Introduction to MATLAB

Mike (Y-Q.)
yqmiao@uwaterloo.ca

Sept 2014

outline

- what's matlab?
- interface
- basics
- an example
- how to access
- help

what is it?

- MATrix LABoratory, by MathWorks
  - high-level programming language
  - so, it's very simple to code
  - rich in libraries and graphical subroutines
  - transportable across platforms
  - a great tool for rapid prototyping
  - not the best for real-time applications

interface

- command window
- workspace
- current directory
- command history
- array editor
- text editor

basics

- ‘Hello World!’
  - a = 3;
  - b = 4;
  - c = a + b
- end each statement with semicolon, if you do not like to see the result in the command window

basics

- arithmetic operators:
  - addition: A+B
  - subtraction: A-B
  - multiplication: A*B
  - right division: A/B = A*inv(B)
  - left division: A\B = inv(A)*B
  - power: A^b
  - transpose: A’
  - colon operator:
    - to create vectors: a:b
    - array subscripting: A(:,b)

basics

- dot operators (a.k.a element-wise operators)
  - A.*B, A./B, A.\B, and A.^B
- relational operators
  - a<b, a<=b, a>b, a>=b, a==b, and a~=b
- logical operations
  - a||b (or), a&&b (and), ~ a(not)
- element-wise logical operators
  - A|B, A&B, ~A

basics

- operator precedence
  - Parentheses
  - transpose and power
  - unary plus, unary minus, and logical negation
  - multiplication(s) and division(s)
  - addition and subtract
  - …

basics

- flow control
  - conditional control
    - if, else, and elseif
    - switch and case
  - loop control
    - for
    - while
    - break
    - continue

basics

- if

```
if expression1
   statements1
elseif expression2
   statements2
else
   statements3
end
```

basics

- for

```
for index = values
   program statements
end
```

- while

```
while expression
   program statements
end
```

basics

- function definition
  - `function [output_variables] = fcn_name(input_variables)`
  - the name of a function should be consistent with the file name

an example

- Given the corresponding coefficients of two lines (ax+by+c=0), calculate the intersection point and plot the lines on a figure.
- Function:

```
[intersection,Runtime] = myPlot(line1,line2)
```

- Script to call this function:

```
clear all;clc
load coeffs
line1 = coeff(1,:);
line2 = coeff(2,:);
[intersection,RunTime] = myPlot(line1,line2)
```

how to access?

- have a license
- Nexus computers
  - on campus
  - remotely
- Octave

help

- where to look for answers?
  - Matlab Help
  - Mathworks website
  - Online forums
  - TAs

refs

- www.mathworks.com
- www.gnu.org/software/octave/
- saw.uwaterloo.ca/matlab/