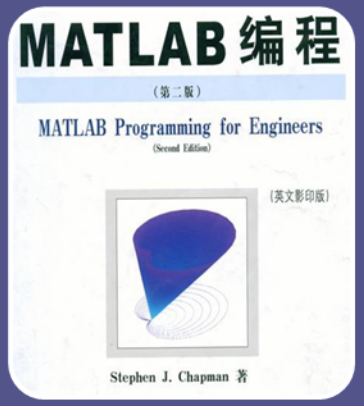
MATLAB Programming
(Lecture 1)

Dr. SUN Bing

School of EIE

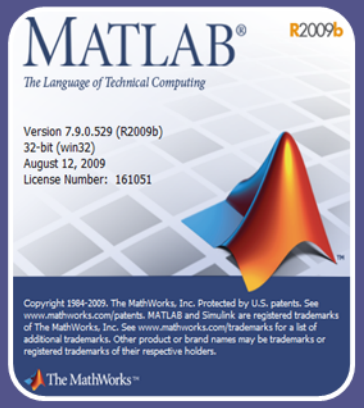
Beihang University

Contents



About this course

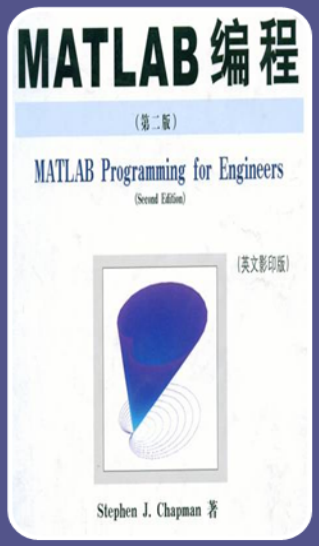
- What's Matlab?
- Why do we learn Matlab?
- How to learn Matlab?
- Course arrangement



About Matlab

- History of MATLAB
- Advantages and disadvantages
- How to start and use Matlab

Contents



About this course

- What's Matlab?
- Why do we learn Matlab?
- How to learn Matlab?
- Course arrangement

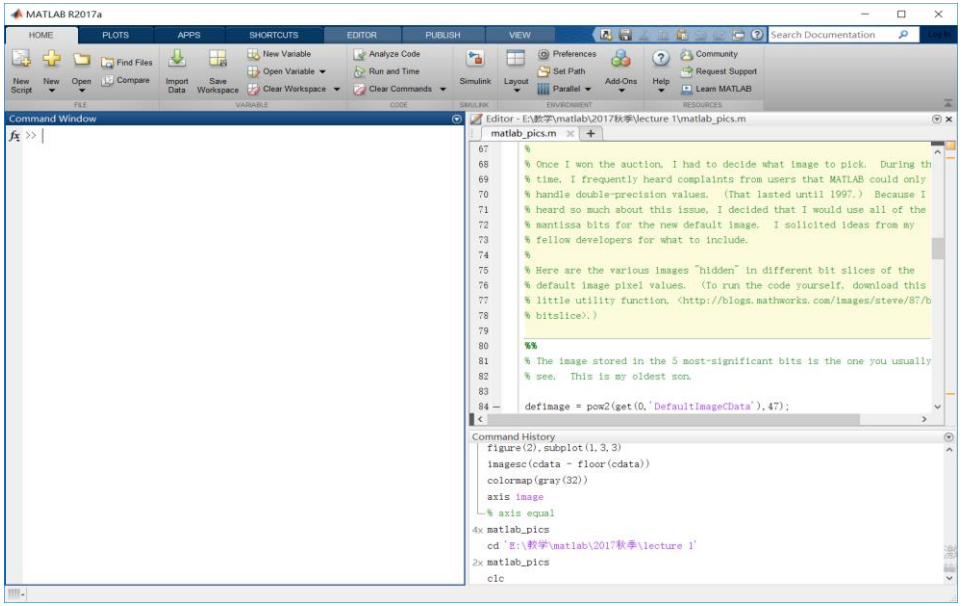
1.1 What's Matlab?

- MATLAB is short for MATrix LABoratory.
- MATLAB is both a powerful computational environment and a programming language for scientific and engineering computations and graphics.
- MATLAB has three parts:
  - *Computational environment and a programming language*
  - *Data visualization*
  - *Symbolic operations*
- Typical uses include:
  - *Math and numeric computation*
  - *Algorithm development*
  - *Scientific and engineering graphics*
  - *Modelling, simulation, and prototyping*

1.1 What is MATLAB?

MATLAB

Developed by	**The MathWorks**
Latest release	**R2019b/ 2019-9-11**
OS	**Cross-platform**
Type	**Technical computing**
License	**Proprietary**
Website	**www.mathworks.com**



M-CODE

Paradigm	**imperative**
Appeared in	**late 1970s**
Designed by	**Cleve Moler**
Developer	**The MathWorks**
OS	**Cross-platform**

```
%%
% The image stored in the 5 most-significant bits
% is the one you usually see.
% This is my oldest son.

defimage = pow2(get(0,'DefaultImageCData'),47);
mag = 200;
figure(2),subplot(3,5,1)
imshow(bitslice(defimage,47,51), 'initialmag', mag);
title('the oldest son');
```

1.1 What is MATLAB?

MATLAB Allows:

- *Easy matrix manipulation*
- *Plotting of functions and data*
- *Implementation of algorithms*
- *Creation of user interfaces*
- *Interfacing with programs in other languages*
- *Access to computer algebra capabilities (Maple engine)*

1.2 Why do we learn Matlab?

- It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

- Now more than 3200 Universities around the world use the MathWorks' products for teaching and research in a broad range of technique.

It's very useful for us to learn Matlab Programming.

1.2 Why do we learn Matlab?

- Now, MATLAB has become a standard tool for many working in science or engineering fields, even in some popular fields such as artificial intelligence.
  - ✓ Numeral Calculations
  - ✓ Engineering and scientific drawing
  - ✓ Control system design and simulation
  - ✓ Digital image processing
  - ✓ User interface design
  - ✓ Artificial intelligence
  - ✓ ……
- See examples.

有用
有趣
好用

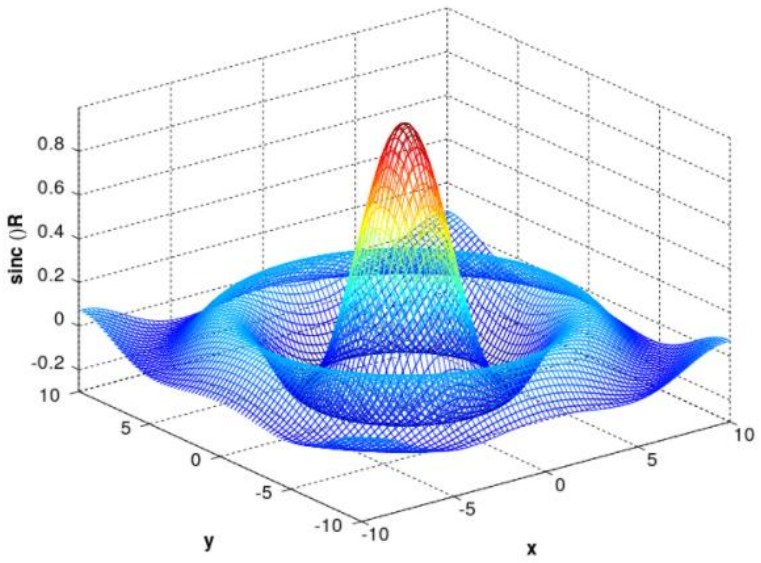
1.2 Why do we learn Matlab?

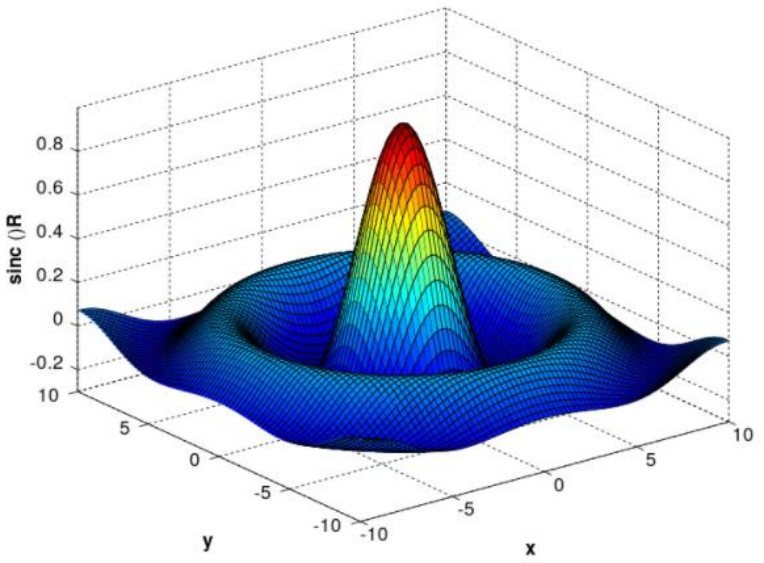
- Numeral Calculations

```
>> A=[1 2 3;4 5 6;4 2 6];
[L,U]=lu(A)
L =
    0.2500   -0.2500    1.0000
    1.0000         0         0
    1.0000    1.0000         0
U =
    4.0000    5.0000    6.0000
         0   -3.0000         0
         0         0    1.5000
```

1.2 Why do we learn Matlab?

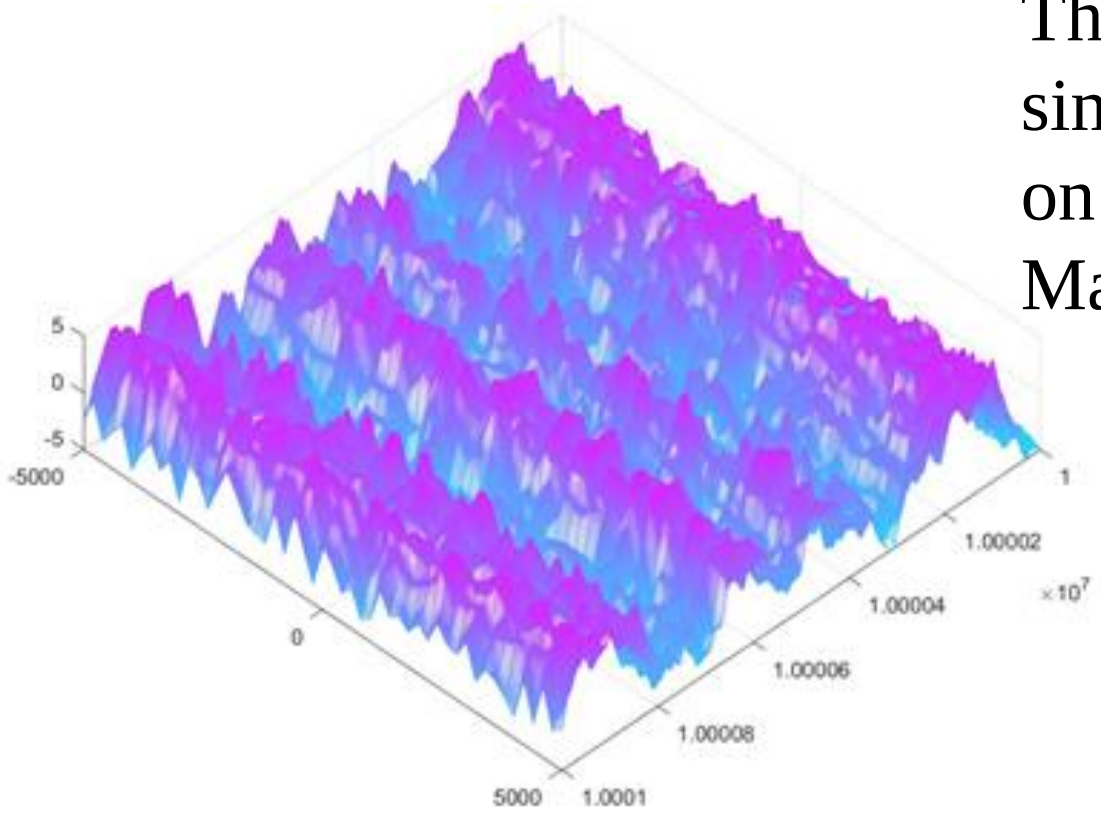
- Engineering and scientific drawing





1.2 Why do we learn Matlab?

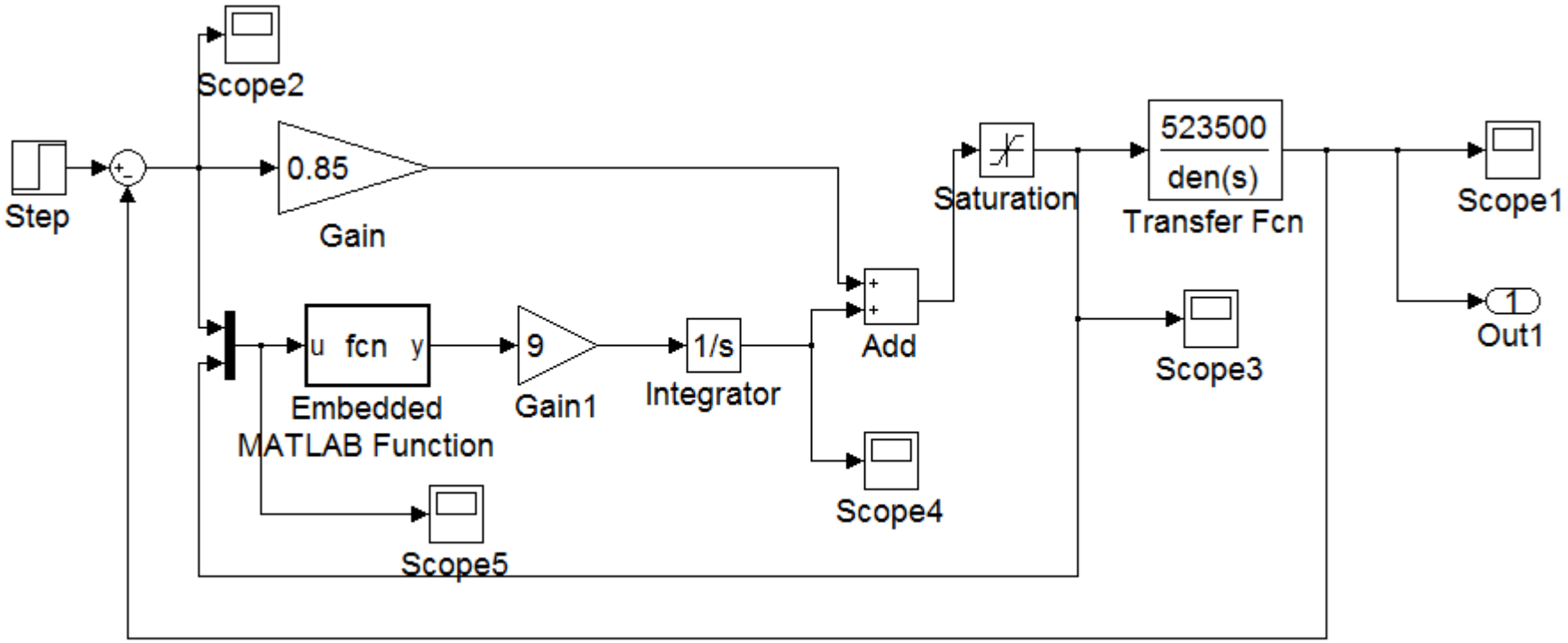
- Engineering and scientific drawing



The picture shows the simulation of a certain time on the dynamic sea on Matlab

1.2 Why do we learn Matlab?

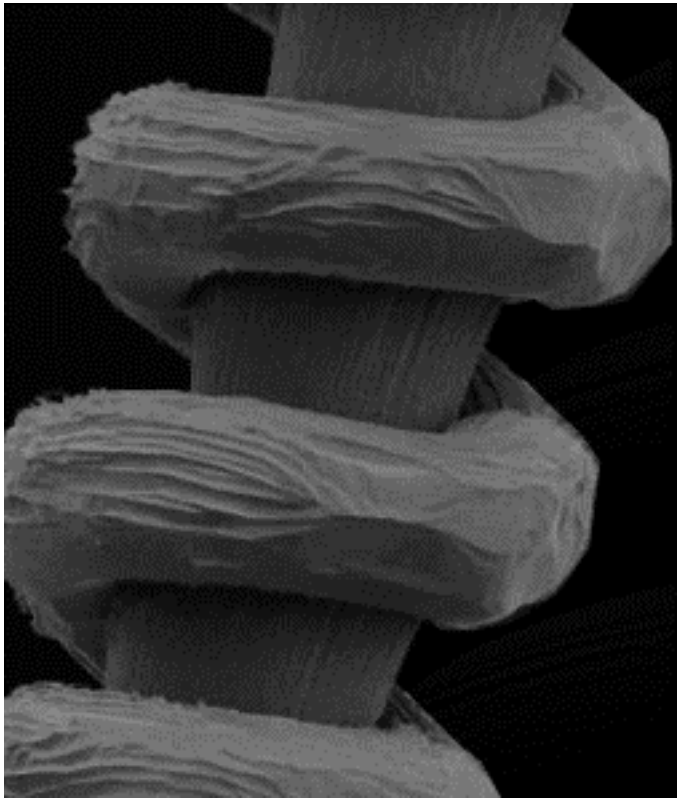
- Control system design and simulation



System function： $G = \dfrac{523500}{s^3 + 87.35s^2 + 10470s}$

1.2 Why do we learn Matlab?

- Digital image processing



Original image

Enhanced image

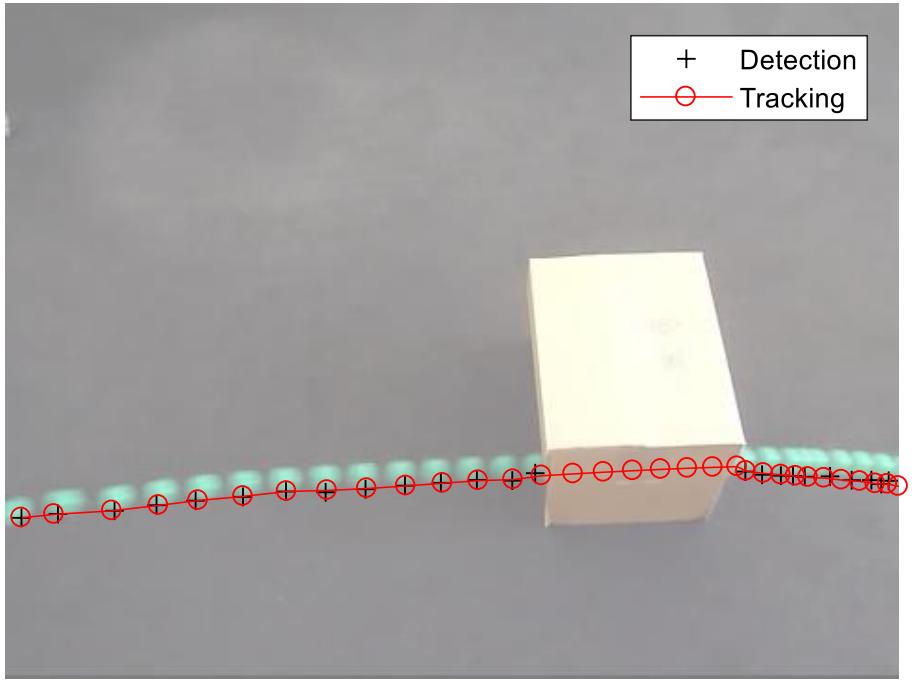
1.2 Why do we learn Matlab?

- Digital image processing

Vehicle license recognition

1.2 Why do we learn Matlab?

- Video signal processing

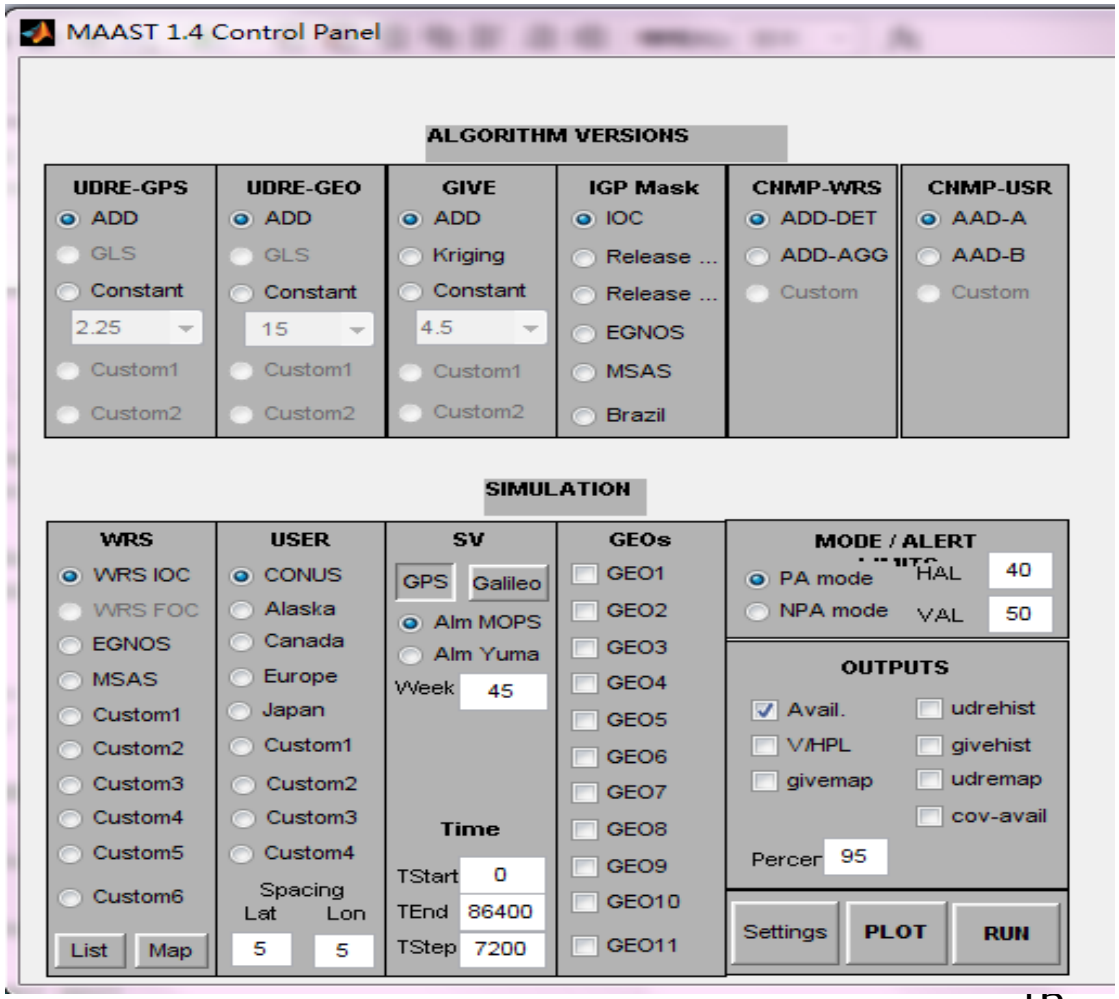


kalmanFilterForTracking

1.2 Why do we learn Matlab?

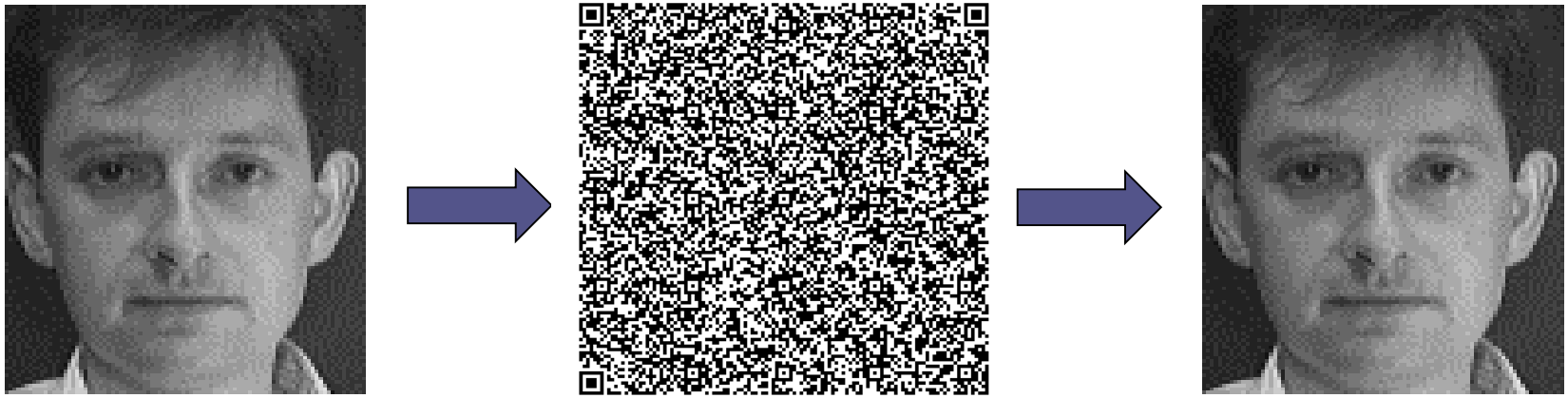
- User interface design

The Wide Area Differential GPS (WADGPS) Laboratory of Stanford University has developed the Matlab Algorithm Availability Simulation Tool, MAAST, to provide a tool for availability simulation of the Wide Area Augmentation System. This software is available for users wishing to simulate the impact on WAAS availability as a result of proposed changes in the system.



1.2 Why do we learn Matlab?

- Artificial intelligence



Face image

face QR code

face recognition

The MATLAB program based on Face QR Code Recognition

1.2 Why do we learn Matlab?

- Artificial intelligence

  The main process：

  - 1. Load the faces image
  - 2. Perform PCA reduce dimension processing
  - 3. Coded dimensionality data
  - 4. Decoded QR code, identify faces

1.2 Why do we learn Matlab?

- Artificial intelligence

Load the face database to form the feature subspace. The face samples involved in this case are all taken from the University of Cambridge face database

Five images of ORL database

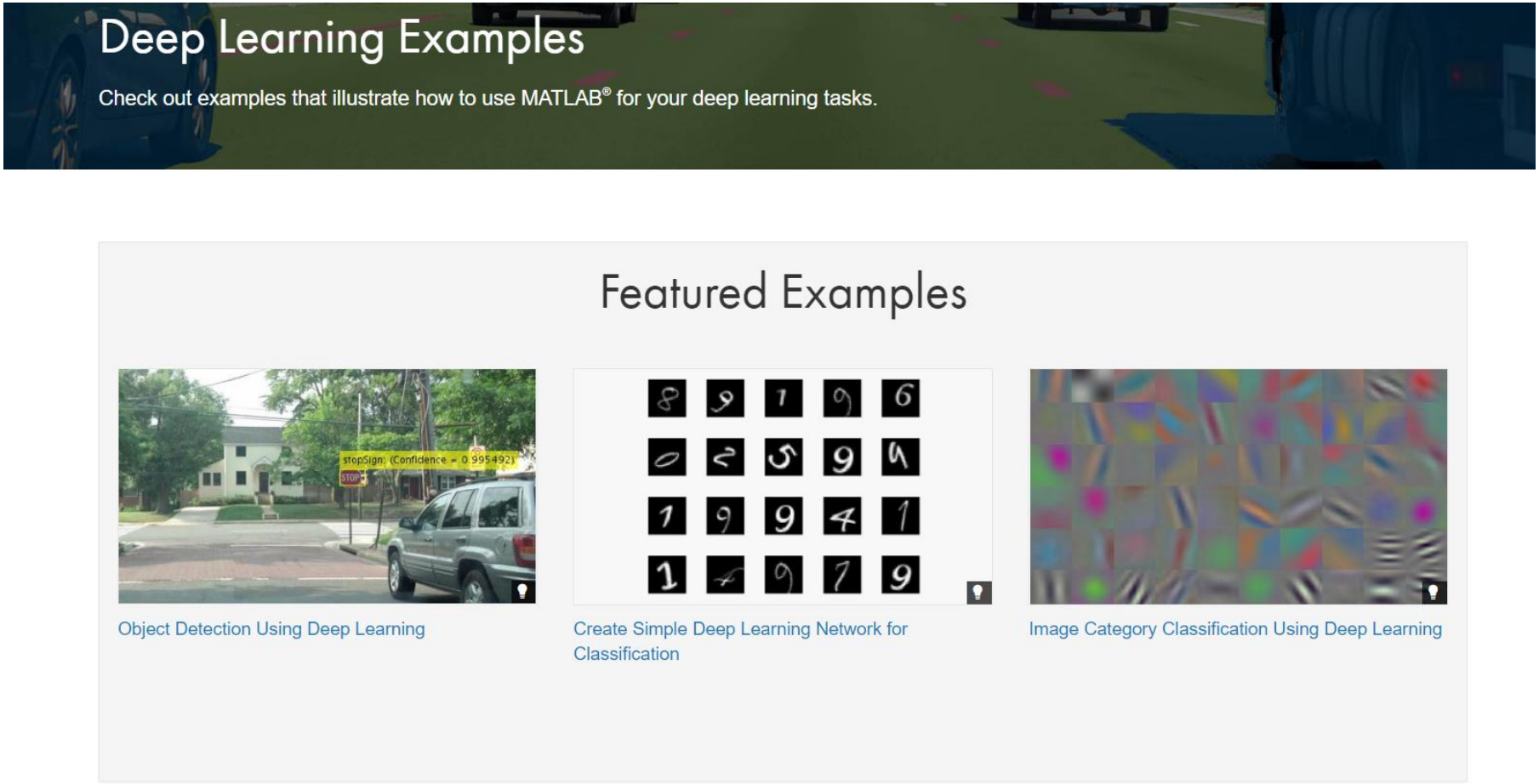
1.2 Why do we learn Matlab?

Construct PCA database：

```
function Construct_PCA_DataBase()
if exist(fullfile(pwd, '人脸库/model.mat'),
'file')
    return;
end
classNum = 40;
sampleNum = 10;
hw = waitbar(0, 'database progress: ', 'Name',
'PCA face identify');
rt = 0.1;
waitbar(rt, hw, sprintf('database progress:
%i%%', round(rt*100)));
allsamples = Get_Samples(classNum, sampleNum);
rt = 0.3;
waitbar(rt, hw, sprintf('database progress:
%i%%', round(rt*100)));
samplemean = mean(allsamples);
xmean = Get_StandSample(allsamples,
samplemean);
rt = 0.5;
waitbar(rt, hw, sprintf('database progress:
%i%%', round(rt*100)));
sigma = xmean*xmean';
[v, d] = eig(sigma);
d1 = diag(d);
rt = 0.7;
waitbar(rt, hw, sprintf('database progress:
%i%%', round(rt*100)));
```

```
dsort = flipud(d1);
vsort = fliplr(v);
p = classNum*sampleNum;
base = xmean' * vsort(:,1:p) *
diag(dsort(1:p).^(-1/2));
rt = 0.9;
waitbar(rt, hw, sprintf('database
progress: %i%%', round(rt*100)));
save(fullfile(pwd, 'face
database/model.mat'), 'base',
'samplemean');
rt = 1;
waitbar(rt, hw, sprintf('database
progress: %i%%', round(rt*100)));
delete(hw);
msgbox('complete the database! ',
'Modal');
```

1.2 Why do we learn Matlab?



https://www.mathworks.com/solutions/deep-learning/examples.html

1.2 Why do we learn Matlab?

Help

Computer Vision System Toolbox Examples

Examples

Search Help

CONTENTS Close

- Examples Home
- Computer Vision System Toolbox
- ON THIS PAGE
- MATLAB Examples
- Simulink Examples
- Documentation
- Functions
- Classes
- Apps
- System Objects
- Blocks
- Release Notes

Computer Vision System Toolbox Examples

MATLAB Examples

Object Detection and Recognition

Automatically Detect and Recognize Text in Natural Images

Detect regions in an image that contain text. This is a common task performed on unstructured scenes. Unstructured scenes are images

Open Script

Detecting Cars Using Gaussian Mixture Models

Detect and count cars in a video sequence using foreground detector based on Gaussian mixture models (GMMs).

Open Script

Object Detection in a Cluttered Scene Using Point Feature Matching

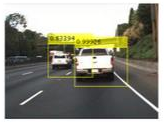
Detect a particular object in a cluttered scene, given a reference image of the object.

Open Script

Object Detection Using Deep Learning

Train an object detector using deep learning and R-CNN (Regions with Convolutional Neural Networks).

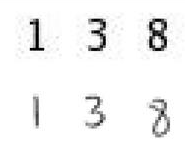
Open Script



Object Detection Using Faster R-CNN Deep Learning

Train an object detector using a deep learning technique named Faster R-CNN (Regions with Convolutional Neural Networks).

Open Script



Digit Classification Using HOG Features

Classify digits using HOG features and a multiclass SVM classifier.

Open Script

Image Category Classification Using Bag of Features

Use a bag of features approach for image category classification. This technique is also often referred to as bag of words. Visual image

Open Script

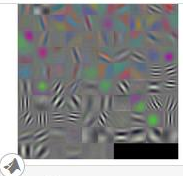


Image Category Classification Using Deep Learning

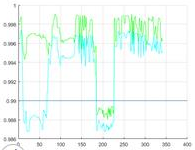
Use a pre-trained Convolutional Neural Network (CNN) as a feature extractor for training an image category classifier.

Open Script

Image Retrieval Using Customized Bag of Features

Create a Content Based Image Retrieval (CBIR) system using a customized bag-of-features workflow.

Open Script



Pattern Matching

Use the 2-D normalized cross-correlation for pattern matching and target tracking. The example uses predefined or user specified target

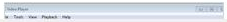
Open Script

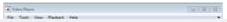
Recognize Text Using Optical Character Recognition (OCR)

Use the ocr function from the Computer Vision System Toolbox™ to perform Optical Character Recognition.

Open Script

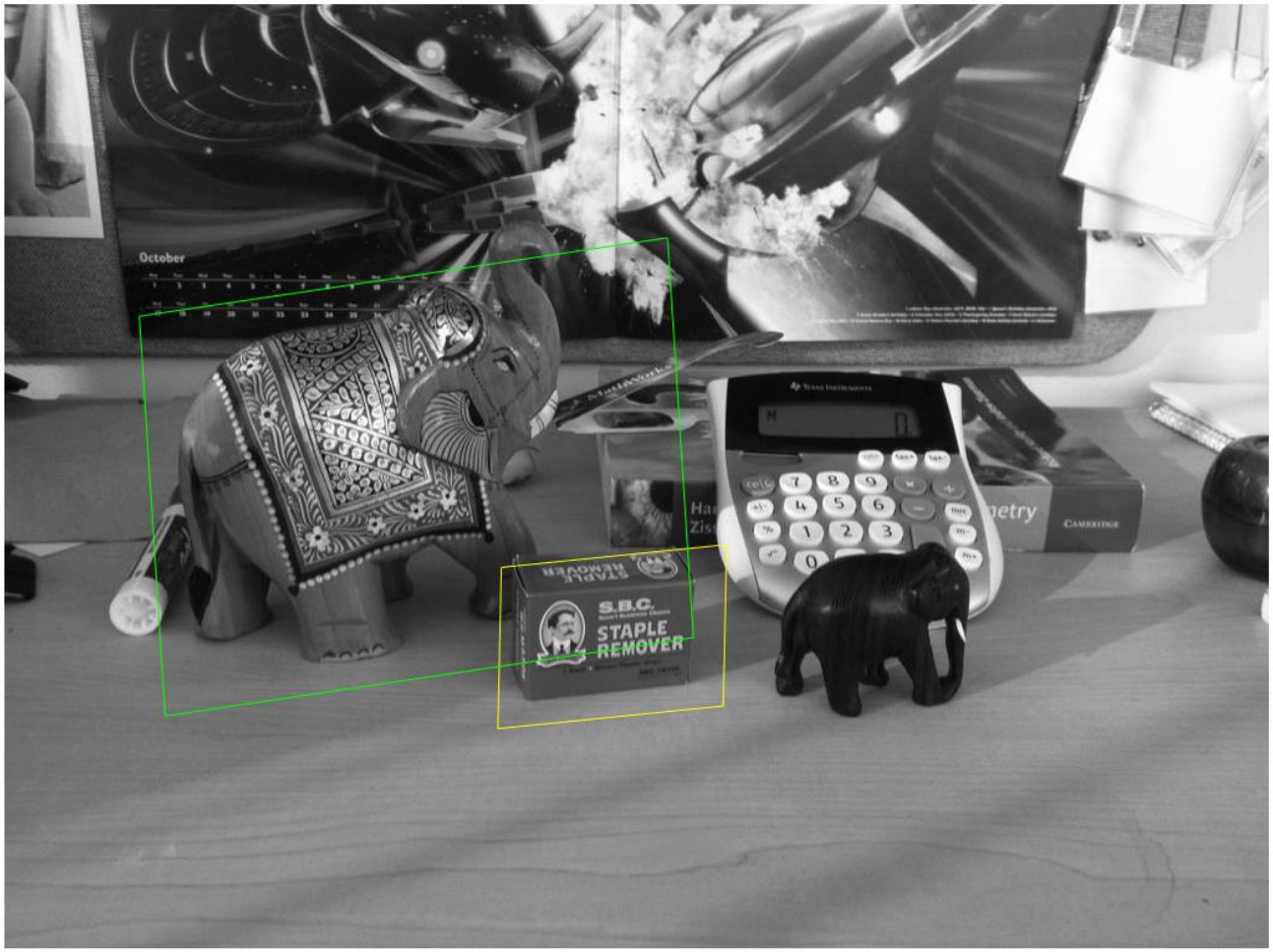
Tracking





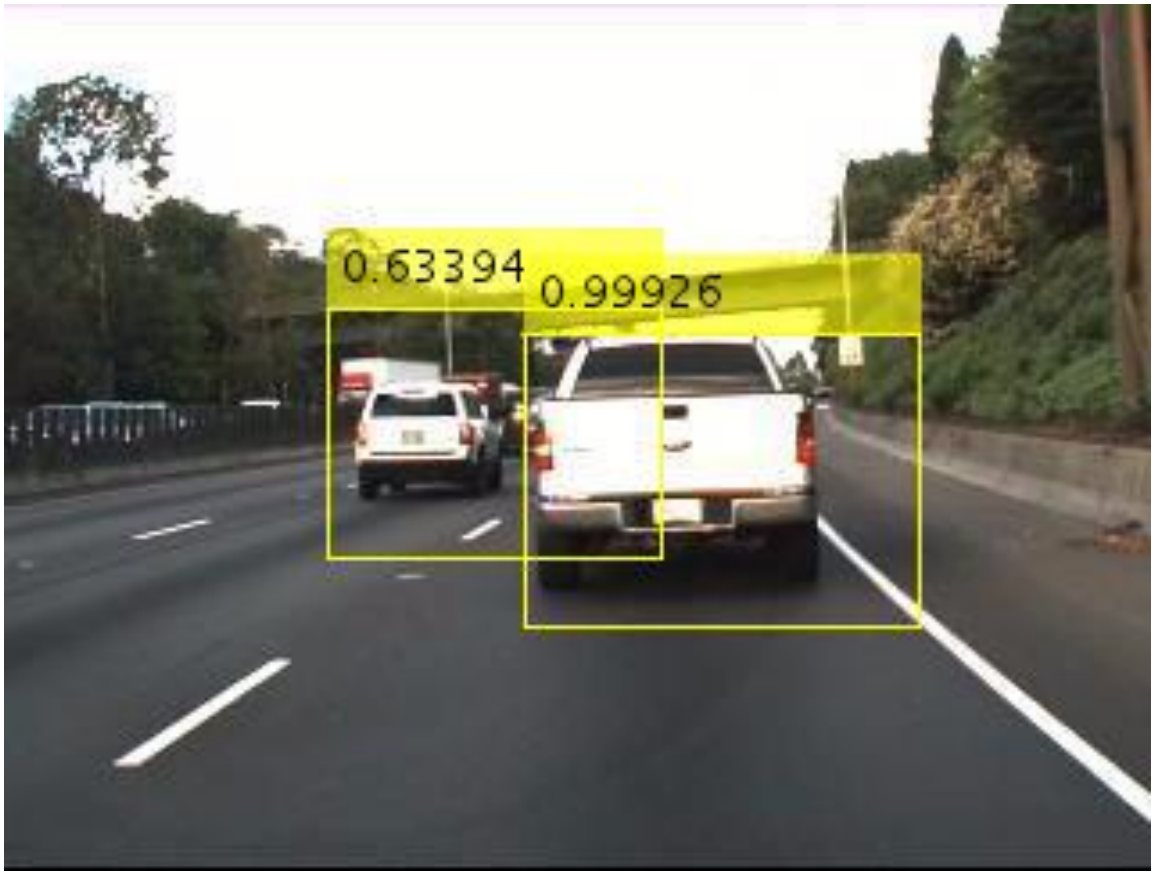
21:10
2018/10/24

1.2 Why do we learn Matlab?



FeatureBasedObjectDetectionExample

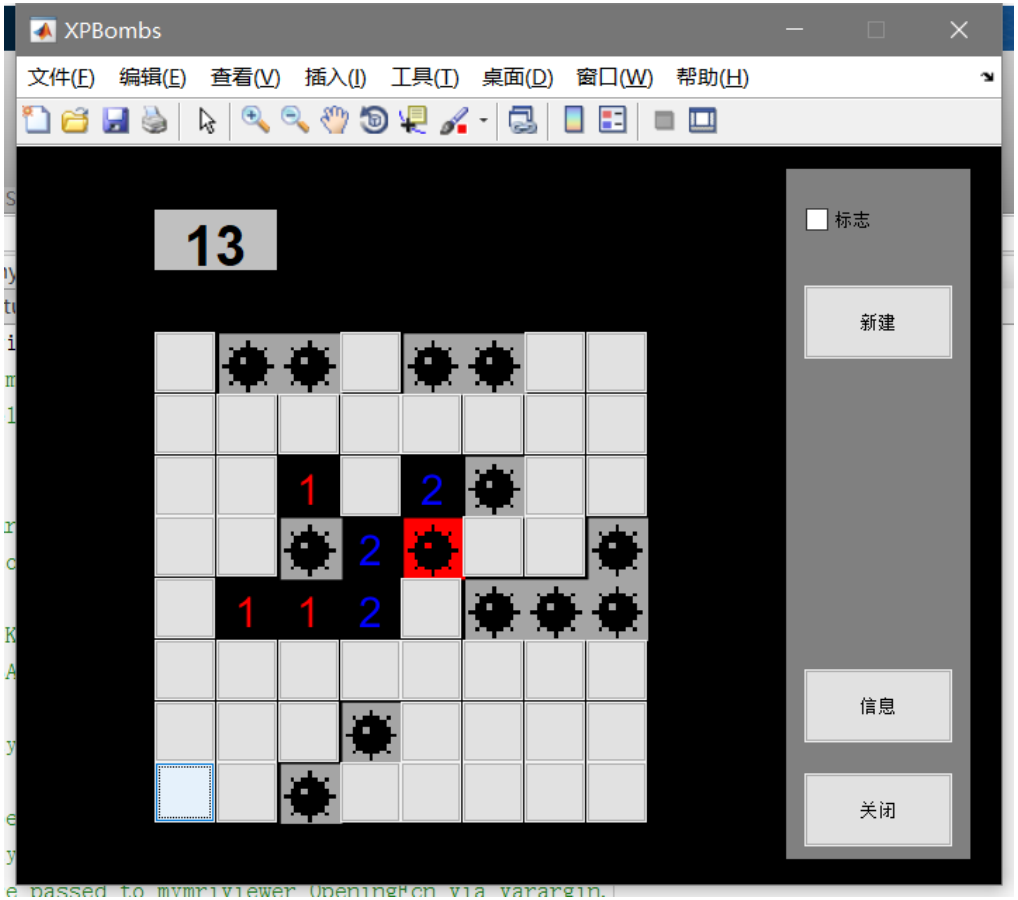
1.2 Why do we learn Matlab?



DeepLearningFasterRCNNObjectDetectionExample

game

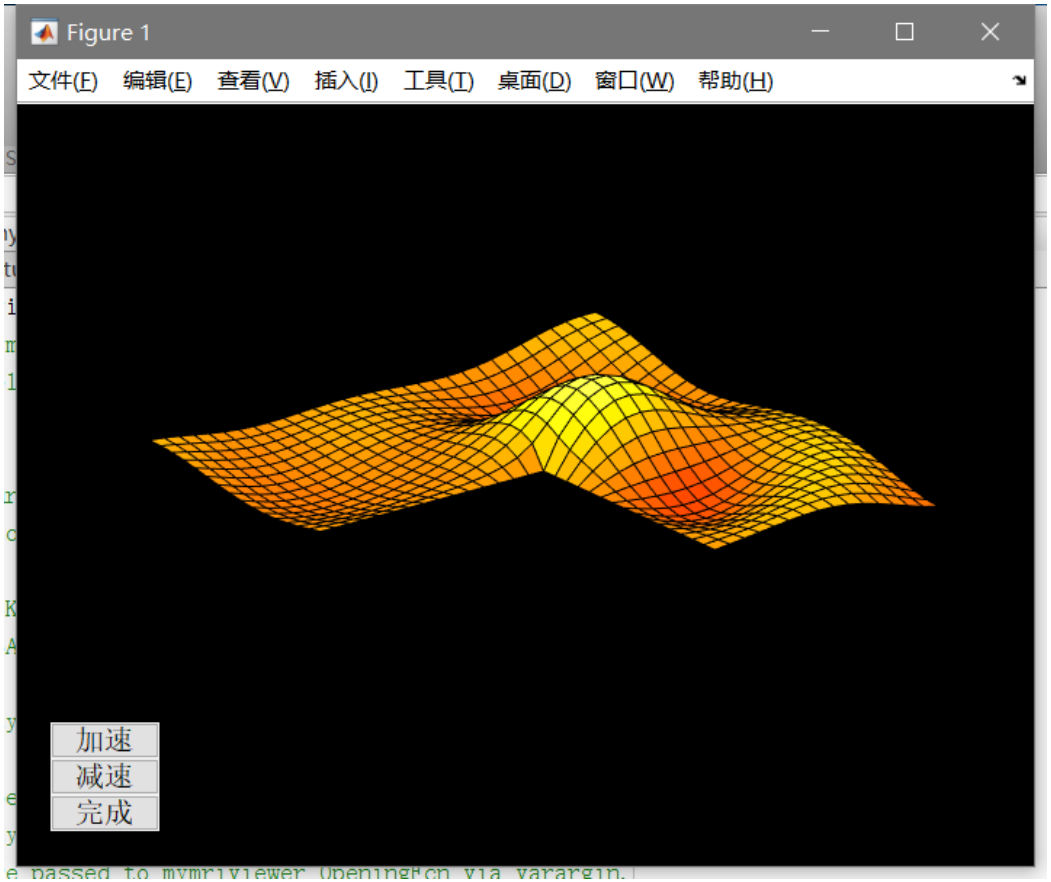
```
>> xpbombs
```



A minesweeper Game.

animation

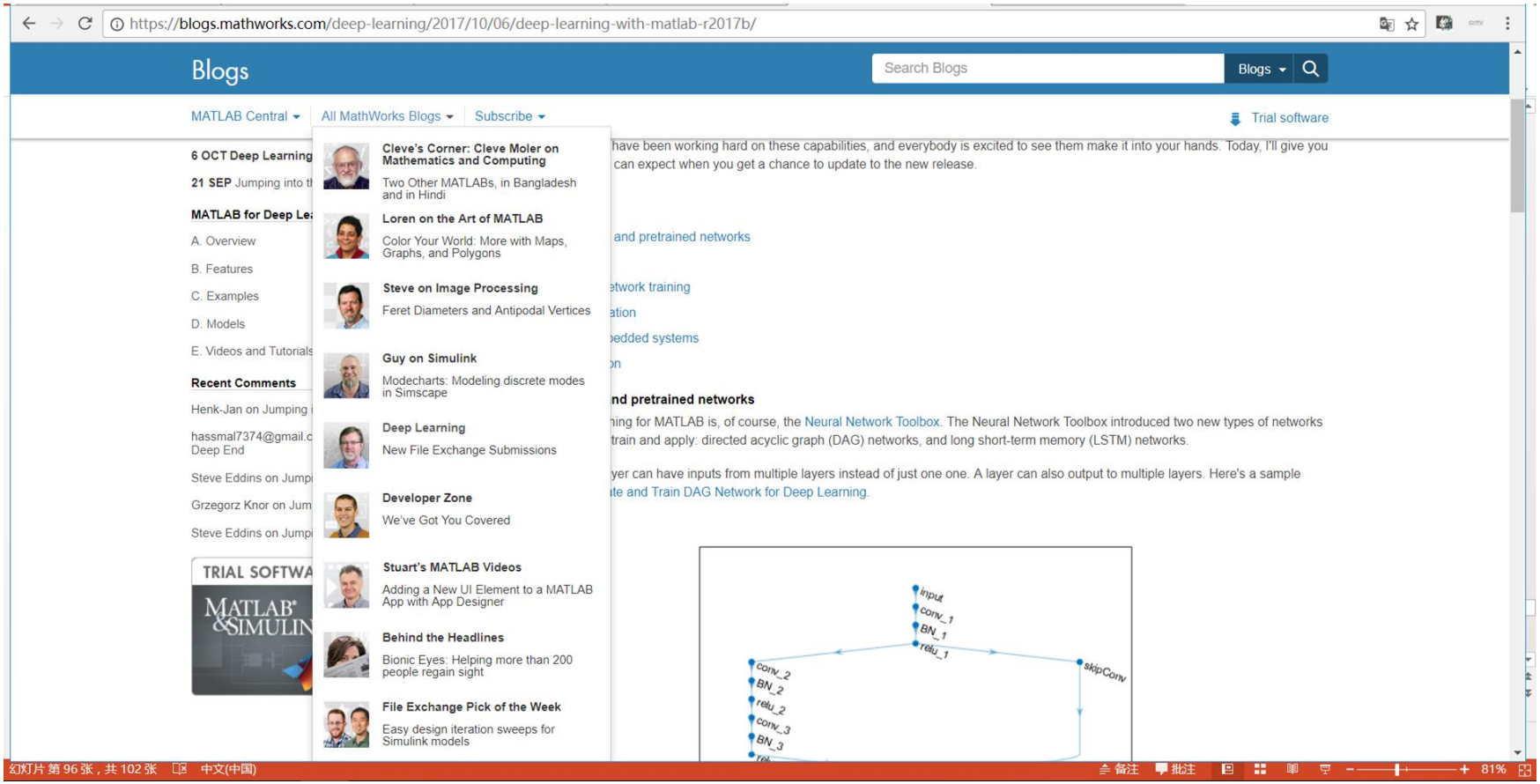
```
>> vibes
```



An animation of vibration.

the-story-behind-the-matlab-default-image

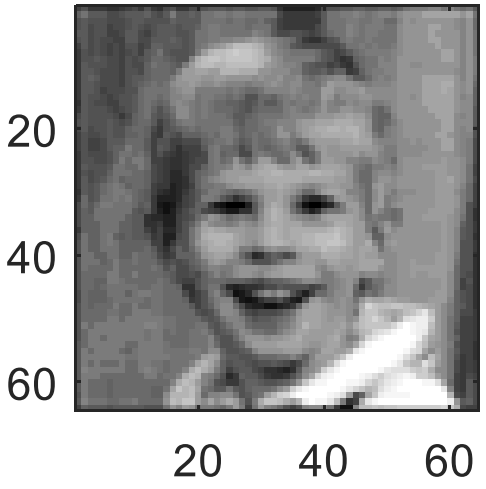
https://blogs.mathworks.com/steve/2006/10/17/the-story-behind-the-matlab-default-image/?utm_medium=social&utm_source=wechat_session



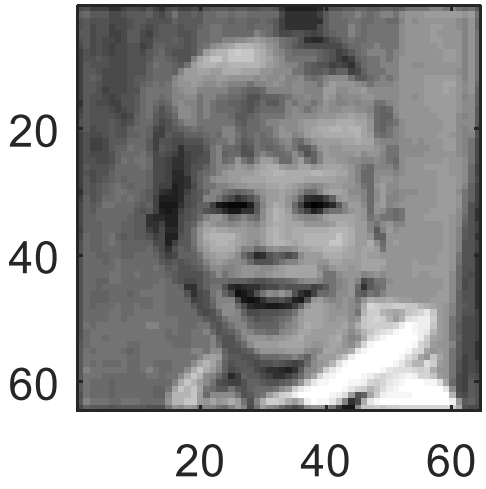
the-story-behind-the-matlab-default-image

```
imagesc(get(0,'DefaultImageCData'))
colormap(gray(32))
axis image
```

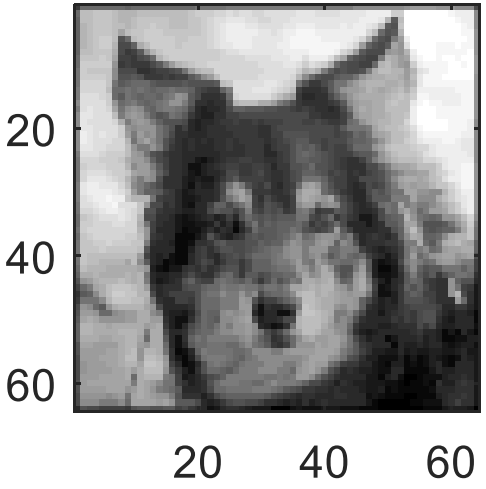
The default image



The integer part



The fractional part



the-story-behind-the-matlab-default-image

the oldest son

the dog of one developer

another pet of one developer

3-by-3 Hilbert matrix

```
  9   -36    30
-36   192  -180
 30  -180   180
```

company's original logo

Loren's favorite number

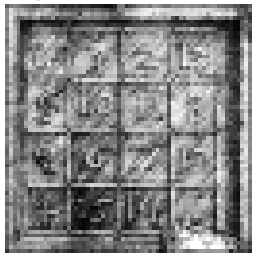
```
17
```

3-by-3 magic square

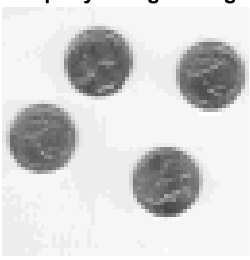
```
8 1 6
3 5 7
4 9 2
```

the youngest son

magic square hidden in Melancolia

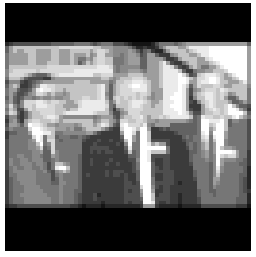


company's original logo



Loren at age 4

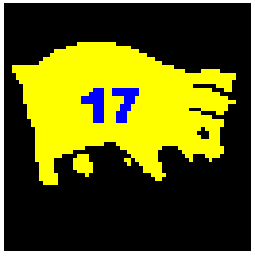
Wilkinson, Givens, and Forsythe



the author

original default image

a yellow pig



15 interesting pictures

1.3 How to learn Matlab?

Class study

- Lectures
- Examples

Exercises

- Homework
- Demo
- help

Projects

- Solve your problems
- Applications

1.4 Course arrangement

- 1.4.1 Course Contents
- 1.4.2 Teaching material
- 1.4.3 Teacher information
- 1.4.4 Assessment
- 1.4.5 Homework
- 1.4.6 Objective

1.4.1 Course Contents

- The MATLAB user interface
- Working with MATLAB variables
- Plotting and data visualization
- Matrix and Array Operations
- M-Files (script and function)
- Data types and Data input and output
- Programming Techniques
- Building graphical user interfaces（GUI）
- Matlab application

1.4.2 Teaching material

MATLAB Programming for Engineers

(Second Edition)

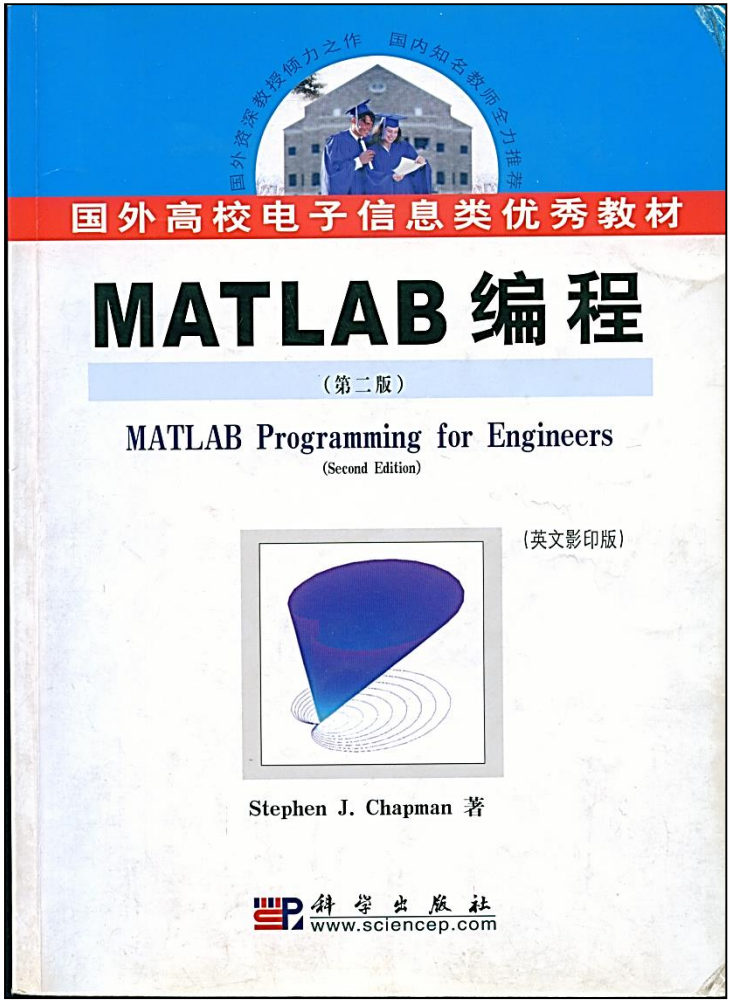
by Stephen J. Chapman

(英文影印版)

科学出版社

Science Press

www.sciencep.com



1.4.3 Teacher information

Teacher: Dr. SUN Bing (孙兵)

- E-mail : bingsun@buaa.edu.cn
- Office room : New Main Building F-617
- Office Tel. : 82338670

Assistants: ABDUL HANAN

- E-mail :ahad_523@Hotmail.com
- Office room : New Main Building G-409

1.4.4 Assessment

Your grade will be computed based upon your final Project, your home works problem set grades, and your attendance record in class.

Course Hours: 32 **Credits: 2**

The weighting on these factors are as follows:

- Attendance records in class 10%
- 6 Home works 60%
- Design and implement a GUI project 30%

1.4.5 Homework

There are 6 homework. The requirements are:

1. Copying is not allowed and will be punished.

2. In your homework M-file the first two comment lines should be your full name and your student ID.

   For example,

   % Student name: xxxxxx xxxxx

   % Student ID : LSxxxxxx

3. Submit your M-files to the following website.

How to submit homework

- Go to **http://www.bhmatlab.cn**
- **Your username and initial password are all your student ID. After login, you can modify them if you need.**
- **After entering the homepage of the website, you can see four sections.**

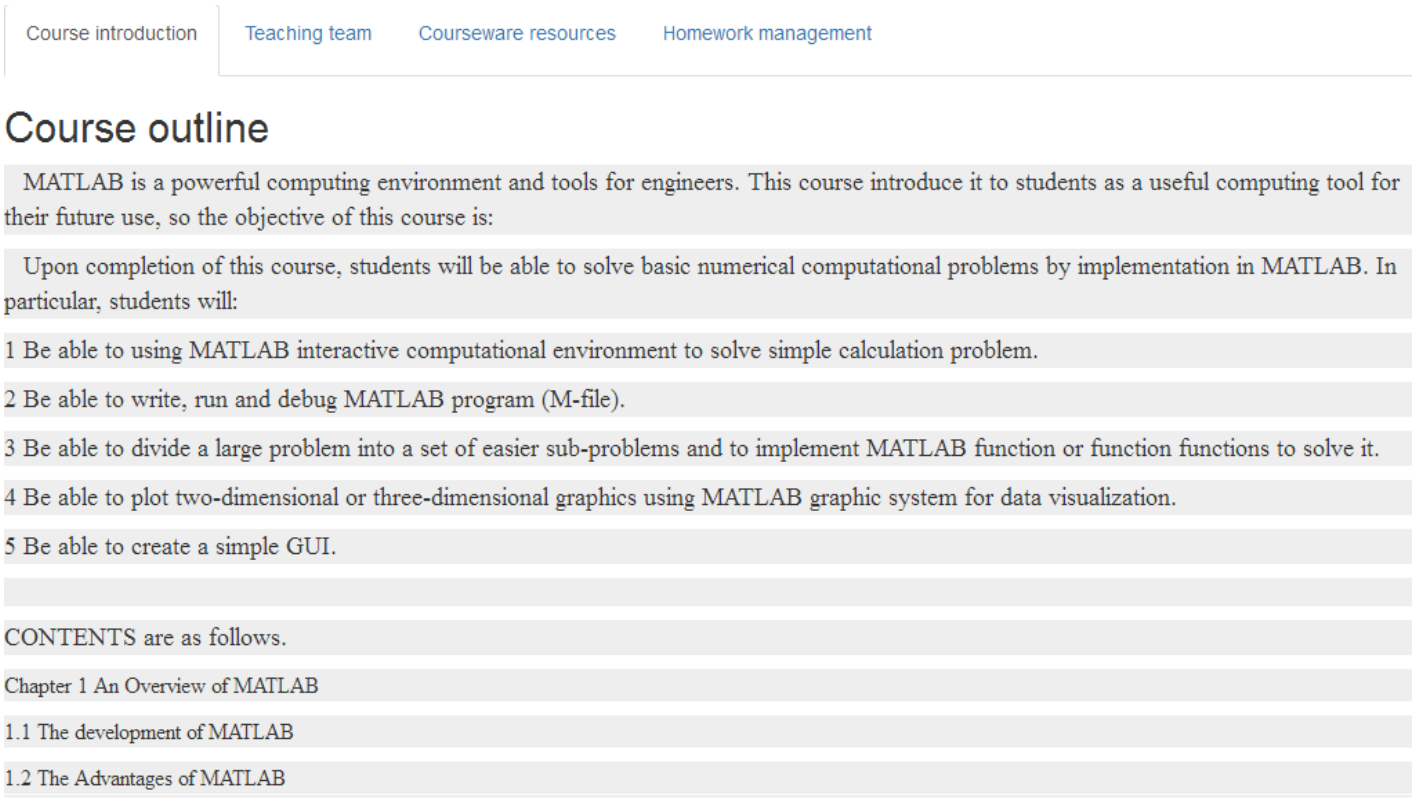
user management

As for the user management in the upper left corner, hereby you can change the password or log out safely.

Course introduction

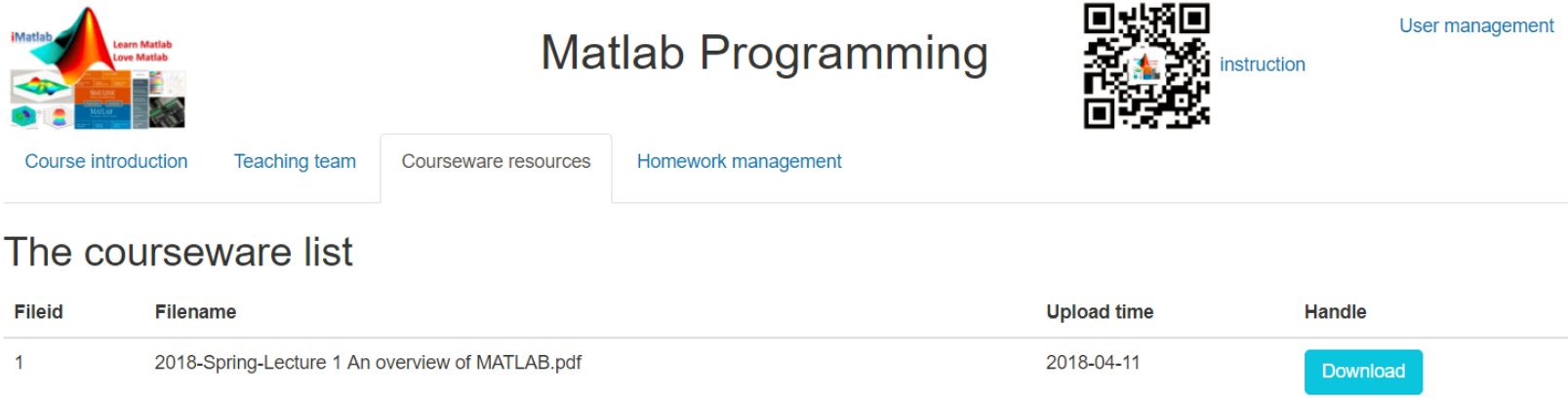
At the section of Course introduction, you can browse the course outline, and then we'll briefly introduce

MATLAB Programming and our course chapters.



Courseware resources

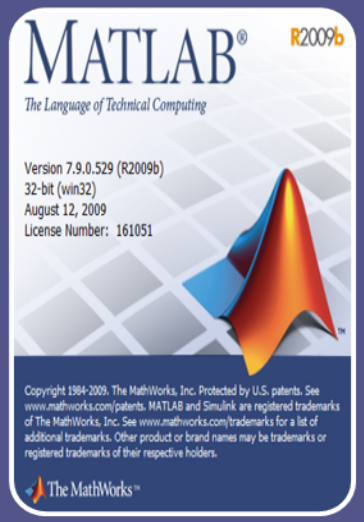
Click Courseware resources, we provide the courseware used in teaching.



Homework management

Click Homework management, here you can download the task and upload the solution. Click the button Upload, it will display the hidden upload part. Please pay attention to the deadline and make sure that you submit your homework in time. If the deadline passed, the button Upload will disappear automatically.

Contents



About Matlab

- History of Matlab
- Advantages
- Disadvantages
- How to start and use Matlab

2.1 The History of MATLAB

- In the mid-1970s, Cleve Moler and several colleagues developed the FORTRAN subroutine libraries called LINPACK and EISPACK under a grant from the National Science Foundation.

- LINPACK is a collection of FORTRAN subroutines for solving linear equations, while EISPACK contains subroutines for solving eigenvalue problems.

2.1 The History of MATLAB

- In the late 1970s, Cleve, who was then chairman of the computer science department at the University of New Mexico, wanted to be able to teach students in his linear algebra courses using the LINPACK and EISPACK software. However, he didn't want them to have to program in FORTRAN, because this wasn't the purpose of the course.

- So, as a "hobby" on his own time, he started to write a program that would provide simple interactive access to LINPACK and EISPACK. He named his program MATLAB, for MATrix LABoratory.

2.1 The History of MATLAB

- Over the next several years, when Cleve would visit another university to give a talk, or as a visiting professor, he would end up by leaving a copy of his MATLAB on the university machines.
- In early 1983, John Little was exposed to MATLAB because of a visit Cleve made to Stanford University. Little, an engineer, recognized the potential application of MATLAB to engineering applications.
- In 1983, Little teamed up with Moler and Steve Bangert to develop a second generation, professional version of MATLAB written in C and integrated with graphics.

2.1 The History of MATLAB

- The MathWorks, Inc. was founded in 1984 to market and continue development of MATLAB.
- Now thousands of Universities around the world use the MathWorks' products for teaching and research in a broad range of technique.

Fortran and Scientific Computing

- Engineering and scientific applications involve a lot of "number crunching".
- For many years, the main language for this was FORTRAN -- first "high level" programming language, and especially designed for numerical computing.
- Here's a Fortran code to solve a $x^2 + b\,x + c = 0$:

```
C     Solve a quadratic equation (this is a comment).
      DESC = B*B - 4*A*C
      IF ( DESC .LT. 0.0 ) GOTO 10
        DESC = SQRT(DESC)
        X1 = (-B + DESC)/(2.0*A)
        X2 = (-B - DESC)/(2.0*A)
        WRITE(6,*) "SOLUTIONS ARE ",X1," AND ", X2
        RETURN
   10 WRITE(6,*) "EQUATION HAS COMPLEX ROOTS"
      RETURN
```

Problems using FORTRAN

"*Number crunching*" on a computer can be tricky.

Problems that occur are:

- *loss of precision and inaccurate results:*

  X = 1/3

  Y = 1.0 - 3*X

  Y "should" equal 0, but probably does not!

- *underflow and overflow:*

  X = 1.0E30, X*X → too big!

- *efficient coding of algorithms not always obvious*

  DO 10 N=1,100000

  10 Y(N) = SQRT(2.0)*X(N) ← inefficient!

- *programming errors!*

Solving a Linear System in Fortran

Here's a Fortran code to solve a linear system b = A*x, solve for x.

It doesn't check for degeneracy or zeros.

```
C Solve B = A*X for X.
C N is dimension of vectors and matrix
C Does not use row interchange, scaling.
      SUBROUTINE LINSYS(N, A, X, B, TMP)
      INTEGER N
      DOUBLE PRECISION A(N,N), X(N), B(N)
      DOUBLE PRECISION TMP(N), RATIO
C...  Forward elimination
      DO 13 J=1,N-1
        DO 12 I=J+1,N
          RATIO = -A(I,J)/A(J,J)
          A(I,*) = A(I,*) +RATIO*ROW(J,*)
          DO 11 K=J+1,N
  11      A(I,K) = A(I,K) + RATIO*A(J,K)
          A(I,J) = 0.0
          X(I) = X(I) + RATIO*X(J)
  12    CONTINUE
  11  CONTINUE
```

Continued...

```
C...  Backwards substitution
      X(N) = X(N)/A(N,N)
      DO 21 I=N-1,1,-1
        TMP = X(I)
        DO 20 J=I+1,N
  20    TMP = TMP - A(I,J)*X(J)
        X(I) = TMP/A(I,I)
  21  CONTINUE
      RETURN
      END
```

This is just a small example.

A full program may be 1000's of lines long.

Need for Numerical Libraries

- The U.S. government recognized these problems, and the inefficiency of many engineers all writing the *same* algorithms... again and again.
- So, they commissioned *numerical analysts* to write good quality algorithms for common tasks.
- Make the results freely available as "libraries" of subroutines than anyone can use in their programs.
- Libraries are available at: www.netlib.org

Examples of Numerical Libraries

- BLAS (Basic Linear Algebra Subroutines): operations on vectors, like adding to vectors, dot product, norm.
- LINPACK: linear algebra subroutines for vector-matrix operations, solving linear systems, factoring a matrix, inverting a matrix. Later replaced by LAPACK.
- EISPACK: compute eigenvalues and eigenvectors of matrices.
- Example: solve A*x = b using LINPACK

```
C.... factor the A matrix
      CALL SGEFA(A, N, N, IPVT, INFO)
C.... copy B vector into X vector
      CALL SCOPY(N, B, 1, X, 1)
C.... solve the system of equations
      CALL SGESL(A, N, N, IPVT, X, 0)
```

Still Not Easy Enough!

- Cleve Moler, mathematician, C.S. Professor, and co-author of LINPACK, thought this is still too much work:
  - write FORTRAN, compile, debug, compile, run...
- He wanted to give students easy access to LINPACK.
- So, he wrote MATLAB ("Matrix Laboratory").
  - interactive
  - easy input, output
  - operations on a whole vector or matrix at once
- Example: solve b = A*x in Matlab...

  x = A \ b

Immediate Popularity!

- MATLAB quickly became quite popular and used for both teaching and research. It was also *free*.
- An engineer, Jack Little, saw Matlab during a lecture by Cleve Moler at Stanford University.
- He saw the commercial potential and (with permission)
  - rewrote Matlab in C
  - added "M-files" (stored programs)
  - many new features and libraries
  - founded *The Mathworks* to market it.

Software principles...

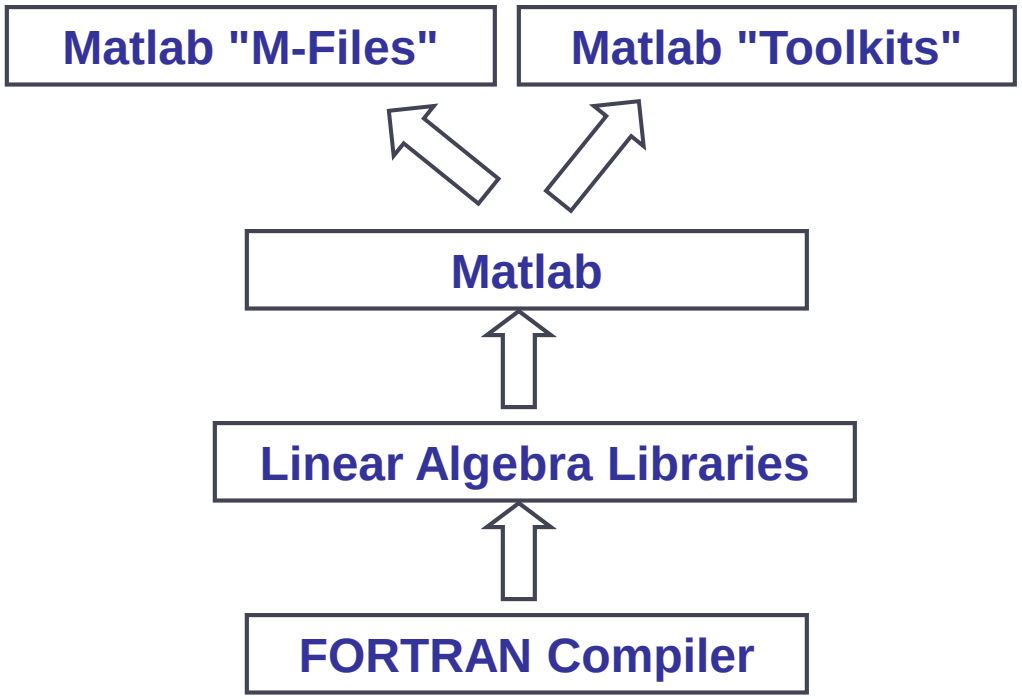
- Matlab illustrates some useful design concepts for software.

Extensible using "Toolkits" or user-contributed programs called M-files.

Interactive user interface; hides boring details

Modular, reusable software components

Standard base platform



The MATLAB SYSTEM

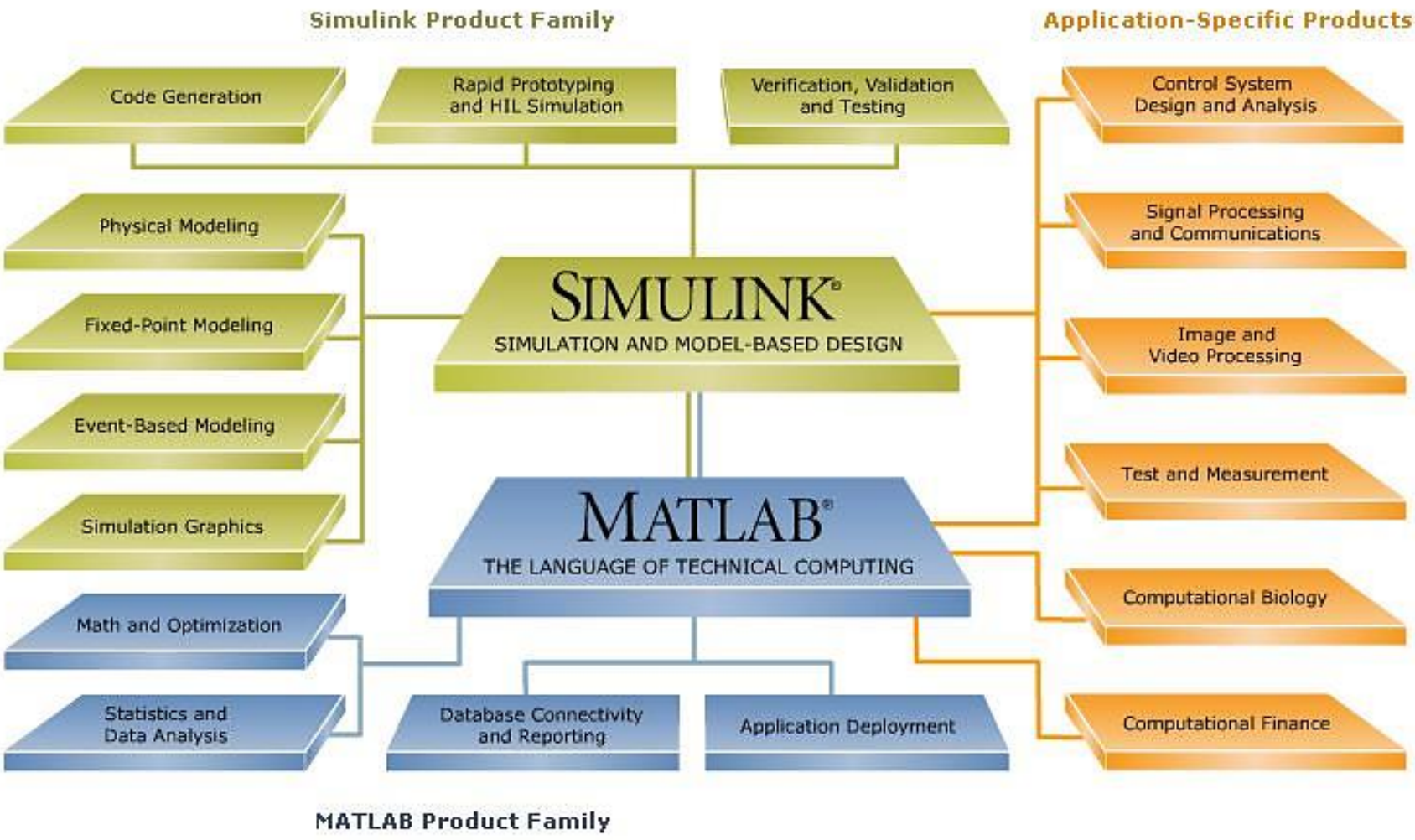
The MATLAB system consists of 5 main parts:

1. Desktop Tools and Development Environment.
2. The MATLAB Mathematical Function Library.
3. The MATLAB Language.
4. Graphics.
5. The MATLAB External Interfaces/API.

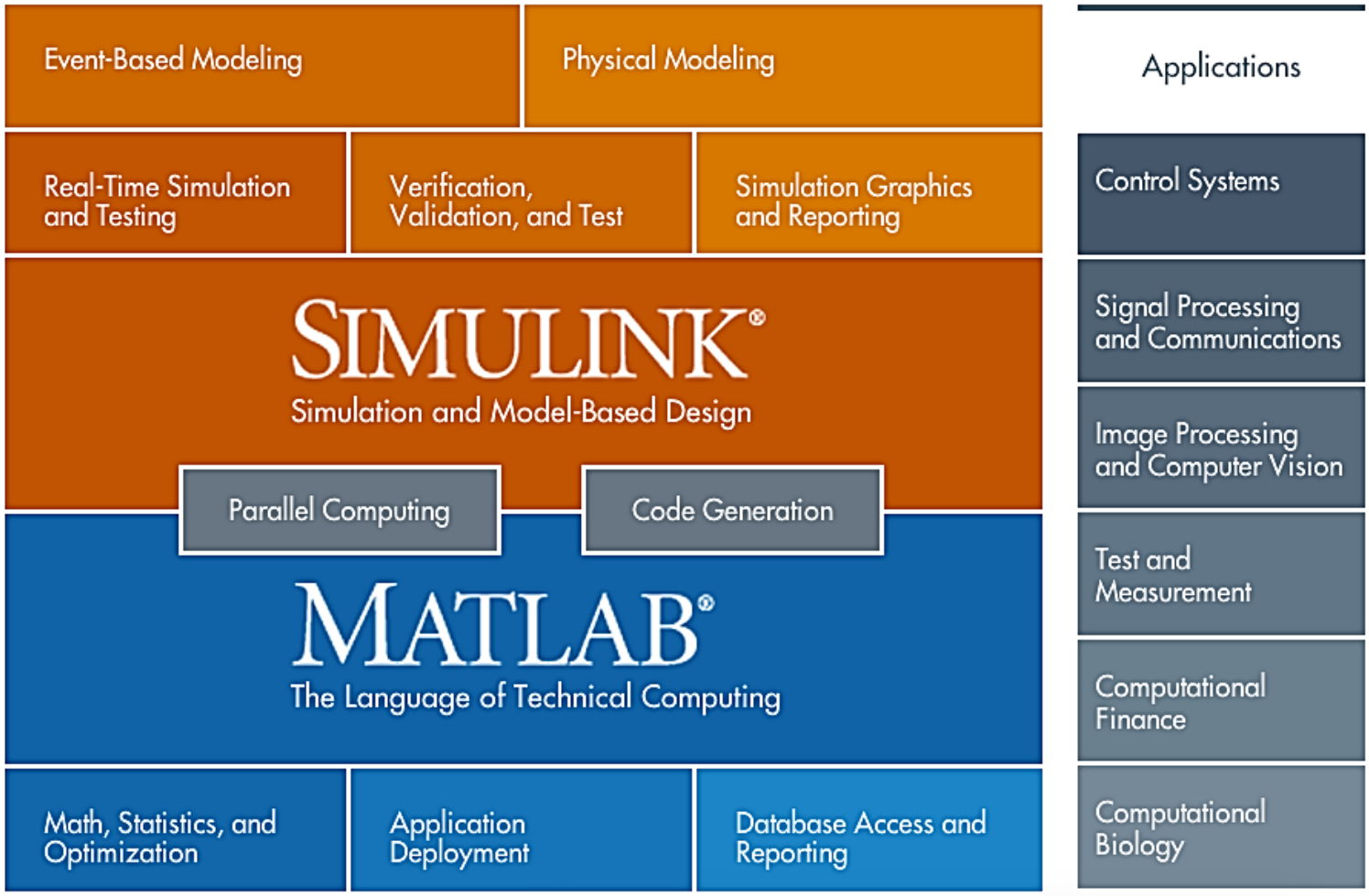
Matlab Today

- Millions of users!
- A standard tool, both professional and academic use
- "Toolboxes" providing functions for many applications:
  - control systems
  - identification
  - neural networks
  - bio-informatics
  - statistics and time-series analysis
- Can do symbolic mathematics, too.
- Simulink: GUI based simulation tool

Matlab Product Family@2009



Matlab Product Family@2015



Explore nearly 100 products in the MATLAB and Simulink product families for technical computing and Model-Based Design.

Release history

Version	*No.*	*Time*	*Version*	*No.*	*Time*
MATLAB 1.0		1984	MATLAB 7.8	R2009a	2009.3.6
MATLAB 2		1986	MATLAB 7.9	R2009b	2009.9.4
MATLAB 3		1987	MATLAB 7.10	R2010a	2010.3.5
MATLAB 3.5		1990	MATLAB 7.11	R2010b	2010.9.3
MATLAB 4		1992	MATLAB 7.12	R2011a	2011.4.8
MATLAB 4.2c	R7	1994	MATLAB 7.13	R2011b	2011.9.1
MATLAB 5.0	R8	1996	MATLAB 7.14	R2012a	2012.3.1
MATLAB 5.1	R9	1997	MATLAB 8.0	R2012b	2012.9.11
MATLAB 5.1.1	R9.1	1997	MATLAB 8.1	R2013a	2013.3.7
MATLAB 5.2	R10	1998	MATLAB 8.2	R2013b	2013.9.9
MATLAB 5.2.1	R10.1	1998	MATLAB 8.3	R2014a	2014.3.6
MATLAB 5.3	R11	1999	MATLAB 8.4	R2014b	2014.10.2
MATLAB 5.3.1	R11.1	1999	MATLAB 8.5	R2015a	2015.3.5
MATLAB 6.0	R12	2000	MATLAB 8.6	R2015b	2015.9.3
MATLAB 6.1	R12.1	2001	MATLAB 9.0	R2016a	2016.3.3
MATLAB 6.5	R13	2002	MATLAB 9.1	R2016b	2016.9.15
MATLAB 6.5.1	R13SP1	2003	MATLAB 9.2	R2017a	2017.3.9
MATLAB 6.5.2	R13SP2	2003	MATLAB 9.3	R2017b	2017.9.26
MATLAB 7	R14	2004	MATLAB 9.4	R2018a	2018.3.15
MATLAB 7.0.1	R14SP1	2004	MATLAB 9.5	R2018b	2019.9.12
MATLAB 7.0.4	R14SP2	2005	MATLAB 9.6	R2019a	2019.3.20
MATLAB 7.1	R14SP3	2005	MATLAB 9.7	R2019b	2019.9.11

Release History

Version[47]	Release name	Number	Bundled JVM	Year	Release date	Notes
MATLAB 1.0				1984		
MATLAB 2				1986		
MATLAB 3				1987		
MATLAB 3.5				1990		Ran on DOS but needed at least a 386 processor; version 3.5m needed math coprocessor
MATLAB 4				1992		Ran on Macintosh
MATLAB 4.2c				1994		Ran on Windows 3.1x, needed a math coprocessor
MATLAB 5.0	Volume 8			1996	December, 1996	Unified releases across all platforms
MATLAB 5.1	Volume 9			1997	May, 1997	
MATLAB 5.1.1	R9.1					
MATLAB 5.2	R10			1998	March, 1998	Last version working on classic Mac's

(40 versions)

MATLAB 9.0	R2016a	35	1.7.0_60	2016	March 3, 2016	Live Scripts: interactive documents that combine text, code, and output (in the style of Literate programming);[86] App Designer: a new development environment for building apps (with new kind of UI figures, axes, and components);[87] pause execution of running programs using a Pause Button
MATLAB 9.1	R2016b	36	1.7.0_60		September 15, 2016	define local functions in scripts;[88] automatic expansion of dimensions (previously provided via explicit call to `bsxfun`); `tall` arrays for Big data;[89] new `string` type;[90] new functions to encode/decode JSON;[91] official MATLAB Engine API for Java[36]
MATLAB 9.2	R2017a	37	1.7.0_60	2017	March 9, 2017	MATLAB Online: cloud-based MATLAB desktop accessed in a web browser;[92] double-quoted strings; new `memoize` function for Memoization; expanded object properties validation;[93] mocking framework for unit testing;[94] MEX targets 64-bit by default; new `heatmap` function for creating heatmap charts[95]
MATLAB 9.3	R2017b	38	1.8.0_121		September 21, 2017	
MATLAB 9.4	R2018a	39	1.8.0_144	2018	March 15, 2018[96]	
MATLAB 9.5	R2018b	40	1.8.0_152		September 12, 2018	

https://en.wikipedia.org/wiki/MATLAB

2.2 The Advantages of MATLAB

- Ease to Use
- Platform Independent
- Predefined Functions
- Device-Independent Plotting
- Graphical User Interface
- MATLAB Compiler

2.2.1. Ease to Use

- MATLAB is an interpreted language, like Basic. The MATLAB program can be used as a scratch pad to evaluate expressions typed at the command line, or can be used to execute large prewritten programs(M-file).
- Many program development tools are provided to make the program easy to use.
- See examples.

Example 1. To solve AX=B.

- It is very easy to solve linear equations system AX=B by using matrix left division operator \ in MATLAB.
- The solution is X=A\B
- Suppose the A is 4×4 square coefficient Matrix, B is 4×1 constants vector.

Used as Scratch Pad

X=A\B

Command Window

```
>> A = [1 2 1 4; 2 0 4 3; 4 2 2 1; -3 1 3 2]

A =

     1     2     1     4
     2     0     4     3
     4     2     2     1
    -3     1     3     2

>> B = [13 28 20 6]'

B =

    13
    28
    20
     6

>> X=A\B

X =

    3.0000
   -1.0000
    4.0000
    2.0000
```

Used to execute M-file

The M-file name is leftdvs.m

```
% sample of Matirx left division operator.
A = [1 2 1 4;2 0 4 3;4 2 2 1;-3 1 3 2]
B = [ 13 28 20 6]'
X = A\B;
% Display the results
Y = X';
str = num2str(Y);
disp(['The X is ',str]);
```

Used to execute M-file

```
>> leftdvs

A =

     1     2     1     4
     2     0     4     3
     4     2     2     1
    -3     1     3     2


B =

    13
    28
    20
     6

The X is 3          -1          4          2
```

2.2.2. Platform Independent

- MATLAB is supported on many different computer systems, providing a large measure of platform independence. It is supported on Windows 95/98/ME/NT/2000/XP and many different versions of UNIX.

- Programs written on any platform will run on all the other platforms. It has very good portability.

2.2.3. Predefined Functions

- The basic MATLAB language has a large build-in library of predefined functions that make your job much easier. For the example 1,we also can use predefined function inv() to solve.
- In addition to build-in library of functions, there are many special-purpose toolboxes available to help solve complex problems in specific areas, such as signal processing, control systems, image processing etc.
- See Example.

Example 2. Solve Linear systems AX=B

- For the example 1, we also can use predefined function inv() to solve it.

```
>> X=inv(A)*B

X =

    3.0000
   -1.0000
    4.0000
    2.0000
```

2.2.4. Device-Independent Plotting

- MATLAB has many integral plotting and imaging commands. The plots and images can be displayed on any graphical output device supported by the computer on which MATLAB is running. It makes MATLAB an outstanding tool for visualizing technical data.

- See Examples.

Example 3. Plots

- MATLAB's plot function makes plotting very easy.

(1) 2-D x-y curve plots

(2) 2-D Polar plots

(3) 3-D curve plots

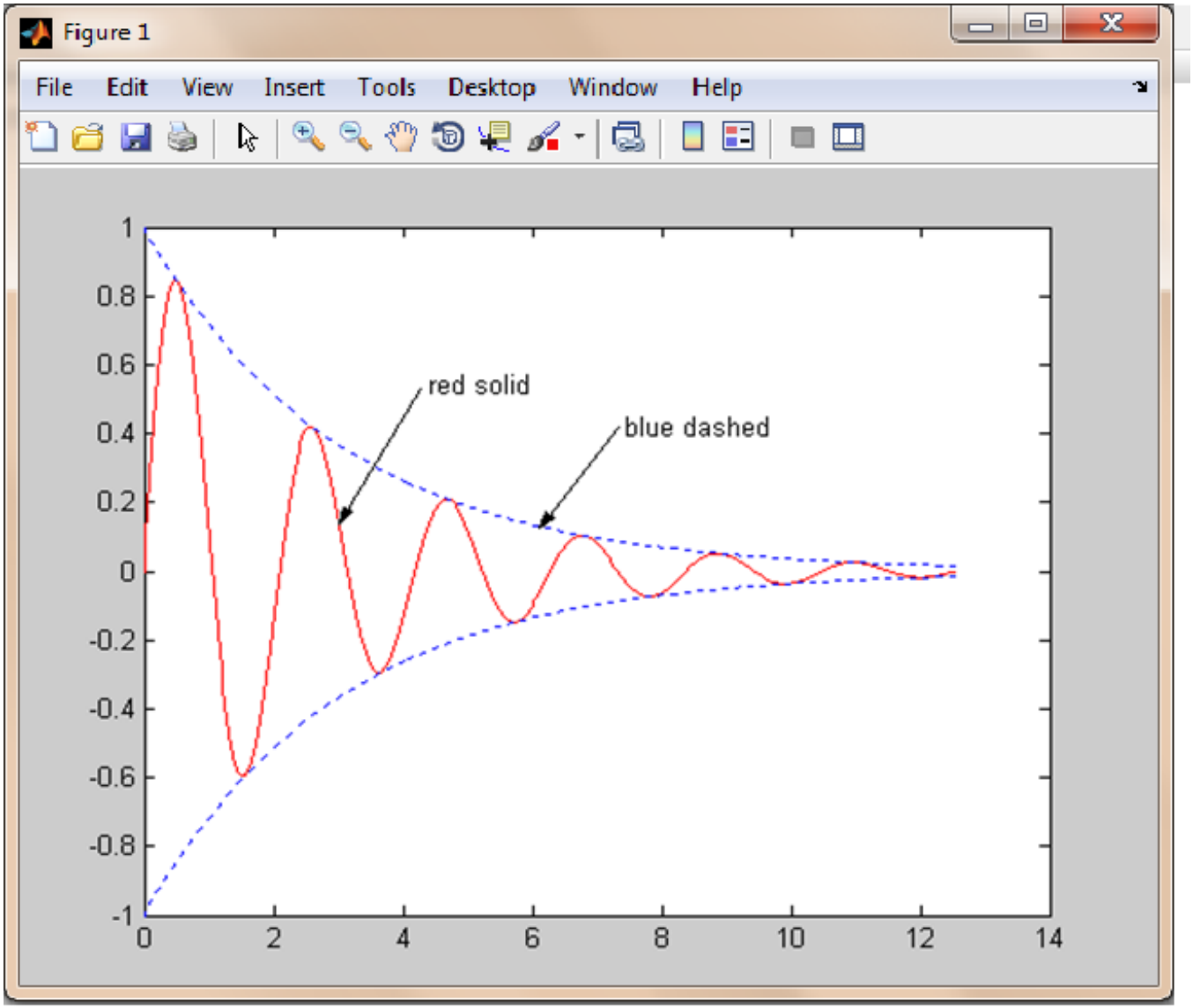
(4) 3-D mesh plots

(5) 3-D surface plots

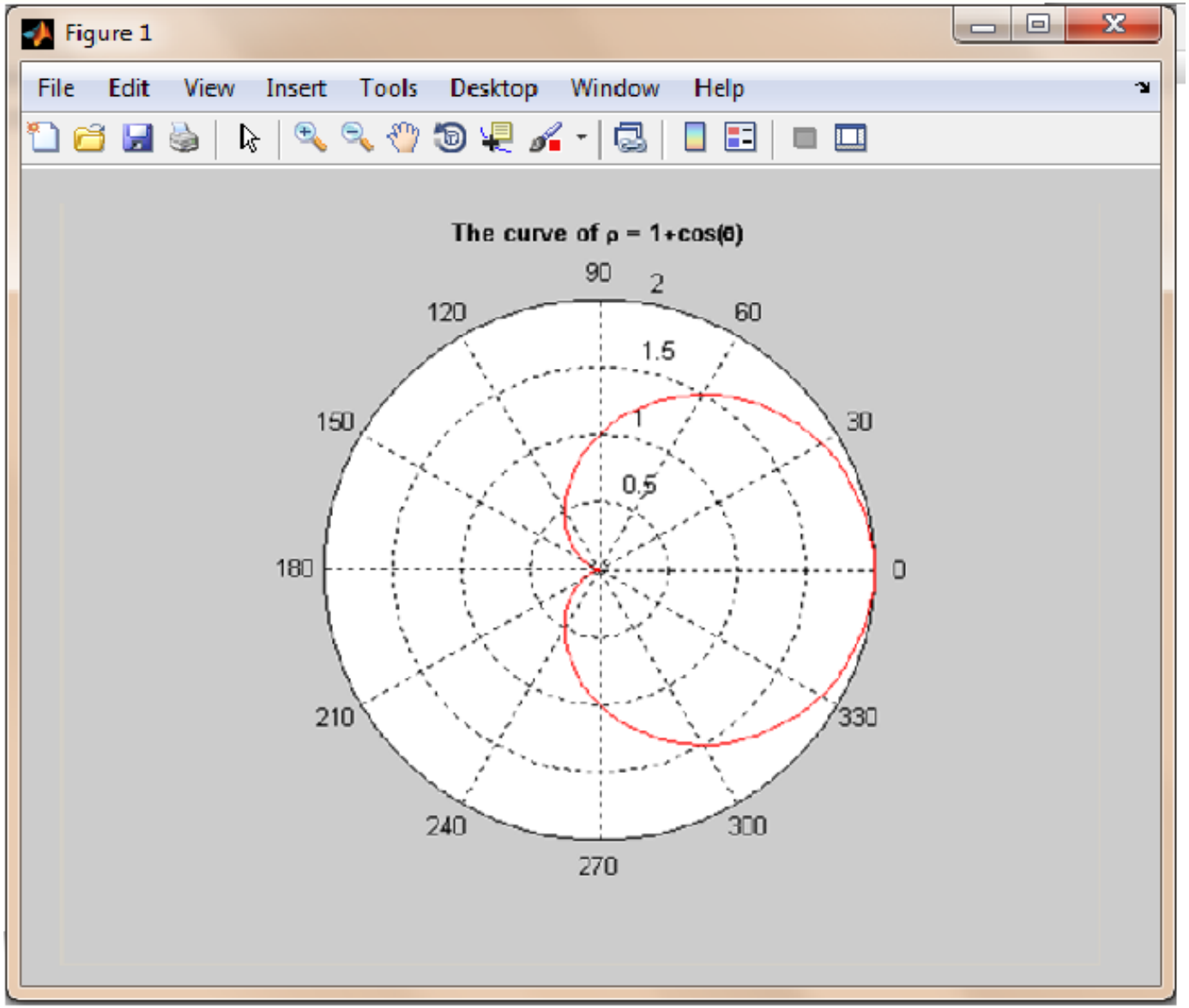
(6) Contour plots

(7) Multi-plotting in One window

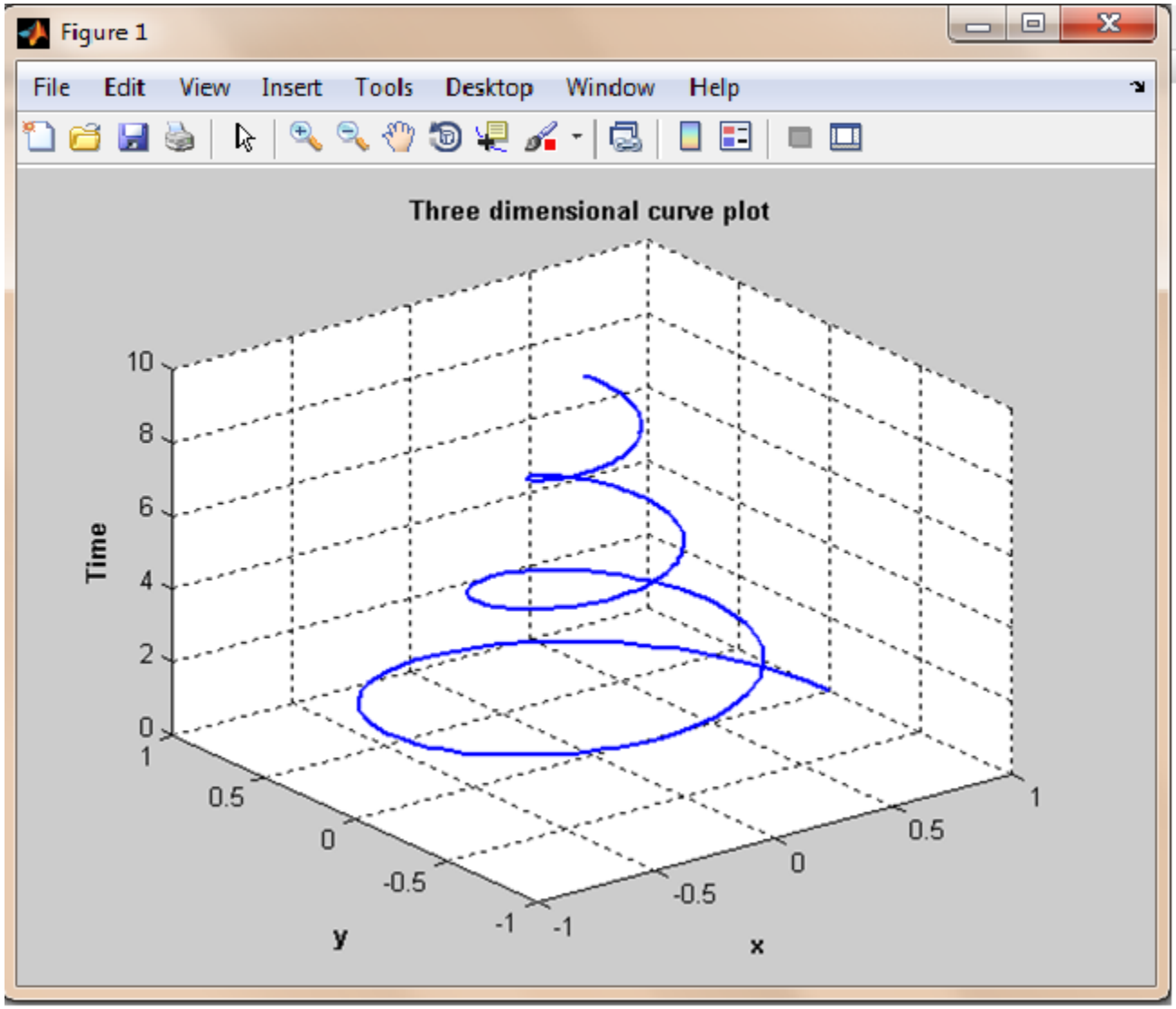
(1) 2-D x-y curve plotting



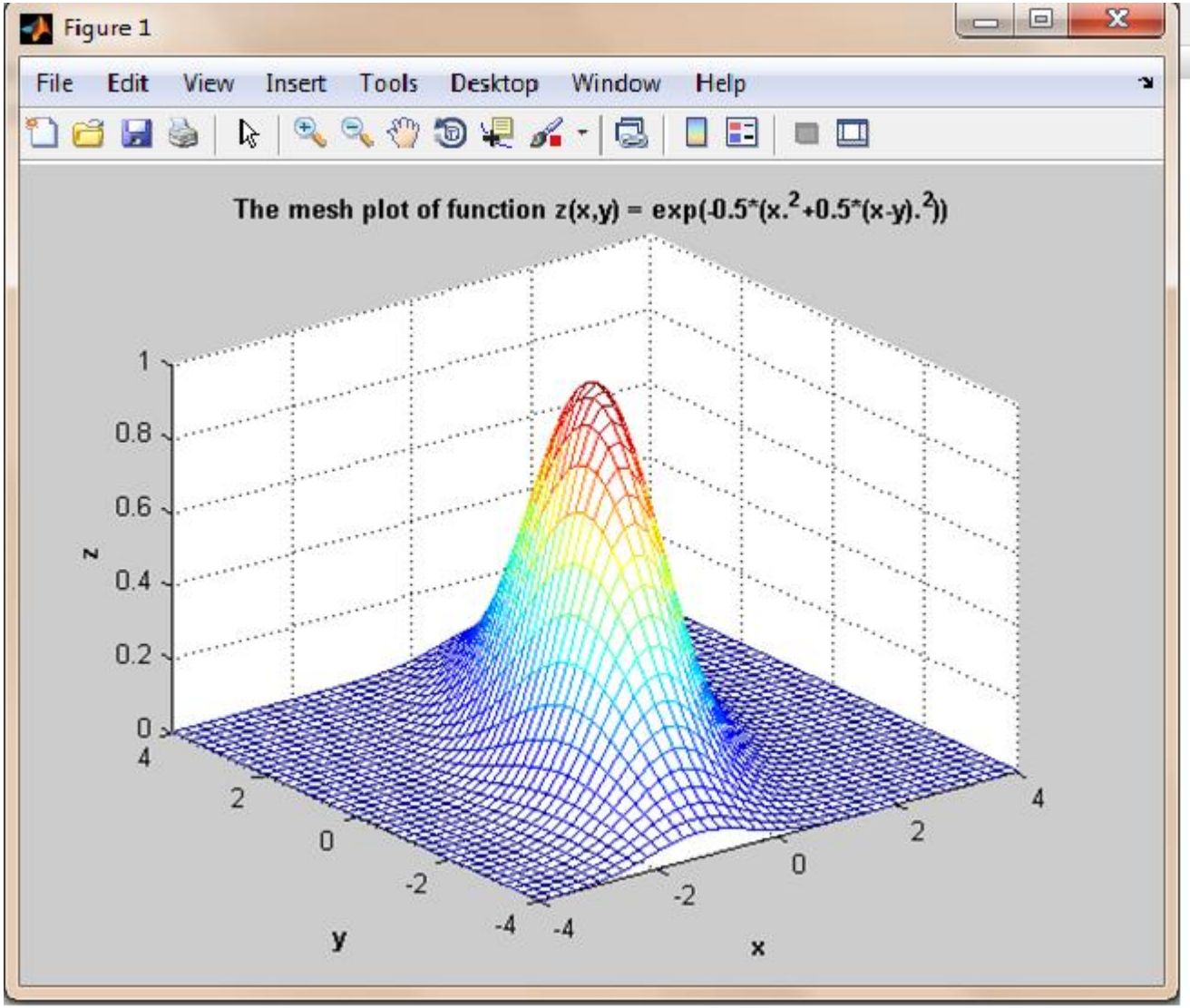
(2) 2-D Polar plotting



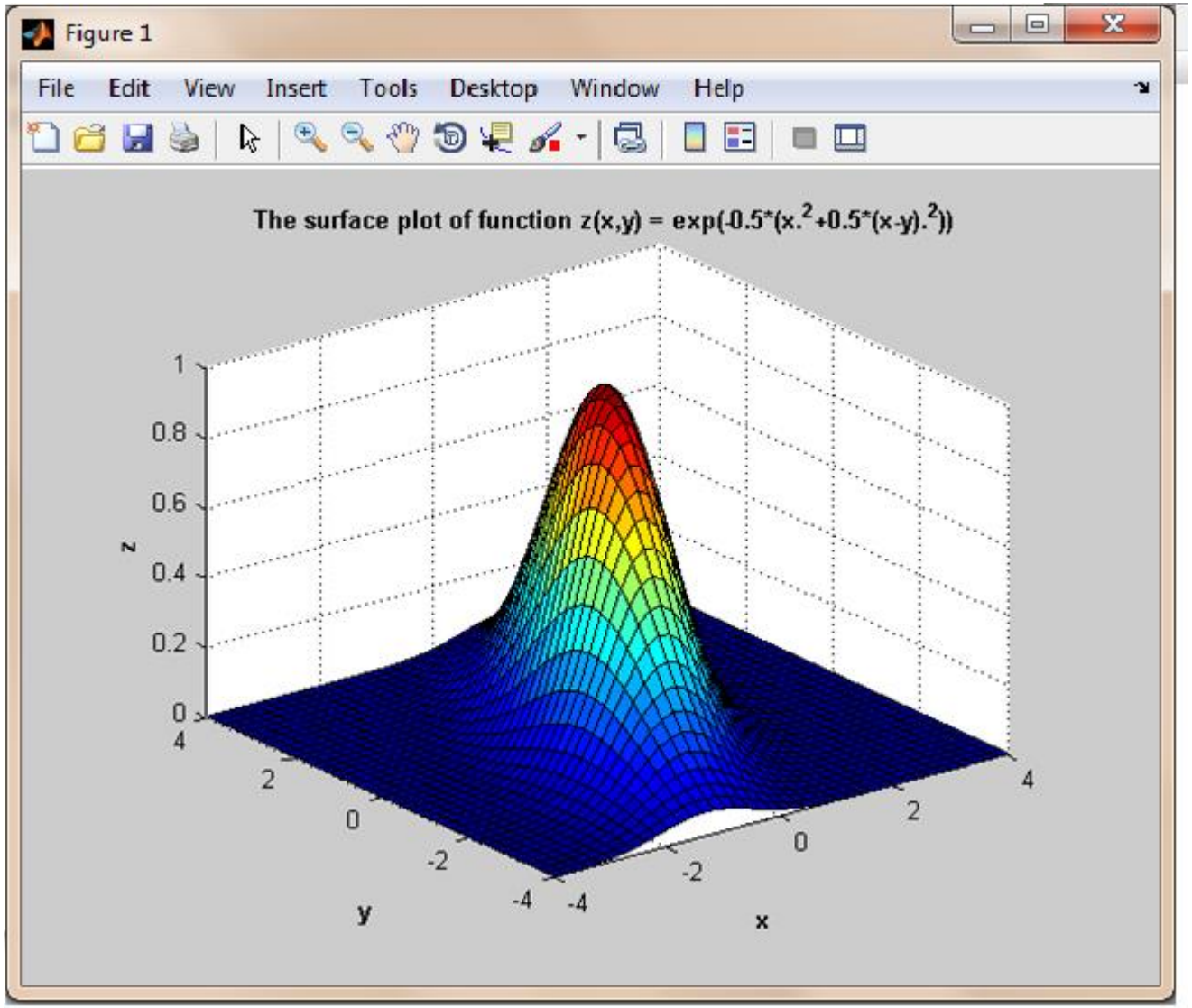
(3) 3-D curve plotting



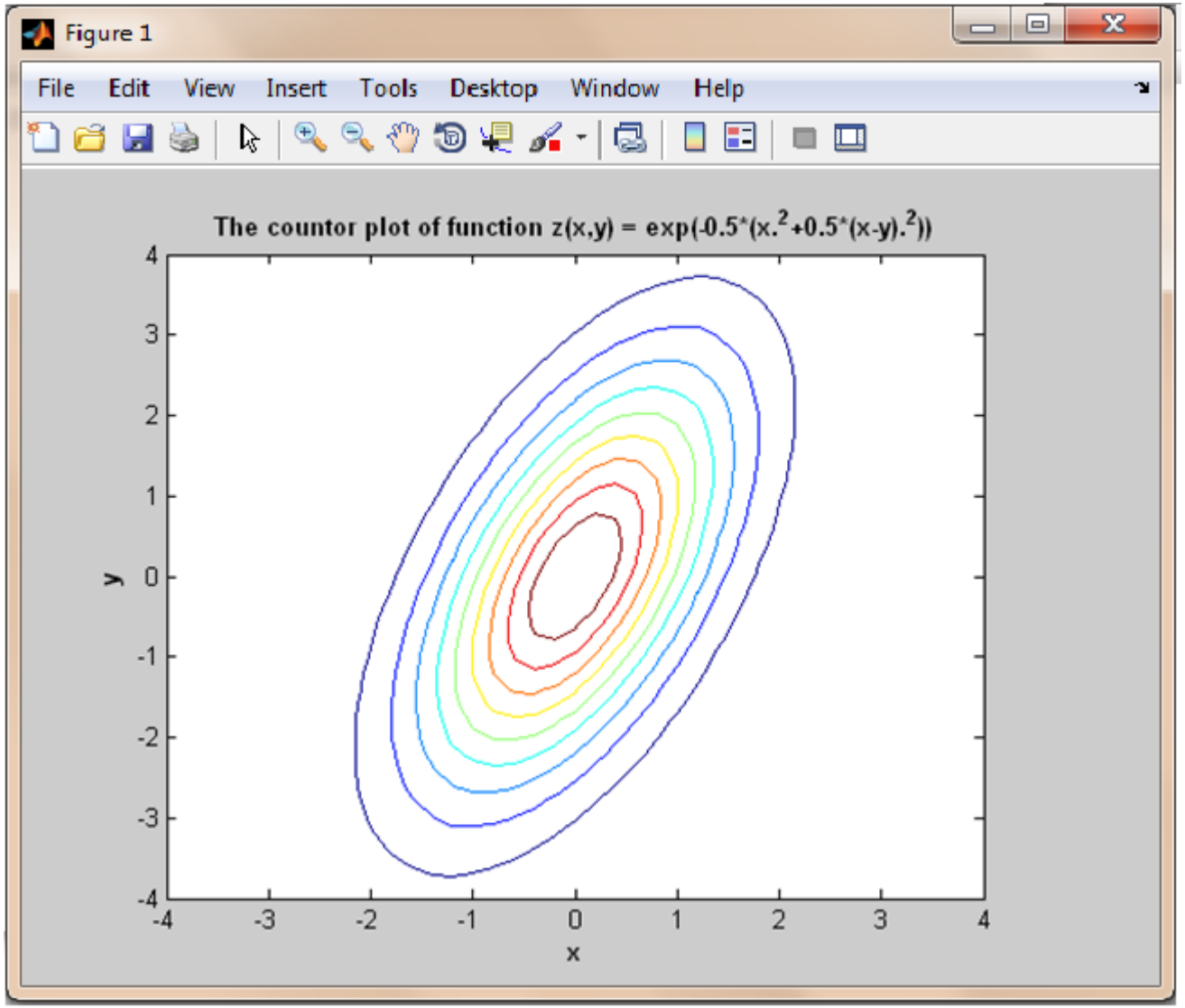
(4) 3-D mesh plotting



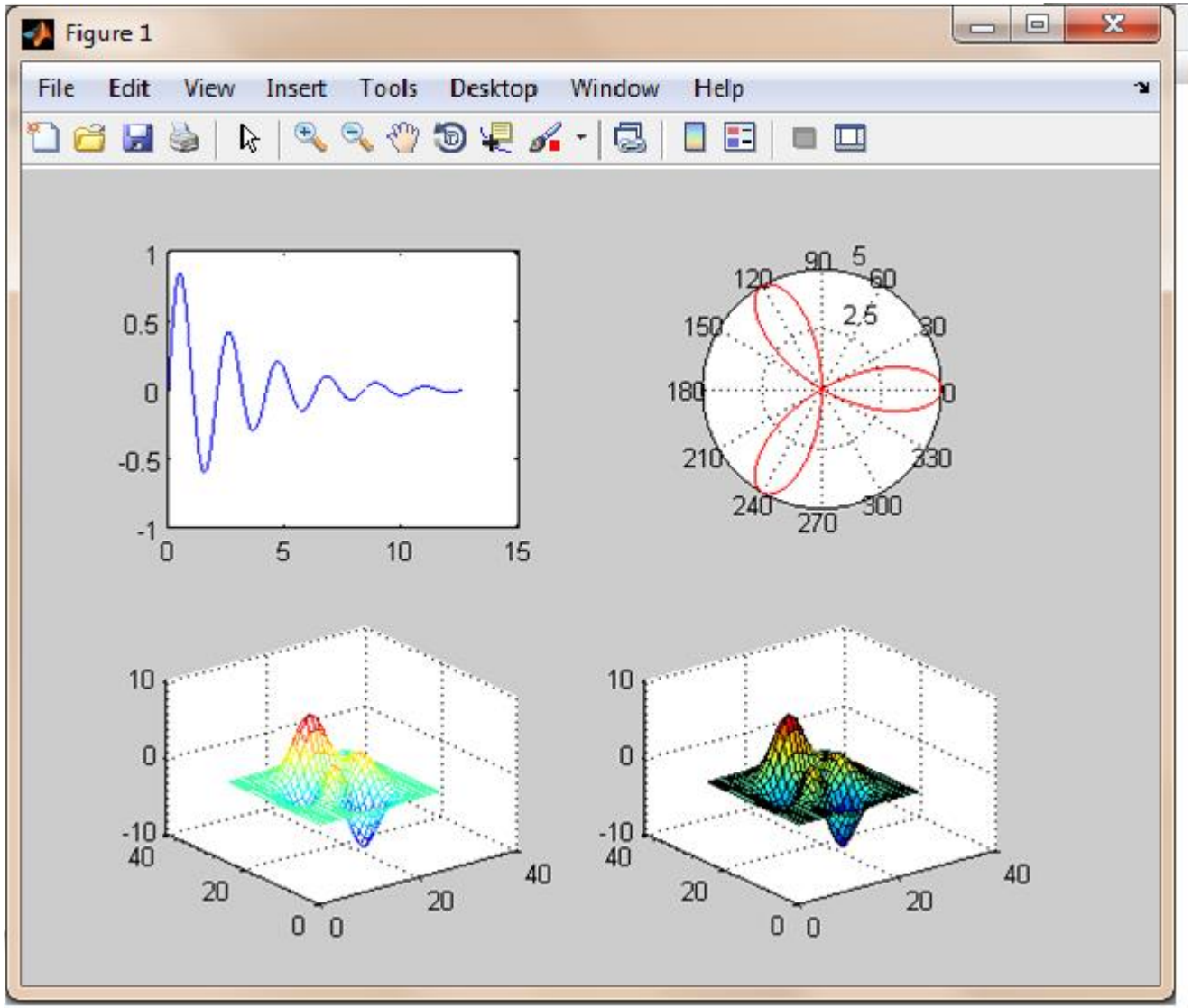
(5) 3-D surface plotting



(6) Contour plotting



(7)Multi-plotting in One window



2.2.4. Device-Independent Plotting

- MATLAB has many integral plotting and imaging commands. The plots and images can be displayed on any graphical output device supported by the computer on which MATLAB is running. It makes MATLAB an outstanding tool for visualizing technical data.

- See Examples.

Example 3. Plots

- MATLAB's plot function makes plotting very easy.

(1) 2-D x-y curve plots

(2) 2-D Polar plots

(3) 3-D curve plots

(4) 3-D mesh plots

(5) 3-D surface plots

(6) Contour plots

(7) Multi-plotting in One window

2.2.5 Graphical User Interface

- MATLAB includes tools that allow a programmer to interactively construct a GUI (like Visual BASIC) for his/her program.
- GUI makes your applications easy to operate.

2.2.6 MATLAB Compiler

- Unfortunately, the program written in MATLAB can sometimes execute slowly because the MATLAB code is interpreted.

- A separate MATLAB compiler is available. This compiler can compile a MATLAB program into a true executable program that runs faster than the interpreted code, and it is suitable for sale and distribution to users.

2.3 Disadvantages of MATLAB

- *MATLAB has two principal disadvantages.*

1. The first is that it is an interpreted language, and therefore can be execute more slowly than compiled language. This problem can be solved by using MATLAB compiler to compile the final MATLAB program before general use.
2. The second is cost: A full copy of MATLAB is very expensive than a conventional C compiler. There is also an inexpensive Student Edition of MATLAB.

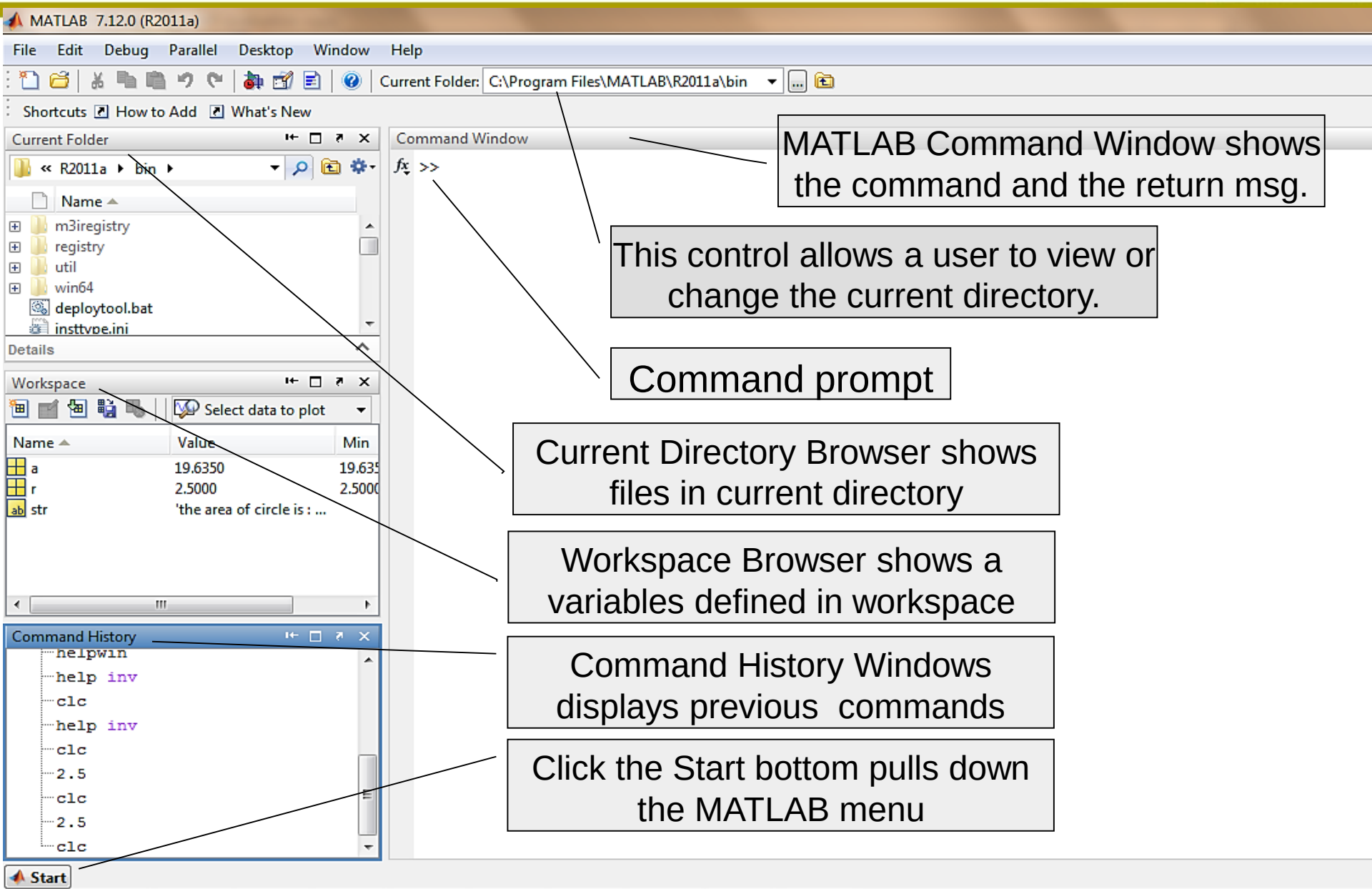
2.4 Getting start

- Click “Start”→ “Programs” → “MATLAB” or double click MATLAB icon  on PC desktop to start the MATALB.

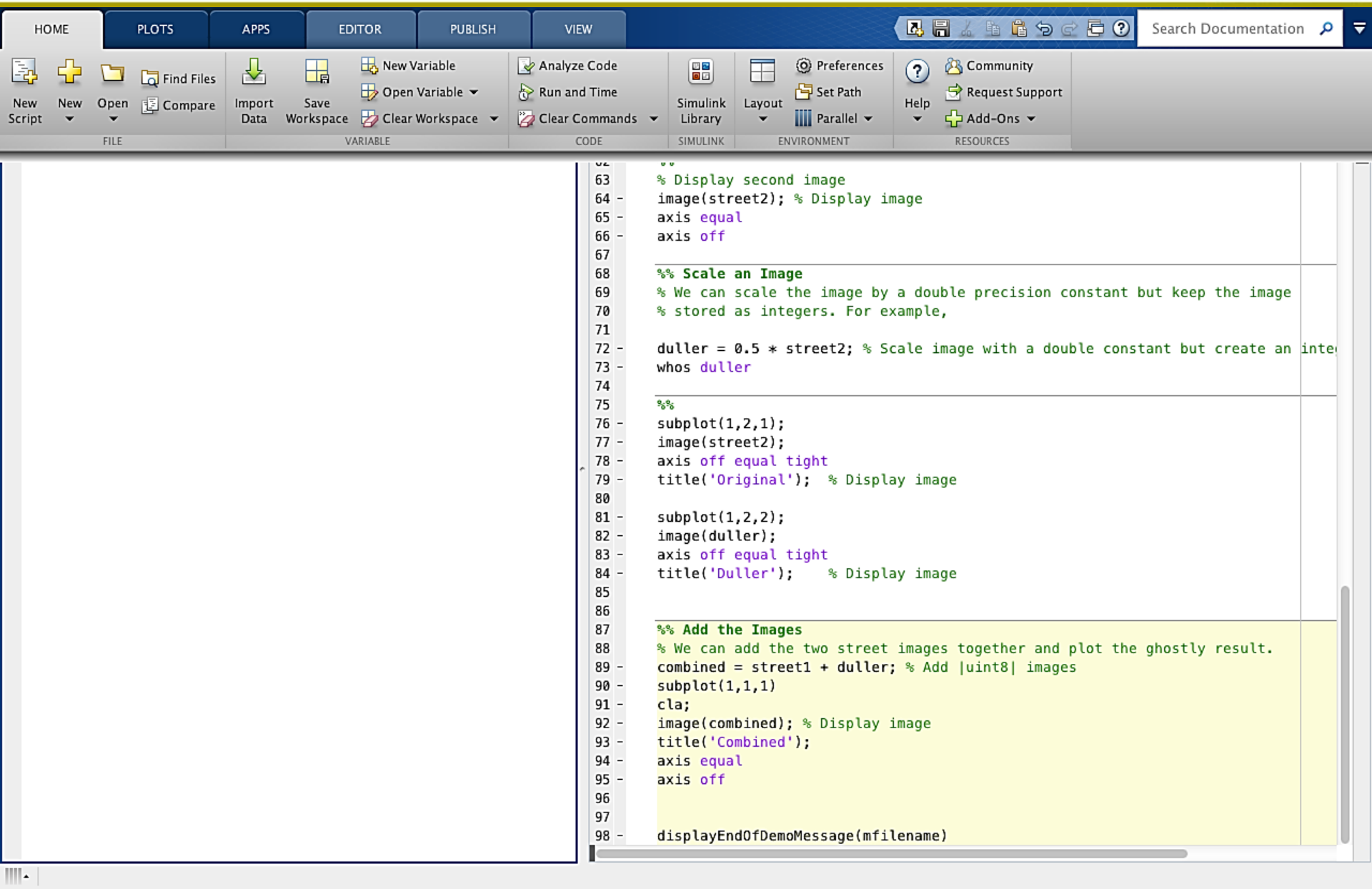
- Click “File” → “Exit MATLAB” can quit the MATLAB.

- When MATLAB is started, The following MATLAB desktop appears.

2.4.1 MATLAB desktop@2009



2.4.1 MATLAB desktop@2015



2.4.2 MATLAB command windows

- You can enter interactive command at the command prompt (>>)in the MATLAB command window, and the command will be executed on the spot. Suppose that we want to calculate the area of a circle with radius 2.5 cm. This can be done in the command window by typing

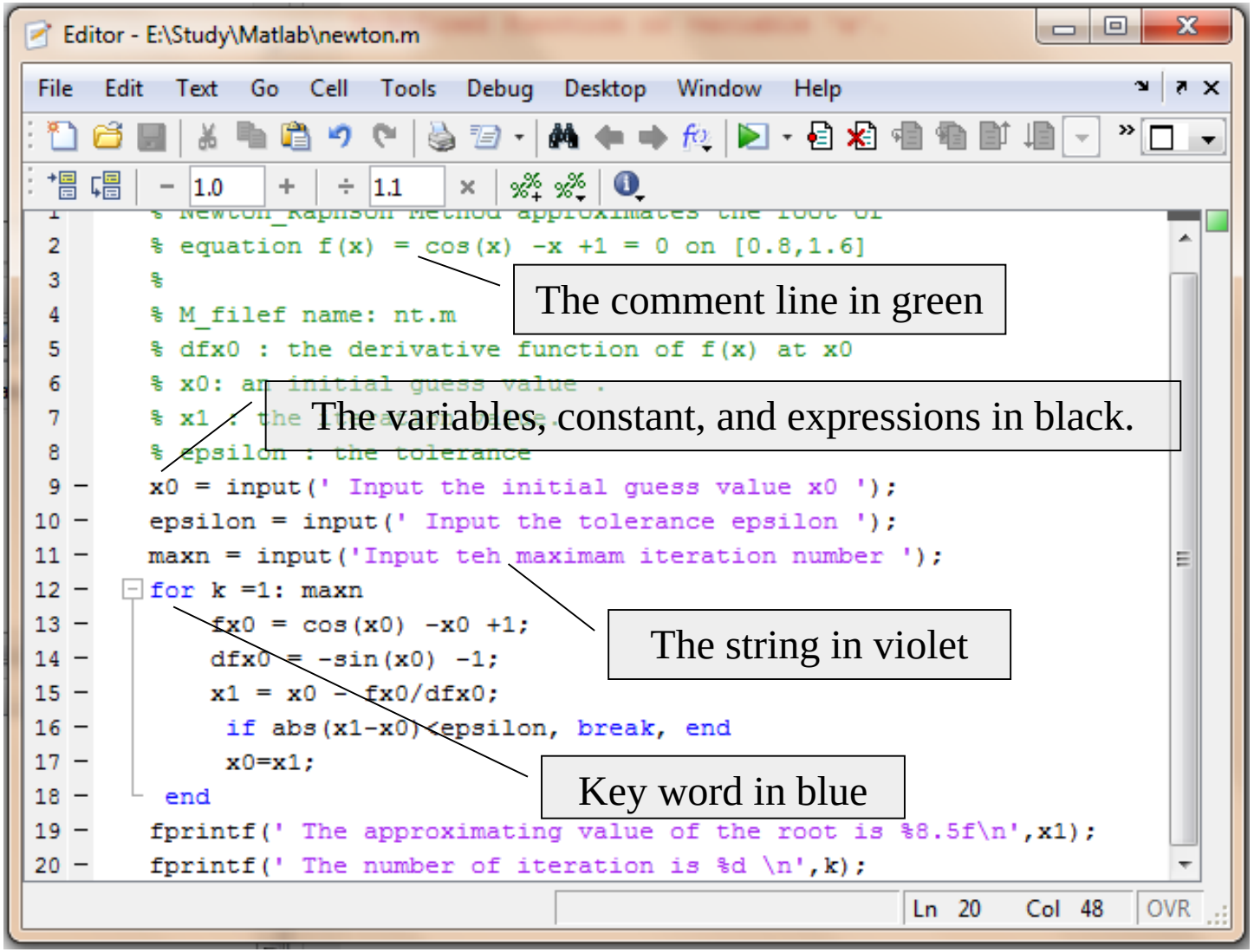
```
>> area = pi*2.5^2 <Enter>
    area =
      19.6350
```

2.4.3 The Edit/Debug Window

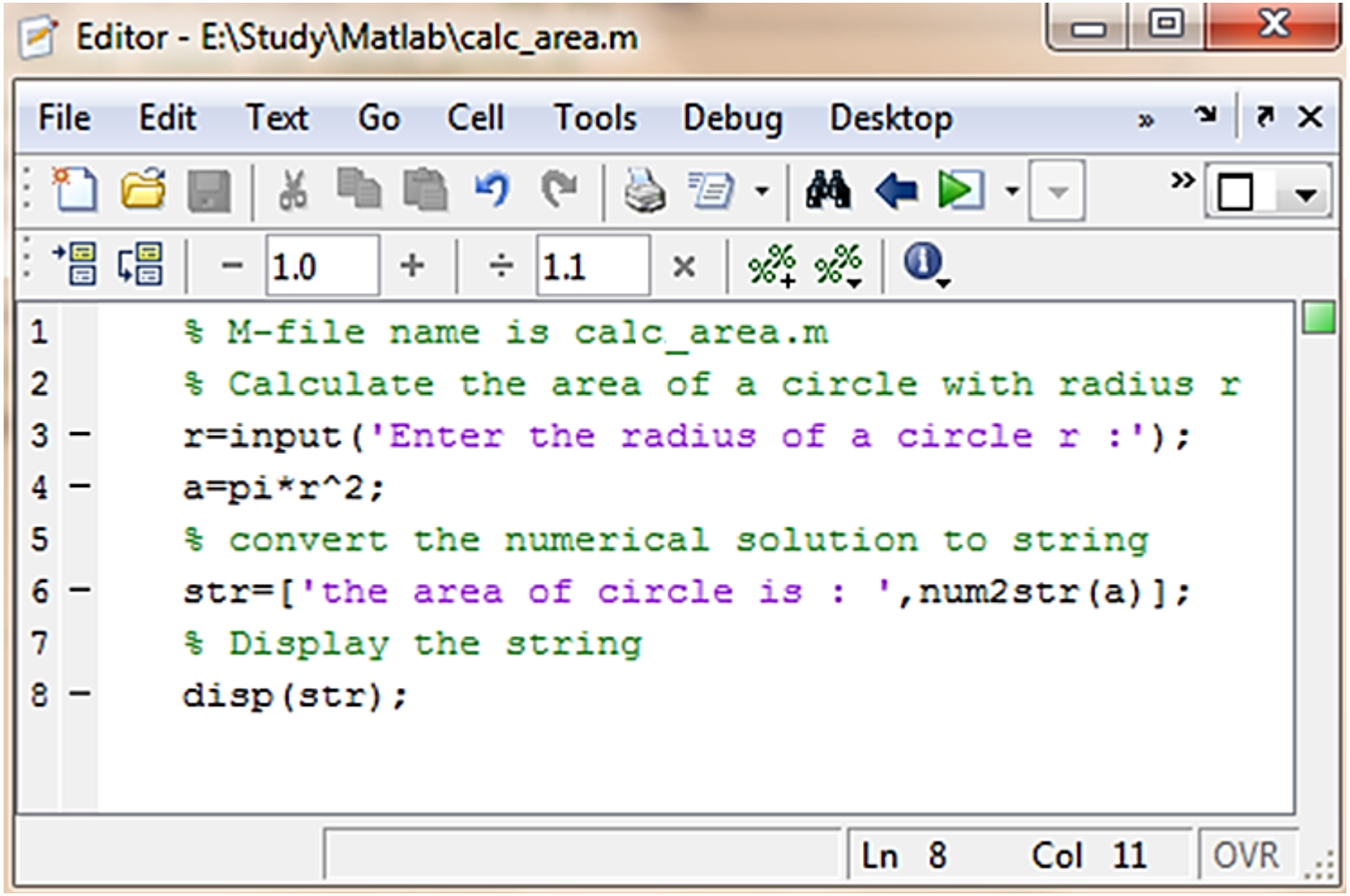
- The Edit/Debug Window is used to create new M-file, or modify old one. It is essentially a programming text editor, with the MATLAB languages features highlighted in different colors.

- Click “File/New/M-file” selection can open Edit/debug window and create new M-file.

The Edit/Debug Window

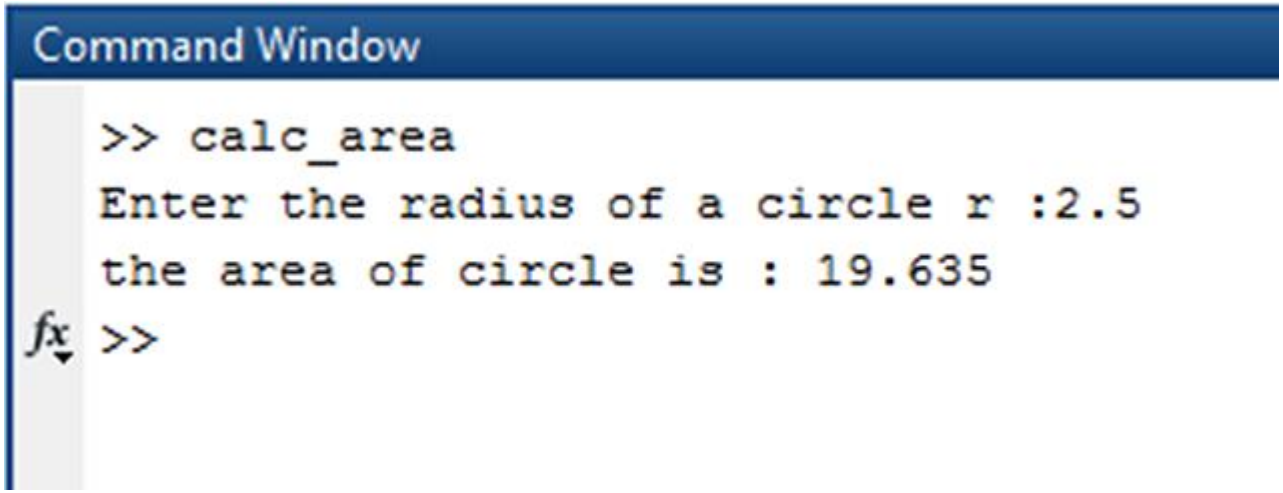


The calc_area.m file



Executing of a M-file

- After M-file is created and saved, it can be executed by simply typing the M-file name (here is calc_area) in the command window. The result is shown below.



2.4.4 Other windows of MATLAB

- Command history window displays previous command. You can select one of them for executing it again by double click it.
- Workspace Browser shows variables defined in workspace. You can edit the data in it.
- Current directory control allows view or change the current directory.

- You can use the menu ‘View’ to set the desktop, and ‘View → Desktop Layout’ command to set the desktop layout.

- You can use the browser of folder button to browser or change the current directory.

2.4.5 Getting Help

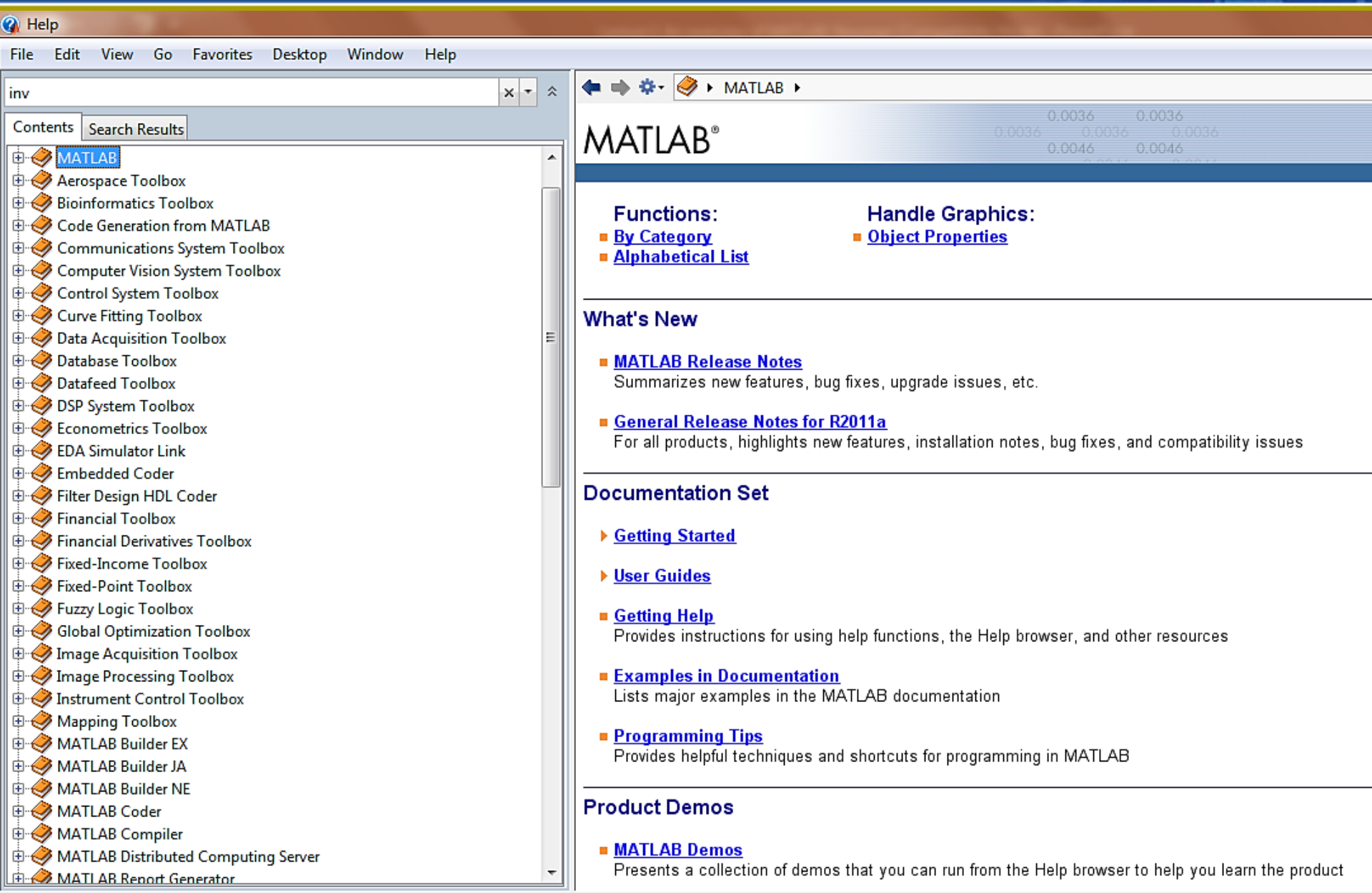
You can get help from the command window in following two ways.

1. By clicking the help icon  or typing *helpwin* in command windows to start the help browser.

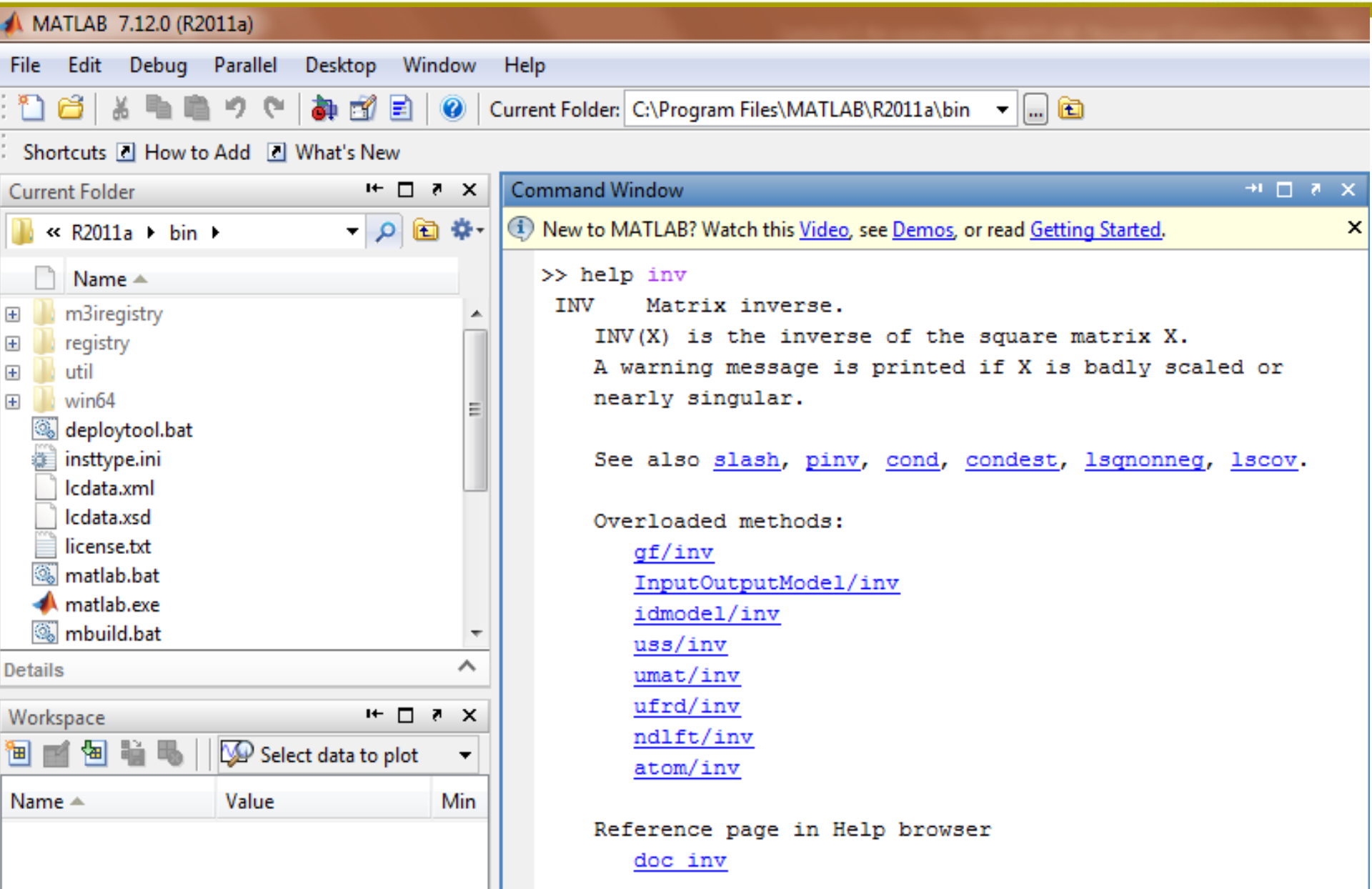
2. By typing *help <function_name>* or doc <function_name> in command line, to show the help document for a given function

For example, command “>> help inv” can get all information about function inv().

The help browser



Command "*help inv*"



Command “doc *inv*”

Help

File Edit View Go Favorites Desktop Window Help

inv

Contents | Search Results

Type | Relevance | Product

inv - Matrix inverse
Y = **inv**(X) returns the inverse of the ... A frequent misuse of **inv** arises when solving the system of ... One way to solve this is with x = **inv**(A)*b ...
MATLAB

inv - Compute symbolic matrix inverse
R = **inv**(A) returns inverse of the symbolic ... A = sym([2,-1,0;-1,2,-1;0,-1,2]); **inv**(A) ... syms a b c d A = [a b; c d]; **inv**(A) ...
Symbolic Math Toolbox

inv - Invert LTI systems
inv inverts the input/output relation ... **inv** handles both continuous- and ... H = [1 tf(1,[1 1]);0 1] Hi = **inv**(H) ... Do not use **inv** to model feedback
Control System Toolbox

lscov - Least-squares solution in presence of known covariance
V, that is, x minimizes (b - A*x)'*inv(V)*(b - A*x) ... x = inv(A'*inv(V)*A)*A'*inv(V)*B ... mse = B'*(inv(V) -
MATLAB

cond - Condition number with respect to inversion
norm(X,p) * norm(**inv**(X),p)
MATLAB

lu - LU matrix factorization
step in obtaining the inverse with **inv** and the determinant with det ... inverse of the example matrix, X = **inv**(A), is actually computed from the
MATLAB

pinv - Moore-Penrose pseudoinverse of matrix
A) is an expensive way to compute **inv**(A) ... or is square and singular, then inv(A) does not exist ... of, but not all, the properties of inv(A)

Search Online Support for inv

fx « Mathematics ▸ Linear Algebra ▸ Linear Equations ▸ inv

inv

Matrix inverse

Syntax

```
Y = inv(X)
```

Description

`Y = inv(X)` returns the inverse of the square matrix `X`. A warning message is printed if `X` is badly scaled or nearly singular.

In practice, it is seldom necessary to form the explicit inverse of a matrix. A frequent misuse of `inv` arises when solving the system of linear equations *Ax* = *b*. One way to solve this is with `x = inv(A)*b`. A better way, from both an execution time and numerical accuracy standpoint, is to use the matrix division operator `x = A\b`. This produces the solution using Gaussian elimination, without forming the inverse. See `mldivide` (\) for further information.

Examples

Here is an example demonstrating the difference between solving a linear system by inverting the matrix with `inv(A)*b` and solving it directly with `A\b`. A random matrix `A` of order 500 is constructed so that its condition number, `cond(A)`, is `1.e10`, and its norm, `norm(A)`, is `1`. The exact solution `x` is a random vector of length 500 and the right-hand side is `b = A*x`. Thus the system of linear equations is badly conditioned, but consistent.

On a 300 MHz, laptop computer the statements

```
n = 500;
```

2.4.6 Using MATLAB

- In its simplest form, MATLAB can be sued as scratchpad to perform mathematical calculations. The Expressions to be typed directly into the command window, using following symbols :

  + Addition - Subtraction

  * Multiplication / Division

  ^ Exponentiation

- See example

Examples of calculation

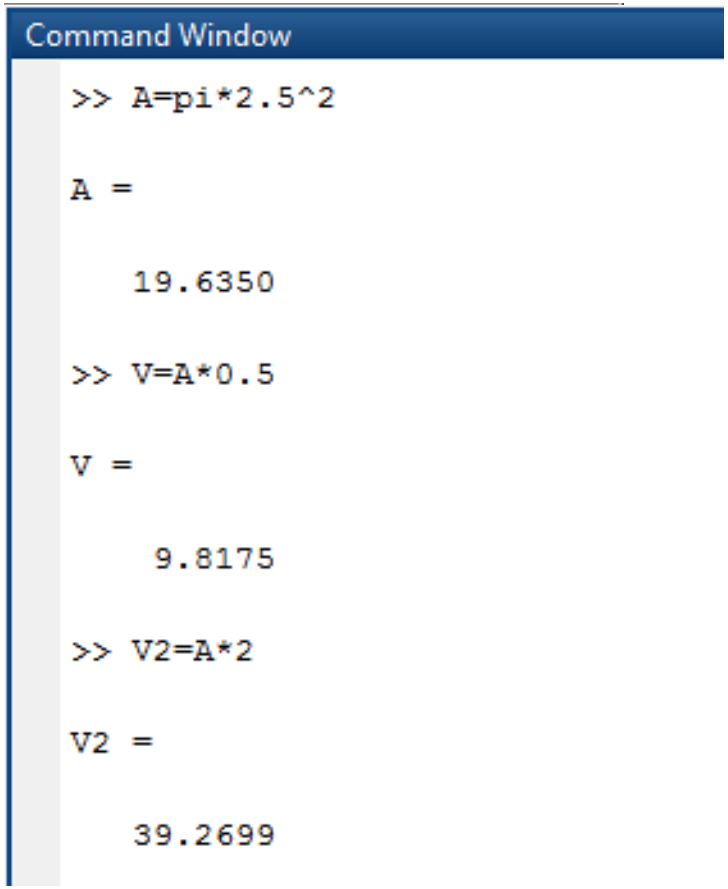
Suppose we want to calculate the volume of a cylinder of radius r and height h. The formula of volume is

$A = \pi r^2$

$V = Ah$

Assume r = 2.5cm h=0.5cm and h2=2cm

Note that pi is predefined to be the value 3.141592.....



The Priority of Arithmetic Operators

Precedence	Operators	Associativity
High ↑ low	( )	Innermost → outward
	^	Left → Right
	* , /	Left → Right
	+,-	Left → Right

2.4.7 Keeping a record

diary filename

- The diary command creates a log of keyboard input and the resulting text output (does not include graphics). The output of diary is an ASCII file, suitable for searching in, printing, inclusion in most reports and other documents.

diary off, on

- diary off suspends the diary.
- diary on resumes diary mode using the current filename.

2.4.8 Saving Work

- **Command *Save <filename>***

  If you enter command save myfile, all the variables in the workspace will be saved to a file called myfile.mat in the current directory. You can also select the variables to be saved by typing them after the filename argument.

- **Command *Load <filename>***

  If you later enter command load myfile, the saved variables are returned to the workspace (overwriting any values with the same names).

2.4.9 Frequently used commands

type : type an M-file (Text file)

who : list current variables

whos : list current variables with memory size

clear : clear current variables and functions from memory

clc : clear command window

cd : change current directory

delete : delete file

dir : directory listing

lookfor : search functions

diary : save text of MATLAB session

save and load

Exercises

1. Suppose that u= 1 and v=3, evaluate the following expressions using MATLAB.

a. $\frac{4u}{3v}$ b. $\frac{4}{3}\pi v^2$ c. $\frac{2v^{-2}}{(u+v)^2}$

d. $\frac{\sqrt{u-3v}}{3v}$ e. $\frac{(u+\cos(v))^2}{v-u}$ f. $\frac{\pi}{3}\sin(0.7\pi)$

2. Get help on the MATLAB function *exp* by typing command “help exp” or “doc exp ” in command window.

Exercises

3. Click “file” → “new” → “M-file”on the desktop menu to create a new M-file, type the following statements into the file, and save with the name try1.m.

```
x = 0:0.1:10.0;
y = 2*exp(-0.2*x);
plot(x, y);
```

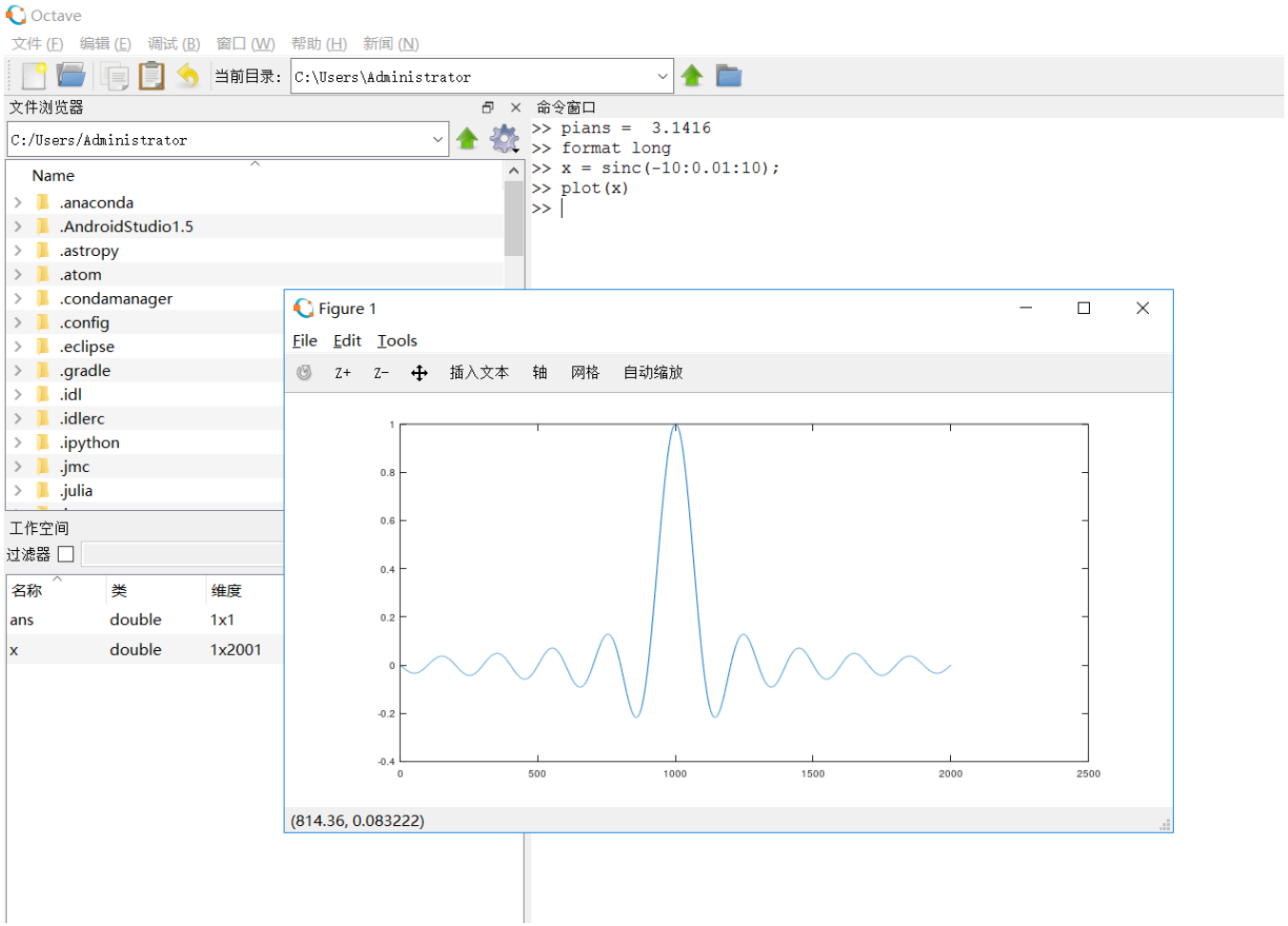
Then execute the program by typing the M-file name try1 in the command window. What result do you get?

4. Try diary command to create a diary file.

Websites

- https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-094-introduction-to-matlab-january-iap-2010/index.htm
- http://www-h.eng.cam.ac.uk/help/tpl/programs/matlab.html
- https://cn.mathworks.com/matlabcentral/fileexchange/?s_tid=gn_mlc_fx
- http://www.cyclismo.org/tutorial/matlab/
- https://www.edx.org/course?search_query=matlab
- ……

Octave


Octave
文件 (F) 编辑 (E) 调试 (B) 窗口 (W) 帮助 (H) 新闻 (N)
当前目录: C:\Users\Administrator
文件浏览器
C:/Users/Administrator
Name
.anaconda
.AndroidStudio1.5
.astropy
.atom
.condamanager
.config
.eclipse
.gradle
.idl
.idlerc
.ipython
.jmc
.julia
工作空间
过滤器
名称 类 维度
ans double 1x1
x double 1x2001
命令窗口
>> pians = 3.1416
>> format long
>> x = sinc(-10:0.01:10);
>> plot(x)
>>
Figure 1
File Edit Tools
Z+ Z- 插入文本 轴 网格 自动缩放
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
500
1000
1500
2000
2500
(814.36, 0.083222)

Thanks