MATH2221A
Mathematics Laboratory II

Lecture 1: Introduction to MATLAB

Gary Choi

January 7, 2025

Teaching team

- Instructor: Prof. CHOI Pui Tung Gary
  - Email: ptchoi@cuhk.edu.hk
  - Office: Lady Shaw Building (LSB) Room 204

- Teaching Assistants:

  - Mr. JIANG Qinghai (LSB 222B, qhjiang@math.cuhk.edu.hk)

  - Ms. WEI Yulin (LSB 222B, ylwei@math.cuhk.edu.hk)

  - Mr. TSANG Hei Tung (Science Center 333B, httsang@math.cuhk.edu.hk)

  - Mr. HUANG Yanwen (Science Center 333B, ywhuang@math.cuhk.edu.hk)

Course logistics (MATH2221A)

- Lectures: Tuesday 11:30am – 12:15pm, LSB 232B
- Lab sessions: Thursday 10:30am – 12:15pm, LSB 232B **(starting from 16/1)**
- Assessment scheme:
  - Lab Assignments 40% (10 Labs, 4% each, to be completed and submitted during each lab session)
  - Test 1 30% (**February 27, 2025**, 90 minutes, during the usual lab session time)
  - Test 2 30% (**April 17, 2025**, 90 minutes, during the usual lab session time)
- Note: You must attend your registered lab session but not the other MATH2221 sessions. Attending the wrong session will result in 0 marks for the assessment!

Course schedule

January

Sun	Mon	Tue	Wed	Thu	Fri	Sat
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	[28]	[29]	[30]	[31]	

February

Sun	Mon	Tue	Wed	Thu	Fri	Sat
						[1]
[2]	[3]	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	1

March

Sun	Mon	Tue	Wed	Thu	Fri	Sat
2	[3]	[4]	[5]	[6]	[7]	[8]
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

April

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17		

Lecture 1- Lecture 13

Lab 1 - Lab 10

Test 1 and Test 2

Course policy

- **Honesty in Academic Work**
  - The Chinese University of Hong Kong places very high importance on honesty in academic work submitted by students, and adopts a policy of zero tolerance on cheating and plagiarism. Any related offence will lead to disciplinary action including termination of studies at the University. Although cases of cheating or plagiarism are rare at the University, everyone should make himself / herself familiar with the content of the following website:
    http://www.cuhk.edu.hk/policy/academichonesty/
    and thereby help avoid any practice that would not be acceptable.

- **Artificial Intelligence (AI) Policy**: Prohibit all use of AI tools

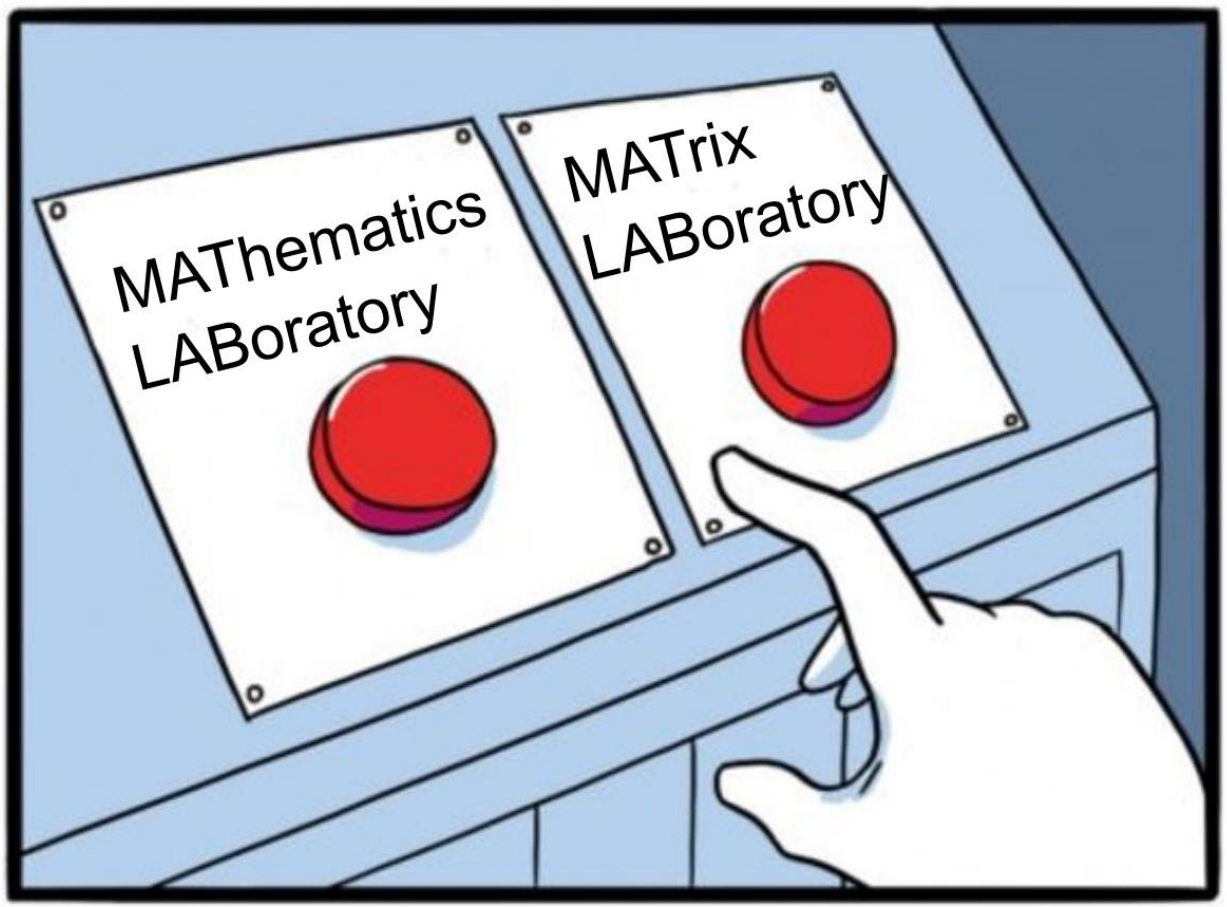
  - This is a first course for you to learn and practice MATLAB
  - AI-generated codes are usually not optimal!

What is MATLAB?

- MATLAB stands for …

What is MATLAB?

- MATLAB stands for …



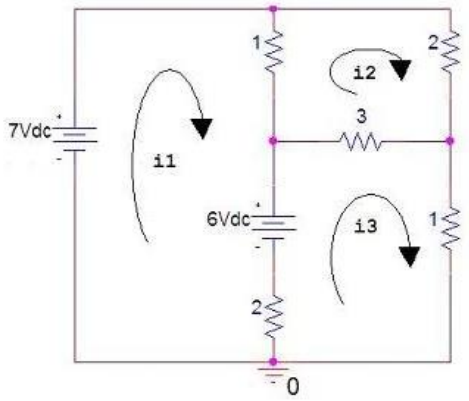
What is MATLAB?

- MATLAB stands for **MATrix LABoratory**

What is MATLAB?

- MATLAB stands for **MATrix LABoratory**
- Why matrix? …

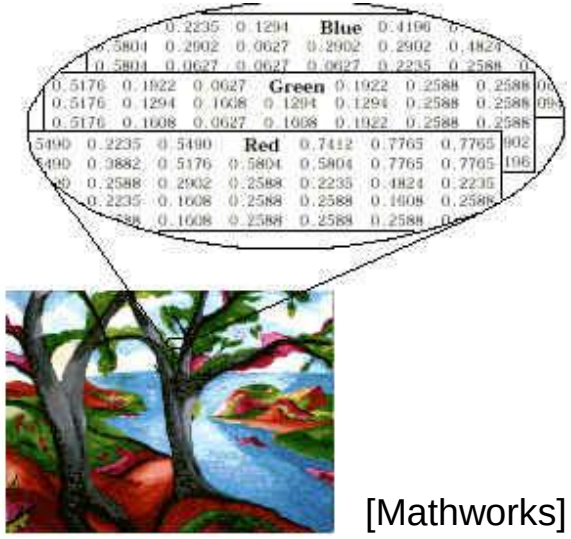
Electric circuits



[matrixlab-examples.com]

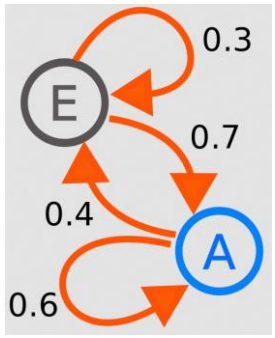
$$\begin{pmatrix} -3 & 1 & 2 \\ 1 & -6 & 3 \\ 2 & 3 & -6 \end{pmatrix} \begin{pmatrix} i_1 \\ i_2 \\ i_3 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ -6 \end{pmatrix}$$

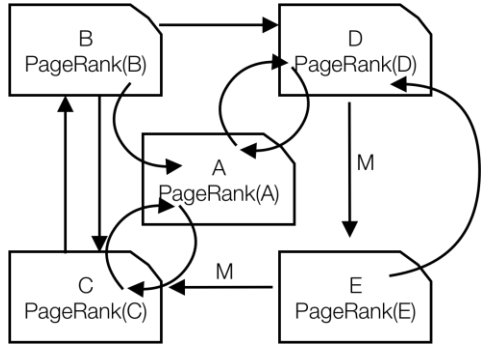
Image processing



[Mathworks]

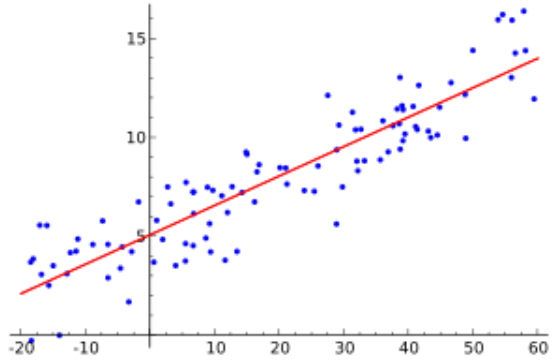
Markov process





[Wikipedia]

Regression analysis

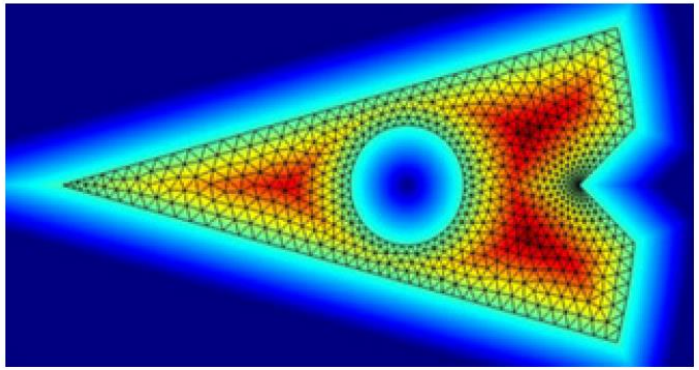


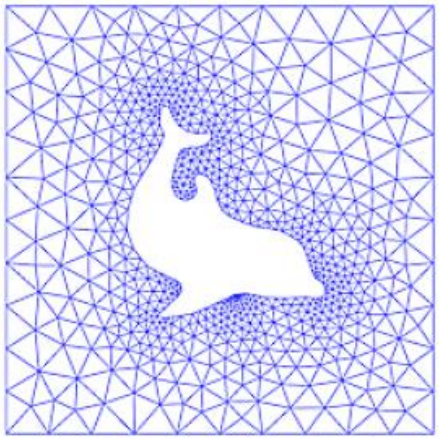
[Wikipedia]

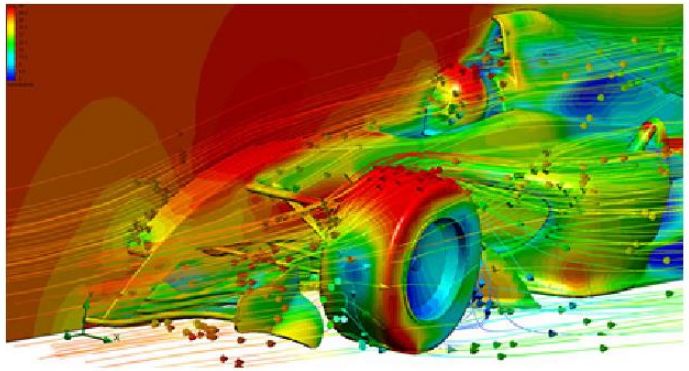
What is MATLAB?

- MATLAB stands for **MATrix LABoratory**
- Why matrix? …

Calculus and Optimization in Engineering







[Mentor Graphics]

Modelling in Biology

[Youtube]

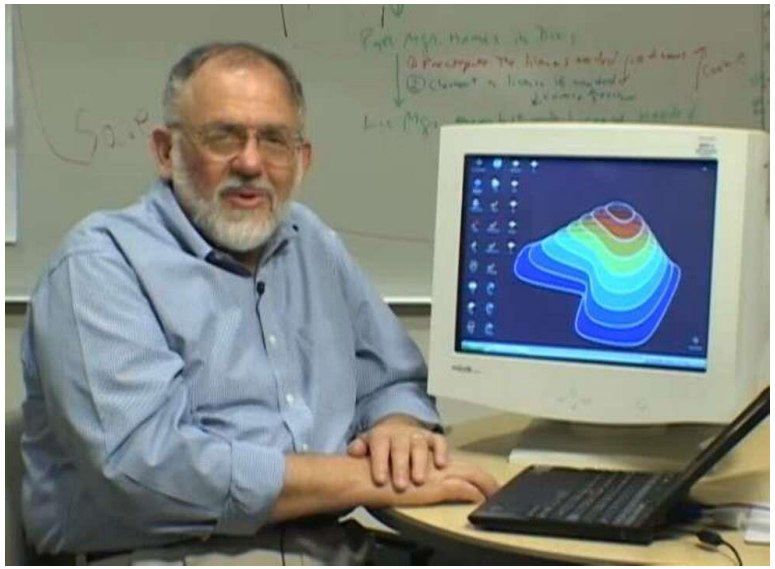
$$\frac{d}{dt}N(t) = \alpha N(t)$$

What is MATLAB?

- MATLAB stands for **MATrix LABoratory**
- Why matrix? …

What is MATLAB?

- MATLAB stands for **MATrix LABoratory**
- Invented by Cleve Moler in late 1970s
- **Calculator x Programming language x Interactive visualization and control tool**
  - Facilitate matrix and vector computations
  - Easy-to-use visualization tools
  - Easier to program than C/C++/Python/…
- MathWorks, Inc established in 1984
  http://www.mathworks.com/
  - MATLAB
  - Simulink



Cleve Moler (1939 -)

MathWorks headquarters
Natick, Massachusetts, USA

MATLAB's main features

MATLAB

Built-in Functions / User-written Functions

Computation

- Linear algebra
- Calculus
- Differential equations
- Optimization
- Data analysis
- Signal/image/video processing

Visualization

- 2D Graphics
- 3D Graphics
- Animation

External Interface (Mex-files)

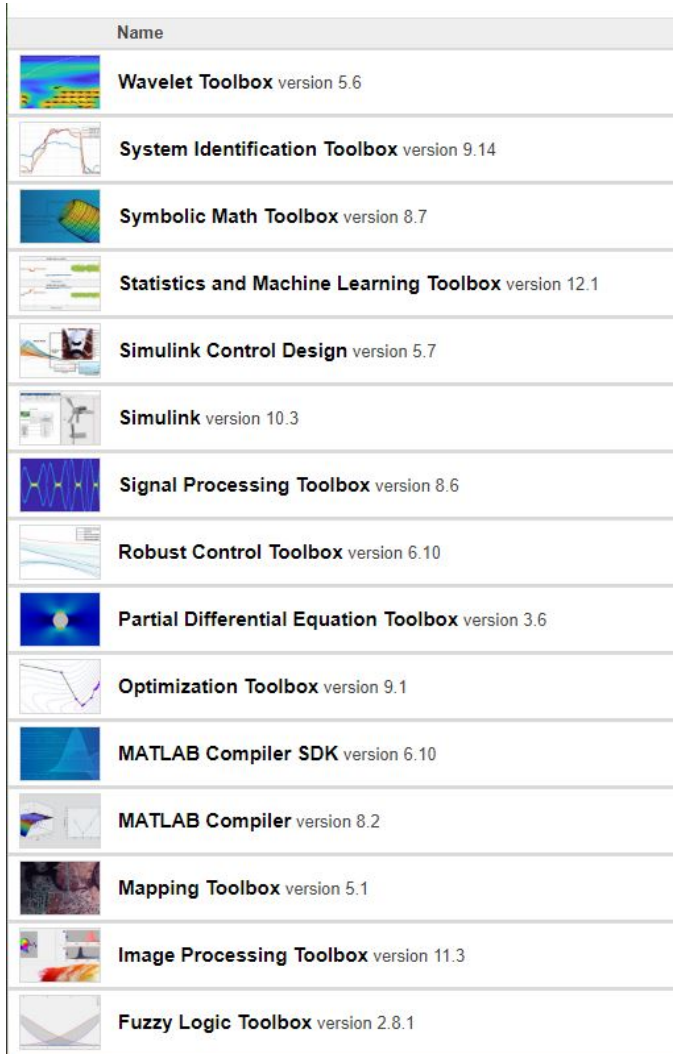
- Interface with C, Java, and Fortran Programs

Toolboxes (Collections of Specialized Functions)

- Optimization
- PDE
- Parallel Computing
- Deep Learning
- Symbolic Math
- Computer Vision
- Radar
- Bioinformatics
- Medical Imaging
- Financial
- Aerospace
- and many more…

Where can I access MATLAB?

- Our computing lab (LSB 232B)
  - MATLAB version R2021a
  - With some toolboxes installed



- CUHK library computers
  https://www.lib.cuhk.edu.hk/en/use/facilities/computer/

- Download and install on your own computers
  - CUHK license
    https://www.itsc.cuhk.edu.hk/all-it/procurement-support/campus-wide-software/matlab-and-simulink/

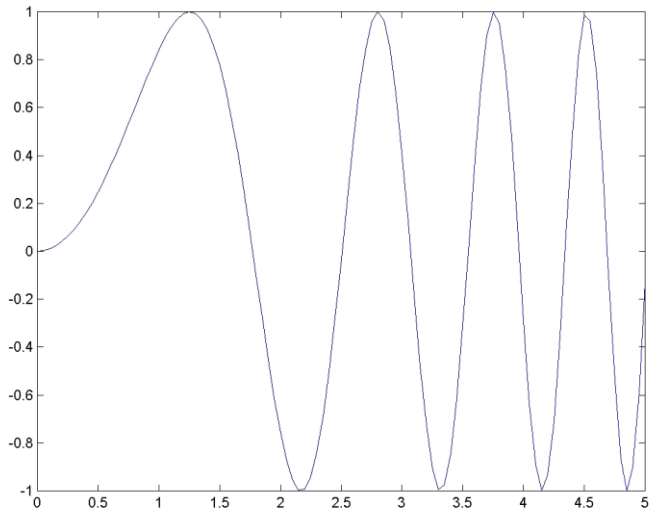
By the end of this course, you will be able to use MATLAB to…

- Perform basic computations
  - Scalar
  - Vector
  - Matrix
  - Commonly used built-in functions
  - Loops and Conditional Statements (if, for, while, …)

- Write your own MATLAB programs with different input and output

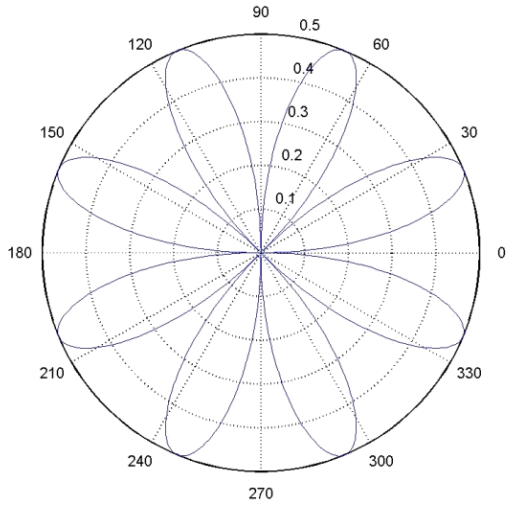
By the end of this course, you will be able to use MATLAB to…

- Create 2D and 3D visualizations



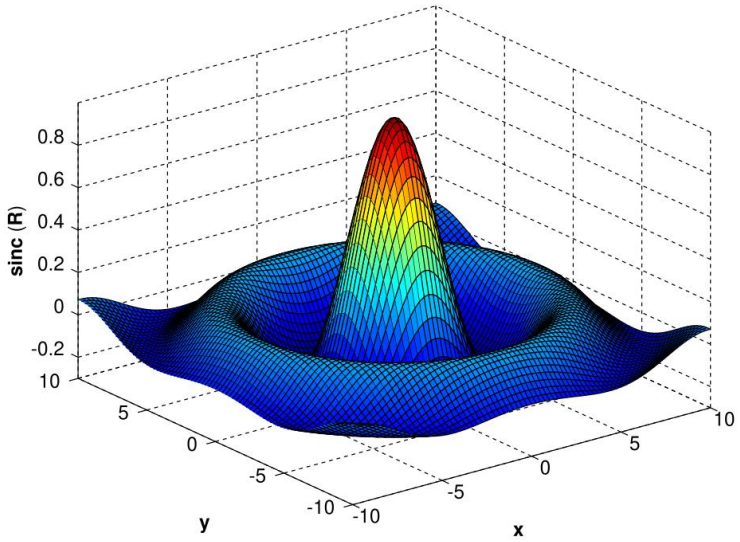
Line plot:

```
x=0:0.05:5;
y=sin(x.^2);
plot(x,y);
```



Polar plot:

```
t=0:.01:2*pi;
polar(t,abs(sin(2*t).*cos(2*t)));
```

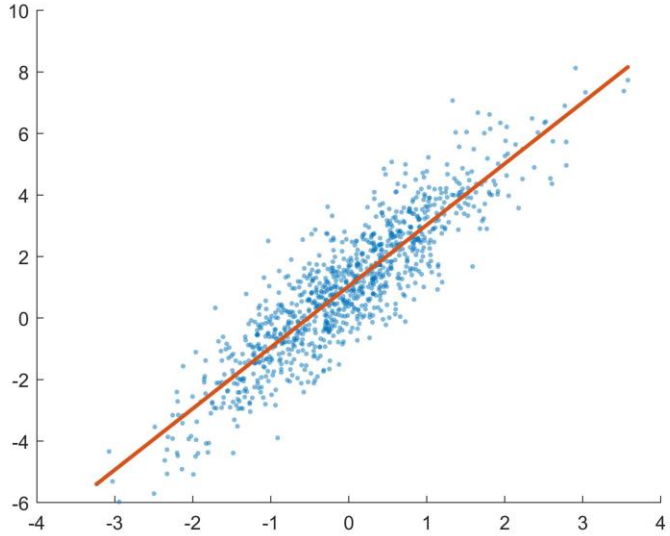


Surface plot:

```
[X,Y] = meshgrid(-10:0.25:10,-10:0.25:10);
f = sinc(sqrt((X/pi).^2+(Y/pi).^2));
surf(X,Y,f);
```

By the end of this course, you will be able to use MATLAB to…

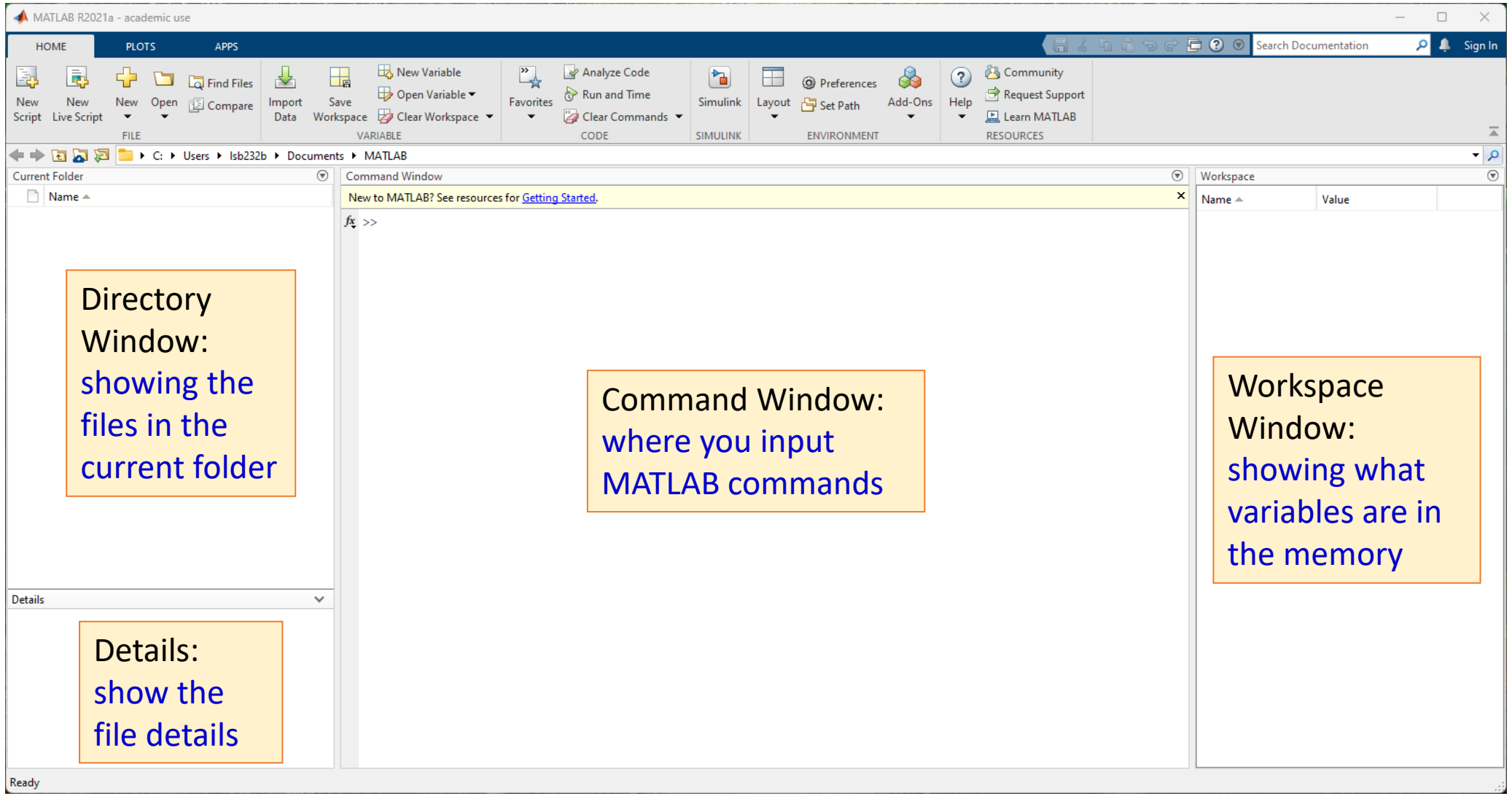
- Perform data analysis
  - Reading numerical data or text data from txt/xlsx/csv/…
  - Analyzing them in MATLAB
  - Saving the results



- Perform other special tasks using different toolboxes
  - Image processing
  - Optimization
  - …

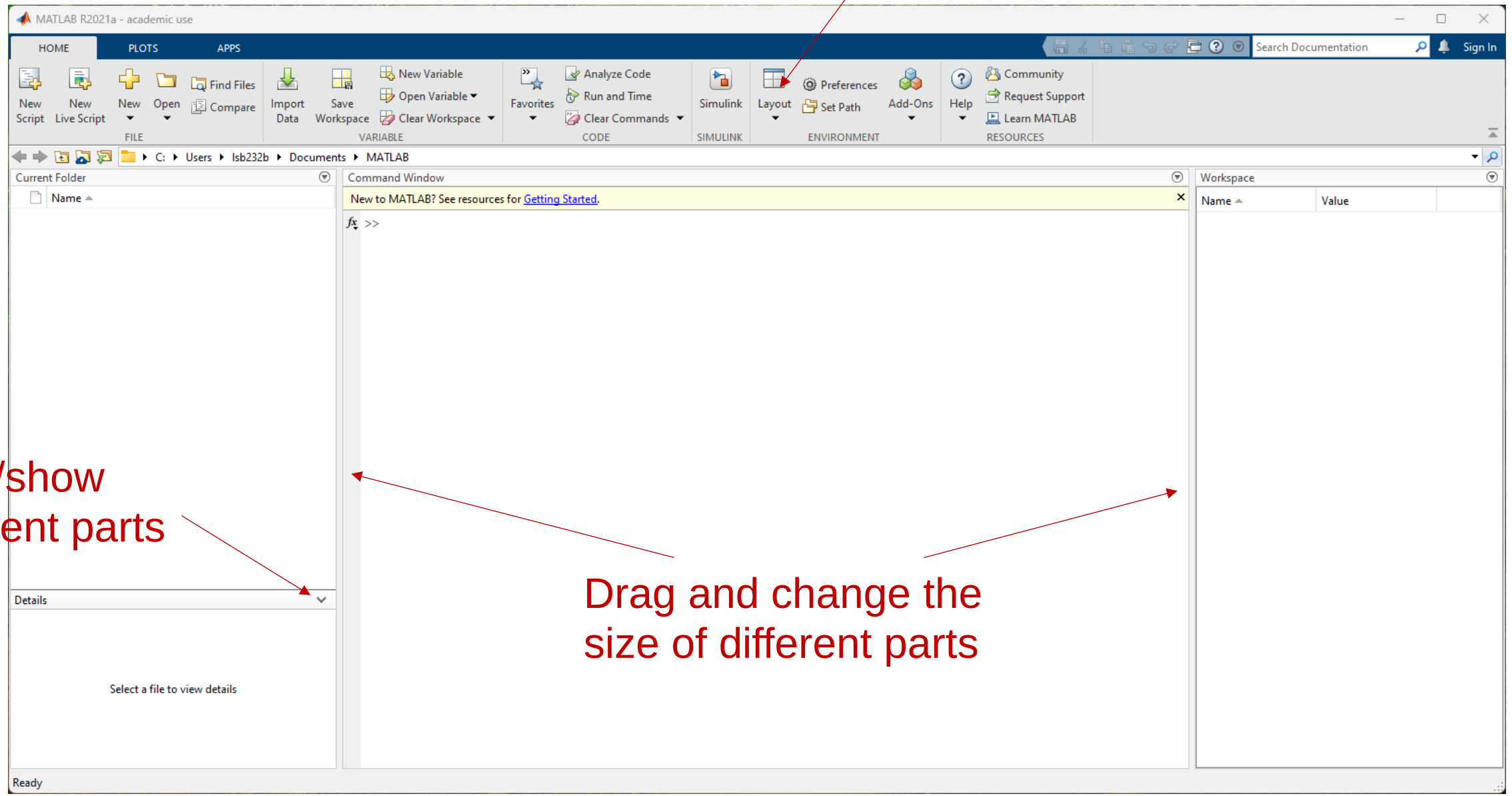
MATLAB software interface

- To start MATLAB: Start → MATLAB R2021a → MATLAB R2021a
  (or look for the MATLAB shortcut on the desktop)



MATLAB software interface

- Note: The interface can be customized!

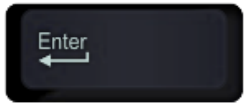


Basic arithmetic operations in MATLAB

```
>>
>>  2*(2 + 2) – 4/2
```

MATLAB prompt

Enter `2*(2 + 2) – 4/2` and hit the return/enter key




```
ans =

     6
```

Note that the result of an unassigned expression is saved in the default variable `ans`

Use the arithmetic operations

- Addition `+`
- Subtraction `-`
- Multiplication `*`
- Division `/`

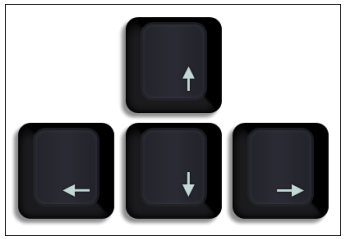
```
>>  x = 2*(2 + 2) – 4/2
```

You can also assign the value of an expression `2*(2 + 2) – 4/2` to a variable `x`

```
x =

     6
```

Hit the up-arrow key. What do you see?



Basic arithmetic operations in MATLAB

```
>>  radius = 2.5;
```

Assign a value to a variable
A semicolon ; at the end suppresses screen output

```
>>  radius

radius =

    2.5000
```

MATLAB remembers radius, though.
You can recall the value radius by simply typing radius

```
>> A = pi*radius^2

A =

   19.6350
```

Use the arithmetic operation:

- Exponentiation ^

Assign the values of an expression pi*radius^2 to a variable A

MATLAB knows the constant value π
(try: >> pi)

Basic arithmetic operations in MATLAB

```
>>  clear A
```
Clear the variable A in the memory

```
>>  A
Unrecognized function or variable 'A'.
```
Now MATLAB does not know what A is

← Giving an error message

(You will get familiar with it very soon!)

```
>>  a = 3;
```
Assign the value 3 to a

```
>>  b = a;
```
Assign the value a (i.e., 3) to b

```
>>  a = 2;
```
Assign the value 2 to a

```
>> a
a =
     2.0000
```

(note: changing the value of a will NOT change the value of b)

```
>> b
b =
     3.0000
```

Basic arithmetic operations in MATLAB

```
>>  pi
pi =
      3.1416
```

The default value of pi

```
>> pi = 3;
>> pi
pi =
      3.0000
```

(Caution) If you assign a value to some special variables (e.g. pi), it will replace the default value of the variables

```
>> pi*radius^2
ans =
      18.7500
```

Now the answer is different from the previous one. Hard to debug!
Be mindful when choosing what variables to use.

```
>> clear pi
>>  pi
pi =
      3.1416
```

Clear the variable pi in the memory.
The default value of pi will be restored.

Variable naming rule in MATLAB

- A valid variable name starts with a letter, followed by letters, digits, or underscores:
  - a ✓
  - b ✓
  - AbC12 ✓
  - Edge_length_1 ✓
  - aa, aaa, aaaa … ✓ (but not recommended, hard to understand)
  - pi ✓ (but not recommended, will replace the default value of pi)
  - 12a ✗
- Case sensitive
  - A and a are treated as different variables
- See https://www.mathworks.com/help/matlab/matlab_prog/variable-names.html

Matrix operations in MATLAB

```
>>  v = [1 3 -1]
v =
     1    3   -1
```

Define a row vector (using space to separate the entries)

```
>> w = [2, 3, 4, 6]
w =
     2    3    4    6
```

One can also use comma “,” or space “ ” to separate the entries

```
>> u = [1; 3; 5]
u =
     1
     3
     5
```

For column vector, use semicolon “;” to separate them

```
>> u(2)
ans =
     3
```

Extract the second entry in the vector

Matrix operations in MATLAB

```
>>  A = [1, 2, 3; 4, 5, 6]
A =
     1    2    3
     4    5    6
```

Define a matrix

- Use comma “,” or space “ ” to separate the entries for each row
- Use semicolon to separate the rows

```
>> A(2,1)
ans =
     4
```

Extract the entry in the 2nd row and 1st column of A

```
>> c = A(2,3);
>> c
c =
     6
```

One can also store the extracted entry as a new variable

Matrix operations in MATLAB

```
>>  M = [1, 2, 3; 4, 5, 6];
>>  u = [1; 3; 5];
>>  v = [1 3 -1];

>> M+1.5
ans =

   2.5000   3.5000   4.5000
   5.5000   6.5000   7.5000
```

Adding a scalar value to all entries of M

```
>> 3*v
ans =

     3    9   -3
```

Scalar multiplication of v

Matrix operations in MATLAB

```
>>  M = [1, 2, 3; 4, 5, 6];
>>  u = [1; 3; 5];
>>  v = [1 3 -1];

>> M*u
ans =
      22
      49
```

Matrix multiplication

```
>> M*v
```

Matrix multiplication with incompatible dimension

Getting an error message!

Error using *

Incorrect dimensions for matrix multiplication. Check that the number of columns in the first matrix matches the number of rows in the second matrix. To perform elementwise multiplication, use '.*'.

Related documentation

Matrix operations in MATLAB

```
>>  A = [1, 2, 3; 4, 5, 6];
>>  v = [1 3 -1];
>>  transpose(v)
ans =
        1
        3
       -1
```

Transpose of v

```
>> v’
ans =
        1
        3
       -1
```

Alternative method (use apostrophe “ ’ ”)

```
>> A*v’
ans =
        4
        13
```

Matrix multiplication with compatible dimension

Matrix operations in MATLAB

```
>>  M = [1, 2, 3; 4, 5, 6];
>>  v = [1 3 -1];
>>  [M; v]
ans =
    1    2    3
    4    5    6
    1    3   -1
```

Forming a new matrix consisting of M followed by v in a new row

```
>> [M(2,2), v]
ans =

    5    1    3   -1
```

Forming a new matrix consisting of the entry in M(2,2) followed by v in the same row

```
>> [M, v]
Error using horzcat
Dimensions of arrays being concatenated are not consistent.
```

Incompatible dimension, getting an error message

Commands for creating vectors efficiently

```
>>  1:5
ans =
    1    2    3    4    5
```

Use colon “m:n” (where n > m) to create a row vector with consecutive entries from m to n

```
>> 3:0.2:4
ans =
   3.0000    3.2000    3.4000    3.6000    3.8000    4.0000
```

Use colon “m:s:n” to create a row vector with entries from m to n with an increment of s

```
>> a = 5:-1:-3
a =
    5    4    3    2    1    0   -1   -2   -3
```

Negative increment can also be used

```
>> 0:pi:10
ans =
         0    3.1416    6.2832    9.4248
```

Values outside the interval will not be included

Commands for creating matrices efficiently

```
>>  M = zeros(2,4)
M =
     0     0     0     0
     0     0     0     0
```

Creating a zero matrix with 2 rows and 4 columns

```
>> A = ones(3,2)
A =
     1     1
     1     1
     1     1
```

Forming a matrix with 3 rows and 2 columns, with all values = 1

```
>> B = ones(2)
B =
     1     1
     1     1
```

Will assume the matrix is a square matrix if only one input is given

Learning resources

- The best way to learn MATLAB is by using it!
  - Type demo in the command window to access the MATLAB Help Center (search and look for certain functions and examples)
  - Type help xxx (where xxx is the function you want to know more about) to read the documentation
- Lecture Notes
- MATLAB Academy: https://matlabacademy.mathworks.com/
- MATLAB Central: https://www.mathworks.com/matlabcentral/

Thank you!

Next time:

- More basic functions
- Writing MATLAB scripts