Introduction to Matlab

CS 471/Math 471

Brian T. Smith, UNM

Andrew C. Pineda, UNM

9/18/1997 matlab 1

What Is Matlab?

- Interactive system for scientific computations
  - matrix computations
  - ODE solvers, approximation, curve fitting, data analysis
  - plotting, image display
- Interpretation language for trying out algorithms and investigating computations
- Extensible language by adding libraries/toolboxes

9/18/1997 matlab 2

Getting Started

- At CIRT, matlab runs on pegasus only?
- Need to set the remote display when the initial login machine is *not* pegasus
  - set the display on pegasus

    setenv DISPLAY <xterm_name>:0.0
  - set permissions on your initial login machine for the display

    xhost +
- Startup matlab by typing: matlab

9/18/1997 matlab 3

Help From Matlab

- Type:

  help

  or

  help <topic>
- Demos
  - Try the demos to see what the capability of matlab is

9/18/1997 matlab 4

Simple Matrix Operations

- To establish a value for a matrix, type:

  a = [1 2; 3 5]
  - creates a 2x2 matrix by rows (that is, a21 is 3)
- Operations on a matrix are linear algebra operations

  c = a + b
  - adds to matrices element by element

  c = a*b
  - performs matrix multiply (linear algebra)

9/18/1997 matlab 5

Other Matrix Operations

- Transpose (uses the quote operator)

  F = A’
  - transpose of A assigned to F
- Solving linear systems (uses the \ operator)

  Y = [1; -1]

  X = A \ Y
  - solves the linear system Ax = y
    - uses the least squares solution when more rows than columns

9/18/1997 matlab 6

Plotting

- Given a set of (x,y) coordinates, it is easy to create a line plot of those data points

```
u = 0:pi/20:pi
w = sin(u)
plot(u,w)
```

  - Plots the sine function using 41 pairs of points and draws a line between them
  - close command removes the plot window
  - Labeling is done by commands
    - » title, xlabel, ylabel

Basic Rules

- There is only one type
  - a rectangular array of complex values
- A convenient model is to assume 3 types
  - a scalar -- actually a 1x1 array
  - a vector -- actually a 1xc array (1 row, c columns)
  - an array -- actually a rxc array (r rows, c columns)
- Variables (names in general)
  - begins with a letter, but uses letters, digits, underscore

Variables Continued

- The first 19 are signification
- Upper/lower case letters are distinguished
- Constants -- arrays in general
  - real
    - 1.1, 1.234e10, -1.22e-3
  - complex
    - 1 + 2i, -7.0+6.123j

Display Formats and Printing

- Typing the name of a variable followed by a carriage return prints the variable
  - actually assignment prints the value assigned
- Suppress print by ending the line with ;
- Printing format using format command
  - option short prints 6 digits rounded
  - option long prints 14 digits rounded
  - option e prints the numbers with an exponent

```
format short e    % prints 6 digit values with e format
help format    % more formats such as hexadecimal
```


Vector Values

- A list of values with no ; enclosed in []
- A triplet forms a sequence of values

  <initial_value> : <step> : <maximum_value>

  - creates a sequence of values in a vector that begins with <initial_value> up to but not exceeding <maximum_value> in steps of <step>
- The sequence of values are treated as a row

```
[ 1 2 3 4]
[1:1:4]
```


Matrix Values

- A vector with a semicolon to separate rows of the matrix

```
[1; 2; 3; 4]    % A column vector
[1 2 3; 4 5 6; 7 8 9; 10 11 12]   % 4x3 matrix
[1:3; 4:6; 7 8 9; 10:12]   % The same 4x3 matrix
```

- A carriage return also separates rows

```
[1 2 3
 4 5 6
 7:9; 10 11 12]
```

Operators

- Operators are: +, -, *, /, \, ^
  - all are interpreted as linear algebra operations
  - ^ is matrix exponentiation
    - » a^3 multiplies a by itself 3 times (lin. Alg. Mult.)
- Place a dot in front of the operator makes it an element by element operation (./ and .\ are the same)
    - » a.*b is Fortran's multiply operation

Expressions

- Combinations of these operators and constant or variable operations
  - shape must be compatible
    - linear algebra rules -- inner dimensions the same -- for non-dot operators
    - shape conformance for dot operators
- Elements and sections
    - A(i,j) is the (i,j)-th element
    - A(:,2) is the second column of A -- uses f90 notation

Relational and Logical Operators

- Relational operators available
  <, <=, >, >=, ==, ~=
- Logical operators to combine relational expressions
  &, |, ~

Built-in Functions

- There are many of these for many kinds applications. The ones used in the tutorial are:
  - » plot % plots x-y points
  - » ones % generates an array of ones
  - » eye % generates the identity matrix
  - » polyfit % performs polynomial curve fitting
  - » eig % computes matrix eigenvalues and eigenvectors

Matlab Scripts and User-defined Functions

- A matlab script is a file whose name is of the form <name>.m (where <name> is a matlab identifier)
  - contains matlab commands that are to be executed
- A matlab function is a script with a special first line that specifies the name, the arguments of the function, and the value to be returned

Function Header Line

- Header line has the format:
  function <var_name> = <name>(<list_of_arguments>)
  - <var_name> is the name of the variable assigned a value that is to be the returned value
  - <name> is the name of the function

```
function y = mycubic(x1,x2)
x = max(x1,x2)
y = ((x + 2)*x)*x + 1
```

  - contents of file mycubic.m
  - referenced as follows: z1 = 1; z2 = 3; z = mycubic(z1,z2)

Returning Array Values

- The following example returns array values
  - that is, a vector of length 2 with sine and cosine of an angle

```
function [c,s] = trigfunction( theta )
c = cos( theta )
s = sin(theta)
```

  - reference

```
[ct,st] = trigfunction( theta )
```

9/18/1997 matlab 19

Control Statement

- Matlab has three control constructs:
  - for, while, if
- for is an iterated loop statement

```
for i = 1:n
    …    % loop body statements
end
```

  - or

```
for i = m:s:n   % from m to n in steps of s
  …
end
```

9/18/1997 matlab 20

Control Constructs Cont'd

- while construct

```
while  <condition>
    ….
end
```

- if construct

```
if  <condition>
   …
else
   …
end
```

9/18/1997 matlab 21

Input/Output and Session Log

- There are functions for input/output much like C functions with format specifications
  - fread, fwrite, fscanf, fprintf, fopen, fclose
  - use help <name> to discover details of these
- Session log

```
diary  <logfile>   % Saves session in <logfile>
diary  off         % Turns off logging of session
```

9/18/1997 matlab 22

Unix Commands in Matlab

- Begin the line with !

```
!ls        % List the names in the current directory
!f90 -O -o myprog myprog.f90    % Compiles myprog
!myprog > myoutput   % Runs a Unix program
fread …        % Now executing a matlab command
```

9/18/1997 matlab 23