

Reconhecimento de Padrões

Guia de Laboratório

Introdução ao MATLAB


Realizado por: Profª Ana Fred

IST - 2000

Guia de Laboratório - Introdução ao MATLAB

❖ Objectivos:

O presente guia tem por objectivo proporcionar uma breve introdução aos conceitos de representação e manipulação de matrizes proporcionado pela aplicação MATLAB. Os conceitos e comandos básicos do MATLAB são abordados através da realização de um conjunto de exercícios de interacção com esta aplicação, indicados numa caixa à esquerda da qual se encontra

o símbolo . No computador do seu laboratório encontrará acessível uma versão do MATLAB, à qual poderá aceder através do menu de aplicações do Windows.

❖ O que é o MATLAB?

- ❖ É um programa interactivo que permite cálculo numérico e visualização de dados. Está construído com base num *software* sofisticado de manipulação de matrizes para a análise de sistemas de equações lineares

❖ Interacção com o MATLAB: janela de comandos

Quando se inicia o MATLAB é criada uma ou mais janelas. A janela de comandos, através da qual se interage com o MATLAB, apresenta o aspecto típico representado na figura seguinte:

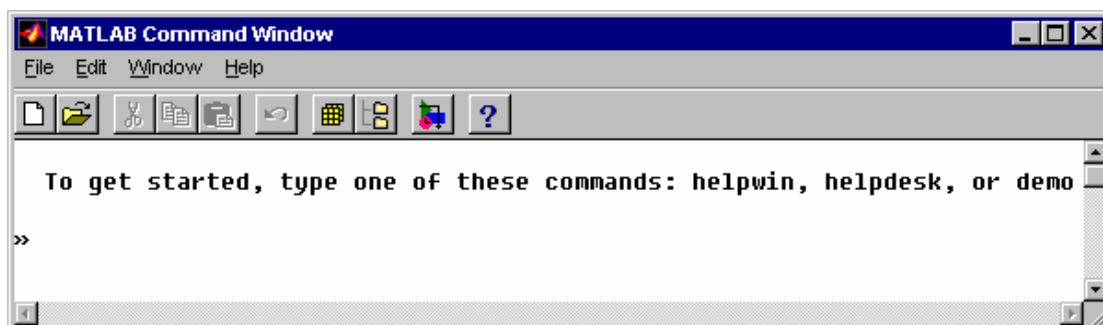


Figura 1: Janela de comandos do MATLAB.

❖ MATLAB e matrizes

- ❖ O MATLAB trabalha com escalares, vectores e matrizes. As matrizes formam o núcleo do MATLAB uma vez que todos os dados são armazenados em matrizes (um escalar é uma matriz 1x1; um vector é uma lista de números, ou seja uma matriz linha ou uma matriz coluna).

❖ Números e operações matemáticas simples

- ❖ O MATLAB usa a convenção decimal normal para a representação de números. Aceita números reais, imaginários e complexos.

- Exemplos de números legais:

3	-99	0.0001
9.76892	1.898E-20	6.8978e23
2i	-3.15666i	3e5I
7j	3+4i	

- ❖ Execução de operações matemáticas simples com números:

- Funcionamento como calculadora



```

» 4+6+2
ans =
    12

» 4*25+6*22+2*99
ans =
    430

» (1+i)*5
ans =
 5.0000+ 5.0000i

```

Obs: *ans* é a abreviatura de “*answer*”.

- Forma alternativa armazenando informação em variáveis:



```

» item1=4
item1 =
     4

» item2=6
item2 =
     6

» item3=2
item3 =
     2

» items=item1+item2+item3
items =
    12

» custo=item1*25+item2*22+item3*99
custo =
    430

» c1=(1+i)*5
c1 =
 5.0000+ 5.0000i

```

- Sumário de operações matemáticas elementares:

Operação	Símbolo	Exemplo
Adição, a+b	+	4+8
Subtração, a-b	-	23-12
Multiplicação, a·b	*	3.14*0.87
Divisão, a÷b	/ ou \	56/8=8\56
Potência, a ^b	^	5^2

❖ Sobre as variáveis ...

❖ Designação das variáveis

- A designação de uma variável segue as seguintes convenções:
 - Deve ser formada por uma única palavra sem espaços;
 - Caracteres em maiúsculas e minúsculas têm significado diferente;
 - Pode ter até 19 caracteres;
 - Deve começar por uma letra;
 - Não são permitidos caracteres de pontuação.
- Algumas variáveis especiais:



Variável	Valor
ans	Resultado anterior
pi	Valor de Π
inf	infinito
i e j	$i=j=\sqrt{-1}$
realmin	O menor valor real positivo
realmax	O maior valor real positivo

❖ Comentários e pontuação

- Comentários: %



```
» items=12 %numero de items
items =
    12
```

- Separação entre comandos: ; ,



```

> item1=4, item2=6; item3=2
item1 =
     4

item3 =
     2

```

- Continuação de um comando na linha seguinte: ...



```

> custo_medio=custo/...
items
> custo_medio =
    71.6667

```

❖ Algumas funções matemáticas

- ❖ `abs(x)`, `acos(x)`, `acosh(x)`, `angle(x)`, `asin(x)`, `asinh(x)`, `atan(x)`, `ceil(x)`, `conj(x)`, `cos(x)`, `cosh(x)`, `exp(x)`, `fix(x)`, `floor(x)`, `gcd(x,y)`, `imag(x)`, `lcm(x,y)`, `log(x)`, `log10(x)`, `real(x)`, `round(x)`, `sign(x)`, `sin(x)`, `sinh(x)`, `sqrt(x)`, `tan(x)`, `tanh(x)`

- ❖ É importante notar que o MATLAB só trabalha com radianos:



```

> x=sqrt(2)/2
x =
    0.7071
> y=asin(x)
y =
    0.7854
> y_graus=y*180/pi
y_graus =
    45.0000

```

❖ Números complexos

- Operações com números complexos



```

» c1=1-2i
c1 =
    1.0000- 2.0000i
» c2=3*(2-sqrt(-1)*3)
c2 =
    6.0000- 9.0000i
» c3=sqrt(-2)
c3 =
         0+ 1.4142i
» c4=6+sin(.5)*i
c4 =
    6.0000+ 0.4794i
» (c1+c2)/c3
ans =
   -7.7782- 4.9497i
» i^2
ans =
   -1.0000+ 0.0000i

```

- Conversão entre coordenadas polares e retangulares



```

» c1 =
    1.0000+ 2.0000i
» m=abs(c1)
m =
    2.2361
» angulo=angle(c1)
angulo =
    1.1071
» angulo_graus=angle(c1)*180/pi
angulo_graus =
    63.4349

» parte_real=real(c1)
parte_real =
    1
» parte_imaginaria=imag(c1)
parte_imaginaria =
    2

```

❖ “Workspace” do MATLAB

- ❖ À medida que os comandos são introduzidos e as variáveis definidas, estes são guardados no “workspace”, podendo ser acedidos quando se desejar.

- Acesso ao valor de variáveis



```

> who
Your variables are:
ans      item1      item3
custo    item2      items

> item2
item2 =
      6

> whos
Name      Size      Bytes  Class
ans       1x1        8  double array
custo     1x1        8  double array
item1     1x1        8  double array
item2     1x1        8  double array
item3     1x1        8  double array
items     1x1        8  double array

Grand total is 6 elements using 48 bytes

```

- Alguns comandos úteis de gestão do “workspace”

Comando	Significado
who	lista de variáveis definidas no workspace
whos	“who” com informação de dimensão
↑ ↓	Ver comandos anteriores
← →	Mover dentro da linha de comando
Clear	Elimina todas as variáveis do workspace
Clear A	Elimina a variável A do workspace
Clc	Limpa a janela
Home	Move o cursor para o canto superior esquerdo

- ❖ Salvaguarda de dados e comandos do “workspace”
 - ❖ Todo o ambiente do “workspace” pode ser guardado em ficheiro e posteriormente recuperado.
- Através do menú “File”:



```
Save Workspace As...
Load Workspace...
```

- Através de comandos:

Comando	Significado
save	Guarda os valores das variáveis no ficheiro matlab.mat
save <i>fich</i>	Guarda os valores das variáveis no ficheiro <i>fich</i> .mat
save <i>fich item1 item2</i>	Guarda os valores das variáveis <i>item1</i> e <i>item2</i> no ficheiro <i>fich</i> .mat
save <i>fich item1</i> -ascii	Guarda os valores das variável <i>item1</i> no ficheiro <i>fich</i> .mat no formato ascii

❖ Ajuda sobre o MATLAB

- Através de menú de “Help”
- Através de comandos:



```
» help
HELP topics:      ...

» help sqrt
SQRT  Square root. SQRT(X) is ...
      See also SQRTM.

» lookfor complex
ctranspose.m: %'  Complex conjugate transpose.
CONJ  Complex conjugate.
CPLXPAIR Sort numbers into complex conjugate
pairs.
IMAG  Complex imaginary part.  ...
```

❖ Arrays❖ Vectores

- Definição por enumeração:



```
» x=[1 2 3 4 5 6 7 8 9]
» y=[0 .1*pi .2*pi .3*pi]
» w=[-1.3 sqrt(3) (1+3)*4/6]
```

- Endereçamento dos valores:



```
» x(3)
» x(1:5)
» x(3:-1:1)
```

- Outras formas de construção:



```
» y=(0: .1 :1)*pi
» y1=linspace(0,pi,11)
» y2=logspace(0,2,11)
» y=0:0.1:10;
» plot(y,sin(y))
```

- Sumário de formas de construção de vectores

```
X=[e11 e12 ... e1n]
X=valor_inicial:valor_final
X=valor_inicial:incremento:valor_final
X=linspace(valor_inicial,valor_final,numero_elementos)
X=logspace(expoente_inicial,expoente_final,numero_elementos)
```

- Vectores coluna:



```
» z=x'
```

❖ Matrizes

- Definição por enumeração:



```
» a=[1 2 3; 4 5 6; 7 8 9]
```

- Operações sobre matrizes / vectores:



```

» g=[1 2 3 4 ; 5 6 7 8 ; 9 10 11 12]
» h=[1 1 1 1 ; 2 2 2 2; 3 3 3 3]

» g-2,g+4,g*3 %operações com escalares
» g+h
» g-h
» g.*h
» g./h
» g.\h
» g.^2
» 2.^g
» g.^h

```

- Manipulação de arrays:



```

» a=[1 2 3; 4 5 6; 7 8 9]
» a(3,3)

» c=a
» c(3,3)=0

» b=a(3:-1:1,1:3)

» c=[1 3]
» b=a(c,c)

» b(:,2)=[] %eliminar uma coluna

» b=a(:,[2 2 2 2])

» b=[1 4 7];
» c=b([1 1 1 1],:)

» b(3:4,:)=a(2:3,:)

» x=-3:3
» x(abs(x)>1)=[]

» b=[5 -3;2 -4]
» x=abs(b)>2

```

- Sumário de endereçamento de *arrays* (preencha o campo de significado):

Comando	Significado
A(r,c)	
A(r,:)	
A(:,r)	
A(:)	
A(i)	
A(x)	

- Algumas funções e operações com matrizes

- Resolução de uma equação linear: $Ax=B$



```

» A=[1 2 3; 4 5 6; 7 8 0]
» B=[ 366;804;351]
» det(A)
» x=inv(A)*B

```

- Exemplo com inversão de matrizes:



```

» n=100;
» a=rand(n);
» imagesc(a)
» b=inv(a);
» imagesc(b);
» imagesc(a*b);

```

- Matrizes especiais:



```

» zeros(3)
» ones(2,4)
» rand(3,1)
» randn(2)
» eye(3)
» ones(size(A))

```

- Funções para análise de dados:

Função	Significado
Cov(x)	Matriz de covariância
Histogram(x)	histograma
Max(x), max(x,y)	Máximo componente
Mean(x)	Valor médio das colunas
Median(x)	Valor mediana das colunas
Min(x), min(x,y)	Mínimo componente
Rand(x)	Números aleatórios uniformemente distribuídos
Randn(x)	Números aleatórios com distribuição gaussiana
Sort(x)	Ordena as colunas por ordem crescente
Std(x)	Desvio padrão das colunas
Sum(x)	Soma dos elementos de cada coluna

❖ Gráficos básicos

- Comando *plot*



```

» x=0:.1:10;
» y=sin(x);
» plot(x,y)

» t=0:1/10:5;
» plot(t,sin(t.^2),'g')
» title('chirp Function');
» ylabel('sin(t.^2),g')
» xlabel('t')

```

❖ Programação em MATLAB: funções

As funções em MATLAB escrevem-se em ficheiros com a extensão .m (ex: plot_sprectr.m) na directoria actual. Quando se chama a função o MATLAB lê o ficheiro e executa o código que aí se encontra.

O código é escrito da seguinte forma:

```
% comentários sobre o ficheiro. Esta informação é impressa quando se executa a seguinte
linha de comando no Matlab: help <nome_da_função>
```

```
%
```

```
function spec = plot_sprectr (signal, fs, N)
```

```
< código da função >
```

```
spec = <resultado do cálculo>
```

O código é executado quando se escreve: `spec = plot_sprectr (signal, fs, N);`

Quando é necessário retornar mais do que um valor, pode-se guardar os valores a retornar num vector ou num array, ou associar cada valor a uma variável da seguinte forma:

```
function [var1, var2, <...>, varn] = calcula_n_entidades (arg1, arg2)
```

- Escreva uma função para calcular o valor máximo, mínimo e médio de um vector



Forma de chamada: `[minv, maxv, av] = mmav(vec)`

Sugestão: use os comandos do matlab: `max`, `min` e `mean`.