

Introdução ao uso de Matlab



Adriano Martins Moutinho

Eugênio Silva

Inteligência Computacional - 2004

Matlab

- Matrix Laboratory, desenvolvido inicialmente pelas universidades de Stanford e novo México.
- Escrito para trabalhar com vetores e matrizes
- Tornou-se ferramenta necessária para a maioria das pesquisas em tecnologia.

Introdução ao Matlab

- Possui um ambiente amigável e de grande capacidade de integração.
- Possui a maioria das funções matemáticas mais utilizadas.
- Fácil utilização. Necessita de pouca implementação do usuário.

Introdução ao Matlab

- Elemento básico de programação do Matlab é a matriz!
- Todas as operações matriciais do Matlab são implementadas automaticamente. Multiplicação, divisão, inversão, soma, subtração, etc...

Aplicações

- Matemática e computação
- Desenvolvimento de algoritmos
- Modelagem, simulação e prototipação
- Análise de dados, exploração e visualização
- Gráficos científicos em duas e três dimensões

Exemplo do uso de Matlab – Soma de escalar com matriz 3x3

Pascal

var

a : array [1..3, 1..3] of integer;

i, j : integer;

begin

... (Entrar com a matriz!)

for i = 1 to 3

for j = 1 to 3

 b(i,j) := b(i,j) + 1;

end;

end;

end.

Matlab

```
>> a = [1 2 3; 4 5 6; 7 8 9]
```

```
>> a = a + 1;
```

Operações elementares

```
>> 4 + 8 - 4
```

```
ans =
```

```
8
```

```
>> 4*3 + 6*2 + 1*9
```

```
ans =
```

```
33
```

```
>> 25 / 5 + 1 / 25
```

```
ans =
```

```
5.0400
```

```
>> 9 ^ 2 * 2 ^ 3 + 2
```

```
ans =
```

```
650
```

Definição de variáveis

- Variáveis são definidas ao digitar o seu valor:

```
>> a = 1
```

```
a =
```

```
1
```

```
>> a = 1 + j*2
```

```
a =
```

```
1.0000 + 2.0000i
```

```
>> a = [1 1 1 1]
```

```
a =
```

```
1 1 1 1
```

```
>> a = [1 1 1 1; 1 1 1 1]
```

```
a =
```

```
1 1 1 1
```

```
1 1 1 1
```


Espaço de trabalho

>> workspace - Inicia janela de workspace

>> who - lista variáveis existentes na memória

>> whos - lista variáveis com informações detalhadas

>> clear - apaga as variáveis da memória

>> save - salva as variáveis em um arquivo .mat

>> load - carrega as variáveis contidas em um .mat

>> format - Formato de apresentação dos resultados

Funções Elementares

>>cos – Cosseno

>>abs - Valor absoluto

>>sin – Seno

>>inv - Matriz inversa

>>acos - Arco cosseno

>>log2 - Log base 2

>>asin - Arco seno

>>log10 – Log base 10

>>inv - Matriz inversa

>>log – Logaritmo natural

>>tan - Tangente

>>round - Arredondamento

>>exp - Exponencial

>>floor - Piso

Comandos do Sistema Operacional

>>quit ou >>exit - sai do ambiente matlab

>>type <arquivo> - mostra o conteúdo do arquivo

>>edit <arquivo> - abre arquivo para edição

>>delete <arquivo> - apaga arquivo

>> cd \ - diretório raiz

>> cd . . - diretório acima do corrente

Trabalhando com vetores

- Criando vetores:

```
>>a = [0 1 2 3 4 5 6 7 8 9 10]
```

```
>>a = 0:10
```

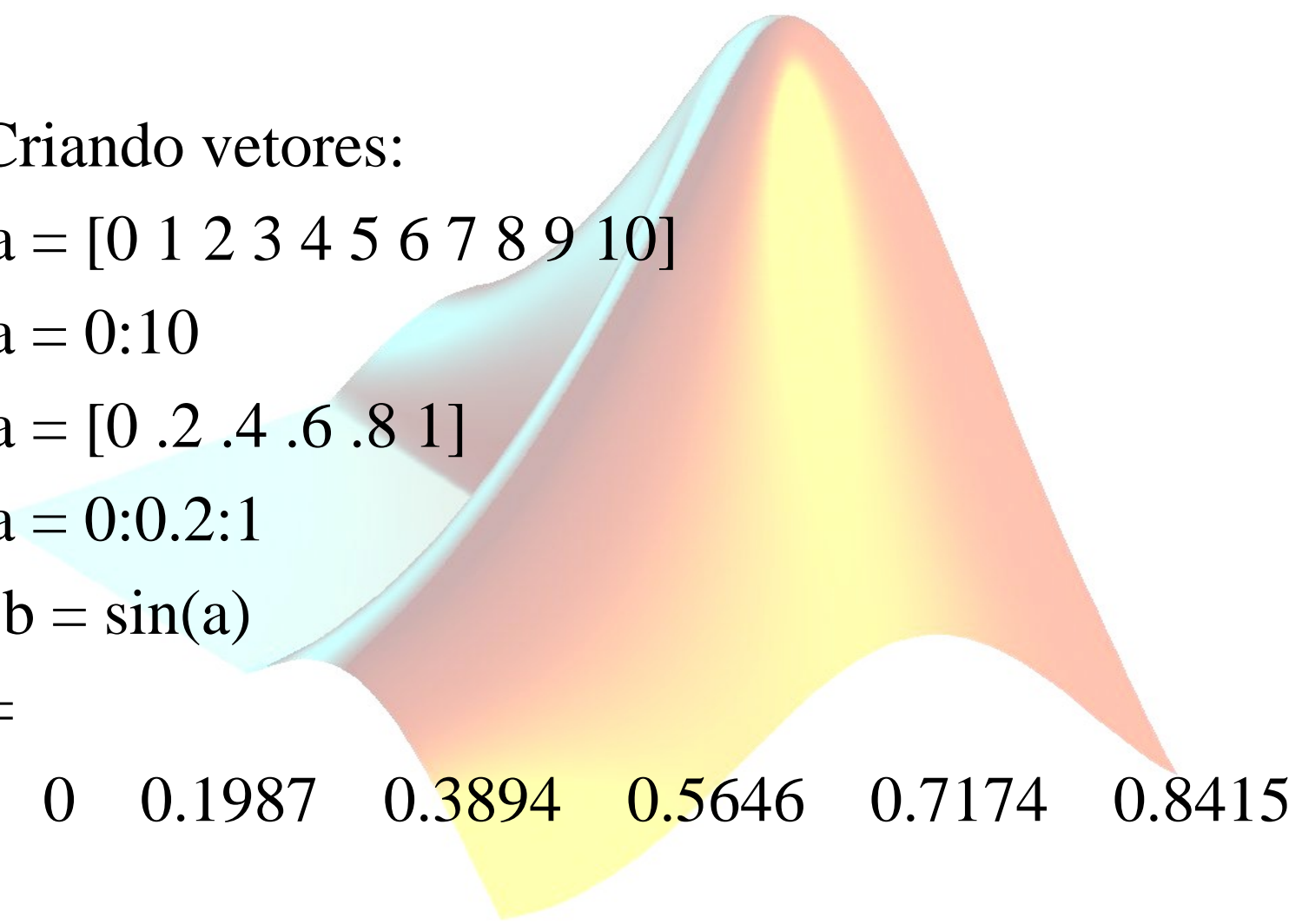
```
>>a = [0 .2 .4 .6 .8 1]
```

```
>>a = 0:0.2:1
```

```
>> b = sin(a)
```

```
b =
```

```
0 0.1987 0.3894 0.5646 0.7174 0.8415
```



Operações com vetores

```
>> x = [1 2 3 4]
```

```
>> y = [7 6 5 4]
```

```
>> x(1:1:3)
```

```
ans =
```

```
1 2 3
```

```
>> x(3:-1:1)
```

```
ans =
```

```
3 2 1
```

```
>> t = [x;y]
```

```
ans =
```

```
1 2 3 4
```

```
7 6 5 4
```

```
>> x'
```

```
ans =
```

```
1
```

```
2
```

```
3
```

```
4
```

Operações com matrizes

```
>> x = [1 2;3 4;5 6]
```

```
ans = 1 2  
      3 4  
      5 6
```

```
>> x(1:2,1:2)
```

```
ans =  
 1 2  
 3 4
```

```
>>x(:,2)
```

```
ans = 2  
      4  
      6
```

```
>>x(1,:)
```

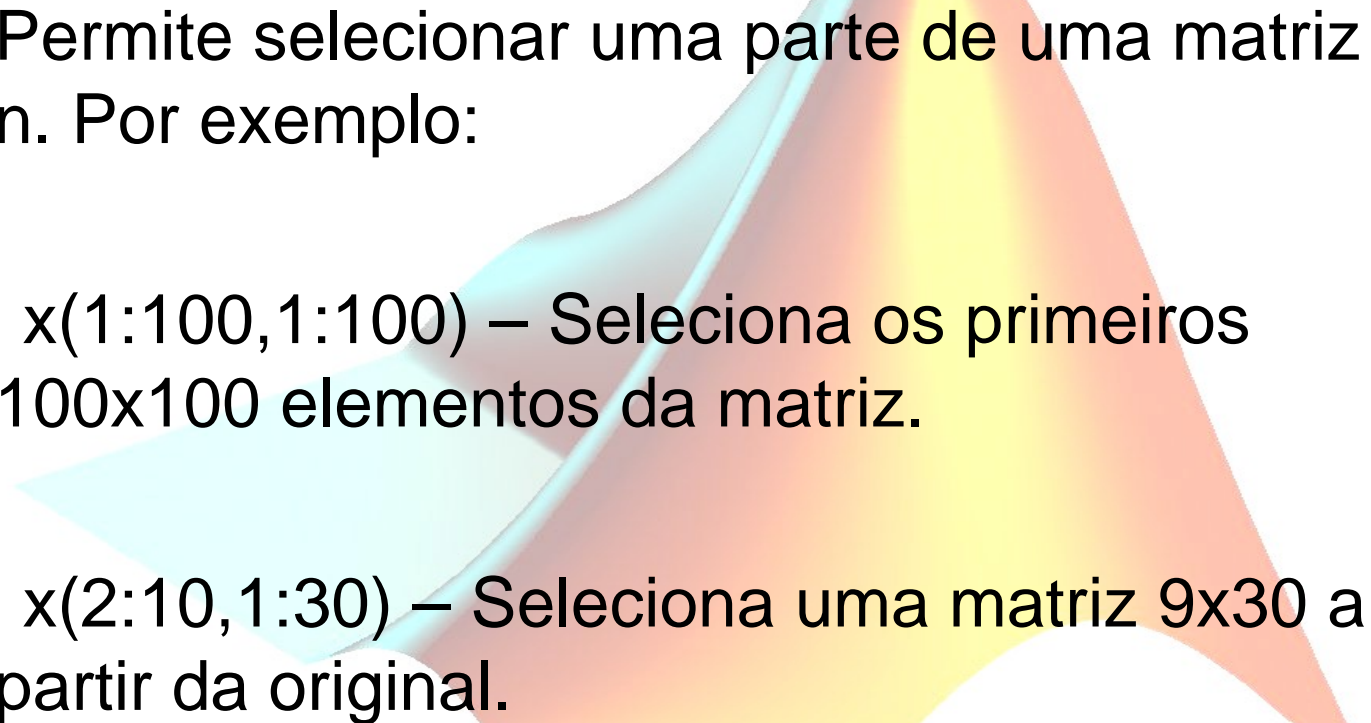
```
ans = 1 2
```

```
>>x(1,:) = x(2,:) + 1;
```

```
>> x
```

```
x =  
 4 5  
 3 4  
 5 6
```

Comando **col** ou :

- Permite selecionar uma parte de uma matriz $n \times n$. Por exemplo:
 - >> `x(1:100,1:100)` – Seleciona os primeiros 100×100 elementos da matriz.
 - >> `x(2:10,1:30)` – Seleciona uma matriz 9×30 a partir da original.
- 

Comando *find*

- Retorna os índices da matriz a partir de uma estrutura de busca.

```
>> x = [1 2 3;4 5 6;7 8 9];
```

```
>> find(x == 1)
```

```
ans = 1;
```

```
>> find(x > 5)
```

```
ans = 3
```

```
6
```

```
8
```

```
9
```

```
>> y = find(x == 1);
```

```
>> x(y) = 0
```

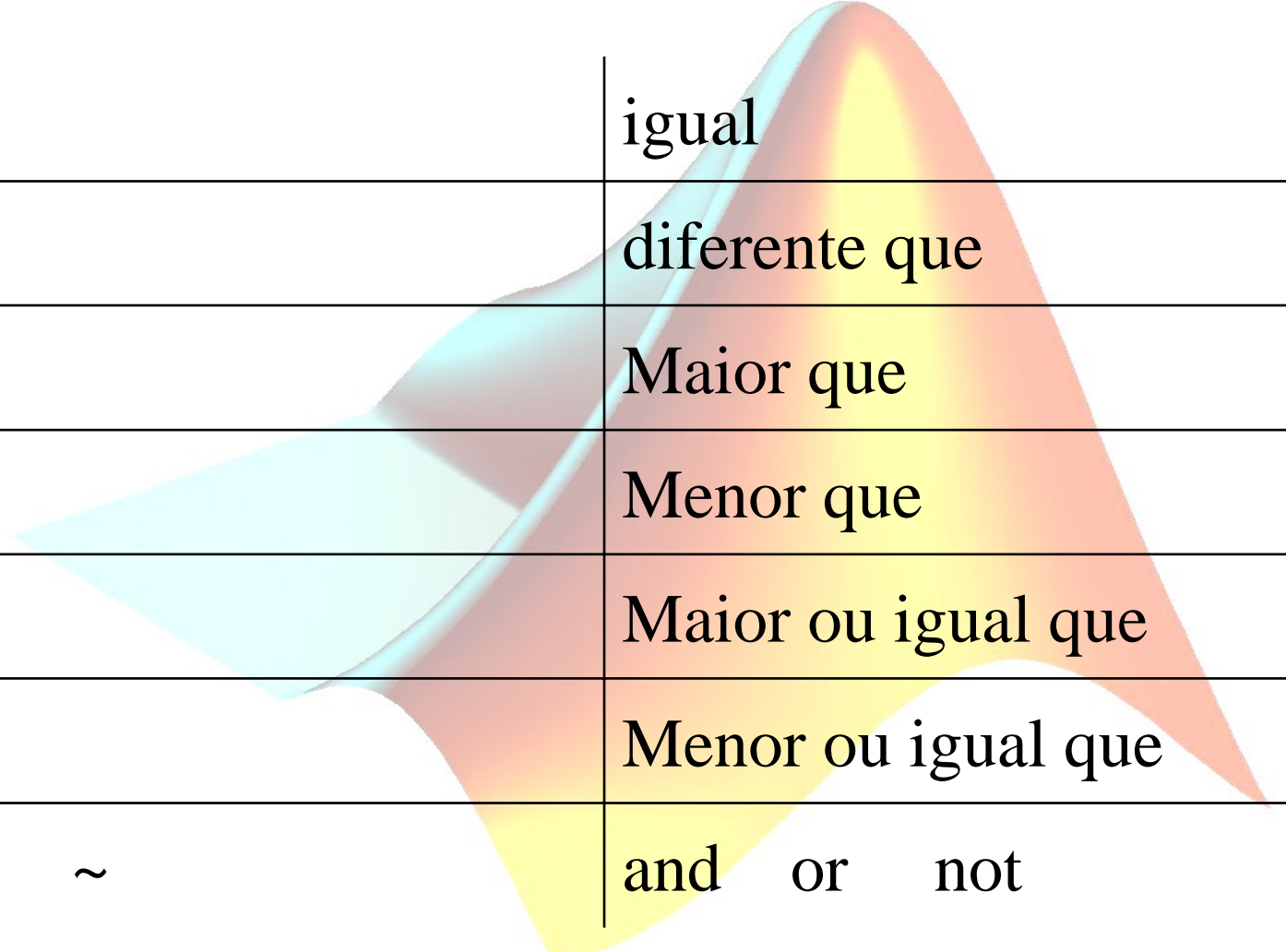
```
x =
```

```
1 2 3
```

```
4 5 0
```

```
0 0 0
```


Operadores lógicos



==	igual
~=	diferente que
>	Maior que
<	Menor que
>=	Maior ou igual que
<=	Menor ou igual que
& ~	and or not

Dimensões de Vetores e Matrizes

<code>size(x)</code>	Tamanho da matrix x
<code>size(x,2)</code>	Número de colunas de x
<code>size(x,1)</code>	Número de linhas de x
<code>length(x)</code>	Número de elementos de um vetor x
<code>ndims(x)</code>	Número de dimensões de x
<code>workspace</code> ou <code>whos</code>	Mostra as dimensões de todas as variáveis.

Operações entre matrizes e vetores

- No ambiente Matlab, $X * Y$ significa uma multiplicação matricial onde o número de linhas em X precisa ser igual ao número de colunas em Y .
- $X .* Y$ significa uma multiplicação de todos os elementos da matriz X pela matriz Y .
- A mesma diferença se obtém em:
 - $./$
 - $.^$

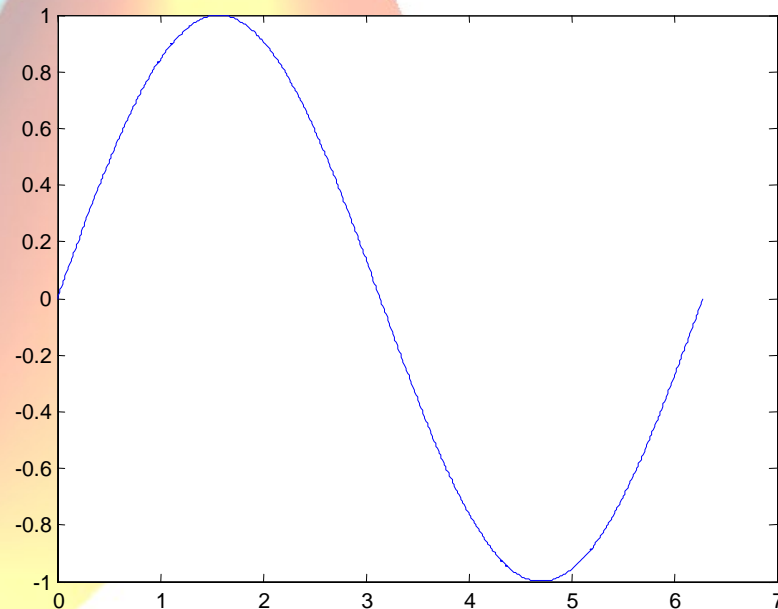
Gráficos

- `plot`: comando usado para plotar qualquer vetor:

```
>> x = [0:0.01:2*pi];
```

```
>> y = sin(x)
```

```
>> plot(x,y)
```



Gráficos

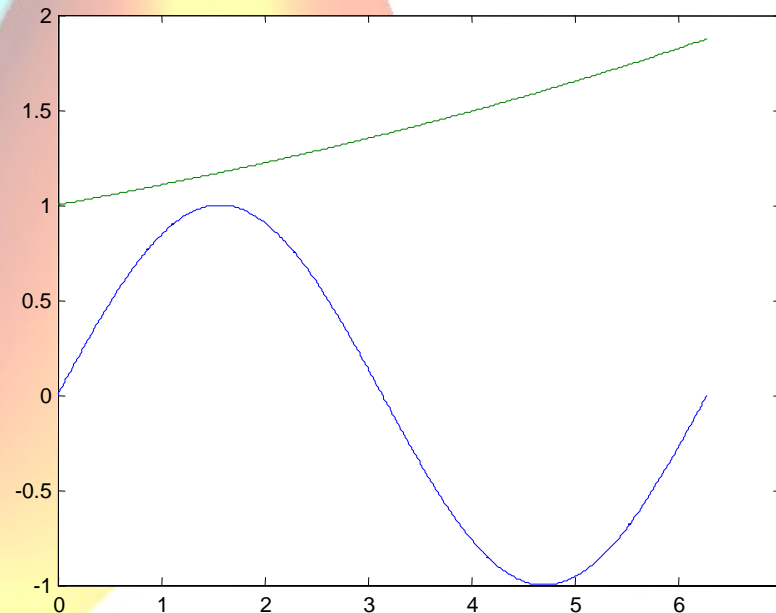
- `plot`: comando usado para plotar qualquer vetor:

```
>> x = [0:0.01:2*pi];
```

```
>> y = sin(x)
```

```
>> z = exp(x/10)
```

```
>> plot(x,y,x,z)
```



Ajuda

- help comando
 - Ajuda em um determinado comando.
 - helpwin comando
 - Ajuda em um determinado comando em uma nova janela.
- 