

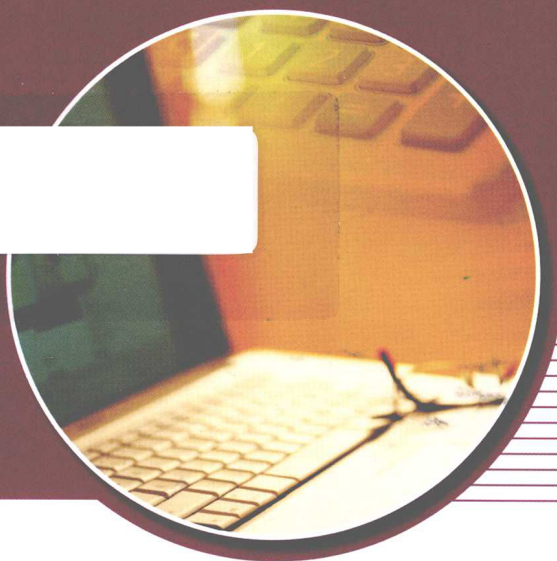
• 金融市场与风险管理系列教材 •

# 金融计算与编程

——基于MATLAB的应用

曹志广 著

*Matlab  
in Finance*

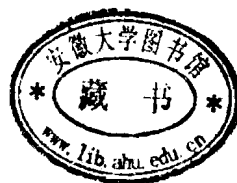


金融市场与风险管理系列教材

# 金融计算与编程

——基于 MATLAB 的应用

曹志广 著



 上海财经大学出版社

## 图书在版编目(CIP)数据

金融计算与编程:基于 MATLAB 的应用/曹志广著. —上海:上海财经大学出版社,2013.5

(金融市场与风险管理系列教材)

ISBN 978-7-5642-1622-1/F·1622

I. ①金… II. ①曹… III. ①金融-计算机辅助计算-高等学校-教材 IV. ①F830.49-39

中国版本图书馆 CIP 数据核字(2013)第 074033 号

- ☐ 责任编辑 台啸天
- ☐ 封面设计 张克瑶
- ☐ 责任校对 胡 芸 卓 妍

JINRONG JISUAN YU BIANCHENG

金融计算与编程

——基于 MATLAB 的应用

曹志广 著

---

上海财经大学出版社出版发行  
(上海市武东路 321 号乙 邮编 200434)

网 址: <http://www.sufep.com>

电子邮箱: [webmaster@sufep.com](mailto:webmaster@sufep.com)

全国新华书店经销

上海华教印务有限公司印刷

上海春秋印刷厂装订

2013 年 5 月第 1 版 2013 年 5 月第 1 次印刷

---

787mm×960mm 1/16 21.75 印张 462 千字

印数:0 001—1 000 定价:59.00 元

# 前 言

我从 2005 年开始,每年都在上海财经大学为硕士研究生开设《MATLAB 与金融计算》这门课程,算起来已八个年头了,后来这门课程更名为《MATLAB 在金融中的应用》。最初这门课程为选修课,后来成为金融工程专业的必修课。2005 年开设这门课程的时候,国内关于这方面合适的教材比较少,2007 年以后这方面的教材才开始多了起来。所以当时借鉴了一些国外的教材内容,再根据自己在金融研究中使用 MATLAB 的心得和体会,逐步开始编写课程的讲义,这些讲义逐步扩充,最后形成了读者现在见到的《金融计算与编程——基于 MATLAB 的应用》这本教材。从我个人体验角度而言,MATLAB 在我的研究,特别是实证研究中起到了特别大的帮助。MATLAB 在金融计算,尤其是在矩阵处理方面的优势,极大地节省了我处理和分析数据的时间。比如,处理上千个回归方程的系数估计,借助 MATLAB 在 1 分钟之内就能完成。我觉得对于经常要进行数据处理和分析的研究者或者从业人员而言,MATLAB 是一个非常好的助手。MATLAB 的编程语言和规则简单,不需要使用者有其他计算机语言的编程基础,非常容易入门。由于我长期在金融领域从事教学和科研,本书的内容局限于在金融领域中应用 MATLAB,因此,本书特别适合在金融领域从事研究的研究人员和在实际金融市场中应用金融理论进行量化投资分析的从业人员,以及有可能从事金融领域研究或应用的金融专业的学生阅读。

本书的主要内容:第一章到第三章的内容为基础篇,第四章到第十二章的内容为金融领域中的应用篇。需要提醒读者注意的是,本书中的所有函数均适用于 MATLAB7.1 版本。书中的某些函数可能无法在低版本的 MATLAB 环境下运行。

第一章介绍 MATLAB 的基本入门知识。这主要是为了方便不熟悉 MATLAB 的读

者能够在最短的时间内了解和熟悉 MATLAB,当然,熟悉 MATLAB 的读者可以跳过这部分内容。

第二章介绍计算中的误差问题,大部分初次从事数值计算的读者可能很少会想到,有时候计算机计算出来的结果会错得离谱,因此,我特别编写单独的一章来介绍这个问题。

第三章是数据的读入和基本的统计分析,这是实证研究和数据分析的基本前提。主要介绍如何读入 EXCEL、CSV 等数据类型的数据文件,还有金融统计分析中经常使用到的 MATLAB 函数的调用。

第四章是回归分析。在长期的研究和教学过程中,我逐渐觉得回归分析在金融研究和金融模型的实际应用过程中非常重要,因此,这章比较详细地介绍了线性回归、非线性回归以及核回归等常用的回归分析方法,以及所使用的 MATLAB 函数,这些函数大部分都是作者基于金融研究的实际需要而编写的。

第五章是金融分析中的优化问题,在金融研究和应用中,优化问题是不可避免的。在很多情形下,我们很难找到优化问题的全局最优解,经常会陷入局部最优解的陷阱。这一章主要介绍我们在金融中经常遇到的一些优化问题,以及如何借助 MATLAB 的帮助比较顺利地找到这些问题的最优解。

第六章是极大似然估计,金融模型都是对现实市场的一种近似刻画,金融模型中的参数估计通常要借助实际数据进行估计,而极大似然估计就是一种常用的参数估计方法。这章介绍了极大似然估计的基本原理,并给出了极大似然估计相应的 MATLAB 函数,借助该函数,读者可以方便地进行许多金融模型的参数估计。

第七章是广义矩估计,广义矩估计是一种非常一般化的估计参数的方法,普通最小二乘法 and 极大似然估计法都可以看成是广义矩估计的特例。因此,广义矩估计在金融研究和实际应用中具有非常广泛的应用。这章介绍了广义矩估计的基本原理,并给出了广义矩估计相应的 MATLAB 函数。最后结合常用的刻画利率随机过程的利率模型,使用实际的市场数据,详细地介绍了如何编写用于广义矩估计的矩条件函数进行参数估计。

第八章是金融资产收益率的波动率估计,波动率估计在风险估计、资产定价和套期保值等方面有着十分重要的意义,这章主要介绍常用的波动率估计方法,比如,ARCH、GARCH、多元 GARCH 等方法,也包含了 GARCHSK 等不太常见的波动率估计方法。

第九章是风险价值和条件风险价值的估计,这章主要介绍了估计风险价值和条件风险价值的一些常用方法:基于正态分布的估计、基于历史数据经验分布的估计、基于 Cornish-Fisher 展开式和极值理论的估计等,也讨论了将波动率替代为风险价值和条件风

险价值后,经典的马克维茨的前沿组合的变化情况。

第十章是利率曲线估计,利率的期限结构在资产定价中有着十分重要的作用。这章给出了样条法和 Svensson 模型估计远期利率曲线和基期利率曲线的 MATLAB 函数,并结合市场实际国债数据详细地介绍了这些函数的调用方法。

第十一章是期权定价的数值方法,主要介绍二叉树方法为欧式和美式期权定价的 MATLAB 函数,蒙特卡罗模拟方法为欧式、亚式和美式期权定价的 MATLAB 函数以及当标的资产收益服从非正态分布时的期权产品定价的 MATLAB 函数,有限差分方法为欧式期权定价的 MATLAB 函数等。

第十二章是状态空间模型在金融中的应用,本章介绍了利用卡曼滤波对某些不可观测的变量进行估计的 MATLAB 函数,还介绍了状态空间模型在我国封闭式基金折价行为实证研究中的具体应用。

最后参考文献部分仅列出了一些重要的参考文献,其他在正文中引用的文献对读者阅读本书并不重要,因此并未在最后的参考文献中列出。尽管我多次校对文中的各处细节,本书依然可能存在错误,欢迎读者指出。

本书的撰写过程中,有幸得到了上海财经大学金融学院同事们的帮助,这里我要特别感谢我的同事韩其恒教授,还要感谢我的同事王安兴教授。感谢英国 Exeter 大学的 Richard Harris 教授,2008 年 Richard Harris 教授邀请我到 Exeter 大学给金融方向的博士研究生开设 MATLAB 的课程,这对本书的内容设置起了很大的作用。

在撰写过程中,我的儿子出生了,这给我带来了以前无法体会的快乐,我要特别感谢妻子为我分担了大部分的家庭琐事和儿子给我带来的无尽欢乐。

曹志广  
2013 年 2 月

# 目 录

|                                 |           |
|---------------------------------|-----------|
| 前言 .....                        | 1         |
| <b>1 MATLAB 入门 .....</b>        | <b>1</b>  |
| 1.1 MATLAB 简介 .....             | 1         |
| 1.2 MATLAB 编程基础 .....           | 4         |
| 1.3 编写 MATLAB 函数 .....          | 18        |
| 1.4 一个混沌游戏 .....                | 20        |
| 1.5 提高 MATLAB 的运算效率 .....       | 22        |
| 1.6 商品交换的例子 .....               | 26        |
| <b>2 数值计算中的误差与误差传播 .....</b>    | <b>29</b> |
| 2.1 认识计算机如何存储数字 .....           | 29        |
| 2.2 误差问题的认识 .....               | 33        |
| 2.3 误差的来源 .....                 | 35        |
| 2.4 误差的度量 .....                 | 37        |
| 2.5 运算中的误差传递 .....              | 37        |
| 2.6 误差控制 .....                  | 38        |
| <b>3 数据的读入和基本统计分析 .....</b>     | <b>40</b> |
| 3.1 金融分析中常见的数据格式 .....          | 40        |
| 3.2 常见的数据获取方式 .....             | 41        |
| 3.3 EXCEL 数据文件的读取 .....         | 41        |
| 3.4 文本数据文件的读取 .....             | 45        |
| 3.5 CSV 格式和 ASCII 格式数据的读取 ..... | 47        |

|          |                         |            |
|----------|-------------------------|------------|
| 3.6      | 通过网络获取数据                | 48         |
| 3.7      | 数据的预处理                  | 49         |
| 3.8      | 数据的描述性统计分析              | 54         |
| 3.9      | 样本分布                    | 55         |
| 3.10     | 产生常见分布的随机数及分布检验         | 62         |
| 3.11     | 自助法                     | 63         |
| 3.12     | 时间序列的基本统计分析             | 65         |
| 3.13     | 常见的假设检验统计方法             | 69         |
| 3.14     | 主成分分析方法                 | 75         |
| 3.15     | 因子分析                    | 79         |
| <b>4</b> | <b>回归分析</b>             | <b>84</b>  |
| 4.1      | MATLAB 在处理回归分析中的优势      | 84         |
| 4.2      | 线性回归                    | 85         |
| 4.3      | 非线性回归                   | 100        |
| 4.4      | 核回归                     | 102        |
| 4.5      | Fama-MacBeth 回归         | 107        |
| 4.6      | 我国股票市场日历效应检验            | 110        |
| 4.7      | 基于线性回归的方差分解             | 117        |
| 4.8      | 一些常见问题的讨论               | 120        |
| <b>5</b> | <b>金融分析中的优化问题</b>       | <b>122</b> |
| 5.1      | 线性规划问题                  | 122        |
| 5.2      | 二次规划问题                  | 124        |
| 5.3      | 无约束非线性函数最优化问题           | 135        |
| 5.4      | 约束非线性函数最优化问题            | 138        |
| 5.5      | 局部最优值和全局最优值             | 140        |
| 5.6      | 优化问题的金融应用:信息交易模型的最优参数估计 | 144        |
| <b>6</b> | <b>极大似然估计</b>           | <b>157</b> |
| 6.1      | 极大似然估计基本原理              | 157        |
| 6.2      | 极大似然估计的 MATLAB 函数       | 159        |
| 6.3      | 二元选择回归问题中的参数估计          | 164        |
| 6.4      | 受限因变量回归模型的参数估计          | 168        |
| 6.5      | 上证综合指数收益率广义双曲线分布的极大似然估计 | 174        |



|                                       |     |
|---------------------------------------|-----|
| <b>7 广义矩估计</b>                        | 181 |
| 7.1 广义矩估计的基本原理                        | 181 |
| 7.2 广义矩估计的参数估计                        | 184 |
| 7.3 广义矩估计的 MATLAB 函数                  | 185 |
| 7.4 广义矩估计的应用                          | 188 |
| <b>8 金融资产收益波动率的估计</b>                 | 199 |
| 8.1 历史波动率                             | 199 |
| 8.2 移动平均模型                            | 199 |
| 8.3 指数加权平均模型                          | 201 |
| 8.4 ARCH 模型                           | 203 |
| 8.5 GARCH 模型                          | 204 |
| 8.6 多元 GARCH 模型                       | 207 |
| 8.7 GARCHSK 模型                        | 210 |
| 8.8 波动率估计的应用:股指期货的套利交易                | 217 |
| <b>9 风险价值和条件风险价值的估计</b>               | 224 |
| 9.1 VaR 和 CVaR 的定义                    | 224 |
| 9.2 基于 Cornish-Fisher 展开式的 VaR 和 CVaR | 225 |
| 9.3 基于正态分布的 VaR 和 CVaR                | 225 |
| 9.4 基于蒙特卡罗模拟的 VaR 和 CVaR              | 225 |
| 9.5 基于历史模拟的 VaR 和 CVaR                | 227 |
| 9.6 极值理论与 VaR 和 CVaR                  | 231 |
| 9.7 均值-方差有效前沿与均值-VaR 及均值-CVaR 有效前沿    | 235 |
| 9.8 不同 VaR 模型套期保值效果的比较                | 244 |
| <b>10 远期利率曲线估计</b>                    | 255 |
| 10.1 即期利率与远期利率                        | 255 |
| 10.2 样条法估计利率曲线                        | 256 |
| 10.3 Svensson 模型估计利率曲线                | 268 |
| <b>11 期权定价的数值方法</b>                   | 274 |
| 11.1 二叉树                              | 274 |
| 11.2 三叉树                              | 277 |
| 11.3 二叉树与期权定价                         | 277 |

|           |                        |            |
|-----------|------------------------|------------|
| 11.4      | 蒙特卡罗模拟和期权定价            | 282        |
| 11.5      | 有限差分方法和期权定价            | 297        |
| 11.6      | 期权定价的应用:银行理财产品的定价      | 306        |
| 11.7      | 期权定价的应用:累计股票期权的定价      | 309        |
| <b>12</b> | <b>状态空间模型在金融中的应用</b>   | <b>312</b> |
| 12.1      | 状态空间模型                 | 312        |
| 12.2      | 状态空间模型与其他时间序列模型        | 313        |
| 12.3      | 卡曼滤波与不可观测变量的估计         | 313        |
| 12.4      | 卡曼滤波的 MATLAB 程序        | 315        |
| 12.5      | 状态空间模型的参数估计            | 320        |
| 12.6      | 应用状态空间模型研究我国封闭式基金的折价行为 | 327        |
|           | <b>参考文献</b>            | <b>335</b> |

# 1 MATLAB 入门

本章主要介绍 MATLAB 编程的入门知识,包括基本的运算和变量命名规则、不同数据类型的生成和调用、矩阵的基本操作、基本的作图功能、编程需要的最基本的循环语句和条件判断语句使用方法,以及如何提高 MATLAB 计算效率的基本技巧。熟悉 MATLAB 的读者可以跳过本章的内容。

## 1.1 MATLAB 简介

### 1.1.1 MATLAB 的优势

MATLAB 是由美国 Mathworks 公司推出的用于数值计算和图形处理的科学计算软件。MATLAB 是英文 Matrix Laboratory(矩阵实验室)的缩写。其第 1 版(DOS 版本 1.0)发行于 1984 年。MATLAB 集中了日常数学处理中的各种功能,包括高效的数值计算、矩阵运算、信号处理和图形生成等功能。在 MATLAB 环境下,用户可以集成地进行程序设计、数值计算、图形绘制、输入输出、文件管理等各项操作。与利用 C 语言或 FORTRAN 语言作数值计算的程序设计相比,利用 MATLAB 可以节省大量的编程时间。在美国的一些大学里, MATLAB 正在成为对数值线性代数以及其他一些高等应用数学课程进行辅助教学的有益工具。在工程技术界, MATLAB 也被用来解决一些实际课题和数学模型问题。典型的应用包括数值计算、算法预设计与验证,以及一些特殊的矩阵计算应用,如自动控制理论、统计、数字信号处理(时间序列分析)等。进入 20 世纪 90 年代的时候, MATLAB 已经成为国际控制界公认的标准计算软件。到 90 年代初期,在国际上 30 多个数学类科技应用软件中, MATLAB 在数值计算方面独占鳌头,而 MATHEMATICA 和 MAPLE 则分居符号计算软件的前两位。

MATLAB 具有以下优势:

(1)编程简单,类似于其他语言,如 C 语言。MATLAB 的语言与数学公式的书写雷同,是一种解释性的语言,非常容易理解。因此,编程者能够以数学公式的思维方式写程序,另外,MATLAB 的程序编写格式简单,不像其他语言有着复杂的格式要求,初学者能够很快掌握基本的编程技能。MATLAB 语言最基本的就是函数的调用,MATLAB 本身提供了丰富的内部函数库,使用者可以直接调用,调用的基本格式是  $[a,b,c,\dots]=\text{funname}(A,B,C,\dots)$ ,其中: $a,b,c,\dots$ 是函数的输出,全部包含在方括号中,不同的输出变量以逗号隔开; $A,B,C,\dots$ 是函数的输入,全部包含在圆括弧中,不同的输入变量以逗号隔开; $\text{funname}$  则是函数的名称。比如使用者想计算  $\sin(3)$  的值,可以直接调用 MATLAB 的内部函数: $y=\sin(3)$ ,正弦函数在 MATLAB 内部函数库中的名字与其数学表示  $\sin$  相同。

(2)集成度更高,扩展性更好。MATLAB 提供了强大的函数库,如优化工具箱、时间序列工具箱、信号处理工具箱等。用户也可以开发自己的函数库,MATLAB 是一个完全开放的系统,其函数库的扩展完全是没有边界的。另外,MATLAB 并不排他,MATLAB 可以方便地与 FORTRAN、C 等语言进行连接。

(3)强大的矩阵处理能力。MATLAB 在进行矩阵运算方面是首屈一指的。虽然 MATLAB 作为解释性的语言,其运算速度相比 C 语言等要逊色,但 MATLAB 借助其在矩阵运算方面的强大优势,在金融工程、信号处理、控制系统等领域显得非常高效、简洁和方便。在控制界,MATLAB 是国际首选的计算机语言,在金融工程领域也是首选计算工具。

(4)丰富的网络资源。MATLAB 的爱好者遍布各个领域,也经常将一些程序放到网上共享,比如:Mathworks 公司的主页 [www.mathworks.com](http://www.mathworks.com) 上就有许多 MATLAB 爱好者发布的免费下载的程序。这些共享资源为 MATLAB 使用者大大节省了时间和精力。

### 1.1.2 MATLAB 的界面

启动 MATLAB 后,进入 MATLAB 默认的桌面平台,图 1.1 显示了 MATLAB 7.1 的桌面平台。

桌面平台主要有四个窗口:当前目录窗口、命令窗口、工作管理窗口和历史命令窗口。当前目录窗口如图 1.1 中的最左边窗口所示。当前目录的位置是可以由使用者设定的,在图 1.1 中,我们将当前目录设定为  $E:\backslash\text{caozhiguang}\backslash\text{matlab\_work}$ 。当前目录窗口显示了在该目录下所有的文件和文件夹,可以方便使用者搜索和查询。命令窗口如图 1.1 中的中间窗口所示。“>>”为运算提示符,表示 MATLAB 正处在等待命令输入状态。在运算提示符后输入命令,比如: $a=1+2$  并按 Enter 键,MATLAB 将给出计算结果, $a=3$ ,然后再次进入等待状态。工作管理窗口如图 1.1 中的右上窗口所示。工作管理窗口显示

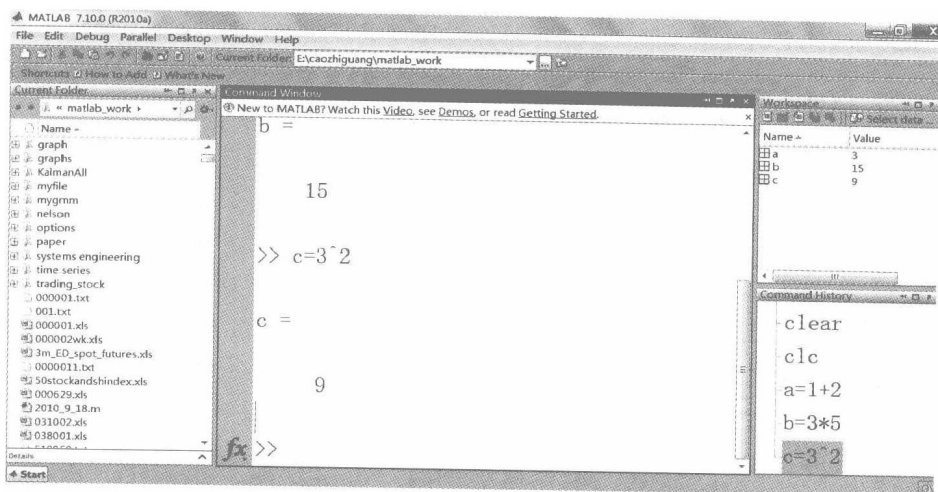


图 1.1 MATLAB 的桌面平台

所有目前内存中 MATLAB 变量的变量名、类型等信息,鼠标左键双击工作管理窗口的某个变量,就可以在新的打开窗口中看到变量的具体内容。历史命令窗口如图 1.1 中的右下窗口所示。历史命令窗口显示了在命令窗口中所有命令的历史记录,方便使用者进行查询。鼠标左键双击历史命令窗口的某一行命令,则该行命令将在命令主窗口中得到执行。

### 1.1.3 MATLAB 的搜索路径

MATLAB 的所有命令操作都是在它的搜索路径中进行的(包括当前路径)。MATLAB 默认的搜索路径是 MATLAB 的安装目录以及所有自带工具箱的路径。这里的路径其实就是目录的意思。经常有初学者自己编写了一个函数,并且保存在某一个目录之下,但在命令窗口下调用该函数时,MATLAB 却提示找不到该函数。这是因为使用者保存该函数的目录并不在 MATLAB 搜索的路径当中,我们需要通过路径设置将该目录添加到 MATLAB 搜索的路径当中。这样 MATLAB 就能调用所有在搜索路径之下的任何函数或文件了。具体地,选择 MATLAB 窗口中的【File】下拉菜单中的【Set Path】,进入设置搜索路径的对话框,如图 1.2 所示。在对话框中选择【Add Folder】,选择所要加入的目录后将设置保存后,则所加目录下的文件和函数都可以被 MATLAB 调用,但该目录下面子目录的文件和函数,还是不在搜索路径之下。如果要将目录下面所有子目录下的文件和函数也置于 MATLAB 搜索路径之下,则选择【Add with Subfolder】进行相应的设置并保存即可。



图 1.2 路径设置对话框

## 1.2 MATLAB 编程基础

### 1.2.1 MATLAB 变量命名规则

在 MATLAB 中,变量名称的英文大小写是有区别的(apple, Apple, AppLe 表示三个不同的变量名)。在 MATLAB7.1 中,变量名的长度上限为 63 位,超过部分将被忽略。不同的 MATLAB 版本对变量名的长度上限规定可能存在差别。在 MATLAB 命令窗口下键入  $N = \text{namelengthmax}$  后,敲回车键即可得到长度上限的值  $N$ 。

变量名的第一个字必须是英文字,随后可以掺杂英文字、数字或是下划线,不可以有其他字符。比如:变量名中不能出现“+”,“.”,“-”,“\*”,“/”等符号。 $A\_1$ ,  $A1$ ,  $ABC123$ ,  $ABC\_123$  都是可以被 MATLAB 识别的有效变量名,而  $2A$ ,  $a * 2$ ,  $a/b$  都是错误的变量名。

MATLAB 的赋值规则很简单,如:要将数值 2 赋给变量  $a$ ,则只需要在 MATLAB 命令窗口中输入  $a=2$  并点击回车键即可完成赋值,事先也无需申明变量  $a$  的类型。 $a=2$  同时完成了赋值和取变量名的任务。在将 2 赋给变量  $a$  后,我们如果在 MATLAB 命令窗口中输入  $b=a+3$  并点击回车键,在命令窗口中就显示得到  $b=5$ 。如果输入  $b=a+3$  并敲回车键,则命令窗口中不显示运算的结果,但实际上 MATLAB 已经完成了计算,并保存在内存中,用户在工作管理窗口可以看到新的变量  $b$ 。接下来用户可以直接调用变量  $b$ 。用户可以通过在 MATLAB 命令窗口中输入  $\text{clear}$  并点击回车键来清除内存中的所有变量,也可以通过  $\text{clear}$  后面跟空格再接变量名来删除特定的变量。比如:我们要删除前

面已经保存在内存中的变量 a, 则输入 `clear a` 并点击回车键即可, 如果要删除前面已经保存在内存中的变量 a 和 b, 但保留内存中其他变量, 则输入 `clear a b` 并点击回车键即可, 变量名之间也以空格隔开。

MATLAB 也自定义了一些变量, 比如: `eps`、`pi`、`Inf`(或 `inf`)、`NaN`(或 `nan`) 等。`eps` 在 MATLAB 中定义为一个非常小的正数:  $2.2204 \times 10^{-16}$ 。`pi` 就被 MATLAB 自定义为无理数  $\pi$  值。`inf` 则表示无穷大的数。`NaN`(或 `nan`) 则被 MATLAB 定义为“不是一个数”。`i` 或 `j` 在 MATLAB 中自定义为虚数  $\sqrt{-1}$ 。`realmin` 在 MATLAB 中自定义为最小的正浮点数, 即  $2.2251 \times 10^{-308}$ 。`realmax` 在 MATLAB 中自定义为最大的浮点数, 即  $1.7977 \times 10^{+308}$ 。用户在定义变量名时, 尽量不要使用上述变量名。如果用户一定要以上述变量名来命名一个变量, MATLAB 也是允许的, 并且以用户定义的为准。比如: 如果我们在 MATLAB 命令窗口中输入 `pi` 并点击回车键后得到 `ans=3.1416`, 这是 MATLAB 默认的变量名, 表示圆周率, 如果数值没有赋给任何变量名称, 则 MATLAB 自动以 `ans` 来命名。如果我们输入 `pi=1` 并点击回车键后得到 `pi=1`, 这时, `pi` 不再表示圆周率, 而是等于数值 1。如果要恢复 MATLAB 默认的变量名, 只要输入 `clear pi` 然后点击回车键, 将用户定义的变量 `pi` 删除即可恢复到 MATLAB 默认的状态, 即 `pi` 仍然表示圆周率。当然 MATLAB 重新启动后, MATLAB 也自动恢复到默认的变量名。

### 1.2.2 简单数学运算

MATLAB 处理简单的数学运算, 与普通的计算器没有什么不同。加、减、乘、除、幂次方在 MATLAB 中分别以 `+`、`-`、`*`、`/`、`^` 表示。比如: 我们要计算:  $5+3$ ,  $5-3$ ,  $5*3$ ,  $5/3$ ,  $5^3$ , 就分别在 MATLAB 命令窗口中输入 `5+3`, `5-3`, `5*3`, `5/3`, `5^3` 即可得到相应的计算结果。如果我们要算  $1 \times 10 + 2 \times 20 + 3 \times 30$  这个式子, 并将计算结果赋给变量名 `X`, 则我们在 MATLAB 命令窗口中输入 `X=1*10+2*20+3*30` 并点击回车键后得到: `X=140`。如果输入 `X=1*10+2*20+3*30`, 则在命令窗口中并不显示计算结果, 如果想知道计算结果, 只需要键入 `X` 并点击回车键即可在命令窗口中显示: `X=140`。

MATLAB 的运算顺序完全遵循数学上的规则。比如: 我们在 MATLAB 命令窗口中输入 `X=(2+2.5)*2/3^2` 并点击回车键后得到: `X=1`, 而不是 9。

常见的数学函数在 MATLAB 中可以直接调用, 函数名与其数学上的表示非常接近或完全相同。表 1.1 列出了 MATLAB 中定义的部分常见函数名称。详细的函数描述可以在 MATLAB 主窗口中点击【Help】进行查询。比如: 我们在 MATLAB 命令窗口中输入 `a=log(5)` 后得到 `a=1.6094`; 输入 `b=ceil(1.2)` 得到 `b=2`; 输入 `c=gamma(2)` 得到 `c=1`。

表 1.1

常用的基本函数

| 函数名     | 数学含义        | 函数名     | 数学含义         |
|---------|-------------|---------|--------------|
| sin     | 正弦函数        | asin    | 反正弦函数        |
| cos     | 余弦函数        | acos    | 反余弦函数        |
| exp     | 指数函数        | log     | 自然对数函数       |
| log2    | 以 2 为底的对数函数 | log10   | 以 10 为底的对数函数 |
| sqrt    | 根号函数        | abs     | 绝对值或模函数      |
| floor   | 朝负方向取整函数    | ceil    | 朝正方向取整函数     |
| round   | 四舍五入取整函数    | sign    | 符号函数         |
| besselj | 第一类贝赛尔函数    | bessely | 第二类贝赛尔函数     |
| besselh | 第三类贝赛尔函数    | besseli | 修正的第一类贝赛尔函数  |
| besselk | 修正的第二类贝赛尔函数 | gamma   | Gamma 函数     |

### 1.2.3 矩阵(向量)的运算

当某一变量为向量或是矩阵时,如果要键入元素,须用中括号[]将各个元素置于其中,元素与元素之间用逗号或空格隔开,“;”则表示换行。例如:在 MATLAB 命令窗口中输入  $a=[1,2,3;4,5,6;7,8,0]$  得到:

```
a=
     1     2     3
     4     5     6
     7     8     0
```

在 MATLAB 命令窗口中输入  $b=[3,8,1;9,5,2;6,2,1]$  得到:

```
b=
     3     8     1
     9     5     2
     6     2     1
```

在 MATLAB 中,我们还能够以更简单的方式来输入一个向量,比如:输入  $s=1:5$  得到:

```
s=
     1     2     3     4     5
```

如果输入  $s=1:2:9$  得到

```
s=
```



```
1    3    5    7    9
```

输入 `s=1:2:9` 中的 2 定义了相邻元素间的间隔,如果间隔为 1,则可以简写为 `s=1:9`。另外,还可以用 `linspace` 来快速生成一个向量。比如:我们输入 `s=linspace(1,9,5)` 得到区间 1 到 9 之间 4 等分后的各区间端点:

```
s=
```

```
1    3    5    7    9
```

而 MATLAB 提供的 `ones`, `zeros`, `eye` 等函数也给出了矩阵的简便输入方法。比如:输入 `ones(3,3)` 得到  $3 \times 3$  的全部是 1 的矩阵:

```
ans=
```

```
1    1    1
1    1    1
1    1    1
```

输入 `zeros(3,3)` 得到  $3 \times 3$  的全部是 0 的矩阵:

```
ans=
```

```
0    0    0
0    0    0
0    0    0
```

输入 `eye(3)` 得到  $3 \times 3$  的单位矩阵:

```
ans=
```

```
1    0    0
0    1    0
0    0    1
```

在 MATLAB 中也能很方便地创建高维数组,比如:输入 `A(:,:,1)=[1,2;3,4]; A(:,:,2)=[5,6;7,8]; A(:,:,3)=[9,8;7,6]; A` 得到:

```
A
```

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

```
1    2
3    4
```

```
A(:,:,2)=
```

```
5    6
7    8
```

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

```
9    8
7    6
```

矩阵或向量的运算在 MATLAB 中十分方便。比如:在 MATLAB 中得到上面的两