

SESSION 1: MATLAB BASICS

<http://uiuc-cse.github.io/matlab-sp17/>

OUTLINE

- Introduction - MATLAB, programming, GUI
- Variables - scalar, vector, matrices & Operators
- Replicating vectors & Reshaping matrices
- Functions
 - Area of a circle & volume of a sphere
 - Fahrenheit/Celsius & plotting
- Matrix Definition
 - Falling ballistic object example
- Element-wise & matrix operators : Truss forces example
- Control Flow – Loop, If Else

INTRODUCTION

- MATLAB Introduction
 - Tool for Linear Algebra
 - Functionality – tool box
- What is programming
 - Commands
 - Data or variables
 - Logic
- MATLAB GUI
 - Workspace
 - Command window
 - Command line
 - Script file - .m file

VARIABLES - SCALAR, VECTOR

- Variable

- Scalar:

`x=2 y=3 x+y x*y x/y x^y`

- Vector:

`V1 = [0 2 4 6 8] V2 = 0:2:8`

`1:2:7 % row vector`

`(1:2:7)' % column vector`

`1:6`

`linspace(0,10,5) % row vector linspace(0,10,5).' % column vector`

- Accessing element

Index in MATLAB starts from 1 (not 0).

`V1(3) M1(2,3) M1(:,2) M1(1,:)`

VARIABLES - MATRICES

- Matrices

- $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ $B = \begin{bmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{bmatrix}$
- $A * B$ A' $A * A$ $A .* A$
- $V = [1 \ 3 \ 4 \ 2]$ V' $V * V'$ $V .* V$ $\text{eye}(3)$ $\text{zeros}(3)$
- $C = \text{ones}(3)$ $D = \text{rand}(3)$ $C * D$ $C .* D$ C^2 $C.^2$
- $A(2,2) = 100;$ $V(5) = 1;$ $B(1:2) = 1;$
 $B(:,2) = 3;$ $B(4,4) = 4;$ $B(2,:) = 5;$

REPLICATING & RESHAPING

- Replicating elements in vectors

- Example : $N = 3$; $A = [4\ 5]$

Create N copies of each element in A, so $B = [4\ 4\ 4\ 5\ 5\ 5]$

Use **kron**: $K = \text{kron}(X,Y)$ returns the Kronecker tensor product of X and Y
`>> kron(A,[1 1 1])`

- Reshaping arrays:

- Example:

Reshape a 3-by-4 matrix into a 2-by-6 matrix.

$A = [1\ 4\ 7\ 10; 2\ 5\ 8\ 11; 3\ 6\ 9\ 12]$

$B = \text{reshape}(A,2,6)$ or $B = \text{reshape}(A,2,[])$

EXAMPLE: AREA OF A CIRCLE & VOLUME OF A SPHERE (FUNCTIONS)

- $A(r) = \pi r^2$ r : radius
- To make it reproducible, use a function
- Try it on 1:1:5
- Create a new function, `volOfSphere`, and make it work:
 `function [V] = volOfSphere(r)`

EXAMPLE: FAHRENHEIT/CELSIUS (FUNCTIONS)

- $T_F(T_C) = T_C \frac{180}{100} + 32$
- Write a function that performs this conversion:
function Tf = TempC2F(Tc)
...
- Take Tc as vector:
 - 1:2:200
 - linspace(0, 200, 101)
- Save the results in a file:
fopen
fprintf – with/without using loop
fclose

EXAMPLE: FALLING BALLISTIC OBJECT (VECTORIZATION, FUNCTIONS)

- $y(t) = \frac{1}{2}at^2 + v_0t + y_0$
 $a = g = -9.81$ $v_0=2520$ $y_0=0$ $t=1$
- Vectorize:
 $t = \text{linspace}(0,5,101);$
 Try: $y=a*t^2+v*t+x0;$
 Not working? Why?
- Plot the t and y
- Create a function: $\text{function } [y] = a_fall(t,v,x0)$
- Modify the function to not yield y -values less than zero.

EXAMPLE: TRUSS FORCES (ELEMENT-WISE & MATRIX OPERATORS)

- $T \cdot x = f \Rightarrow x = T^{-1} \cdot f$
- Let $f_1 = 10$ and $f_2 = 20$
- Define T and f using MATLAB new variable or load it from truss-matrix.mat.
- Solve the matrix using function 'inv' & using '\'
- Compare the speed of two method using tic and toc

CONTROL FLOW(LOOP, IF ELSE) & MATRIX DEFINITION

- for loop :
 for $i=1:n$ or $\text{linspace}(1,n,n)$
 do something on i
 end
- if-elseif – else
 if condition 1
 do something 1
 elseif condition 2
 do something 2
 else
 do something 3
 end
- Example: define matrix using for loop and if-elseif-else