

Introduction to MATLAB

Violeta Ivanova, Ph.D.

Office for Educational Innovation & Technology

violeta@mit.edu

<http://web.mit.edu/violeta/www>

[MATLAB @ MIT]

- Mathematical Tools at MIT
<http://web.mit.edu/ist/topics/math>
- MATLAB on Athena
 - 250 floating licenses (free)
- For student personal computers
 - 300 floating licenses (free)
- For departmental computers
 - Individual & network licenses from IS&T

[This Class]

- Two series, same topics, hands-on training
 - Introduction to MATLAB on Laptops (1-3 pm)
 - Introduction to MATLAB on Athena (4-6 pm)
- Class materials
 - <http://web.mit.edu/acmath/matlab/IAP2007>
 - This session: [InterfaceBasics](#) <.zip, .tar>

[Topics]

- MATLAB Interface and Basics
- Calculus, Linear Algebra, ODEs
- Graphics and Visualization
- Basic Programming
- Programming Practice
- Data Analysis and Statistics

MATLAB Interface

Desktop Interface
Help Browser
Toolboxes

[Starting MATLAB]

- On Athena

```
athena% add matlab
```

```
athena% matlab &
```

```
>> desktop
```

- On laptops

Desktop interface is default.

You must be running MATLAB now ...

[Help in MATLAB]

- Command line help

>> **help** <command>

e.g. help polyint

>> **lookfor** <keyword>

e.g. lookfor polynomial

- Help Browser

- Help->Help MATLAB

[MATLAB Help Browser]

- MATLAB
 - + Getting Started
 - + Desktop Tools and Development Environment
 - + Mathematics
 - + Data Analysis
 - + Programming
 - + Graphics
 - + 3-D Visualization
- Statistics Toolbox
- *Simulink, etc.*

MATLAB Basics

Variables, Vectors & Matrices
Operators & Built-In Functions
Data Input/Output & M-Files

[What is MATLAB?]

- Computational Software

From The MathWorks: www.mathworks.com

- MATrix LABoratory

- Algorithm Development Environment

[Variables]

- Begin with an alphabetic character: `a`
- Case sensitive: `a`, `A`
- No data typing: `a=5; a='ok'; a=1.3`
- Default output variable: `ans`
- Built-in constants: `pi i j Inf`
- `clear` removes variables
- `who` lists variables
- Special characters
`[] () {} ; % : = . ... @`

[Operators]

- Arithmetic operators

`+` `-` `/` `\` `^` `.\` `./` `.*` `.^`

- Relational operators

`<` `>` `<=` `>=` `==` `~=`

- Logical operators

`|` `&` `||` `&&` `true` `false`

- Operator precedence

`()` `{}` `[]` -> Arithmetic -> Relational -> Logical

- Do not use special characters, operators, or keywords in variable names.

[Vectors]

■ Row vector

```
>> R1 = [1 6 3 8 5]
```

```
>> R2 = [1 : 5]
```

```
>> R3 = [-pi : pi/3 : pi]
```

■ Column vector

```
>> C1 = [1; 2; 3; 4; 5]
```

```
>> C2 = R2'
```

[Matrices]

- Creating a matrix

```
>> A = [1 2.5 5 0; 1 1.3 pi 4]
```

```
>> A = [R1; R2]
```

- Accessing elements

```
>> A(1, 1)
```

```
>> A(1:2, 2:4)
```

```
>> A(:, 2)
```

[Matrix Operations]

■ Operators $+$ and $-$

```
>> X = [x1 x2 x3];
```

```
>> Y = [y1 y2 y3];
```

```
>> A = X + Y
```

A =

$x_1 + y_1$ $x_2 + y_2$ $x_3 + y_3$

■ Operators $*$, $/$, and $^$

```
>> Ainv = A-1 Matrix math is default!
```

[Element-Wise Operations]

■ Operators $\cdot^*$, $\cdot/$, and $\cdot^{\wedge}$

```
>> Z = [z1 z2 z3]'
```

```
>> B = [Z.^2      Z      ones(3,1) ]
```

B =

z_1^2 z_1 1

z_2^2 z_2 1

z_3^2 z_3 1

[Graphics]

- 2D linear plots: `plot`

```
>> plot (X, Y, 'r-')
```

Colors: `b`, `r`, `g`, `y`, `m`, `c`, `k`, `w`

Markers: `o`, `*`, `.`, `+`, `x`, `d`

Line styles: `-`, `--`, `-.`, `:`

- Multiple datasets on a plot

```
>> p1 = plot(xcurve, ycurve)
```

```
>> hold on
```

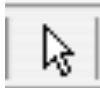
```
>> p2 = plot(Xpoints, Ypoints, 'ro')
```

```
>> hold off
```

[Customizing Graphs]

- Annotating graphs

```
>> plot (X, Y, 'ro')  
>> legend ('Points')  
>> title ('Coordinates')  
>> xlabel ('X')  
>> ylabel ('Y')
```

- Plot Edit mode: icon  in Figure Editor
- Property Editor: View->Property Editor
- Saving figures: File->Save As

[Polynomials]

- Evaluating polynomials

$$y = p_1 x^n + p_2 x^{n-1} \dots + p_n x + p_{n+1}$$

```
>> p = [p1 p2 ... ]
```

```
>> t = [-3 : 0.1 : 3]
```

```
>> z = polyval(p, t)
```

- Curve fitting

```
>> X = [x1 x2 ... xn]; Y = [y1 y2 ... yn]
```

```
>> Pm = polyfit(X, Y, m)
```

[Other Built-In Functions]

■ Matrices & vectors

```
>> [n, m] = size (A)
>> n      = length (X)
>> M1 = ones (n, m)
>> M0 = zeros (n, m)
>> [evecs, evals] = eig (A)
```

■ Descriptive statistics

```
>> m      = mean (X)
>> std    = std (X)
```

[More Built-In Functions]

■ 3D graphics

```
>> [x,y] = meshgrid([-10 : 1 : 10], ...  
                    [-10 : 1 : 10]);  
  
>> z = x .* exp(-(x.^2 + y.^2));  
  
>> s1 = surf(x, y, z)
```

■ And many others ...

```
>> z = sin(t) + cos(t)  
>> Y = fft(X)  
>> R = corrcoef(A)
```

[M-Files: MATLAB Scripts]

- Create or open M-file in **MATLAB Editor**

```
>> edit filename.m
```

- Use **%** for comments
- Use **;** to suppress output at runtime
- Save as *filename.m*
- Run: filename only (no .m extension)

```
>> filename
```

[Data Input / Output]

- **Import Wizard** for data import

File->Import Data ...

- File input with `load`

B = **load**('datain.txt')

- File output with `save`

save('dataout', 'A', '-ascii')

[Interface and Basics Exercises]

- Exercise One: `example1.m`
 - Data input and output
 - Matrices and vectors
 - Operators and built-in functions
- Exercise Two: `example2.m`
 - Polynomials
 - Curve fitting
 - Basic plotting

Follow instructions in M-files ...