

Lecture 4:

Creating and indexing matrices

Lecturer: Jian Wei Tay

Date: 30 August 2021

Learning objectives

- Declaring vectors and matrices
- Definition of functions and keywords in MATLAB
- Addressing matrix elements using subscript indexing
- Modifying, deleting, and inserting elements into a matrix

Vectors and Matrices

Vectors

$$[5 \quad 4 \quad 7]$$

Row vector

$$\begin{bmatrix} 3 \\ 9 \\ 12 \end{bmatrix}$$

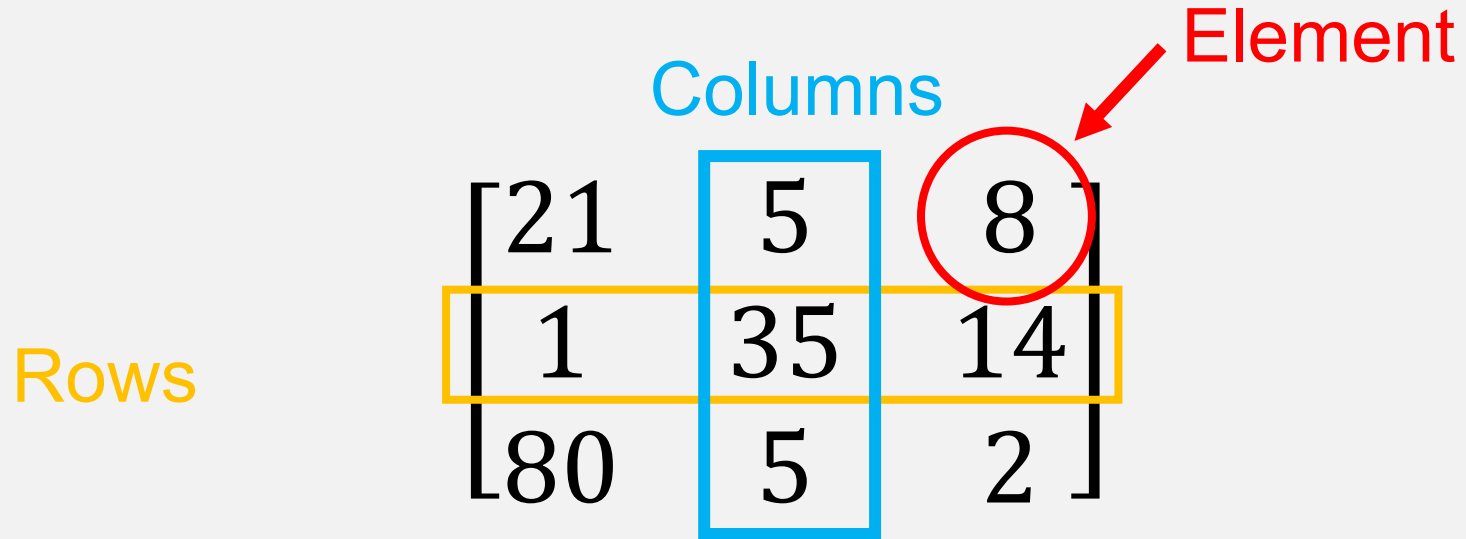
Column vector

Matrices (or arrays)

$$\begin{bmatrix} 21 & 5 & 8 \\ 1 & 35 & 14 \\ 80 & 5 & 2 \end{bmatrix}$$

2D Matrix

MATLAB terminology



Declaring a row vector

- Start and end with a square bracket []
- Separate elements by a space
- Example: To declare the matrix $A = [1 \ 2 \ 3 \ 4]$

```
>> A = [1 2 3 4]
```

Note: You can actually use commas and/or spaces to separate the elements, e.g.

- $A = [1,2,3,4]$
- $A = [1, 2, 3, 4]$

But just spaces look neater

Declaring a column vector

- Separate elements using semicolons (;)
- Example: To create the column vector B

$$B = \begin{bmatrix} 2 \\ 4 \\ 6 \\ 8 \end{bmatrix}$$

```
>> B = [2; 4; 6; 8]
```

Declaring matrices

- Enter a matrix row by row
- Example: To create the matrix C

$$C = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

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

Practice

- What is the command to create the matrix below?

$$M = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

- A. `M = [1 2 3 4]`
- B. `M = [1 3; 2 4]`
- C. `M = [1 2; 3 4]`
- D. `M = [1; 3 2; 4]`

Size of 2D matrices

Number of rows x Number of columns

Example

$$\begin{bmatrix} 21 & 5 \\ 1 & 35 \\ 80 & 5 \end{bmatrix}$$

- This matrix has size 3 x 2 (or 3-by-2)

Matrix dimensions

- MATLAB often uses a number to refer to matrix dimensions (this is more necessary for multi-dimensional matrices)
- By definition:
 - Dimension 1 = rows
 - Dimension 2 = columns

Getting the size of a matrix using a function

$$D = \text{size}(X)$$

For an M-by-N matrix X, `size` returns a two-element row vector $D = [M, N]$

Functions

A function is a named code that takes input(s) and (usually) returns output(s)

The anatomy of a MATLAB function

$D = \text{size}(X)$

Output
arguments

Function
name

Input
arguments

Using built-in documentation

- Access built-in documentation about a function by using the `help` function
- Example:

```
>> help size
```

Note: You do not need to remember function syntax. Rely on the built-in MATLAB documentation.

Using built-in documentation

- To get more detailed documentation, you can also use the doc function

```
>> doc size
```


Questions?

Indexing elements in a matrix

- Types of indexing in MATLAB:
 - Subscript indexing
 - Linear indexing (later in the course)
 - Logical indexing (later in the course)

Subscript indexing

- In subscript indexing, elements in a matrix are identified by their row and column

$$A = \begin{bmatrix} 10 & 30 & 21 \\ 5 & 25 & 3 \\ 9 & 55 & 2 \end{bmatrix}$$

Subscript indexing

- In subscript indexing, elements in a matrix are identified by their row and column

$$A = \begin{bmatrix} 10 & 30 & 21 \\ 5 & 25 & 3 \\ 9 & 55 & 2 \end{bmatrix}$$

Row 3, Column 2

Subscript Indexing in MATLAB

- Declare the matrix below:

$$A = \begin{bmatrix} 10 & 30 & 21 \\ 5 & 25 & 3 \\ 9 & 55 & 2 \end{bmatrix}$$

Subscript Indexing in MATLAB

- Indexing the element in row 3, column 2:

```
>> A(3, 2)
```

Practice

- What is the command that indexes the highlighted element?

$$A = \begin{bmatrix} 10 & 30 & 21 \\ 5 & 25 & 3 \\ 9 & 55 & 2 \end{bmatrix}$$

- A. $A(3, 1)$
- B. $A(3, 3)$
- C. $A(1, 3)$
- D. $A(2, 3)$

The keyword end

- In MATLAB, keywords are words that are reserved and have built-in uses
- **Note:** Reserved words cannot be used as variable or script names
- The keyword end can be used to select the last element in a row or column:

```
>> A(3, end)
```


Indexing entire rows and columns

- Use the colon operator (:) to index a whole row or column

```
>> A(:, 1)
```

Note: One way to remember what the colon operator does is to read it as "all".

For example, $A(:, 1)$ can be read as indexing "all rows, column 1"

Indexing entire rows and columns

- Use the colon operator (:) to index a whole row or column

```
>> A(3, :)
```

Note: "row 3, all columns"

Another use for the : operator

- The colon operator can also be used to generate a range of numbers

- Example:

```
>> R = 1:10
```

Note: A range is a row vector with elements that increase by the same amount. In this example, each subsequent element is +1 greater than the one before it.

Using ranges to select multiple rows or columns

- Example: Selecting rows 1 and 2 of matrix A

```
>> A(1:2, :)
```

Practice

- Which command(s) below will return all elements in columns 4 through 6 of a matrix C? **There may be more than one answer.**

A. `C(4, 6)`

B. `C(:, 4:6)`

C. `C = 4:6`

D. `C[:, 4:6]`

E. `C(:, [4 5 6])`

Modifying matrix elements

- You can modify matrix elements by assigning them a different value using the assignment operator =
- Example:

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

Modifying matrix elements

- You can also change an entire row or column

```
>> A(:, 3) = 1
```

```
>> A(1, :) = [10 20 30]
```

Note: If a vector or matrix is on the right of the = operator, it must have the same number of elements as the elements being replaced.

Example: `A(1, :) = [10 20]` will return the error **Unable to perform assignment because the size of the left side is 1-by-3 and the size of the right side is 1-by-2.**

Questions?

Deleting matrix elements

- You can delete matrix elements by assigning them to an empty matrix `[]`
- Example: Deleting row 1 of matrix A

```
>> A(1, :) = []
```

Note: When deleting matrix elements, the result must still be a valid matrix.

For example, `A(3, 2) = []` will return the error **A null assignment can have only one non-colon index.**

Practice

- Reset matrix A

$$A = \begin{bmatrix} 10 & 30 & 21 \\ 5 & 25 & 3 \\ 9 & 55 & 2 \end{bmatrix}$$

Adding new rows (or columns)

- You can add new rows (or columns) to a matrix by indexing

```
>> A(4, :) = [1 2 3]
```

Remember: The resulting matrix must be rectangular. If in doubt, it can help to draw out the matrix and the operation you are trying to carry out to make sure the dimensions are correct.

Adding new rows (or columns)

- If you skip rows (or columns), MATLAB will simply add zeros in

```
>> A(10, :) = [1 2 3]
```

Practice Questions

What is the size of the matrix below?

$$M = \begin{bmatrix} 1 & 8 & 10 \\ 2 & 4 & 15 \end{bmatrix}$$

- A. 2 x 3
- B. 3 x 2
- C. 6

Which of the following commands will correctly add row(s) or column(s) to matrix A?

$$A = \begin{bmatrix} 10 & 30 & 21 \\ 5 & 25 & 3 \\ 9 & 55 & 2 \end{bmatrix}$$

- A. $A(5, :) = [1 \ 2 \ 3]$
- B. $A(5, :) = [1; 2; 3]$
- C. $A(:, \text{end} + 1) = [1; 2; 3]$
- D. $A(\text{end} + 1) = 10$