

Function in MATLAB

Adapted from Duane Hanselman and Bruce Littlefield

MATLAB: class 4

Md. Jalal Uddin

PhD candidate — Atmospheric Physics and Atmospheric Environment

School of Atmospheric Physics, NUIST, China

Founder and Director of Research Society

Website: <https://researchsociety20.org/founder-and-director/>

Email: dmjalal90@gmail.com, 20205103002@nuist.edu.cn

Function in MATLAB

Output = input

$$Y = f(x)$$

a **function**, a **procedure** or a **method** = collection of commands

Function in MATLAB

outputs in square brackets, []

inputs in parentheses ()

`function` [outputs] = function_name (inputs)



... statements

`return`

Types of function in MATLAB

Built-in: already exist in MATLAB

- **example:** mean, var, std etc.

User-defined: users should write their own function

- **example:** avg, variance, standard_deviation etc.

Rules

- Start with “function” keyword and end with “return” keyword

FUNCTION FILE:

- The first line of a FUNCTION FILE is called the function-declaration line.
- The name of the FUNCTION FILE must be the same name as the FUNCTION.
- FUNCTION FILE names are case sensitive. It is recommended to use only lowercase letters.

Rules

FUNCTION NAMES

- FUNCTION NAMES must begin with a letter.
- Any combination of letters, numbers, and underscores CAN APPEAR after the first character.
- FUNCTION NAMES CANNOT contain spaces or punctuation characters.

Rules

- A **FUNCTION** can report a warning and then continue operation by calling the function warning.
- **FUNCTIONS** can contain nested or sub **FUNCTIONS**.

Input and Output arguments

- **FUNCTIONS** cannot be called with more input or output arguments than the file specifies.
- The number of input and output arguments can be determined by the **FUNCTIONS** `nargin` and `nargout`, respectively.

nargin and nargsout (adapted from MATLAB)

```
function c = addme(a,b)
```

```
switch nargin
```

```
    case 2
```

```
        c = a + b;
```

```
    case 1
```

```
        c = a + a;
```

```
    otherwise
```

```
        c = 0;
```

```
end
```

```
fx = 'addme';  
nargin(fx)
```

```
ans =
```

```
2
```

```
function [dif, absdif] = subtract(y,x)
```

```
dif = y - x;
```

```
if nargsout > 1
```

```
    disp('Calculating absolute value')
```

```
    absdif = abs(dif);
```

```
end
```

```
fx = 'subtract';  
nargsout(fx)
```

```
ans =
```

```
2
```

varargin (adapted from MATLAB)

accept any number of input arguments

```
function varlist(varargin)
    fprintf('Number of arguments: %d\n', nargin)
    celldisp(varargin)
```

```
varlist(ones(3),'some text',pi)
```

```
Number of arguments: 3
```

```
varargin{1} =
    1     1     1
    1     1     1
    1     1     1
```

```
varargin{2} =
some text
```

```
varargin{3} =
    3.1416
```

varargin (adapted from MATLAB)

In cases where one or more input arguments are fixed, **varargin** must appear as the last argument.

```
function varlist2(X,Y,varargin)
    fprintf('Total number of inputs = %d\n', nargin);

    nVarargs = length(varargin);
    fprintf('Inputs in varargin(%d):\n', nVarargs)

    for k = 1:nVarargs
        fprintf(' %d\n', varargin{k})
    end
```

```
varlist2(10,20,30,40,50)
```

```
Total number of inputs = 5
Inputs in varargin(3):
    30
    40
    50
```

varargout

return any number of output arguments

```
function varargout = myfunction(x)
```

```
function [a, b, varargout] = myfunction(x)
```

Let's practice in
MATLAB with real data

Average

$$\bar{X} = \frac{\sum X}{N}$$

Variance for sample

$$s^2 = \frac{\sum (X - \bar{X})^2}{n - 1}$$

Standard deviation for sample

$$s = \sqrt{\frac{\sum (x - \bar{x})^2}{n - 1}}$$