

在线交流，有问有答

(详见封底勒口)



书中所有程序的源代码可通过扫描此二维码免费下载。

MATLAB®&Simulink®
工程师系列丛书

MATLAB在遥感技术中的应用

王成墨 编著

MATLAB®

examples



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

T
MATLAB® & Simulink® 工程师系列丛书

MATLAB 在遥感技术中的应用

王成墨 编著



北京航空航天大学出版社

内 容 简 介

从遥感领域中一些常见应用着手,以 MATLAB 为编程工具,辅以实际的案例说明,详细介绍了 MATLAB 中可以用到的且与遥感相关的一些常见方法。本书分为上、下两篇。上篇为 MATLAB 基础,介绍使用 MATLAB 时的一些基本概念和操作,增加对 MATLAB 本身的一些了解,主要包括:MATLAB 简介、M 语言、MATLAB 代码调试和优化、MATLAB 可视化、MATLAB 文件 I/O、MATLAB 编译与调用。下篇为遥感技术的 MATLAB 应用,介绍遥感领域应用过程中需要的基本方法及其对应 MATLAB 中的相关函数和用法,主要包括:影像灰度处理、影像几何处理、大数据影像处理、特征提取与影像匹配、非监督法学习、监督法学习、坐标转换及地图投影、数值优化、自动微分。

本书可以作为高等院校遥感学科相关课程本科生、研究生的教材或教学参考书,也可供从事数字图像处理、遥感应用相关的人员参考。

图书在版编目(CIP)数据

MATLAB 在遥感技术中的应用 / 王成墨编著. — 北京 :
北京航空航天大学出版社, 2016. 8

ISBN 978-7-5124-2217-9

I. ①M… II. ①王… III. ①Matlab 软件—应用—遥
遥感技术 IV. ①TP7

中国版本图书馆 CIP 数据核字(2016)第 190690 号

版权所有,侵权必究。



MATLAB 在遥感技术中的应用

王成墨 编著

责任编辑 张冀青

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:goodtextbook@126.com 邮购电话:(010)82316936

北京兴华昌盛印刷有限公司印装 各地书店经销

*

开本:787×1 092 1/16 印张:19 字数:499 千字

2016 年 9 月第 1 版 2016 年 9 月第 1 次印刷 印数:4 000 册

ISBN 978-7-5124-2217-9 定价:42.00 元

前 言

MATLAB 作为一种用于科学计算的软件,其强大之处,对初学者而言,在于其精简的编程风格,可以降低学习编程的门槛;对于熟知 MATLAB 的人而言,则在于其内置了非常丰富的函数库,包含了大量的基础算法,在各个领域都可以大显身手,通过合理组合运用,借助 MATLAB,花少量的时间就能完成复杂的工作。有人说,对数表的发明减少了天文学家在计算方面所花费的时间,变相延长了天文学家的生命。MATLAB 中已有的基础算法,同样也减少了很多重复地编程、开发、测试工作,直接拿过来就能用,也变相增加了使用 MATLAB 用于科学计算的人生命。同时, MATLAB 中各个函数的使用方法也不仅限于一种,让使用的过程变得更加简单和灵活。以矩阵为核心单元的各种操作简单、有效、快捷,更是其他软件和语言所难以比拟的。掌握了 MATLAB,科学计算将不再困难。

遥感作为一门基础的空间信息学科,其作用也是不言而喻的,基于位置信息的手机导航应用服务、谷歌地球、数字城市等,都是与遥感息息相关的产物。从当前情况来看,市面上出现的 MATLAB 书籍都偏向于其他专业性的描述,缺乏与遥感学科相关的讲解。而对于遥感学科而言,除了基本的相关专业知识需要了解外,对于编程相关的讲述也比较少。知行合一,不仅需要从专业知识上了解这门学科,在实际动手操作上,同样也需要仔细钻研琢磨。这本书就是希望通过遥感学科作为一个切入点,通过一些案例讲解,介绍 MATLAB 在其中的应用。本书分为上、下两篇,上篇为 MATLAB 基础,包括: MATLAB 简介、M 语言、MATLAB 代码调试和优化、MATLAB 可视化、MATLAB 文件 I/O、MATLAB 编译和调用。下篇为遥感技术的 MATLAB 应用,包括:影像灰度处理、影像几何处理、大数据影像处理、特征提取与影像匹配、非监督法学习、监督法学习、坐标转换及地图投影、数值优化、自动微分。其中,除了包含 MATLAB 帮助文档中介绍的一些试验外,还包含了一些基于不同需要进行算法扩展的试验和见解。

与其临渊羡鱼,不如退而结网。除了通过阅读增加必要的基本知识外,多动手试验也是 MATLAB 学习极为重要的一步。多动手实现算法,能帮助使用者快速熟悉 MATLAB 中的各个功能;试验过程中难免会犯错,通过分析、查找错误,对于 MATLAB 的理解也会变得更加深刻。勤于思考、善于总结也是一个优秀的 MATLAB 使用者所必备的条件。往往好的解决问题的想法是通过不断试验、反复思考之后的灵光一闪。当偶有发现,无论是困惑许久的问题得到解决,还是发现了新的有用函数,就应当及时记录下来。长期的积累才会保证需要用的时候能轻松找到,并与之对应,这样才不会“书到用时方恨少”,做到“为有源头活水来”。从最初学习到慢慢熟练的过程中,会遇到很多困难,多和高手交流、取经,在比较中发现不同,从交流中增长见识。

笔者得到了 MATLAB 论坛版主吴鹏(rocwoods)的热心帮助。作为良师益友,从他身上我学习到了很多关于 MATLAB 的宝贵经验和知识。在本书的写作过程中,感谢北京航空航天大学出版社陈守平编辑的支持与鼓励,陈守平编辑和吴鹏为本书提出了宝贵的修改意见。在此,向他们表示最真诚的感谢!

最后,还要感谢我的家人,是她们在背后默默付出和支持,让我可以顺利完成这本书的写作。在此,向我的家人表示最由衷的谢意。

本书为读者免费提供程序源代码,以二维码的形式印在扉页及前言后,请扫描二维码下载。读者也可以通过网址 <http://pan.baidu.com/s/1skLrpZf> 从“百度云”下载全部资料。同时,北京航空航天大学出版社联合 MATLAB 中文论坛为本书设立了在线交流平台,网址: <http://www.ilovematlab.cn/forum-257-1.html>。我们希望借助这个平台实现与广大读者面对面的交流,解决大家在阅读本书过程中遇到的问题,分享彼此的学习经验,从而达到共同进步的目的。

由于作者水平有限,书中的疏漏和不当之处,欢迎广大读者和同行批评指正。本书勘误网址: <http://www.ilovematlab.cn/thread-478167-1-1.html>。作者邮箱: wangchengmosk@163.com。

王成墨
2016 年于北京



程序源代码下载说明

二维码使用提示:手机安装有“百度云”App 的用户可以扫描并保存到云盘中;未安装“百度云”App 的用户建议使用 QQ 浏览器直接下载文件;ios 系统的手机在扫描前需要打开 QQ 浏览器,单击“设置”,将“浏览器 UA 标识”一栏更改为 Android;Android 等其他系统手机可直接扫描、下载。

配套资料下载或与本书相关的其他问题,请咨询理工图书分社,电话:(010)82317036,(010)82317037。

目 录

上篇 MATLAB 基础

第 1 章 MATLAB 简介	3
1.1 MATLAB 的起源和发展	3
1.2 MATLAB 8.0 时代	3
1.3 MATLAB 使用介绍	4
1.3.1 菜单栏	4
1.3.2 当前目录窗口	5
1.3.3 命令窗口	5
1.3.4 工作空间	6
1.3.5 命令历史窗口	6
1.3.6 编辑界面	7
1.3.7 帮助系统	8
1.4 本章小结	10
第 2 章 M 语言	11
2.1 变 量	11
2.1.1 数据类型	11
2.1.2 矩 阵	12
2.2 符 号	13
2.2.1 单字符	13
2.2.2 多字符	16
2.3 关键字	19
2.4 控制流	20
2.4.1 顺 序	20
2.4.2 选 择	20
2.4.3 循环——特殊的顺序	21
2.4.4 递归——特殊的循环	21
2.4.5 异常捕获	22
2.5 函 数	22
2.5.1 函数类型	22
2.5.2 函数调用	23
2.5.3 函数中变量的生命周期	23
2.5.4 函数句柄	23
2.6 M 语言特性	24
2.7 本章小结	25

第 3 章 MATLAB 代码调试和优化	26
3.1 代码分析	26
3.2 代码调试	28
3.2.1 调试方法	29
3.2.2 调试检查项	30
3.3 代码运行异常	30
3.3.1 错误	30
3.3.2 警告	31
3.3.3 崩溃	31
3.4 代码纠错	31
3.5 代码优化	32
3.6 代码心得	35
3.7 本章小结	36
第 4 章 MATLAB 可视化	37
4.1 二维图形可视化	37
4.2 三维图形可视化	45
4.3 统计图可视化	50
4.4 本章小结	53
第 5 章 MATLAB 文件 I/O	54
5.1 文件路径处理	54
5.2 MAT 文件	57
5.3 文本文件	58
5.4 二进制文件	64
5.5 影像文件	64
5.6 本章小结	66
第 6 章 MATLAB 编译与调用	67
6.1 编译	67
6.1.1 可执行程序编译	68
6.1.2 动态链接库编译	69
6.2 调用	70
6.2.1 MATLAB 调用 C++	71
6.2.2 C++ 调用 MATLAB	73
6.3 本章小结	76

下篇 遥感技术中的 MATLAB 应用

第 7 章 影像灰度处理	79
7.1 二值影像处理	79
7.1.1 影像二值化	79
7.1.2 形态学处理	80
7.1.3 图斑统计	83

7.1.4 边界提取	84
7.2 灰度影像处理	85
7.2.1 影像滤波	86
7.2.2 影像增强	89
7.3 傅里叶变换	91
7.4 色彩空间转换	92
7.5 影像重采样	92
7.6 本章小结	96
第 8 章 影像几何处理	97
8.1 影像坐标系统	97
8.2 影像旋转	98
8.3 影像缩放	105
8.4 影像纠正	109
8.5 本章小结	111
第 9 章 大数据影像处理	112
9.1 影像存储结构	112
9.2 影像分块	112
9.3 分块后处理	116
9.4 本章小结	130
第 10 章 特征提取与影像匹配	131
10.1 特征检测	131
10.1.1 点特征检测	132
10.1.2 线特征检测	136
10.2 特征描述	139
10.3 影像匹配	141
10.3.1 特征匹配	141
10.3.2 配准纠正	154
10.3.3 密集匹配	161
10.4 本章小结	170
第 11 章 非监督法学习	171
11.1 分层聚类	171
11.1.1 函数 pdist	171
11.1.2 函数 linkage	172
11.1.3 函数 cluster	173
11.1.4 函数 clusterdata	174
11.2 K 均值聚类和 K 中心聚类	175
11.3 最邻近聚类	181
11.4 高斯混合模型	185
11.5 隐式马尔可夫模型	190
11.6 聚类评价	193

11.7	本章小结	196
第 12 章	监督法学习	197
12.1	分类树	200
12.2	判别分析	204
12.3	朴素贝叶斯分类	213
12.4	最邻近分类	217
12.5	支持向量机	220
12.6	分类集成	226
12.7	模型构建和评估	227
12.8	本章小结	229
第 13 章	坐标转换及地图投影	230
13.1	坐标基准和空间参考系统	230
13.2	相同坐标基准的坐标转换	231
13.3	不同坐标基准的坐标转换	234
13.4	WKT 与坐标系统	235
13.5	本章小结	241
第 14 章	数值优化	242
14.1	常见的优化方法	242
14.1.1	二分法	242
14.1.2	牛顿法	244
14.1.3	最小二乘法	246
14.2	优化步骤	247
14.2.1	选择方法	248
14.2.2	目标函数	248
14.2.3	约束条件	249
14.2.4	输入参数	249
14.2.5	参数设置	249
14.2.6	优化结果	249
14.3	函数介绍	250
14.3.1	函数 fzero 与函数 fsolve	250
14.3.2	函数 fminunc	252
14.3.3	函数 fminbnd	253
14.3.4	函数 fmincon 与函数 linprog	254
14.3.5	函数 fminsearch	256
14.3.6	函数 lsqlin 与函数 lsqnonlin	256
14.3.7	函数 bintprog 与函数 intlinprog	257
14.4	回归分析	259
14.4.1	线性回归	259
14.4.2	广义线性回归	260
14.5	本章小结	262

第 15 章 自动微分	263
15.1 自动微分的方法	263
15.1.1 代数解	263
15.1.2 数值解	264
15.2 复数法一阶自动微分	265
15.3 面向对象的自动微分	267
15.3.1 一阶自动微分	271
15.3.2 高阶自动微分	277
15.4 自动微分应用	288
15.4.1 雅克比矩阵	288
15.4.2 海森矩阵	289
15.5 本章小结	291
参考文献	292

上 篇

MATLAB 基础

1.1 MATLAB 的起源和发展

MATLAB 是由美国 MathWorks 公司出品的一款商业数学软件,用于算法开发、数据可视化、数据分析以及数值计算,主要包括 MATLAB 和 Simulink 两大部分。MATLAB 这个名称源自 Matrix 与 Laboratory 两个单词前三个字母的组合,意即矩阵实验室。其最重要的核心数据单元是矩阵,计算的核心思想也是矩阵。

20 世纪 70 年代, MATLAB 最初版本采用的编程语言是 FORTRAN,主要用于科学运算、数值分析等领域。随着科技的发展, MATLAB 逐步发展成为国际公认的标准科学计算软件,数值计算分析的功能不断强大、完善。

随着程序语言的演进, MATLAB 的内核也逐步采用 C、C++ 语言进行编写,功能日渐丰富且不断扩展。它不仅具备强大的计算能力,而且拥有丰富的内置函数,其所涉及的领域和专业也越来越广,每一个工具箱(Toolbox),基本上都涉及了一个领域的基本操作。

MATLAB 还提供了便捷的数据可视化功能,将枯燥的数据用生动的图形,整体而全面地展示出来,这无疑对数据分析有着莫大的帮助,而且能从宏观上,更确切地把握数据的趋势和变化。MATLAB 拥有全面易懂的帮助系统,从中能搜索到各个函数的用法和例子,能轻车熟路地模仿和学习,很快掌握其中的用法。

MATLAB 也提供了自己的编译器和编程语言(后文将简写为 M 语言),可以将程序编译运行成动态库、可执行文件,可以创建用户界面进行交互,也可以编写为其他高级语言程序,还能和其他语言混合编程、互相调用,包括 C、C++、Java、Python 和 FORTRAN。从软件移植上来说, MATLAB 具有较好的可移植性,能很好地支持不同操作系统和平台,如 32 位、64 位系统,以及 Microsoft Windows、UNIX 和 Apple Mac OS X 等,程序开发者不用再担忧和苦恼因为平台系统的差异而对代码产生巨大而颠覆性的改动。

Simulink 部分是基于 MATLAB 的框图来设计环境的,因此可以对各种动态系统进行建模、分析和仿真。Simulink 提供了一个可视化开发环境,常用于系统模拟、动态/嵌入式系统开发等方面。

正是 MATLAB 的这些优秀的功能和广泛的应用,因此它与 Maple、Mathmatica 并称三大数学软件。

1.2 MATLAB 8.0 时代

MathWorks 公司从 2006 年开始,每年进行两次 MATLAB 产品发布会,时间分别是在每年的 3 月和 9 月,释放版本以 R 开头加上年份,上半年版本以 a 结尾,下半年版本以 b 结尾。目前, MATLAB 的释放版本已经到了 R2015b。这样的更新速度,预示着 MATLAB 的不断

更迭进步。版本发布随之而来的是每个版本的新特性、错误修复和新用法、新函数、新模块的推出。自 R2012b 推出后,版本号从 MATLAB 7.14 转换到了 MATLAB 8.0, MATLAB 界面也出现了前所未有的焕然一新,从而开启了 MATLAB 8.0 的时代。

MATLAB 8.0 时代最显著的就是界面与以往发生了巨大变化,采用了流行的 Ribbon 风格,界面更加优美简洁,数据可视化变得更加简单直接。最常用的集成开发环境(IDE, Integrated Development Environment)变化很大。原先的菜单样式,追随了如今流行的简约模块化风格,新增了单元测试函数及用例,数据容器 map 映射、table 表格,对大数据的处理,与 Hadoop 的结合使用。工具箱中增加的函数就更多了,就不在这里一一表述了。在 MathWorks 的官方网站,记录了每个工具箱在每个版本中的演变和进展。越来越多的函数将会在相应的工具箱中出现,对 MATLAB 的使用者来说,这些改变将会越来越好地融入到学习 MATLAB 的过程中,也将引领 MATLAB 8.0 时代。

1.3 MATLAB 使用介绍

打开 MATLAB 之后,可以看到一个全局的 MATLAB 显示界面。在这个全局界面中,整体分为两个层次,一个是处在上部的菜单栏,另一个是处在下部的显示部分。其中显示部分又包含了左侧的目录窗口、居中的命令窗口、右侧的工作空间。原有位于右下的命令历史窗口被默认隐藏,只有在命令窗口输入命令时才显示,根据个人习惯也可以将其手动调整到窗口的右下侧。

1.3.1 菜单栏

菜单栏中,HOME 菜单下存放了一些基本的常用菜单项,如新建文本、新建可选的文本模板(例如脚本、函数、类、例子等 m 文件模板)、打开 m 文件及最新编辑过的 m 文件、文件搜索、文件比较、导入数据、保存空间、添加变量、打开变量、清除工作空间、分析代码、运行与时间统计、清除命令等,如图 1.3.1 所示。



图 1.3.1 菜单栏

通常,用得比较多的还是新建文件、运行与时间统计这部分功能。位于菜单栏的功能一般都可以采用一些常用的简单命令方式完成,表 1.3.1 中列出的是命令窗口常用函数。

表 1.3.1 命令窗口常用函数

函 数	说 明
clear	清除当前工作空间的变量。默认清除所有变量。clear a 表示只清除变量 a
clc	清除当前命令窗口中的内容
save	保存当前工作空间的变量到指定 mat 文件中。默认保存所有变量。 有两种写法:save('1.mat','a')和 save 1.mat a。 都代表只保存变量 a 到当前目录下的 1.mat 文件中

续表 1.3.1

函 数	说 明
load	加载指定 mat 文件中的变量到当前工作空间中。默认加载所有变量。 有两种写法:load('1.mat','a')和 load 1.mat a。 都代表只加载当前目录下 1.mat 文件中的变量 a 到当前工作空间中
format	函数 format 需要有额外参数才能达到修改显示数字精度的效果。如: format long 表示对 double 型数据显示 15 位小数位数,对 single 型数据显示 7 位小数位数

在使用 load 函数过程中,如果 1.mat 文件不存在,那么 load 函数会提示是错误的。函数 format 也不只 format long 这一种用法,更多具体使用细则可以到 MATLAB 帮助文档中搜索。

1.3.2 当前目录窗口

当前目录窗口(Current Folder)中显示的是当前 MATLAB 搜索路径下的文件内容,如图 1.3.2 所示。

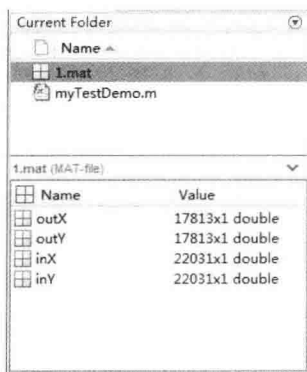


图 1.3.2 当前目录窗口

当前目录窗口中记录了所有文件,其中相关文件(如 m 文件、mat 文件等)还会在下侧的窗口中显示文件内包含的一些信息,不打开文件就能显示和查看一部分内容。

依据不同的需要,用户可以根据自己的喜好选择存放目录,进行文件存储和路径设置。如果想修改当前路径,菜单栏下面有一个文本框,可以按当前计算机文件存储结构,逐级更改文件目录。

每更改一次,左侧的当前目录窗口会及时更新显示当前更改后的目录及该目录下的文件夹、文件。此外,双击左侧当前目录窗口下的文件夹,也可以直接进入该文件夹所在目录。而双击 MATLAB 可识别的文件,则可以进入该文件中进行文件阅读状态,方便后续阅读和修改。

1.3.3 命令窗口

命令窗口(Command Window)是一个脚本命令的载体,里面输入 MATLAB 可以识别的命令,并将该命令显示在其中,用户按下回车键(Enter)后将执行该命令。上文提到的 clear、clc、save、load 等命令,都是在命令窗口中输入并完成其执行过程的。

如果需要输入多条命令,可以在每条命令后用英文分号“;”结束,并写成一行形式。如果希望输入为多行形式,可以通过 Shift+Enter 组合键进行下一行输入。

此外,当执行命令时,如果该命令结尾没有采用英文分号“;”结束,而该命令又有返回值,那么该返回值会被完整地显示在命令窗口中。图 1.3.3 表明了采用英文分号“;”结束和不采用时在显示上的区别。

在显示数值时,由于存在显示精度的问题,系统会采用默认短精度的显示方式,即 format short 的显示方式。该方式将只显示到小数点后 4 位。当用户想查看更多有效数

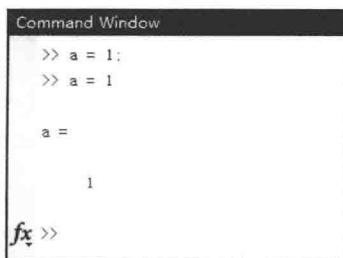


图 1.3.3 命令窗口

字时,只需在命令窗口中输入 `format long`,按下 Enter 键后,再显示一次需要查看的变量,即可得到更多的有效数字。

需要注意,显示精度与计算精度并不是同一个概念。在计算过程中,计算精度随着计算类型的不同而不同,如果没有设置计算类型,MATLAB 会默认精确度最高的双精度(double)作为计算类型使用。显示精度是在计算精度的基础上,用于显示在命令窗口中的精度。

1.3.4 工作空间

工作空间(Workspace)是命令运行后,集中显示生成变量的窗口。每个变量由其变量名、数据类型、数据大小、值组成。当在命令窗口中执行语句时,每条语句的执行结果将被记录在当前的工作空间中。如下所示:

```
>>a = 'MATLAB'
a =
MATLAB
>>
```

上述语句的意思是将字符串 MATLAB 赋值给变量 a。在命令窗口执行语句后,工作空间

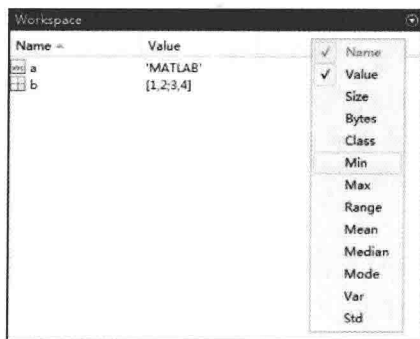


图 1.3.4 工作空间

就会出现一个变量 a,双击该变量,会进入变量窗口,可以看到 a 的变量名是 a,数据大小是 1×6 ,数据类型为 char,a 的值是 MATLAB。

工作空间中除了保存变量的值 Value,还保存了该记录变量的一些基本属性,如图 1.3.4 所示,分别有 Size(形状)、Bytes(大小)、Class(类型)、Min(最小值)、Max(最大值)、Range(范围)、Mean(平均值)、Median(中值)、Mode(最高出现频率值)、Var(方差)、Std(标准差)。最小值 Min 和最大值 Max 仅在变量类型为数值时起作用。

工作空间中的变量可以通过 mat 文件形式,随时保存到计算机的某个位置,也就是上文提及的 save 函数。若要从 mat 文件中读取变量,则可以用 load 函数进行加载。save、load 函数的详细用法可参见帮助文档。后续会对帮助文档的使用作更进一步的解释。

1.3.5 命令历史窗口

命令历史窗口(Command History)用于记录命令窗口中曾经存在的所有命令。如“a = 'MATLAB';”命令,就记录在命令历史窗口中,如图 1.3.5 所示。



图 1.3.5 命令历史窗口

“5x”表示当前命令被连续执行的次数,执行 1 次则默认忽略。此处,命令“a = 'MATLAB';”被连续执行了 5 次。

用户可以查看历史命令,并复制到当前命令窗口,这样可提高输入命令的速度。通常,对

于单行需要复制的历史命令,可通过单击并用鼠标拖拽的方式拉入当前命令窗口。对于多行非连续的历史命令,可以通过 Ctrl 键进行选择;对于多行连续的历史命令,可以通过单击命令第一行并按住 Shift 键后再单击最后一行的方式,进行选择。

用鼠标拖拽到当期命令窗口后的命令,可以对已有的内容进行修改,再按 Enter 键执行。

1.3.6 编辑界面

编辑界面(Editor)用于编辑执行脚本、函数或类,并将其保存在 m 文件中,方便重复调用。新建的 m 文件一般是空文件,里面依据用户需要编写不同的执行语句,组成执行命令,也可以编写为脚本形式(见图 1.3.6)、函数形式(见图 1.3.7)。

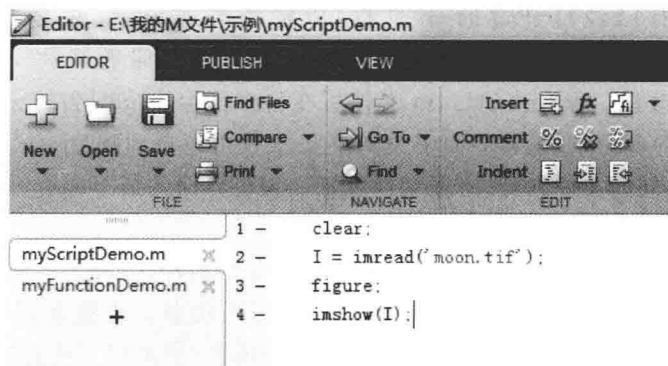


图 1.3.6 脚本编辑界面

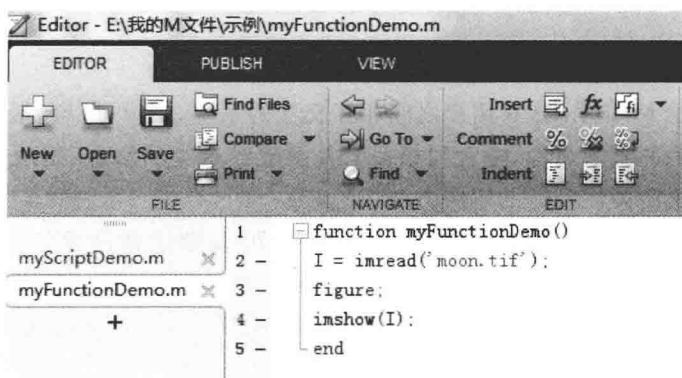


图 1.3.7 函数编辑界面

图 1.3.6 显示的是脚本,图 1.3.7 显示的是函数,两者的区别如下:

(1) 使用脚本时,直接面对的是当前工作空间。如果当前工作空间中存在变量 I,同时又没有首先执行 clear 操作,则很容易发生执行错误。所以,通常需要 clear 清除当前工作空间中的所有变量,防止干扰到后续执行语句。而使用函数则不用担心这种情况,函数执行时采用的工作空间为临时的工作空间,与当前工作空间不同,函数与函数之间调用时也会使用共用的一个临时工作空间。函数与当前工作空间之间的数据交换,是通过函数的输入和输出来完成的。

(2) 执行脚本是将脚本中的所有语句依次在命令窗口中执行,而执行函数则是只执行对应的函数语句,函数内的语句不在命令窗口中体现,在函数内部执行。

此外,还可以建立类的 m 文件。它与脚本、函数的区别在于,类作为功能相关的函数集