


高等院校金融专业应用型特色教材

金融数量方法教程

JINRONG SHULIANG FANGFA JIAOCHENG

张树德 编著



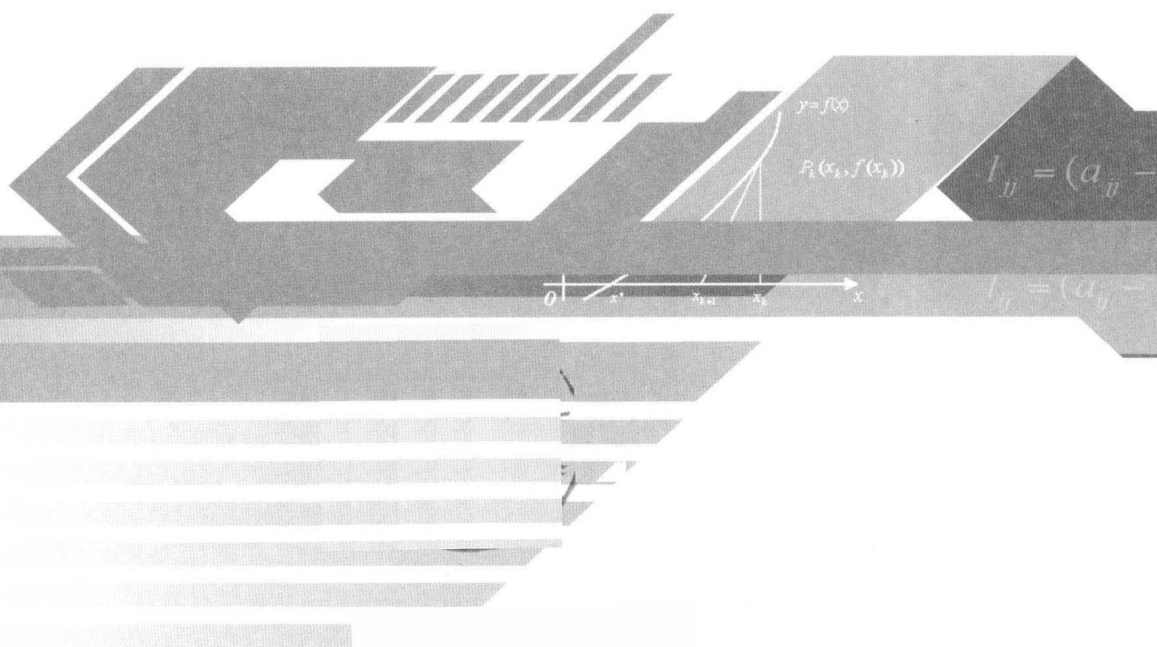
 经济科学出版社
Economic Science Press

高等院校金融专业应用型特色教材

金融数量方法教程

JINRONG SHULIANG FANGFA JIAOCHENG

张树德 编著



 经济科学出版社
Economic Science Press

图书在版编目(CIP)数据

金融数量方法教程 / 张树德编著.—北京: 经济科学出版社, 2010.8

ISBN 978-7-5058-9633-8

I. ①金… II. ①张… III. ①金融学: 数量经济学-教材 IV. ①F830

中国版本图书馆 CIP 数据核字 (2010) 第 127840 号

责任编辑: 段 钢

责任校对: 王肖楠

版式设计: 代小卫

技术编辑: 邱 天

金融数量方法教程

张树德 编著

经济科学出版社出版、发行 新华书店经销

社址: 北京市海淀区阜成路甲 28 号 邮编: 100142

编辑部电话: 88191350 发行部电话: 88191540

网址: www.esp.com.cn

电子邮件: esp@esp.com.cn

北京天宇星印刷厂印刷

三佳装订厂装订

787×1092 16 开 19 印张 480000 字

2010 年 8 月第 1 版 2010 年 8 月第 1 次印刷

印数: 0001—3000 册

ISBN 978-7-5058-9633-8 定价: 38.00 元

(图书出现印装问题, 本社负责调换)

(版权所有 翻印必究)

前 言

金融机构很多产品都属于金融衍生产品，都需要建立衍生品模型，而这些复杂模型都离不开数值计算。目前，金融机构使用量化投资模型的趋势越来越明显，对冲基金、银行及大型投资银行都是量化模型推动者和主要实践者，排名靠前的对冲基金不少依靠量化模型进行决策。

数量投资比较成功的是美国数学教授西蒙斯，西蒙斯 1982 年下海经商，创立文艺复兴科技公司，管理大奖章基金。西蒙斯教授的对冲基金不仅收益率高，而且稳定性好，连续多年在对冲基金行业名列前茅，即使在次贷危机肆虐的 2008 年仍然获得高额回报。笔者认为中国市场的有效性相对美国市场较弱，没有被市场消化的残存信息更多，建立量化投资模型来有效捕捉到这些信息，相信也会获得成功，未来量化模型应该在我国大有用武之地。

量化投资离不开功能强大软件的支持，MATLAB 是国外非常流行的计算软件，不仅计算功能强大，而且提供了很多数据处理与应用平台。MATLAB 和 VC++、VB、Java、Excel VBA、Access、SQL 等都有数据接口相连接，是系统的金融资源整合平台。

近年来，MATLAB 在国内普及速度非常快，甚至有些中学都已经采用 MATLAB 作为数学、物理等课程的辅助教学工具。但遗憾的是，MATLAB 在国内金融业的运用却一直没有得到广泛普及。目前只有部分大学将 MATLAB 作为金融实验课程，而在国外 MATLAB 早已经成为流行的金融工程软件，很多大学金融学都将 MATLAB 作为基础课程。近年来，MATLAB 金融工具箱拓展很快，每年都增加新的金融工程及金融计量模型，金融工具箱越来越完善。通过 MATLAB 学习金融，可以节省时间，起到事半功倍的作用。

笔者认为 MATLAB 应该作为大中专学校金融学核心课程，通过对 MATLAB 金融工具箱的学习，学生可以加深对金融模型的理解，更好地将其与实践结合。本书部分内容源于我为金融机构培训课程。书中所有程序均在 MATLAB 2008b 版本上运行通过，为了方便读者学习，每章例题的 MATLAB 代码都将放到网上，请访问 www.esp.com.cn 点“资源下载”获取。

本书可作为金融研究人员、教师、经济金融工作者参考书，同时也可作为证券公司、基金公司等金融从业人员参考书。

由于水平有限，书中难免存在一些疏漏，恳请读者予以批评指正。
笔者博客：<http://blog.sina.com.cn/matlabfin>，电子邮箱：zhangader@126.com。

张树德 博士

2010年3月28日于浦东陆家嘴

目 录

第 1 章	MATLAB 基本计算	1
1.1	集合运算	1
1.1.1	基本运算	
1.1.2	矩阵逻辑运算	
1.2	范数	7
1.2.1	向量范数	
1.2.2	矩阵范数	
1.3	矩阵分解	9
1.3.1	矩阵 LU 分解	
1.3.2	正定矩阵 Cholesky 分解	
1.4	非线性方程的数值解法	10
1.5	约束最优化	15
1.5.1	基础知识	
1.5.2	约束优化问题的 Kuhn-Tucker 条件	
1.6	罚函数法求解非线性规划	17
1.6.1	罚函数法原理	
1.6.2	外部惩罚函数法	
1.6.3	内部惩罚函数法	
1.6.4	等号约束的乘子法	
1.6.5	不等式约束下的乘子法	
1.7	迭代法求解线性方程	26
1.7.1	雅可比迭代法	
1.7.2	高斯-赛德尔迭代法	
1.7.3	超松弛迭代法	
1.7.4	迭代法收敛条件与误差估计	
1.8	偏导数与卷积	31
1.8.1	偏导数	
1.8.2	卷积	

1.9	句柄函数	33
1.9.1	函数句柄创建和显示	
1.9.2	句柄函数的调用和操作	
1.9.3	避免两个相近的数相减	
1.10	MATLAB 基本操作命令	37
1.10.1	MATLAB 的工作空间	
1.10.2	文件管理	
1.11	MATLAB 程序设计原则	38
1.11.1	程序设计规则	
1.11.2	MATLAB 的程序类型	
1.11.3	声明子程序变量	
1.11.4	字符串及其宏命令	
1.11.5	常用的编程命令	
第 2 章	利率曲线插值与拟合	42
2.1	利率曲线插值	42
2.1.1	插值法的基本原理	
2.1.2	三次样条插值的基本原理	
2.1.3	样条函数插值利率期限结构	
2.1.4	改进样条函数插值利率期限结构	
2.1.5	逐段光滑的三次函数插值	
2.2	最小二乘拟合	54
2.2.1	最小二乘拟合原理	
2.2.2	线性最小二乘拟合	
2.3	分段三次样条拟合	57
2.3.1	分段样条函数拟合利率曲线	
2.3.2	分段三次样条函数拟合价格	
2.4	B 样条函数拟合	69
2.5	Nelson-Siegel 方法拟合	73
2.5.1	Nelson-Siegel 模型	
2.5.2	Nelson-Siegel 模型扩展形式	
2.6	利用互换市场数据拟合利率期限结构	77
第 3 章	资产组合	79
3.1	二次型的基本原理	79

3.2	资产组合的基础知识	83
3.2.1	资产组合收益与风险	
3.2.2	协方差矩阵与相关系数矩阵	
3.2.3	资产组合收益率与标准差	
3.3	资产组合原理	88
3.3.1	均值方差理论	
3.3.2	考虑投资者偏好的组合	
3.4	投资组合评价指标	91
3.4.1	夏普比率	
3.4.2	信息比率	
3.5	资产配置	93
3.5.1	两种资产组合收益期望与方差	
3.5.2	均值方差有效前沿	
3.5.3	带约束条件的资产组合有效前沿	
3.5.4	考虑无风险资产及借贷情况下的资产配置	
3.5.5	线性规划求解资产组合问题	
3.5.6	线性规划求解现金流匹配最小成本	
3.5.7	二次规划求解资产组合问题	
3.6	资产定价理论	107
3.6.1	证券市场线	
3.6.2	CAPM (资本资产定价模型)	
3.6.3	计算经过风险调整的 ALPHA 及回报	
3.7	Black-Litterman 模型	113
3.7.1	Black-Litterman 模型的理论基础	
3.7.2	Black-Litterman 模型的参数说明	
3.7.3	Black-Litterman 模型的评价	
第 4 章 随机过程基本原理及应用		122
4.1	概率论基本知识	122
4.1.1	概率空间	
4.1.2	随机变量	
4.1.3	数学期望与方差	
4.1.4	随机变量相关性	
4.1.5	随机变量的收敛性	
4.1.6	离散型概率转移测度	

4.1.7 Radon-Nikodym 导数	
4.2 随机过程	130
4.2.1 随机过程的概念	
4.2.2 独立增量过程	
4.2.3 随机积分	
4.2.4 Girsannov 定理	
4.2.5 Feynman-Kac 定理	
4.3 马尔可夫过程	134
4.3.1 马尔可夫过程的定义	
4.3.2 转移概率	
4.4 CreditMetrics 模型	137
4.4.1 CreditMetrics 模型概述	
4.4.2 Creditmetrics 模型实例	
4.5 基于马尔可夫链价值评估	146
第 5 章 随机模拟	150
5.1 随机数生成	150
5.1.1 随机数生成原理	
5.1.2 生成正态分布随机数	
5.1.3 生成多元正态分布随机数	
5.2 维纳过程	158
5.2.1 维纳过程性质	
5.2.2 维纳过程实例	
5.3 几何布朗运动模拟	160
5.3.1 随机微分方程	
5.3.2 随机微分的泰勒展式	
5.3.3 几何布朗运动一阶近似	
5.3.4 几何布朗运动二阶近似	
5.3.5 风险中性测度模拟	
5.4 最小二乘蒙特卡罗模拟美式期权	169
5.4.1 最小二乘模拟原理	
5.4.2 美式期权模拟方法	
5.5 障碍期权模拟	175

第 6 章 股票类衍生产品计算	186
6.1 期权基本知识	186
6.1.1 期权概念	
6.1.2 奇异期权	
6.2 Black-Scholes 方程	188
6.2.1 Black-Scholes 方程的推导	
6.2.2 风险中性测度下的期权定价公式	
6.3 看涨期权与看跌期权的平价关系	194
6.3.1 美式看涨期权与看跌期权之差的下界	
6.3.2 美式看涨期权与看跌期权之差的变化区间	
6.3.3 欧式看涨期权与看跌期权的下界	
6.4 二叉树定价	197
6.4.1 单期的二叉树模型	
6.4.2 二项式期权定价	
6.5 有限差分法定价	203
6.5.1 偏微分方程分类	
6.5.2 有限差分离散方法	
6.5.3 显式法求解欧式看跌期权	
6.5.4 显式法求解美式看跌期权	
6.5.5 隐式法求解欧式看跌期权	
6.5.6 偏微方程变量代换	
6.5.7 有限差分法稳定性分析	
第 7 章 动态利率模型	217
7.1 瞬时利率与贴现债券价格	217
7.1.1 瞬时利率	
7.1.2 利率曲线	
7.2 Ho-Lee 利率模型	219
7.2.1 Ho-Lee 模型离散型形式	
7.2.2 利率模型校准	
7.2.3 根据利率期限结构校准	
7.3 基本利率过程	232
7.3.1 O-U 过程	
7.3.2 平方根过程	
7.4 Hull-White 模型三叉树结构	235
7.5 Vasicek 模型	243

7.6 CIR 利率模型	245
第 8 章 利率衍生品定价	247
8.1 构建利率二叉树	247
8.2 可赎回债券定价	250
8.3 回售债券定价	252
8.4 浮动利率上限定价	254
8.5 阶梯可赎回债券定价	260
8.6 美式看涨利率期权二叉树定价	265
8.7 期权调整利差	266
8.8 二叉树计算久期与凸度	269
8.8.1 久期与凸度概念	
8.8.2 凸度计算价格波动	
8.8.3 利率二叉树计算可赎回债券久期与凸度	
附录 1 金融数据函数	275
附录 2 金融衍生品定价函数	281
附录 3 金融时间序列函数	287
附录 4 GARCH 工具箱	290
参考文献	292

1.1 集合运算

1.1.1 基本运算

1. 集合并集运算

在 MATLAB 中, 主要通过 `union()` 函数实现集合并集运算, 其调用方式如下。

- `c=union(A,B)`: 计算 A 、 B 的并集, 即 $c=A \cup B$ 。
- `c=union(A,B,'rows')`: 返回矩阵 A 、 B 不同行向量构成的大矩阵。
- `[c,ia,ib]=union(A,B)`: ia 、 ib 分别表示 c 中行向量在原矩阵 (向量) 中的位置。

【例 1-1】 `union()` 函数的应用 1。

```
>> A=[1 2 3 4];
>> B=[2 4 5 8];
>> C=union(A,B)
C =
    1 2 3 4 5 8
```

【例 1-2】 `union()` 函数的应用 2。

```
>> A=[1 2 3 4;1 2 4 6]
A =
    1 2 3 4
    1 2 4 6
>> B=[1 2 3 8;1 1 4 6]
B =
    1 2 3 8
    1 1 4 6
>> [c,ia,ib]=union(A,B,'rows')
c =
    1 1 4 6
    1 2 3 4
    1 2 3 8
```

```

    1 2 4 6
ia =
    1
    2
ib =
    2
    1

```

2. 集合交集运算

在 MATLAB 中，主要通过 `intersect()` 函数实现集合交集运算，其调用方式如下。

- `c=intersect(A,B)`: 返回向量 A 、 B 的公共部分，即 $c=A \cap B$ 。
- `c=intersect(A,B,'rows')`: A 、 B 为相同列数的矩阵，返回元素相同的行。
- `[c,ia,ib]=intersect(A,B)`: c 为 A 、 B 的公共元素； ia 表示公共元素在 A 中的位置； ib 表示公共元素在 B 中的位置。

【例 1-3】 `intersect()` 函数的应用 1。

```

>> A=[1 2 3 4;1 2 4 6;6 7 1 4]
A =
    1 2 3 4
    1 2 4 6
    6 7 1 4
>> B=[1 2 3 8;1 1 4 6;6 7 1 4]
B =
    1 2 3 8
    1 1 4 6
    6 7 1 4
>> C=intersect(A,B,'rows')
C =
    6 7 1 4

```

【例 1-4】 `intersect()` 函数的应用 2。

```

>> A = [1 9 6 20]; B = [1 2 3 4 6 10 20];
>> [c,ia,ib] = intersect(A,B)
c =
    1 6 20
ia =
    1 3 4
ib =
    1 5 7

```

3. 集合属于运算

在 MATLAB 中, 主要通过 `ismember()` 函数实现集合属于运算, 其调用方式如下。

- $k = \text{ismember}(A, S)$: 当 A 中元素属于 S 时, k 取 1; 否则, k 取 0。
- $k = \text{ismember}(A, S, 'rows')$: A 、 S 有相同列, 如果对应行中元素相同时, k 取 1; 不相同, k 取 0。

【例 1-5】 `ismember()` 函数的应用 1。

```
>> A=[1 2 3 4 5 6];
>> S=[0 2 4 6 8 10 12 14 16 18 20];
>> k=ismember(A,S)
k =
```

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

1 表示相同元素的位置。

【例 1-6】 `ismember()` 函数的应用 2。

```
>> A=[1 2 3 4;1 2 4 6;6 7 1 4]
>> B=[1 2 3 8;1 1 4 6;6 7 1 4]
>> k=ismember(A,B,'rows')
k =
0
0
1
```

1 表示两集合中元素相同的行。

4. 集合差运算

在 MATLAB 中, 主要通过 `setdiff()` 函数实现集合差运算, 其调用方式如下。

- $c = \text{setdiff}(A, B)$: 返回 A 中属于 A 但不属于 B 的元素的集合, 即 $c = A - B$ 。
- $c = \text{setdiff}(A, B, 'rows')$: 返回属于 A 但不属于 B 的不同行。
- $[c, i] = \text{setdiff}(A, B)$: i 表示属于 A 但不属于 B 的元素的位置。

【例 1-7】 `setdiff()` 函数的应用 1。

```
>> A = [1 7 9 6 20]; B = [1 2 3 4 6 10 20];
>> c=setdiff(A,B)
c =
7 9
```

【例 1-8】 `setdiff()` 函数的应用 2。

```
>> A=[1 2 3 4;1 2 4 6;6 7 1 4]
>> B=[1 2 3 8;1 1 4 6;6 7 1 4]
```

```
>> c=setdiff(A,B,'rows')
c =
     1     2     3     4
     1     2     4     6
```

5. 集合交集的非（异或）运算

在 MATLAB 中，主要通过 `setxor()` 函数实现集合交集的非（异或）运算，其调用方式如下。

- `c=setxor(A,B)`: 返回集合 A 、 B 交集的非。
- `c=setxor(A,B,'rows')`: 返回矩阵 A 、 B 交集的非， A 、 B 有相同列数。
- `[c,ia,ib]=setxor(...)`: ia 、 ib 表示 c 中元素分别在 A 、 B 中的位置。

【例 1-9】 `setxor()` 函数的应用 1。

```
>> A=[1 2 3 4];
>> B=[2 4 5 8];
>> C=setxor(A,B)
C =
     1     3     5     8
```

【例 1-10】 `setxor()` 函数的应用 2。

```
>> A=[1 2 3 4;1 2 4 6;6 7 1 4]
A =
     1     2     3     4
     1     2     4     6
     6     7     1     4
>> B=[1 2 3 8;1 1 4 6;6 7 1 4]
B =
     1     2     3     8
     1     1     4     6
     6     7     1     4
>> [C,ia,ib]=setxor(A,B,'rows')
C =
     1     1     4     6
     1     2     3     4
     1     2     3     8
     1     2     4     6
ia =
     1
     2
ib =
```

2

1

6. 取集合单值元素

在 MATLAB 中，主要通过 `unique()` 函数实现取集合单值元素，其调用方式如下。

- `b=unique(A)`: 取集合 A 的不重复元素构成的向量。
- `b=unique(A, B, 'rows')`: 返回 A 、 B 不同行元素组成的矩阵。
- `[b, i, j]=unique(...)`: i 、 j 表示 b 中元素在原向量（矩阵）中的位置。

【例 1-11】 `unique()` 函数的应用 1。

```
>> A=[1 1 2 2 4 4 6 4 6]
A =
     1     1     2     2     4     4     6     4     6
>> [b,i,j]=unique(A)
b =
     1     2     4     6
i =
     2     4     8     9
j =
     1     1     2     2     3     3     4     3     4
```

【例 1-12】 `unique()` 函数的应用 2。

```
>> A=[1 2 2 4;1 1 4 6;1 1 4 6]
A =
     1     2     2     4
     1     1     4     6
     1     1     4     6
>> [b,i,j]=unique(A, 'rows')
b =
     1     1     4     6
     1     2     2     4
i =
     3
     1
j =
     2
     1
     1
```


1.1.2 矩阵逻辑运算

设矩阵 A 和 B 同为 $m \times n$ 矩阵，或其中之一为标量。在 MATLAB 中，可以定义如下逻辑运算。

(1) 与运算，其调用方式如下。

$A \& B$ 或 `and(A, B)`: A 与 B 对应元素进行与运算，若两个数均非 0，则结果为 1；否则为 0。

(2) 或运算，其调用方式如下。

$A | B$ 或 `or(A, B)`: A 与 B 对应元素进行或运算，若两个数均为 0，则结果为 0；否则为 1。

(3) 非运算，其调用方式如下。

$\sim A$ 或 `not(A)`: 对 A 中的元素进行非运算，若 A 的元素为 0，则结果为 1；否则为 0。

(4) 异或运算，其调用方式如下。

$\text{xor}(A, B)$: A 与 B 对应元素进行异或运算，若两个数不同的时候，则结果为 1；否则为 0。

【例 1-13】 矩阵的逻辑运算。

```
>> A=[0 2 3 4;1 3 5 0],B=[1 0 5 3;1 5 0 5]
```

```
A =
```

```
0 2 3 4
1 3 5 0
```

```
B =
```

```
1 0 5 3
1 5 0 5
```

```
>> C1=A&B,C2=A|B,C3=~A,C4=xor(A,B)
```

```
C1 =
```

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

```
C2 =
```

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

```
C3 =
```

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

```
C4 =
```

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