

MA2252 Introduction to Computing

Lecture 4: Variables and Arrays (contd.)

Sharad Kumar Keshari

School of Computing and Mathematical Sciences
University of Leicester

Learning outcomes

- Create some special matrices
- Learn array indexing and arithmetic operations
- Understand Char Arrays
- Use load and save functions

Special matrices

$$\text{ones}(3) \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}_{3 \times 3} \quad \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}_{2 \times 2} \quad \text{ones}(2)$$

- Matrix of ones: use function 'ones'

- Null matrix: use function 'zeros'

$$\text{zeros}(2) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}_{2 \times 2}$$

- Identity matrix: use function 'eye'

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

- Diagonal matrix: use function 'diag'

$$v = [2 \ 3 \ 10]; \quad \text{diag}(v) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{eye}(3)$$

Demo

Special matrices (contd.)

$A = \begin{bmatrix} \quad \end{bmatrix}_{24 \times 8}$ \rightarrow to create bigger matrices from a smaller matrix
Using 'repmat' and 'reshape' functions
 $16 \times 2 = 32$

- **repmat** creates copies of a given matrix

$[1 \ 2 \ 3 \ 4]$

- **reshape** transforms size of a given matrix

$\text{reshape}(A, 1, 4)$ $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}_{2 \times 2}$
 $[1 \ 2 \ 3 \ 4]_{1 \times 4}$ $\text{repmat}(A, 2)$ $=$ $\begin{bmatrix} 1 & 2 & 1 & 2 \\ 3 & 4 & 3 & 4 \\ 1 & 2 & 1 & 2 \\ 3 & 4 & 3 & 4 \end{bmatrix}$
 $\text{reshape}(B, 3, 9)$ $B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}_{3 \times 3}$

Demo

Array indexing

Array indexing means extracting elements of array.

Examples:

$B(:, 2)$ *↑ 1 is the index of element 5 in array A*

- Let $A = [5 \ 3 \ 1 \ 0]$. Then $A(1) = 5$, $A(2) = 3$ and so on.

$A(3) = 1$
 $A(4) = 0$

- If Let $B = \begin{bmatrix} 2 & 6 & 5 \\ 7 & 1 & 3 \\ 5 & 2 & 8 \end{bmatrix}$ *3rd row*
1st column

$B(3, [1, 2, 3])$

Then $B(1, 1) = 2$, $B(1, 2) = 6$ and so on.

Array indexing (contd.)

- Use colon operator to extract a row or column.

Example: $B(3,:) = [5 \ 2 \ 8]$ (extracts the third row of B)

Handwritten notes:
A blue arrow points from the '3' to the text "3rd row".
A blue arrow points from the ':' to the text "all columns".

- You can also extract arrays from arrays!

Example: $B([2,3],[2,3]) =$ extracts the array $\begin{bmatrix} 1 & 3 \\ 2 & 8 \end{bmatrix}$.

Demo

Char Arrays

- Char Arrays store alphanumeric characters such as numbers and letters.

- 1-D char array is called a 'string'.

- Examples of string:

- name = 'Wei Hao'
- a = 'University's address'
- title = 'plot of $\sin x$ vs x '

combination of letters + numbers, '\$', 'x', ...

University's address

Wei Ha	Siao
Yiming	Xinrui

2x2 char array

Demo

Char Arrays (contd.)

→ To modify a pre-defined string

Char arrays can also be made using **sprintf** function.

Examples:

title1=sprintf('plot of %s vs x','sinx')

→ plot of sinx vs x

title2=sprintf('plot of %s vs x','cosx')

→ plot of cosx vs x

→ call the command history

Demo

Arithmetic operations with arrays

$$A^3 = \begin{bmatrix} 0 & 1 \\ 5 & 6 \end{bmatrix} \times \begin{bmatrix} 0 & 1 \\ 5 & 6 \end{bmatrix} \times \begin{bmatrix} 0 & 1 \\ 5 & 6 \end{bmatrix} \rightarrow \text{scalar}$$

$$B = \begin{bmatrix} 1 & 2 \\ 8 & 9 \end{bmatrix}$$

- Operations between a constant (say c) and a matrix (say A)

Examples: $A+c$, $A-c$, $A*c$, A/c , A^c

$$A+B = \begin{bmatrix} 1 & 3 \\ 13 & 15 \end{bmatrix}$$

- Operations between two matrices (say A and B)

Examples: $A+B$, $A-B$, $A*B$, A/B , $A.*B$, $A./B$, $A.^B$

$$A.*B = \begin{bmatrix} 0 & 2 \\ 40 & 54 \end{bmatrix}$$
$$A = \begin{bmatrix} 0 & 1 \\ 5 & 6 \end{bmatrix}$$
$$c = 3$$

$$A+3 = \begin{bmatrix} 3 & 4 \\ 8 & 9 \end{bmatrix}$$

$$A-3 = \begin{bmatrix} -3 & -2 \\ 2 & 3 \end{bmatrix}$$

Arithmetic operations with arrays (contd.)

eg $Y = \begin{bmatrix} 2 \\ 4 \\ 6 \\ 8 \\ 10 \end{bmatrix}$

- Transpose of a matrix

Example:

If $A = \begin{bmatrix} 2 & 3 \\ 7 & 0 \end{bmatrix}$ then $A' = \begin{bmatrix} 2 & 7 \\ 3 & 0 \end{bmatrix}$

Handwritten annotations for matrix A:
- "1st row" with an arrow pointing to the top row [2, 3]
- "2nd row" with an arrow pointing to the bottom row [7, 0]
- "1st column" with an arrow pointing to the left column [2, 7]
- "2nd column" with an arrow pointing to the right column [3, 0]

Demo

Arithmetic operations with arrays (contd.)

$$\text{factorial}(4) = 4 \times 3 \times 2 \times 1 = 24$$

Arithmetic functions can also take arrays as input.

Example:

`x=[1 2 3 4 5];`

`factorial(x)=[1 2 6 24 120]`

`factorial([1 2 3 4 5])`



`sin(x)`
`cos(x)`
`sqrt(x)`
`log(x)`

x as array

output will be an array

Demo

Saving and loading variable data

- **save function:** stores variable data in a .mat file

Command: save 'filename' 'variables to store'

• \uparrow myfile $x, y, title$

\rightarrow .mat
 $x=1, y=2$
title = 'plot of $\sin x$ '

- **load function:** recalls the variable data from a .mat file

Command: load 'filename'

load myfile.mat

Demo

End of Lecture 4

Please provide your feedback [▶ here](#)