Introduction to MATLAB

Arturo Donate
Introduction

- What is MATLAB?
- Environment
- MATLAB Basics
- Programming
- Toolboxes
- Comparison
- Conclusion
What is MATLAB?

- Matrix laboratory
- Programming environment
- High-performance language
- Windows, OSX, Linux/UNIX
- Matrix/Vector computations
- Linear equations, eigenvectors, etc...
- LINPACK, EISPACK, LAPACK, BLAS, etc...
What is MATLAB?

Uses:
- math & computation
- algorithm development
- modeling and simulation
- data analysis and visualization
- application development
What is MATLAB?

- Toolboxes
  - image processing
    - filtering, transforms, analysis, enhancement
  - statistics
    - linear models, probability dist., HMMs
  - optimization
    - max/minimization, least squares, line fitting
What is MATLAB?

- Toolboxes
  - fixed-point
    - fixed-point data type & arithmetic
  - others:
    - symbolic math
    - signal processing
    - virtual reality
function [s, t] = eigencluster(A_orig, level)
    if exist('level')==0
        level=1;
    end
    A = A_orig;
    [num_vect] = size(A,1);
    for i=1:num_vect
        A(i,:) = A(i,:)/sum(A(i,:));
    end
    AAt = A*A';
    p = zeros(num_vect,1);
    R = zeros(num_vect);
    D = zeros(num_vect);
    %step 1:
    for i=1:num_vect
        p(i) = sum(AAt(i,:));
    end
    %step 2:
    pi = (1/sum(p(:))) * p;
    for i=1:num_vect
        R(i,i) = p(i);
        D(i,i) = sqrt(pi(i));
    end
    %step 3:
    s = R(:,1)*W(:,1); %W (P^D)
Basic Console Commands

- help
- ls
- who, whos
- clear, pack
- load, save
- eval
- disp
- what, type
- lookfor, which
- exit, quit
Basic Math Commands

Operations
- mean, median, mode
- sum, abs
- sin, sinh, asin, etc...
- sqrt, log, exp
- floor, ceil, round
- hist, plot

Constants
- $\pi = 3.14159...$
- $i, j = \sqrt{-1}$
- realmin, realmax
- Inf, NaN
Vectors and Matrices

Creating

- \( M = \text{ones}(10,10) \)
- \( M = \begin{bmatrix} 1 & 2 & 3; 4 & 5 & 6; 7 & 8 & 9 \end{bmatrix} \)
- \( M(1,1) = 23 \)
- \( M = \begin{bmatrix} \text{eye}(6) & \text{rand}(6) \end{bmatrix} \)

Accessing

- \( M(x, y) \)
- \( M(x, :) \), \( M(x, i:j) \)

Deleting

- \( M = [] \)
Vectors and Matrices

- Operations
  - + - * / ^ ' 
  - any, all
  - diag
  - rank

- eig
- svd
- trace
- prod
Vectors and Matrices

Sample expressions:

- $A = B \times C$
- $\text{num} = \text{abs}(3 + 4i)$
- $\text{vector} = \sin(1:50)$
- $B = [A' \ C']$
- $B(:,2) = []$
- $\text{determ} = \text{det}(C)$
- $\text{index} = \text{find}(C > 0)$
- $x = \neg \text{isprime}(y)$
- $\text{num} = \text{numel}(x)$
- $[x \ y] = \text{size}(C)$
Vectors and Matrices

Colon notation

- $1 : 50$
- $1 : 2 : 50$
- $\text{A}(:, 2)$
- $\text{A}(1:3, 2)$
- $\text{A}(2:4, :)$
- $\sin(0 : \pi/20 : 2\pi)$
Graphics

2D Example: plot( sin( 0 : pi/20 : 2*pi ) )
Graphics

- 3D example - plot 3D point cloud
Graphics
Programming

Functions

- function \([X \ Y] = \text{func\_name}(\text{arg1, arg2})\)
- func_name.m
- primary vs sub-functions
- nested function declaration/definition

- private functions
- function overloading (int vs double)
- global variables
- pre-allocation
- vectorization
Programming

- script vs function
- if, elseif, else
- for, while
- switch, case
- continue, break
- return
Programming

Relations

\( \sim a \)

\( a == b \)

\( a \sim= b \)

\( a <= b \)

\( a \& b \)

\( a \mid b \)

\( a \mid \sim a \)
Programming

- Data Structures
  - Matrix
    - basic data type
  - Cell
    - multi-dimensional matrices

- String
  - character array

- Structure
  - type with various fields
Simple Example: SVD

Create a function that:
- takes a matrix as input
- make sure the matrix is square
- computes the SVD decomposition
- returns the first singular value

“answer = my_func(matrix)”
Simple Example: Sort

Create a function that:
- takes a list as input
- sorts the list
- returns a list with sorted values
- “answer = my_func(matrix)”
Image Processing

- Toolbox used for image processing, computer vision, and signal processing tasks
  - image transformation
  - registration
  - filtering
  - image analysis/enhancement/de-blurring
  - segmentation
  - etc...
Image Processing

- Noise reduction example
- salt & pepper noise
- mean, median filter
- \texttt{filter2( fspecial('average', 3), image) / 255}
- \texttt{medfilt2(image, [3 3])}
Color segmentation example

cform = makecform('srgb2lab')
lab_img = applycform(image, cform)
ab = double(lab_img(:,:,2:3))
nrows = size(ab, 1)
ncols = size(ab, 2)
ab = reshape(ab, nrows*ncols, 2)
Image Processing

\[
[\text{cluster}_\text{idx}, \text{cluster}_\text{center}] = \text{kmeans}(\text{ab}, 3, \text{'distance'}, \text{'sQEuc\text{lidea}\text{n}'}, \text{'Replicates'}, 3);
\]

\[
\text{label} = \text{reshape}(\text{cluster}_\text{idx}, \text{nrows}, \text{ncols}, 1);
\]
Image Processing

Texture segmentation example

\[ E = \text{entropyfilt}(I); \]
\[ Eim = \text{mat2gray}(E); \]
\[ BW1 = \text{im2bw}(Eim, 0.8); \]
\[ BWao = \text{bwareaopen}(BW1, 2000); \]
\[ nhood = \text{true}(9); \]
Image Processing

- `closeBWao = imclose(BWao, nhood);`
- `roughmask = imfill(closeBWao, 'holes');`
- `I2 = I; I3 = I;`
- `I2(roughmask) = 0;`
- `I3(~roughmask) = 0;`
MATLAB Clones

- Packages: Scilab, Octave, Rlab

- Similar:
  - matrix is basic data type
  - complex number support
  - built-in mathematical functions
  - powerful library
  - user-defined functions
MATLAB Clones

- Scilab
  - www.scilab.org
  - best support & docs
  - good compatibility
- Octave
  - www.gnu.org/
    - software/octave

- Unix-like
- Rlab
  - rlab.sourceforge.net
  - attempts to improve syntax/semantics
  - least compatible
Conclusions

- interactive programming environment
- high performance language
  - algorithm design
  - modeling & simulation
  - analysis & visualization
- linear algebra
- toolboxes
References

MATLAB:
www.mathworks.com

MATLAB Tutorial:
www.math.ufl.edu/help/matlab-tutorial/

MATLAB Comparison:
www.dspguru.com/sw/opendsp/mathclo2.htm