

# Matlab 和 R 中函数使用的异同

陈普

2023 年 8 月 24 日

数据处理、计量分析我通常使用 R，但在宏观经济建模中，又不可避免地要使用到 Matlab，两个软件总体上语法类似。但在函数的撰写和参数传递上存在一些差异。很少看到有资料介绍这种区别，本文做一个简单比较。

## 1 函数脚本放在哪里

**R** 在 R 语言中，函数当然可以用单独的脚本来撰写，然后储存。该脚本可以放在任何地方，但你使用时要利用 `source('/mymean.R/')` 把该脚本载入内存。你也可以把函数直接写在脚本中的任何地方，但一定要放在调用该函数的语句之前。比如下面这个算均值的函数。

Listing 1: R 脚本 main.R

```
x <- seq(1,10)
n <- length(x)

mymean <- function(v){
  # MYMEAN Local function that calculates mean of array.

  a <- sum(v)/n
}

avg <- mymean(x)
```

**Matlab** 在 Matlab 中，你最好把写函数的脚本（比如 `mymean.m`）与你的主程序脚本（`main.m`）放在同一目录中，这样你运行 `main.m` 时 Matlab 就能找到 `mymean.m` 函数。而且与 R 不同的是，函数脚本的名称与函数名称必须一致，即函数脚本是 `mymean.m`，则函数名必须为 `mymean`，R 则没有这样的限制。

当然也可以把函数写在脚本中，但函数必须位于脚本的最后，就像下面这样。而且这样的函数只在该脚本中可以得到识别，Matlab 的工作空间是看不到的。

Listing 2: Matlab 脚本 main.m

```
x = 1:10;
n = length(x);
avg = mymean(x,n);

function a = mymean(v,n)
% MYMEAN Local function that calculates mean of array.

a = sum(v)/n;
end
```

Matlab 和 R 函数最大的差异在于参数的传递，你可能已经注意到在 R 脚本中，我们没有给函数 `mymean` 传递参数 `n` 的值，R 语言会自动在工作空间中寻找该值，即变量 `n` 的值在脚本和函数间实现了共享。但 Matlab 则不会，如果不在 `mymean` 中设置参数 `n`，则会报错。

Matlab 不会在脚本与函数间共享参数值，也不会主函数和子函数<sup>1</sup>间共享参数值，但如果在一个单独的脚本中撰写嵌套函数，则这些函数间的变量值是共享的，比如 Listing 4 中的 `x` 在父函数和子函数间就是共享的。

Listing 3: 脚本 `mystats.m` 中的主函数和子函数

```
function [avg, med] = mystats(x)
n = length(x);
avg = mymean(x,n);
med = mymedian(x,n);
end

function a = mymean(v,n)
% MYMEAN Example of a local function.

a = sum(v)/n;
end

function m = mymedian(v,n)
% MYMEDIAN Another example of a local function.

w = sort(v);
if rem(n,2) == 1
m = w((n + 1)/2);
else
m = (w(n/2) + w(n/2 + 1))/2;
end
end
```

Listing 4: 嵌套函数

```
function main1
x = 5;
nestfun1

function nestfun1
x = x + 1;
end

end
```

## 2 函数中参数的传递

R 语言函数中参数的写法是非常灵活的，比如在 Listing 5 中，既可以用参数名传递值，也可以用位置来传递值，而且参数 `n` 还给出了默认值，即便不传递 `n` 值，也可以使用默认值 5。

Listing 5: R 语言参数的写法

```
mymean <- function(v, n = 5){
  # MYMEAN Local function that calculates mean of array.

  a <- sum(v)/n
}

# 下面的函数计算 1, 2, 3, 4, 5 的均值，会得到相同的值
mymean(n = length(seq(1,5)), v = seq(1,5)) # 按名称传递参数，位置可以随意
mymean(seq(1,5), length(seq(1,5))) # 按位置传递参数，位置不可以随意变动
mymean(v = seq(1,5)) # n 取默认值 5
```

<sup>1</sup>一个单独的脚本中写多个函数，第一个函数称主函数，其他函数称子函数或局部函数。主函数在工作空间可见，子函数不可见，即命令行无法调用子函数，只有主函数可以调用子函数

Matlab 一直的传统都是按位置传递参数，2019 年以后的版本给出了按名称传递参数的语法，但在写法上要复杂很多。因为它要在函数的第一行增加一个参数块`arguments...end`。下面给一个简单的例子。

Listing 6: matlab 中按名称传递函数参数的写法

```
x = 1:10;
n = length(x);
avg = mymean(x, m = n); % 按名称传递参数

function a = mymean(v,opt)
arguments
v double % double 约定了 v 的数值类型，这不是必须的
opt.m double = 5 % double 约定了 m 的数值类型，这不是必须的。同时也给出了该参数的默认值为 5。
end
a = sum(v)/opt.m;
end
```

函数`mymean`包含了两类参数，都包裹在`arguments...end`代码块内。一类是按位置定的参数，如`v`，这样的参数一定要排在前面。另一类是按名称定的参数`opt`，要排在后面。这是一个结构体，它的域名`m`就是调用时参数的名称。如果你有多个按名称定的参数，都要放在`opt`结构体内。