

Computational Methods

CMSC/AMSC 460

Ramani Duraiswami,
Dept. of Computer Science

Syllabus

- **Introduction, Computer Arithmetic and Errors** (Chapter 1) (approx. 4 lectures)
 - course survey
 - introduction to Matlab
 - machine arithmetic and error analysis
 - stability and conditioning
- **Solving Linear Systems of Equations** (Chapter 2) (approx. 6 lectures)
 - Gaussian elimination
 - well-conditioning vs. ill-conditioning, matrix and vector norms
 - Notions of algorithm complexity
 - sparse systems: direct and iterative methods

Syllabus

- **Interpolation** (Chapters 3) (approx. 6 lectures)
 - polynomial interpolation
 - Other basis functions and polynomials
 - piecewise polynomial interpolation
 - spline interpolation
- **Zeros and Roots** (Chapter 4) (approx. 4 lectures)
 - Linear and Nonlinear systems of equations
 - Bisection, Secant and Newton method
 - Introduction to optimization
- **Solving Linear Least Squares Problems** (Chapter 5) (approx. 4 lectures)
 - data-fitting and least squares
 - QR factorization

Syllabus

- **Integration/Quadrature** (Chapter 6) (6 lectures)
 - elementary integration formulas (midpoint, trapezoid, etc.)
 - compound and adaptive integration formulas
 - Gaussian quadrature
- **Fourier Analysis** (Chapter 8) (3 lectures)
- **Ordinary Differential Equations** (Chapter 9) (approx. 6 lectures)
 - ordinary differential equations and Euler's method
 - adaptive methods for ordinary differential equations
 - methods for stiff systems

MATLAB Overview

- History of MATLAB
- Strengths of MATLAB
- Weaknesses of MATLAB

What is MATLAB?

- MATLAB
 - MATrix LABoratory
 - Interactive system
 - Programming language
 - Extendable

What is MATLAB?: 2

- Considering MATLAB at home
 - Standard edition
 - Available for roughly 2 thousand dollars
 - Student edition
 - Available for roughly 1 hundred dollars.
 - Some limitations
 - Shorter license period
- On campus
 - Site license

History of MATLAB

- Ancestral software to MATLAB
 - Fortran subroutines for solving linear (LINPACK) and eigenvalue (EISPACK) problems

History of MATLAB, con't: 2

- One of the developers of these packages, Cleve Moler wanted his students to be able to use LINPACK and EISPACK without requiring knowledge of Fortran
- MATLAB developed as an interactive system to access LINPACK and EISPACK

History of MATLAB, con't: 3

- MATLAB gained popularity primarily through word of mouth because it was not officially distributed
- In the 1980's, MATLAB was rewritten in C with more functionality (such as plotting routines)
- Commercialized by a company (The Mathworks)
- In many fields it is the software for quantitative analysis
 - Finance, biology, defence, image processing, audio, etc.
- Some competing packages
 - Octave (an open source alternative)
 - Mathematica, IDL, ...

Strengths of MATLAB

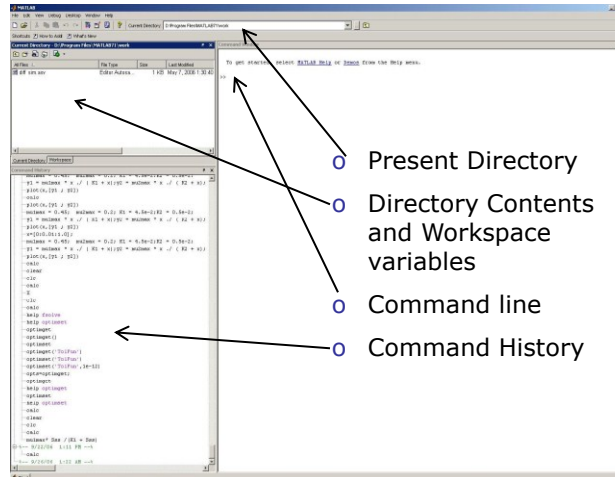
- MATLAB is relatively easy to learn
- MATLAB code is optimized to be relatively quick when performing matrix operations
- MATLAB may behave like a calculator or as a programming language
- MATLAB is interpreted, errors are easier to fix
- Although primarily procedural, MATLAB does have some object-oriented elements

Weaknesses of MATLAB

- MATLAB is NOT a general purpose programming language
- MATLAB is usually used as an interpreted language (making it for the most part slower than a compiled language such as C++)
- MATLAB is designed for scientific computation and is less suitable for some things (such as parsing text)
- Many of these weakness have been addressed somewhat

Matlab Windows

- o Command line Interface (Main Window)
- o Editor Window



Matrices in Matlab

- o Entering a Matrix:

```
>> A = [ 0 -0.8 -0.6 ; 0.8 -0.36 0.48 ; 0.6 0.48 -0.64 ]
A =
```

```
    0 -0.8000 -0.6000
    0.8000 -0.3600  0.4800
    0.6000  0.4800 -0.6400
```

- o Matrix referencing:

```
>> A(1,2)
ans =
```

```
-0.8000
```

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

```
    0.8000 -0.3600  0.4800
```

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

```
ans =
```

```
    0
    0.8000
    0.6000
```

- o Matrix Operations:

```
>> A+A;
>> A.*A;
>> 3*A;
>> A*A
```

```
ans =
```

```
-1.0000    0    0
    0 -0.2800 -0.9600
    0 -0.9600  0.2800
```

Built-in functions

o Determinant

```
>> det(A)
```

```
ans =
```

```
-1.000
```

o Rank

```
>> rank(A)
```

```
ans =
```

```
3
```

o Inverting a Matrix

```
>> inv(A)
```

```
ans =
```

```
-0.0000  0.8000  0.6000
-0.8000 -0.3600  0.4800
-0.6000  0.4800 -0.6400
```

o Transpose of a Matrix

```
>> A'
```

```
ans =
```

```
0  0.8000  0.6000
-0.8000 -0.3600  0.4800
-0.6000  0.4800 -0.6400
```

Solving Linear System

o Linear system of algebraic equations:

$$-x_1 + x_2 + 2x_3 = 2$$

$$3x_1 - x_2 + x_3 = 6$$

$$-x_1 + 3x_2 + 4x_3 = 4$$

$$Ax = b$$

```
>> A = [-1 1 2; 3 -1 1; -1 3 4]
```

```
A =
```

```
-1  1  2
 3 -1  1
-1  3  4
```

```
>> b = [2 6 4]'
```

```
b =
```

```
2
6
4
```

```
>> rank(A)
```

```
ans =
```

```
3
```

```
>> x = A\b
```

```
(also could do inv(A)*b, but not recommended)
```

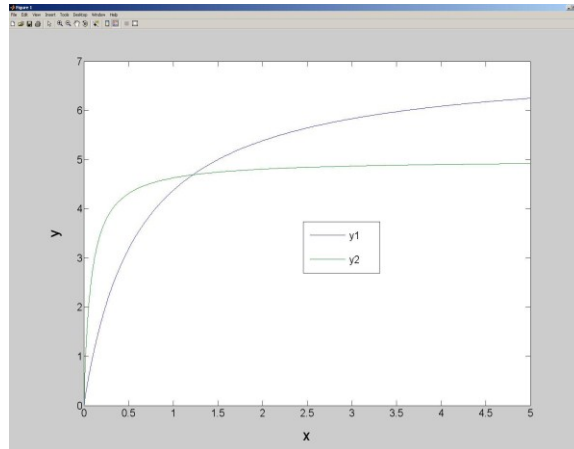
```
x =
```

```
1.0000
-1.0000
2.0000
```

Plotting a function

$$y_1 = \frac{7x}{0.6 + x} \quad y_2 = \frac{5x}{0.08 + x}$$

```
>> x = [ 0:0.01:5];
>> y1 = 7 * x ./ ( 0.6 + x );
>> y2 = 5 * x ./ ( 0.08 + x );
>> plot(x,y1,x,y2)
>> legend('y1','y2')
```



Introduction to MATLAB

- Vectors, Matrices, Syntax
- Vector operations, including the \dot commands
 - length, size, linspace, logspace, size, rand, randn, randperm
- Special vectors and matrices: zeros, ones, eye, magic
- Scripts and functions
 - Diary
- Graphing:
 - plot, special fonts, plot3, semilogx, semilogy, title, xlabel, ylabel, axis, grid, legend, subplot,
- Formatted output:
 - Sprintf, ;, disp, input
- Programming:
 - for, if, while, &, |, ~
- General/misc commands
 - ginput set, size, max, sum, close, figure, hist, any, all, floor, fix, round,
- Graphical programming and callbacks