

# INTRODUÇÃO AO CÁLCULO NUMÉRICO

## LABORATÓRIO – AULA 01

### INTRODUÇÃO AO SCILAB

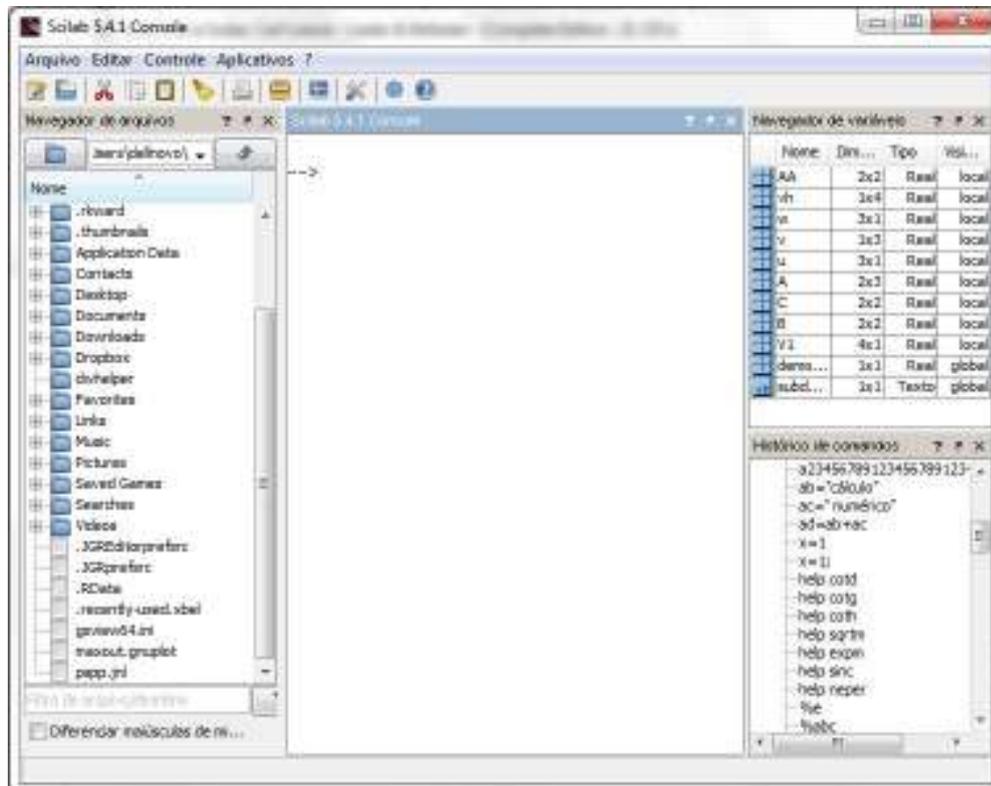
#### 1. INTRODUÇÃO

As atividades de laboratório do Curso de Cálculo Numérico serão desenvolvidas empregando o software Scilab, que é um ambiente numérico, gráfico e programável, desenvolvido pelo Institut Nationale de Recherche en Informatique et en Automatique (INRIA), da França; seu uso é livre e está disponível para plataformas Linux, Windows e Mac.

Se você deseja instalar o Scilab em seu computador pessoal, basta acessar a página <http://www.scilab.org/index.php/download/> e selecionar o programa de instalação adequado a seu sistema operacional. Assim, para uma plataforma Windows-64 bits, será feito o download do arquivo “scilab-5.4.1\_x64.exe” e, a seguir, deve-se executar o procedimento usual de instalação de programas no Windows.

## 2. USANDO O SCILAB

Ao executar o Scilab, é exibida a seguinte janela:



Para esta aula de laboratório, interessa-nos apenas a janela central, chamada Console. É aqui que todos os comandos são executados e os resultados são visualizados. De tempos em tempos, é recomendável limpar o conteúdo da Console, através do comando *clc*.

Mais adiante, veremos que a partir do item Aplicativos do Menu, temos acesso a um Editor de Texto do próprio Scilab, chamado SciNotes e que será importante para a manipulação de funções e scripts.

Veremos, ainda, que gráficos são exibidos em janelas separadas. Outros recursos da interface gráfica do Scilab serão abordados quando necessário.

## 3. ELEMENTOS BÁSICOS DA LINGUAGEM

### 3.1. NOMES DE VARIÁVEIS

Na composição dos nomes das variáveis são permitidos todos os caracteres alfanuméricos (os caracteres %, \_, #, !, \$ e ? também são permitidos, mas o caractere % não deve iniciar o nome). O Scilab distingue caracteres maiúsculos e minúsculos, de modo que *var1*, *Var1*, *VAR1*, por exemplo, são variáveis distintas.

### 3.2. VARIÁVEIS MATEMÁTICAS PRÉ-DEFINIDAS

No Scilab, diversas variáveis são pré-definidas e seus nomes começam com o caracter % e, por esse motivo, os nomes de variáveis do usuário não podem começar com %. Alguns exemplos são:

%pi	$\pi$
%i	O número imaginário, i
%e	Constante de Napier ou num. de Euler

### 3.3. ATRIBUIÇÃO DE VALORES

O Scilab é uma linguagem interpretada e, por isso, não é necessário declarar a variável antes de usá-la (como é o caso de linguagens compiladas como o C). A variável é criada no momento em que lhe atribuímos um valor. Assim, no exemplo abaixo, criamos a variável real x e lhe atribuímos o valor 1.

```
-->x=1
x =
1.
```

Observe que toda vez que um comando é executado, o valor da variável é exibido. Em muitas ocasiões não nos interessa exibir o valor de uma determinada variável e, isso pode ser conseguido terminando a linha de comando com o caracter “;”, como no exemplo abaixo:

```
-->x=1;
-->
```

#### ATIVIDADE PRÁTICA

**Crie algumas variáveis e atribua valores a elas usando, ou não, o caractere ; no final da linha.**

**Em particular, verifique que o Scilab distingue letras maiúsculas de minúsculas. Para tanto, faça as seguintes atribuições (observe os “;”):**

```
-->var1=10;
-->Var1=20;
-->VAR1=30;
```

**Agora, exiba o valor dessas variáveis através dos comandos:**

```
-->var1
-->Var1
-->VAR1
```

**Comprove que os valores numéricos são diferentes e, portanto, são variáveis distintas.**

Na atribuição de valores a variáveis correspondentes a números reais, pode-se usar a notação decimal ou a notação científica. Por exemplo,

```
-->Vdecim = 0.00002
```

```
Vdecim =
```

```
0.00002
```

```
-->Vscient = 2e-5
```

```
Vscient =
```

```
0.00002
```

### 3.4. COMENTÁRIOS E CONTINUAÇÃO DE LINHAS

Qualquer linha iniciada com duas barras “//” é considerada pelo Scilab como um comentário e não como um comando a ser executado e, portanto, o conteúdo dessa linha é ignorado. No Scilab, entretanto, não é possível comentar um bloco inteiro de linhas, isto é, não existe, por exemplo, o recurso da linguagem C de colocar um bloco de comentários entre “/\* ... \*/”.

#### ATIVIDADE PRÁTICA

**Execute o comando:**

```
-->// Comentário: Nada será executado
```

Quando um comando é longo demais para caber em uma única linha ele pode ser subdividido em duas ou mais linhas. Sempre que uma linha terminar com dois pontos “..”, o Scilab entende que a próxima linha é uma continuação da anterior e não um novo comando.

#### ATIVIDADE PRÁTICA

**Execute o comando:**

```
-->x =1..
```

```
-->+2..
```

```
-->+3..
```

```
-->+4
```

**O Scilab deve entender que se trata de um único comando e realizar a soma  $1+2+3+4=10$ . Comprove.**

### 3.5. OPERAÇÕES ARITMÉTICAS

O Scilab oferece um elenco de operações aritméticas elementares como qualquer calculadora. São elas:

+	Adição
-	Subtração
*	Multiplicação
/	Divisão pela Direita ( $x/y$ ou $xy^{-1}$ )
\	Divisão pela Esquerda ( $x\backslash y$ ou $x^{-1}y$ )
^	$x^y$ Potenciação ( $x^y$ )
**	Potenciação (o mesmo que ^)

#### ATIVIDADE PRÁTICA

Exercite livremente algumas das operações aritméticas.

Usando os operadores aritméticos, como você calcularia a raiz n-ésima de x?

### 3.6. FUNÇÕES MATEMÁTICAS ELEMENTARES

O Scilab oferece um grande número de funções matemáticas básicas como funções trigonométricas, exponenciais, logarítmicas, etc, como qualquer calculadora científica. Nos quadros abaixo, é apresentada uma lista (incompleta) das funções matemáticas disponíveis.

```
acos  acosd  acosh  acoshm  acosm  acot  acotd  acoth
acsc  acscd  acsch  asec   asecd  asech  asin   asind
asinh asinhm  asinm  atan   atand  atanh  atanm  atanm
cos   cosd   cosh   coshm  cosm  cotd   cotg   coth
cothm csc    cscd  csch   sec    secd  sech   sin
sinc  sind  sinh  sinhm  sinm  tan   tand  tanh
tanhm tanm
```

```
exp  expm  log  log10  logip  log2  logm  max
maxi  min  mini  modulo  pmodulo  sign  signm  sqrt
sqrtm
```

#### ATIVIDADE PRÁTICA

Experimente usar algumas das funções como as trigonométricas, exponencial e logarítmicas, para argumentos bem conhecidos. Ex.:  $\sin(\pi/2)$ , etc.

### ATIVIDADE PRÁTICA

Aproveitemos a ocasião para conhecermos um pouco melhor o recurso de Ajuda do Scilab. Na lista de funções, algumas são facilmente reconhecíveis, como *cos*, *sin*, *exp* e outras. Mas o que faz a função *sinc* ?

É aqui que entra a Ajuda, o Help do Scilab. Para acessar esse recurso, estando ativa a janela do Scilab, no Menu Principal, selecione a opção ? e, a seguir, Ajuda do Scilab; ou, então, pressione diretamente a tecla F1. Uma janela contendo o Sistema de Ajuda Completo é exibida e, nela, podemos navegar e conhecer todos os recursos do Scilab.

Se, como no caso da função *sinc*, conhecemos o nome do recurso podemos simplesmente executar, na console, o comando

```
-->help sinc
```

Execute esse comando e descubra que *sinc(x)* nada mais é do que *sin(x)/x*. Experimente o comando *help* para outras funções e também com o Navegador de Ajuda Completo.

### 3.7. VARIÁVEIS LÓGICAS (“BOOLEANAS”)

Variáveis lógicas são utilizadas para armazenar os valores VERDADEIRO ou FALSO que são escritos, respectivamente, como %t (ou %T) e %f (ou %F). Além disso, o Scilab fornece vários operadores lógicos e de comparação que resultam em variáveis lógicas, ou seja, em valores VERDADEIRO ou FALSO. O quadro abaixo apresenta o elenco de operadores lógicos:

$a\&b$	Operador E lógico (AND)
$a b$	Operador OU lógico (OR)
$\sim a$	Negação lógica (NOT)
$a==b$	Verdadeiro se a e b iguais
$a\sim=b$ ou $a<>b$	Verdadeiro se a e b diferentes
$a<b$	Verdadeiro se a menor que b
$a>b$	Verdadeiro se a maior que b
$a\leq b$	Verdadeiro se a menor ou igual a b
$a\geq b$	Verdadeiro se a maior ou igual a b

### ATIVIDADE PRÁTICA

Atribua o valor VERDADEIRO a uma variável, digamos, *a*;  
Teste a igualdade entre os números 0 e 1 e atribua o resultado à variável *b*;  
Execute a operação lógica *a&b*

Qual é o resultado encontrado?

### 3.8. NÚMEROS COMPLEXOS

Usando a variável pré-definida `%i`, os números complexos podem ser representados facilmente. Por exemplo, o número  $5+2i$  é definido, no Scilab, como `x = 5 + 2*%i`. O operador `'` é utilizado para calcular o complexo conjugado.

Algumas operações básicas podem ser aplicadas aos números complexos:

<code>real</code>	Retorna a parte real
<code>imag</code>	Retorna a parte imaginária
<code>imult</code>	Multiplica o número por $i$
<code>isreal</code>	Verdadeiro se parte imaginária for zero

#### ATIVIDADE PRÁTICA

Execute os comandos e verifique os resultados:

```
-->x=1+%i
-->isreal(x)
-->y=1-%i
-->real(y)
-->imag(y)
-->x*y
```

### 3.9. A VARIÁVEL *ans*

Toda vez que executamos um cálculo e o resultado não é armazenado explicitamente em uma variável especificada, o Scilab armazena esse resultado em uma variável padrão denominada *ans*. Essa variável pode ser usada em outros cálculos desde que algum valor numérico lhe tenha sido previamente atribuído.

A variável *ans* pode ser útil, por exemplo, quando já realizamos vários cálculos mas esquecemos de atribuir o valor da última operação a uma variável; sabemos que o resultado está armazenado em *ans* e podemos recuperá-lo. No entanto, se estivermos programando uma função ou script, é recomendável não empregar a variável *ans* e, sim, tomar o cuidado de definir variáveis e lhes atribuir os resultados de cada cálculo.

#### ATIVIDADE PRÁTICA

Execute os comandos:

```
-->exp(3)
-->t=log(ans)
```

e comprove o que se afirmou no parágrafo anterior.

### 3.10. CADEIAS DE CARACTERES ("STRINGS")

Strings podem ser armazenadas em variáveis, desde que delimitadas por aspas duplas ("). Duas strings podem ser concatenadas usando o operador +.

#### **ATIVIDADE PRÁTICA**

**Execute os comandos:**

```
-->ab="cálculo"
```

```
-->ac=" numérico"
```

```
-->ad=ab+ac
```

**e verifique que o resultado é "cálculo numérico".**

## 4. MATRIZES E VETORES

No Scilab, o objeto básico é a *Matriz*, um objeto matemático definido por seu número de linhas, número de colunas e tipo do dado armazenado em cada elemento da matriz, que pode ser um número inteiro ou real, uma variável lógica, uma cadeia de caracteres, etc.

Para o Scilab, todos os objetos são matrizes. Assim, vetores são matrizes de dimensão (nx1) (vetor coluna) ou (1xn) (vetor linha) e números usuais são matrizes de dimensão (1x1).

### 4.1. CRIANDO VETORES E MATRIZES

O Scilab possui uma sintaxe simples para a criação de vetores e matrizes. Os seguintes símbolos são empregados na definição de uma matriz:

- . colchetes [ ] delimitam o início e o fim da matriz;
- . vírgulas “,” separam os valores em diferentes colunas;
- . pontos e vírgulas “;” separam os valores de diferentes linhas

#### ATIVIDADE PRÁTICA

**Crie a matriz:**

**A = [1, 2, 3; 4, 5, 6]**

**Crie a matriz**

**A=[1,2;3,4;5,6]**

**Crie um vetor linha ( , separando elementos)**

**V1=[1,2,3,4]**

**Crie um vetor coluna (; separando elementos)**

**V1=[1;2;3;4]**

**Verifique o resultado.**

**A =**

**1. 2. 3.  
4. 5. 6.**

**Verifique o resultado.**

**A =**

**1. 2.  
3. 4.  
5. 6.**

**O resultado é:**

**V1 =**

**1. 2. 3. 4.**

**O resultado é:**

**V1=**

**1.  
2.  
3.  
4.**

Uma sintaxe alternativa e simples não requer o uso de vírgulas e ponto e vírgulas. Neste caso, espaços em branco separam as colunas e as linhas são separadas por <ENTER>.

#### ATIVIDADE PRÁTICA

**Crie a matriz (após o elemento de valor igual a 3, digite <ENTER>)**

-->A = [1 2 3

-->4 5 6]

A =

1. 2. 3.  
4. 5. 6.

## 4.2. MATRIZES ESPECIAIS

Para alguns tipos de matrizes usadas frequentemente o Scilab oferece comandos específicos para sua criação. Nestes casos, é preciso fornecer apenas as dimensões da matriz.

Comando	Resultado
eye(n,m)	Gera a matriz Identidade
zeros(n,m)	Inicializa todos os elementos com 0
ones(n,m)	Inicializa todos os elementos com 1

#### ATIVIDADE PRÁTICA

**Escolha alguns valores para as dimensões n e m da matriz e execute os comandos do quadro acima. Observe os resultados.**

## 4.3. OPERAÇÕES COM VETORES E MATRIZES

Todos os operadores algébricos usuais também estão disponíveis para matrizes quaisquer. O quadro abaixo apresenta o elenco de operadores e o resultado de sua ação. Observe que, na coluna da esquerda, os operadores estão representados pelos símbolos usuais, enquanto na coluna da direita os símbolos dos operadores são precedidos por um ponto “.”. Um ponto antes do símbolo do operador significa que a operação é feita elemento por elemento. Veremos mais sobre isto logo adiante.

+ Adição	
- Subtração	
* Multiplicação	.*
/ Divisão pela Direita	./
\ Divisão pela Esquerda	.\
	.^
' Transposta e Conjugado	.' Transposto (mas não Conjugado)

Vejamos cada operação em detalhe.

### Adição e Subtração

A adição (ou subtração) de duas matrizes é, inerentemente, uma operação elemento por elemento (se  $C=A+B$ ,  $C_{ij}=A_{ij}+B_{ij}$ ); por isso, o operador .+ sequer é definido.

#### ATIVIDADE PRÁTICA

**Crie duas matrizes de mesma dimensão e faça sua adição.  
Tente, agora, somar duas matrizes de dimensões diferentes.**

### Multiplicação

A multiplicação usual entre duas matrizes, representada aqui pelo operador "\*" é aquela em que os elementos da linha  $i$  são multiplicados pelos elementos da coluna  $j$  e somados. Simbolicamente, se  $C=A*B$ , então  $c_{ij}=\sum_{k=1,n}a_{ik}b_{kj}$ ; por sua vez, o operador .\* é executado elemento por elemento de forma que se  $C= A.*B$ , então,  $c_{ij}=a_{ij}*b_{ij}$ .

#### ATIVIDADE PRÁTICA

**Crie as matrizes  $A=[1 \ 2; 3 \ 4]$  e  $B=[1 \ 2; 3 \ 4]$ .  
Execute as operações  $A*B$  e  $A.*B$ .  
Compare os resultados.**

### Divisão

Abordaremos os dois operadores / e \ mais tarde, no contexto da resolução de sistemas de equações lineares. No entanto, como os operadores ./ e .\ são executados elemento por elemento, seu significado já foi visto no item 3.5.

#### ATIVIDADE PRÁTICA

**Crie as matrizes  $A=[1 \ 2; 3 \ 4]$  e  $B=[1 \ 2; 3 \ 4]$ .  
Execute as operações  $A./B$  e  $A.\B$ .  
Compare os resultados.**

### Potenciação

No Scilab não está definida a operação  $A^B$  onde A e B são matrizes. Apenas a potenciação elemento por elemento,  $.^{\wedge}$ , está definida. Neste caso, se  $C=A.^{\wedge}B$  então  $C_{ij}=a_{ij}^{\wedge}b_{ij}$ .

#### ATIVIDADE PRÁTICA

**Crie as matrizes  $A=[1 \ 2; \ 3 \ 4]$  e  $B=[1 \ 2; \ 3 \ 4]$ .  
Execute a operação  $A.^{\wedge}B$ .**

### Transposta e Conjugado

O operador  $'$  resulta na transposição da matriz e, se seus elementos forem números complexos, são substituídos pelos respectivos complexos conjugados. O operador  $./$  resulta apenas na matriz transposta. Se a matriz for real, os dois operadores têm resultado idêntico.

#### ATIVIDADE PRÁTICA

**Crie a matriz (real)  $A=[5 \ 4 \ 3; \ 2 \ 1 \ 7]$   
Execute as operações  $A'$  e  $A.{'}$ . Compare os resultados.**

**Crie a matriz (complexa)  $B=[1+ \%i \ 2+ \%i; \ 3+ \%i \ 4+ \%i]$ .  
Execute as operações  $B'$  e  $B.{'}$ . Compare os resultados.**

## 4.4. MULTIPLICAÇÃO DE DOIS VETORES

Um cuidado especial deve ser tomado quando se multiplicam dois vetores no Scilab. Isto porque, como vimos, o Scilab distingue vetores linha e vetores coluna. Um vetor coluna de 3 elementos é, para o Scilab, uma matriz de dimensão (3x1), enquanto um vetor linha de 3 elementos é uma matriz de dimensão (1x3). Conseqüentemente, se efetuamos a multiplicação  $u*v$ , o resultado é uma matriz de dimensão (3x3). Pela mesma razão, ocorrerá um erro toda vez que multiplicarmos dois vetores coluna (ou dois vetores linha), uma vez que as dimensões são incompatíveis do ponto de vista da multiplicação matricial.

#### ATIVIDADE PRÁTICA

**Crie os três vetores seguintes:  
 $u = [1;2;3]$  (coluna)  
 $v = [4 \ 5 \ 6]$  (linha)  
 $w=[4;5;6]$  (coluna)**

**Execute  $u*v$  e  $u*w$ .  
Verifique os resultados.**

#### 4.5. FUNÇÕES MATEMÁTICAS ELEMENTARES APLICADAS A VETORES E MATRIZES

Como o tipo de dado básico do Scilab é uma matriz, as operações matemáticas apresentam características pouco usuais, às quais não estamos acostumados. Assim, por exemplo, se  $V$  é um vetor, é legítimo determinar o seno de  $V$ . Para o Scilab, as funções, quando aplicadas a vetores ou matrizes, operam elemento por elemento.

##### ATIVIDADE PRÁTICA

**Crie a matriz:**

**$A = [0 \ \pi/2; \ \pi \ 3\pi/2]$**

**Execute  $\sin(A)$  e verifique o que acontece.**

## 5. GRÁFICOS

O Scilab permite a criação de vários tipos de gráficos em 2D e 3D. Aqui veremos apenas um tipo de gráfico e alguns dos recursos de configuração a ele associados. A única forma de conhecer todos os recursos gráficos do Scilab é explorá-los, com o auxílio do Help.

### 5.1. GRÁFICO X-Y

Uma situação particularmente comum é encontrada quando se deseja construir o gráfico de uma função matemática conhecida.

Por exemplo, vamos supor que queremos fazer o gráfico da função  $y=x^2$ , no intervalo  $x=[1,10]$ . A função do Scilab que constrói esse gráfico é

`plot(x,y)`

onde  $x$  é o vetor das abscissas e  $y$  o vetor das ordenadas, contendo a função avaliada em cada um dos elementos do vetor  $x$ . Mas como construir o vetor  $x$ ?

Quando se está fazendo o gráfico de uma função conhecida é usual selecionar as coordenadas  $x$  igualmente espaçadas, dentro do intervalo desejado. Para não ter que criar esse vetor  $x$  manualmente, o Scilab oferece a função

`linspace(x1,x2,n)`

onde  $x1$  e  $x2$  são os extremos do intervalo desejado e  $n$  é o número de pontos igualmente espaçados nesse intervalo (o parâmetro  $n$  é opcional, se não for fornecido o Scilab usa o padrão  $n=100$ ). A função `linspace` retorna um vetor com  $n$  coordenadas (abscissas).

### ATIVIDADE PRÁTICA

Construa o gráfico da função  $y=x^2$  no intervalo  $[1,10]$  usando 50 pontos igualmente espaçados.

Digite a seguinte sequência de comandos na console do Scilab:

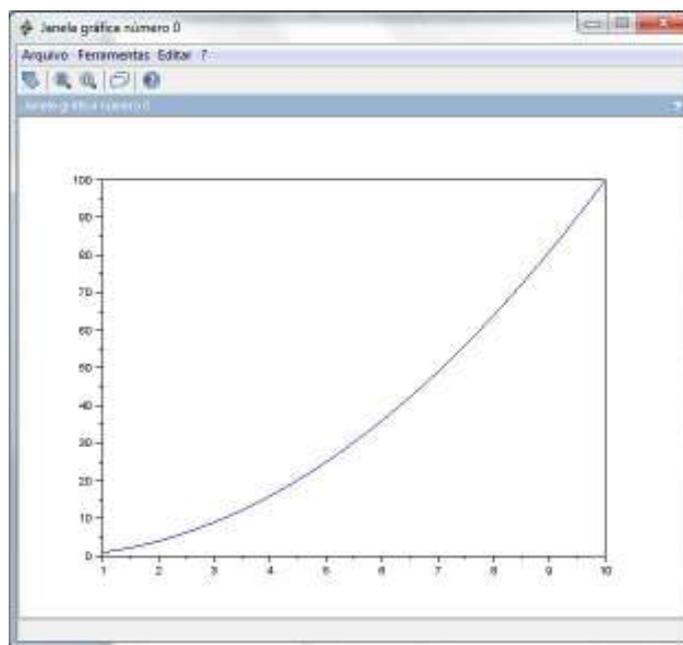
```
-->xx=linspace(1,10,50);
```

```
-->yy=xx^2;
```

```
-->plot(xx,yy)
```

(observe o uso de ; ao final dos 2 primeiros comandos para evitar que o conteúdo dos vetores xx e yy seja exibido, “poluindo” a console)

O resultado dessas operações é a apresentação de uma nova janela contendo o gráfico da função.



### ATIVIDADE PRÁTICA

De maneira similar à ATIVIDADE PRÁTICA anterior, construa o gráfico da função  $y=\sin(x)$ , no intervalo  $[-\pi,\pi]$ , usando 100 pontos igualmente espaçados.

## 5.2. ACRESCENTANDO TÍTULOS, EIXOS E LEGENDAS

Vamos, agora, dar uma apresentação mais profissional aos nossos gráficos. Veremos como o Scilab permite definir títulos, nomear os eixos e apresentar mais de uma curva em um mesmo gráfico junto com as respectivas legendas.

No exemplo a seguir, vamos construir o gráfico da função  $y(x)=1/(1+x^2)$ , no intervalo  $[-5,5]$ .

### ATIVIDADE PRÁTICA

Execute os comandos a seguir.

```
-->x = linspace(-5,5,51);
```

```
-->y = 1 ./ (1+x.^2);
```

```
-->plot(x,y,'o-b');
```

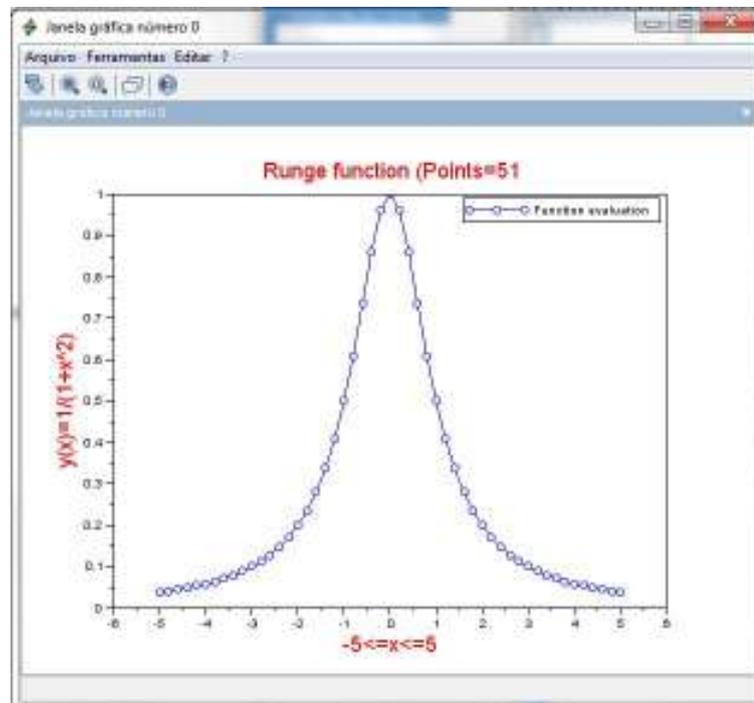
```
-->xlabel("-5<=x<=5", "fontsize",4,"color", "red");
```

```
-->ylabel("y(x)=1 ./ (1+x.^2)", "fontsize",4,"color", "red");
```

```
-->title("Runge function (Points=51)", "color", "red", "fontsize", 4);
```

```
-->legend("Function evaluation");
```

Observe que quando da execução do comando plot, a janela com o gráfico já é exibida mas ainda sem título, legenda, etc. À medida que os outros comandos são executados, a janela gráfica vai sendo atualizada. O resultado final é:



As funções *title*, *xlabel*, *ylabel* e *legend* foram usadas para configurar, respectivamente, o título global do gráfico, os rótulos dos eixos x e y e a legenda no canto superior direito, além de tamanho e cor da letra. Para uma lista completa das opções de configuração de cada função use os comandos “help title”, “help xlabel”, etc.

Além disso, na função *plot* foi usada a opção de configuração ‘o-b’ significando que a linha é contínua (se a opção fosse --, por exemplo, a linha seria tracejada), os pontos de coordenadas (x,y) contidos nos vetores são marcados, sobre a curva, com círculos e a cor da linha é azul (b de blue). Para a lista de opções de estilo de linha, marcadores e cores execute o comando “help LineSpec”.

### 5.3. GRÁFICOS COM VÁRIAS CURVAS

Vamos, agora, exibir os gráficos das funções  $y(x)=1/(1+x^2)$ ,  $\sqrt{y^*(x)}$  e  $y(x)^2$  em uma mesma janela.

#### ATIVIDADE PRÁTICA

Execute os comandos a seguir.

```
-->x = linspace(-5,5,51);
```

```
-->y = 1 ./ (1+x.^2);
```

```
-->plot(x,y,'ro-');
```

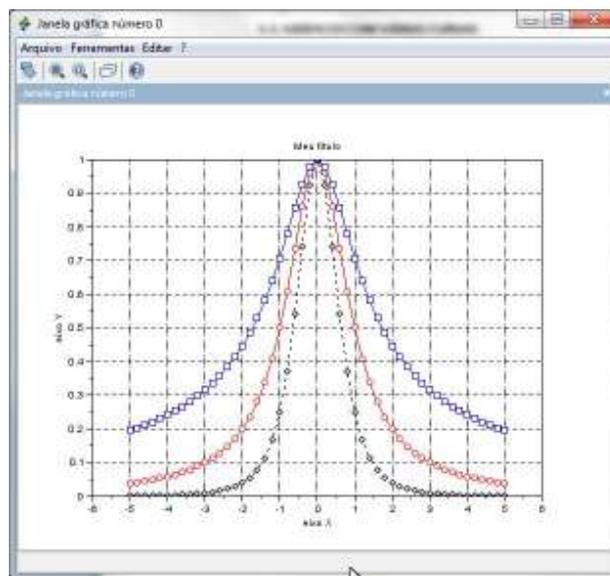
```
-->plot(x,sqrt(y),'bs-');
```

```
-->plot(x,y.^2,'k:d');
```

```
-->xtitle("Meu título", "eixo X", "eixo Y");
```

```
-->xgrid
```

Examine a correspondência entre as opções de configuração dos comandos e o resultado visual:



## 5.4. SUBPLOTS

O comando subplot subdivide uma janela gráfica em várias janelas separadas contendo, cada uma delas, um tipo de gráfico. A sintaxe do comando é subplot(m,n,p), de modo que as sub-janelas serão dispostas em m linhas e n colunas. O número p indica qual subplot está selecionado num dado momento de modo que todos os comandos gráficos são executados nesse subplot até que um novo seja selecionado.

### ATIVIDADE PRÁTICA

Execute os comandos a seguir.

```
-->t = linspace(0,1,101);
-->y1 = exp(%i*t);
-->y2 = exp(%i*t.^2);

-->subplot(2,2,1);
-->plot(t,real(y1),'r');
-->plot(t,real(y2),'b');
-->xtitle("Real part");

-->subplot(2,2,2);
-->plot(t,imag(y1),'r');
-->plot(t,imag(y2),'b');
-->xtitle("Parte imaginária");

-->subplot(2,2,3);
-->plot(t,real(y1)^2,'r');
-->plot(t,real(y2)^2,'b');
// esqueci de definir o título da sub-janela

-->subplot(2,2,4);
-->plot(t,imag(y1)^2,'r');
-->plot(t,imag(y2)^2,'b');
-->xtitle("Parte imaginária ao quadrado");

// havia esquecido de definir o título do subplot 3
// seleciono-o novamente e defino o título
-->subplot(2,2,3);
-->xtitle("Parte real ao quadrado");
```

Procure entender a correspondência entre os comandos e o resultado visual.

