

数学实验教程

(小学教育专业适用)

刘 鹏 编著

清华大学出版社

北 京

内 容 简 介

本书紧密对接高等院校小学教育专业(理科方向)人才培养目标,以小学教育专业“数学实验”课程的核心知识体系为框架,由浅入深、循序渐进地展开讲解,精心设计了一系列数学实验项目,内容主要包括MATLAB入门知识、曲线绘图、极限、导数、级数、积分、常微分方程、迭代法、 π 的近似计算、线性方程组、微积分中的误差分析、平面图形的几何变换、公平分配问题、银行储蓄问题、抛物线穹顶问题、随机模拟问题。每个实验单元包括实验目的、实验内容、实验准备、实验方法与步骤、练习题,并穿插“试一试”“思考”版块。本书既可作为高等院校小学教育专业学生的教材,也可作为小学数学教师的教学参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。举报:010-62782989, beiqinquan@tup.tsinghua.edu.cn。

图书在版编目(CIP)数据

数学实验教程:小学教育专业适用/刘鹏编著.

北京:清华大学出版社,2026.1.--ISBN 978-7-302

-70485-0

I. G623.502

中国国家版本馆 CIP 数据核字第 2025Q9972W 号

责任编辑:高 岫

封面设计:周晓亮

版式设计:思创景点

责任校对:成凤进

责任印制:丛怀宇

出版发行:清华大学出版社

网 址: <https://www.tup.com.cn>, <https://www.wqxuetang.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-83470000

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:天津鑫丰华印务有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:9.25

字 数:208 千字

版 次:2026 年 1 月第 1 版

印 次:2026 年 1 月第 1 次印刷

定 价:45.00 元

产品编号:114195-01

前言

党的二十大报告统筹部署“科教兴国战略、人才强国战略、创新驱动发展战略”，并明确指出，“教育、科技、人才是全面建设社会主义现代化国家的基础性、战略性支撑。”同时，《教育强国建设规划纲要(2024—2035 年)》进一步要求深化教育改革创新，构建支撑中国式现代化的人才培养体系。在此背景下，义务教育阶段数学实验课程应当坚持以核心素养为导向，培养学生适应终身发展和社会需求的关键能力、必备品格与正确价值观。

本书面向小学教育专业学生，响应新时代对教师队伍建设的战略要求，将 MATLAB 科学计算软件引入数学实验教学领域，通过创新性实验资源，帮助师范生深入理解数学知识的本质，掌握信息化教学辅助工具，提升实践能力与创新素养；在数学案例设计中注重融入科学精神，推动小学数学教育从“知识传授”向“素养培育”方向转型，为培养具有坚定信念、高尚师德和卓越专业能力的新时代高素质、专业化小学数学教师贡献力量。

MATLAB 是一款集数值计算、符号运算、图形绘制和编程功能于一体的科学计算软件，简单易用、功能强大，广泛应用于诸多领域。MATLAB 能将抽象的数学概念和复杂的数学问题转化为直观的图形和易懂的代码，从而为小学教育的数学实验教学提供有力支持。通过 MATLAB，学生可更轻松地执行数学建模、数据分析、几何绘图等操作，更好地理解数学知识的本质和应用价值，提高学习数学的兴趣和积极性。

本书紧密贴合小学教育专业培养目标，围绕小学数学课程的核心知识体系，精心设计了一系列数学实验项目。这些实验项目融合了直观演示、动手操作、小组合作、问题探究等多种教学方法，旨在全方位提升未来小学数学教师的教学能力与素养。这些实验项目不仅注重学生对数学知识的理解和掌握，更强调培养学生的数学思维能力、实践操作能力和创新意识，使学生能将所学的数学知识灵活运用到实际问题中，提高解决实际问题的能力。

在编写过程中，本书充分考虑小学教育专业学生的学习特点与未来职业需求，注重强化与小学数学的联系，每一部分都选取了符合小学数学教学实际或贴近学生生活的情景问题，以激发学生的学习兴趣。在内容取舍上，本书注重与理工科专业的区别：不追

求深奥的理论推导，舍弃晦涩的理论内容和计算技巧，利用 MATLAB 的优势，帮助学生直观理解数学的思想方法。

本书注重理论与实践的紧密结合。在数学实验部分，做了精心安排：①明确实验要达到的知识目标和能力目标，使学生弄清楚实验的重点和方向；②简述相关的数学理论基础，为实验操作提供必要的理论支持；③介绍利用 MATLAB 进行实验操作的具体步骤和方法，并通过实验案例演示整个过程，帮助学生更好地理解和掌握实验内容和操作技巧；④设置一些拓展性的实验任务，引导学生进一步探索和思考，培养学生的自主学习能力和创新思维能力。

本书强调学生的主体地位与自主探索精神，鼓励学生在实验过程中积极思考、大胆创新，提出自己的见解与改进方案，为未来的小学数学课堂注入源源不断的创意与活力。

本书提供丰富的教学资源，包括教学课件、教案、教学大纲、学时进度表、习题及答案等(请扫描右侧的二维码下载)，以方便教师教学。本书可作为小学教育专业本科生“数学实验”课程的教材或参考书，建议的课时为 36 课时。限于编者水平，本书必定有不妥之处，望广大读者批评和指正。清华大学出版社的编辑认真审稿，并提出了许多改进意见，在此表示衷心感谢。



教学资源

希望通过本书的学习，小学教育专业的学生能深刻认识到数学实验在小学数学教学中的重要价值和独特作用，熟练掌握 MATLAB 这一教学工具，积极将其应用于未来的小学数学教学实践中。借助数学实验教学，激发学生对数学的学习兴趣，培养学生的数学核心素养，提高小学数学教学质量，为我国义务教育阶段数学教育事业的发展贡献一份力量。

编 者

2025 年 12 月

目 录

第一篇 MATLAB 基础

第一章 MATLAB 入门知识	3
一、历史背景	3
二、语言特点	4
三、操作环境及基本功能	5
四、数组与矩阵	7
五、获取帮助	12
六、矩阵运算	13
七、M 文件	16
八、匿名函数	18
九、控制流	18
练习题	20

第二篇 数学实验

第二章 曲线绘图	23
一、绘制函数 $y=(x-1)(x+1)(x+2)$ 的图形	24
二、绘制摆线 $x=2(t-\sin t)$, $y=2(1-\cos t)$, $t \in [0, 2\pi]$	26
三、抛物线水柱的设计	27
练习题	30
第三章 极限与导数	31
一、计算极限 $\lim_{x \rightarrow 2} \frac{x^2+5}{x-3}$	33
二、判断 $\lim_{x \rightarrow 0} \sin \frac{1}{x}$ 的存在性	34

三、验证 $\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1, \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n = \lim_{x \rightarrow \infty} \left(1 + \frac{1}{x}\right)^x = e = 2.71828 \dots$	35
四、纸盒的最大体积	36
练习题	39
第四章 级数	41
一、麦克劳林级数展开	43
二、级数求和	44
三、验证泊松分布性质	45
四、披萨问题的解	45
练习题	46
第五章 积分	49
一、计算不积分 $\int (e^x - 2\sin x) dx$	51
二、计算定积分 $\int_0^1 \frac{1}{1+x} dx$	52
三、计算二重积分 $\iint_D xy dx dy$	52
四、验证标准正态分布性质	53
五、池塘的水面面积	53
练习题	54
第六章 常微分方程	55
一、dsolve 求解析解	59
二、求解微分方程	60
三、细菌问题	61
练习题	63
第七章 迭代法	65
一、计算数列极限	67
二、利用迭代法求解方程	67
三、用牛顿迭代法求方程 $xe^x = 1$ 的正根	68
四、求解兔子问题	69
练习题	69
第八章 π 的近似计算	71
一、用刘徽的割圆术计算 π 的近似值	74
二、用莱布尼茨级数计算 π 的近似值	74
三、用拉马努金公式计算 π 的近似值	75
四、高斯-勒让德法	76
练习题	78

第九章 线性方程组	81
一、解线性方程组	84
二、求不定方程组的特解	85
三、求超定方程组的最小二乘解	85
四、求奇异方程组的特解	86
五、求方程组的通解	87
六、求解鸡兔同笼问题	88
练习题	89
第十章 微积分中的误差分析	91
一、定积分误差的数值验证	92
二、微分和逼近增量的数值验证	95
练习题	96
第十一章 平面图形的几何变换	99
练习题	104

第三篇 应用实验

第十二章 公平分配问题	107
练习题	113
第十三章 银行储蓄问题	115
一、连续复利与等效利率	118
二、按揭贷款的实际利率	118
三、基金的“拆分定存”方案	119
练习题	121
第十四章 抛物线穹顶问题	123
练习题	127
第十五章 随机模拟问题	129
一、模拟摸球概率	130
二、模拟圆盘与地砖缝相交概率	131
三、模拟布丰投针试验	132
四、转盘问题求解	134
练习题	136
参考文献	137

第一篇

MATLAB基础

第一章 MATLAB入门知识

第一章 MATLAB 入门知识

本章介绍 MATLAB 环境、矩阵操作、基本运算和 MATLAB 编程等方面的基础知识，引导读者熟悉 MATLAB 操作环境，了解软件操作和可视化基础，为后续实验服务。

MATLAB(Matrix Laboratory, 矩阵实验室)是由美国 MathWorks 公司开发的高科技计算环境，集数值计算、可视化、交互式程序设计及多种建模与仿真功能于一体，为科学研究、工程设计和需要进行高效数值计算的诸多领域提供全面的解决方案。它显著区别于传统非交互式语言(如 C、Fortran)的编程模式，代表了当今国际科学计算软件的先进水平。

一、历史背景

20 世纪 70 年代中期，Cleve Moler 博士及其团队在美国国家科学基金资助下开发了调用 EISPACK(特征值求解库)和 LINPACK(线性方程求解库)的 FORTRAN 子程序接口。20 世纪 70 年代后期，时任美国新墨西哥大学计算机系主任的 Cleve Moler 教授，为简化学生使用 EISPACK 和 LINPACK 的过程，利用业余时间开发了接口程序，并命名为 MATLAB(取自 Matrix 和 Laboratory 的前三个字母)。1983 年，Cleve Moler 在斯坦福大学讲学，工程师 John Little 认识到 MATLAB 在工程领域的巨大潜力。同年，Moler、Little 和 Steve Bangert 合作，用 C 语言开发了第二代专业版 MATLAB，其具备了数值计算和数据可视化功能。1984 年，MathWorks 公司成立，正式将 MATLAB 推向市场并持续开发。

在众多数学类科技软件中，按内核可分为以下两类。

(1) 数值计算型：如 MATLAB、Xmath、Gauss，擅长处理大批量数据和高效率数值计算。

(2) 数学分析型：如 Mathematica、Maple，擅长符号计算、解析解和高精度解的计算，处理大数据效率较低。

MathWorks 公司融合了数值计算与可视化的核心优势，率先拓展了符号计算、文字处理、可视化建模和实时控制等专业功能，开发出适应多学科需求的 MATLAB。经过多年发展，MATLAB 已成为数值型软件市场的主导者。

在 MATLAB 出现前，许多应用软件直接基于 FORTRAN/C 等语言开发，存在使用面窄、接口简陋、结构封闭、缺乏标准库等问题，难以适应学科发展。MATLAB 的出现为各学科软件开发提供了新基础。20 世纪 80 年代中期，控制领域原有软件包纷纷被淘汰或在 MATLAB 上重建。

如今，经过 MathWorks 的不断完善，MATLAB 已发展成为功能强大的多学科、多平台大型软件。在欧美高校，它已成为线性代数、自动控制理论、数理统计、数字信号处理、时间序列分析、动态系统仿真等课程的基本教学工具，是理工科学生必须掌握的基本技能。在研究机构和工业界，MATLAB 被广泛应用于科学研究和解决实际问题。

二、语言特点

MATLAB 迅速普及并保持旺盛生命力的核心在于其简洁性，以下是 MATLAB 语言的主要特点。

(1) 简洁紧凑，灵活易用，库函数丰富：程序书写自由，利用丰富的预编译库函数避免烦琐的子程序编写，压缩不必要的工作。库函数由领域专家编写，可靠性高。

(2) 运算符丰富：基于 C 语言开发，提供了几乎与 C 语言一样多的运算符，灵活运用可使程序极为简短。

(3) 结构化与面向对象兼备：支持 for/while 循环、break，支持 if 等结构化控制语句，也具备面向对象编程特性。

(4) 语法限制宽松，编程自由度高：例如，使用矩阵前无须预定义。

(5) 可移植性好：程序基本上不必修改即可在不同计算机和操作系统上运行。

(6) 拥有强大的图形功能：数据可视化简单高效，具备较强的图形用户界面(GUI)编辑能力。

(7) 拥有功能强大的工具箱：由核心部分和众多可选工具箱组成。核心部分包含数百个核心内部函数。工具箱则分为两种。一种是功能性工具箱，具有扩展符号计算、建模仿真、文字处理、硬件交互等功能，适用于多学科；另一种是学科性工具箱，如 Control Toolbox(控制)、Signal Processing Toolbox(信号处理)、Communications Toolbox(通信)等，由领域顶尖专家编写，用户可直接进行高层次研究。

(8) 具有源代码开放性：除少数内部函数外，核心文件和工具箱文件均为可读、可改的源代码。用户可修改源码或添加自定义文件来构建新工具箱。

此外，MATLAB 语言也有自身缺点，比如执行速度相对较慢(缺点)。这是因为作为解释型语言，其无须编译预处理，不生成独立可执行文件，所以执行速度通常慢于编译型语言(如 C/C++)。

三、操作环境及基本功能

启动 MATLAB 后，MATLAB 操作界面如图 1-1 所示。

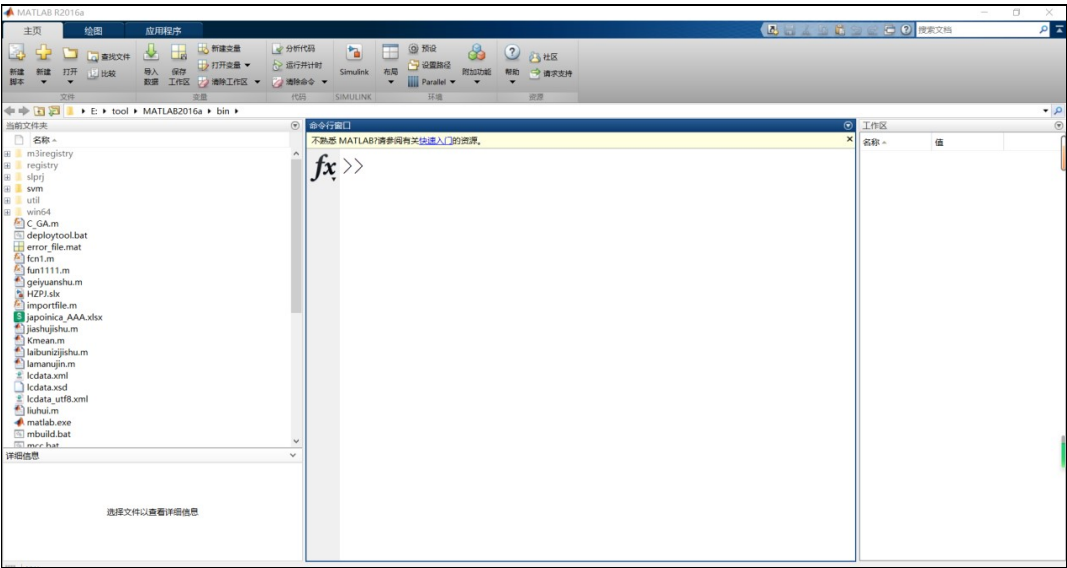


图 1-1 MATLAB 操作界面

主要的界面区域如下。

- 命令窗口(Command Window): 输入命令并显示计算结果的主要窗口。
- 编辑器(Editor): 用于编写和编辑 M 文件(脚本和函数)。
- 工作区(Workspace): 显示当前定义的变量及其信息。
- 命令历史(Command History): 记录已使用过的命令。
- 图形窗口(Figure Window): 显示绘制的图形(绘图时自动弹出)。

MATLAB 计算方式包括交互式计算器模式、变量表达式计算、编程模式三种。

(1) 交互式计算器模式: 直接在命令窗口输入数值和运算符，即时显示结果。

用户在提示符后键入命令，例如，输入“>>3+5”，表示输入运算式 3+5。其中，“>>”是 MATLAB 提示符，表示 MATLAB 已准备好接收用户输入。换行输入时用“...”，表示继续输入剩余部分。上面“>>3+5”也可以多行输入，如：

```
>>3+...  
5
```

一般地，多行输入在代码太多，一行写不下时使用；有时，为了方便阅读，也

可以主动使用多行输入方式。

(2) 变量表达式计算：先给变量赋值，再输入由变量构成的表达式(如 $x=5;y=3;z=x^2+y$)。

(3) 编程模式：编写 M 文件(脚本或函数)解决包含判断、循环、迭代、递归等算法的复杂问题。

MATLAB 计算时需要使用运算符，基本算术运算符包括加(“+”)、减(“-”)、乘(“*”)、除(“/”)、幂(“^”)。

有时还需要进行关系运算和逻辑运算，运算符如表 1-1 所示。

表 1-1 关系、逻辑运算符及其他常用运算符

运算符	功能
=	赋值运算
==	关系运算，相等
< >	不等于
<	小于
<=	小于等于
>	大于
>=	大于等于
%	注释标志
&	逻辑与运算
	逻辑或运算
-	逻辑非运算
xor	逻辑异或运算
...	续行标志
,	分行符，结果不显示
;	分行符，结果显示
'	矩阵转置
.'	向量转置

在 MATLAB 中，一个等号“=”是赋值符号，表示把等号右边的表达式、结果赋值给左边的对象；数学上的相等是“==”。“%”是注释标志，放在命令后面，表示对符号左边的代码进行解释，帮助阅读者理解。注释的内容不是代码，如果注释内容太多，需要跨行，可以使用多行注释符“%{...%}”。一些命令中也使用“%”，如 `fprintf('π 精确值=%.15f\n',pi);`，这时“%”是格式占位符。

MATLAB 算术运算的结果显示可由 `format` 命令来控制，表 1-2 给出了数值显示格式控制命令的说明。

表 1-2 数值显示格式控制(format 命令)

命令	说明
format short	默认显示, 保留小数点后 4 位
format long	有效数字 16 位
format long e	有效数字 16 位加 3 位指数
format short e	有效数字 5 位加 3 位指数
format bank	保留两位小数位
format +	只给出正、负
format rational	以分数形式表示
format hex	16 进制数
format long g	15 位有效数
format short g	5 位有效数

MATLAB 默认显示结果保留小数点后 4 位。

在使用变量之前, 需要对变量命名, 变量命名规则具体如下。

- 区分大小写(如 apple、Apple、AppLe 是三个不同的变量)。
- 变量名长度上限为 63 个字符(MATLAB R2019a 及更新版本); 早期版本中, 长度上限为 31 或 19。
- 必须以字母开头, 后面可跟字母、数字或下画线(_)。

在运算的时候会使用特殊变量, 如表 1-3 所示。

表 1-3 特殊变量

变量名	意义
help	在线帮助, 如 help quit
who	列出所有定义过的变量名称
ans	默认的用来表示计算结果的变量名
eps	极小值(2.2204e-16)
pi	π 值
inf	无穷大的数(∞)
nan	非数值

使用 MATLAB 时遇到最多的是 ans, 输出结果没有赋值时, MATLAB 默认使用 ans 来表示; 计算的结果(尤其是中间结果)还需要参与其余运算时, 最好都进行赋值, 将其值用变量保存下来。

四、数组与矩阵

在线性代数中, 矩阵表示一个多行多列的数表, 单行、单列的称为向量; 在 MATLAB 中, 矩阵一般称为数组。MATLAB 尤其擅长数组(array)和矩阵(matrix)

运算。数组强调元素对元素的运算，矩阵则遵循线性代数规则。数组通常是一维的(向量)，矩阵是多维的(最常见是二维)。

1. 创建矩阵

MATLAB 是以矩阵为主要操作对象的软件，所以首先要学会对矩阵进行输入和访问。创建矩阵的方法一般有以下几种。

方法一：使用函数创建特殊矩阵

表 1-4 显示了生成特殊矩阵的函数及相关说明。

表 1-4 生成特殊矩阵的函数及相关说明

函数符号	说明
<code>zeros(i,j)</code>	创建 i 行 j 列的全零矩阵
<code>ones(i,j)</code>	创建 i 行 j 列的全 1 矩阵
<code>eye(i,j)</code>	创建 i 行 j 列对角线为 1 的矩阵
<code>rand(i,j)</code>	创建 i 行 j 列的随机矩阵

在计算中，若有两个结果，则可以用 `a`、`b` 来保存；若有 10 个结果，则可以事先创建全零矩阵来保存这些值。特殊矩阵多用于预设变量存储空间。

【例 1】创建 2 行 3 列全零矩阵

代码如下：

```
>>A=zeros(2,3) %>>表示后面是 MATLAB 代码，下同
```

输出结果如下：

```
A=
    0    0    0
    0    0    0
```

注意，MATLAB 的代码全部都是英文状态下的符号！使用中文字符时，需要用一对单引号括起来。

【例 2】创建 3×3 全 1 方阵

代码如下：

```
>>B=ones(3)
```

输出结果如下：

```
B=
    1    1    1
    1    1    1
    1    1    1
```

方法二：创建一维数组(向量)

冒号运算符“:”： $X=start:step:end$ 生成从 $start$ 开始，以 $step$ 为步长，到 end 结束的行向量； $step$ 为 1 时可省略。冒号运算符生成数学上的等差数列， $start$ 是首项， $step$ 是公差， end 是终项。

`linspace` 函数： $X=linspace(start,end,n)$ 在区间 $[start,end]$ 上生成 n 个等分点的行向量。

【例 3】创建一维数组

代码如下：

```
>>x=1:2:10    %起点为 1，步长为 2，终点为 10(包含 1,3,5,7,9)
```

输出结果如下：

```
X=
    1     3     5     7     9
```

若想在 $[1,3]$ 区间生成 5 个等分点，可输入如下代码：

```
>>Y=linspace(1,3,5)
```

输出结果如下：

```
Y=
    1.0000    1.5000    2.0000    2.5000    3.0000
```

方法三：直接输入任意矩阵

在输入任意矩阵时，将元素放在 $[]$ 内，行内元素用空格或逗号分隔，行间用分号“;”分隔，这种方法最常用。

【例 4】创建由数字 1~9 组成的 3x3 矩阵 A

代码如下：

```
>>A=[1 2 3;4,5,6;7 8,9]    %空格、逗号分隔列，分号分隔行
```

输出结果如下：

```
A=
    1     2     3
    4     5     6
    7     8     9
```

方法四：使用数组编辑器创建大矩阵

遇到大矩阵时，方法三不太方便，可使用数组编辑器来创建，如图 1-2 所示。

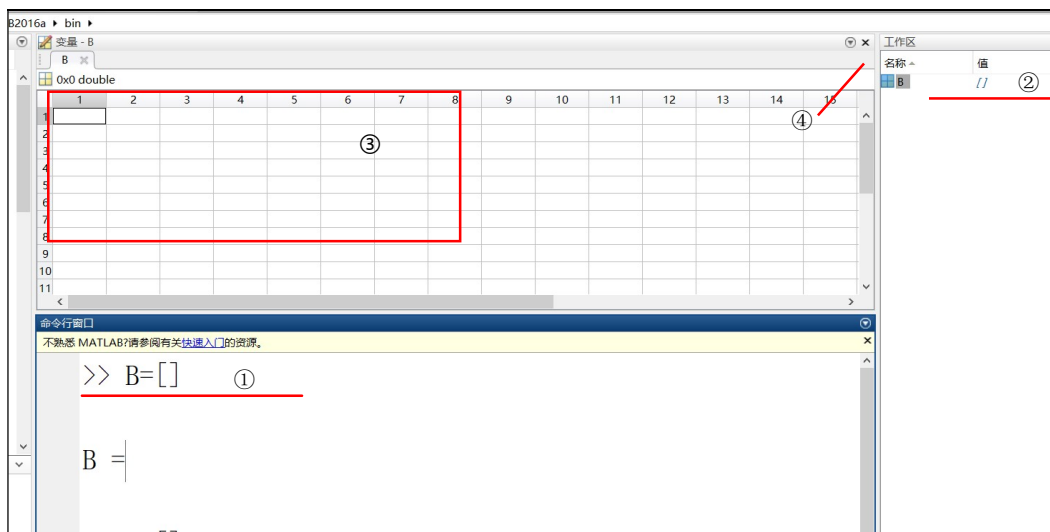


图 1-2 数组编辑器操作步骤

具体步骤如下。

- (1) 在命令窗口输入空矩阵：`>>B=[]`。
- (2) 在工作区窗口双击变量 **B**，打开数组编辑器。
- (3) 在编辑器中像操作 Excel 表格一样输入或修改矩阵元素。
- (4) 关闭编辑器窗口，矩阵 **B** 即被定义。

2. 访问矩阵元素

创建矩阵后，即可访问其元素，访问方法主要有以下两种。

方法一：直接用变量名访问整个矩阵进行运算

【例 5】对两个矩阵做加法运算

代码如下：

```
>>A=[1,2,3;4,5,6;7,8,9];
>>B=[3,2,1;6,5,4;9,8,7];
>>C=A+B                                %计算矩阵 A 与 B 的和 (按元素相加)
```

输出结果如下：

```
C=
     4     4     4
    10    10    10
    16    16    16
```

方法二：访问特定元素或子集

与线性代数中的做法一样，MATLAB 用双下标索引访问矩阵的元素，表 1-5 展示了访问特定元素的命令及相关说明。

表 1-5 访问特定元素的命令

命令	说明
A(i,j)	访问第 i 行第 j 列的元素
A(i,:)	访问第 i 行的元素
A(i,j:k)	访问第 i 行中第 j 列到第 k 列的元素
A(:,j)	访问第 j 列的所有元素(整列)
A(:,j:k)	访问第 j 到 k 列的所有元素
A([i1,i2],[j1,j2])	访问行号为[i1,i2]、列号为[j1,j2]的元素(更通用)

【例 6】访问前面输入的矩阵 A 的元素

代码如下：

```
>>A(2,1:2) %访问第 2 行中，第 1 到第 2 列的元素
```

输出结果如下：

```
ans= %ans 为默认的变量名
    4    5
```

若要访问第 2 行中，第 1、3、2、1 列的元素，可输入如下代码：

```
>>A(2,[1,3,2,1]) %访问第 2 行中，第 1,3,2,1 列的元素(按指定顺序)
```

输出结果如下：

```
ans=
    4    6    5    4
```

【试一试】输入下面的代码，结果会是什么？

```
>>A([1,3,2],[1,3,2,1])
```



1-1 结果

3. 访问数组元素

可使用单下标索引访问数组。

【例 7】访问多个数组元素，并计算它们的和

代码如下：

```
>>X=[1,3,4,2,5,7,3,9];
>>D=X(2)+X(5)+X(7) %计算第 2 个元素(3)+第 5 个元素(5)+第 7 个元素(3)的和
```

输出结果如下：

```
D=
    11 % 结果应为 11
```

五、获取帮助

在使用 MATLAB 时，会遇到许多困难，可通过多种方式获取帮助，主要有以下几种。

命令补全(Tab 键)：输入函数或变量名的前几个字符，然后按 Tab 键，MATLAB 会列出可能的补全项。例如，输入 plo 后按 Tab 键，会看到 plot 等建议(见图 1-3)，其目的是提高输入效率和准确性，避免拼写错误。

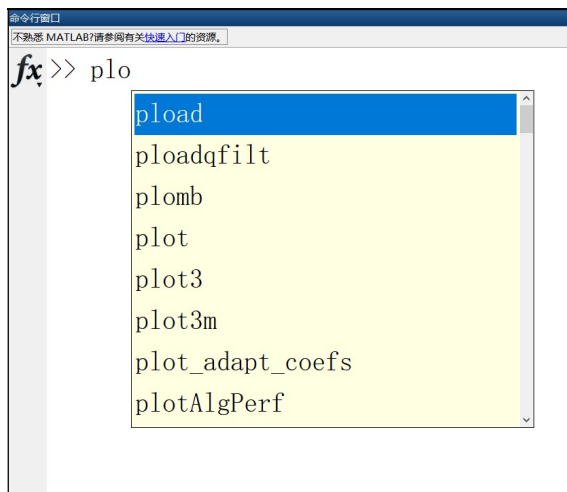


图 1-3 使用 Tab 键获取帮助

help 命令：>>help functionName 在命令窗口显示指定函数、命令的简洁帮助信息。例如>>help plot 会将命令 plot 文件中的注释部分调出，结果如图 1-4 所示。

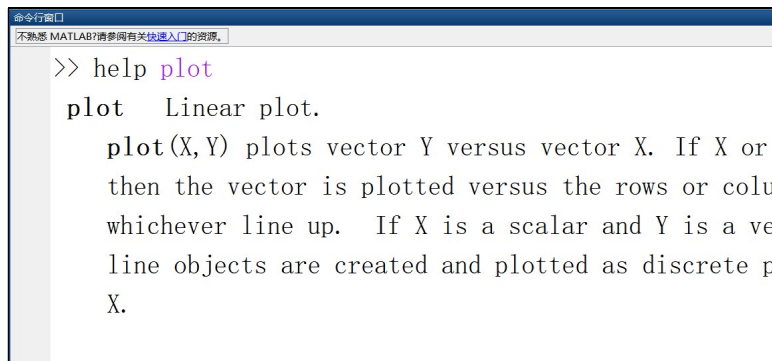


图 1-4 help plot 的输出结果

doc 命令：>>doc functionName 打开更详细、格式化的帮助文档页面(包含示例)。可以试试：>>doc plot。

网络搜索：使用搜索引擎(如百度、谷歌)查询 help、plot、sinx 等关键词，可以找到大量教程、示例和论坛帖子。

六、矩阵运算

下面列举一个例子，来展示矩阵的运算。

【例 8】矩阵的运算

已知矩阵：

$$A = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 6 & 8 \\ 5 & 3 & 9 \end{bmatrix}, \quad B = \begin{bmatrix} 9 & 5 & 3 \\ 8 & 1 & 6 \\ 4 & 2 & 7 \end{bmatrix}$$

求：

- (1) 屏幕输出 A 与 B 。
- (2) A 的转置 A' 。
- (3) 矩阵的加法： $A+B$ 。
- (4) 矩阵的减法： $A-B$ 。
- (5) 矩阵的数乘： $6A$ 。
- (6) 矩阵的乘法： AB 。
- (7) A 的逆矩阵： A^{-1} 。
- (8) 矩阵的点乘： $A.*B$ 。

解：

- (1) 屏幕输出 A 与 B 。

屏幕输出 A 的代码如下：

```
>>A=[1 4 7;2 6 8;5 3 9]    %输出 A
```

输出结果如下：

```
A=
     1     4     7
     2     6     8
     5     3     9
```

屏幕输出 B 的代码如下：

```
>>B=[9 5 3;8 1 6;4 2 7]    %输出 B
```

输出结果如下：

```
B=
     9     5     3
     8     1     6
     4     2     7
```

(2) A 的转置 A' 。

将 A 转置为 A' 的代码如下：

```
>>A'      %A 的转置 (复共轭转置, 对于实数矩阵而言, 等同于转置)
```

输出结果如下：

```
ans=
     1     2     5
     4     6     3
     7     8     9
```

(3) 矩阵的加法： $A+B$ 。

矩阵和的代码如下：

```
>>A+B      %矩阵加法 (对应元素相加)
```

输出结果如下：

```
ans=
    10     9    10
    10     7    14
     9     5    16
```

(4) 矩阵的减法： $A-B$ 。

矩阵差的代码如下：

```
>>A-B      %矩阵减法 (对应元素相减)
```

输出结果如下：

```
ans=
    -8    -1     4
    -6     5     2
     1     1     2
```

(5) 矩阵的数乘： $6A$ 。

矩阵数乘的代码如下：

```
>>6*A      %标量乘法 (矩阵的每个元素乘以 6)
```

输出结果如下：

```
ans=
     6    24    42
    12    36    48
    30    18    54
```

矩阵的和、差与数乘运算，实质上是对应的元素参加运算。

(6) 矩阵的乘法： AB 。

矩阵的乘法遵循线性代数规则运算，其代码如下：

```
>>A*B      %矩阵乘法
```

输出结果如下：

```
ans=
    69    23    76
    98    32    98
   105    46    96
```

(7) A 的逆矩阵： A^{-1} 。

代码如下：

```
>>C=inv(A)   %求 A 的逆矩阵
```

输出结果如下：

```
C=
   -0.6000    0.3000    0.2000
   -0.4400    0.5200   -0.1200
    0.4800   -0.3400    0.0400
```

【思考】逆矩阵存在的条件是什么？

(8) 矩阵的点乘： $A.*B$ 。

这是矩阵的点乘运算，代码如下：

```
>>D=A.*B    %数组乘法(点乘,对应元素相乘)
```

输出结果如下：

```
D=
     9    20    21
    16     6    48
    20     6    63
```

【试一试】试试下面的命令：

```
>>A^3      %矩阵 A 的 3 次幂 (A*A*A)
```

```
>>A.^3     %数组求幂(点幂,每个元素自身的 3 次方)
```

【思考】什么是点运算？



1-2 结果



思考题
答案 1-1

七、M 文件

MATLAB 生成的文件的扩展名是“.m”，称为 M 文件。在 MATLAB “主页”选项卡中，单击“新建”按钮并选择“脚本”或“函数”，即可打开编辑器编写 M 文件。

1. 函数 M 文件

函数 M 文件的作用就是定义一个函数，以 `function` 开头。

基本格式为：

```
function [outputs]=functionName(inputs)
...
end
```

调用时，使用 M 文件名(不含.m 扩展名)，而非 `function` 后的内部函数名，强烈建议使 M 文件与函数同名。

【例 9】创建函数 $y=2x+1$

计算时需要用到函数 $y=2x+1$ ，则单击“新建”按钮并选择“函数”，打开函数 M 文件编辑器(见图 1-5)，在其中输入如下代码。

```
%函数 M 文件:fun.m
function y=fun(x)    %函数定义：输入 x,输出 y
    y=2*x+1;         %函数体：计算 y 的值
end
```

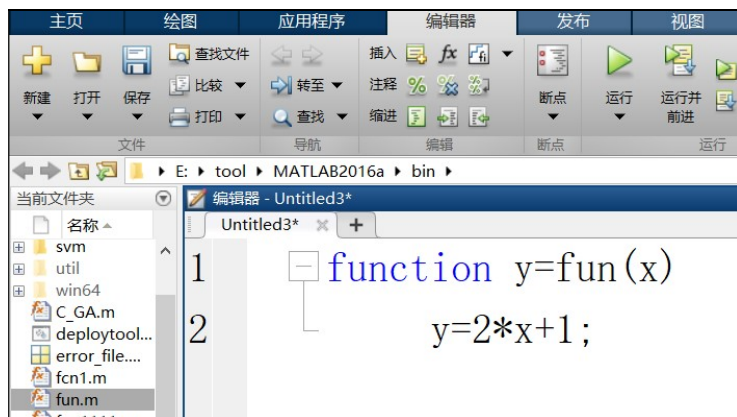


图 1-5 函数 M 文件编辑器

将上述代码保存为 M 文件，将该文件命名为 `fun.m`(特别注意，此处文件名与代码中的函数名保持一致)，并在命令窗口输入以下代码进行调用。

```
>>clear;    %清除工作区变量
>>fun(2)    %调用函数 fun，输入参数 2，就是计算 2×2+1
```

输出结果如下：

```
ans=
    5
```

输入以下代码：

```
>>fun(4)           %调用函数 fun，输入参数 4
```

输出结果如下：

```
ans=
    9
```

这样就定义了函数，并成功调用了该函数计算 `fun(2)`、`fun(4)` 的值。

2. 脚本 M 文件

用 `function` 开头的 M 文件不是脚本 M 文件，脚本 M 文件包含一系列按顺序执行的 MATLAB 命令。

【例 10】创建一个脚本 M 文件

下面将创建脚本 M 文件 `Mydoc.m`，将 4 条命令放在该文件中。单击“新建”按钮并选择“脚本”，打开脚本 M 文件编辑器(见图 1-6)，在其中输入如下代码。

```
%脚本 M 文件:Mydoc.m
a=3;
b=5;
z=a*b+3;      %计算 z 的值
disp(z);      %显示 z 的值 (disp 显示变量内容)
```

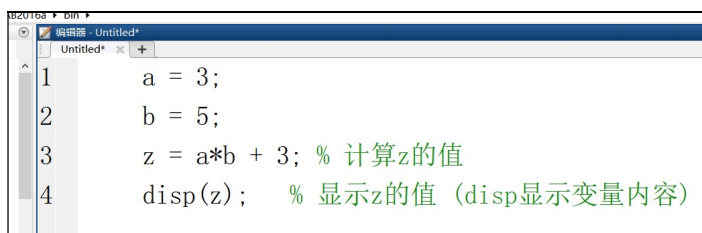


图 1-6 脚本 M 文件编辑器

将上述代码保存为文件名是 `Mydoc.m` 的 M 文件，并在命令窗口输入以下代码进行调用。

```
>>Mydoc    %运行脚本 (输入文件名，不含 .m)
```

输出结果如下：

```
z=
    18
```

编写的脚本 M 文件就是 MATLAB 程序。一般地，命令行数超过 3 时，建议使用脚本 M 文件，以便修改代码、调试程序。

函数 M 文件与脚本 M 文件是不同的，主要区别如下。

(1) 输入/输出：M 函数通常有明确的输入参数和输出返回值，而 M 脚本没有输入参数，输出主要依靠在工作区生成变量或在屏幕显示。

(2) 变量作用域：M 函数内部定义的变量是局部变量，仅在函数内部有效，函数执行完毕后这些变量会被清除，不影响命令窗口或其他函数的工作区。M 脚本中定义的变量存在于基础工作区(即命令窗口的工作区)，是全局可见的。脚本执行完毕后，这些变量依然存在。

八、匿名函数

MATLAB 提供了一种无须创建独立 M 文件即可快速定义简单函数的方法。

匿名函数的定义格式为

$$fhandle=@(arglist) \text{expr}$$

其中，fhandle 为函数句柄，用于调用该匿名函数，@为创建函数句柄的操作符，(arglist)为输入参数列表(逗号分隔)，expr 为具体的函数表达式。

【例 11】创建函数 $y=2x+1$

代码如下：

```
>>f1=@(x) 2*x+1;      %定义匿名函数 f1
>>a=f1(2),b=f1(4)      %调用 f1 计算
```

输出结果如下：

```
a=
    5
b=
    9
```

【试一试】用匿名函数的方式定义一个二元函数：

$f(x,y)=2x+3y-2$ 。



1-3 结果

九、控制流

在使用 MATLAB 计算时，需要用到一些控制结构。

常用的控制流结构有以下几种。

for 循环: for loopVar=start:step:end ... end

while 循环: while condition ... end

if 条件: if condition ... end

if-elseif-else 条件: if cond1 ... elseif cond2 ... else ... end

switch-case-otherwise 分支: switch expr case val1 ... case val2 ... otherwise ... end

break: 立即终止最内层循环的执行。

return: 立即终止当前函数或脚本的执行并返回。

要了解具体用法, 可扫描右侧二维码自行查看。



MATLAB 中
常用控制流
语句的详细
说明及示例

【思考】 for 循环和 while 循环的区别是什么?



思考题
答案 1-2

【例 12】 计算级数 $s = \sum_{n=1}^{100} \frac{1}{n^2}$

代码如下:

```
>>clear;           %清除工作区
>>s=0;             %初始化累加器 s 为 0
>>for n=1:100       %n 从 1 循环到 100, 步长默认为 1
    s=s+1/(n*n);    %通项累加
>>end
>>s                %显示最终结果 s
```

输出结果如下:

```
s=
    1.6350           %近似值 (  $\pi^2/6 \approx 1.6449$  )
```

即级数的值为 1.6350。这里使用了 for 循环结构来求解问题。

一般地, 求级数 $s = \sum_{k=m}^n f(k)$ 的通用代码模板(累加器初始化模式)如下:

```
s=0;               %初始化累加器 s 为 0
for k=m:step:n     %k 从 m 循环到 n, 步长为 step (step 默认为 1)
    s=s+f(k);       %将第 k 项 f(k) 累加到 s 上
end
```

【试一试】计算 $10!$

【思考】在使用 MATLAB 进行编程时，如何体现“严谨求实”的科学态度？

练习题

1. 编写一个 M 函数文件：当 $x < 0$ 时， $y = x + 3$ ；当 $x \geq 0$ 时， $y = x + 10$ 。
2. 用 for 循环语句计算 $s = 1 + 2 + 3 + \cdots + 100$ 。
3. 已知矩阵

$$A = \begin{bmatrix} 1 & 2 & 1 & 5 \\ 3 & 4 & 6 & 7 \\ 2 & 7 & 5 & 7 \\ 8 & 9 & 2 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 3 & 9 & 8 \\ 6 & 8 & 3 & 2 \\ 2 & 4 & 5 & 1 \\ 6 & 7 & 1 & 6 \end{bmatrix}$$

求：(1) A 的转置 A' 。

(2) 矩阵的加法： $A + B$ 。

(3) 矩阵的减法： $A - B$ 。

(4) 矩阵乘法： $A * B$ 。

(5) A 的逆矩阵 A^{-1} 。



1-4 结果



思考题

答案 1-3