x';

>> fzero(f,2)

solución de una ecuación con un valor inicial

>> diff(f)

diferenciación

>> int(f)

integración analítica

>> int(f, 0, 2)

integral definida

>> y=dsolve('Dy=x+y','y(0)=1','x') solución de una EDO de primer orden

Funciones especiales para medir eficiencia de algoritmos

>> flops(0);

Inicia el conteo de operaciones aritméticas

>> a=[2 3 ; 4 5]

un ejemplo para medir la eficiencia

>> b=inv(a);

>> flops

Muestra la cantidad de operaciones aritméticas

Generación de números aleatorios

>> x=rand

Genera un número aleatorio entre 0 y 1

>> a=rand(5)

Genera una matriz nxn con números aleatorios

>> b=rand(4,5)

Genera una matriz nxm con números aleatorios

GRAFICACIÓN

Gráficos de funciones de una variable

>> f='exp(x)-3*x';

>> ezplot(f, 0, 2)

gráfico de una función en un dominio dado

>> x=0:0.1:2*pi;

>> y=sin(x);

puntos de la función seno

>> plot(x,y);

gráfico de la función seno con una línea continua

>> plot(x,y,'o')

gráfico con círculos. Puede elegir: o . * + x --

>> plot(x,y,'r')

gráfico en color rojo. Puede elegir r,b,y,m,g,b,w,k

>> plot(x,y,'og')

gráfico con círculos verdes.

>> grid on

dibujar cuadrículas

>> title('seno de x')

título

>> gtext('seno de x')

posiciona el texto en el gráfico con el mouse

>> xlabel('X')

rotular eje horizontal

>> ylabel('Y')

rotular eje vertical

>> a=[0, 2*pi, -2, 2])

definir región para el gráfico

>> axis(a)	
>> hold on	superponer subsecuentes gráficos
>> figure(1)	puede tener varias figuras abiertas
>> subplot(2,3,1)	puede dividir una figura en subgráficos
>> clf	borrar todos los gráficos
>> x=0:0.1:10;	
>> y=exp(x);	
>> semilogx(x,y)	gráficos en escalas logarítmicas
>> semilogy(x,y)	
>> loglog(x,y)	
>> grid	
>> a=0:0.01:2*pi;	
>> r=sin(3*a);	
>> polar(a,r);	gráfico en coordenadas polares

Gráficos de funciones de dos variables

>> x=-2:0.1:2;	dominio para el ejemplo
>> y=-3:0.1:3;	
>> [u,v]=meshgrid(x,y);	u, v: matrices q' contienen cada par ordenado x,y
>> z=u.^2 - v.^2;	puntos de la función $z = x^2 - y^2$
>> mesh(x,y,z)	gráfico de malla
>> contour(x,y,z)	gráfico de contorno
>> surf(x,y,z)	gráfico de superficie
>> xlabel('X')	rotulación de eje x; también use ylabel, zlabel
>> title('Silla de montar')	título del gráfico
>> colormap(copper);	color del gráfico; también gray, jet, pink
>> shading interp;	suavizado del gráfico

PROGRAMACIÓN EN MATLAB

Archivos de comandos y funciones

Para crear un archivo de comandos, programa, o script, seleccione de la barra de herramientas de Matlab: File -> New -> M-file. Escriba los comandos y almacénelo con algún nombre. Puede incluir comentarios con el símbolo %. Ejemplo:

```
x=0:0.1:2*pi;
y=sin(x);
plot(x,y);
grid
```

Guardar el ejemplo con el nombre **prueba.m**

Para ejecutar el programa escriba en la línea de comandos

```
>> prueba
```

Para crear una función, debe crear el archivo de comandos con un encabezamiento para la función. Si la salida es antes del final, puede usar el comando **return**. Ejemplo:

```
function y=iva(x)
```



```
% cálculo del impuesto a las ventas
y=1.12*x;
```

Guardar el ejemplo con el nombre **iva.m**

Para usar la función use el mismo nombre. Ejemplo

```
>> c=38;
>> t=iva(c)
```

Si el parámetro es un vector o matriz, el resultado también:

```
>> c=[4 5 7.2 6];
>> t=iva(c)
```

Estructuras de control de flujo en Matlab

Decisiones

Escriba y almacene la siguiente función

```
function m=mayor(a, b)
% obtener el mayor entre dos valores
if a>b
    m=a;
else
    m=b;
end
```

Uso de la función:

```
>> t=mayor(3,7)
```

Decisiones múltiples

Escriba, almacene y pruebe el siguiente archivo de comandos

```
x=input('dato');
switch x
    case 1,
        y=3;
    case {2,3},
        y=2*x+1;
    otherwise
        y=exp(x);
end
disp(y);
```

Repetición condicionada al inicio

```
function s=suma(n)
% suma de la serie armónica
s=0;
while n>0
    s=s+1/n;
```

```
n=n-1;  
end
```

Repetición condicionada a una secuencia

```
function s=suma(n)  
% suma de la serie armónica  
s=0;  
for i=1:n;  
    s=s+1/ i;  
end
```

Uso de cualquiera de estas dos funciones:

```
>> t=suma(4)
```

Para interrumpir una repetición puede usar **break**

Para ver la descripción de las estructuras del lenguaje de Matlab, escriba

```
>> help lang
```

INTERACCION CON OTROS ENTORNOS

Interacción con EXCEL

- a) En **Excel** cree la tabla y almacénela con formato tipo texto separado con tabulaciones y almacénela con algún nombre. Ejemplo **q.txt**
- b) En **Matlab** cargue la tabla

```
>> load q.txt;  
>> q
```