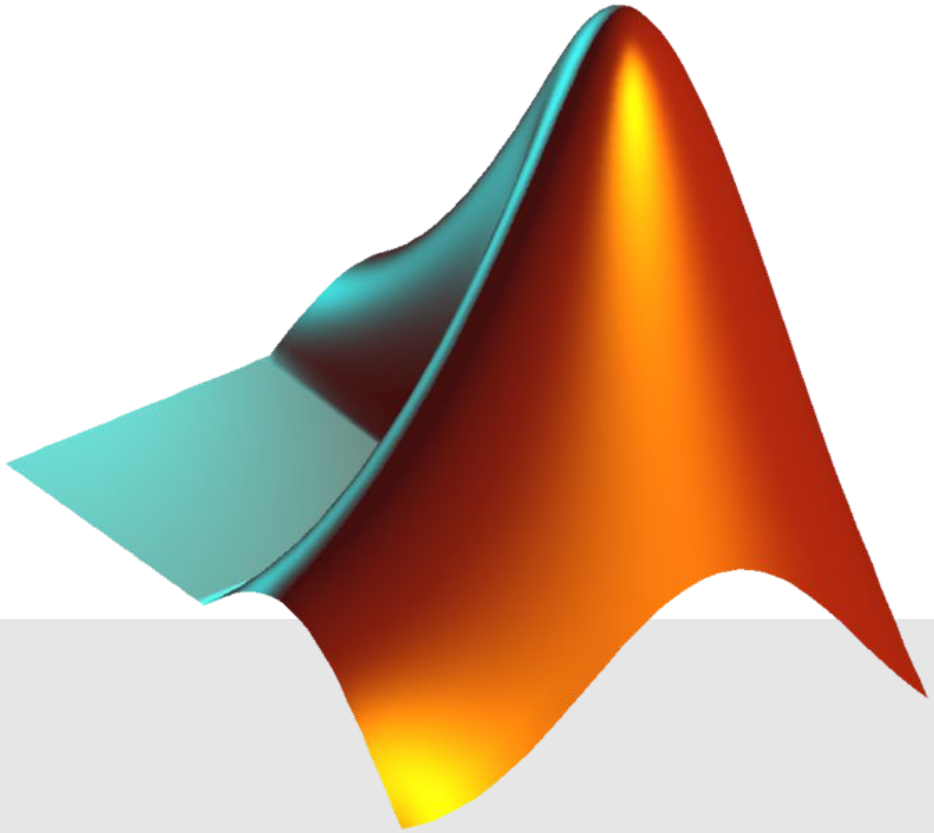
MATLAB Workshop

12/03/21



What is MATLAB?

- High-level language **dedicated to mathematical and technical computing**
- **Mat**rix **Lab**oratory
- Allows for
  - Matrix manipulations
  - Plotting functions and data
  - Implementing algorithms
  - Creation user interfaces,
  - Interfacing with programs written in other languages

MATLAB vs Python

	Python	MATLAB
Definition	High level general purpose programming language	High performance language for technical computing
Primary usage	Developing websites and software, task automation, data analysis, and data visualization	Signal processing & communications, image processing, control systems, computational biology
Advantages	• Free and open • General purpose • Introspection • Portable • Popular	• Matrices • Lots of functions • Simulink • Easy to install • Widespread in academia

/ ▸ Users ▸ daniellegruber ▸ Documents ▸ MATLAB ▸ IEEE-MATLAB-Tutorial ▸

Current Folder

Name | Git

- Helper Scripts
- Beginner Tutorials
  - Essentials.mlx
  - Plotting.mlx
- Intermediate Tuto...
  - Lab 4 — Diode...
  - HW7 — Load Li...
    - Solution
    - load_line_pl...
    - load_lines.m
    - Load Line Pl...
    - Load Line Pl...
    - Load Line Pl...
  - spontaneous_fl...
  - CoolExtras.mlx

load_lines.m (Script)

Setup

- Setup
- Figure settings
- Generate plots
- Current functions
- diode_current(V_D)
- resistor_current(V_D, V_...

Editor - /Users/daniellegruber/Documents/MATLAB/IEEE-MATLAB-Tut...

load_lines.m | load_line_plot_tutorial.m

```
1   %% Setup
2 - addpath(genpath('Helper Scripts'))
3 - addpath(genpath('General Code'))
4 - save_dir = 'Intermediate Tutorials/HW7 - Load Lines';
5
6   %% Figure settings
7 - set(groot,'defaulttextinterpreter','latex');
8 - set(groot,'defaultLegendInterpreter','latex');
9 - set(groot,'defaultAxesTickLabelInterpreter','latex');
10
11  %% Generate plots
12
13  % Part a
14 - V_0 = 2;
15 - R = 400;
16 - V_D = linspace(0,V_0 * 1.05,4000);
17 - I_D = diode_current(V_D);
18 - I_R = resistor_current(V_D, V_0, R);
19 - load_line_plot(V_D, I_D, I_R, 'Part a', save_dir)
20
21  % Part b
```

Command Window

```
>> load_lines
>> close all
>> disp('Hello world!')
Hello world!
>>
```

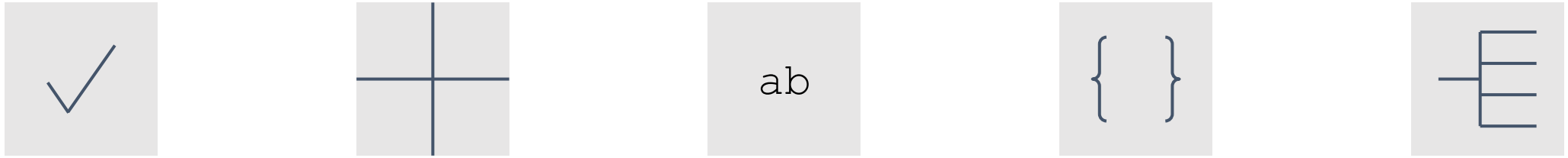
Workspace

Name	Value
ans	'/Users/danielle
I_D	*1x4000 double*
I_R	*1x4000 double*
R	200
save_dir	'Intermediate Tu
V_0	1
V_D	*1x4000 double*

Basic data and variables

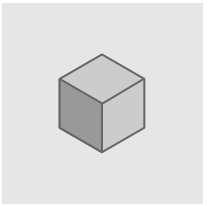
multidimensional array

`logical` `numeric` `char` `cell` `struct`



`int8` `uint8` `single` `double`
`int16` `uint16`
`int32` `uint32`
`int64` `uint64`

other



`function_handle` `sqr = @(n) n.^2`

`sym` $f = x + y$

Arithmetic operators

Operation	Operator	Example
Addition	+	`3 + 2`
Subtraction	–	`3 - 2`
Multiplication	*	`3 * 2`
Right division	/	`3 / 2`
Left division	\	`3 \ 2 = 2 / 3`
Exponentiation	^	`3 ^ 2`

Vectors & matrices

5	10	56

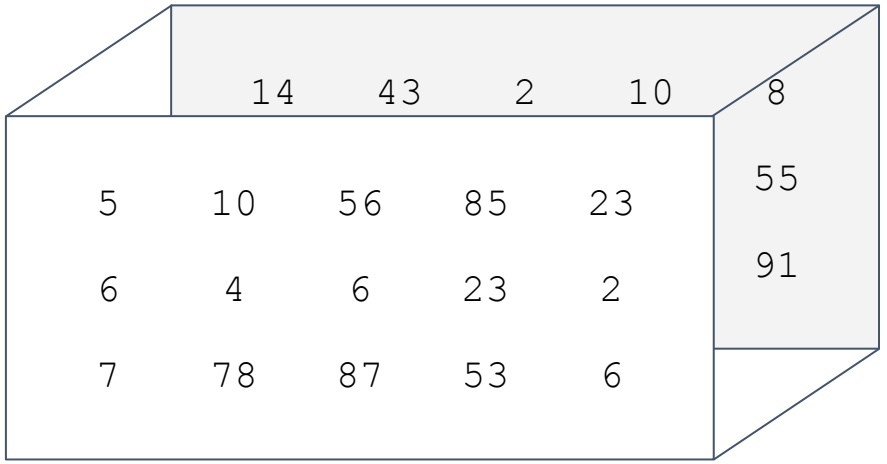
row vector
1 x 3

column vector
3 x 1

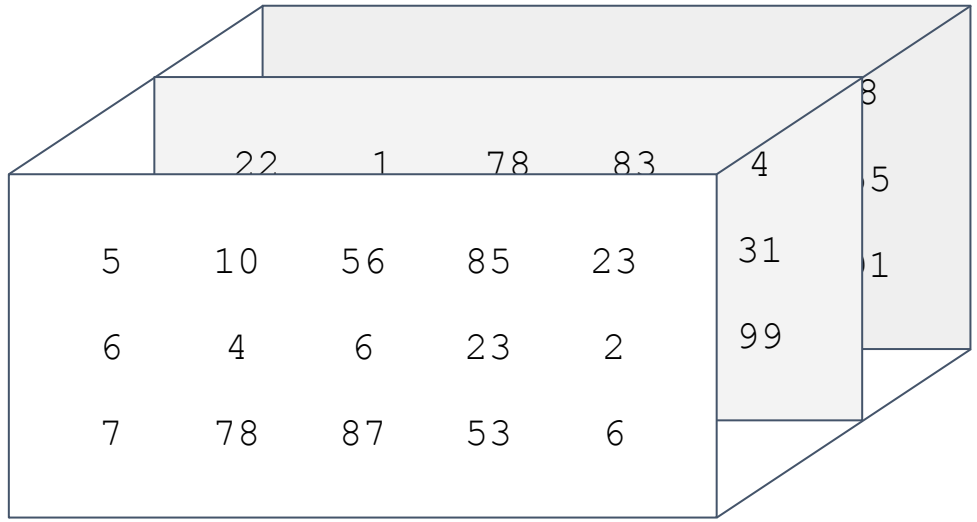
23
2
6

2D matrix
3 x 3

5	10	56
6	4	6
7	78	87



3D matrix
3 x 5 x 2



3D matrix
3 x 5 x 3

Creation

From a known list of numbers

```
x = [vector elements], A = [matrix elements]
```

With constant spacing by specifying first term (m), last term (n), & spacing (q)

```
x = m:q:n, A = reshape(m:q:n, d1, d2)
```

row vector

```
x = [1, 2, 3]
```

```
x = 1:1:3
 x = 1:3
```

column vector

```
x = [2; 4; 6]
```

```
x = [2:2:6]'
```

2D matrix

```
A = [1, 4, 7;
     2, 5, 8;
     3, 6, 9]
```

```
A = reshape(1:9, 3, 3)
```

Array & matrix operators

	Operation	Operator	Description
Array	Addition	+	`A+B` adds `A` and `B`
	Subtraction	–	`A-B` subtracts `B` from `A`
	Element-wise multiplication	.*	`A.*B` is element-by-element product of `A` and `B`
	Right array division	./	`A./B` is matrix with elements `A(i,j)/B(i,j)`
	Element-wise power	.^	`A.^B` is matrix with elements `A(i,j)` to the `B(i,j)` power
Matrix	Matrix multiplication	*	`C = A*B` is linear algebraic product of `A` and `B`
	Matrix left division	/	`x = A\B` is solution to the equation Ax = B
	Matrix right division	\	`x = B/A` is solution to the equation xA = B
	Matrix transpose	'	`A'` is linear algebraic transpose of `A`

Today's tutorial

- Head to github.com/daniellegruber/IEEE-MATLAB-Tutorial and clone the repo
- Read the README file and think about which tutorial(s) you want to try according to your interests and level of experience
- Feel free to work alone or in groups, and we'll be around to answer any questions you have!
- For anyone from EENG 200, feel free to check out the code I wrote for the class: https://github.com/daniellegruber/EENG200