

金融计量经济学：理论、案例与 R 语言

孙佳婧、洪永淼、奥利弗·林顿

2025 年 10 月

插图

1.1 R 软件界面 (版本 4.2.2)	3
1.2 RStudio 的用户界面	5
1.3 预加载数据列表	17
1.4 1952 年以来中国的国内生产总值 (GDP)	21
1.5 1952 年以来中国的国内生产总值 (GDP, 单位: 亿元)	22
1.6 使用 <code>mtcars</code> 数据集绘制的各类图形	24
1.7 带有密度曲线的直方图	28
1.8 使用 <code>ggplot2</code> 对 <code>mtcars</code> 数据集进行可视化 (第 I 部分)	47
1.9 使用 <code>ggplot2</code> 对 <code>mtcars</code> 数据集进行可视化 (第 II 部分)	48
2.1 主要股指的对数收益率	68
2.2 展望理论	77
3.1 学生 t 分布与标准正态分布概率密度函数的比较	97
3.2 汽车重量与 MPG 的关系	108
3.3 比亚迪超额收益与市场超额收益的散点图及 CAPM 拟合直线	112
4.1 随机游走过程的单次实现与 50 条独立随机游走路径的比较	128
4.2 模拟生成的 MA(2) 序列: $\theta_1 = 0.5, \theta_2 = 0.3$, 且 $\varepsilon_t \sim N(0, 1)$	133
4.3 模拟生成的 ARMA(1,1) 序列: $\phi_1 = 0.6, \theta_1 = 0.7$, 且 $\varepsilon_t \sim N(0, 1)$	136
4.4 MA (1) 和 MA (2) 过程的自相关函数	139
4.5 MA (1) 和 MA (2) 过程的偏自相关函数	140
4.6 基于极大似然估计得到的 $\hat{\mu}$ 和 $\hat{\sigma}$ 的分布	154
4.7 模拟生成的 AR (1) 序列: $y_t = 0.9y_{t-1} + \varepsilon_t, \varepsilon_t \sim \text{i.i.d. } N(0, 1)$	157
4.8 模拟生成的 AR (1) 序列的自相关函数 (ACF) 和偏自相关函数 (PACF) 图 .	157
4.9 随机游走过程的时间序列图	158
4.10 随机游走过程的自相关函数 (ACF) 与偏自相关函数 (PACF) 图	158
4.11 AR (1) 过程中负向冲击前后的对比效果	159
4.12 随机游走过程中负向冲击前后的对比效果	159
4.13 2023 年 1 月 3 日至 2023 年 12 月 29 日上证综合指数走势	166
4.14 上证指数对数收益率的自相关与偏自相关 (ACF & PACF)	169
4.15 2023 年上证综合指数对数收益率以及 2023 年 12 月 13 日至 12 月 29 日上 证综合指数对数收益率预测以及预测的置信区间	170
4.16 模拟的时间序列数据	181

5.1 波动聚集现象示例：美元兑人民币汇率的对数日收益率（2018 年 1 月 1 日 - 2024 年 12 月 31 日）	188
5.2 不同时间序列过程的分类	189
5.3 新息冲击曲线	197
5.4 DCC-GARCH 模型的动态条件相关性	214
7.1 均匀核、高斯核、Epanechnikov 核以及四次方核函数	236
7.2 DAX 指数日收益率的核密度与直方图估计	237
7.3 核密度估计的可视化	246
7.4 线性回归与非参数回归的拟合结果及残差图	248
7.5 使用 Nadaraya–Watson 估计量评估马力对每加仑英里数 (MPG) 的影响	250
7.6 Nadaraya-Watson 估计量与局部线性估计量的比较	255
8.1 Wald 检验、似然比 (LR) 检验与 Lagrange 乘数 (LM/Score) 检验的关系	268
10.1 1 个月与 120 个月收益率 (2000–2024 年, FRED 常数到期收益率; 纵轴为百分比)。可见长期平均水平更高、短端在危机期间贴近零下限并在 2022 年后快速上行。	346
10.2 1 个月收益率的日变化 (2000–2024 年, 单位: 百分点)。2008–2009 年与 2020 年附近波动显著放大, 随后总体回落, 反映短端政策与流动性冲击的主要时点。	346
10.3 3M (日频) 收益率的四个条件累积量的非参数估计 (局部线性核; 带宽分别为 1、2、4、8 倍 Silverman 经验法则)。上左: 条件均值近似线性、斜率接近 1; 上右: 条件方差随水平显著非线性; 下左与下右: 条件偏度与条件超峰度在不同利率区间呈现明显的非线性变化。	352
10.4 10Y (月频) 收益率的四个条件累积量的非参数估计 (局部线性核; 带宽分别为 1、2、4、8 倍 Silverman 经验法则)。条件均值基本线性; 条件方差、条件偏度与条件峰度仍存在非线性, 但幅度相对 3M 更温和。	353

目录

第一章 R 语言概述	1
1.1 R 语言简介	1
1.1.1 R 语言概述	1
1.1.1.1 S 语言	1
1.1.1.2 R 语言	2
1.2 R 语言的集成开发环境——RStudio	3
1.3 R 语言编程基础	6
1.3.1 简单数学运算	6
1.3.2 数学常数	7
1.4 R 中的循环	7
1.4.1 For 循环	7
1.4.2 While 循环	8
1.4.3 Repeat 循环	8
1.4.4 循环的替代方法	9
1.4.4.1 lapply	9
1.4.5 R 中的函数	10
1.5 数据结构	13
1.5.1 向量	13
1.5.2 列表	13
1.5.3 数据框	14
1.5.4 矩阵	14
1.5.5 数组	15
1.5.6 因子	15
1.6 数据对象的导入、预处理与存取	15
1.6.1 导入数据对象	15
1.6.2 数据预处理	18
1.6.2.1 子集提取	18
1.6.2.2 过滤	18
1.6.2.3 汇总	18
1.6.2.4 插补缺失值	19
1.6.3 导出数据对象	20
1.7 R 语言中的数据可视化	20
1.7.1 基本折线图	21
1.7.2 改进折线图	21
1.7.3 R 中的其他图形	22

1.8 使用 R 生成随机数	25
1.8.1 使用内置函数生成随机数	25
1.8.2 采用分位数转换法生成随机数	25
1.9 使用“帮助”选项卡	28
1.10 tidyverse 系列软件包	30
1.10.1 dplyr 软件包	30
1.10.2 tidyr 包	34
1.10.2.1 gather () 函数	34
1.10.2.2 separate() 函数	37
1.10.2.3 unite () 函数	38
1.10.2.4 spread () 函数	40
1.10.2.5 nest() 与 unnest() 函数	41
1.10.2.6 数据清洗函数	41
1.10.3 ggplot2 包	42
1.10.3.1 初始化数据层	42
1.10.3.2 完善美学层	42
1.10.3.3 设置几何层	42
1.10.3.4 扩展面板层	43
1.10.3.5 应用统计层	43
1.10.3.6 调整坐标层	43
1.10.3.7 定义主题层	44
1.10.3.8 使用 ggplot2 包绘制各类图表	44
1.10.3.9 保存和提取图表	46
1.11 通过 API 获取开放数据（以世界银行为例）	46
1.11.1 在 R 中访问 API 的两条路径	46
1.11.2 示例：世界银行 API	49
1.11.2.1 方式 A：使用 WDI 包（推荐）	49
1.11.2.2 方式 B：直接调用世界银行 API (httr2 + jsonlite)	50
1.11.3 其他常用公共数据 API 与 R 包速览	51
1.11.4 实践要点与小结	53
1.12 章节总结	53
1.13 习题	54
第二章 引言和背景	57
2.1 金融市场	57
2.1.1 货币市场	57
2.1.2 债券市场	59
2.1.3 股票市场	60
2.2 市场类型以及交易方式	61
2.3 收益率	63
2.3.1 股票指数的编制方法	66
2.3.2 收益率的统计性质	68
2.4 金融经济学基础	70
2.4.1 效用函数以及风险厌恶	71
2.4.2 基于均值-方差框架的投资组合模型	78

2.4.2.1	不存在无风险资产的情形	79
2.4.2.2	存在无风险资产的情形	82
2.4.3	资本资产定价模型	82
2.4.3.1	夏普—林特纳版本的 CAPM 模型	82
2.4.3.2	布莱克版本的 CAPM (零贝塔 CAPM)	83
2.4.4	套利定价理论	84
2.4.5	基于消费的资本资产定价模型	84
2.5	章节总结	86
2.6	习题	87
第三章 线性回归模型		91
3.1	一元线性回归模型	91
3.1.1	一元线性回归模型的基本假设	93
3.1.2	普通最小二乘法	93
3.1.3	学生 t 检验与决定系数 (R^2)	96
3.2	多元线性回归模型	98
3.2.0.1	F 统计量	103
3.2.0.2	调整决定系数 (Adjusted R^2)	104
3.2.0.3	残差标准差 (RSE)	104
3.3	异方差性与多重共线性	105
3.4	线性回归模型相关案例	107
3.4.1	案例：汽车重量是否对燃油效率产生负面影响？	107
3.4.2	案例：比亚迪股票的风险—收益关系	109
3.4.3	案例：Fama-French 因子模型下的个股超额收益 OLS 回归	113
3.5	章节总结	119
3.6	习题	120
第四章 自回归移动平均模型		123
4.1	ARMA 模型	123
4.1.1	平稳性	123
4.1.2	遍历性和混合性条件	125
4.1.3	自回归 (AR) 模型	131
4.1.4	移动平均 (MA) 模型	132
4.1.5	滞后算子 L	133
4.1.6	自回归移动平均过程	135
4.1.7	自相关系数	137
4.1.7.1	AR 模型的自相关结构	137
4.1.7.2	MA 模型的自相关结构	138
4.1.8	偏自相关系数	139
4.1.9	使用 ACF 与 PACF 判断 ARMA 模型的阶数	140
4.1.10	ARMA 模型的估计	141
4.1.10.1	利用尤尔—沃克方程对 AR 模型进行参数估计	141
4.1.11	极大似然法初探	142
4.1.11.1	ARMA 过程的极大似然估计	144
4.1.12	ARMA 过程的统计推断	146

4.1.12.1 基于 AIC 和 BIC 的 ARMA(p,q) 阶数选择	146
4.1.13 极大似然估计量的优良统计性质	148
4.1.13.1 与极大似然估计相关的三个重要概念	149
4.1.13.2 R 中验证极大似然估计的性质	153
4.1.14 单位根	154
4.1.14.1 ADF 检验	159
4.2 结构模型与时间序列模型之间的关系	160
4.3 长期方差	161
4.4 采用 ARMA 模型进行预测	163
4.5 案例：采用 ARMA 模型对上证指数收益率进行建模及预测	166
4.6 向量自回归模型	170
4.6.1 VAR 模型的平稳性条件	171
4.6.2 VAR 模型的极大似然估计	172
4.6.3 VAR 模型的阶数选取	173
4.6.4 格兰杰因果检验	173
4.6.5 脉冲响应函数以及正交脉冲响应函数	175
4.7 结构向量自回归模型	177
4.8 章节总结	182
4.9 习题	183
第五章 波动率模型	187
5.1 ARCH 模型	189
5.1.1 ARCH 过程的平稳性	192
5.2 GARCH 模型	193
5.2.1 GARCH 过程的平稳性	195
5.3 非对称波动率模型	196
5.4 IGARCH 模型	198
5.5 ARCH、GARCH 模型的估计	200
5.6 ARCH、GARCH 模型的诊断检验	203
5.6.1 拉格朗日乘子检验 (LM 检验)	204
5.6.2 Box-Pierce 型混成检验	205
5.7 多元波动率模型	206
5.7.1 案例：基于 BEKK-GARCH 的股指期现货的波动率研究	206
5.7.2 案例：基于 DCC-GARCH 模型的股指联动性分析	210
5.8 章节总结	213
5.9 习题	214
第六章 收益可预测性与有效市场假说	217
6.1 有效市场假说	217
6.2 有效市场假说的提出与发展	217
6.2.1 弱有效市场的检验	218
6.2.2 强有效市场的检验	219
6.3 随机游走价格模型	221
6.4 对 EMH 的假设检验	223
6.4.1 序列相关性检验	223

6.4.2 方差比检验	225
6.4.3 依照 AR 模型对 EMH 进行检验	227
6.4.4 对 rw2 与 rw3 进行假设检验	227
6.5 案例：2023 年中国股市的弱式有效性检验——以贵州茅台为例	228
6.6 章节总结	231
6.7 习题	231
第七章 非参数方法	233
7.1 非参数概率密度估计量	233
7.1.1 单维核密度估计量	238
7.1.1.1 核估计量的偏差	239
7.1.1.2 核估计的边界问题	240
7.1.1.3 核估计的方差	241
7.1.1.4 均方误差和最优带宽	242
7.1.2 多维核密度估计量	243
7.2 非参数回归模型	246
7.2.1 Nadaraya-Watson 估计量	248
7.2.1.1 均方误差和最优带宽	250
7.2.1.2 Nadaraya-Watson 估计量的边界问题	252
7.2.2 局部多项式估计量	254
7.3 非参数统计在金融计量经济学中的应用	257
7.3.1 游程检验	257
7.3.2 非线性可预测性和非参数自回归模型	258
7.3.3 半参数波动率模型	260
7.4 章节总结	261
7.5 习题	261
第八章 金融资产定价模型	263
8.1 资本资产定价模型 (CAPM)	264
8.1.1 模型估计	264
8.1.2 模型检验	265
8.1.2.1 沃尔德 (Wald) 检验	265
8.1.2.2 似然比检验	266
8.1.3 时变 CAPM	267
8.1.4 条件资本资产定价模型	268
8.2 多因子模型	269
8.2.1 以投资组合收益为因子的多因子定价检验	271
8.2.2 含宏观因子和无风险利率的多因子模型的检验	272
8.2.3 案例：Fama-French 三因子模型	273
8.2.4 SMB/HML/UMD 的基本构建思路（实践口径可按研究设计微调） .	276
8.3 基于消费的资本资产定价模型	281
8.3.1 广义矩估计	281
8.3.2 采用广义矩估计方法估计 C-CAPM 模型	282
8.3.3 时变风险厌恶与股权风险溢价之谜	283
8.4 章节总结	284

8.5 习题	284
第九章 连续时间模型与高频波动率估计	287
9.1 布朗运动	287
9.2 随机微分	290
9.3 布朗运动的其他性质	291
9.4 伊藤引理 (Itô's Lemma)	293
9.5 随机积分	293
9.6 扩散过程	294
9.7 扩散过程的极大似然估计	296
9.7.1 采用极大似然估计法估计布莱克-舒尔斯 (Black-Scholes) 模型	297
9.7.2 Aït-Sahalia (1996, 2002), Aït-Sahalia et al. (2009) 提出的近似方法	299
9.7.2.1 第一次变换: $X \rightarrow Y$	299
9.7.2.2 第二次变换: $Y \rightarrow Z$	299
9.7.2.3 第三次变换: $Y \rightarrow X$	300
9.7.2.4 极大似然估计	300
9.7.2.5 连续时间序列模型的模型设定检验	301
9.7.3 案例: 基于 Aït-Sahalia 似然展开的扩散模型近似极大似然估计与稳健推断	301
9.8 从高频数据中估计波动率	306
9.9 测量误差模型	307
9.9.1 基于 Yahoo Finance 日频数据的已实现波动率估计	312
9.10 高频协方差矩阵估计方法	314
9.10.1 双频已实现协方差估计量 (TSRV)	315
9.10.2 多元已实现核协方差估计量 (RK)	317
9.10.3 预平均协方差估计量 (PAV)	319
9.11 章节总结	322
9.12 习题	323
第十章 收益率曲线 (Yield Curve)	325
10.1 贴现函数、收益率曲线与远期利率	325
10.2 由息票债估计收益率曲线	329
10.2.1 基函数展开法	330
10.2.2 参数法	333
10.2.3 Fama-Bliss 法	338
10.3 离散时间的债券定价模型	342
10.3.1 利率的经济学假说	342
10.3.2 收益率的统计性质	342
10.4 无套利与随机贴现因子 (SDF)	347
10.4.1 Vasicek 模型 (离散时间仿射形式)	347
10.4.2 Cox-Ingersoll-Ross (CIR) 模型	348
10.4.3 多因子仿射期限结构	348
10.5 本章小结	354
10.6 习题	354

第十一章 风险管理与尾部风险估计	357
11.1 在险价值 (VaR)	358
11.2 极值理论 (EVT)	360
11.3 尾部厚度的半参数模型	362
11.3.1 尾厚度的估计	363
11.4 动态模型与 VaR	368
11.4.1 VaR 模型的评估	369
11.5 尾部厚度与动态风险度量：基于 S&P 500 的实证与回测	369
11.6 多变量情形	377
11.6.1 多变量依赖与系统性风险：Copula 与 CoVaR 的实证演示	378
11.7 一致性风险度量	382
11.8 期望损失 (ES)	383
11.8.1 期望损失 (ES)：历史模拟法与条件 GARCH 模型的实证计算	384
11.9 黑天鹅、龙王与灰犀牛	388
11.10 本章小结	389
11.11 习题	390
第十二章 AI 与金融计量：模型、应用与实践	393
12.1 树模型（随机森林与 XGBoost）的原理及适用性	394
12.1.1 决策树模型	394
12.1.1.1 分类树：基尼指数与信息增益	394
12.1.1.2 回归树：最小化残差平方和 / 最大化方差减小	395
12.1.1.3 统一的不纯度下降视角	395
12.1.1.4 预测与规则形式	395
12.1.1.5 正则化与剪枝：超参数的含义与作用	395
12.1.1.6 如何使用与调参建议	396
12.1.1.7 金融数据中的建模要点	397
12.1.1.8 与集成学习的承接	397
12.1.1.9 示例（量化 / 风控）	397
12.1.2 随机森林	401
12.1.2.1 构建机制与集成原理	401
12.1.2.2 袋外误差与时间序列评估	402
12.1.2.3 特征重要性与可解释性	403
12.1.2.4 金融场景下的优势与局限	403
12.1.3 梯度提升树与 XGBoost	403
12.1.3.1 提升思想与可加性模型	403
12.1.3.2 XGBoost 的目标函数与正则化	404
12.1.3.3 可定制损失函数与业务约束	404
12.1.3.4 与随机森林的差异与选型	405
12.1.3.5 与金融计量的衔接与注意事项	405
12.1.3.6 与随机森林的差异与选型	406
12.1.3.7 案例：随机森林 vs 逻辑回归（固定窗口，滚动样本外预测）	406
12.1.3.8 代码说明：XGBoost（固定区间，滚动样本外，早停）	411
12.2 神经网络与深度学习方法概述及金融场景应用	417
12.2.1 背景与动机	417

12.2.2 基本模型与经验风险最小化	417
12.2.3 优化与训练细节	418
12.2.4 正则化与归一化	418
12.2.5 序列建模：RNN、LSTM 与 GRU	419
12.2.6 自注意力与 Transformer	419
12.2.7 表示学习与生成模型	419
12.2.8 概率预测、校准与风险度量	419
12.2.9 金融任务与建模要点	420
12.2.10 样本外评估与非平稳性	420
12.2.11 可解释性与经济含义	420
12.2.12 模型治理与合规落地	420
12.3 超参数调优与交叉验证：金融计量视角	421
12.3.1 问题形式化与风险估计	421
12.3.2 嵌套交叉验证与时序滚动验证	421
12.3.3 超参数搜索：网格、随机、贝叶斯与多臂策略	422
12.3.4 金融目标与自定义损失	422
12.3.5 早停、正则与模型选择	423
12.3.6 避免过度搜索与检验偏差	425
12.3.7 随机森林、XGBoost、深度网络的调参要点	426
12.3.8 一个面向时序的调参与评估流程（建议）	426
12.3.9 实践附注：评价指标与报告规范	426
12.4 案例：AI 模型的典型应用场景	426
12.4.1 收益率预测：线性回归 vs. XGBoost 与 LSTM	427
12.4.2 波动率建模：GARCH vs. 注意力机制	428
12.4.3 信用评分：Logit vs. 随机森林 / 神经网络	434
12.4.4 因子提取与风险度量：自编码器与深度贝叶斯	440
12.5 模型可解释性与金融稳定性	447
12.5.1 AI 模型的可解释性挑战	447
12.5.2 可解释性工具：SHAP / LIME / PDP / ICE	447
12.5.3 稳定性与合规	448
12.6 本章小结	449
12.7 习题	450

作者序

金融计量经济学关注金融市场、金融机构和金融工具中的经济规律，核心任务是用可检验的模型和数据证据回答“价格如何形成、风险如何度量、政策如何传导”等问题。本书以“问题——方法——数据——实现”的思路展开：先明确金融问题，再选取合适的计量工具，匹配可获取的数据，最后给出可复现的实现与解读。

全书覆盖的计量经济学方法从入门到进阶，既包括回归分析、时间序列分析，也包括ARCH/GARCH模型及其扩展、非参数方法、资产定价模型、连续时间模型与收益率曲线模型；同时结合风险管理与尾部风险估计，并在最后一章介绍人工智能与金融计量经济学的结合实践。章节安排与读者学习路径相匹配：前几章快速建立计量经济学与编程基础，中段围绕时间序列分析与资产定价理论搭建模型框架，随后深入连续时间模型与收益率曲线建模，最后回到风险管理与人工智能应用，形成相对完整的知识体系闭环。

本书以R语言为主要实现环境，原因在于其数据处理、统计建模与可视化生态系统完备，且易于复现实证结果。书中代码力求简洁、可运行、可修改，读者可在不同数据源（如FRED、WRDS、Yahoo Finance、World Bank等）之间灵活切换，不受特定平台约束。需要强调的是，编程只是手段——我们的重点仍然是“正确提问、恰当建模、谨慎解释”。

关于数据与案例，书中尽量选取公开可获取的市场与宏观数据，以便读者复现与拓展；涉及交易成本、制度背景或口径差异时，会在相应章节给出必要的说明与参考链接。对于模型结果，始终建议在“经济含义——统计检验——稳健性分析——实际可用性”四个维度进行审视，避免仅凭单一指标判断优劣。

读者对象包括经济与金融专业的研究生与高年级本科生，以及需要在工作中开展量化分析的从业者。建议具备基础概率论与数理统计、线性代数知识，熟悉回归分析与时间序列分析的基本概念；编程经验并非必须，但将帮助您更高效地完成练习与项目。

学习建议方面：每章均配有可运行代码与练习题。建议在阅读公式推导的同时，亲手运行代码，替换为您熟悉的市场和时间段，比较不同样本、不同口径和不同评估指标下的结果差异。这样可以更快建立“从理论到数据，从模型到解释”的直觉。

本书难免有疏漏，诚挚欢迎读者批评指正并提出改进建议。来信请联系：
jiajing.sun@gmail.com。您每一条建议，都是我们完善本书的动力。

孙佳婧 洪永森 奥利弗·林顿
2025年10月2日

汪寿阳教授友情作序

本世纪以来，金融计量经济学的发展十分迅速，尤其是大数据与人工智能的兴起，为该领域带来了新的机遇与挑战。《金融计量经济学：理论、案例与 R 语言》一书的出版，必将有力推动金融计量经济学知识在中国的传授与学习，进而促进其理论与方法的创新与广泛应用。

洪永淼教授是国际知名的计量经济学家，在计量经济学与时间序列分析等领域成果卓著。他发展了非参数模型设定检验理论与广义谱分析等方法，在学术界产生了重要影响。在事业巅峰时期回国后，他先后领导中国科学院大学经济与管理学院、中国科学院预测科学研究中心的建设，为我国金融计量经济学与预测科学的发展作出重要贡献。他出任中国科学院大学经济与管理学院院长后，为学院的高质量发展注入了新的思路与活力。

本书由洪永淼教授、我的同事孙佳婧副教授以及剑桥大学经济系主任 Oliver Linton 教授合作完成。我们的理论研究合作由来已久，并建立了深厚友谊。2024 年初，我们的论文发表于计量经济学顶级期刊 *Journal of Econometrics*。本书系统覆盖金融计量经济学的基础理论与方法，内容由经典模型延伸至前沿应用，体系完整、条理清晰。书中以大量实际案例将理论与实践紧密结合，有助于读者深化对相关知识的理解与运用。同时，书中详细介绍了 R 语言在金融计量中的应用，配以丰富的代码示例和详尽说明，显著降低了读者学习使用 R 语言开展金融分析的门槛。对于经济学和金融学专业的学生而言，此书是系统学习金融计量经济学的优质教材，可为其学术研究与职业发展奠定坚实基础；对于金融领域的从业者和研究人员而言，本书提供的案例与方法具有很强的实用性，有助于应对实际工作中问题与挑战，支撑更为科学的决策。

我相信，《金融计量经济学：理论、案例与 R 语言》的出版，必将对我国金融计量经济学的教学、研究与实践产生重要影响。在此，我郑重向各高校经济学院、金融学院与商学院推荐此书。

汪寿阳
中国科学院特聘研究员
中国科学院大学经济与管理学院教授
发展中国家科学院院士
2025 年 1 月

作者简介



孙佳婧, 博士、特许金融分析师 (Chartered Financial Analyst, 简称 CFA)、英国数学及其应用学会会士 (Fellow of the Institute of Mathematics and its Applications, 简称 FIMA)。2011 年 9 月于英国利物浦大学获得管理学博士学位。现任中国科学院大学经济与管理学院统计与数据科学系副主任兼教研室副主任, 曾负责筹备 2022 年和 2023 年亚洲计量经济学与统计学暑期学校, 承担与国际知名学者联络、会议后勤等工作。教学方面, 因教学与研究成果获得青年教师教学特别奖、研究生优秀课程奖等; 其指导的多位学生获得“三好学生”“优秀共青团员”“优秀学生干部”等荣誉。科研方面, 论文发表在《Journal of Econometrics》《Journal of Environmental Management》《Economics Letters》《Environment and Planning C: Politics and Space》《Journal of Multivariate Analysis》《统计研究》《应用概率统计》等期刊, 并多次在国际会议报告研究成果。主持多个国家自然科学基金项目 (含《基于调整样本值域的自正则时间序列分析》面上项目), 并参与多个横向课题, 在金融监管、经济与数字科技领域的人才培养及金融科技研究方面取得积极成果。

洪永淼, 厦门大学物理学学士、经济学硕士, 美国加州大学圣地亚哥分校经济学博士。现任中国科学院数学与系统科学研究院、中国科学院预测科学研究中心特聘研究员, 中国科学院大学经济与管理学院特聘教授, 《计量经济学报》联合主编。系发展中国家科学院院士、世界计量经济学会会士, 国际应用计量经济学会会员, 里米尼经济分析中心高级会士; 兼任中国教育部高等学校经济学类专业教学指导委员会副主任委员、中国光大银行独立董事。在全职回国前, 曾任康奈尔大学经济学系 Ernest S. Liu 讲席教授及统计学与数据科学系教授, 也曾在清华大学、厦门大学任职。研究领域包括计量经济学、时间序列分析、金融计量经济学与统计学。已在国内外主流经济学、金融学与统计学期刊发表 130 余篇论文, 并出版多部中英文著作; 连续八入选 Elsevier 经济学/统计学中国高被引学者。



奥利弗·林顿 (Oliver Linton), 英国剑桥大学三一学院院士、政治经济学教授, 英国社会科学院院士, 计量经济学会和数理统计学会会士。主要研究领域为非参数方法、半参数方法及实证金融, 曾在《Econometrica》《Journal of Econometrics》《Econometric Theory》等国际顶级经济学期刊发表百余篇论文。其研究覆盖广泛的计量经济方法论, 特别在非参数与半参数方法和金融计量经济学方面贡献突出。职业生涯中, 曾在伦敦政治经济学院、耶鲁大学等知名学术机构担任重要职位, 通过教学与科研为学界作出重要贡献; 在加性回归模型与金融领域波动率模型等方面的研究, 为理解复杂经济与金融现象开辟了新的方法路径。

1 R 语言概述

由于本书在介绍金融计量经济学理论的同时，也介绍 R 语言在金融计量经济学分析中的各类应用场景，因此本章对 R 语言作概述性介绍。R 语言是一种流行的编程语言与开发环境，在金融计量经济学研究与实践中尤为重要。

首先，R 语言在金融计量经济学中的重要性体现在其强大的数据分析与可视化能力。面对不断变化的金融市场与经济环境，我们需要对数据进行分析与预测。R 语言具备处理海量数据的能力，可开展多种统计分析，如回归分析、时间序列分析与贝叶斯分析等，并能直观呈现金融数据的可视化结果，便于理解与解释。

其次，R 语言的应用范围广泛，可用于风险管理、投资组合优化、衍生品定价、金融产品设计与金融市场预测等多个领域。例如，在风险管理中，R 语言可用于评估不同投资组合的风险水平，帮助投资者做出更明智的决策；在衍生品定价中，R 语言可用于估算衍生品价格，并借助有效的对冲策略降低市场风险。

最后，R 语言的开源特性使其在学术研究中具有独特优势。作为开源编程语言，R 允许研究者自由共享与修改代码，促进学术合作，推动学科的发展与创新。

值得注意的是，本书对 R 的讲解以问题为导向，侧重展示其在金融计量经济学分析中的应用。若读者希望系统学习 R 软件，建议同时参考以 R 语言教学为主的专著，例如：

1.1 R 语言简介

1.1.1 R 语言概述

R 语言是一种用于统计计算和图形绘制的编程语言，由 Ross Ihaka 与 Robert Gentleman 于 1993 年创建。它最初被开发为 S 语言的免费替代品。S 语言由贝尔实验室研发，是一种面向数据分析与可视化的统计语言，但早期使用需要付费授权。从某种意义上说，R 可视为 S 语言的一种“方言”。¹ 因而，理解 R 的发展历史，需先对 S 语言有所了解。

1.1.1.1 S 语言

S 语言由约翰·钱伯斯 (John Chambers)、里克·贝克尔 (Rick Becker) 和阿伦·威尔克斯 (Allan Wilks) 在贝尔实验室共同研发。最初，S 于 1976 年以 Fortran 库 (Fortran libraries) 的一部分、作为内部统计分析环境的形式启动。² 值得一提的是，早期 S 版本甚至尚未包含用于统计建模的函数。

1980 年，S 的第一个版本开始在贝尔实验室之外分发，1981 年其源代码版本也对外提供。1984 年，团队出版了两部重要著作：《S：一种用于数据分析与图形的交互式环境》

¹S 与 R 的渊源可参见：<https://stat.ethz.ch/~www/SandR.html>

²SPSS 与 SAS 的早期版本同样以可被其他程序（如 Fortran）调用的子程序形式提供。

(Becker & Chambers 1984) 与《扩展 S 系统》(Becker & Chambers 1985)。同年，S 的源代码亦通过 AT&T 以授权形式对教育与商业用途进行销售。

1988 年，S 的核心实现以 C 语言重写，形成通常所称的“S3”系统；随后 S 继续演进，1998 年引入了更为严格的对象系统（通常称为“S4”）。关于 S 在统计建模方面的系统性介绍，可参阅《Statistical Models in S》(Chambers & Hastie 2017) 以及钱伯斯的《Programming with Data: A Guide to the S Language》(Chambers 1998)。同年，S 系统及其实现因对软件生态的突出贡献获得了 ACM 软件系统奖³。

自 20 世纪 90 年代初以来，S 的产业化进程颇为曲折：1993 年，贝尔实验室将 S 的独家开发与销售许可授予 StatSci（后更名 Insightful Corp.）；2004 年，Insightful 以 2000 万美元收购 S 的全部所有权。其后续产品命名为 S-PLUS，意即在 S 的基础上加入了丰富的图形用户界面（GUI）与工具集。⁴ 2008 年，TIBCO 以 2500 万美元收购 Insightful，自此成为 S 的所有者与官方开发方。

1.1.1.2 R 语言

R 语言最初由罗斯·伊哈卡（Ross Ihaka）和罗伯特·詹特尔曼（Robert Gentleman）在新西兰奥克兰大学（University of Auckland）统计系开发，其设计初衷是提供一种便于进行数据分析与可视化的编程语言。“R”这一名称源自两位创始人名字的首字母。自创建以来，R 不断发展壮大，逐渐演变为功能强大且灵活的数据科学工具，广泛应用于统计建模、数据挖掘、可视化、机器学习、生物信息学与金融分析等领域。

R 的起源可追溯至 20 世纪 90 年代初。当时，Ihaka 教授为学生与同事设计一款易用的统计软件，希望通过开源、跨平台的工具摆脱对商业统计软件的依赖。Ihaka 以 S 语言为基础进行修改与扩展，形成了 R 的早期原型。统计学家马丁·梅克勒（Martin Mächler）建议 Ihaka 与 Gentleman 将 R 的源代码以自由软件形式开放，并依据 GNU 通用公共许可证（GNU General Public License，简称 GNU GPL）发布。1995 年，他们发布了第一个公开版本，并在奥克兰大学开设了 R 课程，随后 R 在统计与数据科学界迅速普及，成为主流统计计算平台之一 (Ihaka 1998)。

R 核心开发团队（R Core Team）于 1997 年正式成立，负责 R 的开发、管理与维护 (Fox 2009)。同年，Kurt Hornik 与 Fritz Leisch 创建了综合 R 档案网络（The Comprehensive R Archive Network，简称 CRAN），用于托管 R 的源代码、可执行文件、文档以及用户贡献的软件包 (Hornik 2012)。CRAN 的命名与结构借鉴了综合 TeX 档案网络（CTAN）与综合 Perl 档案网络（CPAN），旨在为 R 社区提供统一、高效的软件分发平台。CRAN 最初仅包含 3 个镜像站与 12 个用户贡献扩展包；截至 2022 年 12 月，已发展为拥有 103 个镜像站与 18 976 个贡献包的重要开源生态系统。

2003 年，R 发布稳定版本，标志其开发进入成熟阶段；同年 4 月，非营利组织 R 基金会（The R Foundation）成立，为 R 项目的持续发展与推广提供支持。2004 年，R 项目在 GNU 通用公共许可证框架下获得开源授权，进一步巩固了其作为自由软件的地位。R 的官方下载地址为：<https://cran.r-project.org/>。目前，R 提供适用于多种操作系统的安装版本，包括 Linux（如 Debian、Fedora/Red Hat、Ubuntu）、macOS 与 Windows，读者可按所用系统选择相应版本安装。

³ ACM 为 Association for Computing Machinery（计算机协会）。该奖项旨在表彰对计算机软件系统作出杰出贡献的个人或团队。

⁴S-PLUS 提供数据编辑器、绘图界面、模型拟合与数据探索等 GUI，以便快速完成分析并生成可视化结果。

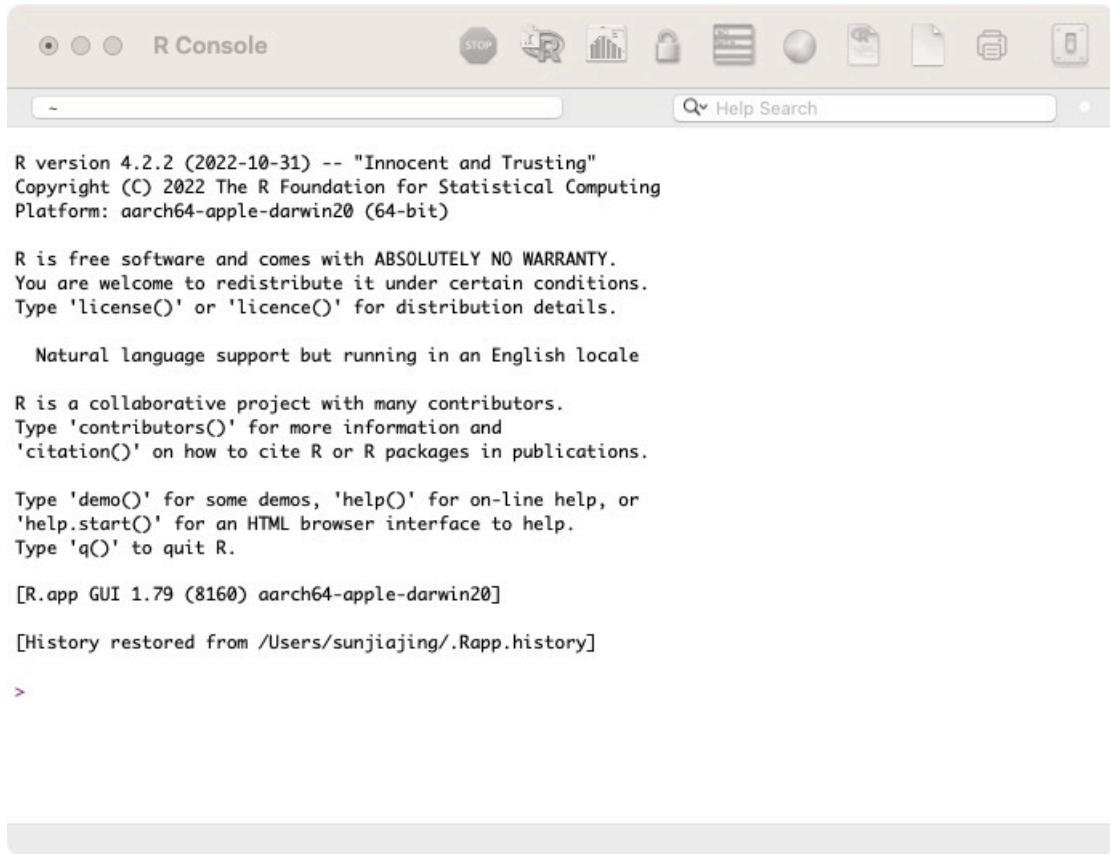


图 1.1: R 软件界面 (版本 4.2.2)

1.2 R 语言的集成开发环境——RStudio

如果您偏好使用集成开发环境 (IDE)，那么强烈推荐安装 RStudio。RStudio 是一款专为 R 语言开发的 IDE，由 Posit PBC (共益企业) 开发。⁵ RStudio 为 R 语言开发者提供了一个便捷、高效、可扩展的开发环境，显著提升编程效率与代码质量。对于 R 语言编程初学者或习惯于图形界面操作的用户来说，RStudio 提供了直观友好的操作体验。与其他编程语言类似，R 语言可通过用户自定义函数进行扩展与开发。为进一步促进 R 语言的编程实践，Posit 设计并推出了这一集成开发平台。RStudio 提供两个版本：RStudio Desktop 和 RStudio Server，均包含免费版与商业版，用户可根据自身需求选择合适的版本进行安装与使用。

RStudio 的首席科学家是 Hadley Wickham。他是 R 社区最受尊敬的数据科学家之一，也是众多广泛使用的 R 扩展包的主要作者与贡献者。RStudio 的第一个版本于 2011 年发布，最初以开源项目面向社区。此后，RStudio 发展为功能强大且广受欢迎的 R 语言集成开发环境，深受研究者、分析师和开发者喜爱。RStudio 的公司主体现名为 Posit PBC (原 RStudio, PBC)。公司创始人为 J.J. Allaire；Wickham 担任首席科学家（非联合创始人）。

除了 RStudio IDE 之外，Posit PBC 还开发了多个与 R 语言密切相关的重要工具与扩展包，例如：

- **Shiny**: 用于从 R 构建交互式 Web 应用程序，广泛应用于数据可视化、仪表盘与交互式建模展示；⁶

⁵Posit 公司原名为 RStudio，2022 年更名为 Posit，详见：<https://posit.co/about/>。

⁶Shiny 官方站点：<https://shiny.posit.co/>

- **R Markdown**（文档格式）与 **rmarkdown**（R 包）：将 R 代码与 Markdown 文本融合，可用于撰写报告、学术论文、博客与幻灯片。⁷

这些工具进一步拓展了 R 语言在数据科学、报告生成与可视化方面的应用范围。

什么是 IDE？

IDE (Integrated Development Environment, 集成开发环境) 是为编程人员设计的综合性开发平台。

与许多基于图形用户界面的统计软件不同，R 用户主要通过命令行界面与软件交互，因此 R 的 IDE 必须支持交互式命令输入。在这一点上，R 并非特例，许多其他交互式科学计算语言（如 Python、Julia）也拥有各自成熟的 IDE 系统。

典型的 IDE 除了包含命令控制台外，还具备以下核心功能组件：

- **源代码编辑器**：用于编写与修改代码，支持快捷键、自动格式化、括号匹配、语法高亮、代码折叠、文件导航与运行接口等功能，是程序开发的核心部分；
- **对象浏览器与对象编辑器**：帮助用户查看并识别变量的类型与数值，可快速检查与编辑数据对象；
- **绘图管理与文档集成**：便于管理图形输出，并与文档系统无缝对接（如 R Markdown、Quarto）；
- **调试与版本控制支持**：提供断点、变量追踪、历史记录等调试工具，以及 Git 等版本控制系统的集成。

这些功能的集成极大提升了数据分析与编程的效率，是现代数据科学工作不可或缺的平台。

RStudio 拥有界面友好、功能丰富且易于操作的集成环境，其主界面由四个主要面板构成：

1. **代码编辑器 (Source Editor)**：位于左上角，用于打开、编辑和运行 R 脚本。用户可以在此编写完整的程序文件，并通过快捷键或按钮执行选中或全部代码。
2. **控制台 (Console)**：位于左下角，是用户直接输入代码并即时执行的区域，也称为命令窗口，适合进行交互式编程与测试。
3. **环境面板 (Environment Pane)**：展示当前工作环境中的对象，如数据框、数组、变量和函数等，便于跟踪内存中已有的数据与结构。
4. **右下角多功能面板**：包含多个选项卡：
 - (a) **绘图 (Plots)**：显示通过代码生成的图形输出；
 - (b) **连接 (Connections)**：连接现有数据源，并可辅助生成导入数据的 R 语句；
 - (c) **包 (Packages)**：列出已安装与可安装的 R 包，并支持一键加载或卸载；
 - (d) **帮助 (Help)**：用于检索 R 帮助文档，是调用帮助命令（如 `?mean`）时显示内容的区域。

⁷knitr: <https://yihui.org/knitr/>; rmarkdown: <https://rmarkdown.rstudio.com/>

需要注意的是，上述面板的位置为 RStudio 的默认布局。若需根据个人习惯调整面板位置，可依次点击“Tools”→“Global Options”，在弹出的设置窗口选择“Pane Layout”进行配置。

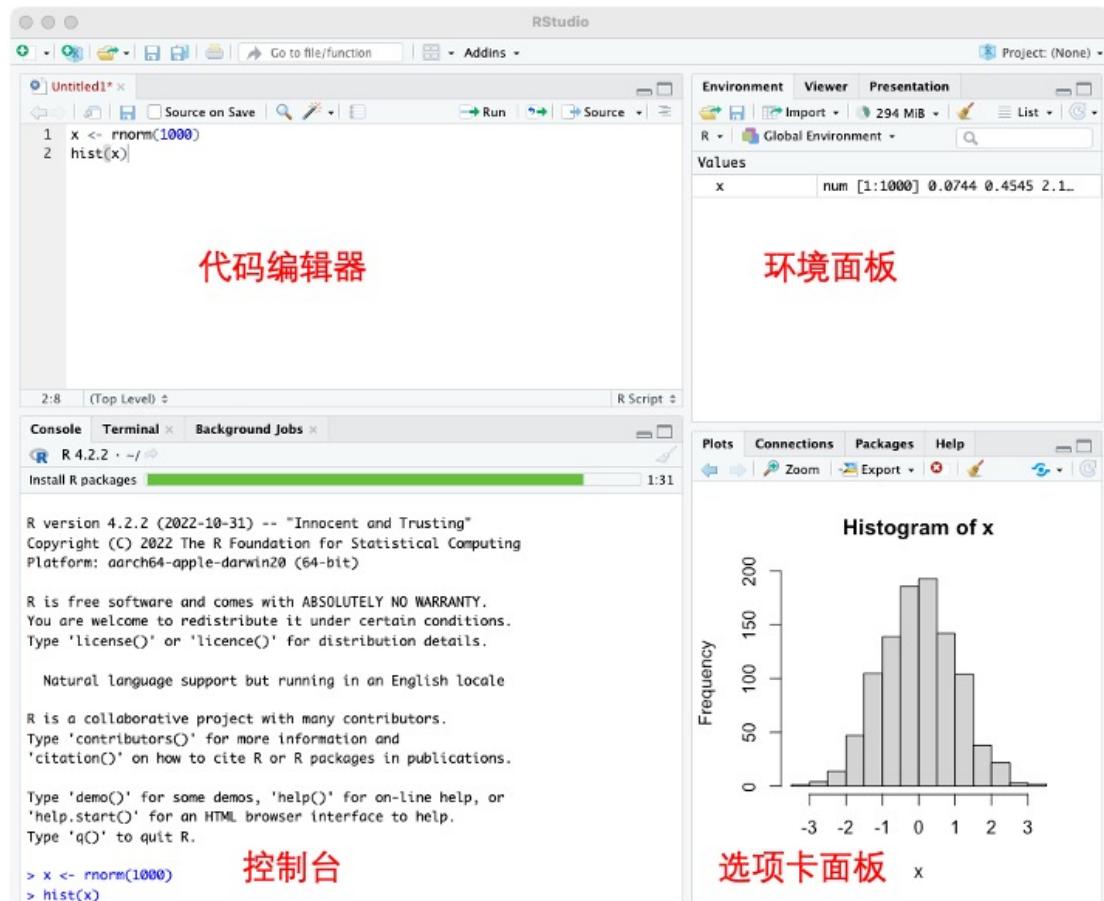


图 1.2: RStudio 的用户界面

RStudio 支持多种操作系统，其支持情况取决于所使用的 IDE 版本。RStudio Desktop 可用于 Windows、macOS 和 Linux 系统；RStudio Server 可运行于 Debian、Ubuntu、Red Hat Linux、CentOS、openSUSE 和 SLES 等主流 Linux 发行版。

用户可以直接在 RStudio 中进行版本管理，支持代码比较、合并与提交等操作。同时，RStudio 集成了包管理工具，便于搜索、安装与更新 R 包。RStudio 还很好地支持多种绘图与报告生成工具，如 `ggplot2`、`Shiny`、`knitr` 与 `rmarkdown` 等。`ggplot2` 能生成多类统计图表（如散点图、直方图、箱线图等）。除 `ggplot2` 外，R 语言还有诸多可视化扩展包（如 `lattice`、`leaflet`、`playwith`、`ggvis`、`ggmaps` 等）。R 亦支持部分地图绘制功能；若对空间数据分析（spatial data analysis）感兴趣，可参考：<https://r-spatial.org/>。

总之，RStudio 是一款功能强大、易于使用的开发环境，集成了多种实用工具与功能，能够有效提升开发效率与代码质量。无论是初学者还是专业数据分析师，都可以在 RStudio 中高效开展 R 项目的开发与管理。

RStudio 下载链接：<https://posit.co/download/rstudio-desktop/>。本书推荐采用 R 与 RStudio 的组合进行学习与实践。

1.3 R 语言编程基础

了解了 R 语言的一些背景历史后，我们现在可以开始学习了！

首先，我们需要安装 R 和 RStudio。请访问 CRAN 官方网站：<https://cran.r-project.org/>。根据所用操作系统选择相应的 R 版本：Windows、macOS 或 Linux（如 Debian、Fedora/Red Hat、Ubuntu）。Windows 用户点击“Download R for Windows”后选择“base”；macOS 用户点击“Download R for macOS”；Linux 用户根据所用发行版进入对应页面。下载完成后，运行安装程序并按照向导完成安装。

获取 RStudio：请访问 <https://posit.co/download/rstudio-desktop/>，按操作系统选择对应的安装包并运行安装程序完成安装。RStudio 安装后会自动检测本机的 R 位置。

请注意：应先安装 R，再安装 RStudio；RStudio 依赖已安装的 R 才能正常工作。

在本章的后续部分，我们将首先介绍 R 编程的基础概念，内容包括：R 环境下的基本数学运算、循环机制的原理，以及如何创建和调用函数。接下来重点讲解数据管理，涵盖 R 中常见的数据结构、导入外部数据的方法、以及数据的可视化与分析技术。此外，我们还将引导读者学习如何高效使用 R 提供的帮助系统，以便在实践中获取函数与包的相关信息。最后，我们将简要介绍 `tidyverse` 软件包，这是一个集成化的数据科学工具集合，在 R 社区具有广泛影响力。

1.3.1 简单数学运算

若要在 R 中尝试基本算术运算，请在 R 控制台中输入以下命令（跳过注释，在每条命令后按下 Enter 键查看结果）。

```

1 # R 中的基本算术
2 # 加法
3 > 8 + 4
4 [1] 12
5 # 减法
6 > 9 - 2
7 [1] 7
8 # 乘法
9 > 7 * 6
10 [1] 42
11 # 除法
12 > 120 / 40
13 [1] 3
14 # 开方
15 > sqrt(64)
16 [1] 8
17 # 指数
18 > 10^2
19 [1] 100

```

下面的例子说明：R 是一门动态类型语言，并在需要时进行类型强制（coercion）。

```

1 # 一个数字与逻辑值相加（逻辑TRUE会被强制为1）
2 > 3 + TRUE
3 [1] 4
4
5 # 检查数组的结构

```

```

6 > str(as.array(1))
7 num [1(1d)] 1
8
9 # 检查数字的结构
10 > str(1)
11 num 1

```

在 R 中，`str` 函数可用于简洁地显示对象的内部结构。对于需要快速概览的数据框、列表或模型等复杂对象而言，该函数尤为实用。处理包含多种数据类型（如数值、因子、字符等列）的数据框时，`str` 会输出列名、数据类型及若干示例值，帮助快速了解数据结构。

1.3.2 数学常数

在 R 中，为了便于计算，系统预定义了若干常见的数学常数。其中最著名的常数之一是欧拉数 e ，它是自然对数的底数。可使用 `exp` 函数进行指数运算，例如 `exp(1)` 返回 e^1 ，即 e 。

另一个重要的常数是 π ，表示圆的周长与直径的比值。在 R 语言中，只需输入 `pi` 即可得到 π 的数值。

此外，R 使用 `Inf` 和 `-Inf` 分别表示正无穷和负无穷。当某些数学操作的结果无法定义（例如“0 除以 0”）时，R 会返回 `NaN`，即“Not a Number”，表示结果不是有效数字。

1.4 R 中的循环

R 语言非常擅长处理重复性任务。为了多次执行一系列操作，我们通常使用“循环”结构。通过循环，R 语言可以按照指定的迭代次数，或直到满足某个特定条件为止，不断执行一组指令。

在 R 语言中，循环结构主要包括三种类型：`for` 循环、`while` 循环和 `repeat` 循环。

循环结构是几乎所有编程语言中的基础功能，R 语言也不例外。它们为自动化任务提供了强大的工具。然而，在 R 编程中，一些开发者认为循环在某些场景下可能被过度使用，特别是在可通过向量化或函数式编程简化代码的情况下。

1.4.1 For 循环

`for` 循环是 R 语言中用于执行重复性任务的常用工具。以下是一个简单的示例：

```

1 for (i in 1:5) {
2   print(i)
3 }

```

该循环使用一个索引变量 `i`，其取值从数字序列 1 到 5（写作 `1:5`）。在 R 语言中，使用 `i` 作为循环变量是一种常见做法，用于表示“迭代”（iteration），但实际上可以使用任意合法的变量名。

循环体中的变量可以在执行过程中被进一步操作。例如，以下代码将在每次迭代中打印 `i + 1` 的结果：

```

1 for (i in 1:5) {
2   print(i + 1)
3 }

```

下面是该循环的功能分解：

1. 初始化一个 `for` 循环，循环变量 `i` 依次从序列 `1:5` 中取值；
2. 在每次循环中，对表达式 `i + 1` 进行求值；
3. 将表达式的结果（即当前 `i` 的值加 1）打印到控制台；
4. 循环继续执行，直到 `i` 遍历完序列 `1:5` 中的所有元素为止。

1.4.2 While 循环

R 语言中的另一种循环是 `while` 循环。与 `for` 循环不同：`for` 在预定义的序列上迭代；而 `while` 只要给定的逻辑条件成立，就会持续执行。

`while` 循环的基本结构如下：

```
while (logical_condition) { expression }
```

一个最小示例（变量从 0 递增到 5）：

```
1 i <- 0
2 while (i <= 5) {
3   print(i)
4   i <- i + 1
5 }
```

该循环结构的功能分解如下：

1. 初始化变量 `i` 为 0；
2. 在每次迭代开始时检查条件 `i <= 5` 是否成立；
3. 若条件成立，则执行花括号 `{}` 中的代码；
4. 在循环体内通过 `i <- i + 1` 使 `i` 递增；
5. 使用 `print(i)` 输出更新后的 `i`；
6. 然后返回循环开头再次检查条件；当 `i` 不再满足条件时，循环终止。

尽管 R 语言的循环结构功能强大，但在处理大型数据集时，效率通常不如向量化或内置函数高。不过，在模拟、递归关系建模、复杂条件判断、需要反复检查某一状态等场景中，`while` 循环仍然非常有用。

1.4.3 Repeat 循环

R 语言中的另一种循环结构是 `repeat` 循环。与 `for` 和 `while` 循环不同，`repeat` 在开始时不进行条件判断，而是持续执行循环体；要结束循环，通常在循环体内设置逻辑判断，并结合 `break` 语句退出循环。

`repeat` 循环的基本结构如下：

```
1 repeat {
2   expression
3 }
```

示例（打印 1 到 5）：

```

1 i <- 0
2 repeat {
3   i <- i + 1
4   if (i > 5) {
5     break
6   }
7   print(i)
8 }
```

该循环的功能如下：

1. 将变量 `i` 初始化为 0；
2. 进入 `repeat` 循环（无初始条件判断）；
3. 在循环体内通过 `i <- i + 1` 使 `i` 递增；
4. 使用 `if` 判断当前 `i` 是否大于 5；
5. 若条件 `i > 5` 成立，则执行 `break` 语句退出循环；
6. 否则使用 `print(i)` 输出 `i` 的当前值；
7. 返回循环起始处，继续执行下一次迭代。

1.4.4 循环的替代方法

在 R 中，通常推荐使用 `apply` 函数家族来替代显式循环结构。这个函数家族包括 `apply()`、`lapply()`、`sapply()`、`vapply()`、`tapply()` 和 `mapply()`。这些函数在执行许多循环能完成的任务时，往往更高效，而且能够显著降低编程错误的可能性。

因此，在创建循环结构时，建议优先考虑是否可以通过 `apply` 系列函数进行重构。如果可实现相同的功能，应优先选择 `apply` 版本。这不仅能够提升代码效率，也能减少因循环产生的细微错误，而这些细微错误往往会在后续处理中被放大，导致难以排查的问题。

1.4.4.1 lapply

对于初学者而言，最常用的 `apply` 函数是 `lapply()`。该函数用于遍历列表中的每一个元素，并对每个元素执行指定的函数。与传统循环相比，`lapply()` 还具有自动返回列表结果的优点，而使用循环实现同样功能通常需要额外的代码编写。

`lapply()` 的基本语法结构如下：

```
1 lapply(X, FUN)
```

其中，`X` 表示需要操作的列表或向量，`FUN` 是要应用于每个元素的函数。

作为一个简单示例，下面的代码使用 `lapply()` 对向量 `0:4` 的每个元素加 1，模拟我们之前用 `for` 循环实现的功能：

```

1 lapply(0:4, function(a) { a + 1 })
2 # [[1]]
3 # [1] 1
4 #
5 # [[2]]
6 # [1] 2
7 ...
```

```
8 # [[5]]
9 # [1] 5
```

请注意，在上述示例中，我们将序列指定为 `0:4`，从而生成长度为 5 的向量，并实现与之前 `for` 循环相同的效果。您可以尝试使用不同的序列，以观察 `lapply()` 在处理不同输入时的表现差异。

此外，您还可以将函数单独定义后，再使用 `lapply()` 将其应用于整个向量，具体如下：

```
1 add_fun <- function(a) { a + 1 }
2 lapply(0:4, add_fun)
```

上述代码的输出结果与前面的例子相同，依然返回一个列表，每个元素为原始序列中对应元素加 1 的结果。

`sapply()` `sapply()` 函数与 `lapply()` 类似，也用于将某个函数作用于向量或列表的每一个元素。不同之处在于，`sapply()` 会尝试将结果简化为向量，而不是列表。例如：

```
1 sapply(0:4, function(a) { a + 1 })
2 # [1] 1 2 3 4 5
```

可以看到，这里返回的是一个向量而不是列表。

无论使用 `lapply()` 还是 `sapply()`，最终的结果都与使用 `for` 循环所得到的输出一致。

1.4.5 R 中的函数

函数是一组组合在一起以执行特定任务的语句集合。R 语言本身提供了大量内置函数，例如 `mean()`、`sum()`、`print()` 和 `lm()` 等，涵盖了从基本运算到回归分析的各类功能。用户也可以根据自身需求自定义函数。

在 R 中定义函数的一般语法如下：

```
1 function_name <- function(arg1, arg2, ...) {
2   # 函数体
3   return(result)
4 }
```

其中，`return` 语句是可选的。R 语言默认会返回函数体中最后一条被求值的表达式结果（这是 R 语言的特性）。

R 语言的函数参数非常灵活。函数可以接受多个参数，其中许多参数可以设定默认值。在调用函数时，参数既可以按位置传递，也可以通过名称指定进行匹配。

R 函数的一个重要特性是它们具有局部作用域。在函数内部创建的变量不会影响全局环境，也不会与其他函数中的变量发生冲突。此外，R 函数还具备闭包（closure）的特性——它们可以“记住”并访问其定义环境中的变量。这种局部作用域与定义环境变量的混合机制，使得 R 函数在组织代码和管理变量作用域方面表现出色。

最后，R 支持函数式编程范式。这意味着函数可以像数据一样被传递给其他函数、作为结果返回，甚至可以存储在列表等数据结构中。这一特性显著增强了 R 语言的灵活性与表达能力。

```
1 # R 中的函数
2 add_numbers <- function(a, b) {
3   result <- a + b
4   return(result)
5 }
6
```

```
7 # 使用函数
8 sum_result <- add_numbers(5, 3)
9 # [1] 8
```

上述示例更具教学意义而非实用价值——毕竟在实际编程中我们通常不会仅为两个数的相加而特意编写一个函数。

接下来，让我们探索一个更具趣味性和实际意义的示例：编写一个函数，用于近似计算欧拉数 e ，即自然对数的底数，也称为自然常数。

自然常数



自然常数 e 的发现归功于 17 世纪末瑞士数学家雅各布·伯努利 (Jacob Bernoulli)，当时他正在研究复利问题。为了在实际情境中展示伯努利的发现，让我们考虑一个账户，最初存入 ¥1.00，年利率为 100%。如果利息在年末一次性结算，账户的总价值将为 ¥2.00。

当利息改为每半年复利一次时，每次的利率为 50%，即初始资金被乘以 1.5 两次，年末账户价值为 ¥2.25。若按季度复利，则为 $\text{¥}1.00 \times 1.25^4 \approx \text{¥}2.4414$ ；若按月复利，则为

$$\text{¥}1.00 \times \left(1 + \frac{1}{12}\right)^{12} \approx \text{¥}2.6130\dots$$

如果我们用 n 表示复利的次数，则每期利率为 $100\%/n$ ，年末账户价值为

$$\text{¥}1.00 \times \left(1 + \frac{1}{n}\right)^n.$$

伯努利注意到，随着 n 越来越大，即复利周期越来越短，该表达式趋于某个极限。例如，每周复利 ($n = 52$) 时，账户价值约为 ¥2.6926；每日复利 ($n = 365$) 时，约为 ¥2.7146。当 $n \rightarrow \infty$ 时，该极限正是我们所说的自然常数 e 。这意味着，在连续复利的极限下，账户最终价值将约为 ¥2.71828。

不过直到 18 世纪初，莱昂哈德·欧拉 (Leonhard Euler) 才正式引入符号 “ e ” 并在数学中广泛推广。欧拉首次证明了 “ e ” 是一个 **无理数**，即它不能表示为两个整数的比。

欧拉还证明了 e 是以下极限的结果：

$$\lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n.$$



同时， e 也等价于以下无穷级数之和：

$$e = \sum_{n=0}^{\infty} \frac{1}{n!} = 1 + 1 + \frac{1}{2!} + \frac{1}{3!} + \dots$$

因此，我们将创建三个函数来近似计算自然常数 e 。

第一个函数将基于表达式

$$\left(1 + \frac{1}{n}\right)^n$$

当 n 趋近于无穷大时的极限，来近似求解 e 。

第二个和第三个函数将基于无穷级数

$$\sum_{n=0}^{\infty} \frac{1}{n!} \quad (\text{这是 } \exp(x) \text{ 在 } x = 1 \text{ 处的泰勒展开})$$

进行近似，其中分别采用不同的实现方式。

以下是相应的 R 语言代码：

```

1 # 首先，使用复利计算方法
2 e_fcn_1 <- function(n = 2000) {
3   (1 + 1 / n)^n
4 }
5
6 e_fcn_1(1)
7 e_fcn_1(100)
8 e_fcn_1(10000) # 随着 n 的增加，我们得到了更接近 e 的近似值
9
10 exp(1) # e 的值
11
12 # 第二，使用无限级数的和来近似 e
13 e_fcn_2 <- function(n = 2000) {
14   e <- 0
15   for (i in 0:n) {
16     e <- e + 1 / factorial(i)
17   }
18   return(e)
19 }
20
21 e_fcn_2(1)
22 e_fcn_2(10)
23
24 # 第三，第二种方法的更简洁版本
25 e_fcn_3 <- function(n = 2000) {
26   e <- sum(1 / factorial(0:n))
27   return(e)
28 }
29
30 e_fcn_3(1)
31 e_fcn_3(10)
```

对数收益率（连续复利收益率）

对数收益率（连续复利收益率）在金融计量经济学中十分重要。当我们衡量一项投资的表现时，算术收益率的计算很直观： $(P_t - P_{t-1})/P_{t-1}$ ；但当资产价格波动较大时，这种做法可能高估收益率。

假设一位投资者初始持有 ¥100 的资产。第一年价格跌至 ¥50，第二年又回升至 ¥100。用算术平均得到的平均收益率为 25%，但投资者实际上并未获得任何净收益；这一方法忽略了复利与波动的影响。

因此，对波动较大的金融资产更适合使用对数收益率，其定义为

$$r_t = \log\left(\frac{P_t}{P_{t-1}}\right).$$

第一年的对数收益率：

$$r_1 = \log\left(\frac{50}{100}\right) \approx -0.6931 \approx -69.31\%.$$

第二年的对数收益率：

$$r_2 = \log\left(\frac{100}{50}\right) \approx 0.6931 \approx 69.31\%.$$

两年的平均对数收益率为 0%（因为 $r_1 + r_2 = 0$ ），与“本金回到初值”的事实一致。对数收益率还具备可加性：若从 0 到 T 的区间被分为若干期，则累计收益率满足

$$\sum_{t=1}^T r_t = \log\left(\frac{P_T}{P_0}\right) \implies \text{累计简单收益率} = \exp\left(\sum_{t=1}^T r_t\right) - 1.$$

这使其在建模与多期汇总时比算术收益率更为方便、稳健。

注：本书统一用 $\log(\cdot)$ 表示自然对数（以 e 为底），即 $\log x \equiv \ln x$ 。

1.5 数据结构

R 语言具有多种基本数据类型，包括字符、数值、整数、复数和逻辑值。除此之外，R 还提供了若干核心数据结构，例如向量、列表、矩阵、数据框和因子。像向量和矩阵这样的结构要求其所有元素具有相同的数据类型，而列表和数据框则可以容纳不同类型的数据。

1.5.1 向量

向量是由相同类型元素组成的有序集合，其长度固定。向量是一种一维的数据结构，R 语言中的大多数对象都可以被视为向量的变体或扩展。

```
1 # 用 R 创建一个数值向量
2 X = c(2, 4, 6, 8, 10)
3
4 # 在控制台打印该向量的元素
5 print(X)
```

请注意，在 R 代码中，# 符号用于添加注释。注释不会被程序执行，但可用于解释代码的用途，增强可读性。

1.5.2 列表

列表是 R 中一种非常灵活的容器，它可以按特定顺序存放多种不同类型的对象。列表中的元素可以是向量、矩阵、字符串、函数等各种形式的数据。您可以将其视为一个有序的容器，每个位置上可以放置不同类型的内容。

```
1 # 用 R 展示一个列表
```

```

2
3 # 员工编号
4 ids <- c(101, 102, 103, 104)
5
6 # 员工姓名
7 names <- c("李华", "王明", "张燕", "刘冰")
8
9 # 员工总数
10 totalEmployees <- 4
11
12 # 整合员工信息为列表
13 employeeList <- list(EmployeeIDs = ids,
14                         Names = names,
15                         TotalEmployees = totalEmployees)
16
17 # 打印员工列表
18 print(employeeList)

```

1.5.3 数据框

数据框是一种用于存储表格格式数据的二维数据结构。它由行和列组成，每一列可以容纳一个不同类型的数据向量，这使得我们可以在同一个数据结构中存储多种类型的信息。

```

1 # 创建一个数据框
2 df <- data.frame(
3   ID = c(1, 2, 3),
4   Name = c("张三", "李四", "王五"),
5   Score = c(85, 90, 78)
6 )
7 print(df)

```

1.5.4 矩阵

矩阵是一种由数字按矩形排列而成的二维数据结构，由行和列构成。矩阵中的所有元素必须具有相同的数据类型，因此它是一种同质的数据结构。矩阵广泛应用于线性代数运算、统计建模及数据变换等场景。

下面是一个在 R 中定义矩阵的示例：

```

1 # 在 R 中定义矩阵
2 M <- matrix(
3   c(1, 2, 3, 4, 5, 6, 7, 8, 9),
4   nrow = 3, ncol = 3,
5   byrow = TRUE
6 )
7 print(M)

```

在该示例中，我们创建了一个 3×3 的矩阵 M。参数 `byrow = TRUE` 指定了矩阵是按行填充的，即数字依次填入每一行，而非默认的按列填充方式。这种结构在多变量分析、回归模型和矩阵代数中有着广泛应用。

1.5.5 数组

数组是 R 语言中一种可以存储多维数据的结构。与矩阵类似，数组也是同质的数据结构，即其中所有元素必须为相同类型。但不同的是，数组支持超过二维的扩展，非常适用于存储和操作三维或更高维度的数据。

例如，一个维度为 $(2, 3, 3)$ 的数组将包含三个 2×3 的矩阵。

以下是一个创建三维数组的示例：

```

1 # 在 R 中定义数组
2 B <- array(
3   c(1, 2, 3, 4, 5, 6, 7, 8),
4   dim = c(2, 2, 2)
5 )
6 print(B)

```

在该示例中，`B` 是一个维度为 $2 \times 2 \times 2$ 的三维数组。我们传入了包含 8 个元素的向量，R 会按照给定的维度将其填充到三个维度的空间中。数组在数值模拟、图像处理以及高维统计分析等方面具有广泛的应用价值。

1.5.6 因子

因子用于将数据分为多个类别或级别，非常适合存储分类数据，例如性别、学历、地区等常见的字符型或整数型变量。它们在统计建模和数据分析中尤为重要，尤其是在使用分类变量作为解释变量时。

```

1 # 在 R 中定义因子
2 fctr <- factor(
3   c("男", "女", "男", "男", "女", "男", "女")
4 )
5 print(fctr)

```

您是否注意到，我们在代码中交替使用了 `<-` 和 `=?` 在 R 语言中，`<-` 是赋值操作符，用于将右边的值赋予左边的变量。箭头方向指出了值的流向。虽然 R 也支持使用 `=` 进行赋值，但 `<-` 更为传统，特别是在函数调用中用于明确区分参数传递与变量赋值，因此许多 R 用户更倾向于使用它。

1.6 数据对象的导入、预处理与存取

1.6.1 导入数据对象

在实际应用中，我们通常不会直接在 R 控制台中输入数据，尤其是在处理大型数据集时，一般需要从外部文件导入数据。本节将以“1952 年以来中国的国内生产总值”为例，介绍如何在 R 中导入数据并进行绘图。该数据以 CSV 格式存储。

R 语言支持灵活的数据导入方式。例如，用户可以使用 `read.csv()` 函数导入以逗号分隔的文本文件（即 CSV 文件）。若需导入 Excel 文件（无论是 `.xls` 还是 `.xlsx`）格式，可以使用 `readxl` 包。对于从其他统计软件（如 SPSS）迁移到 R 的用户，R 还支持通过 `haven` 包导入这些专有格式的数据文件。

什么是 R 软件包？

R 软件包 (Package) 是一个集合，包含函数、数据集、帮助文档以及相关的元信息，用于在 R 环境中执行特定的功能或任务。软件包能够扩展 R 的基础功能，使用户可以进行更复杂、专业化的操作。例如，一些软件包专用于数据可视化（如 `ggplot2`），而另一些则专注于数据处理（如 `dplyr`）。大多数软件包托管在 CRAN (Comprehensive R Archive Network) 上，用户可以从该平台下载安装。安装完成后，需使用 `library()` 函数将包加载到当前的 R 会话中，方可调用其功能。

以下是在 R 中导入 `China_GDP.csv` 数据集的方式：

```
1 folder <- "Z:/金融计量经济学/R代码" # 请替换为您自己的工作目录。
2 setwd(folder)
3 China_GDP <- read.csv("China_GDP.csv")
4 head(China_GDP)
```

在 R 中，函数 `head()` 用于显示数据对象的前几行（默认是前六行），常用于查看数据框、向量等对象的开头部分。当您处理大型数据集时，`head()` 可以帮助您快速了解数据的结构与示例值。

`head(China_GDP)` 命令的输出如下：

```
1 head(China_GDP)
2 1 1952 679.1
3 2 1953 824.4
4 3 1954 859.8
5 4 1955 911.6
6 5 1956 1030.7
7 6 1957 1071.4
```

此外，您还可以使用 `str(China_GDP)` 命令检查 `China_GDP` 数据集的结构，包括变量的名称、类型与数据样本。

并非所有数据都需要通过导入获得。R 的基础包 `datasets` 自带了许多内置数据集，您可以直接调用它们用于学习与演示。例如，常用的数据集有 `mtcars`、`iris` 和 `airquality` 等。

什么是 R 基础包？

在 R 中，“基础包” (base packages) 有其特定含义。它们是在安装 R 时自动包含并加载的核心库，提供了 R 语言的基本功能和运行环境。

这些基础包主要包括：

1. `base`: R 的核心功能，包含基本语法、变量赋值、流程控制、函数定义等。
2. `datasets`: 内置标准数据集，如 `mtcars`、`iris` 等，常用于教学和示例。
3. `graphics`: 基础绘图系统，用于创建散点图、直方图等基本图形。
4. `grDevices`: 处理图形设备相关功能，包括颜色设置与图像输出。
5. `grid`: 支持构建复杂图形布局的网格图形系统。
6. `methods`: 提供对 S4 面向对象编程系统的支持。
7. `stats`: 各种统计分析工具，包括回归、假设检验、时间序列等模型。

8. **tools**: 用于包管理、代码分析和安装支持等功能。
9. **utils**: 通用工具函数，如数据导入导出、进度条、文件操作等。
10. **parallel**: 支持并行计算，适用于多核处理器上的任务分配。
11. **compiler**: 字节编译器，用于加速 R 代码的执行。
12. **splines**: 用于生成和平滑样条函数 (splines)。
13. **tcltk**: 为 GUI 编程提供与 Tcl/Tk 的接口支持。
14. **stats4**: 用于支持 S4 类系统下的统计建模。

因此，用户无需单独安装这些包。启动 R 时，其中的一些包（如 **base** 和 **datasets**）会自动加载，相关函数也即可使用。

补充说明：S4 是 R 中的一种面向对象系统，相较于 S3 更加正式和严格。S4 对象由“槽” (slots) 构成，表示属性，访问方式为 @ 运算符或 **slot()** 函数。此外，S4 支持多重方法分派和类型验证，使对象操作更具规范性和可靠性。

要查看 R 中的内置数据集列表，可以使用函数 **data()**：

```
1 data()
```

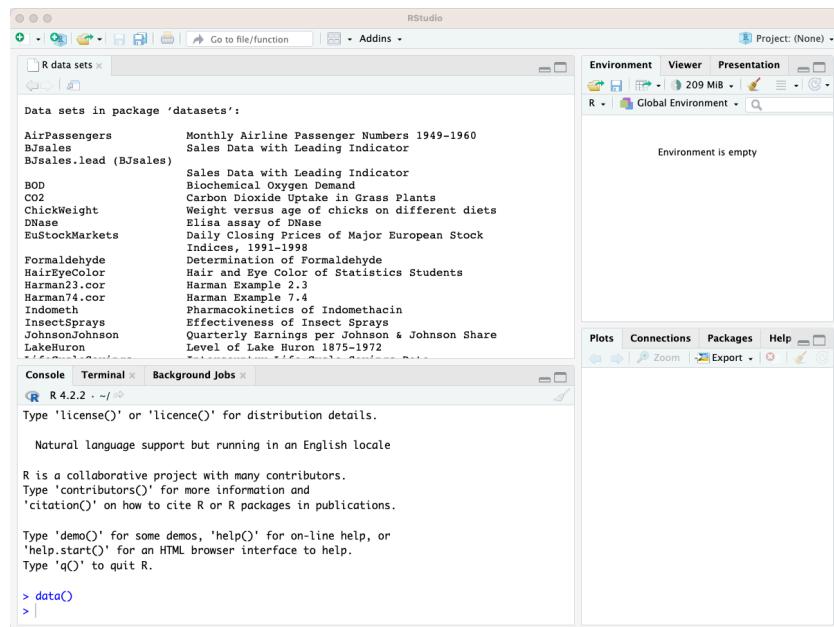


图 1.3: 预加载数据列表

如果我们希望查看某个具体数据集，例如 **mdeaths**，只需输入以下命令即可查看其前几行内容：

```
1 head(mdeaths)
```

该命令将返回该数据集的前六行，便于快速了解其结构和内容。这个方法对于初学者熟悉数据集及其格式非常有帮助。

1.6.2 数据预处理

在 R 中，数据预处理是整个数据分析流程中的关键步骤。该过程涵盖多种技术与方法，旨在对原始数据进行清理、标准化和优化处理，从而为后续分析打下良好基础。最常见的预处理任务之一是数据清洗。例如，可以使用 `na.omit()` 或 `complete.cases()` 函数删除包含缺失值的行；也可以借助如 `imputeTS`、`mice` 或 `Hmisc` 包中的 `impute()` 函数对缺失值进行填补。另外，通过 `duplicated()` 和 `unique()` 函数可以去除重复值。

R 提供了丰富的函数与包用于数据预处理。其中最基本的操作包括子集提取、筛选和汇总。这些步骤通常作为数据分析流程的起点，确保数据结构符合分析要求，从而提高整体建模与推理的效率与准确性。

1.6.2.1 子集提取

在 R 中，子集提取是指从数据对象中提取其一部分内容。对于矩阵或数据框，可以按行和列进行提取；对于向量，则是按元素进行提取。

以下以中国 GDP 数据为例，展示如何从中提取特定子集：

```
1 # 示例：选择第 2 到第 4 个元素
2 subset_China_GDP <- China_GDP$x[2:4]
```

1.6.2.2 过滤

过滤是基于特定条件筛选数据行的过程。此操作在将数据集细分为更相关或更集中的子集时特别有效。

```
1 # 示例：过滤出大于 2000 的元素
2 filtered_China_GDP <- China_GDP$x[China_GDP$x > 2000]
```

1.6.2.3 汇总

R 中的汇总操作是指将大量数据归纳为简明的度量值。常见的汇总操作包括按某一列分组，对另一列进行平均、求和或计数等计算。本节以 R 基础包中的 `mtcars` 数据集为例进行说明。该数据集收录了多种汽车型号及其性能参数，最初发表于 1974 年的美国《汽车趋势》杂志，包含 32 种汽车（1973–74 年款）的油耗数据及其他 10 项设计与性能指标。

- 使用 `aggregate` 函数：这是 R 基础包中的函数，允许通过公式方式进行数据汇总。
- 使用 `dplyr` 包：`dplyr` 提供了更加灵活、可读性更强的聚合语法，例如使用 `group_by()` 和 `summarize()` 等函数。

注意：`dplyr` 并非 R 的基础包，首次使用需先安装。请在 R 控制台运行：

```
1 install.packages("dplyr")
```

安装完成后，使用下述命令加载 `dplyr` 包：

```
1 library(dplyr)
```

接着，我们进行一次简单的汇总计算：

```
1 # 汇总计算
2 avg_mpg_by_cyl <- mtcars %>%
3   group_by(cyl) %>%
```

```

4 summarise(avg_mpg = mean(mpg), .groups = "drop")
5
6 # 显示结果
7 print(avg_mpg_by_cyl)

```

结果将显示四缸、六缸和八缸汽车的平均 MPG 值，反映不同发动机配置下的燃油效率表现。汇总操作有助于从大型数据集中提取概览信息，从而简化分析过程并支持更有效的决策。

总之，掌握这些基础数据操作对 R 用户至关重要，它们为后续构建更复杂的数据处理与分析流程奠定了坚实基础。

1.6.2.4 插补缺失值

许多计量经济学模型（如自回归移动平均模型 ARMA、向量自回归模型 VAR 等）通常假设数据完整且连续。缺失值的存在会干扰模型对数据模式与规律的识别，因此需要在建模前进行恰当的缺失值插补（imputation）。

均值插补是指用变量的平均值填补缺失值；中位数插补则使用变量的中位数填补。这类方法简单直观，适用于分布近似对称的情形。

```

1 # 创建包含缺失值的示例数据集
2 data_set <- data.frame(
3   var1 = c(1, 2, NA, 4, 5),
4   var2 = c(6, NA, 8, 9, 10)
5 )
6
7 # 仅对数值型变量执行：均值插补
8 for (col in names(data_set)) {
9   if (is.numeric(data_set[[col]])) {
10     m <- mean(data_set[[col]], na.rm = TRUE)
11     data_set[[col]][is.na(data_set[[col]])] <- m
12   }
13 }
14 cat("均值插补后的数据: \n")
15 print(data_set)
16
17 # 重新创建示例数据集
18 data_set <- data.frame(
19   var1 = c(1, 2, NA, 4, 5),
20   var2 = c(6, NA, 8, 9, 10)
21 )
22
23 # 仅对数值型变量执行：中位数插补
24 for (col in names(data_set)) {
25   if (is.numeric(data_set[[col]])) {
26     med <- median(data_set[[col]], na.rm = TRUE)
27     data_set[[col]][is.na(data_set[[col]])] <- med
28   }
29 }
30 cat("中位数插补后的数据: \n")
31 print(data_set)

```

最近邻插补 (k-Nearest Neighbors, kNN) 通过寻找与含缺失观测最相似的样本，并用其对应取值来填补缺失。可使用 VIM 包中的 `kNN()` 实现。

```

1 library(VIM)
2
3 # 创建包含缺失值的示例数据集
4 data_set <- data.frame(
5   var1 = c(1, 2, NA, 4, 5),
6   var2 = c(6, NA, 8, 9, 10)
7 )
8
9 # 最近邻插补 (k 表示最近邻数量)
10 imputed_data <- kNN(data_set, k = 1)
11
12 cat("最近邻插补后的数据 (kNN 可能会附加标记列 .imp/.na) : \n")
13 print(imputed_data)

```

此外，还可以使用 `mice` 包进行多重插补；使用 `stats` 包中的 `lm` 函数进行线性回归插补；或结合 `dlm` 包通过卡尔曼滤波进行时间序列场景下的插补。

1.6.3 导出数据对象

R 不仅可以灵活地导入多种数据格式，在数据导出方面也同样非常强大。例如，要将数据框导出为逗号分隔值 (CSV) 文件，用户可以使用 `write.csv()` 函数。语法如下：`write.csv(mydata, "mydata.csv")`。其中，`mydata` 是要导出的数据框，`"mydata.csv"` 是目标文件名。同理，R 也可以导出为 Excel、SPSS、SAS 或 Stata 等格式的数据文件。

```

1 # 导出 Excel 文件
2 library(writexl)
3 write_xlsx(mydata, "mydata.xlsx")
4
5 # 导入/导出 SPSS (.sav) 文件
6 library(haven)
7 write_sav(mydata, "mydata.sav")
8 # read_sav("mydata.sav") # 读取示例

```

此外，还可以使用 `.RData` 或 `.rda` 文件格式来存储 R 中的对象。

```

1 # 创建一些数据
2 x <- 1:10
3 y <- rnorm(10)
4 z <- data.frame(x, y)
5
6 # 保存对象到 .RData 文件
7 save(x, y, z, file = "mydata.RData")

```

如果希望保存当前工作环境中的所有对象，可以使用：

```
1 save.image("mydata.RData")
```

1.7 R 语言中的数据可视化

在 R 或其他平台中，数据可视化在数据分析中至关重要。恰当的图形能够迅速揭示原始数据中难以直接观察到的模式与异常，从而增强对数据的直观理解。除帮助研究者理解

数据之外，可视化还能更清晰地向不同受众传达复杂的数据信息。这种清晰性有助于简化决策流程，使利益相关方识别出对战略行动至关重要的趋势与关联。总而言之，R 的可视化功能能够将原始数据转化为可指导行动的见解。

1.7.1 基本折线图

我们可以使用以下命令绘制我国自 1952 年以来的 GDP 数据：

```
1 # 使用蓝色折线图可视化 GDP
2 plot(China_GDP$GDP, type = "l", col = "blue")
```

请注意，在 R 中可通过数据框的 \$ 运算符按列名访问特定变量，因此可以将数据框中的列当作向量来引用。自 1952 年以来我国的国内生产总值（GDP）变化趋势如图 1.4 所示。不难看出，我国实现了显著的经济增长，尤其是改革开放之后，经济发展与繁荣达到了前所未有的高度。

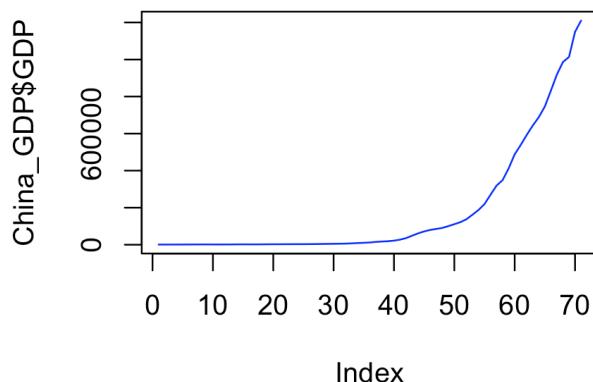


图 1.4: 1952 年以来中国的国内生产总值（GDP）

1.7.2 改进折线图

仔细观察图 1.4 可发现，其清晰度仍有不足。例如，横坐标未显示具体年份，纵坐标也缺少明确标签。

可以先将 GDP 序列转换为时间序列对象，再以更规范的轴与标题进行绘制。我们可以使用基础包 stats 中的 ts() 函数实现：

```
1 GDP.ts <- ts(China_GDP$GDP, start = c(1952), frequency = 1)
```

上述命令 `GDP.ts <- ts(China_GDP$GDP, start = c(1952), frequency = 1)` 完成以下操作：

- `ts()`: R 中用于创建时间序列对象的函数。
- `China_GDP$GDP`: 提取 `China_GDP` 数据集中的 GDP 列。
- `start = c(1952)`: 指定时间序列的起始年份为 1952。

- `frequency = 1`: 表示每年的观测频率为 1, 即该数据是年度数据。
- `GDP.ts <-:` 将 `ts()` 函数的结果赋值给新变量 `GDP.ts`。

随后, 我们可以使用 `plot` 函数对 `GDP.ts` 进行绘图, 效果见图 1.5。

```
1 plot(GDP.ts, type = "l", col = "blue", xlab = "Year", ylab = "GDP",
2      main = "China's GDP",
3      lwd = 2, ylim = c(min(China_GDP$GDP), max(China_GDP$GDP)))
4 grid(col = "gray85", lty = "solid")
```

上述代码在原始折线图的基础上进行了美化处理:

- 设置了 `type = "l"` 表示绘制线图;
- `xlab` 和 `ylab` 分别为横轴与纵轴添加了标签;
- `main` 参数为图形添加了主标题;
- `lwd = 2` 增加了线条粗细;
- `ylim` 明确了纵轴范围, 确保数据完整显示;
- `grid()` 函数增加了背景网格, 使图像更具可读性。

最终结果如图 1.5 所示:

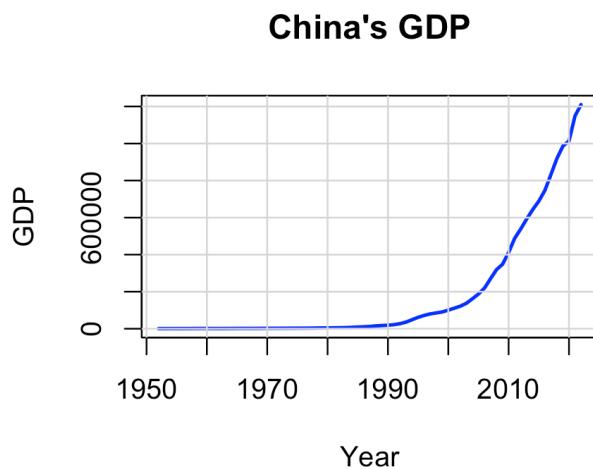


图 1.5: 1952 年以来中国的国内生产总值 (GDP, 单位: 亿元)

1.7.3 R 中的其他图形

除了折线图外, R 还通过基础图形系统及多个扩展包提供了丰富而强大的绘图功能。下面列出基础图形系统中常见的一些图表类型及其对应函数:

- **散点图**: 根据两个变量的数值关系绘制数据点。使用 `plot()`。
- **直方图**: 将数据分箱并统计每个箱的观测数量, 展示单变量分布。使用 `hist()`。

- **箱线图**: 显示五数概括 (最小值、第一四分位数、中位数、第三四分位数、最大值)，适合比较组间分布差异。使用 `boxplot()`。
- **条形图**: 用条形长度表示数值 (通常为频数或比例)。使用 `barplot()`。
- **饼图**: 用扇形切片展示比例或百分比。使用 `pie()`。
- **密度图**: 估计并绘制变量的概率密度函数，可用 `plot(density(x))`；在后文将演示如何在直方图上叠加密度曲线。
- **成对图 (散点图矩阵)**: 展示多个变量的两两关系。使用 `pairs()`。
- **QQ 图 (分位数-分位数图)**: 通过比较分位数检验样本分布与正态分布 (或两分布) 是否相似。使用 `qqnorm()` 与 `qqline()`。
- **轮廓图**: 在二维平面用等高线表示三维数据，常用于展示两个自变量对因变量的影响。使用 `contour()`。
- **图像图 (热度图)**: 与轮廓图相似，但用颜色填充等高线之间的区域。使用 `image()`。
- **三维散点/曲面图**: 基础系统不直接支持，可借助 `rgl`、`lattice` 等扩展包实现。

`mtcars` 数据集包含多种汽车模型的性能属性，适合在 R 中演示各类图形与可视化分析。下面给出若干常见示例。

直方图:

```
1 hist (mtcars$mpg, main="Histogram of Miles-per-Gallon",
2       xlab="mpg", col="lightblue", border="black")
```

散点图:

```
1 plot (mtcars$wt, mtcars$mpg, main="Weight vs. MPG",
2       xlab="Weight", ylab="MPG", pch=19, col="blue")
```

箱线图:

```
1 boxplot (mpg ~ am, data=mtcars, main="MPG by Transmission Type",
2           xlab="Transmission (0=Automatic, 1=Manual)", ylab="MPG",
3           col=c("lightblue", "lightgreen"))
```

成对图:

```
1 pairs (~mpg + hp + wt + qsec, data=mtcars, main="Scatterplot Matrix")
```

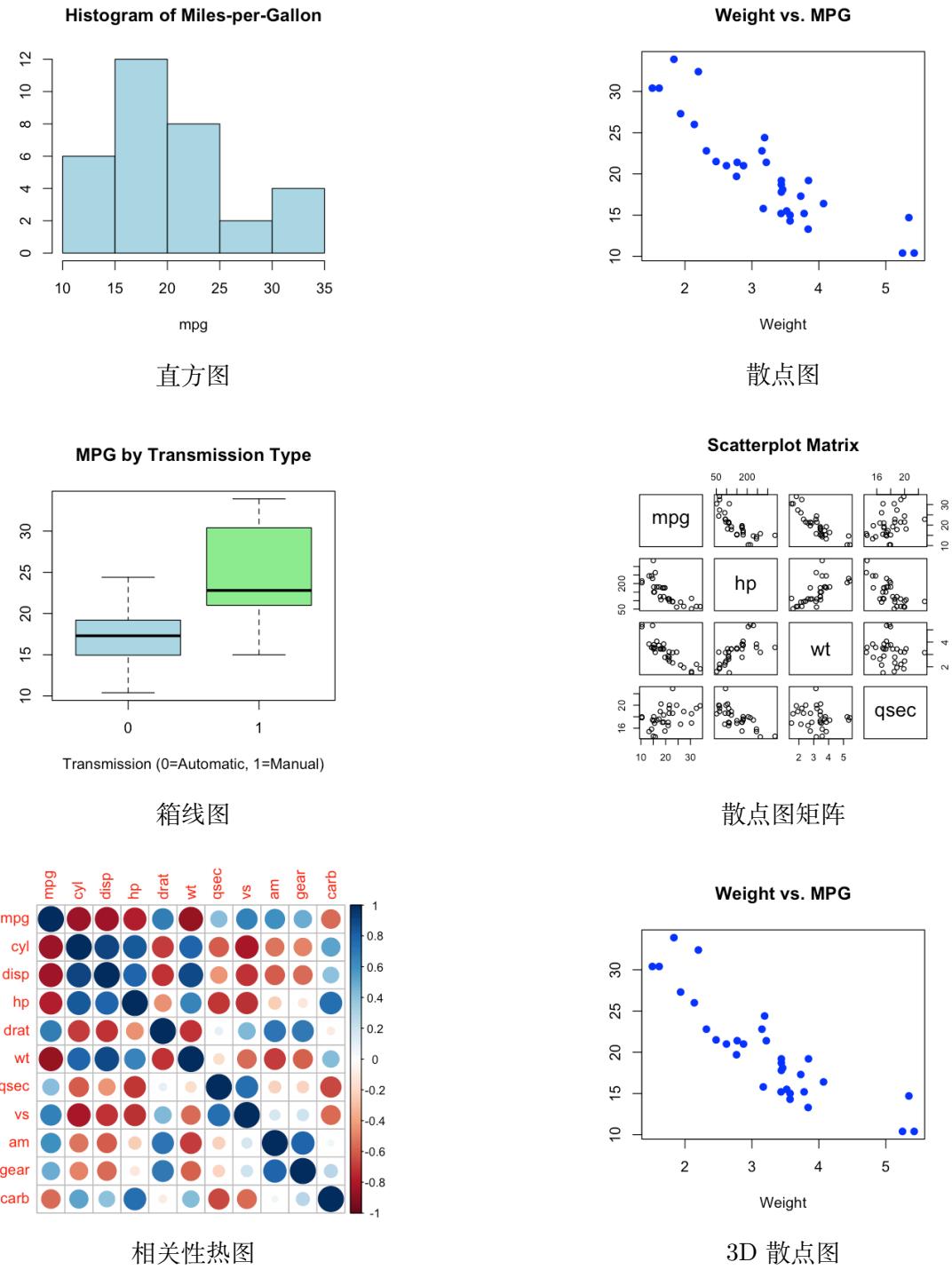
相关性热图:

```
1 library (corrplot)
2 correlations <- cor (mtcars)
3 corrplot (correlations, method="circle")
```

3D 散点图:

```
1 library (scatterplot3d)
2 scatterplot3d (mtcars$hp, mtcars$wt, mtcars$mpg,
3                pch=19, color="blue", main="3D Scatterplot: HP vs. Weight vs. MPG")
```

绘图结果见图 1.6。

图 1.6: 使用 `mtcars` 数据集绘制的各类图形

1.8 使用 R 生成随机数

1.8.1 使用内置函数生成随机数

R 提供了丰富的内置函数，可用于生成来自各类概率分布的随机数。本节以正态分布为例说明基本用法。

与正态分布密切相关的一个核心概念是**中心极限定理** (central limit theorem, CLT)。该定理表明：当一组**独立同分布** (independently and identically distributed, 简称 i.i.d.) 的随机变量样本量足够大时，其**样本均值**的分布将近似服从正态分布——这也是许多统计方法的理论基础。

在 R 中生成正态随机数非常便捷，可直接使用 `rnorm()`：

```
1 # 生成 10 个标准正态随机数
2 rnorm(10, mean = 0, sd = 1)
3 #> [1] -0.8267256 -0.4518683 -1.1265701 -1.7666592 -1.2771818
4 #> [6] -1.1659479  1.3600173 -0.5853499  0.8315215  1.3784917
```

由于 `rnorm()` 的默认参数为 `mean = 0`、`sd = 1`，上式也可简写为：

```
1 rnorm(10)
2 #> [1] 1.574904176 -0.572594806 0.008159591 -0.472049254 0.342730428
3 #> [6] -0.194104579 -0.080820534 0.247510159 1.122421265 2.135382611
```

若需要指定不同的均值与标准差（如均值为 1、标准差为 2）：

```
1 rnorm(5, mean = 1, sd = 2)
2 #> [1] 0.9065939 1.6503606 1.3322302 1.3874926 -0.4869327
```

注意：随机数为伪随机，每次运行结果可能不同。若需结果可复现，请事先设定随机种子，例如 `set.seed(123)`。

除正态分布外，R 还能方便地生成多种其他分布的随机数，常用函数包括（n 表示样本量）：

- 卡方分布：`rchi2(n, df)`，df 为自由度。
- 学生 t 分布：`rt(n, df)`，df 为自由度。
- F 分布：`rf(n, df1, df2)`，df1=df2 分别为分子/分母自由度。
- 几何分布：`rgeom(n, prob)`，prob 为成功概率。
- 二项分布：`rbinom(n, size, prob)`，size 为试验次数，prob 为成功概率。
- 泊松分布：`rpois(n, lambda)`，lambda 为均值（亦为方差）。

1.8.2 采用分位数转换法生成随机数

并不是所有的概率分布在 R 中都提供现成的随机数生成函数。为了生成服从特定分布的随机样本，可以采用**分位数转换法** (quantile transformation method)，亦称**逆累积分布函数法** (inverse CDF method)。其基本思想如下：

设随机变量 $U \sim \text{Unif}[0, 1]$ 。对任意具有连续累积分布函数 $F_X(x)$ 的随机变量 X ，记其分位数函数（反函数）为 $F_X^{-1}(u)$ 。则

$$X = F_X^{-1}(U)$$

的分布即为 F_X 。

该方法的操作步骤如下：

1. 在区间 $[0, 1]$ 内生成一个（或一组）服从均匀分布的随机数 u ；
2. 计算 $x = F_X^{-1}(u)$, 其中 F_X^{-1} 为目标分布的分位数函数；
3. 得到的 x 即为服从分布 F_X 的观测值（一次实现）。

举例而言，若在 R 中没有 `rnorm`, 但可使用 `runif`。下面的代码展示如何用分位数转换法生成标准正态样本（选用标准正态便于检验其统计特性）：

```

1 # 逆变换采样：用分位数法从 Unif[0,1] 生成标准正态样本
2
3 # (可选) 清空环境：注意这会删除当前会话中的所有对象
4 # rm(list = ls())
5
6 # 固定随机种子，保证结果可复现
7 set.seed(123456789)
8
9 # 1) 定义标准正态分布的累积分布函数 Φ(z)
10 #   这里故意用数值积分实现，以展示“自己写 CDF → 再做逆变换”的流程；
11 #   在实务中当然可直接使用内置 pnorm()。
12 compute_phi <- function(z) {
13   integrand <- function(t) (1 / sqrt(2 * pi)) * exp(-t^2 / 2)
14   res <- integrate(integrand, lower = -Inf, upper = z)
15   res$value
16 }
17
18 # 简单校验：与 pnorm 的结果应接近
19 z_val <- 1.96
20 print(compute_phi(z_val))
21 print(pnorm(z_val))
22
23 # 2) 定义 Φ^{-1}(u)：对给定的概率值 phi_value, 求解 Φ(z) - phi_value = 0
24 inverse_phi <- function(phi_value) {
25   f <- function(z) compute_phi(z) - phi_value
26   # 正态分布 99.9999% 的概率落在 [-6, 6] 内；取更宽区间更稳妥
27   root <- uniroot(f, interval = c(-8, 8), tol = 1e-8)
28   root$root
29 }
30
31 # 3) 生成样本：先从 Unif[0,1] 取 1000 个值，再做逆变换得到 N(0,1) 样本
32 u <- runif(1000)
33 x <- as.numeric(lapply(u, inverse_phi))
34
35 # 4) 作图：直方图 + 核密度曲线
36 #   R 的 base graphics 中，绘制“密度直方图”有两种写法：
37 #   - hist(..., probability = TRUE)    # 经典参数，仍然可用
38 #   - hist(..., freq = FALSE)          # 等价写法
39 hist(
40   x, probability = TRUE,
41   main = "Histogram with Density Curve (Inverse CDF Sampling)",
```

```

42   xlab = "Value", col = "lightblue", border = "black"
43 )
44
45 # 叠加核密度曲线
46 lines(density(x), col = "red", lwd = 2)

```

以下是上述代码的详细解释：

1. 初始化

- `rm(list = ls())`: 清空当前 R 环境中的对象（可选，谨慎使用）。
- `set.seed(123456789)`: 设置随机数种子，保证结果可复现。

2. 定义标准正态分布的累积分布函数 (CDF)

- 函数 `compute_phi(z)` 用于计算给定数值 z 下标准正态分布的累积分布函数。
- 数学形式为

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-t^2/2} dt.$$

- 在 R 中通过 `integrate()` 对被积函数从 $-\infty$ 到 z 进行数值积分。

3. 函数校验

- 取 $z = 1.96$ ，比较 `compute_phi(1.96)` 与 R 内置 `pnorm(1.96)` 的结果，二者应高度一致，以验证实现的 CDF 正确性。

4. 逆变换采样：第一步——生成均匀样本

- 使用 `runif(1000)` 在区间 $[0, 1]$ 内生成 1000 个独立均匀分布随机数，记为 `sample.unif`。

5. 定义标准正态分布的逆 CDF 函数

- `inverse_phi(phi_value)`: 对给定概率值 ϕ_value 求解 $\Phi(z) - \phi_value = 0$ 的根。
- 内部函数 `function_to_zero(z)` 返回 `compute_phi(z) - phi_value` 的差值。
- 采用 `uniroot()` 在区间 $[-10, 10]$ 上求根，得到使差值为零的 z 值。

6. 逆变换采样：第二步——生成标准正态样本

- 对 `sample.unif` 中的每个 u ，计算 $z = \text{inverse_phi}(u)$ ，得到一组服从 $N(0, 1)$ 的样本 `sample.norm`。

7. 绘图：直方图与核密度估计曲线

- 用 `hist(sample.norm, probability = TRUE)` 绘制密度直方图（或使用等价写法 `freq = FALSE`）。
- 用 `lines(density(sample.norm))` 叠加核密度估计曲线，以更直观地展示样本分布形态。

以下是绘制直方图及其对应经验密度曲线的方法。

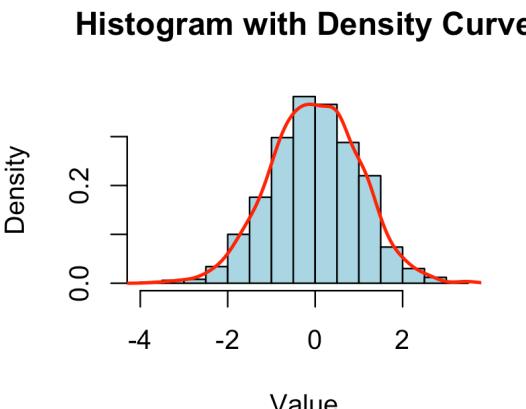


图 1.7: 带有密度曲线的直方图

经验密度曲线是对概率密度函数的一种非参数估计方法。我们将在后续介绍非参数方法时对其进行详细讲解。此处，我们只需关注：通过经验密度曲线绘图，可以看出基于分位数变换法模拟的数据，其分布与标准正态分布较为吻合。

此外，也可以将直方图视为一种使用 ** 均匀核函数 ** 的非参数估计方法。直方图通过将数据划分为若干区间（或称“箱”），统计每个区间内数据点的数量或比例，从而对分布的形状进行估计。

如果图 1.7 仍未能让您信服，我们还可以考察模拟数据的统计特性，例如其均值和方差。

```

1 # 检查生成数据的统计特性
2 mean(sample.norm)
3 [1] -0.007377702
4
5 var(sample.norm)
6 [1] 1.0353

```

可以看到，结果非常接近标准正态分布的理论期望 (0) 和方差 (1)! 由于该算法本身具有随机性，且样本容量有限（并非趋于无穷大），因此模拟结果总是会与理论值存在轻微偏差。

1.9 使用“帮助”选项卡

如果您对某个命令（例如 `rnorm`）的用法不确定，可以将该命令输入至 RStudio 的“帮助”选项卡中。该选项卡会连接至庞大的 R 帮助文档系统，如图 ?? 所示。

此外，您也可以直接在控制台中输入 `help(rnorm)` 或 `?rnorm`，以快速获取相关帮助页面。这些操作将引导您查看以下网页内容：<https://stat.ethz.ch/R-manual/R-devel/library/stats/html/Normal.html> 该网页详细介绍了 `rnorm` 函数的使用方法和参数说明。

R 中的“Help”选项卡

Description

Density, distribution function, quantile function and random generation for the normal distribution with `mean` equal to mean and standard deviation equal to `sd`.

Usage

```
dnorm (x, mean = 0, sd = 1, log = FALSE)
pnorm (q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
qnorm (p, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
rnorm (n, mean = 0, sd = 1)
```

Arguments

[leftmargin=0pt, itemsep=0pt, parsep=0pt]vector of quantiles. vector of probabilities. number of observations. If `length (n) > 1`, the length is taken to be the number required. vector of means. vector of standard deviations. logical; if TRUE, probabilities p are given as $\log(p)$. logical; if TRUE (default), probabilities are $P[X \leq x]$ otherwise, $P[X > x]$.

Details

If `mean` or `sd` are not specified they assume the default values of 0 and 1, respectively. The normal distribution has density:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-\mu)^2/2\sigma^2}$$

where μ is the mean of the distribution and σ the standard deviation.

Value

`dnorm` gives the density, `pnorm` gives the distribution function, `qnorm` gives the quantile function, and `rnorm` generates random deviates. ...

R 的帮助页面对正态分布相关函数提供了详尽说明，涵盖 `dnorm`、`pnorm`、`qnorm` 与 `rnorm` 的用途、参数与常见选项。简要概述如下：

~~log~~~~lower.tail~~ `dnorm`: 计算给定取值处的正态分布概率密度函数值。

- `pnorm`: 计算累积分布函数值，即随机变量不大于给定数的概率。

- `qnorm`: 计算分位数函数值，即给定概率所对应的分位数。
- `rnorm`: 生成正态分布随机数，可指定均值与标准差。

1.10 tidyverse 系列软件包

`tidyverse` 是一组 R 软件包的集合，旨在为数据科学和数据处理任务提供一致且易于使用的功能。`tidyverse` 类软件包具有诸多优点：它们在设计上风格统一，使用类似的数据结构与语法。这不仅降低了学习成本，还使代码更加清晰、易于理解。其中，`dplyr` 提供了强大的数据操作功能，能够轻松应对大规模数据集的处理任务。其简洁的语法使得数据转换步骤更容易编写与维护。此外，`tidyverse` 各个软件包之间能够良好协作，从数据导入（如 `readr`）、数据清洗（如 `tidyr`）到数据可视化（如 `ggplot2`），构成了一个高效、流畅的数据分析工作流。

`tidyverse` 软件包在学习金融计量经济学的过程中也具有重要作用。它不仅可以简化复杂的数据操作，提高处理效率，还能提升分析结果的质量。例如，金融数据中常常包含缺失值、异常值或格式不统一的日期信息。使用 `tidyverse` 中的 `dplyr` 与 `tidyr` 等包，可以轻松完成数据清洗与预处理，构建清洁的数据基础，为后续分析提供保障。金融计量经济学大量依赖时间序列数据，如股票价格、利率等。尽管 `tidyverse` 本身并不专注于时间序列分析，但它可以很好地与其他专门处理时间序列数据的 R 包（如 `xts`、`zoo` 和 `lubridate`）集成使用。`tidyverse` 中的 `ggplot2` 是一个功能强大的数据可视化工具，尤其适用于绘制时间序列图、散点图和直方图，是金融数据分析中的得力助手。

要安装整个 `tidyverse` 包，可以在 R 控制台中输入以下命令：

```
1 install.packages ("tidyverse")
```

1.10.1 dplyr 软件包

R 语言中的 `dplyr` 包以简洁性和可读性为设计核心。它提供了一组常被称为“动词”的核心函数，专注于数据操作中最常见的任务。这种功能聚焦的设计风格避免了因选项繁杂而产生的学习障碍，从而大大降低了上手门槛。利用这些动词，用户可以高效应对各种数据分析挑战。

根据数据的维度，这些动词可大致划分为“行操作”“列操作”“行组操作”三类。

行操作：

- `filter()`: 根据列的取值筛选满足条件的行；
- `slice()`: 通过位置选取特定行；
- `arrange()`: 根据列的取值对数据行进行排列。

列操作：

- `select()`: 选择或排除指定列；
- `rename()`: 修改列名称；
- `mutate()`: 创建或更新列；
- `relocate()`: 调整列的显示顺序。

行组操作 (Groups of Rows)：

- `group_by()`: 指定一个或多个列作为分组变量;
- `summarise()`: 对每组数据进行汇总, 生成摘要统计值。

上述动词清晰地映射到数据分析师在实际工作中所需的常见操作。其命名直观、功能聚焦, 使 `dplyr` 成为数据科学工作流程中的强大工具。

在本节中, 我们将使用 `starwars` 数据集来演示如何通过 `dplyr` 包对数据进行处理与分析。`starwars` 是一个内置于 `dplyr` 和 `tidyverse` 中的数据集, 来源于星球大战 API⁸, 为学习数据整理与转换提供了有趣且信息丰富的案例基础。

通过 `dplyr` 提供的动词函数 (如 `filter()`、`select()`、`mutate()` 等), 我们可以便捷地对 `starwars` 数据集执行过滤、排列、选择变量、计算新列等常见的数据操作任务。

```

1 dim (starwars)
2 #> [1] 87 14
3 starwars
4 # A tibble: 87 x 14
5 #   name height mass hair_color skin_color eye_color birth_year sex gender
6 #   homeworld species films vehicles starships
7 # 1 Luke S... 172 77 blond fair blue 19 male masc... Tatooine Human <chr> <
8 #   <chr> <chr>
9 # 2 C-3PO 167 75 NA gold yellow 112 none masc... Tatooine Droid <chr> <chr>
10 #   <chr>
11 # 3 R2-D2 96 32 NA white, bl... red 33 none masc... Naboo Droid <chr> <chr>
12 #   <chr>
13 # 4 Darth V... 202 136 none white yellow 41.9 male masc... Tatooine Human <
14 #   <chr> <chr> <chr>
15 # 5 Leia O... 150 49 brown light brown 19 fema... femin... Alderaan Human <
16 #   <chr> <chr> <chr>
17 # 6 Owen L... 178 120 brown, gr... light blue 52 male masc... Tatooine
18 #   Human <chr> <chr> <chr>
# 7 Beru W... 165 75 brown light blue 47 fema... femin... Tatooine Human <
#   <chr> <chr> <chr>
# 8 R5-D4 97 32 NA white, red red NA none masc... Tatooine Droid <chr> <chr>
#   > <chr>
# 9 Biggs ... 183 84 black light brown 24 male masc... Tatooine Human <chr>
#   <chr> <chr>
# 10 Obi-W... 182 77 auburn, w... fair blue-gray 57 male masc... Stewjon
#   Human <chr> <chr> <chr>
#   # 77 more rows
#   # 1 more variable: starships <list>
#   # Use `print (n = ...)` to see more rows

```

注: `tibble` 是对 R 传统数据框 (data frame) 的现代化实现。可使用 `as_tibble()` 将数据框转换为 `tibble`。作为 `tidyverse` 生态的一部分, `tibble` 的设计目标是提升数据操作的使用体验。它与传统数据框基本兼容, 但在打印展示、类型保持和子集化等方面更为友好, 更适合现代数据分析工作。更多信息请参见: <https://tibble.tidyverse.org>。

管道操作符 (pipe operator `%>%`), 最初由 `magrittr` 包引入, 并在 `tidyverse` 系列软件包中得到广泛推广。`%>%` 彻底改变了 R 语言中数据操作的书写方式, 它允许一个函数的

⁸星球大战 API (SWAPI) 是一个开放且免费访问的网络服务, 提供关于星球大战宇宙的结构化数据, 涵盖角色、行星、星际飞船、交通工具、物种等信息。该数据集包含 87 个角色的基本资料, 如姓名 (name)、身高 (height)、体重 (mass) 等属性。

输出直接“传递”给下一个函数。

这种命令链式结构不仅简化了代码，使其更具可读性，而且避免了创建和管理临时变量，提高了计算效率。

使用管道操作符的好处如下：

- **可读性**：代码从左到右阅读，避免层层函数嵌套，更易理解数据操作流程。
- **可维护性**：命令链接减少中间变量的创建，避免这些变量在长脚本中成为错误来源。
- **流程逻辑化**：管道操作符使处理步骤按依赖顺序清晰展开，每一步都建立在前一步结果之上。

下例展示了如何使用 `dplyr` 包中的管道操作符。

```

1 library (dplyr)
2 starwars %>%
3   filter (species == "Human") %>%
4   select (name, height, mass) %>%
5   arrange (desc (height) ) %>%
6   head (5)
```

这段代码对《星球大战》(`starwars`) 数据进行了筛选，提取出皮肤颜色为 "light" 且眼睛颜色为 "brown" 的角色，展示了 R 语言中管道操作符的强大功能与代码的可读性。

```

1 starwars %>%
2   filter(skin_color == "light", eye_color == "brown")
```

`dplyr` 包中的 `filter()` 函数用于根据一个或多个条件筛选数据框中的行。在上述示例中，函数应用了两个筛选条件：

- `skin_color == "light"`：选择 `skin_color` 列中值为 “light”的观测，即皮肤颜色为浅色的角色；
- `eye_color == "brown"`：进一步筛选出眼睛颜色为棕色 (“brown”) 的角色。

这些条件组合起来，仅保留同时满足两个条件的观测值。

接下来，我们来看一个更复杂的示例，结合使用了 `select`、`mutate`、`group_by` 和 `summarize` 等函数，展示如何在 `starwars` 数据集上执行更为全面的数据操作任务。

```

1 # 按身高排序
2 starwars %>% arrange (desc (height) )
3
4 # 选择并变更
5 starwars %>% select (name, height, mass) %>%
6   mutate (height_m = height / 100)
7
8 # 汇总和分组
9 starwars %>%
10  group_by (species, sex) %>%
11  summarize (
12    average_height = mean (height, na.rm = TRUE) ,
13    average_mass = mean (mass, na.rm = TRUE)
14  )
```

输出的结果如下：

```

1 # 筛选肤色为浅色且眼睛为棕色的角色
2 starwars %>% filter (skin_color == "light", eye_color == "brown")
3 # A tibble: 7 x 14
4   name height mass hair_color skin_color eye_color birth_year sex gender
5   homeworld species films vehicles
6   <chr> <int> <dbl> <chr> <chr> <dbl> <chr> <chr> <chr> <chr> <lis> <
7   list>
8   1 Leia Or… 150 49 brown light brown 19 fema… femin… Alderaan Human <chr>
9   <chr>
10 ...
11 ...
12 ...
13 ...
14 ...
15 ...
16 ...
17 ...
18 ...
19 ...
20 ...
21 ...
22 ...
23 ...
24 ...
25 ...
26 ...
27 ...
28 ...
29 ...
30 ...
31 ...
32 ...
33 ...
34 ...
35 ...
36 ...
37 ...
38 ...

```

不难看出，R 语言中的 dplyr 包极大地简化了复杂的数据操作流程。通过掌握 dplyr，用户可以高效地完成数据的筛选、排序、变量选择、变换与汇总等任务。

dplyr 的设计理念使用户能够专注于数据分析本身，而无需被繁琐的编程细节所困扰，从而提升分析效率与代码可读性。

1.10.2 tidyverse 包

tidyverse 包主要用于将数据整理为“整洁 (tidy)”格式，从而便于进一步分析和可视化。这里所谓的“整洁数据”，指每个变量对应一列、每个观测对应一行、每类观测单位构成一个独立的表格。

为什么我们应该学习 tidyverse 包？

- 与 tidyverse 的集成：** tidyverse 是 tidyverse 生态系统的一部分，tidyverse 是一组为数据分析量身定制的 R 包集合。tidyverse 与 dplyr 无缝集成，能够高效协同执行数据过滤、汇总、变换等操作任务。
- 促进数据分析：** 整洁的数据结构大大简化了数据分析流程，有助于后续执行统计建模、使用 ggplot2 进行可视化，以及构建预测模型。
- 可重复性与清晰性：** 掌握 tidyverse 能提升数据处理的可重复性，并增强整个分析流程的清晰性，使代码更易于阅读和共享。

1.10.2.1 gather () 函数

首先，输入数据。

```

1 # 加载 tidyverse 包
2 library(tidyverse)
3
4 n = 10
5
6 # 创建数据框
7 tidy_dataframe = data.frame (
8   S.No = c(1:n) ,
9   Group.1 = c(23, 345, 76, 212, 88,
10  199, 72, 35, 90, 265) ,
11  Group.2 = c(117, 89, 66, 334, 90,
12  101, 178, 233, 45, 200) ,
13  Group.3 = c(29, 101, 239, 289, 176,
14  320, 89, 109, 199, 56) )
15
16 # 打印数据框的元素
17 tidy_dataframe

```

结果如下：

```

1 S.No Group.1 Group.2 Group.3
2 1 1 23 117 29
3 2 2 345 89 101
4 3 3 76 66 239
5 4 4 212 334 289
6 5 5 88 90 176
7 6 6 199 101 320
8 7 7 72 178 89
9 8 8 35 233 109
10 9 9 90 45 199
11 10 10 265 200 56

```

`gather()` 函数将多列压缩为“键—值”对 (key-value pairs)，把“宽”数据转换为“长”数据。

“键值对”是指一对数据，其中“键”是标识符或名称，“值”是与该键相关联的数据。键值对常见于数据结构（如字典、映射）中：每个键唯一，并可通过键快速查找对应的值。

例如，一个典型的字典结构 (dictionary data structure) 可能包含如下键值对：

键：“姓名”，值：“张三”；键：“年龄”，值：25。在这种情况下，“姓名”和“年龄”是键，“张三”和 25 是对应的值。该概念在数据整理、数据库与编程等领域非常常见。

```
1 gather (data, key = "key", value = "value", ..., na.rm = FALSE, convert =
  FALSE, factor_key = FALSE)
```

参数	描述
data	数据框对象。
key, value	新键列与值列的名称，可为字符串或符号。
...	待选择的列；若留空，默认选择所有变量，也可显式提供列名。
na.rm	若为 TRUE，从输出中删除 value 列为 NA 的行。
convert	若为 TRUE，尝试将 value 列转换为更合适的数据类型（调用 <code>type.convert()</code> ）。
factor_key	控制 key 列的数据类型；FALSE（默认）存为字符向量，TRUE 存为因子并保留原始顺序。

为了更好地理解，我们将使用 `gather()` 函数将数据转换为长格式。

```
1 # 对 tidy_dataframe 使用 gather () 函数
2 long <- tidy_dataframe %>%
3   gather (Group, Frequency,
4           Group.1:Group.3)
5
6 # 以长格式打印数据框
7 long
```

结果如下：

```
1 > # 以长格式打印数据框
2 > long
3 S.No Group Frequency
4 1 1 Group.1 23
5 2 2 Group.1 345
6 3 3 Group.1 76
7 4 4 Group.1 212
8 5 5 Group.1 88
9 6 6 Group.1 199
10 7 7 Group.1 72
11 8 8 Group.1 35
12 9 9 Group.1 90
13 10 10 Group.1 265
14 11 1 Group.2 117
15 12 2 Group.2 89
16 13 3 Group.2 66
17 14 4 Group.2 334
```

```

18 15 5 Group.2 90
19 16 6 Group.2 101
20 17 7 Group.2 178
21 18 8 Group.2 233
22 19 9 Group.2 45
23 20 10 Group.2 200
24 21 1 Group.3 29
25 22 2 Group.3 101
26 23 3 Group.3 239
27 24 4 Group.3 289
28 25 5 Group.3 176
29 26 6 Group.3 320
30 27 7 Group.3 89
31 28 8 Group.3 109
32 29 9 Group.3 199
33 30 10 Group.3 56

```

长格式：用于描述一类数据结构，每一行通常包含单个变量的一条观测值，亦称“长（long）”或“高（tall）”数据。这种格式在统计分析中特别有用，尤其适合处理时间序列数据。

长格式数据的特点

- **更多的行，较少的列。**在长格式中，通常会有更多的行，因为每一行代表单个变量在某一特定时间点的一条观测值。
- **重复的标识符。**宽格式中可能唯一的标识符（如参与者 ID 或时间点）在长格式中可出现多次，每次对应不同变量或不同时间点。
- **键—值对。**长格式数据常表示为键—值对：一列（键）标识观测的属性，另一列（值）给出相应的观测结果。

长格式的示例

考虑一个宽格式的数据集：

序号	Weight_t1	Weight_t2	Weight_t3
1	70	71	72
2	80	80	81

在这里，每一行表示某人在三个不同时间点的体重测量值。

使用 `tidyverse` 包中的 `gather()` 函数，可以将此数据转换为长格式：

```

1 # 假设 tidy_dataframe 是上面提到的宽格式数据框
2 long <- tidy_dataframe %>%
3   gather (TimePoint, Weight, Weight_t1:Weight_t3)
4

```

转换后的长格式数据如下所示：

序号	时间点	体重
1	Weight_t1	70
1	Weight_t2	71
1	Weight_t3	72
2	Weight_t1	80
2	Weight_t2	80
2	Weight_t3	81

为什么我们要使用“长”格式数据？

- 数据分析的灵活性：**许多统计方法和数据可视化工具更偏好长格式数据，因为它简化了对不同变量的分析。
- 与 R 包的兼容性：**tidyverse 中的工具（如 ggplot2 用于绘图、dplyr 用于数据操作）在处理长格式数据时更直观。
- 汇总更简单：**当每个变量都是单独的列时，跨多个变量汇总（例如计算多个时间点或条件下的平均值）更加直接。

1.10.2.2 separate() 函数

separate() 用于将单个字符列拆分为多个列。

```
1 separate (data, col, into, sep = " ", remove = TRUE, convert = FALSE)
```

参数	描述
data	数据框对象。
col	列名或位置索引。
into	新变量的名称（字符向量）；可用 NA 省略对应输出列。
sep	列之间的分隔符（字符串或正则表达式）。
remove	若为 TRUE，从输出数据框中移除输入列。
convert	若为 TRUE，对新列调用 type.convert()，并设置 as.is = TRUE。

```
1 # 导入 tidyverse 包
2 library (tidyverse)
3 long <- tidy_dataframe %>%
4   gather (Group, Frequency,
5   Group.1:Group.3)
6
7 # 使用 separate () 函数将数据扩展为更宽的格式
8 separate_data <- long %>%
9   separate (Group, c ("Allotment",
10  "Number") )
11
12 # 打印更宽格式的数据
13 separate_data
```

输出的结果：

```

1 S.No Allotment Number Frequency
2 1 1 Group 1 23
3 2 2 Group 1 345
4 3 3 Group 1 76
5 4 4 Group 1 212
6 5 5 Group 1 88
7 6 6 Group 1 199
8 7 7 Group 1 72
9 8 8 Group 1 35
10 9 9 Group 1 90
11 10 10 Group 1 265
12 11 1 Group 2 117
13 12 2 Group 2 89
14 13 3 Group 2 66
15 14 4 Group 2 334
16 15 5 Group 2 90
17 16 6 Group 2 101
18 17 7 Group 2 178
19 18 8 Group 2 233
20 19 9 Group 2 45
21 20 10 Group 2 200
22 21 1 Group 3 29
23 22 2 Group 3 101
24 23 3 Group 3 239
25 24 4 Group 3 289
26 25 5 Group 3 176
27 26 6 Group 3 320
28 27 7 Group 3 89
29 28 8 Group 3 109
30 29 9 Group 3 199
31 30 10 Group 3 56

```

1.10.2.3 unite() 函数

`unite()` 用于将多列合并为一列（可指定分隔符）。

```
1 unite (data, col, ..., sep = " ", remove = TRUE)
```

参数	描述
<code>data</code>	数据框对象。
<code>col</code>	合并后新列的名称。
<code>...</code>	需要合并的列；若留空，默认选择所有变量。
<code>sep</code>	合并时用于连接各值的分隔符。
<code>remove</code>	为 <code>TRUE</code> 时，从输出中移除被合并的原始列。

`unite()` 是 `separate()` 的反向操作。换言之，若要撤销此前的 `separate()`，可用 `unite()` 将拆分后的两列重新合并为一列。下面的示例把 `Group` 与 `Number` 两列合并，并使用点号 “.” 作为分隔符。

```
1 # 导入 tidyverse 包
2 library (tidyverse)
```

```
3  
4 long <- tidy_dataframe %>%  
5 gather (Group, Frequency,  
6 Group.1:Group.3)  
7  
8 # 使用 separate () 函数将数据扩展为更宽的格式  
9 separate_data <- long %>%  
10 separate (Group, c ("Allotment",  
11 "Number") )  
12  
13 # 使用 unite () 函数合并  
14 # Allotment 和 Number 列  
15 unite_data <- separate_data %>%  
16 unite (Group, Allotment,  
17 Number, sep = ".")  
18  
19 # 打印新的数据框  
20 unite_data
```

结果如下：

```
1 S.No Group Frequency  
2 1 1 Group.1 23  
3 2 2 Group.1 345  
4 3 3 Group.1 76  
5 4 4 Group.1 212  
6 5 5 Group.1 88  
7 6 6 Group.1 199  
8 7 7 Group.1 72  
9 8 8 Group.1 35  
10 9 9 Group.1 90  
11 10 10 Group.1 265  
12 11 1 Group.2 117  
13 12 2 Group.2 89  
14 13 3 Group.2 66  
15 14 4 Group.2 334  
16 15 5 Group.2 90  
17 16 6 Group.2 101  
18 17 7 Group.2 178  
19 18 8 Group.2 233  
20 19 9 Group.2 45  
21 20 10 Group.2 200  
22 21 1 Group.3 29  
23 22 2 Group.3 101  
24 23 3 Group.3 239  
25 24 4 Group.3 289  
26 25 5 Group.3 176  
27 26 6 Group.3 320  
28 27 7 Group.3 89  
29 28 8 Group.3 109  
30 29 9 Group.3 199  
31 30 10 Group.3 56
```

1.10.2.4 spread () 函数

`spread()` 是 `gather()` 的反向操作：`spread()` 将“键—值”对展开为多列，使数据变宽，更适合某些类型的分析；`gather()` 则相反，把多列压缩为“键—值”对，使数据变长。`spread ()` 的语法是：

```
1 spread (data, key, value, fill = NA, convert = FALSE)
```

参数	描述
data	数据框对象。
key	列名或位置索引，作为展开后的列名来源。
value	列名或位置索引，作为展开后的取值来源。
fill	缺失值填充值；设置后用该值替换 NA。
convert	若为 TRUE，对新列调用 <code>type.convert()</code> ，并设置 <code>as.is = TRUE</code> 。

```
1 long <- tidy_dataframe %>%
2   gather (Group, Frequency,
3   Group.1:Group.3)
4
5 # 使用 separate () 函数将数据扩展为更宽的格式
6 separate_data <- long %>%
7   separate (Group, c ("Allotment",
8   "Number") )
9
10 # 使用 unite () 函数合并Allotment 和 Number 列
11 unite_data <- separate_data %>%
12   unite (Group, Allotment,
13   Number, sep = ".")
14
15 # 使用 spread () 函数将数据扩展为更宽的格式
16 back_to_wide <- unite_data %>%
17   spread (Group, Frequency)
18
19 # 打印新的数据框
20 back_to_wide
```

结果如下：

```
1 > back_to_wide
2 S.No Group.1 Group.2 Group.3
3 1 23 117 292
4 2 345 89 1013
5 3 76 66 2394
6 4 212 334 2895
7 5 88 90 1766
8 6 199 101 3207
9 7 72 178 898
10 8 35 233 1099
11 9 90 45 1991
12 10 265 200 56
```

1.10.2.5 nest() 与 unnest() 函数

nest() 与 unnest() 用于管理复杂数据结构：

- nest(): 将多列合并为一个列表列 (list-column)，适用于层次化数据模型。
- unnest(): 反向操作，把列表列展开为多列。⁹

语法示例：

```
1 nest (data, ..., .key = "data")
```

参数	描述
data	数据框对象。
...	待选择的列；如果为空，则默认选择所有变量。
key	新列的名称，可以是字符串或符号。

以下代码将数据集中的 tidy_dataframe 的 Group.1 列进行嵌套操作。

```
1 df <- tidy_dataframe
2 # 使用 nest () 函数将 Group.1 列嵌套在 tidy_dataframe 中
3 df %>% nest (data = c (Group.1) )
```

结果如下：

```
1 > df %>% nest (data = c (Group.1) )
2 # A tibble: 10 × 4
3   S.No Group.2 Group.3 data
4     <int> <dbl> <dbl> <list>
5     1 117    29    <tibble [1 × 1]>
6     2 89     101   <tibble [1 × 1]>
7     3 66     239   <tibble [1 × 1]>
8     4 334    289   <tibble [1 × 1]>
9     5 90     176   <tibble [1 × 1]>
10    6 101    320   <tibble [1 × 1]>
11    7 178    89    <tibble [1 × 1]>
12    8 233    109   <tibble [1 × 1]>
13    9 45     199   <tibble [1 × 1]>
14   10 200    56    <tibble [1 × 1]>
```

1.10.2.6 数据清洗函数

fill()、drop_na() 和 replace_na() 是处理缺失数据的关键函数：

- fill(): 按列用相邻非缺失值向前/向后填补缺失值，在时间序列数据中尤为常用。
- drop_na(): 删除包含缺失值的行。
- replace_na(): 用指定值替换数据框中的 NA。

由于篇幅所限，本文不对所有函数逐一介绍；建议在“帮助”选项卡中检索相应函数以获取详细说明。不难看出，tidyr 包具有强大的数据预处理与清洗能力，是数据分析不可或缺的工具。

注意 unnest() 的部分参数已被弃用；使用 unnest() 时，请参考帮助页面获取最新用法。

⁹注意：unnest() 的部分旧参数已弃用，使用时请参考帮助页面获取最新用法。

1.10.3 ggplot2 包

`ggplot2` 是一种基于“图形语法”的绘图系统，由 Hadley Wickham 创建。它提供连贯的框架，便于构建从简单到复杂且美观的可视化。其基于图层的方法允许先绘制基础图形，再逐步添加图层以定制可视化的各个方面。本文将结合 `mtcars` 数据集展示常见用法（参考：<https://cran.r-project.org/web/packages/ggplot2/ggplot2.pdf>）。

1. **数据 (Data)**: 图形的基础，指将用于可视化的实际数据集。
2. **美学映射 (Aesthetics)**: 将数据映射到视觉属性，如 x/y 轴、颜色、大小、形状等。
3. **几何对象 (Geometric Objects)**: 确定可视化类型，如点、线、直方图、条形图、箱线图等。
4. **分面 (Facets)**: 按分组将数据拆分为多个子面板显示，便于对比分析不同子集。
5. **统计变换 (Statistical Transformations)**: 对数据进行统计处理（如分组、汇总、平滑），之后再实施可视化。
6. **坐标 (Coordinates)**: 将数据点映射到坐标系统（常为笛卡尔，也可为极坐标），并可调整比例与坐标轴范围。
7. **主题 (Theme)**: 最高层级的非数据设置，涵盖字体、背景、网格线等影响整体外观的元素。

1.10.3.1 初始化数据层

```
1 ggplot (data = mtcars) +
2   labs (title = "Visualisation of mtcars Dataset")
```

基础数据层的可视化如图 1.8a 所示。

1.10.3.2 完善美学层

```
1 ggplot (data = mtcars, aes (x = hp, y = mpg, col = disp) ) +
2   labs (title = "Automobile Data: Displacement and Efficiency")
```

完善后的美学层可视化如图 1.8b 所示。

1.10.3.3 设置几何层

这里，我们采用简单的散点图来探究马力 (`hp`) 与燃油效率 (`mpg`) 的关系，并通过颜色渐变表示排量 (`disp`)。以下是 R 代码实现，可视化结果见图 1.8c。

```
1 ggplot (data = mtcars, aes (x = hp, y = mpg, col = disp) ) +
2   geom_point () +
3   labs (title = "Horsepower versus Fuel Efficiency",
4         x = "Engine Power (HP) ",
5         y = "Efficiency (MPG) ")
```

通过尺寸、颜色和形状进行增强。

```

1 # 加入尺寸
2 ggplot (data = mtcars, aes (x = hp, y = mpg, size = disp) ) +
3 geom_point () +
4 labs (title = "Engine Power vs Fuel Efficiency",
5 x = "Engine Power (HP) " ,
6 y = "Efficiency (MPG) ")
7
8 # 使用形状和颜色
9 ggplot (data = mtcars, aes (x = hp, y = mpg, col = factor (cyl) ,
10 shape = factor (am) ) +geom_point () +
11 labs (title = "Automobile Efficiency by Cylinder Count and Transmission",
12 x = "Engine Power (HP) " ,
13 y = "Efficiency (MPG) ")

```

这些图表的可视化结果分别在图 1.8d 和图 1.8f 中显示。

接下来，我们绘制一个直方图，R 代码如下：

```

1 # 构造直方图
2 ggplot (data = mtcars, aes (x = hp) ) +
3 geom_histogram (binwidth = 10, fill = "skyblue", color = "black") +
4 labs (title = "Distribution of Horsepower",
5 x = "Engine Power (HP) " ,
6 y = "Frequency")

```

直方图见图 1.8f。

1.10.3.4 扩展面板层

为了深入了解，使用面板来并列展示数据中的子群体。例如，我们可以为具有不同发动机配置或传动方式的汽车单独绘图。结果分别在图 1.9a 和图 1.9b 中展示。

```

1 # 按传动类型分隔行
2 p <- ggplot (data = mtcars, aes (x = hp, y = mpg) ) + geom_point ()
3 p + facet_grid (am ~ .) +
4 labs (title = "Fuel Efficiency vs Engine Power by Transmission Style")

```



```

1 # 按气缸数量分段列
2 p + facet_grid (. ~ cyl) +
3 labs (title = "Fuel Efficiency vs Engine Power by Cylinder Count")

```

1.10.3.5 应用统计层

我们应用线性模型刻画数据中的趋势：

```

1 ggplot (data = mtcars, aes (x = hp, y = mpg) ) +
2 geom_point () +
3 stat_smooth (method = lm, col = "firebrick") +
4 labs (title = "Trend in Fuel Efficiency versus Horsepower")

```

图 1.9c 中展示了相关结果。

1.10.3.6 调整坐标层

这一层精细调整了数据点在图表中的排列方式，调整了坐标轴的刻度和数据的空间分布：

```

1 ggplot (data = mtcars, aes (x = wt, y = mpg) ) +
2 geom_point () +
3 stat_smooth (method = lm, col = "firebrick") +
4 scale_y_continuous ("Efficiency (MPG)", limits = c (5, 35), expand = c
5 (0, 0) ) +
6 scale_x_continuous ("Car Weight", limits = c (1, 20), expand = c (0, 0) )
7 +
8 coord_equal () +
9 labs (title = "The Weight to Fuel Efficiency Relationship",
10 x = "Vehicle Weight (1000 lbs)",
11 y = "Efficiency (MPG)")

```

坐标层的调整详见图 1.9d。

引入 `coord_cartesian()` 可以在图表区域内实现局部放大。

```

1 # Enhanced focus with coord_cartesian ()
2 ggplot (data = mtcars, aes (x = wt, y = hp, col = am) ) +
3 geom_point () + geom_smooth () +
4 coord_cartesian (xlim = c (2, 5) )

```

使用 `coord_cartesian ()` 进行聚焦的效果见图1.9e。

1.10.3.7 定义主题层

通过使用主题进一步自定义，可以精细调整图表的美学元素。在以下代码中，使用了 `theme_minimal()`，它提供了一个干净且简约的外观，没有背景网格线且坐标轴最小化。

`ggplot2` 中有几个内置的主题，包括 `theme_gray ()`（默认主题）、`theme_bw ()`、`theme_linedraw ()`、`theme_light ()`、`theme_dark ()`、`theme_minimal ()` 和 `theme_void ()` 等。每个主题都可以通过额外的主题元素如 `theme ()`、`element_line ()`、`element_rect ()` 和 `element_text ()` 进一步自定义。这一系列函数允许对各种图表组件如文本、线条和框背景进行详细的自定义，使用户能够根据不同的风格偏好和数据展示环境定制他们的可视化效果。

```

1 # 使用主题定制背景
2 ggplot (data = mtcars, aes (x = hp, y = mpg) ) +
3 geom_point () +
4 facet_grid (. ~ cyl) +
5 theme_minimal () +
6 labs (title = "Comparative Engine Power and Efficiency by Cylinder Count")
7
8 # 精细调整图表尺寸
9 ggplot (data = mtcars, aes (x = wt, y = mpg) ) +
10 geom_point () +
11 coord_cartesian (xlim = c (2, 5) ) +
12 labs (title = "Detailed View of Weight and Efficiency Correlation")

```

1.10.3.8 使用 `ggplot2` 包绘制各类图表

与基础 R 的绘图函数如 `hist()`、`boxplot()` 或 `barplot()` 不同，`ggplot2` 提供了一套符合其自身框架和语法的函数集。以下是使用 `mtcars` 数据集介绍 `ggplot2` 中各种图表类型的一些示例。

- 直方图: `geom_histogram ()`。
- 箱形图 (Boxplot): `geom_boxplot ()`。
- 条形图 (Bar Plot): `geom_bar ()` 或 `geom_col ()`。
- 饼图 (Pie Chart): `coord_polar ()`。
- 密度图 (Density Plot): `geom_density ()`。
- QQ 图 (QQ-Plot (Quantile-Quantile Plot)): `stat_qq ()` 和 `stat_qq_line ()`
- 等高线图 (Contour Plot) : `geom_contour ()` 或 `stat_contour ()`。
- 图像图 (Image Plot): `geom_raster ()` 或 `geom_tile ()`。

```

1 library (ggplot2)
2
3 # 直方图
4 ggplot (mtcars, aes (x = mpg) ) +
5   geom_histogram (binwidth = 1, fill = "blue", color = "black") + labs (title
6     = "Histogram of Miles Per Gallon")
7
8 # 箱形图
9 ggplot (mtcars, aes (x = factor (cyl) , y = mpg) ) +
10   geom_boxplot (fill = "orange", color = "black") +
11   labs (title = "Boxplot of Miles Per Gallon by Cylinder Count")
12
13 # 条形图
14 ggplot (mtcars, aes (x = factor (cyl) ) ) +
15   geom_bar (fill = "green", color = "black") +
16   labs (title = "Bar Plot of Car Counts by Cylinder")
17
18 # 饼图 (使用条形图和 coord_polar)
19 ggplot (mtcars, aes (x = factor (1) , fill = factor (cyl) ) ) +
20   geom_bar (width = 1) + coord_polar (theta = "y") + labs (title = "Pie Chart
21     of Cylinder Counts")
22
23 # 密度图
24 ggplot (mtcars, aes (x = mpg) ) +
25   geom_density (fill = "magenta") + labs (title = "Density Plot of Miles Per
26     Gallon")
27
28 # QQ图
29 ggplot (mtcars, aes (sample = mpg) ) +
30   stat_qq () + stat_qq_line () + labs (title = "QQ-Plot of Miles Per Gallon")
31
32 # 二维密度等高线图, 针对 mtcars 数据集
33 ggplot (mtcars, aes (x = wt, y = mpg) ) +
34   stat_density_2d (aes (fill = after_stat (level) ) , geom = "polygon",
35   color = "white") +
36   scale_fill_viridis_c () +

```

```

34 labs (title = "2D Density Contour Plot of mtcars Dataset",
35 x = "Weight (wt) ",
36 y = "Miles Per Gallon (mpg) ",
37 fill = "Density")

```

1.10.3.9 保存和提取图表

最后，我们演示如何保存我们的可视化结果。`extracted_plot <- plot` 将图表对象 `plot` 分配给一个名为 `extracted_plot` 的新变量。这可以帮助我们在同一脚本或会话中稍后使用或修改，而无需从头开始绘图。

```

1 plot <- ggplot (data = mtcars, aes (x = hp, y = mpg) ) + geom_point ()
2 ggsave ("engine_power_vs_efficiency.png", plot)
3 # 将图表作为变量以便后续使用
4 extracted_plot <- plot

```

1.11 通过 API 获取开放数据（以世界银行为例）

现代金融计量研究高度依赖可重复、可扩展的数据获取方式。应用程序编程接口（Application Programming Interface, API）使我们能够通过 R 语言程序化地下载公开数据源（如世界银行、FRED、OECD、IMF、Eurostat 等），并将其无缝嵌入分析流程与可重复报告中。

什么是 API？

API（应用程序编程接口）是不同软件之间进行通信的契约。最常见的是基于 HTTP 的 REST 风格接口：

- **端点 (endpoint)**：一个 URL，例如 `https://api.worldbank.org/v2/...`，表示要访问的资源；
- **方法**：常用 GET（读取数据）；
- **查询参数**：位于 URL 问号后的键值对，例如 `?format=json&per_page=20000`；
- **返回格式**：常见为 JSON（默认）、CSV 或 XML；
- **身份认证**：部分 API 需要 API 密钥；
- **速率限制 (rate limit) 以及分页 (pagination)**：限制单位时间内的请求次数，以及一次返回的记录条数。

1.11.1 在 R 中访问 API 的两条路径

1. **使用专用 R 包（首选）**：很多权威机构已提供 R 包封装其 API，例如：世界银行（WDI, <https://cran.r-project.org/package=WDI>）、FRED（fredr, <https://cran.r-project.org/package=fredr>）、OECD（OECD, <https://cran.r-project.org/package=OECD>）、IMF（imfr, <https://cran.r-project.org/package=imfr>）、Eurostat（eurostat, <https://cran.r-project.org/package=eurostat>）等。优点是语法简洁、内置数据清洗与注释；缺点是覆盖面随包的维护情况而异。

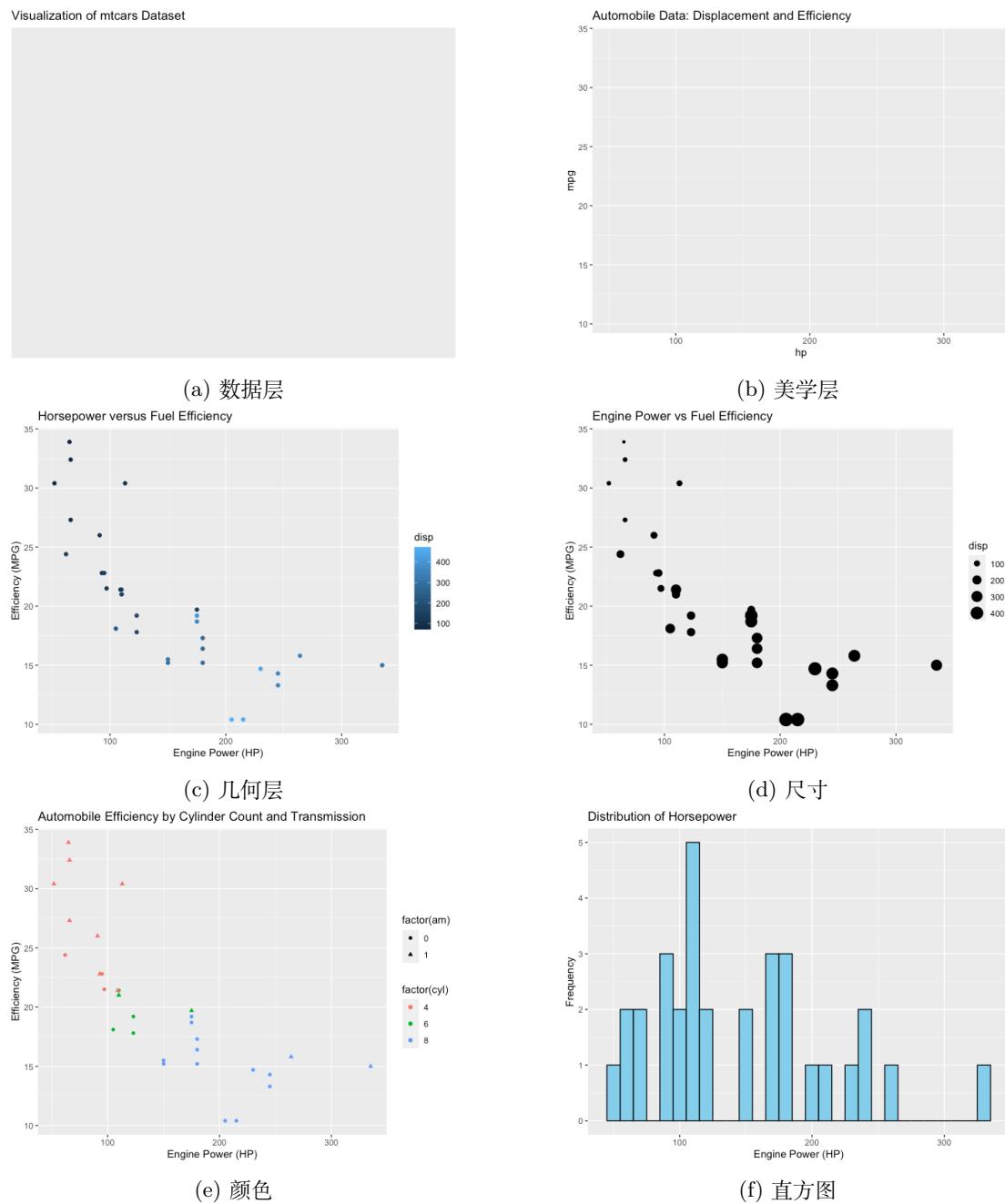


图 1.8: 使用 ggplot2 对 mtcars 数据集进行可视化 (第 I 部分)

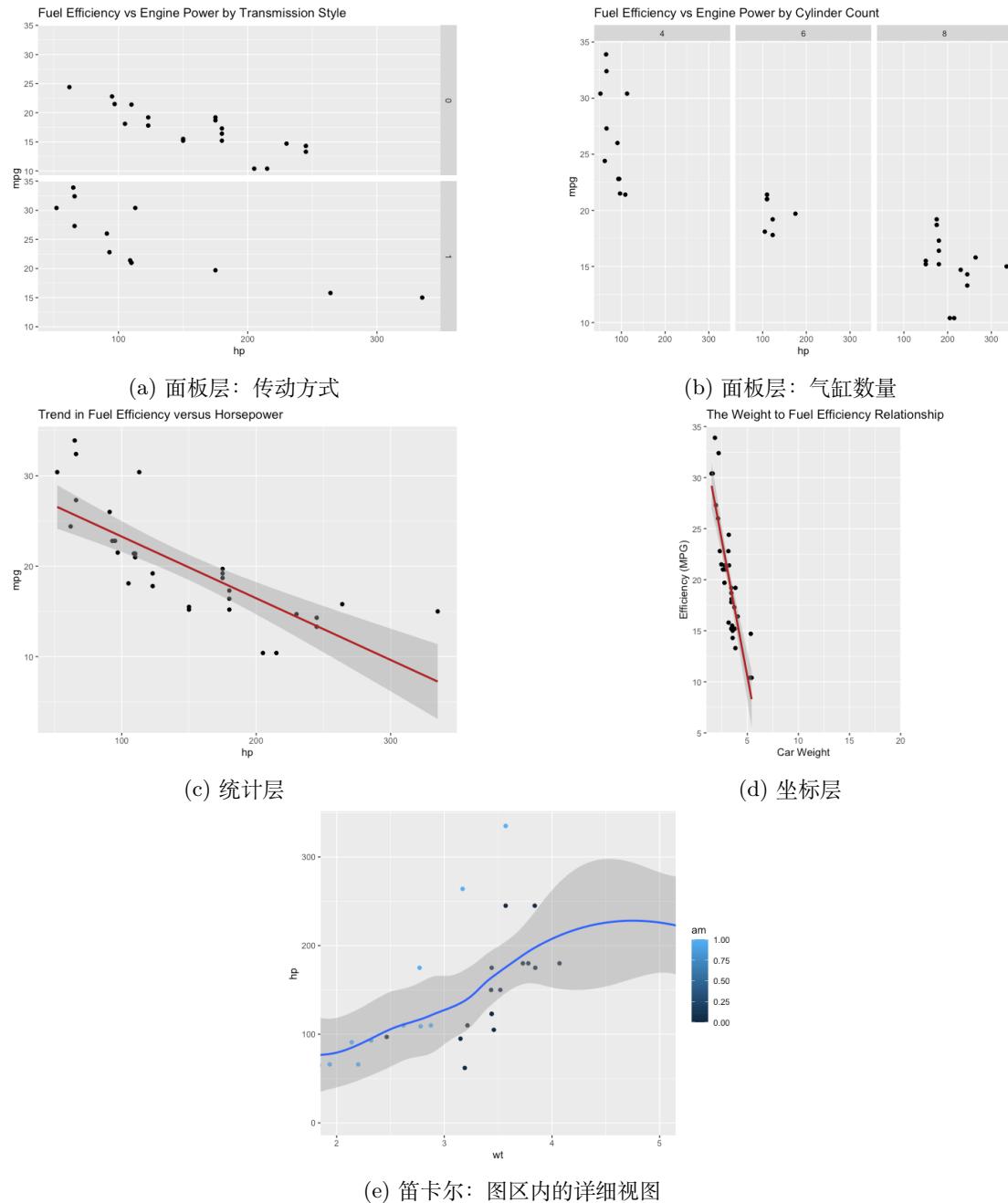


图 1.9: 使用 ggplot2 对 mtcars 数据集进行可视化 (第 II 部分)

2. **直接调用 API**: 用通用 HTTP 客户端（如 `httr2`, <https://httr2.r-lib.org/>）发送请求，配合 `jsonlite` (<https://cran.r-project.org/package=jsonlite>) 解析 JSON¹⁰，自行处理端点、参数、认证、分页与错误处理。适合需要 API 最新能力或定制管线的场景。

安全与可重复性的实践

- 密钥管理**: 将 API Key 写入用户主目录的 `.Renviron` 文件（例如 `FRED_API_KEY=xxxxxxxx`），在脚本中使用 `Sys.getenv("FRED_API_KEY")` 读取，避免将密钥硬编码到代码中。
- 限流与重试**: 对频繁请求添加 `Sys.sleep()`，或在 `httr2` 包中使用 `req_retry()`。
- 缓存与存储**: 将原始响应或处理后的数据保存为 `.rds/.csv` 文件（也可使用 `pins` 包），以减少重复请求。
- 记录元数据**: 保存下载时间、接口端点、请求参数和数据字典，便于审计与实验复现。

1.11.2 示例：世界银行 API

世界银行开放数据涵盖宏观、人口、环境等指标。下面给出两种常用方式：使用 `WDI` 包（推荐）与使用通用 HTTP 客户端直接请求 API（开发者文档参见¹¹；基础端点示例¹²）。

1.11.2.1 方式 A：使用 WDI 包（推荐）

```

1 # 若未安装请先运行: install.packages("WDI")
2 library(WDI)
3 library(dplyr)
4 library(ggplot2)
5
6 # 1) 搜索指标（例如人均GDP（不变价美元，2015基年））
7 WDIsearch("gdp per capita") |> head()
8
9 # 2) 下载中国/英国/美国 1990-2023 年的人均GDP与总人口
10 wb <- WDI(
11   country    = c("CN", "GB", "US"),
12   indicator  = c(gdppc = "NY.GDP.PCAP.KD", pop = "SP.POP.TOTL"),
13   start      = 1990, end = 2023, extra = TRUE
14 )
15
16 # 3) 整理并绘图
17 wb <- wb |>
18   as_tibble() |>
19   arrange(country, year)
20
21 ggplot(wb, aes(x = year, y = gdppc, color = country)) +
22   geom_line() +
23   labs(title = "人均GDP（不变价美元，2015基年）",

```

¹⁰JSON 是 JavaScript Object Notation 的缩写——一种轻量级的文本数据交换格式。它易于人阅读与编写，也便于机器解析与生成，几乎与编程语言无关。

¹¹<https://datahelpdesk.worldbank.org/knowledgebase/topics/125589-developer-information>

¹²<https://api.worldbank.org/>

```

24     x = "年份", y = "人均GDP (常量美元)",
25     color = "国家")

```

说明: WDI() 会自动处理分页与部分元数据; 也可用 WDIsearch() 根据关键词查找指标代码 (WDI 包主页¹³)。

1.11.2.2 方式 B: 直接调用世界银行 API (httr2 + jsonlite)

```

1 # 若未安装请先运行:
2 # install.packages(c("httr2","jsonlite","tibble","purrr","dplyr","ggplot2"))
3
4 library(httr2)
5 library(jsonlite)
6 library(tibble)
7 library(purrr)
8 library(dplyr)
9 library(ggplot2)
10
11 # 世界银行端点示例:
12 # https://api.worldbank.org/v2/country/{ISO3}/indicator/{INDICATOR}?format=
13 #   json&per_page=20000
14
15 fetch_wb <- function(iso3, indicator) {
16   url <- sprintf("https://api.worldbank.org/v2/country/%s/indicator/%s",
17     iso3, indicator)
18
19   req <- request(url) |>
20     req_url_query(format = "json", per_page = 20000) |>
21     req_timeout(60) |>
22     req_retry(max_tries = 3)
23
24   resp <- req_perform(req)
25   j <- resp_body_json(resp, simplifyVector = TRUE)
26
27   # 返回结构通常为 list(meta, data)
28   if (length(j) < 2 || is.null(j[[2]])) {
29     return(tibble(iso3c = iso3, year = integer(), value = numeric()))
30   }
31
32   as_tibble(j[[2]]) |>
33     transmute(
34       country = country$value,
35       iso3c   = country$iso3code,
36       year    = as.integer(date),
37       value   = as.numeric(value)
38     ) |>
39     arrange(year)
40 }
41
42 # 获取中国/英国/美国的人均GDP (NY.GDP.PCAP.KD)

```

¹³<https://cran.r-project.org/package=WDI>

```

41 df <- map_dfr(c("CHN", "GBR", "USA"), fetch_wb, indicator = "NY.GDP.PCAP.KD")
42
43 ggplot(df, aes(year, value, color = iso3c)) +
44   geom_line() +
45   labs(title = "World Bank API (直接请求) — 人均GDP",
46        x = "年份", y = "人均GDP (常量美元)",
47        color = "国家")

```

说明：将 `per_page` 设为较大值可尽量减少分页；在生产环境中应读取响应头中的总页数并分页循环抓取。若请求失败，建议结合 `req_retry()` 配置指数退避（参见 `httr2` 文档¹⁴；参见 `jsonlite` 文档¹⁵）。

1.11.3 其他常用公共数据 API 与 R 包速览

为便于快速定位常用国际宏观与官方统计数据源及其 R 语言接口，表 1.1 汇总了数据平台、推荐 R 包、典型数据类型及 API Key 需求（链接已嵌入表格，便于直接跳转）。

FRED 快速示例（需 API 密钥）

```

1 # 若未安装: install.packages("fredr")
2 library(fredr)
3
4 # 1) 在 ~/.Renviron 写入: FRED_API_KEY=你的密钥
5 fredr_set_key(Sys.getenv("FRED_API_KEY"))
6
7 # 2) 下载美国CPI (示例 series_id)
8 cpi <- fredr(series_id = "CPIAUCSL", observation_start = as.Date("2000-01-01"
  ""))

```

OECD 快速示例

```

1 # 若未安装: install.packages("OECD")
2 library(OECD)
3
4 # 列出可用数据集元信息
5 meta <- get_datasets()
6
7 # 直接抓取某数据集 (示例: 商业景气的综合先行指标 MEI_CLI)
8 cli <- get_dataset("MEI_CLI", start_time = 2015)

```

IMF 快速示例

```

1 # 若未安装: install.packages("imfr")
2 library(imfr)
3
4 # IFS: 消费者价格指数 (示例指标代码根据 IMF 数据字典选择)
5 df_imf <- imf_data(
6   database_id = "IFS",
7   indicator    = "PCPI_IX",
8   country      = c("CN", "GB"),

```

¹⁴<https://httr2.r-lib.org/>

¹⁵<https://cran.r-project.org/package=jsonlite>

表 1.1: 常用宏观/统计数据平台与 R 接口一览

平台	推荐 R 包 (链接)	典型数据	是否需 Key
世界银行 (World Bank)	WDI (https://cran.r-project.org/)	WDI 全库 (宏观、人口、环境、治理等)	否
圣路易斯联邦储备银行 FRED (Federal Reserve Bank of St. Louis, FRED)	fredr (https://cran.r-project.org/)	美国及全球宏观时序 (CPI、GDP、利率等)	是
经合组织 OECD (Organisation for Economic Co-operation and Development)	OECD (https://cran.r-project.org/)	OECD 数据平台 (QNA、MEI、PPP、教育等)	否
国际货币基金组织 IMF (International Monetary Fund)	imfr (https://cran.r-project.org/)	IFS、BOP 等 IMF 数据库	否/可选
欧盟统计局 Eurostat	eurostat (https://cran.r-project.org/)	欧盟官方统计 (国民账户、价格、就业等)	否
美国能源信息署 EIA (U.S. Energy Information Administration)	eia (https://cran.r-project.org/)	能源价格、产量、库存等	是

注：“是否需 Key”指是否需要申请 API 密钥；部分平台在匿名访问时可能存在配额或速率限制。

```

9   start      = 2015, end = 2024
10 )

```

1.11.4 实践要点与小结

- **封装函数**: 将“请求 → 解析 → 整理”封装为可复用函数(如 `fetch_wb`)，便于组合调用与单元测试。
- **统一命名与单位**: 数据下载后应立即规范变量命名、计量单位与注释，并形成数据字典，避免后续歧义与反复。
- **缓存与版本锁定**: 将原始响应或清洗后的数据落盘(如 `.rds`)，同时写入时间戳、请求参数与来源；必要时保留数据快照，确保结果可复现。
- **配额与限流**: 批量抓取时设置请求间隔与并发上限；对 `429/5xx` 等状态码采用带“指数退避”的重试策略，并支持断点续传，降低任务失败风险。

1.12 章节总结

本章围绕 R 语言在金融计量经济学中的应用展开，按“环境 → 编程 → 数据 → 可视化 → 工具 → 网络数据”的脉络系统介绍。首先回顾 R 的起源及其与 S 语言的关系、R Core 与 CRAN 生态，并概述 R 在金融计量中的优势与典型应用场景。随后介绍集成开发环境 RStudio (Posit)，说明 IDE 的核心面板与常用功能(编辑器、控制台、环境/对象浏览、绘图/包/帮助等)，以及 Shiny、R Markdown 等配套工具。

在编程基础部分，依次展示了基本算术与常数(e 、 π 、 Inf / NaN)、循环结构(`for`/`while`/`repeat`)及其替代方案(`apply`家族)，函数定义与作用域/闭包，并通过近似计算自然常数 e 的三个实现与“对数收益率”的金融例子加深理解。数据结构部分系统说明了向量、列表、数据框、矩阵、数组与因子及其典型用法。

数据处理部分涵盖了数据对象的导入(`read.csv`、`readxl`、`haven` 等)、R 基础包与内置数据集、子集/过滤/汇总等常见操作；缺失值处理包含均值/中位数插补、`VIM::kNN` 最近邻、多重插补(`mice`)、回归与卡尔曼滤波等思路；导出与持久化涉及 `write.csv`/`writexl`/`haven` 以及 `save`/`save.image`。可视化方面，先用基础图形绘制 GDP 时间序列(并通过 `ts()` 改善坐标与网格)，再列举散点图、直方图、箱线图、成对图、密度图、QQ 图等常见图形；随后以 `ggplot2` 的“图形语法”为主线，分层讲解 Data/Aesthetics/Geoms/Facets/Stats/Coordinates/Themes，并给出 `mtcars` 的多种示例与图形保存方式。帮助系统部分演示了 RStudio Help 与 `?rnorm` 文档的用法与阅读要点。

在“tidyverse”部分，本章重点介绍 `dplyr` 的“动词”与管道 `%>%`，使用 `starwars` 数据演示筛选、选择、派生、分组汇总与排序；`tidyr` 章节则围绕“整洁数据”阐释 `gather`/`separate`/`unite`/`spread`/`nest` 及 `fill`/`drop_na`/`replace_na` 等清洗工具。随机数与模拟部分先基于 `rnorm` 与其他分布的采样，再用“分位数转换/逆 CDF 法”从均匀分布生成正态样本，通过数值积分与求根实现，并以直方图与核密度估计检验结果特性。

“通过 API 获取开放数据”部分给出两条在 R 中访问 API 的路径：其一是使用权威封装包(如 `WDI/fredr/OECD/imfr/eurostat`)，其二是基于 `httr2` 与 `jsonlite` 直接请求并解析 JSON。示例以世界银行数据为主，分别用 `WDI` 包与手动编写的请求函数抓取人均 GDP 并进行可视化；同时给出 FRED、OECD、IMF、Eurostat、EIA 等常用平台与对应

R 包的速览。最后总结实践要点：函数封装、命名与单位规范/数据字典、缓存与版本锁定、配额管理、指数退避重试以及断点续传等工程实践。

本章末设置了配套习题，覆盖基础编程与可视化、`dplyr/tidyr` 数据整理、随机数与中心极限定理的模拟验证，以及 API 数据抓取与可复现性工作流的综合练习，帮助读者巩固与迁移所学。

1.13 习题

1. 讨论逆变换采样方法及其在从指定分布生成随机变量中的应用。
 - (a) 请用你的话解释什么是逆累积分布函数方法 (inverse transform sampling, ITS)? 为什么均匀分布在逆变换采样中至关重要?
 - (b) 定义什么是累积分布函数 (CDF)，并说明为什么逆变换采样不常用于离散分布 (可提及广义逆函数与实现难点)。
 - (c) 如何检验生成的随机变量是否确实符合预期分布?
 - (d) 描述如何使用逆变换采样方法从指数分布生成随机变量。
2. 假设我们使用 `dplyr` 包来操作 `starwars` 数据集：
 - (a) 使用 `filter()` 函数筛选出 `species` 为 Human 的角色，并选择其 `name`、`height` 与 `mass` 列。用 `dplyr` 代码实现上述操作。结果中有多少个角色？平均身高与平均体重分别是多少？
 - (b) 使用 `mutate()` 为 `starwars` 添加一列 `height_m`，表示以米为单位的身高（将 `height` 除以 100）。编写代码并展示结果前五行。`height_m` 中是否存在缺失值？若有，如何处理？
 - (c) 使用 `arrange()` 按 `mass` 降序排列，并展示前十个角色的信息。说明这些角色中是否存在 `mass` 为 NA 的情况；若存在，应如何在排序前处理缺失值。
 - (d) 使用 `group_by()` 与 `summarize()` 按 `species` 分组，计算各组的平均身高与平均体重。展示前五个物种的结果。哪个物种平均身高最高？平均体重最低？
 - (e) 使用管道操作符 `%>%` 将以下步骤串联：筛选 `species` 为 Droid 的角色，选择 `name` 与 `height` 列，并按 `height` 降序排列。编写代码并展示结果；讨论在这种链式操作中使用管道的优势。
3. 使用 `ggplot2` 包创建可视化有助于深入理解数据、直观呈现结果。在以下练习中，你将应用 `ggplot2` 的语法要素绘制不同图表，并探索可视化方法。
 - (a) 使用 `mtcars` 数据集，创建一个基础的 `ggplot` 对象并设置数据映射（美学映射，`aes`）。
 - i. 编写 R 代码初始化 `ggplot` 对象，设置数据为 `mtcars`，并添加标题“Visualization of mtcars Dataset”。
 - (b) 使用 `aes` 创建散点图，展示汽车马力（`hp`）与燃油经济性（`mpg`）之间的关系。
 - i. 用颜色映射表示排量（`disp`）。完善美学层并添加相应标题。
 - ii. 在散点图基础上，说明并演示如何通过添加趋势线、调整颜色与透明度、设置点形状、添加数据标签等方式提升视觉解读。

- (c) 设置几何层并创建直方图，展示 `mtcars` 中马力的分布。
- 使用 `geom_histogram()` 创建直方图，设定 `binwidth = 10`，并设置合适的填充颜色。
 - 解释分布形态：是否存在某些马力区间集中较多车辆？
- (d) 使用 `facet_grid()` 按变速箱类型 (`am`) 分组，创建分面图。
- 编写 R 代码为不同变速箱类型的车辆创建散点图分面，并添加标题。
 - 解释为何分面能提供额外信息，尤其是在比较不同子群体时。
- (e) 添加统计层，使用线性模型绘制趋势线 (`stat_smooth(method = "lm")`)。
- 为马力 (`hp`) 与燃油经济性 (`mpg`) 的关系添加线性模型趋势线。
 - 说明趋势线对理解数据的帮助；此方法最适合哪类数据关系？
- (f) 应用 `coord_cartesian()` 在图表上聚焦特定区域范围。
- 将散点图的 `x` 轴范围限制在 100 到 300。
 - 说明为何使用 `coord_cartesian()`，而不是直接缩放数据或过滤数据点。
- (g) 使用主题层 (`theme`) 自定义图表外观。
- 为散点图应用 `theme_minimal()`，并添加标题“Engine Power vs Efficiency - Clean Theme”。
 - 讨论自定义主题对图表的影响，以及为何在不同场景下采用不同主题。
- (h) 使用 `ggplot2` 创建箱线图 (`boxplot`)，比较不同气缸数 (`cyl`) 对燃油经济性 (`mpg`) 的影响。
- 使用 `geom_boxplot()` 创建箱线图，设定 `x = cyl, y = mpg`。
 - 解释不同气缸数的燃油经济性分布，是否存在显著的组间差异？
- (i) 保存一个你认为最有趣的图表，并命名为 `my_ggplot.png`。
- 编写 R 代码将该图表保存为 PNG 文件，文件名为 `my_ggplot.png`。
 - 说明保存图表的用途，以及如何在报告或演示中有效使用这些图表。
4. 中心极限定理是统计学的基本概念，描述样本均值的分布形态，对数据分析与假设检验具有重要意义。本题将探索其理论基础及在掷骰子实验中的应用；无需实际计算，但需要分析 R 代码并手动推导关键步骤。
- 解释中心极限定理及其意义；当样本量增大时，样本均值的分布趋近于何种分布？
 - 设函数 `dice_roll(n)` 模拟掷 `n` 次六面骰并返回平均值。用 R 代码描述如何利用此函数验证中心极限定理：例如设定 $n \geq 30$ 与重复次数 `m`，使用 `replicate()` 生成样本均值，绘制直方图并叠加正态曲线，并比较不同 `n` 的分布变化。
 - 给出一段模拟掷骰子 10 次并计算平均值的代码，指出其中的错误或改进点。
- ```
1 n_dice <- 10
2 roll_avg <- mean(sample(1:6, n_dice, replace = TRUE))
```
- 运用中心极限定理，计算 60 次掷骰子的样本均值介于 3 和 4 之间的概率，并清晰展示你的假设与步骤。
  - 下面的 R 代码用于模拟掷骰子以演示中心极限定理。请评估代码是否有缺失或冗余步骤，并给出必要的修正版本。

```

1 # 中心极限定理的蒙特卡罗模拟
2 n_dice <- 30
3 n_rolls <- 1000
4 roll_sums <- replicate(n_rolls, sum(sample(1:6, n_dice, replace =
 TRUE)))
5 roll_means <- roll_sums / n_dice

```

5. 下列 R 代码通过抛硬币实验演示大数定律。

```

1 flips <- sample(c(0, 1), 1000, replace = TRUE)
2 cum_avg <- cumsum(flips) / (1:1000)
3 plot(cum_avg, type = "l", ylim = c(0, 1),
4 ylab = "Cumulative Average", xlab = "Number of Flips")
5 abline(h = 0.5, col = "red")

```

- (a) 描述上述代码的每个步骤，解释其如何模拟抛硬币、计算平均值，并将结果与大数定律关联。
- (b) 编写一组 R 代码，比较大数定律下样本均值与理论期望值的收敛速度。
- (c) 改变样本量（抛硬币次数）对模拟结果有何影响？
6. 作为一名金融分析师，你需要使用过去十多年的上证指数和深证成指的月度收盘数据，利用 `ggplot2` 在 R 中创建可视化，分析股市波动率的时间序列特征与趋势。
- (a) 从相关数据库下载上证指数与深证成指月度收盘数据，计算月度收益率，并评估数据的基本统计特征（如均值、标准差等）。
- (b) 使用 `ggplot2` 创建时间序列图展示两指数的月度收益率。再用 `ggplot2` 绘制直方图或密度图，比较两指数的收益率分布。
- (c) 使用滚动窗口标准差计算并展示波动率的变化，并用 `ggplot2` 绘制波动率的时间序列图。
- (d) 描述你的数据处理与可视化步骤，解释波动率时间序列图反映的风险动态，并讨论如何据此进行投资决策与风险管理。
7. 使用 R 访问开放数据接口，完成以下任务：
- (a) 使用 `WDI`<sup>16</sup> 分别获取 `NY.GDP.PCAP.KD`（人均 GDP）与 `SP.POP.TOTL`（总人口）数据，计算 1990–2023 年期间中国与英国的人均 GDP 增长率并作图。
- (b) 使用 `httr2`<sup>17</sup> 直接请求世界银行 API（端点<sup>18</sup>），获取 `EN.ATM.CO2E.PC`（人均 CO<sub>2</sub> 排放）数据，自行处理分页与缺失值，生成 `tibble` 格式数据并保存为 `.rds`。
- (c) 申请 FRED API 密钥，将 `CPIAUCSL` 与 `FEDFUNDS` 数据导入，合并为按月对齐的面板数据，绘制 2000 年以来通胀率与联邦基金利率的动态对比（使用 `fredr`<sup>19</sup>）。

<sup>16</sup><https://cran.r-project.org/package=WDI>

<sup>17</sup><https://httr2.r-lib.org/>

<sup>18</sup><https://api.worldbank.org/>

<sup>19</sup><https://cran.r-project.org/package=fredr>

## 2 引言和背景

### 2.1 金融市场

金融市场是一个涉及货币资金借贷、外汇买卖、有价证券交易、债券与股票发行，以及黄金等贵金属买卖的复杂网络。它不仅反映国民经济的多个层面，也被誉为经济的“晴雨表”，因为能够提供观测与监控经济运行状态的直观指标。按交易标的不同，金融市场可分为货币市场、债券市场、股票市场、外汇市场、衍生品市场、保险市场和黄金市场等部分；各市场功能与特点各异，共同构成全球经济的基础。按交易中介角色，金融市场又分为直接金融市场和间接金融市场：前者允许经济主体之间直接交易金融资产，后者则涉及金融中介机构，如银行和基金公司，帮助调节资金供需。按金融工具的交易阶段，可分为发行市场与流通市场：发行市场关注新金融工具的初始销售，流通市场处理其后续买卖。按是否有固定交易场所，可分为场内市场与场外市场：场内市场（如证券交易所）具有明确的物理场所和标准化程序；场外市场多指非标准化合约的交易，如部分衍生品。按金融工具的本源与从属关系，可分为传统金融市场与金融衍生品市场，后者专注于期权、期货等衍生工具。最后，按地域范围可分为国内金融市场与国际金融市场，后者跨越国界，涉及多种货币和资本的跨境流动。

这些市场的构成与分类体现了金融市场在全球经济中的核心作用：不仅促进资金的有效配置，也支持全球经济的稳定与增长。通过各类市场的有序运作，金融市场为长期项目提供资金，使企业和个人得以优化资源配置并加强风险管理。

金融市场对中国的重要性不仅体现在对经济发展的推动作用，还体现在它是国家核心竞争力的重要组成部分。金融市场作为国民经济的“血脉”，不仅保障资金高效流动与资源最优配置，还通过支持创新和技术升级，促进实体经济发展与国家战略实施。此外，金融市场的稳定直接关系到整体经济的稳定；通过提供必要的金融服务并保障金融体系安全，促进社会和谐与经济持续健康发展。因此，加快建设金融强国，不仅是中国经济社会发展的需要，也是全面建设社会主义现代化国家的重要战略任务。

按交易的金融工具类型，可作如下分类：货币市场主要交易在1年以内到期的短期工具；债券市场涉及1年以上到期的长期债券；股票市场是交易上市公司股份的市场；外汇市场涉及不同货币之间的交易；衍生品市场包括期货合约、期权等衍生工具；商品市场则交易黄金等贵金属、豆粕、玉米等农产品，以及包括原油在内的能源化工产品。接下来，我们将分别介绍这些市场的运作方式、主要交易工具及其在全球经济中的作用与重要性。

#### 2.1.1 货币市场

货币市场是固定收益市场的重要组成部分，该市场交易的主要金融工具包括：

**定义 2.1 (国库券 (Treasury Bills, 简称 T-bills) ):** 国库券是一种高流动性的短期政府债务工具，通常以低于面值的价格发行，投资者的收益来自到期时面值与购买价格之间的差额。

**定义 2.2 (大额可转让定期存单 (Certificates of Deposit, 简称 CDs) ):** 大额存单是银行发行的一种定期存款凭证，通常到期前不得提前支取，但可在二级市场转让。

**定义 2.3 (欧洲美元存单 (Eurodollar CDs) ):** 此类存单以美元计价，由外国银行或美国机构在其境外分支机构发行，不受美联储 (*Federal Reserve*, 简称 *Fed*) 直接监管。虽然名称中带有“欧洲”二字，但现今已与欧洲无直接关联，该名称源于二战后美元在欧洲银行的广泛使用。

**定义 2.4 (商业票据 (Commercial Paper, 简称 CP) ):** 商业票据是大型企业发行的短期无担保债务工具，常用于在金融市场上直接融资，以替代银行借款。

**定义 2.5 (银行承兑汇票 (Bankers' Acceptances, 简称 BA) ):** 银行承兑汇票是由银行承兑并承诺付款的远期汇票，广泛用于国际贸易，有助于在缺乏信用基础的交易中建立信任。此类汇票通常以贴现形式交易，可在公开市场出售或持有至到期。

**定义 2.6 (回购协议 (Repurchase Agreements, 简称 repos 或 RPs) ):** 回购协议指证券卖方承诺在未来以更高价格回购同一证券的交易安排，实质上构成一项有抵押的短期借款。由于交易中证券所有权发生转移，在法律上与普通借款不同。相应的“逆回购协议” (*Reverse Repo*) 是指证券买方发起的、为交易对手提供有抵押贷款的交易安排。回购利率常被视作抵押融资成本的基准。

货币市场的知名利率包括：

- **联邦基金利率 (Federal Funds Rate):** 指美国存款类金融机构在联邦基金市场上相互拆借准备金余额的隔夜无担保利率。FOMC 设定目标区间，市场成交情况体现在有效联邦基金利率 (EFFR) 上。
- **伦敦银行同业拆借利率 (London Interbank Offered Rate, 简称 LIBOR):** 曾是批发货币市场中的无担保同业资金利率，长期作为全球金融合约的重要参考利率。美元市场已过渡至 SOFR，其他币种亦采用各自的替代基准利率。
- **担保隔夜融资利率 (Secured Overnight Financing Rate, 简称 SOFR):** 美元市场上 LIBOR 的替代基准利率，由纽约联邦储备银行发布，基于以美国国债为担保的隔夜回购交易数据计算，具有覆盖面广、透明度高的特点。

中国货币市场的知名利率包括：

- **人民银行贷款基准利率：**由中国人民银行设定，用于指导商业银行贷款定价。自 2013 年以来，利率市场化改革推进，银行可在基准利率基础上以“加/减点”方式自主确定贷款利率。
- **存款基准利率：**用于指导银行对客户存款定价的基础利率。与贷款基准利率类似，存款利率亦可在基准利率基础上适度调整。

- **上海银行间同业拆放利率 (SHIBOR, Shanghai Interbank Offered Rate)**: 中国金融市场的重要基准利率之一，由上海银行同业拆放市场的主要报价行报出，反映银行间市场的短期资金成本，涵盖从隔夜到一年期的多个期限。
- **香港银行同业拆息 (HIBOR, Hong Kong Interbank Offered Rate)**: 香港银行同业市场的主要基准利率，影响香港地区贷款、抵押贷款及其他金融产品的定价。

### 2.1.2 债券市场

债券是一种在未来特定日期预设支付固定现金流的合约。债券市场规模庞大：根据美国财政部 2024 年 1 月 2 日发布的数据，美国国债总额首次达到 34 万亿美元。在美国市场，债券类型主要包括：

- **零息债券** (零息票债券)：到期一次性偿还本金，期间不发生利息支付（通常以折价发行）。
- **国库券**：原始期限不超过 1 年的政府短期债务工具。
- **附息债券**：定期支付利息，通常每 6 个月或每年支付一次；票面利率以面值百分比表示（默认面值 100 元），到期偿还本金 100 元。
- **中长期国债**：原始期限在 1 至 10 年（含 10 年）的为中期，> 10 年的为长期。
- **债券相关衍生品**：包括期权、期货、互换、互换期权，以及利率上限 (cap) / 利率下限 (floor) / 利率领式 (collar) 期权等。

联邦机构债务及欧洲债券：

- **联邦机构债务**：例如吉利美 (Ginnie Mae)、房利美 (Fannie Mae)、房地美 (Freddie Mac) 等机构发行的与美国住房抵押贷款相关的证券，其募集资金用于支持住房抵押贷款。上述机构的宗旨在于提升住房信贷可获得性、降低对区域经济条件的依赖。由于存在显性或隐性担保，此类证券通常具有较高的信用评级，从而降低融资成本。
- **外国债 (Yankee / Samurai 等)**：在某国本土市场、以该国本币计价、由外国人发行的债券。
- **欧洲债券 (Eurobond) 与离岸美元债 (Eurodollar Bond)**：欧洲债券指以某种货币计价、但在该货币发行国以外的国家或地区发行的债券；离岸美元债是其中以美元计价、在美国境外发行的子类别。

中国债券市场的主要债券类型：

- **国债**：由财政部发行，市场上信用评级较高，通常被视为无风险资产之一。
- **地方政府债券**：用于资助地方基础设施等项目。近年来，随着对地方政府债务的规范化管理，地方政府债券市场发展较快。
- **企业债券**：包括公司债券、中期票据等，由企业发行，用于融资扩展业务或偿还存量债务等。
- **金融债**：主要由政策性银行和商业银行发行，包括次级债券，用于提高银行的资本充足率。
- **同业存单**：由银行业存款类金融机构在银行间市场发行，用于短期筹资，流动性较高。

### 2.1.3 股票市场

股票（股权证券）代表公司所有权的一部分。每一股普通股使持有人在公司年度股东大会（AGM）上拥有一票投票权，并享有公司所有权的财务利益。股东选举董事会，董事会负责选聘并监督管理层。无法出席年会的股东可通过委托他人代为行使投票权。当部分股东试图更换管理层，而管理层通过争取其他股东支持或采取防御策略进行反击时，可能发生代理权之争。

**普通股是一种剩余索取权：**若公司清算，剩余索取人最后从资产中分得收益。普通股有时被称为“次级股”（相对于优先股）。有限责任意味着股东的最大损失限于其原始投资额；与个体业主不同（除非购买董事责任险），债权人不能对其个人资产（如房屋、汽车、家具等）提出索赔。若股票在证券交易所上市，为公开交易股票；未上市的称为私募股权。普通股通常可能每年支付数次股息。公司也可回购自身股票，这可能提升股价，并带来资本利得。以下概念与股票市场相关：

**优先股：**兼具股权与债权特征（例如可转换债券；优先股与次级债务亦称夹层融资/资本）。承诺每年支付固定金额，通常不具备投票权，因而类似于永续债（永久债券、无固定到期期限债券）。不同之处在于，公司对股息支付具有自主权，没有合同义务；股息为累计，未支付部分将累计，并在向普通股股东分配前优先支付。优先股可以是可赎回的（类似可回售债券）、可转换的（按预先设定的比率转换为普通股）或可调整股息率的（股息率与市场利率挂钩）。与债券的利息支付（以及银行贷款利息）不同，优先股股息对公司而言不属于税前扣除费用。

**存托凭证：**代表对一定数量外国股份的所有权（对应股份称为存托股份；也可能是优先股或债券）。由国内存托银行持有（通过其外国分支机构或本地托管银行），并发行相应凭证。存托凭证可在国内交易所上市或进行场外交易，为投资者提供便捷、成本更低的方式投资外国证券。其通常以美元或欧元计价，相关方面临外汇风险。它也使外国公司更易满足纽约或伦敦等地上市的严格注册要求。全球存托凭证（GDRs）在美国以外的一个或多个市场交易；美国存托凭证（ADRs）在美国市场交易。

**市场指数：**市场指数反映股票市场的整体估值水平，覆盖范围有窄（包含几十种证券）也有广（包含数千种证券），如道琼斯工业平均指数、标普 500 指数和罗素 3000 指数。指数构建可采用不同加权方法，详见后续章节。交易所交易基金（ETF）允许投资者交易跟踪宽基指数收益的资产，例如 SPDR ETF 跟踪标普 500 指数。

**股权期货市场：**E-mini S&P 500（常简称 E-mini，亦有其他 E-mini 合约）在芝加哥商品交易所（CME）Globex 电子平台交易，是一种股指期货合约。单个合约名义价值为标普 500 指数的 50 倍。E-mini 自 1997 年 9 月 9 日由 CME 推出，当时大型标普期货合约价值过大（指数的 500 倍，超过 50 万美元），许多小型交易者难以参与。E-mini 迅速成为最受欢迎的股指期货合约之一。对冲基金通常偏好交易 E-mini 而非大型标普期货合约；后者仍采用公开喊价方式，相比之下，Globex 全电子交易系统不存在固有延迟。E-mini 成交极为活跃，名义成交额长期位居全球股指期货市场前列。

**股指期货市场：**股指期货是基于股票市场指数的期货合约，广泛用于投资、对冲风险与价格发现。投资者可在预期指数上涨时买入（做多），或预期下跌时卖出（做空），从而利用波动获取收益。

中国股指期货市场起步于 2010 年，中国金融期货交易所推出首个股指期货产品——沪深 300 股指期货。随着市场发展，中国股指期货市场已成为亚洲重要的衍生品市场之一。

2015 年股市异常波动期间，中国股指期货市场经受严峻考验。2015 年 6 月 15 日至 8 月 26 日的 52 个交易日，上证指数从 5178 点降至 2850 点，跌幅 44.96%。期间，股指期货被部分分析人士指认为波动加剧的原因之一。作为应对，中金所于 2015 年 9 月 2 日

公告，自 9 月 7 日起将非套期保值交易的保证金比例提高至 40%，并将 10 手以上的交易定义为异常交易。此举使股指期货成交量一度暴跌 99% 以上；虽在短期内抑制了波动率溢出，但也显著限制了市场功能发挥。

经过近一年半的严格管理，2017 年 2 月 16 日，中金所对沪深 300、中证 500 和上证 50 三大品种交易规则作出调整，包括异常交易手数认定、交易保证金和平今仓手续费等，自 2 月 17 日起实施，被视为市场逐步回归常态的重要一步。其后，随着市场稳定与发展，相关规则逐步放宽。总体而言，中国股指期货市场的发展体现了我国金融市场从探索走向成熟的进程；尽管面临挑战，该市场在风险管理与提升市场效率方面的作用日益凸显。

在新时代的十年伟大变革中，我国金融领域围绕改革创新与稳定发展持续探索，充分体现了新时代中国特色社会主义思想在金融领域的科学指引。习近平总书记强调，金融安全是国家安全的重要组成部分，金融稳定关乎经济社会发展全局。2015 年异常波动后实施的严格管控措施，看似压制活跃度，实则在关键时刻筑牢金融安全防线，防止风险扩散，保障金融体系稳健运行，是“稳定压倒一切”在金融领域的生动实践。随着市场逐步平稳，自 2017 年起对股指期货交易规则调整并逐步放宽，体现了政策制定者对市场规律的把握与在动态平衡中促进市场发展的智慧。

通过精准的计量模型，可以量化评估股指期货市场的风险状况、波动特征及与股票市场的联动关系，为政策制定提供依据。例如，利用计量分析确定合理的保证金比例和交易手数限制，既有效防范过度投机引发的系统性风险，又能保障市场流动性和价格发现功能，从而在维护金融安全与稳定的同时，释放金融市场活力，助力实体经济高质量发展，推动我国金融市场朝着更加成熟、开放、稳健的方向迈进。

## 2.2 市场类型以及交易方式

本节旨在探讨不同类型市场的运作方式和特点。每种市场都有其独特的结构与交易机制，适用于不同的交易需求与策略。

**定义 2.7 (经纪市场):** 拥有专业知识的经纪人提供搜索服务，通过收取费用（佣金）来撮合供需。例如：房地产市场；证券发行市场（投资银行家作为发行人和投资者之间的经纪人/承销商）；以及大宗股票的“楼上”市场（经纪人在交易所外寻找交易对手进行交易）。

**定义 2.8 (做市商市场):** 做市商为自营账户进行交易，建立资产库存，并通过报出买入价与卖出价来做市。投资者只需查看并比较做市商报出的价格。做市商通过为愿交易者提供即时性维持流动性，因而做市商市场被称为**报价驱动市场**。例如：场外交易市场(OTC)，这是由经纪人与做市商构成的去中心化网络，参与者通过协商决定证券的买卖（不是通过正式的交易所进行交易）。

**定义 2.9 (电子拍卖市场):** 所有交易者汇集到一个单一的场所（物理或“电子/虚拟”）进行买卖。订单在到达并匹配时，由称为匹配引擎的计算机系统执行，这被称为**订单驱动市场**。在此机制下，投资者无需在不同做市商之间搜索最佳价格（也无需支付此类中介费用）。

下面对部分主要交易所作简要介绍。

**纳斯达克交易所 (National Association of Securities Dealers Automated Quotations, 简称 NASDAQ):** 最初作为场外交易市场，采用做市商报价制度；经纪人代表

客户与报价更优的做市商联系并执行交易。1971 年之前，场外交易报价均为手工记录，并通过“粉红单”每日发布。1971 年，全国证券交易商协会自动报价系统（NASDAQ）成立，通过计算机网络连接经纪人和做市商，实现价格的实时显示与调整。最初该系统仅为报价系统，交易仍需经纪人与做市商直接协商；如今纳斯达克已发展为具备电子交易平台的交易所，能够以电子方式执行交易而无需直接协商，绝大多数交易由此完成。

**纽约证券交易所 (New York Stock Exchange, NYSE, 又称“大盘”)**: 过去每种证券由一位专家 (specialist) 通过维护限价订单簿管理该证券的交易；在流动性不足时，专家作为做市商介入以维护市场公平与有序，并通过佣金与买卖差价获利。传统的大厅交易模式下，代表客户的经纪人需直接找到相应专家成交；在如今占主导地位的电子交易模式下，订单可直接提交给专家并自动执行。对于大宗交易，若专家无法在大厅直接处理，将由“楼上”经纪人协商撮合。此外，NYSE 还运营公司债券电子交易平台（自动债券系统）。尽管如此，绝大多数债券交易（包括在 NYSE 上市的债券）仍在场外市场，经由电子报价系统连接的做市商网络完成。

**伦敦证券交易所 (London Stock Exchange, LSE)**: 直至 1997 年，LSE 与早期纳斯达克类似，为场外市场提供自动报价系统，证券公司兼任经纪人与做市商的双重角色；此后主要采用电子限价订单簿方式撮合。对于大宗交易或流动性较低的证券，仍会通过做市商方式成交。

中国的上海证券交易所和深圳证券交易所兼具经纪市场与做市商市场的部分特性。在经纪市场中，经纪人充当中介，帮助买卖双方匹配并完成交易；在两所交易所，经纪人（通常为证券公司）为客户提供买卖股票服务，协助寻找交易对手并成交。做市商市场的特点是做市商提供流动性，通过报出买入价与卖出价进行自营交易；在我国交易所尤其是股票交易中，通常存在做市商，他们通过持有一定库存并双向报价，保障市场流动性与效率。

近年来，随着交易和通信成本下降、信息获取更便捷以及线上经纪低佣金等因素，电子交易取代人工撮合已成全球趋势，推动算法交易与高频交易（High Frequency Trading, HFT）快速发展。在美国，《国家市场系统规章》(Regulation National Market System, Reg NMS) 与欧洲《金融工具市场指令》(Markets in Financial Instruments Directive, MiFID) 不仅鼓励新型电子交易场所的引入，也促进了股票交易电子化。例如，美国新兴的 DirectEdge、BATS，以及英国的 Chi-X 等，均在上述政策推动下设立。

**暗池交易与经纪人交叉网络 (Broker Crossing Networks)** 属于无交易前透明度的电子交易场所，即参与者看不到订单簿。其价格通常设定在一些公开交易所现有买卖报价的中点；此方式适合大宗交易且不希望影响价格的投资者。

在中国，随着金融市场快速发展与对外开放推进，电子交易已逐渐成为主流。大量交易通过上交所与深交所的电子系统完成。同时，我国也在探索暗池等创新机制。这类机制允许大宗交易在非公开环境下完成，从而减少对公开市场价格的直接冲击。

### 订单类型（电子市场）

- **市场订单 (market order)**: 立即以当前市场价格执行的买入 (bid) 或卖出 (ask) 订单。
- **限价订单 (limit order)**: 仅在达到或优于指定价格时执行。买入限价订单 (bid) 以指定价或更低价买入；卖出限价订单 (ask) 以指定价或更高价卖出。若限价不具竞争力或市场变化，订单可能被不执行。
- **停止订单 (stop order)** : 仅当市场价格达到预定价格时触发执行。停止买入订单 (stop-buy) 在价格达到或高于预定价时买入；止损订单 (stop-loss) 在价格达到或低于预定价时卖出。

- **冰山订单 (iceberg order)**: 限价订单的一种，仅显示小部分数量；当部分成交后，剩余数量重新变为可见。
- **挂钩订单 (pegged order)**: 价格随参考价格驱动，如某交易场所的买卖差价中点。

### 时间限制

- **日订单**: 在当日交易结束时到期。
- **开放订单**: 最长可持续六个月，除非被取消。

“直到取消” (Good Till Cancelled, GTC) 与 “全部成交或取消” (Fill or Kill, FOK) 是两类常见的时间/执行条件。GTC 订单一经下达，在被交易者取消或被完全执行前始终有效，可跨越多个交易日，有效期通常为数月（依交易平台而定）。FOK 订单要求交易所在接收指令时立即全额成交，否则将被完全取消，不保留任何挂单；若市场可用数量不足该订单要求，订单将被立即取消，确保不会产生部分成交。

自 2000 年以来，金融市场全球化趋势日益明显。2007 年，欧盟为提升市场透明度、增强竞争并加强消费者保护，制定了 MiFID——《金融工具市场指令》(Markets in Financial Instruments Directive)，并于 2014 年修订；MiFID II 自 2018 年开始实施，旨在提升透明度、保护投资者、促进竞争并规范跨境金融活动。美国证券交易委员会 (SEC) 于 2005 年推出 NMS 法规，全称 “全国市场体系规则” (National Market System)，致力于提高市场效率、增强交易透明度、保护投资者利益并促进市场整合。上述制度为交易所之间引入了竞争机制。与此同时，智能订单路由 (Smart Order Routing, SOR) 推动了市场间的互联互通，激发了场间与场内的竞争；流动性提供者之间的竞争弱化了做市商的基本垄断地位。

在中国，也形成了与 MiFID、NMS 类似的资本市场监管框架，旨在规范市场行为、保护投资者权益，并促进透明与公平。首先，《证券法》为股票、债券等证券市场的基础性法律，构建了关于证券发行、交易、信息披露与监管的总体框架，目标在于保护投资者、维护公平公正的市场秩序、促进资本市场健康发展。其后，作为主要监管机构，中国证监会通过规章与指引细化《证券法》的实施，包括对证券公司监管、交易规范与信息披露要求等，并承担对证券公司合规性的监督职责。最后，我国已建立多层次资本市场体系，包括主板、创业板及近年新设的科创板等，以适配不同类型企业的融资需求与投资者偏好；各板块依据企业发展阶段、规模与行业属性设置差异化上市标准与监管要求。（注：原中小板已于 2021 年并入深交所主板。）

## 2.3 收益率

计算收益率是衡量投资效果的直接指标，对个人投资者、金融分析师、基金经理以及企业决策者至关重要。收益率的波动率常被用作衡量风险的重要指标。此外，收益率还是研究资本市场行为、有效市场假说与资产定价模型等理论的基础数据。本节侧重于**收益率的构建、股指及其对应收益率与收益率的统计性质**。

以下给出简单收益率的定义：

**定义 2.10:** 时间  $s$  到  $t$  之间的简单总收益率为

$$\mathcal{R}_{s,t} = \frac{P_t + D_t}{P_s},$$

其中  $D_t$  为持有期  $[s, t]$  内支付的股息总额。相应的简单净收益率为

$$R_{s,t} = \mathcal{R}_{s,t} - 1.$$

由于价格非负，总收益率也非负；净收益率可以为负，但不低于  $-1$ 。按来源分解，总收益可写为：

$$\text{股息收益} = \frac{D_t}{P_s}, \quad \text{资本增值率} = \frac{P_t - P_s}{P_s},$$

两者之和即为  $\mathcal{R}_{s,t} - 1$ （即  $R_{s,t}$ ）。

接着我们考虑对数收益率，也称连续复利收益率：

**定义 2.11：** 在  $[t-1, t]$  上的连续复合回报或对数收益率为

$$r_t \equiv \log R_t = p_t - p_{t-1},$$

其中  $p_t \equiv \log P_t$ 。多时段的对数收益率为

$$r_{t,t+k} = \log R_{t,t+k} = \log(R_{t-k+1} \times R_{t-k+2} \times \cdots \times R_t) = r_{t-k+1} + r_{t-k+2} + \cdots + r_t.$$

在金融分析中，选择使用对数收益率而非简单收益率主要基于几个关键优点。首先，对数收益率具有对称性，能够均衡地处理资产价格的上涨和下跌。其次，对数收益率的可加性使得计算多个时间段的总收益率变得简单，这对于长期投资分析尤其有用。再者，当价格变动不大时，对数收益率可以作为简单收益率的良好近似，简化了分析过程。最后，对数收益率往往更接近正态分布，这与许多统计模型的基本假设相契合。

我们可以轻松地将日度对数收益率年化。常见做法是将平均日度对数收益率乘以 252，以得到平均年度对数收益率。之所以采用 252，是因为美股每年大约有 250–253 个交易日，常用近似为

$$252 \approx 365.2425 \times \frac{5}{7} - 10.$$

实际工作中，也可用样本期内的实际交易天数进行年化更为稳妥。

考虑一个高波动率市场中的投资组合，其价值先从 100 元下降至 50 元；随后又回升至 100 元。比较简单收益率与对数收益率：

### 1. 简单收益率

- 从 100 元降至 50 元，收益率为  $\frac{50-100}{100} = -50\%$ ；
- 从 50 元升至 100 元，收益率为  $\frac{100-50}{50} = 100\%$ ；
- 算术平均收益率为  $\frac{-50\% + 100\%}{2} = 25\%$ ，但两期累计收益  $(1-50\%)(1+100\%) - 1 = 0$ ，实际应为 0。

### 2. 对数收益率

- 从 100 元降至 50 元： $\log(\frac{50}{100}) = \log(0.5)$ ；
- 从 50 元升至 100 元： $\log(\frac{100}{50}) = \log(2)$ ；
- 对数收益率之和： $\log(0.5) + \log(2) = 0$ ，准确反映投资最终回到起始点。

显然，在高波动率市场中，对数收益率更能准确衡量收益表现。

值得注意的是，对数收益率具有**时间可加性**的特点：

$$r_{t,t+k} = r_t + r_{t+1} + \cdots + r_{t+k-1}.$$

因此，在对数收益率下，周收益率等于五个日收益率之和。而这一性质对简单收益率并不成立，即

$$R_{t,t+k} \neq R_t + R_{t+1} + \cdots + R_{t+k-1}.$$

另一方面，简单收益率具有**投资组合可加性**：对于固定时点  $t$  的投资组合权重  $w_{1t}, w_{2t}, \dots, w_{nt}$ ，有

$$R_t(\mathbf{w}) = w_{1t}R_{1t} + w_{2t}R_{2t} + \cdots + w_{nt}R_{nt}.$$

设  $n_i$  为股票  $i = 1, 2$  的购买数量，令  $V_t = n_1P_{1t} + n_2P_{2t}$  为时点  $t$  的**投资组合价值**， $w_{it} = n_iP_{it}/V_t$  为资产  $i$  在时间  $t$  的投资组合权重。则时点  $t+1$  的投资组合价值为

$$\begin{aligned} V_{t+1} &= n_1P_{1,t+1} + n_2P_{2,t+1} \\ &= \frac{w_{1t}V_t}{P_{1t}}P_{1,t+1} + \frac{w_{2t}V_t}{P_{2t}}P_{2,t+1} \\ &= \frac{w_{1t}V_t}{P_{1t}}P_{1t}(1 + R_{1,t+1}) + \frac{w_{2t}V_t}{P_{2t}}P_{2t}(1 + R_{2,t+1}) \\ &= w_{1t}V_t(1 + R_{1,t+1}) + w_{2t}V_t(1 + R_{2,t+1}) \\ &= V_t(1 + w_{1t}R_{1,t+1} + w_{2t}R_{2,t+1}). \end{aligned}$$

由此可得

$$\frac{V_{t+1} - V_t}{V_t} = w_{1t}R_{1,t+1} + w_{2t}R_{2,t+1}.$$

显见，对数收益率不具备**投资组合层面的可加性**：

$$r_t(\mathbf{w}) = \log\left(\frac{w_{1t}P_{1t} + \cdots + w_{nt}P_{nt}}{w_{1t}P_{1,t-1} + \cdots + w_{nt}P_{n,t-1}}\right) \neq w_{1t}r_{1t} + \cdots + w_{nt}r_{nt}.$$

当然，当时间间隔较短时，可近似认为具有可加性。

**定义 2.12 (累计收益与几何平均收益率)**：给定从  $t$  到  $t+k$  的  $k$  个单期简单总收益率

$R_t, R_{t+1}, \dots, R_{t+k-1}$  (此处  $R_{t+j} \equiv \frac{P_{t+j}+D_{t+j}}{P_{t+j-1}}$ )，则

$$\mathcal{R}_{t,t+k} = \prod_{j=0}^{k-1} R_{t+j}, \quad R_{t,t+k} = \mathcal{R}_{t,t+k} - 1.$$

若用对数收益率  $r_{t+j} = \log R_{t+j}$  表示，则

$$\log \mathcal{R}_{t,t+k} = \sum_{j=0}^{k-1} r_{t+j}, \quad \iff \quad \mathcal{R}_{t,t+k} = \exp\left(\sum_{j=0}^{k-1} r_{t+j}\right).$$

对应的几何平均（单期）简单收益率定义为

$$\bar{R}_{t,t+k}^{(G)} = (\mathcal{R}_{t,t+k})^{1/k} - 1 = \exp\left(\frac{1}{k} \sum_{j=0}^{k-1} r_{t+j}\right) - 1.$$

几何平均收益率与复利机制一致，能精确反映“先跌后涨”情形下的真实累计效果（见前述  $100 \rightarrow 50 \rightarrow 100$  的例子：累计收益为 0，而算术平均给出 25% 的偏高值）。因此，在跨期比较与长期评估中，几何平均较算术平均更稳健、更符合资产收益的复合本质。

### 2.3.1 股票指数的编制方法

目前，全球股票指数的编制主要采用算术平均法和加权平均法。**算术平均法**是通过计算一组选定股票价格的**简单算术平均数**来构建股价指数的方法。首先，选择一组具有代表性的样本股票，并以某年某月某日作为基期，确定基期指数。随后，在任何给定日，计算所有样本股票的平均价格，并将该平均价格与基期的平均价格进行比较。将此比值乘以基期指数，即得到当日的股票价格指数。这种方法简单直观，但可能无法准确反映市场中各股票的实际影响力，因为它假设所有股票对指数的影响相同。

**道琼斯工业平均指数 (Dow Jones Industrial Average, 简称 DJIA)** 是一个价格加权指数，由查尔斯·道和爱德华·琼斯于 1896 年创建，最初仅包含 12 家公司的股票。该指数旨在提供一个反映美国工业领域主要上市公司股票表现的简单有效指标。随着时间的推移，该指数已扩展至包含 30 家大型蓝筹股公司，这意味着指数的计算是基于其成分股的股票价格。具体来说，该指数由所有成分股的价格之和除以一个特定的除数（称为“道琼斯除数”）得出。该除数会随着时间推移进行调整，以应对成分股的股票分割、股息支付或成分股更换等事件。

**加权平均法则**考虑市场中不同股票的影响力，并给予不同的权重。这种方法首先根据股票在市场中的重要性赋予不同的权重，例如根据股票的市值、成交金额或流通股数来确定权重。然后，将每只股票的价格与其权重相乘并求和，再除以总权重，从而计算得到加权平均值。加权平均法更能反映市场整体的真实动态，尤其适用于那些股票差异较大的市场。

**标准普尔 500 指数 (S&P 500 Index)** 是一个基于市值加权的指数，包含美国股市中 500 家最大且最具代表性的上市公司。S&P 500 指数的成分股由标准普尔公司的选择委员会挑选，主要基于市值、流动性、行业分类以及其他经济因素。成分股必须为美国公司，且其主板上市须在标准普尔指数方法学所列的合格美国交易所（如 NYSE、Nasdaq 及其他合格交易所）进行。S&P 500 指数使用市值加权法计算，这意味着每家公司在指数中的权重与其调整后市值成正比。调整后市值指公司的股票价格乘以可自由流通的股票数量（自由流通市值）。自由流通市值的计算排除了主要股东、政府持有的股份以及其他不能自由交易的股份。指数值计算公式为：指数 = 总市值 / 除数。总市值是所有成分股的自由流通市值的总和。除数是一个经调整的数字，用来保持指数的连续性；在成分股发生股票分割、股息支付、增发新股或成分股变更时进行调整，以确保这些事件不会影响指数的整体水平。

**上证指数 (Shanghai Composite Index) 和深证成指 (Shenzhen Component Index)** 的编制方法如下。上证指数是以流通市值加权的价格指数，旨在反映上海证券交易所所有上市股票的整体表现。上证指数的计算涵盖在上海证券交易所上市的 A 股和 B 股。上证指数的基期为 1990 年 12 月 19 日，基点为 100 点，采用流通市值加权法计算，计算公式为：

$$\text{指数} = \frac{\sum(\text{个股价格} \times \text{自由流通股本})}{\text{基期自由流通总市值}} \times \text{基点}.$$

类似地，深证综指涵盖深圳证券交易所上市的 A 股和 B 股，其基期为 1991 年 4 月 3 日，基点同样为 100 点。深证成份指数同样采用流通市值加权计算方式，公式此处从略。

以下 R 代码可用于获取全球主要股指的历史数据，计算其对数收益率，并绘制对数收益率的时间序列图。

```
1 # 安装并加载quantmod和ggplot2包
2 if (!require ("quantmod")) install.packages ("quantmod")
3 if (!require ("ggplot2")) install.packages ("ggplot2")
4 library (quantmod)
5 library (ggplot2)
6 # 定义主要股指的符号
7 # 标准普尔500, 日经225, 富时100, 德国DAX
8 symbols <- c (^GSPC, ^N225, ^FTSE, ^GDAXI)
9
10 # 使用getSymbols函数从Yahoo Finance下载数据
11 stock_data <- lapply (symbols, function (sym) {
12 getSymbols (sym, src = "yahoo", from = "2020-01-01",
13 to = Sys.Date () , auto.assign = FALSE)
14 })
15 names (stock_data) <- c ("S&P 500", "Nikkei 225", "FTSE 100", "DAX")
16 # 计算每个股指的对数收益率
17 log_returns <- lapply (stock_data, function (x) {
18 x <- na.omit (x) # 去除缺失值
19 Delt (Cl (x) , type = "log")
20 })
21
22 # 将对数收益率数据框的列名设为股指名称
23 log_returns_df <- do.call (merge, log_returns)
24 names (log_returns_df) <- c ("S&P 500", "Nikkei 225", "FTSE 100", "DAX")
25 # 将xts对象转换为data.frame
26 log_returns_df <- data.frame (date = index (log_returns_df) ,
27 coredata (log_returns_df))
28 # 确保所有缺失值被移除
29 log_returns_long <- na.omit (log_returns_long)
30
31 # 绘制对数收益率图表
32 ggplot (log_returns_long, aes (x = date, y = `Log Return` , color = Index))
33 +
34 geom_line () +
35 theme_minimal () +
36 labs (title = "Logarithmic Return of Major Stock Indices",
37 x = "Date",
38 y = "Logarithmic Return",
39 color = "Stock Index") +
40 theme (legend.position = "bottom")
```

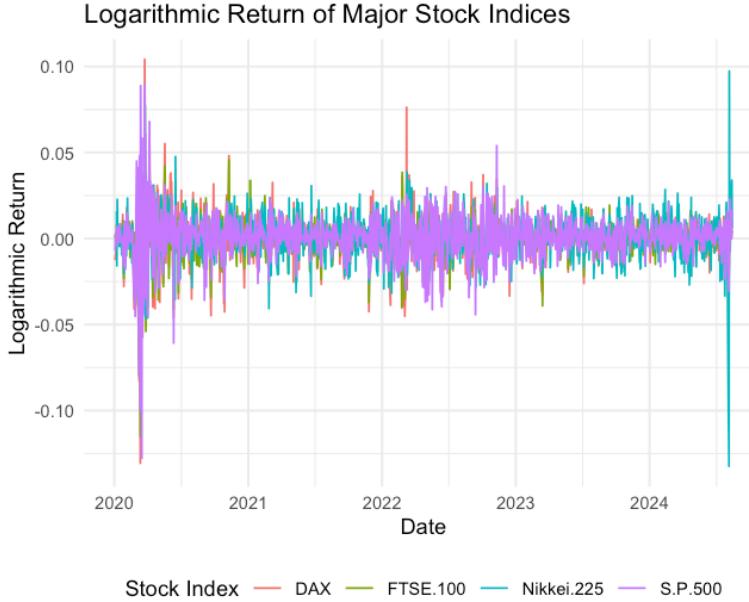


图 2.1: 主要股指的对数收益率

### 2.3.2 收益率的统计性质

由于受到市场条件、经济因素和投资者行为等多种不可预测因素的影响，收益率是随机变量，其数值无法确定。下面首先探讨收益率与对数收益率之间的关系。根据詹森不等式，

$$E[r] = E[\log(1 + R)] \leq \log(1 + E[R]).$$

又因当  $E[R] \neq 0$  时， $\ln(1 + E[R]) \leq E[R]$ ，故  $E[r] \leq E[R]$ 。一般而言， $E[R] \in (-1, 1)$ ，于是

$$\ln(1 + E[R]) = E[R] - \frac{1}{2}(E[R])^2 + \frac{1}{3}(E[R])^3 + \dots$$

当  $\Pr(-1 < R \leq 1)$  接近于 1 时，可在  $R = E[R]$  处对  $\ln(1 + R)$  作泰勒展开，得

$$\begin{aligned} E[r] &= E[\ln(1 + R)] = \ln(1 + E[R]) + \frac{E[(R - E[R])]}{1 + E[R]} - \frac{1}{2} \frac{E[(R - E[R])^2]}{(1 + E[R])^2} + \dots \\ &\simeq \ln(1 + E[R]) - \frac{1}{2} \frac{\text{Var}(R)}{(1 + E[R])^2} \simeq E[R] - \frac{1}{2} \text{Var}(R), \end{aligned}$$

其中最后一近似在  $|E[R]| \ll 1$  时成立。

通常用收益率的期望衡量投资的收益水平，而用回报的方差衡量投资风险。常用的综合性价比指标是夏普比率 (Sharpe Ratio)：

$$S = \frac{E[R] - R_f}{\sigma(R)} = \frac{E[R] - R_f}{\sqrt{\text{Var}(R)}},$$

其中  $R_f$  为无风险收益率， $E[R] - R_f$  称为风险溢价。直观上，夏普比率为正表示平均回报高于无风险收益；为负则低于无风险收益；夏普比率越高，投资组合的吸引力通常越强。

除夏普比率外，以下比率也可用于衡量投资绩效并考虑风险因素。

**索提诺比率 (Sortino Ratio)**: 与夏普比率类似，但仅关注投资组合回报的下行风险（即负回报的风险），而非总体波动率，因而更适合评估对下行风险敏感的策略。其计算公式为

$$\text{Sortino Ratio} = \frac{E[R] - R_f}{\sigma_{\text{down}}},$$

其中  $\sigma_{\text{down}}$  为下行标准差，即负回报的标准差。

**信息比率 (Information Ratio)**: 衡量投资组合相对基准（如市场指数）的超额回报，并按超额回报的波动率进行调整，用于评价投资经理相对基准的表现。其计算公式为

$$\text{Information Ratio} = \frac{E[R - R_b]}{\sqrt{\text{Var}(R - R_b)}},$$

其中  $R_b$  为基准回报率。

由于金融资产通常具备有限责任，例如股票价格的下界为 0，故简单收益率的下界为  $-100\%$ （即  $R_t \geq -1$ ）。因此，简单收益率本身不可能服从正态分布。即使单期  $R_t$  近似正态，多期复合  $R_{t,t+k}$  也不会是正态，因为正态变量的乘积并不服从正态分布。

更常见且合理的建模是假设：对数收益率（连续复利）服从正态，而简单总收益率服从对数正态：

$$1 + R \sim \text{Lognormal}(\mu, \sigma^2) \iff r = \ln(1 + R) \sim \mathcal{N}(\mu, \sigma^2).$$

**为何强调“金融资产一般为有限责任”？** 多数常见的投资形式（如普通股、债券及大部分衍生品）遵循有限责任原则。然而也存在例外情形，投资者可能承担超出初始投入的额外责任。例如：在一般合伙企业中，合伙人对企业债务与义务负有无限责任；若企业资产不足以清偿负债，合伙人的个人资产可能需用于偿债。再如，在证券或期货市场使用保证金账户交易时，投资者可能收到追加保证金通知（margin call），被要求追加资金以维持头寸；若行情不利，除亏损初始投资外，还可能产生超出该金额的负债。此外，某些衍生品（如特定类型的期权或未平仓的期货合约）亦可能带来超过初始投资的损失，尤其当投资者卖出裸期权（未持有相应标的资产）时，潜在损失可能极大。

**原油宝事件（2020 年 4 月）简述：** “原油宝”为中国银行推出的衍生品投资产品，允许个人投资者通过该行平台参与国际原油期货价格表现，主要追踪西得克萨斯中质油（WTI, West Texas Intermediate）与布伦特原油（Brent Crude Oil）的期货价格，区别于直接买入期货合约。2020 年 4 月 20 日，受疫情导致的需求暴跌与储存能力不足影响，WTI 期货价格历史性跌至负值（每桶  $-37.63$  美元）。由于产品设计存在缺陷，叠加中国银行未能及时平仓，不少持有近月合约的投资者出现巨额亏损。事件冲击了发行机构声誉与相关衍生品的市场信心，事后中国银行对内部风险管理与客户服务流程进行了审查与改进。

此外，我们可以采用偏度 (Skewness) 和峰度 (Kurtosis) 来衡量收益率的分布特征：前者度量分布的不对称性，后者衡量分布的尖峰性或重尾性。其定义为

$$\kappa_3 \equiv E\left[\frac{(r - \mu)^3}{\sigma^3}\right]; \quad \kappa_4 \equiv E\left[\frac{(r - \mu)^4}{\sigma^4}\right].$$

对正态分布变量，有  $\kappa_3 = 0$  且  $\kappa_4 = 3$ 。

当偏度  $> 0$  时, 右尾 (较大值) 较长、左尾较短, 主体更集中于较小值 (右偏/正偏); 当偏度  $< 0$  时, 左尾 (较小值) 较长、右尾较短, 主体更集中于较大值 (左偏/负偏)。正峰度 (Leptokurtic) 指  $\kappa_4 > 3$ , 分布较正态更尖、尾更厚; 负峰度 (Platykurtic) 指  $\kappa_4 < 3$ , 分布较正态更平、尾更薄。

设样本为  $\{r_1, \dots, r_T\}$ , 其样本均值、方差、偏度与峰度如下:

$$\bar{r} = \frac{1}{T} \sum_{t=1}^T r_t, \quad s^2 = \frac{1}{T} \sum_{t=1}^T (r_t - \bar{r})^2,$$

$$\hat{\kappa}_3 = \frac{\frac{1}{T} \sum_{t=1}^T (r_t - \bar{r})^3}{\left(\frac{1}{T} \sum_{t=1}^T (r_t - \bar{r})^2\right)^{3/2}}, \quad \hat{\kappa}_4 = \frac{\frac{1}{T} \sum_{t=1}^T (r_t - \bar{r})^4}{\left(\frac{1}{T} \sum_{t=1}^T (r_t - \bar{r})^2\right)^2}.$$

下例使用 R 的 `quantmod` 包获取上证指数(Shanghai Composite Index, 代码 000001.SS) 2023 年日收益率数据, 并计算平均收益率、方差、偏度与峰度 (偏度/峰度示例函数取自 `e1071` 包)。实际运行需安装并加载所需包, 并进行缺失值处理。

```

1 # 加载软件包
2 library (quantmod)
3
4 # 设置开始和结束日期
5 start_date <- "2023-01-01"
6 end_date <- "2023-12-31"
7
8 # 获取上证指数数据
9 getSymbols ("000001.SS", src = "yahoo", from = start_date, to = end_date)
10
11 # 计算日收益率
12 returns <- dailyReturn (Cl (get ("000001.SS")))
13
14 # 计算平均收益率和方差
15 r_bar <- mean (returns, na.rm = TRUE)
16 s_squared <- var (returns, na.rm = TRUE)
17
18 # 计算偏度和峰度
19 T <- length (na.omit (returns))
20 kappa_3 <- sum ((returns - r_bar) ^3, na.rm = TRUE) / (T * s_squared^(3/
 2))
21 kappa_4 <- sum ((returns - r_bar) ^4, na.rm = TRUE) / (T * s_squared^2)
22
23 # 输出结果
24 cat ("平均收益率: ", r_bar, "\n")
25 cat ("方差: ", s_squared, "\n")
26 cat ("偏度: ", kappa_3, "\n")
27 cat ("峰度: ", kappa_4, "\n")

```

## 2.4 金融经济学基础

金融计量经济学中广泛使用的数学与统计模型需要依托坚实的经济理论基础。例如资本资产定价模型 (CAPM) 等, 均源于严格的经济学推导。掌握这些模型的经济学原理, 是

正确应用并批判性评价这些模型的前提。因此本章将对金融经济学基础作简要回顾。

### 2.4.1 效用函数以及风险厌恶

有很多人认为经济学，与数学、物理、化学等基础学科相比是一个新兴的学科。但事实上经济学和概率论一样古老。在 1730 年代，数学家丹尼尔·伯努利的堂兄尼古拉一世·伯努利在给法国数学家皮耶·黑蒙·德蒙马特的信中提出了一个问题：设想有一个赌局，规则是连续掷硬币直到出现正面为止。如果第一次掷出正面，你赢得 1 元。如果第一次掷出反面，则需再掷一次，若第二次掷出正面，则赢得 2 元。如果第二次还是反面，就继续掷第三次，若这次掷出正面，则赢得 4 元，依此类推，每次赌注翻倍，直到掷出正面为止。赌局可能在第一次就结束，也可能一直掷下去。问题是，你最多愿意付出多少钱来参加这个赌局？

你愿意付出的最大金额应等于这个游戏的期望值：

$$\begin{aligned} E &= \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 4 + \frac{1}{16} \cdot 8 + \dots \\ &= \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \dots \\ &= \sum_{k=1}^{\infty} \frac{1}{2} = \infty. \end{aligned}$$

不难看出，这个赌局的期望是无穷大的，即理论上你应该愿意支付无限多的金钱来参加。但事实上很少有人愿意花大价钱参加这个赌局。丹尼尔·伯努利在 1738 年指出，人类决策不是基于金额期望，而是效用。并提出效用的边际效用递减原理以及最大效用原理。丹尼尔·伯努利用对数函数度量效用，因此，该赌局的期望效用如下：

$$\sum_{t=1}^{\infty} \left(\frac{1}{2}\right)^t \cdot \ln(2^{t-1}) = \ln 2 < \infty.$$

也就是说，如果效用函数是对数函数，人们愿意为这个赌局支付的入门费只有 2 元 ( $= e^{\ln 2}$ )。

之后，约翰·冯·诺依曼 (John von Neumann) 和奥斯卡·摩根斯特恩 (Oskar Morgenstern) 将期望效用理论 (expected utility theory) 系统化、公理化。在经济学中，理解和建模个体的偏好至关重要，因为它直接关系到如何解释和预测个体在不确定环境下的决策行为。学过经济学的读者应该知道，一个理性 (rational) 的偏好是具有完备性 (completeness) 与传递性 (transitivity) 的。用  $x$  以及  $y$  分别代表两个“商品束” (commodity bundle)<sup>1</sup>。我们使用符号  $\succeq$  来表示一个人对两个不同选项的偏好关系。 $x \succsim y$  ( $x, y \in X$ ) 表示对某人而言， $x$  至少和  $y$  一样好。

**定义 2.13：理性偏好：**偏好关系  $\succeq$  被称为理性，当且仅当满足以下两项：  
(i) 完备性：任意  $x, y \in X$ ,  $x \succeq y$  或  $y \succeq x$ ;  
(ii) 传递性：任意  $x, y, z \in X$ , 若  $x \succeq y$  且  $y \succeq z$ , 则  $x \succeq z$ 。

完备性意味着对于任意两个选项，人们都能进行比较；传递性意味着若  $x$  至少与  $y$  一样好，且  $y$  至少与  $z$  一样好，则  $x$  也至少与  $z$  一样好。

此外，还需对偏好的连续性作出假设。连续性使偏好关系可由效用函数表示，从而可以使用（包含微积分在内的）优化理论工具进行求解；反之，若偏好不连续，许多数学工具将

<sup>1</sup> “商品束”是一个经济学术语，它指的是由不同商品和服务的特定数量组成的集合。在微观经济学中，商品束通常用来分析消费者的选择行为、偏好和消费决策。商品束可以涵盖各种类型的商品和服务，如食品、服装、娱乐和交通等。

难以应用，问题的解析处理会变得复杂或不可行。显然，这一假设基本符合经济规律，能够描述并解释现实中观察到的多数经济行为。连续偏好意味着当选择的变动很小时，个体的偏好反应也是平滑的；在许多经济模型中（如消费者理论、生产理论），连续性是理解边际变化的基础。

**定义 2.14 (连续性):** 若偏好关系  $\succeq$  在极限下也能保持，则称其为连续。具体而言，给定一系列点对  $\{(x^n, y^n)\}_{n=1}^{\infty}$ ，若对所有  $n$  有  $x^n \succeq y^n$ ，则当  $x = \lim_{n \rightarrow \infty} x^n$ 、 $y = \lim_{n \rightarrow \infty} y^n$  时，必有  $x \succeq y$ 。

若偏好关系既理性又连续，则必然存在一个连续效用函数可以表示该偏好关系；即存在  $u(\cdot)$  使得当且仅当  $x \succeq y$  时， $u(x) \geq u(y)$ 。

**命题 2.1:** 若偏好关系  $\succeq$  理性且连续，则存在连续效用函数  $u : X \rightarrow \mathbb{R}$  表示该偏好关系： $x \succeq y \iff u(x) \geq u(y)$ 。

以上讨论有助于更深入地理解人类行为与决策的复杂性。这种理解使我们在政策设计与战略制定中，能够更准确地匹配个体的真实行为与需求，从而更有效地解决现实经济问题。

首先，我们将彩票分为两类：**简单彩票**与**复合彩票**。简单彩票是一种基本的随机化金融工具，由一组可能结果及其对应概率构成。

**定义 2.15 (简单彩票):** 一张简单彩票  $L$  定义为一个概率向量  $L = (p_1, \dots, p_n)$ ，其中  $p_i$  表示第  $i$  个结果发生的概率，满足  $p_i \geq 0$  ( $\forall i$ ) 且  $\sum_{i=1}^n p_i = 1$ 。

简单彩票的典型例子是**掷骰子**：若骰子公平，则六个面的出现概率均为  $1/6$ 。

复合彩票更为复杂：它由若干张简单彩票以及一套决定“选中哪张简单彩票”的**额外概率机制**组成。

**定义 2.16 (复合彩票):** 给定  $K$  张简单彩票  $L_k = (p_1^k, \dots, p_n^k)$  ( $k = 1, \dots, K$ )，以及一组选择概率  $\{\alpha_k\}_{k=1}^K$  ( $\alpha_k \geq 0$ ,  $\sum_{k=1}^K \alpha_k = 1$ )，先以概率  $\alpha_k$  选中彩票  $L_k$ ，再按  $L_k$  中的概率决定最终结果，则所得随机机制称为**复合彩票**。

直观地看，复合彩票是“彩票的彩票”：先随机选一张简单彩票，再由该彩票的概率产生最终结果。这种结构便于刻画更复杂的风险情形，常用于多阶段博弈与复杂金融产品的定价建模。

**独立性公理 (independence axiom)** 是期望效用理论的核心公理之一。其含义是：偏好关系应当具有与无关选项“独立”的一致性与线性——即在两个彩票之间的比较中，将二者按同一权重与任意第三个彩票混合 (convex combination) 后，原先的偏好顺序不应因该第三彩票的加入而改变。换言之，偏好不应受与当前选择无关的外部选项影响。

**命题 2.2 (独立性公理):** 在彩票集合上，若偏好关系满足：对任意三张彩票  $A, B, C$  与任意  $\alpha \in [0, 1]$ ，都有

$$A \succeq B \iff \alpha A + (1 - \alpha)C \succeq \alpha B + (1 - \alpha)C,$$

则称该偏好关系满足**独立性公理**。

独立性公理保证偏好的一致性：决策者在加入不相关选项（如  $C$ ）时，其原有偏好不

变。这一特征体现了理性选择的内在一致性和可预测性。在期望效用理论中，独立性公理使复合彩票的效用可简化为简单彩票效用的加权和，进而利用数学工具进行优化决策。

期望效用定理是期望效用理论的核心。该定理说明：在适当假设下，决策者的偏好可以由一个数学函数表示，此函数称为效用函数。

**命题 2.3 (期望效用):** 若定义在彩票空间  $L$  上的偏好关系  $\succeq$  同时满足**理性、连续性与独立性公理**，则该偏好可由**期望效用**表示。具体地，设结果集合为  $\{x_1, \dots, x_n\}$ ，对任意两张彩票  $L = (p_1, \dots, p_n)$  与  $L' = (p'_1, \dots, p'_n)$ ，存在效用函数  $u: X \rightarrow \mathbb{R}$  使得

$$L \succeq L' \iff \sum_{i=1}^n p_i u(x_i) \geq \sum_{i=1}^n p'_i u(x_i).$$

通过期望效用理论，决策者可以将复杂的抉择问题化简为对数值效用的计算与比较。这不仅有助于理解并预测个体在风险情境下如何做出选择，也是设计经济政策与评估金融产品的基础工具。此外，该方法能反映个体的风险态度——无论风险厌恶、风险中性还是风险偏好——都能在效用函数的形态中得到体现。下一节将介绍风险厌恶的相关概念。

期望效用理论自约翰·冯·诺依曼 (John von Neumann) 与奥斯卡·摩根斯特恩 (Oskar Morgenstern) 提出以来，已成为决策科学、经济学与心理学中用于理解与建模个体在风险环境下如何选择的主导理论。然而，它在许多方面也遭遇了挑战。首先，实验经济学与行为经济学的研究发现，人们在实际决策中常常违背期望效用理论，经典例子包括阿莱悖论 (Allais

**例 2.1:** 阿莱悖论 (*Allais Paradox*)，由法国经济学家莫里斯·阿莱 (Maurice Allais) 于 1953 年提出。

悖论通过以下两组偏好说明：

**第一组偏好：**

- 1A: 以 100% 的概率获得 100 万美元；
- 1B: 以 89% 的概率获得 100 万美元、10% 的概率获得 500 万美元、1% 的概率什么也得不到。

**第二组偏好：**

- 2A: 以 11% 的概率获得 100 万美元、89% 的概率什么也得不到；
- 2B: 以 10% 的概率获得 500 万美元、90% 的概率什么也得不到。

实验中，多数参与者在第一组选择中偏好 1A (确定获得 100 万美元)。若用期望效用表示，则

$$1.00 u(100 \text{ 万美元}) > 0.89 u(100 \text{ 万美元}) + 0.10 u(500 \text{ 万美元}) + 0.01 u(0 \text{ 美元}). \quad (2.1)$$

而在第二组中，多数参与者偏好 2B (高额但不确定的回报)，于是

$$0.89 u(0 \text{ 美元}) + 0.11 u(100 \text{ 万美元}) < 0.90 u(0 \text{ 美元}) + 0.10 u(500 \text{ 万美元}). \quad (2.2)$$

由式 (2.2) 可得

$$0.11 u(100 \text{ 万}) < 0.01 u(0) + 0.10 u(500 \text{ 万}),$$

进而

$$1.00 u(100 \text{ 万}) - 0.89 u(100 \text{ 万}) < 0.01 u(0) + 0.10 u(500 \text{ 万}),$$

从而推出

$$1.00 u(100 \text{ 万}) < 0.89 u(100 \text{ 万}) + 0.10 u(500 \text{ 万}) + 0.01 u(0), \quad (2.3)$$

这与 (2.1) 矛盾。

显然，阿莱悖论展示的偏好结果违背了独立性公理：参与者的偏好受到与当前选择无关的外部结果影响。按独立性公理，若某人偏好  $1B$  于  $1A$ ，则应当偏好  $2B$  于  $2A$ ，因为两组实验的概率结构相似；但事实表明，人们对确定性结果存在非线性偏好，而这是在标准期望效用理论中不被支持。

**例 2.2：** 埃尔斯伯格悖论 (*Ellsberg Paradox*) 通过实验展示：人们对已知风险（概率可计算）与未知风险（概率不可计算或不确定）的反应不同。设有两个盒子，每个盒子装有红球与黑球：第一个盒子中确切有 50 个红球与 50 个黑球；第二个盒子共有 100 个球，但红黑球比例未知。参与者需从某一盒子中抽球，若抽到红球即可获奖。

多数人选择从第一个盒子抽取，因为其风险是“已知”的；当规则改为抽到黑球获奖时，多数人仍然偏好第一个盒子。该行为体现了对未知风险的回避，即使这可能违背最大化期望效用的原则。

通过提出效用概念，并指出人们在决策时追求的是最大化期望效用而非仅仅最大化期望收益，伯努利解释了人们在高风险博弈中表现出的风险厌恶。

看待圣彼得堡悖论的另一角度是考察在资本约束下的破产问题：例如，若连续 99 次为正面、随后一次为反面，参与者是否真的愿意支付  $2^{100}$  元的入场费？

考虑一个押大押小的赌局。

**例 2.3：** 假设你有 1 元，并且每次都押注 1 元在“大”上，直到资金耗尽。设  $p$  为“大”出现的概率，记  $P_{j,k}$  为资金从  $j$  元增长到  $k$  元的概率。易得

$$P_{1,0} = (1-p) + p P_{2,0}.$$

由对称性  $P_{2,1} = P_{1,0}$ ；由独立性  $P_{2,0} = P_{2,1}P_{1,0}$ 。由此可得

$$P_{2,0} = P_{1,0}^2, \quad P_{1,0} = 1 - p + p P_{1,0}^2.$$

求解二次方程得

$$P_{1,0} = 1, \quad P_{1,0} = \frac{1-p}{p}.$$

当  $p \leq \frac{1}{2}$  时取前者；否则取后者。若在赌场中  $p < \frac{1}{2}$ ，则你最终将以概率 1 破产。

风险厌恶是经济学和金融学中的一个概念，描述了个体或组织在面对不确定性时倾向于规避风险或选择风险较低选项的行为。风险厌恶者更愿意接受一个较低但是确定的收益，而不愿承担可能带来更高收益但伴随更高不确定性的风险。

那么风险厌恶程度如何度量呢？首先，确定性等值 (Certainty Equivalent, CE)。确定性等值是指一个确定的结果，其效用等同于一个不确定结果（如彩票或投资）的期望效用。换言之，确定性等值是人们愿意接受的、与参与一个概率性风险项目等效用的确定金额。例

如，如果一个投资者面对一个期望收益为 100 元的彩票，该彩票有 50

确定性等值的计算基于个体的效用函数。假设个体的效用函数是  $u(x)$ ，并且一个随机收益  $X$  的期望效用是  $E[u(X)]$ 。那么确定性等值  $CE$  是满足以下条件的金额  $x$ :

$$u(CE) = E[u(X)].$$

确定性等值与风险厌恶程度之间存在以下关系：如果个体是风险中性的，他们的确定性等值将等于风险项目的期望值。在这种情况下，效用函数是线性的。大多数投资者都是风险厌恶的，这意味着他们的确定性等值通常低于风险项目的期望值。这是因为风险厌恶者的效用函数是凹的，表示他们对潜在损失的重视程度超过了对同等收益的重视程度。如果个体是风险偏好的，他们的确定性等值会高于风险项目的期望值，因为他们愿意支付额外的金额来享受参与风险的“刺激感”。这种情况下，效用函数是凸的。

除了确定性等值之外，风险厌恶程度还可通过效用函数的曲率与风险厌恶系数进行度量。

令效用函数为  $u(y)$ ，其中  $y$  表示消费或财富。若  $u(y)$  为凹函数（即  $u''(y) < 0$ ），则个体风险厌恶；凹性越强，通常表示风险厌恶程度越高。

**定义 2.17 (绝对风险厌恶系数):** (*Coefficient of Absolute Risk Aversion*; 亦称 Arrow-Pratt 绝对风险厌恶度量，简称 *ARA*)

$$R_A(y) = -\frac{u''(y)}{u'(y)}.$$

其中  $u'(y)$  与  $u''(y)$  分别为一阶、二阶导数。 $R_A(y)$  描述在财富（或消费）为  $y$  时的风险厌恶局部强度；数值越大，表示风险厌恶程度越高。

为了更好地理解绝对风险厌恶系数，可以考虑以下直观情形。对一位拥有财富水平  $y$  的投资者而言，存在这样一个投资机会：该投资有  $p$  的概率赢得数额为  $\delta$  的货币，同时有  $1-p$  的概率输掉同样数额的货币。假设  $\delta$  是一个很小的数值。显然，投资者是否选择参与这项投资取决于  $p$  的大小； $p$  越大，越多的投资者愿意参加。特别是当  $p=1$  时，投资者可以确保赢得  $\delta$ ，此时所有人都会选择参与。相反，如果  $p$  很小，则参与者会相应减少。

显然，对于那些风险厌恶程度较高的投资者来说，需要更高的  $p$ （赢钱概率）才足以吸引他们参与这项投资。我们定义  $p^*$  为使投资者在参与和不参与投资之间处于完全无差异状态的临界概率值。 $p^*$  可以看作是衡量投资者风险厌恶程度的指标。接下来，我们将详细阐述如何将  $p^*$  表示为投资者偏好的函数。

按照  $p^*$  的定义，可得：

$$u(y) = p^* \cdot u(y + \delta) + (1 - p^*) \cdot u(y - \delta).$$

将  $u(y + \delta)$  与  $u(y - \delta)$  在  $y$  处进行泰勒展开：

$$\begin{aligned} u(y + \delta) &= u(y) + \delta u'(y) + \frac{\delta^2}{2} u''(y) + o(\delta^2), \\ u(y - \delta) &= u(y) - \delta u'(y) + \frac{\delta^2}{2} u''(y) + o(\delta^2), \end{aligned}$$

其中  $o(\delta^2)$  为高阶余项，在  $\delta$  很小的情况下可以忽略。代入上述展开式，简化得：

$$u(y) = p^* \left[ u(y) + \delta u'(y) + \frac{\delta^2}{2} u''(y) \right] + (1 - p^*) \left[ u(y) - \delta u'(y) + \frac{\delta^2}{2} u''(y) \right].$$

整理后可得：

$$0 = (2p^* - 1)\delta u'(y) + \frac{\delta^2}{2} u''(y),$$

从中解出：

$$p^* = \frac{1}{2} + \frac{\delta}{4} \left( -\frac{u''(y)}{u'(y)} \right).$$

我们定义：

$$R_A(y) \equiv -\frac{u''(y)}{u'(y)}.$$

$R_A(y)$  为绝对风险厌恶系数，该系数由 Pratt (1964) 与 Arrow (1965) 最先提出。绝对风险厌恶系数  $R_A(y)$  越大，为了吸引投资者参与投资，就需要更高的获胜概率。

**定义 2.18：** 相对风险厌恶系数 (*Relative Risk Aversion*, 简称 *RRA*)，亦称 *Arrow-Pratt-De Finetti* 相对风险厌恶度量 (*Arrow-Pratt-De Finetti measure of relative risk aversion*)，定义为

$$R_R(y) = -\frac{y u''(y)}{u'(y)}.$$

值得注意的是，虽然我们经常假设投资者是风险厌恶的，但实际上投资者往往是损失厌恶 (loss aversion) 的。除了阿莱悖论 (Allais Paradox) 与埃尔斯伯格悖论 (Ellsberg Paradox) 外，另一个对传统期望效用理论的重要补充来自心理学和行为经济学的前景理论 (Prospect Theory)，该理论由丹尼尔·卡尼曼 (Daniel Kahneman) 和阿莫斯·特沃斯基 (Amos Tversky) 于 1979 年提出。前景理论指出，人们在做出决策时，并非基于绝对的财富水平，而是依据相对于某个参照点 (通常是当前状态或期望) 的相对变化。这意味着同一收益或损失在不同参照点下可能引致截然不同的效用感受。前景理论提出了一个关键的价值函数，该函数在参照点处不对称：对损失比对收益更加敏感，通常呈“S”形 (见图 2.2)。其中，

- **凹形区域 (收益)：**当收益为正时，价值函数呈凹形 (递增但递增速度递减)，表明效用随收益增加而上升，但边际效用递减；
- **凸形区域 (损失)：**当收益为负 (损失) 时，价值函数呈凸形 (递减但递减速度递增)，表明损失带来的负效用大于同等金额收益带来的正效用。

对一位拥有财富水平  $y$  的投资者，考虑一项按比例的小赌注：以概率  $p$  赢得  $cy$ ，以概率  $1-p$  损失  $cy$ ，其中  $c$  很小。类似之前的方法，定义使其在“参与/不参与”之间完全无差异的临界概率  $p^*$ ：

$$u(y) = p^* u(y + cy) + (1 - p^*) u(y - cy).$$

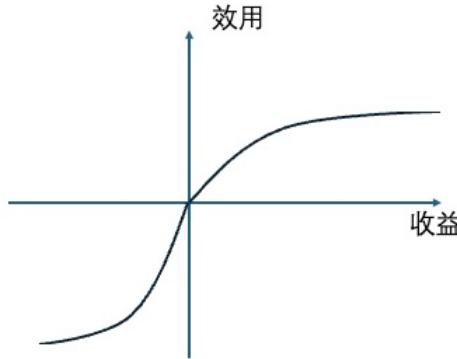


图 2.2: 展望理论

在  $y$  处对  $u(y \pm cy)$  进行泰勒展开，并忽略二阶以上高阶项，可得

$$\begin{aligned} u(y + cy) &= u(y) + cy u'(y) + \frac{c^2}{2} y^2 u''(y), \\ u(y - cy) &= u(y) - cy u'(y) + \frac{c^2}{2} y^2 u''(y). \end{aligned}$$

代入并化简：

$$\begin{aligned} u(y) &= p^* \left[ u(y) + cy u'(y) + \frac{c^2}{2} y^2 u''(y) \right] + (1 - p^*) \left[ u(y) - cy u'(y) + \frac{c^2}{2} y^2 u''(y) \right], \\ 0 &= (2p^* - 1) c u'(y) y + \frac{c^2}{2} u''(y) y^2, \quad p^* = \frac{1}{2} + \frac{c}{4} \left( -\frac{y u''(y)}{u'(y)} \right). \end{aligned}$$

定义相对风险厌恶系数

$$R_R(y) \equiv -\frac{y u''(y)}{u'(y)}.$$

$R_R(y)$  是衡量投资者风险厌恶程度的重要指标。

不难看出，绝对风险厌恶系数衡量的是个体在每单位财富增加或减少时对风险的厌恶程度，它反映了个体对小额风险的敏感性；而相对风险厌恶系数则表明个体在其总财富的某一比例发生变化时的风险厌恶程度。因此，绝对风险厌恶系数可以帮助金融顾问或决策者了解投资者对风险的敏感度，从而设计适合其风险承受能力的投资组合；相对风险厌恶系数对于理解大规模财富变化下的投资行为尤为重要，适用于评估大额投资或彩票等极端情况。从应用范围来看，绝对风险厌恶系数更适用于日常小额投资或保险决策，而相对风险厌恶系数更适合用于分析涉及较大比例财富变动的决策，如购买房产或其他大宗商品。虽然两者都是重要的经济分析工具，但相对风险厌恶系数由于能够涵盖更广泛的经济行为和决策背景，通常被认为在理论和实践中更为关键。

另外，值得注意的是，绝对风险厌恶系数和相对风险厌恶系数均是财富水平  $y$  的函数；在不同的财富水平上，投资者的风险厌恶程度可能不同。当然，风险厌恶系数也依赖于效用函数。

下面我们介绍几个常用的效用函数，它们因具有不同的风险厌恶特性而在模型分析中得到广泛应用：

**定义 2.19 (常绝对风险厌恶型效用函数 (CARA) ):** 效用函数为  $u(y) = -e^{-\alpha y}$ , 其中  $y$  为财富水平,  $\alpha > 0$  表示风险厌恶程度。对应的绝对风险厌恶系数为  $R_A(y) = \alpha$ 。在该模型中, 无论财富水平如何, 绝对风险厌恶系数保持不变。

**定义 2.20 (常相对风险厌恶型效用函数 (CRRA) ):** 效用函数通常表示为  $u(y) = \frac{y^{1-\gamma}-1}{1-\gamma}$ , 其中  $\gamma$  为相对风险厌恶系数。当  $\gamma = 1$  时, CRRA 退化为对数效用函数, 即  $u(y) = \ln y$  (自然对数)。

**定义 2.21 (双曲绝对风险厌恶型效用函数 (HARA) ):** 绝对风险厌恶系数为  $R_A(y) = \frac{1}{ay+b}$ 。解相应微分方程得效用函数  $u(y) = \frac{(y-c_s)^{1-\gamma}}{1-\gamma}$ , 其中  $\gamma = 1/a$ ,  $c_s = -b/a$ 。当  $a = 0$  时, HARA 退化为 CARA; 当  $b = 0$  时, HARA 退化为 CRRA。

**定义 2.22 (线性效用函数 (风险中性) ):** 效用函数为  $u(y) = \alpha y$  ( $\alpha > 0$ )。其特点是无论财富水平如何变化, 绝对风险厌恶系数和相对风险厌恶系数均为 0, 表示对风险持中立态度。

## 2.4.2 基于均值-方差框架的投资组合模型

Markowitz (1959) 首次提出现代投资组合理论, 为风险与收益的权衡关系奠定了量化基础, 因而被誉为现代投资组合理论之父。其思想启发了 Sharpe (1964) 与 Lintner (1965), 二人进一步发展出资本资产定价模型 (CAPM), 该模型已成为现代金融市场价格理论的核心。

在统计学与金融分析中, 相关系数  $\rho_{A,B}$  用于量化两个变量之间线性关系的强弱, 定义为

$$\rho_{A,B} = \text{corr}(A, B) = \frac{\text{Cov}(A, B)}{\sigma_A \sigma_B} = \frac{E[(A - \mu_A)(B - \mu_B)]}{\sigma_A \sigma_B},$$

其中  $\text{Cov}(A, B)$  为协方差,  $\sigma_A, \sigma_B$  为标准差,  $\mu_A, \mu_B$  为均值。

在资产配置中, 投资者常以相关系数评估不同资产间的关联性, 并据此进行分散化与风险管理。例如, 在理想情形  $\rho_{A,B} = -1$  时, 若两资产的权重按

$$w^* = \frac{\sigma_B}{\sigma_A + \sigma_B}, \quad 1 - w^* = \frac{\sigma_A}{\sigma_A + \sigma_B},$$

配置, 则组合方差可为零 (当且仅当  $\rho_{A,B} = -1$  且权重满足上式)。只有在  $\sigma_A = \sigma_B$  时,  $w^* = 1/2$  才对应 “50%-50%” 的无风险对冲。

在金融市场中, 投资者常用相关系数评估不同证券或资产之间的关联性, 并据此进行资产配置与风险管理。例如, 为降低组合风险, 投资者会寻找相关系数较低或负相关的资产进行组合。在理想情形, 若  $A$  与  $B$  完全负相关, 因二者波动相互抵消, 按适当权重配置即可实现完全对冲; 只有当  $\sigma_A = \sigma_B$  时, 50%-50% 的配置才对应无风险对冲。

在金融投资领域, 投资组合的风险管理是至关重要的环节。一个投资组合的总体风险受多种因素影响: 各个资产的个别风险、各资产在组合中的权重, 以及资产收益率之间的相关性。理解这些因素对于制定有效的投资策略与风险控制措施至关重要。

对于由两种资产组成的投资组合, 其总体风险可用组合标准差衡量:

$$\sigma_p = \sqrt{\sigma_A^2 w_A^2 + \sigma_B^2 w_B^2 + 2w_A w_B \rho_{A,B} \sigma_A \sigma_B},$$

其中  $\sigma_A, \sigma_B$  为资产  $A, B$  的标准差,  $w_A, w_B$  为它们在组合中的权重,  $\rho_{A,B}$  为两资产的收益率相关系数。

由上式可见, 相关性  $\rho_{A,B}$  在风险管理中至关重要: 若  $\rho_{A,B} = 1$ , 两资产风险为线性加权; 若  $\rho_{A,B} < 1$ , 尤其为负相关时, 组合风险低于单一资产风险的加权平均, 体现出分散化效果。因此, 投资者在构建组合时倾向纳入相关性低或负相关资产, 以有效地分散风险。这一分散化原则是现代投资组合理论的核心内容之一, 广泛应用于资产管理与金融策略制定。

古人虽未建立如现代般精密的投资组合理论体系, 但在长期实践中早已蕴含风险管理智慧。例如“狡兔三窟”: 狡兔为保性命, 营造多个藏身之所, 以防一处被破坏后无处可逃, 体现了分散风险的朴素思维, 正如现代组合中纳入相关性低的资产, 避免“把鸡蛋放在一个篮子里”。当市场环境复杂多变时, 若单一资产面临风险, 其他相关性较低的资产或能保持稳定, 从而维护组合整体稳健。

再看古代的仓储管理, 以常平仓为例: 丰年时政府收购粮食储存, 灾年则开仓放粮以稳定粮价。其逻辑是对粮食供应风险的管理, 通过跨期资源调配, 降低自然灾害与收成波动带来的民生与经济风险。本质上, 这与现代投资组合通过资产配置分散风险的做法异曲同工——在不同条件下运用多种策略应对风险, 确保整体平稳运行。这些古老智慧与现代投资组合理论中的分散化原则相呼应, 值得在金融实践中汲取与传承。

随着资产数量增加, 马科维茨投资组合的计算量也随之大幅提升。若资产组合由三种资产构成, 需要计算 3 个相关系数:  $\rho_{A,B}$ 、 $\rho_{B,C}$  与  $\rho_{A,C}$ 。此时组合风险(标准差)为

$$\sigma_p = \sqrt{\sigma_A^2 w_A^2 + \sigma_B^2 w_B^2 + \sigma_C^2 w_C^2 + 2w_A w_B \rho_{A,B} \sigma_A \sigma_B + 2w_B w_C \rho_{B,C} \sigma_B \sigma_C + 2w_A w_C \rho_{A,C} \sigma_A \sigma_C}.$$

接着我们将考虑  $n$  个资产的一般情形, 分为不存在无风险资产和存在无风险资产两种情况。

#### 2.4.2.1 不存在无风险资产的情形

首先, 我们讨论不存在无风险资产的情形。事实上, 在严格意义的金融市场中并不存在完全无风险的资产; 在极度不稳定或高风险的环境下, 更难以找到可视为无风险的资产。因此, 一些理论或实证研究会选择不引入无风险资产, 以更贴近真实风险环境。

设有  $n$  个资产, 其随机回报为  $R_1, \dots, R_n$ , 且  $E(R_j) = \mu_j$ 、 $\text{Var}(R_j) = \sigma_{jj}$ 、 $\text{Cov}(R_j, R_k) = \sigma_{jk}$ 。令

$$R = (R_1, \dots, R_n)'$$

为  $n \times 1$  的回报向量, 其期望向量与协方差矩阵为

$$E(R) = \boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix}, \quad \text{Var}(R) = E[(R - \boldsymbol{\mu})(R - \boldsymbol{\mu})'] = \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1n} \\ \vdots & \ddots & \vdots \\ \sigma_{n1} & \cdots & \sigma_{nn} \end{pmatrix}.$$

设  $R(\mathbf{w})$  为随机投资组合回报:

$$R(\mathbf{w}) = \sum_{j=1}^n w_j R_j = \mathbf{w}' R,$$

其中  $\mathbf{w} = (w_1, \dots, w_n)'$  为权重向量，且  $\sum_{j=1}^n w_j = 1$ 。投资组合的期望回报与方差分别为

$$\mu_{\mathbf{w}} = E(R(\mathbf{w})) = \mathbf{w}'\boldsymbol{\mu} = \sum_{j=1}^n w_j \mu_j, \quad \sigma_{\mathbf{w}}^2 = \text{Var}(R(\mathbf{w})) = \mathbf{w}'\boldsymbol{\Sigma}\mathbf{w} = \sum_{j=1}^n \sum_{k=1}^n w_j w_k \sigma_{jk}.$$

**定义 2.23:** 最小方差组合  $\mathbf{w}$  是如下优化问题的解：

$$\min_{\mathbf{w} \in \mathbb{R}^n} \mathbf{w}'\boldsymbol{\Sigma}\mathbf{w} \quad \text{subject to} \quad \mathbf{w}'\mathbf{1} = 1,$$

其中  $\mathbf{1} = (1, \dots, 1)'$  为全 1 向量。

这是一个经典的约束优化问题，构建拉格朗日函数（目标函数加上拉格朗日乘子  $\lambda$  乘以约束）：

$$\mathcal{L}(\mathbf{w}, \lambda) = \frac{1}{2}\mathbf{w}'\boldsymbol{\Sigma}\mathbf{w} + \lambda(1 - \mathbf{w}'\mathbf{1}).$$

一阶条件为

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \boldsymbol{\Sigma}\mathbf{w} - \lambda\mathbf{1} = 0 \Rightarrow \mathbf{w} = \lambda\boldsymbol{\Sigma}^{-1}\mathbf{1}.$$

左乘  $\mathbf{1}'$  并利用约束可得

$$\mathbf{1}'\mathbf{w} = \lambda\mathbf{1}'\boldsymbol{\Sigma}^{-1}\mathbf{1} \Rightarrow \lambda = \frac{1}{\mathbf{1}'\boldsymbol{\Sigma}^{-1}\mathbf{1}},$$

因此全局最小方差组合权重为

$$\mathbf{w}_{\text{GMV}} = \frac{\boldsymbol{\Sigma}^{-1}\mathbf{1}}{\mathbf{1}'\boldsymbol{\Sigma}^{-1}\mathbf{1}}.$$

其期望与方差为

$$\mu_{\text{GMV}} = \frac{\mathbf{1}'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}}{\mathbf{1}'\boldsymbol{\Sigma}^{-1}\mathbf{1}}, \quad \sigma_{\text{GMV}}^2 = \frac{1}{\mathbf{1}'\boldsymbol{\Sigma}^{-1}\mathbf{1}}.$$

最小方差组合虽使风险最小，但可能无法提供足够的期望回报。因此，考虑如下优化情形：在给定目标回报  $m$  的约束下最小化方差。

**定义 2.24:** 保持  $m$  的回报水平使方差最小化，需求解

$$\min_{\mathbf{w} \in \mathbb{R}^n} \mathbf{w}'\boldsymbol{\Sigma}\mathbf{w}$$

并满足约束： $\mathbf{w}'\mathbf{i} = 1$  与  $\mathbf{w}'\boldsymbol{\mu} \geq m$ 。

拉格朗日函数为

$$\mathcal{L}(\mathbf{w}, \lambda, \gamma) = \frac{1}{2}\mathbf{w}'\boldsymbol{\Sigma}\mathbf{w} + \lambda(1 - \mathbf{w}'\mathbf{i}) + \gamma(m - \mathbf{w}'\boldsymbol{\mu}),$$

其中  $\lambda, \gamma \in \mathbb{R}$  为拉格朗日乘子。关于  $\mathbf{w}$  的一阶条件：

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \boldsymbol{\Sigma}\mathbf{w} - \lambda\mathbf{i} - \gamma\boldsymbol{\mu} = 0,$$

得到

$$\mathbf{w}_{\text{opt}} = \lambda(m)\boldsymbol{\Sigma}^{-1}\mathbf{i} + \gamma(m)\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}. \quad (2.4)$$

由两个约束

$$1 = i' w_{\text{opt}} = \lambda i' \Sigma^{-1} i + \gamma i' \Sigma^{-1} \mu, \quad m = \mu' w_{\text{opt}} = \lambda \mu' \Sigma^{-1} i + \gamma \mu' \Sigma^{-1} \mu,$$

可得关于  $\lambda, \gamma$  的方程组, 解得

$$\lambda = \frac{C - Bm}{\Delta}, \quad \gamma = \frac{Am - B}{\Delta},$$

其中

$$A = i' \Sigma^{-1} i, \quad B = i' \Sigma^{-1} \mu, \quad C = \mu' \Sigma^{-1} \mu, \quad \Delta = AC - B^2,$$

且  $\Delta > 0$ 。于是该组合的期望回报为  $m$ , 方差为

$$\sigma_{\text{opt}}^2(m) = \frac{Am^2 - 2Bm + C}{\Delta}.$$

不难看出  $\sigma_{\text{opt}}^2(m)$  是  $m$  的二次函数。集合  $\{w_{\text{opt}}(m) : m \geq 0\}$  称为均值-方差有效集 (mean-variance efficient set)。若以均值为横轴、标准差为纵轴, 则得到如下双曲线:

$$\left\{ (m, \sigma_{\text{opt}}(m)) \mid \sigma_{\text{opt}}(m) = \sqrt{\frac{Am^2 - 2Bm + C}{\Delta}}, m \geq 0 \right\}.$$

在上述双曲线方程中, 不同的预期回报 (收益)  $m$  可以对应不同的  $\sigma_{\text{opt}}(m)$  (风险)。在均值-方差投资组合理论中, 有效前沿 (Efficient Frontier) 是由那些在给定风险水平下提供最高预期回报的投资组合构成的集合。换言之, 有效前沿通常位于双曲线的右侧或上侧, 具体位置取决于参数  $A, B, C$  和  $\Delta$  的取值。有效前沿一般呈上凸曲线: 高风险通常对应高回报, 而位于有效前沿上的投资组合在每一风险水平上都提供可能的最高回报。双曲线的形状还表明, 随着预期回报  $m$  的增加, 标准差  $\sigma_{\text{opt}}(m)$  也随之增加, 但不是线性增加, 而是以减速的方式增加 (即边际风险增加速度逐渐下降)。这与现实相符——为了获取更高的额外回报, 所需承担的额外风险也会逐步上升。

此外, 在投资组合管理中通常有两类基本优化问题: 其一, 在给定最大风险水平的约束下最大化投资组合的平均回报; 其二, 在保持固定平均回报 (或预期回报) 的条件下最小化投资组合的风险。两种方法虽然出发点不同, 但在数学上是对偶的, 通常会产生相同的有效前沿。

接着, 我们介绍两基金分离定理 (Two-Fund Separation Theorem)。

**定理 2.1 (两基金分离定理):** 在均值-方差分析框架下, 任何一个位于均值-方差有效集 (Mean-variance efficient set) 上的投资组合, 都可以表示为两个 (任意两个) 有效投资组合的线性组合。

换言之, 只需选择两个有效的基金或资产组合, 投资者便可通过调整二者的权重比例, 构建出任意其他有效投资组合。

**证明:** 由式 (2.4) 可知, 最小方差投资组合在  $\mathbb{R}^n$  空间中构成一条直线。设给定最优投资组合的预期回报为  $m$ , 并取任意两个不同实数  $m_1, m_2$ 。存在唯一  $\alpha \in \mathbb{R}$  使

$m = \alpha m_1 + (1 - \alpha)m_2$ 。由  $\lambda(m), \gamma(m)$  的表达式可得

$$\begin{aligned}\lambda(m) &= \alpha \lambda(m_1) + (1 - \alpha) \lambda(m_2), \\ \gamma(m) &= \alpha \gamma(m_1) + (1 - \alpha) \gamma(m_2), \\ w(m) &= \alpha w(m_1) + (1 - \alpha) w(m_2).\end{aligned}$$

因此，任一最优投资组合的权重向量  $w(m)$  可由两个有效投资组合的权重向量线性组合得到。

#### 2.4.2.2 存在无风险资产的情形

这里我们讨论存在无风险资产的情形。经典投资理论（如马科维茨的投资组合选择理论和资本资产定价模型（CAPM））中，引入无风险资产可以帮助简化模型，使理论分析更为清晰。无风险资产提供基准利率，允许投资者在有风险与无风险资产之间进行权衡，从而得到效用最大化的投资组合。在现实世界中，虽然严格意义上不存在完全无风险的资产，但一些资产（例如政府债券等）可视为接近无风险资产，因为其违约风险极低。因此，在实际应用中，这些资产经常被用作无风险资产的代理。

#### 2.4.3 资本资产定价模型

Markowitz (1959) 首次提出现代投资组合理论，为风险与收益之间的权衡提供了量化基础，因而被誉为现代资产组合理论之父。其思想启发了 Sharpe (1964) 与 Lintner (1965)，两人在此基础上进一步发展了资本资产定价模型（Capital Asset Pricing Model，简称 CAPM）。CAPM 是金融经济学中的核心模型，主要用于刻画投资的风险与预期收益之间的关系。该模型基于市场组合的概念；假设所有投资者都选择同一市场组合，该组合包含所有可投资资产，并按各自在市场中的价值比例加权。CAPM 认为，任何资产的预期收益率可通过其与市场组合的相关性来预测。具体而言，CAPM 指出：资产的预期收益率等于无风险利率，加上该资产相对于市场组合的系统性风险（以贝塔系数衡量）与市场风险溢价的乘积。贝塔系数衡量单项资产相对于整个市场的风险程度。

CAPM 强调一个重要原则：在市场均衡下，并非所有风险都会得到回报。市场中的可分散风险（即特定于单项资产的风险）不会带来额外预期收益；只有不可分散的、市场整体风险才会通过资产的贝塔系数影响该资产的预期收益率。

##### 2.4.3.1 夏普-林特纳版本的 CAPM 模型

首先，我们考虑存在无风险资产的情形，即 Sharpe (1964) 与 Lintner (1965) 提出的资本资产定价模型。在该模型中，资产  $i$  的预期收益率可以用下式表示：

$$E[R_i] = R_f + \beta_{im} (E[R_m] - R_f), \quad (2.5)$$

其中  $\beta_{im}$  为资产  $i$  与市场组合收益率之间的协方差与市场组合收益率方差之比：

$$\beta_{im} = \frac{\text{Cov}(R_i, R_m)}{\text{Var}(R_m)}.$$

在上式中， $R_m$  表示市场组合的收益率， $R_f$  为无风险资产的收益率。该公式表明，资产的预期收益率由无风险利率与该资产的市场风险溢价（其贝塔系数乘以市场组合的超额收益

率) 构成。

夏普-林特纳版本的 CAPM 亦可用超额收益率最简洁地表达。设  $Z_i = R_i - R_f$  为第  $i$  项资产的超额收益率，则

$$E[Z_i] = \beta_{im} E[Z_m], \quad (2.6)$$

$$\beta_{im} = \frac{\text{Cov}[Z_i, Z_m]}{\text{Var}[Z_m]}, \quad (2.7)$$

其中  $Z_m$  为市场资产组合的超额收益率。

由于无风险收益率被假定为非随机量，式 (2.5) 与上式是等价的。然而，在实证研究中，用作无风险收益率的代理变量通常是随机的，因此两个  $\beta$  的估计可能并不完全一致。多数与夏普-林特纳版本相关的实证研究采用超额收益率 (式 (2.6))，其更符合经济直觉：投资者关注的是相对于无风险投资的回报，而非绝对回报。

#### 2.4.3.2 布莱克版本的 CAPM (零贝塔 CAPM)

我们讨论不含无风险资产的 CAPM。Black (1972) 提出：当不存在无风险资产时，可用零贝塔投资组合替代无风险资产；该组合的预期收益率与市场波动无关。传统 CAPM 假定存在无风险资产并以其收益率为基准，但在现实中严格意义的“无风险”资产并不存在（即便是通常被视为无风险的政府债券，也会受到通胀、利率变动、汇率波动等因素影响）。

根据 Black (1972)：前沿投资组合的凸组合仍在前沿上；有效投资组合的集合是凸集。这并不意味着有效前沿在  $(\mu, \sigma)$  空间中是凸的，而是指任一有效投资组合的凸组合仍位于有效前沿。若每位投资者都持有某个有效组合，则市场投资组合（个体组合的加权平均）也是有效的。对于任何位于有效前沿上的投资组合  $p$ （除最小方差组合外），存在唯一的位于有效前沿上的投资组合  $Z_p$ ，使得  $\text{Cov}(R_{Z_p}, R_p) = 0$ 。 $Z_p$  被称为相对于  $p$  的零协方差组合。若  $p$  有效，则  $Z_p$  无效；反之亦然。

**定理 2.2：** 对任意资产或投资组合  $i$  和任一位于有效前沿上的投资组合  $p$ ，有

$$E(R_i) = E(R_{Z_p}) + \beta_{ip}(E(R_p) - E(R_{Z_p})), \quad \beta_{ip} = \frac{\text{Cov}(R_i, R_p)}{\text{Var}(R_p)}.$$

取  $p = M$  (市场组合)，记相对于市场组合  $M$  的零贝塔组合为  $Z$  (文献中常写作  $R_0$ )，则

$$E(R_i) = E(R_Z) + \beta_i(E(R_M) - E(R_Z)),$$

等价地，

$$E(R_i) = \beta_i E(R_M) + (1 - \beta_i) E(R_Z).$$

还可写成

$$E(R_i) = \alpha_i + \beta_i E(R_M), \quad \alpha_i = (1 - \beta_i) E(R_Z).$$

零贝塔 CAPM 在缺乏无风险资产的环境下更贴近现实，但其实证实现较传统 CAPM 更复杂，因为零贝塔组合收益率是未观测变量。尽管资本资产定价模型被广泛用于资产定价与投资决策，但其理论基础依赖期望效用框架；围绕期望效用的质疑（如阿莱悖论、前景理论等）也对 CAPM 的适用性提出了挑战。

#### 2.4.4 套利定价理论

套利定价理论 (Arbitrage Pricing Theory, 简称 APT) 由经济学家斯蒂芬·罗斯 (Stephen Ross) 于 1976 年提出 (Ross 1976)。APT 的核心思想是：如果某资产价格偏离由多个宏观经济因素共同决定的理论价格，那么通过套利将使价格回归其理论价值。APT 假设市场上存在无风险套利机会，即投资者可以通过构建零净投资组合获得无风险利润。在实际应用中，APT 通过构建多因素模型来估计资产收益，因素既可以是宏观经济变量，也可以是其他任何影响资产收益的因素。与 CAPM 不同，APT 并非仅基于市场风险单一因素，而是认为资产的预期收益受多种宏观经济因素影响，如通货膨胀率、利率、工业产出变化等。显然，APT 提供了一种更为灵活的方法来分析和预测资产收益，尤其适用于受多种经济因素影响较大的资产。APT 在投资组合管理、风险评估和资产定价方面具有广泛应用，是现代金融理论与实践中不可或缺的一部分。

具体来讲，资本资产定价模型是基于市场组合的单因素模型，它认为资产的预期收益率主要由市场风险（即系统性风险或不可分散风险）决定。资本资产定价模型的核心是贝塔系数 (Beta)，用于衡量个别证券或投资组合相对于整个市场的风险。而套利定价理论对应多因子模型，认为资产收益不仅受市场风险的影响，还受到多种宏观经济因素的影响。套利定价理论并不局限于单一风险源，因此其对应的多因子模型会考虑多种风险因素（如通货膨胀率、利率、工业产值等）。显然，APT 比 CAPM 更具一般性，因为它允许存在多个风险因素；不同于 CAPM，APT 也不要求识别市场组合。从实践角度来看，APT 和因子模型较为抽象：既没有告诉我们因子是什么、该如何选取，也没有给出资产对各个因子的敏感性应如何估计。但这正是 APT 理论一般性与灵活性的体现。

APT 理论认为金融市场是竞争性 (competitive) 且无摩擦的 (frictionless)。根据 Ross (1976) 的研究，APT 意味着

$$\boldsymbol{\mu} \approx \iota \lambda_0 + \mathbf{B} \boldsymbol{\lambda}_K,$$

其中， $\mathbf{B} \in \mathbb{R}^{N \times K}$ ,  $\boldsymbol{\lambda}_K \in \mathbb{R}^{K \times 1}$ 。 $\boldsymbol{\mu}$  为  $(N \times 1)$  维预期回报向量； $\lambda_0$  为零贝塔参数（若存在无风险资产，则等于无风险回报率）； $\boldsymbol{\lambda}_K$  为  $(K \times 1)$  维因子风险溢价向量； $\iota$  为全 1 向量（维度  $N \times 1$ ）。由于 Ross (1976) 中的 APT 为近似关系，不能直接用于严格的资产定价。在实际操作中，通常因近似误差可忽略，常取

$$\boldsymbol{\mu} = \iota \lambda_0 + \mathbf{B} \boldsymbol{\lambda}_K.$$

#### 2.4.5 基于消费的资本资产定价模型

接着我们介绍基于消费的资本资产定价模型 (Consumption Based Capital Asset Pricing Model, 简称 C-CAPM)。C-CAPM 模型是在传统的 CAPM 基础上发展起来的，考虑到消费者消费习惯对资本市场的影响，认为消费者的消费习惯与其投资决策密切相关，进而对资产定价产生重要影响。

投资者可以自由交易资产  $i$ ，并希望最大化时间可分离型效用函数 (time-separable utility function) 的期望值：

$$\max E_t \left[ \sum_{j=0}^{\infty} \delta^j U(C_{t+j}) \right], \quad (2.8)$$

其中  $\delta$  是时间折现因子， $C_{t+j}$  是投资者在时间  $t+j$  的消费， $U(C_{t+j})$  是在  $t+j$  时期消费带来的效用。

投资者最优消费和投资计划的一阶条件（欧拉方程）为：

$$U'(C_t) = \delta E_t[(1 + R_{i,t+1}) U'(C_{t+1})]. \quad (2.9)$$

等式 (2.9) 的左边表示在时间  $t$  少消费一单位人民币的边际效用成本，右边则表示在时间  $t$  将这一单位人民币投资于资产  $i$ ，在  $t+1$  时以  $(1 + R_{i,t+1})$  人民币出售并消费收益的预期边际效用收益。最优化条件使得投资者的边际成本等于边际收益，因此方程 (2.9) 描述了最优条件。

将等式 (2.9) 的左右两边都除以  $U'(C_t)$ ，可得：

$$1 = E_t[(1 + R_{i,t+1}) M_{t+1}], \quad (2.10)$$

其中  $M_{t+1} = \delta U'(C_{t+1}) / U'(C_t)$ 。变量  $M_{t+1}$  被称为随机贴现因子 (stochastic discount factor)，或定价核 (pricing kernel)。在当前模型中，它等于边际效用折现比  $\delta U'(C_{t+1}) / U'(C_t)$ ，也被称为跨期边际替代率 (intertemporal marginal rate of substitution)。值得注意的是，因为边际效用总是正的，所以随机贴现因子以及跨期边际替代率均为正。

对等式 (2.10) 的左右两边取无条件期望，并将时间滞后一期以简化表达式，可得：

$$1 = E[(1 + R_{it}) M_t]. \quad (2.11)$$

等式 (2.11) 表明： $E[(1 + R_{it}) M_t] = E[1 + R_{it}] E[M_t] + \text{Cov}(R_{it}, M_t)$ ，因此

$$E[1 + R_{it}] = \frac{1}{E[M_t]}(1 - \text{Cov}(R_{it}, M_t)). \quad (2.12)$$

如果存在一个与随机贴现因子的无条件协方差为零的资产——即一个“无条件零贝塔资产”，则等式 (2.12) 可简化为该资产的预期总回报： $E[1 + R_{0t}] = 1/E[M_t]$ 。

将其代入等式 (2.12) 中，可得资产  $i$  相对于零贝塔资产的超额回报  $Z_{it}$  的表达式为：

$$E[Z_{it}] \equiv E[R_{it} - R_{0t}] = -E[1 + R_{0t}] \cdot \text{Cov}(R_{it}, M_t). \quad (2.13)$$

通过上式不难看出，一个资产与随机贴现因子的协方差越小，该资产的预期回报越高。与  $M_{t+1}$  协方差较小的资产往往在投资者的边际消费效用较高时具有较低的回报，这通常发生在消费较低时；换而言之，该资产“锦上添花”，而非“雪中送炭”。这种资产在风险的意义上表现为在财富最有价值时无法提供财富，因此投资者要求更高的风险溢价。

在实证研究中，通常假定个人可以汇总为单一的代表性投资者，因此可以用总消费代替任意个体的消费。等式 (2.10) 可以改写为：

$$1 = E_t[(1 + R_{i,t+1}) M_{t+1}], \quad (2.14)$$

其中  $M_{t+1} = \delta U'(C_{t+1}) / U'(C_t)$ ， $C_t$  为总消费，等式 (2.14) 被称为基于消费的资本资产定价模型 (consumption based capital asset pricing model，简称 C-CAPM)。

假设存在一个代表性代理人，其目标是最大化时间可分离的幂效用函数：

$$U(C_t) = \frac{C_t^{1-\gamma} - 1}{1 - \gamma}, \quad (2.15)$$

其中  $\gamma$  是相对风险厌恶系数。当  $\gamma \rightarrow 1$  时， $U(C_t) \rightarrow \log C_t$ 。

幂效用函数有几个重要的性质，其中各有利弊：首先，幂效用函数具有尺度不变性（scale-invariant）：在回报分布恒定的情况下，随着总财富和经济规模的增加，风险溢价不会随时间变化。这使得我们可以将所有投资者汇总为一个代表性投资者，只要他们具有相同的幂效用函数，不论其财富水平如何。这也解释了在 C-CAPM 的实证中为什么采用总消费而不是个体消费——一方面这是由于微观数据很难获得，另一方面是因为 C-CAPM 模型保证了可以将多个消费者的数据汇总。值得注意的是，现实中投资者的风险态度可能会随着财富的增加而发生变化。其次，当效用具有幂函数形式时，跨期替代弹性（即计划的对数消费增长对对数利率的导数） $\psi$  是相对风险厌恶系数  $\gamma$  的倒数——这一点较不理想。此外，由于幂效用函数在理论上对高收益的极端值（尾部）较为敏感，可能导致在面对厚尾分布的资产时，估计出的风险溢价偏高。

在后续的章节中，我们将介绍如何把本章所述的金融学理论模型转化为计量经济学模型，并对其进行估计、检验以及预测。

### 均衡定价与套利定价

均衡定价与套利定价是金融经济学中用于描述资产价格形成的两种重要理论。均衡定价（Equilibrium Pricing）基于市场供需平衡的观点，认为市场中的资产价格是在市场供需力量相互作用下自然形成的。在均衡状态下，资产的市场价格反映了所有市场参与者对其未来现金流的风险与回报的合理预期。典型的均衡定价模型包括资本资产定价模型（CAPM）以及基于消费的资本资产定价模型（C-CAPM）。

套利定价理论基于无套利原则，认为资产的价格应由一组系统性风险因子决定，任何价格偏离这些因子的线性组合都会引发套利机会，从而导致市场价格的调整，使其回归合理水平。

尽管均衡定价和套利定价都用于理解资产价格的形成，但两者有不同的出发点：均衡定价强调市场的整体均衡及其动态调整，而套利定价强调通过无套利机会保证价格的合理性。均衡定价模型（如 CAPM, C-CAPM）提供了简明的市场风险与回报之间的关系，而套利定价模型则允许更复杂的多因子分析，以解释资产收益率的变化。均衡定价与套利定价各有优势，通常在金融实践中相辅相成，共同用于资产定价和风险管理。

## 2.5 章节总结

本章围绕金融市场、收益率及金融经济学基础展开介绍。在金融市场方面，详细阐述了不同类型的市场，涵盖货币市场、债券市场和股票市场等，使读者对金融市场的主要构成有清晰认知。对于市场类型以及交易方式虽未展开细述，但点明了此部分内容在金融体系中的重要地位。

收益率部分介绍了股票指数的编制方法，同时探讨了收益率的统计性质，为进一步研究金融资产的收益特征奠定基础。

金融经济学基础是本章的重点内容之一。首先讲解了效用函数以及风险厌恶的概念，这是理解投资者行为和决策的关键。接着在均值一方差框架下的投资组合理论中，分别讨论了存在无风险资产和不存在无风险资产的情形。资本资产定价模型方面，介绍了夏普-林特纳版本的 CAPM 和布莱克版本；此外，还提及了套利定价理论（APT）和基于消费的资本资产定价模型（C-CAPM）等内容。上述模型与理论为金融资产定价和投资决策提供了重要的理论依据。通过本章的学习，读者能够对金融市场、收益计算以及相关经济学理论形成较为系统的认识。

## 2.6 习题

1. 金融市场是现代经济体系中的关键组成部分，涉及各种金融工具和交易市场。在本节的习题中，我们将探讨金融市场的不同类型及其运作方式，分析主要的金融工具以及它们在经济中的角色。
2. 货币市场：
  - (a) 列举三种主要的货币市场工具，并描述它们的特点和功能。
  - (b) 简述国库券与商业票据之间的主要区别。
  - (c) 伦敦银行同业拆借利率 (LIBOR) 和有担保隔夜融资利率 (SOFR) 有何不同？它们在金融市场中的作用是什么？
3. 债券市场：
  - (a) 描述零息债券的特点，以及它们与附息债券的区别。
  - (b) 解释联邦机构债务的概念，并列举两个常见的发行机构。
  - (c) 在中国的债券市场中，哪些主要类型的债券由政府发行？简述这些债券的用途。
4. 股票市场：
  - (a) 什么是普通股？普通股持有者在公司清算时的清偿顺序如何？
  - (b) 优先股有哪些特点？它们如何兼具债权和股权的特征？
  - (c) 什么是市场指数？举例说明两个常见的股票市场指数。例如：道琼斯工业平均指数、沪深 300 指数。
5. 市场类型以及交易方式：
  - (a) 什么是电子拍卖市场？它与庄家（做市商）市场的主要区别是什么？
  - (b) 解释什么是“长期有效订单”（Good-Till-Cancelled, GTC），以及它如何适用于电子交易系统。
  - (c) 在中国，上海证券交易所和深圳证券交易所的交易模式有哪些特点？描述其经纪市场与做市商相关安排（如价格优先、时间优先的集中竞价原则，以及在部分板块实施的做市商制度及其报价义务）。
6. 股指期货：
  - (a) 描述中国股指期货市场的发展历程，包括 2015 年股市动荡对股指期货的影响。
  - (b) 股指期货有哪些主要功能？它们在风险管理的价格发现中的作用是什么？
7. 金融市场的监管和透明度：
  - (a) MiFID II 与美国的 Reg NMS 规则有何共同目标？它们如何促进金融市场的透明度和公平性？
  - (b) 在中国，哪些主要法律和机构负责对金融市场进行监管？简述其主要职责。
8. 交易订单类型及其在金融市场中的应用：
  - (a) 什么是市价单？它与限价单的主要区别是什么？

- (b) 解释“挂钩订单”(Pegged Order)的概念，并说明其在交易系统中的适用机制。
- (c) 什么是暗池交易？它在金融市场中的作用是什么？
9. 从上证指数或者深证成指中获取某只股票 2019 年至 2024 年的日度价格数据。您选择的股票不应与班上其他同学重复。计算可以在 Excel 和/或 R 中执行，也可在其他软件包中完成。
- (a) 计算股票收益率（基于每日收盘价计算）序列的样本统计量，包括均值、标准差、偏度和峰度。可忽略股息，仅考虑资本增值部分。
- (b) 计算前 20 个自相关系数，并检验该序列是否具有线性可预测性。
- (c) 对数收益率和算术收益率的计算有何不同？
10. 在博弈论中的“圣彼得堡悖论”中，期望值为无穷大，但实际上很少有人愿意支付高价来参加此赌局。基于效用理论的观点，回答以下问题。
- (a) 为什么圣彼得堡悖论的期望值为无穷大？
- (b) 使用对数效用函数，计算圣彼得堡悖论的期望效用。
- (c) 对比期望效用与期望值，说明人们在决策中考虑的因素为何不只是数学上的期望值。
11. 假设您有一个对数效用函数： $u(y) = \ln(y)$ ，请回答以下问题：
- (a) 您愿意为参与一个 50% 概率赢得 1000 元、50% 概率一无所获的赌局支付多少入场费？
- (b) 解释对数效用函数下投资者的风险厌恶特性 ( $U'(W) = 1/W > 0$ ,  $U''(W) = -1/W^2 < 0$ , 相对风险厌恶系数为常数 1, 表现为 CRRA; 绝对风险厌恶系数  $A(W) = 1/W$  呈递减, 符合 DARA)。
- (c) 若效用函数改为线性函数： $u(y) = y$ , 如何解释这一投资决策的变化？
12. 关于期望效用理论，请回答以下问题：
- (a) 什么是独立性公理？
- (b) 结合阿莱悖论 (Allais Paradox)，说明独立性公理在实际决策中的局限性。
- (c) 如何通过行为经济学解释人们在阿莱悖论中的选择？
13. 前景理论指出，人们在做出决策时，更加重视损失而非同等程度的收益。请回答以下问题：
- (a) 描述前景理论的价值函数的形状及其特征。
- (b) 什么是损失厌恶？请举例说明它在日常生活中的表现。
- (c) 如何利用前景理论解释股市中的投资者行为，例如牛市与熊市中的投资决策差异。
14. 假设一个投资者的效用函数为  $u(y) = -e^{-\alpha y}$ , 其中  $\alpha > 0$ , 请回答以下问题：
- (a) 计算该投资者的绝对风险厌恶系数  $R_A(y)$ 。
- (b) 解释为什么该效用函数被称为常绝对风险厌恶型 (CARA) 效用函数。
- (c) 在该效用函数下，财富的增加对风险厌恶程度有何影响？

15. 在均值一方差分析中，投资者会根据期望收益率和方差来选择投资组合。请回答以下问题：

- (a) 什么是最小方差组合？如何计算其权重？
- (b) 解释两基金分离定理，并说明其在投资组合管理中的重要性。
- (c) 如果存在无风险资产，如何使用资本市场线（CML）来选择最优投资组合？

16. 资本资产定价模型（CAPM）在金融经济学中被广泛应用。请回答以下问题：

- (a) 使用 CAPM 模型，解释某项资产的预期收益率由哪些因素决定。
- (b) 计算某假设股票的  $\beta$  值，并解释其经济含义。
- (c) 结合夏普-林特纳版本的 CAPM，说明如何用市场组合的超额收益率来解释个别资产的超额收益率。

17. 假设你是一位投资顾问，目前正为一位客户管理其投资组合。该客户希望在未来一年获得尽可能高的期望回报率，同时也希望控制投资风险。你的任务是使用均值一方差分析为客户设计一个理想的投资组合。以下是三种资产的预期年收益率和标准差，以及它们之间的相关系数：

- 资产 A：预期收益率 = 8%，标准差 = 15%
- 资产 B：预期收益率 = 12%，标准差 = 20%
- 资产 C：预期收益率 = 10%，标准差 = 18%

资产间的相关系数如下：

- 资产 A 和资产 B: 0.5
- 资产 A 和资产 C: -0.2
- 资产 B 和资产 C: 0.3

- (a) 计算资产 A、B、C 的组合（假设每种资产的投资比例分别为  $w_A$ 、 $w_B$  和  $w_C$ ，且  $w_A + w_B + w_C = 1$ ）的预期收益率和组合风险（以组合标准差衡量）。
- (b) 假设客户愿意接受最大 16% 的年化波动率，找出在这个风险水平下能够提供最高预期回报的投资组合。
- (c) 使用拉格朗日乘数法，解释如何找到在一定风险水平下预期收益率最大的投资组合。

18. 考虑一个经济体，其中所有投资者都遵循幂效用函数

$$U(C_t) = \frac{C_t^{1-\gamma}}{1-\gamma},$$

其中  $\gamma$  为相对风险厌恶系数。在这个环境中，投资者可以投资两种资产：无风险资产和一种风险资产。无风险资产的收益率  $R_f$  已知且固定；风险资产的收益率  $R$  为随机变量，满足  $E[R] = \mu$ ,  $\text{Var}(R) = \sigma^2$ 。假设投资者的初始财富为  $W_0$ ，他们希望在未来一期最大化其期望效用。

- (a) 使用 C-CAPM 模型推导风险资产的风险溢价  $E[R] - R_f$  的表达式。
- (b) 讨论风险厌恶系数  $\gamma$  如何影响投资者对风险资产的需求。
- (c) 假设随机贴现因子  $M$  可以表示为  $M = \beta \frac{U'(C_{t+1})}{U'(C_t)}$ ，其中  $\beta$  是贴现因子。根据  $E_t[(1+R)M] = 1$  的条件，说明为何  $\text{Cov}_t(R, M)$  与  $E_t[R]$  之间存在关系。



# 3 线性回归模型

在金融计量经济学中，回归模型发挥着核心作用，尤其体现在对金融市场、投资决策和风险管理的深入分析上。这些模型被关键地应用于资产定价，如资本资产定价模型（CAPM）和套利定价理论（APT），通过回归分析来估计资产的预期收益率和风险。在风险管理方面，它们用于评估金融资产的波动率和相关性，这对于构建投资组合和计算风险调整后的投资收益率至关重要。回归模型还被用于测试金融市场的效率，探究信息如何被市场吸收并反映在股票价格中。此外，回归模型可被用来分析宏观经济变量与金融市场之间的互动，帮助理解宏观经济政策对金融市场的影响。在后续的金融时间序列分析中，特别是自回归模型和协整模型，被广泛用于分析和预测市场趋势。总之，回归模型在金融计量经济学中不仅加深了理论研究的深度，也为实际的金融操作和决策制定提供了强大的分析工具。

构建计量经济学模型首先需要明确问题定义和理论背景。这要求深入理解相关的经济理论与假设，以指导模型构建。然后应根据研究问题与可用数据的性质，选择合适的模型类型，如线性或非线性模型、静态或动态模型。随后进行数据的收集与处理，包括数据清理、处理缺失值与异常值，以及必要的数据转换。模型设定是接下来的步骤，需确定哪些变量作为自变量和因变量及其预期关系。然后，使用统计方法（例如最小二乘法）来估计模型参数，目标是找到最能解释数据的参数值。紧接着进行模型检验，通过各种统计测试检验模型的有效性与假设的合理性。一旦模型被验证有效，便可以解释估计结果，并在可能的情况下使用不同的数据集或方法进行再验证，确保模型的稳健性。最后，若模型有效，可将其用于政策分析与预测，这是将模型应用于实际经济问题的关键一步。整个过程中，对数据质量、模型假设的合理性以及结果的解释都需给予充分关注，因为计量经济模型虽然是强大的工具，但使用时需要谨慎和具备批判性思维。

本章将深入探讨计量经济学中的关键概念，结合 R 语言案例，细致介绍一元以及多元线性回归模型。

## 3.1 一元线性回归模型

本节将介绍一元回归模型（也称简单线性回归模型），这是计量经济学中用于分析一个自变量与一个因变量之间线性关系的基本工具。简单线性回归关注的是两个变量之间的直线关系。

一元回归模型的基本形式可以表示为：

$$y = \beta_0 + \beta_1 x + \varepsilon \quad (3.1)$$

其中， $y$  为被解释变量； $x$  为解释变量，用于预测或解释  $y$ ； $\beta_0$  为截距，表示当  $x = 0$  时  $y$  的预期值； $\beta_1$  为斜率，表示  $x$  每变化一个单位时  $y$  的预期变化； $\varepsilon$  为随机误差项，代表除  $x$  之外影响  $y$  的其他因素。

一元线性回归的目标是找到最佳的  $\beta_0$  和  $\beta_1$ ，使得模型预测的  $y$  值尽可能接近实际观

测到的  $y$  值。 $y$  和  $x$  在不同语境下对应的术语见表 3.1。值得注意的是，虽然表 3.1 针对一元回归模型，其中术语在多元回归模型中也同样适用。

表 3.1: 回归中  $x$  和  $y$  的术语

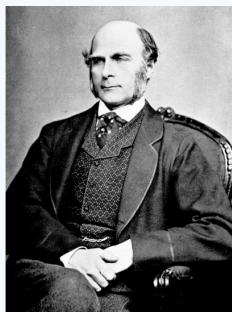
| $y$                                        | $x$                         |
|--------------------------------------------|-----------------------------|
| 因变量 (Dependent variable)                   | 自变量 (Independent variable)  |
| 被解释变量 (Explained variable)                 | 解释变量 (Explanatory variable) |
| 被预测变量 / 响应变量 (Predicted/Response variable) | 预测变量 (Predictor variable)   |
| 被回归变量 (Regressand)                         | 回归变量 (Regressor)            |
| —                                          | 控制变量 (Control variable)     |

一元回归模型在金融经济学中有许多实际应用场景。比如，分析股票指数与基准利率之间的关系时，可以研究当利率变动时，股市整体表现如何受到影响。再比如，可以通过一元回归模型研究房地产市场价格与国家 GDP 增长率之间的关系，帮助理解在经济扩张或衰退期间房地产市场的表现趋势。此外，可以探讨货币汇率变化如何影响进出口量，例如：分析美元兑人民币汇率的变化如何影响美国对中国的出口。

一元回归模型还可以用于研究公司财务指标（如市盈率、股息率）与其股票收益率之间的关系。一元回归模型还可以用来分析金融政策与债券市场，例如量化央行政策（如量化宽松）对长期债券收益率的影响，这有助于投资者理解宏观经济政策如何影响债券市场的表现。

总之，通过分析一元回归模型的参数（如斜率和截距），我们可以了解变量之间关系的强度与方向。

### “回归”与“中庸之道”



“回归”一词源于“回归现象”，也被称作“高尔顿定律”，由英国统计学家弗朗西斯·高尔顿在 1889 年提出。当时，高尔顿在研究遗传学中身高特征的传递时，发现了一个饶有趣味的现象：在观察父母与子女的身高关系时，他注意到，高个子父母所生子女往往较高，但子女的身高通常低于父母的平均身高；而较矮的父母所生子女一般也偏矮，可子女的平均身高却高于父母。这表明，无论父母身高处于何种极端情况，后代的身高都倾向于向群体平均身高靠拢。“高尔顿定律”不仅揭示了遗传特征在代际传递中的规律，也体现出一种趋向平均水平的趋势，因此成为统计学和遗传学领域的重要概念。

这种“回归”现象与中国传统文化中的“中庸之道”有着异曲同工之妙。“中庸之道”作为儒家思想的重要核心理念之一，旨在追求万事万物的平衡与和谐，避免走向极端。正如身高遗传的回归现象所示，无论父母身高如何，子女身高总会趋近于群体平均身高，仿佛有一种潜在的平衡机制在发挥作用。

在日常生活里，“中庸之道”指导人们为人处世、做决策时把握分寸，既不激进冒进，也不保守退缩。这与回归现象中趋向平均水平的本质高度契合。在经济领域的计量分析中，运用回归模型研究变量关系时，同样需要秉持“中庸之道”的理念，精准探寻数据间的最佳关系。只有这样，才能避免模型出现过度拟合或欠拟合的问题，进而更准确地揭示经济现象背后隐藏的规律。

这种跨越时空与学科界限的呼应，充分展现了中国传统文化中“中庸之道”的深邃智慧和普遍适用性。它激励着我们在现代科学的研究与学习过程中，持续从传统文化中汲取

智慧，不断增强文化认同感与民族自豪感，让古老的智慧在现代社会中绽放新的光彩。

在 R 中，有几个内置或常用包自带的数据集非常适合用来演示一元回归模型。例如：ggplot2 包的 economics 数据集：该数据集包含失业率、个人收入等经济时间序列，可用于构建与经济趋势相关的一元回归模型；quantmod 包中的 stocks 数据集。此外，美国圣路易斯联邦储备银行（Federal Reserve Economic Data, FRED）的宏观经济与金融数据可通过 quantmod 或 fredr 包访问。

### 3.1.1 一元线性回归模型的基本假设

令  $x = \{x_i\}$  以及  $y = \{y_i\}$ ，其中  $i = 1, \dots, n$ ， $n$  称为样本量。一元线性回归模型需要满足以下基本假设：

1. 线性关系： $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ 。
2. 随机误差的期望值：随机误差  $\varepsilon_i$  的期望值为  $E(\varepsilon_i) = 0$ ；等价地， $E(y_i | x_i) = \beta_0 + \beta_1 x_i$ 。
3. 方差齐性（同方差性）：误差项具有常方差，

$$\text{Var}(\varepsilon_i) = \sigma^2 \quad (\text{或 } \text{Var}(\varepsilon_i | x_i) = \sigma^2). \quad (3.2)$$

4. 随机误差的无相关性：任意两个不同观测的误差不相关 ( $i \neq j$ )，

$$\text{Cov}(\varepsilon_i, \varepsilon_j) = 0. \quad (3.3)$$

5. 正态性（可选）：若响应变量条件分布为正态，则误差亦为正态，反之亦然，

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2) \Leftrightarrow \varepsilon_i \sim N(0, \sigma^2). \quad (3.4)$$

假设 2 表明模型没有遗漏任何与  $x$  相关的变量。假设 4 确保误差项相互独立，保证一个观测值的误差不会影响另一个观测值的误差。它的一个更强版本的假设是随机误差  $\varepsilon_i$  与  $\varepsilon_j$  ( $i \neq j$ ) 相互独立。值得注意的是，统计独立性能够确保协方差为零，但零协方差并不能保证统计独立性。以下为一反例：

**例 3.1：**令随机变量  $x$  满足  $E(x) = 0$  且  $E(x^3) = 0$ （如均值为零的正态分布）。取  $y = x^2$ 。显然， $x$  和  $y$  并非独立，但

$$\text{Cov}(x, y) = E[(x - Ex)(y - Ey)] = E(xy) - E(x)E(y) = E(x^3) - 0 = 0, \quad (3.5)$$

其中最后一步利用了给定条件  $E(x^3) = 0$ 。

### 3.1.2 普通最小二乘法

在实际研究中，我们通常无法直接知道总体回归线的具体参数。因此，我们需要利用收集的数据估计这些未知的参数（回归线的截距和斜率）。理论上，任意两个数据点  $(x_i, y_i)$  就能确定一条直线。但在实际操作中，面对众多可能的直线，我们该如何选择最合适的一条呢？

最常见且有效的方法是“最小二乘法”(Ordinary Least Squares, 简称 OLS)。根据这一方法, 我们应当选择一条直线, 使得所有数据点到这条直线的垂直距离的平方和最小。这种方法的目的是让估计出的回归线尽可能接近实际观测到的数据点, 其中“接近程度”是通过预测给定  $x$  值时  $y$  的观测值与预测值之差的平方和来衡量的。虽然这个选择规则是基于一定假设的, 但它能有效地描绘出一条贯穿数据中心趋势的回归线。

**定义 3.1:** 在所有可能的直线中, 最佳拟合线可以使以下目标函数最小化:

$$Q(\beta_1, \beta_2) = \sum_{i=1}^n (y_i - E(y_i))^2 = \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2. \quad (3.6)$$

令回归线参数  $\beta_0$  和  $\beta_1$  的估计量为  $\hat{\beta}_0$  和  $\hat{\beta}_1$ 。使得等式 (3.6) 达到最小的  $\hat{\beta}_0$  和  $\hat{\beta}_1$  的值被称为最小二乘估计值 (Ordinary Least Squares Estimators)。 $\hat{\beta}_0$  和  $\hat{\beta}_1$  可以通过将以下偏导数为零的方程设为 0 获得:

$$\frac{\partial Q}{\partial \beta_0} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i), \quad (3.7)$$

$$\frac{\partial Q}{\partial \beta_1} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) x_i. \quad (3.8)$$

令  $y$  和  $x$  的样本均值分别为:

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i, \quad \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i. \quad (3.9)$$

由式(3.8)-(??)可得  $\hat{\beta}_0$  和  $\hat{\beta}_1$  的估计量:

$$\sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0 \Rightarrow n \hat{\beta}_0 = \sum_{i=1}^n y_i - \hat{\beta}_1 \sum_{i=1}^n x_i \Rightarrow \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}, \quad (3.10)$$

以及

$$\sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) x_i = 0 \Rightarrow \sum_{i=1}^n y_i x_i - \hat{\beta}_0 \sum_{i=1}^n x_i - \hat{\beta}_1 \sum_{i=1}^n x_i^2 = 0 \Rightarrow \hat{\beta}_1 = \frac{S_{xy}}{S_{xx}}, \quad (3.11)$$

其中

$$S_{xy} = (n-1) \text{Cov}(x_i, y_i) = \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = \sum_{i=1}^n x_i y_i - n \bar{x} \bar{y}, \quad (3.12)$$

$$S_{xx} = (n-1) \text{Var}(x_i) = \sum_{i=1}^n (x_i - \bar{x})^2 = \sum_{i=1}^n x_i^2 - n \bar{x}^2. \quad (3.13)$$

因此, 基于最小二乘法的最佳拟合直线为:

$$y = \hat{\beta}_0 + \hat{\beta}_1 x = \bar{y} - \hat{\beta}_1 \bar{x} + \hat{\beta}_1 x = \bar{y} + \hat{\beta}_1 (x - \bar{x}) = \bar{y} + \frac{S_{xy}}{S_{xx}} (x - \bar{x}). \quad (3.14)$$

模型拟合的残差由下式给出

$$\hat{\varepsilon}_i = y_i - \hat{y}_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i = y_i - \bar{y} - \hat{\beta}_1 (x_i - \bar{x}) . \quad (3.15)$$

最小二乘法的最佳拟合线的几个重要性质包括：

1. 最小二乘法的最佳拟合线经过均值点  $(\bar{x}, \bar{y})$ , 即模型在整体上对数据集的中心趋势进行了准确拟合;
2. 残差和为 0:

$$\sum_{i=1}^n \hat{\varepsilon}_i = \sum_{i=1}^n (y_i - \hat{y}_i) = \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = n (\bar{y} - \hat{\beta}_0 - \hat{\beta}_1 \bar{x}) = 0 \quad (3.16)$$

这表明模型在整体上没有系统性的偏差。

3. 由  $x_i$  加权的残差之和为零:

$$\begin{aligned} \sum_{i=1}^n x_i \hat{\varepsilon}_i &= \sum_{i=1}^n x_i (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = \sum_{i=1}^n x_i y_i - n \bar{x} \hat{\beta}_0 - \hat{\beta}_1 \sum_{i=1}^n x_i^2 \\ &= \sum_{i=1}^n x_i y_i - n \bar{x} (\bar{y} - \hat{\beta}_1 \bar{x}) - \hat{\beta}_1 \sum_{i=1}^n x_i^2 \\ &= \sum_{i=1}^n x_i y_i - n \bar{x} \bar{y} + \hat{\beta}_1 n \bar{x}^2 - \hat{\beta}_1 \sum_{i=1}^n x_i^2 \\ &= S_{xy} - \hat{\beta}_1 S_{xx} = 0 , \end{aligned} \quad (3.17)$$

即模型在不同水平的自变量上没有系统性偏差。

4.  $x_i$  与  $\hat{\varepsilon}_i$  不相关:

$$\text{Cov}(x_i, \hat{\varepsilon}_i) = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x}) (\hat{\varepsilon}_i - \bar{\varepsilon}) = \frac{1}{n-1} \left( \sum_{i=1}^n x_i \hat{\varepsilon}_i - n \bar{x} \bar{\varepsilon} \right) = 0 , \quad (3.18)$$

其中  $\bar{\varepsilon} = \frac{1}{n} \sum_{i=1}^n \hat{\varepsilon}_i$ , 即模型不存在遗漏变量导致的偏差;

5. 由  $\hat{y}_i$  加权的残差之和为零:

$$\sum_{i=1}^n \hat{y}_i \hat{\varepsilon}_i = \sum_{i=1}^n (\hat{\beta}_0 + \hat{\beta}_1 x_i) \hat{\varepsilon}_i = \hat{\beta}_0 n \bar{\varepsilon} + \hat{\beta}_1 \sum_{i=1}^n x_i \hat{\varepsilon}_i = 0 , \quad (3.19)$$

即残差在不同拟合水平上平均分布;

6.  $\hat{y}_i$  和  $\hat{\varepsilon}_i$  不相关:

$$\text{Cov}(\hat{y}_i, \hat{\varepsilon}_i) = \frac{1}{n-1} \sum_{i=1}^n (\hat{y}_i - \bar{y}) (\hat{\varepsilon}_i - \bar{\varepsilon}) = \frac{1}{n-1} \left( \sum_{i=1}^n \hat{y}_i \hat{\varepsilon}_i - n \bar{y} \bar{\varepsilon} \right) = 0 , \quad (3.20)$$

其中  $\bar{y} = \frac{1}{n} \sum_{i=1}^n \hat{y}_i$ , 这点进一步证明了模型拟合的有效性。

误差 (Error) 和残差 (Residual) 是回归分析中两个重要但不同的概念。误差是指在回归模型中，真实数据点与理论模型预测值之间的差异。由于误差涉及到真实但未知的模型参数，因此是不可观测的。误差通常被假定为具有均值为零的正态分布，并反映了数据中未被模型解释的随机波动。相比之下，残差是观测值与回归模型预测值之间的差异，是可以直接观测到的。残差在模型诊断、验证假设（如方差齐性和无自相关性）以及改进模型中起着重要作用。它们可以帮助识别模型是否适合数据、是否存在异常值等。

### 3.1.3 学生 t 检验与决定系数 ( $R^2$ )

本节主要介绍 t 统计量与  $R^2$ 。在线性回归模型中，对  $\beta_i$  的显著性检验通常采用 t 统计量，其公式为：

$$t_i = \frac{\hat{\beta}_i - 0}{SE(\hat{\beta}_i)}, \quad (3.21)$$

其中  $\hat{\beta}_i$  是  $\beta_i$  的估计值 ( $i = 0, 1$ )， $SE(\hat{\beta}_i)$  是该估计值的标准差。



学生 t 检验 (Student's t-test) 有着一段有趣的历史。这个统计方法由威廉·戈塞特 (William Sealy Gosset) 发明，他是一位在都柏林的吉尼斯啤酒厂工作的化学家和统计学家。由于当时的样本量通常很小，戈塞特需要一种方法来确定小样本数据集之间的差异是否具有统计学意义，因此推导出了一种新的概率分布，即现在所称的 t 分布。这个分布适用于估计的标准误差，特别是当样本量较小时。

戈塞特在 1908 年发表了他的工作，但由于他的雇主（吉尼斯啤酒厂）担心竞争对手可能利用他的发现，他使用了笔名 “Student” 来发表论文。因此，这项检验被称为 “Student's t-test”。最初，学术界对戈塞特的工作不太重视，部分原因是他在几乎没有正式统计训练。但随着时间的推移，他的方法因其实用性和有效性而被广泛接受，并在各种科学的研究中得到应用。t 检验对统计学和实验科学产生了深远影响。它不仅为小样本研究提供了一种有效工具，而且也在教育、社会科学、生物学等许多领域中成为标准的分析方法。

图 3.1 展示了不同自由度 (degrees of freedom, 简称 DF) 下 t 分布与标准正态分布的概率密度函数 (probability density function, 简称 PDF) 的比较。从图中可以看出：当自由度较小时（例如  $df = 1$ ），t 分布的尾部较厚，即在分布的远端（横轴两端）概率密度较高；随着自由度的增加，t 分布的峰变得更高耸、尾部更薄，这意味着它逐渐接近正态分布的形状；当自由度很大时（例如  $df = 30$ ），t 分布的形状与标准正态分布非常接近，几乎重合。这是中心极限定理的一种体现：随着样本量（这里类比为自由度）的增加，样本均值的分布趋向于正态分布。<sup>1</sup>

总平方和 (total sum of squares, 简称 SST)、回归平方和 (regression sum of squares,

<sup>1</sup> 自由度可以简单地理解为独立观测值的数量减去待估参数的数量。例如，在单样本 t 检验中，自由度为样本量减去 1 ( $df = n - 1$ )；在独立样本 t 检验 (双样本) 中，若假设两总体方差相等 (方差齐性)，则自由度为两样本量之和减去 2 ( $df = n_1 + n_2 - 2$ )。

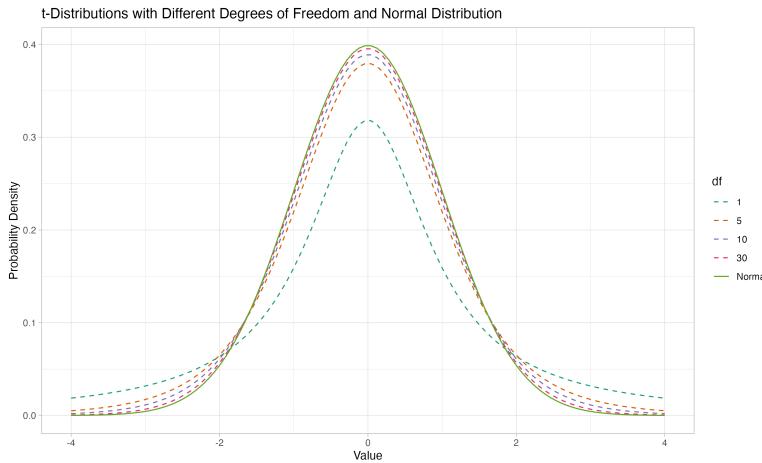


图 3.1: 学生 t 分布与标准正态分布概率密度函数的比较

简称 SSR) 和残差平方和 (error sum of squares, 简称 SSE)，它们的定义如下：

$$\begin{aligned} SST &= \sum_{i=1}^n (y_i - \bar{y})^2, \\ SSR &= \sum_{i=1}^n (\hat{y}_i - \bar{y})^2, \\ SSE &= \sum_{i=1}^n \hat{\varepsilon}_i^2. \end{aligned} \quad (3.22)$$

SST 是样本被解释变量  $y$  的总变异程度的度量，即  $y$  的分散程度；SSR 衡量  $\hat{y}$  的变异，SSE 则度量  $\hat{\varepsilon}_i$  的样本变异。 $y$  的总变异程度为已解释的变异和未解释的变异之和：

$$SST = SSR + SSE. \quad (3.23)$$

除非所有的  $y_i$  都相同，否则总平方和 (SST) 不等于零。将式 (3.23) 两边同时除以 SST 可得

$$1 = \frac{SSR}{SST} + \frac{SSE}{SST}.$$

回归的  $R^2$  (决定系数) 定义为：

$$R^2 \equiv \frac{SSR}{SST} = 1 - \frac{SSE}{SST}. \quad (3.24)$$

$R^2$  是解释变异与总变异的比率，即  $y$  的样本变异中可以由  $x$  解释的部分。式 (3.24) 中的第二个等式提供了另一种计算  $R^2$  的方法。

从式 (3.24) 可知， $R^2$  的值总是在 0 和 1 之间。如果数据点都位于同一条直线上，OLS 可以完美拟合数据，此时  $R^2 = 1$ 。 $R^2$  越接近于 0，表示最小二乘回归线的拟合程度越差： $y_i$  的变异中只有很少部分能被  $\hat{y}_i$  的变异所解释。

$R^2$  等于  $y_i$  和  $\hat{y}_i$  之间的样本相关系数的平方。这就是“ $R$  平方”这个术语的由来。字母  $R$  传统上被用来表示总体相关系数的估计值，其用法在回归分析中一直保留下来。

## 3.2 多元线性回归模型

多元回归模型在金融计量经济学中发挥着关键作用，因为它能够处理多个复杂的影响因素，帮助我们更好地理解并解决金融与经济领域的问题。它提供了一种强大的分析工具，广泛用于金融市场分析、风险管理、政策制定与研究等场景。例如，金融市场中的资产价格受多种因素影响（如利率、通货膨胀率、公司盈利水平等），多元回归模型可用于分析这些因素对资产价格的综合作用；又如，金融机构和企业需要管理市场风险、信用风险和操作风险，多元回归模型可用于识别并量化不同风险因素之间的关系；此外，政府和中央银行常用多元回归模型评估政策对经济与金融市场的影响（如利率、货币供应量与通货膨胀的数量关系）；最后，分析师也可用多元回归模型进行公司估值，综合公司盈利、市盈率、市净率等多项指标以更准确地评估企业价值。

多元线性回归模型用于描述因变量  $y$  与多个解释变量  $x_1, x_2, \dots, x_k$  之间的关系，其基本形式为：

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon. \quad (3.25)$$

其中， $y$  为被解释变量； $x_1, x_2, \dots, x_k$  为解释变量<sup>2</sup>； $\beta_0$  为截距，表示当所有解释变量均为零时  $y$  的预期值； $\beta_1, \beta_2, \dots, \beta_k$  为斜率系数，表示每个解释变量  $x_i$  每变化一个单位对  $y$  预期值的影响程度； $\varepsilon$  为误差项，代表除解释变量之外影响  $y$  的其他随机因素。

多元线性回归的目标是找到最佳的  $\beta_0, \beta_1, \dots, \beta_k$ ，使模型预测值  $\hat{y}$  尽可能接近观测值  $y$ 。通常通过最小二乘法实现：寻找估计值  $\hat{\beta}_0, \dots, \hat{\beta}_k$  使残差平方和

$$\sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \dots - \hat{\beta}_k x_{ik})^2$$

最小。由此得到关于  $k+1$  个未知数  $\hat{\beta}_0, \dots, \hat{\beta}_k$  的  $k+1$  个一阶条件（正规方程）：

$$\begin{aligned} \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \dots - \hat{\beta}_k x_{ik}) &= 0, \\ \sum_{i=1}^n x_{i1} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \dots - \hat{\beta}_k x_{ik}) &= 0, \\ \sum_{i=1}^n x_{i2} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \dots - \hat{\beta}_k x_{ik}) &= 0, \\ &\vdots \\ \sum_{i=1}^n x_{ik} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \dots - \hat{\beta}_k x_{ik}) &= 0. \end{aligned}$$

对于掌握线性代数的读者而言，最直观、易懂的多元线性回归模型的表达式是矩阵形式：

$$Y = \mathbf{X}\boldsymbol{\beta} + \mathbf{u} \quad (3.26)$$

其中， $Y$  为维度  $n \times 1$  的因变量（响应变量）， $\mathbf{X}$  为维度  $n \times (k+1)$  的自变量（预测变量）矩阵且秩为  $k+1$ ， $\boldsymbol{\beta}$  为待估参数向量（维度  $(k+1) \times 1$ ）， $\mathbf{u}$  为维度  $n \times 1$  的随机误差向量，且  $E[\mathbf{u}] = \mathbf{0}$ 。

---

<sup>2</sup>我们可以将  $y$  以及  $x_1, x_2, \dots, x_k$  视为  $n \times 1$  的列向量。

普通最小二乘法旨在最小化残差平方和：

$$\mathbf{u}'\mathbf{u} = (Y - \mathbf{X}\boldsymbol{\beta})'(Y - \mathbf{X}\boldsymbol{\beta}) = Y'Y - Y'\mathbf{X}\boldsymbol{\beta} - (\mathbf{X}\boldsymbol{\beta})'Y + (\mathbf{X}\boldsymbol{\beta})'\mathbf{X}\boldsymbol{\beta}.$$

最小二乘估计量为：

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'Y.$$

注：在矩阵表达式 (3.26) 中，矩阵  $\mathbf{X}$  第一列为  $\mathbf{1}_n$  ( $n \times 1$  维列向量，每个元素均为 1)，其余每列代表一个自变量  $x_j$  ( $j = 1, 2, \dots, p$ )。例如，如果  $\mathbf{X}$  是一个  $n \times 4$  维矩阵，

$$\mathbf{X} = \begin{bmatrix} \mathbf{1}_n & x_1 & x_2 & x_3 \end{bmatrix}.$$

其中第一列为常数项  $\mathbf{1}_n$ ，第二至四列分别为自变量  $x_1, x_2, x_3$ 。写成 (3.25) 的形式为：

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon.$$

$Y$  和  $y$  的关系是： $Y$  为一个  $n \times 1$  维列向量，包含所有观测点的因变量值；每个观测点的因变量值  $y_i$  是  $Y$  向量的一个元素。若将  $y$  认定为一个  $n \times 1$  的列向量，则可记为  $Y = y$ 。

接着，我们介绍线性模型中最重要的定理——高斯-马尔可夫 (Gauss-Markov) 定理，它提供了普通最小二乘 (OLS) 估计量的一些关键性质。

**定理 3.1 (高斯-马尔可夫 (Gauss-Markov) 定理)：** 设线性回归模型为

$$Y = \mathbf{X}\boldsymbol{\beta} + \varepsilon,$$

若模型满足以下假设：

1. 线性性与外生性：回归模型为线性形式，且误差项与解释变量不相关，即对每个解释变量列  $\mathbf{X}_{\cdot j}$  ( $j = 1, \dots, k$ )，有

$$\text{Cov}(\mathbf{X}_{\cdot j}, \varepsilon) = 0.$$

2. 误差项均值为零：

$$\text{E}(\varepsilon_i) = 0, \quad \text{对于所有 } i.$$

3. 同方差性 (Homoscedasticity)：所有观测点的误差项具有相同的方差，即

$$\text{Var}(\varepsilon_i) = \sigma^2, \quad \text{对于所有 } i.$$

4. 无自相关 (误差项不相关)：

$$\text{Cov}(\varepsilon_i, \varepsilon_j) = 0, \quad \text{对于所有 } i \neq j.$$

5. 解释变量无完全多重共线性： $\mathbf{X}$  的各列线性无关，即  $\mathbf{X}$  满秩。

则在上述条件下，普通最小二乘法 (OLS) 估计量是所有线性无偏估计量中方差最小者，即最佳线性无偏估计量 (Best Linear Unbiased Estimator, BLUE)。

若进一步假设误差项服从正态分布, 即  $\varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ , 则 OLS 估计量不仅是 BLUE, 还是最小方差无偏估计量 (Minimum Variance Unbiased Estimator, 简称 MVUE); 相应的统计推断 (如  $t$  检验、 $F$  检验) 亦可成立。

由于  $\mathbf{X}$  的随机性, 很多教材或参考书将高斯-马尔可夫的假设的条件以  $\mathbf{X}$  为条件进行设定, 例如:

1. **模型定义:**  $Y = \mathbf{X}\beta + \varepsilon$ , 其中  $Y$  为被解释变量 (因变量),  $\mathbf{X}$  是解释变量 (自变量) 随机矩阵,  $\beta$  是参数向量,  $\varepsilon$  是随机误差项。
2.  **$X$  的性质:**  $\mathbf{X}$  为  $n \times (k+1)$  矩阵, 且其秩为  $k+1$ 。这确保了模型中的每个变量都提供了独特的信息, 没有完全的多重共线性。<sup>3</sup>
3. **条件均值假设:**  $E[\varepsilon | \mathbf{X}] = \mathbf{0}$ 。这意味着给定  $\mathbf{X}$ , 误差项  $\varepsilon$  的期望值为零。这个假设是为了确保模型不会系统地偏离其真实值。条件性表明  $\varepsilon$  的期望不仅在整体上是零, 而且对于每个给定的  $\mathbf{X}$  值也是零。
4. **同方差和不相关性:**  $E[\varepsilon \varepsilon' | \mathbf{X}] = \sigma^2 \mathbf{I}$ 。这表示误差项  $\varepsilon$  在给定  $\mathbf{X}$  时具有恒定的方差 (同方差性) 且不相关。这保证了模型的预测误差在不同的  $\mathbf{X}$  值下是均匀的, 不会随  $\mathbf{X}$  的变化而变化。
5. **正态分布假设:**  $\varepsilon | \mathbf{X} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ 。这表明在给定  $\mathbf{X}$  的条件下, 误差项  $\varepsilon$  服从均值为  $\mathbf{0}$ 、方差为  $\sigma^2$  的正态分布。这个假设对于推导 OLS 估计量的抽样分布和进行假设检验是关键。

在本章中, 我们将  $\mathbf{X}$  视为固定的, 这样做可以简化模型的解释以及后续分析。

普通最小二乘估计量是最佳线性无偏估计量意味着:

1. 线性: OLS 估计量在  $Y$  中是线性的。
2. 无偏性: OLS 估计量是无偏的, 即  $E(\hat{\beta}) = \beta$ 。
3. 有效性: 在所有线性无偏估计量中, OLS 估计量具有最小方差。

**证明 (高斯-马尔可夫 (Gauss-Markov) 定理的证明):** 1. 【估计 (Estimator)】

判断最小二乘估计量  $\hat{\beta}$  是否为 BLUE 首先是证明  $\hat{\beta}$  是估计量的概念。显见,  $\hat{Y}$  和  $\hat{\beta}$  为估计量, 因为它们代表了基于样本数据  $\mathbf{X}$  以及  $Y$  的函数。

2. 【无偏 (Unbiased)】要证明矩阵形式的最小二乘估计量是无偏的, 我们必须证明  $\hat{\beta}$  的期望等于总体系数 (真值)  $\beta$ 。

$$Y = \mathbf{X}\beta + \varepsilon,$$

$$\varepsilon = Y - \mathbf{X}\beta.$$

在最小二乘法估计量的推导中, 我们的目标是找到使平方残差最小化的  $\beta$  的值:

<sup>3</sup> 多重共线性发生在当一个或多个解释变量在多元线性回归模型中高度相关时。在方程 (3.25) 中, 如果某些解释变量  $x_i$  和  $x_j$  ( $i \neq j$ ) 之间的相关性很高, 就可能出现这种情况。

$$\begin{aligned}
 \varepsilon' \varepsilon &= (Y - \mathbf{X}\beta)'(Y - \mathbf{X}\beta) \\
 &= [Y' - (\mathbf{X}\beta)'] (Y - \mathbf{X}\beta) \\
 &= Y'Y - Y'\mathbf{X}\beta - (\mathbf{X}\beta)'Y + (\mathbf{X}\beta)' \mathbf{X}\beta.
 \end{aligned}$$

由于  $Y'\mathbf{X}\beta$  以及  $(\mathbf{X}\beta)'Y$  为常数，因此：

$$Y'\mathbf{X}\beta = (\mathbf{X}\beta)'Y$$

因此可得：

$$\varepsilon' \varepsilon = Y'Y - 2(\mathbf{X}\beta)'Y + (\mathbf{X}\beta)' \mathbf{X}\beta.$$

为了确定使我们的目标函数 ( $\varepsilon' \varepsilon$ ) 最小化的  $\beta$  的值，我们对  $\beta$  进行求导并令其等于零：

$$\frac{\partial \varepsilon' \varepsilon}{\partial \beta} = -2\mathbf{X}'Y + 2\mathbf{X}'\mathbf{X}\hat{\beta} = 0.$$

因此可得：

$$\mathbf{X}'\mathbf{X}\hat{\beta} = \mathbf{X}'Y.$$

即：

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'Y. \quad (3.27)$$

将  $\hat{\beta}$  带入  $Y$ ，可得：

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'(\mathbf{X}\beta + \varepsilon).$$

展开上式可得：

$$\hat{\beta} = \beta + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\varepsilon.$$

求期望，可得：

$$E(\hat{\beta}) = \beta.$$

由此可见，最小二乘估计量式无偏的。

3. 【线性 (Linear)】显见，最小二乘估计量  $\hat{\beta}$  可以改写为：

$$\hat{\beta} = \beta + A\varepsilon,$$

其中

$$A = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'.$$

由于  $A$  是矩阵的线性组合，所以  $\hat{\beta}$  为线性估计量。

4. 【最佳 (Best)】最佳意味着其他任何的无偏线性无偏估计量中，最小二乘估计量的方差最小。

以下为最小二乘估计量。

$$\text{Var}(\hat{\beta}) = E[(\hat{\beta} - \beta)(\hat{\beta} - \beta)'] \quad (3.28)$$

将  $\hat{\beta} - \beta = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\varepsilon$  带入式(3.28)可得:

$$\text{Var}(\hat{\beta}) = E \left\{ [(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\varepsilon] [(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\varepsilon]' \right\}$$

对上式的第二项进行转置, 可得:

$$\begin{aligned} \text{Var}(\hat{\beta}) &= E \left\{ [(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\varepsilon] [\varepsilon'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}] \right\} \\ &= E [(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\varepsilon\varepsilon'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}] \end{aligned} \quad (3.29)$$

采用最小二乘法的同方差假设 (假设 2), 进一步简化上式可得:

$$E(\varepsilon\varepsilon') = \sigma^2 \mathbf{I} \quad (3.30)$$

将式(3.30)带入式(3.29)可得:

$$\begin{aligned} \text{Var}(\hat{\beta}) &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\sigma^2\mathbf{I}\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} \\ &= \sigma^2(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} \\ &= \sigma^2(\mathbf{X}'\mathbf{X})^{-1} \end{aligned}$$

为了得到其他线性无偏估计量的方差, 定义矩阵  $C$  且  $C\mathbf{X} = \mathbf{I}$ 。此外,  $C$  满足  $C = D + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$  where  $D\mathbf{X} = 0$ 。显见,  $D = C - (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}$ , 以及  $\beta_0 = CY$ , 由此可得  $\beta_0$  是线性无偏的, 即

$$\begin{aligned} E(\beta_0) &= E(CY) \\ &= E[DY + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'Y] \\ &= E[D(\mathbf{X}\beta + \varepsilon) + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'Y] \\ &= E[DX\beta + D\varepsilon + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'Y] \\ &= 0 + 0 + \beta \end{aligned}$$

接着, 我们计算此线性无偏估计量的方差, 并将之与最小二乘估计量  $\hat{\beta}$  的方差进行对比, 可得:

$$\text{Var}(\beta_0) = E[(\beta_0 - \beta)(\beta_0 - \beta)'].$$

如果  $\beta_0 = CY$  以及  $Y = \mathbf{X}\beta + \varepsilon$ , 则

$$\begin{aligned} \beta_0 &= C(\mathbf{X}\beta + \varepsilon), \\ \beta_0 &= \beta + C\varepsilon, \\ \beta_0 - \beta &= C\varepsilon. \end{aligned}$$

将上式带入方差等式，并进一步展开可得：

$$\begin{aligned}\text{Var}(\beta_0) &= \mathbb{E}[(C\boldsymbol{\varepsilon})(C\boldsymbol{\varepsilon})'] \\ &= \mathbb{E}(C\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}'C').\end{aligned}$$

根据同方差性假设（式(3.30)），并依照  $C = D + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ ，表达式可进一步简化为：

$$\begin{aligned}\text{Var}(\beta_0) &= \sigma^2 \mathbf{I}(CC') \\ &= \sigma^2 [D + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'] [D' + \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}] \\ &= \sigma^2 [DD' + D\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'D' + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}] \\ &= \sigma^2 DD' + \sigma^2(\mathbf{X}'\mathbf{X})^{-1}.\end{aligned}$$

显见，最小二乘估计量  $\hat{\beta}$  的方差小于其他任意线性无偏估计量。

由此可得最小二乘估计量为最佳无偏线性估计量 (BLUE)。

### 3.2.0.1 F 统计量

F 统计量用于检验线性回归模型中所有自变量是否整体上对因变量具有显著解释力。

考虑如下多元线性回归模型：

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon \quad (3.31)$$

其中， $y$  为因变量， $x_1, x_2, \dots, x_k$  为  $k$  个解释变量， $\beta_0, \beta_1, \dots, \beta_k$  为待估计系数， $\varepsilon$  为误差项。

我们感兴趣的假设检验为：

原假设  $H_0$ ：

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_k = 0 \quad (3.32)$$

即所有自变量的系数为零，表示它们对因变量没有显著影响。

备择假设  $H_1$ ：至少存在一个  $\beta_j \neq 0$ ，说明至少有一个自变量对因变量有显著作用。

对应的 F 统计量定义为：

$$F = \frac{\text{SSE}/k}{\text{SSR}/(n - k - 1)} \quad (3.33)$$

其中，SSE 表示解释平方和（回归平方和），SSR 表示残差平方和， $n$  为样本容量， $k$  为自变量的个数（不含截距项）。

该检验的自由度为：分子自由度为  $k$ ，分母自由度为  $n - k - 1$ 。若计算所得的  $F$  值大于显著性水平  $\alpha$  下的临界值  $F_{\alpha}(k, n - k - 1)$ ，则拒绝原假设  $H_0$ ，说明至少有一个解释变量在统计上显著影响因变量。也可使用  $F$  检验的  $p$  值作为判据，当  $p$  值足够小，同样拒绝原假设。

值得注意的是，F 统计量与回归模型的决定系数  $R^2$  存在如下关系：

$$\begin{aligned}
 F &= \frac{n - k - 1}{k} \cdot \frac{\text{SSR}}{\text{SSE}} \\
 &= \frac{n - k - 1}{k} \cdot \frac{\text{SSR}}{\text{SST} - \text{SSR}} \\
 &= \frac{n - k - 1}{k} \cdot \frac{\text{SSR}/\text{SST}}{1 - (\text{SSR}/\text{SST})} \\
 &= \frac{n - k - 1}{k} \cdot \frac{R^2}{1 - R^2} \\
 &= \frac{R^2/k}{(1 - R^2)/(n - k - 1)}.
 \end{aligned} \tag{3.34}$$

因此，F 检验不仅反映了整体模型的拟合优度，还通过  $R^2$  与样本量和模型复杂度之间的权衡，度量所有解释变量对因变量的整体解释能力。这一关系有助于我们理解回归拟合优度与显著性检验之间的联系。

### 3.2.0.2 调整决定系数 (Adjusted $R^2$ )

在第3.1.3节中，我们介绍了  $R^2$  统计量，它是衡量回归模型拟合优度的常用指标。 $R^2$  的取值范围在 0 到 1 之间，表示模型所能解释的因变量总变异的比例。然而， $R^2$  存在一个显著局限性：随着解释变量个数  $k$  的增加， $R^2$  的值往往不会减小，即使新引入的变量对因变量几乎没有解释力。这可能导致模型在变量增多时出现“虚假的拟合提升”，从而掩盖了过拟合的风险。

为了解决这一问题，引入调整决定系数  $\bar{R}^2$  (Adjusted  $R^2$ )，它在  $R^2$  的基础上对变量数量进行了校正。其计算公式如下：

$$\bar{R}^2 = 1 - \frac{\sum \hat{\varepsilon}_i^2 / (n - k - 1)}{\sum (y_i - \bar{y})^2 / (n - 1)}. \tag{3.35}$$

其中， $\hat{\varepsilon}_i$  表示残差项， $y_i$  为因变量观测值， $\bar{y}$  为样本均值， $n$  为样本容量， $k + 1$  表示模型中估计的参数个数（包含截距项）。

$\bar{R}^2$  被称为“校正后的”决定系数，是因为它在度量模型拟合优度时，显式惩罚了解释变量个数的增加，从而在模型复杂度与样本容量之间做出平衡。当我们向模型中加入更多的预测变量时，虽然  $R^2$  可能上升，但  $\bar{R}^2$  可能会下降，反映出新增变量并未显著提升模型解释能力。因此， $\bar{R}^2$  往往更保守，是评估回归模型拟合效果和选择变量数量的重要依据，尤其在高维数据情境下，对于防止过拟合具有重要意义。

### 3.2.0.3 残差标准差 (RSE)

残差标准差 (Residual Standard Error, RSE) 是统计学中用于衡量模型对数据拟合程度的指标，也是评估线性回归模型拟合优度的重要依据之一。RSE 是对回归模型中残差（即观测值与模型预测值之间的差异）的标准差的估计，反映模型预测值与实际观测值之间差异的平均幅度。RSE 的计算公式为：

$$RSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \tag{3.36}$$

其中  $y_i$  是响应变量第  $i$  个观测值， $\hat{y}_i$  是其对应的预测值， $n$  是观测值的总数（样本量），

$p$  是模型中自变量的数量（包含截距项，故  $p = k + 1$ ）。

RSE 为模型误差的度量。它越小，模型预测的准确性越高。RSE 是一个绝对度量，其大小取决于因变量的单位和数量级，因此通常与因变量的均值相比较，以评估模型的相对误差大小。

### 3.3 异方差性与多重共线性

在多元线性回归模型 (3.25) 中，可能存在影响模型有效性和估计准确性的若干问题，包括异方差性与多重共线性。这些问题可能对回归系数的估计、标准误差以及模型整体的统计推断产生影响。

首先，考虑异方差性 (Heteroscedasticity)。在理想的多元线性回归模型假设中，误差项  $\varepsilon$  应具有恒定方差，即  $\text{Var}(\varepsilon_i) = \sigma^2$  对所有  $i$  相同。当这一假设不成立时，就出现了异方差性。在异方差性的情况下，误差项的方差依赖于解释变量，即  $\text{Var}(\varepsilon_i) = \sigma_i^2$  可能随  $x_i$  而变化。异方差性的存在会导致最小二乘估计的非有效性，虽然估计值仍是无偏的，但标准误差可能被错误估计，从而影响统计显著性的判断。

异方差性可能由以下情形引起：(1) 截面型数据结构：不同截面数据的性质和特征不同，例如在对不同国家或地区进行研究时，各自的经济规模和发展水平差异可能导致方差大小不一。(2) 重要解释变量的遗漏：若模型中忽略了某些重要解释变量，其效应可能被纳入误差项中，增加误差项的方差。(3) 模型设定错误：如将本质上非线性的关系错误设定为线性模型。(4) 经济结构变动：存在结构性断点或政策变化，可能导致模型在不同阶段的方差不同。(5) 其他经济原因：如市场心理、突发事件等也可能引起方差变化。

异方差性的识别可采用以下方法：首先，图形分析。对于一元回归，可通过绘制自变量与因变量的散点图观察方差的变化；对于多元回归，可绘制残差平方与各解释变量的散点图观察趋势。其次，统计检验。(1) 相关系数检验：检查残差的绝对值与自变量之间的相关系数；(2) White 检验：对残差平方进行回归分析，包括所有自变量的一次项、平方项及交互项；(3) Breusch-Pagan (BP) 检验：与 White 检验类似，但通常只包括自变量的一次项和平方项。

异方差性的常见处理方法包括：(1) 加权最小二乘法 (Weighted Least Squares, 简称 WLS)：通过对观测值赋予不同权重来调整方差的不一致性。(2) 稳健标准误差 (robust standard error)：使用稳健估计的标准误差（例如 HAC 标准差，见第4.3节）进行统计推断，以减轻异方差性的影响。通过上述方法，研究者可以更准确地估计模型参数，提高模型的可靠性与解释力。

多重共线性发生在一个或多个解释变量在多元线性回归模型中高度相关时。在式 (3.25) 中，若某些解释变量  $x_i$  与  $x_j$  ( $i \neq j$ ) 之间的相关性很高，就可能出现这种情况。多重共线性会导致回归系数估计不稳定、方差增大，从而降低估计值的统计显著性。尽管多重共线性不会使最小二乘 (OLS) 估计产生偏差，但会显著放大系数的标准误差，削弱对参数的推断能力。实践中可能表现为：即便预测变量与响应变量存在显著关系，部分系数仍不显著；高度相关的预测变量在不同样本中的估计差异很大；移除任一高度相关变量后，其他变量的估计系数发生明显变化，甚至出现符号错误。

令人惊叹的是，中国传统文化中的“和而不同”思想，竟能与现代计量经济学中的多重共线性产生奇妙共鸣。“和而不同”出自《论语》，倡导在和谐共处的同时保持各自独特的个性与观点。处理多重共线性时，这一思想颇具启发：模型中的各个解释变量如同社会个体，应当和谐共处、协同作用；一旦变量间相关性过高，便犹如失去“不同”的特质而过度趋同，导致信息冗余，反而削弱模型的准确性与可靠性。

正如生活中尊重个体独特性才能构建和谐多元的社会，在计量分析中也需要清楚分辨每个解释变量独立的影响，避免过度相关。借助方差膨胀因子（variance inflation factor, VIF）和条件数等方法诊断多重共线性，实质上是在寻找解释变量之间“和而不同”的平衡。VIF 衡量因预测变量相关而导致回归系数方差增加的幅度：当所有 VIF 值为 1 时，说明各变量“和而不同”，不存在多重共线性；当 VIF 值大于 5 时应提高警惕。条件数作为度量整体共线性的指标，若超过 10，通常意味着存在中等程度的共线性问题，提醒我们重视变量之间的关系，保持适度的“不同”。

这种古老思想与现代计量经济学的结合，彰显了中华优秀传统文化跨越时空的生命力与价值。它启示我们，在学习和研究计量经济学时，不仅要注重理论与技术，也要从传统文化中汲取养分、增强文化认同，让古老智慧在现代学术领域绽放新光彩，助力我们更好地理解和解决复杂的经济问题。

**定义 3.2 (方差膨胀因子 (VIF) 与条件数):** 方差膨胀因子 (*Variance Inflation Factor, VIF*) 是衡量多重共线性强度的常用指标。对于多元线性回归模型中的某个解释变量  $x_i$ ，其方差膨胀因子定义为：

$$\text{VIF}_i = \frac{1}{1 - R_i^2}.$$

其中  $R_i^2$  是以  $x_i$  为因变量、其余解释变量为自变量所建立回归模型的决定系数。若  $\text{VIF}_i = 1$ ，表示该变量与其他变量无线性相关关系，不存在多重共线性； $\text{VIF}_i$  越大，说明  $x_i$  与其他变量的线性相关性越强。通常当  $\text{VIF}_i > 5$  或  $\text{VIF}_i > 10$  时（具体阈值因研究领域而异），可认为存在较为严重的多重共线性问题。

另一种衡量多重共线性的方法是条件数 (*Condition Number*)。其计算方法为：首先构造解释变量矩阵  $\mathbf{X}$  的相关矩阵

$$\mathbf{R} = \mathbf{X}'\mathbf{X},$$

然后求解其特征值  $\lambda_1, \lambda_2, \dots, \lambda_p$ ，其中  $p = k + 1$  表示解释变量的个数（含截距项）。条件数  $\kappa$  定义为

$$\kappa = \sqrt{\frac{\lambda_{\max}}{\lambda_{\min}}}.$$

其中  $\lambda_{\max}$  和  $\lambda_{\min}$  分别是相关矩阵  $\mathbf{R}$  的最大与最小特征值。条件数越大，说明变量之间的共线性越严重。一般认为，当  $\kappa > 10$  时即存在明显的共线性问题。

当处理多元线性回归模型中的多重共线性时，可以考虑以下几种策略。首先，进行变量中心化处理，特别是在多项式拟合中，从每个预测变量中减去其均值，有助于降低变量间的线性相关强度。其次，删除模型中部分高度相关的预测变量也是有效策略。此举虽未必显著降低模型的解释力（即  $R^2$  值），但能明显提升参数估计的稳定性。实际操作中，可采用逐步回归或最优子集回归识别并移除冗余变量。最后，采用正则化方法，如岭回归或主成分回归，通过引入适度偏差换取方差减小，从而提升模型的预测精度与稳定性，尤其适用于变量众多且相关性强的复杂模型。

异方差性与多重共线性在许多计量经济学教材中均有详尽论述，此处不再赘述。感兴趣的读者可参考 Greene (2000)，该书系统讨论了包括多元线性回归在内的各类计量经济模型及相关问题的处理。另可参考 James et al. (2013)，其中对线性回归、预测误差、模型选择方法与正则化方法（如岭回归与套索回归）有清晰介绍。

## 3.4 线性回归模型相关案例

### 3.4.1 案例：汽车重量是否对燃油效率产生负面影响？

除了 `economics` 数据集之外，R 语言还自带了几个内置的数据集，可用于演示简单线性回归，这里我们采用 `mtcars` 数据集，用来研究汽车重量是否对燃油效率产生负面影响。

```

1 # 加载mtcars数据集
2 data (mtcars)
3
4 # 执行回归
5 model <- lm (mpg ~ wt, data=mtcars)
6
7 # 打印回归模型摘要
8 summary (model)
9
10 # 绘制数据和回归线
11 ggplot (mtcars, aes (x=wt, y=mpg)) +
12 geom_point () +
13 geom_smooth (method="lm", se=FALSE, color="blue")

```

回归模型摘要如下：

```

1 Call:
2 lm (formula = mpg ~ wt, data = mtcars)
3
4 Residuals:
5 Min 1Q Median 3Q Max
6 -4.5432 -2.3647 -0.1252 1.4096 6.8727
7
8 Coefficients:
9 Estimate Std. Error t value Pr (>|t|)
10 (Intercept) 37.2851 1.8776 19.858 < 2e-16 ***
11 wt -5.3445 0.5591 -9.559 1.29e-10 ***
12
13 ---
14 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
15
16 Residual standard error: 3.046 on 30 degrees of freedom
17 Multiple R-squared: 0.7528, Adjusted R-squared: 0.7446
18 F-statistic: 91.38 on 1 and 30 DF, p-value: 1.294e-10

```

回归分析是为了探究汽车重量 (`wt`) 与其燃油效率 (以每加仑英里数，即 `mpg` 计) 之间的关系。分析结果表明，汽车重量越大，其燃油效率越低，且这种关系在统计上是显著的。

图3.2 证实了汽车重量 (`wt`) 与其燃油效率之间存在显著的线性关系。

回归分析中，可视化至关重要。首先，图形表示提供了对数据模式、趋势和可能的异常值的直观理解，而这在数值摘要中可能不明显。如图3.2 可快速显示关系的强度和方向。其次，可视化对于验证统计方法的基本假设（如线性关系假设）至关重要。没有这些视觉检查，可能会从数据中得出错误的结论。

我们进一步使用 R 语言中的 `anova` 函数，对回归模型进行方差分析 (Analysis of Variance, 简称 ANOVA)，这可以帮助确定模型中的预测变量是否对响应变量存在统计上显著的影响。

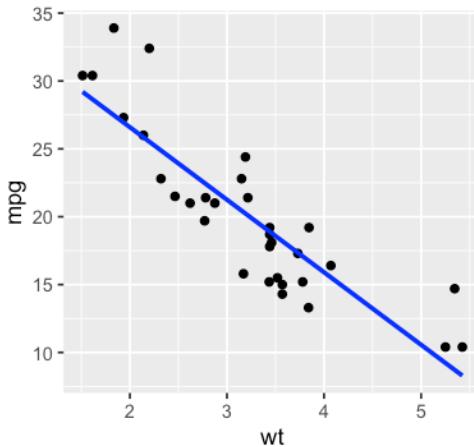


图 3.2: 汽车重量与 MPG 的关系

```

1 # 对回归模型进行 ANOVA
2 anova_result <- anova (model)
3
4 # 打印结果
5 print (anova_result)

```

结果如下：

```

1 Analysis of Variance Table
2 Response: mpg
3 Df Sum Sq Mean Sq F value Pr (>F)
4 wt 1 847.73 847.73 91.375 1.294e-10 ***
5 Residuals 30 278.32 9.28
6 ---
7 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

方差分析表评估了回归模型的整体拟合度，并检验了响应变量 (mpg) 在预测变量 (wt) 的不同水平上的均值是否存在显著差异。

1. 变异来源 (Source of Variation): 这代表了因变量方差的组成部分。

- wt: 由于汽车重量 (weight, 变量名 wt) 而导致的变异。
- 残差 (Residuals): 模型未能解释的变异 (通常称为误差)。

2. Df (自由度, Degrees of Freedom):

- wt: 1 个自由度。因为这是一个只有一个预测变量的简单线性回归模型。
- 残差 (Residuals): 30 个自由度。按总观测数  $n$  减去预测变量数  $p$  (不含截距) 再减 1 计算，即  $n - p - 1$ 。

3. 平方和 (Sum of Squares):

- 对于 wt, 为 847.73 (wt 的平方和除以其自由度)。
- 对于 残差, 为 9.28 (残差的平方和除以其自由度)。

4. F 值:

- F 值为 91.375，是 `wt` 的均方与误差项均方的比值，用于检验 `wt` 的回归系数是否等于零（即无效应）的假设。较大的 F 值表明观察到的关系并非偶然产生。

### 5. $\Pr (>F)$ :

- F 统计量的 p 值为 1.294e-10，远小于 5% 和 1% 的显著性水平，因此原假设被拒绝，这表明汽车的重量 (`wt`) 是其每加仑英里数 (`mpg`) 的一个重要预测变量。

## 3.4.2 案例：比亚迪股票的风险—收益关系

本节我们使用 CAPM 模型来研究比亚迪股票的风险—收益关系。关于 CAPM 模型的论述，详见第8.1节。比亚迪股份有限公司创立于 1995 年，是一家涵盖汽车、轨道交通、新能源和电子四大产业的高新技术企业。总部位于中国广东省深圳市，在深圳证券交易所上市。比亚迪以技术创新为核心，致力于推动可持续发展，其产品遍布全球多个国家和地区。在本节中，我们将使用比亚迪股票（股票代码：002594）的收盘价格（经后复权处理）、深证成份指数以及无风险利率；无风险利率采用银行间同业拆借加权利率的 7 天期利率。这些数据分别来源于 Wind 数据库、深圳证券交易所及中国货币网。研究的时间区间为 2022 年 1 月 4 日至 2023 年 6 月 30 日。通过 CAPM 模型的分析，我们将探讨比亚迪股票在这一期间内的风险与预期收益之间的关系。

CAPM 模型的表达式为：

$$E(R_i) = R_f + \beta_{im}(E(R_m) - R_f), \quad (3.37)$$

其中  $E(R_i)$  是比亚迪股票的预期收益率， $R_f$  是无风险利率， $\beta_{im}$  是贝塔系数，用以衡量相对于市场的波动性， $E(R_m)$  是市场的预期收益率，因此  $E(R_m) - R_f$  也被称为市场风险溢价。

我们采用 CAPM 模型的“超额收益”形式，具体来说，

$$R_{it} - R_{ft} = \alpha_i + \beta_{im}(R_{mt} - R_{ft}) + \varepsilon_t, \quad (3.38)$$

其中  $R_{it}$  是时间  $t$  的比亚迪股票收益率， $R_{ft}$  是时间  $t$  的无风险利率， $\alpha_i$  是截距项， $R_{mt}$  是时间  $t$  的市场收益率， $\varepsilon_t$  是误差项（假设为白噪声）。在这种形式下，截距项  $\alpha_i$  有一个清晰的解释：统计上显著的正  $\alpha_i$  表明资产平均而言提供了高于基于其贝塔系数和市场风险溢价所预期的回报；相反，统计上显著的负  $\alpha_i$  表明资产表现不佳。

```

1 library (ggplot2)
2 library (zoo)
3
4 # 设置工作目录
5 if (requireNamespace("rstudioapi", quietly = TRUE) &&
6 rstudioapi::isAvailable()) {
7 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
8 }
9
10 # 读取 BYD 数据
11 BYD <- read.csv ("BYD.csv")
12
13 # 显示数据的前几行
14 head (BYD)

```

```

15 BYD$Date <- as.Date (BYD$Date, format="%Y-%m-%d")
16
17
18 # 使用近似法处理NA值
19 BYD$BYD <- na.approx (BYD$BYD)
20 BYD$SZ <- na.approx (BYD$SZ)
21 BYD$Rf <- na.approx (BYD$Rf)
22
23 # 计算收益率
24 N <- length (BYD$Date)
25 BYD$BYD_RR <- c (NA, log (BYD$BYD[2:N]) - log (BYD$BYD[1: (N-1)]))
26 BYD$SZ_RR <- c (NA, log (BYD$SZ[2:N]) - log (BYD$SZ[1: (N-1)]))
27
28 # 计算超额收益率
29 BYD$ExcessReturn_SZ <- BYD$SZ_RR - BYD$Rf / 365
30
31 # 删除NA
32 BYD <- BYD[-1,]
33
34 # 创建xts时间序列对象
35 xts_BYD <- xts (BYD[,c ("BYD_RR", "ExcessReturn_SZ")], order.by=BYD$Date)
36
37 # CAPM回归
38 CAPM_model <- lm (BYD_RR ~ ExcessReturn_SZ, data = xts_BYD)
39
40 # 查看模型摘要
41 summary (CAPM_model)
42
43 # 绘制散点图和回归线
44 p <- ggplot (BYD, aes (x = ExcessReturn_SZ, y = BYD_RR)) +
45 geom_point () +
46 geom_smooth (method = "lm", col = "blue")
47
48 # 显示图形
49 print (p)
50
51 # 保存图形为PNG文件
52 ggsave ("capm_plot.png", plot = p, width = 6, height = 4, dpi = 250)

```

在 R 语言的数据分析与编程实践中，设置工作目录是确保数据文件顺利读写的重要前提。常见的做法主要有两类：其一是直接指定本地文件夹路径，如：

```

1 folder <- "Z:/金融计量经济学/R代码"
2 setwd(folder)

```

这种方式的优点在于路径清晰、可控，尤其适合文件结构固定且多台电脑共享同一网盘目录（如 OneDrive、Google Drive 或企业内网盘）的场景。只要路径正确，不依赖于开发环境，也不受脚本文件位置变化影响，适合批量化、自动化数据处理。

另一种常用方式是借助 `rstudioapi` 包自动将工作目录设置为当前 R 脚本所在位置，例如：

```

1 if (requireNamespace("rstudioapi", quietly = TRUE) && rstudioapi::
2 isAvailable()) {
3 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
4 }
```

这种自动化设定最大的优势在于便于团队协作、脚本迁移和分享。无论脚本放在哪台电脑或哪个文件夹中，运行时都能自动定位到脚本自身所在目录，避免手动修改路径，极大地提升了代码的可复用性和可移植性。尤其适合教学演示、项目开发，或将脚本与数据、结果一起打包分享的场景，能有效减少“路径找不到”的类错误。

需要注意的是，`rstudioapi` 方式要求在 RStudio 环境下运行，并且需要脚本已被保存（而非临时控制台代码）。在某些服务器环境或纯命令行下则推荐直接使用绝对路径设置。

在实际 R 语言项目中，为了增强代码的可移植性和适用性，很多时候需要自动设置当前脚本所在目录为工作目录。下面这段代码即实现了对不同运行环境的兼容和自动识别：

```

1 # 设置工作目录
2 if (requireNamespace("rstudioapi", quietly = TRUE) &&
3 rstudioapi::isAvailable() &&
4 !is.null(rstudioapi::getActiveDocumentContext()$path)) {
5 script.dir <- dirname(rstudioapi::getActiveDocumentContext()$path)
6 setwd(script.dir)
7 } else {
8 args <- commandArgs(trailingOnly = FALSE)
9 file.arg <- grep("--file=", args, value = TRUE)
10 if (length(file.arg) > 0) {
11 script.path <- normalizePath(sub("--file=", "", file.arg))
12 setwd(dirname(script.path))
13 }
14 }
```

上述代码的设计思路是：优先判断是否处于 RStudio 环境下，如果是，则利用 `rstudioapi` 包自动获取当前脚本文件的路径，并将其设为工作目录。这使得无论脚本文件被移动到哪里，数据读写操作都能在与脚本同级的目录下顺利进行，极大提升了代码的可移植性和协作友好性。

查看模型摘要：

```

1 Call:
2 lm (formula = BYD_RR ~ ExcessReturn_SZ, data = xts_BYD)
3
4 Residuals:
5 Min 1Q Median 3Q Max
6 -0.061509 -0.010366 -0.001136 0.008762 0.063269
7
8 Coefficients:
9 Estimate Std. Error t value Pr (>|t|)
10 (Intercept) 0.008144 0.001038 7.846 4.69e-14 ***
11 ExcessReturn_SZ 1.261996 0.072979 17.293 < 2e-16 ***
12 ---
13 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

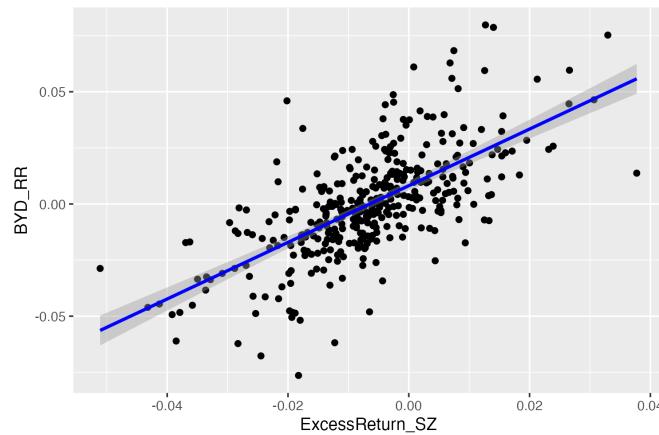


图 3.3: 比亚迪超额收益与市场超额收益的散点图及 CAPM 拟合直线

```

14
15 Residual standard error: 0.01774 on 369 degrees of freedom
16 Multiple R-squared: 0.4476, Adjusted R-squared: 0.4461
17 F-statistic: 299 on 1 and 369 DF, p-value: < 2.2e-16

```

模型摘要解释如下：

### 1. 模型概要

```

1 Call:
2 lm(formula = BYD-RR ~ ExcessReturn-SZ, data = xts_BYD)
3

```

这表明使用 BYD-RR 作为因变量, ExcessReturn-SZ 作为自变量拟合了一个线性模型。

### 2. 残差分析

残差的范围从最小值  $-0.061509$  到最大值  $0.063269$ , 中位数接近于 0。

### 3. 系数解释

- Intercept: 截距为 0.008144, 标准误差为 0.001038, t 值为 7.846。该显著截距表示当 ExcessReturn-SZ 为零时, 个股超额收益的预期值。
- ExcessReturn-SZ: 系数为 1.261996, 标准误差为 0.072979, t 值为 17.293。这表明比亚迪的系统性风险 (Beta) 高于市场平均水平 ( $\beta > 1$ )。

### 4. 模型拟合度

- Multiple R-squared:  $R^2 = 0.4476$ , 说明模型解释了约 44.76% 的因变量变异。
- Adjusted R-squared:  $\bar{R}^2 = 0.446$ 。
- F-statistic: 299, 自由度为 1 和 369,  $p < 2.2 \times 10^{-16}$ 。这表明模型在统计上显著; F 统计量用于检验“至少有一个回归系数不为零”的原假设, F 值越大越有利于拒绝原假设。

### 3.4.3 案例：Fama-French 因子模型下的个股超额收益 OLS 回归

为避免价格水平非平稳导致的伪回归风险，本节仍采用月度收益率，并将单因子的 CAPM 扩展为 Fama-French 多因子模型。个股与市场代理说明如下：AAPL 指美国苹果公司（Apple Inc.）在纳斯达克交易所的股票代码，具有较长的可得样本期与高流动性；SPY 指追踪标普 500 指数的交易所交易基金（SPDR S&P 500 ETF Trust），因覆盖面广、流动性高、跟踪误差小而被广泛用作“市场组合”的实践代理。风险因子与无风险利率（RF）均直接来自 Kenneth French 数据库的月度因子表（无需再用 TB3MS [3 个月期国债利率] 近似 RF）。

**模型设定** 以 Fama-French 三因子 + 动量（4 因子）为例：

$$R_{i,t} - R_{f,t} = \alpha + \beta_m (\text{Mkt}-\text{RF})_t + \beta_s \text{SMB}_t + \beta_h \text{HML}_t + \beta_u \text{UMD}_t + \varepsilon_t.$$

亦可使用 Fama-French 五因子 + 动量（6 因子）：

$$R_{i,t} - R_{f,t} = \alpha + \beta_m (\text{Mkt}-\text{RF})_t + \beta_s \text{SMB}_t + \beta_h \text{HML}_t + \beta_r \text{RMW}_t + \beta_c \text{CMA}_t + \beta_u \text{UMD}_t + \varepsilon_t.$$

其中  $\text{Mkt}-\text{RF}$ ,  $\text{SMB}$ ,  $\text{HML}$ ,  $\text{RMW}$ ,  $\text{CMA}$ ,  $\text{UMD}$  与  $R_f$  皆来自 Fama-French 月度因子表，单位为 % 的算术收益。为与之匹配，个股收益亦采用月度算术收益。

**数据获取与回归（R 代码，2005M1–2024M12）** 本节采用纯 base R 实现数据抓取与回归，不依赖其他第三方包：个股与市场的月度收益通过 Yahoo Finance 的 v8 chart 接口 (`interval=1mo`) 直接获取；Fama-French 因子（含 RF 以及动量 UMD）则从肯尼思·弗伦奇（Kenneth French）官方网站下载 ZIP 压缩包并解析其中的 CSV 文件。鉴于 Fama-French 因子以“百分比口径”的月度算术收益提供，个股/市场端亦统一为月度算术收益，并在回归前换算为百分比口径。为进行稳健推断，我们在 OLS 基础上使用 Newey-West (HAC) 稳健协方差（Bartlett 核）计算标准误，以便在残差同时存在异方差与自相关时仍能获得一致的方差估计与有效的  $t$  检验/置信区间。关于 LRV 的介绍，见第 4.3 节。

```

1 # ----- 0. 参数设置 -----
2 from_date <- "2005-01-01"
3 to_date <- "2024-12-31"
4 lag_nw <- 6 # Newey-West 的 Bartlett 截断阶数 (月度可取 6~12)
5
6 # ----- 1. 常用小工具 -----
7 ym_first_day <- function(xDate) {
8 # 把任意日期转换为当月第一天 (Date)
9 y <- as.integer(format(as.Date(xDate), "%Y"))
10 m <- as.integer(format(as.Date(xDate), "%m"))
11 as.Date(sprintf("%04d-%02d-01", y, m))
12 }
13
14 as_ym01_from_YYYYMM <- function(YYYYMM) {
15 # 把如 196307 或 "196307" → Date "1963-07-01"
16 s <- sprintf("%06d", as.integer(YYYYMM))
17 as.Date(paste0(substr(s, 1, 4), "-", substr(s, 5, 6), "-01"))
18 }
```

```

20 # Newey - West (HAC) 协方差 (Bartlett 核) , 纯 base R 实现
21 nw_cov <- function(X, u, lag = 0) {
22 n <- nrow(X); k <- ncol(X)
23 S <- matrix(0, k, k)
24 for (t in 1:n) {
25 xt <- X[t, , drop = FALSE]
26 S <- S + t(xt) %*% xt * (u[t]^2)
27 }
28 if (lag > 0) {
29 for (L in 1:lag) {
30 wL <- 1 - L/(lag + 1)
31 S_L <- matrix(0, k, k)
32 for (t in (L+1):n) {
33 xt <- X[t, , drop = FALSE]
34 xtL <- X[t - L, , drop = FALSE]
35 S_L <- S_L + t(xt) %*% xtL * (u[t] * u[t - L])
36 }
37 S <- S + wL * (S_L + t(S_L))
38 }
39 }
40 XtX_inv <- solve(t(X) %*% X)
41 V <- XtX_inv %*% S %*% XtX_inv
42 V
43 }
44
45 print_lm_with_nw <- function(mod, lag = 6, title = "") {
46 # 打印 OLS 系数 + Newey - West 稳健标准误/t/p
47 cat("\n=====\n", title, "\n", sep = "")
48 s <- summary(mod)
49 co <- coef(mod)
50 X <- model.matrix(mod)
51 u <- residuals(mod)
52 Vnw <- nw_cov(X, u, lag = lag)
53 se_nw <- sqrt(diag(Vnw))
54 est <- as.numeric(co)
55 tval <- est / se_nw
56 df <- nrow(X) - ncol(X)
57 pval <- 2 * pt(abs(tval), df = df, lower.tail = FALSE)
58 out <- cbind(Estimate = est, `NW_Std.Err` = se_nw, `t(NW)` = tval, `Pr(>|t|)` = pval)
59 rownames(out) <- names(co)
60 print(round(out, 6))
61 cat(sprintf("\nR-squared: %.3f Adj R-squared: %.3f RSE: %.4f N: %d\n",
62 s$r.squared, s$adj.r.squared, s$sigma, s$df[1] + s$df[2]))
63 }
64
65 # ----- 2. 从 Yahoo v8 chart 获取“月度”K线并构造月度算术收益
66 # -----
67 # 说明: 使用 v8 chart JSON (无需 crumb/cookie)。interval=1mo 返回月度序列
这里用正则从 JSON 中提取 timestamp 与 adjclose 两个数组 (纯 base R

```

```

) .

68 fetch_yahoo_chart_monthly <- function(symbol, from_date, to_date) {
69 p1 <- as.integer(as.POSIXct(as.Date(from_date), tz = "UTC"))
70 p2 <- as.integer(as.POSIXct(as.Date(to_date), tz = "UTC"))
71 url <- paste0(
72 "https://query1.finance.yahoo.com/v8/finance/chart/", symbol,
73 "?period1=", p1, "&period2=", p2,
74 "&interval=1mo&events=div%2Csplit"
75)
76 tf <- tempfile(fileext = ".json")
77 utils::download.file(url, tf, quiet = TRUE, mode = "wb")
78 js <- paste(readLines(tf, warn = FALSE), collapse = "")
79
80 # 提取 timestamp 数组
81 ts_pat <- '"timestamp"\s*:\s*\[\s*([^\]]+)\s*]'
82 ts_m <- regexpr(ts_pat, js, perl = TRUE)
83 if (ts_m[1] == -1) stop("未找到 timestamp 数组: 可能是符号无数据或网络受限。")
84 ts_txt <- regmatches(js, ts_m)
85 ts_inside <- sub('"\timestamp"\s*:\s*\[', "", sub("\]\$", "", ts_txt))
86 ts_vals <- as.numeric(unlist(strsplit(ts_inside, ",")))
87 dates <- as.POSIXct(ts_vals, origin = "1970-01-01", tz = "UTC")
88 months <- as.Date(format(dates, "%Y-%m-01"))
89
90 # 提取 adjclose 数组 (在 "indicators": {"adjclose": [{"adjclose": [...]}]}
91 # 内)
92 ac_pat <- '"adjclose"\s*:\s*\[\s*\{\s*"\adjclose"\s*:\s*\[\s*'
93 ac_m <- regexpr(ac_pat, js, perl = TRUE)
94 if (ac_m[1] == -1) stop("未找到 adjclose 数组: 结构变化或无数据。")
95 ac_txt <- regmatches(js, ac_m)
96 ac_inside <- sub('"\adjclose"\s*:\s*\[\s*\{\s*"\adjclose"\s*:\s*\[\s*',
97 ' ", "", sub("\]\$", "", ac_txt))
98 # 可能包含 "null", 用 NA 替代
99 ac_inside <- gsub("null", "NA", ac_inside, fixed = TRUE)
100 adj <- as.numeric(unlist(strsplit(ac_inside, ",")))
101
102 # 对齐长度与 NA
103 n <- min(length(months), length(adj))
104 months <- months[seq_len(n)]
105 adj <- adj[seq_len(n)]
106 ok <- !is.na(months) & !is.na(adj)
107 months <- months[ok]; adj <- adj[ok]
108
109 # 月度算术收益: Adj_t / Adj_{t-1} - 1
110 ret <- c(NA, adj[-1] / adj[-length(adj)] - 1)
111 out <- data.frame(month = months, ret = ret, stringsAsFactors = FALSE)
112 out <- out[!is.na(out$ret),]
113 out
}

```

```

114 # 获取 AAPL / SPY 月度收益
115 aapl <- fetch_yahoo_chart_monthly("AAPL", from_date, to_date)
116 spy <- fetch_yahoo_chart_monthly("SPY", from_date, to_date)
117
118 # 合并为同一表
119 rets <- merge(aapl, spy, by = "month", all = FALSE, suffixes = c("_AAPL", "_SPY"))
120 names(rets) <- c("month", "AAPL", "SPY")
121
122 # ----- 3. 从 Kenneth French 官网下载“月度”因子 ZIP 并解析 -----
123 read_ff_zip_csv <- function(zip_url, inner_name_pattern, skip_lines) {
124 tf <- tempfile(fileext = ".zip")
125 utils::download.file(zip_url, tf, mode = "wb", quiet = TRUE)
126 lst <- utils::unzip(tf, list = TRUE)$Name
127 target <- lst[grep(inner_name_pattern, lst, ignore.case = TRUE)][1]
128 csv_path <- utils::unzip(tf, files = target, exdir = tempdir(), overwrite
129 = TRUE)[1]
130 dat <- utils::read.csv(csv_path, skip = skip_lines, header = TRUE,
131 check.names = FALSE, stringsAsFactors = FALSE)
132 colnames(dat)[1] <- "Date"
133 # 只保留 YYYYMM 为纯数字的行 (剔除尾注/Annual Factors)
134 ok <- !is.na(suppressWarnings(as.integer(dat$Date)))
135 dat <- dat[ok, , drop = FALSE]
136 dat
137 }
138
139 # 5 因子 + RF (月度, 单位%)
140 ff5_url <- "https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/ftp/F-F_
141 Research_Data_5_Factors_2x3_CSV.zip"
142 ff5_raw <- read_ff_zip_csv(ff5_url, "F-F_Research_Data_5_Factors_2x3.CSV",
143 skip_lines = 3)
144 ff5 <- data.frame(
145 month = as_ym01_from_YYYYMM(ff5_raw$date),
146 Mkt_RF = as.numeric(ff5_raw[["Mkt-RF"]]),
147 SMB = as.numeric(ff5_raw[["SMB"]]),
148 HML = as.numeric(ff5_raw[["HML"]]),
149 RMW = as.numeric(ff5_raw[["RMW"]]),
150 CMA = as.numeric(ff5_raw[["CMA"]]),
151 RF = as.numeric(ff5_raw[["RF"]]),
152 stringsAsFactors = FALSE
153)
154 ff5 <- ff5[!is.na(ff5$month),]
155 ff5 <- ff5[!duplicated(ff5$month),]
156 ff5 <- ff5[ff5$month >= as.Date(from_date) & ff5$month <= as.Date(to_date),
157]
158
159 # Momentum 因子 (月度, 单位%)
160 mom_url <- "https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/ftp/F-F_
161 Momentum_Factor_CSV.zip"
162 mom_raw <- read_ff_zip_csv(mom_url, "F-F_Momentum_Factor.CSV", skip_lines =
163 13)

```

```

158 mom <- data.frame(
159 month = as_ym01_from_YYYYMM(mom_raw$Date),
160 UMD = as.numeric(mom_raw[["Mom"]]),
161 stringsAsFactors = FALSE
162)
163 mom <- mom[!is.na(mom$month),]
164 mom <- mom[!duplicated(mom$month),]
165 mom <- mom[mom$month >= as.Date(from_date) & mom$month <= as.Date(to_date),
166]
167 # 合并 5 因子与 UMD, 确保 1:1
168 ff_factors <- merge(ff5, mom, by = "month", all.x = TRUE, all.y = FALSE)
169 ff_factors <- ff_factors[order(ff_factors$month),]
170
171 # ----- 4. 合并收益与因子, 构造超额收益 (%) -----
172 # 说明: FF 因子(含 RF)为“百分比口径”; AAPL/SPY 为“小数”
173 dat <- merge(rets, ff_factors, by = "month", all = FALSE)
174 dat$Ri_excess_pct <- 100 * dat$AAPL - dat$RF # AAPL 超额收益 (%)
175 dat$Rm_excess_pct <- 100 * dat$SPY - dat$RF # SPY 超额收益 (%)
176 dat <- dat[complete.cases(dat),]
177
178 # ----- 5. 回归: CAPM / FF 4 因子 / FF 6 因子 -----
179 capm_fit <- lm(Ri_excess_pct ~ Rm_excess_pct, data = dat)
180 ff4_fit <- lm(Ri_excess_pct ~ Rm_excess_pct + SMB + HML + UMD, data = dat)
181 ff6_fit <- lm(Ri_excess_pct ~ Rm_excess_pct + SMB + HML + RMW + CMA + UMD,
182 data = dat)
183
184 # ----- 6. 输出: OLS + Newey-West 稳健 -----
185 print_lm_with_nw(capm_fit, lag = lag_nw, title = "CAPM (AAPL 月度超额收益) ")
186 print_lm_with_nw(ff4_fit, lag = lag_nw, title = "Fama-French 4 因子(含
187 UMD) ")
188 print_lm_with_nw(ff6_fit, lag = lag_nw, title = "Fama-French 6 因子(5 因
189 子 + UMD) ")
190
191 # ----- 7. 使用说明 -----
192 # - 若 Yahoo 返回空/NA, 请稍后重试(可能限流); 或缩短样本期以减少请求窗口。
193 # - 因子端是百分比, 个股/市场端是小数; 代码已在合并时统一为“%”口径做回归。
194 # - Newey-West 的 lag(本例=6)可按月度频率设为 6~12; 更高的 lag 更保守。
195 nw4; nw6

```

估计结果如下:

```

1 > # ----- 6. 输出: OLS + Newey-West 稳健 -----
2 > print_lm_with_nw(capm_fit, lag = lag_nw, title = "CAPM (AAPL 月度超额收
3 益) ")
4 =====
5 CAPM (AAPL 月度超额收益)
6 Estimate NW_Std.Err t(NW) Pr(>|t|)
7 (Intercept) 1.575746 0.483029 3.26222 0.001268
8 Rm_excess_pct 1.219974 0.094798 12.86914 0.000000
9

```

```

10 R-squared: 0.360 Adj R-squared: 0.358 RSE: 7.1578 N: 239
11 > print_lm_with_nw(ff4_fit, lag = lag_nw, title = "Fama - French 4 因子 (含
12 UMD) ")
13 =====
14 Fama - French 4 因子 (含 UMD)
15 Estimate NW_Std.Err t(NW) Pr(>|t|)
16 (Intercept) 1.428389 0.450778 3.168723 0.001735
17 Rm_excess_pct 1.311730 0.123130 10.653253 0.000000
18 SMB -0.030293 0.167474 -0.180882 0.856617
19 HML -0.677579 0.146684 -4.619309 0.000006
20 UMD 0.030832 0.158998 0.193913 0.846412
21
22 R-squared: 0.422 Adj R-squared: 0.412 RSE: 6.8483 N: 239
23 > print_lm_with_nw(ff6_fit, lag = lag_nw, title = "Fama - French 6 因子 (5
24 因子 + UMD) ")
25 =====
26 Fama - French 6 因子 (5 因子 + UMD)
27 Estimate NW_Std.Err t(NW) Pr(>|t|)
28 (Intercept) 1.271435 0.478409 2.657633 0.008416
29 Rm_excess_pct 1.298031 0.121741 10.662247 0.000000
30 SMB 0.100117 0.176136 0.568410 0.570306
31 HML -0.571982 0.186466 -3.067488 0.002415
32 RMW 0.549749 0.259180 2.121105 0.034974
33 CMA -0.360846 0.318217 -1.133961 0.257981
34 UMD 0.054781 0.158873 0.344812 0.730548
35
36 R-squared: 0.436 Adj R-squared: 0.421 RSE: 6.7957 N: 239

```

### 结果解读与比较（结合本次估计）

- **市场因子敞口 ( $\beta_m$ )**: 三组回归的市场贝塔均显著大于 1 (CAPM:  $\hat{\beta}_m = 1.220$ ; FF4: 1.312; FF6: 1.298, Newey-West  $t$  值均在 10 以上), 表明 AAPL 对**市场超额收益**的敏感度高于市场平均水平, 属于高系统性风险/高弹性资产。以 FF6 为例,  $\hat{\beta}_m = 1.298$  (NW 标准误 0.122), 对应的 95% 置信区间约为 [1.06, 1.54]。
- **风格因子负载:**
  1. **规模 (SMB)**: 在 FF4 与 FF6 中均不显著 ( $t \approx 0.57$  或更小), 与 AAPL 的**大市值特征**一致, 说明其超额收益并不依赖小盘因子。
  2. **价值 (HML)**: 显著为负 (FF4:  $-0.678$ ,  $p < 10^{-5}$ ; FF6:  $-0.572$ ,  $p \approx 0.002$ ), 表明 AAPL 具有**显著的成长因子特征** (反价值、偏成长), 该负载在多因子控制后仍稳健。
  3. **盈利能力 (RMW)**: 在 FF6 中显著为正 (0.550,  $p \approx 0.035$ ), 与 AAPL 长期较强的盈利能力相符, 说明其超额收益与高盈利组合同向。
  4. **投资 (CMA)**: 在 FF6 中为负但不显著 ( $-0.361$ ,  $p \approx 0.258$ ), 提示投资因子并非 AAPL 的主要解释来源。

5. 动量 (UMD): 在 FF4 与 FF6 中均不显著 ( $t \approx 0.19$  与  $0.34$ )，说明在控制其他因子后，AAPL 的月度超额收益不存在稳定的独立动量暴露。

- $\alpha$  (Jensen's alpha): 三种设定下  $\hat{\alpha}$  均显著为正且量级接近 (CAPM: 1.576%/月；FF4: 1.428%/月；FF6: 1.271%/月，均在 1% 或 5% 水平显著)。随着因子增多， $\alpha$  的数值与显著性有所下降 (由  $t = 3.26$  降至  $t = 2.66$ )，说明部分原本在 CAPM 下被误归入  $\alpha$  的可解释成分被风格因子 (尤以价值/盈利) 吸收；但剩余的正  $\alpha$  仍显著，可能与遗漏因子 (如质量、无形资产/研发)、样本结构变化、微观结构或交易/融资约束等因素有关，建议做子样本/滚动窗口检验以增强稳健性。
- 拟合度与残差:  $R^2$  随因子维度提升而上升 (CAPM: 0.360；FF4: 0.422；FF6: 0.436)，残差标准误 (RSE) 亦小幅下降 (约  $7.16 \rightarrow 6.85 \rightarrow 6.80$ )。这表明相较小因子 CAPM, Fama–French 多因子模型能解释更多的月度收益率波动，但仍存在较大的特质波动未被捕捉。
- 稳健推断与口径一致性: 上述  $t$  值与  $p$  值均基于 Newey–West (HAC) 稳健标准误，控制了月度收益残差中常见的异方差性与自相关性；同时我们将 AAPL/SPY 与 Fama–French 因子统一转换为按月度的算术收益率，并以百分制进行回归，确保估计与比较在口径上的一致性。关于 HAC / 长期方差 (LRV) 的原理与实现，请参见第 4.3 节。

**经济含义总结:** AAPL 对市场具有较高的系统性暴露 ( $\beta_m > 1$ )，呈现鲜明的成长特征 (HML 显著为负)，并伴随盈利因子的正向负载 (RMW 显著为正)，而规模、投资与动量负载不显著。即便在六因子模型下， $\alpha$  仍为每月约 1.27% 且显著为正 (Newey–West 调整后 95% 置信区间约 [0.33%, 2.21%])，提示可能存在超出标准因子的剩应回报，值得通过扩展因子集与稳定性检验进一步考察。

**口径与单位提示** Fama–French 因子与 RF 为月度算术收益率 (%)；本文将 AAPL、SPY 也统一为月度算术收益率并转换为百分制后回归，从而保证口径一致。若需年化，仅作量级参考 (受复利与波动影响，简单乘以 12 为便捷近似<sup>4</sup>)。

## 3.5 章节总结

本章围绕线性回归模型的建模与实证展开，以“理论—方法—案例—诊断—稳健”贯穿全章：先从一元线性回归切入，梳理线性、零均值、同方差与无自相关等基本假设，给出普通最小二乘法 (OLS) 估计、残差性质与几何含义，并介绍参数的  $t$  检验、拟合优度  $R^2$  及其分解 ( $SST = SSR + SSE$ )；通过 `mtcars` 的  $mpg \sim wt$  示例配合图形直观呈现负斜率和较强拟合。随后拓展至多元线性回归，采用矩阵形式  $Y = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$  统一表述，给出 OLS 闭式解与整体显著性的  $F$  检验，并引入调整后的决定系数  $\bar{R}^2$  与残差标准差 (RSE) 以平衡拟合与复杂度；在模型诊断层面，讨论了异方差性与多重共线性的影响、识别与对策 (如 White/BP 检验、VIF/条件数、中心化、删减与正则化等)。在金融应用方面，以比亚迪 (BYD) CAPM 为案例构造超额收益回归，解释  $\beta$  的市场暴露与  $\alpha$  的 Jensen's alpha 含义，并提示其显著性可能受样本期与遗漏因子影响；进一步指出在美国市场的 AAPL/SPY 上将 CAPM 扩展为 Fama–French 多因子模型通常可提升  $R^2$ ，且由价值与盈利等风格因子部分吸收 CAPM 下的  $\alpha$  (代码置于附录/在线材料)。为保障推断有效性，文中在 OLS 基础上配套 Newey–West (HAC) 稳健协方差 (Bartlett 核) 以应对收益残差的异方差性与自

<sup>4</sup>建议在严谨报告中使用基于复利与标准差的年化方式，并明确样本频率与换算方法。

相关性，其本质可视为对长期方差（LRV）的带核估计，理论与细节见第 4.3 节。通过本章学习，读者应能在金融数据上规范完成从一元到多元回归的建模、估计、检验与稳健推断，并在资产定价等场景中正确解读  $\alpha/\beta$  与多因子结果。

### 3.6 习题

- 假设  $(X, Y)$  服从二元正态分布，其联合密度函数为

$$f_{XY}(x, y) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left\{-\frac{1}{2(1-\rho^2)} \left[ \left(\frac{x-\mu_1}{\sigma_1}\right)^2 - 2\rho\left(\frac{x-\mu_1}{\sigma_1}\right)\left(\frac{y-\mu_2}{\sigma_2}\right) + \left(\frac{y-\mu_2}{\sigma_2}\right)^2 \right]\right\},$$

其中  $-1 < \rho < 1$ ,  $-\infty < \mu_1, \mu_2 < \infty$ ,  $0 < \sigma_1, \sigma_2 < \infty$ 。求：

- (a)  $E(Y | X)$ ;
- (b)  $\text{Var}(Y | X)$ 。

提示：可用变量替代法积分，并利用等式

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx = 1.$$

- 假设您是一名金融分析师，正在研究利率变动对股市收益率的影响。给定下表，包含过去 10 个月的利率（月度数据）与同期的股票收益率（单位：%）。

| 月份 | 月度利率 (%) | 股市收益率 (%) |
|----|----------|-----------|
| 1  | 0.5      | 1.5       |
| 2  | 0.7      | 1.7       |
| 3  | 0.8      | 1.9       |
| 4  | 1.0      | 2.1       |
| 5  | 1.2      | 2.4       |
| 6  | 1.4      | 2.6       |
| 7  | 1.6      | 2.9       |
| 8  | 1.8      | 3.1       |
| 9  | 2.0      | 3.5       |
| 10 | 2.2      | 3.7       |

您的任务是：

- (a) 使用最小二乘法（OLS）估计一元线性回归模型

$$y = \beta_0 + \beta_1 x + \varepsilon,$$

其中  $y$  为股票收益率， $x$  为利率（月度数据）。

- (b) 计算回归线的截距 ( $\beta_0$ ) 与斜率 ( $\beta_1$ ) 的估计值。
- (c) 计算该回归模型的  $R^2$  值，并解释其含义。
- (d) 讨论利率每增加 1 个百分点时，预期股票收益率将如何变化。

提示：可用下式手算回归系数与  $R^2$ ：

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}, \quad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2},$$

其中  $\bar{x}$ 、 $\bar{y}$  分别为样本均值， $\hat{y}_i$  为回归线对应的预测值。

3. 假设  $y = \beta_0^* + \beta_1^* x_1 + u$ ，其中  $y$  与  $x_1$  为随机变量， $\beta^* = (\beta_0^*, \beta_1^*)'$  为最优线性最小二乘近似系数。

(a) 证明

$$\beta_1^* = \frac{\text{Cov}(y, x_1)}{\sigma_{x_1}^2}, \quad \beta_0^* = \text{E}(y) - \beta_1^* \text{E}(x_1),$$

且均方误差

$$\text{MSE}(\beta^*) = \text{E}\left[\left(y - (\beta_0^* + \beta_1^* x_1)\right)^2\right] = \sigma_y^2(1 - \rho_{x_1 y}^2),$$

其中  $\sigma_y^2 = \text{Var}(y)$ ， $\rho_{x_1 y}$  为  $y$  与  $x_1$  的相关系数。

(b) 若  $y$  与  $x_1$  服从二元正态分布，证明

$$\text{E}(y | x_1) = \beta_0^* + \beta_1^* x_1, \quad \text{Var}(y | x_1) = \sigma_y^2(1 - \rho_{x_1 y}^2),$$

即条件均值等于最优线性最小二乘预测值，条件方差等于该预测的均方误差。

4. 假设函数  $g(X)$  用于预测  $Y$ ，评估标准为平均绝对误差 (MAE)，定义

$$\text{MAE}(g) = \text{E}[|Y - g(X)|].$$

证明最小化  $\text{MAE}(g)$  的最优解是给定  $X$  的  $Y$  的条件中位数。

5. 假设

$$y = \beta_0^o + \beta_1^o x_1 + |x_1| \varepsilon,$$

其中  $\text{E}(x_1) = 0$ ， $\text{Var}(x_1) = \sigma_{x_1}^2 > 0$ ， $\text{E}(\varepsilon) = 0$ ， $\text{Var}(\varepsilon) = \sigma_\varepsilon^2 > 0$ ，且  $\varepsilon$  与  $x_1$  相互独立， $\beta_0^o, \beta_1^o$  为常数。

(a) 求  $\text{E}(y | x_1)$ ；

(b) 求  $\text{Var}(y | x_1)$ ；

(c) 证明  $\beta_1^o = 0$  当且仅当  $\text{Cov}(x_1, y) = 0$ 。

6. 设  $X, Y$  为随机变量，且

$$\text{E}(Y | X) = 7 - \frac{1}{4}X, \quad \text{E}(X | Y) = 10 - Y.$$

计算  $X$  与  $Y$  的相关系数。

7. 假设数据生成过程为

$$y_t = \mathbf{x}'_t \boldsymbol{\beta}^o + \varepsilon_t = \beta_1^o x_{1t} + \beta_2^o x_{2t} + \varepsilon_t,$$

其中  $\mathbf{x}_t = (x_{1t}, x_{2t})'$ ,  $E(\mathbf{x}_t \mathbf{x}'_t)$  非奇异, 且  $E(\varepsilon_t | \mathbf{x}_t) = 0$ 。为简便起见, 进一步假设  $E(x_{2t}) = 0$ ,  $E(x_{1t} x_{2t}) \neq 0$ , 且  $x_{2t}$  不是  $x_{1t}$  的确定性函数 (不存在可测函数  $g(\cdot)$  使  $x_{2t} = g(x_{1t})$ ), 并且  $\beta_2^o \neq 0$ 。考虑以下一元线性回归模型

$$y_t = \beta_1 x_{1t} + u_t.$$

- (a) 证明:  $E(y_t | \mathbf{x}_t) = \mathbf{x}'_t \boldsymbol{\beta}^o \neq E(y_t | x_{1t})$ 。即该一元回归遗漏了变量  $x_{2t}$ 。
- (b) 证明: 对所有  $\beta_1 \in \mathbb{R}$ ,  $E(y_t | x_{1t}) \neq \beta_1 x_{1t}$ 。即一元线性回归是  $E(y_t | x_{1t})$  的错误设定。
- (c) 该一元模型的最小二乘估计系数  $\hat{\beta}_1$  是否等于真实参数  $\beta_1^o$ ? 请解释。

# 4 自回归移动平均模型

自回归移动平均 (autoregressive moving average, 简称 ARMA) 模型是时间序列模型的一种。时间序列是指在一段时间内以等时间间隔采集的连续数据点所组成的序列，通常用于研究时间变化规律、趋势与周期性等。它可以是一维序列（每个时间点对应一个观测值），也可以是多维序列（每个时间点对应多个变量的观测值）。时间序列广泛应用于经济学、金融学、气象学、医学等领域。

本章首先介绍一元 ARMA 模型。AR 部分 (autoregressive) 与 MA 部分 (moving average) 的结合，使模型能够刻画序列当前值受自身历史值与历史预测误差共同作用的机制。在金融计量经济学中，ARMA 模型广泛用于短期金融资产价格预测，如股票与债券价格。随后拓展至多维情形，引入向量自回归模型 (vector autoregressive model, VAR)。VAR 模型通过利用所有变量的历史值来预测各变量未来值，能够刻画多个时间序列之间的相互依赖关系，在宏观预测与金融风险管理中具有重要作用，例如分析经济政策变动对市场的影响。进一步地，在 VAR 的基础上引入结构向量自回归模型 (structural vector autoregressive model, SVAR)。SVAR 通过加入对冲击的结构性假设，更深入地揭示变量之间的动态相互作用，已广泛用于货币与财政政策冲击效应的分析。

关于市场波动率模型：一维模型如自回归条件异方差 (autoregressive conditional heteroskedasticity, 简称 ARCH) 与广义自回归条件异方差 (generalized autoregressive conditional heteroskedasticity, 简称 GARCH)；多维模型如 BEKK (Baba, Engle, Kraft 和 Kroner) 与动态条件相关 (dynamic conditional correlation, 简称 DCC) 等。这些波动率模型将在第 5 章中单独讨论。

## 4.1 ARMA 模型

ARMA 模型是一种用于描述时间序列数据的统计模型，其一般形式最早由彼得·惠特尔 (Peter Whittle) 在 1951 年的论文 “Hypothesis Testing in Time Series Analysis” (Whittle 1951) 中提出，并在乔治·E·P·博克斯 (George E. P. Box) 和格威利姆·詹金斯 (Gwilym Jenkins) 1970 年的著作 “Time Series Analysis: Forecasting and Control” (Box & Jenkins 1976) 中得到系统推广。

### 4.1.1 平稳性

时间序列分析的目的在于从历史数据中探寻背后的规律，并基于此预测未来趋势。换言之，未来的观测值在概率层面上与过去相似。用计量经济学 (统计学) 的术语讲，该时间序列是平稳的。

**定义 4.1 (严格平稳性):** 若时间序列  $Y$  的概率分布不随时间改变，则称其为严格平稳时间序列。即对任意  $k$ ,  $(Y_t, Y_{t-1}, \dots, Y_{t-k+1})$  的联合分布不依赖于  $t$ 。换句话说，任意相邻的  $k$  个观测值的联合分布 ( $k \geq 1$ ) 在时间轴上任意平移时保持不变。

在实践中，严格平稳性难以检验，因此通常采用较弱的条件。只要时间序列的均值、方差与自协方差函数不随时间而变化，即可称为“弱平稳性”或“协方差平稳性”。

**定义 4.2 (弱平稳性):** 若过程  $\{Y_t\}$  对所有  $t$  满足以下条件，则称其为弱平稳过程：

$$\mathbb{E}(Y_t) = \mu < \infty \quad (4.1)$$

$$\text{Var}(Y_t) = \mathbb{E}[(Y_t - \mu)^2] = \gamma_0 < \infty \quad (4.2)$$

$$\text{Cov}(Y_t, Y_{t-k}) = \mathbb{E}[(Y_t - \mu)(Y_{t-k} - \mu)] = \gamma_k, \quad k = 1, 2, 3, \dots \quad (4.3)$$

式 (4.3) 定义了过程  $Y_t$  滞后  $k$  阶的自协方差 (autocovariance)；式 (4.1) 与 (4.2) 要求均值与方差为有限常数。式 (4.3) 表明自协方差仅依赖于两个观测值之间的时间间隔，而与具体时点无关。

### 严平稳的时间序列一定是弱平稳的吗？



答案是否定的。柯西分布 (Cauchy distribution)，亦称洛伦兹分布 (Lorentz distribution)，其均值与方差均不存在。柯西分布以法国数学家奥古斯丁·路易·柯西 (Augustin-Louis Cauchy, 1789–1857) 命名。关于发现者的归属，学界仍存争议：斯蒂格勒 (Stigler, 2002) 第 18 章指出，1659 年费马 (Fermat) 曾从几何角度研究过具有柯西密度的曲线；该曲线后来也被称为“阿涅西的女巫” (witch of Agnesi)，因阿涅西 (Agnesi) 在其 1748 年的微积分教材中以此为例。尽管如此，泊松 (Poisson) 于 1824 年首先清晰阐述了柯西分布的性质；直到 1853 年的一场学术争论中，柯西的名字才与该分布正式关联。

柯西分布的概率密度函数为

$$f(x, x_0, \gamma) = \frac{1}{\pi \gamma \left[ 1 + \left( \frac{x-x_0}{\gamma} \right)^2 \right]},$$

其中  $x_0$  为位置参数 (中位数)， $\gamma$  为尺度参数 (半峰全宽的一半)。其形状近似钟形，但尾部较正态分布更“厚”，远离中心的概率衰减更慢。柯西分布在物理学中可用于描述共振谱线形状，在金融等领域也常作为厚尾现象的范例。一组服从柯西分布的随机变量序列是严格平稳的，但因其均值与方差不存在，不满足弱平稳的要求。

在第 5.2.1 节中，我们介绍了 GARCH 过程的弱平稳与强平稳条件。Nelson (1990b) 表明，某些 GARCH 过程可能强平稳但非弱平稳。

你还能想到其他反例吗？

### 4.1.2 遍历性和混合性条件

遍历性和混合条件对于确保统计推断与预测的稳健性和可靠性至关重要。遍历性确保时间平均数收敛至集合平均值 (ensemble averages)，为利用历史时间序列数据来推断系统更广泛的统计特性提供了理论基础。这在经济与金融领域尤为关键，因为长期预测与政策决策常基于“过去行为可用于指导未来预期”的假设。混合条件描述了时间序列“遗忘”其历史的速度：随着观测间隔的增加，不同观测之间的相关性逐渐减弱。这些条件的存在使我们能够应用中心极限定理以及其他针对相关序列的统计工具，从而推导渐近分布、构建置信区间并进行假设检验；同时，它们也是构建第 4.1 节 ARMA 模型的基础。

为了能够使用过去的数据预测未来的结果，过去与未来必须在概率层面具有一定的相似性，这一原则被囊括在平稳性的概念中。然而，当过去与未来“过于相似”时，也会出现的问题（例如非平稳或高度持久性的情形会导致伪回归等现象），因此需谨慎检验与处理。

**例 4.1：**考虑时间序列  $\{Y_t = Z\}$ ，其中  $Z$  为从  $N(0, 1)$  抽取的单一样本。显然，该过程是平稳的，因为每个  $Y_t$  都服从相同分布，其均值与方差不随时间变化。然而，由于各期  $Y_t$  并非独立抽取，而是同一个值  $Z$ ，我们无法用样本均值来估计分布的均值。

通常情况下，大数定律 (*law of large number*, 简称 *LLN*) 保证随着样本容量增大，样本均值  $\bar{Y}_T = \frac{1}{T} \sum_{t=1}^T Y_t$  收敛到期望值 (本例为 0)。但在此例中，因为每个  $Y_t$  都取相同值  $Z$ ，无论  $T$  多大，样本均值都不会收敛到分布的期望值，而是收敛到随机变量  $Z$ 。这表明：即便序列是平稳的，样本均值也未必是总体均值的相合估计。

**定义 4.3 (均值遍历性 (Ergodicity in Mean))**：给定随机过程  $\{Y_t\}$ 。先定义两种平均：

1. 集合平均 (跨样本平均)

$$\bar{Y} = \frac{1}{m} \sum_{i=1}^m Y_i.$$

2. 时间平均 (单一路径在长度为  $T$  的区间上的平均)

$$\bar{Y}_T = \frac{1}{T} \sum_{t=1}^T Y_t.$$

当  $T \rightarrow \infty$  时，若时间平均满足

$$\bar{Y}_T \xrightarrow{p} E(Y_t) = \mu \quad \text{或} \quad \bar{Y}_T \xrightarrow{a.s.} E(Y_t) = \mu,$$

则称过程  $\{Y_t\}$  在均值上是遍历的 (均值遍历性成立)。

特别地，若一个协方差 (弱) 平稳过程满足

$$\lim_{T \rightarrow \infty} \bar{Y}_T = E(Y_t) = \mu \quad (\text{依概率或几乎必然}),$$

则该过程在均值上具有遍历性。

**条件 4.1 (均值遍历的充分条件):** 若

$$\sum_{k=0}^{\infty} |\gamma_k| < \infty \quad \text{或} \quad \sum_{k=0}^{\infty} |\rho_k| < \infty,$$

则称该过程满足均值遍历性的充分条件；此时  $k \rightarrow \infty$  意味着  $\rho_k \rightarrow 0$ 。其中  $\gamma_k$  为自协方差函数在滞后  $k$  处的取值， $\rho_k$  为自相关系数。

上述条件称为绝对可和条件 (absolute summability condition)。

**定义 4.4 (自相关系数 (Autocorrelation Coefficient)):** 自相关系数用于衡量随机过程在不同时刻取值之间线性相关性的强弱。对于平稳随机过程  $\{Y_t\}$ ，其自相关系数  $\rho_k$  定义为

$$\rho_k = \frac{\gamma_k}{\gamma_0},$$

其中  $\gamma_k = \text{Cov}(Y_t, Y_{t+k})$  为  $Y_t$  与  $Y_{t+k}$  的协方差， $\gamma_0 = \text{Var}(Y_t)$  为  $Y_t$  的方差。直观地说， $\rho_k$  反映时间序列在相隔  $k$  个时间间隔时的线性关联程度； $\rho_k$  的取值范围为  $[-1, 1]$ ， $\rho_k = 1$  表示完全正相关， $\rho_k = -1$  表示完全负相关， $\rho_k = 0$  表示不存在线性相关性。

在计量经济学与时间序列分析中，自相关系数具有重要地位。它是衡量变量与其自身滞后项之间关联程度的关键指标。简单而言，若考察变量  $y$ ，自相关系数可清晰展现  $y$  与其过去取值（滞后项）之间的联系。这种联系对于分析时间序列数据的趋势、周期性与稳定性等特征至关重要，有助于洞察数据背后的规律，从而做出更为准确的预测与决策。

回望历史，我国古人虽未提出“自相关系数”这一术语，却早已蕴含与之契合的智慧。秦始皇推行“车同轨、书同文、度同衡”，将度量衡统一到同一标准。表面上是计量规范，实则为跨时期、跨地域的可比性奠定了基础：在商业活动中，统一尺度保证了不同时点交易数据的一致与可比，正如时间序列分析要求数据具备平稳与连贯。若度量衡混乱，则各期数据难以对照，更遑论识别随时间演变的规律。自相关系数的分析以这种稳定性为前提；从这个意义上说，古人统一度量衡的制度安排，正是现代统计方法（包括自相关分析）所依赖的数据基础的早期雏形，体现出跨越时空的制度智慧与方法启示。

```

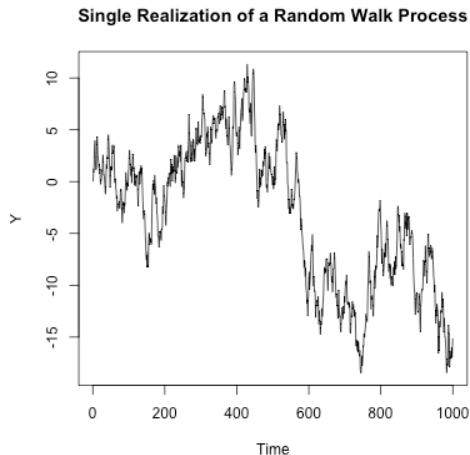
1 # 载入必要的库
2 library (ggplot2)
3
4 # 设置种子以确保结果可重现
5 set.seed (123)
6
7 # 设置工作目录
8 if (requireNamespace("rstudioapi", quietly = TRUE) &&
9 rstudioapi::isAvailable()) {
10 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
11 }
12
13 # 定义模拟参数
14 set.seed (0)
15 T <- 1000
16
17 # 模拟一个单一DGP: Y_t = Y_{t-1} + epsilon_t
18 Y <- numeric (T)

```

```

19 for (t in 2:T) {
20 Y[t] <- Y[t - 1] + rnorm (1)
21 }
22
23 # 绘制单一实现过程图
24 png ('single_realization.png', width = 400, height = 400)
25 plot (Y, type = 'l', main = '单一随机游走过程实现',
26 xlab = '时间', ylab = 'Y')
27 dev.off ()
28
29 # 通过取样本平均值进行估计
30 mu_hat <- mean (Y)
31 sigma_hat_sq <- var (Y)
32
33 # 定义样本大小和路径数
34 N <- 50
35 t_point <- 100
36
37 # 初始化数组
38 random_walks <- matrix (nrow = N, ncol = T)
39
40 # 生成 N 个随机游走
41 for (i in 1:N) {
42 epsilon <- rnorm (T) # 生成随机正态增量
43 random_walks[i,] <- cumsum (epsilon) # 累积求和以获取随机游走路径
44 }
45
46 # 根据所有路径的最小值和最大值确定 y 轴最小、最大值
47 y_min <- min (random_walks)
48 y_max <- max (random_walks)
49
50 # 绘制 50 条随机游走路径，调整 y 轴限制
51 png ('50_random_walks_plot.png', width = 400, height = 400)
52 par (mar = c (4, 4, 2, 1)) # 设置边距
53 matplot (t (random_walks) , type = 'l', lty = 1,
54 col = alpha ('black', 0.5) , ylim = c (y_min, y_max) ,
55 xlab = '时间', ylab = '随机游走', main = '50 条独立随机游走路径')
56 dev.off ()
57
58 # 在 t=100 时计算系统平均数和方差
59 ensemble_average_t100 <- mean (random_walks[, t_point])
60 ensemble_variance_t100 <- var (random_walks[, t_point])
61
62 # 打印单一实现的估计值
63 cat ("单一实现估计值:\n")
64 cat ("mu_hat:", mu_hat, "\nsigma_hat_sq:", sigma_hat_sq, "\n\n")
65
66 # 输出 t=100 时集体的结果
67 cat (sprintf ("t=100 时的系统平均数: %f\n", ensemble_average_t100))
68 cat (sprintf ("t=100 时的集体方差: %f", ensemble_variance_t100))

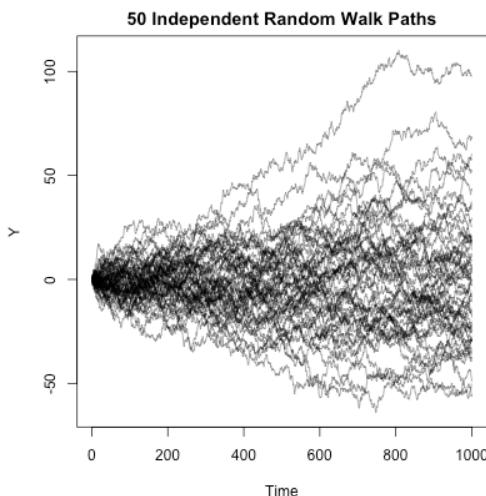
```



(a) 随机游走过程的单次实现

$$\hat{\mu} = \frac{1}{T} \sum_{t=1}^T Y_t = -3.192129$$

$$\hat{\sigma}^2 = \frac{1}{T} \sum_{t=1}^T (Y_t - \hat{\mu})^2 = 48.2165$$



(b) 50 条独立随机游走路径

$$\hat{\mu}_{100} = \frac{1}{50} \sum_{s=1}^{50} Y_{100}^s = -1.015433$$

$$\hat{\sigma}_{100}^2 = \frac{1}{50} \sum_{s=1}^{50} (Y_{100}^s - \hat{\mu}_{100})^2 = 92.785795$$

图 4.1: 随机游走过程的单次实现与 50 条独立随机游走路径的比较

对于任意可测函数  $f$  与  $g$ , 当且仅当

$$\mathbb{E}[f(X)g(Y)] = \mathbb{E}[f(X)]\mathbb{E}[g(Y)],$$

随机变量  $X$  与  $Y$  相互独立。

这是因为对于任意可测函数  $f$  与  $g$ , 有

$$\mathbb{E}[f(X)g(Y)] = \iint f(x)g(y) dF_{X,Y}(x,y),$$

其中  $F_{X,Y}(x,y)$  为  $X$  与  $Y$  的联合分布函数。若  $X$  与  $Y$  相互独立, 则其联合分布为两者边缘分布之乘积, 即

$$F_{X,Y}(x,y) = F_X(x)F_Y(y) \quad (\text{等价地, 若存在密度, 则 } f_{X,Y}(x,y) = f_X(x)f_Y(y)).$$

因此, 在  $X$  与  $Y$  的支持集上积分得

$$\mathbb{E}[f(X)g(Y)] = \int f(x) dF_X(x) \int g(y) dF_Y(y) = \mathbb{E}[f(X)]\mathbb{E}[g(Y)].$$

我们可以据此给出遍历性的一个定义。

**定义 4.5 (遍历性的一种表述):** 设  $\{Y_t\}$  为严格平稳过程。若对任意有界可测函数  $f : \mathbb{R}^k \rightarrow \mathbb{R}$  与  $g : \mathbb{R}^\ell \rightarrow \mathbb{R}$  ( $k, \ell \in \mathbb{N}$ ), 都有

$$\lim_{T \rightarrow \infty} \left\{ \mathbb{E}[f(Y_t, \dots, Y_{t+k-1})g(Y_{t+T}, \dots, Y_{t+T+\ell-1})] - \mathbb{E}[f(Y_t, \dots, Y_{t+k-1})]\mathbb{E}[g(Y_t, \dots, Y_{t+\ell-1})] \right\} = 0,$$

则称该过程在此意义下具有遍历性 (*ergodic*)。

接着, 我们介绍混合性。需要说明的是, 遍历性与混合性都是评估随机过程统计性质的重要工具, 二者虽有相近之处, 但侧重点与应用场景不同: 遍历性关注时间序列的长期行为, 尤其是时间平均是否收敛到集合平均; 混合性则衡量序列中不同时间点之间的依赖性如何随时间衰减 (或等价地, 独立性如何随时间增强)。满足遍历性的序列, 通常可以应用弱大数定律 (Weak Law of Large Numbers, 简称 WLLN) 与中心极限定理 (Central Limit Theorem, 简称 CLT) 开展统计推断; 而混合条件常用于保证更高阶统计方法的适用性, 例如在建立预测模型时, 强混合可保证估计的一致性与有效性。

若一个随机过程在足够长的时间间隔之后近似独立, 则称其满足混合条件。混合与遍历性相似, 但混合更侧重 “随时间递减的依赖性/相关性”, 而遍历性则侧重 “时间平均 (time average) 是否收敛至集合平均 (ensemble average)”。

**定义 4.6 ( $\alpha$ -混合 (强混合)):** 设  $\{X_t\}_{t=1}^\infty$  为随机变量序列, 记

$$\mathcal{F}_a^b = \sigma(X_a, X_{a+1}, \dots, X_b), \quad \mathcal{F}_{b+1}^\infty = \sigma(X_{b+1}, X_{b+2}, \dots).$$

定义强混合系数为

$$\alpha(n) = \sup_{k \in \mathbb{Z}} \sup_{A \in \mathcal{F}_{-\infty}^k, B \in \mathcal{F}_{k+n}^\infty} |\Pr(A \cap B) - \Pr(A)\Pr(B)|.$$

若随着  $n \rightarrow \infty$ ,  $\alpha(n) \rightarrow 0$ , 则称序列  $\{X_t\}$  满足强混合条件, 亦称为  $\alpha$ -混合序列。

**定义 4.7 ( $\beta$ -混合 (绝对正则性)):** 设  $\{X_t\}_{t=1}^{\infty}$  为随机变量序列, 记

$$\mathcal{F}_a^b = \sigma(X_a, X_{a+1}, \dots, X_b), \quad \mathcal{F}_{b+1}^{\infty} = \sigma(X_{b+1}, X_{b+2}, \dots).$$

定义  $\beta$ -混合系数为

$$\beta(n) = \sup_{k \in \mathbb{Z}} \text{E} \left[ \sup_{\{B_i\} \subset \mathcal{F}_{k+n}^{\infty}} \frac{1}{2} \sum_i |\Pr(B_i | \mathcal{F}_{-\infty}^k) - \Pr(B_i)| \right],$$

其中上确界取遍  $\mathcal{F}_{k+n}^{\infty}$  的有限可测划分  $\{B_i\}$ 。

若随着  $n \rightarrow \infty$ ,  $\beta(n) \rightarrow 0$ , 则称序列  $\{X_t\}$  满足  $\beta$ -混合条件 (或绝对正则性条件), 亦称为  $\beta$ -混合序列。

**定义 4.8 ( $\phi$ -混合 (phi-mixing)):** 设随机变量序列  $\{X_t\}_{t=1}^{\infty}$ , 并记

$$\mathcal{F}_a^b = \sigma(X_a, X_{a+1}, \dots, X_b), \quad \mathcal{F}_{-\infty}^k = \sigma(\dots, X_{k-1}, X_k), \quad \mathcal{F}_{k+n}^{\infty} = \sigma(X_{k+n}, X_{k+n+1}, \dots).$$

定义  $\phi$ -混合系数为

$$\phi(n) = \sup_{k \in \mathbb{Z}} \sup_{\substack{A \in \mathcal{F}_{k+n}^{\infty} \\ B \in \mathcal{F}_{-\infty}^k, \Pr(B) > 0}} |\Pr(A | B) - \Pr(A)|.$$

若随着  $n \rightarrow \infty$ ,  $\phi(n) \rightarrow 0$ , 则称序列  $\{X_t\}$  满足  $\phi$ -混合条件 (phi-mixing), 亦称为  $\phi$ -混合序列。

**定义 4.9 ( $\rho$ -混合 (rho-mixing)):** 设随机变量序列  $\{X_t\}_{t \in \mathbb{Z}}$ , 并记

$$\mathcal{F}_{-\infty}^k = \sigma(\dots, X_{k-1}, X_k), \quad \mathcal{F}_{k+n}^{\infty} = \sigma(X_{k+n}, X_{k+n+1}, \dots).$$

定义  $\rho$ -混合系数为

$$\rho(n) = \sup_{k \in \mathbb{Z}} \sup_{\substack{U \in L^2(\mathcal{F}_{-\infty}^k), V \in L^2(\mathcal{F}_{k+n}^{\infty}) \\ \text{Var}(U) > 0, \text{Var}(V) > 0}} |(U, V)| = \sup_k \sup_{U, V} \frac{|\text{Cov}(U, V)|}{\sqrt{\text{Var}(U) \text{Var}(V)}}.$$

若随着  $n \rightarrow \infty$ ,  $\rho(n) \rightarrow 0$ , 则称序列  $\{X_t\}$  满足  $\rho$ -混合条件 (rho-mixing), 亦称为  $\rho$ -混合序列。

### 直观解读

$\alpha$ -混合 (强混合) 衡量 “过去—未来” 之间最坏情形下的概率乘法偏离, 即

$$|\Pr(A \cap B) - \Pr(A) \Pr(B)| \text{ 的上确界; 当滞后 } n \text{ 增大时应收敛为 } 0$$

它刻画的是最基本的依赖衰减 (强度层级中最弱)。

$\beta$ -混合 (绝对正则性) 以全变差距离度量 “给定过去信息后的未来分布” 与 “未来边

缘分布”的差异，要求该差异随滞后增大趋于零；可理解为“过去信息”与“未来事件”之间的最大可能依赖度在时间上消散。

$\rho$ -混合以最大相关系数（在  $L^2$  可测函数类上取上确界）度量过去与未来的线性一二型依赖，要求该最大相关系数随滞后衰减至零；在经济与金融时间序列中常用来约束长期自相关的强度。

$\phi$ -混合以“最坏条件概率偏离”刻画依赖强度：

$$|\Pr(A | B) - \Pr(A)| \text{ 在所有 } A, B \text{ 上的上确界随滞后 } \rightarrow 0$$

等价于“给定过去信息时，未来观测的条件分布逐步靠拢其边缘分布”。在常见层级中， $\phi$  最强， $\beta$ 、 $\rho$  居中且一般不可比， $\alpha$  最弱。

关于混合条件的系统综述，可参见Bradley (2005)。该文在其 1986 年工作基础上进行了更新与补充，回顾了关键定义，引入了多种依赖性度量方式，并讨论了概率论中与强混合条件相关的最新进展与若干未解决问题。

### 4.1.3 自回归 (AR) 模型

时间序列是按时间顺序排列的数据点集合。虽然这些数据点通常按照固定的间隔（如每天、每月或每年）收集，但这并非绝对必要。这类序列刻画了随机变量随时间变化的趋势与模式，时间序列分析旨在揭示这些规律并据此进行预测。

自回归过程（autoregressive process，简称 AR）是时间序列分析中的基础模型，用于描述序列与其自身滞后值之间的关系。给定白噪声序列  $\varepsilon_t$ ，AR(1) 模型定义为

$$Y_t = c + \phi_1 Y_{t-1} + \varepsilon_t,$$

其中  $Y_t$  为时间序列， $c$  为常数项， $\phi_1$  为自回归系数， $\varepsilon_t$  为白噪声。

**定义 4.10 (白噪声 (white noise, WN)):** 白噪声序列是值之间不相关的随机序列，具有以下统计特性：

1.  $E(\varepsilon_t) = 0$ ;
2.  $\text{Var}(\varepsilon_t) = \sigma^2$  为常数，表示各期波动幅度相同；
3. 自协方差函数

$$\gamma(\tau) = E[\varepsilon_t \varepsilon_{t-\tau}] = \begin{cases} \sigma^2, & \text{当 } \tau = 0, \\ 0, & \text{当 } \tau \neq 0, \end{cases}$$

即除零滞后外，其余滞后处自协方差为零。

其中  $\gamma(\tau)$  为滞后  $\tau$  的自协方差函数。

“白噪声”中的“白”一词与光谱中的“白”有着密切的关联性。这种联系主要源于频谱特性，即白噪声与白光的类比。在光学中，白光包含人眼可见的所有波长（从红到紫），其光谱连续并覆盖可见光范围内的全部颜色。同理，白噪声在频谱上的特性是：各频率成分近乎均匀分布、具有恒定的功率谱密度。这意味着白噪声在所有频率上具有大致相同的能量，类似于白光中各颜色的均匀分布。因此，从这个角度看，白噪声可视作声音或信号领域的

“白光”。

AR(1) 模型可推广为高阶的 AR(p) 模型，其表达式为

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t,$$

其中  $\phi_1 \phi_2 \dots \phi_p$  为自回归系数， $c$  为常数项。

AR(p) 模型是时间序列分析中的基本模型。在金融学中，AR(p) 及其变体广泛应用于多种场景。以股票价格研究为例，在收益率预测中常可观察到短期的自回归特性；借助 AR(p) 模型，投资者能够更好地理解收益率的动态特征。又如，许多宏观经济指标（如短期利率、通货膨胀率等）也呈现明显的时间序列属性；在固定收益市场中，短期利率常以 AR(p) 模型进行建模与预测。

#### 4.1.4 移动平均 (MA) 模型

移动平均过程 (moving average process, 简称 MA) 与自回归过程不同，描述的是一个时间序列与其白噪声误差项的过去值之间的关系。

给定一个白噪声序列  $\varepsilon_t$ ，MA(1) 模型可以定义为：

$$Y_t = \mu + \theta_1 \varepsilon_{t-1} + \varepsilon_t,$$

其中  $Y_t$  是时间序列数据， $\mu$  是常数项（序列的均值）， $\theta_1$  是移动平均系数， $\varepsilon_t$  是白噪声或误差项。

与 AR(p) 模型类似，MA(1) 模型可以扩展为 MA(q) 模型，其表达式为：

$$Y_t = \mu + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t,$$

其中  $\theta_1, \theta_2, \dots, \theta_q$  是移动平均系数。

移动平均 (MA) 模型是时间序列分析中另一种基本模型。在金融学中，MA(q) 模型及其变种被广泛用于分析和预测金融市场的各种时间序列数据。例如，在分析股票的交易量或其他金融资产的波动率时，移动平均模型可以提供有用的信息。与 AR 模型结合，移动平均模型还可以形成 ARMA 模型，进一步提高预测的准确性，详见第 4.1.6 节。

本节提供如何模拟 MA(2) 过程的 R 代码示例：

```

1 # 1. 将工作目录设为当前脚本所在目录 (在 RStudio 下有效)
2 if (requireNamespace("rstudioapi", quietly = TRUE) &&
3 rstudioapi::isAvailable() &&
4 !is.null(rstudioapi::getActiveDocumentContext()$path)) {
5 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
6 }
7
8 # 2. 可复现随机种子
9 set.seed(123)
10
11 # 3) 定义 MA(2) 模拟函数: X_t = _t + 1 _{t-1} + 2 _{t-2}
12 simulate_ma2 <- function(n, theta1, theta2, sigma = 1) {
13 stopifnot(n >= 1, is.numeric(theta1), is.numeric(theta2), sigma > 0)
14 eps <- rnorm(n + 2, sd = sigma) # 预留两期噪声
15 x <- numeric(n)
16 for (i in seq_len(n)) {

```

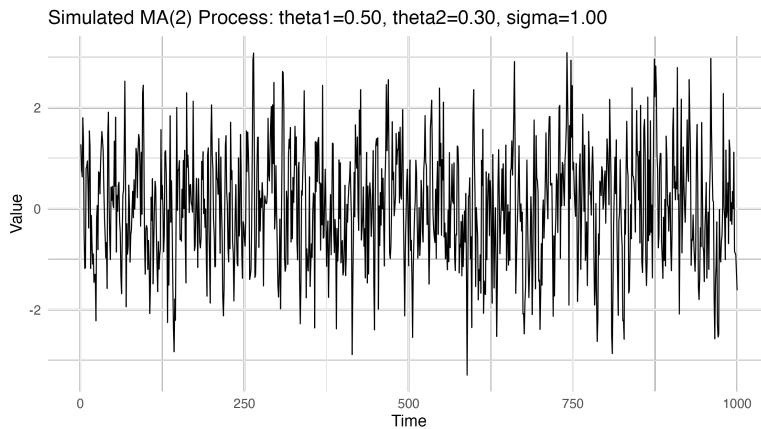


图 4.2: 模拟生成的 MA(2) 序列:  $\theta_1 = 0.5$ ,  $\theta_2 = 0.3$ , 且  $\varepsilon_t \sim N(0, 1)$

```

17 x[i] <- eps[i + 2] + theta1 * eps[i + 1] + theta2 * eps[i]
18 }
19 x
20 }
21
22 # 4. 参数与数据
23 n <- 1000 # 产生的时间序列长度
24 theta1 <- 0.5 # 第一个 MA 参数
25 theta2 <- 0.3 # 第二个 MA 参数
26 sigma <- 1 # 噪声标准差
27
28 x <- simulate_ma2(n, theta1, theta2, sigma) # 生成数据
29 df <- data.frame(time = seq_len(n), value = x) # 整理为绘图数据框
30
31 # 5. 加载绘图包并绘图
32 if (!requireNamespace("ggplot2", quietly = TRUE)) install.packages("ggplot2")
33 library(ggplot2)
34
35 p <- ggplot(df, aes(x = time, y = value)) +
36 geom_line(linewidth = 0.4) +
37 labs(
38 title = sprintf("Simulated MA(2) Process: theta1=%.2f, theta2=%.2f,
39 sigma=%.2f", theta1, theta2, sigma),
40 x = "Time", y = "Value"
41) +
42 theme_minimal(base_size = 12)
43 print(p)

```

#### 4.1.5 滞后算子 $L$

滞后算子 (lag operator), 通常记为  $L$ , 用于表示时间序列的滞后: 对序列  $\{Y_t\}$ , 定义

$$LY_t = Y_{t-1}.$$

应用滞后算子  $L$  于时间序列  $\{Y_t\}$  会得到该序列在前一时点的取值。

在时间序列分析中，滞后算子在 ARMA 模型及其扩展中发挥着重要作用，因为它提供了一种简洁的方式来表示滞后的依赖关系。通过使用滞后算子  $L$ （其中  $LY_t = Y_{t-1}$ ），AR(1) 模型  $Y_t = c + \phi_1 Y_{t-1} + \varepsilon_t$  可以更为简洁地表示：

$$(1 - \phi_1 L)Y_t = c + \varepsilon_t.$$

### 滞后算子 $L$ 的若干代数性质

1. 滞后算子可以求幂：对于  $L^p Y_t$  表示  $Y_{t-p}$ ； $L^0 Y_t = Y_t$ ， $L^{-1} Y_t = Y_{t+1}$ 。对任意常数  $\mu$ ，有  $L\mu = \mu$ 。
2. 可以用滞后算子构造多项式。例如：

$$\begin{aligned}\phi(L) &= \phi_1 L + \cdots + \phi_p L^p \\ \phi(L)Y_t &= \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p}\end{aligned}$$

3. 滞后多项式在一定条件下可逆。定义

$$(1 - \phi L)(1 - \phi L)^{-1} \equiv 1. \quad (4.4)$$

### 当 $|\phi| < 1$ 时，式 (4.4) 是否意味着

$$(1 - \phi L)^{-1} = \sum_{j=0}^{\infty} \phi^j L^j? \quad (4.5)$$

严格地讲，我们需要证明式 (4.5)。首先，要保证无穷和  $\sum_{j=0}^{\infty} \phi^j L^j$  在  $\mathcal{L}^2$  空间（平方可积函数类）上定义良好——也即它是从弱平稳序列映射到弱平稳序列的有界算子。换言之，若  $Y_t$  为弱平稳，则对所有  $t$ ，部分和  $(\sum_{j=0}^J \phi^j L^j)Y_t$  在  $\mathcal{L}^2$  中收敛于某个极限  $z_t$ ，且极限序列  $z_t$  仍为弱平稳（略）。

当  $|\phi| < 1$  时，需要检验  $\sum_{j=0}^J \phi^j L^j Y_t$  在  $\mathcal{L}^2$  中构成柯西序列。由于  $\mathcal{L}^2$  是完备空间，柯西序列必然收敛，因此存在极限  $z_t$  使得上述部分和在  $\mathcal{L}^2$  中收敛到  $z_t$ 。

最后验证

$$(1 - \phi L) \sum_{j=0}^{\infty} \phi^j L^j = \phi^0 L^0 = 1.$$

如您对  $\mathcal{L}^2$  空间与柯西序列的性质不熟悉，此部分可略读。

当然，滞后算子多项式不一定是一阶的。更高阶多项式如何求逆？一种做法是因子分解。二阶滞后算子多项式可分解为

$$1 - \alpha_1 L - \alpha_2 L^2 = (1 - \lambda_1 L)(1 - \lambda_2 L), \quad |\lambda_j| < 1 \ (j = 1, 2).$$

求逆：

$$(1 - \alpha_1 L - \alpha_2 L^2)^{-1} = (1 - \lambda_1 L)^{-1} (1 - \lambda_2 L)^{-1} = \left( \sum_{j=0}^{\infty} \lambda_1^j L^j \right) \left( \sum_{k=0}^{\infty} \lambda_2^k L^k \right) = \sum_{j=0}^{\infty} L^j \left( \sum_{k=0}^j \lambda_1^k \lambda_2^{j-k} \right).$$

需要注意，为了确保  $Y_t$  平稳，需要满足  $|\lambda_1| < 1$  且  $|\lambda_2| < 1$ 。

其次，部分分式分解也是常用方法：

$$\frac{1}{(1 - \lambda_1 x)(1 - \lambda_2 x)} = \frac{a}{1 - \lambda_1 x} + \frac{b}{1 - \lambda_2 x}, \quad a = \frac{\lambda_1}{\lambda_1 - \lambda_2}, \quad b = \frac{\lambda_2}{\lambda_2 - \lambda_1}.$$

但该技巧在下述情形才有效： $\lambda_j$  ( $j = 1, 2$ ) 互异，且  $|\lambda_1| < 1$  和  $|\lambda_2| < 1$ 。

**滞后因子多项式**:  $\alpha(L) = 1 - \alpha_1 L - \alpha_2 L^2$ , 其对应的**特征方程**为  $\alpha(z) = 1 - \alpha_1 z - \alpha_2 z^2$ 。方程的根称为**特征根**，例如  $\alpha(z) = (1 - \lambda_1 z)(1 - \lambda_2 z)$ 。若  $|\lambda_j| < 1$  则  $|z_j| > 1$ ；若  $|z_j| \leq 1$ ，则对应多项式不可逆；当特征方程的根为 1 时，相应的  $Y_t$  过程称为**单位根过程** (unit root process)。

由于滞后算子多项式的可逆性，AR 与 MA 过程在形式上可相互表示。在适当条件下（如  $|\theta| < 1$ ），AR(1) 模型

$$Y_t = \theta Y_{t-1} + \varepsilon_t$$

与 MA( $\infty$ ) 等价，即

$$Y_t = (1 - \theta L)^{-1} \varepsilon_t = \sum_{j=0}^{\infty} \theta^j L^j \varepsilon_t = \sum_{j=0}^{\infty} \theta^j \varepsilon_{t-j}.$$

#### 4.1.6 自回归移动平均过程

我们可以将自回归模型与移动平均模型结合，构成 ARMA( $p, q$ ) 模型，其定义为

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q},$$

其中  $Y_t$  为时间序列数据， $c$  为常数项， $\phi_1, \phi_2, \dots, \phi_p$  为自回归部分参数， $\varepsilon_t$  为白噪声（误差项）， $\theta_1, \theta_2, \dots, \theta_q$  为移动平均部分参数。

ARMA 模型融合了 AR 与 MA 的特征，因此在不少场景下较单独使用 AR 或 MA 模型更具优势：其一，通过结合两类结构，ARMA 能更全面地刻画复杂的时间序列；其二，相比单一模型，ARMA 往往以较少参数捕捉更丰富的动态模式；其三，在部分数据的拟合与预测中（尤其同时呈现 AR 与 MA 特征时），ARMA 表现更佳。当然，特定应用与数据下，单独的 AR 或 MA 亦可能更为合适。总体而言，模型选择应基于数据特性、研究目的及其他相关因素综合权衡。

由于 R 语言提供了生成 ARMA 过程的函数 `arima.sim`，无需自行编写生成 AR、MA 或 ARMA 过程的代码。下面的代码可用于模拟 ARMA(1,1) 过程并绘制结果；若不熟悉 `arima.sim` 的用法，可参见第 1.9 节，或通过搜索获取相关信息。

```

1 # 1) 将工作目录设为当前脚本所在目录（在 RStudio 下有效，可按需保留/删除）
2 if (requireNamespace("rstudioapi", quietly = TRUE) &&
3 rstudioapi::isAvailable() &&
4 !is.null(rstudioapi::getActiveDocumentContext()$path)) {
5 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
6 }
7
8 # 2) 导入必要的库
9 if (!requireNamespace("ggplot2", quietly = TRUE)) install.packages("ggplot2")

```

```

10 library(ggplot2) # 画图
11 # stats 包为基包， 默认已加载；此处无需额外 library(stats)
12
13 # 3) 设置随机种子，确保结果可复现
14 set.seed(12345)
15
16 # 4) 参数设置
17 n <- 1000 # 产生的时间序列长度
18 ar_param <- 0.6 # AR(1) 参数
19 ma_param <- 0.7 # MA(1) 参数
20
21 # 5) 使用 arima.sim 模拟 ARMA(1,1) 过程
22 x <- arima.sim(model = list(order = c(1, 0, 1), ar = ar_param, ma = ma_param),
 n = n)
23
24 # 6) 整理数据并绘图（大小与之前一致：8 x 4.5 英寸，300 dpi）
25 df <- data.frame(time = seq_len(n), value = as.numeric(x))
26
27 p <- ggplot(df, aes(x = time, y = value)) +
28 geom_line(linewidth = 0.4) +
29 labs(
30 title = sprintf("Simulated ARMA(1,1) Process: ar=%.2f, ma=%.2f", ar_
31 param, ma_param),
32 x = "Time",
33 y = "Value"
34) +
35 theme_minimal(base_size = 12)
36 print(p)

```

注：`stats` 是 R 语言的基础（base）包的一部分，调用 `arima.sim` 或该包中其他函数时，一般无需显式 `library(stats)`。为使代码更清晰，示例中保留了 `library(stats)`。

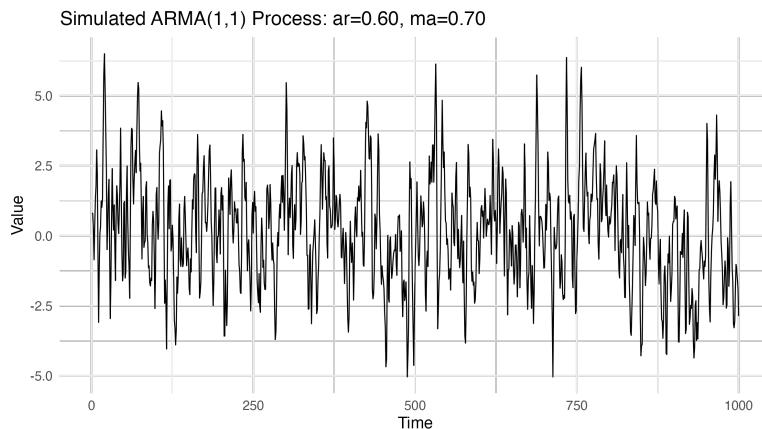


图 4.3: 模拟生成的 ARMA(1,1) 序列： $\phi_1 = 0.6$ ,  $\theta_1 = 0.7$ , 且  $\varepsilon_t \sim N(0, 1)$

### 4.1.7 自相关系数

在时间序列分析中，自协方差 (autocovariance) 和自相关 (autocorrelation) 是两个重要概念，它们帮助我们理解序列与自身在不同滞后项 (lag) 之间的关系。

**定义 4.11 (自协方差 (Autocovariance))**: 对任意整数  $k$ , 时间序列  $\{Y_t\}$  与其滞后  $k$  阶项的自协方差定义为

$$\gamma_k = \text{Cov}(Y_t, Y_{t-k}) = E[(Y_t - \mu)(Y_{t-k} - \mu)],$$

其中  $E[\cdot]$  表示期望,  $\mu$  为时间序列的无条件期望。当  $k = 0$  时, 自协方差即方差  $\text{Var}(Y_t) = \gamma_0$ 。

**定义 4.12 (自相关 (Autocorrelation))**: 对任意整数  $k$ , 自相关函数定义为

$$\rho_k = \frac{\gamma_k}{\gamma_0},$$

其中  $\gamma_k$  为滞后  $k$  阶的自协方差,  $\gamma_0$  为滞后 0 阶的自协方差 (即方差)。自相关的取值范围为  $[-1, 1]$ 。当  $\rho_k$  接近 1 或 -1 时, 表示与滞后  $k$  阶项高度正 (或负) 相关; 当  $\rho_k$  接近 0 时, 表示与滞后  $k$  阶项几乎不相关。

若将自相关系数  $\rho_k$  视为关于滞后阶数  $k$  的函数, 得到自相关函数 (autocorrelation function, 简称 ACF)。ACF 描述时间序列  $Y_t$  的跨期相关结构: 它刻画  $Y_t$  与  $Y_{t \pm k}$  之间的线性相关程度, 可用于评估序列的持续性与冲击  $\varepsilon_t$  的影响幅度与持续时间。对单变量弱平稳序列, 由于自协方差函数满足  $\gamma_{-k} = \gamma_k$ , ACF 亦满足对称性  $\rho_{-k} = \rho_k$ 。

#### 4.1.7.1 AR 模型的自相关结构

对于 AR(1) 模型:

$$Y_t = c + \phi_1 Y_{t-1} + \varepsilon_t, \quad |\phi_1| < 1, \quad \varepsilon_t \stackrel{\text{i.i.d.}}{\sim} (0, \sigma^2).$$

先推导  $\text{Var}(Y_t)$ :

$$\text{Var}(Y_t) = \phi_1^2 \text{Var}(Y_{t-1}) + \sigma^2.$$

在平稳下  $\text{Var}(Y_t) = \text{Var}(Y_{t-1}) = \gamma_0$ , 故

$$\gamma_0 = \frac{\sigma^2}{1 - \phi_1^2}.$$

一阶自相关为

$$\rho_1 = \frac{\text{Cov}(Y_t, Y_{t-1})}{\sqrt{\text{Var}(Y_t) \text{Var}(Y_{t-1})}} = \phi_1,$$

而一般滞后  $k$  的自相关

$$\rho_k = \phi_1^k \quad (k = 1, 2, \dots).$$

高阶 AR 模型的自协方差/自相关结构可由尤尔-沃克 (Yule-Walker) 方程推导。以 AR(2) 为例:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \varepsilon_t, \quad \varepsilon_t \sim (0, \sigma^2).$$

将方程两边分别与  $Y_t$ 、 $Y_{t-1}$ 、 $Y_{t-2}$  取协方差，得到

$$\gamma_0 = \phi_1\gamma_1 + \phi_2\gamma_2 + \sigma^2, \quad \gamma_1 = \phi_1\gamma_0 + \phi_2\gamma_1, \quad \gamma_2 = \phi_1\gamma_1 + \phi_2\gamma_0,$$

其中  $\gamma_k$  为滞后  $k$  阶自协方差。进一步可得对任意  $k \geq 3$  的递推式

$$\gamma_k = \phi_1\gamma_{k-1} + \phi_2\gamma_{k-2}.$$

对于一般 AR( $p$ )：

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t,$$

对  $Y_{t-k}$  与模型两边取协方差得到尤尔-沃克方程

$$\gamma_0 = \phi_1\gamma_1 + \phi_2\gamma_2 + \cdots + \phi_p\gamma_p + \sigma^2 \quad (k=0),$$

$$\gamma_k = \phi_1\gamma_{k-1} + \phi_2\gamma_{k-2} + \cdots + \phi_p\gamma_{k-p} \quad (k=1, 2, 3, \dots).$$

#### 4.1.7.2 MA 模型的自相关结构

针对 MA(1) 模型：

$$y_t = \theta \varepsilon_{t-1} + \varepsilon_t, \quad \varepsilon_t \text{ i.i.d. } (0, \sigma^2),$$

其中  $\{\varepsilon_t\}$  为白噪声。其自相关函数为

$$\rho_1 = \frac{\theta}{1+\theta^2}, \quad \rho_2 = \rho_3 = \cdots = 0.$$

针对 MA(2) 模型：

$$y_t = \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \varepsilon_t, \quad \varepsilon_t \text{ i.i.d. } (0, \sigma^2),$$

其自相关函数为

$$\rho_1 = \frac{\theta_1(1+\theta_2)}{1+\theta_1^2+\theta_2^2}, \quad \rho_2 = \frac{\theta_2}{1+\theta_1^2+\theta_2^2}, \quad \rho_k = 0 \quad (k \geq 3).$$

针对 MA( $q$ ) 模型：

$$y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}.$$

不难看出，对于  $k \leq q$ ，ACF 为非零并取决于  $\theta_1, \theta_2, \dots, \theta_q$ ；对于  $k > q$ ，ACF 为零。

```

1 library (stats)
2
3 # 设定随机种子以保证结果的可复制性
4 set.seed (123)
5
6 # 模拟MA (1) 模型的数据
7 ma1_sim <- arima.sim (n = 1000, model=list (ma=c (0.5)))
8
9 # 将输出图形保存为PNG图片

```

```
10 png (filename="ma1_acf.png", width=400, height=300)
11
12 # 计算并绘制MA (1) 模型的ACF
13 acf (ma1_sim, main="ACF for MA (1) Process")
14
15 # 结束图形输出
16 dev.off()
17
18 # 模拟MA (2) 模型的数据
19 ma2_sim <- arima.sim (n = 1000, model=list (ma=c (0.5, 0.3)))
20
21 # 将输出图形保存为PNG图片
22 png (filename="ma2_acf.png", width=400, height=300)
23
24 # 计算并绘制MA (2) 模型的ACF
25 acf (ma2_sim, main="ACF for MA (2) Process")
26
27 # 结束图形输出
28 dev.off()
```

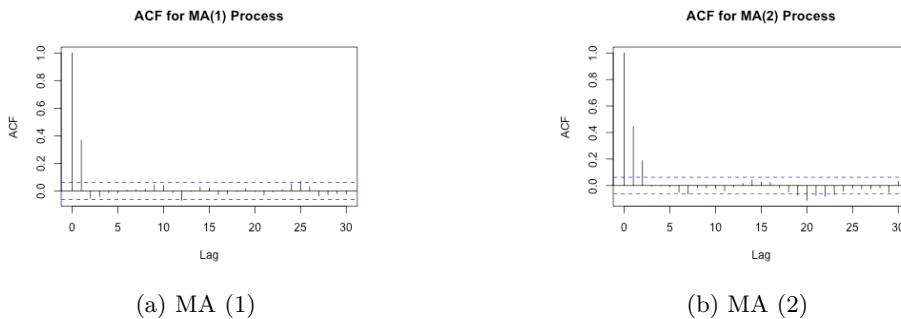


图 4.4: MA (1) 和 MA (2) 过程的自相关函数

#### 4.1.8 偏自相关系数

在时间序列分析中，偏自相关函数（partial autocorrelation function，简称 PACF）用于度量在控制所有中间滞后项后  $Y_t$  与  $Y_{t-k}$  之间的纯线性相关程度。第  $k$  阶偏自相关系数可由辅助自回归（auxiliary autoregression） $AR(k)$  中对应的系数  $\phi_k$  给出。具体而言，考虑下列  $AR(k)$  ( $k = 1, 2, \dots$ ) 模型：

1. AR(1):  $Y_t = \delta + \phi_1 Y_{t-1} + \varepsilon_t$ , 其偏自相关系数的估计为  $\hat{\phi}_{11}$ 。
  2. AR(2):  $Y_t = \delta + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \varepsilon_t$ , 其偏自相关系数的估计为  $\hat{\phi}_{22}$ 。
  3. ...
  4. AR( $k$ ): 对于更高阶模型  $Y_t = \delta + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_k Y_{t-k} + \varepsilon_t$  ( $k = 1, 2, \dots$ ), 其偏自相关系数的估计为  $\hat{\phi}_{kk}$ 。

下面我们模拟一组来自 MA(1) 模型的数据，并绘制其 PACF（偏自相关函数）。

```
1 # 加载必要的包
2 # install.packages ("forecast")
```

```

3 library (forecast)
4
5 # 设置随机种子以保证结果的可复制性
6 set.seed (123456)
7
8 # 模拟AR (1) 模型的数据
9 ar1_sim <- arima.sim (n = 1000, model=list (ar=c (0.9)))
10
11 # 将输出图形保存为PNG图片
12 png (filename="ar1_pacf.png", width=400, height=300)
13
14 # 计算并绘制AR (1) 模型的PACF
15 pacf (ar1_sim, main="PACF for AR (1) Process")
16 dev.off ()
17
18 # 模拟MA (1) 模型的数据
19 ma1_sim <- arima.sim (n = 1000, model=list (ma=c (0.9)))
20
21 # 将输出图形保存为PNG图片
22 png (filename="ma1_pacf.png", width=400, height=300)
23
24 # 计算并绘制MA (1) 模型的PACF
25 pacf (ma1_sim, main="PACF for MA (1) ")
26 dev.off ()
27
28 # 如果需要, 也可以存储PACF的数值
29 pacf_values <- pacf (ma1_sim, plot=FALSE) $acf
30 print (pacf_values)

```

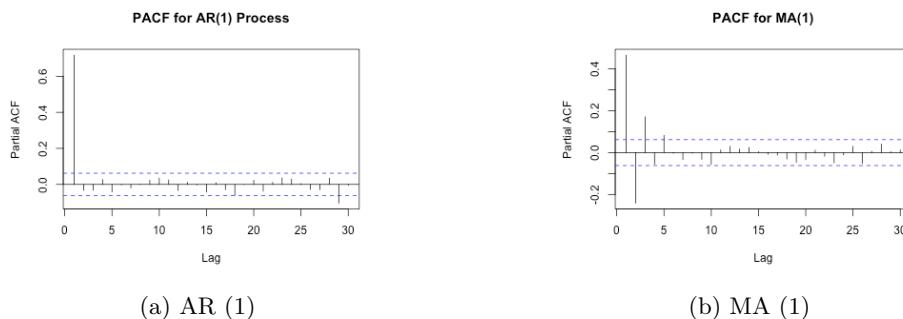


图 4.5: MA (1) 和 MA (2) 过程的偏自相关函数

#### 4.1.9 使用 ACF 与 PACF 判断 ARMA 模型的阶数

从图 4.4 与图 4.5 可见：若  $Y_t$  遵循  $\text{AR}(p)$ ，则当  $k > p$  时第  $k$  阶偏自相关  $\hat{\phi}_{kk}$  近似为 0，因而 PACF 对 AR 阶数呈“截尾”特性；而对 MA 模型，由于可表示为无穷阶 AR，PACF 一般不截尾而是随滞后逐步衰减趋零。由此，PACF 常用于判断  $\text{AR}(p)$  的阶数。

实际建模时，应同时参考自相关函数图（ACF）与偏自相关函数图（PACF）：单看其中一幅图可能导致误判。关于常用的 ACF/PACF 判别规则与对应的 ARMA 阶数组合，见表 4.1。

表 4.1: ARMA 模型的 ACF 与 PACF 判断标准

| 模型               | ACF     | PACF    |
|------------------|---------|---------|
| AR( $p$ )        | 拖尾      | $p$ 阶截尾 |
| MA( $q$ )        | $q$ 阶截尾 | 拖尾      |
| ARMA( $p, q$ )   | 拖尾      | 拖尾      |
| 其他线性过程（非 ARMA 型） | 截尾      | 截尾      |

为什么 AR(1) 过程高阶的 PACF 理论上应为 0，但在图 4.5a 中仍可见  $k > 1$  阶的 PACF 估计值不为 0？

0 是 PACF 的理论取值；但是由于样本的随机性，基于样本计算得到的 PACF 本身也是随机变量。对于连续随机变量，即使其期望为 0，实际等于 0 的概率也为 0。因此，在有限样本下看到  $k > 1$  的 PACF 估计值偏离 0 是常见的。关键在于显著性：从图 4.5a 可见高阶 ( $k > 1$ ) PACF 的估计值大都位于两条蓝色显著性界限之间，因而统计上不显著，与 AR(1) 的理论特征并不矛盾。

ACF 可以用来判断 MA 模型的阶数，PACF 可以用来判断 AR 模型的阶数。为什么 ARMA 模型不能用 ACF 以及 PACF 来判断阶数呢？ARMA 模型的阶数判定不像纯 AR 模型和纯 MA 模型那样直观，因为 ARMA 模型是 AR 和 MA 两个过程的组合。这是因为如果把 AR 部分求逆，会发现 AR 过程是可以表示为无穷阶的 MA 过程；同理，将 MA 部分求逆，会发现 MA 过程是可以表示为无穷阶的 AR 过程。因此，ARMA 过程对应的 ACF 以及 PACF 都是拖尾的。ARMA 过程的阶数，需要采用其他方法或信息准则，例如：赤池信息量准则 (Akaike information criterion，简称 AIC) 以及贝叶斯信息量准则 (Bayesian information criterion，简称 BIC)，来判断模型的最佳阶数，详见第 4.1.10 节，而不能仅依赖 ACF 和 PACF 的图形表现。

### 4.1.10 ARMA 模型的估计

本节主要介绍如何对 ARMA 模型进行估计，且主要侧重于极大似然估计方法 (maximum likelihood estimation)；随后说明如何使用 Akaike 信息准则与贝叶斯信息准则进行 ARMA 模型的定阶选择。

#### 4.1.10.1 利用尤尔–沃克方程对 AR 模型进行参数估计

在第 4.1.7.1 节，我们介绍了如何采用尤尔–沃克方程计算高阶自协方差。事实上，尤尔–沃克方程也可用于 AR 模型参数估计。以下给出 AR(2) 过程的 R 代码示例。

```

1 # 按照AR(2)模型模拟数据
2 set.seed (123456)
3 ts_data <- arima.sim (n = 1000, model=list (ar=c (0.6, -0.4)))
4
5 # 计算ACF
6 acf_vals <- acf (ts_data, plot=FALSE, lag.max=2) $acf
7
8 # 从ACF中取出需要的值
9 gamma0 <- acf_vals[1]
10 gamma1 <- acf_vals[2]
11 gamma2 <- acf_vals[3]
```

```

12 # 设置 Yule-Walker 方程
13 mat <- matrix (c (gamma0, gamma1, gamma1, gamma0) , nrow=2)
14 vec <- c (gamma1, gamma2)
15
16
17 # 求解线性方程
18 phi <- solve (mat, vec)
19
20 # 输出结果
21 print (phi)

```

输出结果如下：

```

1 > # 输出结果>
2 print (phi) [1] 0.576045 -0.410378

```

可以看出，结果与设定的真值 0.6 与 -0.4 十分接近。注意：仅当  $k > 0$  时，末项才为 0，因此此处只对  $\gamma_{k+1}$  与  $\gamma_{k+2}$  求解。

此外，R 语言提供了 `ar.yw()` 函数（`stats` 包的一部分），可直接用于 AR 模型参数的估计。

```

1 # 模拟一个AR (2) 数据
2 set.seed (123456)
3 ts_data <- arima.sim (n = 1000, model=list (ar=c (0.6, -0.4)))
4
5 # 使用 Yule-Walker 方程估计
6 ar_model <- ar.yw (ts_data, order.max=2)
7
8 # 输出结果
9 print (ar_model$ar)

```

针对 AR ( $p$ ) 阶模型的尤尔—沃克方程也可以表达为矩阵形式：

$$\begin{bmatrix} \gamma(0) & \gamma(1) & \cdots & \gamma(p-1) \\ \gamma(1) & \gamma(0) & \cdots & \gamma(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(p-1) & \gamma(p-2) & \cdots & \gamma(0) \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_p \end{bmatrix} = \begin{bmatrix} \gamma(1) \\ \gamma(2) \\ \vdots \\ \gamma(p) \end{bmatrix}.$$

类似地，第一个方程

$$\gamma(0) = \phi_1\gamma(1) + \phi_2\gamma(2) + \cdots + \phi_p\gamma(p) + \sigma^2. \quad (4.6)$$

被略去，原因是随机扰动项的方差  $\sigma^2$  未知。因此方程组通过  $k \geq 2$  的方程来求解。

#### 4.1.11 极大似然法初探

极大似然估计法（Maximum Likelihood Estimation，简称 MLE）是统计学和计量经济学中一种常用且重要的参数估计方法。MLE 具有优良的统计性质：在某些常见假设下，MLE 是一致的（当样本量趋向无穷时，估计值收敛于真实值），并且是渐近有效的（具有最小的渐近方差）。为更直观地展示极大似然估计方法，下面给出一个简例。

首先，给出极大似然函数的定义。

**定义 4.13:** 设有来自某一分布的随机样本  $\mathbf{x} = (x_1, x_2, \dots, x_n)$ , 其分布含有一个或多个未知参数, 记为向量  $\boldsymbol{\theta}$ 。则似然函数定义为

$$L(\boldsymbol{\theta} | \mathbf{x}) = f(x_1, x_2, \dots, x_n | \boldsymbol{\theta}),$$

其中  $f(x_1, x_2, \dots, x_n | \boldsymbol{\theta})$  为该样本的联合概率密度函数或联合概率质量函数 (分别对应连续型与离散型情形)。

注: 在许多文献中, 似然函数亦写作  $L(\boldsymbol{\theta})$ , 即省略“给定数据”的条件部分。下文在讨论极大似然估计的性质时, 也沿用此记法。

为什么我们需要对似然函数取对数呢? 对似然函数直接求最大值不好吗? 首先, 因为似然函数通常为乘积形式, 取对数可将连乘转化为求和, 使表达与求解更为简单。其次, 当样本量大或概率值很小时, 直接相乘易导致数值下溢, 对数似然可避免该问题。此外, 对数变换在部分情形下可将非凸优化近似为凸优化, 便于计算。更重要的是, 取对数后出现的是“和”, 便于使用大数定律与中心极限定理对估计量进行渐近分析。

针对独立同分布服从  $N(\mu, \sigma^2)$  的样本的极大似然估计。假设有随机序列, 每个元素  $X_i \sim \text{i.i.d. } N(\mu, \sigma^2)$ , 考虑一个随机样本  $x_1, x_2, \dots, x_n$ , 如何使用 MLE 来估计参数  $\boldsymbol{\theta} = (\mu, \sigma^2)$ ? 样本对应的似然函数为:

$$L(\boldsymbol{\theta} | x_1, \dots, x_n) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \quad (4.7)$$

可以看出似然函数与联合概率密度函数在形式上相似, 唯一不同的是: 似然函数是  $\boldsymbol{\theta} = (\mu, \sigma^2)$  的函数。对数似然函数为:

$$l(\boldsymbol{\theta}) = \ln L(\mu, \sigma^2 | x_1, \dots, x_n) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2 \quad (4.8)$$

通过对对数似然函数求导并令其为零, 我们可以得到 MLE:

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i \quad (4.9)$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})^2 \quad (4.10)$$

注: 式 (4.10) 为正态模型下的 MLE (有偏); 若需无偏估计, 可将分母改为  $n - 1$ 。

以下代码展示了如何通过极大似然估计法对  $\mu$  和  $\sigma^2$  进行估计:

```

1 # 使用MLE估计正态分布的参数
2
3 # 生成一些模拟数据
4 set.seed (123)
5 data <- rnorm (100, mean = 5, sd = 3)
6
7 # 定义对数似然函数
8 log_likelihood <- function (params) {
9 mu <- params[1]

```

```

10 sigma2 <- params[2]
11
12 # 注意我们取了负的对数似然，因为optim默认是进行最小化
13 n <- length (data)
14 - (-n/2 * log (2 * pi) - n/2 * log (sigma2) - 1/ (2*sigma2) * sum ((data
15 - mu) ^2))
16 }
17
18 # 优化对数似然函数以得到MLE
19 start_values <- c (mu = 0, sigma2 = 1) # 设置初始值
20 result <- optim (start_values, log_likelihood)
21
22 # 输出MLE估计值
23 cat ("MLE for mu:", result$par[1], "\n")
24 cat ("MLE for sigma^2:", result$par[2], "\n")
25
26 # 使用极大似然估计法估计均值和方差
27 mle_mu <- mean (data)
28 mle_sigma2 <- sum ((data - mle_mu) ^2) / length (data)
29
30 cat ("MLE for mu:", mle_mu, "\n")
31 cat ("MLE for sigma^2:", mle_sigma2, "\n")

```

当然，由于我们已经在等式 (4.9) 及 (4.10) 中推导得到极大似然估计量，我们也可直接采用以下 R 代码估计极大似然计量。

```

1 # MLE估计正态分布的参数
2
3 # 生成一组模拟数据
4 set.seed (123)
5 data <- rnorm (100, mean = 5, sd = 3)
6
7 # 使用极大似然估计法估计均值和方差
8 mle_mu <- mean (data)
9 mle_sigma2 <- sum ((data - mle_mu) ^2) / length (data)
10
11 cat ("MLE for mu:", mle_mu, "\n")
12 cat ("MLE for sigma^2:", mle_sigma2, "\n")

```

#### 4.1.11.1 ARMA 过程的极大似然估计

这里我们以 ARMA 模型为例，介绍 MLE 方法及其统计性质。考虑以下 ARMA(p, q) 过程：

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}. \quad (4.11)$$

其中  $\varepsilon_i \sim \text{i.i.d. } N(0, \sigma^2)$ 。我们的目的是估计总体参数向量  $\theta = (c, \phi_1, \phi_2, \dots, \phi_p, \theta_1, \theta_2, \dots, \theta_q, \sigma^2)'$ 。假设我们得到过程 (4.11) 的一组观测值  $\{y_1, y_2, \dots, y_T\}$ ；此外，为了构建极大似然函数，给

定初始值  $y_0 \equiv (y_0, y_{-1}, \dots, y_{-p+1})'$  和  $\varepsilon_0 \equiv (\varepsilon_0, \varepsilon_{-1}, \dots, \varepsilon_{-q+1})'$ , 可得

$$\begin{aligned}\varepsilon_t &= y_t - c - \phi_1 y_{t-1} - \phi_2 y_{t-2} - \cdots - \phi_p y_{t-p} \\ &\quad - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q}.\end{aligned}\tag{4.12}$$

即  $t = 1, 2, \dots, T$  时的序列  $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_T\}$ 。由于  $\varepsilon_t \sim \text{i.i.d. } N(0, \sigma^2)$ , 可得条件对数似然函数为:

$$\begin{aligned}l(\boldsymbol{\theta}) &= \log f_{Y_T, Y_{T-1}, \dots, Y_1 | \mathbf{y}_0, \varepsilon_0}(y_T, y_{T-1}, \dots, y_1 | \mathbf{y}_0, \varepsilon_0; \boldsymbol{\theta}) \\ &= -\frac{T}{2} \log(2\pi) - \frac{T}{2} \log(\sigma^2) - \sum_{t=1}^T \frac{\varepsilon_t^2}{2\sigma^2}.\end{aligned}\tag{4.13}$$

值得注意的是, 无论是如何设定初始值, 如果  $p < \infty$  以及  $q < \infty$ , 在样本容量  $T \rightarrow \infty$  时 (也就是渐近分析中), 初始值的影响可以忽略。

一般来讲, 初始值  $\mathbf{y}_0 \equiv (y_0, y_{-1}, \dots, y_{-p+1})'$  和  $\varepsilon_0 \equiv (\varepsilon_0, \varepsilon_{-1}, \dots, \varepsilon_{-q+1})'$  无法被观测到。参考 Hamilton (1994) 第 5.6 节中的做法, 我们可以令  $s = 0, -1, \dots, -p+1$  时的

$$y_s = \frac{c}{1 - \phi_1 - \phi_2 - \cdots - \phi_p},$$

以及  $s = 0, -1, \dots, -q+1$  时的  $\varepsilon_s = 0$ , 然后通过等式 (4.12) 进行迭代, 并通过 (4.13) 构造极大似然函数。或者, 直接假设  $s = 0, -1, \dots, -q+1$  时的  $\varepsilon_s = 0$ , 但  $s = 0, -1, \dots, -p+1$  时  $y_s$  等于其实际值, 等等。相应地, 极大似然函数的形式也会略有变化。但是, 不难看出, 对数极大似然函数的构建依赖  $\varepsilon_t \sim \text{i.i.d. } N(0, \sigma^2)$  这一假设, 所以形式上仍为正态分布联合密度的对数形式。值得注意的是, 无论如何设定初始值, 只要  $p < \infty$  且  $q < \infty$ , 在样本容量  $T \rightarrow \infty$  时 (即渐近分析中), 初始值的影响可以忽略。

为了方便大家理解, 我们以 ARMA (1,1) 模型为例:

$$Y_t = \phi Y_{t-1} + \varepsilon_t + \theta \varepsilon_{t-1}$$

介绍如何在 R 中采用 MLE 方法对 ARMA 模型进行估计。

```
1 # 模拟 ARMA (1,1) 数据
2 set.seed (123)
3 n <- 500
4 data <- arima.sim (n = n, model = list (ar = 0.5, ma = 0.4))
5
6 # 定义 ARMA (1,1) 的对数似然函数
7 loglik_arma11 <- function (par, data) {
8 phi <- par[1] # AR (1) 系数
9 theta <- par[2] # MA (1) 系数
10 sigma2 <- par[3]^2 # 用平方参数化, 确保 ^2 > 0
11 n <- length (data)
12 eps <- rep (0, n) # 初始化残差序列
13 for (t in 2:n) {
14 eps[t] <- data[t] - phi * data[t-1] - theta * eps[t-1]
15 }
16 sum_ll <- -n/2 * log (2*pi*sigma2) - sum (eps^2) / (2*sigma2)
17 return (-sum_ll) # 返回负值, 因为 optim () 会最小化目标函数
18}
19
```

```

20 # 优化对数似然函数
21 start_values <- c (0.5, 0.5, 1) # phi, theta和sigma的初始猜测值
22 result <- optim (par = start_values, fn = loglik_arma11, data = data,
23 method = "BFGS", hessian = TRUE)
24
25 # 输出结果
26 cat ("估计的参数:", result$par, "\n")
27 cat ("估计值处的对数似然值:", -result$value, "\n")

```

上述代码的教学功能要高于其实用性，但是按照某一特定表达式在 R 中通过编程表示出来，是 R 语言的基本功，我们建议大家在有时间的时候尝试一下 ARMA (1,2) 等高阶 ARMA 模型的估计。

如果仅从实用角度考虑，在 R 语言中有现成的函数来对 ARMA 模型进行极大似然估计。以下是简化的代码：

```

1 # 模拟ARMA (1,1) 数据
2 set.seed (123)
3 n <- 500
4 data <- arima.sim (n = n, model = list (ar = 0.5, ma = 0.4))
5
6 # 使用arima函数进行极大似然估计
7 fit <- arima (data, order = c (1, 0, 1) , method = "ML")
8
9 # 输出结果
10 print (fit)

```

### 4.1.12 ARMA 过程的统计推断

统计推断 (statistical inference) 是通过样本对总体特征作出推断的方法。在时间序列分析中，我们基于观测数据建立模型，以刻画其内在的模式、趋势、周期性等结构，并据此推断总体属性或预测未来观测。例如，我们通常需要估计 ARMA 模型的参数，利用模型进行预测，并对参数或模型结构提出并检验相应的假设。

#### 4.1.12.1 基于 AIC 和 BIC 的 ARMA(p,q) 阶数选择

在第 4.1.9 节，我们介绍了如何基于 ACF 和 PACF 选择 MA 与 AR 模型的阶数。但由于 AR 模型（或 MA 模型）与无穷阶的 MA 模型（或 AR 模型）等价，因此无法直接依靠 ACF/PACF 为 ARMA(p,q) 精确定阶。常用做法是采用信息准则进行比较，下面给出 AIC 与 BIC 的定义：

$$AIC = -2\ell(\hat{\theta}) + 2k \quad (4.14)$$

$$BIC = -2\ell(\hat{\theta}) + k \log(T) \quad (4.15)$$

其中  $\ell(\theta)$  为极大似然在估计量  $\hat{\theta}$  处的取值， $k$  为待比较模型的参数个数， $T$  为样本长度。AIC/BIC 值越小，表示模型在“拟合优度与复杂度惩罚”的综合意义下越优。

信息准则不仅考察拟合 ( $-2\ell$ )，还通过惩罚项限制不必要的复杂度：AIC 的  $2k$  在拟合与复杂之间做权衡；BIC 的  $k \log T$  随样本增大而加重惩罚，因而比 AIC 更倾向选择更简洁的模型。<sup>1</sup>

<sup>1</sup> 过拟合是指模型过度贴合样本中的噪声或偶然波动，导致样本内拟合很好而样本外表现变差。信息准则通过对参数个数加

在实际操作中,可在预设的  $p_{\max}, q_{\max}$  网格上对 ARMA(p,q) 逐一估计并计算 AIC/BIC, 选择使准则值最小的阶数组合:

$$(\hat{p}, \hat{q}) = \arg \min_{0 \leq p \leq p_{\max}, 0 \leq q \leq q_{\max}} \{\text{AIC}(p, q) \text{ 或 } \text{BIC}(p, q)\}.$$

### 简约原则 (Principle of Parsimony)

简约原则 (Principle of Parsimony) 也被称为奥卡姆剃刀原则 (Occam's Razor), 它在科学和哲学中是一个被广泛采纳的方法论原则。其基本思想是: 在所有可能的解释或模型中, 最简单的解释 (假设最少、前提或假设数量最少) 往往更可取。或者更直白地说, 当面对两个解释都足以解释某一事实或一系列观察结果时, 应选择更为简洁的那个。该原则鼓励研究者寻找最经济、最简洁的解释, 避免过度复杂和不必要的假设。尽管这并不意味着最简单的模型或理论总是最好的, 但在缺乏证据支持更复杂假设的情况下, 更为简洁的理论通常被优先考虑。

在统计学中, 奥卡姆剃刀原则常被用来解释为何我们倾向于选择简单模型, 特别是在复杂模型并未提供显著更好拟合时。例如, 使用信息准则 (如 AIC 和 BIC) 进行模型比较时, 这些准则通常会对模型复杂性进行惩罚, 这与奥卡姆剃刀原则的精神一致。

大家可以回想一下“公理”和“定理”之间的区别: 公理 (Axiom) 是一个在特定数学体系中被认为是基本的、显而易见的陈述, 不需要证明; 定理 (Theorem) 是一个可以被证明的命题。两者之中哪个更基础、更广泛适用呢? 显然是公理。定理是在公理以及特定前提条件或 (和) 其他已经被证明的定理的基础上推导得出的。

要使用 R 语言为 ARMA 模型选择最佳的阶数, 可以使用 `forecast` 包中的 `auto.arima()` 函数。该函数会基于 AIC、BIC 或其他信息准则, 在  $(p, q)$  空间自动搜索最优模型。

```

1 # 模拟ARMA(1,1) 数据
2 set.seed (123)
3 n <- 500
4 data <- arima.sim (n = n, model = list (ar = 0.5, ma = 0.4))
5
6 # 如果没有安装过forecast软件包, 第一次使用需要安装
7 # install.packages ("forecast")
8
9 library (forecast)
10
11 # 使用AIC选择ARMA的阶数
12 fit_aic <- auto.arima (data, ic="aic", stepwise=FALSE, approximation=FALSE)
13
14 # 使用BIC选择ARMA的阶数
15 fit_bic <- auto.arima (data, ic="bic", stepwise=FALSE, approximation=FALSE)
16
17 # 输出选择的模型参数
18 print (fit_aic)
19 print (fit_bic)

```

不难看出, AIC 以及 BIC 准则都选择了 ARMA (1,1) 模型, 结果如下:

```

1 > # 输出选择的模型参数
2 > print (fit_aic)

```

---

以惩罚来缓解此问题。

```

3 Series: data
4 ARIMA (1,0,1) with zero mean
5
6 Coefficients:
7 ar1 ma1
8 0.3955 0.4489
9 s.e. 0.0589 0.0592
10
11 sigma^2 = 0.9446: log likelihood = -694.57
12 AIC=1395.14 AICc=1395.19 BIC=1407.78
13 > print (fit_bic)
14 Series: data
15 ARIMA (1,0,1) with zero mean
16
17 Coefficients:
18 ar1 ma1
19 0.3955 0.4489
20 s.e. 0.0589 0.0592
21
22 sigma^2 = 0.9446: log likelihood = -694.57
23 AIC=1395.14 AICc=1395.19 BIC=1407.78

```

显然，这个结果在我们的预期范围之内，因为我们将是按照 ARMA(1,1) 模型模拟生成的序列。

### 4.1.13 极大似然估计量的优良统计性质

在第4.1.11 节中，我们简单地介绍了极大似然函数的概念以及其在 ARMA(p,q) 模型中的应用，在本章的最后部分，我们补充介绍极大似然函数的一些性质。极大似然估计量 (Maximum Likelihood Estimator, 简称 MLE) 具有很多优良的性质，它具有一致性 (Consistency)、有效性 (Efficiency) 和渐近正态性 (Asymptotic Normality)。一致性意味着当样本量增大时，极大似然估计量  $\hat{\theta}$  会依概率收敛到真值  $\theta$ 。有效性意味着极大似然估计量是最小方差无偏估计量 (Minimum Variance Unbiased Estimator, 简称 MVUE)。渐近正态性 (Asymptotic Normality) 指的是在一定的条件下，极大似然估计量随着样本量的增加会呈渐近正态分布。在本节中，我们将对这三个性质进行证明。但是，如果您的目的是 R 软件的学习，以及 ARMA 模型的应用，那么我们建议您略过本节。

#### 一致性 (Consistency) 和无偏性 (Unbiasedness) 是同一个概念吗？

答案是否。

**无偏性 (Unbiasedness)**: 一个估计量的期望值等于参数的真值，即  $E(\hat{\theta}) = \theta$ 。

**一致性 (Consistency)**: 当样本容量趋近无穷大时，估计量依概率收敛于参数的真值。因此，一个估计量可以是无偏但不一致，也可以是一致但有偏；当然也可能同时满足这这两个性质或都不满足。这两个概念都在评估估计量的质量时起关键作用，但描述的是不同的特性。考虑一个简单的均值估计。对于一个独立同分布的样本  $X_1, X_2, \dots, X_n$  来自正态分布  $N(\mu, \sigma^2)$ ，样本均值  $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$  是对  $\mu$  的无偏且一致的估计量，因

为  $E(\bar{X}) = \mu$  且当  $n \rightarrow \infty$  时,  $\bar{X} \xrightarrow{P} \mu$ 。但是, 方差的极大似然估计量

$$\hat{\sigma}_{\text{MLE}}^2 = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2$$

则是有偏但一致的 (相对于无偏的  $\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$ ), 详见等式 (4.9)。

#### 4.1.13.1 与极大似然估计相关的三个重要概念

为了更好地理解基于极大似然估计量的一致性、有效性和渐近正态性, 这里我们首先介绍基于极大似然估计的统计推断中的三个重要概念。它们分别是 **得分 (Score)**、**费希尔信息 (Fisher information)** 矩阵以及 **海森 (Hessian) 矩阵**。

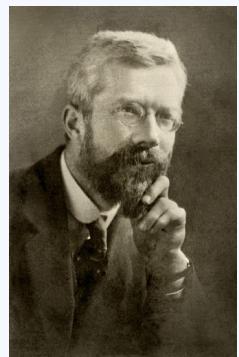
首先, 考虑**得分 (Score)**: 对数似然函数关于参数的一阶导数。我们学习微积分的时候, 知道如果想求某个函数的极值, 需要对之求一阶导数, 令一阶导数为 0 的点, 有可能就是极大值<sup>2</sup>。虽然这么说不怎么严格, 但这就是得分 (Score) 的粗浅意义。事实上, 从第 4.1.11.1 节中不难看出, 我们在对对数极大似然函数求最大值时的第一步, 就是先对对数似然求一阶导数。

当然, 针对得分 (Score) 的解释也可以更严谨一些。假设随机样本  $X_1, X_2, \dots, X_T$  都是独立同分布 (i.i.d.) 的<sup>3</sup>, 并从给定的概率密度 (或概率质量函数)  $f(X; \theta)$  中抽取。不难得到

$$\begin{aligned} \mathbb{E}\left[\frac{\partial}{\partial \theta} \log f(X; \theta) \mid \theta\right] &= \int_{\mathbb{R}} \frac{\partial}{\partial \theta} \log f(x; \theta) f(x; \theta) dx \\ &= \int_{\mathbb{R}} \frac{1}{f(x; \theta)} \frac{\partial}{\partial \theta} f(x; \theta) f(x; \theta) dx \\ &= \frac{\partial}{\partial \theta} \int_{\mathbb{R}} f(x; \theta) dx \\ &= \frac{\partial}{\partial \theta} 1 = 0. \end{aligned}$$

其次, **费希尔 (Fisher) 信息**。费希尔信息等式是极大似然估计中的核心概念; 费希尔信息衡量可观测随机变量  $X$  就未知参数  $\theta$  所“携带”的信息量。

Fisher 信息是以著名统计学家、遗传学家罗纳德 · 费希尔 (Ronald Fisher) 命名的。罗纳德 · 费希尔是 20 世纪统计学领域的杰出人物。由于他在统计学方面的工作, 他被描述为“几乎独自创建了现代统计科学基础的天才” (Hald 1998) 以及“20 世纪统计学中最重要的人物” (Efron 1998)。罗纳德 · 费希尔不仅在统计学领域取得了卓越的成就, 他对现代生物学的核心理论也做出了重要贡献。他的著作《天择的遗传理论 (The Genetical Theory of Natural Selection)》为现代生物学奠定了坚实的基础, 明确提出孟德尔的遗传定律与达尔文的自然选择理论相辅相成。费希尔认为, 演化的主要驱动力是自然选择而非突变。这一观点以及他将统计方法应用于



<sup>2</sup>当我们确定函数  $f(x)$  在某一区间上的极值时, 一般对  $f(x)$  求一阶导数 (记为  $f'(x)$ ), 再通过二阶导数  $f''(x)$  判断临界点是极大、极小还是拐点: 若在某临界点  $x_0$ ,  $f''(x_0) > 0$  则为 (局部) 极小值; 若  $f''(x_0) < 0$  则为 (局部) 极大值; 若  $f''(x_0) = 0$ , 则二阶导数测试失效, 需结合其他信息判断。除临界点外, 还需考虑边界点。

<sup>3</sup>事实上, 在 ARMA 模型中,  $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_T$  也是 i.i.d. 的, 且在进行极大似然估计时, 通常假设  $\varepsilon_t \sim \text{i.i.d. } N(0, \sigma^2)$ 。

演化论研究的做法，为解释和验证达尔文的理论提供了重要支持。

Fisher 信息被定义为得分函数的方差：

$$\mathcal{I}(\boldsymbol{\theta}) = \text{E} \left[ \left( \frac{\partial}{\partial \boldsymbol{\theta}} \log f(X; \boldsymbol{\theta}) \right)^2 \mid \boldsymbol{\theta} \right] = \int_{\mathbb{R}} \left( \frac{\partial}{\partial \boldsymbol{\theta}} \log f(x; \boldsymbol{\theta}) \right)^2 f(x; \boldsymbol{\theta}) dx$$

不难看出， $0 \leq \mathcal{I}(\boldsymbol{\theta})$ 。注意  $0 \leq \mathcal{I}(\boldsymbol{\theta})$ 。高 Fisher 信息的随机变量意味着得分的绝对值经常很高。Fisher 信息不是某个特定观测值的函数，因为随机变量  $X$  已经被平均掉了。

如果  $\log f(x; \boldsymbol{\theta})$  关于  $\boldsymbol{\theta}$  是二次可导的，在某些正则性条件下，Fisher 信息矩阵也可以写为<sup>4</sup>：

$$\mathcal{I}(\boldsymbol{\theta}) = -\text{E} \left[ \frac{\partial^2}{\partial \boldsymbol{\theta}^2} \log f(X; \boldsymbol{\theta}) \mid \boldsymbol{\theta}, \right] \quad (4.16)$$

原因在于

$$\frac{\partial^2}{\partial \boldsymbol{\theta}^2} \log f(X; \boldsymbol{\theta}) = \frac{\frac{\partial^2}{\partial \boldsymbol{\theta}^2} f(X; \boldsymbol{\theta})}{f(X; \boldsymbol{\theta})} - \left( \frac{\partial}{\partial \boldsymbol{\theta}} \log f(X; \boldsymbol{\theta}) \right)^2$$

以及

$$\text{E} \left[ \frac{\frac{\partial^2}{\partial \boldsymbol{\theta}^2} f(X; \boldsymbol{\theta})}{f(X; \boldsymbol{\theta})} \mid \boldsymbol{\theta} \right] = \frac{\partial^2}{\partial \boldsymbol{\theta}^2} \int_{\mathbb{R}} f(x; \boldsymbol{\theta}) dx = 0$$

最后，海森 (Hessian) 矩阵，其定义是对数极大似然函数的二阶导数，即：

$$H(\boldsymbol{\theta}) = \frac{\partial^2}{\partial \boldsymbol{\theta}^2} \log f(X; \boldsymbol{\theta}). \quad (4.17)$$

从等式 (4.16) 和 (4.17) 中不难看出，Fisher 信息的负值是对数似然函数的 Hessian 矩阵的期望值。

与 Fisher 信息相关的一个概念是 Cramér–Rao 下界 (Cramér–Rao lower bound, 简称 CRLB)。首先，我们介绍 Cramér–Rao 不等式，其描述了参数估计的最低方差：

$$\text{Var}(\hat{\boldsymbol{\theta}}) \geq \frac{1}{\mathcal{I}(\boldsymbol{\theta})} \quad (4.18)$$

在模型正确设定且满足常见正则条件时，极大似然估计量 (MLE) 是渐近有效的，因此其（渐近）方差达到该下界：

$$\text{avar}(\hat{\boldsymbol{\theta}}) = \mathcal{I}(\boldsymbol{\theta})^{-1}. \quad (4.19)$$

首先，我们引入经验损失函数 (Empirical Loss Function)：

$$R_T(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}) = \frac{1}{T} \sum_{i=1}^T \log \frac{f(X_i; \boldsymbol{\theta})}{f(X_i; \hat{\boldsymbol{\theta}})}$$

**定义 4.14 (损失函数 (Loss Function)):** 损失函数也被称为代价函数 (*Cost Function*)，在机器学习和统计学中，是用来估量模型的预测值与真实值之间的差异的一个函数。通过最小化这个函数，我们可以优化模型的参数来获得更好的预测效果。

虽然  $f(X, \boldsymbol{\theta})$  是未知的，但是它可以被视为常数，因此对上述损失函数  $R_T(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta})$  的最小化与对极大似然函数的最大化是等价的。

<sup>4</sup>Lehmann & Casella (2006)，公式 (2.5.16)，引理 5.3

熟悉统计学的读者不难看出,  $R_T(\hat{\theta}, \theta)$  对应的总体风险函数是:

$$R(\hat{\theta}, \theta) = E_{\theta} \log \frac{f(X; \theta)}{f(X; \hat{\theta})} = \int_{\mathbb{R}} f(x; \theta) \log \left( \frac{f(x; \theta)}{f(x; \hat{\theta})} \right) dx.$$

这就是我们熟知的 Kullback-Leibler 散度 (Kullback-Leibler divergence), 简称 KL 散度, 也称相对熵 (relative entropy)。注意  $E_{\theta}$  指的是期望是在某个特定的参数  $\theta$  下计算的。例如: 有一个随机变量  $X$ , 其概率分布是由参数  $\theta$  定义的, 那么  $E_{\theta}$  表示的是在这个分布下  $X$  的期望。这里, 我们定义积分范围为  $\mathbb{R}$ , 是因为  $X$  的支撑集 (support)<sup>5</sup> 一定是  $\mathbb{R}$  的子集。

**定义 4.15 (KL 散度):** 是用于衡量两个概率分布间差异的非对称性度量。给定两个离散概率分布  $P$  和  $Q$ , 其 KL 散度定义为:

$$KL(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}.$$

如果分布  $p$  和  $q$  是连续的, 则

$$KL(P||Q) = \int p(x) \log \frac{p(x)}{q(x)} dx.$$

这里, 我们没有定义求和和积分的范围, 但默认求和以及积分覆盖整个支撑集。

不难看出, 经验损失函数是一组独立同分布随机变量的和。因此, 应用大数定律, 我们可得对于任意的  $\theta$  有

$$R_T(\tilde{\theta}, \theta) \xrightarrow{p} R(\tilde{\theta}, \theta).$$

**条件 4.2 (可识别性):** 构建任何一致估计量的基本要求是模型必须是可识别的, 即如果  $\theta_1 \neq \theta_2$ , 那么必定有  $f(x; \theta_1) \neq f(x; \theta_2)$ 。

通常, 我们需要比这更强的条件:

**条件 4.3 (强可识别性):** 我们假设对于每一个  $\varepsilon > 0$ , 有

$$\inf_{\tilde{\theta}: |\tilde{\theta} - \theta| \geq \varepsilon} KL(f(x; \theta) \| f(x; \tilde{\theta})) > 0.$$

条件 4.3 本质上与条件 4.2 相同, 只是条件 4.3 不允许两个分布之间的差异无限小。如果  $\theta$  被限制在一个紧集中, 那么两个条件是等价的。

**条件 4.4 (一致大数定律 (Uniform Law of Large Numbers)): 假设**

$$\sup_{\tilde{\theta}} |R_T(\tilde{\theta}, \theta) - R(\tilde{\theta}, \theta)| \xrightarrow{p} 0.$$

**定理 4.1:** 在条件 4.3 和 4.4 下, 极大似然估计量具有一致性。

<sup>5</sup>随机变量  $X$  的支撑集 (support) 是所有使其概率密度函数或概率质量函数非零的值的集合。

**证明：**令  $\varepsilon > 0$ , 基于条件 4.3, 若  $|\tilde{\theta} - \theta| \geq \varepsilon$ , 对于每一个  $\varepsilon > 0$ , 则存在  $\eta > 0$  使得

$$\text{KL}(f(x; \theta) \| f(x; \tilde{\theta})) \geq \eta.$$

证明极大似然估计量的一致性, 我们需要证明当  $T \rightarrow \infty$  时,  $\text{KL}(f(x; \theta) \| f(x; \hat{\theta})) \leq \eta$  成立的概率为 1, 由此可得  $|\hat{\theta} - \theta| \leq \varepsilon$ , 即  $\hat{\theta} \xrightarrow{P} \theta$ 。也就是说我们需证明当  $T \rightarrow \infty$  时  $\text{KL}(f(x; \theta) \| f(x; \hat{\theta})) \leq \eta$ 。

注意

$$\text{KL}(f(X; \theta) \| f(X; \hat{\theta})) = R(\hat{\theta}, \theta) = R(\hat{\theta}, \theta) - R_T(\hat{\theta}, \theta) + R_T(\hat{\theta}, \theta) \leq R(\hat{\theta}, \theta) - R_T(\hat{\theta}, \theta) \xrightarrow{P} 0.$$

其中, 条件 4.3 意味着

$$R(\hat{\theta}, \theta) - R_T(\hat{\theta}, \theta) \xrightarrow{P} 0,$$

而

$$R(\hat{\theta}, \theta) - R_T(\hat{\theta}, \theta) + R_T(\hat{\theta}, \theta) \leq R(\hat{\theta}, \theta) - R_T(\hat{\theta}, \theta)$$

成立则是因为

$$R_T(\hat{\theta}, \theta) = \frac{1}{T} \sum_{i=1}^T \log \frac{f(X_i; \theta)}{f(X_i; \hat{\theta})} \leq 0,$$

而上式成立的原因是  $\hat{\theta}$  为极大似然估计量。

注: “当  $T \rightarrow \infty$  时,  $\text{KL}(f(x; \theta) \| f(x; \hat{\theta})) \leq \eta$  成立的概率为 1” 这是一个“几乎必然”或“几乎处处”收敛的概念, 这句话可以用以下公式更简明的表达:

$$\lim_{T \rightarrow \infty} P\left(\text{KL}(f(x; \theta) \| f(x; \hat{\theta})) \leq \eta\right) = 1.$$

其次,  $\hat{\theta}$  为极大似然估计量, 意味着  $\frac{1}{T} \sum_{i=1}^T \log f(X_i; \hat{\theta}) \geq \frac{1}{T} \sum_{i=1}^T \log f(X_i; \theta)$  .

极大似然估计量的有效性以及渐近正态性需要以下充分正则条件 (sufficient regularity condition) :

**条件 4.5:** 参数空间的维度不随  $T$  变化, 即  $\theta \in \mathbb{R}^d$  且  $d$  为固定正整数。

这里  $\mathbb{R}^d$  是  $d$  维实数空间。若  $d$  随  $T$  增加而增长, 极大似然估计甚至可能不是一致的。

**条件 4.6:**  $f(x, \theta)$  是  $\theta$  的光滑函数 (三次可微)。

**条件 4.7:** 我们可以交换关于  $\theta$  的微分和关于  $X$  的积分。这要求  $X$  的取值范围不依赖于  $\theta$ , 以及  $f(x, \theta)$  满足一些可积性条件。

**条件 4.8:** 参数  $\theta$  是可识别的。

**条件 4.9:** 如果参数空间受到限制, 即  $\theta \in \Theta$ , 其中  $\Theta$  为某给定集合, 那么  $\theta$  需要在集合  $\Theta$  的内部 (即不能在其边界上)。

我们将重点关注参数为一维的情况，尽管在一般（固定） $d$  维情况下几乎完全相同。

**定理 4.2：** 在上述规范条件（条件 4.5-4.9）下，

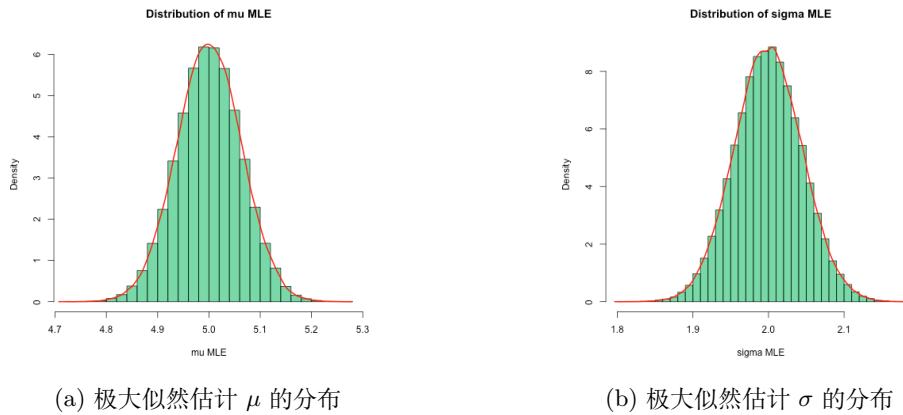
$$\sqrt{T}(\hat{\theta} - \theta) \xrightarrow{d} N(0, 1/\mathcal{I}(\theta)).$$

极大似然估计量的有效性以及渐近正态性的证明相对较为简单，仅需要得分函数、Fisher 信息以及泰勒级数展开，读者可自行完成作为练习。

#### 4.1.13.2 R 中验证极大似然估计的性质

```

1 # 设置种子以确保结果可重复性
2 set.seed (123)
3
4 # 设置参数
5 mu_true <- 5
6 sigma_true <- 2
7 n <- 1000 # 样本大小
8 num_simulations <- 100000 # 模拟次数
9
10 # 初始化存储MLE估计值的向量
11 mu_mle_values <- numeric (num_simulations)
12 sigma_mle_values <- numeric (num_simulations)
13
14 # 进行多次模拟
15 for (i in 1:num_simulations) {
16 sample <- rnorm (n, mu_true, sigma_true)
17 mu_mle_values[i] <- mean (sample)
18 sigma_mle_values[i] <- sqrt(sum((sample - mean(sample))^2) / length(
19 sample))
20 }
21
22 # 验证有效性
23 cat ("Average of mu MLE:", mean (mu_mle_values) , "\n")
24 # 应该接近真实的 mu
25 cat ("Average of sigma^2 MLE:", mean (sigma_mle_values^2) , "\n")
26 # 应该接近真实的 sigma^2
27
28 # 保存mu MLE分布为PNG
29 png ("mu_mle_distribution.png")
30 hist (mu_mle_values, main="Distribution of mu MLE", xlab="mu MLE",
31 breaks=30, probability=TRUE, col=rgb (0.2,0.8,0.5,0.7))
32 lines (density (mu_mle_values) , col="red", lwd=2)
33 dev.off ()
34
35 # 保存sigma MLE分布为PNG
36 png ("sigma_mle_distribution.png")
37 hist (sigma_mle_values, main="Distribution of sigma MLE", xlab="sigma MLE",
38 breaks=30, probability=TRUE, col=rgb (0.2,0.8,0.5,0.7))
39 lines (density (sigma_mle_values) , col="red", lwd=2)
40 dev.off ()
```

图 4.6: 基于极大似然估计得到的  $\hat{\mu}$  和  $\hat{\sigma}$  的分布

#### 4.1.14 单位根

许多宏观经济和金融时间序列显示出长期记忆特性，即历史信息对未来值有持续的影响。例如，历史上的经济冲击可能对未来的产出水平、物价、利率等产生长期影响，这些长期影响反映在单位根的存在上。单位根在经济学和统计学中具有重要的意义，尤其是在时间序列分析领域。单位根过程是一类特殊的随机过程，其特征在于序列中的随机冲击可能会对未来的所有值产生持久影响。单位根过程与宏观经济学和金融市场分析中的许多关键概念如均衡、冲击响应和长期趋势密切相关。以宏观经济领域为例，宏观经济学中经常讨论经济系统达到的长期均衡状态。如果一个经济变量（如 GDP、物价水平）具有单位根，它可能需要更长的时间来响应经济政策或其他外部冲击，从而影响经济系统达到新均衡的速度和路径。在金融市场分析中，理解资产价格或收益率对市场冲击的响应是至关重要的。单位根过程意味着时间序列对冲击的反应可能是永久性的。在分析市场冲击（如金融危机或政策变动）的影响时，单位根的存在需要特别注意。单位根过程的非平稳性对统计推断提出了挑战。标准的统计模型和测试方法可能在这些过程中不适用或产生误导性的结果。因此，对单位根的检测和适当的模型调整（如差分或协整分析）对于进行有效的统计分析和准确的预测至关重要。

弱平稳性要求方差和自协方差是有限的并且与时间无关。考虑如下的自回归过程：

$$y_t = \theta y_{t-1} + \varepsilon_t, \quad (4.20)$$

其中，若  $\theta = 1$ ，则对等式两边取方差得  $\text{Var}(y_t) = \text{Var}(y_{t-1}) + \sigma^2$ ，该式在平稳性条件下无解。除非  $\sigma^2 = 0$ ，此时存在无穷多解。

式 (4.20) 意味着该一阶自回归过程具有单位根，也称为随机游走（random walk，简称 RW）。这种情况下， $y_t$  的无条件方差不存在（无穷大），故该过程是非平稳的。任何  $|\theta| \geq 1$  的情况下，式 (4.20) 描述的都是非平稳过程。总结来说，AR (1) 过程只有当多项式  $1 - \theta L$  可逆时才是平稳的，即特征方程  $1 - \theta z = 0$  在单位圆外。

该结果可以直接推广到任意的 ARMA 模型，以 ARMA (p,q) 模型为例：

$$\theta(L)y_t = \alpha(L)\varepsilon_t$$

此 ARMA (p,q) 模型是平稳的，当且仅当特征方程  $\theta(z) = 0$  的根  $z_1, z_2, \dots, z_p$  绝对值大于 1（在模的意义上），即当 AR 多项式可逆时。

一个重要的情形是当其中一个根恰好等于 1，而其他根的模大于 1 时，我们将  $y_t$  过程表达为：

$$\theta^*(L)(1-L)y_t = \theta^*(L)\Delta y_t = \alpha(L)\varepsilon_t, \quad (4.21)$$

其中  $\theta^*(L)$  是一个关于  $L$  的可逆多项式，阶数为  $p-1$ ， $\Delta = 1-L$  是一阶差分算子。因此，式 (4.21) 表明如果  $y_t$  过程有一个单位根，那么  $\Delta y_t$  是一个平稳的 ARMA 过程。

考虑如下 ARMA(2,1) 过程：

$$y_t = 1.2y_{t-1} - 0.2y_{t-2} + \varepsilon_t - 0.5\varepsilon_{t-1}. \quad (4.22)$$

使用滞后算子  $L$ ，式 (4.22) 可写为

$$(1 - 1.2L + 0.2L^2)y_t = (1 - 0.5L)\varepsilon_t = (1 - 0.2L)(1 - L)y_t = (1 - 0.5L)\varepsilon_t. \quad (4.23)$$

其特征多项式

$$1 - 1.2z + 0.2z^2 = (1 - 0.2z)(1 - z) \quad (4.24)$$

在  $z=1$  处有根，因此 (4.22) 为非平稳过程。

对 (4.22) 做一阶差分（记  $\Delta y_t \equiv (1-L)y_t$ ），可得平稳的 ARMA(1,1)：

$$\Delta y_t = 0.2\Delta y_{t-1} + \varepsilon_t - 0.5\varepsilon_{t-1}. \quad (4.25)$$

经过一阶差分后变为平稳的序列称为“一阶单整”，记作  $I(1)$ 。若  $\Delta y_t$  可由一个平稳的  $ARMA(p-1, q)$  模型刻画，则称  $y_t$  服从自回归-单整-滑动平均 (ARIMA) 模型，其阶数为  $(p, 1, q)$ ，简记为  $ARIMA(p, 1, q)$ 。一阶差分通常可以把非平稳序列转化为平稳序列，尤其在处理经济或金融时间序列（包括其对数）时尤为常见。也有少数序列需要两次差分，即

$$\Delta(\Delta y_t) = \Delta y_t - \Delta y_{t-1},$$

这对应于对数变量“增长率的变化”。若某序列在平稳化前必须进行两次差分，则称为“二阶单整”，记作  $I(2)$ ，其特征多项式具有两个单位根。

$I(0)$  与  $I(1)$  过程的主要差别可概括如下：首先， $I(0)$  过程是平稳的，长期围绕其均值波动，方差有限且不随时间变化；相较之下， $I(1)$  过程的波动范围通常更大。其次， $I(0)$  序列对过去冲击的记忆是有限的，任一随机创新的影响是暂时的；而  $I(1)$  过程具有“长记忆”，一次创新会对路径产生持久影响。最后，对于  $I(0)$  序列，随着滞后阶数的增加，自相关系数会较快衰减；而对于  $I(1)$  过程，自相关衰减至零的速度往往较慢。这些对比反映了两类序列在统计特性上的显著差异，并直接关系到建模与推断策略的选择。

下例模拟一个  $AR(1)$  过程  $y_t = 0.9y_{t-1} + \varepsilon_t$ ，并绘制时间序列图以及对应的自相关函数 (ACF) 与偏自相关函数 (PACF)，输出为 PNG 图。见图 4.7 和图 4.8。

```

1 # 1. 将工作目录设为当前脚本所在目录（在 RStudio 下有效，可按需保留/删除）
2 if (requireNamespace("rstudioapi", quietly = TRUE) &&
3 rstudioapi::isAvailable() &&
4 !is.null(rstudioapi::getActiveDocumentContext()$path)) {
5 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
6 }
7
```

```

9 # 2. 加载必要的库
10 library(ggplot2)
11 library(forecast)
12
13 # 3. 设置种子以保证结果可重现
14 set.seed(123)
15
16 # 4. 模拟 AR(1): $y_t = \alpha * y_{t-1} + \epsilon_t$
17 n_samples <- 500
18 alpha <- 0.9
19 y <- numeric(n_samples)
20 epsilon <- rnorm(n_samples, mean = 0, sd = 1)
21 for (t in 2:n_samples) y[t] <- alpha * y[t - 1] + epsilon[t]
22
23 # 5. 时间序列图 (尺寸与前文统一: 8 x 4.5 英寸)
24 df <- data.frame(t = seq_len(n_samples), y = y)
25
26 ts_plot <- ggplot(df, aes(x = t, y = y)) +
27 geom_line(linewidth = 0.4) +
28 labs(
29 title = expression("Simulated series for " ~ y[t] == 0.9 * y[t-1] +
30 epsilon[t]),
31 x = "t",
32 y = expression(y[t])
33) +
34 theme_minimal(base_size = 12)
35
36 print(ts_plot)
37
38 # 保存为 PNG
39 ggsave("AR1_plot.png", plot = ts_plot, width = 8, height = 4.5, dpi = 300)
40 cat("时间序列图已保存: ", normalizePath("AR1_plot.png"), "\n")
41
42 # 6. ACF 与 PACF 图
43 png("AR1_acf_pacf_plot.png", width = 8, height = 4.5, units = "in", res =
44 300)
45 par(mfrow = c(1, 2), mar = c(4, 4, 3, 1) + 0.1)
46 Acf(
47 y,
48 main = expression("ACF for " ~ y[t] == 0.9 * y[t-1] + epsilon[t]),
49 lag.max = 40
50)
51 Pacf(
52 y,
53 main = expression("PACF for " ~ y[t] == 0.9 * y[t-1] + epsilon[t]),
54 lag.max = 40
55)
56 dev.off()

```

可以将数据生成过程替换为以下代码:

```

1 # 随机游走模型不需要前一个状态的权重, 因此alpha被设为1
2 for (t in 2:n_samples) { y[t] <- y[t-1] + epsilon[t]}

```

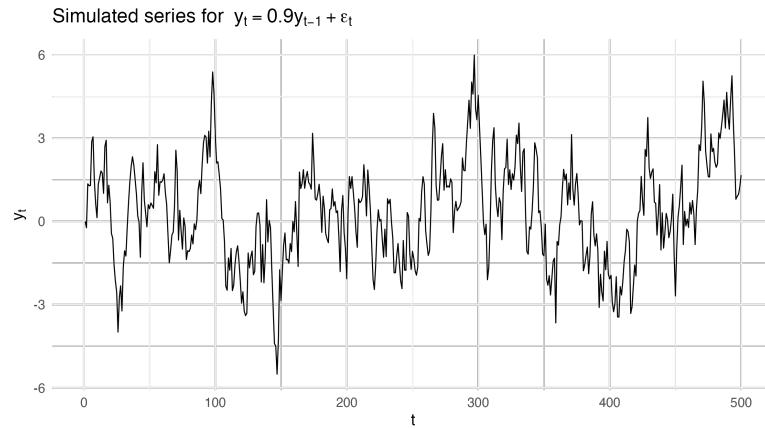


图 4.7: 模拟生成的 AR (1) 序列:  $y_t = 0.9y_{t-1} + \varepsilon_t, \varepsilon_t \sim \text{i.i.d. } N(0, 1)$

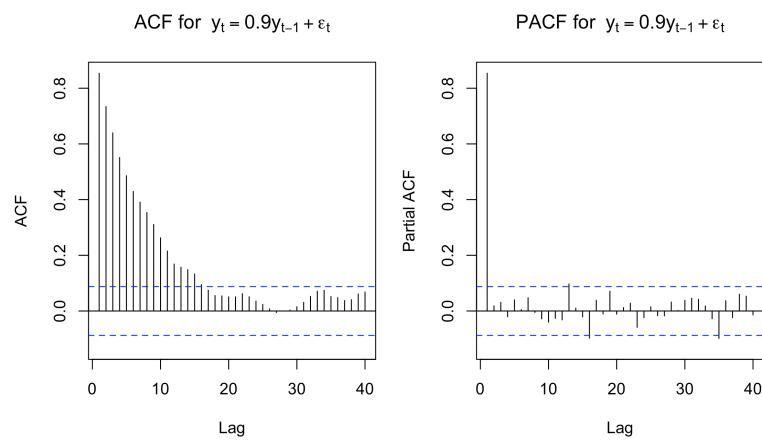


图 4.8: 模拟生成的 AR (1) 序列的自相关函数 (ACF) 和偏自相关函数 (PACF) 图

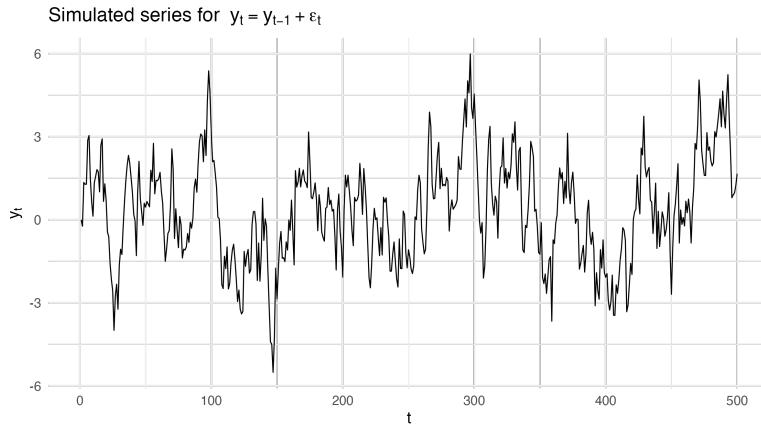


图 4.9: 随机游走过程的时间序列图

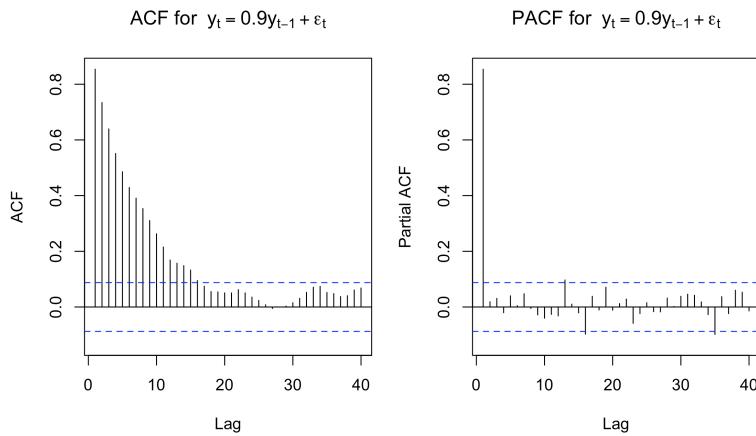


图 4.10: 随机游走过程的自相关函数 (ACF) 与偏自相关函数 (PACF) 图

并得到随机游走过程的时间序列以及自相关函数 (ACF) 和偏自相关函数 (PACF) 图, 见图4.9 以及4.10。<sup>6</sup> 因为  $\alpha = 1$ , 所以时间序列值会显示出无约束的累积效应。在 ACF 图中, 不难看出衰减的速度非常慢, 以至于前几期的 ACF 非常接近 1, 因为随机游走是非平稳的, 过去的值会持续影响未来的值。

在计量经济学中, 理解时间序列对冲击的响应至关重要。由图 4.11 与图 4.12 可见, AR(1) 过程与随机游走 (RW) 过程对同一负向冲击的反应截然不同。

对 AR(1) 过程而言, 若在  $t = 100$  出现负冲击, 序列会瞬间下挫; 由于其具有均值回归特性, 冲击效应随后逐步衰减, 路径重新向长期均值收敛, 可较快回到未受冲击时的轨迹。这反映了 AR(1) 的自我校正机制: 外部扰动虽会造成偏离, 但偏离是暂时的。

相反, 对于随机游走过程, 同样的负冲击会造成水平的永久性下移。冲击发生后, 序列不会回到原有水平, 而是从新的基准继续随机游走, 表现出典型的非平稳与“持久效应”。换言之, RW 的当前值累积了历史冲击的影响, 每一次冲击都会在路径上留下长期“印记”。

这种差异深刻反映了两种过程对于冲击的适应和调整方式的本质区别。在实际应用中, 这意味着服从随机游走的非平稳序列在遭遇诸如市场冲击等突发因素时, 可能需要漫长的时间来消化这些影响; 而平稳序列 (例如 AR(1) 过程) 则能更快地适应并返回到稳定状态。

有趣的是, 这种数据规律的差异与中国古代文化中对自然规律的认知有着奇妙的联系。中国古代的农耕文化依据四季更迭安排农事, 无论每年的气候如何变化, 四季的顺序和大

<sup>6</sup> 由于篇幅原因生成图4.9 以及4.10的代码在附录的软件包中提供。

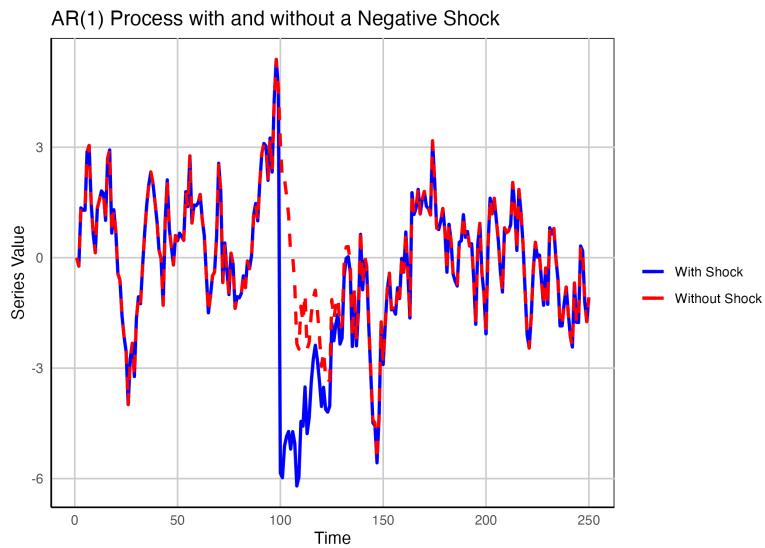


图 4.11: AR (1) 过程中负向冲击前后的对比效果

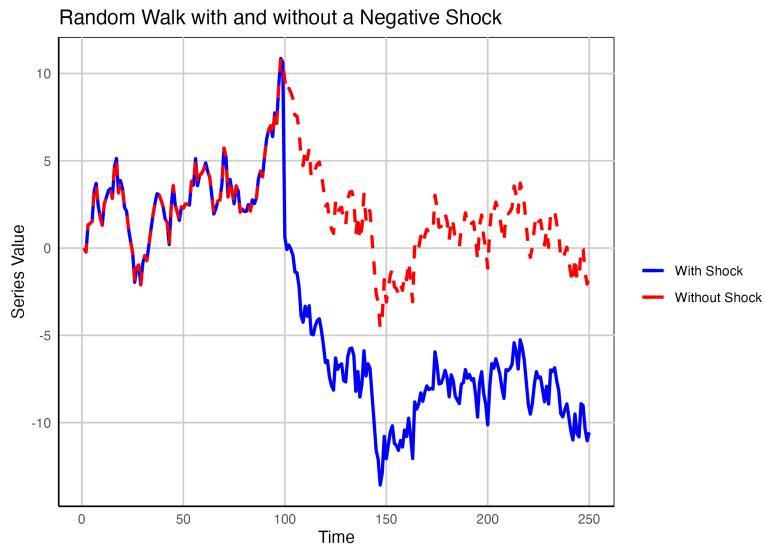


图 4.12: 随机游走过程中负向冲击前后的对比效果

致的时间范围总是稳定的，就像 AR(1) 过程能回归到均值，具有稳定的规律和自我调节能力；春天播种、夏天耕耘、秋天收获、冬天贮藏，年年如此，即便遇到异常气候，来年依旧是四季轮转。天干地支纪年法以 60 年为一个完整周期循环往复，无论朝代更替、世事变迁，这种纪年方式的周期规律始终稳定，如同平稳的时间序列。与之相对，随机游走过程的非平稳特性，就如同古人眼中难以捉摸的天灾，一旦发生，其影响往往难以在短期内消除，可能会打乱原本的生活节奏和秩序，需要漫长的时间去恢复和调整。

这些古老的智慧为我们理解计量经济学中的时间序列数据提供了独特视角。在现代经济研究中，我们不应忽视传统文化的价值，而是要从中汲取灵感，增强对传统文化的认同感和民族自豪感。让古人对自然规律的深刻理解，在现代计量经济学的研究中发挥新的作用。

#### 4.1.14.1 ADF 检验

一个具有单位根的序列被称为具有**随机趋势** (stochastic trend)。这在分析和预测时会引起若干问题。首先，若采用最小二乘法估计  $Y_t = Y_{t-1} + \varepsilon_t$ ，其估计量趋向于零，但真实值

为 1。即当时间序列具有随机趋势时，最小二乘参数估计会得到有偏估计。其次，若回归变量具有随机趋势，则该变量对应的估计量在大样本下也不会收敛于常见的  $t$  分布；若采用  $t$  检验，可能导致错误推断。再次，若两个序列完全独立但均具有随机趋势，它们很可能会误导性地表现出相关性，这一现象被称为**伪回归** (spurious regression)，可能误导政策制定和投资决策。随机趋势还会使传统的预测模型难以捕捉未来的变动趋势，从而增加预测难度。此外，它还可能增强序列对历史信息的依赖，进而影响序列的记忆特性与动态关系建模。在金融市场分析中，随机趋势的存在使价格走势预测更加复杂，并可能影响市场的有效性。因此，识别和处理随机趋势对于时间序列分析的准确性与可靠性至关重要。

## 4.2 结构模型与时间序列模型之间的关系

在第三章中，我们介绍了回归模型（也称结构模型，structural model），此类模型旨在理解和解释经济现象背后的真实机制和关系。这些模型通常基于经济理论或者研究人员对经济金融现象的认知，试图建立经济变量之间的因果关系。而第4.1节主要介绍时间序列模型中的 ARMA 模型，此类模型用于分析和预测随时间变化的经济数据。这些模型假设数据中存在时间依赖性，即未来的观测值与过去的观测值相关。那么，结构模型和时间序列模型之间是否互相矛盾呢？答案是否定的。

为了说明二者之间的关系，假设以下回归模型可以刻画  $Y_t$  和  $X_t$  之间的关系

$$Y_t = \beta_0 + \beta_1 X_t + \varepsilon_t,$$

其中  $\varepsilon_t$  是一个白噪声误差项。若  $X_t$  可以由某个 ARMA 模型描述，则  $Y_t$  为 ARMA 过程与白噪声过程之和，因此也遵循 ARMA 过程。例如，若  $X_t$  服从 MA(1) 过程，即

$$X_t = u_t + \alpha u_{t-1},$$

其中  $u_t$  为与  $\varepsilon_t$  独立的白噪声误差项。由此可得

$$Y_t = \beta_0 + \beta_1(u_t + \alpha u_{t-1}) + \varepsilon_t = \beta_0 + \beta_1 u_t + \alpha \beta_1 u_{t-1} + \varepsilon_t.$$

由此可得：

1.  $E(Y_t) = \beta_0$ ;
2.  $\text{Var}(Y_t) = \beta_1^2(1 + \alpha^2) \sigma_u^2 + \sigma_\varepsilon^2$ , 其中  $\sigma_u^2$  和  $\sigma_\varepsilon^2$  分别为  $u_t$  和  $\varepsilon_t$  的方差;
3.  $\text{Cov}(Y_t, Y_{t-1}) = \beta_1^2 \alpha \sigma_u^2$ ;
4. 对于  $k = 2, 3, \dots$ ,  $\text{Cov}(Y_t, Y_{t-k}) = 0$ 。

因此， $Y_t$  亦服从一个 MA(1) 过程。可见，变量间存在因果关系并不与纯时间序列方法相矛盾。

尽管结构模型和时间序列模型的方法和重点有所不同，它们并不是彼此矛盾的。相反，它们可以互相补充，提供对经济现象更为全面的理解。例如，一个基于结构模型的因果关系可以通过时间序列模型在数据中找到支持，反之亦然。在后续章节中，特别是在介绍向量自回归模型 (VAR, 第4.6节) 和结构向量自回归模型 (SVAR, 第4.7节) 时，我们将更深入地探讨这种联系。VAR 和 SVAR 模型都是时间序列分析中的重要工具，它们可以同时捕捉多个时间序列变量之间的动态关系。VAR 模型是一种多变量时间序列模型，它将每个变量

作为过去值的线性函数来建模，从而揭示变量之间的动态关联。VAR 模型对于预测和政策分析特别有用，因为它能够捕捉变量间的相互影响。SVAR 模型在 VAR 的基础上增加了结构上的假设，它尝试将经济理论引入模型中，以区分变量之间的因果关系。通过对模型结构的假设，SVAR 模型可以帮助我们更好地理解经济变量之间的内在关系，例如识别哪些变量是导致变化的原因，哪些是结果。总之，结构模型和时间序列模型在理解和分析经济数据方面各有其优势。结构模型强调理论和因果关系，而时间序列模型强调数据模式和预测。将两者结合使用，可以确保更全面和深入的分析。

### 4.3 长期方差

长期方差 (long-run variance, 简称 LRV) 的估计对于统计推断至关重要。在金融时间序列中，由于观测值之间存在相关性，准确估计 LRV 具有一定挑战。应用最广的方法之一是异方差和自相关一致 (Heteroskedasticity and Autocorrelation Consistent, 简称 HAC) 估计方法。HAC 推断在计量经济学中尤为重要：当同方差性或独立性等常见假设被违反时，它能够提供更为可靠的标准误估计，从而保证假设检验与置信区间的有效性；同时，HAC 可避免因低估或高估估计量方差而导致的误导性结论，进而提升实证研究的稳健性与可信度。

我们考虑时间序列的加总。假设  $y_t$  满足  $E(y_t) = 0$  且  $E(y_t^2) \leq C < \infty$ 。对  $S_T = \sum_{t=1}^T y_t$  有  $E(S_T) = 0$  且

$$E(S_T^2) = \sum_{t=1}^T E(y_t^2) + \sum_{t=1}^T \sum_{\substack{s=1 \\ s \neq t}}^T E(y_t y_s).$$

若序列弱平稳，则对任意  $t, s$ ，自协方差仅依赖于间隔  $|t - s|$ ，记  $\gamma(k) = E(y_0 y_k)$ （因  $E(y_t) = 0$ ）。变量代换可得

$$E(S_T^2) = T \gamma(0) + 2 \sum_{k=1}^{T-1} (T - k) \gamma(k) = T \gamma(0) + 2T \sum_{k=1}^{T-1} \left(1 - \frac{k}{T}\right) \gamma(k).$$

若进一步假设弱相关 (绝对可和)：

$$\sum_{k=1}^{\infty} |\gamma(k)| < \infty,$$

则有

$$\frac{1}{T} E(S_T^2) = E \left[ \left( \frac{1}{\sqrt{T}} \sum_{t=1}^T y_t \right)^2 \right] \rightarrow \gamma(0) + 2 \sum_{k=1}^{\infty} \gamma(k).$$

根据大数定律 (LLN)，可得  $S_T/T \xrightarrow{P} 0$ 。如果序列  $y_t$  的均值不为零，我们只需减去均值得到

$$\text{Var} \left( \frac{1}{\sqrt{T}} S_T \right) \rightarrow \text{Var}(y_0) + 2 \sum_{k=1}^{\infty} \text{Cov}(y_0, y_k) = \text{lrvvar},$$

上式为长期方差 (与短期方差或普通方差相对)。

在频谱分析中，这是频率零处的谱密度的  $2\pi$  倍。[Parzen \(1957, 式 6.5\)](#) 在 [Bartlett \(1950\)](#)

的基础上，提出了该估计量

$$\widehat{\text{lrvar}} = \sum_{|k| \leq M_T} \left(1 - \frac{k}{M_T}\right) \widehat{\gamma}(k),$$

其中  $\widehat{\gamma}(j)$  是样本自协方差， $M_T$  式满足  $M_T \rightarrow \infty$  以及  $M_T/T \rightarrow 0$  的一个序列。上述方法由 Newey & West (1987, 1994) 提出，因此也称 Newey-West 估计量，该方法允许  $y_t$  是回归模型的残差，并在某些类型的非平稳条件下也适用。我们将详细讨论各类核函数以及 LRV 估计。

在简单回归模型中，长期方差的估计尤为重要。考虑以下回归模型：

$$Y_t = X'_t \beta + u_t \quad t = 1, 2, \dots, T, \quad (4.26)$$

其中  $\beta$  和  $X_t$  是  $q \times 1$  向量， $E(\beta) = \beta_0$ ， $u_t$  是外生误差项，满足  $E(u_t | X_t) = 0$ ， $u_t$  可能具有自相关和条件异方差性。

普通最小二乘法 (OLS) 估计量  $\widehat{\beta}$  为

$$\widehat{\beta} = \left( \sum_{t=1}^T X_t X'_t \right)^{-1} \sum_{t=1}^T X_t Y_t. \quad (4.27)$$

且

$$\sqrt{T}(\widehat{\beta} - \beta) = \left( \frac{1}{T} \sum_{t=1}^T X_t X'_t \right)^{-1} \left( \frac{1}{\sqrt{T}} \sum_{t=1}^T X_t u_t \right).$$

假设大数定律和中心极限定理成立，可得

$$\frac{1}{T} \sum_{t=1}^T X_t X'_t \xrightarrow{p} Q \quad \text{and} \quad \frac{1}{\sqrt{T}} \sum_{t=1}^T X_t u_t \xrightarrow{} N(0, \Omega_v),$$

其中  $\Omega_v$  是  $v_t = X_t u_t$  的渐近方差协方差矩阵。

因此， $\sqrt{T}(\widehat{\beta} - \beta)$  的渐近分布为：

$$\sqrt{T}(\widehat{\beta} - \beta) \xrightarrow{} N(0, Q^{-1} \Omega_v Q^{-1})$$

虽然  $Q$  相对容易估计，但针对  $\widehat{\text{lrvar}}$  的估计却相当复杂。首先定义  $v_t = X_t u_t$ ，由于  $E(u_t | X_t) = 0$ ，因此  $E[v_t] = 0$ 。假设  $v_t$  是弱平稳的，则有

$$\widehat{\text{lrvar}} = \Omega_v = \text{Var} \left( \frac{1}{\sqrt{T}} \sum_{t=1}^T v_t \right) = \frac{1}{T} \sum_{t=1}^T \sum_{s=1}^T E(v_t v'_s)$$

令  $\gamma(j)$  为  $j = t - s$  阶自协方差，可得

$$\widehat{\text{lrvar}} = \Omega_v = \frac{1}{T} \sum_{j=-(T-1)}^{T-1} (T - |j|) \gamma(j) \rightarrow \sum_{j=-\infty}^{\infty} \gamma(j).$$

值得注意的是，在计量经济学建模过程中，模型的误差项一般非独立，通常假设误差项为鞅差序列。鞅差序列是一种特殊的随机变量序列，对于序列中的任何时间点  $t$ ，给定过去

所有信息，该时间点的条件期望值为零。具体来说，对于一个鞅差序列  $\{y_t\}$ ，可得：

$$E(y_t | \mathcal{F}_{t-1}) = 0$$

其中  $\mathcal{F}_{t-1}$  表示时间  $t-1$  之前（包括  $t-1$ ）的所有信息的 sigma 代数。模型的误差项为鞅差序列意味着误差项与解释变量不相关，进而保证回归系数的一致性和有效性。针对鞅差序列的相关理论（大数定律和中心极限定理）可以参考 Hall & Heyde (2014)。

Herrndorf (1984, 推论 1) 针对一般的随机变量序列（不一定是平稳的）确立了以下结果：

**定理 4.3:** 假设  $E(y_t) = 0$ ，对于  $t = 1, 2, \dots$  有  $E(|y_t|^{2+\delta}) \leq C < \infty$ ，且存在某  $\sigma^2 = lrvar > 0$

$$\frac{1}{T} E(S_T^2) \rightarrow \sigma^2,$$

且

$$\sum_{k=1}^{\infty} \alpha(k)^{\frac{\delta}{2+\delta}} < \infty,$$

那么

$$\frac{1}{\sigma\sqrt{T}} S_T \Rightarrow N(0, 1).$$

上述中心极限定理体现了矩条件与混合系数衰减速率之间的关系。如果  $\delta = \infty$ ，仅需  $\sum_{k=1}^{\infty} \alpha(k) < \infty$ ，此时可以允许较慢的衰减速率： $\alpha(k) = (k \log k)^{-1}$ 。另一方面，如果  $\delta$  较小，则要求  $\alpha(k) \rightarrow 0$  的速率快于  $k \rightarrow \infty$  的速率。

## 4.4 采用 ARMA 模型进行预测

首先，我们考虑采用一些简单的时间序列模型进行预测，例如：AR(1) 模型

$$y_t = \theta y_{t-1} + \varepsilon_t,$$

其中  $\varepsilon_t$  均值为零，且为独立同分布，方差为  $\sigma^2 < \infty$ 。我们希望基于样本信息  $\{y_1, \dots, y_T\}$  预测  $y_{T+1}, y_{T+2}, \dots, y_{T+r}$ 。首先假设  $\theta$  是已知的，可得  $y_{T+1} = \theta y_T + \varepsilon_{T+1}$ 。

这里，我们采用条件期望来预测  $y_{T+1}$ ：

$$\hat{y}_{T+1|T} = E(y_{T+1} | y_1, \dots, y_T) = \theta y_T.$$

选择条件期望作为预测量的原因在于，它能够最小化基于  $\{y_1, \dots, y_T\}$  的均方预测误差。对应的预测误差为

$$e_{T+1|T} = y_{T+1} - \hat{y}_{T+1|T} = \varepsilon_{T+1},$$

其均值为零，方差为  $\sigma^2$ 。因此，任何其他基于样本信息的预测方法都将具有不小于此的均方预测误差。

如何针对  $r$  期之后进行预测呢？不难看出

$$y_{T+r} = \theta^r y_T + \theta^{r-1} \varepsilon_{T+1} + \dots + \theta^0 \varepsilon_{T+r}.$$

因此，我们可以设定预测值为

$$\hat{y}_{T+r|T} = \theta^r y_T.$$

其预测误差为

$$y_{T+r} - \hat{y}_{T+r|T} = \theta^{r-1} \varepsilon_{T+1} + \cdots + \varepsilon_{T+r},$$

其均值为零，方差为

$$\sigma^2(1 + \theta^2 + \cdots + \theta^{2(r-1)}).$$

在实际应用中，我们必须使用  $\theta$  的估计量，因此计算方式为

$$\hat{y}_{T+r|T} = \hat{\theta}^r y_T,$$

其中  $\hat{\theta}$  基于样本数据估计得到。预测误差可分解为

$$\begin{aligned} y_{T+r} - \hat{y}_{T+r|T}(\hat{\theta}) &= [y_{T+r} - \hat{y}_{T+r|T}(\theta)] + [\hat{y}_{T+r|T}(\theta) - \hat{y}_{T+r|T}(\hat{\theta})] \\ &= \theta^{r-1} \varepsilon_{T+1} + \cdots + \varepsilon_{T+r} + (\theta^r - \hat{\theta}^r) y_T. \end{aligned}$$

只要  $\hat{\theta}$  是  $\theta$  的一致估计量，随着样本量的增加，上式中  $(\hat{\theta}^r - \theta^r) \xrightarrow{p} 0$ ，即估计误差相对于固有预测误差可以忽略。因此，

$$\text{Var}(y_{T+r} - \hat{y}_{T+r|T}(\theta)) = \sigma^2(1 + \theta^2 + \cdots + \theta^{2(r-1)}).$$

在特定条件下，可以为预测值构建置信区间

$$\hat{y}_{T+r|T} \pm z_{\alpha/2} \sqrt{\hat{\sigma}^2(1 + \hat{\theta}^2 + \cdots + \hat{\theta}^{2(r-1)})},$$

其中  $\hat{\sigma}^2$  与  $\hat{\theta}$  分别为  $\sigma^2$  与  $\theta$  的样本估计。

在大样本情况下，近似为

$$\hat{y}_{T+r|T} \pm z_{\alpha/2} \sqrt{\sigma^2(1 + \theta^2 + \cdots + \theta^{2(r-1)})}.$$

这里  $\alpha$  为显著性水平， $z_{\alpha/2}$  为标准正态分布  $(1 - \alpha/2)$  分位数（例如  $z_{0.025} \approx 1.96$  对应 95% 置信区间）。

**注意：**该置信区间的长度不会因样本量增大而收缩至零，因为其主要由未来冲击的方差决定。在  $\varepsilon_t \sim N(0, \sigma^2)$  等常规条件下，上述区间具有渐近精确性（覆盖率收敛到  $1 - \alpha$ ）。

现在我们假设数据由一个一般的数据生成过程（无穷阶 AR 表示）：

$$y_t = \mu + \sum_{j=1}^{\infty} \theta_j (y_{t-j} - \mu) + \varepsilon_t,$$

其中  $\theta_j$  为系数。由于 ARMA 过程的可逆性，服从 ARMA( $p, q$ ) 的  $y_t$  可以表示为无穷阶 AR 过程。我们希望通过下式预测  $y_{T+1}$ ：

$$\hat{y}_{T+1|T} = \mu + \sum_{j=1}^{\infty} \theta_j (y_{T+1-j} - \mu) = \mu + \sum_{j=1}^T \theta_j (y_{T+1-j} - \mu) + \sum_{j=T+1}^{\infty} \theta_j (y_{T+1-j} - \mu).$$

在实际操作中，由于无法观测到  $y_0, y_1, \dots$ ，且参数  $\mu, \theta_j$  的真值未知，我们只使用第

一个求和项并用样本估计值替代参数：

$$\hat{y}_{T+1|T} = \hat{\mu} + \sum_{j=1}^T \hat{\theta}_j (y_{T+1-j} - \hat{\mu}).$$

接着，我们探讨如何对预测精准度进行评估。一般来讲，我们将总样本分为估计样本 (estimation sample) 和评估样本 (evaluation sample)。设完整样本为  $\{y_1, \dots, y_n\}$ ，其中  $n = T + K$ ，估计样本为  $\{y_1, \dots, y_T\}$ ，评估样本为  $\{y_{T+1}, \dots, y_{T+K}\}$ 。

首先，我们计算对所有评估样本的预测值  $\hat{y}_{T+1|T}, \dots, \hat{y}_{T+K|T}$ ，并计算一些衡量指标。Campbell & Thompson (2008) 提出了样本外  $R^2$ ，在这种情况下定义为

$$R_{OOS}^2 = 1 - \frac{\sum_{j=1}^K (y_{T+j} - \hat{y}_{T+j|T})^2}{\sum_{j=1}^K (y_{T+j} - \bar{y})^2},$$

其中  $\bar{y}$  是整个样本的均值。

其次，如果我们考虑向前一步预测 (one-step ahead forecast)，那么需要逐步修正估计样本的预测方法。也就是说，对于预测  $T + j$  时，我们使用估计样本  $\{y_1, \dots, y_{T+j-1}\}$  并采用新的估计样本的均值  $\bar{y}_{1:T+j-1}$ ，即：

$$R_{OOS,j}^2 = 1 - \frac{\sum_{j=1}^K (y_{T+j} - \hat{y}_{T+j|T+j-1})^2}{\sum_{j=1}^K (y_{T+j} - \bar{y}_{1:T+j-1})^2}.$$

不难看出，样本外  $R^2$  用于衡量某个预测模型相对于简单的样本均值（即  $\bar{y}$ ）的预测表现。如果我们的目的是比较多个模型在未来样本上的预测精度，尤其是要验证预测的改进程度，这种方法非常有效。如果目标是动态地更新预测模型以反映最新的信息，则需要考虑逐步修正估计样本的预测方法。

此外，我们还可以通过假设检验判断哪类预测方法更有效。假设我们采用两种方法对数据进行预测，令原假设为这两种预测误差的均值相同，备择假设为其中一个预测优于另一个。Diebold & Mariano (1995) 提出了一个等预测准确性的检验。设序列  $y_{T+j}$  的两个预测的误差序列分别为  $\hat{\varepsilon}_j$  和  $\hat{\varepsilon}_j^*$ ,  $j = 1, \dots, K$ 。在原假设下，这两种预测误差的均值相同；但由于在时间序列模型下它们可能都具有高度的自相关性，因此在统计量设计时需要考虑这一点。

令

$$S = \frac{\frac{1}{\sqrt{K}} \sum_{j=1}^K d_j}{\widehat{\text{lrvar}}(d_j)},$$

其中  $d_j = \hat{\varepsilon}_j - \hat{\varepsilon}_j^*$ ,  $\widehat{\text{lrvar}}(d_j)$  是序列  $d_j$  的长期方差  $\text{lrvar}(d_j)$  的估计值。如果评估样本大小  $K$  足够大，并且预测误差是平稳的或近似平稳的，则  $S$  可以近似为标准正态随机变量。针对长期方差的介绍见第4.3节。

进行预测时需要对样本外的预测能力进行检验。你会发现，一个在样本内数据上拟合得很好的模型在样本外的预测表现可能并不理想。极端情况下，对样本数据进行完美拟合（即样本内  $R^2$  等于 1）的模型可能无法准确地预测未来数据，这也就是所谓的“过拟合”问题。过拟合的模型过度依赖历史数据的特征，甚至包括噪声，导致其无法准确做出预测。此外，即便某个预测方法在特定时期的预测效果非常好，也无法保证它在另一个时期仍然表现出色。这是因为经济环境、市场条件、政策变化等因素可能会发生显著变化，进而影响模型的预测效果。因此，在预测中保持对模型的谨慎态度并不断对其进行调整是至关重要的，

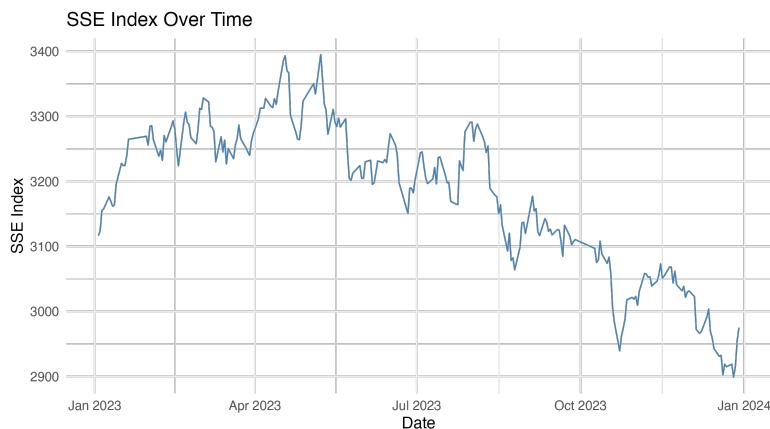


图 4.13: 2023 年 1 月 3 日至 2023 年 12 月 29 日上证综合指数走势

不能仅仅依赖单一模型。

## 4.5 案例：采用 ARMA 模型对上证指数收益率进行建模及预测

上证综合指数，最初名为静安指数，是中国 A 股市场的第一个股票指数。由静安证券业务部于 1989 年初编制并发布，后被上证指数取代。上证指数于 1991 年 7 月 15 日正式发布，其编制方法与静安指数基本相同。它是一种市值加权指数，反映了挂牌股票的整体走势。静安指数虽然发布的时间短暂，但上证指数随着中国资本市场的发展，逐渐成为国际上具有影响力的股票指数之一。<sup>7</sup> 本节采用 ARMA 模型对 2023 年 1 月 3 日至 2023 年 12 月 29 日（2023 年）上证指数收益率进行分析。

我们采用对数收益率。对于价格（或者指数）在两个时刻之间的变化，对数收益率的计算公式为：

$$r_t = \log\left(\frac{p_t}{p_{t-1}}\right).$$

其中  $r_t$  是  $t$  时刻的对数收益率， $p_t$  是  $t$  时刻的价格， $\log$  表示自然对数（以  $e$  为底的对数）。

以下是分析的 R 代码，主要包括数据的导入、预处理、可视化（包括时间序列图和 ACF、PACF 图）、时间序列模型拟合（ARIMA）、使用模型进行预测以及结果的可视化。这是一个典型的金融时间序列分析流程。

```

1 # ---- 上证指数：数据读取、可视化、对数收益率计算与ARIMA 预测（图片尺寸统一
2 # 为 8 x 4.5 英寸，300 dpi） ----
3
4 # 1) 将工作目录设为当前脚本所在目录（在 RStudio 下有效，可按需保留/删除）
5 if (requireNamespace("rstudioapi", quietly = TRUE) &&
6 rstudioapi::isAvailable() &&
7 !is.null(rstudioapi::getActiveDocumentContext()$path)) {
8 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
9 }
10 # 2) 加载必要的库（若缺失则安装）

```

<sup>7</sup> 详见：<https://finance.sina.com.cn/roll/20090601/04456285781.shtml>

```
11 if (!requireNamespace("ggplot2", quietly = TRUE)) install.packages("ggplot2")
12 if (!requireNamespace("forecast", quietly = TRUE)) install.packages("forecast")
13 if (!requireNamespace("tseries", quietly = TRUE)) install.packages("tseries")
14 if (!requireNamespace("dplyr", quietly = TRUE)) install.packages("dplyr")
15 library(ggplot2)
16 library(forecast)
17 library(tseries)
18 library(dplyr)
19
20 # 3) 读取数据并预处理
21 sse <- read.csv("sse.csv", stringsAsFactors = FALSE)
22 sse$date <- as.Date(sse$date) # 确保日期类型
23 sse <- sse[order(sse$date),] # 按日期排序，避免乱序
24
25 # 4) 指数随时间的折线图（统一尺寸与风格）
26 sse_plot <- ggplot(sse, aes(x = Date, y = SSE)) +
27 geom_line(color = "steelblue", linewidth = 0.5) +
28 labs(title = "SSE Index Over Time", x = "Date", y = "SSE Index") +
29 theme_minimal(base_size = 12)
30
31 print(sse_plot)
32 ggsave("sse_index_plot.png", plot = sse_plot, width = 8, height = 4.5, dpi =
300)
33 cat("指数折线图已保存：", normalizePath("sse_index_plot.png"), "\n")
34
35 # 5) 计算对数收益率并划分训练/测试（训练 95%，测试 5%）
36 sse$Log_Returns <- c(NA, diff(log(sse$SSE)))
37 sse <- na.omit(sse) # 去除由差分引入的 NA 值
38 split_index <- floor(0.95 * nrow(sse))
39 train_data <- sse$Log_Returns[1:split_index]
40 test_data <- sse$Log_Returns[(split_index + 1):nrow(sse)]
41 train_date <- sse$date[1:split_index]
42 test_date <- sse$date[(split_index + 1):nrow(sse)]
43
44 # 6) ACF / PACF（训练样本）图，统一为 8 x 4.5 英寸，300 dpi
45 png("sse_acf_pacf.png", width = 8, height = 4.5, units = "in", res = 300)
46 par(mfrow = c(1, 2), mar = c(4, 4, 3, 1) + 0.1)
47 Acf(train_data, main = "ACF of SSE Log Returns (Train)", lag.max = 40)
48 Pacf(train_data, main = "PACF of SSE Log Returns (Train)", lag.max = 40)
49 dev.off()
50 cat("ACF/PACF 图已保存：", normalizePath("sse_acf_pacf.png"), "\n")
51
52 # 7) 拟合 ARIMA（训练样本），并预测测试期长度
53 set.seed(123) # 可复现性 (ARIMA 初始值/搜索)
54 model <- auto.arima(train_data)
55 cat("\n===== ARIMA(Train) 摘要 =====\n")
56 print(summary(model))
57
```

```

58 forecasts <- forecast(model, h = length(test_data))
59
60 # 8) 组织 ggplot 绘图数据 (训练黑线; 预测蓝线; 真实红点; 带蓝色区间带)
61 train_df <- data.frame(Date = train_date, Value = as.numeric(train_data))
62 fc_df <- data.frame(
63 Date = test_date,
64 Forecast = as.numeric(forecasts$mean),
65 Lower = as.numeric(forecasts$lower[, "80%"]),
66 Upper = as.numeric(forecasts$upper[, "80%"])
67)
68 actual_df <- data.frame(Date = test_date, Actual = as.numeric(test_data))
69
70 # 9) 训练+预测+真实值合图 (统一为 8 x 4.5 英寸, 300 dpi)
71 final_plot <- ggplot() +
72 geom_line(data = train_df, aes(x = Date, y = Value), color = "black",
73 linewidth = 0.5) +
74 geom_line(data = fc_df, aes(x = Date, y = Forecast), color = "steelblue",
75 linewidth = 0.6) +
76 geom_ribbon(data = fc_df, aes(x = Date, ymin = Lower, ymax = Upper), fill
77 = "steelblue", alpha = 0.2) +
78 geom_point(data = actual_df, aes(x = Date, y = Actual), color = "red",
79 size = 1.2) +
80 scale_x_date(date_breaks = "3 month", date_labels = "%Y-%m") +
81 labs(title = "SSE Log Returns: Train, Forecast and Actual",
82 x = "Date", y = "Log Returns") +
83 theme_minimal(base_size = 12)
84
85 print(final_plot)
86 ggsave("sse_forecast_plot.png", plot = final_plot, width = 8, height = 4.5,
87 dpi = 300)
88 cat("预测对比图已保存: ", normalizePath("sse_forecast_plot.png"), "\n")

```

图4.14 为上证指数收益率的自相关和偏自相关系数图。ACF 图显示，没有滞后期超出蓝色虚线的置信区间，表明自相关性不显著；PACF 图也显示没有滞后期是显著的，表明偏自相关性不显著。<sup>8</sup>

`auto.arima` 函数是 R 语言中 `forecast` 包提供的一个功能强大的工具，用于自动拟合单变量时间序列数据的最佳 ARIMA 模型。`auto.arima` 函数在默认情况下使用极大似然估计，并通过比较不同模型的信息准则值（如 AIC、AICc 或 BIC）来确定最佳模型，且这一切都是自动完成的。

```

1 > print(summary(model))
2 Series: train_data
3 ARIMA(0,0,0) with zero mean
4
5 sigma^2 = 5.308e-05: log likelihood = 798.65
6 AIC=-1595.31 AICc=-1595.29 BIC=-1591.88
7
8 Training set error measures:
9
 ME RMSE MAE MPE MAPE MASE
 ACF1

```

<sup>8</sup>这是资产收益的一个常见特征，资产收益率经常显示出很小的自相关性，尤其是在收益率已调整以消除任何非平稳性之后。

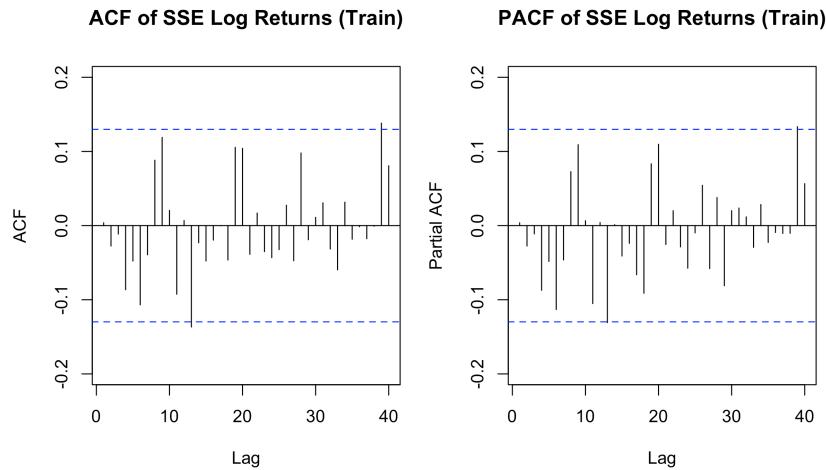


图 4.14: 上证指数对数收益率的自相关与偏自相关 (ACF &amp; PACF)

```
10 Training set -0.0001620855 0.007285922 0.005679844 100 100 0.6898531
 0.003751747
```

其中各误差指标及其英文全称如下：

- ME (Mean Error, 平均误差): 预测误差的算术平均, 反映总体偏差的方向与大小, 易受正负误差抵消影响。
- RMSE (Root Mean Squared Error, 均方根误差): 误差平方的均值再开方, 放大较大误差的影响, 常用于模型整体拟合优度评价。
- MAE (Mean Absolute Error, 平均绝对误差): 误差绝对值的平均, 稳健性较好、解释直观 (与数据量纲一致)。
- MPE (Mean Percentage Error, 平均百分比误差): 相对误差 (带符号) 的平均, 体现系统性高估/低估倾向; 对接近零的实际值较敏感。
- MAPE (Mean Absolute Percentage Error, 平均绝对百分比误差): 百分比误差绝对值的平均, 便于不同量纲/规模间的比较; 同样对零或极小实际值敏感。
- MASE (Mean Absolute Scaled Error, 均值绝对标准化误差): 将模型的 MAE 与“基准模型”(如一步前随机游走或均值模型)的 MAE 之比进行比较,  $MASE < 1$  表示优于基准,  $MASE > 1$  表示劣于基准; 适合跨序列/尺度比较。
- ACF1 (First-order Autocorrelation of Residuals, 一阶残差自相关): 检验残差的一阶相关性, 数值越接近 0 表明相邻残差相关性越弱。

不难看出, 拟合的模型为 ARIMA(0,0,0) 模型, 其中括号内的三个数字分别表示 AR、差分 (I) 和 MA 部分的阶数。在该模型中, AR 和 MA 的阶数均为 0, 意味着不包含自回归与移动平均项, 仅包含常数项 (若采用 zero mean 规格, 则常数项设为 0)。

图4.15 右侧显示了 2023 年 12 月 13 日至 12 月 29 日上证综合指数对数收益率的真实值 (红色)、预测值 (蓝色) 及其预测置信区间; 可以看到, 大多数真实值落在预测置信区间内。

接着我们计算样本外  $R^2$ , 代码如下:

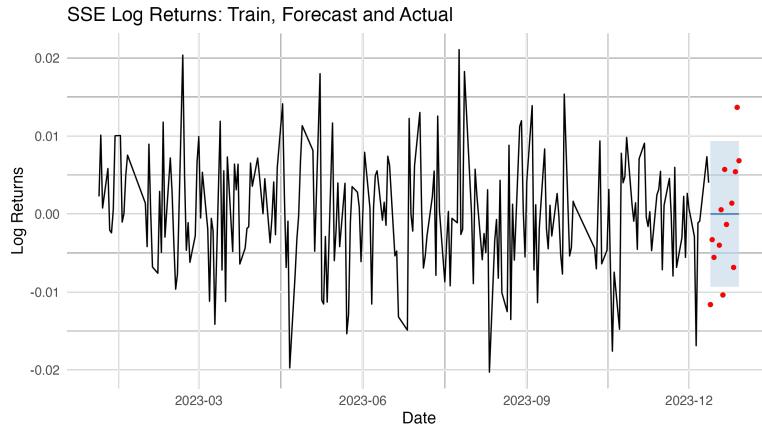


图 4.15: 2023 年上证综合指数对数收益率以及 2023 年 12 月 13 日至 12 月 29 日上证综合指数对数收益率预测以及预测的置信区间

```

1 mean <- mean (sse$Log_Returns)
2 ss_total <- sum ((test_data - mean_train) ^2)
3 ss_residual <- sum ((test_data - forecasts$mean) ^2)
4 R_OOS_2 <- 1 - (ss_residual / ss_total)
5 print (paste ("Out-of-sample R^2:", R_OOS_2))

```

## 4.6 向量自回归模型

向量自回归 (VAR) 模型是一种自回归 (AR) 过程的多维拓展。在一个  $p$  阶的 VAR( $p$ ) 模型中，每个变量不仅是其自身过去值的函数，还受到其他所有变量过去值的影响。这种模型结构允许我们捕捉多个经济或金融变量之间的相互依赖性。

VAR( $p$ ) 模型的表达式为：

$$\mathbf{y}_t = \mathbf{c} + \Phi_1 \mathbf{y}_{t-1} + \Phi_2 \mathbf{y}_{t-2} + \cdots + \Phi_p \mathbf{y}_{t-p} + \boldsymbol{\varepsilon}_t, \quad (4.28)$$

其中  $\mathbf{c}$  表示一个  $n \times 1$  的常数向量， $\Phi_j$  为  $n \times n$  的自回归系数矩阵 ( $j = 1, 2, \dots, p$ )。 $n \times 1$  向量  $\boldsymbol{\varepsilon}_t$  为白噪声，且满足：

$$E(\boldsymbol{\varepsilon}_t) = 0, \quad (4.29)$$

$$E(\boldsymbol{\varepsilon}_t \boldsymbol{\varepsilon}'_{\tau}) = \begin{cases} \Omega & \text{当 } t = \tau \\ 0 & \text{当 } t \neq \tau \end{cases}, \quad (4.30)$$

其中  $\Omega$  是一个  $n \times n$  的对称正定矩阵。

我们可以用滞后算子将上述 VAR( $p$ ) 模型重新表示为：

$$(\mathbf{I}_n - \Phi_1 L - \Phi_2 L^2 - \cdots - \Phi_p L^p) \mathbf{y}_t = \mathbf{c} + \boldsymbol{\varepsilon}_t,$$

或者

$$\Phi(L) \mathbf{y}_t = \mathbf{c} + \boldsymbol{\varepsilon}_t,$$

其中  $\Phi(L)$  是  $n \times n$  的滞后算子  $L$  的矩阵多项式。矩阵  $\Phi(L)$  中的第  $i$  行第  $j$  列元素是一

个标量多项式：

$$\Phi_{ij}(L) = \delta_{ij} - \phi_{ij}^{(1)}L - \phi_{ij}^{(2)}L^2 - \cdots - \phi_{ij}^{(p)}L^p,$$

其中  $\delta_{ij}$  在  $i = j$  时为 1，其他情况下为 0。

#### 4.6.1 VAR 模型的平稳性条件

VAR(p) 模型的平稳性是分析其动态特性的关键前提。如果一个 VAR 模型是协方差平稳的，意味着模型描述的经济系统能够在受到外部冲击后，随时间恢复到某种长期均衡状态。这种平稳性保证了模型的预测结果在长期内是可信的。

(Hamilton 1994, pp. 258–259) 将 VAR(p) 模型转化为 VAR(1) 模型：

$$\boldsymbol{\xi}_t = \mathbf{F} \boldsymbol{\xi}_{t-1} + \mathbf{v}_t, \quad (4.31)$$

其中

$$\begin{aligned} \boldsymbol{\xi}_t &\equiv \begin{bmatrix} \mathbf{y}_t - \boldsymbol{\mu} \\ \mathbf{y}_{t-1} - \boldsymbol{\mu} \\ \vdots \\ \mathbf{y}_{t-p+1} - \boldsymbol{\mu} \end{bmatrix}, \\ \mathbf{F} &\equiv \begin{bmatrix} \Phi_1 & \Phi_2 & \Phi_3 & \cdots & \Phi_{p-1} & \Phi_p \\ \mathbf{I}_n & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I}_n & \mathbf{0} \end{bmatrix}, \end{aligned} \quad (4.32)$$

以及

$$\mathbf{v}_t \equiv \begin{bmatrix} \boldsymbol{\varepsilon}_t \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}.$$

误差项  $\mathbf{v}_t$  满足

$$E(\mathbf{v}_t \mathbf{v}'_\tau) = \begin{cases} \mathbf{Q}, & t = \tau, \\ \mathbf{0}, & t \neq \tau, \end{cases}$$

其中协方差矩阵

$$\mathbf{Q} = \begin{bmatrix} \Omega & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}.$$

式 (4.31) 意味着

$$\boldsymbol{\xi}_{t+s} = \mathbf{v}_{t+s} + \mathbf{F}\mathbf{v}_{t+s-1} + \mathbf{F}^2\mathbf{v}_{t+s-2} + \cdots + \mathbf{F}^{s-1}\mathbf{v}_{t+1} + \mathbf{F}^s\boldsymbol{\xi}_t.$$

换言之，VAR(p) 模型的协方差平稳性要求矩阵  $\mathbf{F}$  的所有特征值位于单位圆内（模长小于 1）。此时，任何由白噪声引起的冲击都会随时间逐渐衰减并最终消失。因此，需检验 VAR

模型的特征值是否位于单位圆内<sup>9</sup>；若不满足，该模型可能产生不切实际的预测，且难以准确刻画系统的真实动态。

以下命题为 (Hamilton 1994, p. 259) 的命题 10.1。

**命题 4.1：** 矩阵  $\Phi$  的特征值满足

$$|\lambda^p \mathbf{I}_n - \Phi_1 \lambda^{p-1} - \Phi_2 \lambda^{p-2} - \cdots - \Phi_p| = 0.$$

即当所有满足上式的  $\lambda$  的模长小于 1 时， $VAR(p)$  模型是协方差平稳的。换言之，

$$|\mathbf{I}_n - \Phi_1 z - \Phi_2 z^2 - \cdots - \Phi_p z^p| = 0$$

的所有根  $z$  均应位于单位圆外。

#### 4.6.2 VAR 模型的极大似然估计

$p$  阶高斯向量自回归模型可以表示为：

$$\mathbf{y}_t = \mathbf{c} + \Phi_1 \mathbf{y}_{t-1} + \Phi_2 \mathbf{y}_{t-2} + \cdots + \Phi_p \mathbf{y}_{t-p} + \varepsilon_t, \quad (4.33)$$

其中  $\varepsilon_t \sim i.i.d. N(\mathbf{0}, \Omega)$ 。

变量数为  $n$ ，样本容量为  $T$ （时间跨度为  $T+p$ ）。将前  $p$  个观测值  $(\mathbf{y}_{-p+1}, \mathbf{y}_{-p+2}, \dots, \mathbf{y}_0)$  作为已知条件，并采用  $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_T$  这  $T$  组观测值用于估计和预测，我们可得如下条件极大似然函数：

$$f_{\mathbf{Y}_T, \mathbf{Y}_{T-1}, \dots, \mathbf{Y}_1 | \mathbf{Y}_0, \mathbf{Y}_{-1}, \dots, \mathbf{Y}_{-p+1}}(\mathbf{y}_T, \mathbf{y}_{T-1}, \dots, \mathbf{y}_1 | \mathbf{y}_0, \mathbf{y}_{-1}, \dots, \mathbf{y}_{-p+1}; \boldsymbol{\theta}). \quad (4.34)$$

极大似然估计量  $\boldsymbol{\theta}$  包含  $\mathbf{c}$ 、 $\Phi_1, \Phi_2, \dots, \Phi_p$  以及  $\Omega$ ，其应确保式 (4.34) 最大化。

参考 Hamilton (1994, p. 292) 的做法，令

$$\mathbf{x}_t \equiv \begin{bmatrix} 1 \\ \mathbf{y}_{t-1} \\ \mathbf{y}_{t-2} \\ \vdots \\ \mathbf{y}_{t-p} \end{bmatrix}, \quad \boldsymbol{\Pi}' \equiv [\mathbf{c} \quad \Phi_1 \quad \Phi_2 \quad \cdots \quad \Phi_p]. \quad (4.35)$$

可得  $\mathbf{y}_t$  的条件分布为：

$$\mathbf{y}_t | \mathbf{y}_{t-1}, \mathbf{y}_{t-2}, \dots, \mathbf{y}_{t-p+1} \sim N(\boldsymbol{\Pi}' \mathbf{x}_t, \Omega). \quad (4.36)$$

进而可得极大似然函数：

$$\begin{aligned} L(\boldsymbol{\theta}) &= \sum_{t=1}^T \log f_{\mathbf{Y}_t | \mathbf{Y}_{t-1}, \mathbf{Y}_{t-2}, \dots, \mathbf{Y}_{t-p}}(\mathbf{y}_t | \mathbf{y}_{t-1}, \mathbf{y}_{t-2}, \dots, \mathbf{y}_{t-p}; \boldsymbol{\theta}) \\ &= -\frac{Tn}{2} \log(2\pi) + \frac{T}{2} \log |\Omega^{-1}| - \frac{1}{2} \sum_{t=1}^T \left[ (\mathbf{y}_t - \boldsymbol{\Pi}' \mathbf{x}_t)' \Omega^{-1} (\mathbf{y}_t - \boldsymbol{\Pi}' \mathbf{x}_t) \right]. \end{aligned}$$

---

<sup>9</sup>单位圆指复平面上半径为 1 的圆。

对之求最大值可得极大似然估计量：

$$\hat{\Pi}'_{n \times (np+1)} = \left[ \sum_{t=1}^T \mathbf{y}_t \mathbf{x}_t' \right] \left[ \sum_{t=1}^T \mathbf{x}_t \mathbf{x}_t' \right]^{-1}, \quad \hat{\Omega} = \frac{1}{T} \sum_{t=1}^T \hat{\varepsilon}_t \hat{\varepsilon}_t'.$$

高斯 VAR(p) 模型的极大似然估计量渐近服从正态分布，详见 Hamilton (1994) 中命题 11.1 和 11.2。

### 4.6.3 VAR 模型的阶数选取

选择 VAR(p) 模型中的滞后阶数  $p$  是一个关键步骤，因为它决定了模型包含多少历史信息。选择过多的滞后阶数可能导致过拟合，即模型过于复杂而无法有效捕捉数据的一般规律；而选择过少的滞后阶数则可能导致模型无法充分捕捉数据中的所有相关信息，从而导致欠拟合。因此，选择合适的  $p$  对于构建有效的 VAR 模型至关重要。

通常，选择  $p$  的方法包括信息准则，例如赤池信息准则 (Akaike information criterion, 简称 AIC)、贝叶斯信息准则 (Bayesian information criterion, 简称 BIC)、汉南-奎因准则 (Hannan–Quinn information criterion, 简称 HQIC) 和经验性检验 (如序列相关检验和残差分析)。

信息准则通过平衡模型复杂度 (滞后阶数) 和拟合优度来确定最优的  $p$ 。具体来说，这些准则评估不同  $p$  值下模型的表现，选择使准则值最小化的  $p$ 。但是，AIC、BIC 以及 HQIC 可能选取不同的阶数，这主要是由于它们对模型复杂度的惩罚方式不同。AIC 的计算公式为

$$AIC = 2k - 2 \log L,$$

其中  $k$  是模型中参数的数量， $L$  是模型的极大似然值。BIC 的计算公式为

$$BIC = \log(T) k - 2 \log L,$$

其中  $T$  为样本量。

HQIC 在惩罚模型复杂度方面介于 AIC 和 BIC 之间。当数据量较大时，HQIC 相比于 AIC 可能更不容易选择过于复杂的模型，从而降低过拟合的风险。值得注意的是，没有单一的准则在所有情况下都是最优的，模型选择准则的应用应该基于具体的数据特性和分析需求。此外，经验检验 (如观察残差的自相关性) 也是选择  $p$  的一种方法，其目的是确保模型残差之间没有自相关，从而保证模型的有效性。

### 4.6.4 格兰杰因果检验

格兰杰因果检验 (Granger causality test) 在金融计量经济学中占据举足轻重的地位。对于金融市场和宏观经济分析而言，透彻理解不同经济与金融变量之间的动态关系，是制定投资策略、作出政策决策以及实现有效风险管理的关键所在。格兰杰因果检验恰为探索这些变量之间是否存在预测性因果关系提供了有力方法。

其核心思想是：若一个变量 (如变量  $X$ ) 的过去值能够显著提升对另一变量 (如变量  $Y$ ) 未来值的预测精度，便可认定  $X$  是  $Y$  的格兰杰原因 ( $X$  Granger-causes  $Y$ )。需要注意的是，格兰杰因果关系并非传统意义上的因果关系，其本质在于检验一个时间序列对另一个时间序列的预测能力。

在实际应用中，格兰杰因果检验的身影广泛出现在各类场景，如分析利率变化对股市走向的影响、探究政府政策对经济增长的作用，或是挖掘汇率变动与国际贸易之间的内在

联系。通过揭示变量间的相互作用，它帮助经济学家和金融分析师深入洞察市场行为与宏观经济趋势，从而在复杂多变的经济环境中作出更为明智的决策。

值得一提的是，这种对事物因果关联的探索，与中国传统文化中对事物关系的深刻认知有着异曲同工之妙。中国古代哲学强调“相生相克”，认为世间万物相互依存、相互影响，形成复杂而有序的关系网络。这一思想映射到格兰杰因果检验中，就如同在经济金融领域，不同变量相互交织、彼此作用，牵一发而动全身。例如，在分析利率与股市的关系时，就如同思考“相生相克”中两种元素的相互作用：利率的调整如同一股力量，作用于股市这一复杂系统，可能引发股市的涨跌起伏，如同水与火的相互制衡。古人通过长期对自然与社会现象的观察总结，形成了独特的认知体系，其中蕴含的因果思维为现代计量经济学研究提供了深厚的文化根基。在学习和运用格兰杰因果检验等现代金融计量方法时，我们应从传统文化中汲取养分，感受古人智慧跨越时空的魅力；这不仅能加深我们对专业知识的理解，更能激发我们的文化认同感与民族自豪感，让古老智慧在现代金融领域焕发新的光芒，实现传统文化与现代科学的有机融合。

格兰杰因果关系的定义如下：

**定义 4.16：**如果  $y$  不能帮助预测  $x$ ，即  $y$  不是  $x$  的格兰杰原因，则对于所有  $s > 0$ ，基于  $(x_t, x_{t-1}, \dots)$  的  $x_{t+s}$  的均方误差 (mean squared error, 简称 MSE) 与同时使用  $(x_t, x_{t-1}, \dots)$  和  $(y_t, y_{t-1}, \dots)$  的均方误差相同，即

$$\text{MSE}[E(x_{t+s} | x_t, x_{t-1}, \dots)] = \text{MSE}[E(x_{t+s} | x_t, x_{t-1}, \dots, y_t, y_{t-1}, \dots)].$$

反之，如果

$$\text{MSE}[E(x_{t+s} | x_t, x_{t-1}, \dots)] > \text{MSE}[E(x_{t+s} | x_t, x_{t-1}, \dots, y_t, y_{t-1}, \dots)],$$

则可认定  $y$  是  $x$  的格兰杰原因。

换而言之，在包含  $x$  和  $y$  的双变量 VAR (p) 模型中，如果所有的系数矩阵  $\Phi_j$  均为下三角形矩阵，那么  $y$  不是  $x$  的格兰杰因 ( $y$  对  $x$  的预测没有帮助)：

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} + \begin{bmatrix} \phi_{11}^{(1)} & 0 \\ \phi_{21}^{(1)} & \phi_{22}^{(1)} \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} + \begin{bmatrix} \phi_{11}^{(2)} & 0 \\ \phi_{21}^{(2)} & \phi_{22}^{(2)} \end{bmatrix} \begin{bmatrix} x_{t-2} \\ y_{t-2} \end{bmatrix} + \dots \\ + \begin{bmatrix} \phi_{11}^{(p)} & 0 \\ \phi_{21}^{(p)} & \phi_{22}^{(p)} \end{bmatrix} \begin{bmatrix} x_{t-p} \\ y_{t-p} \end{bmatrix} + \begin{bmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{bmatrix}. \quad (4.37)$$

等式(4.37) 是约束模型 (constrained model)，对应的约束条件是

$$\phi_{12}^{(1)} = \phi_{12}^{(2)} = \dots = \phi_{12}^{(p)} = 0. \quad (4.38)$$

针对式(4.37)进行检验时，需要构建无约束模型 (unconstrained model)，即：

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} + \begin{bmatrix} \phi_{11}^{(1)} & \phi_{12}^{(1)} \\ \phi_{21}^{(1)} & \phi_{22}^{(1)} \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} + \begin{bmatrix} \phi_{11}^{(2)} & \phi_{12}^{(2)} \\ \phi_{21}^{(2)} & \phi_{22}^{(2)} \end{bmatrix} \begin{bmatrix} x_{t-2} \\ y_{t-2} \end{bmatrix} + \dots \\ + \begin{bmatrix} \phi_{11}^{(p)} & \phi_{12}^{(p)} \\ \phi_{21}^{(p)} & \phi_{22}^{(p)} \end{bmatrix} \begin{bmatrix} x_{t-p} \\ y_{t-p} \end{bmatrix} + \begin{bmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{bmatrix}. \quad (4.39)$$

格兰杰因果检验一般通过对比这两个模型，即 (4.37) 以及 (4.39)，通常采用 F 统计量

或似然比统计量来检验约束条件 (4.38) 是否显著。若统计检验结果显示约束模型与非约束模型之间存在显著差异，则可以认为  $y$  是  $x$  的格兰杰原因。

值得注意的是，格兰杰因果关系并不总是等同于真正的因果关系。Hamilton (1994, pp. 306–307) 举例说明，在金融市场中，尽管股价是股息的格兰杰原因（即股价的变动能够预测股息的变动），但实际上股息才是股价的真正原因。这一现象体现了市场的前瞻性行为 (forward-looking behavior)：投资者会根据预期的未来股息来评估股票价值，因此股票价格已经包含未来股息的信息。换句话说，即使在统计意义上存在格兰杰因果关系，也不能直接判定真实的因果关系，还需要进一步分析并理解背后的经济逻辑。

此外，Lütkepohl (2005, p. 48) 也提供了一个反例，考虑以下 VAR (1) 模型，

$$\begin{bmatrix} z_t \\ x_t \end{bmatrix} = \begin{bmatrix} \alpha_{11} & 0 \\ \alpha_{21} & \alpha_{22} \end{bmatrix} \begin{bmatrix} z_{t-1} \\ x_{t-1} \end{bmatrix} + \begin{bmatrix} u_{1t} \\ u_{2t} \end{bmatrix}.$$

由于参数矩阵为下三角矩阵，可以看出  $x_t$  并不是  $z_t$  的格兰杰原因。但是，如果我们将上述模型两边同时乘以非奇异矩阵  $B$ ：

$$B = \begin{bmatrix} 1 & \beta \\ 0 & 1 \end{bmatrix}$$

由此可得：

$$\begin{bmatrix} z_t \\ x_t \end{bmatrix} = \begin{bmatrix} 0 & -\beta \\ 0 & 0 \end{bmatrix} \begin{bmatrix} z_t \\ x_t \end{bmatrix} + \begin{bmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{bmatrix} \begin{bmatrix} z_{t-1} \\ x_{t-1} \end{bmatrix} + \begin{bmatrix} v_{1t} \\ v_{2t} \end{bmatrix},$$

其中， $\gamma_{11} := \alpha_{11} + \alpha_{21}\beta$ ,  $\gamma_{12} := \alpha_{22}\beta$ ,  $\gamma_{21} := \alpha_{21}$ ,  $\gamma_{22} := \alpha_{22}$  以及  $(v_{1t}, v_{2t})' := B(u_{1t}, u_{2t})'$ 。不难看出，无论是否在两边乘以  $B$ ，过程  $(z_t, x_t)'$  都是等价的。然而，若乘以  $B$ ，则意味着  $x_t$  的变动可能通过第一个方程中系数  $-\beta$  的项传递至  $z_t$ 。换言之，一个变量组对另一变量组没有格兰杰因果关系，并不等同于两者之间不存在因果联系。即便在统计意义上未发现格兰杰因果关系，也不能据此断言变量间不存在真正的因果关系。

#### 4.6.5 脉冲响应函数以及正交脉冲响应函数

在金融计量经济学中，脉冲响应函数 (impulse response function, 简称 IRF) 与正交脉冲响应函数 (orthogonalized impulse response function, 简称 OIRF) 具有重要作用。脉冲响应函数主要用于刻画一个经济变量对另一变量冲击的动态响应，这对于理解和预测经济金融时间序列的变化规律具有重要意义。借助脉冲响应函数，可以衡量某一经济冲击（如政策调整、市场突发事件）在不同时点对其他变量（如资产收益率）的影响。与此同时，脉冲响应函数及其正交形式能够有效揭示经济与金融数据中潜在而复杂的动态联系，对政策制定与投资决策均具有重要参考价值。

为了更好地理解脉冲响应函数，我们首先将 VAR 模型转换为无穷阶的向量移动平均模型 (vector moving average, 简称 VMA)，即

$$\mathbf{y}_t = \boldsymbol{\mu} + \boldsymbol{\varepsilon}_t + \boldsymbol{\Psi}_1 \boldsymbol{\varepsilon}_{t-1} + \boldsymbol{\Psi}_2 \boldsymbol{\varepsilon}_{t-2} + \dots$$

不难看出，

$$\frac{\partial \mathbf{y}_{t+s}}{\partial \boldsymbol{\varepsilon}'_r} = \boldsymbol{\Psi}_s$$

换而言之，参数矩阵  $\boldsymbol{\Psi}_s$  的第  $i$  行第  $j$  列度量了在其他时刻的随机扰动保持不变的情况下，

$\varepsilon_{jt}$  增加一单位对时刻  $t + s$  的第  $i$  个变量  $y_{i,t+s}$  的影响。

如果  $\varepsilon$  的第一个元素变化  $\delta_1$ , 第二个元素变化  $\delta_2$ , 以此类推, 第  $n$  个元素变化  $\delta_n$ , 那么这些变化对  $y_{i,t+s}$  的影响则会得到如下累积:

$$\Delta \mathbf{y}_{t+s} = \frac{\partial \mathbf{y}_{t+s}}{\partial \varepsilon_{1t}} \delta_1 + \frac{\partial \mathbf{y}_{t+s}}{\partial \varepsilon_{2t}} \delta_2 + \cdots + \frac{\partial \mathbf{y}_{t+s}}{\partial \varepsilon_{nt}} \delta_n = \Psi_s \boldsymbol{\delta},$$

其中  $\boldsymbol{\delta} = (\delta_1, \delta_2, \dots, \delta_n)'$ 。关于  $\Psi_s$  的性质, 详见 Hamilton (1994) 第 10.1 节。提取  $\Psi_s$  的第  $i$  行第  $j$  列元素, 并将其视为  $s$  的函数, 则

$$\frac{\partial y_{i,t+s}}{\partial \varepsilon_{j,t}}$$

即为脉冲响应函数。

但是实际经济分析中, 我们更感兴趣的往往是模型中某个变量变动对其他变量的影响, 即

$$\frac{\partial \widehat{E}(\mathbf{y}_{t+s} | y_{jt}, y_{j-1,t}, \dots, y_{1t}, \mathbf{x}_{t-1})}{\partial y_{jt}}$$

由于  $\varepsilon_{i,t}$  与  $\varepsilon_{j,t}$  之间可能相关, 因此通常来说

$$\frac{\partial \widehat{E}(\mathbf{y}_{t+s} | y_{jt}, y_{j-1,t}, \dots, y_{1t}, \mathbf{x}_{t-1})}{\partial y_{jt}} \neq \frac{\partial \widehat{E}(\mathbf{y}_{t+s} | y_{jt}, y_{j-1,t}, \dots, y_{1t}, \mathbf{x}_{t-1})}{\partial \varepsilon_{jt}}$$

为了得到某个变量变动对其他变量的影响, 我们可以利用  $\varepsilon$  的方差协方差矩阵是对称正定矩阵的性质, 对之进行 LDL 分解, 使得  $\Omega = \mathbf{A}\mathbf{D}\mathbf{A}'$ , 其中  $\mathbf{A}$  为唯一的主对角线上元素均为 1 的下三角矩阵, 以及唯一的正定对角矩阵  $\mathbf{D}$ 。<sup>10</sup> 我们可以通过  $\mathbf{u}_t = \mathbf{A}^{-1}\varepsilon_t$  构造一个  $(n \times 1)$  向量  $\mathbf{u}_t$ 。 $\mathbf{u}_t$  的元素彼此不相关, 即  $E(\mathbf{u}_t \mathbf{u}_t') = \mathbf{D}$ 。由此可得正交脉冲响应函数:

$$\frac{\partial \widehat{E}(\mathbf{y}_{t+s} | y_{jt}, y_{j-1,t}, \dots, y_{1t}, \mathbf{x}_{t-1})}{\partial y_{jt}} = \Psi_s \mathbf{a}_j, \quad (4.40)$$

$\mathbf{a}_j$  是矩阵  $\mathbf{A}$  的第  $j$  列。

当然, 式 (4.40) 是针对总体而言的。在实际操作中, 我们需要利用最小二乘法或极大似然法对参数矩阵  $\Phi_1, \dots, \Phi_p$  进行估计, 从而得到  $\widehat{\Phi}_1, \dots, \widehat{\Phi}_p$ 。此外, 采用

$$\widehat{\Omega} = \frac{1}{T} \sum_{t=1}^T \widehat{\varepsilon}_t \widehat{\varepsilon}_t'$$

来估计扰动项  $\varepsilon_t$  的方差—协方差矩阵; 并通过分解

$$\widehat{\Omega} = \widehat{\mathbf{A}} \widehat{\mathbf{D}} \widehat{\mathbf{A}}'$$

求得  $\widehat{\mathbf{A}}$  与  $\widehat{\mathbf{D}}$ , 进而构造

$$\widehat{\mathbf{u}}_t = \widehat{\mathbf{A}}^{-1} \widehat{\varepsilon}_t.$$

最终可以得到样本正交脉冲响应函数:

$$\widehat{\Psi}_s \widehat{\mathbf{a}}_j,$$

<sup>10</sup> LDL 分解是线性代数中的一个数学方法也被称为“无平方根的 Cholesky 分解”(square root-free Cholesky decomposition), 原因在于这种分解方法在计算过程中不需要计算平方根; 在传统的 Cholesky 分解中, 一个正定对称矩阵  $A$  被分解成一个下三角矩阵  $L$  和它的转置  $L'$  的乘积, 且传统的 Cholesky 分解通常涉及到平方根的计算。

其中,  $\hat{\mathbf{a}}_j$  表示矩阵  $\hat{\mathbf{A}}$  的第  $j$  列。同样地, 可以将  $\hat{\Psi}_s \hat{\mathbf{a}}_j$  视为随时间  $s$  变化的函数, 这便构成了正交脉冲响应函数。正交脉冲响应函数不仅揭示了特定时间点上某一变量冲击对系统的即时影响, 还刻画了这一影响随时间的动态演化过程。

## 4.7 结构向量自回归模型

普通的向量自回归 (VAR) 模型存在一个主要挑战, 即它们无法刻画变量之间的同期 (即时) 互动。Hamilton (1994, p. 324) 在其著作中指出, VAR 模型 “没有利用关于这些变量如何预期相关的先验理论观点, 因此无法用来检验理论或根据经济原则解释数据” (made no use of prior theoretical ideas about how these variables are expected to be related, and therefore cannot be used to test theories or interpret the data in terms of economic principles)。为解决这一问题, 研究者通常通过对误差项协方差矩阵进行下三角 Cholesky 分解来构造正交冲击响应函数。但这种方法要求对冲击的传导机制和模型中变量的排序作出一定假设。此外, 正交冲击响应函数只能刻画冲击从第一期开始至特定时点的动态过程, 因而存在一定局限。

另一种更为灵活的方法是使用结构向量自回归 (SVAR) 模型。SVAR 模型允许在建模过程中明确刻画变量之间的同期关系, 从而更好地识别冲击并研究其传导路径。SVAR 模型已成为计量经济学中广泛应用的工具, 尤其适用于分析突发事件或“冲击”如何在整个经济体系中传导。与标准 VAR 模型相比, SVAR 不仅能够刻画变量间的时序动态, 还能揭示它们在同期内的相互影响, 从而更加完整地反映经济机制的运行过程。

Stock & Watson (2001) 强调, SVAR 的优势在于能够将经济理论纳入实证模型结构中, 使得经济学家不再只是观察变量之间的相关性, 而是可以识别其因果机制。为实现这一目标, SVAR 模型通常需要设定识别约束, 用于明确变量之间在同期内的结构关系。这种识别对于政策分析和宏观经济动态推断具有重要意义。

Kilian (2009) 指出, 在能源经济学中区分不同类型的石油价格冲击至关重要。例如, 供应冲击 (如石油产量的变化)、需求冲击 (由全球经济波动引发) 以及石油特定需求冲击 (如地缘政治风险或预防性库存行为) 都会对市场产生不同的影响。一旦识别出这些冲击, SVAR 模型即可用于估计它们对石油实际价格的时间路径的影响。这一过程有助于政策制定者理解冲击的来源, 并制定相应的应对政策。例如, 因地缘政治紧张导致的供应冲击, 其应对措施将不同于由全球经济扩张推动的需求冲击。

此外, SVAR 模型还广泛应用于分析资产价格之间的动态关系, 例如股市、债市和外汇市场之间的联动性。例如, Kilian & Park (2009) 利用 SVAR 模型研究油价冲击对股市的影响, 系统考察了三类冲击 (供应、需求、石油特定需求) 对股市表现的异质性影响。这些研究充分展示了 SVAR 模型在高维经济系统中揭示结构关系和动态传导机制方面的强大能力。

让我们重新审视一下 VAR 模型:

$$\mathbf{y}_t = \Phi_1 \mathbf{y}_{t-1} + \dots + \Phi_p \mathbf{y}_{t-p} + \boldsymbol{\varepsilon}_t, \text{ 其中 } \boldsymbol{\varepsilon}_t \sim (0, \Omega), \quad (4.41)$$

其中,  $\mathbf{y}_t$  为  $p \times 1$  向量,  $\Phi_1$  为  $p \times p$  维系数矩阵,  $\boldsymbol{\varepsilon}_t$  为  $p \times 1$  误差向量, 服从均值为零、协方差矩阵为  $\Omega$  的多元正态分布。为简化起见, 这里省略了常数项。

然而, 当一个变量对另一个变量存在同期影响时, 实际的数据生成过程可能并非方程 (4.41), 而是:

$$\mathbf{y}_t = A_1^* \mathbf{y}_{t-1} + \dots + A_p^* \mathbf{y}_{t-p} + \mathbf{e}_t,$$

其中  $A_j^* = A\Phi_j$ , 且  $\mathbf{e}_t = A\varepsilon_t \sim (0, \Sigma_e = A\Omega_\varepsilon A')$ 。简化形式的参数（即方程 (4.41) 中的  $(\Phi_1, \dots, \Phi_p, \Omega_\varepsilon)$ ）可以通过对观测数据使用最小二乘法 (OLS) 进行估计。但一个变量对另一个变量的同期影响这一核心问题，则需要对结构参数  $(A_1^*, \dots, A_p^*, \Omega_e)$  进行识别。

显然, SVAR 模型的识别是一项具有挑战性的任务。如果不对系数矩阵施加任何限制, 模型将保持未识别状态。这是因为理论上, SVAR 模型包含  $(p+1)n^2 + \frac{n(n+1)}{2}$  个参数, 而简化形式仅有  $pn^2 + \frac{n(n+1)}{2}$  个参数。不难看出, 如果不设置任何限制条件, SVAR 模型估计将缺少  $n^2$  个参数。因此, 在应用 SVAR 模型时, 需要对协方差矩阵以及其他矩阵施加一定的约束。

结构向量自回归模型 (SVAR) 在许多教科书中均有详细讨论, 感兴趣的读者可以参考 Lütkepohl (2005) 和 Amisano & Giannini (2012)。Lütkepohl (2005) 介绍了四种在 VAR 模型中构建结构关系的方法: A 模型、B 模型、AB 模型, 以及由 Blanchard & Quah (1989) 提出的长期约束方法。关于 SVAR 模型的背景及其识别的相关文献, 读者可以参考 Lütkepohl (2005) 第 358–368 页。本节将简要介绍 A 模型、B 模型、AB 模型以及 Blanchard–Quah 式的长期约束, 重点介绍 A 模型和 B 模型。

A 模型通过将协方差矩阵限制为对角矩阵, 并设定一个额外的矩阵以描绘同期关系, 从而直接刻画观测变量之间的即时互动。B 模型则通过将结构关系直接嵌入误差项中, 规范误差方差, 并提供了一种直接分析冲击传导机制的方法。AB 模型融合了 A 模型和 B 模型的优势, 提供了一个框架来同时考察同期和滞后互动, 但需要引入更多的限制条件以实现模型识别。由 Blanchard & Quah (1989) 提出的长期约束方法, 则避免了对结构矩阵施加限制, 而是通过关注经济冲击的长期效应来揭示其持久影响。总之, 这些模型为经济分析提供了一个系统的工具箱, 每种模型都从不同的角度揭示驱动经济现象的复杂交互机制。

A 模型假设协方差矩阵  $\Omega$  为对角矩阵, 这意味着模型中只包含误差项的方差。观测变量之间的同期关系通过一个附加矩阵  $A$  来刻画:

$$A\mathbf{y}_t = A_1^*\mathbf{y}_{t-1} + \cdots + A_p^*\mathbf{y}_{t-p} + \mathbf{e}_t,$$

其中,  $A_j^* = A\Phi_j$ ,  $\mathbf{e}_t = A\varepsilon_t \sim (0, \Sigma_e = A\Omega_\varepsilon A')$ 。矩阵  $A$  的结构为下三角矩阵, 对角线上元素均为 1。这种结构有助于规范化并施加  $p(p-1)/2$  个约束, 从而实现对结构系数的唯一识别。

矩阵  $A$  的具体形式为:

$$A = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ a_{21} & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ a_{p1} & a_{p2} & \cdots & 1 \end{bmatrix}.$$

B 模型则更直接地处理了误差项之间的结构关系。其方法是在误差项中显式引入矩阵  $B$ , 并将误差方差标准化为 1:

$$\mathbf{y}_t = \Phi_1\mathbf{y}_{t-1} + \cdots + \Phi_p\mathbf{y}_{t-p} + B\mathbf{e}_t,$$

其中,  $\varepsilon_t = B\mathbf{e}_t$ , 且  $\mathbf{e}_t \sim (0, I_p)$ 。

下面的 R 代码演示了一个基于预定系数矩阵和冲击矩阵的三变量时间序列模拟过程。该序列包含变量之间的动态交互与同期关系。随后, 我们从生成的数据中估计简化形式的 VAR 模型, 并进一步开展 SVAR 建模与分析。代码首先通过指定带有下三角约束的矩阵, 采用 A 模型识别变量之间的同期关系; 随后, 再利用 B 模型及其约束条件对模型进行估计。

```
1 # 清空环境并设置随机种子以保证可重复性
2 rm (list = ls ())
3 set.seed (123)
4
5 # 设置时间序列的观测数
6 timeLength <- 500
7
8 # 定义VAR过程的系数矩阵
9 CoeffMatrix <- matrix (c (0.25, 0.05, 0.20,
10 0.10, 0.28, 0.22,
11 0.65, 0.45, 0.28) , 3, byrow = TRUE)
12
13 # 定义冲击矩阵的结构系数
14 ImpactMatrix <- diag (1, 3)
15 ImpactMatrix[lower.tri (ImpactMatrix)] <- c (-0.1, -0.06, 0.25)
16
17 # 初始化并生成时间序列数据
18 timeSeriesData <- matrix (rnorm (3 * (timeLength + 1) , 0, 1) , 3,
19 timeLength + 1)
20 for (i in 2: (timeLength + 1)) {
21 timeSeriesData[, i] <- CoeffMatrix %*% timeSeriesData[, i - 1] +
22 ImpactMatrix %*% rnorm (3, 0, 1)
23 }
24 timeSeriesData <- ts (t (timeSeriesData)) # 转换为时间序列格式
25 dimnames (timeSeriesData) [[2]] <- c ("Series1", "Series2", "Series3")
26
27 # 绘制生成的时间序列
28 plot.ts (timeSeriesData, main = "模拟的时间序列数据")
29
30 # 加载估计VAR模型所需的库
31 library (vars)
32
33 # 估计简化形式的VAR模型
34 varModelEstimate <- VAR (timeSeriesData, p = 1, type = "none")
35
36 # 估计A模型
37 # 定义A矩阵以用于结构估计的约束
38 A_matrix <- diag (1, 3)
39 A_matrix[lower.tri (A_matrix)] <- NA
40
41 # 使用A模型的约束估计结构VAR
42 SVAR_A_Model <- SVAR (varModelEstimate, Amat = A_matrix, max.iter = 1000)
43
44 # 显示估计的A模型
45 SVAR_A_Model
46
47 # 估计A模型中结构系数的标准误差
48 SVAR_A_Model$Ase
49
50 # 对A矩阵求逆以得到A模型中的B矩阵
51 solve (SVAR_A_Model$A)
```

```

51
52 # 估计A模型中结构系数的标准误差
53 SVAR_A_Model$Ase
54
55 # 估计B模型
56 # 定义B矩阵以用于结构估计的约束
57 B_matrix <- diag(1, 3)
58 B_matrix[lower.tri(B_matrix)] <- NA
59
60 # 使用B模型的约束估计结构VAR
61 SVAR_B_Model <- SVAR(varModelEstimate, Bmat = B_matrix)
62
63 # 显示估计的B模型
64 SVAR_B_Model
65
66 # 估计B模型中结构系数的标准误差
67 SVAR_B_Model$Bse

```

在上述代码中, `B_matrix <- diag(1, 3)` 用于创建一个名为 `B_matrix` 的  $3 \times 3$  对角矩阵, 其对角线元素均为 1。这种初始化方式在结构 VAR 模型中十分常见, 通常用于对误差方差进行标准化。

随后, `B_matrix[lower.tri(B_matrix)] <- NA` 将矩阵 `B_matrix` 的下三角元素全部设为 `NA`。这一操作意味着在该特定模型中假定变量之间不存在下三角部分的即时相互作用, 从而对结构估计施加了约束条件。

接着, `SVAR_B_Model <- SVAR(varModelEstimate, Bmat = B_matrix)` 使用预定义的 `B_matrix` 作为约束, 对先前估计的简化形式 VAR 模型 (存储在 `varModelEstimate` 中) 进行结构 VAR 的估计。

运行 `SVAR_B_Model` 后, 将得到 SVAR 模型的估计结果, 其中通常包含结构系数及其统计显著性。最后, 通过 `SVAR_B_Model$Bse` 可提取 `B` 矩阵中结构系数的标准误差, 为评估估计结果的精度提供了重要信息。

模拟得到的时间序列结果如图 4.16 所示。

A 模型的估计结果如下所示。

```

1 > SVAR_A_Model
2 SVAR Estimation Results:
3 =====
4 Estimated A matrix:
5 Series1 Series2 Series3
6 Series1 1.00000 0.00000 0
7 Series2 0.05055 1.00000 0
8 Series3 0.04069 -0.31911 1
9
10
11 > SVAR_A_Model$Ase
12 Series1 Series2 Series3
13 Series1 0.00000000 0.00000000 0
14 Series2 0.04472136 0.00000000 0
15 Series3 0.04477846 0.04472136 0

```

`Estimated A matrix` 给出了模型中各变量之间同期关系的系数。所有对角线元素均固定为 1, 这是多数 SVAR 模型中的常见设定。此种标准化处理意味着每个变量对自身的同

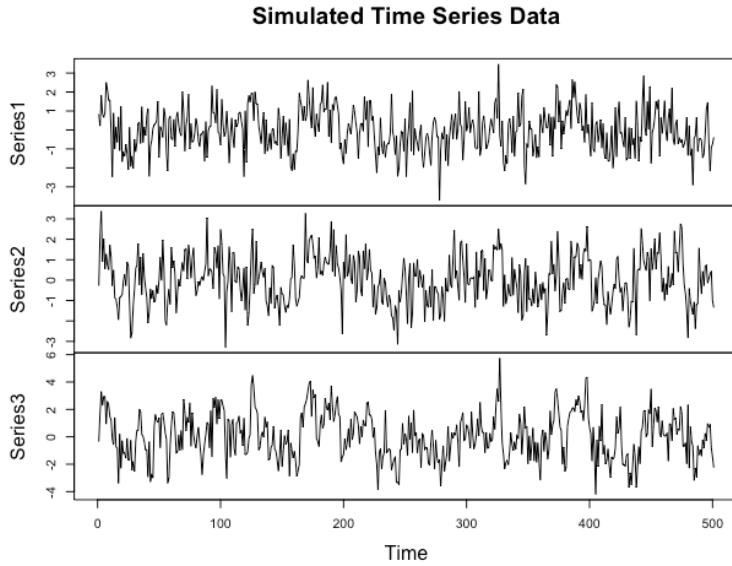


图 4.16: 模拟的时间序列数据

期效应被归一化为 1，从而简化了对非对角线系数的解释。非对角线元素则表示不同变量之间的即时效应。例如，Series2 对 Series1 的系数为 0.05055，表明 Series2 在同期内对 Series1 具有正向影响；而 Series3 对 Series2 的系数为 -0.3191，则说明 Series3 在同期内对 Series2 具有负向影响。

`SVAR_A_Model$A_se` 列示了  $A$  矩阵中各系数的标准误差，这些数值在估计结果下方给出，用以衡量估计系数的统计精度。所有对角线元素的标准误差为零，这是由于模型规范中这些元素被固定（标准化）的典型情形；而非对角线元素的非零标准误差则反映了相应估计量的不确定性。

下面的代码展示了如何利用  $B$  模型对 SVAR 进行估计：

```

1 > solve (SVAR_A_Model$A)
2 Series1 Series2 Series3
3 Series1 1.00000000 0.0000000 0
4 Series2 -0.05054793 1.0000000 0
5 Series3 -0.05681683 0.3190886 1
6
7
8
9 > SVAR_B_Model
10 SVAR Estimation Results:
11 =====
12 Estimated B matrix:
13 Series1 Series2 Series3
14 Series1 1.00000 0.00000 0
15 Series2 -0.05055 1.00000 0
16 Series3 -0.05682 0.31911
17
18 > SVAR_B_Model$Bse
19 Series1 Series2 Series3
20 Series1 0.00000000 0.00000000 0
21 Series2 0.04472136 0.00000000 0

```

```
22 Series3 0.04694289 0.04472136 0
```

`solve(SVAR_A_Model$A)` 用于计算 A 模型中矩阵 A 的逆。在 SVAR 模型的框架下，矩阵 A 通常表示变量之间的同期关系。通过对 A 矩阵求逆，可以得到矩阵 B，后者能够有效刻画结构冲击（区别于观测扰动）对变量的即时影响。可以看到，A 的逆矩阵与在 SVAR B 模型中估计得到的 B 矩阵结果非常接近。

`Estimated B matrix` 显示 B 矩阵的估计结果。所有对角元素均被设定为 1，这种标准化设定在 SVAR 模型中十分常见，意味着每个变量的冲击对自身具有单位效应，且作用是即时的。非对角元素则反映了不同变量之间的即时效应。例如，`Series2` 对 `Series1` 的值为 -0.05055，表明 `Series2` 在同期内对 `Series1` 具有负向影响。

`SVAR_B_Model$B_se` 给出了 B 矩阵各系数的标准误差。对角项的标准误差为零，这是因为这些系数在模型中是预设或标准化的，而非通过估计得到。非对角项的标准误差为非零值，例如 `Series2` 对 `Series1` 的标准误差为 0.04472136，反映了该估计量的精确程度。标准误差越小，说明估计量越精确。

最后，可以使用以下 R 命令生成脉冲响应函数。由于数据为模拟结果，且脉冲响应在统计上显著性极高，因此此处省略具体结果：

```
1 # 计算 A 模型的脉冲响应函数
2 irf_A_model <- irf (SVAR_A_Model, n.ahead = 10, boot = TRUE, ci = 0.95)
3
4 # 绘制 A 模型的脉冲响应
5 plot (irf_A_model)
6
7 # 计算 B 模型的脉冲响应函数
8 irf_B_model <- irf (SVAR_B_Model, n.ahead = 10, boot = TRUE, ci = 0.95)
9
10 # 绘制 B 模型的脉冲响应
11 plot (irf_B_model)
```

结合 A 模型与 B 模型，可以得到 AB 模型，其误差定义为：

$$A\boldsymbol{\varepsilon}_t = B\mathbf{e}_t, \quad \text{其中 } \mathbf{e}_t \sim (0, I_p)$$

与单独的 A 模型或 B 模型相比，AB 模型通常需要施加更多的约束条件。一般而言，可以将其中一个矩阵设为单位矩阵，而对另一个矩阵施加必要的约束，以实现模型的识别。

Blanchard & Quah (1989) 提出的方法并不直接对矩阵 A 或 B 施加约束，而是关注经济冲击的长期效应。具体而言，该方法假定某些冲击可能对变量水平产生永久性影响，而另一些冲击的影响则是暂时的，并会随时间逐步消退。例如，在宏观经济分析中，技术创新可被视为影响 GDP 长期趋势的永久性冲击；而政策调整、自然灾害等通常属于暂时性冲击，其影响会随着时间推移而减弱。通过考察这些冲击的累积效应或长期效应，我们能够识别经济运行的潜在规律以及不同冲击的性质。

## 4.8 章节总结

本章围绕自回归移动平均模型 (ARMA) 及其相关拓展模型展开了系统介绍。首先阐述了 ARMA 模型的基本概念与性质，包括平稳性、可逆性和可识别性条件；随后介绍了自回归 (AR) 与移动平均 (MA) 模型、滞后算子以及自回归移动平均过程，并讲解了自相关系数与偏自相关系数的性质，说明如何利用自相关函数 (ACF) 和偏自相关函数 (PACF)

判断模型阶数。

在估计与推断方面，本章介绍了利用尤尔-沃克方程估计 AR 模型的方法，讨论了极大似然法及其估计量的渐近性质，并在 R 软件中进行了验证。同时，还说明了通过 AIC 与 BIC 准则选择模型阶数、利用 ADF 单位根检验判断平稳性，以及模型长期方差的相关问题。

在预测方面，本章介绍了 ARMA 模型的预测方法，并给出了以上证指数收益率为例的建模与预测案例。随后介绍了向量自回归模型 (VAR)，包括其平稳性条件、极大似然估计、阶数选择、格兰杰因果检验及脉冲响应函数等内容。最后，介绍了结构向量自回归模型 (SVAR)，并讨论了其在识别与应用中的重要作用。

## 4.9 习题

1. 为什么平稳性和遍历性的概念对金融计量经济学至关重要？

2. 假设时间序列  $\{y_t\}$  服从一阶自回归 AR(1) 过程

$$\begin{aligned} y_t &= \theta y_{t-1} + \varepsilon_t, \quad |\theta| < 1, \\ \{\varepsilon_t\} &\sim \text{WN}(0, \sigma^2). \end{aligned}$$

(a) 求  $E(y_t)$ 、 $\text{Var}(y_t)$ 、 $\text{Cov}(y_t, y_{t-j})$  及  $\text{corr}(y_t, y_{t-j})$ ，其中  $j = 0, \pm 1, \dots$ 。

(b)  $\{y_t\}$  是一个弱平稳过程吗？

(c) 若  $|\theta| \geq 1$ ， $\{y_t\}$  是否为弱平稳过程？请给出具体解释。

3. 检查以下过程是否是弱平稳的：

(a)  $y_t = 1.3y_{t-1} - 0.3y_{t-2} + \varepsilon_t$ ,  $\varepsilon_t \sim \text{i.i.d.}(0, \sigma^2)$ ;

(b)  $y_t = 2 + 0.5y_{t-1} - 0.25y_{t-2} + \varepsilon_t$ ,  $\varepsilon_t \sim \text{i.i.d.}(0, \sigma^2)$ ;

(c)  $y_t = 2 + 0.1t + \varepsilon_t + 0.5\varepsilon_{t-1}$ ,  $\varepsilon_t \sim \text{i.i.d.}(0, \sigma^2)$ 。

4. 考虑一个资产的日度对数收益率，记为  $r_t$ ，服从一个 AR(2) 模型：

$$r_t = 0.01 + 0.2 r_{t-2} + a_t,$$

其中  $\{a_t\}$  为高斯白噪声序列，均值为 0，方差为 0.02。

(a) 计算收益率序列  $r_t$  的均值和方差。

(b) 计算  $r_t$  的一阶和二阶自相关系数。

(c) 给定  $r_{100} = -0.01$  和  $r_{99} = 0.02$ ，计算在  $t = 100$  时收益率序列的一步和两步预测。

(d) 计算相应的预测误差。

5. 考虑以下一阶自回归 AR(1) 模型：

$$y_t = \theta y_{t-1} + \varepsilon_t, \tag{4.42}$$

其中  $\varepsilon_t$  为白噪声过程。请回答：

(a) 当  $\theta = 1$  时, 为什么该过程是非平稳的? 解释单位根过程和平稳过程之间的主要区别。

(b) 如何通过单位根检验判断时间序列  $y_t$  是否存在单位根?

(c) 假设  $y_t$  为单位根过程, 如何将其转化为平稳过程?

6. 判断下列说法正确与否, 并给出理由:

(a) 一个零均值 i.i.d. 序列是白噪声 (white noise, 简称 WN) 过程。

(b) 一个零均值 i.i.d. 序列是鞅差分序列 (martingale difference sequence, 简称 MDS)。

(c) MDS 是一个 WN 过程。

(d) WN 过程可能不是 i.i.d. 序列。

(e) WN 过程可能不是 MDS。

7. 进一步拓展第 4.5 节案例的内容:

(a) 采用向前一步预测 (one-step-ahead forecast) 进行预测。

(b) 采用 Diebold & Mariano (1995) 提出的等预测准确性的检验比较两种预测方法 (Diebold–Mariano 检验)。

8. 序列相关与鞅差分序列的关系:

(a) 假设利用某一检验发现  $\{\varepsilon_t\}$  存在序列相关。那么, 是否可以确定  $\{\varepsilon_t\}$  不是一个鞅差分序列? 请给出具体解释。

(b) 假设  $\{\varepsilon_t\}$  不存在序列相关, 是否能确定  $\{\varepsilon_t\}$  是一个鞅差分序列? 请给出具体解释。提示: 考虑时间序列  $\varepsilon_t = y_{t-1}y_{t-2} + \eta_t$ , 其中  $\eta_t \sim \text{i.i.d.}(0, \sigma^2)$ 。

9. 假设我们有一个时间序列  $\{y_t\}$ , 其中  $t = 1, 2, \dots, T$ , 且满足  $E(y_t) = 0$  且  $E(y_t^2) \leq C < \infty$ 。考虑时间序列的部分和  $S_T = \sum_{t=1}^T y_t$ , 回答:

(a) **长期方差的定义**

i. 解释长期方差的概念, 以及为什么在时间序列分析中需要估计长期方差。

ii. 结合弱平稳的定义, 写出  $E(S_T^2)$  的表达式, 并推导长期方差的估计公式。

(b) **HAC 估计方法**

i. 解释 HAC (heteroskedasticity-and-autocorrelation-consistent) 估计量及其在长期方差估计中的作用。

ii. 在下式中识别并解释各项的意义:

$$\widehat{\text{lrvar}} = \sum_{|k| \leq M_T} \left(1 - \frac{|k|}{M_T}\right) \hat{\gamma}(k),$$

其中  $\hat{\gamma}(k)$  为样本自协方差,  $M_T$  满足  $M_T \rightarrow \infty$  且  $M_T/T \rightarrow 0$ 。

(c) **长期方差的估计**

i. 假设  $y_t$  弱相关, 且满足  $\sum_{k=1}^{\infty} |\text{Cov}(y_0, y_k)| < \infty$ , 推导长期方差的渐近表达式。

- ii. 如何使用 Newey-West 方法估计下列回归模型残差的长期方差?

$$Y_t = X'_t \beta + u_t, \quad t = 1, 2, \dots, T,$$

其中  $X_t$  为  $q \times 1$  向量,  $u_t$  为可能存在自相关性和异方差性的残差项。

- iii. 说明如何选择窗口大小  $M_T$  来平衡估计量的偏差与方差。

(d) 中心极限定理与长期方差

- i. 在 Herrndorf (1984) 的结果中, 若  $E(y_t) = 0$ , 且存在某个  $\sigma^2 > 0$  使得

$$\frac{1}{T} E[S_T^2] \rightarrow \sigma^2,$$

推导中心极限定理的渐近分布。

- ii. 若  $y_t$  为鞅差分序列, 解释其满足中心极限定理的原因, 并简要给出鞅差分序列的定义。

(e) 频谱分析中的长期方差

- i. 解释长期方差与零频率处谱密度之间的关系。

- ii. 描述如何使用 Bartlett 核函数估计长期方差。

10. 考虑以下 VAR 模型:

$$\mathbf{y}_t = \boldsymbol{\mu} + \boldsymbol{\varepsilon}_t + \boldsymbol{\Psi}_1 \boldsymbol{\varepsilon}_{t-1} + \boldsymbol{\Psi}_2 \boldsymbol{\varepsilon}_{t-2} + \dots.$$

请回答:

- (a) 写出如何从 VAR 模型得到向量移动平均 (VMA) 模型的表达式。
- (b) 给定脉冲响应函数  $\boldsymbol{\Psi}_s$  (即  $\partial \mathbf{y}_{t+s} / \partial \boldsymbol{\varepsilon}'_t = \boldsymbol{\Psi}_s$ ), 解释其经济含义。
- (c) 在脉冲响应函数分析中, 为何需要进行正文化处理? 请简要说明正文化脉冲响应函数的求解过程。
- (d) 结合经济学案例, 说明脉冲响应函数与正文化脉冲响应函数在政策冲击分析中的应用。

11. 考虑以下双变量 VAR(2) 模型:

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} + \begin{bmatrix} \phi_{11}^{(1)} & \phi_{12}^{(1)} \\ \phi_{21}^{(1)} & \phi_{22}^{(1)} \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} + \begin{bmatrix} \phi_{11}^{(2)} & \phi_{12}^{(2)} \\ \phi_{21}^{(2)} & \phi_{22}^{(2)} \end{bmatrix} \begin{bmatrix} x_{t-2} \\ y_{t-2} \end{bmatrix} + \begin{bmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{bmatrix}.$$

请回答:

- (a) 根据上述模型, 列出  $y_t$  不是  $x_t$  的格兰杰原因的限制条件。
- (b) 针对上述模型, 写出受限模型与非受限模型的表达式。
- (c) 如何使用 F 统计量检验  $y_t$  是否为  $x_t$  的格兰杰原因? 请简要说明。
- (d) 在检验格兰杰因果关系时, 为什么说“格兰杰因果关系并不总是等同于真正的因果关系”? 请结合实际经济学案例说明。

12. 研究包含三个宏观经济变量的经济体：国内生产总值（GDP）、通货膨胀率（ $\pi$ ）和货币供应量（M2）。使用 SVAR 模型，得到估计方程

$$A\mathbf{y}_t = B\mathbf{e}_t, \quad \mathbf{y}_t = \begin{bmatrix} \text{GDP}_t \\ \pi_t \\ \text{M2}_t \end{bmatrix}, \quad \mathbf{e}_t \sim N(\mathbf{0}, \mathbf{I}_3),$$

矩阵  $A$  与  $B$  形式如下：

$$A = \begin{bmatrix} 1 & 0 & 0 \\ a_{21} & 1 & 0 \\ a_{31} & a_{32} & 1 \end{bmatrix}, \quad B = \begin{bmatrix} b_{11} & 0 & 0 \\ b_{21} & b_{22} & 0 \\ b_{31} & b_{32} & b_{33} \end{bmatrix}.$$

- (a) 说明为何需要对矩阵  $A$  与  $B$  施加约束才能识别 SVAR 模型。在当前设定中，对  $A$  与  $B$  施加了哪些约束？这些约束如何帮助识别？
  - (b) 解释矩阵  $A$  中系数  $a_{21}$  与  $a_{31}$  的含义，说明其所代表的经济关系。若  $a_{32} < 0$ ，如何解释通货膨胀对货币供应量的即时影响？
  - (c) 使用矩阵  $B$  进行脉冲响应分析，解释  $b_{11}$  与  $b_{22}$  的含义。若  $b_{11} > 1$ ，意味着什么？请描述货币供应量（M2）对 GDP 冲击的响应路径，该响应是即时的还是存在滞后？
  - (d) 考虑 Blanchard–Quah 方法，若希望区分对 GDP 的永久性冲击与暂时性冲击，说明如何对  $A$  或  $B$  施加长期约束以实现这种区分。在该框架下，哪些变量更可能受到永久性冲击影响，哪些更可能只受暂时性冲击？
  - (e) 编写 R 代码模拟包含上述三个变量的 SVAR 模型，其中设定  $a_{21} = 0.5$ 、 $a_{31} = 0.3$ 、 $b_{11} = 1.0$ 、 $b_{22} = 0.8$ ；使用生成数据先估计简化形式 VAR 模型，再在  $A$  矩阵约束下进行结构估计。
13. 通过 SVAR 模型分析货币政策冲击对中国产出与通货膨胀的影响，使用 2013–2023 年度的季度宏观数据。首先，从中国国家统计局和中国人民银行网站（或 WIND 等数据库）下载实际 GDP、通货膨胀率与短期利率，并进行季节调整、取对数等预处理。
- (a) 构建 VAR 模型，并根据理论与数据特性确定变量顺序和滞后阶数；检验是否存在从短期利率到 GDP 与通货膨胀的 Granger 因果关系。
  - (b) 构建 SVAR 模型，使用带有长期约束的 AB 模型识别需求与供给冲击，分析不同类型冲击对经济变量的动态影响；并分析一次货币政策紧缩冲击（例如短期利率上升）对 GDP 与通货膨胀率的影响。
  - (c) 撰写报告，总结货币政策冲击的分析结果，包括模型诊断、主要发现与政策含义；报告需图表清晰、论述严谨。另请提供完整的 R 脚本文件，包括数据处理、模型估计与结果输出。

## 5 波动率模型

在金融时间序列中，常常可以观察到所谓的波动聚集现象：较大的冲击（如误差或模型的残差）往往伴随着后续的较大冲击（无论方向如何），而较小的冲击也倾向于被较小的冲击所跟随。这种现象在较高频的数据（例如每日或每周的数据）中尤为明显。通过建模波动率，我们可以更好地捕捉和解释市场中这种动态变化。波动率模型，特别是条件异方差模型（如 ARCH、GARCH 模型），允许误差项的方差随时间而变化；这类模型能够灵活刻画金融时间序列中波动率随时间变化的特征，提供比传统模型更强的预测与解释能力。

此外，波动率（尤其是资产回报的条件标准差）是期权定价与风险管理中的关键因素。期权价值高度依赖于基础资产的波动率；波动率越大，期权价格通常可能越高。通过研究波动率模型，交易者与投资者能够更精确地估计与预测市场波动，从而制定更有效的投资策略。金融市场通常会经历波动较大的时期与波动较小的“平稳”时期，例如股票市场常见的持续高波动率阶段与相对平静的低波动率阶段。借助波动率模型，我们能够更好地理解并预测市场在不同阶段的表现，并据此做出相应决策。总之，学习与应用波动率模型对于理解和预测金融市场中的不确定性至关重要，有助于提升金融分析与投资决策的质量。

之前学习的 ARMA 模型属于条件均值模型，用于描述时间序列数据的条件期望值（即均值）的动态变化。它结合了自回归 (AR) 部分和移动平均 (MA) 部分来建模时间序列的依赖结构。本章侧重于条件方差模型，它可以用于描述时间序列数据的条件波动率（即条件方差）的变化。这类模型可用于分析 ARMA 模型或者其他模型的误差项波动率随时间变化的动态特征，适用于金融时间序列中常见的波动聚集现象。

那么，究竟什么是波动聚集现象呢？从图 5.1 不难看出以下几点：某些时间段波动率较高，而在其他时间段则较低，即存在波动聚集现象；同时，波动率随时间连续演变，换言之，剧烈跳变较为少见；波动率不会无限增长，而是在一定范围内变化，从统计角度看，这意味着波动率序列通常是平稳的。最后，虽然图 5.1 所示美元兑人民币汇率的波动聚集现象十分明显，但在股票市场中，波动率对价格的大幅上涨与大幅下跌的反应有所不同，这被称为“杠杆效应”。后续在建模与估计部分我们将系统地加以讨论。

在 ARMA 模型中，我们将随机扰动项（误差项）定义为具有零均值和常数方差的独立同分布随机变量，也就是说

$$\varepsilon_t \sim \text{i.i.d. } (0, \sigma^2).$$

严格来说，上述条件比推导弱平稳性（或协方差平稳性）所需的条件更强。例如，考虑基本的 MA(1) 模型：

$$Y_t = \mu + \varepsilon_t + \alpha \varepsilon_{t-1},$$

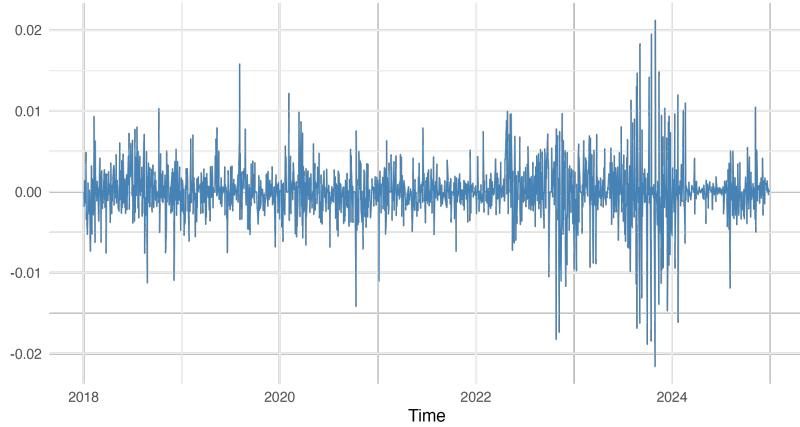


图 5.1: 波动聚集现象示例: 美元兑人民币汇率的对数日收益率 (2018 年 1 月 1 日 - 2024 年 12 月 31 日)

其中我们已经证明

$$\begin{aligned} E(Y_t) &= E(\mu + \varepsilon_t + \alpha\varepsilon_{t-1}) = \mu \\ \text{Var}(Y_t) &= \text{Var}(\varepsilon_t + \alpha\varepsilon_{t-1}) = (1 + \alpha)\sigma^2 \\ \text{Cov}(Y_t, Y_{t-1}) &= \text{Cov}(\varepsilon_t + \alpha\varepsilon_{t-1}, \varepsilon_{t-1} + \alpha\varepsilon_{t-2}) \\ &= \text{Cov}(\varepsilon_t, \varepsilon_{t-1}) + \text{Cov}(\varepsilon_t, \alpha\varepsilon_{t-2}) \\ &\quad + \text{Cov}(\alpha\varepsilon_{t-1}, \varepsilon_{t-1}) + \text{Cov}(\alpha\varepsilon_{t-1}, \alpha\varepsilon_{t-2}) \\ &= \alpha\sigma^2. \end{aligned}$$

值得注意的是, 上述结果成立只需要  $\varepsilon_t$  满足 **无条件均值为零、常数方差以及无序列相关性**, 即

$$E(\varepsilon_t) = 0 \tag{5.1}$$

$$E(\varepsilon_t^2) = \sigma^2 \tag{5.2}$$

$$E(\varepsilon_t \varepsilon_{t-k}) = 0 \text{ for } k \neq 0. \tag{5.3}$$

满足式 (5.1) - (5.3) 的过程称为 **白噪声过程**。需要注意的是, 这并不排除条件方差对过去的依赖, 例如

$$\text{Cov}(\varepsilon_t^2, \varepsilon_{t-k}^2) \neq 0,$$

依然可能成立。

ARMA 模型关注的是时间序列中 **条件期望** (条件均值) 的相依性, 它通过自回归 (AR) 与移动平均 (MA) 两部分来刻画序列与其滞后项之间的线性关系。ARMA 模型通常假设误差项的 **无条件方差为常数** (同方差), 并不对 **条件方差** 进行建模。在金融计量经济学中, 可以用条件期望模型刻画收益, 用条件方差模型刻画风险; 条件期望层面的相依性与条件方差层面的相依性并不矛盾。

图 5.2 诠释了一些常见平稳过程之间的关系。不难看出, 独立同分布 (i.i.d.) 的限制最强, 其次是鞅差分序列 (MDS), 其含义为

$$E(\varepsilon_t | \mathcal{F}_{t-1}) = 0,$$

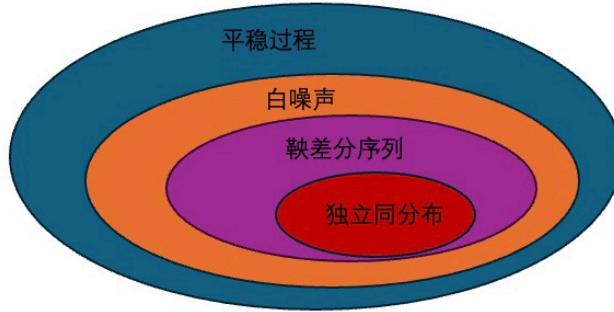


图 5.2: 不同时间序列过程的分类

即在给定过去信息时, 当前的最佳预测为零。相比之下, 白噪声的定义更为宽泛, 仅要求序列之间不存在线性可预测模式。换句话说, 如果相关系数

$$\rho(\varepsilon_t, \varepsilon_s) = 0 \quad \text{对所有 } t \neq s \text{ 均成立,}$$

则可记为  $\varepsilon_t \sim WN(0, \sigma^2)$ 。

## 5.1 ARCH 模型

自回归条件异方差模型 (Autoregressive Conditional Heteroskedasticity, 简称 ARCH) 中, 当前时期的条件方差取决于过去若干期条件均值模型误差项的平方值。也就是说, 方差是随条件而变的: 在某些时段可能较高, 在其他时段可能较低。ARCH 模型用于刻画时间序列数据的条件方差随时间变化的动态特征, 特别适合描述金融时间序列中常见的“波动聚集”现象。该模型通过引入误差项平方的滞后项来建模方差的时间依赖性, 从而在均值方程与方差方程两个层面同时反映数据的统计特性。

罗伯特·F. 恩格尔 (Robert F. Engle III), 美国著名计量经济学家, 以其创立的 ARCH 模型闻名于世, 该模型在经济、金融和统计等多个领域具有广泛应用。

Engle (1982) 提出了 ARCH 模型。2003 年, 因这一开创性贡献, 恩格尔获颁诺贝尔经济学奖。



恩格尔于 1969 年在康奈尔大学获得经济学博士学位, 曾在麻省理工学院 (MIT) 经济学系任教; 1975 年至 2003 年在加州大学圣地亚哥分校 (UCSD) 经济学系任教; 现任纽约大学斯特恩商学院 Michael Armellino 金融服务管理讲座教授。其荣誉包括: 计量经济学会会士、美国艺术与科学院院士、美国统计学会会士、美国金融学会院士及美国国家科学院院士等。此外, 他获多所国际知名高校授予荣誉博士学位, 如法国高等经济商业学院 (HEC Paris)、法国萨伏依大学与瑞士卢加诺大学等。

Engle (1982) 指出, 在  $t$  时的误差项的平方取决于之前时期误差项的平方。ARCH 模型最简单的形式是 ARCH(1) 模型, 即:

$$\varepsilon_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + u_t,$$

其中  $u_t$  是一个新的 **白噪声过程**。不难看出 ARCH(1) 模型是关于  $\varepsilon_t^2$  的 AR(1) 模型。

按定义，时间  $t$  的  $\varepsilon_t$  的条件方差由下式给出：

$$\sigma_t^2 \equiv E(\varepsilon_t^2 | \mathcal{F}_{t-1}) = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2, \quad (5.4)$$

其中  $\mathcal{F}_{t-1}$  表示包含  $\varepsilon_{t-1}$  及其整个历史的信息集。

式 (5.4) 为 **ARCH(1)** 过程。为确保  $\sigma_t^2 \geq 0$ ，无论  $\varepsilon_{t-1}^2$  的值如何，我们需要规定  $\alpha_0 > 0$  且  $\alpha_1 \geq 0$ 。(5.4) 并不意味着  $\varepsilon_t$  的过程是非平稳的，它只是说明  $\varepsilon_{t-1}$  的平方值与  $\sigma_t^2$  相关。

$\varepsilon_t$  的无条件方差由下式给出：

$$\sigma^2 = E(\varepsilon_t^2) = \alpha_0 + \alpha_1 E(\varepsilon_{t-1}^2).$$

在  $0 \leq \alpha_1 < 1$  时，其有一个平稳解：

$$\sigma^2 = \frac{\alpha_0}{1 - \alpha_1}.$$

注：无条件方差不依赖于  $t$ 。

ARCH(1) 模型很容易扩展到 ARCH( $p$ ) 过程：

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \alpha_2 \varepsilon_{t-2}^2 + \cdots + \alpha_p \varepsilon_{t-p}^2,$$

也可以将 ARCH( $p$ ) 过程改写为  $\varepsilon_t^2$  的 AR( $p$ ) 过程：

$$\varepsilon_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \alpha_2 \varepsilon_{t-2}^2 + \cdots + \alpha_p \varepsilon_{t-p}^2 + u_t,$$

其中  $u_t$  是新的白噪声过程。为确保条件方差非负，所有系数  $\alpha_0, \alpha_1, \dots, \alpha_p$  必须为非负；并为确保过程平稳，还需令

$$\alpha_1 + \alpha_2 + \cdots + \alpha_p < 1.$$

由此可得无条件方差：

$$\sigma^2 = \frac{\alpha_0}{1 - (\alpha_1 + \alpha_2 + \cdots + \alpha_p)}.$$

第  $j$  时期前的冲击对当前波动率的影响由系数  $\alpha_j$  决定。在一个 ARCH( $p$ ) 模型中，超过  $p$  时期的旧冲击对当前波动率没有影响。

Ljung–Box 检验用于检测时间序列中的自相关性，所以可以用来检验 ARIMA 模型是否存在自相关；在 ARCH( $p$ ) 模型中也有一定应用。ARCH( $p$ ) 模型

$$\varepsilon_t^2 = \varpi + \sum_{i=1}^p \alpha_i \varepsilon_{t-i}^2 + u_t$$

是  $p$  阶针对  $\varepsilon_t^2$  的 AR 模型，因此可以对  $\varepsilon_t^2$  进行检验：若  $\varepsilon_t^2$  存在序列相关，则意味着  $\varepsilon_t$  存在条件异方差。我们可以对 ARCH( $p$ ) 模型的残差进行自相关检验，以判断残差是否具有随机性。拟合后，可对 ARCH 模型的残差  $\hat{u}_t$  进行自相关性检测。Ljung–Box 检验的统计量  $Q$  如下：

$$Q = T(T+2) \sum_{k=1}^h \frac{\hat{\rho}_k^2}{T-k},$$

其中  $T$  是样本总数,  $h$  为最大滞后阶数,  $\hat{\rho}_k$  为残差在第  $k$  阶的自相关系数。 $Q$  统计量较大表明残差中存在自相关。

以下 R 代码模拟一个 ARCH( $p$ ) 时间序列, 然后使用 `rugarch` 包拟合不同阶的 ARCH 模型, 并基于信息准则 (AIC 和 BIC) 选出表现最佳的模型。最后采用 Ljung–Box 检验对最佳模型的残差进行检验。

```

1 # 加载必要的库
2 library (rugarch)
3 library (ggplot2)
4 library (FinTS)
5
6 # 1. 模拟ARCH (p) 模型
7 set.seed (123)
8 n = 1000
9 omega = 0.1
10 alphas = c (0.5, 0.2, 0.1) # 初始化alpha值, 表示p = 3
11 eps = rnorm (n)
12 sigma2 = rep (0, n)
13 sigma2[1:length (alphas)] = omega / (1 - sum (alphas))
14
15 for (i in (length (alphas) +1) :n) {
16 sigma2[i] = omega + sum (alphas * eps[(i-1) : (i-length (alphas))]^2)
17 eps[i] = rnorm (1, sd = sqrt (sigma2[i]))
18 }
19
20 # 2. 使用ggplot2绘制时间序列
21 df = data.frame (time = 1:n, eps = eps)
22 ggplot (df, aes (x = time, y = eps)) +
23 geom_line () +
24 labs (title = "模拟的ARCH (p) 过程", y = expression (epsilon[t]))
25
26 # 3. 估计不同的ARCH(p)模型并选择最优模型
27 aic_values <- rep (NA, 5)
28 bic_values <- rep (NA, 5)
29 p_max <- 5 # 可以设置一个上限, 如5
30
31 for (p in 1:p_max) {
32 spec = ugarchspec (variance.model = list (model = "sGARCH", garchOrder = c
33 (p, 0)) ,
34 mean.model = list (armaOrder = c (0, 0) , include.mean = FALSE))
35 fit = try (ugarchfit (spec, data = eps) , silent = TRUE)
36
37 aic_values[p] <- infocriteria (fit) [1]
38 bic_values[p] <- infocriteria (fit) [2]
39 }
40
41 # 找出AIC和BIC最小的p值
42 best_p_aic <- which.min (aic_values)
43 best_p_bic <- which.min (bic_values)
44

```

```

45 print (paste ("最佳的 p 值 (AIC) : ", best_p_aic))
46 print (paste ("最佳的 p 值 (BIC) : ", best_p_bic))
47
48 # 使用最优 p 值重新拟合模型，并进行模型检验
49 best_spec = ugarchspec (variance.model = list (model = "sGARCH", garchOrder
50 = c (best_p_aic, 0)) ,
51 mean.model = list (armaOrder = c (0, 0) , include.mean = FALSE))
52 best_fit = ugarchfit (best_spec, data = eps)
53 summary (best_fit)
54
55 # Ljung-Box 检验
56 ljung_box_test = Box.test (best_fit@fit$residuals / best_fit@fit$sigma, lag
57 = 10, type = "Ljung-Box")
58 print (ljung_box_test)

```

上述 R 代码的结果如下：AIC、BIC 准则均选择的最佳阶数为 3，与真实的数据生成过程相一致。

```

1 > print (paste ("最佳的 p 值 (AIC) : ", best_p_aic))
2 [1] "最佳的 p 值 (AIC) : 3"
3 > print (paste ("最佳的 p 值 (BIC) : ", best_p_bic))
4 [1] "最佳的 p 值 (BIC) : 3"
5
6 > print (ljung_box_test)
7
8 Box-Ljung test
9
10 data: best_fit@fit$residuals/best_fit@fit$sigma
11 X-squared = 10.295, df = 10, p-value = 0.415

```

需要指出的是，ARCH 模型针对条件方差建模，刻画序列的波动聚集；而第 4.1 节侧重 ARMA 过程，针对条件期望建模。二者并不矛盾；实务中通常以 ARMA 模型的残差替代  $\varepsilon_t$ 。

### 5.1.1 ARCH 过程的平稳性

首先考虑 ARCH 过程的平稳性，即  $\varepsilon_t$  是否是一个平稳过程；如果是，其平稳方差是多少。假设该过程是弱平稳的，令  $\text{Var}(\varepsilon_t) = \sigma^2 < \infty$ ，且不随时间变化，则必须满足

$$\sigma^2 = E(\sigma_t^2) = \alpha_0 + \alpha_1 E(\varepsilon_{t-1}^2) = \alpha_0 + \alpha_1 \text{Var}(\varepsilon_{t-1}) = \alpha_0 + \alpha_1 \sigma^2.$$

如果  $|\alpha_1| < 1$ ，可得

$$\sigma^2 = \frac{\alpha_0}{1 - \alpha_1}.$$

在上述条件下，该方程具有唯一解。另一方面，如果  $\alpha_1 > 1$ ，则该值为负数。由于  $\sigma^2$  是方差，显然这是不可能的。

进一步地，假设该过程从给定的初始值  $\sigma_1^2$  开始，并探讨远期的波动率是否趋近于平稳

值。若  $|\alpha_1| < 1$ , 则有

$$\begin{aligned} E(\sigma_t^2) &= \alpha_0 + \alpha_1 E(\sigma_{t-1}^2) \\ &= \alpha_0 + \alpha_1 \alpha_0 + \alpha_1^2 E(\sigma_{t-2}^2) \\ &= \alpha_0 \sum_{j=1}^t \alpha_1^{j-1} + \alpha_1^j E(\sigma_1^2) \\ &\rightarrow \frac{\alpha_0}{1 - \alpha_1}. \end{aligned}$$

当  $t \rightarrow \infty$  时, 只要  $E(\sigma_1^2) < \infty$ , 无论  $E(\sigma_1^2)$  取何值,  $E(\sigma_t^2)$  都会趋向于  $\frac{\alpha_0}{1 - \alpha_1}$ 。因此, 过程  $\varepsilon_t$  是渐近弱平稳的 (asymptotically weakly stationary)。如果初始随机变量满足  $E(\sigma_1^2) = \frac{\alpha_0}{1 - \alpha_1}$ , 那么  $E(\sigma_t^2) = E(\sigma_1^2)$  对所有  $t$  均成立。

## 5.2 GARCH 模型

ARCH 模型在描述波动率方面能够取得良好效果, 但在实际建模中往往需要较高的阶数。为解决这一问题, Bollerslev (1986) 提出了 ARCH 模型的一个拓展, 即 generalized ARCH, 简称 GARCH 模型。



蒂姆·博勒斯勒夫 (Tim Bollerslev) 是杜克大学 (Duke University) 经济学系胡安妮塔和克利夫顿·克雷普斯杰出教授 (Juanita and Clifton Kreps Distinguished Professor), 同时担任福库商学院 (Fuqua School of Business) 金融学教授及杜克金融经济中心 (Duke Financial Economics Center, 简称 DFE) 研究主任。他是丹麦皇家科学与文学院及多个国际学术组织的成员, 并长期担任国家经济研究局 (National Bureau of Economic Research, 简称 NBER) 研究员。

博勒斯勒夫博士在其研究领域的顶级学术期刊上发表了大量论文。他是《Journal of Econometrics》中被引用次数第一和第三的论文作者, 并且常被列为全球被引次数最多的经济学家之一。2018 年, 因其在金融与时间序列计量经济学领域的杰出研究, 博勒斯勒夫获得卡尔斯伯格基金会研究奖 (Carlsberg Foundation Research Prize)。尽管 GARCH 模型未直接获得诺贝尔奖, 但在 2003 年诺贝尔经济学奖新闻稿中被提及, 被认为是当时应用最广泛的时变波动率经济时间序列模型之一。

详见 [https://public.econ.duke.edu/~boller/Papers/bollerslev\\_bio.pdf](https://public.econ.duke.edu/~boller/Papers/bollerslev_bio.pdf)。

在其一般形式中, GARCH( $p, q$ ) 模型的表达式为:

$$\sigma_t^2 = \alpha_0 + \sum_{k=1}^p \alpha_k \varepsilon_{t-k}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2,$$

或写成滞后算子形式

$$\sigma_t^2 = \alpha_0 + \alpha(L) \varepsilon_{t-1}^2 + \beta(L) \sigma_{t-1}^2,$$

其中  $\alpha(L) = \sum_{k=1}^p \alpha_k L^k$ 、 $\beta(L) = \sum_{j=1}^q \beta_j L^j$  为滞后算子多项式。

下面以最简单的 GARCH(1,1) 模型为例介绍 GARCH 模型。该模型因其简洁性和有效性而受到广泛青睐。GARCH(1,1) 的表达式为:

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2. \quad (5.5)$$

它只有三个未知参数 ( $\alpha_0$ 、 $\alpha_1$  和  $\beta_1$ ) 需要估计。为确保  $\sigma_t^2 > 0$ , 通常要求  $\alpha_0 > 0$ 、 $\alpha_1 \geq 0$ 、 $\beta_1 \geq 0$ 。在平稳性条件下,  $\varepsilon_t$  的无条件方差可表示为

$$\sigma^2 = \alpha_0 + \alpha_1\sigma^2 + \beta_1\sigma^2,$$

即

$$\sigma^2 = \frac{\alpha_0}{1 - (\alpha_1 + \beta_1)}.$$

显然, 上式仅当  $\alpha_1 + \beta_1 < 1$  时存在。

将式 (5.5) 的滞后项迭代展开, 可得

$$\begin{aligned}\sigma_t^2 &= \alpha_0(1 + \beta_1 + \beta_1^2 + \cdots) + \alpha_1(\varepsilon_{t-1}^2 + \beta_1\varepsilon_{t-2}^2 + \beta_1^2\varepsilon_{t-3}^2 + \cdots) \\ &= \frac{\alpha_0}{1 - \beta_1} + \alpha_1 \sum_{j=1}^{\infty} \beta_1^{j-1} \varepsilon_{t-j}^2.\end{aligned}$$

不难看出, GARCH(1,1) 与无限阶 ARCH 模型等价, 该 ARCH 模型的系数按几何级数方式减弱。换言之, 过去冲击对当前波动率的影响会随时间推移逐步减弱。因此, GARCH 模型较高阶 ARCH 过程更为简洁。依照“简约原则”(principle of parsimony), 在拟合程度相同的情况下, 一般倾向于选择更简洁的 GARCH 模型, 因为高阶 ARCH 模型会损失更多自由度。

以下 R 代码首先模拟一组符合 GARCH(1,1) 模型的数据, 然后根据不同的  $p$  和  $q$  组合定义并拟合 GARCH 模型, 计算每个模型的 AIC 和 BIC 值; 最后输出具有最小 AIC 与 BIC 值的模型阶数。

```

1 # 加载 rugarch 包
2 library (rugarch)

3
4 # 模拟一些数据或使用实际的金融时间序列数据
5 set.seed (123)
6 n = 1000
7 eps = rnorm (n, mean = 0, sd = 1)
8 sigma2 = rep (1, n)
9 y = rep (0, n)
10 alpha0 = 0.01; alpha1 = 0.05; beta1 = 0.9
11 sigma2[1] = alpha0 / (1 - alpha1 - beta1)
12
13 for (i in 2:n) {
14 sigma2[i] = alpha0 + alpha1 * y[i-1]^2 + beta1 * sigma2[i-1]
15 y[i] = rnorm (1, mean = 0, sd = sqrt (sigma2[i]))
16 }
17
18 # 设定一个 (p, q) 的组合范围
19 p_max = 3
20 q_max = 3
21
22 # 存储结果
23 aic_values = matrix (NA, nrow = p_max, ncol = q_max, dimnames = list (p =
24 1:p_max, q = 1:q_max))
25 bic_values = matrix (NA, nrow = p_max, ncol = q_max, dimnames = list (p =
26 1:p_max, q = 1:q_max))
```

```

25
26 # 尝试所有 (p, q) 组合
27 for (p in 1:p_max) {
28 for (q in 1:q_max) {
29 spec = ugarchspec (variance.model = list (model = "sGARCH", garchOrder =
29 c (p, q)) ,
30 mean.model = list (armaOrder = c (0, 0) , include.mean = FALSE))
31 fit = try (ugarchfit (spec = spec, data = y) , silent = TRUE)
32 aic_values[p, q] = infocriteria (fit) [1]
33 bic_values[p, q] = infocriteria (fit) [2]
34 }
35 }
36
37
38 # 找到AIC和BIC最小值的位置
39 best_aic = which (aic_values == min (aic_values, na.rm = TRUE) , arr.ind =
40 TRUE)
40 best_bic = which (bic_values == min (bic_values, na.rm = TRUE) , arr.ind =
41 TRUE)
42
42 # 输出最佳模型阶数
43 cat ("Best (p,q) by AIC: (", best_aic[1, "p"], ",",
43 best_aic[1, "q"], ") \n"
44 ")
44 cat ("Best (p,q) by BIC: (", best_bic[1, "p"], ",",
44 best_bic[1, "q"], ") \n"
45 ")

```

### 5.2.1 GARCH 过程的平稳性

首先，我们讨论 GARCH 过程的弱平稳性。对于所有  $t$ , GARCH( $p, q$ ) 过程定义如下：

$$\sigma_t^2 = \alpha_0 + \sum_{j=1}^p \alpha_j \varepsilon_{t-j}^2 + \sum_{k=1}^q \beta_k \sigma_{t-k}^2.$$

当

$$\sum_{j=1}^p \alpha_j + \sum_{k=1}^q \beta_k < 1$$

时，过程  $\varepsilon_t$  为弱平稳，且无条件方差为

$$\sigma^2 = \text{Var}(\varepsilon_t) = E(\sigma_t^2) = \frac{\alpha_0}{1 - \sum_{j=1}^p \alpha_j - \sum_{k=1}^q \beta_k}.$$

其次，我们探讨 GARCH 过程的强平稳性。考虑如下 GARCH(1,1) 过程：

$$\varepsilon_t = \sigma_t \nu_t, \quad \sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2,$$

其中  $\nu_t$  为独立同分布的非退化随机变量，且  $\beta_1 \geq 0$ 、 $\alpha_0, \alpha_1 > 0$ 。通常假设  $E(\nu_t) = 0$ 、 $\text{Var}(\nu_t) = 1$ ，因此  $\sigma_t^2$  为条件方差。弱平稳的充要条件为  $\beta_1 + \alpha_1 < 1$ 。下面进一步讨论强平稳的条件；这些条件仅要求  $\nu_t$  独立同分布，并不要求其矩存在。

**定理 5.1:** Nelson (1990a) 推导了 GARCH 过程的严平稳 (*strictly stationary*) 性质。

设  $t = 0$  时以随机值  $\sigma_0^2 > 0$  初始化, 且

$$E[\log(\beta_1 + \alpha_1 \nu_t^2)] < 0.$$

则有:

1. 令

$$\sigma_{t,\text{st}}^2 = \alpha_0 \left( 1 + \sum_{j=1}^{\infty} \prod_{i=1}^j (\alpha_1 \nu_{t-i}^2 + \beta_1) \right),$$

则  $\sigma_{t,\text{st}}^2$  为严平稳过程;

$$2. \sigma_{t,\text{st}}^2 \in \left[ \frac{\alpha_0}{1 - \beta_1}, \infty \right);$$

3. 当  $t \rightarrow \infty$  时,  $\sigma_t^2 - \sigma_{t,\text{st}}^2 \rightarrow 0$  的概率为 1;

4. 当  $t \rightarrow \infty$  时,  $\sigma_t^2$  的分布收敛到  $\sigma_{t,\text{st}}^2$  的一个非退化且定义良好的分布 (*non-degenerate and well-defined distribution*)。

当  $E(\nu_t^2) = 1$  时, 根据 Jensen 不等式可得:

$$E[\log(\beta_1 + \alpha_1 \nu_t^2)] < \log(E[\beta_1 + \alpha_1 \nu_t^2]) = \log(\alpha_1 + \beta_1).$$

因此, 即使  $\alpha_1 + \beta_1 \geq 1$ ,  $E[\log(\beta_1 + \alpha_1 \nu_t^2)] < 0$  也是有可能的。换言之, GARCH 过程可以是强平稳的, 同时却不是弱平稳的。在第 4.1.1 节中, 我们给出了服从柯西分布的序列是强平稳而非弱平稳的例子, GARCH 过程在特定情况下也是这样一个强平稳而非弱平稳的例子。

### 5.3 非对称波动率模型

上述 ARCH 和 GARCH 模型设定存在一个重要局限, 即它们具有对称性: 只关注新息绝对值, 而不考虑其正负号。也就是说, 幅度相同的负向冲击对未来波动率的影响与同幅度的正向冲击被视为相同。然而在现实中, 往往是相同幅度的负向冲击对未来波动率的影响更大。这是因为在金融市场中, 投资者通常具有风险厌恶特性: 当出现负面消息时, 投资者更可能恐慌抛售, 导致不确定性上升、波动率剧增; 而正向冲击 (如业绩超预期、宏观数据向好) 虽会引发买入, 但由于担忧利好难以持续, 买入力度相对克制。此外, 许多市场存在卖空限制, 负面冲击时投资者难以通过卖空充分表达悲观预期, 只能离场, 进一步加剧波动。因此, 在同等幅度下, 负面冲击对未来波动率的影响通常大于正向冲击。

为刻画这种不对称性, Nelson (1991) 提出了指数 GARCH (exponential GARCH, 简称 EGARCH) 模型。考虑 EGARCH(1,1):

$$\log \sigma_t^2 = \omega + \beta \log \sigma_{t-1}^2 + \gamma \frac{\varepsilon_{t-1}}{\sigma_{t-1}} + \alpha \frac{|\varepsilon_{t-1}|}{\sigma_{t-1}}.$$

其中,  $\omega, \beta, \gamma, \alpha$  为常数参数。对数变换 ( $\log \sigma_t^2$ ) 保证了条件方差永不为负, 因此这些参数无需非负约束。由于包含  $\frac{\varepsilon_{t-1}}{\sigma_{t-1}}$  与  $\frac{|\varepsilon_{t-1}|}{\sigma_{t-1}}$  两项, 只要  $\gamma \neq 0$ , EGARCH 模型即体现杠杆效应。当  $\gamma < 0$  时, 正向冲击引致的波动率增幅小于负向冲击 (“坏消息”)。EGARCH 可通过增

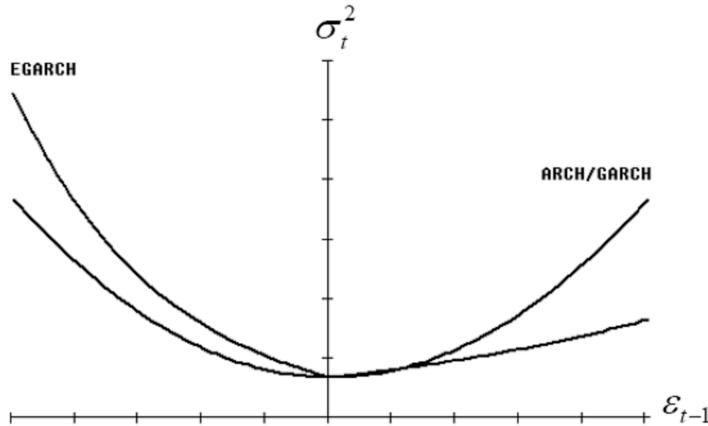


图 5.3: 新息冲击曲线

加额外滞后项进行扩展。

此外, Glosten et al. (1993) 提出了 GJR 模型。GJR-GARCH(1,1) 定义如下: 对所有  $t$ ,

$$\sigma_t^2 = \omega + \beta_1 \sigma_{t-1}^2 + \gamma y_{t-1}^2 + \delta y_{t-1}^2 \mathbf{1}\{y_{t-1} < 0\},$$

其中  $\mathbf{1}\{\cdot\}$  为指示函数。该模型与 GARCH 过程类似, 不同之处在于当  $\delta \neq 0$  时, 允许新息影响曲线 (news impact curve) 呈现不对称性。

**定义 5.1:** 新息冲击曲线 (*news impact curve*) 用于刻画“新息”(如价格变动等冲击) 对未来条件方差(波动率)的影响强度与方向。它展示不同规模与符号的冲击对后续波动率的作用, 从而揭示波动率的不对称(杠杆)效应: 在许多市场中, 同幅度的负向新息对未来波动率的提升通常大于正向新息。示意见图 5.3。

除了 EGARCH 和 GJR 模型外, 常见的非对称波动率模型还包括门限 GARCH(Threshold GARCH) 模型 (Glosten et al. 1993)、门限 ARCH 模型 (Zakoian 1994) 以及马尔可夫区制转换 (Markov Regime-Switching) GARCH 模型 (Hamilton & Susmel 1994)。

门限 GARCH(1,1) 模型定义如下:

$$\begin{cases} \varepsilon_t = \nu_t \sigma_t, \\ \sigma_t^2 = \psi + \beta \sigma_{t-1}^2 + \alpha_1 \varepsilon_{t-1}^2 \mathbf{1}\{\nu_{t-1} < 0\} + \alpha_2 \varepsilon_{t-1}^2 \mathbf{1}\{\nu_{t-1} \geq 0\}, \\ \{\nu_t\} \sim \text{i.i.d. } (0, 1). \end{cases}$$

其中  $\mathbf{1}\{\cdot\}$  为示性(指示)函数。

门限 ARCH 模型定义为:

$$\sigma_t = \psi + \beta \sigma_{t-1} + \alpha_1 |\varepsilon_{t-1}| \mathbf{1}\{\nu_{t-1} > 0\} + \alpha_2 |\varepsilon_{t-1}| \mathbf{1}\{\nu_{t-1} \leq 0\}.$$

如果  $\alpha_1 \neq \alpha_2$ , 则波动率呈现非对称性。一般而言, 对于多数高频金融时间序列,  $\alpha_1 > \alpha_2$  表明负面冲击对条件方差  $h_t$  的影响更大。

Schwert (1989)、Hamilton & Lin (1996) 与 McQueen & Thorley (1993) 发现, 股票收益率在衰退期的波动率高于扩张期的波动率。

马尔可夫区制转换 GARCH 模型的形式如下：

$$\begin{cases} \varepsilon_t = \nu_t \sigma_t, \\ \sigma_t^2 = \psi(S_t) + \beta(S_t) \sigma_{t-1}^2 + \alpha(S_t) \varepsilon_{t-1}^2, \\ \{\nu_t\} \sim \text{i.i.d. } (0, 1), \text{ 其概率密度为 } f(\nu). \end{cases}$$

其中，系数  $\psi(S_t)$ 、 $\beta(S_t)$  和  $\alpha(S_t)$  取决于潜在状态变量  $S_t$ ； $S_t$  为具有固定转移概率的马尔可夫链。例如，

$$\begin{aligned} P(S_t = 1 | S_{t-1} = 0) &= p_{01}, \\ P(S_t = 0 | S_{t-1} = 1) &= p_{10}. \end{aligned}$$

一些实证研究表明，股票收益率在衰退期的波动率高于扩张期的波动率。还可考虑扰动项分布的形状参数随状态变量而变的情形。在这种情况下，扰动项  $\{\nu_t\}$  为均值为 0、方差为 1 的 MDS（非独立同分布）序列，而非 i.i.d. 序列。

## 5.4 IGARCH 模型

假设弱平稳性成立，并且给定  $E(\varepsilon_t | \mathcal{F}_{t-1}) = 0$ ，根据迭代期望定律可得  $E(\varepsilon_t^2) = E(\sigma_t^2)$ 。此外，由于

$$\sigma_t^2 = \omega + \beta \sigma_{t-1}^2 + \alpha \varepsilon_{t-1}^2,$$

有

$$E(\sigma_t^2) = \omega + \beta E(\sigma_{t-1}^2) + \alpha E(\varepsilon_{t-1}^2).$$

若记平稳方差为  $\sigma^2$ ，则

$$\sigma^2 = \omega + (\beta + \alpha)\sigma^2 \Rightarrow \sigma^2 = \frac{\omega}{1 - \beta - \alpha}.$$

为保证弱平稳性，需要满足

$$\omega > 0, \quad \beta \geq 0, \quad \alpha \geq 0, \quad \beta + \alpha < 1.$$

对 GARCH(1,1) 而言，当  $\beta + \alpha = 1$  时称为单整(Integrated)GARCH(1,1)，即 IGARCH(1,1)。因此，IGARCH 是带“单位根”的 GARCH 模型。与 ARIMA 模型类似，IGARCH 的一个重要特征是历史平方扰动对  $\sigma_t^2$  的影响持续存在、不会消失。这可由下面的表示看出。令

$$u_t = \varepsilon_t^2 - \sigma_t^2 \quad (\text{MDS}),$$

则由  $\varepsilon_{t-1}^2 = u_{t-1} + \sigma_{t-1}^2$  得

$$\sigma_t^2 = \omega + (\alpha + \beta)\sigma_{t-1}^2 + \alpha u_{t-1}.$$

在 IGARCH 情形 ( $\alpha + \beta = 1$ ) 下，

$$\sigma_t^2 = \omega + \sigma_{t-1}^2 + \alpha u_{t-1}.$$

递归可得

$$\sigma_t^2 = \omega t + \sigma_0^2 + \alpha \sum_{l=1}^{t-1} u_{t-l},$$

其中  $\sigma_0^2$  为  $t = 0$  的条件方差，从而

$$E(\sigma_t^2 | \mathcal{F}_0) = \omega t + \sigma_0^2.$$

类似地，平方过程有 ARMA(1,1) 表示。一般情形：

$$\varepsilon_t^2 = \omega + (\alpha + \beta) \varepsilon_{t-1}^2 - \beta u_{t-1} + u_t,$$

故 IGARCH(1,1) 的 ARMA(1,1) 表示为

$$\varepsilon_t^2 = \omega + \varepsilon_{t-1}^2 - \beta u_{t-1} + u_t.$$

对于 IGARCH 模型，波动率过程  $\{\sigma_t^2\}$  或平方过程  $\{\varepsilon_t^2\}$  具有“单位根”特性；当滞后阶数  $j \rightarrow \infty$  时，冲击  $u_{t-j}$  对  $\sigma_t^2$  或  $\varepsilon_t^2$  的影响不会消失。满足 IGARCH 的  $\varepsilon_t$  的无条件方差非良好定义（即  $Var(\varepsilon_t) = \infty$ ）。这对资产的超额收益而言并不合理，因为其无条件波动应是有限的。从理论角度看，IGARCH(1,1) 现象可能由波动率水平的偶然变化（即截距的结构性突变）引起；波动率持续性的实际成因仍有待进一步研究。

尽管 ARIMA 过程与 IGARCH 过程中的冲击都不会随时间消失，但均值回归模型与方差整合过程之间仍存在差异。[Nelson \(1990b\)](#) 给出了 IGARCH 的若干渐近性质。

IGARCH(1,1) 的一个特例是  $\omega = 0$ 。这对应 J.P. Morgan ([1997](#)) RiskMetrics 中的指  
数平滑波动率：当  $\omega = 0$  时，

$$E(\sigma_{t+j}^2 | \mathcal{F}_{t-1}) = \sigma_t^2.$$

在这种设定下，通常认为  $\sigma_t^2$  的分布呈退化趋势，即随着  $t \rightarrow \infty$ ， $\sigma_t^2 \rightarrow 0$  的概率为 1。

### RiskMetrics

J.P. Morgan ([1997](#)) 提出的 RiskMetrics 模型是 IGARCH(1,1) 的一个特例。其采用指  
数平滑法计算时变波动率：

$$\begin{cases} \varepsilon_t = \nu_t \sigma_t, \\ \sigma_t^2 = (1 - \beta) \sum_{j=1}^{\infty} \beta^{j-1} \varepsilon_{t-j}^2, \quad \beta \in (0, 1), \\ \{\nu_t\} \sim i.i.d. N(0, 1). \end{cases}$$

也可写成递推形式：

$$\sigma_t^2 = \beta \sigma_{t-1}^2 + (1 - \beta) \varepsilon_{t-1}^2.$$

该式对应于 IGARCH(1,1) 过程（其中  $\omega = 0$ ,  $\alpha = 1 - \beta$ ）。常数  $\beta$  称为衰减因子，控制历史波动率对当前波动率的影响衰减至零的速度。对于大多数日度金融时间序列， $\beta = 0.94$  是较为常见的设定。RiskMetrics 模型常用于金融风险管理领域。

如果  $\omega > 0$ ，则过程  $\{\varepsilon_t\}$  不是弱平稳的，因为  $E(\varepsilon_t^2) = \infty$ 。

IGARCH(1,1) 不具弱平稳性；关于严格（严）平稳，可用 Jensen 不等式得到

$$E[\log(\beta + \alpha u_t^2)] \leq \log(\beta + \alpha E[u_t^2]) = \log(\beta + \alpha) = 0.$$

因此即使  $\beta + \alpha$  略大于 1，也可能出现  $E[\log(\beta + \alpha u_t^2)] < 0$  的情形；但需注意：IGARCH 的无条件方差发散，上述结论并不意味着其具有良好的二阶平稳性。

IGARCH(1,1) 模型可用 `rugarch` 包估计：

```

1 # 设置模型
2 spec <- ugarchspec(
3 variance.model = list(model = "iGARCH", garchOrder = c(1,1)),
4 mean.model = list(armaOrder = c(0,0)),
5 distribution.model = "norm"
6)
7 # 估计
8 fit <- ugarchfit(spec, data)

```

## 5.5 ARCH、GARCH 模型的估计

这里以 GARCH(1,1) 模型为例，从理论和实操角度讨论 ARCH、GARCH 模型的估计。考虑

$$\varepsilon_t = \nu_t \sigma_t,$$

$$\sigma_t^2(\theta) = \omega + \beta \sigma_{t-1}^2(\theta) + \alpha \varepsilon_{t-1}^2,$$

其中  $t = 2, \dots, T$ ，待估参数为  $\theta = (\omega, \beta, \alpha)'$ 。ARMA 模型从条件期望建模，ARCH/GARCH 模型从条件方差建模；此处  $\varepsilon_t$  可为条件期望模型（如 ARMA）的残差。通常假设  $\nu_t$  服从正态分布；对许多金融时间序列，正态假设可能不合理，此时可假设  $\nu_t$  服从学生  $t$  分布 (Student's  $t$ -distribution)。

对应的对数极大似然函数为

$$\ell(\theta) = \sum_{t=2}^T \ell_t(\theta), \quad \ell_t(\theta) = -\frac{1}{2} \log \sigma_t^2(\theta) - \frac{1}{2} \left( \frac{\varepsilon_t}{\sigma_t(\theta)} \right)^2,$$

其中  $\sigma_t^2(\theta)$  由初始值递归得到。记  $\hat{\theta}$  为在参数空间  $\Theta$  上最大化  $\ell(\theta)$  的值。

对 GARCH(1,1) 可采用以下初始设定：

1.  $\sigma_1^2(\theta) = \frac{\omega}{1 - \beta - \alpha}$ ;
2.  $\sigma_1^2 = \frac{1}{T} \sum_{t=1}^T \varepsilon_t^2$  ( $T$  为样本量);
3.  $\sigma_1^2 = \varepsilon_1^2$ ;
4. 将  $\sigma_1^2$  作为未知参数与  $\theta$  一并估计。

第一种假定序列弱平稳，并令初始值为无条件方差；第二种不作平稳性约束，令初始值为波动率均值；第三种为任意设定，不强加平稳性要求；第四种不作平稳性限制，但需额外估计参数  $\sigma_1^2$ 。

对数似然的导数递归为

$$\begin{aligned} \frac{\partial \ell_t}{\partial \theta}(\theta) &= -\frac{1}{2} (\varepsilon_t^2(\theta) - 1) \frac{\partial \log \sigma_t^2(\theta)}{\partial \theta}, \quad \varepsilon_t^2(\theta) = \frac{\varepsilon_t^2}{\sigma_t^2(\theta)}, \\ \frac{\partial^2 \ell_t}{\partial \theta \partial \theta'}(\theta) &= -\frac{1}{2} (\varepsilon_t^2(\theta) - 1) \frac{\partial^2 \log \sigma_t^2(\theta)}{\partial \theta \partial \theta'} + \frac{1}{2} \varepsilon_t^2(\theta) \frac{\partial \log \sigma_t^2(\theta)}{\partial \theta} \frac{\partial \log \sigma_t^2(\theta)}{\partial \theta'}, \\ \frac{\partial \log \sigma_t^2(\theta)}{\partial \theta} &= \frac{1}{\sigma_t^2(\theta)} \frac{\partial \sigma_t^2(\theta)}{\partial \theta}, \end{aligned}$$

其中  $t = 2, \dots, T$ , 其递推为

$$\frac{\partial \sigma_t^2}{\partial \omega} = 1 + \beta \frac{\partial \sigma_{t-1}^2}{\partial \omega}, \quad \frac{\partial \sigma_t^2}{\partial \beta} = \sigma_{t-1}^2 + \beta \frac{\partial \sigma_{t-1}^2}{\partial \beta}, \quad \frac{\partial \sigma_t^2}{\partial \alpha} = \varepsilon_{t-1}^2 + \beta \frac{\partial \sigma_{t-1}^2}{\partial \alpha}.$$

在较强条件下（例如误差分布  $\nu_t$  为条件正态分布），极大似然估计量是一致、渐近正态且有效的；在较弱条件下（例如不要求  $\nu_t$  条件正态），估计量仍是一致、渐近正态。尤其当条件均值与条件方差设定正确（半强 GARCH）时，上述结论成立。在此更一般设定下，估计过程常称为拟极大似然估计（quasi-maximum likelihood estimator, QMLE）。QMLE 中“quasi”意指估计不要求假定误差为正态分布，甚至无需指定误差分布形式，只需保证条件均值与条件方差的设定是正确的。

**定理 5.2:** 在 *Bollerslev & Wooldridge (1992)* 的条件下，

$$\sqrt{T}(\hat{\theta} - \theta_0) \xrightarrow{d} N(0, \mathcal{J}^{-1} \mathcal{I} \mathcal{J}^{-1}),$$

其中

$$\mathcal{J} = E\left[-\frac{\partial^2 \ell_t(\theta_0)}{\partial \theta \partial \theta'}\right], \quad \mathcal{I} = E\left[\frac{\partial \ell_t(\theta_0)}{\partial \theta} \frac{\partial \ell_t(\theta_0)}{\partial \theta'}\right].$$

令  $\nu_t$  为独立同分布 (i.i.d.)，则信息矩阵  $\mathcal{I}$  与 Fisher 信息矩阵  $\mathcal{J}$  成比例，记为  $\mathcal{I} \propto \mathcal{J}$ 。特别地，

$$\mathcal{I} = \frac{1}{4} E\left[(\nu_t^2 - 1)^2\right] E\left(\frac{\partial \log \sigma_t^2(\theta_0)}{\partial \theta} \frac{\partial \log \sigma_t^2(\theta_0)}{\partial \theta'}\right),$$

$$\mathcal{J} = \frac{1}{2} E\left(\frac{\partial \log \sigma_t^2(\theta_0)}{\partial \theta} \frac{\partial \log \sigma_t^2(\theta_0)}{\partial \theta'}\right),$$

$$\mathcal{J}^{-1} \mathcal{I} \mathcal{J}^{-1} = E\left[(\nu_t^2 - 1)^2\right] \left\{ E\left(\frac{\partial \log \sigma_t^2(\theta_0)}{\partial \theta} \frac{\partial \log \sigma_t^2(\theta_0)}{\partial \theta'}\right) \right\}^{-1}.$$

当  $\nu_t \sim N(0, 1)$  时，可得

$$E\left[(\nu_t^2 - 1)^2\right] = E(\nu_t^4) - 1 = 2.$$

最简单的做法是假设误差服从高斯（正态）分布。其二，假设误差  $\nu_t$  为独立同分布 (i.i.d.) 但非高斯分布。此时

$$Avar(\hat{\theta}) = E\left[(\nu_t^2 - 1)^2\right] \mathcal{J}^{-1},$$

其中  $E\left[(\nu_t^2 - 1)^2\right]$  一般不等于 2。可用一致估计式

$$\left(\frac{1}{T} \sum_{t=1}^T (\hat{\nu}_t^2 - 1)^2\right) \hat{\mathcal{J}}^{-1}$$

来估计渐近方差。若仅假设  $\nu_t$  与  $\nu_t^2 - 1$  为鞅差分序列 (MDS)，应采用三明治估计量

$$\hat{\mathcal{J}}^{-1} \hat{\mathcal{I}} \hat{\mathcal{J}}^{-1},$$

其在一般条件下是一致的，因为  $\hat{\mathcal{J}} \xrightarrow{p} \mathcal{J}$ 、 $\hat{\mathcal{I}} \xrightarrow{p} \mathcal{I}$ 。上式亦可用于参数置信区间与假设检验。例如，若原假设为“序列无异方差” ( $\beta = \alpha = 0$ )，更一般的约束  $R\theta = r$  ( $R$  为  $q \times p$ ,  $q < p$ )，

$r$  为  $q \times 1$  向量) 的 Wald 统计量为

$$W = T(\widehat{R\theta} - r)' \left( R \widehat{\mathcal{J}}^{-1} \widehat{\mathcal{I}} \widehat{\mathcal{J}}^{-1} R' \right)^{-1} (\widehat{R\theta} - r),$$

在大样本下  $W \sim \chi_q^2$ 。

下面的 R 代码用于分析与建模上证指数 (SSE Index) 的日对数收益率。首先, 加载存放于指定路径的 CSV 文件中的上证指数数据, 并计算其对数收益率。其次, 调用 `auto.arima` 自动选择对数收益率的 ARMA 模型参数, 为后续波动率建模提供基础; 结果显示最合适的选择为 ARMA(0,0), 与市场有效性理论 (Efficient Market Hypothesis, 简称 EMH) 相符<sup>1</sup>。最后, 我们分别拟合多种 GARCH 模型: 标准 GARCH(1,1)、EGARCH(1,1)、GJR-GARCH(1,1)、门限 GARCH, 以及马尔可夫区制转换 GARCH, 并输出各模型的拟合结果。注: 门限 ARCH 模型只需将 `garchOrder = c(1,1)` 调整为 `garchOrder = c(1,0)` 即可。

```

1 set.seed (123)
2
3 # 设置工作目录 (在 RStudio 中)
4 if (requireNamespace("rstudioapi", quietly = TRUE) &&
5 rstudioapi::isAvailable()) {
6 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
7 }
8
9 # 加载包
10 library (rugarch)
11 library (forecast)
12 library (zoo)
13
14 # 载入数据
15 sse_data <- read.csv ("sse.csv", stringsAsFactors = FALSE)
16 sse_data$date <- as.Date (sse_data$date)
17
18 # 计算对数收益率
19 sse_data$logReturns <- c (NA,diff (log (sse_data$SSE) , lag = 1))
20 sse_data <- sse_data[-1,]
21
22 # 自动选择 ARMA 模型
23 arma_fit <- auto.arima (sse_data$logReturns, seasonal = FALSE, stepwise =
24 FALSE, approximation = FALSE)
25 summary (arma_fit)
26
27 # 拟合 GARCH(1,1)
28 spec_garch <- ugarchspec (variance.model = list (model = "sGARCH",
29 garchOrder = c (1, 1)) ,
30 mean.model = list (armaOrder = c (0, 0) , include.mean = TRUE))
31 garch_fit <- ugarchfit (spec = spec_garch, data = sse_data$logReturns)
32
33 # 拟合 EGARCH(1,1)
34 spec_egarch <- ugarchspec (variance.model = list (model = "eGARCH",
35 garchOrder = c (1, 1)) ,
36 mean.model = list (armaOrder = c (0, 0) , include.mean = TRUE))

```

<sup>1</sup>市场有效性理论认为价格已反映全部公开信息, 收益近似不可预测。ARMA(0,0) 被选中意味着样本中无显著的自回归或移动平均成分。参见第 六 章。

```

34 egarch_fit <- ugarchfit (spec = spec_egarch, data = sse_data$LogReturns)
35
36 # 拟合 GJR-GARCH(1,1)
37 spec_gjrgarch <- ugarchspec (variance.model = list (model = "gjrGARCH",
38 garchOrder = c (1, 1)) ,
39 mean.model = list (armaOrder = c (0, 0) , include.mean = TRUE))
40 gjrgarch_fit <- ugarchfit (spec = spec_gjrgarch, data = sse_data$LogReturns
41
42 # 设定门限ARCH (TARCH) 模型
43 tgarch_spec <- ugarchspec (
44 variance.model = list (model = "fGARCH", submodel = "TGARCH", garchOrder =
45 c (1, 1)) ,
46 mean.model = list (armaOrder = c (0, 0) , include.mean = TRUE)
47
48 # 设定马尔可夫区制转换GARCH模型
49 library (MSwM)
50
51 # 创建数据框
52 data_df <- data.frame (log_returns = sse_data$LogReturns)
53
54 # 设定模型公式
55 formula <- log_returns ~ 1
56
57 # 使用msmFit来拟合马尔可夫状态切换模型
58 # 设置k = 2 (2个状态) , 并确保sw参数的长度与系数数量匹配
59 model_ms_garch <- msmFit (formula, data = data_df, k = 2, sw = c (TRUE,
60 TRUE))
61
62 # 汇总输出
63 garch_fit
64 egarch_fit
65 gjrgarch_fit
66 tarch_fit
67 model_ms_garch

```

## 5.6 ARCH、GARCH 模型的诊断检验

在对波动聚类进行建模之前，需先检验其是否存在。对高频金融数据，这通常不是难点；而对宏观经济序列，由于更为平滑且样本量较小，证据可能较弱。因此，在建立波动率动态之前，采用合适的方法检验波动聚类是否存在至关重要。

令  $Y_t$  为下列模型的回归误差：

$$Y_t = \mu(\mathcal{F}_{t-1}, \theta_0) + \varepsilon_t.$$

上式从条件期望层面对  $Y_t$  建模；随后通过分析残差  $\varepsilon_t$  的条件方差来检验波动率动态。

原假设（条件同方差）：

$$\mathbb{H}_0 : \text{Var}(\varepsilon_t | \mathcal{F}_{t-1}) = \sigma_0^2, \quad \exists \sigma_0^2 > 0.$$

备择假设 (条件异方差):

$$\mathbb{H}_A : \text{Var}(\varepsilon_t | \mathcal{F}_{t-1}) \text{ 非常数 a.s. (等价地: 对任意常数 } \sigma^2 > 0, \text{Var}(\varepsilon_t | \mathcal{F}_{t-1}) \neq \sigma^2 \text{ a.s.)}.$$

### 5.6.1 拉格朗日乘子检验 (LM 检验)

考虑时间序列回归模型:

$$Y_t = \mu(\mathcal{F}_{t-1}, \theta_0) + \varepsilon_t,$$

其中参数  $\theta_0$  可通过非线性最小二乘或拟极大似然估计得到。一般而言, 假设条件均值模型  $\mu(\mathcal{F}_{t-1}, \theta)$  对真实条件均值  $\mu_t$  的设定是正确的, 即存在某个参数  $\theta_0$  使得  $\mu_t = \mu(\mathcal{F}_{t-1}, \theta_0)$ , 从而

$$E(\varepsilon_t | \mathcal{F}_{t-1}) = 0.$$

条件均值模型的正确设定对识别 ARCH 效应至关重要。若  $\mu(\mathcal{F}_{t-1}, \theta)$  是对  $E(Y_t | \mathcal{F}_{t-1})$  的错误设定, 则估计残差

$$\hat{\varepsilon}_t = \underbrace{\varepsilon_t}_{\text{真实误差}} + \underbrace{[\mu_t - \mu(\mathcal{F}_{t-1}, \theta_0)]}_{\text{模型误设误差, 依赖 } \mathcal{F}_{t-1}},$$

后项依赖于  $\mathcal{F}_{t-1}$ , 即便真实误差  $\varepsilon_t$  条件同方差,  $\hat{\varepsilon}_t$  也可能呈现 ARCH 效应。

现在, 介绍如何利用随机样本  $\{Y_t\}_{t=1}^T$  对 ARCH 效应进行检验。LM 检验 (Engle 1982) 的基本步骤如下:

1. 得到残差估计量:

$$\hat{\varepsilon}_t = Y_t - \mu(\mathcal{F}_{t-1}, \hat{\theta}),$$

其中  $\hat{\theta}$  是  $\theta$  的一致估计量 (可以采用最小二乘法或拟极大似然法得到)。

2. 进行以下辅助自回归, 并用 OLS 估计:

$$\hat{\varepsilon}_t^2 = \phi_0 + \sum_{j=1}^p \phi_j \hat{\varepsilon}_{t-j}^2 + u_t, \quad t = p+1, \dots, T,$$

其中  $p$  为预先给定的滞后阶数。计算该辅助回归的  $TR^2$  或  $(T-p)R^2$ , 此处  $R^2$  为回归的决定系数 (coefficient of determination)。

3. 在原假设下 (不存在 ARCH 效应), 当  $T \rightarrow \infty$  时,

$$TR^2 \xrightarrow{d} \chi_p^2.$$

$\hat{\theta}$  的估计不确定性不影响  $TR^2$  或  $(T-p)R^2$  的极限分布: 可用其估计量  $\hat{\theta}$  替代未知的  $\theta_0$ 。

LM 检验统计量的计算较为简单。然而, 当  $p$  较大时, 检验功效可能下降, 因为辅助自回归对各滞后阶的权重被设定为相等, 从而削弱了统计量的辨识能力。

### 5.6.2 Box–Pierce 型混成检验

基于与 Box–Pierce 混成检验类似的思路, McLeod & Li (1983) 提出了如下统计量:

$$\text{ML}(p) = T \sum_{j=1}^p \hat{\rho}_2^2(j),$$

其中  $\hat{\rho}_2(j)$  为残差平方序列  $\{\hat{\varepsilon}_t^2\}$  的样本自相关系数, 定义为

$$\hat{\rho}_2(j) = \frac{\hat{\gamma}_2(j)}{\hat{\gamma}_2(0)},$$

$\hat{\gamma}_2(j)$  为  $\{\hat{\varepsilon}_t^2\}$  的样本自协方差。该统计量在原假设  $H_0$  (不存在 ARCH 效应) 下, 渐近服从卡方分布  $\chi_p^2$ 。

与 Engle (1982) 的 LM 检验类似, 在原假设下有

$$\text{ML}(p) \xrightarrow{d} \chi_p^2 \quad \text{当 } T \rightarrow \infty,$$

因此 McLeod & Li (1983) 的统计量与 LM 检验统计量渐近等价。

需要注意, 条件均值模型  $\mu(\mathcal{F}_{t-1}, \theta)$  的参数估计不确定性对  $\text{ML}(p)$  的极限分布无影响; 无需因估计的参数个数而调整卡方分布自由度。直观地说, 用  $\varepsilon_t^2$  替代  $\hat{\varepsilon}_t^2$  的影响为  $O_p(T^{-1})$  量级, 因而对  $\text{ML}(p)$  的渐近分布影响可以忽略。

下面这段代码实现了对 ARCH 效应检验的 LM 检验和 Box–Pierce 型混成检验。首先, 模拟了一个 AR(1) 过程的时间序列数据, 并拟合了一个 AR(1) 模型来估计残差。然后, LM 检验通过对残差的平方进行回归, 计算统计量以判断是否存在 ARCH 效应。接着, Box–Pierce 检验通过计算残差平方的自相关系数, 进一步检验 ARCH 效应。最终, 代码输出了两个检验的统计量。

```

1 library (lmtest)
2 library (fUnitRoots)
3
4 # 模拟时间序列数据 (以 AR(1) 模型为例)
5 set.seed (123)
6 n <- 100
7 Y <- arima.sim (model = list (ar = 0.5) , n = n)
8
9 # 拟合 AR(1) 模型
10 model <- lm (Y[2:n] ~ Y[1: (n-1)])
11
12 # 计算残差
13 residuals <- model$residuals
14
15 # 首先考虑LM检验统计量
16 # 进行辅助回归, 计算残差的平方与滞后平方的自回归模型
17 p <- 1 # 设置滞后阶数
18 residuals_squared <- residuals^2
19 auxiliary_model <- lm (residuals_squared[(p+1) :n] ~ residuals_squared[1:
19 (n-p)])
20
21 # 计算 LM 检验统计量 TR^2 或 (T-p) R^2
22 lm_stat <- summary (auxiliary_model)$r.squared * (n - p)

```

```

23
24 # 输出 LM 检验统计量
25 lm_stat
26
27 # 然后考虑 Box-Pierce 型混成检验统计量
28 sigma_hat_squared <- mean (residuals_squared)
29 demeaned_residuals_squared <- residuals_squared - sigma_hat_squared
30
31 # 计算计算自相关系数
32 gamma_hat_0 <- var (demeaned_residuals_squared)
33 p <- 1 # 这里令阶数为 1, 可以适当调整。
34 gamma_hat_j <- sapply (1:p, function (j) mean (demeaned_residuals_squared [
35 (j+1) :n] * demeaned_residuals_squared[1: (n-j)], na.rm = TRUE))
36 rho_hat <- gamma_hat_j / gamma_hat_0
37
38 # 计算 Box-Pierce 型混成检验统计量 ML (p)
39 ml_stat <- n * sum (rho_hat^2)
40
41 # 输出 Box-Pierce 型混成检验统计量
42 ml_stat

```

## 5.7 多元波动率模型

接着我们分析多个金融资产或时间序列的波动率。在金融市场中，资产之间通常是互相关联的，尤其在经济危机或市场大幅波动时，多个资产可能同时出现大的波动。多元波动率模型通过捕捉这些资产之间的相关性，可以为分析资产间的依赖关系提供量化工具。常见的多元波动率模型包括 BEKK 模型 (Baba, Engle, Kraft, and Kroner) 和 DCC-GARCH 模型 (Dynamic Conditional Correlation GARCH) 等。[Tsay \(2013\)](#) 一书的第十章以及[Linton \(2019\)](#) 第 11.14 节也对多元波动率模型进行了详细的介绍。该章从基础的指数加权估计开始，逐步深入到多种常见的多元 GARCH 模型，如对角化向量化 (VEC) 模型和 BEKK 模型。书中还探讨了相关性模型，包括常数相关性模型、时变相关性模型和动态相关性模型，进一步扩展到更高维度的波动率模型以及因子-波动率模型。本节通过案例介绍 BEKK-GARCH 以及 DCC-GARCH 模型。

### 5.7.1 案例：基于 BEKK-GARCH 的股指期现货的波动率研究

股指期货作为重要的金融衍生品，自诞生之初即受到学界与监管部门的广泛关注，尤其是在“异常波动”情形下对现货市场的影响。1982 年，美国堪萨斯期货交易所推出全球首张股指期货合约。期货合约因其套期保值与风险对冲功能而受到青睐，但其“杠杆—双刃剑”属性亦可能在特定阶段放大市场波动。鉴于交易成本较低、杠杆率较高，期货市场更易吸引投机者，进而可能提高期货与现货市场的波动率，使其“稳定市场”的功能难以完全实现。以 2015 年二季度为例，我国 A 股主要指数先后触及阶段高位后快速下跌，上证综指由 5178.19 点降至 8 月 26 日的 2850.71 点，沪深 300 指数由 5380.43 点降至 2952.01 点。其间舆论一度将异常波动部分归因于股指期货。因此，系统研究股指期货与现货之间的波动率溢出效应，对于理解金融市场的稳定性与完善监管政策具有重要意义。

本节以 2010 年 4 月 16 日至 2018 年 8 月 23 日为样本区间，研究股指与股指期货之

间的波动率溢出，并采用 VAR–BEKK–GARCH 模型 (Engle & Kroner 1995)：

$$R_t = \mu + \sum_{i=1}^{k-1} \Gamma_i R_{t-i} + \varepsilon_t, \quad \varepsilon_t | \mathcal{F}_{t-1} \sim N(0, H_t)$$

$$H_t = C' C + A' \varepsilon_{t-1} \varepsilon'_{t-1} A + B' H_{t-1} B$$

其中  $R_t = \begin{pmatrix} r_{f,t} \\ r_{s,t} \end{pmatrix}$ ,  $\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$ ,  $\Gamma_i = \begin{pmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{pmatrix}$ ,  $\varepsilon_t = \begin{pmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{pmatrix}$ ,  $H_t = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix}$ 。

$r_{s,t}$  表示股指（收盘价）的对数收益率,  $r_{f,t}$  表示股指期货（当月合约收盘价）的对数收益率； $A, B, C$  为系数矩阵。 $A, B$  的对角元素刻画各自市场的自身冲击与历史波动对本市场波动的影响，非对角元素反映期现货两大市场之间的波动溢出与信息传递。 $\mathcal{F}_{t-1}$  为  $t-1$  时刻的信息集合,  $H_t$  为  $t$  时刻的条件协方差矩阵。

VAR 部分的滞后阶数可依据 HQ (Hannan–Quinn)、AIC 或 BIC 选择；三种准则的结果可能略有差异。需注意，这些准则通常基于同方差扰动的假定，仅可作为初步参考；因此还应对残差向量  $\varepsilon_t$  进行相关性与异方差性诊断。鉴于样本仅包含股指与股指期货两类资产，本文采用二元 BEKK–GARCH(1,1) 模型进行估计与检验。

$$H_t = \begin{pmatrix} c_{11} & c_{12} \\ 0 & c_{22} \end{pmatrix}' \begin{pmatrix} c_{11} & c_{12} \\ 0 & c_{22} \end{pmatrix} + \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}' \begin{pmatrix} \varepsilon_{1,t-1}^2 & \varepsilon_{1,t-1}\varepsilon_{2,t-1} \\ \varepsilon_{2,t-1}\varepsilon_{1,t-1} & \varepsilon_{2,t-1}^2 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

$$+ \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}' \begin{pmatrix} h_{11,t-1} & h_{12,t-1} \\ h_{21,t-1} & h_{22,t-1} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}. \quad (5.6)$$

等式 (5.6) 阐述了信息与波动率如何在股指期货与股票市场之间传递。具体而言，波动率在两大市场之间的传递主要通过以下三条渠道：

1. 直接经由滞后均方误差  $(\varepsilon_{1,t-1}^2, \varepsilon_{2,t-1}^2)$ ；该渠道反映来源于收益率冲击角度的波动溢出效应 (the return innovation effect)。
2. 通过滞后自身条件方差  $(h_{11,t-1}, h_{22,t-1})$  以及协方差项  $h_{12,t-1}$ ；其中，自身条件方差刻画波动的持续性，协方差项反映历史信息角度的跨市场波动溢出。
3. 间接经由误差的交叉乘积  $(\varepsilon_{1,t-1}\varepsilon_{2,t-1})$ ；可理解为潜在的双向冲击 (the potential presence of bi-directional shock)。

来自资产收益的当期冲击首先通过均方误差  $(\varepsilon_{1,t-1}^2, \varepsilon_{2,t-1}^2)$  产生第一轮作用，随后经由滞后条件方差  $H_{t-1}$  对当期条件方差  $H_t$  产生第二轮影响。换言之，这两类作用均体现在方差方程 (5.6) 中。因此，由股票市场传向股指期货市场的信息可用系数  $a_{21}^2$  度量，反向则用  $a_{12}^2$  度量；相应地，历史信息渠道的溢出由  $b_{21}^2$  (股票  $\rightarrow$  期货) 与  $b_{12}^2$  (期货  $\rightarrow$  股票) 加以刻画。

表 5.1: 沪深 300 股指期货（当月合约）与沪深 300 股指的描述性统计量

|                        | 均值                        | 标准差        | 偏度         | 峰度       |
|------------------------|---------------------------|------------|------------|----------|
| 沪深 300 股指期货（当月合约）日度收益率 | $-1.66081 \times 10^{-5}$ | 0.01667454 | -0.6091457 | 9.400874 |
| 沪深 300 股指日度收益率         | $-5.34889 \times 10^{-6}$ | 0.01489299 | -0.7382722 | 5.066927 |

数据来源：中国金融期货交易所（WIND 数据库）

以下 R 代码演示如何使用 BEKK-GARCH 模型对股指与股指期货的波动率进行建模与检验，从而刻画两者的动态关联与风险溢出效应。

```

1 # 清空工作空间
2 rm (list = ls ())
3
4 # 设置工作目录
5 if (requireNamespace("rstudioapi", quietly = TRUE) &&
6 rstudioapi::isAvailable()) {
7 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
8 }
9
10 # 加载所需的库
11 library (readxl) # 用于读取Excel文件
12 library (forecast) # 用于时间序列分析
13 library (vars) # 用于VAR模型分析
14 library (mgarchBEKK) # 用于BEKK-GARCH模型
15
16
17 # 读取数据
18 dat <- read_excel ("沪深300股指期现货收盘价.xlsx")
19
20 # 去除缺失值
21 dat <- na.omit (dat)
22
23 # 提取数据
24 date <- dat$日期
25 LnS <- log (dat$沪深300股指收盘价)
26 LnF <- log (dat$沪深300股指期货当月收盘价)
27
28 # 数据长度
29 n <- length (LnS)
30
31 # 计算收益率
32 date.ret <- date[-1]
33 RS <- LnS[-1] - LnS[-n]
34 RF <- LnF[-1] - LnF[-n]
35
36 # 创建时间序列数据框
37 RS.tSeries <- data.frame (time = date.ret, RS)
38 RF.tSeries <- data.frame (time = date.ret, RF)
39
40 # 将收益率合并为一个矩阵进行VAR分析
41 dat.var <- cbind (RF, RS)
42
43 # 进行VAR模型选择
44 VARselect (dat.var, lag.max = 12, type = "const")
45
46 # 拟合VAR模型
47 var.2c <- VAR (dat.var, p = 3, type = "const")
48
49 # 提取VAR模型的残差（白噪声）

```

```
50 ut <- na.omit (resid (var.2c))
51
52 # 拟合BEKK-GARCH模型
53 estimated <- BEKK (ut)
54
55 # 模型诊断
56 diagnoseBEKK (estimated)
```

估计结果如下：

```
1
2 Number of estimated series : 4060
3 Length of estimated series : 2030
4 Estimation Time : 1.113319
5 Total Time : 1.186359
6 BEKK order : 1 1
7 Eigenvalues : 3.754045 0.6389963 0.3486411 0.1339566
8 aic : -13854.91
9 unconditional cov. matrix : 0.05916318 -0.06260839 -0.06260839 0.06668292
10 var (resid 1) : 0.9546205
11 mean (resid 1) : 0.01773091
12 var (resid 2) : 0.9477698
13 mean (resid 2) : -0.01272984
14 Estimated parameters :
15
16 C estimates:
17 [,1] [,2]
18 [1,] -0.002390247 -0.002540273
19 [2,] 0.000000000 -0.002144994
20
21 ARCH estimates:
22 [,1] [,2]
23 [1,] -0.09025784 0.6614547
24 [2,] -0.18887369 -0.9614542
25
26 GARCH estimates:
27 [,1] [,2]
28 [1,] 0.5162522 -0.08867566
29 [2,] -1.5260502 -0.82329481
30
31 asy.se.coef :
32
33 C estimates, standard errors:
34 [,1] [,2]
35 [1,] 0.0005725315 0.0005194220
36 [2,] 0.0000000000 0.0002547972
37
38 ARCH estimates, standard errors:
39 [,1] [,2]
40 [1,] 0.1223140 0.1334035
41 [2,] 0.1148105 0.1213120
42
43 GARCH estimates, standard errors:
```

```

44 [,1] [,2]
45 [1,] 0.1649024 0.1297107
46 [2,] 0.1729286 0.1368526
47 Called from: diagnoseBEKK (estimated)

```

其中, `C estimates` 对应  $\begin{pmatrix} c_{11} & c_{12} \\ 0 & c_{22} \end{pmatrix}$ , `ARCH estimates` 对应  $\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ , `GARCH estimates` 对应  $\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$ 。从估计结果看, 若  $|a_{12}| > |a_{21}|$ , 说明期货市场的当期冲击对股票市场的影响更为显著。此外, 在历史信息渠道方面, 若  $|b_{21}| > |b_{12}|$ , 则表明期货市场对股票市场的波动溢出效应更强。

本案例样本区间为 2010 年 4 月 16 日至 2018 年 8 月 23 日。该期间中国金融期货交易所对交易规则进行了多次调整(见表 5.2)。为增强稳健性, 建议将样本按制度变动分段进行子样本分析。

### 5.7.2 案例: 基于 DCC-GARCH 模型的股指联动性分析

DCC-GARCH 模型是对经典 GARCH 模型的扩展。GARCH 模型用于描述单一资产或时间序列的条件方差(波动率), 而 DCC-GARCH 模型在此基础上进一步刻画资产之间的动态条件相关性。

DCC-GARCH 模型的估计步骤如下: 首先, 为每个资产分别设定 GARCH 模型; 对各资产的收益率序列, 通常采用 GARCH(1,1) 来刻画其条件方差, 使当期波动取决于过去的波动与误差项。其次, 通过估计各资产的 GARCH 模型得到每期条件方差。再次, 将各资产收益率去均值并除以条件标准差, 得到标准化残差(亦称“标准化收益率”)。最后, 基于这些标准化残差由 DCC 模型估计资产间的动态条件相关性; DCC 通过对条件相关系数矩阵进行动态更新, 能够捕捉金融市场中资产关系随时间的演变。

考虑以下二元 DCC-GARCH 模型:

$$\begin{aligned}
\tilde{r}_t &= H_t^{1/2} \epsilon_t, \quad H_t = D_t S_t D_t, \\
h_{11,t} &= c_{11} + a_{11,1} \tilde{r}_{1,t-1}^2 + g_{11,1} h_{11,t-1}, \\
h_{22,t} &= c_{22} + a_{22,1} \tilde{r}_{2,t-1}^2 + g_{22,1} h_{22,t-1}, \\
S_t &= \begin{pmatrix} 1 & s_{12,t} \\ s_{12,t} & 1 \end{pmatrix}, \quad s_{12,t} = \frac{q_{12,t}}{\sqrt{q_{11,t} q_{22,t}}} \\
q_{12,t} &= (1 - \alpha - \beta) \bar{q}_{12} + \alpha \frac{\tilde{r}_{1,t-1} \tilde{r}_{2,t-1}}{\sqrt{h_{11,t-1} h_{22,t-1}}} + \beta q_{12,t-1}, \\
q_{11,t} &= (1 - \alpha - \beta) \bar{q}_{11} + \alpha \left( \frac{\tilde{r}_{1,t-1}}{\sqrt{h_{11,t-1}}} \right)^2 + \beta q_{11,t-1}, \\
q_{22,t} &= (1 - \alpha - \beta) \bar{q}_{22} + \alpha \left( \frac{\tilde{r}_{2,t-1}}{\sqrt{h_{22,t-1}}} \right)^2 + \beta q_{22,t-1},
\end{aligned}$$

其中,  $\{\tilde{r}_t\}$  表示使用 ARMA 模型调整后的收益率, 即 ARMA 过程的残差项;  $H_t$  为  $\{\tilde{r}_t\}$  的条件方差矩阵,  $\epsilon_t$  表示时点  $t$  的创新项。矩阵  $D_t$  为条件标准差的对角矩阵,  $S_t$  为时点  $t$  的时变条件相关矩阵。该模型含有 8 个参数:  $\theta = (\alpha, \beta, c_{11}, c_{22}, a_{11,1}, a_{22,1}, g_{11,1}, g_{22,1})$ 。模型估计在假设创新项  $\{\epsilon_t\}$  独立同分布且服从二维正态分布的前提下进行, 即  $N(\mathbf{0}, I_2)$ 。

在 DCC-GARCH(1,1) 模型中，各类参数共同刻画收益率序列的动态相关结构。首先，DCC 方程中的  $\alpha$  与  $\beta$  分别反映条件相关性对短期冲击的即时响应及其随时间的衰减速度。其次，GARCH 方程中的  $c_{11}$  与  $c_{22}$  为各序列的方差截距，代表在无新信息时的基准（平均）波动水平。再次，ARCH 参数  $a_{11,1}$ 、 $a_{22,1}$  衡量滞后平方创新对当期方差的影响，体现市场冲击的即时效应；GARCH 参数  $g_{11,1}$ 、 $g_{22,1}$  则度量滞后条件方差对当期波动的持续作用。总体而言，这些参数揭示了波动率集聚现象：参数取值越大，历史波动对当前市场状态的影响越显著。

下述实证以道琼斯工业平均指数 (Dow Jones Industrial Average, 简称 DJIA) 与加拿大标普/TSX 综合指数 (Standard & Poor's/Toronto Stock Exchange Composite Index, 简称 S&P/TSX Composite Index) 的日收益率为样本 (2012 年 1 月 1 日—2020 年 12 月 31 日)。Hong et al. (2024) 使用同一数据对 DCC-GARCH 参数进行结构性变化检验。研究流程如下：首先，自 CSV 文件读取价格数据并剔除周末观测；随后，采用自定义的 `price2return` 函数将价格转换为对数收益率，并对潜在缺失值进行插值处理。接着，利用 `auto.arima` 对两条收益率序列分别拟合 ARIMA 模型并提取残差；这些残差作为 DCC-GARCH 的输入，用以刻画两市场收益的动态条件相关性 (DCC)。模型估计完成后，提取条件相关矩阵并获得两指数之间的动态相关序列，最终借助 `ggplot2` 绘制其随时间的演化轨迹。

```

1 # 清除所有的变量
2 rm (list=ls ())
3
4 # 设置工作目录
5 if (requireNamespace("rstudioapi", quietly = TRUE) &&
6 rstudioapi::isAvailable()) {
7 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
8 }
9
10 # 载入必要的库
11 library (rmgarch) # 多变量GARCH模型
12 library (ggplot2) # 用于绘图
13
14 # 定义一个函数，用于计算收益率
15 price2return<- function (p)
16 { N<-length (p)
17 return.series<- (log (p[2:N]) - log (p[1: (N-1)])) * 100
18 return.series<-na_interpolation (return.series) # 对缺失值进行插补
19 return (return.series) }
20
21 # 读取数据，并去除周末的数据（周末数据为NA）
22 temp.dat <- read.csv ("stock_indices.csv")
23 dat <- temp.dat[setdiff ((1:dim (temp.dat) [1]) , which (temp.dat$weekdays
24 == "Saturday" | temp.dat$weekdays=="Sunday")),]
25
26 # 提取数据列
27 Dow_Jones <- dat$Dow_Jones_Index_Industrial_Average
28 SP <- dat$S.P.TSX_Composite_Index
29
30 # 计算收益率
31 Dow_Jones.return <- price2return (Dow_Jones)
32 SP.return <- price2return (SP)

```

```

33 # 合并收益率数据
34 return_series <- cbind (Dow_Jones.return, SP.return)
35
36 # 使用ARIMA模型拟合数据并提取残差
37 arima_dj <- auto.arima (Dow_Jones.return)
38 arima_sp <- auto.arima (SP.return)
39 residuals_dj <- residuals (arima_dj)
40 residuals_sp <- residuals (arima_sp)
41
42 # 为DCC-GARCH模型准备数据
43 ut <- cbind (residuals_dj, residuals_sp)
44
45 # 定义GARCH(1,1)模型规格
46 garch11.spec = ugarchspec (mean.model = list (armaOrder = c (0,0) , include
47 .mean = FALSE) ,
48 variance.model = list (garchOrder = c (1,1) ,
49 model = "sGARCH") ,
50 distribution.model = "norm")
51
52 # 定义DCC-GARCH模型规格
53 dcc.garch11.spec = dccspec (uspec = multispec (replicate (2, garch11.spec)
54) ,
55 dccOrder = c (1,1) ,
56 distribution = "mvn")
57
58 # 拟合DCC-GARCH模型
59 dcc.fit = dccfit (dcc.garch11.spec, data = ut)
60
61 dynamic_correlation <- rcor (dcc.fit)
62
63 # 提取动态相关系数的非对角元素（即Dow_Jones.return与SP.return之间的相关性）
64 off_diagonal_correlation <- dynamic_correlation[2, 1,] # 提取非对角 (1, 2)
65 相关性
66
67 # 创建数据框以便绘图
68 dynamic_corr_df <- data.frame (
69 time = as.Date (dimnames (dynamic_correlation) [[3]]) ,
70 correlation = off_diagonal_correlation
71)
72
73 # 绘制动态相关图
74 ggplot (dynamic_corr_df, aes (x = time, y = correlation)) +
75 geom_line (color = 'blue') +
76 labs (title = "Dynamic Conditional Correlation (Off-Diagonal) " ,
77 x = "Time" ,
78 y = "Correlation") +
79 theme_minimal () +
80 theme (axis.text.x = element_text (angle = 45, hjust = 1))

```

DCC模型的估计结果如下：

```
1 *-----*
2 * DCC GARCH Fit *
3 *-----*
4
5 Distribution : mvnorm
6 Model : DCC (1,1)
7 No. Parameters : 9
8 [VAR GARCH DCC UncQ] : [0+6+2+1]
9 No. Series : 2
10 No. Obs. : 2385
11 Log-Likelihood : -4411.584
12 Av.Log-Likelihood : -1.85
13
14 Optimal Parameters
15 -----
16 Estimate Std. Error t value Pr (>|t|)
17 [residuals_dj].omega 0.041274 0.008534 4.8364 0.000001
18 [residuals_dj].alpha1 0.222284 0.032933 6.7496 0.000000
19 [residuals_dj].beta1 0.734947 0.030983 23.7210 0.000000
20 [residuals_sp].omega 0.020112 0.005474 3.6738 0.000239
21 [residuals_sp].alpha1 0.188567 0.035891 5.2539 0.000000
22 [residuals_sp].beta1 0.785539 0.033671 23.3301 0.000000
23 [Joint]dcca1 0.072732 0.014655 4.9630 0.000001
24 [Joint]dccb1 0.793087 0.044881 17.6708 0.000000
25
26 Information Criteria
27 -----
28
29 Akaike 3.7070
30 Bayes 3.7288
31 Shibata 3.7070
32 Hannan-Quinn 3.7149
33
34
35 Elapsed time : 0.5664971
```

## 5.8 章节总结

本章聚焦于金融时间序列的波动率模型。首先介绍了 ARCH 模型及其平稳性条件，为理解该模型的稳定性与适用性提供理论依据；接着阐述了 GARCH 模型及其平稳性条件，该模型是对 ARCH 的重要拓展，能更好地捕捉波动聚集性特征；还探讨了考虑正负冲击不对称影响的非对称波动率模型以及具有特殊性质的 IGARCH 模型。模型处理方面，说明了 ARCH 和 GARCH 的估计方法，并介绍了拉格朗日乘子检验与 Box-Pierce 型混成检验等诊断工具，以确保模型的拟合效果与可靠性。最后，引入多元波动率模型，并给出基于 BEKK 和 DCC-GARCH 的案例，展示多元模型在处理多个金融变量波动关系方面的应用，帮助读者将理论应用于实际。

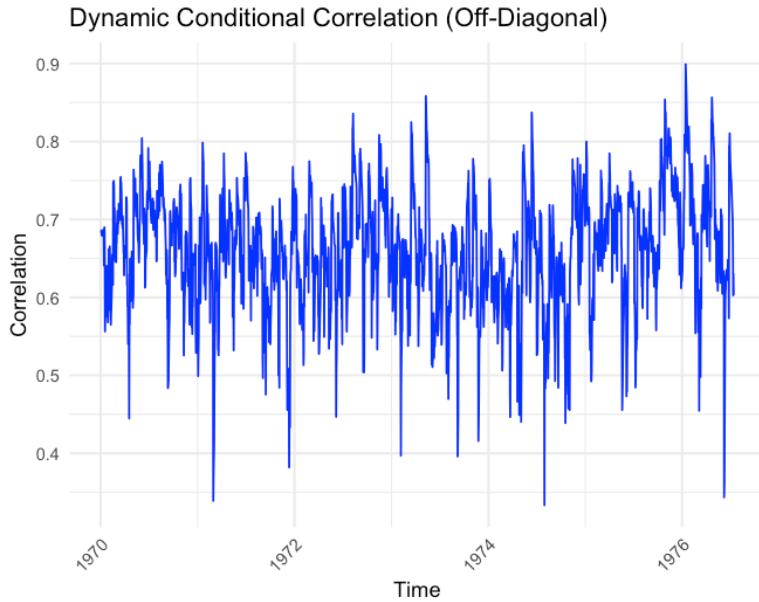


图 5.4: DCC-GARCH 模型的动态条件相关性

## 5.9 习题

1. 为什么波动率模型在金融计量经济学中非常重要?
2. ARCH 模型有哪些重要的统计性质? 其平稳性条件是什么?
3. 假设  $\{\varepsilon_t\}$  为四阶平稳序列。为保证 ARCH(1) 过程的四阶矩  $E(\varepsilon_t^4)$  有限,  $\alpha_1$  需满足何种条件? (设  $\varepsilon_t = \sigma_t z_t$ ,  $z_t \sim \text{i.i.d. } N(0, 1)$ ,  $\sigma_t^2 = \omega + \alpha_1 \varepsilon_{t-1}^2$ , 且  $\omega > 0$ ,  $\alpha_1 \geq 0$ )
4. 假设  $\{\varepsilon_t\}$  弱平稳, 且标准化扰动项  $\{\nu_t\} \sim \text{i.i.d. } N(0, 1)$ 。那么,  $\{\varepsilon_t\}$  的边际分布是高斯分布吗?
5. 对于 GARCH(1,1) 模型  $\sigma_t^2 = \omega + \beta \sigma_{t-1}^2 + \alpha \varepsilon_{t-1}^2$ , 其强平稳与弱平稳条件分别是什么?
6. IGARCH 模型具有哪些渐近性质?
7. 从 A 股市场任意选择一只股票, 下载至少过去一年的日收盘价数据 (可使用 Wind、Choice 智能金融终端、雅虎财经等平台)。将数据导入 Python 或 R, 计算对数收益率  $r_t = \ln(P_t/P_{t-1})$ , 其中  $P_t$  为第  $t$  期收盘价。
  - (a) 利用 ACF (自相关函数) 与 PACF (偏自相关函数) 图初步确定 ARMA( $p, q$ ) 的取值范围;
  - (b) 采用 AIC (赤池信息准则) 与 BIC (贝叶斯信息准则) 在不同  $p, q$  组合中选择最优 ARMA 模型;
  - (c) 估计所选 ARMA 模型并进行残差白噪声检验;
  - (d) 对 ARMA 残差序列进行 ARCH/GARCH 效应检验;
  - (e) 若存在 ARCH/GARCH 效应, 则建立 GARCH( $p, q$ ) 模型, 并用 AIC、BIC 选择最优  $p, q$ ;
  - (f) 从拟合优度、残差特性、对波动率刻画能力等方面比较模型表现, 并说明原因。

8. 为刻画复杂且持续的波动动态, ARCH( $q$ ) 往往需要较大阶数。在实践中, 用普通最小二乘法 (OLS) 得到的  $\{\hat{\beta}_j\}_{j=1}^q$  可能出现负值, 应如何处理?
9. 设  $Y_t = X'_t \beta + \varepsilon_t$ ,  $\varepsilon_t = \sigma_t^{1/2} \nu_t$ ,  $h_t = \psi + \alpha \varepsilon_{t-1}^2$ , 且  $\{\nu_t\} \sim \text{i.i.d. } N(0, 1)$ ; 并且  $\{X_t\}$  与  $\{\nu_t\}$  相互独立。
- 计算  $\beta$  的 OLS 估计量  $\hat{\beta}$ ;
  - 用高斯–马尔可夫定理 (Gauss–Markov theorem) 证明  $\beta$  的 OLS 估计是 BLUE;
  - 计算  $\beta$  的极大似然估计  $\tilde{\beta}$ ;
  - 讨论  $\hat{\beta}$  与  $\tilde{\beta}$  的相对效率并给出理由。
10. 在第 5.7.1 节, 我们讨论了全样本区间内沪深 300 股指期货与股票市场的波动溢出效应。由表 5.2 可见, 为稳定市场, 中金所多次调整股指期货交易规则。请分别对股指与股指期货分段建立 VAR–BEKK–GARCH 模型, 分析其波动溢出效应。

表 5.2: 中金所历次交易规则变化

| 日期         | 非套期保值持仓交易保证金 | 套期保值持仓交易保证金 | 交易手续费  | 投机交易单边持仓限额           |
|------------|--------------|-------------|--------|----------------------|
| 2010/04/16 | 从 15% 到 18%  | 从 15% 到 18% | 0.05%  | 未指定                  |
| 2012/06/01 | —            | —           | 0.035% | 300 手                |
| 2012/06/29 | 12%          | 12%         | —      | —                    |
| 2012/09/01 | —            | —           | 0.025% | 1200 手               |
| 2013/03/12 | —            | —           | —      | 600 手                |
| 2014/09/01 | 10%          | 10%         | —      | —                    |
| 2015/04/10 | —            | —           | —      | 5000 手               |
| 2015/08/03 | —            | —           | 0.023% | —                    |
| 2015/08/26 | 12%          | —           | 0.115% | —                    |
| 2015/08/27 | 15%          | —           | —      | —                    |
| 2015/08/28 | 20%          | —           | —      | —                    |
| 2015/08/31 | 30%          | —           | —      | —                    |
| 2015/09/07 | 40%          | 20%         | 2.3%   | 认定“日内开仓超过 10 手”为异常交易 |
| 2017/02/17 | 30%          | 10%         | 0.46%  | 认定“日内开仓超过 20 手”为异常交易 |



# 6 收益可预测性与有效市场假说

本章主要研究收益可预测性以及有效市场假说。资产收益的可预测性与有效市场假说(EMH)之间的关系至关重要，原因有几个。首先，研究资产收益的可预测性是对 EMH 核心论点的检验——即所有相关信息已经反映在资产价格中，且没有任何投资策略能够持续实现超额回报。该研究促使人们深入审视市场动态与投资者行为，从而揭示市场效率。此外，探索收益的可预测性有助于识别与 EMH 相悖的异常或模式，即市场何时、因何而低效。这些研究可以指导投资者优化投资策略，不仅具有深刻的学术意义，而且还能有效改善投资组合管理，提升金融市场效率。进一步地，了解资产收益的可预测条件，有助于提高市场透明度，并推动制定更为公平的监管政策。

## 6.1 有效市场假说

## 6.2 有效市场假说的提出与发展

有效市场假说是金融经济学中的核心概念之一。EMH 的历史可追溯至 20 世纪 60 年代，由保罗 · A · 萨缪尔森 (Paul A. Samuelson) 与尤金 · F · 法玛 (Eugene F. Fama) 分别提出。EMH 在理论建模与金融资产价格的实证研究中被广泛应用，为价格发现过程提供了基础性见解，并引发了学界的激烈争论。

萨缪尔森最早在 1965 年提出有效市场的思想。他在《证明正确预期的价格随机波动》(Proof that Properly Anticipated Prices Fluctuate Randomly) 一文中 ([Samuelson 1965](#)) 指出，在有效市场中，若价格变动能够恰当反映所有市场参与者的信息与预期，则价格变动应不可预测。萨缪尔森对价格运作机制在确定性与不确定性条件下的长期关注，促成了其与学生的一系列重要研究。他研究可储存商品在存在储存成本与损耗条件下的跨期定价，由此获得了关于有效市场的启发。

与之相比，法玛形成 EMH 的路径有所不同。他对股票价格的统计特性，以及技术分析与基本面分析的争论具有浓厚兴趣 ([Fama 1963, 1965a,b](#), [Fama & Blume 1966](#), [Fama 1970](#))。法玛是较早使用现代计算机开展金融实证研究并首次使用“有效市场”一词的学者。他将 EMH 表述为“价格完全反映所有可用信息”，并通过划分市场参与者可获得的信息集合，使 EMH 假设具备可操作性。法玛对实证分析的热情推动了包括事件研究法在内的方法论与实证贡献，同时也促进了对股票、债券、货币与商品市场一系列实证规律与异常现象的发现。

EMH 的信息效率概念具有一定的反直觉性：市场越高效，其价格变动序列越接近随机；最高效的市场则表现为价格变动完全随机且不可预测。这并非偶然，而是众多积极的市场参与者试图利用其所掌握的信息获利所致。受利润机会驱动，投资者会迅速将哪怕极为细小的信息优势反映到价格中，并迅速消除激发交易的初始套利空间。

十年后 (20 世纪 70 年代)，许多学者在萨缪尔森与法玛的框架上进一步扩展，引入风

险规避投资者，形成“新古典”版本的 EMH；在该版本中，价格变动之所以不可预测，是因为市场价格已充分反映所有可用信息。此后，EMH 又被延伸至多个方向，包括非交易资产（如人力资本）、状态依赖偏好、投资者异质性、信息不对称与交易成本等领域，但总体结论保持一致：在任一时点，证券价格已充分反映所有可用信息。更系统的综述可参见 Lo (2008), Lim & Brooks (2011), Sewell (2011)。

有效市场假说将“价格反映信息”的命题划分为三个层次：弱式有效、半强式有效与强式有效。弱式有效市场假说认为，价格已吸收全部历史信息，因此任何基于历史价格的技术分析或交易策略都无法获得超额收益；半强式有效市场假说进一步主张，价格还会迅速反映所有公开信息（如财务报告、行业动态），投资者据此亦难以获得系统性超额回报；强式有效市场假说最为严苛，认为价格包含所有信息（包括内部信息），因此即便内部人士也无法持续获取超额收益。当然，真实市场运行更为复杂，参与者行为并非总是理性，行为金融学通过情绪、认知偏差等机制解释了市场短期偏离有效性的可能原因。

对 EMH 的检验通常分为三个层面：弱式有效检验关注历史价格信息，半强式有效检验关注公开信息的及时反映，强式有效检验则关注内部信息的影响。借助多种统计方法，研究者评估价格是否能够准确、及时地反映相关信息。尽管 EMH 在理论与实践中面临诸多质疑与挑战，但作为金融学最具影响力的理论之一，它仍为投资者、分析师、政策制定者与学者提供理解与预测市场行为的基本框架。关于市场有效性的检验，见表 6.1。

表 6.1: 有效市场假说的检验

| 市场假说类型 | 特点                  | 检验方法                 | 结论                    |
|--------|---------------------|----------------------|-----------------------|
| 弱式有效   | 历史价格信息已反映在当前股价中     | 检验历史趋势（如游程检验、方差比检验等） | 历史信息不能预测未来股价、无法获得超额收益 |
| 半强式有效  | 所有公开信息已反映在股价中       | 事件研究法                | 公开信息被迅速反映于价格、无法获得超额收益 |
| 强式有效   | 所有信息（含非公开信息）已反映在股价中 | 内幕信息持有者绩效分析          | 即便掌握内部信息亦难以获得持续超额收益   |

### 6.2.1 弱有效市场的检验

很多实证研究都支持弱式 EMH。例如，1980 年至 1984 年的标准普尔 500 指数与抛硬币模型的比较显示，基于历史价格趋势的技术交易策略并不能持续产生利润 (Brealey et al. 2014)。这是因为在弱式有效市场假说下，过去的价格信息被认为与未来价格无关，因此不能用来预测未来的市场走势。Solnik (1973) 对 9 个国家股票市场的日收益率序列相关性进行研究，发现其接近零，这进一步支持了弱式 EMH 的观点。类似地，在第 4.5 节采用 ARMA 模型对上证指数收益率进行建模时，我们也发现上证指数的收益率没有显著自相关性，这一发现与弱式 EMH 相符，即历史信息已经被当前股价所反映。而 Barber & Odean (2000) 的研究关注个人投资者的月交易活动与年化收益之间的关系，并将其与标准普尔 500 指数基金的表现进行比较。该研究发现，标准普尔 500 指数基金的净回报相对稳定，整体高于平均个人投资者的净回报。这进一步表明，频繁交易可能会降低投资者的净收益，而长期投资标准普尔 500 指数基金可能是更为稳健的选择。这些研究结果对投资者具有重要意义：在弱式有效市场上，基于历史价格动态的交易策略难以带来持续超额回报，而长期投资与分散配置更为理智。

弱式有效性检验主要包括两方面：一是对证券价格或收益的随机特征（如自相关性、波动率）的检验；二是对技术分析交易规则预测力与获利性的检验。要检验弱式市场有效性，可采用多种方法。例如，检查收益率时间序列是否存在自相关性；另一种方法是检验技术分析中使用的历史趋势交易规则是否能够持续产生超额回报——若不存在此类规则，则至少在弱式有效性意义上市场是有效的。常用统计方法既包括非参数检验（如游程检验、方差比率检验），也包括参数检验（如扩展的迪基-富勒检验〔Augmented Dickey-Fuller，简称 ADF〕和菲利普斯-佩龙检验〔Phillips-Perron，简称 PP〕）。若检验结果表明历史数据无法预测短期未来价格走势，则支持弱式有效市场假说。

不难看出，有效市场假说（EMH）与部分著名投资策略（尤其是技术分析策略）存在张力：弱式 EMH 认为技术分析难以持续获得超额收益。下面简要介绍 10 种常见的股票图表分析形态，这些形态在技术分析中用于识别潜在趋势与买卖机会：

- 双顶形态（Double Tops）：常预示上涨乏力，可能反转向下。
- 双底形态（Double Bottoms）：常预示下跌结束，可能反转向上。
- 趋势线与突破（Trendline & Breakout）：用趋势线识别走势延续或反转。
- 头肩形态（Head and Shoulders）：典型反转形态，预示由升转跌。
- 三角形（Triangle）：连续形态，价格多延续既有趋势。
- 楔形（Wedge）：提示即将突破并延续原有趋势。
- 三角旗形（Pennant）：短暂整理后延续强势走势。
- 旗形（Flag）：与三角旗形相似，为强劲走势中的短暂停顿。
- 圆弧底（Rounding Bottom）：长期下跌后的缓慢筑底与反转。
- 杯柄形（Cup with Handle）：多次回测后或将向上突破。

半强式有效市场理论认为，所有公开信息（包括公司财报信息）都会迅速反映在股票价格中。早期研究为此提供了实证支持。例如，Ball & Brown (1968) 发现，财报发布后公司股价出现显著变动，表明市场能够迅速吸收并反映财报信息。Foster (1973) 进一步证明了股价对盈利公告的快速反应，从而支持半强式有效市场假说。Keown & Pinkerton (1981) 亦发现，无论是股利公告还是收购尝试，股价都会迅速作出反应。

后续研究将视野扩展至其他信息类型。Asquith & Mullins Jr (1986) 显示，市场对股票分拆等事件也能快速、有效地反应。Hayn (1995) 则聚焦于“盈余反应系数”(earnings response coefficient，简称 ERC)，用于度量财报信息对股价的影响强度。总体来看，这些研究表明，在多种情境下，公开信息能够被市场及时反映于价格之中。

### 6.2.2 强有效市场的检验

在市场效率研究中，许多学者通过分析不同市场数据与策略，对市场反应的速度与准确性进行检验，以验证强式有效市场假说是否成立。Jensen (1969) 的研究指出，资本管理者通常难以持续战胜市场。这一发现为“市场有效性较高”的观点提供了支持：若价格能够充分反映全部可得信息，则通过分析或预测获得超额收益将非常困难。

Meulbroek (1992) 通过对非法内幕交易的实证分析发现，内幕交易在某些情形下具有获得累积异常收益 (cumulative abnormal return, CAR) 的可能性；其研究还表明，内幕交易与股价的即时波动及价格发现速度密切相关。上述结论与 Manne (1966) 以及 Carlton & Fischel (1982) 关于“内幕交易有助于提升资本配置效率”的观点相呼应——即内幕交易通过加快价格发现，降低了多位市场参与者重复收集同一信息的动机，从而可能提高资本配置效率。

由此可见，首先需明确“异常收益”(abnormal returns) 的含义。异常收益是金融市场的核心概念之一，指投资者实际获得的回报与依据某一资产定价模型所预期的正常回报之间的差额。若假设正常收益率为  $R_t^*$ ，则在有效市场假说下，任何资产在时间间隔  $[t, T]$  内的实际回报  $R_T$  应满足：

$$\mathbb{E}(R_T | \mathcal{F}_t) = R_t^*.$$

其中， $R_t^*$  为无风险利率  $R_{ft}$  与风险溢价  $\pi_t$  之和，即  $R_t^* = R_{ft} + \pi_t$ 。这里的  $\pi_t$  由某一资产定价模型决定，通常假设为非负。

在有效市场的不同形态下，信息集  $\mathcal{F}_t$  的内容也不同。在弱形式有效市场中， $\mathcal{F}_t$  仅包含历史价格信息；在半强形式有效市场中， $\mathcal{F}_t$  包含所有公开可用的信息；而在强形式有效市场中， $\mathcal{F}_t$  包含所有信息（包括公开与非公开）。根据迭代期望法则，如果  $\mathcal{F}'_t \subseteq \mathcal{F}_t$ ，则有：

$$\mathbb{E}(R_T - R_t^* | \mathcal{F}'_t) = 0.$$

此外，由于

$$\text{Var}[R_{t+j}] = \mathbb{E}[\text{Var}(R_{t+j} | \mathcal{F}_t)] + \text{Var}[\mathbb{E}(R_{t+j} | \mathcal{F}_t)],$$

不难看出，随着信息集的增加，我们对未来收益的预测更为准确，其条件方差减小：

$$\text{Var}(R_{t+j} | \mathcal{F}'_t) \geq \text{Var}(R_{t+j} | \mathcal{F}_t) \quad (\mathcal{F}_t \subseteq \mathcal{F}'_t).$$

因此，异常收益的存在挑战了有效市场假说 (EMH)。该假说认为在特定情况下，投资者或许可以利用额外信息或市场的效率不足获取超出一般风险调整回报的收益。对异常收益的研究有助于评估资产管理策略、市场有效性与投资行为。

当然，EMH 的定义存在很多模糊之处。尤其是如何理解“信息被价格充分反映”。首先，信息反映速度：通常理解为所有信息被立即反映在价格中，但“立即”并不现实，因为所有的行为和反应都受到光速的限制。因此，实证中必须定义一个合理的时间框架来符合理论的实质。其次，信息集的定义：对于不同形式的市场有效性，信息集的定义也存在讨论。弱式有效性通常指证券的历史价格，但具体是哪个历史价格并不明确。通常我们使用收盘价，但也可以包含开盘价、日内最高价、最低价，甚至整个交易记录。此外，不同交易所（例如：上交所和港交所）的价格也可以纳入考量范围。最后，信息集的扩展也有难度。弱式有效性的信息集是历史价格，而半强式有效性则包括公开发布的信息以及非固定时间表发布的其他信息。强式有效性则包括所有信息，无论是公开的还是非公开的，例如只有公司内部人士或监管机构才知道的信息，这些信息几乎不可能获得。所以很难针对强式有效性的信息集进行研究。在现实中，大多数市场被认为接近弱式或半强式有效，完全的强式有效性往往难以实现。

**定义 6.1:** 设定  $\mathcal{H}_t$  为时间  $t \geq 0$  的市场价格历史,  $\mathcal{F}_t$  为包含  $\mathcal{H}_t$  的更广泛的公共信息集, 而  $\mathcal{G}_t$  为包含  $\mathcal{F}_t$  的全部信息集。考虑一个资产在未来时间  $T > t$  的价格  $P_T$ , 如果满足:

$$\Pr(P_T \leq x | \mathcal{F}_t) = \Pr(P_T \leq x | \mathcal{H}_t). \quad (6.1)$$

对于所有  $t \geq 0$  和  $x \in \mathbb{R}$ , 这表明市场已经完全吸收了公共信息集  $\{\mathcal{F}_t\}$ 。

公式 (6.1) 表示在给定价格历史  $\mathcal{H}_t$  的条件下, 资产未来价格  $P_T$  的分布与公共信息集  $\mathcal{F}_t$  无关。这暗示在半强式和强式有效市场中, 除历史价格信息外的其他公开信息不会对资产价格产生额外影响。在强式有效市场中, 价格历史  $\mathcal{H}_t$  已经包含了所有公开和非公开信息, 因此在估算未来价格  $P_T$  的条件分布时, 较弱的信息集  $\mathcal{H}_t$  和较强的信息集  $\mathcal{F}_t$  同样有效, 额外的信息被视为无用。

验证条件 (6.1) 在实际操作中十分困难, 因此我们会对一个更弱的条件进行检验。

$$\mathbb{E}(P_T | \mathcal{F}_t) = \mathbb{E}(P_T | \mathcal{H}_t),$$

进而可得

$$\mathbb{E}(R_T | \mathcal{F}_t) = \mathbb{E}(R_T | \mathcal{H}_t),$$

根据迭代期望法则 (Law of Iterated Expectation), 我们有:

$$\mathbb{E}((R_T - \mathbb{E}(R_T | \mathcal{H}_t)) | \mathcal{F}_t) = 0.$$

我们需要对信息集和期望值等进行约束来测试该假设, 通常我们假设期望收益率、无风险利率和风险溢价都是恒定的。

此外, 值得注意的是, 在针对 EMH 进行假设检验时, 即使我们拒绝了“投资者无法获得超额收益”的原假设, 也并不能据此证伪 EMH, 原因可能在于收益率模型设定有误 (联合假设问题)。因此, 我们实际上无法真正地拒绝弱式 EMH。进一步地, 若在较小信息集中未能拒绝 EMH, 可能仅因信息集的信息量不足; 而所有信息又难以被完全纳入。换言之, 我们无法对半强式、强式 EMH 进行证伪。

## 6.3 随机游走价格模型

值得注意的是, 由于内部交易信息难以获得, 市场有效性的实证研究多数集中在弱式以及半强式有效市场中; 又由于对半强式有效市场进行检验需要分析会计盈余、资产负债表和现金流量表等公开披露的财务数据, 以及分析行业发展报告、政策变化等公开信息, 观察市场价格是否迅速整合这些信息, 因此本节对 EMH 的检验侧重于弱式有效性检验。

弱式市场有效假设认为资产价格服从随机游走模型, 即价格变动是随机的, 并非基于任何可识别的趋势。随机游走模型是评估股市弱式有效性的一个方法。考虑一个离散时间随机过程  $P_t$ ,  $t = 1, 2, \dots$ :

$$P_t = P_{t-1} + \varepsilon_t,$$

其中,  $\varepsilon_t$  是均值为零的创新过程 (innovation process), 该过程也称为随机游走。此外, 还可以考虑有漂移的随机游走:

$$P_t = \mu + P_{t-1} + \varepsilon_t$$

其中漂移项  $\mu$  可以是非零的；通过迭代可得：

$$P_t = P_0 + t\mu + \sum_{s=1}^t \varepsilon_s$$

在金融应用中， $P_t$  可以是价格或是对数收益率。因此漂移  $\mu$  对应市场的正常回报。在高频数据（毫秒、秒、分钟等）中， $\mu$  通常被假设为零。其他情形允许  $\mu$  随时间变化，例如基于某种资产定价模型： $\mu_t = \Psi(P_{t-1}, P_{t-2}, \dots)$ 。此外，一般认为市场收益率可视为风险的函数，例如：

$$\mu_t = h(\text{Var}(r_t | \mathcal{F}_{t-1})).$$

其中  $h(\cdot)$  为单调递增函数，条件方差  $\text{Var}(r_t | \mathcal{F}_{t-1})$  衡量资产风险。如果风险  $\text{Var}(r_t | \mathcal{F}_{t-1})$  是时变的，那么预期回报  $E(r_t | \mathcal{F}_{t-1})$  也应如此。对于日度或周度数据，可认为  $\mu_t$  变化很小，甚至可以忽略，即  $\mu_t = \mu$ ；而对季度或年度等频率较低的数据，则需对  $\mu_t$  进行建模与估计。

为了使随机游走模型更加精确，我们需要对创新项  $\varepsilon_t$  做出明确的假设。与 Campbell et al. (1997) 类似，我们考虑以下几类假设：

- rw1 假设： $\varepsilon_t$  是独立同分布的 (i.i.d.) 且期望为零。这是最强的假设，意味着每一个  $\varepsilon_t$  都是完全独立的，并且具有相同的分布特性。这种假设使得模型简单并且容易处理，因为它排除了序列之间的任何形式的时间依赖性或结构变化。
- rw2 假设： $\varepsilon_t$  在时间上是独立的，且期望为零（即  $E(\varepsilon_t) = 0$ ）。rw2 假设不如 rw1 假设强，因为它仅要求  $\varepsilon_t$  在不同时间点上的独立性，而不要求同分布。因此， $\varepsilon_t$  的分布可以随时间变化，这为模型提供了更大的灵活性，但也增加了模型的复杂性和不确定性。
- rw3 假设：所有的  $k, t$  对于  $\varepsilon_t$  期望为零，并且协方差  $\text{Cov}(\varepsilon_t, \varepsilon_{t-k}) = 0$ 。这意味着创新项之间不仅是独立的，而且相互之间没有线性关联（即没有相关性）。值得注意的是，除了缺乏相关性外， $\varepsilon_t$  与  $\varepsilon_s$  之间的依赖程度完全不受限制。这是三种假设中最弱的。

这些假设之间的强弱关系表明：如果更强的假设（如 rw1）成立，那么较弱的假设（如 rw2 和 rw3）也必然成立，但反之则不成立。这种层次结构允许研究者根据实际数据的特性选择最合适的假设，以平衡模型的严格性和灵活性。在实际应用中，选择哪种假设往往取决于数据的性质和分析目标。例如，高频交易数据可能更适合使用强假设（如 rw1 或 rw2），因为在短时间内，市场的微观结构可能保持相对稳定；而对于频率较低的数据，可能需要采用更弱的假设（如 rw3），以适应潜在的市场结构变化和更复杂的动态。

在金融分析中，理解回报是按交易时间还是按日历时间生成非常重要，因为这会直接影响我们如何分析和解释数据。

首先，交易时间。根据这一假设，回报是在交易时间内产生的，这意味着分析时使用的数据是在市场开放期间收集的。由于这些数据是在固定的交易窗口（例如每个交易日）内收集的，因此可以直接利用日回报数据进行分析。这种方法简单且常用，因为它假设每天都有交易发生，且交易间隔固定。

其次，关于日历时间。如果假设回报是按日历时间生成的，情况则更为复杂。在这种情况下，即使在非交易日也会产生回报，例如周末或假日。这导致我们观察到的数据（即交易时间的回报）可能不完整，因为有些日子没有交易，因此没有回报数据。此时，如果简单地使用观察到的回报进行分析，可能会导致误解。为准确反映日历时间的回报

变化，需要对数据进行调整，考虑非交易日的影响。

在 rw1 下，若进一步假设  $E(r_t) = \mu$  以及  $\text{Var}(r_t) = \sigma^2$ ，则可以使用  $r_t^O = \mu D_t$  与  $\text{Var}(r_t^O) = \sigma^2 D_t$ ；其中  $r_t^O$  表示观测到的收益率， $D_t$  表示从上一次交易日到当前观察日的天数 ( $D_t = 1$  对应周二至周五， $D_t = 3$  对应周一)。

除非特别说明，一般假设数据按交易时间生成。

## 6.4 对 EMH 的假设检验

### 6.4.1 序列相关性检验

如果一个市场是有效的，那么资产价格的变动（例如股票价格）应类似于随机游走，即价格变动不可预测。这意味着价格序列的自相关系数应非常接近于零，表明过去的价格变动与未来的价格变动之间不存在明显的线性关系。在有效市场中，价格变动应独立于其历史数据；若自相关系数显著不为零，则可能表明市场参与者可用历史信息预测未来价格，从而违反 EMH 的信息效率观点。

自协方差函数  $\gamma(j)$  定义为时间序列  $Y_t$  与其滞后  $j$  期值  $Y_{t-j}$  之间的协方差，计算公式为：

$$\gamma(j) = \text{Cov}(Y_t, Y_{t-j}) = E[(Y_t - \mu)(Y_{t-j} - \mu)],$$

其中  $\mu$  为  $Y_t$  的期望。注意此处  $Y_t$  指价格增量： $Y_t = P_t - P_{t-1}$ 。自相关函数  $\rho(j)$  定义为自协方差与  $j = 0$  时自协方差之比，即

$$\rho(j) = \frac{\gamma(j)}{\gamma(0)} \quad (j = 0, 1, 2, \dots).$$

该比值反映  $Y_t$  与  $Y_{t-j}$  的线性关联程度。若序列非平稳，可将自协方差定义为时间跨度  $T$  取极限：

$$\gamma(j) = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=j+1}^T \text{Cov}(Y_t, Y_{t-j}),$$

若该极限存在。自相关函数在衡量线性依赖或可预测性方面非常重要，并且在尺度变换（如按日度或年度）下保持不变；其取值范围为  $[-1, 1]$ 。

$Y_t$  在滞后  $j$  期信息下的最佳线性预测可写为

$$\mathbb{P}(Y_t | Y_{t-j}) = \mu + \beta_j (Y_{t-j} - \mu),$$

其中  $\mu$  为均值， $\beta_j$  为线性投影系数。

实证中，用样本自协方差  $\hat{\gamma}(j)$  与样本自相关  $\hat{\rho}(j)$  检验 EMH。对  $j = 0, 1, 2, \dots, T-1$ ，有

$$\begin{aligned} \hat{\gamma}(j) &= \frac{1}{T} \sum_{t=j+1}^T (Y_t - \bar{Y})(Y_{t-j} - \bar{Y}), \quad \bar{Y} = \frac{1}{T} \sum_{t=1}^T Y_t, \\ \hat{\rho}(j) &= \frac{\sum_{t=j+1}^T (Y_t - \bar{Y})(Y_{t-j} - \bar{Y})}{\sqrt{\sum_{t=j+1}^T (Y_t - \bar{Y})^2 \sum_{t=j+1}^T (Y_{t-j} - \bar{Y})^2}}. \end{aligned}$$

不难看出，在 rw1 假设下（即  $Y_t$  为具有有限方差的独立同分布随机变量），当样本量  $T \rightarrow \infty$  时， $\hat{\rho}(j)$  的分布趋于正态。基于此有如下定理：

**定理 6.1:** 设  $Y_t$  为具有有限方差的独立同分布随机变量。对任意固定的  $p$ , 当  $T \rightarrow \infty$  时,

$$\sqrt{T}(\hat{\rho}(1), \dots, \hat{\rho}(p))' \implies N(0, I_p),$$

即

$$\Pr(\sqrt{T}\hat{\rho}(1) \leq x_1, \dots, \sqrt{T}\hat{\rho}(p) \leq x_p) \rightarrow \Pr(Z_1 \leq x_1, \dots, Z_p \leq x_p) = \prod_{j=1}^p \Phi(x_j),$$

其中  $Z = (Z_1, \dots, Z_p) \sim N(0, I_p)$ ,  $\Phi$  为标准正态分布的累积分布函数,  $I_p$  为  $p \times p$  的单位矩阵。

因此, 对任意固定的  $k$ , 当  $T \rightarrow \infty$  时都有

$$\sqrt{T}\hat{\rho}(k) \implies N(0, 1).$$

对 rw1 的 EMH 检验, 其原假设为  $\rho(k) = 0$ 。据此可将  $\hat{\rho}(k)$  与巴特利特区间 (Bartlett intervals) 比较:

$$\left[ -\frac{z_{\alpha/2}}{\sqrt{T}}, \frac{z_{\alpha/2}}{\sqrt{T}} \right],$$

其中  $z_{\alpha/2}$  为标准正态分布的临界值,  $1 - \alpha$  为双侧检验的置信水平。若  $\hat{\rho}(k)$  落在区间之外, 则拒绝原假设 (rw1), 即历史信息能够预测未来价格。

定理 6.1 进一步意味着 Box–Pierce 统计量

$$Q = T \sum_{j=1}^p \hat{\rho}(j)^2$$

在 rw1 假设下具有近似  $\chi_p^2$  的渐近分布。可据此检验联合原假设  $\rho(1) = 0, \dots, \rho(p) = 0$  与“至少存在某个  $j$  使得  $\rho(j) \neq 0$ ”的备择假设。当  $Q > \chi_p^2(\alpha)$  时拒绝原假设, 其中  $\alpha$  为显著性水平。

在实际应用中, 受样本量限制, 样本自相关常存在偏差, 可能导致检验倾向于错误拒绝或接受原假设。独立同分布且正态时可部分缓解小样本问题 (参见 Anderson (1942))。

针对小样本和非正态情形, 可改进估计与检验。其一, 用  $T - j$  替代  $T$  作为分母计算自协方差:

$$\tilde{\gamma}(j) = \frac{1}{T-j} \sum_{t=j+1}^T (Y_t - \bar{Y}_j)(Y_{t-j} - \bar{Y}_j),$$

其中  $\bar{Y}_j$  为  $t = j + 1$  至  $T$  的样本均值。该修正可使估计的自协方差  $\tilde{\gamma}(j)$  更接近真实自协方差, 尤其在真实自协方差  $\gamma(j) = 0$  时。

其二, 引入偏差校正项, 构造偏差校正的样本自相关:

$$\hat{\rho}^{bc}(j) = \hat{\rho}(j) + \frac{T-j}{(T-1)^2} (1 - \hat{\rho}(j)^2).$$

模拟研究显示, 对中等样本量  $T$ , 该估计量表现更好。

其三, 采用 Ljung–Box 统计量 (对 Box–Pierce 的偏差校正):

$$Q = T(T+2) \sum_{j=1}^p \frac{\hat{\rho}(j)^2}{T-j},$$

其在小样本下通常优于未进行偏差校正的 Box–Pierce 统计量。需注意，该校正在原假设 rw1（即独立同分布）下有效；在更弱的假设下，统计量的极限分布可能不同，需附加条件以确保中心极限定理的适用性。

### 6.4.2 方差比检验

方差比是评估金融市场有效性的一个重要工具，最初由 Poterba & Summers (1988) 以及 Lo & MacKinlay (1988) 引入。方差比检验通过比较不同周期  $p$  的对数回报方差与单期回报方差的  $p$  倍来进行。令  $r_t$  为对数收益率，假设其构成一个平稳序列且其方差有界，则

$$r_t(p) = r_{t+1} + \cdots + r_{t+p}, \quad q = 2, 3, \dots$$

方差比的公式为

$$VR(p) = \frac{\text{Var}(r_t(p))}{p \cdot \text{Var}(r_t)},$$

其中  $r_t(p)$  表示  $p$  期对数收益率为连续  $p$  期收益率之和。

在回报之间没有相关性的情形（随机游走假设）下，序列中各期回报相互独立，方差比  $VR(p)$  应等于 1，即  $p$  期回报的方差等于单期回报方差的  $p$  倍。

在实际应用中，若回报过程平稳但存在自相关，方差比可揭示其自相关结构。例如，当序列呈正（或负）相关时， $p$  期回报方差将大于（或小于）单期回报方差的  $p$  倍。此时

$$VR(p) = 1 + 2 \sum_{j=1}^{p-1} \left(1 - \frac{j}{p}\right) \rho(j),$$

表明方差比是高频数据自相关函数平均值的一个线性函数。

方差比能够反映时间序列中是否存在稳定或可靠的可预测结构，这对预测目的有参考价值。不同于 Box–Pierce 统计量，方差比保留了相关性的符号信息，依赖于前  $p$  个自相关系数及其相对大小：若所有  $\rho(j) > 0$ ，则  $VR(p) > 1$ ；若均小于 0，则  $VR(p) < 1$ 。换言之，通过检验  $VR(p)$  的稳定性可以判断收益率是否具有可预测性。

接着，我们探讨如何在实证中应用方差比对 EMH 进行检验。给定  $np+1$  个（对数）价格观测  $p_0, \dots, p_{np}$ ，称这组观测为高频观测数据，并据此计算高频收益率序列  $\{r_1, \dots, r_T\}$ （其中  $T = np$ ）。随后计算间隔为  $p$  的低频收益率序列  $\{r_0(p), r_p(p), \dots, r_{(n-1)p}(p)\}$ ，显然该序列包含  $n$  个观测值；另需估计重叠的低频收益率序列  $\{r_0(p), r_1(p), \dots, r_{T-p}(p)\}$ ，其包含  $T - p + 1$  个观测值。

我们为每个抽样框计算收益率方差。首先是高频抽样框：

$$\hat{\sigma}_H^2 \equiv \frac{1}{T} \sum_{t=1}^T (r_t - \hat{\mu})^2, \quad \hat{\mu} = \frac{1}{T} \sum_{t=1}^T r_t.$$

其次是低频（例如“星期一至星期五”）的方差：

$$\hat{\sigma}_L^2(p) \equiv \frac{1}{n} \sum_{t=0}^{n-1} (r_t(p) - p \hat{\mu})^2.$$

最后，方差比定义为

$$\widehat{VR}(p) = \frac{\hat{\sigma}_L^2(p)}{p \cdot \hat{\sigma}_H^2}.$$

假设  $rw_1$  成立且  $p$  固定，则

$$\sqrt{T}(\widehat{VR}(p) - 1) \implies N(0, 2(p-1))$$

因此，定义检验统计量

$$S = \frac{\sqrt{T}(\widehat{VR}(p) - 1)}{\sqrt{2(p-1)}}$$

当  $|S| > z_{\alpha/2}$  时拒绝原假设 (EMH)，其中  $z_{\alpha/2}$  为显著性水平  $\alpha$  下的标准正态分布临界值。

$S$  统计量的构造采用非重叠的低频周度回报，此处假定以“星期一”为起始日。当然，我们可以选择不同的起始日，例如我们可以考虑“星期二到星期二”、“星期三到星期三”等等。

此外，对  $j = 1, \dots, p$ ，可用滚动窗口计算收益率序列  $\{r_0(p), r_1(p), \dots, r_{T-p}(p)\}$ 。定义：

$$\hat{\sigma}_{LO}^2(p) \equiv \frac{1}{T-p+1} \sum_{t=0}^{T-p} (r_t(p) - \hat{\mu})^2,$$

其中下标“LO”分别取 Low frequency 和 Overlapping 的首字母，即“低频”和“重叠”之意。这里的“高频/低频”为相对概念；真正的高频金融数据通常为秒级或毫秒级交易数据。由此可得

$$\hat{\sigma}_{LO}^2(p) \equiv \frac{1}{T-p+1} \sum_{t=0}^{T-p} (r_t(p) - p\hat{\mu})^2.$$

由此得到的方差比在渐近意义下（即大样本情况下）呈正态分布，其渐近方差为

$$\omega_1(p) = \frac{4}{6p}(2p-1)(p-1),$$

且  $\omega_1(p) < 2(p-1)$ 。可见使用重叠回报相较非重叠回报效率更高，相当于最多为非重叠序列两倍的样本量。

方差比的另一种估计基于自相关系数：

$$\widetilde{VR}(p) = 1 + 2 \sum_{j=1}^{p-1} \left(1 - \frac{j}{p}\right) \hat{\rho}(j)$$

其中  $\hat{\rho}(j)$  用频率数据估计。

**定理 6.2：** 在  $rw_1$  条件下，

$$\sqrt{T}(\widetilde{VR}(p) - 1) \implies N(0, \omega_1(p))$$

定理 6.2 是定理 6.1 的直接推论，因为  $4 \sum_{j=1}^{p-1} \left(1 - \frac{j}{p}\right)^2 = \omega_1(p)$ 。当存在异方差性（如  $rw_3$ ）时，可对异方差进行校正；此外，可按第 6.4.1 节提出的方法对相关系数进行小样本校正。实际操作中，可采用 wild bootstrap 方法估计方差比的长期方差，如 Choi (1999)。

Bootstrap 方法是一种非参数统计推断方法，主要用于估计样本统计量（如均值、标准误差、置信区间等）的分布。它通过重复从原始数据中重抽样生成多个“Bootstrap”样本，以模拟统计量的抽样分布。该方法广泛用于样本量较小、模型复杂或传统方法难以适用的情形。Wild Bootstrap 是一种特别适用于存在异方差问题的 Bootstrap 方法，最

早由 Wu (1986) 提出, 用于在回归模型中处理异方差性。与传统 Bootstrap 不同, Wild Bootstrap 通过在回归残差上引入随机噪声构造样本, 而非直接对原始观测值抽样。关于 Bootstrap 的系统介绍, 可参见 Davison & Hinkley (1997)。

### 6.4.3 依照 AR 模型对 EMH 进行检验

这里我们介绍一种基于自回归模型对有效市场假说 (EMH) 的检验方法。首先, 考虑以下  $\text{AR}(p)$  模型:

$$Y_t = \mu + \beta_1 Y_{t-1} + \cdots + \beta_p Y_{t-p} + \varepsilon_t,$$

其中  $\varepsilon_t$  满足  $E(\varepsilon_t | Y_{t-1}, \dots, Y_{t-p}) = 0$ 。在 EMH 下,  $Y_t$  的滞后项对  $Y_t$  不应具有预测力; 对应原假设  $H_0: \beta_1 = \cdots = \beta_p = 0$ , 备择假设为“至少存在某个  $j$  使得  $\beta_j \neq 0$ ”。

令  $X$  为  $(T-p-1) \times (p+1)$  维矩阵,  $X$  的第一列全为 1; 其余列由观测值  $Y_p, \dots, Y_{T-1}$  的滞后项构成。Wald 检验统计量<sup>1</sup>为

$$W = T(\hat{\beta} - \beta)' \hat{V}^{-1} (\hat{\beta} - \beta),$$

其中  $\hat{\beta} = (\hat{\beta}_1, \dots, \hat{\beta}_p)'$  为  $\beta$  的最小二乘估计量,  $\hat{V}$  为  $\hat{\beta}$  漐近方差的一致估计。

在独立同分布 (i.i.d.) 情形下,

$$\hat{V} = \hat{\sigma}_\varepsilon^2 (X' X / T)^{-1}.$$

在存在异方差时 (rw2 与 rw3 均允许异方差), 可通过 White 异方差稳健标准误估计:

$$\hat{V}_W = (X' X / T)^{-1} (X' D X / T) (X' X / T)^{-1}, \quad D = \text{diag}(\hat{\varepsilon}_{p+1}^2, \dots, \hat{\varepsilon}_T^2),$$

其中  $\hat{\varepsilon}_t = Y_t - \hat{\mu} - \hat{\beta}_1 Y_{t-1} - \cdots - \hat{\beta}_p Y_{t-p}$ 。在原假设下,  $W \sim \chi_p^2$ 。

在实际金融市场中, 异方差性——即资产收益率的波动率随时间变化——十分常见, 可能受宏观新闻、市场情绪、政策变化等影响。异方差性并不与有效市场假说相冲突, 尤其在弱式有效市场假说下。关键在于市场吸收与反应信息的速度, 而非价格波动是否稳定。因此, 有必要针对 rw2 及 rw3 对 EMH 进行假设检验。下文各节将分别介绍 rw2 与 rw3 的检验方法。

### 6.4.4 对 rw2 与 rw3 进行假设检验

首先, 我们考虑 rw2。此时  $\bar{Y}_t = Y_t - E(Y_t)$  是独立但不同分布的。令

$$\lambda_{ij} = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=j+1}^T E(\bar{Y}_{t-j}^2) E(\bar{Y}_t^2), \quad \gamma_0 = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T E(\bar{Y}_t^2),$$

$$\omega_2(p) = \sum_{j=1}^{p-1} c_{j,p}^2 V_{2;jj}, \quad c_{j,p} = 2 \left(1 - \frac{j}{p}\right),$$

$$V_2(p) = \text{diag}(V_{2;11}, \dots, V_{2;pp}), \quad V_{2;jj} = \frac{\lambda_{ij}}{\gamma_0^2}.$$

其中  $T = np$  为高频收益率样本量,  $p$  为固定正整数且  $n$  较大。

<sup>1</sup>Wald 检验以匈牙利数学家沃德·亚伯拉罕 (Abraham Wald) 命名。

**定理 6.3:** 在 rw2 下, 若  $E(|Y_t|^{2+\delta}) \leq M$ , 且  $\gamma_0$  与  $\lambda_j$  存在 ( $j = 1, \dots, p$ ), 则当  $T \rightarrow \infty$ ,

$$\sqrt{T} (\hat{\rho}(1), \dots, \hat{\rho}(p))' \xrightarrow{d} N(0, V_2(p)), \quad \sqrt{T} (\widehat{VR}(p) - 1) \xrightarrow{d} N(0, \omega_2(p)).$$

接着, 考虑 rw3。按照 Linton (2019), 对 Campbell et al. (1997) 的假设作如下修正。设  $\tilde{Y}_t = Y_t - E(Y_t)$ 。

**假设 6.1:** 对所有  $t$  与任意  $j \geq 0$ , 有  $E(\tilde{Y}_t) = 0$  且  $E(\tilde{Y}_t \tilde{Y}_{t-j}) = 0$ ; 并且对任意  $s \neq t$  与  $j, k = 1, \dots, p$ ,

$$E(\tilde{Y}_t \tilde{Y}_{t-j} \tilde{Y}_s \tilde{Y}_{s-k}) = 0.$$

**假设 6.2:**  $\tilde{Y}_t$  为强混合过程, 混合系数  $\alpha(m)$  的衰减率为  $O(m^{-r})$ , 其中  $r > 1$ 。对所有  $t$  与任意  $j \geq 0$ , 存在  $\delta > 0$  使

$$E(|\tilde{Y}_t \tilde{Y}_{t-j}|^{2(r+\delta)}) \leq C < \infty.$$

**假设 6.3:**

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T E(\tilde{Y}_t^2) = \sigma^2 < \infty.$$

**假设 6.4:** 对所有  $t$  与任意  $j, k \geq 0$  (且  $j \neq k$ ), 有

$$E(\tilde{Y}_t^2 \cdot \tilde{Y}_{t-j} \cdot \tilde{Y}_{t-k}) = 0.$$

**假设 6.5:** 对  $j = 1, \dots, p$ ,

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T E(\tilde{Y}_t^2 \tilde{Y}_{t-j}^2) = \lambda_j < \infty.$$

假设 6.1 中  $E(\tilde{Y}_t \tilde{Y}_{t-j} \tilde{Y}_s \tilde{Y}_{s-k}) = 0$  的条件为 Linton (2019, p. 109) 新引入, 旨在保证渐近方差不再是无穷和。Campbell et al. (1997) 采用 bootstrap 方法估计  $\widehat{VR}(p)$  的渐近方差。在假设 6.1–6.5 下, 定理 6.3 的渐近正态性结果仍然成立。实际应用中, 我们以样本估计量替代定理 6.3 中各参数, 详见 Linton (2019, p. 108) 中式 (3.40)–(3.43)。

## 6.5 案例: 2023 年中国股市的弱式有效性检验——以贵州茅台为例

这里我们采用序列相关性检验与方差比检验, 检验贵州茅台 2023 年 1 月 3 日至 2023 年 12 月 29 日 (2023 年) 期间的价格变动是否服从弱式有效市场假说的随机游走 1 模型 (rw1)。贵州茅台酒股份有限公司成立于 1999 年 11 月 20 日, 是中国著名的大曲酱香型白酒生产商, 总部位于贵州省赤水河畔茅台镇。公司的主导产品贵州茅台酒不仅是中国大曲酱香型白酒的典型代表, 也被视为中国文化的象征。2023 年, 贵州茅台品牌价值达 875.24 亿美元, 成为 “BrandZ 最具价值全球品牌排行榜” 中全球最具价值的酒类品牌。<sup>2</sup> 贵州茅台酒股份有限公司股票代码为 600519, 是上证指数的成分股之一。需要说明的是, 为便于展

<sup>2</sup> 详见: <https://www.moutaichina.com/maotaigf/qygk32/gsjj/index.html>.

示检验流程，本节以贵州茅台为例；若要系统评估我国股市的有效性，仍需对更多代表性股票进行检验。读者可以自行下载中国平安、招商银行、兴业银行、中国石化、上汽集团等多只股票的收盘价数据，并据此进行分析。

```

1 # 清除环境中的所有对象
2 rm (list=ls ())
3
4 # 设置工作目录
5 if (requireNamespace("rstudioapi", quietly = TRUE) &&
6 rstudioapi::isAvailable()) {
7 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
8 }
9
10 library (tseries)
11 library (ggplot2)
12 library (vrtest)
13
14 Moutaidata <- read.csv ("Moutai.csv")
15
16 # 提取数据，并转换为 data frame 格式
17 Moutaidata <- data.frame (Date = Moutaidata$date,
18 MoutaiPrice = Moutaidata$GuizhouMoutai_600519_ClosePrice_Adj)
19
20 Moutaidata$Date <- as.Date (Moutaidata$Date)
21
22 # 计算对数收益率差分以及对数收益率
23
24 Moutaidata$DiffPrice <- c (NA, diff (Moutaidata$MoutaiPrice))
25 Moutaidata$LogReturns <- c (NA, diff (log (Moutaidata$MoutaiPrice)))
26 Moutaidata <- Moutaidata[-1,]
27
28 # Box-Pierce 检验：基于对数收益率进行检验
29 Box_Pierce_result <- Box.test (Moutaidata$DiffPrice, lag = 10, type = "Box-
 Pierce")
30 print (Box_Pierce_result)
31
32 # Box-Ljung 检验：基于对数收益率进行检验
33 Box_Ljung_result <- Box.test (Moutaidata$DiffPrice, lag = 10, type = "Ljung-
 -Box")
34 print (Box_Ljung_result)
35
36 # 方差比检验：基于对数收益率进行检验
37 AutoBoot_result <- AutoBoot.test (Moutaidata$LogReturns, nboot = 500, wild
 = "Normal")
38 print (AutoBoot_result)

```

首先，上述代码使用 ADF 检验、Box-Pierce 检验以及 Ljung-Box 检验来测试贵州茅台股票价格是否服从随机游走。ADF 检验的结果如下：

```

1 > # Box-Pierce 检验：基于对数收益率进行检验
2 > Box_Pierce_result <- Box.test (Moutaidata$DiffPrice, lag = 10, type = "
 Box-Pierce")
3 > print (Box_Pierce_result)

```

```

4
5 Box-Pierce test
6
7 data: Moutaidata$DiffPrice
8 X-squared = 11.957, df = 10, p-value = 0.2879
9
10 >
11 > # Box-Ljung 检验: 基于对数收益率进行检验
12 > Box_Ljung_result <- Box.test (Moutaidata$DiffPrice, lag = 10, type =
13 Ljung-Box)
14 > print (Box_Ljung_result)
15
16 Box-Ljung test
17
18 data: Moutaidata$DiffPrice
X-squared = 12.384, df = 10, p-value = 0.2602

```

不难看出, Box-Pierce 检验和 Ljung-Box 检验在滞后 10 期时均显示价格差分不存在显著的自相关性。

接着我们采用自动方差比率检验 (Auto-VR) 来评估股票收益率序列的随机游走特性。在有效市场中, 价格变动被认为是随机的, 因此无法预测。方差比率检验比较不同时间间隔的收益率方差, 检查它们是否符合随机游走模型的预期; 若实测方差比率显著偏离理论值, 则表明市场可能不符合随机游走假设。

`Auto.VR` 是 `vrtest` 包中的函数, 由 Choi (1999) 提出。`Auto.VR` 通过分析时间序列数据, 自动选择合适的滞后期数 (lag) 并计算方差比率, 适用于评估金融时间序列 (如股票价格或收益率序列) 是否符合随机游走假设。但由于 `Auto.VR` 不提供  $p$  值, 通常采用 `vrtest` 中的 `AutoBoot.test` 进行判断; 该函数采用 wild bootstrap 方法计算出方差比检验的置信区间, 便于统计推断。

检验的结果如下:

```

1 > # 方差比检验: 基于对数收益率进行检验
2 > AutoBoot_result <- AutoBoot.test (Moutaidata$LogReturns, nboot = 500,
3 wild = "Normal")
4 > print (AutoBoot_result)
5 $test.stat
6 [1] -0.49597
7
8 $VRsum
9 [1] 0.9459821
10
11 $pval
12 [1] 0.324
13
14 $CI.stat
15 2.5% 97.5%
16 -1.120601 1.254277
17
18 $CI.VRsum
19 2.5% 97.5%
0.8533894 1.1777045

```

方差比检验结果如下：

- `test.stat = -0.49597`: 方差比检验的统计量，数值接近 0，未显示显著偏离随机游走假设。
- `VRsum = 0.9459821`: 方差比约为 0.946，接近 1，表明在 2 期及更长区间内，收益率方差与随机游走模型的理论方差相近。
- `p-value = 0.324`:  $p = 0.324 > 0.05$ ，无法拒绝弱式有效市场假说（价格遵循随机游走）的原假设。
- `CI.stat`: 统计量的置信区间为  $(-1.12, 1.25)$ ，包含 0，进一步说明未见显著偏离。
- `CI.VRsum`: 方差比的置信区间为  $(0.85, 1.18)$ ，包含 1，支持方差比接近 1 的判断。

总之，上述结果均不显著，因而无法否定价格遵循随机游走的假设。就本样本期而言，贵州茅台股票的走势与弱式有效一致，即市场价格已反映全部历史信息，历史价格与收益信息不具备可用于预测未来价格走势的统计依据。

## 6.6 章节总结

本章围绕收益可预测性与有效市场假说（EMH）展开系统论述，旨在探讨金融市场中资产价格的变动规律及市场有效性。首先，介绍了有效市场假说这一核心理论，它是理解金融市场价格形成机制的基础；接着阐述随机游走价格模型，为后续检验提供理论框架。在对有效市场假说的假设检验部分，详细介绍多种方法：序列相关性检验用于检测资产收益率序列是否存在自相关性；方差比检验通过比较不同时间间隔的收益率方差判断是否符合随机游走；基于 AR 模型对 EMH 进行检验，借助自回归模型分析收益率的可预测性；对随机游走模型的第二种与第三种形式进行假设检验，从不同角度验证市场有效性。最后，为加深对理论与方法的理解，给出实际案例——以 2023 年中国 A 股市场中贵州茅台股票为例进行弱式有效性检验，便于将所学应用于实际金融分析。

## 6.7 习题

1. 假设资产价格的对数回报  $r_t$  是平稳过程，定义多期回报的方差比率  $VR(p)$ ：

$$VR(p) = \frac{\text{Var}(r_t(p))}{p \cdot \text{Var}(r_t)}.$$

- (a) 解释在随机游走假设下，为什么方差比率应等于 1。  
 (b) 说明如何使用方差比率检验来检验弱式有效市场假说。若检验中方差比率显著大于 1 或显著小于 1，应如何解释?  
 (c) 证明在非重叠情形下，方差比统计量在随机游走（rw1）假设下的极限分布，即定理6.2。
2. 虽然在本章正文中未能提及，但游程检验（Runs Test）亦可用于检验 EMH。游程检验是一种非参数方法，通过考察序列中“游程”（连续相同符号的一段）的分布来判断序列是否偏离随机性。请下载贵州茅台近一年的收盘价并完成以下检验：

(a) 计算对数收益率，并将其转换为“+”（上涨，对应 +1）与“-”（下跌，对应 -1），统计游程数（runs）。例如，序列“+ + -- + - + + - - + + +”的游程总数为 10。

(b) 计算正值与负值的数量，分别记为  $n_1$  与  $n_2$ 。计算期望游程数  $\bar{R}$  与游程数标准差  $s_R$ ：

$$\bar{R} = \frac{2n_1 n_2}{n_1 + n_2} + 1, \quad s_R^2 = \frac{2n_1 n_2 (2n_1 n_2 - n_1 - n_2)}{(n_1 + n_2)^2 (n_1 + n_2 - 1)}.$$

(c) 构建检验统计量

$$Z = \frac{R - \bar{R}}{s_R}.$$

(d) 检验假设：零假设  $H_0$  为“序列为随机产生”，备择假设  $H_a$  为“序列非随机产生”。当样本量较大（如  $n_1 > 20$  且  $n_2 > 20$ ）时， $Z$  近似服从标准正态分布。在 5% 显著性水平下，临界值为 1.96。若  $|Z| > 1.96$ ，则拒绝零假设，表明序列可能为非随机产生。

注：可使用 R 软件包 `randtests` 中的 `runs.test` 函数进行游程检验。

3. 第 6.5 节中的案例针对 `rw1` 进行检验。请针对 `rw2` 和 `rw3` 构建检验统计量，并讨论茅台股票价格是否服从 EMH？
4. 令对数收益率服从 AR(1) 过程，

$$r_t = \rho r_{t-1} + \varepsilon_t,$$

其中  $\varepsilon_t$  为独立同分布（i.i.d.）且期望为 0 的随机扰动项，且  $|\rho| < 1$ 。请推导  $VR(p)$ 。

5. 令对数收益率  $r_t$  服从 MA(2) 过程，

$$r_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2},$$

其中  $\varepsilon_t$  为独立同分布（i.i.d.）且期望为 0 的随机扰动项。请推导  $VR(p)$ ，并讨论当  $r_t$  服从 MA 过程时是否可能出现  $VR(p) = 1$ 。

# 7 非参数方法

非参数方法在金融计量经济学中的应用非常广泛，其特点是不依赖于数据分布假设。该方法被用于估计资产价格的波动率，这是因为传统的参数方法（如 GARCH 模型）可能无法准确捕捉金融时间序列的复杂性。非参数方法（如局部线性回归与核平滑）更具灵活性，并对异常值与尾部行为具有更好的适应性。此外，非参数技术在分析市场微观结构方面也非常有用。[Zhou \(1996\)](#) 首次探讨了高频数据中的微观结构噪声，并采用核方法对高频数据的波动率进行估计；非参数方法尤其适用于高频数据分析，因为它们能够适应数据的不规则时间间隔与噪声。

在金融风险管理中，非参数方法可用于估计损失分布的尾部风险，这对计算在险价值 (VaR, value at risk 的简称) 和条件在险价值 (CVaR, conditional value at risk 的简称) 等风险指标至关重要。

本章旨在介绍非参数统计的一些基本概念与方法，为后续章节采用非参数统计方法对模型进行估计和假设检验提供知识储备。具体而言，本章首先讨论独立同分布情形下的概率密度估计，说明其性质同样适用于时间序列数据，并讨论多元概率密度核估计量的渐近性质；接着介绍用于回归函数估计的 Nadaraya–Watson 估计量；随后说明 Nadaraya–Watson 估计量是局部常数平滑的特例，并引入局部多项式估计量；最后简述非参数统计在金融计量经济学中的应用。

## 7.1 非参数概率密度估计量

在金融数据分析中，核平滑法作为非参数密度估计的常用技术，其核心优势在于能够自动适应数据的局部特性，从而揭示真实的分布结构，尤其适用于处理偏态、多峰和异方差等非标准分布特征的数据。对概率密度函数的准确估计可帮助分析者理解资产价格的分布特征，进而更有效地评估风险与机会。核密度估计通过在各观测点处叠加核函数的贡献，形成对整体分布的平滑估计。

核估计中的核函数  $K(\cdot)$  负责对指定区间内的数据点赋予权重，常见的核函数包括高斯核、Epanechnikov 核等。这些核函数通常具有非负、积分为一的特性，确保权重分配的合理性与估计的一致性。

在核估计中，带宽  $h$  是一个极为关键的参数，它对核估计的平滑程度起着直接的决定性作用。带宽的取值，实际上决定了核函数在数据空间中所覆盖的数据范围大小。当带宽较小时，核函数仅会考虑较少的局部数据点，这使得估计对局部数据的变化极为敏感，能够敏锐捕捉到数据的细微波动。然而，这种高敏感性也带来了弊端，那就是可能会增大估计结果的波动率，导致估计值不够稳定。相反，若带宽取值过大，核函数会覆盖过多的数据点，虽然会使估计结果更加平滑，但却可能过度平滑，从而丢失数据中蕴含的关键细节信息，无法精准地反映数据的真实特征。

带宽的选择通常基于均方误差 (MSE) 准则进行优化。理想带宽在偏差与方差之间取

得平衡，使均方误差最小。在实际应用中，带宽往往依据经验法则（如 Silverman 法则）、交叉验证或插值法等自动确定。

**定义 7.1 (二阶核函数  $K(\cdot)$ )：** 二阶或正核函数  $K(\cdot)$  是一个预先指定的对称概率密度函数，满足以下条件：

- $\int_{-\infty}^{\infty} K(u) du = 1;$
- $\int_{-\infty}^{\infty} K(u)u du = 0;$
- $\int_{-\infty}^{\infty} u^2 K(u) du = C_K < \infty;$
- $\int_{-\infty}^{\infty} K^2(u) du = D_K < \infty.$

**定义 7.2 ( $q$  阶核)：**  $K(\cdot)$  满足以下条件：

- $\int_{-\infty}^{\infty} K(u) du = 1;$
- 对于  $1 \leq j \leq q - 1$ ,  $\int_{-\infty}^{\infty} u^j K(u) du = 0;$
- $\int_{-\infty}^{\infty} u^q K(u) du < \infty;$
- $\int_{-\infty}^{\infty} K^2(u) du < \infty.$

$q \geq 2$  阶核的主要优点是能显著减少估计偏差。当核函数的阶数  $q$  增加时，对应的核函数能更有效地捕捉目标函数的高阶变化，从而减少估计中的系统偏差。在处理具有复杂局部结构的数据时，高阶核能提供更好的拟合效果。例如，在金融数据分析中，市场数据可能在短时间内快速变动，使用高阶核能更准确地逼近这种动态变化。尽管  $q$  阶核在理论上具有优越的性能，但在实际应用中也需要进行权衡：使用高阶核可能需要更多的数据来支撑复杂的模型估计，否则可能会增加方差，导致过拟合。

以下，我们给出了一些二阶核的例子：

- 均匀核 (Uniform Kernel)

$$K(u) = \frac{1}{2} \mathbf{1}(|u| \leq 1);$$

- 高斯核 (Gaussian Kernel)

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}u^2\right); \quad -\infty < u < \infty;$$

- Epanechnikov 核

$$K(u) = \frac{3}{4} (1 - u^2) \mathbf{1}(|u| \leq 1);$$

- 四次方核 (Quartic Kernel)

$$K(u) = \frac{15}{16} (1 - u^2)^2 \mathbf{1}(|u| \leq 1).$$

```
1 # 均匀核
2 func_uniform <- function (x) {
3 y <- numeric (length (x))
```

```

4 idx <- abs (x) <= 1
5 y[idx] <- 1/2
6 return (y)
7 }
8 # 高斯核
9 func_gaussian <- function (x) {
10 y <- (1 / sqrt (2 * pi)) * exp (-0.5 * x^2)
11 return (y)
12 }
13 # Epanechnikov核
14 func_e <- function (x) {
15 y <- numeric (length (x))
16 idx <- abs (x) <= 1
17 y[idx] <- (3 / 4) * (1 - x[idx]^2)
18 return (y)
19 }
20 # 四次方核
21 func_quartic <- function (x) {
22 y <- numeric (length (x))
23 idx <- abs (x) <= 1
24 y[idx] <- (15 / 16) * (1 - x[idx]^2) ^2
25 return (y)
26 }
27
28

```

我们可以用以下代码生成这些核函数的图像，见图7.1。

```

1 library (ggplot2)
2 library (dplyr)
3
4 # 生成x值的序列
5 x_vals <- seq (-3, 3, by=0.01)
6
7 # 为每个核函数创建数据框
8 df_uniform <- data.frame (x = x_vals, y = func_uniform (x_vals) , kernel =
9 "Uniform")
9 df_gaussian <- data.frame (x = x_vals, y = func_gaussian (x_vals) , kernel =
10 "Gaussian")
10 df_e <- data.frame (x = x_vals, y = func_e (x_vals) , kernel =
11 "Epanechnikov")
11 df_quartic <- data.frame (x = x_vals, y = func_quartic (x_vals) , kernel =
12 "Quartic")
12
13 # 将所有数据框合并成一个数据框
14 df_kernels <- bind_rows (df_uniform, df_gaussian, df_e, df_quartic)
15
16 # 使用ggplot2绘图
17 ggplot (df_kernels, aes (x = x, y = y, color = kernel)) +
18 geom_line (size = 1.2) +
19 labs (title = "Kernel Functions", x = "x", y = "Density") +
20 scale_color_manual (values = c ("blue", "red", "green", "purple")) +
21 theme_minimal () +

```

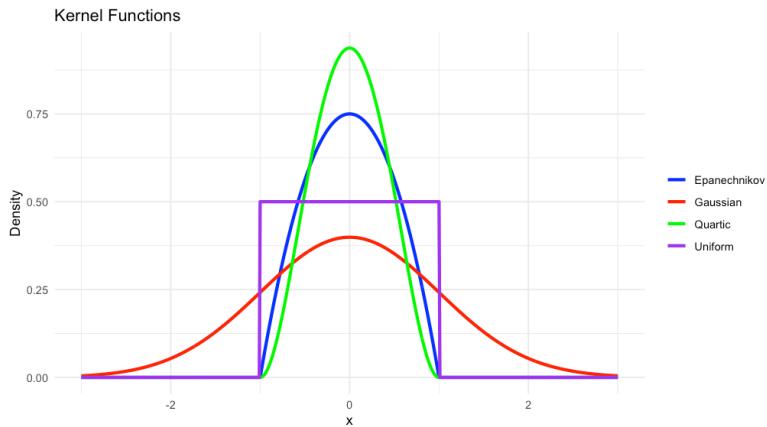


图 7.1: 均匀核、高斯核、Epanechnikov 核以及四次方核函数

```
22 theme (legend.title = element_blank ())
23
24
```

在这些核函数中，高斯核的支撑集不受限制，其余核的支撑集均在  $[-1, 1]$  上。均匀核在其支撑集内赋予等权重；相比之下，其它核多采用类似“钟形曲线”的逐渐递减权重。需要注意的是，Epanechnikov 核由 Epanechnikov (1969) 提出。对于给定的带宽，使用 Epanechnikov 核可以最小化均方误差，从而使估计更为精确。同时，估计值对核函数的选择并不十分敏感，更依赖于带宽的选取。

**例 7.1 (直方图):** 如果  $K(u) = \frac{1}{2}\mathbf{1}(|u| \leq 1)$ ，那么：

$$\hat{g}(x) = \frac{1}{2hT} \sum_{t=1}^T \mathbf{1}(|x - X_t| \leq h).$$

直观来看，当选用均匀核时，核密度估计  $\hat{g}(x)$  表示的是以  $x$  为中心、长度为  $2h$  的区间  $[x - h, x + h]$  上观测值的样本相对频率。当区间长度  $2h$  足够小时， $2hT$  近似为小区间  $[x - h, x + h]$  内的样本量。

另一方面， $T^{-1} \sum_{t=1}^T \mathbf{1}(|x - X_t| \leq h)$  是落入小区间  $[x - h, x + h]$  内观测值的相对频率。若  $h$  足够小，并且  $g(x)$  在点  $x$  附近连续，根据大数定律，它近似等于该区间的概率：

$$E[\mathbf{1}(|x - X_t| \leq h)] = P(x - h \leq X_t \leq x + h) = \int_{x-h}^{x+h} g(y) dy \approx 2hg(x).$$

因此，直方图是  $g(x)$  的一个合理估计量。实际上，如果  $h$  趋于零，但其趋于零的速率慢于样本量  $T$  趋于无穷大的速率，那么  $\hat{g}(x)$  是  $g(x)$  的一致估计量。

以下代码主要用于分析和可视化股票指数的日收益率分布特性。具体实现方式是，通过绘制直方图和核密度估计图，直观展示相关数据特征。数据方面，采用 `datasets` 包中 `EuStockMarkets` 数据集里德国 DAX 指数的历史收盘价，以此为基础计算日收益率，并进行后续的分析与可视化处理。

```
1 # 加载数据集包和数据
2 library (datasets)
```

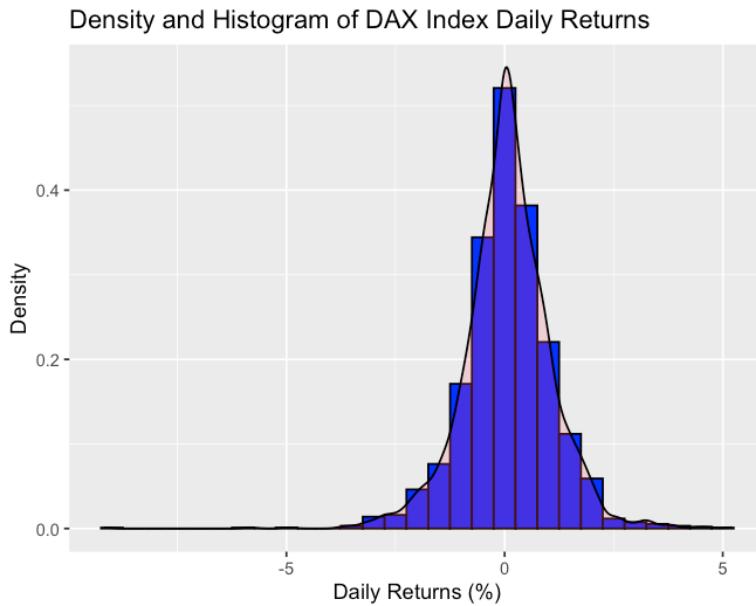


图 7.2: DAX 指数日收益率的核密度与直方图估计

```

3 data ("EuStockMarkets")
4
5 # EuStockMarkets 数据集是一个矩阵，每一列代表一个指数
6 dax_prices <- EuStockMarkets[, "DAX"]
7
8 # 计算日收益率为连续交易日之间的百分比变化
9 dax_returns <- diff (dax_prices) / lag (dax_prices, -1) * 100
10
11 # 移除由于 diff 函数生成的 NA 值（第一个值）
12 dax_returns <- na.omit (dax_returns)
13
14 # 将 DAX 日收益率数据转换为 dataframne，以便使用 ggplot2
15 dax_returns_df <- data.frame (DAX_Returns = dax_returns)
16
17 # 加载 ggplot2 包进行绘图
18 library (ggplot2)
19
20 # 绘制 DAX 日收益率的密度直方图
21 ggplot (dax_returns_df, aes (x = DAX_Returns)) +
22 geom_histogram (aes (y = ..density..) , binwidth = 0.5, color = "black",
23 fill = "blue") +
24 geom_density (alpha = .2, fill = "#FF6666") +
25 labs (title = "Density and Histogram of DAX Index Daily Returns",
26 x = "Daily Returns (%) " ,
27 y = "Density")

```

大家可以观察一下生成的概率密度图—图7.2。首先，金融数据的收益率往往不是对称的；例如，股票收益率在市场崩溃时可能出现尖锐的下跌，导致负向偏态。其次，与正态分布相比，金融收益率的分布往往具有更重的尾部。这意味着极端值的发生概率远高于正态分布所预测的概率，这是金融市场中常见的大幅波动或“黑天鹅”事件。

### 7.1.1 单维核密度估计量

为了方便对核密度估计量的偏差和方差进行分析。我们对数据生成过程以及概率密度函数进行如下假定。

**假设 7.1 (概率密度函数的平滑性):** (i)  $\{X_t\}$  是一个严平稳过程，具有边际概率密度函数  $g(x)$ ; (ii)  $g(x)$  在有界支撑  $[a, b]$  上连续两次可微，且  $g''(\cdot)$  在  $[a, b]$  上是 Lipschitz 连续的，即对所有  $x_1, x_2 \in [a, b]$ ，有  $|g''(x_1) - g''(x_2)| \leq C|x_1 - x_2|$ ，这里  $a, b$  和  $C$  是有限常数。

函数  $g(\cdot)$  在边界点  $a$  和  $b$  处的导数定义为：

$$g'(a) = \lim_{x \rightarrow 0^+} \frac{g(a+x) - g(a)}{x},$$

$$g'(b) = \lim_{x \rightarrow 0^-} \frac{g(b+x) - g(b)}{x}.$$

**定义 7.3 (Lipschitz 连续性):** 存在一个非负常数  $L$ ，使得对于所有  $x$  和  $y$  在函数定义域内，都有：

$$|f(x) - f(y)| \leq C|x - y|,$$

这里的  $C$  称为 Lipschitz 常数。Lipschitz 连续是一种比一般连续性（即使得函数图像无断点）更强的条件，它确保了函数在其定义域内的变化不会突然急剧增大。Lipschitz 连续性实际上意味着函数的最大斜率受到限制，因此不会有任何“垂直”的部分。

为方便起见，我们对核函数  $K(\cdot)$  引入有界性条件：

**假设 7.2 (二阶正值核):**  $K(u)$  是一个二阶正值核函数，且其支撑集为  $[-1, 1]$ 。

该有界支撑的假设不是必需的；没有有界支撑假设也能进行渐近分析，但该假设可简化渐近分析。

如果针对给定支撑集中的点  $x$ ， $\hat{g}(x)$  是  $g(x)$  的一致估计量，我们可以对  $\hat{g}(x) - g(x)$  进行如下分解： $\hat{g}(x) - g(x) = [\mathbb{E}\hat{g}(x) - g(x)] + [\hat{g}(x) - \mathbb{E}\hat{g}(x)]$ 。因此，可得核密度估计量  $\hat{g}(x)$  的均方误差（mean squared error，简称 MSE）为

$$\text{MSE}(\hat{g}(x)) = [\mathbb{E}\hat{g}(x) - g(x)]^2 + \mathbb{E}[\hat{g}(x) - \mathbb{E}\hat{g}(x)]^2 = \text{Bias}^2[\hat{g}(x)] + \text{Var}[\hat{g}(x)].$$

其中第一项是估计量  $\hat{g}(x)$  的偏差平方，它是非随机的；第二项是在点  $x$  处  $\hat{g}(x)$  的方差。在合适的正则条件下，随着样本大小  $T$  趋于无穷大，若  $\hat{g}(x)$  是  $g(x)$  的一致估计量，则其偏差和方差都应趋于零。

### 7.1.1.1 核估计量的偏差

我们首先考虑偏差。对于支撑集  $[a, b]$  的内部区域  $[a+h, b-h]$  中的任意点  $x$ ，我们有：

$$\begin{aligned}
 E[\hat{g}(x)] - g(x) &= \frac{1}{T} \sum_{t=1}^T EK_h(x - X_t) - g(x) \\
 &= E[K_h(x - X_t)] - g(x) \quad (\text{由同分布性质}) \\
 &= \int_a^b \frac{1}{h} K\left(\frac{x-y}{h}\right) g(y) dy - g(x) \\
 &= \int_{(a-x)/h}^{(b-x)/h} K(u) g(x + hu) du - g(x) \quad \left(\text{变量替换 } \frac{y-x}{h} = u\right) \\
 &= \int_{-1}^1 K(u) g(x + hu) du - g(x) \\
 &= g(x) \int_{-1}^1 K(u) du - g(x) + hg'(x) \int_{-1}^1 u K(u) du + \frac{1}{2} h^2 \int_{-1}^1 u^2 K(u) g''(x + \lambda hu) du \\
 &= \frac{1}{2} h^2 C_K g''(x) + \frac{1}{2} h^2 \int_{-1}^1 [g''(x + \lambda hu) - g''(x)] u^2 K(u) du \\
 &= \frac{1}{2} h^2 C_K g''(x) + o(h^2),
 \end{aligned}$$

其中第二项，当  $h \rightarrow 0$  时

$$\int_{-1}^1 [g''(x + \lambda hu) - g''(x)] u^2 K(u) du \rightarrow 0.$$

这是由勒贝格控制收敛定理 (Lebesgue's dominated convergence theorem) 得出的。因此，对于内部区域  $[a+h, b-h]$  中的点  $x, \hat{g}(x)$  的偏差与  $h^2$  成正比。因此，为了使偏差随  $T \rightarrow \infty$  而趋向于零，我们必须令  $T \rightarrow \infty, h \rightarrow 0$ 。

勒贝格控制收敛定理是实分析中的一个重要定理，用于处理函数序列的极限与积分的交换。其基本思想是：若可测函数列  $f_n(x)$  几乎处处收敛到  $f(x)$ ，且存在一个可积函数  $g(x)$  使得  $|f_n(x)| \leq g(x)$  对所有  $n$  与几乎处处的  $x$  成立，则  $f$  可积，且

$$\lim_{n \rightarrow \infty} \int f_n(x) dx = \int \lim_{n \rightarrow \infty} f_n(x) dx = \int f(x) dx.$$

在上例中，控制函数是  $g(\cdot)$  的二阶导数  $g''(\cdot)$ 、核  $K(u)$  与  $u^2$  的乘积  $u^2 K(u)$ ，而被控制的函数是  $[g''(x + \lambda hu) - g''(x)] u^2 K(u)$ 。

之所以能使用勒贝格控制收敛定理，主要是因为：

1.  $g''(\cdot)$  的有界性与连续性：这意味着  $g''(\cdot)$  在整个考虑的区间上既不会趋向于无限大也不会有突变，因此  $g''(x + \lambda hu)$  和  $g''(x)$  的差异可以被控制，并且随着  $h$  趋向于 0，这种差异趋向于 0。
2. 积分  $\int_{-1}^1 u^2 K(u) du < \infty$ ：这说明  $u^2 K(u)$  是可积的，因此可以作为控制函数，确保每个项  $|(g''(x + \lambda hu) - g''(x)) u^2 K(u)|$  都不会超过这个可积的函数。

这些条件使得我们将极限操作  $h \rightarrow 0$  与积分操作交换，从而得出偏差项中包含的二阶导数项随  $h$  趋向 0 时的行为。这是分析核密度估计中偏差行为的一个重要步骤。

上述关于偏差的结果是在  $\{X_t\}$  独立同分布的假设下获得的。但是在  $\{X_t\}$  是序列相关时也成立，感兴趣的读者可以尝试自己推导一下。

### 7.1.1.2 核估计的边界问题

当  $x$  位于支撑区间  $[a, b]$  的边界区域  $[a, a+h]$  或  $[b-h, b]$  时，称  $x$  位于边界区域。由于这两个区域的长度均为  $h$ ，随着样本量  $T$  的增加，带宽  $h$  趋于零。当估计点  $x$  接近支撑区间的边界，例如位于区间的起始部分  $[a, a+h)$  或结束部分  $(b-h, b]$  时，会出现所谓的边界效应（或边界问题）。其根本原因是，在边界附近，核函数  $K$  无法对跨越边界的的数据点进行完全对称的加权。

通常，核密度估计是通过平滑核函数  $K$  和数据点  $X_t$  来计算的，核函数通常是对称的。在数据的中心区域，每个点  $x$  都被其附近的数据点对称地加权。然而，在边界附近，尤其是  $x$  靠近  $a$  或  $b$  时，核函数对  $x$  左侧或右侧的数据加权不再对称。这是因为一旦  $x$  超出边界，其对应的核函数部分将超出数据的支撑集  $[a, b]$ ，导致在该方向上的数据贡献丧失或减少。

令  $x = a + \lambda h \in [a, a+h]$ ，其中  $\lambda \in [0, 1)$ 。若  $g$  在区间  $[a, b]$  上有界且远离零，即存在常数  $\epsilon > 0$  使得对所有  $x \in [a, b]$ ，均有  $g(x) \geq \epsilon$ ，则有

$$\begin{aligned} E[\hat{g}(x)] - g(x) &= E[K_h(x - X_t)] - g(x) \\ &= \frac{1}{h} \int_a^b K\left(\frac{x-y}{h}\right) g(y) dy - g(x) \\ &= \int_{(a-x)/h}^{(b-x)/h} K(u) g(x+hu) du - g(x) \\ &= \int_{-\lambda}^1 K(u) g(x+hu) du - g(x) \\ &= g(x) \int_{-\lambda}^1 K(u) du - g(x) + h \int_{-\lambda}^1 u K(u) g'(x+\tau hu) du \\ &= g(x) \left[ \int_{-\lambda}^1 K(u) du - 1 \right] + O(h) \\ &= O(1). \end{aligned}$$

因为对于任何  $\lambda < 1$ ，有  $\int_{-\lambda}^1 K(u) du = 1 - \int_{-1}^{\lambda} K(u) du \in (0, 1)$ ，所以  $E[\hat{g}(x)] - g(x) = O(1)$ 。这是因为在边界区域  $[a, a+h)$  或  $(b-h, b]$  中没有对观测值进行对称覆盖。这种现象被称为核估计的边界问题。

边界问题存在多种解决方法：其中最简单的方法是不使用边界区域中的估计值  $\hat{g}(x)$ ，仅对内部区域  $[a+h, b-h]$  中的密度进行估计和使用。然而，这种方法存在缺点：它会导致重要信息的丢失，因为边界区域的  $\hat{g}(x)$  包含了关于  $\{X_t\}$  尾部分布的信息。在金融经济学中，极端市场下行风险的分析极为重要，所以这种方法并不推荐。此外，还可以采用局部多项式自动适应边界区域，这将在后续部分详细讨论。在这里，我们介绍由 Hong & Li (2005) 提出的边界核（boundary kernel）方法，以及由 Schuster (1985) 提出的数据反射（data reflection）方法。

当  $x$  处于边界区域时，需要对核函数进行修正，使其成为一个依赖于位置的函数。例如，Hong & Li (2005) 使用了如下核的密度估计量：

$$\hat{g}(x) = \frac{1}{T} \sum_{t=1}^T K_h(x, X_t),$$

其中

$$K_h(x, y) \equiv \begin{cases} h^{-1} K\left(\frac{x-y}{h}\right) / \int_{-(x/h)}^1 K(u) du & \text{如果 } x \in [0, h], \\ h^{-1} K\left(\frac{x-y}{h}\right) & \text{如果 } x \in [h, 1-h], \\ h^{-1} K\left(\frac{x-y}{h}\right) / \int_{-1}^{(1-x)/h} K(u) du & \text{如果 } x \in (1-h, 1], \end{cases}$$

且  $K(\cdot)$  是一个标准的二阶核。该思路的核心思想是在边界区域修改核函数，使其积分为 1。在内部区域的所有  $x \in [a+h, b-h]$  处的偏差为  $O(h^2)$ ，在边界区域  $x \in [a, a+h]$  和  $(b-h, b]$  处的偏差至多为  $O(h)$ 。这种方法的优点是非常简单，并且总能给出正的密度估计；缺点是边界区域的偏差收敛速度  $O(h)$  比内部区域的  $O(h^2)$  更慢。

数据反射法是基于增广数据 (augmented data) 构建核密度估计，该数据结合“反射”数据  $\{-X_t\}_{t=1}^T$  和原始数据  $\{X_t\}_{t=1}^T$ ，其支撑集在  $[0, 1]$  上。假设  $x$  是  $[0, h)$  中的边界点，且  $x \geq 0$ ，那么反射法给出的估计量为

$$\hat{g}(x) = \frac{1}{2T} \sum_{t=1}^T \{K_h(x - X_t) + K_h(x + X_t)\}.$$

当核  $K(\cdot)$  的支撑集为  $[-1, 1]$  时，若  $x$  离边界较远，则第二项为零。因此，这种方法仅在边界区域修正密度估计。该方法首先由 Schuster (1985) 提出，后续由 Chen & Hong (2012) 进一步拓展，并应用于时变函数的估计。

### 7.1.1.3 核估计的方差

我们已经讨论了  $\hat{g}(x)$  的偏差，接下来考虑其方差。为简化分析，假设样本独立同分布 (i.i.d.)。

**假设 7.3 (独立同分布观测):** 样本  $\{X_t\}_{t=1}^T$  相互独立且同分布 (i.i.d.)。

假设 7.3 可在很大程度上简化  $\hat{g}(x)$  漐近方差的计算过程。

给定闭区间  $[a, b]$  中的任意点  $x$ ，

$$Z_t \equiv Z_t(x) = K_h(x - X_t) - E[K_h(x - X_t)],$$

可得  $\{Z_t\}_{t=1}^T$  均值为零且独立同分布。由此可得  $\hat{g}(x)$  的方差：

$$\begin{aligned} E[\hat{g}(x) - E\hat{g}(x)]^2 &= E\left(T^{-1} \sum_{t=1}^T Z_t\right)^2 = \frac{1}{T^2} \sum_{t=1}^T \text{Var}(Z_t) = \frac{1}{T} \text{Var}(Z_t) \\ &= \frac{1}{T} \left[ E[K_h^2(x - X_t)] - [E K_h(x - X_t)]^2 \right] \\ &= \frac{1}{Th^2} \int_a^b K^2\left(\frac{x-y}{h}\right) g(y) dy - \frac{1}{T} \left[ \frac{1}{h} \int_a^b K\left(\frac{x-y}{h}\right) g(y) dy \right]^2 \\ &= \frac{1}{Th} g(x) \int_{-1}^1 K^2(u) du [1 + o(1)] + O(T^{-1}) \\ &= \frac{1}{Th} g(x) D_k + o(T^{-1} h^{-1}), \end{aligned}$$

上式中最后第二个等式是通过变量变换  $\frac{x-y}{h} = u$  得到的。 $\hat{g}(x)$  的方差与  $(Th)^{-1}$  成比例，其中  $Th$  是落在间隔  $[x-h, x+h]$  内的观测值的近似样本大小。

我们可以放宽  $\{X_t\}$  的独立性假设，例如假设  $\{X_t\}$  为  $\alpha$ -混合过程（详见第 4.1.2 节）。然而，这并不会改变  $\hat{g}(x)$  的渐近方差结果，感兴趣的读者可以尝试推导这一结论。Hart (1996) 为这一结果提供了直观的解释。

假设核函数  $K(\cdot)$  的支撑集为  $[-1, 1]$ ，则在点  $x$  处的核密度估计仅使用局部区间  $[x - h, x + h]$  内的数据点。落在该局部区间内的观测值在时间上通常相距较远。因此，尽管原始序列  $\{X_t\}_{t=1}^T$  可能高度相关，但围绕  $x$  的局部区间内子序列的依赖性往往显著减弱。结果， $[x - h, x + h]$  内的数据近似于来自一个独立样本。因此，在满足某些混合条件下，核密度估计量的渐近方差与独立观测情形下的方差相同。这表明序列相关性不会影响核密度估计的渐近方差。

#### 7.1.1.4 均方误差和最优带宽

均方误差 (mean squared error, 简称 MSE) 的定义如下：

$$\begin{aligned}\text{MSE}[\hat{g}(x)] &= \mathbb{E}[\hat{g}(x) - g(x)]^2 \\ &= \text{Var}[\hat{g}(x)] + \text{Bias}^2[\hat{g}(x), g(x)] \\ &= \frac{1}{Th}g(x)D_K + \frac{1}{4}h^4[g''(x)]^2C_K^2 + o(T^{-1}h^{-1} + h^4) \\ &= O(T^{-1}h^{-1} + h^4)\end{aligned}$$

根据切比雪夫不等式，对于任意给定点  $x$  在内部区域  $[a + h, b - h]$ ，我们有：

$$\hat{g}(x) - g(x) = O_P(T^{-1/2}h^{-1/2} + h^2)$$

因此，为了使  $\hat{g}(x) \rightarrow^p g(x)$ ，需要当  $T \rightarrow \infty$  时有  $Th \rightarrow \infty, h \rightarrow 0$ 。在给定的假设下，估计量  $\hat{g}(x)$  对未知密度函数  $g(x)$  总是一致的，但收敛速度比参数收敛速度  $T^{-1/2}$  要慢，这意味着需要大样本才能获得  $g(x)$  的合理估计。

此外， $\hat{g}(x)$  的偏差取决于未知函数  $g(\cdot)$  的平滑性。若二阶导数  $g''(x)$  在点  $x$  存在较大波动，则难以在该点获得对  $g(\cdot)$  的良好估计。

当  $g(x) > 0$  时，我们可得如下相对均方误差 (mean square error, 简称 MSE) 标准：

$$\begin{aligned}\text{MSE}[\hat{g}(x)/g(x)] &= \frac{\text{MSE}[\hat{g}(x)]}{g^2(x)} = \mathbb{E} \left[ \frac{\hat{g}(x) - g(x)}{g(x)} \right]^2 \\ &= \frac{1}{Thg(x)}D_K + \frac{1}{4}h^4 \left[ \frac{g''(x)}{g(x)} \right]^2 C_K^2 + o(T^{-1}h^{-1} + h^4) \\ &= O(T^{-1}h^{-1} + h^4)\end{aligned}$$

相对 MSE 的表达式表明，在观测值相对稀少的稀疏区域，很难获得  $g(x)$  的合理估计。此外，当  $g(x)$  在  $x$  点附近的变化非常剧烈（例如，二阶导数  $g''(x)$  很大）时，小区间内的数据可能不足以捕捉这种快速变化。

从  $\hat{g}(x)$  的均方误差公式可以看出，较小的带宽  $h$  将减少偏差但增加方差，而较大的带宽  $h$  会减少方差但增加偏差。带宽是一个平滑参数。当带宽  $h$  很小，以至于平方偏差小于方差时，我们称存在欠平滑；当带宽很大，以至于其平方偏差大于方差时，我们称存在过平滑。若带宽能够平衡  $\hat{g}(x)$  的平方偏差和方差，则达到了最佳平滑。因此，我们考虑带宽  $h$  的最优选择。

最优带宽可以通过最小化均方误差  $\text{MSE}[\hat{g}(x)]$  这一全局误差度量来获得：

$$h_0 = \left[ \frac{D_K}{C_K^2} \frac{1/g(x)}{[g''(x)/g(x)]^2} \right]^{\frac{1}{8}} T^{-1/5}$$

概率密度函数  $g(x)$  的平滑程度越低，或者在点  $x$  附近的观测数据越稀疏，则对于任意给定的样本量  $T$ ，最优带宽  $h_0$  越小。最优带宽  $h_0$  使得  $\hat{g}(x)$  达到最优收敛速率：

$$\hat{g}(x) - g(x) = O_P(T^{-2/5})$$

收敛速度  $T^{-2/5}$  比参数估计的收敛速率  $T^{-1/2}$  慢。

### 7.1.2 多维核密度估计量

首先，我们考虑估计量  $\hat{f}(x)$  的偏差。假设  $x$  是一个内部点，满足对于所有的维度  $i = 1, \dots, d$  都有  $x_i \in [a_i + h, b_i - h]$ 。这意味着  $x$  位于一个  $d$  维的矩形区域中，该区域在每个维度上的边界为  $[a_i + h, b_i - h]$ 。

多维核密度估计量：

$$\hat{f}(x) = \frac{1}{T} \sum_{t=1}^T \prod_{i=1}^d K_h(x_i - X_{it}) = \frac{1}{T} \sum_{t=1}^T \mathcal{K}_h(x \mid X_t)$$

$\hat{f}(x)$  的偏差为：

$$\begin{aligned} \mathbb{E}[\hat{f}(x)] - f(x) &= \mathbb{E}K_h(x - X_t) - f(x) = \mathbb{E} \prod_{i=1}^d K_h(x_i - X_{it}) - f(x) \\ &= \int \cdots \prod_{i=1}^d \frac{1}{h} K\left(\frac{x_i - y_i}{h}\right) \left| f(y) dy - f(x) \right| \\ &= \prod_{i=1}^d \int_{(a_i - x_i)/h}^{(b_i - x_i)/h} K(u_i) f(x + hu) du - f(x) \\ &= \int_{-1}^1 \cdots \int_{-1}^1 \prod_{i=1}^d K(u_i) f(x + hu) du - f(x) + h \sum_{i=1}^d f_i(x) \prod_{i=1}^d \int_{-1}^1 K(u_i) du_i - f(x) \\ &\quad + \frac{1}{2} h^2 \sum_{i=1}^d \sum_{j=1}^d \int_{-1}^1 \int_{-1}^1 u_i u_j K(u_i) du_i \\ &= \frac{1}{2} h^2 C_K \sum_{i=1}^d f_{ii}(x) + o(h^2) = O(h^2), \end{aligned}$$

其中  $f_i(x) = \frac{\partial}{\partial x_i} f(x)$ ,  $f_{ij}(x) = \frac{\partial^2}{\partial x_i \partial x_j} f(x)$ ,  $\sum_{i=1}^d f_{ii}(x)$  是联合概率密度函数  $f(x)$  的拉普拉斯算子 (Laplacian) 作用结果。它可以帮助我们理解概率密度函数在空间中的弯曲程度，同时也有助于分析多变量之间的关系和相互作用。

接下来，设

$$Z_t \equiv Z_t(x) = \mathcal{K}_h(x - X_t) - \mathbb{E} \mathcal{K}_h(x - X_t) = \prod_{i=1}^d K_h(x_i - X_{it}) - \mathbb{E} \prod_{i=1}^d K_h(x_i - X_{it}).$$

若  $\{X_t\}$  是独立同分布的，则  $\{Z_t\}$  也是独立同分布的且均值为零，由此可得  $\hat{f}(x)$  的方差为

$$\begin{aligned} \mathrm{E}[\hat{f}(x) - \mathrm{E}\hat{f}(x)]^2 &= \mathrm{E}\left[T^{-1} \sum_{t=1}^T [\mathcal{K}_h(x - X_t) - \mathrm{E}\mathcal{K}_h(x - X_t)]\right]^2 \\ &= \frac{1}{T^2} \sum_{t=1}^T \mathrm{E}(Z_t^2) \text{ (由独立性得出)} \\ &= \frac{1}{T} \mathrm{E}\left[\prod_{i=1}^d K_h(x_i - X_{it}) - \mathrm{E}\prod_{i=1}^d K_h(x_i - X_{it})\right]^2 \\ &= \frac{1}{T} \left[\mathrm{E}\prod_{i=1}^d K_h^2(x_i - X_{it}) - \left[\mathrm{E}\prod_{i=1}^d K_h(x_i - X_{it})\right]^2\right] \\ &= \frac{1}{Th^d} f(x) D_K^d + o(T^{-1}h^{-d}). \end{aligned}$$

不难看出  $\hat{f}(x)$  的渐近方差与  $Th^d$  的倒数成比例，其中  $Th^d$  大约是落在以点  $x$  为中心、边长为  $2h$  的  $d$  维超立方体中的有效样本大小。虽然上述结论是在独立同分布的情况下推导得到的，但是在适当的  $\alpha$ -混合条件下，上述结果仍然成立。

接下来， $\hat{f}(x)$  的均方误差 (MSE) 为

$$\mathrm{MSE}[\hat{f}(x)] = \frac{1}{Th^d} f(x) D_K^d + \frac{1}{4} C_K^2 h^4 \left[\sum_{i=1}^d f_{ii}(x)\right]^2 + o(T^{-1}h^{-d} + h^4) = O(T^{-1}h^{-d} + h^4).$$

通过适当选择带宽  $h$ ，可以获得  $\hat{f}(x)$  向  $f(x)$  的最优均方误差 (MSE) 收敛速度： $T^{-\frac{4}{4+d}}$ ，最优带宽为：

$$h_0 = \left[ \frac{dD_K^2}{C_K^2} \frac{1/f(x)}{\left[\sum_{i=1}^d f_{ii}(x)/f(x)\right]^2} \right]^{\frac{1}{d+4}} T^{-\frac{1}{d+4}}.$$

基于此可得 MSE 的收敛率：

1. 如果  $d = 1$ ，则  $\mathrm{MSE}[\hat{f}(x)] \propto T^{-\frac{4}{5}}$ ，

2. 如果  $d = 2$ ，则  $\mathrm{MSE}[\hat{f}(x)] \propto T^{-\frac{2}{3}}$ ，

3. 如果  $d = 3$ ，则  $\mathrm{MSE}[\hat{f}(x)] \propto T^{-\frac{4}{7}}$ 。

不难看出，维度  $d$  越大， $\hat{f}(x)$  的收敛速度越慢。这就是所谓的“维数诅咒 (curse of dimensionality)”。维数诅咒”意味着需要大样本才能对  $f(x)$  进行合理的估计，且样本大小  $T$  需要随着维度  $d$  的增加而指数级增长。因此，在金融学及经济学实证研究中，受限于样本大小，很难看到维度  $d > 5$  的非参数估计。

以下是几种处理维数诅咒的常用方法：可以假设联合概率密度函数  $f(x)$  为各个分量密度函数的乘积，即  $X_{1t}, X_{2t}, \dots, X_{dt}$  互相独立，可得

$$f(x) = \prod_{i=1}^d g_i(x_i).$$

此外，还可以假设时间序列  $\{X_t\}$  是一个马尔可夫过程，则

$$\begin{aligned} f(X_t | \mathcal{F}_{t-1}) &= f(X_t | X_{t-1}) \\ &= \frac{f(X_t, X_{t-1})}{g(X_{t-1})}. \end{aligned}$$

其中， $\mathcal{F}_{t-1} = (X_{t-1}, X_{t-2}, \dots)$  是无限维的信息集。在这种情况下， $f(X_t, X_{t-1})$  只依赖于  $X_t$  和  $X_{t-1}$ ，该假设同样可以减少所需估计的维度。最后，可以采用投影寻踪法 (projection pursuit)，即假设多变量函数是一些解释变量的线性组合的未知函数。通过非参数方法估计未知函数和组合系数。概率密度函数估计在金融计量经济学中有许多应用。[Aït-Sahalia \(1996\)](#) 使用基于核的边际密度估计量  $\hat{g}(x)$  来检验短期利率扩散模型是否正确设定；而 [Hong & Li \(2005\)](#) 则采用非参数方法估计联合密度函数  $\hat{f}_j(x, y)$  以检验连续时间模型的适用性，并探讨其在利率仿射期限结构模型 (affine term structure models) 中的应用。

在金融学中，“仿射期限结构模型”是一种用于描述和预测利率期限结构的数学模型。这些模型被称为“仿射”，是因为它们假设债券的即期利率（或零息债券收益率）可以被表示为一组状态变量的仿射（即线性加常数）函数。具体来说，在仿射期限结构模型中，即期利率  $r(t)$  被建模为：

$$r(t) = a + b'X(t),$$

其中  $a$  是一个常数， $b$  是参数向量， $X(t)$  是由一个或多个风险因素构成的向量。这些风险因素的动态通常由随机微分方程描述。仿射模型的一个关键特性是，它们简化了金融衍生品（如债券、利率互换和期权）的定价，因为这些模型给出了债券价格和其他衍生品的解析表达式。这使得风险管理和服务更为高效。

在 R 中，可以使用 `ks` 包来进行多维核密度估计。以下是一个使用 R 代码估计多维核密度函数的示例。此代码首先提取 FTSE 以及 DAX 股指数据，然后使用 `ks::kde` 函数来估计其核密度。结果见图7.3。

```

1 library (datasets)
2 library (ks)
3
4 # 加载EuStockMarkets 数据集
5 data ("EuStockMarkets")
6
7 # 提取英国 (FTSE) 和德国 (DAX) 的指数
8 ftse_prices <- EuStockMarkets[, "FTSE"]
9 dax_prices <- EuStockMarkets[, "DAX"]
10
11 # 计算两个指数的日收益率
12 ftse_returns <- diff (ftse_prices) / lag (ftse_prices, -1) * 100
13 dax_returns <- diff (dax_prices) / lag (dax_prices, -1) * 100
14
15 # 移除因使用diff函数产生的NA值
16 ftse_returns <- na.omit (ftse_returns)
17 dax_returns <- na.omit (dax_returns)
18
19 # 创建包含两个收益率的数据框
20 market_returns <- data.frame (FTSE_Returns = ftse_returns, DAX_Returns =
 dax_returns)

```

```

21
22 # 使用ks包进行多变量核密度估计
23 # 确保数据的维度正确以便进行多变量分析
24 data_matrix <- as.matrix(market_returns)
25 kde_result <- kde(x = data_matrix)
26
27 # 2D核密度估计结果的可视化
28 plot(kde_result, display = "filled.contour")
29
30 # 3D核密度估计结果的可视化
31 plot(kde_result, display = "persp")
32

```

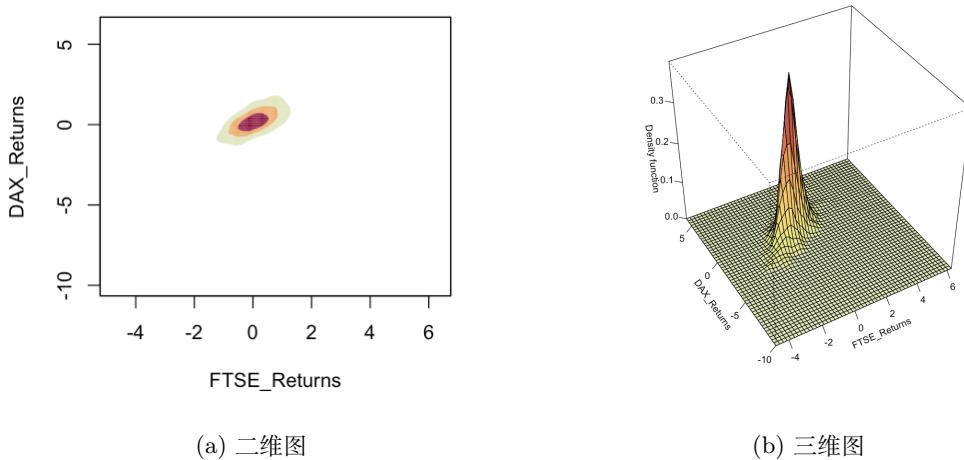


图 7.3: 核密度估计的可视化

## 7.2 非参数回归模型

与以往我们熟悉的回归模型相比，非参数回归模型有很多优点。首先，非参数回归不需要预先假设数据之间的关系具有特定的数学形式（如线性或多项式关系），这使得它能够适应更复杂和未知的关系模式。其次，由于非参数方法不依赖于严格的模型假设，因此它们在处理非线性、异方差性（heteroscedasticity）或复杂依赖结构的数据时，可以减少因模型设定不当引入的偏差。最后，非参数回归完全由数据驱动（data-driven），不需要对数据的分布形态或关系进行严格设定。非参数方法特别适合于探索性数据分析，可以帮助研究者发现数据中的新现象或规律。

以下代码通过模拟非线性数据，使用两种不同的回归方法（线性回归和半参数非参数回归）来拟合数据，并绘制拟合结果和残差图。这里我们采用 LOESS 函数，其英文全称为 Locally Estimated Scatterplot Smoothing，默认采用局部加权回归（即一阶多项式）来拟合数据；在函数中指定 `degree = 2` 参数来使用二次多项式等。回归结果见图 7.4。图中黑色点为模拟的数据点，由  $y = \sin(x)$  模型加上噪声生成。蓝色实线是线性回归模型的拟合结果，该模型假设数据点与  $x$  之间存在线性关系。红色点表示非参数回归模型的残差。非参数回归明显优于线性回归，原因如下。首先，从拟合值图可以看到，LOESS 更好地拟合了数据的非线性关系；其次，线性回归的残差图呈现出系统性的结构，而非参数回归的残差分

布较为随机。不难看出，非参数回归方法能够更准确地捕捉数据中的非线性关系，而线性回归则由于假设数据呈线性关系，因此无法灵活地适应复杂的非线性模式。

本节介绍了 Nadaraya-Watson 估计量与局部多项式估计量。这类方法无需对数据的全局关系作出明确假设，而是通过局部加权拟合进行预测，因而具有更强的适应性、对模型设定更为稳健。相较之下，线性回归或多项式回归等参数模型往往依赖较强的先验假设（如线性或多项式关系）。这些假设在实际中未必成立，易导致拟合效果不佳。

```

1 # 加载必要的库
2 library (ggplot2)
3 library (ggpubr)
4
5 # 设置随机种子以保证结果可重复
6 set.seed (123)
7
8 # 生成非线性数据: y = sin (x) + 噪声
9 x_vals <- seq (-3, 3, length.out = 100)
10 y_vals <- sin (x_vals) + rnorm (length (x_vals) , sd = 0.3) # 添加噪声
11
12 # 创建数据框
13 data <- data.frame (x = x_vals, y = y_vals)
14
15 # 线性回归模型
16 linear_model <- lm (y ~ x, data = data)
17
18 # 非参数回归使用LOESS (局部多项式回归)
19 loess_model <- loess (y ~ x, data = data)
20
21 # 从两个模型中获得预测值
22 data$y_pred_linear <- predict (linear_model, newdata = data)
23 data$y_pred_loess <- predict (loess_model, newdata = data)
24
25 # 计算两个模型的残差
26 data$residuals_linear <- data$y - data$y_pred_linear
27 data$residuals_loess <- data$y - data$y_pred_loess
28
29 # 绘制结果:
30 # 1. 显示两个模型的拟合值
31 # 2. 显示两个模型的残差
32 plot1 <- ggplot (data, aes (x = x)) +
33 geom_point (aes (y = y) , color = "black") + # Actual data points
34 geom_line (aes (y = y_pred_linear) , color = "blue", size = 1) + # 线性回
归拟合曲线
35 geom_line (aes (y = y_pred_loess) , color = "red", size = 1) + # LOESS拟合
曲线
36 labs (title = "Fitted Values", x = "x", y = "y") +
37 theme_minimal () +
38 theme (legend.title = element_blank ())
39
40 plot2 <- ggplot (data, aes (x = x)) +
41 geom_point (aes (y = residuals_linear) , color = "blue") + # 线性回归残差
42 labs (title = "Residuals of Linear Regression", x = "x", y = "Residuals")

```

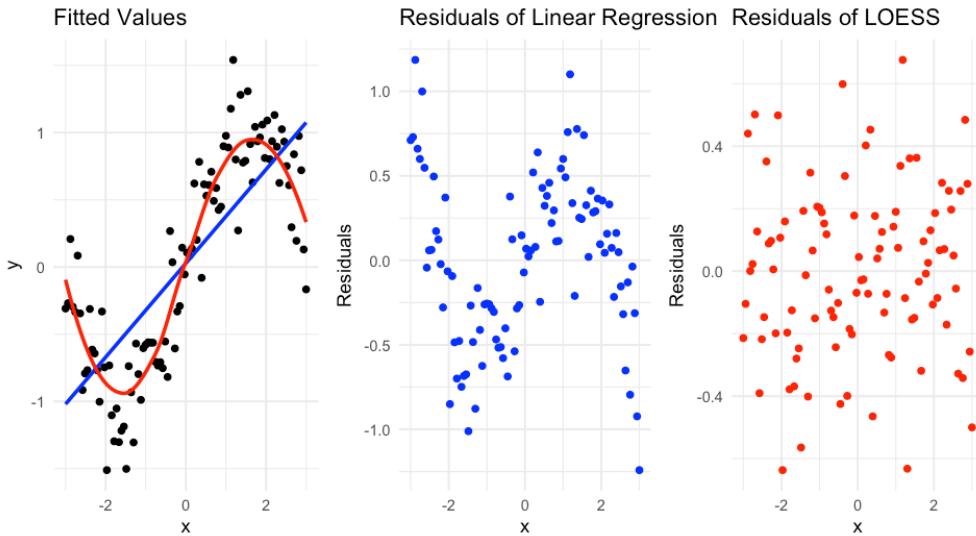


图 7.4: 线性回归与非参数回归的拟合结果及残差图

```

43 +
44 theme_minimal_()
45
46 plot3 <- ggplot (data, aes (x = x)) +
47 geom_point (aes (y = residuals_loess) , color = "red") + # LOESS 残差
48 labs (title = "Residuals of LOESS" , x = "x" , y = "Residuals") +
49 theme_minimal_()
50
51 # 将三个图安排在一个1行3列的网格中
52 ggarrange (plot1, plot2, plot3, ncol = 3, nrow = 1)

```

### 7.2.1 Nadaraya-Watson 估计量

顾名思义, Nadaraya-Watson 估计量由 Nadaraya (1964) 和 Watson (1964) 分别提出。对任意给定的  $x$ , 其定义为

$$\hat{r}(x) = \frac{\hat{m}(x)}{\hat{g}(x)},$$

其中分子是  $\{Y_t\}$  的加权样本均值,

$$\hat{m}(x) = \frac{1}{T} \sum_{t=1}^T Y_t K_h(x - X_t),$$

分母是点  $x$  处密度  $g(x)$  的核估计量,

$$\hat{g}(x) = \frac{1}{T} \sum_{t=1}^T K_h(x - X_t).$$

如果定义权重函数

$$\hat{W}_t(x) = \frac{K_h(x - X_t)}{\sum_{t=1}^T K_h(x - X_t)},$$

且令

$$\sum_{t=1}^T \widehat{W}_t = 1,$$

则可得

$$\widehat{r}(x) = \sum_{t=1}^T \widehat{W}_t(x) Y_t.$$

不难看出, Nadaraya–Watson 估计量是  $\{Y_t\}_{t=1}^n$  的局部加权样本均值, 其中权重  $\widehat{W}_t(x)$  在核函数  $K(u)$  的支撑集为  $[-1, 1]$  时, 对区间  $[x - h, x + h]$  之外的观测赋零权。

如果我们假设核函数是均匀核  $K(u) = \frac{1}{2}\mathbf{1}(|u| \leq 1)$ , 则 Nadaraya–Watson 估计量为:

$$\widehat{r}(x) = \frac{\sum_{t=1}^T Y_t \mathbf{1}(|X_t - x| \leq h)}{\sum_{t=1}^T \mathbf{1}(|X_t - x| \leq h)}.$$

即这是一个局部样本均值。Tukey (1961) 将之称为回归直方图 (regressogram)。

以下代码采用 `mtcars` 数据集, 演示如何使用 Nadaraya–Watson 估计量分析马力对每加仑英里数的影响。

```

1 # 加载 KernSmooth 包
2 if (!requireNamespace ("KernSmooth", quietly = TRUE)) {
3 install.packages ("KernSmooth")
4 }
5 library (KernSmooth)
6
7 # 加载 mtcars 数据集
8 data (mtcars)
9
10 # 提取变量
11 x <- mtcars$hp # 解释变量: 马力
12 y <- mtcars$mpg # 被解释变量: 每加仑英里数
13
14 # 使用 KernSmooth 的 dpill 函数选择带宽
15 bw <- dpill (x, y)
16
17 # 使用 locpoly 执行 Nadaraya-Watson 核回归
18 fit <- locpoly (x, y, bandwidth=bw, degree=0, kernel="normal", gridsize
19 =100)
20
21 # 创建图形以可视化结果
22 plot (x, y, main="Nadaraya-Watson Estimation",
23 xlab="Horsepower", ylab="Miles per Gallon", pch=19)
24 lines (fit, col="blue") # 添加回归线

```

在代码中, 我们采用 `dpill` 函数自动选择核平滑带宽, 并使用 `locpoly` 函数拟合 Nadaraya–Watson 模型。该函数支持局部多项式回归 (Nadaraya–Watson 平滑对应多项式的阶数为 0) 的情况。后续我们将介绍局部多项式回归。

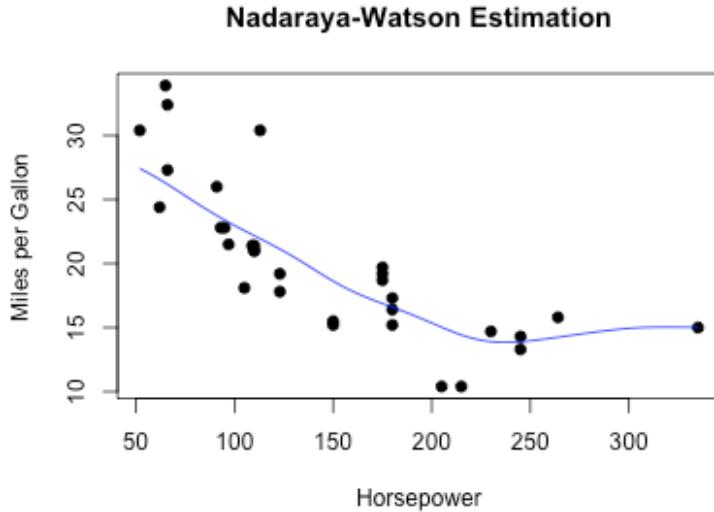


图 7.5: 使用 Nadaraya-Watson 估计量评估马力对每加仑英里数 (MPG) 的影响

### 7.2.1.1 均方误差和最优带宽

Nadaraya-Watson 估计量  $\hat{r}(x)$  是两个随机变量的比率。为了简化渐近分析，对于任意给定的点  $x$ ，我们考虑以下分解：

$$\begin{aligned}\hat{r}(x) - r(x) &= \frac{\hat{m}(x) - r(x)\hat{g}(x)}{\hat{g}(x)} \\ &= \frac{\hat{m}(x) - r(x)\hat{g}(x)}{\mathbb{E}[\hat{g}(x)]} + \frac{[\hat{m}(x) - r(x)\hat{g}(x)]}{\mathbb{E}[\hat{g}(x)]} \cdot \frac{[\mathbb{E}[\hat{g}(x)] - \hat{g}(x)]}{\hat{g}(x)} \\ &= \frac{\hat{m}(x) - r(x)\hat{g}(x)}{\mathbb{E}[\hat{g}(x)]} + \text{高阶项}.\end{aligned}$$

由于当  $T \rightarrow \infty$  时

$$\hat{g}(x) - \mathbb{E}[\hat{g}(x)] \xrightarrow{p} 0,$$

$$\mathbb{E}[\hat{g}(x)] \rightarrow g(x) \int_{-1}^1 K(u)du > 0 \text{ 当 } h \rightarrow 0,$$

且第二项是一个高阶项，所以在 Nadaraya-Watson 估计量的渐近分析中，第一项（即核回归的偏差项）主导整个估计量的收敛行为。具体来讲，第一项包含了估计量相对于真实回归函数的偏差，而这种偏差随着  $T \rightarrow \infty$  和  $h \rightarrow 0$  而减小的速度慢于第二项中随机波动的消失速度。

考虑第一项，基于  $Y_t = r(X_t) + \varepsilon_t$ ，分子表达式为：

$$\begin{aligned}\hat{m}(x) - r(x)\hat{g}(x) &= \frac{1}{T} \sum_{t=1}^T [Y_t - r(x)] K_h(x - X_t) \\ &= \frac{1}{T} \sum_{t=1}^T \varepsilon_t K_h(x - X_t) + \frac{1}{T} \sum_{t=1}^T [r(X_t) - r(x)] K_h(x - X_t) \\ &= \hat{V}(x) + \hat{B}(x), \text{ 即} \\ &= \text{方差部分} + \text{偏差部分}.\end{aligned}$$

为简化讨论，假设  $\{Y_t, X_t\}$  为独立同分布。针对方差部分，我们有：

$$\begin{aligned}
 \text{E}[\hat{V}(x)^2] &= \text{E}\left[\frac{1}{T} \sum_{t=1}^T \varepsilon_t K_h(x - X_t)\right]^2 \\
 &= \frac{1}{T^2} \text{E}\left[\sum_{t=1}^T \varepsilon_t K_h(x - X_t)\right]^2 \\
 &= \frac{1}{T^2} \sum_{t=1}^T \text{E}[\varepsilon_t^2 K_h^2(x - X_t)] \quad (\text{由独立性和 } \text{E}(\varepsilon_t | X_t) = 0 \text{ 可得}) \\
 &= \frac{1}{T} \text{E}[\varepsilon_t^2 K_h^2(x - X_t)] \\
 &= \frac{1}{T} \text{E}[\sigma^2(X_t) K_h^2(x - X_t)] \quad (\text{由 } \text{E}(\varepsilon_t^2 | X_t) = \sigma^2(X_t) \text{ 可得}) \\
 &= \frac{1}{T} \int_a^b \left[ \frac{1}{h} K\left(\frac{x-y}{h}\right) \right]^2 \sigma^2(y) g(y) dy \\
 &= \frac{1}{Th} \sigma^2(x) g(x) \int_{-1}^1 K^2(u) du [1 + o(1)],
 \end{aligned}$$

最后一行是基于变量替换以及  $\sigma^2(\cdot)g(\cdot)$  的连续性，此外  $\sigma^2(x) = \text{E}(\varepsilon_t^2 | X_t = x)$  是给定  $X_t = x$  时  $\varepsilon_t$  或  $Y_t$  的条件方差。不难看出，方差  $\text{E}[\hat{V}(x)]^2$  与  $Th$  成反比，因为  $Th$  是落入区间  $[x-h, x+h]$  的有效样本大小。

对于分母部分，当  $h \rightarrow 0$  时，如果  $\int_{-1}^1 K(u) du = 1$ ，有：

$$\begin{aligned}
 \text{E}[\hat{g}(x)] &= \text{E}[K_h(x - X_t)] = \int_a^b \frac{1}{h} K\left(\frac{x-y}{h}\right) g(y) dy \\
 &\rightarrow g(x) \int_{-1}^1 K(u) du = g(x),
 \end{aligned}$$

进而可得：

$$\text{E}\left[\frac{\hat{V}(x)}{\text{E}[\hat{g}(x)]}\right]^2 = \frac{1}{Th} \frac{\sigma^2(x)}{g(x)} \int_{-1}^1 K^2(u) du [1 + o(1)].$$

Nadaraya-Watson 估计量在点  $x$  的渐近方差与  $(Th)^{-1}$  同阶，且与局部噪声水平  $\sigma^2(x)$  以及核的粗糙度  $R(K) = \int K^2$  成正比，与局部密度  $g(x)$  成反比。位于区间  $[x-h, x+h]$  内的样本量的期望约为  $2Th g(x)$ 。

针对偏差部分  $\hat{B}(x)$ ，有：

$$\hat{B}(x) = \text{E}\hat{B}(x) + [\hat{B}(x) - \text{E}\hat{B}(x)].$$

对于任意给定的内部点  $x \in [a+h, b-h]$ ，定义  $m(x) = r(x)g(x)$ ，有：

$$\begin{aligned}
E\widehat{B}(x) &= E[r(X_t)K_h(x-X_t)] - r(x)E[K_h(x-X_t)] \\
&= \int_a^b r(z)K_h(x-z)g(z)dz - r(x)\int_a^b K_h(x-z)g(z)dz \\
&= \int_a^b m(z)K_h(x-z)dz - r(x)\int_a^b g(z)K_h(x-z)dz \\
&= \int_{(a-x)/h}^{(b-x)/h} m(x+hu)K(u)du - r(x)\int_{(a-x)/h}^{(b-x)/h} g(x+hu)K(u)du \\
&= m(x)\int_{-1}^1 K(u)du + hm'(x)\int_{-1}^1 uK(u)du + \frac{1}{2}h^2m''(x)\int_{-1}^1 u^2K(u)du[1+o(1)] \\
&\quad - r(x)g(x)\int_{-1}^1 K(u)du - hr(x)g'(x)\int_{-1}^1 uK(u)udu - \frac{1}{2}h^2r(x)g''(x)\int_{-1}^1 u^2K(u)du[1+o(1)] \\
&= \frac{1}{2}h^2[m''(x) - r(x)g''(x)]\int_{-1}^1 u^2K(u)du[1+o(1)] \\
&= \frac{1}{2}h^2[r''(x)g(x) + 2r'(x)g'(x)]C_K + o(h^2),
\end{aligned}$$

其中我们使用如下链式法则：

$$\begin{aligned}
m''(x) &= [r(x)g(x)]'' = [r'(x)g(x) + r(x)g'(x)]' \\
&= r''(x)g(x) + 2r'(x)g'(x) + r(x)g''(x).
\end{aligned}$$

当  $h \rightarrow 0$  时, 有:

$$E[\widehat{g}(x)] \rightarrow g(x) \int_{-1}^1 K(u)du = g(x),$$

和  $\int_{-1}^1 K(u)du = 1$ , 因此标准化偏差可以表示为:

$$\begin{aligned}
E\left[\frac{\widehat{B}(x)}{E\widehat{g}(x)}\right] &= \frac{h^2}{2}\left[\frac{m''(x)}{g(x)} - \frac{r(x)g''(x)}{g(x)}\right]C_K + o(h^2) \\
&= \frac{h^2}{2}\left[r''(x) + \frac{2r'(x)g'(x)}{g(x)}\right]C_K + o(h^2).
\end{aligned}$$

不难看出,  $\widehat{r}(x)$  的偏差包含两部分: 第一部分是由分子  $\widehat{m}(x)$  贡献的  $\frac{1}{2}h^2[m''(x)/g(x)]C_K$ ; 第二部分是由分母  $\widehat{g}(x)$ , 即密度  $g(x)$  的估计量贡献的  $-\frac{1}{2}h^2[r(x)g''(x)/g(x)]C_K$ 。

### 7.2.1.2 Nadaraya-Watson 估计量的边界问题

当  $x$  处于边界上 (即  $x \in [a, a+h] \cup (b-h, b]$ ) 时, 随着  $h \rightarrow 0$ , 有  $E[\widehat{B}(x)] \rightarrow 0$ , 并且

$$\frac{E[\widehat{B}(x)]}{E[\widehat{g}(x)]} = O(h) = o(1).$$

由

$$E[\widehat{B}(x)] = [m(x) - r(x)g(x)] \int_{-\tau}^1 K(u)du + O(h) = O(h),$$

可得

$$\frac{\mathbb{E}[\widehat{B}(x)]}{\mathbb{E}[\widehat{g}(x)]} = \frac{[m(x) - r(x)g(x)] \int_{-\tau}^1 K(u) du}{g(x) \int_{-\tau}^1 K(u) du} + O(h) = O(h).$$

然而, 由于  $\int_{-\tau}^1 u K(u) du \neq 0$ , 边界处的偏差阶数为  $O(h)$ , 较内部区域的  $O(h^2)$  收敛更慢。

接着, 我们还需证明  $\widehat{B}(x) - \mathbb{E}[\widehat{B}(x)]$  是一个高阶项。为了简化证明, 我们假设  $\{Y_t, X_t\}$  为独立同分布序列。令

$$Z_t \equiv Z_t(x) = [r(X_t) - r(x)] K_h(x - X_t),$$

由此可得,

$$\begin{aligned} \mathbb{E}[\widehat{B}(x) - \mathbb{E}\widehat{B}(x)]^2 &= E \left[ \frac{1}{T} \sum_{t=1}^T (Z_t - EZ_t) \right]^2 \\ &= \frac{1}{T^2} \sum_{t=1}^T \mathbb{E} (Z_t - EZ_t)^2 \text{ 由独立性可得} \\ &\leq \frac{1}{T} \mathbb{E} (Z_t^2) \\ &= \frac{1}{T} \mathbb{E} \left\{ [r(X_t) - r(x)]^2 K_h^2(x - X_t) \right\} \\ &\leq \frac{Ch}{T} [1 + o(1)] \end{aligned}$$

为一个高阶无穷小项。

由此可得,

$$\begin{aligned} \mathbb{E}[\widehat{m}(x) - r(x)\widehat{g}(x)]^2 &= \mathbb{E}[\widehat{V}(x) + \widehat{B}(x)]^2 \\ &= \mathbb{E}[\widehat{V}^2(x)] + \mathbb{E}[\widehat{B}^2(x)] \\ &= \mathbb{E}[\widehat{V}^2(x)] + \mathbb{E}[\widehat{B}^2(x)] + \mathbb{E}[\widehat{B}(x) - \mathbb{E}\widehat{B}(x)]^2 \\ &= \frac{1}{Th} D_K \sigma^2(x) g(x) + \frac{h^4}{4} C_K^2 [m''(x) - r(x)g''(x)]^2 \\ &\quad + o((Th)^{-1} + h^4). \end{aligned}$$

因此, 漐近 MSE 为

$$\begin{aligned} \mathbb{E}[\widehat{r}(x) - r(x)]^2 &= \frac{1}{Th} \frac{\sigma^2(x)}{g(x)} D_K + \frac{h^4}{4} \left[ \frac{r''(x) + 2r'(x)g'(x)}{g(x)} \right]^2 C_K^2 \\ &\quad + o((Th)^{-1} + h^4) \\ &= O(T^{-1}h^{-1} + h^4). \end{aligned}$$

方差与  $(Th)^{-1}$  成正比, 均方偏差与  $h^4$  成正比。因此, 增加  $h$  会减少方差但会增加偏差, 减少  $h$  会增加方差但会减少偏差。通过平衡方差和均方偏差可以实现最优平滑。

最优的带宽  $h$  可以通过最小化  $\widehat{r}(x)$  的均方误差 (MSE) 获得:

$$h^* = c^* T^{-1/5},$$

其中最优比例常数  $c^*$  为：

$$\begin{aligned} c^* &= \left[ \frac{D_K}{C_K^2} \frac{\sigma^2(x)/g(x)}{[m''(x)/g(x) - r(x)g''(x)/g(x)]^2} \right]^{\frac{1}{5}} \\ &= \left[ \frac{D_K}{C_K^2} \frac{\sigma^2(x)g(x)}{[r''(x) + 2r'(x)g'(x)]^2} \right]^{\frac{1}{5}}. \end{aligned}$$

因此，当数据的  $\sigma^2(x)$  较大时，最优带宽  $h^*$  也会较大；而当回归函数  $r(x)$  不平滑（其导数较大）时， $h^*$  会比较小。

和密度估计情形类似，核回归估计的最优核函数仍然是 Epanechnikov 核，

$$K(u) = \frac{3}{4} (1 - u^2) \mathbf{1}(|u| < 1).$$

不过，在实证分析中，带宽  $h$  的选择通常比核函数  $K(\cdot)$  的具体形状更为关键。其原因在于，常用的二阶核在满足  $\int K = 1$ 、对称且可用于二阶近似等基本条件时，只会在 AMISE 的常数项上产生细微差异；相较之下，带宽直接决定方差项（随  $(Th)^{-1}$  变化）与偏差项（随  $h^4$  变化）之间的权衡，使误差对  $h$  的变化远比对核形状的变化更为敏感。另一方面，规范化与局部性意味着核权重主要集中在估计点附近，远离估计点的数据权重极小，因此在选定合适带宽后，改换不同的标准二阶核往往难以显著改变估计结果。实践中更应侧重采用数据驱动的带宽选择方法（如交叉验证、Plug-in 或规则法则）来控制整体误差。

AMISE 是“Asymptotic Mean Integrated Squared Error”（渐近积分均方误差）的缩写，它是

$$\text{MISE} = \mathbb{E} \left[ \int (\hat{\theta}(x) - \theta(x))^2 dx \right]$$

大样本近似主项，其中  $\theta$  表示目标函数（如密度  $f$  或回归函数  $r$ ）， $\hat{\theta}$  为其估计量。以 Nadaraya-Watson 回归为例，AMISE 可写成“方差项 + 偏差项”的形式：

$$\text{AMISE}(h) \approx \frac{R(K)}{Th} \int \frac{\sigma^2(x)}{g(x)} dx + \frac{1}{4} \mu_2(K)^2 h^4 \int (r''(x))^2 g(x) dx,$$

其中  $R(K) = \int K^2(u) du$  度量核的“粗糙度”， $\mu_2(K) = \int u^2 K(u) du$  为核的二阶矩， $g$  为  $X$  的密度， $\sigma^2(x) = \text{Var}(Y | X = x)$ 。最小化该近似得到最优带宽阶数  $h_{\text{opt}} \propto T^{-1/5}$ ，AMISE 可用于比较不同核与带宽对整体误差的影响；这也是为何在满足基本条件的前提下，带宽选择通常比具体核形状更为重要。

## 7.2.2 局部多项式估计量

本节我们介绍局部多项式估计量。局部多项式方法能够更好地处理边界效应，从而在边界处提供更准确的估计。此外，局部多项式允许通过调整多项式的阶数来控制偏差与方差之间的权衡，使得我们可以根据具体的数据结构选择最合适的模型复杂度。

为了更好得解释局部多项式估计量，我们首先考虑 Nadaraya-Watson 估计量  $\hat{r}(x)$  的另一种解释。考虑最小化残差平方和（sum of squared residuals，简称 SSR）

$$\min_r \sum_{t=1}^T (Y_t - r)^2,$$

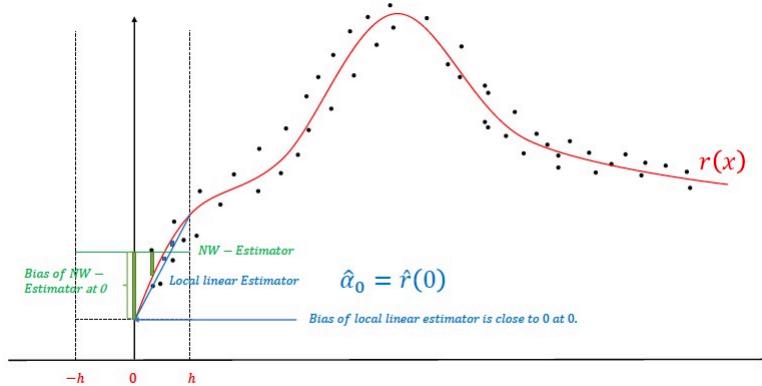


图 7.6: Nadaraya-Watson 估计量与局部线性估计量的比较

其中  $r$  是一个常数。

最优解为样本均值:

$$\hat{r} = \bar{Y} \equiv \frac{1}{T} \sum_{t=1}^T Y_t.$$

接着, 我们讨论最小化局部加权残差平方和 (locally weighted sum of squared residuals):

$$\min_r \sum_{t=1}^T (Y_t - r)^2 K_h(x - X_t).$$

类似地, 这里的  $r$  仍为常数。当核函数  $K(u)$  的支撑集为  $[-1, 1]$  时, 上式对应解释变量  $\{X_t\}$  落在区间  $[x - h, x + h]$  内、预测变量为  $\{Y_t\}$  的加权残差平方和。最小化问题对应的一阶条件 (first order condition, 简称 FOC) 为:

$$\sum_{t=1}^T (Y_t - \hat{r}(x)) K_h(x - X_t) = 0,$$

由此得到

$$\hat{r}(x) = \frac{\sum_{t=1}^T Y_t K_h(x - X_t)}{\sum_{t=1}^T K_h(x - X_t)} = \frac{\hat{m}(x)}{\hat{g}(x)}.$$

上式即为局部常数估计量。换言之, Nadaraya-Watson 估计量在每个小区间内进行样本加权平均。

在图 7.6 中, 我们使用 Nadaraya-Watson 估计量与局部线性估计量 (即阶数为 1 的局部多项式) 来估计回归函数  $r(x)$ 。可以看出, 相比于简单的 Nadaraya-Watson 加权平均, 局部线性估计的偏差更小。

若不止采用线性函数, 而进一步以更高阶多项式在局部区间内拟合: 给定  $X_t$  的支撑区间内一点  $x$ , 设  $z$  为  $x$  的一个小邻域内的任意点, 并假设在该邻域内  $r(z)$  的导数连续至  $p+1$  阶。由此可得  $(p+1)$  阶的泰勒级数展开如下:

$$\begin{aligned} r(z) &= \sum_{j=0}^p \frac{1}{j!} r^{(j)}(x) (z - x)^j + \frac{1}{(p+1)!} r^{(p+1)}(\xi) (z - x)^{p+1} \\ &= \sum_{j=0}^p \alpha_j (z - x)^j + \frac{1}{(p+1)!} r^{(p+1)}(\xi) (z - x)^{p+1}. \end{aligned}$$

上式是中值定理的一个应用，其中  $\xi$  位于  $x$  与  $z$  之间。多项式系数

$$\begin{aligned}\alpha_j &\equiv \alpha_j(x) \\ &= \frac{1}{j!} r^{(j)}(x), \quad j = 0, 1, \dots, p\end{aligned}$$

依赖于点  $x$ 。因此，只要该邻域内的观测数据足够多，就可以使用局部多项式模型来拟合函数  $r(z)$ 。

将局部加权最小二乘残差和最小化

$$\min_{\boldsymbol{\alpha}} \sum_{t=1}^T \left[ Y_t - \sum_{j=0}^p \alpha_j (X_t - x)^j \right]^2 K_h(X_t - x) = \sum_{t=1}^T (Y_t - \boldsymbol{\alpha}' Z_t)^2 K_h(X_t - x),$$

其中  $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \dots, \alpha_p)'$ ,  $Z_t = Z_t(x) = [1, (X_t - x), \dots, (X_t - x)^p]'$ 。注意,  $Z_t = Z_t(x)$  是一个  $(p+1)$  维的局部多项式向量，其依赖于位置  $x$ 。可得局部加权最小二乘估计量

$$\hat{r}(z) = \sum_{j=0}^p \hat{\alpha}_j (z - x)^j \quad (\text{对所有在 } x \text{ 邻域内的 } z),$$

其中  $\hat{\boldsymbol{\alpha}}$  为局部加权最小二乘估计量，其显式形式将在后文推导。

截距  $\hat{\alpha}_0$  是回归函数  $r(x)$  的估计量，而  $\nu! \hat{\alpha}_\nu$  是导数  $r^{(\nu)}(x)$  的估计量 ( $1 \leq \nu \leq p$ )。

由于

$$\hat{r}(z) = \sum_{j=0}^p \hat{\alpha}_j (z - x)^j \quad (\text{对所有在 } x \text{ 邻域内的 } z),$$

因此点  $x$  的回归估计量为

$$\hat{r}(x) = \sum_{j=0}^p \hat{\alpha}_j (x - x)^j = \hat{\alpha}_0.$$

此外,  $r^{(\nu)}(z)$  在  $x$  附近的导数估计量为

$$\hat{r}^{(\nu)}(z) = \sum_{j=\nu}^p j(j-1)\cdots(j-\nu+1) \hat{\alpha}_j (z - x)^{j-\nu}, \quad \nu \leq p,$$

从而点  $x$  处的  $\nu$  阶导数估计量是

$$\hat{r}^{(\nu)}(x) = \nu! \hat{\alpha}_\nu.$$

局部多项式平滑能够同时估计  $r^{(\nu)}(x)$ ,  $\nu = 0, 1, \dots, p$ 。

值得注意的是，局部多项式平滑估计量的残差平方和总是小于或等于 Nadaraya-Watson 估计量的加权残差平方和。这是因为 Nadaraya-Watson 估计量是局部多项式平滑估计量的一种特殊形式（当  $p=0$  时），这一点在图 7.6 中得到了很好的诠释。

此外，局部多项式方法需要设置多项式的阶数  $p$ 、带宽  $h$  以及核函数  $K(\cdot)$ 。当  $p=1$  时，对应的是局部线性平滑估计量。带宽  $h$  的选择可以基于数据驱动方法，例如交叉验证 (cross-validation) 或插件方法 (plug-in methods)。

用矩阵和向量的方式可以更好得诠释局部多项式平滑估计量。设  $(p+1) \times 1$  的多项式回归向量为：

$$Z_t = Z_t(x) = [1, (X_t - x), (X_t - x)^2, \dots, (X_t - x)^p]',$$

以及权重函数

$$W_t = W_t(x) = K_h(x - X_t) = \frac{1}{h} K\left(\frac{x - X_t}{h}\right).$$

局部加权残差平方和为：

$$\sum_{t=1}^T \left[ Y_t - \sum_{j=0}^p \alpha_j (X_t - x)^j \right]^2 K_h(x - X_t) = \sum_{t=1}^T (Y_t - Z_t' \alpha)^2 W_t = (Y - Z\alpha)' W (Y - Z\alpha),$$

其中 FOC 为

$$\sum_{t=1}^T Z_t W_t (Y_t - Z_t' \hat{\alpha}) = 0,$$

即

$$\sum_{t=1}^T Z_t W_t Y_t = \left( \sum_{t=1}^T Z_t W_t Z_t' \right) \hat{\alpha}.$$

由此可得：

$$\hat{\alpha} \equiv \hat{\alpha}(x) = \left( \sum_{t=1}^T Z_t W_t Z_t' \right)^{-1} \sum_{t=1}^T Z_t W_t Y_t = (Z' W Z)^{-1} Z' W Y,$$

其中  $W = W(x) = \text{diag}(W_1, \dots, W_T)$  为  $T \times T$  对角矩阵， $Z$  为  $T \times (p+1)$  矩阵， $Y$  为  $T \times 1$  列向量， $K(\cdot)$  为支持支撑集是  $[-1, 1]$  的当核函数， $\hat{\alpha}$  为局部加权最小二乘估计量。

在 R 中采用局部多项式估计量，仅需将 `locpoly (x, y, bandwidth=bw, degree=0, kernel="normal", gridsize=100)` 中的 `degree=0` 调高，例如令 `degree=1`，`degree=2` 等。

相比 Nadaraya-Watson 估计量，局部多项式估计量的拟合效果更好；并且可以证明，它等价于一个已知  $g(x)$  的 Nadaraya-Watson 估计量。此外，局部多项式在边界处的表现更优，具有良好的应用价值。在资产定价与风险管理中，利用局部多项式回归可以在不同时间点估计资产的风险溢价或利率，避免对模型形式作过强假设；同时，局部多项式估计量还可用于金融因子的建模，当不同因子之间存在非线性关系时，能够灵活刻画因子效应。

## 7.3 非参数统计在金融计量经济学中的应用

非参数方法可以广泛应用于各种类型的数据，包括分类数据、顺序数据（或排序数据）和计数数据等，特别适用于处理金融市场中复杂且非正态分布的数据。

### 7.3.1 游程检验

考虑第六章中针对 EMH 的检验，EMH 意味着资本市场是有效的，股票价格反映了所有相关信息。针对 EMH 的检验归根结底是对过去信息预测能力的检验。在第6.7 节的习题中，我们提到游程检验也可用于检测 EMH。这里我们将详细介绍游程检验。

游程检验是一种非参数统计方法，主要用于检验样本数据是否随机，或判断多个样本是否来自相同的总体分布。其原理是：若样本数据随机，则序列中不同类型元素的出现顺序也应随机；游程的长度与数量应遵循特定的概率分布。若实际观测到的数量与零假设（样本数据随机）相差过大，则有理由怀疑数据非随机，或样本间存在分布差异。

在金融市场中，“游程”通常指价格连续上涨或连续下跌的序列。例如，我们可以考察

市场连续多少个交易日出现上涨或下跌。这个问题与初学概率论中的硬币投掷情形类似：可连续观察到多少次正面或反面。下面给出游程长度的定义。

**定义 7.4 (游程长度):** 给定时间序列  $\{X_t\}_{t=1}^T$ , 令  $s_t = \text{sign}(X_t) \in \{-1, 0, 1\}$ 。定义游程长度序列  $\{Z_t\}$  为：

1. 取初值:  $Z_0 = 0$ 。
2. 对于  $t = 1, \dots, T$ , 按下述规则递推:
  - (a) 若  $s_t \neq 0$  且  $s_t = s_{t-1}$ , 则设  $Z_t = Z_{t-1} + 1$ ;
  - (b) 否则 ( $s_t = 0$  或  $s_t \neq s_{t-1}$ ), 设  $Z_t = 1$ 。

上述  $Z_t$  记录了同号连续段（游程）的当前长度。

不难看出,  $Z_t$  给出了当前游程的长度及其符号, 因此, 如果过去七天的回报是正的, 则  $Z_t = 7$ 。

我们可以在一些假设下对这一系列的游程进行统计检验。假设  $X_t$  是独立同分布且中位数为零, 那么  $\text{sign}(X_t)$  是独立同分布的伯努利 (Bernoulli) 随机变量, 取值为  $\pm 1$  的概率为  $1/2$ 。在这种情况下, 通过模拟可以验证, 对于  $n = 5000$ , 最长游程的均值 (即  $\max_{1 \leq t \leq 5000} Z_t$ ) 大约为 10.6, 标准差为 1.87。实际上, 最大游程长度的分布是已知的, 尽管它的解析表达式比较复杂。对于较大的  $n$ , 它大致遵循如下分布:

$$Z_n = \left\lfloor \frac{W}{\log(2)} + \frac{\log(n)}{\log(2)} - 1 \right\rfloor,$$

其中  $W$  是一个随机变量, 其分布函数为  $\Pr(W \leq t) = \exp(-\exp(-t))$ , 而  $\lfloor \cdot \rfloor$  表示向下取整函数。注: 第6.7节习题给出了游程检验的其他数学表达式。

### 7.3.2 非线性可预测性和非参数自回归模型

计量经济学建模的核心在于用自变量解释因变量, 而解释的前提是自变量对因变量具有预测能力。计量经济学中有许多相关检验, 例如广为人知的格兰杰因果检验。严格而言, 该检验更应称为“格兰杰预测能力检验”, 其核心在于检验在加入新的解释变量后, 因变量的预测能力是否得到提升。

然而, 许多检验受模型结构所限。例如, 第 4.6.4 节中的格兰杰因果检验是在 VAR 框架下提出的, 只能检验解释变量对被解释变量的线性预测能力。若要评估更全面的预测关系, 则需引入广义格兰杰因果关系或完全格兰杰因果关系。

**定义 7.5:** 广义格兰杰因果关系。假设  $\{X_t, Y_t\}$  是一个二元严格平稳时间序列过程, 令  $\mathcal{F}_{t-1}^X = \{X_{t-1}, X_{t-2}, \dots\}$  和  $\mathcal{F}_{t-1}^Y = \{Y_{t-1}, Y_{t-2}, \dots\}$  分别表示  $X$  和  $Y$  的滞后信息集。若

$$\Pr(Y_t \leq y | \mathcal{F}_{t-1}) \neq \Pr(Y_t \leq y | \mathcal{F}_{t-1}^Y),$$

则称相对于信息集  $\mathcal{F}_{t-1} = (\mathcal{F}_{t-1}^X, \mathcal{F}_{t-1}^Y)$ ,  $\{X_t\}$  对  $\{Y_t\}$  存在格兰杰原因。这里的“格兰杰原因”指分布上的格兰杰因果关系 (*Granger causality in distribution*), 亦称广义或完全格兰杰因果关系 (*complete Granger causality*)。

设  $X = \cos \theta$ 、 $Y = \sin \theta$ , 其中  $\theta \sim \text{Unif}[0, 2\pi]$ 。显然有  $Y^2 = 1 - X^2$ , 故  $X$  与  $Y$  之

间存在函数关系，并非独立。然而

$$E(XY) = \int_0^{2\pi} \cos \theta \sin \theta d\theta = 0, \quad E(X) = \int_0^{2\pi} \cos \theta d\theta = 0, \quad E(Y) = \int_0^{2\pi} \sin \theta d\theta = 0,$$

因此  $\text{Cov}(X, Y) = E(XY) - E(X)E(Y) = 0$ ，即  $X$  与  $Y$  不相关（线性关系为零）——从线性角度看， $X$  对  $Y$  没有预测能力。

针对非线性可预测性，可考虑检验  $\gamma(g) = \text{Cov}(Y_t, g_{t-1})$  或  $\text{corr}(Y_t, g_{t-1})$ ，其中  $g_{t-1}$  来自某个给定函数  $g$ （如  $g_{t-1} = Y_{t-1}^2$ 、 $g_{t-1} = \text{sign}(Y_{t-1})$ ，或  $g_{t-1} = \sum_{j=1}^p Y_{t-j}^2$ ）。对应的样本协方差为

$$\hat{\gamma}(g) = \frac{1}{T} \sum_{t=2}^T (Y_t - \bar{Y})(g_{t-1} - \bar{g}), \quad \bar{g} = \frac{1}{T} \sum_{t=1}^T g_t.$$

在适当的正则条件下，对序列  $\{Y_t\}, \{g_t\}$  可建立大数定律与中心极限定理。在原假设  $\gamma(g) = 0$  下，有

$$\sqrt{T}(\hat{\gamma}(g) - \gamma(g)) \implies N(0, V(g)), \quad V(g) = E[(Y_t - EY_t)^2(g_{t-1} - Eg_{t-1})^2].$$

进一步定义

$$\hat{S}(g) = \hat{V}(g)^{-1/2} \sqrt{T}(\hat{\gamma}(g) - \gamma(g)) = \frac{\sqrt{T}(\hat{\gamma}(g) - \gamma(g))}{\sqrt{\frac{1}{T} \sum_{t=1}^T (Y_t - \bar{Y})^2(g_{t-1} - \bar{g})^2}} \implies N(0, 1).$$

据此可构造不可预测性的检验：当  $|\hat{S}(g)| > z_{\alpha/2}$  时拒绝原假设。但需注意，上述检验的结论在一定程度上取决于函数  $g$  的具体选取。理想情况下，我们希望对所有由过去信息生成的函数集合  $\mathcal{G}$  同时进行联合检验；然而  $\mathcal{G}$  极其庞大（包含所有线性函数在内），因此直接对其整体进行检验在实践中不可行。

我们可以退而求其次，采用非参数自回归模型

$$m_k(y_1, \dots, y_p) = E(Y_{t+k} | Y_t = y_1, \dots, Y_{t+p-1} = y_p).$$

当  $m_k$  满足某些平滑性条件时，我们可以基于数据样本  $\{Y_1, \dots, Y_T\}$ ，采用非参数回归一致地估计  $m_k(\cdot)$ 。设  $K$  是一个连续且对称的核函数，定义在区间  $[-1, 1]$  上，满足  $\int K(u)du = 1$ ，并设带宽  $h > 0$ 。在单变量的情况下（即  $p = 1$ ），可得局部加权估计量：

$$\hat{m}_k(y) = \frac{\sum_{t=1}^{T-k} K\left(\frac{Y_t-y}{h}\right) Y_{t+k}}{\sum_{t=1}^{T-k} K\left(\frac{Y_t-y}{h}\right)}.$$

Härdle & Linton (1994) 讨论了带宽和核函数的选择问题。在某些条件下， $\hat{m}_k(y) \rightarrow m_k(y)$  成立，且满足中心极限定理。

非参数自回归模型也可用于检测 EMH。这里我们进一步拓展第六章内容，并引入一个介于 rw2 与 rw3 之间的 EMH 条件：

**rw3' 条件：** $\varepsilon_t$  是一个马尔可夫差分序列 (MDS)，即对于每一个时刻  $t$ ，都有：

$$E(\varepsilon_t | \varepsilon_{t-1}, \varepsilon_{t-2}, \dots) = 0 \quad (\text{概率为 } 1).$$

其中  $\varepsilon_t$  是描述市场价格变动的误差项。 $\varepsilon_t$  为 MDS，意味着对任何给定的历史信息， $\varepsilon_t$

的条件期望为零（即  $E(\varepsilon_t | \varepsilon_{t-1}, \varepsilon_{t-2}, \dots) = 0$ ），这表明当前的市场价格变化没有系统性的预测模式。

**定理 7.1：**假设  $Y_t$  满足 **rw3'** 条件，是平稳且方差有限的，则对所有  $k$  和  $y$ ， $m_k(y) = \mu$  均成立。假设  $Y_t$  为概率密度函数为  $f_Y$  的连续分布，且在  $y$  处可微且为正。假设  $h \rightarrow 0$  且  $Th \rightarrow \infty$ ，有：

$$\sqrt{Th} (\hat{m}_k(y) - \hat{\mu}) \Rightarrow N(0, \omega), \quad \omega = \int K(u)^2 du \frac{\sigma_k^2(y)}{f_Y(y)},$$

$$\hat{S}_k(y) = \frac{\hat{m}_k(y) - \hat{\mu}}{\sqrt{\frac{\sum_{i=1}^{T-k} K^2\left(\frac{Y_i-y}{h}\right)(Y_{t+k}-\hat{m}_k(y))^2}{\left(\sum_{t=1}^{T-k} K\left(\frac{Y_t-y}{h}\right)\right)^2}}} \Rightarrow N(0, 1),$$

其中  $\sigma_k^2(y) = \text{Var}(Y_{t+k} | Y_t = y)$ ，而  $f_Y(y)$  是平稳过程  $Y_t$  的边际密度。 $\hat{\mu} = \frac{1}{T} \sum_{t=1}^T Y_t$ 。且当  $y \neq y'$  时， $\hat{S}_k(y)$  和  $\hat{S}_l(y')$  是渐近独立的。

不难看出，定理 7.1 可以用来检验原假设（无预测能力）。如果  $\sum_{k=1}^K \sum_{l=1}^L \hat{S}_k(y_l)^2 > \chi_{KL}^2(\alpha)$ ，则拒绝零假设，其中  $\{y_1, \dots, y_L\}$  是  $Y_t$  支持集上的一组不同点的网格。

### 7.3.3 半参数波动率模型

在第 7.3.2 节中，我们将非参数统计方法引入到自回归 (AR) 模型中，类似地，我们也可以将非参数统计方法引入到波动率模型中。

假设收益率遵循一个 GARCH 模型，但误差分布是未知的，即：

$$y_t = \nu_t \sigma_t, \quad \sigma_t^2 = \omega + \beta \sigma_{t-1}^2 + \alpha y_{t-1}^2,$$

其中  $\nu_t$  是独立同分布且密度为  $f$ 。标准化残差  $\nu_t = y_t / \sigma_t$  并非正态分布。

在实际操作中，我们可以假设  $\nu_t$  服从学生 t 分布 (Student's t-distribution) 或其他厚尾的参数分布。然而，这种假设可能会影响参数估计的稳健性，因为与之相关的似然函数无法为感兴趣参数提供稳健的估计 (Newey & Steigerwald 1997)。相比之下，半参数方法允许同时估计  $\nu_t$  的分布以及 GARCH 过程的参数 (Engle & Gonzalez-Rivera 1991, Linton 1993, Drost et al. 1997)。

给定  $y_T, \dots, y_2$  和  $f$  的条件下，对数似然函数为：

$$\ell(\theta) = -\frac{1}{2} \sum_{t=1}^T \log \sigma_t^2(\theta) - \sum_{t=1}^T \log f\left(\frac{y_t}{\sigma_t(\theta)}\right),$$

得分函数 (score) 的形式为：

$$\frac{\partial \ell}{\partial \theta}(\theta) = -\frac{1}{2} \sum_{t=1}^T \left( \nu_t(\theta) \frac{f'}{f}(\nu(\theta)) + 1 \right) \frac{\partial \log \sigma_t^2(\theta)}{\partial \theta}.$$

估计步骤如下：

1. 使用如 QMLE 方法取得参数  $\theta$  的一致估计量，并计算标准化残差；
2. 基于标准化残差，采用非参数方法估计误差分布函数  $f$ ；

3. 使用估计得到的误差分布重新估计参数，进行两阶段极大似然估计。

其中第二步针对误差分布  $f$  的估计在本章第7.1节中有详尽的介绍。值得注意的是，为了确保模型的可识别性，不失一般性，我们需要设定  $\omega = 1$ ，或者假设  $E(\nu_t^2) = 1$ 。

此外，参数 GARCH (1,1) 模型的另一个问题在于其条件方差函数形式本身可能存在过度的局限性。我们可以进一步放宽假设，允许  $\sigma_t^2(\mathcal{F}_{t-1})$  采用任意函数形式：

$$\sigma_t^2 = g(y_{t-1}, \dots, y_{t-p}),$$

其中  $g$  为未知函数， $p$  为滞后阶数。此时，我们可以采用非参数回归方法对  $g$  进行估计。假设  $p = 1$ ，令：

$$\hat{g}(y) = \frac{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right) y_t^2}{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right)},$$

其中  $h$  是带宽， $K$  是核函数。这个估计量是对平方回报的局部加权平均。

我们也可以考虑在估计波动率时调整条件均值。这可以通过两种方式来实现。令：

$$\begin{aligned}\hat{g}_m(y) &= \frac{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right) y_t^2}{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right)} - \left( \frac{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right) y_t}{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right)} \right)^2, \\ \hat{g}_r(y) &= \frac{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right) \hat{u}_t^2}{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right)}, \quad \hat{u}_t = y_t - \frac{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right) y_t}{\sum_{t=2}^T K\left(\frac{y-y_{t-1}}{h}\right)}.\end{aligned}$$

## 7.4 章节总结

本章围绕非参数统计方法在金融计量经济学中的应用展开，系统讨论了非参数概率密度估计与非参数回归等内容。在概率密度估计方面，首先给出一维核密度估计的基本框架，分析其偏差、方差与边界问题，并据此讨论均方误差及最优带宽选择，为准确应用提供理论依据。随后将方法推广至多维情形，说明核密度估计在高维数据中的适用性与注意事项。在非参数回归部分，介绍了 Nadaraya-Watson 估计量，探讨其均方误差与最优带宽，并指出该方法在边界处的局限性；继而引入局部多项式估计量，说明其在减小边界偏差与提升拟合稳健性方面的优势。最后，本章概述了若干金融计量应用，包括游程检验、非线性可预测性检验、非参数自回归与半参数波动率模型等，展示了非参数方法在风险管理与资产定价中的实践价值，旨在帮助读者将理论工具与实际问题有效对接。

## 7.5 习题

1. 非参数统计方法的边界问题是什么？如何解决边界问题？
2. 假设  $\{X_t\}_{t=1}^T$  是一个独立同分布的随机样本，其边际密度函数  $g(x)$  在支撑区间  $[a, b]$  上二次连续可微。定义核密度估计量

$$\hat{g}(x) = \frac{1}{T} \sum_{t=1}^T K_h(x - X_t),$$

其中  $K_h(x - X_t) = h^{-1} K\left(\frac{x-X_t}{h}\right)$ ， $K(\cdot)$  为支撑集为  $[-1, 1]$  的正核函数， $h = h(T) \rightarrow 0$  为带宽。

- (a) 当  $x \in [a+h, b-h]$  时，推导  $E[\hat{g}(x)] - g(x)$  的渐近偏差表达式。

- (b) 当  $x \in [a+h, b-h]$  时, 推导  $\hat{g}(x)$  的渐近方差表达式, 即  $\text{Var}(\hat{g}(x)) = \mathbb{E}[(\hat{g}(x) - \mathbb{E}\hat{g}(x))^2]$ 。
- (c) 求均方误差 (MSE) 的渐近表达式  $\mathbb{E}[\hat{g}(x) - g(x)]^2$ 。
- (d) 推导使渐近 MSE 最小化的最优带宽  $h^*$ 。
- (e) 在最优带宽  $h^*$  处, 渐近均方误差 (MSE) 是什么?
3. 假设  $K(\cdot)$  为高阶 ( $q$  阶) 核函数, 满足

$$\int_{-1}^1 K(u) du = 1, \quad \int_{-1}^1 u^j K(u) du = 0 \quad (1 \leq j \leq q-1), \quad \int_{-1}^1 u^q K(u) du = C_K(q), \quad \int_{-1}^1 K^2(u) du = D_K,$$

并且假设  $g(x)$  在  $[a, b]$  上  $q$  次连续可微。请再次回答第 2 题中的 (a) 与 (e) 两问。

4. 假设有一组数据, 其中自变量  $X$  在区间  $[-3, 3]$  上均匀分布, 因变量  $Y$  由

$$Y = X^3 + \varepsilon,$$

生成, 其中  $\varepsilon$  为正态噪声, 标准差为 0.3。请编写 R 代码, 实现以下功能:

- (a) 用 R 模拟生成  $X$  与  $Y$  的数据, 其中  $X$  服从均匀分布,  $Y$  按  $Y = X^3 + \varepsilon$  生成;
- (b) 使用 Nadaraya–Watson 估计量与局部多项式估计量拟合数据;
- (c) 绘制实际数据点、Nadaraya–Watson 拟合曲线与局部多项式拟合曲线, 比较拟合效果;
- (d) 计算并绘制各方法的残差图, 分析残差分布特征, 评估拟合效果;
- (e) 进行多次实验, 研究非参数回归在不同样本量与噪声水平下的表现。
5. 下载最近一年的上证指数数据, 使用第 7.3.3 节提出的半参数 GARCH (1,1) 模型估计其波动率, 并与传统 GARCH (1,1) 模型的估计结果进行比较; 讨论非参数模型在捕捉波动率特性 (如波动聚集性) 方面的表现。

## 8 金融资产定价模型

在第 2.4 节中，我们系统地介绍了一系列重要的金融学理论模型，其中包括资本资产定价模型 (CAPM)、套利定价理论 (APT) 以及基于消费的资本资产定价模型 (C-CAPM)。这些经典模型构成了现代金融学的理论基石，在金融学术研究和实际投资决策中都发挥着不可替代的作用。CAPM 通过对风险与收益关系的精确刻画，为投资者评估资产的预期回报提供了简洁而有力的分析框架；APT 则从多因素角度出发，拓展了对资产定价的理解，使我们能够更全面地考虑影响资产价格的多种潜在因素；C-CAPM 则将消费行为与资产定价相联系，为宏观经济与金融市场的互动研究提供关键的理论支撑。

在此基础上，我们将这些理论模型转化为金融计量经济模型。通过计量经济模型的构建，我们能够利用实际市场数据对理论模型进行量化估计，在检验理论有效性的同时，也能更准确地把握市场规律。这不仅有助于学术界深入理解金融市场的运行机制，也为金融从业者在资产定价、投资组合管理等实际操作中提供科学的方法与工具。

一般来讲，将理论模型转换为计量经济学模型通常涉及以下几个步骤：

1. 模型规范化：首先，需要将理论模型中的关键概念和关系明确地表达为数学方程。例如，在 CAPM 模型中，这涉及资产收益、市场收益、无风险利率等要素的数学表示。随后需赋予理论模型计量经济学结构。这一步非常重要。从理论上看，CAPM 是单期模型，不含时间维度，因此需将其拓展至时间维度。对于单一资产，这意味着：

$$Z_{it} = \alpha_i + \beta_i Z_{mt} + \varepsilon_{it}, \quad (8.1)$$

其中， $i$  表示资产编号， $t$  表示时间 ( $t = 1, \dots, T$ )。 $Z_{it}$  和  $Z_{mt}$  分别表示在时间  $t$  的资产  $i$  与市场组合的超额收益率。

在进行计量经济学分析时，通常需对收益的时间序列行为作出假设：一般假设资产收益率在时间上独立同分布 (i.i.d.)；多元模型可能进一步假设其服从联合多元正态分布。当然，也可引入非参数方法以放松对收益分布的限制。

2. 数据收集与处理：收集与模型相关的数据以开展实证检验，数据可能包括股票价格、市场指数、利率等。在此过程中需进行数据清洗与整理，以确保数据质量与适用性。
3. 参数估计：使用统计方法（如最小二乘、极大似然等）对模型参数进行估计。
4. 模型检验：使用各类统计检验方法验证模型的有效性，考察模型对数据的拟合程度以及各参数的显著性。
5. 模型应用：在验证模型有效性后，可据此进行分析或预测未来趋势。

## 8.1 资本资产定价模型 (CAPM)

本节主要侧重于夏普-林特纳版本的 CAPM 模型的估计与检验；关于布莱克版本的 CAPM 模型，详见 Campbell et al. (1997) 第 5.4 节。

### 8.1.1 模型估计

令  $Z_t$  为  $(N \times 1)$  的超额收益向量，代表  $N$  种资产的超额收益率。针对  $N$  个资产的 CAPM 模型为：

$$\begin{aligned} Z_t &= \alpha + \beta Z_{mt} + \varepsilon_t, \\ E[\varepsilon_t] &= 0, \quad E[\varepsilon_t \varepsilon'_t] = \Sigma, \\ E[Z_{mt}] &= \mu_m, \quad E[(Z_{mt} - \mu_m)^2] = \sigma_m^2, \quad \text{Cov}[Z_{mt}, \varepsilon_t] = 0. \end{aligned}$$

其中， $\beta$  为  $(N \times 1)$  的贝塔向量， $Z_{mt}$  表示时点  $t$  的市场组合超额收益， $\alpha$  为截距向量， $\varepsilon_t$  为扰动项向量。此处  $\mu_m$  为市场超额收益的期望；若记各资产期望超额收益向量为  $\boldsymbol{\mu} = E[Z_t]$ ，则有  $\boldsymbol{\mu} = \alpha + \beta \mu_m$ 。

上述模型可采用极大似然方法进行估计，详见 Campbell et al. (1997) 第 5 章第 3 节，此处不再赘述。极大似然估计量如下：

$$\hat{\alpha} = \hat{\mu} - \hat{\beta} \hat{\mu}_m, \quad \hat{\beta} = \frac{\sum_{t=1}^T (Z_t - \hat{\mu})(Z_{mt} - \hat{\mu}_m)}{\sum_{t=1}^T (Z_{mt} - \hat{\mu}_m)^2}, \quad \hat{\Sigma} = \frac{1}{T} \sum_{t=1}^T (Z_t - \hat{\alpha} - \hat{\beta} Z_{mt}) (Z_t - \hat{\alpha} - \hat{\beta} Z_{mt})'$$

其中

$$\hat{\mu} = \frac{1}{T} \sum_{t=1}^T Z_t, \quad \hat{\mu}_m = \frac{1}{T} \sum_{t=1}^T Z_{mt}.$$

极大似然估计量的分布为：

$$\hat{\alpha} \sim N\left(\alpha, \frac{1}{T} \left[1 + \frac{\hat{\mu}_m^2}{\hat{\sigma}_m^2}\right] \Sigma\right), \quad \hat{\beta} \sim N\left(\beta, \frac{1}{T} \left[\frac{1}{\hat{\sigma}_m^2}\right] \Sigma\right),$$

其中

$$\hat{\sigma}_m^2 = \frac{1}{T} \sum_{t=1}^T (Z_{mt} - \hat{\mu}_m)^2.$$

$T\hat{\Sigma} \sim \mathcal{W}_N(T-2, \Sigma)$  表明  $(N \times N)$  矩阵  $T\hat{\Sigma}$  服从自由度为  $(T-2)$ 、协方差矩阵为  $\Sigma$  的威沙特 (Wishart) 分布。这个分布是卡方分布的多变量推广。 $\hat{\alpha}$  和  $\hat{\beta}$  的协方差为：

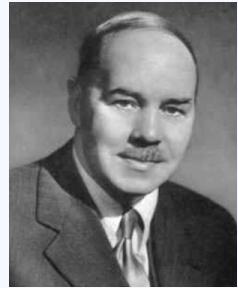
$$\text{Cov}[\hat{\alpha}, \hat{\beta}'] = -\frac{1}{T} \left[ \frac{\hat{\mu}_m}{\hat{\Sigma}_m^2} \right] \Sigma.$$

其中  $\hat{\Sigma}$ 、 $\hat{\alpha}$  以及  $\hat{\beta}$  都是互相独立的。

**威沙特分布**是以统计学家约翰·威沙特 (John Wishart) 的名字命名的，是统计学中用于描述半正定矩阵随机分布的一种概率分布。它在多变量统计分析中，尤其是在协方差矩阵估计方面，具有重要意义。

假设  $X$  是一个  $n \times p$  矩阵，其每一行  $X_{(i)}$  都独立地服从均值向量为  $\mathbf{0}$ 、协方差矩阵为  $\mathbf{V}$  的  $p$  维多元正态分布，即

$$X_{(i)} = (x_i^1, \dots, x_i^p)' \sim N(\mathbf{0}, \mathbf{V}),$$



则  $X$  的转置  $X'$  与  $X$  的乘积

$$\mathbf{S} = X'X = \sum_{i=1}^n X_{(i)}X'_{(i)},$$

遵循一个  $p \times p$  的威沙特分布。这个分布通常表示为

$$\mathbf{S} \sim W_p(\mathbf{V}, n),$$

其中  $n$  为自由度。当  $p = 1$  且  $\mathbf{V} = 1$  时，威沙特分布退化为自由度为  $n$  的卡方分布，即  $\chi_n^2$ 。

威沙特分布的参数包括：自由度  $n$ (正实数) 和尺度矩阵  $\mathbf{V}$ (正定矩阵)。当  $n \geq p + 1$  时，其期望值为  $n\mathbf{V}$ ，而众数为  $(n - p - 1)\mathbf{V}$ 。威沙特分布的特征函数为：

$$\Theta \mapsto |I - 2i\Theta\mathbf{V}|^{-n/2},$$

其中  $I$  为单位矩阵， $\Theta$  为参数矩阵。

## 8.1.2 模型检验

### 8.1.2.1 沃尔德 (Wald) 检验

我们通过检验截距项 (超额收益率) 是否为零来判断 CAPM 模型的正确性。换而言之，如果 CAPM 模型是正确的，那么超额收益率的期望值应为零。

我们构建以下 Wald 假设检验。原假设为

$$H_0 : \boldsymbol{\alpha} = \mathbf{0},$$

备择假设为

$$H_1 : \boldsymbol{\alpha} \neq \mathbf{0}.$$

Wald 检验统计量为

$$J_0 = \hat{\boldsymbol{\alpha}}' [\text{Var}(\hat{\boldsymbol{\alpha}})]^{-1} \hat{\boldsymbol{\alpha}} = T \left[ 1 + \frac{\hat{\mu}_m^2}{\hat{\sigma}_m^2} \right]^{-1} \hat{\boldsymbol{\alpha}}' \hat{\Sigma}^{-1} \hat{\boldsymbol{\alpha}}.$$

在原假设下， $J_0$  服从自由度为  $N$  的卡方分布。由于  $\Sigma$  未知，我们一般采用其一个极大似然估计量  $\hat{\Sigma}$  (一致估计量) 进行替代。

值得注意的是， $J_0$  是基于大样本设计的。依照 Campbell et al. (1997)，对于小样本情

形, 建议使用下面的统计量:

$$J_1 = \frac{T-N-1}{N} \left[ 1 + \frac{\hat{\mu}_m^2}{\hat{\sigma}_m^2} \right]^{-1} \hat{\alpha}' \hat{\Sigma}^{-1} \hat{\alpha}.$$

在原假设下,  $J_1$  服从自由度为  $N$  与  $T-N-1$  的  $F(N, T-N-1)$  分布。

### 8.1.2.2 似然比检验

为了进行似然比检验 (likelihood ratio test, 简称 LRT), 需要在约束模型下获得参数估计值。约束模型令  $\boldsymbol{\alpha} = \mathbf{0}$ , 即

$$\mathbf{Z}_t = \beta Z_{mt} + \varepsilon_t.$$

约束模型的极大似然估计量为

$$\hat{\boldsymbol{\beta}}^* = \frac{\sum_{t=1}^T \mathbf{Z}_t Z_{mt}}{\sum_{t=1}^T Z_{mt}^2}, \quad \hat{\Sigma}^* = \frac{1}{T} \sum_{t=1}^T (\mathbf{Z}_t - \hat{\boldsymbol{\beta}}^* Z_{mt}) (\mathbf{Z}_t - \hat{\boldsymbol{\beta}}^* Z_{mt})'$$

在原假设 ( $\boldsymbol{\alpha} = \mathbf{0}$ ) 下, 极大似然估计量的渐近分布为

$$\hat{\boldsymbol{\beta}}^* \sim N\left(\boldsymbol{\beta}, \frac{1}{T} \frac{1}{\hat{\mu}_m^2 + \hat{\sigma}_m^2} \Sigma\right), \quad T \hat{\Sigma}^* \sim \mathcal{W}_N(T-1, \Sigma).$$

令似然比为

$$\Lambda \equiv \frac{L^*}{L}, \quad \ln \Lambda = \mathcal{L}^* - \mathcal{L} = -\frac{T}{2} [\ln |\hat{\Sigma}| - \ln |\hat{\Sigma}^*|].$$

据此, 似然比检验统计量为

$$J_2 = -2 \ln \Lambda = T [\ln |\hat{\Sigma}^*| - \ln |\hat{\Sigma}|] \underset{a}{\sim} \chi_N^2.$$

其中  $\hat{\Sigma}^*$  与  $\hat{\Sigma}$  分别是约束与非约束模型下的协方差矩阵估计量,  $T$  为样本量,  $\chi_N^2$  表示自由度为  $N$  的卡方分布 (此处  $N$  为原假设下的约束数量, 即  $N$  个资产的超额收益率均为零)。

由于 Campbell et al. (1997) 对  $J_2$  的推导已有详细论述, 这里不再赘述。Campbell et al. (1997) 指出约束模型与非约束模型的估计量之间存在如下关系:

$$\hat{\boldsymbol{\beta}}^* = \hat{\boldsymbol{\beta}} + \frac{\hat{\mu}_m}{\hat{\mu}_m^2 + \hat{\sigma}_m^2} \hat{\alpha},$$

以及

$$\begin{aligned} \hat{\Sigma}^* &= \frac{1}{T} \sum_{t=1}^T (Z_t - \hat{\boldsymbol{\beta}}^* Z_{mt}) (Z_t - \hat{\boldsymbol{\beta}}^* Z_{mt})' \\ &= \frac{1}{T} \sum_{t=1}^T \left[ (Z_t - \hat{\alpha} - \hat{\beta} Z_{mt}) + \left(1 - \frac{\hat{\mu}_m Z_{mt}}{\hat{\mu}_m^2 + \hat{\sigma}_m^2}\right) \hat{\alpha} \right] \\ &\quad \times \left[ (Z_t - \hat{\alpha} - \hat{\beta} Z_{mt}) + \left(1 - \frac{\hat{\mu}_m Z_{mt}}{\hat{\mu}_m^2 + \hat{\sigma}_m^2}\right) \hat{\alpha} \right]', \end{aligned}$$

进一步可得:

$$\hat{\Sigma}^* = \hat{\Sigma} + \left( \frac{\hat{\sigma}_m^2}{\hat{\mu}_m^2 + \hat{\sigma}_m^2} \right) \hat{\alpha} \hat{\alpha}',$$

即：

$$|\widehat{\Sigma}^*| = |\widehat{\Sigma}| \left[ \left( \frac{\widehat{\sigma}_m^2}{\widehat{\mu}_m^2 + \widehat{\sigma}_m^2} \right) \widehat{\alpha}' \widehat{\Sigma}^{-1} \widehat{\alpha} + 1 \right].$$

因此，似然比可以用非限制模型的估计量表达如下：

$$\mathcal{LR} = -\frac{T}{2} \log \left[ \left( \frac{\widehat{\sigma}_m^2}{\widehat{\mu}_m^2 + \widehat{\sigma}_m^2} \right) \widehat{\alpha}' \widehat{\Sigma}^{-1} \widehat{\alpha} + 1 \right],$$

进一步整理可得似然比检验统计量：

$$J_1 = -2\mathcal{LR} = \frac{(T-N-1)}{N} \left( \exp \left[ \frac{J_2}{T} \right] - 1 \right).$$

与 Wald 检验的情形类似， $J_1$  适用于大样本的情形，在有限样本的情况下建议采用

$$J_3 = \frac{(T-\frac{N}{2}-2)}{T} J_2 = \left( T - \frac{N}{2} - 2 \right) \left[ \log |\widehat{\Sigma}^*| - \log |\widehat{\Sigma}| \right] \implies \chi_N^2,$$

换而言之，当样本量  $T$  趋向于无穷大时， $J_3$  的分布会趋近于自由度为  $N$  的卡方分布  $\chi_N^2$ 。

### 沃尔德 (Wald) — 似然比 (LR) — 拉格朗日乘数 (LM/Score) 检验的比较

在检验线性模型约束是否成立时，除似然比 (LR) 检验与 Wald 检验外，常用的还有 Lagrange 乘数 (LM, 亦称得分/Score) 检验。三者在大样本下等价，但思路与计算量不同：Wald 检验基于非受限估计量的“水平距离”，即比较参数估计值与原假设约束值之间的偏离，并用估计方差进行标准化；LR 检验基于非受限模型与受限模型最大对数似然值的差，衡量“纵向高度差”；LM/Score 检验仅需估计受限模型，利用受限点处的得分向量与信息矩阵检验“切线斜率是否为零”。为避免符号歧义，本文约定： $\theta$  表示标量参数， $\widehat{\theta}$  为非受限极大似然估计（最大化  $\mathcal{L}(\theta)$  所得）， $\theta_0$  为原假设（受限）下的参数取值， $\mathcal{L}(\theta)$  为对数似然函数。由此，LR 的统计量可写作  $2[\mathcal{L}(\widehat{\theta}) - \mathcal{L}(\theta_0)]$ ；Wald 的统计量为  $(\widehat{\theta} - \theta_0)^2 / \widehat{\text{Var}}(\widehat{\theta})$ ；LM/Score 的统计量为  $[s(\theta_0)]^2 / \mathcal{I}(\theta_0)$ ，其中  $s(\theta) = \partial \mathcal{L}(\theta) / \partial \theta|_{\theta=\theta_0}$ ， $\mathcal{I}(\theta_0)$  为（期望或观测）信息矩阵。图 8.1 直观展示了“水平距离—纵向增益—切线斜率”的几何对应关系；更系统的示意与讨论见 Fox (1997, p. 570)。

上述记号也可自然推广到参数向量情形：令  $\boldsymbol{\theta} \in \mathbb{R}^q$ ，则 Wald 由“平方差”推广为二次型  $(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0)' \widehat{\text{Var}}(\widehat{\boldsymbol{\theta}})^{-1} (\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0)$ ，LR 仍为  $2[\mathcal{L}(\widehat{\boldsymbol{\theta}}) - \mathcal{L}(\boldsymbol{\theta}_0)]$ ，LM/Score 则为  $\mathbf{s}(\boldsymbol{\theta}_0)' \mathcal{I}(\boldsymbol{\theta}_0)^{-1} \mathbf{s}(\boldsymbol{\theta}_0)$ ，其中  $\mathbf{s}(\boldsymbol{\theta}_0) = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0}$ 。在原假设下，上述统计量（在常规正则条件下）渐近服从自由度为  $r$  的  $\chi_r^2$  分布；其中  $r$  为原假设中的独立约束数量。在计算量上，Wald 与 LR 需做受限与非受限两套估计，LM/Score 仅需受限一次估计，在复杂约束场景下更为经济。

### 8.1.3 时变 CAPM

时变参数的资本资产定价模型 (time-varying CAPM) 定义为

$$Z_{it} = \alpha_i(\mathcal{F}_{t-1}) + \beta_i(\mathcal{F}_{t-1})' Z_{mt} + \varepsilon_{it}, \quad i = 1, \dots, n, \quad t = 1, \dots, T,$$

其中  $E(\varepsilon_{it} | \mathcal{F}_{t-1}) = 0$ ； $Z_{it}$  表示资产  $i$  在时点  $t$  的超额收益率， $\alpha_i(\mathcal{F}_{t-1})$  与  $\beta_i(\mathcal{F}_{t-1})$  为未知函数系数； $Z_{mt}$  为市场投资组合在时点  $t$  的超额收益率； $\mathcal{F}_t$  为时点  $t$  可得的信息集。

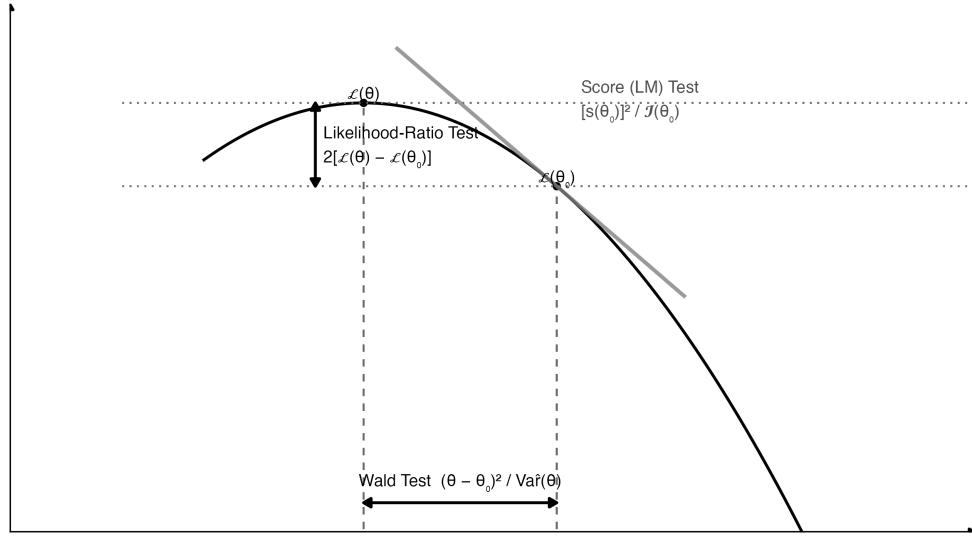


图 8.1: Wald 检验、似然比 (LR) 检验与 Lagrange 乘数 (LM/Score) 检验的关系

在讨论上述模型时，研究者常关注因子载荷是否随时间变化。由欧拉方程出发，可将

$$\alpha_{it} \equiv \alpha_i(\mathcal{F}_{t-1}), \quad \beta_{it} \equiv \beta_i(\mathcal{F}_{t-1})$$

视为潜在的时变系数。进一步设

$$\alpha_{it} = \alpha_i(S_t), \quad \beta_{it} = \beta_i(S_t),$$

其中  $S_t$  为  $\mathcal{F}_{t-1}$  中给定的状态变量向量，但具体函数形式  $\alpha_i(\cdot)$  与  $\beta_i(\cdot)$  未知。可通过最小化下述局部残差平方和进行估计：

$$\min_{\{\alpha_i(\cdot), \beta_i(\cdot)\}_{i=1}^n} \sum_{i=1}^n \sum_{t=1}^T \left[ Z_{it} - \alpha_i(S_t) - \beta_i(S_t)' Z_{mt} \right]^2 K_h(S_t - z).$$

#### 8.1.4 条件资本资产定价模型

假设有多种风险资产  $i = 1, \dots, n$  以及一种无风险资产。定义  $Z_t = (Z_{1t}, \dots, Z_{nt})'$  为时点  $t$  的风险资产超额收益向量，即各资产收益率减去无风险资产收益率得到的差值向量； $Z_{mt}$  表示市场投资组合在时点  $t$  的超额收益。市场投资组合是所有可交易资产的加权组合，其表达式为

$$Z_{mt} = Z_t' W_t,$$

其中  $W_t = (W_{1t}, \dots, W_{nt})'$  为依据资本资产定价模型 (CAPM) 均衡确定的权重向量。权重可随信息集  $\mathcal{F}_{t-1}$  的更新而调整。

设

$$\mu_t = (\mu_{1t}, \dots, \mu_{nt})',$$

其中  $\mu_{it} = E(Z_{it} | \mathcal{F}_{t-1})$  表示在给定  $\mathcal{F}_{t-1}$  的条件下，资产  $i$  的预期超额收益。进一步定义

$$H_t = E[(Z_t - \mu_t)(Z_t - \mu_t)' | \mathcal{F}_{t-1}],$$

为给定  $\mathcal{F}_{t-1}$  时  $Z_t$  的方差—协方差矩阵。由此可得

$$\mu_{mt} = \mathbb{E}(Z_{mt} | \mathcal{F}_{t-1}) = \mu'_t W_t, \quad \text{Var}(Z_{mt} | \mathcal{F}_{t-1}) = W_t' H_t W_t.$$

因此，条件 CAPM 模型为

$$\mu_t = \beta_t \mu_{mt} = \frac{\mu_{mt}}{W_t' H_t W_t} H_t W_t,$$

其中市场贝塔系数

$$\beta_t = \frac{H_t W_t}{W_t' H_t W_t}.$$

该模型可对应为下面的回归形式：

$$Z_t = \beta_t Z_{mt} + \varepsilon_t,$$

其中

$$\mathbb{E}(\varepsilon_t | \mathcal{F}_{t-1}) = 0, \quad \text{Var}(\varepsilon_t | \mathcal{F}_{t-1}) = H_t,$$

即风险系数  $\beta_t$  随时间变化，并依赖于单个资产收益的条件方差—协方差矩阵  $H_t$ 。

## 8.2 多因子模型

多因子定价模型的理论基础主要有两类：套利定价理论（Arbitrage Pricing Theory, 简称 APT；(Ross 1976)）和跨期资本资产定价模型（Intertemporal Capital Asset Pricing Model, 简称 ICAPM；(Merton 1973)）。套利定价理论基于套利论证，认为资产的预期收益可以由若干宏观因子或市场指标解释：若某资产价格偏离由这些因子所蕴含的均衡水平，即存在套利机会，从而推动价格回到均衡。跨期资本资产定价模型则基于均衡论证，强调投资者在进行投资决策时不仅权衡当期风险与收益，还会考虑未来投资机会集的变化，从而在多个期次上优化投资组合，因而可视作对传统 CAPM 的扩展。

但是，学界与教材更普遍接受“无套利”作为多因子模型的主要理论基石，对 Merton (1973) 的讨论相对较少。此外，APT 在金融实务（尤其投资银行的资产定价）中作用突出。与均衡定价模型（如 CAPM、C-CAPM）不同，APT 更侧重在已知部分资产价格的前提下对其他相关资产进行定价，因而亦称“相对定价理论”。

在现代金融理论不断发展的进程中，资本资产定价模型（CAPM）和套利定价理论（APT）作为重要的理论基石，为理解资产收益与风险的关系提供了关键视角。不难看出，CAPM 基于市场组合的单因子框架，认为资产的预期收益主要由市场风险（即系统性风险、不可分散风险）决定，其核心度量为贝塔系数（Beta），用于衡量个别证券或投资组合相对于市场的风险。而 APT 则对应多因子模型，认为资产收益不仅受市场风险影响，还受到多种宏观经济因素的共同驱动；因此其对应的实证设定通常纳入通货膨胀率、利率、工业产出等多个风险因子。显然，APT 比 CAPM 更具一般性，因为它允许存在多个风险源；不同于 CAPM，APT 亦不要求识别市场组合。

回溯中国传统文化，其中蕴含着与现代金融理论相契合的智慧。《周易》提出“方以类聚，物以群分”的思想，体现了深刻的分类理念，与现代金融中的因子模型异曲同工。在金融市场中，不同板块（如大盘股、小盘股）及不同行业的股票，往往呈现出各自的收益特征与风险属性；正如 APT 多因子模型中对通货膨胀率、利率等风险因素的分类考量，板块与行业的划分本质上也是对影响收益的共性因子的归纳。这种分类有助于投资者更精准地开

展组合分析与风险管理，在控制风险的同时追求收益最大化。

根据 Ross (1976), APT 意味着

$$\mu \approx \iota \lambda_0 + \mathbf{B} \lambda_K$$

其中， $\mu$  为  $(N \times 1)$  维预期回报向量， $\lambda_0$  为零贝塔参数（若存在无风险资产，则等于无风险回报）， $\lambda_K$  为  $(K \times 1)$  维因子风险溢价向量， $\iota$  为每个元素均为 1 的  $(N \times 1)$  维向量。显见，由于 Ross (1976) 中的 APT 为近似关系，不能直接用于资产定价。在实际操作中，一般近似误差可忽略不计，即

$$\mu = \iota \lambda_0 + \mathbf{B} \lambda_K.$$

关于 APT 的文献，参见 Campbell et al. (1997) 中第 6.1 节中的文献综述。

接着，我们将理论模型转换为计量经济学模型。 $K$  因子回归模型（用于回报或超额回报）可以写为：

$$Z_{it} = \alpha_i + \sum_{j=1}^K b_{ij} f_{jt} + \varepsilon_{it}, \quad i = 1, \dots, N, \quad t = 1, \dots, T;$$

上式也可以用矩阵形式表示

$$Z_t = \alpha + B Z_{Kt} + \varepsilon_t,$$

其中  $Z_t$  是  $N \times 1$  的回报或超额回报向量（当无风险利率  $R_{ft}$  已知时）， $Z_{Kt}$  是一个  $K \times 1$  的因子超额回报向量， $\varepsilon_t = (\varepsilon_{1t}, \dots, \varepsilon_{Nt})'$  是误差项向量，这些误差项随时间独立并满足：

$$E(\varepsilon_t | Z_{K1}, \dots, Z_{KT}) = 0, \quad E(\varepsilon_t \varepsilon_t' | Z_{K1}, \dots, Z_{KT}) = \Sigma_\varepsilon.$$

条件期望  $E(\varepsilon_t | Z_{K1}, \dots, Z_{KT}) = 0$  表明，在给定所有时期的因子超额回报  $Z_{K1}, \dots, Z_{KT}$  的条件下，误差项的期望为 0。这意味着从平均意义上，在考虑了这些因子的影响后，剩余的随机因素不会系统性地对资产回报产生正向或负向的影响。条件协方差  $E(\varepsilon_t \varepsilon_t' | Z_{K1}, \dots, Z_{KT}) = \Sigma_\varepsilon$  说明，在给定所有时期的因子超额回报的条件下，误差项向量的协方差矩阵是一个常数矩阵  $\Sigma_\varepsilon$ 。这体现了误差项之间的相关性结构在不同时间点是相对稳定的，不会随着因子的变化而发生改变。

有时我们引入误差项的 i.i.d. 假设： $\varepsilon_t \sim \text{i.i.d. } N(0, \Sigma_\varepsilon)$ 。该假设简化了模型的统计推断过程。在正态独立同分布的假设下，我们可以运用许多经典的统计方法来估计模型参数、进行假设检验和预测等，这无疑提高了模型的实用性和可操作性。

因子模型中的因子可以涵盖多种多种类型，以下列举了一些常见的因子类别：

### 1. 宏观经济因子：

[label=] 利率因子：利率变动对各类资产价格与贴现率产生广泛影响。通货膨胀因子：影响资产的实际收益率。经济增长因子：通常以 GDP 增长率衡量，影响企业盈利与市场风险偏好。

### (2) 市场因子：

[label=] 市场组合收益率：如沪深 300 指数、标普 500 指数等的收益率，反映市场整体走势；亦可采用同一行业构建的投资组合收益率以反映行业信息。市场风险溢价因子：投资者为承担市场风险所要求的额外回报（市场组合收益率减无风险利率）。

**(B) 公司特征因子:**

[label=)] 规模因子: 通常以市值分组构建“小盘减大盘 (SMB)”组合的收益差来表征; 历史上小盘股平均收益往往较高。价值因子: 以账面市值比 (B/M)、市净率等指标分组构建的“高减低 (HML)”组合收益, 刻画被低估股票的相对收益。盈利能力因子: 例如净资产收益率 (ROE)、总资产收益率 (ROA) 等指标所反映的盈利特征。

**(H) 统计因子:**

[label=)] 主成分因子: 通过主成分分析从资产收益率中提取, 解释共同波动。特征因子: 基于价格序列的波动性特征、相关性等统计性质提取的因子。

多因子模型的估计与第8.1.1节的方法相同, 这里不再赘述。后续我们将侧重多因子模型的检验。

### 8.2.1 以投资组合收益为因子的多因子定价检验

套利定价理论 (APT) 的检验与资本资产定价模型 (CAPM) 的检验非常相似, 但向量  $\beta$  被  $N \times K$  矩阵  $B$  所取代。

我们首先假设存在无风险资产, 因子收益  $F$  是可观测的, 这样我们就得到了一个多元高斯似不相关回归 (seemingly unrelated regression equations, 简称 SURE) 模型。

似不相关回归 (SURE) 模型中, 虽然每个方程单独来看都是普通的线性回归方程, 但不同方程的误差项之间可能存在相关性。例如, 在研究多个不同行业的企业业绩与若干宏观经济变量的关系时, 企业业绩方程的误差项可能因共同的宏观经济因素而相关。似不相关回归模型能够显式考虑这种误差项相关性, 从而提高估计的效率与准确性。

在给定  $Z_{K1}, \dots, Z_{KT}$  的条件下, 数据  $Z_1, \dots, Z_T$  (其中  $Z_{it} = R_{it} - R_{ft}$ ) 的对数似然函数为:

$$\ell(\alpha, B, \Sigma_\varepsilon) = c - \frac{T}{2} \log \det(\Sigma_\varepsilon) - \frac{1}{2} \sum_{t=1}^T (Z_t - \alpha - BZ_{Kt})' \Sigma_\varepsilon^{-1} (Z_t - \alpha - BZ_{Kt}). \quad (8.2)$$

令:

$$\begin{aligned} \hat{\mu}_K &= \frac{1}{T} \sum_{t=1}^T Z_{Kt}, & \hat{\mu} &= \frac{1}{T} \sum_{t=1}^T Z_t, \\ \hat{\Sigma}_K &= \frac{1}{T} \sum_{t=1}^T (Z_{Kt} - \hat{\mu}_K)' (Z_{Kt} - \hat{\mu}_K), & \hat{\Sigma}_{ZK} &= \frac{1}{T} \sum_{t=1}^T (Z_t - \hat{\mu})' (Z_{Kt} - \hat{\mu}_K). \end{aligned}$$

$\alpha$ 、 $B$  的无约束极大似然估计 (MLE) 是逐个方程的普通最小二乘 (OLS) 估计量:

$$\hat{\alpha} = \hat{\mu} - \hat{B}\hat{\mu}_K, \quad \hat{B} = \hat{\Sigma}_{ZK} \hat{\Sigma}_K^{-1}, \quad \hat{\Sigma}_\varepsilon = \frac{1}{T} \sum_{t=1}^T \hat{\varepsilon} \hat{\varepsilon}'_t. \quad (8.3)$$

其中  $\hat{\varepsilon} = Z_t - \hat{\alpha} - \hat{B}Z_{Kt}$ , 前提是  $\hat{\Sigma}_K$  的估计量存在且满秩 ( $K < T$ )。在正态分布假设下, 在给定因子的条件下, 我们有:

$$\hat{\alpha} \sim N \left( \alpha, \frac{1}{T} \left( 1 + \hat{\mu}'_K \hat{\Sigma}_K^{-1} \hat{\mu}_K \right) \Sigma_\varepsilon \right). \quad (8.4)$$

与 CAPM 的情形类似，如果模型是正确设置的，则  $\alpha = 0$ 。针对  $\alpha = 0$  的原假设， $F$  检验为：

$$F = \frac{(T - N - K)}{N} \left( 1 + \hat{\mu}'_K \hat{\Sigma}_K^{-1} \hat{\mu}_K \right)^{-1} \hat{\alpha}' \hat{\Sigma}_\varepsilon^{-1} \hat{\alpha}. \quad (8.5)$$

只要误差项  $\varepsilon_t$  服从正态分布假设，且  $\hat{\Sigma}_\varepsilon$  满秩（要求  $N < T$ ），该统计量就精确服从  $F_{N,T-N-K}$  分布。在非正态情况下，在某些条件下， $N \times F$  渐近地服从自由度为  $N$  的卡方分布，沃尔德 (Wald) 统计量、拉格朗日乘数 (Lagrange multiplier) 统计量和似然比 (likelihood ratio) 统计量也是如此。

我们也可以逐个方程地收集参数，即令  $\Theta = (\theta_1, \dots, \theta_N)'$  为  $N \times (K + 1)$  矩阵，其中  $\theta_i = (\alpha_i, b'_i)'$ ， $b_i$  是  $B$  的第  $i$  行，令  $X$  为  $T \times (K + 1)$  矩阵，包含一列全为 1 的列向量以及  $Z_{kt}$  ( $k = 1, \dots, K$ ) 的观测值列。我们可以将  $\Theta$  的估计量简写为：

$$\hat{\Theta} = Z' X (X' X)^{-1}, \quad (8.6)$$

上式与 OLS 估计量的表达式类似。

除了对  $\alpha = 0$  进行检验之外，我们还可以对  $B_2 = 0$  进行检验，其中  $B = (B_1, B_2)$ ，即  $B_2$  对应的某些因子对收益率没有影响。

类似地，我们可以构建似然比统计量：

$$J_1 = T \left( \log |\tilde{\Sigma}_\varepsilon| - \log |\hat{\Sigma}_\varepsilon| \right), \quad (8.7)$$

其中  $\tilde{\Sigma}_\varepsilon$  是受限模型 ( $B_2 = 0$ ) 中的估计误差协方差矩阵，其定义与公式(8.3)相同，但仅包含与  $B_1$  对应的因子子集。当样本量  $T$  较大时，该统计量的分布近似为自由度为  $K_2$  的卡方分布，且在非正态条件下该结果依然成立。

### 8.2.2 含宏观因子和无风险利率的多因子模型的检验

我们考虑存在无风险利率的情形，其中一些因子是可交易投资组合 ( $(f_{1t} \in \mathbb{R}^{K_1})$ )，而其他因子是不可交易的宏观因子 ( $(f_{2t} \in \mathbb{R}^{K_2})$ )。在这种情况下，宏观因子的预期收益率不受原假设的限制。

在这种情形下，无约束模型为：

$$\begin{aligned} R_t - R_{ft} &= \alpha + B_1 (f_{1t} - R_{ft}) + B_2 f_{2t} + \varepsilon_t, \\ E(f_{2t}) &= \mu_{K2}. \end{aligned}$$

在此情形下，与套利定价理论 (APT) 一致的原假设 ( $H_0$ ) 是：对于某个未知的  $\gamma_2 \in \mathbb{R}^{K_2}$ ，有  $\alpha = B_2 \gamma_2$ 。假设  $B_2$  满秩，我们可以通过  $\gamma_2 = (B'_2 B_2)^{-1} B_2 \alpha$  求出  $\gamma_2$ 。由此可知，约束条件可以重写为  $M_{B_2} \alpha = 0$ ，其中  $M_{B_2} = I_{K_2} - B_2 (B'_2 B_2)^{-1} B'_2$   $N \times N$  矩阵  $M_{B_2}$  是对称且幂等的，秩为  $N - K_2$ 。将  $\alpha$  代入收益率方程可得：

$$R_t - R_{ft} = B_1 (f_{1t} - R_{ft}) + B_2 (f_{2t} + \gamma_2) + \varepsilon_t.$$

在给定  $\gamma_2$  时，上式关于  $B$  是线性的（在给定  $B_2$  时，关于  $\gamma_2$  是线性的）。可以先设定一个  $\gamma_2$ ，求出  $\tilde{B}(\gamma_2)$ ，然后对参数  $\gamma_2$  进行优化，从而估计受限模型。设  $\tilde{B}$ 、 $\tilde{\gamma}_2$ 、 $\tilde{\Omega}_\varepsilon$  为受限极大似然估计值。检验该假设的最简单方法是使用似然比统计量，该统计量渐近服从自由度为  $N - K_2$  的卡方分布。

接下来，我们考虑沃尔德检验，或者说是一种无需明确使用约束条件  $M_{B_2}\alpha = 0$ （由于秩降低，该约束条件存在问题）的沃尔德类检验。设  $\hat{\theta}_i = (\alpha_i, b'_{2i})'$  且  $\hat{\theta} = (\hat{\theta}'_1, \dots, \hat{\theta}'_N)'$   $\in \mathbb{R}^{(K_2+1)N}$ ，以及  $\delta = (\gamma'_2, b'_{21}, \dots, b'_{2N})'$ 。然后令：

$$J_2 = T \min_{\delta \in \mathbb{R}^{K_2(1+N)}} (\hat{\theta} - h(\delta))' \hat{\Xi}^{-1} (\hat{\theta} - h(\delta)),$$

其中  $h(\delta) = (b'_{21}\gamma_2, b'_{21}, \dots, b'_{2N}\gamma_2, b'_{2N})'$  且  $\hat{\Xi} = \hat{\Omega}_\varepsilon \otimes (X'X)^{-1}$ 。根据Chamberlain (1984, 定理 1)，当  $T \rightarrow \infty$  时：

$$J_2 \implies \chi^2_{N-K_2}.$$

### 8.2.3 案例：Fama-French 三因子模型

Fama-French 三因子模型，由Fama & French (1993) 提出，是对资本资产定价模型(CAPM) 的重要拓展。该模型认为，股票的预期回报不仅仅取决于市场风险这单一因素，还与公司规模和账面市值比相关。具体来说，三因子模型的表达式为：

$$R_{it} - R_{ft} = \alpha_i + \beta_i (R_{mt} - R_{ft}) + s_i SMB_t + h_i HML_t + \varepsilon_{it},$$

其中  $R_{it}$  是股票  $i$  在时间  $t$  的收益率， $R_{ft}$  是无风险收益率， $R_{mt}$  是市场组合收益率； $SMB_t$  (Small Minus Big) 是规模因子，衡量了小市值股票组合和大市值股票组合的收益率之差，反映了公司规模对股票收益的影响； $HML_t$  (High Minus Low) 是价值因子，代表高账面市值比组合和低账面市值比组合的收益率之差，体现了价值型股票和成长型股票的收益差异； $\beta_i$ ,  $s_i$  和  $h_i$  分别是对应因子的系数。相较于 CAPM，三因子模型能更好地解释股票收益率的变动，在投资组合管理、资产定价等金融领域得到了广泛应用，帮助投资者更全面地分析股票的风险和收益特征，制定更为合理的投资策略。

以下 R 代码操作包括下载并处理 Fama/French 三因子模型的月度和日度数据，计算指定日期范围内（2013 年 1 月 1 日至 2023 年 12 月 31 日）多家能源公司股票的日收益率。接着我们将股票收益与 Fama/French 三因子进行回归分析。代码首先设置工作环境和所需的库，下载并清洗数据，随后对股票收益数据进行日收益率计算并存储，最后将每只股票的收益数据与市场因子数据合并，并运行线性回归模型以分析市场因子对股票收益的影响。整个过程还包括了数据的保存和回归分析结果的输出。这为金融市场分析提供了一个详细的数据处理和统计分析流程。这里我们主要使用了 `frenchdata` R 语言包，该软件包由 Nelson Areal 设计<sup>1</sup>，它可以从[https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data\\_library.html](https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html) 金融数据库中下载 Fama-French 三因子数据。

```

1 rm (list=ls ())
2
3 # 设置工作目录
4 if (requireNamespace("rstudioapi", quietly = TRUE) &&
5 rstudioapi::isAvailable()) {
6 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
7 }
8
9 # 加载所需的库
10 library (tidyverse)
11 library (scales)
```

<sup>1</sup>详见<https://www.tidy-finance.org/r/accessing-and-managing-financial-data.html>。

```
12 library (quantmod)
13 library (PerformanceAnalytics)
14
15 # 设置日期范围
16 start_date <- ymd ("2013-01-01")
17 end_date <- ymd ("2023-12-31")
18
19 # 加载Fama-French数据集库
20 library (frenchdata)
21
22 # 下载并处理Fama/French 3因子日度数据
23 factors_ff3_daily_raw <- download_french_data ("Fama/French 3 Factors [
24 Daily]")
25
26 factors_ff3_daily <- factors_ff3_daily_raw$subsets$data[[1]] |>
27 mutate (
28 date = ymd (date) ,
29 across (c (RF, `Mkt-RF` , SMB, HML) , ~as.numeric (.) / 100) ,
30 .keep = "none"
31) |>
32 rename_with (str_to_lower) |>
33 rename (mkt_excess = `mkt-rf`) |>
34 filter (date >= start_date & date <= end_date)
35
36 # 保存Fama-French日度数据
37 write.csv (factors_ff3_daily, "factors_ff3_daily.csv")
38
39 # 指定股票代码
40 tickers <- c ("NEE", "ENPH", "SEDG", "FSLR", "BEP", "PLUG", "TSLA", "VWDRY"
41 "")
42
43 # 初始化存储股票数据的列表
44 stock_data <- list ()
45
46 # 循环下载每个股票的数据
47 for (ticker in tickers) {
48 stock_data[[ticker]] <- getSymbols (ticker, src = "yahoo", from = start_
49 date, to = end_date, auto.assign = FALSE)
50
51 # 计算日收益率
52 stock_data[[ticker]] <- dailyReturn (C1 (stock_data[[ticker]]))
53 # 保存股票收益数据
54 write.csv (stock_data[[ticker]], paste0 (ticker, "_stock_data.csv"))
55 }
56
57 # 将Fama-French日度数据转换为xts格式以便合并
58 factors_ff3_daily_xts <- xts (factors_ff3_daily[,-1], order.by = factors_
59 ff3_daily$date)
60
61 # 对每只股票进行回归分析
62 results <- lapply (stock_data, function (stock) {
```

```

59 # 合并股票收益与Fama-French因子
60 merged_data <- merge (stock, factors_ff3_daily_xts, join = 'inner')
61 # 进行回归分析
62 fit <- lm (daily.returns ~ mkt_excess + smb + hml, data = merged_data)
63 return (summary (fit))
64 }

65
66 # 设置结果的名称为股票代码
67 names (results) <- names (stock_data)
68
69 # 查看每只股票的分析结果
70 results$NEE # NextEra Energy (美国最大的风能和太阳能发电公司)
71 results$ENPH # Enphase Energy (太阳能技术公司, 提供微型逆变器系统)
72 results$SEDG # SolarEdge Technologies (太阳能光伏逆变器生产商)
73 results$FSLR # First Solar (生产光伏模块的公司)
74 results$BEP # Brookfield Renewable Partners (可再生能源公司, 涵盖水力、风
 能等)
75 results$PLUG # Plug Power (提供氢燃料电池系统的公司)
76 results$TSLA # Tesla, Inc. (电动车和能源存储解决方案的制造商)
77 results$VWDRY # Vestas Wind Systems (风力发电设备制造商)
78
79 # 保存合并数据和回归输出
80 for (ticker in tickers) {
81 # 合并数据并再次进行回归分析
82 merged_data <- merge (stock_data[[ticker]], factors_ff3_daily_xts, join =
 'inner')
83 fit <- lm (daily.returns ~ mkt_excess + smb + hml, data = merged_data)
84
85 # 保存合并数据
86 write.csv (merged_data, paste0 (ticker, "_combined_data.csv"))
87
88 # 保存回归输出
89 fit_summary <- capture.output (summary (fit))
90 write (fit_summary, file = paste0 (ticker, "_regression_output.txt"))
91 }
92

```

以下为一组分析结果:

```

1 Call:
2 lm (formula = daily.returns ~ mkt_excess + smb + hml, data = merged_data)
3
4 Residuals:
5 Min 1Q Median 3Q Max
6 -0.165796 -0.012465 -0.000798 0.011249 0.165252
7
8 Coefficients:
9 Estimate Std. Error t value Pr (>|t|)
10 (Intercept) 0.001039 0.000476 2.183 0.0291 *
11 mkt_excess 0.931388 0.043636 21.344 <2e-16 ***
12 smb 0.077768 0.079293 0.981 0.3268
13 hml -0.035087 0.056460 -0.621 0.5344
14 ---

```

```

15 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
16
17 Residual standard error: 0.02492 on 2744 degrees of freedom
18 Multiple R-squared: 0.1513, Adjusted R-squared: 0.1504
19 F-statistic: 163.1 on 3 and 2744 DF, p-value: < 2.2e-16
20

```

除了 Fama–French 三因子模型之外，常用的扩展还包括 Carhart 四因子模型和 Fama–French 五因子模型等。Carhart 四因子模型由 Carhart (1997) 提出，在三因子基础上加入动量因子，用以刻画股票收益的短期持续性。其计量模型为

$$R_{it} - R_{ft} = \alpha_i + \beta_i (R_{mt} - R_{ft}) + s_i \text{SMB}_t + h_i \text{HML}_t + u_i \text{UMD}_t + \varepsilon_{it},$$

其中 UMD (Up Minus Down) 为动量因子，通常定义为“过去表现好（赢家）组合与表现差（输家）组合”的收益差。四因子模型在刻画动量效应与评价投资组合绩效方面往往表现更好。

#### 8.2.4 SMB/HML/UMD 的基本构建思路（实践口径可按研究设计微调）

##### 构建 SMB (Small Minus Big) 因子：小市值组合收益率减去大市值组合收益率

- (a) **数据准备：** 获取样本股票的市值 (MV) 与收益率 (一般为月度)，并确定滚动的形成期与持有期 (常见为月度再平衡，历史窗口 3–5 年等)。
- 2. **分组排序：** 在每个形成时点 (如每月末)，按市值从小到大排序，将样本分为“小盘 (Small)”与“大盘 (Big)”。常见做法是按中位数或按 50% 分位点切分；亦可做三分位/五分位切分。
- 3. **组合收益：** 分别计算 Small 与 Big 的下期组合收益 (等权或市值加权两种计算方式均可)。
- 4. **因子值：定义**

$$\text{SMB}_t = R_{\text{Small}, t} - R_{\text{Big}, t}.$$

##### 构建 HML (High Minus Low) 因子：高账面市值比减去低账面市值比

- 1. **数据准备：** 获取股票的账面价值和市值，计算账面市值比 (BM = Book-to-Market；或使用市净率 PB 的倒数)，并同步获取收益率数据。
- 2. **分组排序：** 在每个形成时点按 BM 排序，将样本划分为高 BM (Value) 与低 BM (Growth)。常见做法是取最高 30% 分位为 High、最低 30% 分位为 Low (中间 40% 可予以排除)。
- 3. **组合收益：** 计算高 BM 与低 BM 组合的下期收益 (等权或市值加权)。
- 4. **因子值：定义**

$$\text{HML}_t = R_{\text{High BM}, t} - R_{\text{Low BM}, t}.$$

### 构建 UMD (动量) 因子：赢家组合减去输家组合

1. **数据准备：**以日度或月度价格计算收益率，设定排序期（如过去 6/12 个月）与持有期（如 1/3/6 个月）。实务中常采用“跳月 (skip-month)”方法以弱化短期反转效应（例如用  $t - 12$  至  $t - 2$  的累计收益进行排序）。
2. **形成组合：**按排序期内累计收益率降序排列，取前 30% 作为赢家投资组合 (Winners)，后 30% 作为输家投资组合 (Losers)，中间 40% 可予以排除。
3. **组合收益：**在持有期内分别计算赢家与输家组合的收益（等权或市值加权），并按月度滚动再平衡，计算“组合族”的平均收益率。
4. **因子值：定义**

$$\text{UMD}_t = R_{\text{Winners}, t} - R_{\text{Losers}, t}.$$

以下 R 代码给出一个教学版的模拟示例（仅用于演示计算流程）：

```

1 set.seed(123)
2
3 # ----- 基本参数 -----
4 n_stocks <- 100
5 n_periods <- 24 # 24 期 (如 24 个月)
6
7 # ----- 模拟市值 / PB / 收益率 (示例数据) -----
8 market_values <- matrix(runif(n_stocks * n_periods, 100, 1100), nrow = n_
 stocks)
9 pb_ratios <- matrix(runif(n_stocks * n_periods, 1, 6), nrow = n_
 stocks)
10 returns <- matrix(runif(n_stocks * n_periods, -0.05, 0.05), nrow = n_
 stocks)
11
12 rownames(market_values) <- rownames(pb_ratios) <- rownames(returns) <-
 paste0("Stock_", 1:n_stocks)
13 colnames(market_values) <- colnames(pb_ratios) <- colnames(returns) <-
 paste0("Period_", 1:n_periods)
14
15 # ----- SMB 与 HML (等权口径) 的简单构造 -----
16 smb_values <- numeric(n_periods - 1)
17 hml_values <- numeric(n_periods - 1)
18
19 for (period in 2:n_periods) {
20 # SMB: 按市值二分
21 mv <- market_values[, period]
22 ord_mv <- order(mv) # 从小到大
23 k_half <- floor(length(mv) / 2)
24 small_ids <- rownames(market_values)[ord_mv[1:k_half]]
25 big_ids <- rownames(market_values)[ord_mv[(length(mv) - k_half + 1):
 length(mv)]]
26 small_ret <- mean(returns[small_ids, period], na.rm = TRUE)
27 big_ret <- mean(returns[big_ids, period], na.rm = TRUE)
28 smb_values[period - 1] <- small_ret - big_ret
29

```

```

30 # HML：按 BM=1/PB 三分的两端 30%
31 bm <- 1 / pb_ratios[, period]
32 ord_bm <- order(bm, decreasing = TRUE) # 高 BM 在前
33 k_top <- max(1, round(0.30 * length(bm)))
34 high_ids <- rownames(pb_ratios)[ord_bm[1:k_top]]
35 low_ids <- rownames(pb_ratios)[ord_bm[(length(bm) - k_top + 1):length(
 bm)]]]
36 high_ret <- mean(returns[high_ids, period], na.rm = TRUE)
37 low_ret <- mean(returns[low_ids, period], na.rm = TRUE)
38 hml_values[period - 1] <- high_ret - low_ret
39 }
40
41 names(smb_values) <- paste0("Period_", 2:n_periods)
42 names(hml_values) <- paste0("Period_", 2:n_periods)
43
44 cat("SMB 因子 (示例) :\n"); print(smb_values)
45 cat("HML 因子 (示例) :\n"); print(hml_values)

```

### UMD（动量因子）构造的教学示例

以下代码演示“排序期 12 个月、持有期 1 个月”的简单口径；以前 30% 与后 30% 的股票分别构建赢家组合与输家组合。实际研究中，建议采用跳月（skip-month）处理与滚动持有策略，并保持再平衡方式与权重计算口径的一致性。

```

1 set.seed(123)
2
3 # 模拟价格并计算收益率（仅示例）
4 prices <- matrix(runif(n_stocks * n_periods, 50, 150), nrow = n_stocks,
5 dimnames = list(paste0("Stock_", 1:n_stocks),
6 paste0("Period_", 1:n_periods)))
7 returns <- apply(prices, 1, function(p) c(NA, diff(p) / head(p, -1)))
8 returns <- t(returns) # 行=股票，列=期间
9
10 sorting_period <- 12 # 排序期
11 holding_period <- 1 # 持有期（本例为 1 期）
12
13 # 用最后 12 期做排序，最后 1 期做持有（示例）
14 sort_cols <- (n_periods - sorting_period + 1):n_periods
15 hold_col <- n_periods - holding_period + 1
16
17 # 计算排序期累计收益率（可换为累计 log-return 等）
18 sorting_returns <- rowSums(returns[, sort_cols, drop = FALSE], na.rm = TRUE)
19
20 # 划分赢家/输家（30% - 30%）
21 k_top <- max(1, round(0.30 * n_stocks))
22 ord_sort <- order(sorting_returns, decreasing = TRUE)
23 winners <- rownames(returns)[ord_sort[1:k_top]]
24 losers <- rownames(returns)[ord_sort[(n_stocks - k_top + 1):n_stocks]]
25
26 # 计算持有期收益并形成 UMD（等权）
27 winner_ret <- mean(returns[winners, hold_col], na.rm = TRUE)

```

```

28 loser_ret <- mean(returns[losers, hold_col], na.rm = TRUE)
29 UMD <- winner_ret - loser_ret
30 cat(sprintf("动量因子 UMD (示例) = %.6f\n", UMD))

```

Fama–French 五因子模型（包含 RMW 与 CMA）由 Fama & French (2015) 在三因子模型的基础上加入盈利能力因子 RMW (Robust Minus Weak) 和投资风格因子 CMA (Conservative Minus Aggressive) 构建而成：

$$R_{it} - R_{ft} = \alpha_i + \beta_i (R_{mt} - R_{ft}) + s_i \text{SMB}_t + h_i \text{HML}_t + r_i \text{RMW}_t + c_i \text{CMA}_t + \varepsilon_{it}.$$

其中 RMW 衡量“高盈利组合减低盈利组合”的收益差；CMA 衡量“保守投资组合减激进投资组合”的收益差。它们在解释横截面收益差异方面通常较三因子模型更有力。

### RMW (盈利能力因子) 构造要点

- 数据：**用年度/季度财报计算盈利能力指标（如 ROE/ROA、净利润/所有者权益等），同时准备价格数据以计算组合收益率。
- 分组：**在形成时点按盈利能力排序，取前 30% 为高盈利组 (Robust)、后 30% 为低盈利组 (Weak)。
- 组合与因子：**计算下一期高/低盈利组合收益率（等权或市值加权），定义

$$\text{RMW}_t = R_{\text{robust}, t} - R_{\text{weak}, t}.$$

### CMA (投资风格因子) 构造要点

- 数据：**以总资产增长率等指标衡量公司投资风格（“保守” / “激进”），同时准备价格数据计算收益率。
- 分组：**按资产增长率排序，取最“保守”（增长率较低）的前 30% 为 Conservative 组合，最“激进”（增长率较高）的后 30% 为 Aggressive 组合。
- 组合与因子：**计算下一期保守/激进组合收益率（等权或市值加权），定义

$$\text{CMA}_{t+1} = R_{\text{conservative}, t+1} - R_{\text{aggressive}, t+1}.$$

下列代码将展示基于模拟的 RMW/CMA 计算流程：

```

1 set.seed(123)
2
3 n_stocks <- 100
4 n_periods <- 24
5
6 # 模拟盈利能力 (如 ROE)、资产增长率、收益率
7 roe <- matrix(runif(n_stocks * n_periods, min = -0.10, max = 0.10), nrow = n_stocks,
8 dimnames = list(paste0("Stock_", 1:n_stocks), paste0("Period_",
9 , 1:n_periods)))
9 asset_growth <- matrix(runif(n_stocks * n_periods, min = 0.00, max = 0.50),
10 nrow = n_stocks,
11 dimnames = list(rownames(roe), colnames(roe)))
11 returns <- matrix(runif(n_stocks * n_periods, min = -0.05, max = 0.05), nrow =

```

```

12 = n_stocks,
13 dimnames = list(rownames(roe), colnames(roe)))
14
15 rmw_values <- numeric(n_periods - 1)
16 cma_values <- numeric(n_periods - 1)
17
18 for (period in 2:n_periods) {
19 # RMW: 按 ROE 两端 30%
20 rvec <- roe[, period]
21 ord_roe <- order(rvec) # 从小到大
22 k_low <- max(1, floor(0.30 * length(rvec)))
23 k_high <- k_low
24 weak_ids <- rownames(roe)[ord_roe[1:k_low]]
25 robust_ids <- rownames(roe)[ord_roe[(length(rvec) - k_high + 1):length(
26 rvec)]]
27 rmw_values[period - 1] <- mean(returns[robust_ids, period], na.rm = TRUE)
28
29 # CMA: 按资产增长率两端 30%
30 avec <- asset_growth[, period]
31 ord_ag <- order(avec) # 从小到大 (低=更保守)
32 conservative_ids <- rownames(asset_growth)[ord_ag[1:k_low]]
33 aggressive_ids <- rownames(asset_growth)[ord_ag[(length(avec) - k_high +
34 1):length(avec)]]
35 cma_values[period - 1] <- mean(returns[conservative_ids, period], na.rm =
36 TRUE) -
37 mean(returns[aggressive_ids, period], na.rm =
38 TRUE)
39 }
40
41 names(rmw_values) <- paste0("Period_", 2:n_periods)
42 names(cma_values) <- paste0("Period_", 2:n_periods)
43
44 cat("RMW 因子 (示例) : \n"); print(rmw_values)
45 cat("CMA 因子 (示例) : \n"); print(cma_values)

```

**实践提示与口径说明（中国市场）：**本章所展示的构建流程与代码主要用于示范，旨在清晰呈现实证分析的基本思路。用于生产级因子研究时，建议在中国 A 股或中证全指等样本范围内按以下统一口径执行：(i) 形成期与持有期设置：动量类因子建议“跳过最近 1 个月”，即在形成期与持有期之间设置 1 个月跳空；(ii) 组合加权方式：同时报告等权加权与市值加权两种口径，并明确“市值”为总市值还是流通市值；(iii) 组合分组断点：优先采用横截面分位点分组（如 30–40–30），或以全部 A 股/中证全指为基准计算断点，确保口径统一；(iv) 收益率计算口径：统一采用复权收益率，并说明使用前复权还是后复权价格，以及收益率为简单收益还是对数收益；(v) 数据清洗与调整：对缺失值、停复牌、涨跌停导致的不可交易样本，以及除权除息（现金分红、配股、送转股等）进行一致处理；(vi) 极端值处理：对极端值采用稳健方法（如 1% 分位截断或 1%/99% 温莎化），并说明是否进行了行业内标准化或市值中性化；(vii) 无风险利率选择：明确口径，例如中债 1 个月/3 个月国债到期收益率，或银行间 7 天回购加权利率；(viii) 组合构建与样本筛选：考虑滚动再平衡与跨期持有（尤其动量）引致的“组合族”平均效应，并披露是否剔除新股、次新股及

ST 样本。不同口径与设定会显著影响因子的时序特征与回归结果；报告时应明确披露以上关键设定，并提供稳健性检验与敏感性分析。

## 8.3 基于消费的资本资产定价模型

在金融资产定价领域，传统的资本资产定价模型（CAPM）和套利定价理论（APT）虽构建起资产定价的基本框架，但存在明显局限。传统模型多将资产价格单纯归结于投资者基于未来单期财富偏好所做出的投资组合选择，完全忽视了消费决策在其中扮演的关键角色。而现实世界里，投资者的决策并非局限于单一时期，其投资组合决策往往涉及多个时期的考量，消费与投资紧密相连，在这种跨期背景下，同时对消费和投资组合选择进行建模就显得尤为重要。此外，传统模型难以有效解释资产收益率的动态变化。真正的系统性风险因素从本质上讲是宏观经济层面的，主要源于消费的跨期边际替代率，资产价格也内生地由这些因素推导得出。但传统模型在确定无风险利率、投资者承担风险所要求的回报，以及解释资产收益率的可预测变化方面存在不足。例如，在 CAPM 中，无风险利率、零贝塔收益率以及承担市场风险的回报均为外生参数；APT 虽将单一市场风险价格替换为因子风险价格向量，但风险价格依旧在模型外确定。同时，无风险实际利率和股票超额收益率会随时间变化，这与市场有效性的关系以及能否构建合理模型来模拟现实，传统模型均无法给出令人满意的解答。而基于消费的资本资产定价模型（C-CAPM）则从消费视角出发，为解决上述问题提供了新的思路和方法。它将消费纳入资产定价的核心考量，能够更全面深入地剖析系统性风险因素与资产价格的内在联系，进而明确风险回报的决定机制。通过 C-CAPM，我们有望更准确地解释资产收益率的动态变化，使金融资产定价理论更加贴合现实金融市场的运行规律，为投资者决策、金融市场监管等提供更为可靠的理论依据，从而推动金融资产定价理论的进一步发展。在第 2.4.5 节中，我们介绍了 C-CAPM 的相关理论，C-CAPM 模型一般采用广义矩估计方法进行估计。

### 8.3.1 广义矩估计

考虑一个经济模型，该模型意味着一组  $r$  个总体矩约束满足：

$$\underbrace{E\{\mathbf{h}(\boldsymbol{\theta}, \mathbf{w}_t)\}}_{(r \times 1)} = 0, \quad (8.8)$$

其中  $\mathbf{w}_t$  是一个在  $t$  时刻已知的  $h \times 1$  维变量向量， $\boldsymbol{\theta}$  是一个待估计的  $a \times 1$  维未知参数向量。其思路是选择  $\boldsymbol{\theta}$ ，使得样本矩尽可能接近总体矩。在任何广义矩估计（GMM）中，将样本矩记为  $\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)$ ：

$$\underbrace{\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)}_{(r \times 1)} \equiv (1/T) \sum_{t=1}^T \mathbf{h}(\boldsymbol{\theta}, \mathbf{w}_t),$$

其中  $T$  是样本量， $\mathbf{y}_T \equiv (\mathbf{w}'_T, \mathbf{w}'_{T-1}, \dots, \mathbf{w}'_1)'$  是一个  $T \cdot h \times 1$  维的观测向量。GMM 估计量  $\hat{\boldsymbol{\theta}}$  使以下标量达到最小：

$$Q(\boldsymbol{\theta}; \mathbf{y}_T) = [\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)]' \mathbf{W}_T [\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)], \quad (8.9)$$

其中  $\{\mathbf{W}_T\}_{T=1}^\infty$  是一个  $r \times r$  维的正定矩阵序列，该矩阵可以是数据  $\mathbf{y}_T$  的函数。

如果  $r = a$ ，则通过令每个  $\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)$  等于零来估计  $\boldsymbol{\theta}$ 。GMM 一般指的是当  $r > a$  时，使用式(8.9) 来估计  $\boldsymbol{\theta}$ 。Hansen (1982) 证明了 GMM 估计量的渐近性质，指出在一定的正

则条件以及数据是严格平稳的条件下, GMM 估计量  $\hat{\boldsymbol{\theta}}$  是一致的, 其收敛速度与样本量的平方根成正比, 并且渐近服从正态分布。

Hansen (1982) 还确定了最优权重矩阵  $\mathbf{W}_T = \mathbf{S}^{-1}$ , 在 GMM 估计量类中, 它能使  $\hat{\boldsymbol{\theta}}$  成为最小方差估计量。最优权重矩阵是下式的逆矩阵:

$$\mathbf{S}_{r \times r} = \sum_{j=-\infty}^{\infty} E \left\{ [\mathbf{h}(\boldsymbol{\theta}_0, \mathbf{w}_t)] [\mathbf{h}(\boldsymbol{\theta}_0, \mathbf{w}_{t-j})]' \right\}.$$

最优权重矩阵依赖于真实参数值  $\boldsymbol{\theta}_0$ 。在实践中, 这意味着  $\hat{\mathbf{S}}_T$  依赖于  $\hat{\boldsymbol{\theta}}_T$ , 而  $\hat{\boldsymbol{\theta}}_T$  又依赖于  $\hat{\mathbf{S}}_T$ 。这种同时性问题通常通过迭代过程来处理: 首先, 在任意权重矩阵 (例如  $\mathbf{W} = \mathbf{I}$ ) 的条件下最小化  $Q(\boldsymbol{\theta}; \mathbf{y}_T)$ , 得到  $\boldsymbol{\theta}$  的初始估计值  $\hat{\boldsymbol{\theta}}_T^{(1)}$ ; 然后, 使用  $\hat{\boldsymbol{\theta}}_T^{(1)}$  得到  $\mathbf{S}$  的初始估计值  $\hat{\mathbf{S}}_T^{(1)}$ ; 接着, 使用初始估计值  $\hat{\mathbf{S}}_T^{(1)}$  重新最小化  $Q(\boldsymbol{\theta}; \mathbf{y}_T)$ , 得到新的估计值  $\hat{\boldsymbol{\theta}}_T^{(2)}$ 。持续迭代直至收敛, 或者在完成一次完整迭代后停止 (尽管这两个估计量的有限样本性质可能不同, 但它们具有相同的渐近分布)。另外, 也可以找到一个不动点。

Hansen (1982) 还基于检验统计量  $J_T$  提出了一个过度识别 (over-identifying, 简称 OID) 约束检验:

$$J_T \equiv T Q(\hat{\boldsymbol{\theta}}; \mathbf{y}_T) \Rightarrow \chi^2(r - a),$$

该检验要求  $r > a$ 。OID 检验是对模型设定本身的检验。若模型正确且总体矩约束成立, 则该检验在给定置信水平下考察矩条件 (8.8) 是否近似为零。统计量  $J_T$  计算较为简便, 等于样本量  $T$  乘以在估计参数值处的 GMM 目标函数值。参见 Chaussé (2010), 其对基于 R 语言进行 GMM 估计有详细论述, 此处不再赘述。

### 8.3.2 采用广义矩估计方法估计 C-CAPM 模型

Hansen (1982) 在引入 GMM 估计时就对 C-CAPM 模型进行了估计。具体来讲, Hansen (1982) 运用该方法对标准的基于消费的模型进行估计和检验。在这个模型中, 投资者追求效用最大化:

$$\max_{C_t} E_t \left[ \sum_{i=0}^{\infty} \beta^i u(C_{t+i}) \right],$$

其中效用函数采用幂效用形式: 当  $\gamma > 0$  时,  $u(C_t) = \frac{C_t^{1-\gamma}}{1-\gamma}$ ; 当  $\gamma = 1$  时,  $u(C_t) = \ln(C_t)$ .

最优消费选择的一阶条件为:

$$C_t^{-\gamma} = \beta E_t \left\{ (1 + R_{i,t+1}) C_{t+1}^{-\gamma} \right\}, \quad i = 1, \dots, N,$$

其中  $i = 1, \dots, N$  对应可交易的资产。上述矩条件构成了 GMM 估计的基础。根据 GMM 理论的要求, 这些矩条件必须改写为用严格平稳变量表示的形式:

$$0 = E_t \left\{ 1 - \beta \left[ (1 + R_{i,t+1}) \frac{C_{t+1}^{-\gamma}}{C_t^{-\gamma}} \right] \right\}. \quad (8.10)$$

所以我们选择消费增长率, 因为消费水平一般呈现出明显的趋势, 为非平稳序列。

模型有两个待估计的参数, 即  $\boldsymbol{\theta} = (\beta, \gamma)'$ 。等式(8.10) 对应一个横截面资产定价: 给定一组  $i = 1, \dots, N$  种资产的收益率, 预期收益率的横截面变化可以由收益率与  $M_{t+1} = \beta (C_{t+1}/C_t)^{-\gamma}$  的协方差来解释。

用  $\mathcal{F}_t$  表示投资者的信息集。等式(8.10) 意味着:

$$0 = E \left\{ \left[ 1 - \left\{ \beta (1 + R_{i,t+1}) C_{t+1}^{-\gamma} / C_t^{-\gamma} \right\} \right] | \mathcal{F}_t \right\}, \quad i = 1, \dots, N. \quad (8.11)$$

令  $\mathbf{x}_t \subseteq \mathcal{F}_t$  为可观测信息集  $\mathcal{F}_t$  的一个子集。那么条件期望方程(8.11) 意味着以下无条件模型:

$$0 = E \left\{ \left[ 1 - \left\{ \beta (1 + R_{i,t+1}) \frac{C_{t+1}^{-\gamma}}{C_t^{-\gamma}} \right\} \right] \mathbf{x}_t \right\}, \quad i = 1, \dots, N.$$

如果  $\mathbf{x}_t$  是一个  $M \times 1$  的向量, 那么存在  $r = N \cdot M$  个矩约束, 可用于检验资产定价模型, 其中:

$$\mathbf{h}(\boldsymbol{\theta}, \mathbf{w}_{t+1}) = \begin{bmatrix} \left[ 1 - \beta \left\{ (1 + R_{1,t+1}) \frac{C_{t+1}^{-\gamma}}{C_t^{-\gamma}} \right\} \right] \mathbf{x}_t \\ \left[ 1 - \beta \left\{ (1 + R_{2,t+1}) \frac{C_{t+1}^{-\gamma}}{C_t^{-\gamma}} \right\} \right] \mathbf{x}_t \\ \vdots \\ \vdots \\ \left[ 1 - \beta \left\{ (1 + R_{N,t+1}) \frac{C_{t+1}^{-\gamma}}{C_t^{-\gamma}} \right\} \right] \mathbf{x}_t \end{bmatrix}. \quad (8.12)$$

只要  $r \geq 2$ , 就可以对该模型进行估计和检验。对式(8.12) 取样本均值可得到  $\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)$ 。Hansen (1982) 通过最小化下式来估计参数:

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \mathbf{y}_T) = [\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)]' \hat{\mathbf{S}}_T^{-1} [\mathbf{g}(\boldsymbol{\theta}; \mathbf{y}_T)],$$

其中  $\hat{\mathbf{S}}_T^{-1}$  是最优权重矩阵  $\mathbf{S}^{-1}$  的一个估计值。

Hansen (1982) 在  $\mathbf{x}_t$  中使用了消费增长率的滞后项和资产收益率的滞后项。他们使用股票市场指数和行业股票收益率作为  $R_{i,t}$  的数据。消费以国民收入和生产账户中的非耐用品和服务支出进行衡量。他们发现, 在大多数设定下,  $\beta$  的估计值约为 0.99。他们还发现, 相对风险厌恶系数的估计值  $\hat{\gamma}$  相当低, 范围在 0.35 到 0.999 之间。

注: 这里不存在股权溢价之谜 (equity premium puzzle), 因为该模型是利用  $\mathbf{x}_t$  中的条件信息进行估计的, 即通过  $\mathbf{R}_{t+1}\mathbf{x}_t$  进行估计的。这些收益率不同于简单 (未调整) 的股票市场超额收益率, 后者才是股权溢价之谜所涉及的内容。

### 8.3.3 时变风险厌恶与股权风险溢价之谜

股权溢价之谜是金融经济学中的经典难题, 最早由 Mehra & Prescott (1985) 提出。传统的消费型资本资产定价模型 (C-CAPM) 在理论上将资产收益与投资者消费联系起来; 但用该模型检验实际数据时, 预测的股权溢价远低于观测值, 由此形成“股权溢价之谜”。长期样本显示, 股票年均收益率通常高于无风险资产 (如短期国债) 数个百分点, 而在合理的风险厌恶系数下, C-CAPM 难以匹配这一差距。

一种可能的解释是假定结构参数随时间变化: 令

$$\beta_t = \beta(\mathcal{F}_{t-1}), \quad \gamma_t = \gamma(\mathcal{F}_{t-1}),$$

更具体地, 以状态向量  $\mathbf{x}_t \in \mathcal{F}_{t-1}$  为自变量,

$$\beta_t = \beta(\mathbf{x}_t), \quad \gamma_t = \gamma(\mathbf{x}_t),$$

其中  $\beta(\cdot)$  与  $\gamma(\cdot)$  为未知光滑函数，用于刻画风险态度随状态变化的规律。

由欧拉方程（见式 (8.11)）可得一个等价的广义回归式：

$$\beta(X_t) \left( \frac{C_{t+1}}{C_t} \right)^{-\gamma(X_t)} (1 + R_{i,t+1}) = 1 + \varepsilon_{t+1},$$

其中  $\varepsilon_{t+1}$  为随机定价误差，满足鞅差性质  $E(\varepsilon_{t+1} | \mathcal{F}_t) = 0$ ； $X_t$  为状态变量。

据此，可用低阶局部多项式（如局部线性）最小化下式的局部广义残差平方和以估计  $\beta(\cdot)$  与  $\gamma(\cdot)$ ：

$$\min_{\beta(\cdot), \gamma(\cdot)} \sum_{t=1}^T \left[ \beta(X_t) \left( \frac{C_{t+1}}{C_t} \right)^{-\gamma(X_t)} (1 + R_{i,t+1}) - 1 \right]^2 K_h \left( \frac{x - X_t}{h} \right).$$

更一般地，也可将欧拉方程写为时变 GMM 的矩条件：

$$E \left\{ \left[ \beta(X_t) \left( \frac{C_{t+1}}{C_t} \right)^{-\gamma(X_t)} (1 + R_{i,t+1}) - 1 \right] \mathbf{x}_t \mid \mathcal{F}_t \right\} = 0,$$

其中  $\mathbf{x}_t \subseteq \mathcal{F}_t$  为工具变量。用局部多项式逼近  $\beta(\cdot)$  与  $\gamma(\cdot)$ ，并最小化样本矩的局部二次型，可得到相应的局部多项式 GMM 估计量。

## 8.4 章节总结

本章聚焦金融资产定价模型，系统介绍不同类型的模型及其检验方法，并结合案例加深理解。首先阐述资本资产定价模型 (CAPM)，包括模型估计与检验；在模型检验中介绍了沃尔德 (Wald) 检验与似然比检验，并讨论时变 CAPM 与条件 CAPM，以反映市场条件与资产风险的动态性。随后引入多因子模型，既包括基于投资组合收益的多因子模型，也包括包含宏观因子与无风险利率的多因子模型，并给出相应的检验方法；通过 Fama–French 三因子模型的案例展示其在资产定价中的应用。最后介绍基于消费的资本资产定价模型，该模型立足宏观一般均衡框架，以消费者的消费行为为切入点，为金融资产定价提供新的视角，有助于更全面地理解价格形成机制。

## 8.5 习题

1. 在第 8.2.3 节中，我们仅对 Fama–French 三因子模型进行了估计。请构建假设检验统计量，对  $\alpha = 0$  进行检验，并对  $B_2 = 0$  进行检验，其中  $B_2$  包括  $SMB_t$  和  $HML_t$  的系数。使用如沪深 300 指数等市场指数的日收盘价数据，计算并整理用于构建多因子模型的因子数据，包括规模因子 (SMB)、价值因子 (HML) 和动量因子 (UMD) 等。
2. 用本章学习的模型对中国股市进行分析。
  - (a) 数据下载：从合适的金融数据平台（如 Wind 等）获取 A 股市场相关数据。选取至少 30 只不同行业的 A 股股票作为样本（覆盖大盘股、中盘股、小盘股），下载 2020 年 1 月 1 日至 2024 年 12 月 31 日期间的日收盘价；同时下载同期无风险利率（可用中国国债收益率替代）和市场指数（如沪深 300 指数）的日收盘价。计算并整理多因子模型所需的因子数据，包括 SMB、HML、UMD 等。

- (b) 数据预处理：对下载数据进行清洗和整理；处理缺失值与异常值——缺失值可采用均值填充、删除等方法；用统计方法识别并修正异常值。
- (c) 模型构建与分析：运用处理后的数据，构建 CAPM、Fama-French 三因子模型、Carhart 四因子模型；分析各因子对股票收益率的影响是否显著，并比较各模型对股票收益率的解释力度。
- (d) 报告撰写：形成一份详细分析报告。内容包括数据来源、数据处理方法、模型构建过程、回归结果分析以及结论与建议等，要求条理清晰、逻辑严谨、图表使用恰当。
3. 请使用中国的宏观经济数据和金融市场数据，对基于消费的资本资产定价模型（C-CAPM）进行实证分析。具体要求如下：
- (a) 数据：收集 2012 年 1 季度至 2019 年 4 季度的季度宏观经济数据（如居民人均消费支出、国内生产总值（GDP）等）及金融市场数据（如沪深 300 指数收益率、无风险利率；建议采用国债收益率替代）。数据来源可选用国家统计局网站、Wind 数据库等权威渠道。对数据进行预处理，包括缺失值处理、季节性调整和对数差分等操作，以获取平稳的时间序列。
  - (b) 估计与检验：依据 C-CAPM 基本理论设定模型形式，考虑时变参数的可能性（如假设主观贴现率因子  $\beta$  和相对风险厌恶系数  $\gamma$  为时变），构建欧拉方程。使用广义矩估计（GMM）进行估计，确定最优权重矩阵并得到参数估计值；对结果进行统计检验，包括参数显著性检验和模型拟合优度检验，判断模型对中国市场资产定价现象的解释程度。
  - (c) 经济含义：分析估计的参数值，解释主观贴现率因子  $\beta$  与相对风险厌恶系数  $\gamma$  的经济含义，讨论其在中国金融市场中的合理性。
  - (d) 解释股权风险溢价之谜：检验模型能否解释“股权风险溢价之谜”，即模型预测的股权溢价与实际观测值之间的差异；若存在差异，需分析可能原因，如投资者非理性行为、市场摩擦因素、宏观经济不确定性等。
  - (e) 稳定性与适用性：比较不同时期或不同市场状态下的估计结果，分析模型的稳定性与适用性，探讨影响 C-CAPM 在中国市场表现的主要因素。
  - (f) 研究展望：指出研究的局限性，并对未来方向提出展望，如引入更多解释变量、改进模型设定等。
  - (g) 政策与实践建议：基于实证结果，为投资者资产配置与监管政策制定提供合理建议。



# 9 连续时间模型与高频波动率估计

连续时间模型与高频波动率估计共同构成当代金融计量的两大支柱：前者为资产价格的动态建模与推断提供统一的随机分析框架，后者则将高频成交/报价数据转化为可用于风险管理与资产配置的统计量。二者在理念与工具上相互补充——扩散型随机微分方程（SDE）给出价格在无穷小时间步上的近似法则，而实现波动率/协方差等“高频统计量”在离散观测下逼近 SDE 的二次变差，从而把连续理论落实为可计算的样本外度量。

本章围绕两个核心问题展开：第一，如何以最少的建模假设刻画价格过程的“漂移 + 扩散”结构，并在无法得到闭式转移密度时仍能进行有效估计与设定检验；第二，如何在包含微观结构噪声与异步交易的现实数据中，稳健地估计波动率与协方差，并使之服务于组合与风险度量。为此，我们采用如下路线：首先由布朗运动出发，介绍随机积分、伊藤引理与扩散过程的统计推断；继而转向高频统计量，说明如何在“无噪声”与“有噪声”两种情形下得到一致估计，并处理多资产的时间同步问题。

章节结构如下：第 9.5 节前给出布朗运动、随机积分与伊藤引理，随后在式 (9.2) 的框架内介绍扩散过程的存在唯一性与常见模型族；在“扩散过程的极大似然估计”部分，先给出可解密度下的精确 MLE，再转向 Aït-Sahalia 展开的近似 MLE 与设定检验；接着进入高频波动率与协方差的估计：从 RV/CLT 与  $\widehat{IQ}$  出发，系统讨论微观结构噪声的偏差及其修正（TSRV、MSRV、RK、PAV）与时间同步；最后以组合最小方差实例收尾，给出从估计矩阵到投资权重的可操作链条，并在“章节总结”中归纳要点。

## 9.1 布朗运动

布朗运动也称维纳过程（Wiener process），源于诺伯特·维纳（Norbert Wiener）在 1923 年构建的数学模型，用以描述粒子在液体中的随机运动。资产价格的随机演化可视为布朗运动的一个应用。布朗运动（Brownian motion）的发现要追溯到 1827 年，当时英国植物学家罗伯特·布朗（Robert Brown）利用普通显微镜观察悬浮于水中由花粉裂解出的微粒，发现这些微粒呈现出不规则的运动，于是这种运动便被命名为布朗运动。自 1860 年起，众多科学家投身于对布朗运动现象的研究，逐渐揭示出它的一系列主要特性：粒子的运动由平移及转动构成，毫无规则可言，其轨迹几乎处处不可微；粒子间的移动相互独立，即便彼此接近到小于直径的距离亦是如此；粒子越小、液体黏度越低或者温度越高，粒子运动就越剧烈；粒子的成分与密度对其运动毫无影响；并且粒子的运动永不停息。1905 年，爱因斯坦为解释布朗运动提出理论。因布朗粒子受撞击频繁，经典力学难以测量其运动距离，他转而研究粒子群体的行为。将粒子一维运动增量视为随机变量，给出概率密度函数，经泰勒级数展开、积分化简等推导，得出布朗粒子密度满足的扩散方程  $\partial \rho / \partial t = D \cdot \partial^2 \rho / \partial x^2$ ，并给出初始时刻  $t = 0$  的解，为布朗运动研究奠定基础，使其成为随机分析中的关键概念。

虽然有多种介绍布朗运动的方法，为便于解释，我们采用最直观的方法，即从随机游走引入布朗运动。我们以标准正态分布这一相对简单的正态分布作为基础“砖石”，并逐步构

建用于描述布朗运动的维纳过程。首先，我们着眼于离散时间（discrete time）的随机游走。

假设有一系列随机变量  $\varepsilon_t$ ，它们均服从标准正态分布， $\varepsilon_t \sim N(0, 1)$ 。 $\varepsilon_t$  两两独立同分布，简称为“i.i.d.”。再假设有一系列随机变量  $\{z_t\}$ ，满足如下条件：

$$z_1 - z_0 = \varepsilon_0,$$

⋮

$$z_{t+1} - z_t = \varepsilon_t,$$

⋮

即  $z_t$  的变化量是服从标准正态分布的  $\varepsilon_t$ 。可以把  $z_t$  想象成一个粒子在数轴上所处的位置；该粒子每向前移动一步，其移动距离是随机的并服从标准正态分布。随机变量序列  $\{z_t\}$  共同构成一个随机过程（stochastic process），即随机游走（random walk）。

由上式可得

$$z_t - z_0 = \sum_{j=0}^{t-1} \varepsilon_{t-1-j}.$$

鉴于  $\{\varepsilon_t\}$  独立同分布，我们可进一步分析其期望与方差。对于期望，

$$\mathbb{E}(z_t - z_0) = \mathbb{E}\left(\sum_{j=0}^{t-1} \varepsilon_{t-1-j}\right) = \sum_{j=0}^{t-1} \mathbb{E}(\varepsilon_{t-1-j}) = 0.$$

对于方差，

$$\text{Var}(z_t - z_0) = \text{Var}\left(\sum_{j=0}^{t-1} \varepsilon_{t-1-j}\right) = \sum_{j=0}^{t-1} \text{Var}(\varepsilon_{t-1-j}) = t.$$

因此，对于处于随机游走状态的粒子而言，其在任意时刻所处的位置也是随机变量：其期望等于初始位置（即  $z_0$ ），方差等于经历的时间  $t$ 。相应地，标准差（即波动率）与  $\sqrt{t}$  成正比。

接下来由离散时间随机游走拓展至连续时间下的布朗运动。假设随机过程  $\{z_t\}$  具有如下性质：任意两个时刻之间的增量服从正态分布，其均值为 0、方差等于这两个时刻之间的时间差，即

$$z_{t+\Delta} - z_t \sim N(0, \Delta), \quad \Delta > 0.$$

限定  $\Delta$  只能取正整数则得到前述随机游走；在连续时间情形，仅要求  $\Delta$  为正实数（无论多小），即可把离散情形扩展到连续情形。

此外，还需设定  $\{z_t\}$  为独立增量过程：在任意一组两两不相交的时间区间上， $z_t$  的增量相互独立。可将其视为对离散时间下 i.i.d. 随机变量  $\varepsilon_t$  特性的拓展。满足上述条件的随机过程称为布朗运动，亦即连续时间下的随机游走。

下面给出布朗运动（维纳过程）的严格数学定义。

**定义 9.1：** 若随机过程  $\{X(t), t \geq 0\}$  满足：

1.  $X(t)$  为独立增量过程；
2. 对任意  $s, t > 0$ ， $X(s+t) - X(s) \sim N(0, \sigma^2 t)$ （即该增量均值为 0、方差为  $\sigma^2 t$  的正态分布）；
3.  $X(t)$  关于  $t$  连续。

则称  $\{X(t), t \geq 0\}$  为布朗运动（或维纳过程）。若  $\sigma = 1$ ，则称之为标准布朗运动。

一般用  $B_t$  表示标准布朗运动，具有如下性质：

1. **独立增量性**：对于任意两个时间点  $s < t$ ，增量  $B_t - B_s$  与历史信息  $\mathcal{F}_s$  独立（其中  $\mathcal{F}_s$  表示由时刻  $s$  及其之前的观测所生成的滤过）。
2. **正态分布的增量**：增量  $B_t - B_s$  服从均值为 0、方差为  $t - s$  的正态分布，即  $B_t - B_s \sim N(0, t - s)$ 。
3. **非平稳性与平稳增量**：虽然过程  $B_t$  本身非平稳（其协方差函数为  $\text{Cov}(B_s, B_t) = \min(s, t)$ ），但其增量是平稳的，即增量的分布仅取决于时间间隔  $|t - s|$ ，与具体时点无关。
4. **无限可分性**：从任意初始时刻  $t_0 = 0$  开始，布朗运动可以写为一系列独立正态增量之和，

$$B_t = \sum_{i=1}^n (B_{t_i} - B_{t_{i-1}}) = \sum_{i=1}^n Z_i \sqrt{t_i - t_{i-1}}, \quad 0 = t_0 < t_1 < \dots < t_n = t,$$

其中  $\{Z_i\}_{i=1}^n$  为独立同分布的  $N(0, 1)$  随机变量。

```

1 # 设置参数
2 set.seed (123) # 设置随机种子以获得可复现的结果
3 n <- 1000 # 时间步数
4 T <- 1 # 总时间
5 dt <- T / n # 时间间隔
6 t <- seq (0, T, length.out = n + 1) # 时间序列
7
8 # 生成标准正态随机变量
9 dW <- rnorm (n, mean = 0, sd = sqrt (dt))
10
11 # 计算布朗运动的路径
12 W <- c (0, cumsum (dW))
13
14 # 绘制布朗运动的路径
15 plot (t, W, type = 'l', main = "标准布朗运动", xlab = "时间", ylab = "W (t)",
16 col = "blue", lwd = 2)
17

```

接着，我们介绍布朗桥的概念。

**定义 9.2：** 设  $B_t$  为标准布朗运动，其中  $B_0 = 0$ ，定义至时间  $T > 0$  的布朗桥为：

$$\mathbb{B}_t = B_t - \frac{t}{T} B_T.$$

布朗桥是金融数学中的一个重要概念，在期权及其他衍生品定价领域有着广泛应用。由于布朗桥能够模拟资产价格在特定时间点结束时达到特定水平的路径，因此可用于计算路径依赖型期权的价格，例如亚式期权<sup>1</sup>。

<sup>1</sup> 亚式期权（Asian option）是一种金融衍生品，其收益取决于标的资产在一段时间内的平均价格，而非某一具体到期时刻的价格。

**定义 9.3:** 若函数  $f : [0, T] \rightarrow \mathbb{R}$  满足：对区间  $[0, T]$  的任一分割  $0 = t_0 < t_1 < \dots < t_n = T$  都有

$$\sup_{\{0=t_0 < \dots < t_n=T\}} \sum_{i=0}^{n-1} |f(t_{i+1}) - f(t_i)|^p < \infty, \quad p > 0,$$

则称  $f$  具有有界  $p$ -变差 (*bounded  $p$ -variation*)。

连续可微函数具有有界 1-变差 (bounded 1-variation) (或简称有界变差)。布朗运动的样本路径在任何紧区间上均为无界变差 (unbounded variation)；其路径四处游走。因而无法采用黎曼积分对函数  $f$  关于  $B$  进行积分，需要引入随机积分的概念，见第 9.5 节。

## 9.2 随机微分

为了更深入地理解布朗运动，我们从微分和积分两方面进行探究。首先，从微分的角度入手。当我们考虑非常小的时间间隔  $\Delta \rightarrow 0$  时，布朗运动的表现形式可通过以下微分方程来表述：

$$dz_t = \lim_{\Delta \rightarrow 0} (z_{t+\Delta} - z_t),$$

这里的时间间隔  $\Delta$  始终为正，且总是从上方趋近于 0。对应的离散时间表达式为：

$$\varepsilon_{t+1} = z_{t+1} - z_t.$$

随机变量的量级通常由其标准差决定，因为随机变量的实际取值通常接近其标准差。作为随机变量的布朗运动微分  $dz_t$  具有一个独特性质，即它的标准差与  $\sqrt{\Delta}$  相当。这是因为  $dz_t$  的方差为  $\Delta$ ，根据标准差与方差的关系可知，其标准差就是  $\sqrt{\Delta}$ 。

这一点导出了两个重要的结论：第一，布朗运动虽连续无间断，但处处不可导（导数趋向无穷大）。这是因为在  $\Delta$  极小的情况下， $\sqrt{\Delta}$  变得非常大。因此，按照导数的定义，布朗运动的导数应为  $\lim_{\Delta \rightarrow 0} \frac{\sqrt{\Delta}}{\Delta} \rightarrow +\infty$ 。这表明布朗运动在任何点上都不可导。第二，无论时间区间多么短，布朗运动始终显示为随机波动的图像，而不是平滑的直线。这表明布朗运动无法通过一系列微小的确定性运动来合成，它本质上是不可约分的最基本随机过程之一。



在数学的发展历史中，人们曾长期秉持这样一种直观的观点：连续函数在某些点上应该是可导的。然而，19世纪时，卡尔·魏尔斯特拉斯 (Karl Weierstrass) 构造出了一个处处连续却处处不可导的函数，彻底颠覆了这一传统认知。

这个函数通常被称为 **魏尔斯特拉斯函数** (Weierstrass function)，其常见形式是通过一个无限级数来定义的：

$$W(x) = \sum_{n=0}^{\infty} a^n \cos(b^n \pi x),$$

其中  $0 < a < 1$ ,  $b$  为正奇数，并且要求满足  $ab > 1 + \frac{3}{2}\pi$ 。这些限制条件确保该级数在所有点上都连续，但在任意一点上都不可导。

实际上，如果将布朗运动视为时间的函数，它也是一个处处连续但处处不可导的函数。

当  $\Delta \rightarrow 0$  时，得到布朗运动，记作  $B_t$ 。对于初次接触的随机变量，我们通常先关注其

期望与方差，布朗运动的微分  $dB_t$  也不例外。根据定义：

$$E_t(dB_t) = 0,$$

其中下标  $t$  表示该期望在时点  $t$  计算。由于研究对象是随机过程，不同时间点的期望可能不同。此外， $dB_t$  在  $t$  时刻的方差为  $dt$ ，即

$$dt = \text{Var}_t(dB_t) = E_t[(dB_t - E_t(dB_t))^2] = E_t[dB_t^2],$$

这表明  $dB_t^2$  与  $dt$  为同阶无穷小，因此  $dB_t$  与  $\sqrt{dt}$  亦为同阶无穷小。

基于标准布朗运动的微分，可构建更一般的资产价格变动模型，广义布朗运动表述为

$$dX_t = \mu dt + \sigma dB_t. \quad (9.1)$$

其中  $dX_t$  表示位置（价格）的瞬时变化， $\mu dt$  为漂移项（drift term）， $\sigma$  为位置变化的标准差，亦称扩散系数（diffusion coefficient）。因此，

$$\begin{aligned} E_t[dX_t] &= \mu dt + \sigma E_t[dB_t] = \mu dt, \\ \text{var}_t[dX_t] &= \sigma^2 E_t[dB_t^2] = \sigma^2 dt. \end{aligned}$$

在金融计量经济学中，式 (9.1) 常用于表征资产价格的变动，各项具有明确的经济含义： $\mu dt$  体现“资金的时间价值”，即在无风险条件下价格随时间的预期增长，通常与通胀、基准利率和经济增长预期等宏观因素相关； $\sigma dB_t$  则刻画风险部分，其中  $\sigma$  度量价格波动幅度， $dB_t$  代表随机扰动，反映市场情绪、供需变化、政治事件等难以预测因素对资产价值的即时影响。

### 9.3 布朗运动的其他性质

接下来我们介绍布朗运动样本轨道的一些性质。布朗运动具有连续的样本轨道，且在局部 Hölder 连续性方面表现良好：对任意  $\gamma < \frac{1}{2}$ ，几乎必然有

$$|B_t - B_s| \leq C |t - s|^\gamma,$$

其中  $C$  为有限常数。然而，对任何  $\gamma > \frac{1}{2}$ ，样本轨道在局部都不是 Hölder 连续的，且几乎处处不可微。实际上，其连续性模量为  $g(\delta) = (2\delta \ln(1/\delta))^{1/2}$ ，即

$$\Pr \left( \limsup_{\delta \rightarrow 0^+} \frac{1}{g(\delta)} \max_{\substack{0 \leq s < t \leq 1 \\ t-s \leq \delta}} |B_t - B_s| = 1 \right) = 1.$$

这表明，当时间间隔  $\delta \rightarrow 0$  时，布朗运动的最大增量在适当标准化后，其上极限几乎必然等于 1，凸显了布朗运动在微观尺度上的极端不规则性与剧烈随机波动。

接着讨论布朗运动的穿越时间（crossing times）。所谓穿越时间，是指过程首次越过某阈值的时刻。对标准布朗运动  $B_t$ ，定义首次穿越（到达）时间

$$\tau_a = \inf\{t : |B_t| > a\},$$

其中  $B_0 = x$ 。这是一个停时的例子：停时要求在任意时刻  $s$  之前是否停止（即  $\tau \leq s$ ）的

判定只能依赖于过程在  $r \leq s$  时刻所包含的信息。随机变量  $\tau_a$  已被广泛研究。

**定义 9.4 (停时 (stopping time) ):** 给定随机过程  $\{X_t\}_{t \geq 0}$  及其自然滤过  $\mathcal{F}_t^X \equiv \sigma(X_u : 0 \leq u \leq t)$ 。若随机变量  $\tau : \Omega \rightarrow [0, \infty]$  满足对任意  $s \geq 0$ , 事件  $\{\tau \leq s\} \in \mathcal{F}_s^X$ , 则称  $\tau$  为 (相对于  $\mathcal{F}_t^X$  的) 停时。直观地说, 是否在时刻  $s$  停止, 只依赖于过程在  $s$  及之前的路径信息  $\{X_t : t \leq s\}$ 。

停时可以视为游戏或实验中的“停止按钮”：当某个特定事件发生时按下即停。例如，赌徒在赌场中可能将停止时刻设为其资本首次达到或超过某个预设目标金额, 或资本耗尽的时刻。该停止时刻就是一个停时, 因为它完全由赌徒资本过程的历史路径决定。

自 Bachelier (1900) 的研究以来,  $\tau_a$  的分布就已为人所知, 其分布为:

$$\Pr(\tau_a \leq t | x, a) = 1 - \frac{2}{\pi} \sum_{j=0}^{\infty} \frac{(-1)^j}{j + \frac{1}{2}} \cos\left(\left(j + \frac{1}{2}\right) \frac{\pi x}{a}\right) \exp\left(-\left(j + \frac{1}{2}\right)^2 \pi^2 t / 2a^2\right).$$

(见 Darling & Siegert (1953, 第 632 页))。密度函数通过求导得出:

$$f_0(t | x, a) = \frac{\pi}{a^2} \sum_{j=0}^{\infty} (-1)^j \left(j + \frac{1}{2}\right) \cos\left(\left(j + \frac{1}{2}\right) \frac{\pi x}{a}\right) \exp\left(-\left(j + \frac{1}{2}\right)^2 \pi^2 t / 2a^2\right).$$

当  $x = 0$  时,  $E(\tau_a) = a^2 / \sigma^2$ 。

接着我们考虑单侧穿越概率 (one-sided crossing probability)。

**定义 9.5:** 考虑一个随机过程  $X(t)$ , 例如布朗运动  $B(t)$ 。设定一个阈值  $a$ 。单侧穿越概率定义为该过程在某个特定时间  $t$  或之前首次越过阈值  $a$  的概率, 即:

$$\Pr(\tau_a^+ \leq t) = \Pr(\min\{s : X(s) > a\} \leq t),$$

这里  $\tau_a^+$  是随机过程  $X(t)$  首次达到阈值  $a$  的时刻。

相比停时, 单侧穿越概率要简单得多。设:

$$\tau_a^+ = \min\{t : B_t > a\}.$$

假设  $B_0 = x = 0$ 。根据 Feller (1991 第 171 页) 的研究, 我们有:

$$\begin{aligned} \Pr(\tau_a^+ \leq t) &= 2 \left(1 - \Phi\left(\frac{a}{\sqrt{t}}\right)\right), \\ f_0(t | x, a) &= \frac{a}{\sqrt{2\pi t^3}} \exp\left(-\frac{a^2}{2t}\right). \end{aligned}$$

注意,  $f_0$  是  $(0, \infty)$  上的有界密度函数。该密度的形状随  $a$  而变化。例如: 当  $a = 5$  时, 穿越概率非常低, 且密度在期末单调上升。

单侧穿越概率有助于理解“熔断机制”。以伦敦证券交易所 (LSE) 为例, 在单个交易日的连续竞价过程中, 如大盘股价格相对开盘价的涨跌幅超过预设阈值 (历史上常设为 10%), 熔断机制即刻触发。市场实证表明, 熔断在交易日的早盘与尾盘更为高发。

停时和单侧穿越概率在金融市场监控、风险管理以及市场行为预测等诸多领域有很大的价值。金融机构借助对停止时间以及单侧穿越概率的研究, 能够设计应对市场极端波动

情形的策略，并开发高效实用的交易策略及各类金融产品，保障交易与投资活动正常开展，助力市场参与者在错综复杂、变幻莫测的金融环境中始终维保持竞争力。

## 9.4 伊藤引理 (Itô's Lemma)

在建立布朗运动的数学模型之后，一个自然的问题是：若某随机变量遵循布朗运动，那么该随机变量的函数的运动规律如何？这在资产定价中极为常见。以期权为例，期权价格是股票价格的函数；当股票价格服从布朗运动时，期权价格将如何变化？

伊藤引理在此成为回答此类问题的核心数学工具。它是随机微积分中的基础而关键的结果，尤其在金融数学中用于刻画资产价格的随机演化。该引理为分析和处理依赖时间与随机过程的函数提供了数学基础；可以将其视为“适用于随机过程的链式法则”，是经典微积分链式法则的推广。

在常规微积分中，函数的泰勒展开通常保留到一阶项，高阶项因属于高阶无穷小而可忽略；但在随机过程的微分计算中情形不同，展开需考虑到二阶项。这是因为在随机微分运算中， $(dB_t)^2$  与  $dt$  同阶，即  $(dB_t)^2$  不能被忽略。下文给出的伊藤引理微分法则表可帮助理解与记忆。

|        | $dB_t$ | $dt$ |
|--------|--------|------|
| $dB_t$ | $dt$   | 0    |
| $dt$   | 0      | 0    |

首先，考虑  $dt \cdot dt = 0$ 。在随机微积分中， $dt$  表示极小的时间增量；当计算二次或更高次的时间增量乘积（如  $dt \cdot dt$ ）时，这些乘积属于高阶无穷小，在极限意义下可视为 0。类似地，有  $dt \cdot dB_t = 0$  与  $dB_t \cdot dt = 0$ 。再看  $dB_t \cdot dB_t = dt$ ：依据布朗运动的定义与性质，增量  $dB_t$  的方差与时间间隔  $dt$  成正比，因而  $dB_t \cdot dB_t \approx dt$ 。这些规则是随机分析中的关键工具。

伊藤引理使我们能够计算随机过程  $X_t$  的可微函数  $f(t, X_t)$  的微分。若  $X_t$  为伊藤过程（如布朗运动或含一般漂移与扩散项的过程），则

$$dX_t = \mu(t, X_t) dt + \sigma(t, X_t) dB_t,$$

其中  $B_t$  为标准布朗运动， $\mu$  与  $\sigma$  分别为漂移系数与扩散系数。若  $f(t, x)$  关于  $t, x$  光滑，则  $f$  关于  $X_t$  的伊藤微分为

$$df(t, X_t) = \left( \frac{\partial f}{\partial t} + \mu(t, X_t) \frac{\partial f}{\partial x} + \frac{1}{2} \sigma^2(t, X_t) \frac{\partial^2 f}{\partial x^2} \right) dt + \sigma(t, X_t) \frac{\partial f}{\partial x} dB_t.$$

## 9.5 随机积分

在常规高等数学中，积分被定义为函数曲线下方的面积。然而，由于布朗运动的曲线几乎处处不可导，传统高等数学中的积分定义在此情形并不适用。首先，我们可利用“布朗运动是随机游走的极限”来定义随机积分：

$$\int_0^T dB_t = \lim_{\Delta t \rightarrow 0} [(B_{\Delta t} - B_0) + (B_{2\Delta t} - B_{\Delta t}) + \cdots + (B_T - B_{T-\Delta t})] = B_T - B_0 \sim \mathcal{N}(0, T).$$

上式表明，随机积分的结果是一个随机变量，反映了布朗运动粒子在一段时间内位置的增量，该增量服从正态分布。

不难看出，随机微分方程描述了布朗运动粒子每时每刻位置的变化规律，而随机积分可用于计算任意时刻粒子的位置：

$$\int_0^T dx_t = \mu \int_0^T dt + \sigma \int_0^T dB_t \Rightarrow x_T - x_0 = \mu T + \sigma \int_0^T dB_t,$$

由此可得

$$E_0(x_T - x_0) = \mu T, \quad \text{Var}_0(x_T - x_0) = \sigma^2 T,$$

即：在时点  $T$ ，粒子的位置是一个均值为  $x_0 + \mu T$ 、方差为  $\sigma^2 T$  的正态分布随机变量。

其次，我们可以对随机积分进行更严谨的定义。考虑以下更一般的积分形式：

$$x_t = \int_a^t f(B_s, s) dB_s.$$

**定义 9.6 (随机积分)：**设  $B$  为标准布朗运动。令  $\{f_t, t \in [a, b]\}$  为关于  $\{B_s, s \leq t\}$  可测并适应 (adapted) 于该布朗运动的随机过程，且  $\int_a^b E[f_t^2] dt < \infty$ 。定义

$$I(f) = \int_a^b f_t dB_t = \lim_{n \rightarrow \infty} I_n(f), \quad I_n(f) = \sum_{i=0}^{n-1} f_{t_i} (B_{t_{i+1}} - B_{t_i}),$$

其中  $a = t_0 < t_1 < \dots < t_n = b$  为区间  $[a, b]$  的任意划分，且当  $n \rightarrow \infty$  时，  
 $\max_{0 \leq i \leq n-1} |t_{i+1} - t_i| \rightarrow 0$ ；极限在均方意义下成立，即

$$\lim_{n \rightarrow \infty} E[(I_n(f) - I(f))^2] = 0.$$

与黎曼积分一样，随机积分是线性运算。对于满足上述条件的任意函数  $f, g$  和常数  $\alpha, \beta$ ，有

$$I(\alpha f + \beta g) = \alpha I(f) + \beta I(g).$$

在被积函数  $f$  满足一定条件下，随机积分过程是一个鞅，即对所有  $s < t$ ，几乎必然 (概率为 1) 有

$$E(X_t | \mathcal{F}_s) = E\left(\int_a^t f_u dB_u | \mathcal{F}_s\right) = \int_a^s f_u dB_u = X_s.$$

此外，随机积分满足 Itô 等距性质 (isometry property)

$$E\left(\left(\int_a^b f_t dB_t\right)^2\right) = E\left(\int_a^b E(f_t^2) dt\right).$$

## 9.6 扩散过程

扩散过程通常被定义为一个连续的随机过程，其特点是具有连续的概率密度函数，且该过程的路径在概率意义上是连续的。布朗运动 (或维纳过程) 是最常见的扩散过程。

首先，定义马尔可夫过程 (Markov process)。

**定义 9.7:** 若随机过程  $\{X_t, t \in [0, T]\}$  对所有  $x, t, s$  满足

$$\Pr(X_t \leq x | \mathcal{F}_s) = \Pr(X_t \leq x | X_s),$$

则称其为马尔可夫过程。即在给定  $X_s$  后,  $X_t$  与  $\{X_r : r < s\}$  独立。

以下为扩散过程的定义:

**定义 9.8 (扩散过程):** 扩散过程是具有连续样本路径的连续时间强马尔可夫过程。

强马尔可夫性质是普通马尔可夫性质的扩展: 将固定时间参数替换为停时  $\tau$ 。换而言之, 对于每一个停时  $\tau$ , 在  $\tau < \infty$  条件下, 对所有  $t \geq 0$ , 过程在时刻  $\tau + t$  的状态  $X_{\tau+t}$  在给定  $X_\tau$  的条件下独立于  $\tau$  之前的历史信息, 这意味着过程的未来状态只依赖于停时  $\tau$  的状态。

扩散过程通常通过随机微分方程 (stochastic differential equation, 简称 SDE) 来定义。随机微分方程包括一个确定性的漂移部分 (drift term) 和一个随机的扩散部分 (diffusion term)。

**定理 9.1:** 假设  $X_0 = X$ , 且

$$dX_t = \mu(X_t, t)dt + \sigma(X_t, t)dB_t,$$

其中  $X$  是一个给定的随机变量,  $B_t$  是标准布朗运动。广义扩散过程可以等价地写为

$$X_t = X_0 + \int_0^t \mu(X_s, s)ds + \int_0^t \sigma(X_s, s)dB_s, \quad (9.2)$$

其中第一个积分为普通的黎曼积分, 第二个积分则为随机积分。

**定理 9.2:** 下列条件足以确保方程 (9.2) 存在唯一解  $\{X_t, t \in [0, T]\}$ , 且该解是一个具有连续样本路径的马尔可夫过程, 并满足  $\int E(X_t^2) < \infty$ :

1. **矩条件:**  $E(X^2) < \infty$ ;
2. **Lipschitz 条件:**  $\mu, \sigma$  是 Borel 可测函数, 且存在有限常数  $K$ , 使得对所有的  $x, y \in \mathbb{R}$ ,

$$|\mu(x, t) - \mu(y, t)| \leq K|x - y|, \quad |\sigma(x, t) - \sigma(y, t)| \leq K|x - y|$$

3. **增长条件:** 对某个  $K > 0$  和所有的  $x \in \mathbb{R}$ ,

$$|\mu(x, t)| \leq K(1 + x^2)^{1/2}, \quad |\sigma(x, t)| \leq K(1 + x^2)^{1/2}.$$

以下是一些常见的扩散过程。

- 布莱克-舒尔斯 (Black-Scholes) 模型:

$$dX_t = \beta X_t dt + \sigma X_t dB_t;$$

- Ornstein-Uhlenbeck 过程 (Vasicek 1977) :

$$dX_t = \beta(\alpha - X_t) dt + \sigma dB_t;$$

- 费勒的平方根 (Feller's square root) 过程 (Cox et al. 1985):

$$dX_t = \beta(\alpha - X_t) dt + \sigma\sqrt{X_t} dB_t;$$

- Courtadon (1982):

$$dX_t = \beta(\alpha - X_t) dt + \sigma X_t dB_t;$$

- Marsh & Rosenfeld (1983):

$$dX_t = (\alpha X_t^{-(1-\delta)} + \beta) dt + \sigma X_t^{\delta/2} dB_t;$$

- Cox (1975):

$$dX_t = \beta(\alpha - X_t) dt + \sigma X_t^\gamma dB_t;$$

- Constantinides (1992):

$$dX_t = (\alpha_0 + \alpha_1 X_t + \alpha_2 X_t^2) dt + (\sigma_0 + \sigma_1 X_t) dB_t;$$

- 仿射模型 (affine models) (Duffie & Kan 1996, Dai & Singleton 2000), :

$$dX_t = \beta(\alpha - X_t) dt + (\sigma_0 + \sigma_1 X_t)^{1/2} dB_t;$$

- 非线性均值回归模型 (nonlinear mean reversion) (Aït-Sahalia 1996):

$$dX_t = (\alpha_0 + \alpha_1 X_t + \alpha_2 X_t^2 + \alpha_{-1}/X_t) dt + (\beta_0 + \beta_1 X_t + \beta_2 X_t^{\beta_3}) dB_t.$$

## 9.7 扩散过程的极大似然估计

本节首先展示如何使用极大似然估计来估计扩散模型，并以布莱克-舒尔斯 (Black-Scholes) 模型为例进行详细介绍。随后，在转移密度 (transition density) 未知的情况下，介绍采用 Aït-Sahalia (1996, 2002) 提出的封闭形式似然展开方法 (closed-form likelihood expansions) 对扩散过程参数进行估计。

事实上，任何满足马尔可夫性质的连续金融模型都可以通过最大似然方法进行估计。假设我们有一系列  $n+1$  个历史观测数据  $X_t$ ，在非随机时间点  $t_0 < t_1 < \dots < t_n$  采样，其中采样不必等间隔。为便于说明，这里假设采样间隔为  $\Delta$ ，因此序列可记为  $\{X_0, X_\Delta, \dots, X_{n\Delta}\}$ 。

由此可得样本的联合密度函数：

$$\begin{aligned} & \Pr(X_{n\Delta}, X_{(n-1)\Delta}, \dots, X_0; \theta) \\ &= \Pr(X_{n\Delta} | X_{(n-1)\Delta}, \dots, X_0; \theta) \times \Pr(X_{(n-1)\Delta}, \dots, X_0; \theta) \\ &= \Pr(X_{n\Delta} | X_{(n-1)\Delta}; \theta) \times \Pr(X_{(n-1)\Delta}, \dots, X_0; \theta) \\ &= \Pr(X_{n\Delta} | X_{(n-1)\Delta}; \theta) \times \dots \times \Pr(X_\Delta | X_0; \theta) \times \Pr(X_0; \theta). \end{aligned} \quad (9.3)$$

其中  $p_X(\Delta; X_{i\Delta} | X_{(i-1)\Delta}; \theta)$  为给定  $X_{(i-1)\Delta}$  后  $X_{i\Delta}$  的条件密度函数，也称为**转移密度函数 (transition density)**。根据式 (9.3) 和观测数据  $\{X_0, X_\Delta, \dots, X_{n\Delta}\}$ ，我们可以采用极大似然估计法估计该参数向量。考虑以下连续时间金融模型，

$$dX_t = \mu(X_t; \theta)dt + \sigma(X_t; \theta)dB_t, \quad (9.4)$$

其中  $W_t$  是标准布朗运动。马尔可夫过程  $\{X_t\}$  的转移函数是状态变量  $x$  在固定未来时间  $\Delta$  给定当前状态  $x_0$  的条件密度  $p_X(\Delta, x|x_0; \theta)$ 。

因此，对于式 (9.4) 中的扩散过程，其对数似然函数定义为

$$l_n(\theta) \equiv \sum_{i=1}^n \ln p_X(\Delta; X_{i\Delta} | X_{(i-1)\Delta}; \theta), \quad (9.5)$$

对数似然估计量由下式给出

$$\hat{\theta} = \arg \max_{\theta \in \Theta} l_n(\theta). \quad (9.6)$$

显然，对于任何连续时间金融模型，如果能够确定转移密度  $p_X$ ，则可以采用极大似然估计法对模型参数进行估计。

此外，在适当的正则条件下， $\hat{\theta}$  服从如下极限正态分布：

$$\begin{aligned} \sqrt{n}(\hat{\theta} - \theta) &\xrightarrow{} N(0, \mathcal{I}^{-1}(\theta)), \\ \mathcal{I}(\theta) &= \lim_{n \rightarrow \infty} -E \left[ \frac{1}{n} \frac{\partial^2 l_n(\theta)}{\partial \theta \partial \theta'} \right], \end{aligned}$$

其中  $\mathcal{I}(\theta)$  是信息矩阵 (Information matrix)。当  $n$  很大时，估计量  $\hat{\theta}$  的方差为

$$\text{Var}[\hat{\theta}] \approx \frac{1}{n} \mathcal{I}^{-1}(\theta), \quad (9.7)$$

信息矩阵的估计量为：

$$\hat{\mathcal{I}} = \frac{1}{n} \frac{\partial^2 \mathcal{L}(\hat{\theta})}{\partial \theta \partial \theta'}, \quad (9.8)$$

$\hat{\theta}$  在所有相合且统一渐近正态 (Consistent and Uniformly Asymptotically Normal, 简称 CUAN) 估计量中是最有效的。即  $\hat{\theta}$  不仅相合，而且其标准化误差在一整片参数集合上一致地趋于正态极限。MLE 的渐近协方差达到 Cramér–Rao 下界  $\mathcal{I}(\theta)^{-1}$ ，因此在所有 CUAN 估计量中“最有效”（渐近方差最小）。具体细节参见Campbell et al. (1997, 的 A.4 节)。

### 9.7.1 采用极大似然估计法估计布莱克-舒尔斯 (Black-Scholes) 模型

在布莱克-舒尔斯模型中，股票价格动态由几何布朗运动描述，

$$dX_t = \beta X_t dt + \sigma X_t dB_t. \quad (9.9)$$

Black-Scholes 模型被广泛地用于股票价格和其他金融衍生品的定价中。

将伊藤引理应用于  $\log X_t$ ，并代入式 (9.9) 中的  $dX_t$  可得

$$d \log X_t = \left( \beta - \frac{1}{2} \sigma^2 \right) dt + \sigma dW_t = \alpha dt + \sigma dB_t, \quad (9.10)$$

其中  $\alpha = \beta - \frac{1}{2}\sigma^2$ 。因此，连续复利收益率

$$r_t(\Delta) \equiv \log\left(\frac{X_t}{X_{t-1}}\right)$$

是独立同分布的正态随机变量，其均值为  $\alpha\Delta$ ，方差为  $\sigma^2\Delta$ 。因此  $r_t(\Delta)/\sqrt{\Delta}$  的样本方差也是  $\sigma^2$  的估计量。事实上，它就是  $\sigma$  的极大似然估计量。

在等式 (9.10) 下，连续复利收益率样本  $r_1(\Delta), \dots, r_n(\Delta)$  的极大似然函数为：

$$l_n(\alpha, \sigma) = -\frac{n}{2} \log(2\pi\sigma^2\Delta) - \frac{1}{2\sigma^2\Delta} \sum_{t=1}^n (r_t(\Delta) - \alpha\Delta)^2. \quad (9.11)$$

$\alpha$  和  $\sigma^2$  的极大似然估计量为：

$$\begin{aligned}\hat{\alpha} &= \frac{1}{n\Delta} \sum_{t=1}^n r_t(\Delta), \\ \hat{\sigma}^2 &= \frac{1}{n\Delta} \sum_{t=1}^n (r_t(\Delta) - \hat{\alpha}\Delta)^2.\end{aligned}$$

此外，由于在等式 (9.9) 下  $r_t(\Delta)$  是独立同分布的正态变量，满足一致性和渐近正态性估计量的正则条件（参见 Campbell et al. (1997) 中的第 A.4 节），因此  $\hat{\alpha}$  和  $\hat{\sigma}^2$  在所有一致渐近无偏正态（consistent, asymptotically unbiased normal）估计量中是最渐近有效的，其渐近协方差矩阵见式 (9.7)。

几何布朗运动（Black-Scholes 模型）

$$dX_t = \beta X_t dt + \sigma X_t dB_t$$

具有对数正态分布的转移密度；Ornstein-Uhlenbeck 过程

$$dX_t = \beta (\alpha - X_t) dt + \sigma dB_t$$

具有正态转移密度；而 Feller 平方根过程

$$dX_t = \beta (\alpha - X_t) dt + \sigma \sqrt{X_t} dB_t$$

则具有非中心卡方分布的转移密度。但是，在很多情形下，转移概率  $p_X$  不可知，例如在第9.6章开始时介绍的一些扩散模型，详见 Courtadon (1982)、Marsh & Rosenfeld (1983)、Constantinides (1992)、Duffie & Kan (1996), Dai & Singleton (2000)、Aït-Sahalia (1996) 等文献中引入的扩散模型。

Campbell et al. (1997) 概述了几种规避这一问题的方法，包括广义矩估计法（GMM）和蒙特卡洛模拟等；Pedersen (1995) 提出了模拟似然法。各种近似方法的实施细节可参见 Jensen & Poulsen (1999)。此外，Linton (2019) 第十二章第 4 节介绍了如何采用偏微分方程估计及广义矩估计等方法对扩散模型进行估计。因此，本章侧重介绍极大似然估计方法。

### 9.7.2 Aït-Sahalia (1996, 2002), Aït-Sahalia et al. (2009) 提出的近似方法

本节介绍 Aït-Sahalia (1996, 2002), Aït-Sahalia et al. (2009) 提出的近似方法。为简便起见，仅涵盖单变量情形。

1. 进行两次连续变换  $X \mapsto Y \mapsto Z$ ，使  $Z$  的分布依分布收敛地接近正态分布。
2. 围绕该近似正态的随机变量构建  $P_Z$  的序列。
3. 再做反向还原  $Z \rightarrow Y \rightarrow X$ 。

通常，对固定采样间隔  $\Delta$ ，直接用围绕正态密度的标准级数去近似  $p_X$  往往不可行，因为  $X$  与正态分布差异过大。例如，若  $X$  遵循几何布朗运动，其右尾过厚，Edgeworth 展开可能发散。因此需先施行  $X \mapsto Y \mapsto Z$  变换，使  $Z$  的分布更接近正态，再开展近似。

#### 9.7.2.1 第一次变换： $X \rightarrow Y$

考虑以下通用连续时间金融模型（扩散模型），

$$dX_t = \mu(X_t; \theta) dt + \sigma(X_t; \theta) dB_t. \quad (9.12)$$

定义  $Y_t \equiv \gamma(X_t; \theta) = \int^{X_t} du / \sigma(u; \theta)$ ，通过伊藤引理，可得

$$dY_t = \mu_Y(Y_t; \theta) dt + dB_t, \quad (9.13)$$

其中

$$\mu_Y(y; \theta) = \frac{\mu(\gamma^{-1}(y; \theta); \theta)}{\sigma(\gamma^{-1}(y; \theta); \theta)} - \frac{1}{2} \frac{\partial \sigma}{\partial x}(\gamma^{-1}(y; \theta); \theta).$$

因此消除了 (9.12) 中的  $\sigma(X_t; \theta)$  项，但代价是使漂移项变得非常复杂。

#### 9.7.2.2 第二次变换： $Y \rightarrow Z$

我们将  $Y$  标准化为

$$Z_t \equiv \Delta^{-1/2} (Y_t - y_0).$$

注意这里不需要  $\Delta \rightarrow 0$ ，因为此处的标准化对精确度没有具体要求；只要  $Z_t$  “足够接近”一个正态随机变量，该近似的效果就很好。我们使用 Hermite 多项式来近似  $p_Z$ ，

$$H_j(z) \equiv \exp(-z^2/2) \frac{d^j}{dz^j} [\exp(-z^2/2)],$$

其中  $z$  服从标准正态密度，即  $\phi(z) \equiv \exp(-z^2/2) / (2\pi)^{1/2}$ 。 $Z$  的密度函数是在阶数  $J$  下  $z$  的 Hermite 展开。在给定  $\Delta$ 、 $y_0$  和  $\theta$  时，该展开为：

$$p_Z^{(j)}(\Delta, z | y_0; \theta) \equiv \phi(z) \sum_{j=0}^J \eta_j(\Delta, y_0; \theta) H_j(z).$$

Hermite 展开中的未知数是系数  $\eta_j$ 。根据 Hermite 多项式的正交性，可得

$$\eta_j (\Delta, y_0; \theta) \equiv \frac{1}{j!} \int_{-\infty}^{\infty} H_j(z) P_Z(\Delta, z | y_0; \theta) dz.$$

这意味着  $\eta_j$  是在密度  $p_Z$  上的期望，因此可以对这些系数进行测算。

假设  $p_Z^{(J)}$  已知，根据  $p_Z^{(J)}$ ，我们可得到一系列  $p_Y$  的近似，

$$p_Z^{(J)}(\Delta, y | y_0; \theta) \equiv \Delta^{-1/2} p_Z^{(J)}(\Delta, \Delta^{-1/2}(y - y_0) | y_0; \theta);$$

进而得到  $p_X$ ，即

$$p_X^{(J)}(\Delta, x | x_0; \theta) \equiv \sigma(x; \theta)^{-1} p_Y^{(J)}(\Delta, \gamma(x; \theta) | \gamma(x; \theta); \theta).$$

### 9.7.2.3 第三次变换： $Y \rightarrow X$

令  $Y_t \equiv \gamma(X_t; \theta) = \int^{X_t} du / \sigma(u; \theta)$ ，即  $Y$  是  $X$  的非线性变换<sup>2</sup>。首项  $p_X^{(J)}$  通常不服从正态分布，因为它是服从正态分布变量的非线性函数。

**定理 9.3：** 存在一个  $\bar{\Delta} > 0$ ，使得对于每一个  $\Delta \in (0, \bar{\Delta})$ ， $\theta \in \Theta$  和  $(x, x_0) \in D_X^2$ ：

$$p_X^{(J)}(\Delta, x | x_0; \theta) \rightarrow p_X(\Delta, x | x_0; \theta) \text{ 随着 } J \rightarrow \infty.$$

此外，这种收敛在  $\Theta$  上关于  $\theta$  是均匀的，在  $D_X$  上关于  $x$  是均匀的，且在  $D_X$  的任意紧子集上关于  $x_0$  是均匀的。通过最大化

$$l_n^{(J)} \equiv \sum_{i=1}^n \ln p_X^{(J)}(\Delta, X_{i\Delta} | X_{(i-1)\Delta}; \theta)$$

得到的估计量  $\hat{\theta}_n^{(J)}$ ，随着  $J \rightarrow \infty$  收敛到真实的（但无法计算的）极大似然估计  $\hat{\theta}_n$ ，并继承了该极大似然估计的所有渐近性质。

具体细节请参见 Aït-Sahalia (2002)。

### 9.7.2.4 极大似然估计

那么问题来了，如何获得  $p_Z^{(J)}$  呢？显然，随着 Hermite 多项式的数量  $J \rightarrow \infty$ ，近似的精度会逐渐提高。

$$p_Z^{(\infty)}(\Delta, z | y_0; \theta) = \phi(z) \sum_{j=0}^{\infty} \eta_j(\Delta, y_0; \theta) H_j(z).$$

为了计算系数  $\eta_j(\Delta, y_0; \theta)$ ，我们对  $\eta_j$  在  $\Delta$  处进行泰勒展开，展开至  $\Delta^K$  阶。我们称之为  $\tilde{p}_Z^{(K)}$  的展开。

<sup>2</sup>除非  $\sigma$  是常数参数，在这种情况下  $Y_t = X_t / \sigma$ 。

**定理 9.4:**

$$\tilde{p}_Z^{(K)}(\Delta, z | y_0; \theta) = \Delta^{-1/2} \phi\left(\frac{y - y_0}{\Delta^{1/2}}\right) \exp\left(\int_{y_0}^y \mu_Y(\varpi; \theta) d\varpi\right) \sum_{k=0}^K c_k(y | y_0; \theta) \frac{\Delta^k}{k!}.$$

可得系数的封闭形式 (closed form) 表达式：

$$\begin{aligned} c_0(y | y_0; \theta) &= 1 \\ c_j(y | y_0; \theta) &= j(y - y_0)^{-j} \int_{y_0}^y (\varpi - y_0)^{j-1} \left\{ \lambda_Y c_{j-1}(\varpi | y_0; \theta) + \frac{1}{2} \frac{\partial^2 c_{j-1}(\varpi | y_0; \theta)}{\partial \varpi^2} \right\} d\varpi \\ \lambda_Y(y; \theta) &= -\frac{1}{2} \left( \mu_Y^2(y; \theta) + \frac{\partial \mu_Y(y; \theta)}{\partial y} \right). \end{aligned}$$

Aït-Sahalia (2002) 提出的方法相较其他近似方法具有更好的表现，详见其论文中的图 1。

#### 9.7.2.5 连续时间序列模型的模型设定检验

当在众多候选中选择模型时，一个自然思路是选取最能生成数据所蕴含  $\mu$  与  $\sigma$  关系的模型。Aït-Sahalia (1996) 一方面利用漂移项与扩散项之间的映射关系，另一方面以边缘密度  $\pi_X$  与转移密度  $p_X$  为基础来检验模型是否被正确设定 (correctly specified)。由于真正的高频连续轨迹通常不可得，Aït-Sahalia (1996) 使用的是离散采样下的密度  $(\pi_X, p_X)$ ，而非连续过程参数  $(\mu, \sigma^2)$ ； $\mu$  与  $\sigma^2$  均可看作与  $\pi_X, p_X$  相联系的参数。例：Ornstein–Uhlenbeck 过程  $dX_t = \beta(\alpha - X_t) dt + \sigma dB_t$  具有高斯 (正态) 形式的边缘密度与转移密度；Feller 平方根 (CIR) 过程  $dX_t = \beta(\alpha - X_t) dt + \sigma \sqrt{X_t} dW_t$  产生伽马形式的边缘密度与非中心卡方形式的转移密度。

Hong & Li (2005) 同样围绕转移密度构建设定检验：在原假设 (模型正确设定) 下，序列  $\{P_X(X_{i\Delta} | X_{(i-1)\Delta}, \Delta, \theta)\}$  为独立同分布的均匀随机变量。

Aït-Sahalia et al. (2009) 则将非参数估计的转移密度函数与假设的参数转移密度函数直接比较：

$$\begin{aligned} H_0: \quad p_X(y | x, \Delta) &= p_X(y | x, \Delta, \theta), \\ \text{vs. } H_1: \quad p_X(y | x, \Delta) &\neq p_X(y | x, \Delta, \theta). \end{aligned}$$

Chen et al. (2008) 采用类似思路，使用基于核函数的非参数方法估计转移密度。

#### 9.7.3 案例：基于 Aït–Sahalia 似然展开的扩散模型近似极大似然估计与稳健推断

连续时间序列模型的估计与检验可使用 R 软件包 MLEMVD 实现，详见：<https://rdrr.io/github/mfrdixon/MLEMVD/>。值得注意的是，MLEMVD 不在 CRAN 上，需要输入以下代码进行安装：

```
1 install.packages("remotes")
2 remotes::install_github("mfrdixon/MLEMVD")
```

使用 GitHub 安装 R 软件包的优点在于：开发者可快速推送更新与修复，用户可即时获取最新版本，无需等待 CRAN 审核。GitHub 上通常还托管包的开发版本，便于在现有源码基础上改进，并扩展至不同模型与估计方法。

本节代码演示在无法获得闭式转移密度时，如何基于 Aït–Sahalia 的封闭形式似然展开对单变量扩散模型实施极大似然估计，并提供稳健标准误。数据方面，代码默认从 Yahoo

Finance 抓取 SPY 的日频复权收盘价（起始于 2015-01-01），以价格“层级”入模并采用日频步长  $\Delta = 1/252$ ；若读者已有自有数据，只需替换 `price` 向量。方法方面，先对价格序列做统一缩放以改善数值条件，然后拟合三次漂移、常数扩散的近似模型（记为 U6），通过对全样本过渡对数似然求和进行优化，并由近似信息矩阵计算协方差与标准误；当信息矩阵近奇异或存在边界粘连导致标准误不可得时，改用数值 Hessian 兜底。考虑到高阶漂移在有限样本下的潜在不稳定，代码同时提供二次漂移、常数扩散的备选模型（记为 U4）：先进行随机可行起点搜索以确保全样本似然函数有限，再使用无导数的 Subplex 算法优化，最终按与 U6 相同规则输出估计值与标准误。整套流程的要点在于：通过缩放与边界设置稳定似然函数展开，通过可行起点与无导数优化提升收敛鲁棒性，并以“信息矩阵优先、数值 Hessian 兜底”的推断口径确保结果可报告且口径统一（如  $\Delta = 1/252$ 、缩放/还原与单位解释保持一致）。

```

1 # =====
2 # 扩散模型的 Ait-Sahalia 似然展开：U6 + U4 稳健估计与标准误
3 # - U6: 三次漂移 + 常数扩散（近似 MLE）
4 # - U4: 二次漂移 + 常数扩散（当 U6 条件数差/边界粘连时的稳健备选）
5 # - 统一对价格做一次缩放，改善数值稳定性
6 # 依赖: MLEMVD, nloptr, MASS, numDeriv, quantmod(可选取数)
7 # =====
8
9 suppressPackageStartupMessages({
10 library(MLEMVD)
11 library(nloptr)
12 library(MASS) # ginv()
13 library(numDeriv) # hessian()
14 library(quantmod) # 取数用（如已自备 price，可跳过）
15 })
16
17 # -----
18 # 0) 数据（如已自备 price，可注释本段）
19 #
20 if (!exists("price")) {
21 getSymbols("SPY", src = "yahoo", from = "2015-01-01")
22 price <- as.numeric(Ad(SPY)) # GBM/扩散以价格“层级”拟合，而非收益率
23 }
24 n <- length(price)
25 del <- 1/252 # 日频步长的年化口径
26
27 #
28 # 1) 统一缩放（数值稳定性关键）
29 #
30 scale_factor <- 1000
31 price_s <- price / scale_factor
32
33 #
34 # 2) 适配器（让 mle() 可传入 args 而不报错）
35 #
36 ModelU6_with_args <- function(x, x0, del, param, args = NULL) MLEMVD::
37 ModelU6(x, x0, del, param)
38 ModelU4_with_args <- function(x, x0, del, param, args = NULL) MLEMVD::
```

```

 ModelU4(x, x0, del, param)
38
39 # -----
40 # 3) 工具函数 (一次性定义)
41 # -----
42 # (a) 将“过渡对数似然”在全样本上求和 (便于数值 Hessian)
43 total_llik <- function(logdensity_fun, param, x, del) {
44 s <- 0
45 for (i in 1:(length(x) - 1)) s <- s + logdensity_fun(x[i+1], x[i], del,
46 param)$llk
47 s
48 }
49
50 # (b) 由信息矩阵求协方差: 对称化 + 微小岭回归 + 不可逆则用广义逆
51 safe_vcov_from_I <- function(I, n) {
52 I <- 0.5 * (I + t(I)) # 对称化
53 I <- I + 1e-10 * diag(nrow(I)) # 加微小对角岭
54 tryCatch(solve(I) / n, error = function(e) MASS::ginv(I) / n)
55 }
56
57 # (c) 生成与参数维度匹配的上下界 (最后一维为扩散 f)
58 make_args <- function(param0, f_upper = 50, maxeval = 1500,
59 method = c("LBFGS", "SBPLX"), deoptim_iter = 0, print_
60 level = 0) {
61 method <- match.arg(method)
62 p <- length(param0)
63 l <- rep(-1, p); u <- rep(1, p)
64 l[p] <- 1e-8; u[p] <- f_upper
65 algo <- if (method == "LBFGS") "NLOPT_LD_LBFGS" else "NLOPT LN SBPLX"
66 list(
67 mode = "direct",
68 nloptr = list(
69 method = algo,
70 maxeval = maxeval, # 注意: 此处用 maxeval 更稳妥
71 xtol_rel = 1e-8,
72 ftol_rel = 1e-10,
73 ftol_abs = 0,
74 print_level = print_level,
75 l = l, u = u
76),
77 DEoptim = list(maxiter = deoptim_iter, population = 80, strategy = 2),
78 eval_g_ineq = NULL,
79 eval_jac_g_ineq = NULL
80)
81 }
82
83 # (d) 全样本起点可行性检查 (目标是否有限)
84 series_ll_finite <- function(logdensity_fun, x, del, par) {
85 s <- 0
86 for (i in 1:(length(x) - 1)) {
87 li <- try(logdensity_fun(x[i+1], x[i], del, par)$llk, silent = TRUE)

```

```

86 if (!is.numeric(li) || !is.finite(li)) return(FALSE)
87 s <- s + li
88 }
89 is.finite(s)
90 }
91
92 # -----
93 # 4) U6: 三次漂移 + 常数扩散 (近似 MLE)
94 # —— 常规一阶法 (LBFGS) , 失败再用数值 Hessian 兜底
95 #
96 start_u6 <- c(a = 0, b = 0, c = 0, d = 0, f = 0.05)
97 args_u6 <- make_args(start_u6, f_upper = 50, maxeval = 1500, method = "
98 LBFGS", deoptim_iter = 0)
99
100 cat("\n==== U6 (近似 MLE, 三次漂移, 常数扩散) ====\n")
101 fit_u6 <- mle(
102 logdensity = ModelU6_with_args,
103 x = price_s,
104 del = del,
105 param0 = start_u6,
106 args = args_u6
107)
108 theta_u6 <- setNames(as.numeric(fit_u6$solution), names(start_u6))
109 print(theta_u6)
110
111 # 信息矩阵 → 协方差 → 标准误
112 I_u6 <- as.matrix(logdensity2info(
113 logdensity = ModelU6_with_args, x = price_s, del = del, param = theta_u6
114))
115 vcov_u6 <- safe_vcov_from_I(I_u6, n)
116 se_u6 <- sqrt(diag(vcov_u6))
117
118 # 若仍数值不稳, 改用数值 Hessian 兜底
119 if (any(!is.finite(se_u6))) {
120 cat(" (提示) U6 信息矩阵条件数差, 改用数值 Hessian.\n")
121 H_u6 <- numDeriv::hessian(function(p) -total_loglik(ModelU6_with_args, p
122 , price_s, del), theta_u6)
123 vcov_u6 <- safe_vcov_from_I(H_u6, n)
124 se_u6 <- sqrt(diag(vcov_u6))
125 }
126 cat("\nU6 估计与标准误: \n")
127 print(round(cbind(Estimate = theta_u6, Std.Error = se_u6), 6))
128
129 # -----
130 # 5) U4: 二次漂移 + 常数扩散 (稳健备选)
131 # —— 强化缩放后, 随机可行起点 + 无导数优化 (Subplex)
132 #
133 # 随机搜索一个 “全样本目标有限” 的可行起点 (漂移均匀 [-0.8,0.8], f 为 log-
134 uniform)
135 set.seed(123)

```

```

134 best_par <- NULL; best_ll <- -Inf
135 for (k in 1:500) {
136 a <- runif(1, -0.8, 0.8)
137 b <- runif(1, -0.8, 0.8)
138 c <- runif(1, -0.8, 0.8)
139 f <- exp(runif(1, log(1e-3), log(50)))
140 par <- c(a=a,b=b,c=c,f=f)
141 if (series_ll_finite(ModelU4_with_args, price_s, del, par)) {
142 ll <- total_loglik(ModelU4_with_args, par, price_s, del)
143 if (ll > best_ll) { best_ll <- ll; best_par <- par }
144 }
145 }
146 if (is.null(best_par)) stop("未能找到 U4 的可行起点：可进一步增大 scale_
147 factor 或放宽边界。")
148 start_u4 <- best_par
149 message("U4 可行起点：", paste(round(start_u4, 6), collapse = ", "))
150 # 无导数优化 (Subplex)，并限制评估次数避免过慢
151 args_u4 <- make_args(start_u4, f_upper = 100, maxeval = 2500, method =
152 "SBPLX", deoptim_iter = 0, print_level = 1)
153 cat("\n==== U4 (稳健近似 MLE, 二次漂移, 常数扩散) ====\n")
154 fit_u4 <- mle(
155 logdensity = ModelU4_with_args,
156 x = price_s,
157 del = del,
158 param0 = start_u4,
159 args = args_u4
160)
161
162 theta_u4 <- setNames(as.numeric(fit_u4$solution), c("a", "b", "c", "f"))
163 print(theta_u4)
164
165 # 信息矩阵或数值 Hessian 计算标准误
166 I_u4_try <- try(as.matrix(logdensity2info(
167 logdensity = ModelU4_with_args, x = price_s, del = del, param = theta_u4
168)), silent = TRUE)
169
170 if (inherits(I_u4_try, "try-error") || any(!is.finite(I_u4_try))) {
171 cat(" (提示) U4 信息矩阵不可用/不稳，改用数值 Hessian。 \n")
172 H_u4 <- numDeriv::hessian(function(p) -total_loglik(ModelU4_with_args, p
173 , price_s, del), theta_u4)
174 vcov_u4 <- safe_vcov_from_I(H_u4, n)
175 } else {
176 vcov_u4 <- safe_vcov_from_I(I_u4_try, n)
177 }
178 se_u4 <- sqrt(diag(vcov_u4))
179 cat("\nU4 估计与标准误：\n")
180 print(round(cbind(Estimate = theta_u4, Std.Error = se_u4), 6))
181

```

182

```
cat("\n完成。\\n")
```

## 9.8 从高频数据中估计波动率

首先，我们考虑一段时间内的累积波动率。

**定义 9.9 (二次变差):** 对于一个平方可积的过程  $X_t$ , 其二次变差 (*quadratic variation*) 为:

$$\langle X, X \rangle_{0:t} = \text{plim}_{\max x(t_{k+1}-t_k) \rightarrow 0} \sum_{t_k \leq t} |X_{t_{k+1}} - X_{t_k}|^2,$$

其中  $0 = t_1 < t_2 < \dots < t_n = t$ 。

后续，我们将二次变差简写为 QV，即英文 quadratic variation 的两个首字母缩写。

对于有界变差的函数 (定义9.3)，其 QV 存在且为零。此外，Andersen et al. (2003) 指出，在一些相当普遍的条件下

$$E((X(t+h) - X(t))^2 | \mathcal{F}_t) = E(\langle X, X \rangle_{t:t+h} | \mathcal{F}_t),$$

即回报的条件方差等于同一时间间隔上 QV 的条件期望。换言之，可以用 QV 来估计回报波动率。

针对一般扩散过程

$$dX_t = \mu(X_t) dt + \sigma(X_t) dB_t,$$

其二次变差  $\langle X \rangle_t$  一般为随机过程；若  $\sigma^2(X_t) \equiv \sigma^2$  (常数)，则  $\langle X \rangle_t = t\sigma^2$ 。此外， $\langle X, X \rangle_t$  对于连续平方可积的半鞅过程总是存在。下面引入局部鞅 (local martingale) 与半鞅 (semi-martingale)。

**定义 9.10 (局部鞅):** 若存在一列停时  $(\tau_k)_{k \geq 1}$  使得  $\tau_k \uparrow \infty$  a.s.，且对每个  $k$ ，停止过

$$M_t^{\tau_k} := M_{t \wedge \tau_k}$$

为鞅，则称适应过程  $M$  为局部鞅。

每一个鞅都是局部鞅；每一个有界的局部鞅都是鞅，但并非每个局部鞅都是鞅。无漂移项的某些扩散过程是局部鞅，但未必是鞅。

**定义 9.11 (半鞅):** 一个实值过程  $X$  称为半鞅，若可分解为

$$X(t) = M(t) + A(t),$$

其中  $M$  为局部鞅， $A$  为适应过程，具有局部有限 (有界) 变差的样本路径，且样本路径右连续并具有左极限 (*càdlàg*)。

过程  $A$  的二次变差 (quadratic variation) 为零；本质上它比  $M$  变动得更慢，因此其的可预测性帮助不大。半鞅类过程可以定义随机积分，是可以进行随机积分的过程类。半鞅过程包含所有的鞅过程。此外，半鞅过程还包括由布朗运动驱动的扩散过程以及具有跳跃的过程。Girsanov 定理适用于半鞅。资产定价基本定理表明，无套利意味着存在等价鞅测度；因此若资产价格为半鞅，则不存在套利机会。

我们现在考虑 QV 的一个一致估计量—已实现波动率 (realized volatility, 简称 RV)。假设我们有一个在间隔  $[0, 1]$  内等间区间观测的  $n$  个对数价格样本, 记为  $X_t$ , 则有

$$RV_X^n = \sum_{l=1}^{n-1} \left( X_{\frac{t+1}{n}} - X_{\frac{t}{n}} \right)^2$$

这一在  $[0, 1]$  上  $X$  的二次变差 (QV) 的一致估计。[Jacod & Protter \(1998\)](#) 建立了  $RV_X^n$  关于伊藤类半鞅的中心极限定理。[Barndorff-Nielsen & Shephard \(2002\)](#) 则推导了布朗运动类半鞅的一致性和极限分布。

**定义 9.12 (布朗半鞅):** 布朗半鞅  $X$  满足

$$X_t = \int_0^t \mu_v dv + \int_0^t \sigma_v dB_v,$$

其中过程  $\mu, \sigma$  可预见 (仅依赖于过去), 且  $\sigma$  为右连续并具有左极限的 *càdlàg* 过程。

[Barndorff-Nielsen & Shephard \(2002\)](#) 在无杠杆 (no-leverage) 情形下——即  $\mu, \sigma$  与驱动布朗运动  $B$  相互独立——证明

$$\sqrt{n} (RV_X^n - QV) \xrightarrow{d} MN \left( 0, 2 \int_0^1 \sigma_u^4 du \right) =: MN(0, 2IQ), \quad IQ := \int_0^1 \sigma_u^4 du.$$

并给出  $IQ$  的一致估计量

$$\widehat{IQ} = \frac{n}{3} \sum_{i=1}^n (\Delta_i^n X)^4, \quad \Delta_i^n X := X_{i/n} - X_{(i-1)/n}.$$

由此得到中心极限定理

$$\frac{\sqrt{n} (RV_X^n - QV)}{\sqrt{2 \widehat{IQ}}} \xrightarrow{d} N(0, 1). \quad (9.14)$$

式 (9.14) 可用于构造波动率的置信区间并进行假设检验。

## 9.9 测量误差模型

由于市场摩擦和交易基础设施的局限性, 高频金融数据中存在市场微观结构噪声。该噪声通常体现在交易与报价数据的离散性上, 其成因包括交易员的行为策略、交易系统的执行延迟、最小报价单位的限制以及交易场所的规则等因素, 这些因素都会影响价格的连续记录。

市场微观结构噪声会对基于高频数据的统计推断和模型估计带来挑战, 尤其是在波动率估计、相关性分析和市场效率评估中。波动率签名图 (volatility signature plot) 用于展示波动率估计量随时间尺度 (通常指交易或采样频率) 变化的规律。由于微观结构噪声, 在高频率采样下, 已实现波动率 (realized volatility, RV) 的估计会发散至无穷大, 参见 [Linton \(2019, 第 454 页, 图 12.4\)](#)。

显然, 若未适当处理微观结构噪声, 基于高频数据的结论可能产生偏差, 并导致对市场动态的误解。

以下示例使用纽约证券交易所 (NYSE) TAQ 数据库中美国铝业公司 (英文名: Alcoa Corporation, 简称 AA) 2012 年 1 月 3 日的逐笔交易与报价数据。

```

1 2012 1 3 79 8.96 32599 1 8.95 8.96
2 2012 1 3 80 8.96 100 1 8.95 8.96
3 2012 1 3 83 8.95 400 -1 8.95 8.96
4 2012 1 3 85 8.96 5300 1 8.95 8.96
5 2012 1 3 87 8.96 2200 1 8.95 8.96
6 2012 1 3 89 8.955 3600 1 8.95 8.96
7 2012 1 3 94 8.95 200 -1 8.95 8.96
8 2012 1 3 96 8.95 800 -1 8.95 8.96
9 2012 1 3 98 8.96 7400 1 8.95 8.96
10 2012 1 3 109 8.96 6764 1 8.95 8.96
11 2012 1 3 115 8.96 100 1 8.95 8.96
12 2012 1 3 118 8.96 100 1 8.95 8.97
13 2012 1 3 120 8.96 100 1 8.95 8.97
14 2012 1 3 124 8.96 100 -1 8.955 8.97
15 2012 1 3 139 8.97 6200 1 8.96 8.97
16 2012 1 3 140 8.98 200 1 8.96 8.98
17 2012 1 3 150 8.97 200 -1 8.97 8.98

```

其中每列数据的意义如下：

1. **Year (年份)**：数据记录的年份，这里为 2012 年。
2. **Month (月份)**：数据记录的月份。
3. **Day (日期)**：数据记录的日期，表示对应当月的具体日期。
4. **Timestamp (时间戳)**：记录交易发生的确切时间点。
5. **Price (交易价格)**：即在该时间点记录的交易发生时的价格。
6. **Volume (交易量)**：表示在该时间点成交的股票数量。
7. **Direction (交易方向)**：+1 表示买入，-1 表示卖出。
8. **Bid Price (买入价)**：通常指买方愿意支付的最高价格。
9. **Ask Price (卖出价)**：通常指卖方愿意接受的最低价格。

假设我们观察以下对数价格序列  $\{Y_{t_j}\}_{j=1}^n$ ,

$$Y_{t_j} = X_{t_j} + \varepsilon_{t_j}. \quad (9.15)$$

其中， $\varepsilon_{t_j}$  为独立同分布 (i.i.d.) 的随机变量，满足  $E[\varepsilon_{t_j}] = 0$ 、 $\text{Var}(\varepsilon_{t_j}) = \sigma_\varepsilon^2$ ，且测量误差过程  $\varepsilon$  与过程  $X$  独立。

我们将误差项  $\varepsilon_{t_j}$  视为简化的市场微观结构噪声，参见 Zhang et al. (2005)。长期来看  $X_{t_j}$  占主导地位；但在短期内，测量误差  $\varepsilon$  可能在已实现波动率 (RV) 中占主导。

基于式(9.15)，可得 RV 估计量：

$$\begin{aligned} RV_Y^n &= \sum_{i=1}^{n-1} \left( Y_{\frac{i+1}{n}} - Y_{\frac{i}{n}} \right)^2 \\ &= \sum_{i=1}^{n-1} \left( X_{\frac{i+1}{n}} - X_{\frac{i}{n}} \right)^2 + \sum_{i=1}^{n-1} \left( \varepsilon_{\frac{i+1}{n}} - \varepsilon_{\frac{i}{n}} \right)^2 + 2 \sum_{i=1}^{n-1} \left( \varepsilon_{\frac{i+1}{n}} - \varepsilon_{\frac{i}{n}} \right) \left( X_{\frac{i+1}{n}} - X_{\frac{i}{n}} \right). \end{aligned}$$

基于大数定律，可得

$$\frac{1}{n} \sum_{t=1}^{n-1} \left( \varepsilon_{\frac{t+1}{n}} - \varepsilon_{\frac{t}{n}} \right)^2 \xrightarrow{p} 2\sigma_\varepsilon^2.$$

基于柯西-施瓦茨 (Cauchy-Schwarz) 不等式，可得

$$\left| \frac{1}{n} \sum_{i=1}^{n-1} \left( \varepsilon_{\frac{i+1}{n}} - \varepsilon_{\frac{i}{n}} \right) \left( X_{\frac{i+1}{n}} - X_{\frac{i}{n}} \right) \right| \leq \frac{1}{\sqrt{n}} \left( \frac{1}{n} \sum_{i=1}^{n-1} \left( \varepsilon_{\frac{i+1}{n}} - \varepsilon_{\frac{i}{n}} \right)^2 \right)^{1/2} \left( \sum_{l=1}^{n-1} \left( X_{\frac{l+1}{n}} - X_{\frac{l}{n}} \right)^2 \right)^{1/2} \xrightarrow{p} 0.$$

不难看出，当  $n \rightarrow \infty$  时，

$$RV_Y^n \xrightarrow{p} \infty.$$

即，RV 不是波动率的一致估计量。

为减弱市场微观结构噪声的影响，可采用较低频率的数据估计已实现波动率 (RV)。此外，Zhang et al. (2005) 提出两尺度已实现波动率 (two scales realized volatility, 简称 TSRV) 估计量：将两种不同采样尺度（高频与低频）下计算的已实现方差加权结合，得到对二次变差的一致估计，并通过加性偏差修正，有效消除由微观结构噪声引入的偏差。

**定义 9.13：**设  $K \times (m + 1) = n$ ，令第一个子样本为  $\{Y_0, Y_{K/n}, \dots, Y_{mK/n}\}$ ，第二个子样本为  $\{Y_{1/n}, Y_{(K+1)/m}, \dots, Y_{(mK+1)/n}\}$ ，以此类推。在每个子样本中，我们有  $m + 1$  个（对数）价格，可得  $m$  个收益率。对于  $j = 1, \dots, K$ ，我们有

$$RV_{sub,j} = \sum_{i=1}^m \left( Y_{\frac{(j+iK)}{m}} - Y_{\frac{(j+(i-1)K)}{m}} \right)^2.$$

不难看出， $RV_{sub,j}$  是基于较低频率或较慢时间尺度计算的实际波动率。在没有微观结构噪声的情况下，该估计量在  $m \rightarrow \infty$  时是一致的。

当存在微观结构噪声时，

$$\begin{aligned} \frac{1}{m} RV_{sub} &= \frac{1}{m} \sum_{k=1}^n \left( Y_{\frac{(j+iK)}{n}} - Y_{\frac{(j+(i-1)K)}{n}} \right)^2 \\ &= \frac{1}{m} \sum_{i=1}^m \left( X_{\frac{(j+iK)}{n}} - X_{\frac{(j+(i-1)K)}{n}} \right)^2 + \frac{1}{m} \sum_{i=1}^m \left( \varepsilon_{\frac{(j+iK)}{n}} - \varepsilon_{\frac{(j+(i-1)K)}{n}} \right)^2 \\ &\quad + \frac{2}{m} \sum_{i=1}^m \left( X_{\frac{(j+iK)}{n}} - X_{\frac{(j+(i-1)K)}{n}} \right) \left( \varepsilon_{\frac{(j+iK)}{n}} - \varepsilon_{\frac{(j+(i-1)K)}{n}} \right) \\ &\xrightarrow{p} 2\sigma_\varepsilon^2 \quad (\text{当 } m \rightarrow \infty \text{ 时}). \end{aligned}$$

上式表明，在存在测量误差（见式 (9.15)）时， $RV_{sub}$  的估计是不一致的，随  $m$  的增大而发散。为改进估计效果，采用两尺度已实现波动率 (RV) 的线性组合

$$RV_{sub,j} - \frac{m}{n} RV.$$

因此主项的偏差为

$$2m\sigma_\varepsilon^2 - \frac{m}{n} \cdot 2n\sigma_\varepsilon^2 = 0.$$

这意味着可以将全样本与子样本的估计量按上述方式进行线性组合，从而消除偏差项。

此时的主导项为

$$\sum_{i=1}^m \left\{ \left( \varepsilon_{\frac{(j+iK)}{n}} - \varepsilon_{\frac{(j+(i-1)K)}{n}} \right)^2 - 2\sigma_\varepsilon^2 \right\} = \sqrt{m} \times \frac{1}{\sqrt{m}} \sum_{i=1}^m \left\{ \left( \varepsilon_{\frac{(j+iK)}{n}} - \varepsilon_{\frac{(j+(i-1)K)}{n}} \right)^2 - 2\sigma_\varepsilon^2 \right\},$$

该项由噪声驱动。对  $K$  个子样本作平均，得到

$$\begin{aligned} T &= \frac{1}{K} \sum_{j=1}^K \sum_{i=1}^m \left( \left( \varepsilon_{\frac{(j+iK)}{n}} - \varepsilon_{\frac{(j+(i-1)K)}{n}} \right)^2 - 2\sigma_\varepsilon^2 \right) \\ &= \frac{1}{K} \sum_{j=1}^K \sum_{l=1}^m \left( \varepsilon_{\frac{(j+iK)}{n}}^2 - \sigma_\varepsilon^2 \right) + \frac{1}{K} \sum_{j=1}^K \sum_{l=1}^m \left( \varepsilon_{\frac{(j+(i-1)K)}{n}}^2 - \sigma_\varepsilon^2 \right) \\ &\quad - 2 \frac{1}{K} \sum_{j=1}^K \sum_{l=1}^m \varepsilon_{\frac{(j+iK)}{n}} \varepsilon_{\frac{(j+(i-1)K)}{n}}, \end{aligned}$$

上述三项的期望均为零（利用  $\varepsilon$  独立同分布且  $E\varepsilon = 0$ 、 $\text{Var}(\varepsilon) = \sigma_\varepsilon^2$ ）。此外，每一项都可视为  $n$  个独立随机变量的和，因此  $T$  的概率阶为  $\sqrt{m/K}$ ；在适当中心化并按方差归一化后满足中心极限定理。若  $K/m \rightarrow \infty$ ，这些噪声项的概率阶更小。由此即可构造 TSRV 估计量。

**定义 9.14 (TSRV 估计量):** TSRV 估计量定义为

$$\hat{\theta}_{\text{TSRV}} = \frac{1}{K} \sum_{j=1}^K RV_{\text{sub},j} - \frac{m}{n} RV_n.$$

以下 R 代码用来测算 TSRV 估计量。

```

1 rm (list=ls ())
2 set.seed (123) # 设置随机种子以获得可复现的结果
3
4 # 设置工作目录
5 if (requireNamespace("rstudioapi", quietly = TRUE) &&
6 rstudioapi::isAvailable()) {
7 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
8 }
9
10 library (readxl)
11 # 读取Excel文件, 不假定第一行为列名
12 aa <- read_excel ("AA_daily20120103.xlsx", col_names = FALSE)
13
14 TSRV <- function (priceX, K, m, n) {
15 # 计算价格的对数
16 lprice <- log (priceX)
17 # 初始化一个向量, 用于存储每个子样本的实际波动率
18 RV_sub <- numeric (K)
19 # 为每个子样本计算RV_sub
20 for (j in 1:K) {
21 # 计算每个子样本的价格序列
22 sub_sample <- lprice[seq (j, by = K, length.out = m + 1)]
23 # 计算收益率 (对数价格差)
```

```

24 returns <- diff (sub_sample)
25 # 计算子样本的实际波动率
26 RV_sub[j] <- sum (returns^2)
27 }
28 # 计算全样本的实际波动率
29 returns_full <- diff (lprice)
30 RV_n <- sum (returns_full^2)
31 # 计算TSRV估计量
32 TSRV_estimator <- (1 / K) * sum (RV_sub) - (m / n) * RV_n
33 # 返回TSRV估计量
34 return (TSRV_estimator)
35 }
36
37 priceX <- aa$...5 # 假设第5列包含价格数据
38 K <- 5 # 子样本的数量
39 m <- 100 # 每个子样本中的观测值数量
40 n <- length (priceX) # 数据的总观测数
41
42 # 计算TSRV估计量
43 TSRV_estimate <- TSRV (priceX, K, m, n)
44 TSRV_estimate
45

```

Zhang (2006) 提出了多尺度已实现波动率 (multi-scale realized volatility, 简称 MSRV) 估计量。MSRV 通过结合数量约为  $\simeq n^{1/2}$  的不同时间尺度 (或采样频率) 下计算的已实现方差, 来估计二次变差, 并进行加性偏差修正, 从而在降低噪声与偏差的同时, 提高波动率估计的准确性与稳健性。

**定义 9.15 (多尺度已实现波动率 (MSRV) ):** MSRV 为

$$\hat{\theta}_{\text{MSRV}} = \sum_{\ell=1}^L \alpha_\ell \frac{1}{K_\ell} \sum_{j=1}^{K_\ell} RV_{sub,j}^{K_\ell},$$

其中  $\alpha_1, \dots, \alpha_L$  需满足若干约束 (包括  $\sum_{\ell=1}^L \alpha_\ell \simeq 1$ ), 并对  $L$  与  $K$  的增长给出相应条件, 详见 Zhang (2006)。

Zhang (2006) 证明: 在常数波动率这一特殊情形下, 其收敛速度达到最优, 与高斯极大似然估计量一致。Aït-Sahalia et al. (2011) 进一步对 TSRV 与 MSRV 作了改进, 使其在存在序列相关的微观结构噪声时仍保持一致性, 显著拓展了两类估计量的适用范围。

此外, 还有预平均 (pre-averaging) 和实现核 (realized kernels, 简称 RK) 等方法。RK 方法的定义如下:

**定义 9.16 (实现核 (RK) 估计量):** 实现核 (RK) 估计量定义为

$$RK_H = \sum_{|h|<n} k\left(\frac{h}{H+1}\right) \gamma_h(Y), \quad \gamma_h(Y) := \sum_{j=h+1}^n Y_{t_j} Y_{t_{j-h}}, \quad h = 0, \pm 1, \dots$$

其中核函数  $k$  满足  $k(0) = 1$ , 且当  $|s| \rightarrow \infty$  时  $k(s) \rightarrow 0$ ; 带宽参数  $H$  用于控制偏差-方差的权衡。

预平均方法见第 9.10.3 节的预平均方差-协方差估计量。

在高频数据研究领域, Zhou (1996) 率先探讨了微观结构噪声的利用问题, 聚焦于噪声相互独立的情形, 并设定  $H = 1$ 。随后, Hansen & Lunde (2006) 对 Zhou (1996) 的估计量性质进行研究: 在独立同分布的微观结构噪声下该估计量无偏, 但不一致; 同时指出, 即便所用核方法不具一致性, 仍能揭示噪声的部分特性。Barndorff-Nielsen et al. (2008) 进一步为该方法提供了相应的理论框架。Jacod et al. (2009) 提出了预平均方法: 先在固定窗口内对若干时点的观测价格取平均以降低测量误差, 再基于预平均后的数据计算已实现波动率 (RV)。此外, Barndorff-Nielsen et al. (2008) 对连续时间框架下的波动率估计进行了系统综述。另见 Mykland & Zhang (2012), 该研究系统阐述了高频数据的概率模型基础与估计方法, 重点讨论波动率估计, 并介绍了用于杠杆效应估计、实现回归、半方差与方差分析、跳跃检测, 以及利用微观结构噪声衡量流动性等多种技术。

### 9.9.1 基于 Yahoo Finance 目频数据的已实现波动率估计

本节演示如何直接从 Yahoo Finance 获取实际标的资产数据 (默认为 SPY 的复权收盘价), 构造等间隔的对数价格序列, 并以最近一年约  $n = 252$  个交易日为离散网格, 计算已实现波动率 ( $RV = \sum_{i=1}^n (\Delta X_i)^2$ ) 及积分四次方估计量  $\widehat{IQ} = (n/3) \sum_{i=1}^n (\Delta X_i)^4$ 。在  $\sqrt{n}(RV - QV)/\sqrt{2\widehat{IQ}} \Rightarrow N(0, 1)$  的结论下, 进一步给出二次变差  $QV$  的渐近置信区间, 并提供滚动一年窗口的  $RV$  序列以刻画波动率的时变性。读者可按需更换股票代码与时间窗, 并将  $n$  与取样步长 (如  $\Delta = 1/252$ ) 调整为与全文一致的口径; 若采用分钟级及更高频数据, 应使用可获取高频行情的接口, 并注意微观结构噪声对  $RV$  的上行偏差及其稳健修正。

```

1 # =====
2 # 高频波动率估计 (RV, IQ) 与 CLT (Barndorff-Nielsen & Shephard)
3 # - 等间隔观测: X_0, X_{1/n}, ..., X_{1}
4 # - RV_n = sum (ΔX_i)^2
5 # - IQ_hat = (n/3) * sum (ΔX_i)^4
6 # - CLT: sqrt(n) (RV_n - QV) / sqrt(2 * IQ) -> N(0,1)
7 # - 置信区间 (以 RV 代 QV, IQ_hat 代 IQ):
8 # QV RV ± z_{/2} * sqrt(2 * IQ_hat / n)
9 # =====
10
11 set.seed(123)
12
13 # -----
14 # 1) 基础函数: 给定等间隔的对数价格 X_t, 计算 RV, IQ 与 CI
15 # -----
16 rv_iq_ci <- function(x, level = 0.95) {
17 # x: 等间隔对数价格向量 (长度 n+1)
18 n <- length(x) - 1
19 dX <- diff(x) # ΔX_i
20 RV <- sum(dX^2)
21 IQ_hat <- (n/3) * sum(dX^4)
22 # 基于 CLT 的标准误 (以 RV 估 QV, IQ_hat 估 IQ)
23 se <- sqrt(2 * IQ_hat / n)
24 z <- qnorm((1 + level) / 2)
25 ci <- c(RV - z * se, RV + z * se)
26 list(RV = RV, IQ_hat = IQ_hat, SE = se, CI = ci)

```

```

27 }
28
29 # -----
30 # 2) 模拟一个“无杠杆”的布朗半鞅:
31 # X_t = _0^t _s ds + _0^t _s dB_s
32 # _s 与 B 独立(无杠杆), _s 连续右极限(cadlag)
33 # 这里取: _s = 0; _s^2 = 0.04 + 0.02 * sin(2 * s)
34 #
35 simulate_semiMartingale <- function(n) {
36 # 等间隔网格: t_i = i/n, i=0,...,n
37 t <- seq(0, 1, length.out = n + 1)
38 dt <- 1 / n
39 # 漂移设为 0(也可替换为缓变函数)
40 mu <- rep(0, n)
41 # 随机波动(与 B 独立, 这里直接给定确定性函数; 满足“无杠杆”假设)
42 sigma2 <- 0.04 + 0.02 * sin(2 * pi * t[-1]) # 使用右端点 {t_i}
43 sigma <- sqrt(pmax(sigma2, 1e-10))
44 # 生成标准正态增量 _i
45 eps <- rnorm(n)
46 # 构造 ΔX_i = _i * dt + _i * sqrt(dt) * _i
47 dX <- mu * dt + sigma * sqrt(dt) * eps
48 # 对数价格路径(以 X_0 = 0 为例, 不影响 RV 与 QV)
49 X <- c(0, cumsum(dX))
50 # 真值 QV = _0^1 _s^2 ds (在等间隔网格上用黎曼和逼近)
51 QV_true <- sum(sigma^2) * dt
52 list(path = X, QV_true = QV_true, sigma2 = sigma2, t = t)
53 }
54
55 # -----
56 # 3) 单次模拟: 计算 RV、IQ、标准化统计量与置信区间
57 #
58 single_run <- function(n = 23*60) {
59 # n: 把 [0,1] 划分为 n 段, 可理解为 n 次等间隔高频观测
60 sim <- simulate_semiMartingale(n)
61 out <- rv_iq_ci(sim$path, level = 0.95)
62 # 基于真值 QV 的标准化统计量(用于验证 CLT)
63 Z_true <- sqrt(n) * (out$RV - sim$QV_true) / sqrt(2 * (sum(diff(sim$path)
64 ^4) * n/3))
65 list(
66 n = n,
67 RV = out$RV, IQ_hat = out$IQ_hat, SE = out$SE, CI = out$CI,
68 QV_true = sim$QV_true, Z_true = Z_true
69)
70 }
71 # 示例: 把 [0,1] 视作“一天”, 设 n = 23*60 每分钟一笔(仅为演示)
72 res1 <- single_run(n = 23 * 60)
73 cat("单次模拟结果: \n")
74 print(res1[c("n", "RV", "QV_true", "SE", "CI", "Z_true")])
75
76 #

```

```

77 # 4) 多次重复：检验 CLT 与置信区间覆盖率
78 #
79 mc_check <- function(n = 23*60, R = 500, level = 0.95) {
80 z <- qnorm((1 + level) / 2)
81 cover <- numeric(R)
82 Z <- numeric(R)
83 for (r in 1:R) {
84 sim <- simulate_semiMartingale(n)
85 x <- sim$path
86 out <- rv_iq_ci(x, level = level)
87 # 检验 CI 是否覆盖 QV 真值
88 cover[r] <- (sim$QV_true >= out$CI[1] && sim$QV_true <= out$CI[2])
89 # 基于真值 QV 的标准化统计量 (用于查看是否接近 N(0,1))
90 Z[r] <- sqrt(n) * (out$RV - sim$QV_true) / sqrt(2 * out$IQ_hat)
91 }
92 list(
93 n = n, R = R, level = level,
94 coverage = mean(cover),
95 Z_mean = mean(Z), Z_sd = sd(Z)
96)
97 }
98
99 res_mc <- mc_check(n = 23 * 60, R = 200, level = 0.95)
100 cat("\n蒙特卡洛检验:\n")
101 print(res_mc)

```

## 9.10 高频协方差矩阵估计方法

第 9.8 节集中讨论一维情形，然而高频数据同样可用于估计方差与协方差矩阵。本节从应用角度简要介绍若干方差-协方差矩阵的估计方法。

高频数据的方差协方差估计受多种因素影响，以下是影响较为显著的几个因素：

- 做市商的双边报价**：在同一时刻，同一资产的买入价与卖出价存在差异，这种差异可能导致资产价格偏离布朗运动的理论轨迹。尤其在有效市场假设下，价格的这种偏移会对波动率的估计产生影响。
- 离散价格**：实际资产价格的变化是离散的，在价格剧烈波动时，尤其可能出现价格跳跃现象。
- 异步交易 (non-synchronous trading)**：市场上不同资产的交易时间点通常并非完全一致，在高频交易场景下，多个资产的价格难以在同一时刻获取。这种非同步现象成因多样，包括交易量不均衡、不同交易时段重叠等。此外，股票价格以十进制形式报价与变动，这种离散化的价格表示方式也会对相关系数的计算造成影响。上述问题会引发 Epps 效应，使得协方差矩阵的非对角元素趋近于零。而在金融分析中，协方差矩阵用于衡量多个资产之间的相互关系，非对角元素趋近于零意味着资产之间的相关性被低估，这可能误导投资者对资产关系的判断，进而影响投资组合的构建、风险评估以及各类基于资产相关性的金融模型的准确性与有效性。

相较于一维已实现波动率的估计量，多维已实现协方差估计量及其修正方法需要解决时间同步问题。在高频数据中，由于数据采集间隔不固定，特别是秒级的高频交易数据，很

难确保多个资产的价格数据在同一时间点均有观测值，从而导致数据不同步。除数据采集间隔外，诸如交易频率差异等其他因素也会引发异步交易现象，进一步加剧该问题。因此，异步交易对多维协方差建模会产生影响，致使协方差的非对角元素趋近于零。

为解决这一问题，学者们提出了多种时间同步方法。针对高频数据不对齐问题，常见的方法有插值法和删除法。插值法通过填补缺失的观测值来对齐数据，例如用前一个样本值替代缺失值，或者借助自回归模型预测缺失数据。然而，插值法在实证研究中的表现欠佳，因此目前主流方法是删除法，即剔除那些无法同步的数据，仅保留时间上对齐的数据。

其中，最常用的时间同步方法是刷新时间同步法 (Harris et al. 1995)。在高维数据中，很难保证每个资产在同一时刻都有交易记录，直观的做法是舍弃不同步的时间点。具体而言，设首次所有资产都进行交易的时间点为  $t_1$ ，第二次所有资产都交易的时间点为  $t_2$ ，依此类推，直至最后一个时间点  $t_n$ ，那么经过刷新时间同步法得到的采样时间序列为  $T = (t_1, t_2, \dots, t_n)$ 。但这种方法会造成数据损失，在高维资产的情况下，数据损失尤为显著。详见第9.10.2节的介绍。

此外，由于噪声的存在，高频方差-协方差估计还需考虑降噪。本文介绍三类主流的稳健已实现协方差修正方法：双频已实现波动率协方差估计量 (Two-Scale Realized Variance-Covariance Estimator, 简称 TSRV)，多元已实现核协方差估计量 (Realized Kernel Variance-Covariance Estimator, 简称 RK)，预平均协方差估计量 (Pre-averaging Variance-Covariance Estimator, 简称 PAV)。鉴于篇幅限制，以下主要侧重这些方法的表达形式及其 R 语言实现；相关理论可参见本章参考文献。

### 9.10.1 双频已实现协方差估计量 (TSRV)

双频已实现协方差估计量 (TSRV) 由 Zhang (2011) 提出。首先，TSRV 的对角元为方差项。针对时间点  $\tau_1, \tau_2, \dots, \tau_n$  对应的  $n$  个价格观测值构成的对数价格序列  $X$ ，其方差可按如下方式计算：

$$\left(1 - \frac{\bar{n}_K}{\bar{n}_J}\right)^{-1} \left( [X, X]_T^{(K)} - \frac{\bar{n}_K}{\bar{n}_J} [X, X]_T^{(J)} \right),$$

其中  $\bar{n}_K = \frac{n-K+1}{K}$ 、 $\bar{n}_J = \frac{n-J+1}{J}$ ，且

$$[X, X]_T^{(K)} = \frac{1}{K} \sum_{i=1}^{n-K+1} (X_{t_{i+K}} - X_{t_i})^2.$$

其次，TSRV 的非对角元为协方差矩阵元素，其时间同步与 Harris et al. (1995) 提出的刷新时间 (refresh time) 方法一致：在每个刷新时间点，从上一个刷新时间点起，所有资产至少完成一次交易。例如，第一个刷新时间对应所有股票首次均完成交易的时刻；后续刷新时间依次定义为所有股票再次均完成交易的第一个时刻。该过程重复直至时间序列终点。该方法可通过 R 语言软件包 `highfrequency` 中的 `refreshTime` 函数实现。第 9.10.2 节亦阐述刷新时间的确定方法及数据同步操作。

假设存在两个对数价格序列： $X$  和  $Y$ 。设  $\Gamma = \{\tau_1, \tau_2, \dots, \tau_{N_T}^X\}$  与  $\Theta = \{\theta_1, \theta_2, \dots, \theta_{N_T}^Y\}$  分别为两资产的交易时间集合。第一个刷新时间为  $\phi_1 = \max(\tau_1, \theta_1)$ 。后续刷新时间定义为  $\phi_{j+1} = \max(\tau_{N_{\phi_j}^X+1}, \theta_{N_{\phi_j}^Y+1})$ 。完整刷新时间样本网格  $\Phi = \{\phi_1, \phi_2, \dots, \phi_{M_{N+1}}\}$ ，其中  $M_N$  为配对收益率总数。资产  $X$ 、 $Y$  的采样点分别为  $t_i = \max\{\tau \in \Gamma : \tau \leq \phi_i\}$  与  $s_i = \max\{\theta \in \Theta : \theta \leq \phi_i\}$ 。

根据上述刷新时间及对应收益率，协方差计算公式为：

$$c_N \left( [X, Y]_T^{(K)} - \frac{\bar{n}_K}{\bar{n}_J} [X, Y]_T^{(J)} \right),$$

其中

$$[X, Y]_T^{(K)} = \frac{1}{K} \sum_{i=1}^{M_N-K+1} (X_{t_i+K} - X_{t_i}) (Y_{s_i+K} - Y_{s_i}).$$

需要注意，TSRV 估计量并不总是半正定的。若使用 R 语言 `highfrequency` 包的 `rTSCov` 函数计算 TSRV，需将参数 `makePsd` 设为 `TRUE`，函数将把负特征值置零，从而将非正定矩阵转换为半正定矩阵。

下面给出用于计算 TSRV 估计量的 R 代码。

```

1 # TSRV
2 TSRV <- function (priceX, slows, fasts) {
3 # 计算价格的对数
4 lprice <- log (priceX)
5
6 # 确定每个子网格的大小
7 bin11 <- floor (nrow (lprice) / slows)
8 bin22 <- floor (nrow (lprice) / fasts)
9
10 # 计算慢频率的已实现波动
11 rv5m1 <- matrix (0, ncol = ncol (lprice) , nrow = ncol (lprice))
12 for (j in 1:slows) {
13 # 切分数据
14 cut <- seq (j + nrow (lprice) %% slows, by = slows, length.out = bin11)
15 cut1 <- lprice[cut,]
16 diff_cut1 <- diff (cut1)
17 rv5m1 <- rv5m1 + t (diff_cut1) %*% diff_cut1
18 }
19 rv5m <- rv5m1 / slows
20
21 # 计算快频率的已实现波动
22 rv30s1 <- matrix (0, ncol = ncol (lprice) , nrow = ncol (lprice))
23 for (j in 1:fasts) {
24 # 切分数据
25 cut <- seq (j + nrow (lprice) %% fasts, by = fasts, length.out = bin22)
26 cut1 <- lprice[cut,]
27 diff_cut1 <- diff (cut1)
28 rv30s1 <- rv30s1 + t (diff_cut1) %*% diff_cut1
29 }
30 rv30s <- rv30s1 / fasts
31
32 # 计算TSRV估计值
33 TSRV <- rv5m - (fasts / slows) * rv30s
34
35 # 计算年化估计值
36 return (TSRV * 252)
37}
38
```

### 9.10.2 多元已实现核协方差估计量 (RK)

已实现核方法最初用于单变量波动率的稳健估计 (Barndorff-Nielsen et al. (2008)), 随后推广至多元协方差矩阵估计 (Barndorff-Nielsen et al. (2011))。本文在该框架下讨论多元已实现核 (Realized Kernel, RK) 协方差估计量。为便于应用, 先给出刷新时间的定义。

**定义 9.17:** [刷新时间] 对于  $t \in [0, 1]$ , 首次刷新时间为  $\tau_1 = \max(t_1^{(1)}, \dots, t_1^{(d)})$ ;

随后

$$\tau_{j+1} = \max\left(t_{N_{\tau_j}^{(1)}+1}^{(1)}, \dots, t_{N_{\tau_j}^{(d)}+1}^{(d)}\right), \quad (9.16)$$

由此得到的刷新时间样本数量记为  $N$ , 并记  $n^{(i)} = N^{(i)}(1)$ 。

由上可知,  $\tau_1 = \max(t_1^{(1)}, \dots, t_1^{(d)})$  表示所有  $d$  个资产首次均完成交易的最晚时刻; 仅当全部资产都发生首次成交时, 才达到第一次刷新时间。式 (9.16) 给出了递推规则: 在第  $j$  次刷新时刻  $\tau_j$  之后, 各资产出现新的成交时刻,  $\tau_{j+1}$  取这些时刻中的最大值, 即“再次全部完成一次成交”的最晚时刻。按此规则即可从各资产的成交时间序列构造刷新时间序列  $\{\tau_j\}$ , 作为多元已实现核等方法的统一时间标尺。刷新时间的样本量  $N$  由不同步程度以及  $n^{(1)}, n^{(2)}, \dots, n^{(d)}$  决定。保留数据比例用“已保留规模与原始规模之比”衡量; 对刷新时间, 该比值为  $p = dN / \sum_{i=1}^d n^{(i)}$ 。Barndorff-Nielsen et al. (2011) 的图 1 展示了  $d = 3$  时的构造过程。

已实现核以刷新后  $N$  个同步向量价格生成的  $n$  个高频收益率为基础。渐近理论要求在日内开、收盘处对  $m$  个价格做平均以定义端点收益率。设  $n, m \in \mathbb{N}$  且  $n - 1 + 2m = N$ , 令向量观测  $X_0, X_1, \dots, X_n$  满足  $X_j = X(\tau_{j+m})$  ( $j = 1, 2, \dots, n - 1$ ), 并且

$$X_0 = \frac{1}{m} \sum_{j=1}^m X(\tau_j), \quad X_n = \frac{1}{m} \sum_{j=1}^m X(\tau_{N-m+j}).$$

因此,  $X_0$  与  $X_n$  通过对端点做“抖动”得到。通常取  $m$  适中但相对  $n$  很小, 使端点误差被平均从而更接近有效价格。Barndorff-Nielsen et al. (2011) 建议  $m$  取 2 左右。

据此定义高频向量收益率  $r_j = X_j - X_{j-1}$  ( $j = 1, 2, \dots, n$ )。

**定义 9.18:** [多元已实现核协方差估计量 (RK)] 对同步后的高频收益率序列  $\{r_j\}$ , 定义

$$K(X) = \sum_{h=-n}^n k\left(\frac{h}{H+1}\right) \Gamma_h,$$

其中  $k(\cdot)$  为非随机核函数。第  $h$  阶已实现自协方差

$$\Gamma_h = \begin{cases} \sum_{j=|h|+1}^n r_j r'_{j-h}, & h \geq 0, \\ \sum_{j=|h|+1}^n r_{j-h} r'_j, & h < 0. \end{cases}$$

核函数  $k$  应具有以下特征:

1.  $k(0) = 1, k'(0) = 0$ ;
2.  $k$  二阶可导且导数连续, 以保证函数的平滑性, 避免出现不规则或跳跃行为;

3.  $\int_0^1 k(x)^2 dx, \int_0^1 k'(x)^2 dx, \int_0^1 k''(x)^2 dx < \infty$ <sup>3</sup>;
4. 对任意  $\lambda \in \mathbb{R}$ ,  $\int_{-\infty}^{\infty} k(x) \exp(ix\lambda) dx \geq 0$ 。该条件确保由  $k(x)$  生成的相关矩阵为半正定。

可直接使用 R 语言 `highfrequency` 包中的 `rKernelCov` 计算 RK。与 TKX 及随后介绍的 PAV 相比, RK 的显著优势在于能保证估计矩阵半正定。

下面给出基于 Parzen 核的示例代码。<sup>4</sup>

```

1 # RK
2 RK <- function (priceX, slows, fasts) {
3 # 计算价格的对数
4 lprice <- log (priceX)
5
6 # 确定每个子网格的大小
7 bin11 <- floor (nrow (lprice) / slows)
8 bin22 <- floor (nrow (lprice) / fasts)
9
10 # 计算慢频率的已实现波动率
11 rv10m1 <- matrix (0, nrow = slows, ncol = ncol (lprice))
12 for (j in 1:slows) {
13 # 切分数据
14 cut <- seq (j + nrow (lprice) %% slows, length.out = bin11, by = slows)
15 cut1 <- lprice[cut, , drop = FALSE]
16 rv10m1[j,] <- colSums (diff (cut1) ^2)
17 }
18 rv10m <- colMeans (rv10m1)
19
20 # 计算快频率的已实现波动率
21 rv30s1 <- matrix (0, nrow = fasts, ncol = ncol (lprice))
22 for (j in 1:fasts) {
23 # 切分数据
24 cut <- seq (j + nrow (lprice) %% fasts, length.out = bin22, by = fasts)
25 cut1 <- lprice[cut, , drop = FALSE]
26 rv30s1[j,] <- colSums (diff (cut1) ^2)
27 }
28 rv30s <- colMeans (rv30s1)
29
30 # 计算噪声的方差
31 noisev <- rv30s / ((bin22 - 1) * 2)
32
33 # 计算平滑窗口的大小
34 h <- round (mean (3.51 * (noisev / rv10m) ^0.4 * (nrow (lprice) - 1) ^0.6)
35)
36
37 # 计算价格变动
38 r <- diff (lprice)
39
计算已实现协方差矩阵

```

<sup>3</sup> “矩有界”可解读为:  $\int_0^1 k(x)^2 dx < \infty$  表示核函数整体不致过大;  $\int_0^1 k'(x)^2 dx < \infty$  进一步保证了平滑性与可导性;  $\int_0^1 k''(x)^2 dx < \infty$  确保函数变化不过于剧烈。

<sup>4</sup>Parzen 核: 当  $x \in [0, 0.5]$ ,  $k(x) = 1 - 6x^2 + 6x^3$ ; 当  $x \in [0.5, 1]$ ,  $k(x) = 2(1 - x)^3$ 。

```

40 RK <- t (r) %*% r
41 for (hh in 1:h) {
42 x <- (hh - 1) / h
43 if (x <= 0.5) {
44 k <- 1 - 6 * x^2 + 6 * x^3
45 } else {
46 k <- 2 * (1 - x)^3
47 }
48 RK <- RK + k * (t (r[(hh + 1) : nrow (r), , drop = FALSE]) %*% r[1: (nrow
49 (r) - hh), , drop = FALSE] +
50 t (r[1: (nrow (r) - hh), , drop = FALSE]) %*% r[(hh + 1) : nrow (r), ,
51 drop = FALSE])
52 # 计算年化的估计值
53 return (RK * 252)
54 }
55

```

### 9.10.3 预平均协方差估计量 (PAV)

预平均协方差估计方法 (PAV) 由 Christensen et al. (2010) 与 Hautsch & Podolskij (2013) 在 Jacod et al. (2009) 的基础上发展而来。该方法吸收了高频数据降噪的既有经验，在思路上兼具双频 (TSRV) 与已实现核 (RK) 的优点：先对原始对数价格（收益率）序列进行平滑，再借鉴更高频协方差作为噪声估计的作差思路以实现降噪，最终得到预平均协方差估计量。（术语上常用“PAV”称呼；为与下式记号一致，公式中保留 PAV 的符号。）

考虑与公式 (9.15) 相同结构的多变量对数价格模型：

$$Y_\tau = X_\tau + \epsilon_\tau,$$

其中观测到的  $d$  维对数价格由潜在布朗半鞅  $X$  与噪声项  $\epsilon_\tau$  相加而成； $\epsilon_\tau$  与  $X$  相互独立，且构成 i.i.d. 随机变量序列。

直观地说，在零均值的 i.i.d. 微观结构噪声下，对观测对数价格做局部加权平均能够削弱噪声的影响。这正是预平均方法的基本思想：在给定邻域内对  $Y$  的观测求加权平均，以近似其连续部分，从而“平均掉”噪声。

设某一刷新时段  $\tau$ （数据对齐后）包含  $N$  个等间隔的收益率观测。记  $r_{\tau_i}$  为该时段某资产的第  $i$  个收益率 ( $i = 1, \dots, N$ )，共有  $d$  个资产。

为给出单变量的预平均估计量，先定义预平均收益率：

$$\bar{r}_{\tau_j}^{(k)} = \sum_{h=1}^{k_N-1} g\left(\frac{h}{k_N}\right) r_{\tau_j+h}^{(k)}, \quad (9.17)$$

其中  $g : [0, 1] \rightarrow \mathbb{R}$  为非零实值函数，取  $g(x) = \min(x, 1-x)$ 。量  $k_N$  随样本量  $N$  变化，令  $k_N = \lfloor \theta N^{1/2} \rfloor$ 。

Hautsch & Podolskij (2013) 建议  $\theta = 0.8$ （见其第 2.4 节）。预平均收益率可理解为局部窗口内收益率的加权平均，能够减弱噪声影响；窗口阶  $k_N$  的设定旨在获得良好的收敛性质。对应的预平均方差思想与已实现方差 (RV) 一致，但基于预平均收益率并加入偏差修

正项:

$$\hat{C} = \frac{N^{-1/2}}{\theta\psi_2} \sum_{i=0}^{N-k_N+1} (\bar{r}_{\tau_i})^2 - \frac{\psi_1^{k_N} N^{-1}}{2\theta^2\psi_2^{k_N}} \sum_{i=0}^N r_{\tau_i}^2,$$

其中

$$\psi_1^{k_N} = k_N \sum_{j=1}^{k_N} \left( g\left(\frac{j+1}{k_N}\right) - g\left(\frac{j}{k_N}\right) \right)^2, \quad \psi_2^{k_N} = \frac{1}{k_N} \sum_{j=1}^{k_N-1} g^2\left(\frac{j}{k_N}\right), \quad \psi_2 = \frac{1}{12}.$$

多变量情形与之相似, 其定义为:

$$\text{PAV} = \frac{N}{N - k_N + 2} \frac{1}{\psi_2 k_N} \sum_{i=0}^{N-k_N+1} \bar{r}_{\tau_i} \cdot \bar{r}'_{\tau_i} - \frac{\psi_1^{k_N}}{\theta^2 \psi_2^{k_N}} \hat{\Psi}_N.$$

其中  $\bar{r}_{\tau_i}$  为多维预平均收益率 (即 (9.17) 的扩展),  $\hat{\Psi}_N = \frac{1}{2N} \sum_{i=1}^N \bar{r}_{\tau_i} (\bar{r}_{\tau_i})'$ 。该项用于偏差校正以确保一致性 (见 Christensen et al. (2010) 第 3.1 节), 但可能导致估计矩阵并非半正定。一个实用做法是适度增大带宽, 令  $k_N = \lfloor \theta N^{1/2+\delta} \rfloor$ 。当  $\delta = 0.1$  时, 可在无需偏差校正的情况下得到一致且半正定的估计, 即

$$\text{PAV}^\delta = \frac{N}{N - k_N + 2} \frac{1}{\psi_2 k_N} \sum_{i=0}^{N-k_N+1} \bar{r}_i \cdot \bar{r}'_i.$$

在 R 语言中, 可使用 `highfrequency` 包的 `rMRCov` 函数计算 (本书记作) PAV (公式中记号为 PAV)。由于存在偏差修正, 所得矩阵不一定半正定; 此时可将参数设为 `makePsd = TRUE` 以进行半正定化处理。

以下为 PAV 估计量的 R 语言实现示例。

```

1 # PAV
2 PAV <- function (priceX, theta) {
3 # 计算价格的对数
4 lprice <- log (priceX)
5
6 # 设置计算的参数
7 delta_n <- 1 / nrow (lprice)
8 K_n <- round (theta / sqrt (delta_n) / 2) * 2
9
10 # 常数值, 根据Jacod等人的研究
11 psi1 <- 1
12 psi2 <- 1 / 12
13 psi_K <- (1 + 2 / K_n^2) / 12
14
15 # 初始化协方差矩阵
16 Z <- matrix (0, ncol = ncol (lprice) , nrow = ncol (lprice))
17
18 # 计算Z矩阵
19 for (j in 1: (nrow (lprice) - K_n + 1)) {
20 r <- (1 / K_n) * colSums (lprice[(j + K_n/2) : (j + K_n - 1) ,] -
21 lprice[j: (j + K_n/2 - 1) ,])
22 Z <- Z + r %*% t (r)
23 }

```

```

24 # 计算估计值
25 CX <- sqrt (delta_n) / (theta * psi2) * Z - psi1 * delta_n / (2 * theta^2
26 * psi2) *
27 t (diff (lprice)) %*% diff (lprice)
28
29 # 计算年化估计值
30 return (CX * 252)
31 }

```

综上所述，TSRV 方法通过在不同时间尺度上构造方差-协方差估计量并作差以降噪，能够同时减弱（一次）Epps 效应与微观结构噪声的影响，但其不足在于难以保证估计矩阵的正定性。RK 方法引入核估计思路，对数据条件要求更低：与 TSRV、PAV 需假设对数价格中的噪声独立同分布不同，RK 允许噪声具有内生性与序列相关性。PAV 方法在一定程度上融合了 RK 与 TSRV 的思路——先对原始价格序列做平滑以削弱噪声，再据此进行协方差估计；其收敛速度在上述三类方法中通常较快。

得到方差-协方差矩阵后，可进一步求解全局最小方差投资组合（global minimum variance portfolio，简称 GMVP）的权重，以最小化组合方差：

$$\mathbf{w} = \frac{\Sigma^{-1}\mathbf{1}}{\mathbf{1}'\Sigma^{-1}\mathbf{1}},$$

其中  $\mathbf{1}$  为维度等于资产数量的全 1 列向量。

```

1 compute_mvp_weights_simple<- function (cov_matrix) {
2 n <- nrow (cov_matrix) # 资产数
3 ones <- rep (1, n)
4 inv_cov_matrix <- solve (cov_matrix) # 协方差矩阵的逆
5 # 计算最小方差组合的权重
6 num <- inv_cov_matrix %*% ones
7 den <- t (ones) %*% inv_cov_matrix %*% ones
8 weights <- num / as.numeric (den)
9 return (weights)
10 }
11

```

若 GMVP 权重出现较多负值且市场不允许卖空，可采用受约束优化。其目标函数为

$$\min_{\mathbf{w}} \mathbf{w}'\Sigma\mathbf{w},$$

对应约束条件为

$$\sum_{i=1}^n w_i = 1, \quad w_i \geq 0 \quad \forall i.$$

其对应的 R 代码示例如下：

```

1 compute_mvp_weights <- function(cov_matrix) {
2 n <- nrow(cov_matrix) # 资产数
3 Dmat <- 2 * cov_matrix # 乘以2，因为quadprog最小化的是 0.5*t(x)%*%Dmat%*
4 %x + t(dvec)%*%x
5 dvec <- rep(0, n) # 因为函数中没有线性项
6 Amat <- cbind(rep(1, n), diag(n)) # Amat矩阵描述不等式和等式约束（权重之
7 和为1）

```

```

6 bvec <- c(1, rep(0, n) # bvec的第一个元素是1 (等式约束), 其余元素是0 (不
 等式约束)
7 meq <- 1 # meq是等式约束的数量 (权重之和 = 1)
8 # 求解二次规划问题
9 solution <- solve.QP(Dmat, dvec, Amat, bvec, meq)
10 return(solution$solution)
11 }

```

## 9.11 章节总结

本章围绕连续时间序列模型，搭建了由“基础—工具—模型—估计—应用”构成的完整链条。首先，以布朗运动为起点，系统阐述其独立增量、正态增量与样本路径处处不可导等核心性质，并据此引入局部 Hölder 连续性与穿越时间等结果，为后续随机分析奠定概率与路径两方面的直觉基础。随后，结合随机游走极限定理与等距性质，给出随机积分的严格定义，并借助伊藤引理将“随机版链式法则”形式化，使由 SDE 所定义、含漂移与扩散两部分的过程能够实施函数变换与微分运算。

在模型层面，我们以广义布朗运动为核心，将扩散过程表示为随机微分方程

$$dX_t = \mu(X_t, t) dt + \sigma(X_t, t) dB_t,$$

并给出解的存在唯一的充分条件。估计方面，先在可获得闭式转移密度的情形下，给出布莱克-舒尔斯 (Black-Scholes) / 几何布朗运动模型的极大似然估计；在转移密度不可得时，引入 Aït-Sahalia 的封闭形式似然展开，通过  $X \rightarrow Y \rightarrow Z$  的正态化变换与 Hermite 展开构造近似似然，并论证其与“真实”MLE 的一致性与渐近等价。与此同时，我们回顾了连续时间模型的设定检验：或基于边际/转移密度一致性的检验，或以非参数转移密度对照假设模型，从而在估计之外提供模型诊断与稳健性保障。

在高频应用层面，本章区分一维与多维两条主线。一维方面，利用已实现波动率 (RV) 一致逼近二次变差，并配合  $\widehat{IQ}$  构造中心极限定理下的标准化与置信区间；同时强调微观结构噪声会导致高频 RV 的上行偏差，需通过降频或两尺度/多尺度修正 (TSRV、MSRV) 加以校正。多维方面，重点讨论异步交易引发的 Epps 效应与时间同步问题，并介绍三类常用协方差估计量及其 R 实现：双频 TSRV (作差降噪但可能非半正定)、实现核 RK (核平滑、可容许内生性与序列相关噪声、半正定)、以及预平均 PAV (窗口平滑加偏差修正、收敛速度较快)。实际操作中，需明确取样步长与年化口径，合理设置同步规则 (如刷新时间或公共网格对齐)，依据样本量选择 TSRV 的  $K/J$  参数，并在需要时对估计矩阵做半正定化处理，以确保后续组合优化 (如 GMVP) 可解且稳定。

总体而言，本章在“从随机基础到计量方法、从极大似然到高频协方差、从理论到实现”的主线之下，提供了统一的分析框架。读者在掌握布朗运动、伊藤引理与随机积分等工具后，能够对扩散模型进行参数估计与设定检验；面对高频数据时，能识别并修正微观结构噪声与异步交易造成的偏差，择优使用波动率与协方差估计方法，并将其服务于风险度量与资产配置等实务问题。后续章节可在此基础上延展至跳跃—扩散、随机波动率与仿射类模型，以及以期权价格或收益率曲线为目标的结构化估计与定价应用。

## 9.12 习题

1. 假设你有一条标准布朗运动样本路径  $B_t$ , 已知其满足局部 Hölder 连续性条件:

$$|B_t - B_s| \leq C|t - s|^\gamma, \quad \text{其中 } \gamma < \frac{1}{2}.$$

- (a) 证明: 对任意  $t, s$ , 当  $|t - s|$  充分小时, 上述不等式成立。
- (b) 若  $\gamma > \frac{1}{2}$ , 讨论样本路径的 Hölder 连续性将如何表现。

2. 假设  $X_t$  是一个伊藤过程, 满足

$$dX_t = \mu(t, X_t) dt + \sigma(t, X_t) dB_t,$$

其中  $B_t$  为标准布朗运动,  $f(t, x)$  为关于  $t$  与  $x$  的光滑函数。

- (a) 应用伊藤引理, 推导  $f(t, X_t)$  的伊藤微分。
- (b) 当  $f(t, x) = x^2$  时, 计算其伊藤微分。
- (c) 假设  $X_t = \mu t + \sigma B_t$ , 试用伊藤引理计算  $X_t^2$  的微分。

3. 下载多只股票的高频交易数据, 并完成下列步骤:

- (a) 使用已实现波动率 (RV)、多元已实现协方差矩阵 (RCOV)、核估计方法 (RK) 和预平均协方差估计 (PAV) 来估计方差-协方差矩阵。
  - (b) 依据估计的协方差矩阵设定给定投资组合的权重。
  - (c) 若 GMVP 权重出现较多负值且市场不允许卖空, 应如何处理?
4. 高频数据通常包含噪声, 噪声对协方差估计的影响需要通过降噪方法处理。在含噪模拟数据上, 应用 TSRV 和 RK 方法估计协方差矩阵, 并与 RV 方法对比; 比较不同估计方法 (如 PAV) 对协方差估计结果的改善效果。
- 提示: 通过向模拟数据中添加白噪声来引入噪声:  $Y_t = X_t + \varepsilon_t$ , 其中  $\varepsilon_t$  为白噪声。



# 10 收益率曲线 (Yield Curve)

在本章中，我们讨论利率与货币时间价值 (time value of money) 的度量问题。收益率曲线在诸多应用中至关重要：例如资产配置、预测未来利率（如 Campbell & Shiller 1991）、通胀与国民收入（如 Estrella & Mishkin 1997）、利率衍生品定价以及理性预期的检验。此外，我们还讨论收益率曲线的离散时间建模。在本章中，我们将讨论利率与货币时间价值 (time value of money) 的度量问题。收益率曲线在诸多应用中至关重要：例如资产配置、未来利率预测（如 Campbell & Shiller, 1991）、通胀与国民收入分析（如 Estrella & Mishkin, 1997）、利率衍生品定价以及理性预期检验。此外，我们还将讨论收益率曲线的离散时间建模方法。

**定义 10.1 (收益率曲线 (Yield Curve))**: 收益率曲线描述在特定时点、信用质量相同而期限不同的债券到期收益率随期限变化所形成的曲线。通常呈上倾斜形态（长期利率高于短期利率），也可能平坦或倒挂。若出现倒挂（长期利率低于短期利率），通常被视为经济衰退风险上升的先行信号。

## 10.1 贴现函数、收益率曲线与远期利率

首先，我们考虑零息债券。零息债券（到期一次支付面值  $M$ ，期间无息票）将“时间价值”从其他复杂因素（如票息频率、嵌入式期权等）中剥离出来，使利率度量对应单一现金流的“现值—未来值”关系，是构建整条收益率曲线最为干净的基石。在有息票债券定价中，所有现金流都可拆解为若干面值为 1 的零息债券之和，因此先明确零息债券上的利率度量口径至关重要。

**定义 10.2 (现货利率 (spot rate) 的经济含义与口径)**: 现货利率  $y_n$  指在“今天”买入、于  $n$  期后收回面值  $M$  的零息债券之到期收益率（按约定复利口径计算），满足

$$p = \frac{M}{(1 + y_n)^n}.$$

关于上述现货利率的定义，这里我们补充四点。第一是口径依赖：期数的单位与复利频率（年、半年、季度、月、日或连续复利）需要与  $y_n$  的口径一致；若改用连续复利  $y^{(c)}(\tau)$ ，应采用  $p = M e^{-\tau y^{(c)}(\tau)}$ 。第二是期限特异：每个期限对应一个现货利率  $\{y_1, \dots, y_n\}$ ，这些点共同构成零息收益率曲线，说明现货利率是期限的函数而非单一常数。第三是无套利解释：若两种无风险策略在  $n$  期末产生同额确定现金流，其现值必须一致，由此在给定价格  $p$  时确定  $y_n$ ，从而形成一一对应。第四是与到期收益率的区别：有息票债的到期收益率 (Yield to Maturity, 简称 YTM) 是使整串现金流现值等于价格的内部收益率，本质上将不同期限的贴现“混合”；现货利率  $y_n$  则只针对单一到期点的贴现。

**定义 10.3 (贴现因子 (discount factor) 的作用与性质):** 贴现因子  $d_n$  是  $n$  期后 1 单位确定现金流的标准化价格:

$$d_n = \frac{1}{(1 + y_n)^n}.$$

关于贴现因子的作用与性质, 这里我们补充四点说明。第一是可加性与线性定价: 对任意确定现金流向量  $b = (b_{\tau_1}, \dots, b_{\tau_m})$ , 其价格写作

$$p = \sum_{j=1}^m b_{\tau_j} d_{\tau_j},$$

由此可见  $\{d_{\tau}\}$  构成对确定现金流的完整定价基, 因而成为估计整条曲线的自然对象。第二是单调性: 在无套利且利率非负的情形下,  $d_{\tau}$  随期限  $\tau$  单调递减; 在负利率环境下, 一般仍随期限递减, 但曲线可能更为平缓或出现局部形变, 极短端出现  $d_{\tau} > 1$  也并非不可能。第三是口径统一: 在连续复利记号下, 贴现因子、收益率与远期利率可以写成统一框架  $d(\tau) = \exp\{-\tau y^{(c)}(\tau)\}$ , 便于进行微分、积分以及与  $f(\tau)$  的相互转换。第四是插值与平滑: 实务估计通常先在  $d(\tau)$  上实施平滑或正则化 (例如样条、指数样条、带单调约束的样条), 再推得  $y(\tau)$  与  $f(\tau)$ , 以在插值与外推时保持与线性定价的一致。

**定义 10.4 (远期利率 (forward rate) 的含义、计算与度量口径):** 一期远期利率  $f_n$  是今天锁定、从第  $n$  期到第  $n+1$  期的一期无风险借贷利率, 满足

$$1 + f_n = \frac{d_n}{d_{n+1}}.$$

连续复利下的瞬时远期利率  $f(\tau)$  定义为

$$f(\tau) = -\frac{d}{d\tau} \ln d(\tau), \quad y(\tau) = \frac{1}{\tau} \int_0^\tau f(s) ds.$$

关于远期利率的含义与度量, 这里我们补充四点。第一是离散情形下的无套利推导: 可以比较两种无风险策略——其一是先借  $n$  期、到期后再滚动至  $n+1$  期; 其二是直接借  $n+1$  期。两者在  $n+1$  期末的确定现金流应当一致, 由此得到等式  $1 + f_n = d_n/d_{n+1}$ 。第二是度量含义: 瞬时远期利率  $f(\tau)$  衡量“瞬时的期限增量”的价格; 它与收益率的关系  $y(\tau) = \tau^{-1} \int_0^\tau f(s) ds$  说明, 收益率是区间  $[0, \tau]$  上远期利率的算术平均。第三是预期解释与测度: 在以  $d(\tau)$  为计价基准的远期测度下, 远期利率等于相应未来短端利率的条件期望; 在物理测度或风险中性测度下, 还需要考虑风险溢价项, 二者并不必然相等。第四是数值稳定性: 由贴现函数  $d(\tau)$  做数值微分以获得  $f(\tau)$  往往对噪声较为敏感, 实务中通常先对  $d(\tau)$  进行单调平滑或正则化, 再计算  $f(\tau)$ ; 或者直接对  $f(\tau)$  采取一个结构化的函数族 (如 Nelson–Siegel 型), 以获得更稳健的曲线与导数。

若一债券在未来  $\tau_j$  时刻支付  $b_j$  (息票与本金),  $j = 1, \dots, m$ , 无套利给出线性定价公式:

$$p = \sum_{j=1}^m b_j d_{\tau_j} = \sum_{j=1}^m \frac{b_j}{(1 + y_{\tau_j})^{\tau_j}}.$$

零息收益率曲线即  $\{y_{\tau}\}$  的集合, 也可等价用  $\{d_{\tau}\}$  或  $\{f_{\tau}\}$  表示。

**定义 10.5 (到期收益率 (Yield to Maturity) ):** 面值为 1、到期期限为  $n$  期、每期付息  $c$  的息票债券，其价格  $p$  与到期收益率  $y$  满足

$$p = \frac{c}{(1+y)} + \frac{c}{(1+y)^2} + \cdots + \frac{1}{(1+y)^n}.$$

以面值成交 ( $p = 1$ ) 的息票债的到期收益率随到期变化构成票面收益率曲线 (*par curve*)。

在连续复利口径下更为便捷：

**定义 10.6 (贴现函数与连续复利收益率曲线):** 贴现函数  $d(\tau)$  为未来  $\tau$  时点 1 美元的现值，收益率曲线  $y(\tau)$  与  $d(\tau)$  关系为

$$d(\tau) = \exp(-\tau y(\tau)), \quad y(\tau) = \frac{1}{\tau} \int_0^\tau f(s) ds,$$

其中  $f(s)$  为瞬时远期利率。给定边界条件  $d(0) = 1, d(\infty) = 0$ ,  $d, y, f$  可相互唯一转换，且  $d(\tau)$  单调递减。

以下 R 代码以 FRED 公布的美国国债常数到期收益率 (DGS 系列) 为输入，对同一观测日的各期限收益率进行口径统一与“曲线三件套”计算：先将原始年化的简单/债券口径收益率换算为连续复利  $y(\tau)$ ，据此得到贴现函数  $d(\tau) = \exp\{-\tau y(\tau)\}$ ；随后对  $g(\tau) = \tau y(\tau)$  进行样条平滑并求导，从而稳健地估计瞬时远期利率  $f(\tau) = \frac{d}{d\tau}\{\tau y(\tau)\}$ ，同时给出相邻期限间的一期离散远期利率  $1 + f_{i \rightarrow i+1} = d(\tau_i)/d(\tau_{i+1})$ 。代码最终输出按期限排列的表格（含原始年化、连续复利、贴现、瞬时远期）与简要三联图 ( $y(\tau)$ 、 $d(\tau)$ 、 $f(\tau)$ )，便于教学展示与快速质检。若后续以息票债逐点抽取得到真实零息贴现函数  $\{d(\tau)\}$ ，只需将代码中的  $d(\tau)$  替换为抽取结果，即可在同一框架下即时生成一致口径的  $y(\tau)$ 、 $f(\tau)$  与离散远期利率，用于定价、久期/凸性与风险管理等分析。

```

1 # =====
2 # 收益率曲线三件套: y(), d(), f() — 以 FRED 美债常数到期为例
3 # 依赖: quantmod, xts, stats (base) , utils
4 # 功能:
5 # - 从 FRED 拉取常数到期收益率 (DGSx)
6 # - 统一为“连续复利”口径: y_cc()
7 # - 计算贴现函数 d()=exp{- y_cc()}
8 # - 以样条对 g()= y_cc() 平滑并求导, 得瞬时远期 f()=g'()
9 # - 计算相邻节点离散远期: 1+f_disc = d(_i)/d(_{i+1})
10 # - 结果打印 (仅四舍五入数值列) , 并作简单三联图
11 # =====
12
13 suppressPackageStartupMessages(library(quantmod))
14
15 # 1) 选择期限 (单位: 年) 与 FRED 代码
16 taus <- c(0.25, 0.5, 1, 2, 3, 5, 7, 10, 20, 30)
17 fred <- c("DGS3MO", "DGS6MO", "DGS1", "DGS2", "DGS3", "DGS5", "DGS7", "DGS10", "
18 DGS20", "DGS30")
19
20 # 2) 拉取 FRED 数据并对齐到“最后一个共同日期”
21 getSymbols(fred, src = "FRED", auto.assign = TRUE, warnings = FALSE)
22 Y <- do.call(merge, lapply(fred, get)) # 合并为 xts, 一列一个期限

```

```

22 Yc <- Y[stats::complete.cases(Y)] # 仅保留各列都有值的日期
23 stopifnot(nrow(Yc) > 0)
24 last_row <- tail(Yc, 1) # 取最后一个共同日期
25 asof <- index(last_row)
26 y_annual_pct <- as.numeric(last_row) # 百分比
27 y_annual <- y_annual_pct/100 # 年化小数 (简单/债券口径)
28
29 # 3) 统一口径: 连续复利 y_cc() = ln(1 + y_annual())
30 y_cc <- log(1 + y_annual)
31
32 # 4) 贴现函数 d() = exp{- y_cc()}
33 d_tau <- exp(-taus * y_cc)
34
35 # 5) 瞬时远期: f() = d/d { y_cc() } , 用样条平滑 g()= y_cc() 再求导
36 g_tau <- taus * y_cc
37 fit <- smooth.spline(x = taus, y = g_tau, spar = NULL) # 自动选择平滑
 参数
38 gprime <- predict(fit, x = taus, deriv = 1)$y
39 f_tau <- gprime # 连续复利的瞬时
 远期
40
41 # 6) 相邻节点的一期“离散远期” : 1 + f_disc = d(_i)/d(_{i+1})
42 f_disc <- rep(NA_real_, length(taus)-1)
43 for (i in 1:(length(taus)-1)) {
44 f_disc[i] <- d_tau[i]/d_tau[i+1] - 1
45 }
46
47 # 7) 结果表: 连续复利 y_cc、贴现 d_tau、瞬时远期 f_tau; 并保留原年化简单 y_
 annual 对照
48 res <- data.frame(
49 asof = as.character(asof),
50 tau_year = taus,
51 y_annual = y_annual, # 简单/债券式年化
52 y_cc = y_cc, # 连续复利 (年)
53 d_tau = d_tau, # 贴现因子
54 f_cc = f_tau # 瞬时远期 (连续复利)
55)
56
57 disc_tab <- data.frame(
58 from_tau = taus[-length(taus)],
59 to_tau = taus[-1],
60 f_disc = f_disc # 离散复利
61)
62
63 # 8) 结果打印
64 cat("==== FRED Constant Maturity US Treasury — as of", as.character(asof), "
 ===\n")
65 num_cols_res <- sapply(res, is.numeric)
66 res_print <- res
67 res_print[num_cols_res] <- lapply(res_print[num_cols_res], round, 6)
68 print(res_print, row.names = FALSE)

```

```

69
70 cat("\n--- Discrete one-period forwards between adjacent nodes (annual
 compounding) ---\n")
71 num_cols_disc <- sapply(disc_tab, is.numeric)
72 disc_print <- disc_tab
73 disc_print[num_cols_disc] <- lapply(disc_print[num_cols_disc], round, 6)
74 print(disc_print, row.names = FALSE)
75
76 # 9) 简单三联图: y() (连续) 、 d() 、 f() (连续)
77 op <- par(mfrow = c(1,3), mar = c(4,4,2,1))
78 plot(taus, y_cc*100, type="b", pch=19,
79 xlab="Maturity (years)", ylab="Yield y() [% p.a., cont.]",
80 main="Zero-coupon Yield (cont.)")
81 plot(taus, d_tau, type="b", pch=19,
82 xlab="Maturity (years)", ylab="Discount d()",
83 main="Discount Function")
84 plot(taus, f_tau*100, type="b", pch=19,
85 xlab="Maturity (years)", ylab="Instantaneous f() [% p.a.]",
86 main="Instantaneous Forward (cont.)")
87 par(op)

```

## 10.2 由息票债估计收益率曲线

实际中并无全期限的零息债券可供观察，多为息票债券。记价格为  $p$ 、现金流为  $b$ 、支付时点为  $\tau$ ，目标是用样本  $\{p_i, b_{ij}, \tau_{ij}\}$  提取  $d(\cdot), y(\cdot), f(\cdot)$ 。直接反演线性系统常会出现解不唯一、波动过大，且无法保证  $d$  的单调性或插值合理性，因而需进行正则化与平滑。

**定义 10.7 (统计模型):**

$$p_i = \sum_{j=1}^{m_i} b_{ij} d(\tau_{ij}) + \varepsilon_i, \quad E(\varepsilon_i) = 0, \quad \text{Var}(\varepsilon_i) < \infty.$$

令  $p = Bd + \varepsilon$ ，其中  $B$  为现金流矩阵， $d$  为未知贴现因子在若干节点处的取值向量。  
最小二乘估计

$$\hat{d} = (B'B)^{-1}B'p$$

在  $m$  较大、矩阵病态或需满足单调性/插值条件时并不理想，因而常采用正则化或基函数展开等方法。

上述统计模型可以这样理解。对任意一只息票债券  $i$ ，其价格等于未来各支付时点  $\{\tau_{ij}\}_{j=1}^{m_i}$  上现金流  $\{b_{ij}\}$  的贴现值之和；为容纳市场摩擦（流动性、买卖价差、计息与税务口径差异）、个券特征与数据噪声等因素，我们引入一个均值为零、方差有限的误差项  $\varepsilon_i$ ：

$$p_i = \sum_{j=1}^{m_i} b_{ij} d(\tau_{ij}) + \varepsilon_i.$$

将所有债券在所有支付时点上的贴现因子汇成未知向量  $d$ ，把相应现金流系数汇成现金流矩阵  $B$ ，即可写成矩阵形式  $p = Bd + \varepsilon$ 。形式上，用最小二乘可得  $\hat{d} = (B'B)^{-1}B'p$ 。但在实际估计中，直接逐点求  $d$  往往面临三类问题：第一，结点数量  $m$  通常很大，甚至超过

样本量  $n$ , 从而使  $(B'B)$  病态或近似不可逆, 解对噪声极其敏感; 第二, 逐点估计难以自然满足无套利约束 (例如  $d(\tau)$  应随  $\tau$  递减、且  $d(0) = 1$ ); 第三, 逐点结果只在有限支付时点上有效, 难以提供平滑、可插值与可外推的整条曲线。基于此, 实务与文献更倾向于把  $d(\tau)$  作为一个光滑函数来估计: 要么采用基函数 (级数) 展开, 用少量系数刻画整条曲线并配合单调/正性等约束; 要么在目标函数中加入平滑惩罚 (如对二阶导数的积分惩罚) 进行正则化。通过这种方式, 可以在“拟合误差—曲线平滑—无套利约束”之间取得平衡, 得到一条在全期限上可用的贴现曲线  $d(\tau)$ , 并据此一并推导收益率  $y(\tau)$  与远期利率  $f(\tau)$ 。

### 10.2.1 基函数展开法

这里我们将贴现函数  $d(t)$  视为一条未知但平滑的曲线, 用少量“基函数”的线性组合来逼近:

$$d(t) \approx \sum_{\ell=1}^L \theta_\ell g_\ell(t), \quad L \ll n.$$

思路是将债券价格的线性定价  $p_i = \sum_{j=1}^{m_i} b_{ij} d(\tau_{ij}) + \varepsilon_i$  改写为关于系数  $\theta = (\theta_1, \dots, \theta_L)'$  的线性回归:

$$p_i = \sum_{\ell=1}^L \theta_\ell X_{\ell i} + \varepsilon_i, \quad X_{\ell i} = \sum_{j=1}^{m_i} b_{ij} g_\ell(\tau_{ij}),$$

再用最小二乘 (必要时配合正则化与约束) 来估计  $\theta$ 。估计完成后, 即可在任意期限点上计算  $\hat{d}(t) = \sum_{\ell=1}^L \hat{\theta}_\ell g_\ell(t)$  并据此得到  $\hat{y}(t) = -\ln \hat{d}(t)/t$ ,  $\hat{f}(t) = -\frac{d}{dt} \ln \hat{d}(t)$  等曲线。

为了让这一路径在统计与金融上都“站得住脚”, 通常需要注意以下几个方面。第一是基函数的选择与复杂度控制: 早期文献采用分段多项式 (McCulloch 1971)、Bernstein 多项式 (Schaefer 1981)、指数基/指数样条 (Vasicek & Fong 1982)、平滑样条 (Fisher et al. 1995) 等。经验上, 分段三次样条或指数样条能在“平滑—灵活”之间取得较好平衡。第二是平滑与单调 (无套利) 约束: 直接最小二乘容易在稀疏期限段出现局部震荡甚至违背单调性。做法上, 一种是对二阶导加入惩罚 (如  $\lambda \int (d''(t))^2 dt$ ), 兼顾拟合与平滑; 另一种是在参数化上直接保证  $d(t)$  正且随  $t$  递减 (如用“积分—指数”重参数化, 或在线性规划/二次规划中加入  $d(t_{k+1}) \leq d(t_k)$ 、 $d(t) \geq 0$  等约束)。第三是加权与稳健性: 不同债券的报价误差与流动性差异较大, 实践中常对回归做加权 (如按久期或价差反比加权), 或在目标函数中采用稳健损失。第四是结点评价与外推: 基函数法天然给出全期限的  $\hat{d}(t)$ , 便于插值; 外推时应配合经济约束 (如远端贴现率趋于零) 与基函数的尾部形状, 避免不合理的远端波动。

严格条件下, 若基函数族足够丰富、调参 (如  $L, \lambda$ ) 得当, 则  $\hat{d}(\cdot)$  在有界区间上一致收敛, 即

$$\sup_{\tau \in [0, \tau^{\max}]} |\hat{d}(\tau) - d(\tau)| \xrightarrow{p} 0,$$

参见 Silverman (1986) 关于平滑估计量的一般一致性结果。

下列 R 代码使用真实数据 (FRED 公布的美国国债常数到期收益率, 作“近似零息的到期收益率”) 构造一篮子“面值成交的附息债券”(各期限息票率取对应常数到期收益率, 半年付息), 并在此基础上采用基函数展开 + 平滑惩罚估计  $d(t)$ ——即贴现函数的全期限曲线。随后由  $\hat{d}(t)$  推得  $\hat{y}(t) = -\ln \hat{d}(t)/t$  与  $\hat{f}(t) = \frac{d}{dt} \{t \hat{y}(t)\}$ 。该代码是“由附息债估计收益率曲线”的最小可运行示例: 将“价格 = 现金流  $\times$  贴现因子”的线性定价关系, 改写为“关于基函数系数的线性回归 + 惩罚项”, 以在 {拟合误差—曲线平滑—(软) 无套利} 之间取得平衡。需注意: FRED 的常数到期收益率并非逐点抽取的零息曲线, 故本例主要用于方法流程演示; 如有 CRSP/STRIPS 或交易所实际债券的现金流与价格数据, 可在同

一框架下以实盘现金流矩阵  $B$  与价格向量  $p$  替换之。

```

1 # =====
2 # 基函数展开法：由“附息债（基于 FRED 常数到期收益率构造）”估计贴现曲线 d(t)
3 # 步骤：
4 # 1) 从 FRED 获取美国国债常数到期收益率（DGS 系列，年化简单利率，%）。
5 # 2) 构造一篮子“面值成交”的半年付息债券：票面利率 = 对应常数到期收益率。
6 # 3) 用 B 样条基函数 g_i(t) 展开 d(t) $\sum_i g_i(t)$ ，目标函数 = 最小二乘 + 平滑惩罚项。
7 # 4) 通过高权重伪观测施加强约束“t=0 时 d(0)=1”，并施加单调性/平滑的软约束。
8 # 5) 得到 d_hat(t)，并计算 y_hat(t)、f_hat(t)，绘制曲线图。
9 # 依赖: quantmod, splines
10 # =====
11
12 suppressPackageStartupMessages({
13 library(quantmod)
14 library(splines)
15 })
16
17 # ----- 1) 选择期限与FRED代码 -----
18 taus <- c(0.5, 1, 2, 3, 5, 7, 10, 20, 30) # 年，至少半年起，便于半年付息
19 fred <- c("DGS6MO", "DGS1", "DGS2", "DGS3", "DGS5", "DGS7", "DGS10", "DGS20", "DGS30")
20
21 getSymbols(fred, src = "FRED", auto.assign = TRUE, warnings = FALSE)
22 Y <- do.call(merge, lapply(fred, get))
23 Yc <- Y[stats::complete.cases(Y)]
24 stopifnot(nrow(Yc) > 0)
25 last_row <- tail(Yc, 1)
26 asof <- index(last_row)
27 y_ann_pct <- as.numeric(last_row) # FRED: 百分比，年化简单/债券口径
28 y_ann <- y_ann_pct/100
29
30 # ----- 2) 构造“面值成交”的半年付息息票债 -----
31 # 每个期限 tau 对应一只债：票息率=该期限常到收益率（年化简单），半年付息
32 # 票面=1，则每期息票=c/2，最后一期=1+c/2；理论上面值成交 -> 价格=1
33 bond_list <- vector("list", length(taus))
34 names(bond_list) <- paste0("T", taus, "Y")
35
36 for (k in seq_along(taus)) {
37 Tm <- taus[k]
38 cR <- y_ann[k] # 年化简单票息率（用于示范）
39 N <- as.integer(round(2*Tm)) # 半年期数
40 tps <- (1:N)/2 # 现金流时点（单位：年）
41 cf <- rep(cR/2, N); cf[N] <- cf[N] + 1
42 price <- 1 # 面值成交（示范：把par yield当成券的票息）
43 bond_list[[k]] <- list(t=tps, cf=cf, p=price)
44 }

```

```

46 # ----- 3) 基函数设计: B样条 + 二阶差分惩罚 -----
47 # 设 $d(t) = \sum g_i(t)$ 。构建回归矩阵 $X_{ij} = \sum_j b_{ij} g_i(t_j)$
48 # 选择样条结点 (含端点), 次数=3, 含截距
49 t_max <- max(taus)
50 n_basis <- 12 # 基函数数量 (可调; 越多越灵活)
51 knots_inner <- seq(0.5, t_max-0.5, length.out = max(0, n_basis-4))
52 gfun <- function(tt) bs(tt, degree = 3, knots = knots_inner, Boundary.knots
53 = c(0, t_max), intercept = TRUE)
54
55 # 构造X与p
56 # 收集所有债的价格
57 p_vec <- sapply(bond_list, function(b) b$p)
58 # 计算每只债的“基函数现金流矩阵行”
59 X <- matrix(0, nrow=length(bond_list), ncol=ncol(gfun(c(0, t_max))))
60 colnames(X) <- paste0("g", seq_len(ncol(X)))
61
62 for (i in seq_along(bond_list)) {
63 bi <- bond_list[[i]]
64 G <- gfun(bi$t) # 维度: 期数 × 基函数数
65 X[i,] <- colSums(G * bi$cf) # $\sum_j cf_j g_i(t_j)$
66 }
67
68 # 在 $t=0$ 处加入强约束 $d(0)=1$ —— 用“高权重伪观测”实现
69 G0 <- gfun(0) # $g_i(0)$
70 w0 <- 1e6 # 权重够大 -> 近似等式约束
71 X_aug <- rbind(X, sqrt(w0)*G0) # 伪观测行: $\sqrt{w_0} \cdot g(0)$
72 p_aug <- c(p_vec, sqrt(w0)*1) # 伪价格: $\sqrt{w_0} \cdot 1$
73
74 # 二阶差分惩罚 (近似 $(d''(t))^2$) , 对 的二阶差分实施L2惩罚
75 D2 <- diff(diag(ncol(X)), differences = 2)
76 lambda <- 1e-2 # 平滑强度 (可调: $10^{-4} \sim 10^{-1}$)
77 R <- crossprod(D2) # 惩罚矩阵
78 # 也可考虑在极端场景下用更大的 lambda, 或在短端加局部权重
79
80 # 带惩罚的闭式解: $(X'X + R)^{-1} X'p$
81 XtX <- crossprod(X_aug)
82 Xtp <- crossprod(X_aug, p_aug)
83 theta_hat <- solve(XtX + lambda*R, Xtp)
84
85 # ----- 4) 由 θ 得到 $d_{\hat{\theta}}(t)$, $y_{\hat{\theta}}(t)$, $f_{\hat{\theta}}(t)$ -----
86 grid <- seq(0, t_max, by = 0.01)
87 G_grid <- gfun(grid)
88 d_hat <- as.vector(G_grid %*% theta_hat)
89 # 数值安全: 避免极小负值 (数值误差)
90 d_hat[d_hat <= 1e-10] <- 1e-10
91 # $y_{\hat{\theta}}$, $f_{\hat{\theta}}$ (连续复利口径)
92 y_hat <- -log(d_hat) / pmax(grid, 1e-8) # $y(0)$ 用极小数规避除零
93 # $f(t) = d/dt \{ e^{y(t)} \}$, 用局部差分近似
94 g_tau <- grid * y_hat
95 f_hat <- c(diff(g_tau)/diff(grid), NA)
96 f_hat[length(f_hat)] <- f_hat[length(f_hat)-1]

```

```

96
97 # ----- 5) 质量检查: 用 d_hat 回定价, 看与 “面值=1” 的误差 -----
98 p_fitted <- numeric(length(bond_list))
99 for (i in seq_along(bond_list)) {
100 bi <- bond_list[[i]]
101 d_i <- approx(grid, d_hat, xout = bi$t, rule = 2)$y
102 p_fitted[i] <- sum(bi$cf * d_i)
103 }
104 fit_tab <- data.frame(
105 tau = taus,
106 p_target = p_vec,
107 p_fitted = p_fitted,
108 abs_err = p_fitted - p_vec
109)
110
111 cat("==== FRED par-yield 构造息票债; 基函数平滑贴现曲线 ===\n")
112 cat("As of:", as.character(asof), "\n\n")
113 print(round(fit_tab, 6), row.names = FALSE)
114
115 # ----- 6) 作图 -----
116 op <- par(mfrow=c(1,3), mar=c(4,4,2,1))
117 plot(grid, d_hat, type="l", lwd=2, xlab="t (years)", ylab="d(t)",
118 main="贴现函数 d(t) (B样条 + 平滑惩罚) ")
119 abline(h=1, v=0, col="grey80", lty=3)
120 points(0, 1, pch=19, col="steelblue")
121
122 plot(grid, y_hat*100, type="l", lwd=2, xlab="t (years)",
123 ylab="y(t) [% p.a., cont.]", main="连续复利收益率 y(t)")
124 points(taus, log(1+y_ann)/taus * taus * 100 / taus, pch=19, col="grey40") #
125 仅作参考
126 # 注: 上句只是提醒原始点, 并非严格对齐口径; 正式应逐点统一口径
127
128 plot(grid, f_hat*100, type="l", lwd=2, xlab="t (years)",
129 ylab="f(t) [% p.a.]", main="瞬时远期 f(t)")
129 par(op)

```

读者在实证应用中可将示例中的 `bond_list` 直接替换为真实的“息票债现金流与价格”样本  $(t, cf, p)$ , 回归矩阵的构造方式  $X_{i\ell} = \sum_j cf_{ij} g_\ell(t_{ij})$  保持不变。为得到更平滑或更灵活的曲线, 可适当增大平滑参数  $\lambda$  或基函数数量 (并相应增大  $\lambda$  以抑制振荡)。若需严格确保贴现函数单调递减, 可将“软约束”的平滑惩罚替换为带单调性 (及非负性) 约束的二次规划求解。

### 10.2.2 参数法

参数法通过给定一个结构化、经济含义明确的函数族来描述整条曲线, 以少量参数同时控制短端、长端与中段的形状。常用模型为 Nelson–Siegel (NS, Nelson & Siegel 1987) 及其扩展 Svensson (NSS, Svensson 1994)。NS 从“瞬时远期利率”出发进行设定:

$$f(t) = \beta_0 + (\beta_1 + \beta_2 t/\tau_0) \exp(-t/\tau_0),$$

其中  $\beta_0$  决定远端水平 (level),  $\beta_1$  主要决定斜率 (slope, 短端相对长端的抬升或压低),  $\beta_2$  与  $\tau_0$  共同刻画中段的曲率 (curvature) 位置与强度。由  $f$  可一体化推出贴现  $d_\theta(t)$  与收益率  $y_\theta(t)$ 。Svensson 在 NS 基础上加入二次衰减分量:

$$f(t) = \beta_0 + \beta_1 e^{-t/\tau_1} + \beta_2(t/\tau_1) e^{-t/\tau_1} + \beta_3(t/\tau_2) e^{-t/\tau_2},$$

以增强对长端/次峰的刻画能力。

估计上, 通常以“价格误差最小二乘法”拟合参数  $\theta$ :

$$\hat{\theta} = \arg \min_{\theta} \sum_{i=1}^n \left( p_i - \sum_{j=1}^{m_i} b_{ij} d_\theta(\tau_{ij}) \right)^2,$$

亦可用“收益率误差”或“远期误差”作为准则 (但价格准则更符合无套利线性定价)。实践中若干要点: 其一, 初值与尺度——NS/NSS 为非线性最小二乘, 良好初值有助于收敛性与全局最优化;  $\tau_0, \tau_1, \tau_2$  的量纲为“年”, 初值可取 1–3 年与 5–10 年以覆盖短中长期限。其二, 约束——为避免不合理的极端曲线形态, 常加参数边界 (如  $\tau_k > 0, |\beta_\ell|$  不宜过大), 必要时在目标函数中加入轻度正则项。其三, 加权——为反映报价精度与流动性差异, 可按债券或期限段设置权重; 亦可先将个券转换为“面值息票”的等价现金流以降低异质性。其四, 诊断——拟合后检查残差 (是否系统性偏离短端或长端)、曲线形状 (是否出现不必要的振荡), 并与市场平价曲线/掉期曲线对照。NS/NSS 的优点在于可解释性强、插值与外推稳定、参数维度低; 但在极端短端或长端处可能失配, 且难以同时兼顾局部细节。

在模型正确设定且常规正则条件下,  $\hat{\theta}$  具有一致性与渐近正态性,

$$\sqrt{n}(\hat{\theta} - \theta) \xrightarrow{d} N(0, \Omega), \quad d_{\hat{\theta}}(\tau) \rightarrow d_\theta(\tau) \quad (n \rightarrow \infty),$$

但在真实市场中, 曲线形态会随制度、税收、基差与流动性而变, 因此“模型错设”较常见。实务上宜将 NS/NSS 视为稳健的低维近似基, 并配合滚动估计、参数平滑与异常值处理。

以下 R 代码通过“价格准则”估计 Nelson–Siegel (NS) 与 Svensson (NSS) 参数, 并由此生成整条贴现、收益率与远期曲线。思路与前一节一致: 先用 FRED 的美国国债常数到期收益率构造一篮子“面值成交”的半年付息息票债 (各期限票面利率取对应到期收益率, 作为教学近似), 再以 {非线性最小二乘} 最小化“模型价格与面值之差”的平方和拟合参数。拟合完成后, 用解析积分公式由瞬时远期  $f(t)$  推出贴现  $d_\theta(t) = \exp\{-\int_0^t f(s) ds\}$ 、收益率  $y_\theta(t) = -\ln d_\theta(t)/t$  与远期  $f_\theta(t)$ , 并给出残差诊断与曲线作图。需强调: FRED 的长期收益率并非逐点抽取的零息利率, 故本例侧重演示方法流程; 若有真实“息票现金流 + 市场价格”样本, 可在相同框架下直接替换现金流与价格矩阵。

```

1 # =====
2 # 参数法 (NS / NSS) : 利用“价格准则”估计曲线, 并输出 d(t), y(t), f(t)
3 # 依赖: quantmod
4 # =====
5
6 suppressPackageStartupMessages(library(quantmod))
7
8 # 1) 选择期限与FRED代码 (单位: 年)
9 taus <- c(0.5, 1, 2, 3, 5, 7, 10, 20, 30)
10 fred <- c("DGS6MO", "DGS1", "DGS2", "DGS3", "DGS5", "DGS7", "DGS10", "DGS20", "DGS30")

```

```

11
12 getSymbols(fred, src = "FRED", auto.assign = TRUE, warnings = FALSE)
13 Y <- do.call(merge, lapply(fred, get))
14 Yc <- Y[stats::complete.cases(Y)]
15 stopifnot(nrow(Yc) > 0)
16 last_row <- tail(Yc, 1)
17 asof <- index(last_row)
18 y_ann <- as.numeric(last_row)/100 # 年化简单 (债券口径)
19
20 # 2) 构造“面值成交”的半年付息票债 (票面=1; 票息率取对应常数)
21 bond_list <- vector("list", length(taus))
22 names(bond_list) <- paste0("T", taus, "Y")
23 for (k in seq_along(taus)) {
24 Tm <- taus[k]
25 cR <- y_ann[k] # 年化简单票息率
26 N <- as.integer(round(2*Tm)) # 半年期数
27 tps <- (1:N)/2 # 支付时点 (年)
28 cf <- rep(cR/2, N); cf[N] <- cf[N] + 1
29 price <- 1 # 面值成交 (用于价格准则)
30 bond_list[[k]] <- list(t=tps, cf=cf, p=price)
31 }
32
33 # 3) 定义 NS 与 NSS 的贴现/远期函数
34 # NS: f(t) = 0 + (1 + 2 t/0) e^{-t/0}
35 # _0^t f(s) ds = 0 t + 1 0 (1 - e^{-t/0}) + 2 [0 - (t+0) e^{-t/0}]
36 A_NS <- function(t, theta) {
37 b0 <- theta[1]; b1 <- theta[2]; b2 <- theta[3]; tau0 <- theta[4]
38 E <- exp(-t/tau0)
39 b0*t + b1*tau0*(1 - E) + b2*(tau0 - (t + tau0)*E)
40 }
41 d_NS <- function(t, theta) exp(- A_NS(t, theta))
42 f_NS <- function(t, theta) {
43 b0 <- theta[1]; b1 <- theta[2]; b2 <- theta[3]; tau0 <- theta[4]
44 b0 + (b1 + b2*t/tau0) * exp(-t/tau0)
45 }
46
47 # NSS: 在NS基础上加 3 (t/2) e^{-t/2}
48 # _0^t 3 (s/2) e^{-s/2} ds = 3 [2 - (t+2) e^{-t/2}]
49 A_NSS <- function(t, theta) {
50 b0 <- theta[1]; b1 <- theta[2]; b2 <- theta[3]; tau1 <- theta[4]; b3 <-
51 theta[5]; tau2 <- theta[6]
52 E1 <- exp(-t/tau1); E2 <- exp(-t/tau2)
53 b0*t + b1*tau1*(1 - E1) + b2*(tau1 - (t + tau1)*E1) + b3*(tau2 - (t + tau2)
54)*E2)
55 }
56 d_NSS <- function(t, theta) exp(- A_NSS(t, theta))
57 f_NSS <- function(t, theta) {
58 b0 <- theta[1]; b1 <- theta[2]; b2 <- theta[3]; tau1 <- theta[4]; b3 <-
59 theta[5]; tau2 <- theta[6]
60 b0 + b1*exp(-t/tau1) + b2*(t/tau1)*exp(-t/tau1) + b3*(t/tau2)*exp(-t/tau2)

```

```

58 }
59
60 # 4) 定义“价格准则”目标函数 (NS / NSS)
61 price_error_NS <- function(theta) {
62 # = (0, 1, 2, 0), 约束: 0 > 0
63 if (theta[4] <= 1e-6) return(1e12)
64 err <- 0
65 for (i in seq_along(bond_list)) {
66 bi <- bond_list[[i]]
67 di <- d_NS(bi$t, theta)
68 p_hat <- sum(bi$cf * di)
69 err <- err + (p_hat - bi$p)^2
70 }
71 err
72 }
73 price_error_NSS <- function(theta) {
74 # = (0, 1, 2, 1, 3, 2), 约束: 1, 2 > 0
75 if (theta[4] <= 1e-6 || theta[6] <= 1e-6) return(1e12)
76 err <- 0
77 for (i in seq_along(bond_list)) {
78 bi <- bond_list[[i]]
79 di <- d_NSS(bi$t, theta)
80 p_hat <- sum(bi$cf * di)
81 err <- err + (p_hat - bi$p)^2
82 }
83 err
84 }
85
86 # 5) 选择初值与边界并拟合
87 theta0_NS <- c(b0=mean(y_ann), b1=-0.02, b2= 0.02, tau0=2.0)
88 lower_NS <- c(-0.10, -5, -5, 1e-3)
89 upper_NS <- c(0.20, 5, 5, 10.0)
90
91 fit_NS <- optim(par=theta0_NS, fn=price_error_NS, method="L-BFGS-B",
92 lower=lower_NS, upper=upper_NS, control=list(maxit=2000))
93
94 theta0_NSS <- c(b0=mean(y_ann), b1=-0.03, b2=0.06, tau1=2.0, b3=-0.02, tau2
95 =8.0)
96 lower_NSS <- c(-0.10, -5, -5, 1e-3, -5, 1e-3)
97 upper_NSS <- c(0.20, 5, 5, 10.0, 5, 30.0)
98
99 fit_NSS <- optim(par=theta0_NSS, fn=price_error_NSS, method="L-BFGS-B",
100 lower=lower_NSS, upper=upper_NSS, control=list(maxit=3000))
101 cat("== NS 拟合 (价格准则) ==\nAs of:", as.character(asof), "\n")
102 print(fit_NS$par); cat("obj =", fit_NS$value, "\n\n")
103 cat("== NSS 拟合 (价格准则) ==\nAs of:", as.character(asof), "\n")
104 print(fit_NSS$par); cat("obj =", fit_NSS$value, "\n\n")
105
106 # 6) 用拟合参数生成整条曲线并回定价诊断
107 grid <- seq(0.01, max(taus), by=0.01)

```

```

108
109 # NS 曲线
110 d_ns <- d_NS(grid, fit_NS$par)
111 y_ns <- -log(d_ns)/grid
112 f_ns <- f_NS(grid, fit_NS$par)
113
114 # NSS 曲线
115 d_nss <- d_NSS(grid, fit_NSS$par)
116 y_nss <- -log(d_nss)/grid
117 f_nss <- f_NSS(grid, fit_NSS$par)
118
119 # 回定价误差
120 price_fit_tab <- function(d_fun, theta) {
121 data.frame(
122 tau = taus,
123 p_target = sapply(bond_list, function(b) b$p),
124 p_fitted = sapply(bond_list, function(b) sum(b$cf * d_fun(b$t, theta)))
125)->DF
126 DF$abs_err <- DF$p_fitted - DF$p_target
127 DF
128 }
129 tab_NS <- price_fit_tab(d_NS, fit_NS$par)
130 tab_NSS <- price_fit_tab(d_NSS, fit_NSS$par)
131
132 cat("NS 回定价误差 (价格 - 1) :\n"); print(round(tab_NS, 6), row.names =
133 FALSE)
134 cat("\nNSS 回定价误差 (价格 - 1) :\n"); print(round(tab_NSS, 6), row.names =
135 FALSE)
136
137 # 7) 作图: d(t), y(t), f(t)
138 op <- par(mfrow=c(1,3), mar=c(4,4,2,1))
139 plot(grid, d_ns, type="l", lwd=2, col="steelblue",
140 xlab="t (years)", ylab="d(t)", main="贴现函数 d(t)")
141 lines(grid, d_nss, lwd=2, col="tomato")
142 legend("topright", c("NS", "NSS"), lty=1, col=c("steelblue", "tomato"), bty="
143 n")
144
145 plot(grid, 100*y_ns, type="l", lwd=2, col="steelblue",
146 xlab="t (years)", ylab="y(t) [% p.a., cont.]", main="连续复利收益率 y(t")
147 ")
148 lines(grid, 100*y_nss, lwd=2, col="tomato")
149 par(op)

```

使用说明——若有真实的息票债现金流与价格数据，只需将示例中“以 FRED 常见收益率构造的 `bond_list`”替换为你的  $(t, cf, p)$  样本，目标函数与求解流程保持不变；若需加入更严格的无套利约束（如  $d(\tau)$  单调、正性），可在价格准则外叠加惩罚项或改用带约束的二次规划/非线性规划；为提升稳定性，建议对短端和流动性差的债券加权，或对参数

加入温和的边界与正则化，并在拟合后检查残差与曲线形状（短端是否过度弯折、长端是否出现不必要振荡）。

### 10.2.3 Fama–Bliss 法

Fama & Bliss (1987) 提出一种“由短到长、逐段锁定远期利率”的序贯拟合思路。直观地讲，把整个期限区间切分为若干相邻到期区间，并在每个小区间上把远期利率视为常数；然后按照到期从短到长的顺序，依次用较为“干净”的样本精确定定价该区间，从而逐段推出整条远期曲线与贴现函数。该方法的优点在于：每一步只处理一个结构简单的小问题，避免一开始就同时估计大量自由度；同时通过严格的样本筛选尽量剔除噪声债券，因此带有“稳健递推”的特点。与其他曲线拟合方法的系统比较，可参见 Bliss (1997)。

**定义 10.8 (Fama–Bliss 的分段常数远期利率与序贯定价):** 将样本债券按到期由短到长排序，记第  $i$  只债券的到期为  $\tau^i$ （设  $\tau^0 = 0$ ）。在相邻区间  $(\tau^{i-1}, \tau^i]$  上令远期利率保持常数  $f^i$ ，即

$$f(\tau) = f^i, \quad \tau \in (\tau^{i-1}, \tau^i].$$

据此可将贴现函数写为依赖  $\{f^1, \dots, f^i\}$  的形式；然后使用“第  $i$  只债券的价格等于其现金流贴现值之和”的方程求解  $f^i$ ：

$$p_i = \sum_{j=1}^{m_i} b_i(\tau_{ij}) d(\tau_{ij}),$$

其中  $d(\cdot)$  由已求得的  $\{f^1, \dots, f^{i-1}\}$  与当前未知的  $f^i$  共同确定。求得  $f^i$  后进入下一只更长期限债券，如此递推，直至样本最长期限。

在实施时，样本选择尤为关键。通常仅保留完全征税、不可赎回 (non-callable)、非含权 (non-option) 等标准化品种；剔除到期不足一年的票据；并通过阈值规则过滤异常观测值，例如：某只债券的到期收益率与相邻期限的收益率差异不应超过约 0.2%（或位于相邻两者之间），且纳入该债后不应出现“相邻期限大幅且方向相反的跳变”。这些规则的目的，是尽量将曲线的结构信息与个券的交易噪声分离开来，使得每一步递推都基于尽可能干净的信号。

为了方便理解，我们可以把 Fama–Bliss 看作一种“带筛选的序贯回归”。设想最简单的一元回归  $y_i = \beta x_i + u_i$ 。先用全部样本做 OLS，得到残差  $\hat{u}_i$ ，再按  $|\hat{u}_i|$  从小到大排序；如果只“相信”最干净的那一小部分观测（残差最小），以此做一个极简估计，在满足常见正则条件时，该估计仍可与 OLS 在大样本下等价。这个类比表达的意思是：只要筛选机制能有效剔除噪声，基于干净子样本的序贯估计与基于全样本的理想化估计在统计上可以对齐。Fama–Bliss 的递推逻辑与此类似：用严格筛选的样本逐段锁定远期利率，往往能在稳健性与可解释性之间取得较好平衡。

需要注意，Fama–Bliss 的统计性质并非在所有设定下都有完整的极限理论支撑；而“分段常数”远期在曲线非常光滑或存在多个局部“弯点”时，可能显得偏粗。较稳妥的做法，是将其作为“初始曲线”的快速构造工具，再配合样条平滑或 NS/NSS 参数化细化；或在每一步定价等式中叠加温和的正则项与局部单调约束，以提升数值稳定性与无套利一致性。

总体而言，Fama–Bliss 通过“分段常数远期 + 序贯定价 + 严格样本筛选”的组合来拟合收益率曲线；它并非“一步到位”的全局优化，而是“由近及远”的稳健递推。只要筛选规则能有效排除噪声观测，此方法在统计与实务两端均具可取之处。

以下 R 代码按 Fama–Bliss 的“分段常数远期+序贯定价”思路，自短到长逐段抽取远期利率。为便于演示，与前文一致，我们用 FRED 的常数到期收益率在同一观测日 {构造} 一篮子“面值成交”的半年付息票债（各期限票息率取对应到期收益率）；随后在每个相邻期限区间  $(\tau^{i-1}, \tau^i]$  假设瞬时远期为区间常数  $f^i$ ，并用第  $i$  只债券的 {价格等式} 逐段求解  $f^i$ 。得到分段常数远期后，生成贴现函数  $d(t) = \exp\{-\int_0^t f(s) ds\}$ 、连续复利收益率  $y(t) = -\ln d(t)/t$  与整条远期曲线  $f(t)$ ，并给出回定价误差与作图。若有真实“息票债现金流+价格”样本，可将构造环节替换为你的  $(t, cf, p)$  数据，后续步骤与求解不变。

```

1 # =====
2 # Fama – Bliss 序贯引导 (分段常数远期) : 由短到长抽取 f^i
3 # 依赖: quantmod
4 # =====
5
6 suppressPackageStartupMessages(library(quantmod))
7
8 # 1) 期限与 FRED 代码 (单位: 年) ; 至少半年起, 便于半年付息
9 taus <- c(0.5, 1, 2, 3, 5, 7, 10, 20, 30)
10 fred <- c("DGS6M0", "DGS1", "DGS2", "DGS3", "DGS5", "DGS7", "DGS10", "DGS20", "DGS30"
11 ")
12
13 getSymbols(fred, src = "FRED", auto.assign = TRUE, warnings = FALSE)
14 Y <- do.call(merge, lapply(fred, get))
15 Yc <- Y[complete.cases(Y)]
16 stopifnot(nrow(Yc) > 0)
17 last_row <- tail(Yc, 1)
18 asof <- index(last_row)
19 par_y <- as.numeric(last_row)/100 # 年化简单 (债券口径)
20
21 # 2) 构造“面值成交”的半年付息票债 (票面=1; 票息率取对应常到)
22 # 仅用于演示; 有真实 (t,cf,p) 时, 直接替换 bond_list
23 make_par_bond <- function(Tm, cR) {
24 N <- as.integer(round(2*Tm)) # 半年期数
25 tps <- (1:N)/2 # 支付时点 (年)
26 cf <- rep(cR/2, N); cf[N] <- cf[N] + 1
27 list(t = tps, cf = cf, p = 1)
28 }
29 bond_list <- mapply(make_par_bond, Tm = taus, cR = par_y, SIMPLIFY = FALSE)
30 names(bond_list) <- paste0("T", taus, "Y")
31
32 # 3) 分段常数远期下的贴现: d(t)=exp{-_0^t f(s)ds}, 其中 f(s)=f^k on (^{k-1}, ^k]
33 # 给定分段端点 tau_knots 与 f_vec (长度等于区间数), 计算任意 t 的 d(t)
34 discount_from_piecewise_f <- function(t, tau_knots, f_vec) {
35 # tau_knots: c(^0, ^1, ..., ^I); f_vec: length I (每段常数)
36 I <- length(f_vec)
37 stopifnot(length(tau_knots) == I + 1)
38 # 对每个 t, 累加整段 f^k Δ_k + 最后一段的残余
39 integ <- sapply(t, function(tt) {
40 if (tt <= 0) return(0)
41 # 找到落在哪一段
42 k <- max(which(tau_knots < tt)))

```

```

42 k <- min(k, I)
43 # 完整段的和
44 full <- 0
45 if (k >= 1) {
46 full <- sum(f_vec[1:(k-1)] * diff(tau_knots)[1:(k-1)])
47 # 第 k 段的残余长度
48 rem <- tt - tau_knots[k]
49 full <- full + f_vec[k] * rem
50 }
51 full
52 })
53 exp(-integ)
54 }
55
56 # 4) 第 i 段引导: 已知 f^1, \dots, f^{i-1} , 用第 i 只债的价格等式解 f^i
57 # 单变量方程, 用 uniroot 求解; 必要时扩张搜索区间
58 solve_f_i <- function(i, f_vec, tau_knots, bond_list, bracket = c(-0.05,
59 0.20)) {
60 bi <- bond_list[[i]]
61 obj <- function(fi) {
62 f_tmp <- f_vec
63 f_tmp[i] <- fi
64 d_i <- discount_from_piecewise_f(bi$t, tau_knots[1:(i+1)], f_tmp[1:i])
65 sum(bi$cf * d_i) - bi$p
66 }
67 a <- bracket[1]; b <- bracket[2]
68 # 扩张搜索区间, 直到目标函数异号或达到上限
69 iter <- 0; max_iter <- 20
70 while (iter < max_iter && obj(a)*obj(b) > 0) {
71 a <- a - 0.05; b <- b + 0.05; iter <- iter + 1
72 }
73 if (obj(a)*obj(b) > 0) stop("无法在合理区间内找到根: 请检查样本或调整初值/
74 区间。")
75 uniroot(obj, interval = c(a, b), tol = 1e-10)$root
76 }
77
78 # 5) 自短到长顺序引导 f^i
79 I <- length(taus)
80 tau_knots <- c(0, taus) # ^0=0, ^I=最长期限
81 f_hat <- rep(NA_real_, I)
82 for (i in 1:I) {
83 # 第 i 段只需要用到第 i 只债 (其现金流不会超过 ^i)
84 # 已有 $f^1 \dots f^{i-1}$, 现在解 f^i
85 if (i == 1) {
86 f_prev <- numeric(1)
87 } else {
88 f_prev <- f_hat[1:(i-1)]
89 }
90 f_hat[i] <- solve_f_i(i,
91 f_vec = c(f_prev, NA),

```

```

91 tau_knots = tau_knots,
92 bond_list = bond_list)
93 }
94
95 # 6) 生成整条曲线: 分段常数 f(t)、贴现 d(t)、收益率 y(t)
96 grid <- seq(0, max(taus), by = 0.01)
97 # f(t) piecewise-constant: 在每个格点找所属区间
98 f_grid <- sapply(grid, function(tt) {
99 if (tt <= 0) return(f_hat[1])
100 k <- max(which(tau_knots < tt)); k <- min(k, I)
101 f_hat[k]
102 })
103 d_grid <- discount_from_piecewise_f(grid, tau_knots, f_hat)
104 d_grid[d_grid <= 1e-12] <- 1e-12
105 y_grid <- -log(d_grid)/pmax(grid, 1e-8) # 连续复利收益率
106
107 # 7) 固定价误差 (应基本接近 0)
108 p_fitted <- sapply(seq_along(bond_list), function(i) {
109 bi <- bond_list[[i]]
110 d_i <- discount_from_piecewise_f(bi$t, tau_knots[1:(i+1)], f_hat[1:i])
111 sum(bi$cf * d_i)
112 })
113 fit_tab <- data.frame(
114 tau = taus,
115 p_target = 1,
116 p_fitted = p_fitted,
117 abs_err = p_fitted - 1
118)
119
120 cat("==== Fama - Bliss 分段常数远期 (序贯引导) ===\n")
121 cat("As of:", as.character(asof), "\n\n")
122 print(round(fit_tab, 8), row.names = FALSE)
123
124 # 8) 作图
125 op <- par(mfrow=c(1,3), mar=c(4,4,2,1))
126 # d(t)
127 plot(grid, d_grid, type="l", lwd=2, xlab="t (years)", ylab="d(t)",
128 main="贴现函数 d(t) (Fama - Bliss) ")
129 abline(h=1, v=0, col="grey80", lty=3)
130 # y(t)
131 plot(grid, 100*y_grid, type="l", lwd=2, xlab="t (years)",
132 ylab="y(t) [% p.a., cont.]", main="连续复利收益率 y(t)")
133 # f(t) 分段常数
134 plot(grid, 100*f_grid, type="s", lwd=2, xlab="t (years)",
135 ylab="f(t) [% p.a.]", main="分段常数远期 f(t)")
136 abline(v = taus, col="grey85", lty=3)
137 par(op)

```

## 10.3 离散时间的债券定价模型

本节讨论在离散时间框架下的收益率曲线建模：收益率如何随时间演化？如何在该框架下对固定收益衍生品进行定价？

### 10.3.1 利率的经济学假说

关于利率与收益率的决定，经典文献中有四类常见假说，它们从不同角度刻画远期利率、短期利率的预期以及期限溢价之间的关系。

**定义 10.9 (预期假说 (Expectations Hypothesis) )：** 在适当的测度下，远期利率等于未来对应一期短期即期利率的条件期望。

**定义 10.10 (流动性 (风险) 溢价假说 (Liquidity/Risk Premium Hypothesis) )：** 远期利率等于未来短期利率的条件期望，加上一项与期限相关的风险溢价（或流动性溢价）。

**定义 10.11 (市场分割假说 (Market Segmentation Hypothesis) )：** 不同期限区间构成相对独立的子市场，各期限的利率主要由该期限特定的供需关系与制度约束决定。

**定义 10.12 (偏好栖息地 (Preferred Habitat) )：** 投资者对某些期限区间存在偏好；若给予足够溢价，投资者会偏离其偏好期限，从而在不同期限的资产之间形成可交易的风险补偿。

现代期限结构理论在上述直觉基础上，对风险溢价的形式施加更明确的可检验约束（例如射类模型下的线性风险价格假设），使“预期与溢价”的分解能够被识别与估计。

### 10.3.2 收益率的统计性质

早期研究（例如 Campbell et al. (1997) 基于 McCulloch & Kwon (1993) 的 1952–1991 年固定到期样本）总结了若干稳健事实：平均收益率曲线整体上行且呈凹形；单期曲线在截面上可能上行、下行或呈驼峰形态；收益率的时间序列标准差对期限不敏感；短端偏度与超额峰度较高且为正，并随期限延长而下降；原序列自相关显著且随期限增加而增强。本节在这一脉络下，利用近二十五年的数据做一组可比的更新。

具体地，我们选取 FRED 公布的美国国债“固定期限收益率”（Constant Maturity Yield）作为近似零息的截面（到期期限为 1、3、6、12、24、36、60、120、240、360 个月），样本区间为 2000–2024 年日频数据。需要说明的是，1 个月期与 30 年期在部分年份存在数据缺失，但不影响在各自可得区间内的描述统计与相关性计算。图 10.1 给出了 1 个月期与 120 个月期两条典型期限的时间序列：可以看到，短端在 2008–2016 年长期贴近零利率下限（ZLB），而 2022 年后伴随快速加息显著抬升；相较之下，10 年期收益率的趋势更平缓、波动更小。图 10.2 则展示了 1 个月期收益率的日变化，2008–2009 年与 2020 年附近波动显著放大，随后总体回落，反映了危机与流动性冲击聚集在少数时段的事实。

表 10.1 报告了 2000–2024 年各期限收益率的均值 ( $m$ )、标准差 ( $s$ )、偏度 ( $\kappa_3$ ) 与超额峰度 ( $\kappa_4$ )。可以看到，平均曲线整体随期限上行（从 1M 的 1.55% 升至 30Y 的 3.87%），与“上凸”的经典结论一致；标准差随期限递减（短端  $s \approx 1.99\%$ ，长端  $s \approx 1.18\%$ ），显

示短端更易受政策与流动性冲击影响；偏度在短端显著为正（1M 为 1.03），随期限接近零后在远端略为负；超额峰度在本样本期内多数期限为负值，说明相较正态分布的尖峰厚尾特征有所收敛（与 1952–1991 年的结果不同，这一变化与 ZLB 时期的长期压平效应不无关系）。

表 10.2 报告了差分收益率  $\Delta y_t$  的 1–5 阶自相关系数。极短端（1M、3M）在一阶自相关上仍表现出一定的短期可预测性（0.14 与 0.13 左右），而期限拉长后，自相关迅速逼近零并在若干阶呈现轻微负值，这与“原序列近似单位根、差分后弱相关”的认识相符。与早期样本相比，本期短端差分相关性总体偏低，反映出 ZLB 与逆回购等微观机制的变化，使短端变化更多由少数事件驱动而非常态持续性。

就整体结论而言：在 2000–2024 年这一更“现代”的样本里，平均收益率曲线仍随期限上行；短端的波动与偏度明显高于长端；差分收益率的短期相关性集中于最短端并快速衰减。与 1952–1991 年相比，长期的零利率下限约束与疫情冲击使短端水平与波动呈现阶段性“极值化”，而长端统计特征相对稳定。这些事实提示，在离散时间建模与预测时，短端宜采用能容纳结构性约束与突发波动的模型（如分段/断点、时变波动或跳跃成分），而中长端可继续使用线性—高斯或温和非线性的传统计量结构。

以下 R 代码以 FRED 的固定期限收益率（2000–2024）为样本，计算各期限日频收益率的均值、标准差、偏度与超额峰度，并给出差分收益率的一到五阶自相关系数，从而复现实证文献中关于收益率统计性质的关键事实；同时绘制并保存“1 个月期与 120 个月期收益率”的对比图和“1 个月期收益率日变化”的时间序列图，以便直观呈现短端贴近零利率下限、加息期陡升与波动聚集等特征。

```

1 # =====
2 # 统计性质复现：2000 – 2024 日频收益率的描述统计与差分自相关
3 # 依赖：quantmod, zoo, moments
4 # =====
5
6
7 # 设置工作目录
8 if (requireNamespace("rstudioapi", quietly = TRUE) &&
9 rstudioapi::isAvailable()) {
10 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
11 }
12 suppressPackageStartupMessages({
13 library(quantmod)
14 library(zoo)
15 library(moments) # skewness(), kurtosis()
16 })
17
18 # 1) 设定期限与FRED代码（单位：月 -> 年）
19 mats_month <- c(1,3,6,12,24,36,60,120,240,360)
20 fred_codes <- c("DGS1MO", "DGS3MO", "DGS6MO", "DGS1", "DGS2", "DGS3",
21 "DGS5", "DGS10", "DGS20", "DGS30")
22
23 # 2) 下载 2000-01-01 至 2024-12-31 的数据；注意 1MO 与 30Y 存在缺段
24 from <- as.Date("2000-01-01"); to <- as.Date("2024-12-31")
25 getSymbols(fred_codes, src = "FRED", from = from, to = to, auto.assign =
26 TRUE)
27 # 合并为 xts，列名为到期（月）

```

```

28 Y <- do.call(merge, mget(fred_codes))
29 colnames(Y) <- paste0(mats_month, "M")
30
31 # 3) 转换为数值矩阵并去掉全NA的列; 转为小数 (而非%)
32 Ynum <- Y / 100
33 # 这里只在各列各自可用的时间区间内计算统计量, 不强求完全交集
34 # 函数: 对单列xts计算统计量
35 stats_one <- function(x) {
36 x <- na.omit(as.numeric(x))
37 if (length(x) < 10) return(c(m=NA, s=NA, k3=NA, k4=NA))
38 c(m = mean(x, na.rm=TRUE)*100, # 转回百分数便于展示
39 s = sd(x, na.rm=TRUE)*100,
40 k3= skewness(x, na.rm=TRUE),
41 k4= kurtosis(x, na.rm=TRUE) - 3) # 超峰度
42 }
43
44 # 4) 生成表13.1: 各期限收益率的 m, s, 3, 4
45 tab1 <- t(apply(Ynum, 2, stats_one))
46 tab1 <- as.data.frame(tab1)
47 tab1$Maturity <- rownames(tab1)
48 tab1 <- tab1[, c("Maturity", "m", "s", "k3", "k4")]
49 rownames(tab1) <- NULL
50
51 # 5) 生成表13.2: 差分收益率的自相关 (1-5阶)
52 acf_one <- function(x, maxlag=5) {
53 x <- diff(na.omit(as.numeric(x)))
54 if (length(x) < (maxlag+10)) return(rep(NA, maxlag))
55 acf(x, plot=FALSE, na.action=na.pass)$acf[2:(maxlag+1)]
56 }
57 acf_mat <- t(apply(Ynum, 2, acf_one, maxlag=5))
58 colnames(acf_mat) <- paste0("rho_D(", 1:5, ")")
59 tab2 <- as.data.frame(acf_mat)
60 tab2$Maturity <- rownames(acf_mat)
61 tab2 <- tab2[, c("Maturity", colnames(acf_mat))]
62 rownames(tab2) <- NULL
63
64 # 6) 打印结果 (四舍五入以便排版)
65 cat("==== Table 13.1 日频收益率的描述统计 (2000-2024) ===\\n")
66 print(within(tab1, { m=round(m,4); s=round(s,4); k3=round(k3,4); k4=round(k4,4)}), row.names=FALSE)
67
68 cat("\\n==== Table 13.2 差分收益率的自相关 (1-5阶, 2000-2024) ===\\n")
69 print(within(tab2, {
70 `rho_D(1)`=round(`rho_D(1)`,4); `rho_D(2)`=round(`rho_D(2)`,4);
71 `rho_D(3)`=round(`rho_D(3)`,4); `rho_D(4)`=round(`rho_D(4)`,4);
72 `rho_D(5)`=round(`rho_D(5)`,4)
73 }), row.names=FALSE)
74
75 # 7) 可选: 绘制两条典型期限 (1个月 与 120个月) 的曲线, 以对比不同期限的水平
76 与趋势

```

```

77 # 使用分析区间 (2000-01-01 至 2024-12-31)
78 Ywin <- window(Y, start = from, end = to)
79
80 # 1个月期与120个月期共用纵轴范围
81 ylim12 <- range(Ywin[, c("1M", "120M")], na.rm = TRUE)
82
83 # 图1: 1个月期 vs 120个月期收益率 (保存为 PNG)
84 png("fig_yields_1M_120M_2000_2024.png", width = 1600, height = 900, res =
 200)
85 par(mar = c(4, 4, 2, 1))
86 plot(index(Ywin), coredata(Ywin[, "1M"]),
 type = "l", col = "steelblue", lwd = 1.5,
 ylim = ylim12, xlab = "", ylab = "Percent",
 main = "1M vs 120M Yields (2000 - 2024)")
87 lines(index(Ywin), coredata(Ywin[, "120M"]), col = "tomato", lwd = 1.5)
88 abline(h = 0, col = "grey80", lty = 3)
89 legend("topright", c("1M", "120M"), lty = 1, col = c("steelblue", "tomato"),
90 lwd = 1.5, bty = "n")
91 dev.off()
92
93 # 图2: 1个月期收益率的日度增量 Δy (保存为 PNG)
94 Dy1 <- diff(Ywin[, "1M"])
95 png("fig_dyield_1M_2000_2024.png", width = 1600, height = 900, res = 200)
96 par(mar = c(4, 4, 2, 1))
97 plot(index(Dy1), coredata(Dy1),
 type = "h", col = "steelblue",
 xlab = "", ylab = "Δ Percent",
 main = "ΔYield: 1M (2000 - 2024)")
98 abline(h = 0, col = "grey70", lty = 3)
99 dev.off()
100
101
102
103
104
105

```

表 10.1: 日频收益率的描述统计 (2000–2024 年) —— 均值  $m$ 、标准差  $s$  (单位: %), 偏度  $\kappa_3$  与超峰度  $\kappa_4$

| 统计量            | 1M      | 3M      | 6M      | 12M     | 24M     | 36M     | 60M     | 120M    | 240M    | 360M    |
|----------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 均值 $m$         | 1.5525  | 1.8535  | 1.9534  | 2.0237  | 2.1983  | 2.3717  | 2.7284  | 3.2833  | 3.7965  | 3.8742  |
| 标准差 $s$        | 1.7899  | 1.9856  | 1.9846  | 1.9016  | 1.7732  | 1.6630  | 1.4982  | 1.3058  | 1.3021  | 1.1793  |
| 偏度 $\kappa_3$  | 1.0263  | 0.8257  | 0.7801  | 0.7063  | 0.6392  | 0.5624  | 0.4056  | 0.1212  | -0.0287 | -0.1052 |
| 超峰度 $\kappa_4$ | -0.2768 | -0.7543 | -0.8070 | -0.8556 | -0.7504 | -0.7105 | -0.6551 | -0.7196 | -0.9562 | -0.8926 |

表 10.2: 差分收益率的自相关 (1–5 阶, 2000–2024 年)

| 阶数               | 1M      | 3M      | 6M      | 12M     | 24M     | 36M     | 60M     | 120M    | 240M    | 360M    |
|------------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| $\rho_\Delta(1)$ | 0.1412  | 0.1326  | 0.0632  | 0.0433  | -0.0185 | -0.0086 | -0.0051 | -0.0037 | -0.0165 | -0.0099 |
| $\rho_\Delta(2)$ | -0.0716 | -0.0890 | -0.0317 | -0.0021 | -0.0417 | -0.0483 | -0.0597 | -0.0492 | -0.0419 | -0.0471 |
| $\rho_\Delta(3)$ | -0.1488 | -0.1220 | -0.0587 | -0.0169 | -0.0045 | -0.0018 | 0.0055  | 0.0064  | -0.0024 | -0.0075 |
| $\rho_\Delta(4)$ | 0.0125  | 0.0366  | 0.0837  | 0.0324  | 0.0066  | 0.0111  | 0.0080  | -0.0049 | -0.0108 | -0.0094 |
| $\rho_\Delta(5)$ | 0.1110  | 0.0584  | 0.0833  | -0.0034 | -0.0064 | -0.0050 | -0.0057 | -0.0048 | 0.0022  | -0.0034 |

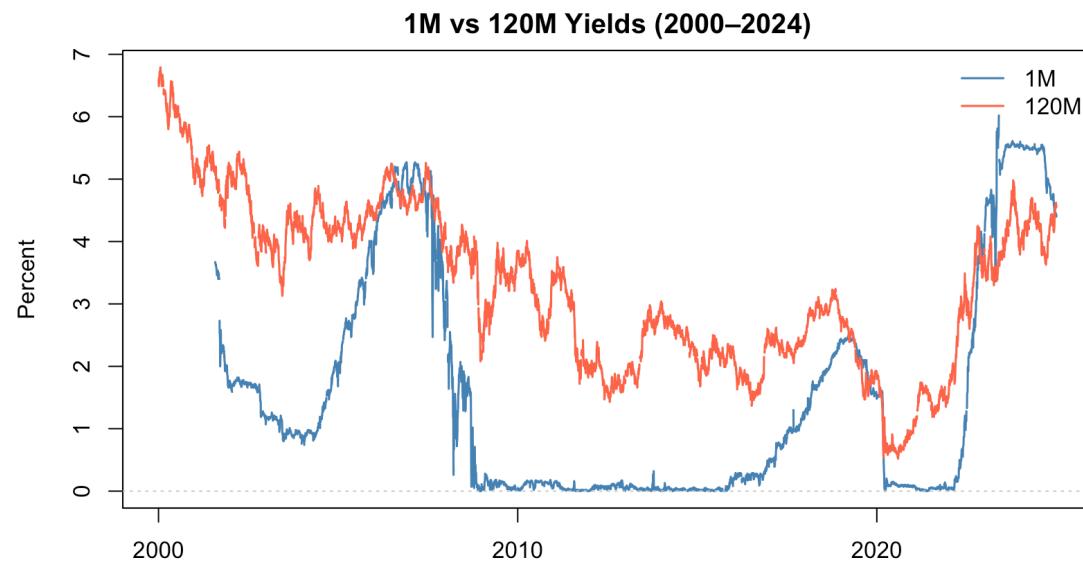


图 10.1: 1 个月与 120 个月收益率 (2000–2024 年, FRED 常数到期收益率; 纵轴为百分比)。可见长期平均水平更高、短端在危机期间贴近零下限并在 2022 年后快速上行。

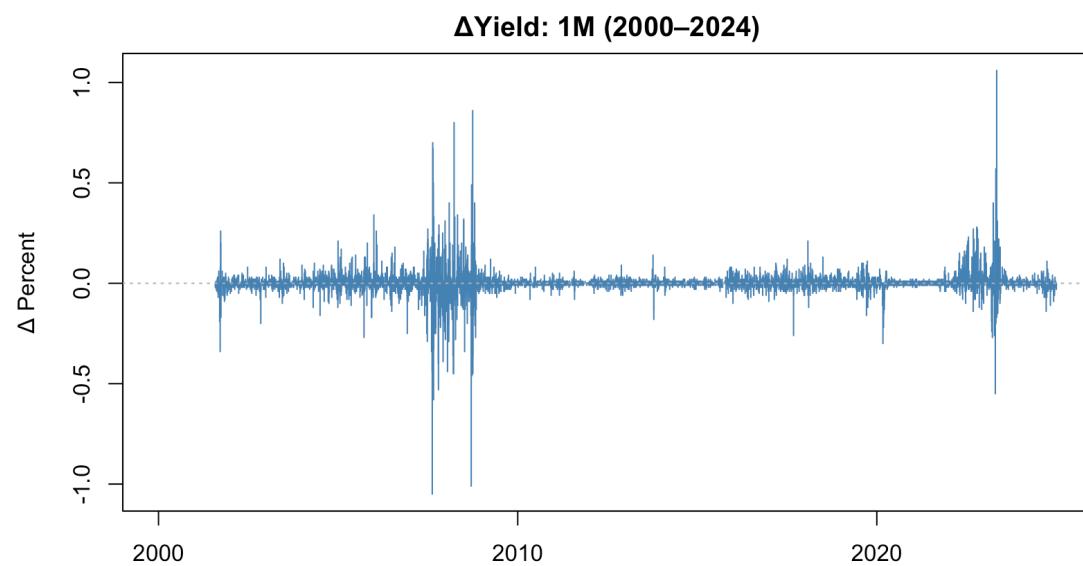


图 10.2: 1 个月收益率的日变化 (2000–2024 年, 单位: 百分点)。2008–2009 年与 2020 年附近波动显著放大, 随后总体回落, 反映短端政策与流动性冲击的主要时点。

## 10.4 无套利与随机贴现因子 (SDF)

令  $p_t^n$  为  $t$  时点购买、 $t+n$  兑付 1 美元的价格。一期收益  $R_{t+1} = p_{t+1}^{n-1}/p_t^n$ 。若存在 SDF (随机贴现因子)  $M_{t+1}$ , 则无套利要求

$$p_t^n = E_t(M_{t+1} p_{t+1}^{n-1}), \quad \text{递推得} \quad p_t^n = E_t\left(\prod_{j=1}^n M_{t+j}\right).$$

在风险中性测度  $\mathbb{Q}$  下, 常写  $M_{t+1} = e^{-r_t \frac{d\mathbb{Q}}{d\mathbb{P}}}$ , 从而

$$p_t^n = E_t^{\mathbb{Q}}\left[e^{-(r_t + \dots + r_{t+n-1})}\right], \quad y_t^n := -\frac{1}{n} \log p_t^n, \quad 1 + f_t^{n \rightarrow n+1} = \frac{p_t^n}{p_t^{n+1}}.$$

这一区分强调: 定价在  $\mathbb{Q}$  (或 SDF) 下进行, 而统计建模多在物理测度  $\mathbb{P}$  下进行, 二者通过“风险价格/期限溢价”相联。

### 10.4.1 Vasicek 模型 (离散时间仿射形式)

设随机贴现因子 (SDF) 与单因子状态变量满足

$$-\log M_{t+1} = \delta + z_t + \lambda \varepsilon_{t+1}, \quad z_{t+1} = \varphi z_t + (1 - \varphi)\theta + \sigma \varepsilon_{t+1}, \quad \varepsilon_{t+1} \stackrel{\text{i.i.d.}}{\sim} N(0, 1),$$

于是  $E(z_t) = \theta$ 。由无套利定价理论

$$p_t^0 = 1, \quad p_t^1 = E_t(M_{t+1}), \quad p_t^2 = E_t(M_{t+1} p_{t+1}^1), \quad \dots$$

若取  $\delta = \lambda^2/2$ , 利用正态矩母函数的性质可得

$$\log p_t^1 = -z_t \quad \Rightarrow \quad r_t := -\log p_t^1 = z_t,$$

即状态  $z_t$  恰为年化对数短期利率。对长期债券, 假设价格为仿射形式

$$-\log p_t^n = A_n + B_n z_t \quad \Longleftrightarrow \quad p_t^n = \exp(-A_n - B_n z_t),$$

其中系数  $(A_n, B_n)$  随期限  $n$  递推确定。将上式代回定价方程并配对  $z_t$  的常数项与系数项, 可得 (离散时间 Riccati 型) 递推方程:

$$\begin{aligned} A_{n+1} &= A_n + \delta + B_n(1 - \varphi)\theta - \frac{1}{2}(\lambda + B_n \sigma)^2, & A_1 &= 0, \quad B_1 = 1. \\ B_{n+1} &= 1 + \varphi B_n, \end{aligned}$$

(上式表示:  $B_n$  线性地累积均值回复系数  $\varphi$ ;  $A_n$  聚合了贴现常数项、均值回复项与由创新  $\varepsilon_{t+1}$  引致的“风险价格一方差”项。)

Vasicek 模型属于离散时间的一因子高斯—仿射期限结构模型: 债券对数价格对状态变量线性, 短期利率服从均值回复的 AR(1) 过程; 计算上通过简单递推即可得到任意期限的价格、收益率与远期利率。由于是高斯结构, 短期利率允许取负值, 且单因子形状受限——虽然可以产生随期限上行的平均收益率曲线, 但其长期端的“弯曲程度”(凹度/驼峰) 通常弱于实际数据; 在需要更丰富曲率与协同变动的应用中, 常用多因子或非高斯扩展以提升拟合能力。

### 10.4.2 Cox–Ingersoll–Ross (CIR) 模型

离散时间下，令状态变量与随机贴现因子满足

$$\begin{aligned} z_{t+1} &= \varphi z_t + (1 - \varphi) \theta + \sigma z_t^{1/2} \varepsilon_{t+1}, \\ -\log M_{t+1} &= \left(1 + \frac{\lambda^2}{2}\right) z_t + \lambda z_t^{1/2} \varepsilon_{t+1}, \end{aligned} \quad \varepsilon_{t+1} \stackrel{\text{i.i.d.}}{\sim} N(0, 1).$$

这是连续时间费勒平方根过程（见第 9.7.1 节）在离散时间中的对应设定：创新项的幅度与利率水平的平方根成正比，短端越高、波动越大，且更自然地保持非负性。

在离散时间 SDF 框架下，债券价格保持仿射形式：

$$-\log p_t^n = A_n + B_n z_t \iff p_t^n = \exp(-A_n - B_n z_t),$$

其中系数  $(A_n, B_n)$  满足 (Riccati 型) 递推关系

$$A_{n+1} = A_n + B_n(1 - \varphi) \theta, \quad B_{n+1} = 1 + \frac{\lambda^2}{2} + \varphi B_n - \frac{1}{2}(\lambda + B_n \sigma)^2, \quad A_1 = 0, \quad B_1 = 1.$$

这与连续时间 CIR 模型在风险中性测度下得到的 Riccati 常微分方程相呼应：在离散时间里由“一期滚动定价”的仿射性质给出差分递推。

CIR 模型的核心是“水平—波动联动”：短端较高时， $\sqrt{z_t}$  放大创新项，体现出更强波动性与更厚尾部；同时，均值回复特性使利率长期受  $\theta$  参数锚定，避免无约束漂移。与高斯模型 (Vasicek) 不同，平方根扩散过程更自然地保证利率非负性，对 ZLB (零利率下限) 前后也更为稳健。

尽管 CIR 模型引入了与水平相关的波动并保持非负，但单因子版本在实践中仍显简单：若将  $\varphi$  调整到能较好匹配短端自相关，平均收益率曲线往往比实际数据更“平”，即凸度不足、难以呈现清晰的驼峰；同时，模型预测不同期限收益率的自相关形状几乎一致，这与经验事实（长端自相关更高、短端更快均值回复）并不吻合。此外，离散时间设定中仍使用正态创新  $\varepsilon_{t+1}$ ，难以充分刻画尾部与跳跃，极端时期（如危机与非常规政策阶段）可能出现系统性失配。

### 10.4.3 多因子仿射期限结构

离散时间的仿射类期限结构可以从两条路径理解：一是从若干具体可解的特例出发，二是构建一套在运算上保持封闭、解析性强且适用范围更广的模型族。前一条路径中，CLM 与 Backus et al. (1998) 讨论了多种带高斯创新的具体情形；例如，下式的一因子“平方根短端 + 线性—平方根 SDF”就是离散时间的 CIR 思路：

$$\begin{aligned} r_{t+1} &= \alpha + \beta r_t + \gamma r_t^{1/2} \varepsilon_{t+1}, \\ M_{t+1} &= a + b r_t + c r_t^{1/2} \varepsilon_{t+1}, \end{aligned}$$

其中  $\varepsilon_{t+1} \stackrel{\text{i.i.d.}}{\sim} N(0, 1)$ 。由于 SDF 的定价约束存在结构性限制， $(a, b, c, \alpha, \beta, \gamma)$  不能任取；Dai & Singleton (2000) 系统刻画并检验了这类“过度识别限制”。与此同时，CLM 指出：这类“具体仿射规格”在现实中常见三种不足——(i) 当用  $K$  个因子（或解释  $n$  个期限、 $K \geq n$ ）时，模型隐含的债券协方差矩阵可能退化；(ii) 波动率的函数形式受限，难以反映厚尾/异方差；(iii) 长期债券的风险溢价往往被约束为固定符号，与数据的阶段性变化不完全一致。

为提升覆盖面与可用性, Gourieroux et al. (2006) 提出了一套多因子仿射模型族。令状态  $z_t \in \mathbb{R}^n$  为马尔可夫过程, 并假设其条件对数矩母函数关于当前状态呈仿射形式:

$$\log E(\exp(u' z_{t+1}) | z_t) = a(u)' z_t + b(u),$$

其中  $a(\cdot), b(\cdot)$  为给定函数。可将短端与若干“形状因子”(如斜率、曲率)并入状态(例如取  $z_t = (r_{t+1}, f_t)$ ), 并令 SDF 的对数在“当前—未来状态”上仿射:

$$m_{t+1} := \log M_{t+1} = \gamma_0 + \gamma'_1 z_t + \gamma'_2 z_{t+1}.$$

基本定价关系(式(10.3))对  $\gamma_0, \gamma_1, \gamma_2$  施加了可检验限制。该模型族把 Vasicek、CIR 及其多因子扩展等众多离散时间仿射模型统一纳入; 作者的目标是: 在尽量不改变既有定价工具的前提下, 提供一个既便于衍生品定价、又更贴近数据的统一框架。

**价格—收益率的仿射表示与递推。**在上述设定下, 剩余期限为  $n$  的债券价格与收益率可写成

$$\text{Price } P_t^n = \exp(c'_n z_t + d_n), \quad \text{Yield } y(t, t+n) = -\frac{c'_n z_t + d_n}{n},$$

其中  $n = 2, 3, \dots$  的系数  $(c_n, d_n)$  满足非线性差分递推关系

$$\begin{aligned} c_n &= a(c_{n-1} + \gamma_2) - a(\gamma_2) - e_1, \\ d_n &= d_{n-1} - b(\gamma_2) + b(c_{n-1} + \gamma_2) - e_1, \end{aligned}$$

并取  $e_1 = (1, 0)', c_1 = -e_1, d_1 = 0$ 。Vasicek 模型的特殊情况将该递推简化为线性形式, CIR 模型的特殊情况则会出现二次项; 在定价某些复杂现金流时, 还可能需要在此基础上做一次积分(或采用数值近似方法)。

**条件累积量的线性结构。**该模型族的一个关键结论是: 在给定  $z_t = z$  时, 各阶条件累积量均为  $z$  的线性函数:

$$\begin{aligned} E(z_{t+1} | z_t = z) &= a'(0)z + b'(0), \\ \text{Var}(z_{t+1} | z_t = z) &= a''(0)z + b''(0), \\ \kappa_3(z_{t+1} | z_t = z) &= a'''(0)z + b'''(0), \\ \kappa_4(z_{t+1} | z_t = z) &= a^{(4)}(0)z + b^{(4)}(0). \end{aligned}$$

这与广义线性模型(GLM)的思想相近: 条件分布由有限维状态控制, 且在计算与识别上保持简明。而与之对照, 诸如 GARCH 等常见离散时间条件方差模型往往让方差成为状态变量与滞后项的二次函数, 甚至依赖无穷多阶过去值, 这类模型在本框架中被自然排除。

Vasicek / CIR 模型的优点在于计算简洁、直观可解释, 但在曲线形状的灵活度、期限联动与尾部刻画上往往不足; 广义多因子仿射模型族在保持解析友好与运算封闭性的同时, 放宽了对创新分布与状态维度的限制, 允许更丰富的风险价格结构, 因此能够在不大幅改造既有定价流程的前提下, 统一处理更复杂的现金流与衍生品定价问题。

以下 R 代码用于复现实证文献中的收益率统计事实。具体做法是: 从 FRED 抓取 2000–2024 年的美国国债常数到期收益率(1、3、6、12、24、36、60、120、240、360 个月), 在各自可用区间内分别计算日频收益率的四个描述统计量——均值  $m$ 、标准差  $s$  (单位: %)、偏度  $\kappa_3$  与超额峰度  $\kappa_4$ ——并输出为“表 13.1”。随后对差分收益率  $\Delta y_t$  计算 1–5 阶自相关, 形成“表 13.2”, 以检验极短端的短期可预测性与长期端近单位根的特征。同时, 代码将“1 个月与 120 个月收益率”的时序对比图和“1 个月收益率日变化”的图形分别保存为 PNG 文件, 以直观展示零利率下限阶段的短端压平、加息期的陡升以及波动的阶段性聚集。

这些结果与经典样本 (1952–1991) 相比, 既保留了“平均曲线随期限上行”的稳健事实, 也清晰呈现出现代样本 (ZLB、疫情冲击与后续加息) 下的结构性差异。

```

1 # =====
2 # 统计性质复现: 2000-2024 日频收益率的描述统计与差分ACF
3 # 依赖: quantmod, zoo, moments
4 # =====
5
6 # 设置工作目录
7 if (requireNamespace("rstudioapi", quietly = TRUE) &&
8 rstudioapi::isAvailable()) {
9 setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
10 }
11 suppressPackageStartupMessages({
12 library(quantmod)
13 library(zoo)
14 library(moments) # skewness(), kurtosis()
15 })
16
17 # 1) 设定期限与FRED代码 (单位: 月 -> 年)
18 mats_month <- c(1,3,6,12,24,36,60,120,240,360)
19 fred_codes <- c("DGS1MO", "DGS3MO", "DGS6MO", "DGS1", "DGS2", "DGS3",
20 "DGS5", "DGS10", "DGS20", "DGS30")
21
22 # 2) 下载 2000-01-01 至 2024-12-31 的数据; 注意 1MO 与 30Y 存在缺段
23 from <- as.Date("2000-01-01"); to <- as.Date("2024-12-31")
24 getSymbols(fred_codes, src = "FRED", from = from, to = to, auto.assign =
25 TRUE)
26
27 # 合并为 xts, 列名为到期 (月)
28 Y <- do.call(merge, mget(fred_codes))
29 colnames(Y) <- paste0(mats_month, "M")
30
31 # 3) 转换为数值矩阵并去掉全NA的列; 转为小数 (而非%)
32 Ynum <- Y / 100
33 # 这里只在各列各自可用的时间区间内计算统计量, 不强求完全交集
34 # 函数: 对单列xts计算统计量
35 stats_one <- function(x) {
36 x <- na.omit(as.numeric(x))
37 if (length(x) < 10) return(c(m=NA, s=NA, k3=NA, k4=NA))
38 c(m = mean(x, na.rm=TRUE)*100, # 转回百分数便于展示
39 s = sd(x, na.rm=TRUE)*100,
40 k3= skewness(x, na.rm=TRUE),
41 k4= kurtosis(x, na.rm=TRUE) - 3) # 超峰度
42 }
43
44 # 4) 生成表13.1: 各期限收益率的 m, s, 3, 4
45 tab1 <- t(apply(Ynum, 2, stats_one))
46 tab1 <- as.data.frame(tab1)
47 tab1$Maturity <- rownames(tab1)
48 tab1 <- tab1[, c("Maturity", "m", "s", "k3", "k4")]
rownames(tab1) <- NULL

```

```

49
50 # 5) 生成表13.2: 差分收益率的自相关 (1-5阶)
51 acf_one <- function(x, maxlag=5) {
52 x <- diff(na.omit(as.numeric(x)))
53 if (length(x) < (maxlag+10)) return(rep(NA, maxlag))
54 acf(x, plot=FALSE, na.action=na.pass)$acf[2:(maxlag+1)]
55 }
56 acf_mat <- t(apply(Ynum, 2, acf_one, maxlag=5))
57 colnames(acf_mat) <- paste0("rho_D(", 1:5, ")")
58 tab2 <- as.data.frame(acf_mat)
59 tab2$Maturity <- rownames(acf_mat)
60 tab2 <- tab2[, c("Maturity", colnames(acf_mat))]
61 rownames(tab2) <- NULL
62
63 # 6) 打印结果 (四舍五入以便排版)
64 cat("==== Table 13.1 日频收益率的描述统计 (2000-2024) ====\n")
65 print(within(tab1, { m=round(m,4); s=round(s,4); k3=round(k3,4); k4=round(k4,
66 ,4)}), row.names=FALSE)
67
68 cat("\n==== Table 13.2 差分收益率的自相关 (1-5阶, 2000-2024) ====\n")
69 print(within(tab2, {
70 `rho_D(1)`=round(`rho_D(1)`,4); `rho_D(2)`=round(`rho_D(2)`,4);
71 `rho_D(3)`=round(`rho_D(3)`,4); `rho_D(4)`=round(`rho_D(4)`,4);
72 `rho_D(5)`=round(`rho_D(5)`,4)
73 }), row.names=FALSE)
74
75 # 7) 可选: 画两条典型期限 (1M 与 120M) 以对照不同时期的水平与趋势
76
77 # 使用分析区间 (2000-01-01 至 2024-12-31)
78 Ywin <- window(Y, start = from, end = to)
79
80 # 1个月期与120个月期共用纵轴范围
81 ylim12 <- range(Ywin[, c("1M", "120M")], na.rm = TRUE)
82
83 # 图1: 1个月期 vs 120个月期收益率 (保存为 PNG)
84 png("fig_yields_1M_120M_2000_2024.png", width = 1600, height = 900, res =
85 200)
86 par(mar = c(4, 4, 2, 1))
87 plot(index(Ywin), coredata(Ywin[, "1M"]),
88 type = "l", col = "steelblue", lwd = 1.5,
89 ylim = ylim12, xlab = "", ylab = "Percent",
90 main = "1M vs 120M Yields (2000 - 2024)")
91 lines(index(Ywin), coredata(Ywin[, "120M"]), col = "tomato", lwd = 1.5)
92 abline(h = 0, col = "grey80", lty = 3)
93 legend("topright", c("1M", "120M"), lty = 1, col = c("steelblue", "tomato"),
94 lwd = 1.5, bty = "n")
95 dev.off()
96
97 # 图2: 1个月期收益率的日度增量 Δy (保存为 PNG)
98 Dy1 <- diff(Ywin[, "1M"])
99 png("fig_dyield_1M_2000_2024.png", width = 1600, height = 900, res = 200)

```

```

98 par(mar = c(4, 4, 2, 1))
99 plot(index(Dy1), coredata(Dy1),
100 type = "h", col = "steelblue",
101 xlab = "", ylab = " Δ Percent",
102 main = " Δ Yield: 1M (2000 - 2024)")
103 abline(h = 0, col = "grey70", lty = 3)
104 dev.off()
105
106

```

图 10.3 与图 10.4 展示了 3 个月（日频）与 10 年（月频）收益率在不同利率水平下四个{条件累积量}（条件均值、条件方差、条件偏度、条件超峰度）的局部线性核估计。可以看到：条件均值大体呈近似线性，斜率接近 1，表明收益率具有较强的持续性；条件方差在不同水平区间显著非线性，短端尤为明显，说明波动性随水平而变。更高阶的条件偏度与条件超峰度同样随水平呈系统性变化，说明尾部形态与偏斜在不同利率环境下并不恒定。相较之下，10 年期曲线的非线性幅度整体更温和，符合“长端更平稳、短端更敏感”的经验事实。

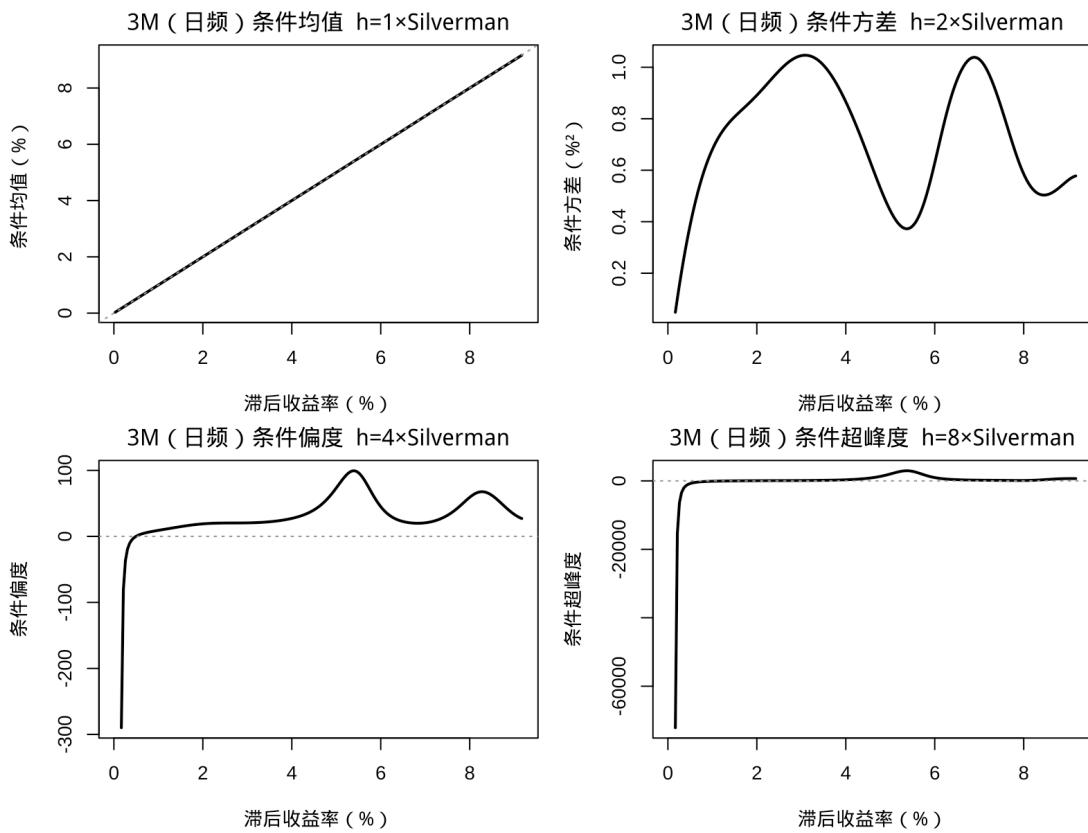


图 10.3: 3M (日频) 收益率的四个条件累积量的非参数估计 (局部线性核; 带宽分别为 1、2、4、8 倍 Silverman 经验法则)。上左: 条件均值近似线性、斜率接近 1; 上右: 条件方差随水平显著非线性; 下左与下右: 条件偏度与条件超峰度在不同利率区间呈现明显的非线性变化。

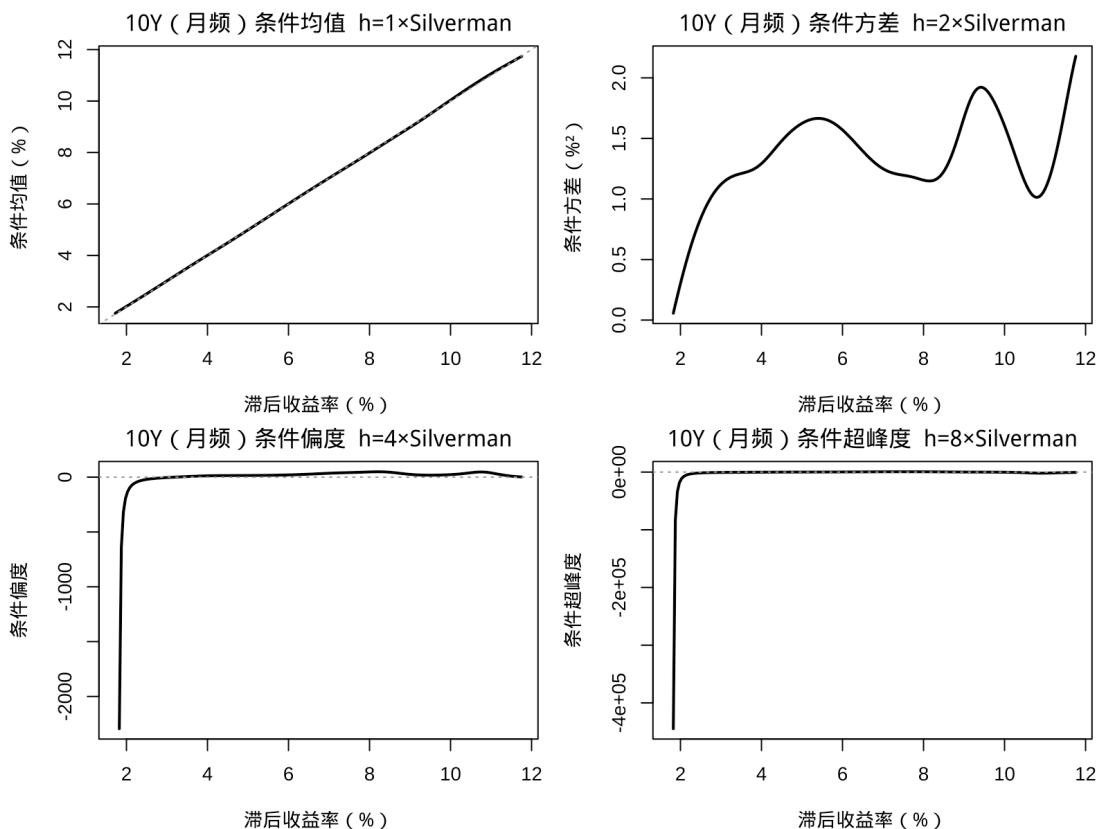


图 10.4: 10Y (月频) 收益率的四个条件累积量的非参数估计 (局部线性核; 带宽分别为 1、2、4、8 倍 Silverman 经验法则)。条件均值基本线性; 条件方差、条件偏度与条件峰度仍存在非线性, 但幅度相对 3M 更温和。

## 10.5 本章小结

本章主要涵盖两个核心内容：一是“如何量化利率期限结构”，二是“怎样稳健地估计曲线并加以应用”。我们先从零息债券出发，梳理贴现函数、收益率曲线和远期利率之间的定义与换算关系，并用统一口径的真实数据计算了“曲线三件套” $(y, d, f)$ 的示例。随后给出两条实用的曲线估计路径：一种视贴现函数为平滑曲线，在“线性定价”框架下使用样条/基函数进行平滑处理并施加（软）无套利约束；另一种使用少量参数控制整体形状（Nelson–Siegel/Svensson 模型），并配合 Fama–Bliss 的分段远期引导。为将直觉落实到数据上，我们使用 2000–2024 年的美国国债样本，展示了“平均曲线随期限上升、短端波动更大、差分相关性集中在最短端”等常见现象。最后，在“无套利 + 随机贴现因子 (SDF)”框架下，介绍了从单因子模型 (Vasicek, CIR) 到多因子仿射模型的基本思路，说明如何将价格—收益率关系表示为仿射形式，并用简单的递推关系连接“预期”和“期限溢价”。

整体而言，本章希望提供一套可操作的流程：明确口径，选好方法（样条/参数化/引导），在约束与平滑之间寻找平衡，然后将曲线用于定价、久期、凸性、预测与风险管理。读者可将本章的示例代码作为起点，替换为自身的现金流与价格数据，按需添加无套利约束或更灵活的因子设定，在真实业务中落地使用。

## 10.6 习题

1. (概念与推导) 结合第 10.1 节，证明下列结论：
  - (a) 若采用连续复利口径，贴现函数与收益率曲线满足  $d(\tau) = \exp\{-\tau y(\tau)\}$ ，瞬时远期利率  $f(\tau) = -\frac{d}{d\tau} \ln d(\tau)$ ，并由此推出  $y(\tau) = \tau^{-1} \int_0^\tau f(s) ds$ 。
  - (b) 设一组确定现金流  $\{(\tau_j, b_j)\}_{j=1}^m$ ，在连续复利口径下给出价格的两种等价表示： $p = \sum_{j=1}^m b_j d(\tau_j) = \sum_{j=1}^m b_j \exp\{-\tau_j y(\tau_j)\}$ 。
  - (c) 在利率非负时证明  $d(\tau)$  关于  $\tau$  单调递减；当存在负利率时分析  $d(\tau) > 1$  的可能性与经济含义。
2. (离散—连续口径统一) 给定年化简单利率口径的即期收益率点集  $\{(\tau_i, \tilde{y}_i)\}_{i=1}^N$ ，将其转换为连续复利  $y_i^{(c)} = \ln(1 + \tau_i \tilde{y}_i)$ ，并据此计算  $\{d(\tau_i), f(\tau_i)\}$ 。请讨论口径转换对短端 ( $\tau \leq 1$ ) 的数值影响。
3. (Bootstrapping) 设有一组息票债券（半年付息），价格向量  $p$ 、现金流矩阵  $B$  与支付时点集合  $\{\tau_k\}$ 。
  - (a) 采用“自短至长”逐点抽取零息贴现率的 Bootstrapping 程序，给出算法步骤并实现（可用 R 或 Python）。
  - (b) 在出现病态问题或解不唯一时，给出三种改进措施（如曲线平滑、正则化、插值策略或剔除异常债券）并比较其利弊。
4. (基函数/样条法，见第 10.2.1 节) 将贴现函数写作  $d(t) \approx \sum_{\ell=1}^L \theta_\ell g_\ell(t)$ ：
  - (a) 选择三次 B 样条为基，写出最小二乘 + 二阶差分惩罚的目标函数，并解释惩罚项近似  $\int (d''(t))^2 dt$  的含义。
  - (b) 说明如何用“高权重伪观测”实现  $d(0) = 1$  的强约束；给出保证单调性  $d'(t) \leq 0$  的一种实现思路（如差分不增约束或再参数化）。

(c) 用你选取的国债样本复现一条  $\hat{d}(t)$ , 并据此给出  $\hat{y}(t)$ 、 $\hat{f}(t)$ 。比较不同  $\lambda$  (平滑强度) 对短端与长端的影响。

5. (参数法, 见第 10.2.2 节) Nelson–Siegel 与 Svensson 模型:

- (a) 从瞬时远期利率出发, 推导 NS 与 NSS 模型的贴现函数  $d_\theta(t) = \exp\{-\int_0^t f(s) ds\}$  的闭式解。
- (b) 采用“价格准则”拟合 NS 与 NSS (R 代码可沿用文中示例), 报告参数、定价误差, 以及与样条曲线的对比。讨论何时 NSS 明显优于 NS。
- (c) 设计一个“短端权重更高”的加权最小二乘法, 并比较其对  $\tau \leq 2$  年期限段拟合的改善与对长端外推的影响。

6. (Fama–Bliss, 引导法, 见第 10.2.3 节)

- (a) 在相邻区间  $(\tau_{i-1}, \tau_i]$  假设远期利率为常数  $f_i$ , 写出基于第  $i$  只债券的单变量方程并阐述序贯求解步骤; 讨论无法找到根时的处置方式 (如放宽搜索区间或剔除异常债券)。
- (b) 使用同一交易日的样本实现 FB 自举法, 绘制分段常数远期利率曲线  $f(t)$ 、贴现因子曲线  $d(t)$  与收益率曲线  $y(t)$ 。将 FB 方法的长端外推能力和短端稳定性与 NS/NSS 模型及样条方法进行比较。

7. (非参数检验, 见第 10.3.2 节) 对 3M (日频) 与 10Y (月频) 收益率:

- (a) 采用局部线性核、Silverman 带宽的 1、2、4、8 倍, 分别估计条件均值、条件方差、条件偏度与条件超峰度随滞后收益率变化的函数关系并作图。
- (b) 讨论“条件均值近似线性且斜率趋近于 1”、“条件方差与高阶矩存在非线性特征”的经济含义; 指出这对短端/长端利率建模的启示 (例如水平依赖的波动性、跳跃行为或厚尾分布)。

8. (SDF——Vasicek, 参见第 10.4.1 节) 已知

$$-\log M_{t+1} = \delta + z_t + \lambda \varepsilon_{t+1}, \quad z_{t+1} = \varphi z_t + (1 - \varphi) \theta + \sigma \varepsilon_{t+1},$$

取  $\delta = \lambda^2/2$  并假设  $-\log p_t^n = A_n + B_n z_t$ 。请推导 Riccati 递推

$$A_{n+1} = A_n + \delta + B_n(1 - \varphi)\theta - \frac{1}{2}(\lambda + B_n\sigma)^2, \quad B_{n+1} = 1 + \varphi B_n,$$

并说明  $r_t = z_t$  的由来。用一组收益率序列 (国家/期限自行选择) 校准参数, 并报告样本内拟合与样本外预测性能。

9. (SDF——CIR, 参见第 10.4.2 节) 在离散时间设定

$$z_{t+1} = \varphi z_t + (1 - \varphi)\theta + \sigma z_t^{1/2} \varepsilon_{t+1}, \quad -\log M_{t+1} = \left(1 + \frac{\lambda^2}{2}\right) z_t + \lambda z_t^{1/2} \varepsilon_{t+1},$$

证明仿射价格与递推

$$A_{n+1} = A_n + B_n(1 - \varphi)\theta, \quad B_{n+1} = 1 + \frac{\lambda^2}{2} + \varphi B_n - \frac{1}{2}(\lambda + B_n\sigma)^2.$$

讨论单因子 CIR 模型的优缺点 (非负性、水平—波动联动 vs. 曲率不足/相关结构受限), 并尝试加入第二因子或厚尾创新项, 比较拟合效果。

10. (广义多因子仿射族, 见第 10.4.3 节) 令

$$\log E\left(e^{u'z_{t+1}} \mid z_t\right) = a(u)'z_t + b(u), \quad m_{t+1} = \gamma_0 + \gamma_1'z_t + \gamma_2'z_{t+1}.$$

- (a) 证明在上述设定下,  $-\log p_t^n = A_n + B_n'z_t$  并给出  $(A_n, B_n)$  的递推形式 (可引用文中  $c_n, d_n$  的差分关系)。
- (b) 说明“各阶条件累积量对状态线性”的意义, 并与 GARCH 的条件方差结构作对比: 何以说该族在计算与识别上更简单?
- (c) 设计一套基于状态空间模型—卡尔曼滤波的 ML/EM 估计算法流程 (观测方程、状态方程、初值与识别约束), 并在一组 5–10 个期限的利率上演示。

11. (稳健性与外推) 选择你认为最合适两种的曲线拟合方法 (例如“样条曲线 + 单调约束”与 NSS 模型), 对同一交易日横截面数据进行拟合, 并完成:

- (a) 短端外推至  $\tau = 1$  月与长端外推至  $\tau = 50$  年, 分析两种方法在外推端的数值稳定性与经济合理性。
- (b) 通过置换/加噪 (对个券价格加上  $\pm 5$  bp) 做敏感性实验, 报告对  $\hat{d}(\tau), \hat{f}(\tau)$  的影响。

12. (综合实践) 任选一个国家的国债市场 (或利率互换曲线):

- (a) 选取 1–2 年的日度横截面数据, 滚动拟合样条与 NS/NSS 模型, 绘制“时间–期限”的热力图 (heatmap), 以展示  $y(\tau)$  与  $f(\tau)$  的时变结构。
- (b) 使用你认为合适的 SDF—仿射模型 (单因子或多因子) 做样本内及样本外预测, 比较预测误差与参数稳定性; 对 ZLB 时期单独分析。
- (c) 撰写简短报告, 总结方法选择、校准细节、主要发现与局限, 并提出进一步改进的方向 (如影子利率、宏观因子嵌入、厚尾/跳跃等)。

# 11 风险管理与尾部风险估计

金融市场的风险并不总是“温和”的。长期以来，实务界与监管部门常以标准差/方差配合正态分布来度量不确定性，原因在于计算简洁、沟通成本低 (Jorion 1997)。但自 Mandelbrot (1963) 以来的大量研究表明，收益分布往往呈现尖峰厚尾、波动具有时变聚集性，极端事件（所谓“ $6\sigma$ ”）出现的频率远高于正态假设下的理论预期。2008 年危机后的监管评估也强调：仅依赖线性—高斯框架，可能系统性低估尾部风险，从而误导资本计提与压力测试安排 (Turner 2009, Basel Committee on Banking Supervision 2012)。

为什么要专门研究“尾部”？至少有三方面动因。第一，许多关键决策依赖高置信分位数（如 99%/99.9%）的损失度量，而正态缩放 ( $\sqrt{T}$ ) 在这些分位数常常失真；第二，历史样本有限，真正令人担忧的情形往往位于“样本之外”的更深尾部，需要在可控假设下进行外推；第三，危机时多资产常常“同跌”，尾部依赖与常规相关性（线性/均值一方差框架）并不等价，这直接关系到分散化是否有效以及系统性风险的识别与定价。

中国市场的经验同样凸显尾部问题的重要性：2013 年银行间“钱荒”在极短时间内推高短端利率，暴露出流动性尾部的传导链条；2015 年 A 股异常波动与 2016 年初熔断事件，提示价格限制与交易微观结构在极端条件下可能放大波动与相关性；近年来房地产及其上下游的信用事件、城投平台再融资压力，以及部分衍生品/结构化产品在极端行情下的穿仓与追加保证金风险，都表明：尾部不仅体现在价格维度，也会在流动性、信用与机制设计层面显性化。这些案例并非为了追责，而在于强调：忽视尾部，往往叠加模型风险与治理风险。

因此，围绕尾部的工具箱必须超出均值一方差分析的范畴：一方面，以分位数/在险价值 (Value-at-Risk, VaR) 为“通用语言”，结合稳定分布与非参数分位数估计，获得对中高分位数的稳健估计；另一方面，引入极值理论 (Extreme Value Theory, 简称 EVT) 与正则变化理论，通过尾指数识别并对“样本之外”的极端分位数进行外推；同时，在时间维度采用 GARCH/CaViaR 等模型刻画风险的时变性，在多维维度利用 Copula/CoVaR 描述尾部依赖与系统性贡献；最后，将 VaR 的可沟通性与期望损失 (Expected Shortfall, ES) 的理论一致性结合起来，形成“可检验、可披露、可操作”的尾部度量闭环。

本章的结构按照“诊断—建模—外推—评估—沟通”的实践路径展开：首先，从分位数视角重述 VaR 的优势与局限，对比正态分布与稳定分布下的时间缩放差异，明确正态缩放失效的边界；随后，系统介绍 EVT 与半参数尾部模型的核心结果与估计方法（包括 Hill 估计、阈值选择与敏感性分析），并讨论如何在有限样本下稳健地外推深尾分位数；接着，给出条件 VaR/ES 的建模思路与回测检验 (backtesting) 框架，讨论不同置信水平、滚动窗口与样本长度对检验结果的影响；在多资产情境中，重点阐述尾部依赖的识别与度量（如 Copula、CoVaR、系统性风险分摊），以及其对分散化与资本配置的意义；最后，结合监管与实务需求，给出度量选型与披露建议：在“可沟通性—一致性—可检验性—可实施性”之间取得平衡，明确模型假设与口径，配套回测检验与敏感性分析，并在压力测试与资本计提场景中给出可操作的落地方案。

## 11.1 在险价值 (VaR)

**定义 11.1 (分位数 (Quantile) ):** 设随机变量  $X$  的分布函数为  $F(x) = \Pr(X \leq x)$ 。  
 $X$  的  $\alpha$  分位数定义为

$$q_\alpha = \inf\{q : \Pr(X \leq q) \geq \alpha\}.$$

若  $F$  在  $q_\alpha$  附近严格单调, 则有  $q_\alpha = F^{-1}(\alpha)$ 。另外, 对任意  $a > 0$  与  $b \in \mathbb{R}$ , 分位数满足仿射变换性质

$$q_\alpha(aX + b) = a q_\alpha(X) + b,$$

且当  $X$  连续时, 左右尾之间满足

$$q_{1-\alpha}(X) = -q_\alpha(-X).$$

因而, 无论我们以“收益”还是“损失”为建模对象, 只要明确号向, 分位数的定义与换算都是一致的。

当假定

$$X \sim N(\mu, \sigma^2),$$

则

$$q_\alpha = \mu + \sigma z_\alpha,$$

其中  $z_\alpha$  为标准正态分布的  $\alpha$  分位点。以  $\alpha = 1\%$  为例,  $z_\alpha = -2.33$ 。在样本  $X_1, \dots, X_n$  下, 可用样本均值与方差估计

$$\hat{q}_\alpha = \bar{X} + s z_\alpha,$$

这里  $\bar{X}, s^2$  分别是样本均值与样本方差。正态且 i.i.d. 的设定带来时间聚合上的便利: 若“日收益”

$$X_D \sim N(\mu_D, \sigma_D^2),$$

则“月收益”作为  $T$  个日收益之和, 有

$$X_M \sim N(\mu_M, \sigma_M^2), \quad \mu_M = T\mu_D, \quad \sigma_M^2 = T\sigma_D^2,$$

从而

$$q_\alpha(M) = \mu_M + \sigma_M z_\alpha = T\mu_D + T^{1/2}\sigma_D z_\alpha.$$

当  $\mu_D$  相对  $\sigma_D$  很小 (常见于高频收益), 得到熟悉的“平方根法则”: 月度 VaR 约为日度 VaR 的  $\sqrt{T}$  倍。这一近似使得我们能以日度参数快速推导任意持有期的 VaR, 并基于正态近似给出标准误与区间估计。

然而, 真实金融数据往往不是正态且独立同分布: 收益序列具有尖峰厚尾与时变相关, 正态尾部“过薄”, 会把“大亏损”的概率估得过小; 序列依赖也破坏了简单的  $\sqrt{T}$  缩放有效性。因此, 正态—平方根法则在实务中常成为低估尾部风险的来源。

因为金融收益常呈厚尾且持有期风险缩放常偏离  $\sqrt{T}$  规律, 稳定分布以尾指数  $\theta$  同时刻画极端尾概率与时间尺度的幂律缩放 ( $\text{VaR}_\alpha(T) = T^{1/\theta} \text{VaR}_\alpha(1)$ ), 即便二阶矩不存在也能给出分位数, 因此比正态更贴合 VaR 的尾部风险度量与期限换算。

**定义 11.2 (稳定分布):** 称实随机变量  $X$  服从稳定分布, 若其特征函数满足

$$\log E(e^{iuX}) = i\mu u - |\gamma u|^\theta \left(1 + i\beta \operatorname{sign}(u) w(u, \theta)\right), \quad u \in,$$

其中参数取值为

$$\theta \in (0, 2], \quad \beta \in [-1, 1], \quad \gamma > 0, \quad \mu \in,$$

并定义

$$w(u, \theta) = \begin{cases} \tan(\frac{\pi\theta}{2}), & \theta \neq 1, \\ -\frac{2}{\pi} \log |u|, & \theta = 1. \end{cases}$$

此时  $\theta$  为**特征指数/尾指数** (尾越厚则  $\theta$  越小)、 $\beta$  为**偏度** ( $\beta = 0$  为对称)、 $\gamma$  为**尺度**、 $\mu$  为**位置**。

**例 11.1 (常见特例):** • **正态分布:** 当  $\theta = 2$  时,  $w \equiv 0$ , 有  $\log E(e^{iuX}) = i\mu u - \gamma^2 u^2$ , 即  $X \sim N(\mu, 2\gamma^2)$ 。

- **柯西分布:** 当  $\theta = 1$  时,  $w(u, 1) = -(2/\pi) \log |u|$ , 得到厚尾的 *Cauchy* ( $\beta$  控制偏斜)。
- **矩存在性:**  $E|X|^p < \infty$  当且仅当  $p < \theta$ ; 因此除  $\theta = 2$  外方差通常不存在, 仅当  $\theta > 1$  时才有有限均值。

**定义 11.3 (稳定性/可加性与持有期缩放):** 若  $X_1, \dots, X_T$  独立同分布且各自满足上述稳定分布 (相同的  $\theta, \beta$ ), 则其和  $S_T = \sum_{t=1}^T X_t$  仍为稳定分布, 且

$$S_T \stackrel{d}{=} T^{1/\theta} X_1 + T \mu_0,$$

其中  $\mu_0$  为适当的中心化常数 (对称情形可取  $\mu_0 = \mu$ )。据此 (忽略平移项或在零均值下), 分位数与 VaR 呈**持有期幂律缩放**:

$$\text{VaR}_\alpha(T) = T^{1/\theta} \text{VaR}_\alpha(1).$$

当  $\theta = 2$  时退化为熟悉的  $\sqrt{T}$  尺度; 当  $\theta < 2$  时尾部更厚、VaR 随  $T$  的增长更快。

以下是值得注意的几个事项:

- **方差并非风险唯一刻画:** 对于  $\theta < 2$  的稳定分布, 方差不存在, 但分位数/尾部概率仍有明确定义, 适合采用 VaR/ES 等指标进行量化。
- **时间聚合不服从平方根法则:** 一般当  $\theta \neq 2$  时, VaR 随  $T$  呈  $T^{1/\theta}$  变化, 这解释了厚尾资产“持有期风险膨胀”现象往往快于  $\sqrt{T}$ 。
- **估计难点:** 稳定分布参数 (特别是  $\theta, \beta$ ) 的极大似然估计容易数值不稳定; 实务中常结合非参数分位数法 (中等尾部) 与 EVT (极端尾部) 来进行校准。

针对稳定分布的研究表明, 正态分布在尾部往往过于“薄”, 会低估极端损失出现的概率 (Mandelbrot 1963, Fama 1965c)。由于稳定分布族在独立同分布下具有可加性, 收益的持有期扩展并不遵循平方根法则, 而是  $\text{VaR}_\alpha(T) = T^{1/\theta} \text{VaR}_\alpha(1)$ , 这说明时间尺度上的 VaR 取

决于尾部厚度，而不仅仅是方差。不过，稳定分布在参数估计上较为困难 (McCulloch 1986)，因此在实证中常用非参数分位数来推断中等水平的尾部风险。例如，样本分位数  $\hat{q}_\alpha$  在温和条件下满足

$$\sqrt{T}(\hat{q}_\alpha - q_\alpha) \xrightarrow{d} N\left(0, \frac{\alpha(1-\alpha)}{[f(q_\alpha)]^2}\right),$$

从而可构造近似的置信区间 (Koenker 2005)。然而，当  $\alpha < 1/T$  或  $\alpha > 1 - 1/T$  时，非参数方法无法外推“样本之外”的极端分位数。为此，需要引入极值理论 (EVT) 与半参数尾部模型，对尾部作结构化建模。

尽管有上述补充方法，VaR 仍是风险管理的核心工具。定义上，若损失随机变量为  $X$ ，则

$$\text{VaR}_\alpha = q_\alpha(X), \quad \Pr(X \leq \text{VaR}_\alpha) = \alpha,$$

例如在正态假设下有

$$\text{VaR}_\alpha = \mu + \sigma z_\alpha,$$

其中  $z_\alpha$  为标准正态分布的  $\alpha$  分位点。VaR 以分位数为核心，既计算简洁又便于沟通，且已被纳入《巴塞尔协议》等监管框架 (Basel Committee on Banking Supervision 2012)。因此，虽然 VaR 存在一致性不足和尾部低估的问题，但在与极值理论 (EVT) 和期望损失 (Expected Shortfall, 简称 ES) 等工具结合后，它依然是现代风险管理体系不可或缺的基石。

## 11.2 极值理论 (EVT)

从风险管理视角，我们关心的是样本极端值在样本量增大时的“极限行为”。令样本极大值

$$M_T = \max_{1 \leq t \leq T} \{X_t\}$$

当  $T$  增大时， $M_T$  会如何变化？总体最大值对应总体分布的  $\alpha = 1$  分位数；在样本层面， $M_T$  可被视作“ $1 - 1/T$  分位数”的经验对应物。对称问题——样本极小值——可通过考虑  $\max(-X_t)$  处理，所以下文集中讨论极大值。若  $X_t$  为向量，还可逐分量定义坐标极大值，并进一步追问：各分量的极端事件之间是否存在系统性的同步关系？

这一路线可追溯至 Fisher & Tippett (1928) 的先驱性工作；Gnedenko (1943) 给出了严整的极限定理，系统综述参见 Embrechts et al. (1997)。

**定理 11.1 (i.i.d. 情形的极值极限定理):** 设  $X_t$  独立同分布。在温和的正则性条件下，存在常数序列  $a_T > 0, b_T \in \mathbb{R}$ ，使得

$$a_T(M_T - b_T) \xrightarrow{d} \mathcal{E}. \tag{11.1}$$

其中极限随机变量  $\mathcal{E}$  的分布属于三种类型之一，统一表示为

$$G_\gamma(x) = \Pr(\mathcal{E} \leq x) = \exp(-(1 + \gamma x)^{-1/\gamma}), \quad 1 + \gamma x > 0. \tag{11.2}$$

详见：Fisher & Tippett (1928), Gnedenko (1943), Embrechts et al. (1997)。

注：形状参数（极值指数） $\gamma$  决定广义极值 (GEV) 分布的三种情形： $\gamma > 0$  (Fréchet, 重尾)、 $\gamma = 0$  (Gumbel, 指数尾) 与  $\gamma < 0$  (Weibull, 有限端点)。此外，极小值可以通过对  $-X$  应用同样结果得到 ( $\min_{1 \leq t \leq T} X_t = -\max_{1 \leq t \leq T} (-X_t)$ )。一般而言，样本极大值与极

小值并不独立，仅在特殊分布或构造下才可能出现独立性。

**例 11.2 (均匀分布):** 若  $X_t \sim \text{Unif}[0, 1]$  且相互独立同分布，则  $M_T \rightarrow 1$ ，且

$$\Pr(M_T \leq x) = \prod_{t=1}^T \Pr(X_t \leq x) = [F(x)]^T = x^T.$$

取  $x_T = 1 - \frac{x}{T}$  ( $x > 0$ )，有

$$\Pr(T(M_T - 1) \leq -x) = \Pr(M_T \leq x_T) = \left(1 - \frac{x}{T}\right)^T \rightarrow \exp(-x).$$

即 (11.1) 成立， $a_T = T$ ,  $b_T = 1$ ，极限分布为指数分布 ( $\gamma = -1$ )。

**例 11.3 (正态分布):** 若  $X_t \sim N(0, 1)$ ，则  $M_T \xrightarrow{P} \infty$ ，且在  $\gamma = 0$  下 (11.1) 成立：

$$a_T = \sqrt{2 \ln T}, \quad b_T = \sqrt{2 \ln T} - \frac{\ln(4\pi) + \ln \ln T}{2\sqrt{2 \ln T}},$$

极限 CDF:  $G(x) = \exp(-\exp(-x))$ .

其密度函数为  $g(x) = e^{-e^{-x}} e^{-x}$ 。

**例 11.4 (柯西分布):** 若  $X_t$  为标准柯西分布，密度  $f_X(x) = 1/\{\pi(1+x^2)\}$ ，则  $M_T \xrightarrow{P} \infty$ ，且 (11.1) 在  $\gamma = 1$  的情形下成立：

$$a_T = \pi/T, \quad b_T = 0, \quad \text{极限 CDF } G(x) = \exp(-x^{-1}), \quad x > 0,$$

密度函数  $g(x) = \exp(-x^{-1})/x^2$ 。

**命题 11.1 (自相关情形: Berman (1964) 条件下的高斯过程极值):** 若  $\{X_t\}$  为边缘分布为  $N(0, 1)$  的平稳高斯过程，自相关函数  $\rho(k)$  满足

$$\rho(k)/\ln k \rightarrow 0 \quad (k \rightarrow \infty),$$

则式 (11.1) 仍成立，且可选用与独立同分布情形相同的一组归一化序列 (即同样的  $a_T > 0$  与  $b_T$ )。对于更一般的非高斯过程也可得到相应结论：弱依赖通常不改变收敛速度，但可能影响极限分布的具体形态 (Embrechts et al. 1997)。

**命题 11.2 (多维高斯分布: 极大值的渐近独立性):** 若

$$\begin{pmatrix} X_t \\ Y_t \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \quad |\rho| < 1,$$

则当  $T \rightarrow \infty$  时， $\max_{1 \leq t \leq T} \{X_t\}$  与  $\max_{1 \leq t \leq T} \{Y_t\}$  (以及对应的极小值) 渐近独立；该性质在一定弱依赖的高斯过程中亦成立 ((Embrechts et al. 1997))。这意味着：高斯范式下，不同资产的极端值不易同时发生 (同一时点出现的概率较低)；而实证市场常见危机期的“尾部共振”现象，提示高斯假设对尾部依赖的刻画不足。

### 11.3 尾部厚度的半参数模型

我们希望在样本范围之外进行外推：仅凭样本，最多只能确定  $\alpha \geq 1/T$  与  $\alpha \leq 1 - 1/T$  的分位数。当  $T$  较小，这不足以覆盖极端小/大的概率水平。引入模型可以对非常小或非常大的  $\alpha$  给出合理预期。下面讨论若干能刻画尾部厚度 (tail thickness) 的模型。正态分布属于薄尾分布 (当  $x \rightarrow \infty$  时  $\Pr(X > x)$  迅速衰减，且所有矩存在)，对日度股票收益通常过于严格； $t$  分布允许更厚的尾；Pareto 分布广泛用于收益与收入分布，对尾部形状有较强的灵活性。

**定义 11.4 (Pareto 分布):** 若对  $x \geq L$  有

$$\Pr(X \leq x) = F(x) = 1 - L^\kappa x^{-\kappa}, \quad \kappa > 0, \quad L > 0,$$

则称  $X$  在  $[L, \infty)$  上服从 Pareto 分布。其密度为  $f(x) = \kappa L^\kappa / x^{\kappa+1}$ ，分位函数为

$$q_{1-\alpha} = L \alpha^{-1/\kappa}.$$

参数  $\kappa$  控制尾部厚度： $\kappa$  越小，尾部越厚；且  $E(|X|^\gamma) < \infty$  当且仅当  $\gamma < \kappa$ 。Pareto 分布可扩展到包含正负值的情形；虽然它排除了如正态分布、GED 等更薄的尾部，但取很大的  $\kappa$  可粗略逼近。完全参数化的 Pareto 分布可用极大似然估计 VaR；然而它刻画整个分布，而实务中我们往往只关心尾部甚至极端尾部。一个折中思路是仅在尾部模仿 Pareto (半参数)，例如柯西分布的尾部与 Pareto 分布同阶，但主体部分行为不同。

**定义 11.5 (半参数尾模型):** 设  $X$  的分布函数为  $F$  (未知)，并且当  $x \rightarrow \infty$  时

$$1 - F(x) \sim L(x) x^{-\kappa},$$

其中  $\kappa$  为尾指数， $L(x)$  为常数或慢变函数 ( $\lim_{x \rightarrow \infty} L(ax)/L(x) = 1$  对所有  $a > 0$  成立)。若  $F$  可导、密度函数为  $f$ ，则  $f(x) \sim \kappa L(x) x^{-(\kappa+1)}$ 。对应的分位函数满足 (当  $\alpha \rightarrow 0$ )

$$q_{1-\alpha} = L^*(1/\alpha) \alpha^{-1/\kappa},$$

其中  $L^*(1/\alpha)$  为常数或慢变函数。

可以分别对左右尾建模，但通常关注下侧尾。柯西分布在上下尾均有  $\kappa = 1$ ；更一般地， $t_\kappa$  分布也满足上述规律。下面给出一个更精确的“尾部厚度”定义。

**定义 11.6 (正则变化尾；尾部厚度):** 若存在  $\kappa \in (0, \infty)$  使得对任意  $x > 0$ ,

$$\lim_{t \rightarrow \infty} \frac{1 - F(tx)}{1 - F(t)} = x^{-\kappa},$$

则称  $X$  的尾部厚度为  $\kappa$ 。

**例 11.5 (慢变函数示例):** 若取  $L(x) = c \log x$  (慢变函数)，则

$$\frac{1 - F(tx)}{1 - F(t)} \sim \frac{c \log t + c \log x}{c \log t} x^{-\kappa} \sim x^{-\kappa} \quad (t \rightarrow \infty),$$

满足上式的形式化定义。

在上述尾假设中,  $L(x)$  的角色弱于  $\kappa$ 。若  $\gamma \geq \kappa$  则  $E(|X|^\gamma) = \infty$ , 而  $\gamma < \kappa$  时有限; 因此  $\kappa$  是刻画极端事件频率与矩存在性的核心参数。该模型仅针对“大小值”(半参数), 不约束  $x$  近 0 的行为; 但“大小值”正是研究样本极值所需。样本最大值的极限律与参数  $\gamma$  相关, 且有  $\kappa = 1/\gamma$ ; 因此  $\kappa > 0$  排除了  $\gamma \leq 0$  的情形——模型强制厚尾, 如正态分布(薄尾)不在其范畴内; 也排除超厚尾(如对数柯西分布, 其尾比任意  $x^{-\kappa}$  衰减更慢)。

Pareto 尾模型广泛见于物理与社会科学(Zipf、Gibrat 定律)。相较高斯分布, 它更能描述高频收益(及成交量)的尾部行为, 也比稳定分布更灵活。事实上, 基于高斯分布创新的强 GARCH 模型可生成满足该尾模型的观测收益。

### 11.3.1 尾厚度的估计

设排序

$$X_{(1)} > \dots > X_{(T)},$$

取阈值整数  $M < T$ 。对  $j \geq M + 1$  的大阶次统计量, 有

$$\log \Pr(X > X_{(j)}) \simeq \kappa \log X_{(j)} + \log L(X_{(j)}).$$

两边减去  $\log \Pr(X > X_{(M+1)})$ , 并利用慢变性  $L(X_{(j)})/L(X_{(M+1)}) \rightarrow 1$ , 得到近似

$$\log\left(\frac{j}{M+1}\right) \simeq \kappa \log\left(\frac{X_{(j)}}{X_{(M+1)}}\right).$$

再由  $\int_1^M \ln x dx = M \ln M - M + 1$  可得  $\log(M+1) - \frac{1}{M} \sum_{j=1}^M \log j \rightarrow 1$ , 于是

$$\frac{1}{M} \sum_{j=1}^M \log\left(\frac{X_{(j)}}{X_{(M+1)}}\right) \simeq \frac{1}{\kappa},$$

据此得到估计思路。

**定义 11.7 (Hill 估计量):** 给定阈值  $M$ , 定义

$$\frac{1}{\hat{\kappa}} = \frac{1}{M} \sum_{j=1}^M \log \frac{X_{(j)}}{X_{(M+1)}}, \quad \hat{\kappa}_{1-\alpha} = \hat{L}_T \alpha^{-1/\hat{\kappa}}, \quad \hat{L}_T = X_{(M+1)} \left(\frac{M}{T}\right)^{1/\hat{\kappa}}, \quad \alpha < M/T.$$

分位数亦可外推为  $\hat{L}_T = X_{(1)}/T^{1/\hat{\kappa}}$ , 此时当  $\alpha = 1/T$  时与样本最大值一致。

**定理 11.2 (渐近正态性):** 若满足冯·米塞斯(von Mises)条件

$$\lim_{x \rightarrow \infty} \frac{xf(x)}{1 - F(x)} = \kappa > 0, \quad f = F',$$

则在  $M \rightarrow \infty$  且  $M/T \rightarrow 0$  下,

$$M^{1/2}(\hat{\kappa} - \kappa) \xrightarrow{d} N(0, \kappa^2).$$

阈值  $M$  的选择需“刚刚好”。最优  $M$  的理论(在附加条件下)此处不赘述, 实证中可

比较不同  $M$  的影响。除 Hill 估计量外，还有替代估计：Gabaix & Ibragimov (2011) 的对数秩  $-1/2$  估计基于回归

$$\log(i - \delta) = a - b \log X_{(i)}, \quad i = 1, \dots, M, \quad \delta \in [0, 1],$$

令  $\hat{b}_\delta \rightarrow \kappa$  (随  $M \rightarrow \infty$ )，并推荐  $\delta = 1/2$  以改善有限样本表现。Dekkers et al. (1989) 指出 Hill 仅覆盖  $\gamma > 0$  的情形，提出  $\gamma = 1/\kappa$  的 DEM 估计量：

$$\hat{\gamma}_{\text{DEM}} = H_T^{(1)} + 1 - \frac{1}{2} \left( 1 - \frac{(H_T^{(1)})^2}{H_T^{(2)}} \right)^{-1}, \quad H_T^{(j)} = \frac{1}{M} \sum_{i=1}^M (\log X_{(i)} - \log X_{(M+1)})^j, \quad j = 1, 2,$$

其对任意  $\gamma \in \mathbb{R}$  相合，并与 Hill 同阶渐近正态；在  $\gamma > 0$  时方差为  $1 + \gamma^2$ ，较 Hill 效率低。时间序列扩展方面，在弱依赖条件下，Hill 的相合性可推广到平稳序列（见 (Resnick & Stărică 1998))；Hill (2010) 在若干条件下给出

$$M^{1/2}(\hat{\kappa} - \kappa) \xrightarrow{d} N(0, \Phi),$$

一般  $\Phi \geq \kappa^2$  且与依赖结构相关；在部分强 GARCH 模型中  $\Phi = \kappa^2$ 。Hill (2006) 给出  $\Phi$  的一致估计：

$$\hat{\Phi} = \frac{1}{m} \sum_{s=1}^T \sum_{t=1}^T K\left(\frac{s-t}{b_T}\right) \hat{Z}_s \hat{Z}_t,$$

其中  $\hat{Z}_t$ 、带宽  $b_T$  与核函数  $K(\cdot)$  的具体实现可参照原文设定。

下文给出的 R 脚本以标普 500 指数（日对数收益率）为样本，构建并检验“尾部厚度”的半参数模型。具体包括：

1. 数据获取与预处理（将收益取负得到亏损变量  $Y = -r$ ，以聚焦左尾分布）；
2. 基于正则变化假设的尾指数估计，给出 Hill 估计与 DEdH（或 DEM）估计及其随阈值  $M$  变化的 Hill 图，用于辅助选择阈值；
3. 采用对数秩回归 ( $\delta = 1/2$ ) 作为 rank-size 关系的佐证性估计；
4. 提供幂律/正则变化的可视化诊断，包括互补累积分布函数 (CCDF) 的双对数图与平均超额函数 (mean excess) 图；
5. 在选择好的  $M$  下，按  $\hat{q}_{1-\alpha} = \hat{L}_T \alpha^{-1/\hat{\kappa}}$  对极端分位数 (VaR) 进行样本外推，并与经验分位数在双对数坐标下比较。该脚本仅覆盖“无条件尾部”部分；若需进一步给出条件 VaR (如 GARCH 与 CAViaR) 及回测检验，可直接与前文的“全流程”脚本拼接使用。

```

1 # ----- 0) 环境 -----
2 pkgs_needed <- c("quantmod", "zoo")
3 to_install <- setdiff(pkgs_needed, rownames(installed.packages()))
4 if (length(to_install)) install.packages(to_install, repos="https://cloud.r-
 project.org")
5 invisible(lapply(pkgs_needed, library, character.only=TRUE))
6
7
8 set.seed(123)

```

```

9
10 # ----- 1) 数据: ^GSPC 与收益 -----
11 # ^GSPC 是 S&P 500 指数 (指数点位, 非可交易资产)
12 getSymbols("^GSPC", src="yahoo", from="1980-01-01", auto.assign=TRUE,
13 warnings=FALSE)
14 px <- Cl(GSPC)
15 ret <- na.omit(100*diff(log(px))) # 日对数收益 (%)
16 colnames(ret) <- "ret"
17
18 # 亏损变量: Y = -ret (只研究左尾)
19 Y <- -as.numeric(ret)
20 Tn <- length(Y)
21
22 # 估计/评估拆分 (本段仅做无条件尾, 全部用于估计和作图)
23 Y_est <- Y
24
25 # ----- 2) 工具函数 -----
26 order_desc <- function(x) sort(x, decreasing=TRUE)
27
28 # Hill 估计 (式 14.13)
29 hill_once <- function(x, M) {
30 x <- x[is.finite(x) & x > 0]
31 xsort <- order_desc(x)
32 stopifnot(M < length(xsort))
33 xm1 <- xsort[M+1]
34 hk <- mean(log(xsort[1:M] / xm1))
35 kappa_hat <- 1 / hk
36 se <- kappa_hat / sqrt(M) # i.i.d. 漐近标准误 (定理 14.1 的方差 σ^2 / M)
37 list(kappa=kappa_hat, se=se, xm1=xm1, xsort=xsort)
38 }
39
40 # DEM (Dekkers - Einmahl - de Haan, 1989) 估计 $\hat{\gamma} = 1 / \hat{\kappa}$ (适用于 $x > 0$ 时方差
41 σ^2)
42 dem_once <- function(x, M) {
43 x <- x[is.finite(x) & x > 0]
44 xsort <- order_desc(x)
45 stopifnot(M < length(xsort))
46 z <- log(xsort[1:M]) - log(xsort[M+1])
47 H1 <- mean(z); H2 <- mean(z^2)
48 gamma_hat <- H1 + 1 - 0.5 * (1 - (H1^2)/H2)^(-1)
49 kappa_hat <- 1/gamma_hat
50 list(kappa=kappa_hat, gamma=gamma_hat)
51 }
52
53 # 分位外推 (式 14.12) : $q_{1-\alpha} = L_T^{-1/\gamma}$, $L_T = x_{(M+1)} (M/T)^{1/\gamma}$
54 tail_quantile_uncond <- function(alpha, x, M, kappa_hat) {
55 x <- x[is.finite(x) & x > 0]
56 xsort <- order_desc(x)
57 Tn <- length(xsort)
 xm1 <- xsort[M+1]
```

```

58 L_T <- xm1 * (M/Tn)^(1/kappa_hat)
59 L_T * alpha^(-1/kappa_hat)
60 }
61
62 # ----- 3) Hill 曲线与阈值 M 的粗选 -----
63 M_grid <- seq(50, max(2000, round(0.1*Tn)), by=10)
64 hill_path <- sapply(M_grid, function(M) hill_once(Y_est, M)$kappa)
65
66 # 选一个平台段内的 M*: 用滚动标准差 (窗口=5) 最小化作启发
67 roll_sd <- zoo::rollapply(hill_path, width=5, sd, align="right", fill=NA)
68 idx_best <- which.min(roll_sd)
69 M_star <- M_grid[idx_best]
70 HILL <- hill_once(Y_est, M_star)
71 DEM <- dem_once(Y_est, M_star)
72
73 cat(sprintf("Hill: ^ =%.3f (se %.3f) at M*=%d\n", HILL$kappa, HILL$se, M_
 star))
74 cat(sprintf(" DEM: ^ =%.3f (^ =%.3f) at M*=%d\n", DEM$kappa, DEM$gamma,
 M_star))
75
76 # ----- 4) 对数秩 (=1/2) 回归 (Gabaix, 2011) -----
77 # 回归: log(i-) = a - b log X_(i), i=1..M*, =1/2, b
78 rank_size_reg <- function(x, M, delta=0.5) {
79 xsort <- order_desc(x)
80 i <- 1:M
81 y <- log(i - delta)
82 X <- log(xsort[i])
83 fit <- lm(y ~ X)
84 b <- -coef(fit)[2] # y = a + (-b) * logX => 斜率的负号是 b
85 list(b=b, fit=fit, i=i, x_top=xsort[i], y=y, X=X)
86 }
87 RS <- rank_size_reg(Y_est, M_star, delta=0.5)
88 cat(sprintf("Rank - Size(=1/2): 斜率 b %.3f ()\n", RS$b))
89
90 # ----- 5) 幂律/正则变动诊断图 -----
91 par(mfrow=c(2,2))
92
93 # (a) Hill 曲线 + M*
94 plot(M_grid, hill_path, type="l", lwd=2, xlab="M (阈值个数)", ylab="Hill 尾
 指数估计 ^",
95 main="Hill 曲线 (选择 M*)")
96 abline(v=M_star, lty=2); abline(h=HILL$kappa, lty=3)
97
98 # (b) Rank - Size 图 (=1/2)
99 plot(log(RS$x_top), RS$y, pch=16, cex=0.7,
100 xlab=expression(log~X[(i)]~"(i=1..M*)"),
101 ylab=expression(log(i-1/2)),
102 main="Rank - Size 回归 (=1/2) ")
103 abline(RS$fit, lwd=2, lty=2)
104 legend("bottomleft", sprintf("b %.2f ()", RS$b), bty="n")
105

```

```

106 # (c) CCDF 双对数图: P(Y>y) vs y
107 Yp <- Y_est[is.finite(Y_est) & Y_est>0]
108 Yp <- sort(Yp)
109 emp_ccdf <- 1 - (1:length(Yp))/length(Yp)
110 plot(log(Yp), log(pmax(emp_ccdf, 1e-8)), type="l", lwd=2,
111 xlab=expression(log~y), ylab=expression(log~bar(F)(y)),
112 main="CCDF 双对数图 (幂律应近线性) ")
113
114 # (d) Mean Excess Plot: e(u)=E[Y-u | Y>u]
115 u_grid <- quantile(Yp, probs=seq(0.8, 0.99, by=0.01), names=FALSE)
116 e_u <- sapply(u_grid, function(u) {
117 exc <- Yp[Yp>u]
118 if (length(exc)>0) mean(exc - u) else NA
119 })
120 plot(u_grid, e_u, type="b", pch=16, xlab="阈值 u (右尾)", ylab="均超额 e(u)"
121 ,
122 main="Mean Excess Plot (Pareto 下近线性上升)")
123 par(mfrow=c(1,1))
124
125 # ----- 6) 极端分位 (VaR) 外推 (式 14.12) -----
126 alphas <- 10^seq(log10(1/50), log10(1/20000), length.out=100) # 从 2% 到
127 0.005%
128 q_emp <- as.numeric(quantile(Y_est, probs=1 - alphas, type=7)) # 经验分位
129 (上限受样本限制)
130 q_ext <- sapply(alphas, tail_quantile_uncond, x=Y_est, M=M_star, kappa_hat=
131 HILL$kappa)
132
133 # 在双对数坐标下比较 经验 vs 外推
134 plot(alphas, q_emp, log="xy", type="l", lwd=2,
135 xlab=expression(alpha), ylab=expression(q[1-alpha]~"(亏损阈值)"),
136 main="经验分位 vs Pareto 外推")
137 lines(alphas, q_ext, lwd=2, lty=2)
138 legend("topright", c("经验分位", "Pareto 外推 (Hill) "), lwd=c(2,2), lty=c
139 (1,2), bty="n")
140
141 # 输出若干代表性 的外推数值
142 alpha_list <- c(1/100, 1/250, 1/1000, 1/5000)
143 q_hat_tab <- sapply(alpha_list, function(a) tail_quantile_uncond(a, x=Y_est,
144 M=M_star, kappa_hat=HILL$kappa))
145 names(q_hat_tab) <- paste0(" =", alpha_list)
146 cat("\n极端分位 (外推, 单位=收益%的亏损阈值)\n")
147 print(round(q_hat_tab, 3))
148
149 # ----- 7) 摘要汇总 -----
150 cat("\n===== 摘要 =====\n")
151 cat(sprintf("Hill(M*=%d): ^ =%.3f, se %.3f\n", M_star, HILL$kappa, HILL$se)
152)
153 cat(sprintf(" DEM(M*=%d): ^ =%.3f (^ =%.3f)\n", M_star, DEM$kappa, DEM$gamma
154))
155 cat(sprintf(" Rank - Size(=1/2, M*=%d): 斜率 b %.3f ()\n", M_star, RS$b

```

```

))
149 cat(" 诊断: CCDF 双对数近线性 & mean excess 上升 幂律尾 (正则变动) 相容。\
 \n")
150 cat(" 分位外推: q_{1- } ~ ^{-1/\gamma} (式 14.12), 仅在 < M/T 的范围内可信。
 \n")
151
152

```

## 11.4 动态模型与 VaR

前述关注无条件分布。现在考虑给定可得信息的条件分布。纳入最新信息可改善当期风险评估，但需要时间序列模型与相应假设。设收益服从 GARCH(1,1) 模型：

$$X_t = \mu + \sigma_t \varepsilon_t, \quad \sigma_t^2 = \omega + \alpha \sigma_{t-1}^2 + \gamma (X_{t-1} - \mu)^2, \quad \varepsilon_t \sim N(0, 1).$$

则条件分位

$$q_\alpha(X_t | \mathcal{F}_{t-1}) = \mu + \sigma_t z_\alpha$$

随  $\sigma_t$  时变。亦可令  $\mu = \mu_t$  随时间变动，实务中用估计的  $\hat{\mu}_t$ 、 $\hat{\sigma}_t$  替代。若认为正态性假设过强，可仅假设  $\varepsilon_t$  为 i.i.d.、密度未知  $f$ ，此时

$$q_\alpha(X_t | \mathcal{F}_{t-1}) = \mu_t + \sigma_t w_\alpha,$$

其中  $w_\alpha$  为  $\varepsilon_t$  的  $\alpha$  分位数； $w_\alpha$  可由标准化残差  $\hat{\varepsilon}_t = (X_t - \hat{\mu}_t)/\hat{\sigma}_t$  的样本分位得到；对极端分位可对  $\hat{\varepsilon}_t$  应用 Hill。若  $w_\alpha > z_\alpha$ ，则条件 VaR 高于高斯假设，但形态相近。

**定义 11.8 (CaViaR 思路; Engle & Manganelli 2004):** 直接对 VaR 动态建模：令

$$\Pr(X_t < -\text{VaR}_t(\alpha) | \mathcal{F}_{t-1}) = \alpha,$$

并考虑如下规格（略去  $\alpha$  的显式记号）：

$$\text{VaR}_t = \text{VaR}_{t-1} + \beta [1\{X_{t-1} \leq -\text{VaR}_{t-1}\} - \alpha],$$

$$\text{VaR}_t = \beta_0 + \beta_1 \text{VaR}_{t-1} + \beta_2 |X_{t-1} - \mu|,$$

$$\text{VaR}_t = \beta_0 + \beta_1 \text{VaR}_{t-1} + \beta_2 |X_{t-1}|.$$

估计上，使用分位回归最小化

$$Q_T(\beta) = \frac{1}{T} \sum_{t=1}^T \rho_\alpha(X_t - \text{VaR}_t(\beta)),$$

其中  $\text{VaR}_t(\beta)$  递推得到；在适度平稳与混合假设下，可得到参数的渐近正态性 (CLT)。作为参照，GARCH(1,1) 的 VaR 可写为

$$\text{VaR}_t = \sigma_t z_\alpha = (\omega + \alpha \sigma_{t-1}^2 + \gamma X_{t-1}^2)^{1/2} z_\alpha,$$

对应 CaViaR 的“间接 GARCH”规格。

### 11.4.1 VaR 模型的评估

给定由  $\{X_{t-T}, \dots, X_{t-1}\}$  构造的  $\text{VaR}_t(\alpha)$ , 可在评估期  $\{1, \dots, H\}$  内检验其校准性。令

$$\hat{p}_\alpha = \frac{1}{H} \sum_{t=1}^H \psi_\alpha(X_t - \text{VaR}_t), \quad \psi_\alpha(x) = \mathbf{1}\{x > 0\} - (1 - \alpha).$$

若  $\text{VaR}_t(\alpha)$  正确, 则  $\hat{p}_\alpha$  的期望为零 (等价地, 其“超越”频率与名义  $\alpha$  一致; 可用 Kupiec 无条件覆盖检验进行形式化检验)。

## 11.5 尾部厚度与动态风险度量：基于 S&P 500 的实证与回测

本节提供一份整合脚本, 对标普 500 指数 (Yahoo Finance 代码 `^GSPC`) 开展从无条件尾部风险到条件风险度量的全流程实证。脚本自动安装并加载所需依赖, 并将重复的工具函数 (如 Hill/DEM 估计、Pareto 外推、Kupiec 检验等) 统一封装, 避免跨章节代码冗余。为便于读者按需运行, 脚本在顶部提供开关 `RUN_A`、`RUN_B`、`RUN_C` 分别控制以下三个模块:

#### 1. 模块 A: 滚动 VaR 与样本外回测。

从 Yahoo Finance 获取 `^GSPC` 的日度收盘价, 转换为对数收益率, 并在固定宽度的滚动窗口下计算 1 日期 VaR ( $\alpha \in \{1\%, 5\%\}$ )。实现四种常见方法: 历史模拟法 (窗口内经验分位数)、高斯 i.i.d. (滚动均值与标准差)、GARCH(1,1)-t (使用 `rugarch` 包的 `ugarchroll` 函数, 每 250 日重估; 学生  $t$  分布用于刻画厚尾特征和条件异方差) 与 EVT-POT (以训练集损失的 95% 分位数为阈值拟合 GPD, 并按闭式公式给出无条件 VaR)。回测采用 Kupiec 无条件覆盖检验, 手工计算违约次数与  $p$  值; 图形部分展示最近一年的日收益与四种方法产生的 99% VaR 轨迹 (将“正的损失阈值”取负以与收益轴对齐)。

#### 2. 模块 B: 尾部厚度的半参数模型。

基于正则变化假设, 仅在尾部引入 Pareto 型近似, 给出尾指数的 Hill 与 DEM 估计, 并通过 Hill 曲线 (随阈值  $M$  变化) 辅助选择阈值。提供幂律/正则变化的可视化诊断: CCDF 双对数图、平均超额函数 (mean excess) 与对数秩 ( $\delta = 1/2$ ) 回归。在所选  $M$  下, 按照

$$\hat{q}_{1-\alpha} = \hat{L}_T \alpha^{-1/\hat{\kappa}}, \quad \hat{L}_T = X_{(M+1)} \left( \frac{M}{T} \right)^{1/\hat{\kappa}},$$

对极端分位数 (VaR) 进行样本外推, 并与经验分位数在双对数坐标下进行对比。需要强调: 该外推仅在  $\alpha < M/T$  范围内可信。

#### 3. 模块 C: 动态模型与 VaR。

考虑条件分布: GARCH(1,1)-正态分布 (先在估计期拟合, 再固定参数对全样本滤波获取  $\mu_t, \sigma_t$ ), 据此得到  $q_\alpha(X_t | \mathcal{F}_{t-1}) = \mu_t + \sigma_t z_\alpha$ ; 进一步以估计期标准化残差估计  $\varepsilon_t$  的左尾分位数  $w_\alpha$ , 在极端  $\alpha$  时采用 Hill 外推, 得到“准非参” VaR:  $q_\alpha = \mu_t + \sigma_t w_\alpha$ 。此外, 构建 CAViaR (SAV) 模型

$$\text{VaR}_t = \beta_0 + \beta_1 \text{VaR}_{t-1} + \beta_2 |X_{t-1}|,$$

用分位回归损失  $Q_T(\beta) = T^{-1} \sum_t \rho_\alpha(X_t - \text{VaR}_t(\beta))$  极小化估计参数, 并在评估期上进行回测 (Kupiec 无条件覆盖与独立性检验)。作图部分对比三种条件 VaR 的轨迹。

注：(i) Yahoo 代码前缀“ $\sim$ ”表示指数（非可交易资产）；(ii) GARCH 的 ugarchfilter 需在 spec 中固定参数后使用（通过 setfixed(spec) 实现），本脚本已统一处理；(iii) EVT-POT 示例采用“固定阈值 + 常数参数”的闭式 VaR 计算方式，若要更贴近业界实践，可改为“滚动阈值 + 滚动拟合”的 POT 方法；(iv) Hill 外推法依赖阈值选择，建议结合 Hill 曲线与诊断图进行稳健性检验。

```

1 # ----- 0) CONFIG -----
2 RUN_A <- TRUE # 四种滚动 VaR + 回测 + 作图
3 RUN_B <- TRUE # 半参数尾厚度 (Hill/DEM 等)
4 RUN_C <- TRUE # 动态 VaR (GARCH-norm/准非参/CaViaR) + 回测
5
6
7 # 数据/窗口/回测参数
8 ticker <- " \sim GSPC" # 可改为 AAPL / MSFT / \sim NDX / \sim HSI 等
9 start_date <- "1950-01-01"
10 window_days <- 250 # 约一年的交易日
11 forecast.length <- 1000 # 回测长度 (A 模块使用)
12 alpha_set <- c(0.01, 0.05)
13 set.seed(1)
14
15 # ----- 1) PACKAGES -----
16 req <- c("quantmod", "PerformanceAnalytics", "rugarch", "evir", "xts", "zoo")
17 inst <- req[!req %in% installed.packages()[,"Package"]]
18 if (length(inst)) install.packages(inst, repos="https://cloud.r-project.org")
19 suppressPackageStartupMessages(invisible(lapply(req, library, character.only
20 = TRUE)))
21 options(stringsAsFactors = FALSE)
22
23 # ----- 2) COMMON UTILS -----
24 # 分位回归损失
25 rho_alpha <- function(u, alpha) u * (alpha - (u < 0))
26
27 # 排序 (降序)
28 order_desc <- function(x) sort(x, decreasing = TRUE)
29
30 # Hill 估计 (i.i.d. 漐近 se kappa/sqrt(M))
31 hill_once <- function(x, M){
32 x <- x[is.finite(x) & x > 0]
33 xsort <- order_desc(x); stopifnot(M < length(xsort))
34 xm1 <- xsort[M+1]
35 hk <- mean(log(xsort[1:M] / xm1))
36 kappa_hat <- 1 / hk
37 se <- kappa_hat / sqrt(M)
38 list(kappa=kappa_hat, se=se, xm1=xm1, xsort=xsort)
39 }
40
41 # DEM (Dekkers – Einmahl – de Haan) , 输出 kappa=1/gamma
42 dem_once <- function(x, M){
43 x <- x[is.finite(x) & x > 0]
44 xsort <- order_desc(x); stopifnot(M < length(xsort))

```

```

44 z <- log(xsort[1:M]) - log(xsort[M+1])
45 H1 <- mean(z); H2 <- mean(z^2)
46 gamma_hat <- H1 + 1 - 0.5*(1 - (H1^2)/H2)^(-1)
47 kappa_hat <- 1 / gamma_hat
48 list(kappa=kappa_hat, gamma=gamma_hat)
49 }
50
51 # Pareto 外推: q_{1- } L_T * ^{-1/ }, L_T = X_{(M+1)} * (M/T)^{1/ }
52 tail_quantile_uncond <- function(alpha, x, M, kappa_hat){
53 x <- x[is.finite(x) & x > 0]
54 xsort <- order_desc(x)
55 Tn <- length(xsort); xm1 <- xsort[M+1]
56 L_T <- xm1 * (M/Tn)^(1/kappa_hat)
57 L_T * alpha^(-1/kappa_hat)
58 }
59
60 # Kupiec UC + 简化独立性检验
61 backtest_basic <- function(x, VaR, alpha){
62 hit <- (x < -VaR)
63 H <- length(hit); n <- sum(hit); p_hat <- n/H
64 LR_uc <- -2*((H-n)*log(1-alpha) + n*log(alpha) -
65 ((H-n)*log(1-p_hat) + n*log(p_hat)))
66 p_uc <- 1 - pchisq(LR_uc, df=1)
67 # 一阶马尔可夫独立性
68 h1 <- hit[-length(hit)]; h2 <- hit[-1]
69 n00 <- sum(h1==0 & h2==0); n01 <- sum(h1==0 & h2==1)
70 n10 <- sum(h1==1 & h2==0); n11 <- sum(h1==1 & h2==1)
71 pi0 <- ifelse(n00+n01>0, n01/(n00+n01), 0)
72 pi1 <- ifelse(n10+n11>0, n11/(n10+n11), 0)
73 pi <- (n01+n11)/(n00+n01+n10+n11)
74 logL_H1 <- n00*log(1-pi0) + n01*log(pi0) + n10*log(1-pi1) + n11*log(pi1)
75 logL_H0 <- (n00+n10)*log(1-pi) + (n01+n11)*log(pi)
76 LR_ind <- -2*(logL_H0 - logL_H1); p_ind <- 1 - pchisq(LR_ind, df=1)
77 list(H=H, n=n, p_hat=p_hat, LR_uc=LR_uc, p_uc=p_uc, LR_ind=LR_ind, p_ind=p_ind,
78 hit=hit)
79 }
80 # 打印回测汇总 (便于不同模型复用)
81 print_bt <- function(name, res){
82 cat(sprintf("%-18s | 命中率=%.3f (n=%d/H=%d) | Kupiec p=%.3f | 独立性 p=%.3
83 f\n",
84 name, resp_hat, resn, resH, resp_uc, res$p_ind))
85 }
86 # ----- 3) DATA -----
87 message("Downloading data from Yahoo Finance ...")
88 x <- suppressWarnings(getSymbols(ticker, from=start_date, auto.assign=FALSE
89))
90 px <- Ad(x)
91 ret <- na.omit(dailyReturn(px, type="log"))
92 colnames(ret) <- "ret"

```

```

92 loss <- -ret # 损失为正
93 stopifnot(NROW(ret) > forecast.length + window_days + 20)
94
95 # 切分 (A 模块)
96 ret_in <- head(ret, -forecast.length)
97 ret_out <- tail(ret, forecast.length)
98 loss_in <- -ret_in
99 loss_out <- -ret_out
100
101 # 切分 (B/C 模块)
102 ret_all <- ret
103 y_all <- as.numeric(ret_all)
104 T_total <- length(y_all)
105 T_est <- floor(0.8 * T_total)
106 ret_est <- ret_all[1:T_est]
107 ret_eval <- ret_all[(T_est+1):T_total]
108 y_est <- y_all[1:T_est]
109
110 # ----- A) 四种滚动 VaR (样本外) -----
111 if (RUN_A){
112 message("A) Rolling VaR: Hist / Gaussian / GARCH-t / EVT-POT ...")
113
114 # A1 历史模拟 (滚动窗口经验分位)
115 hist_roll_var <- sapply(1:forecast.length, function(i){
116 idx_end <- NROW(loss_in) + i - 1
117 idx_start <- idx_end - window_days + 1
118 w <- loss[idx_start:idx_end, 1]
119 c(VaR_99 = as.numeric(quantile(w, 0.99, na.rm=TRUE)),
120 VaR_95 = as.numeric(quantile(w, 0.95, na.rm=TRUE)))
121 })
122 hist_roll_var <- xts(t(hist_roll_var), order.by=index(ret_out))
123 colnames(hist_roll_var) <- c("VaR_99", "VaR_95")
124
125 # A2 高斯 i.i.d. (滚动均值/标准差)
126 gauss_roll_var <- sapply(1:forecast.length, function(i){
127 idx_end <- NROW(loss_in) + i - 1
128 idx_start <- idx_end - window_days + 1
129 w <- loss[idx_start:idx_end, 1]
130 mu <- mean(w); sig <- sd(w)
131 c(VaR_99 = as.numeric(qnorm(0.99, mu, sig)),
132 VaR_95 = as.numeric(qnorm(0.95, mu, sig)))
133 })
134 gauss_roll_var <- xts(t(gauss_roll_var), order.by=index(ret_out))
135 colnames(gauss_roll_var) <- c("VaR_99", "VaR_95")
136
137 # A3 GARCH(1,1)-t (ugarchroll, 每 250 日重估)
138 spec_t <- ugarchspec(
139 variance.model=list(model="sGARCH", garchOrder=c(1,1)),
140 mean.model =list(armaOrder=c(0,0), include.mean=TRUE),
141 distribution.model="std"
142)

```

```

143 roll <- ugarchroll(
144 spec_t, data=ret_all, n.ahead=1, forecast.length=forecast.length,
145 refit.every=window_days, refit.window="moving",
146 solver="hybrid", solver.control=list(trace=0), keep.coef=TRUE
147)
148 dens_df <- as.data.frame(roll@forecast$density)
149 nms <- colnames(dens_df)
150 idx_mu <- which(nms == "Mu")
151 idx_sigma <- which(nms == "Sigma")
152 idx_shape <- grep("shape|df|nu", nms, ignore.case=TRUE)
153 stopifnot(length(idx_mu)*length(idx_sigma)*length(idx_shape) > 0)
154 mu_t <- as.numeric(dens_df[, idx_mu[1]])
155 sg_t <- as.numeric(dens_df[, idx_sigma[1]])
156 nu_t <- as.numeric(dens_df[, idx_shape[1]])
157 q01 <- qdist("std", 0.01, mu=0, sigma=1, skew=1, shape=nu_t)
158 q05 <- qdist("std", 0.05, mu=0, sigma=1, skew=1, shape=nu_t)
159 garch_t_var_loss <- xts(cbind(
160 VaR_99 = -(mu_t + sg_t*q01),
161 VaR_95 = -(mu_t + sg_t*q05)
162), order.by=index(ret_out))
163
164 # A4 EVT-POT (训练集 95% 阈值, 常参闭式 VaR)
165 u <- as.numeric(quantile(loss_in, 0.95))
166 fit_gpd <- evir::gpd(as.numeric(loss_in), u)
167 pe <- fit_gpd$par.est; nm <- tolower(names(pe))
168 xi <- as.numeric(pe[which(nm=="xi")[1]])
169 beta <- as.numeric(pe[which(nm %in% c("beta", "scale"))[1]])
170 lambda_hat <- mean(as.numeric(loss_in) > u)
171 gpd_var_uncond <- function(p, u, xi, beta, lambda){
172 if (abs(xi) < 1e-8) u + beta*log(lambda/(1-p))
173 else u + (beta/xi) * ((lambda/(1-p))^xi - 1)
174 }
175 evt_99 <- gpd_var_uncond(0.99, u, xi, beta, lambda_hat)
176 evt_95 <- gpd_var_uncond(0.95, u, xi, beta, lambda_hat)
177 evt_var_loss <- xts(cbind(
178 VaR_99 = rep(evt_99, NROW(ret_out)),
179 VaR_95 = rep(evt_95, NROW(ret_out))
180), order.by=index(ret_out))
181
182 # A5 回测 (UC)
183 uc <- function(actual, var, a, name){
184 a_num <- as.numeric(actual); v_num <- as.numeric(var)
185 keep <- is.finite(a_num) & is.finite(v_num)
186 res <- backtest_basic(a_num[keep], v_num[keep], a)
187 print_bt(name, res); invisible(res)
188 }
189 cat("\n[A] 回测 (样本外) \n")
190 btA <- list(
191 hist99 = uc(ret_out, hist_roll_var$VaR_99, 0.01, "Hist 99%"),
192 hist95 = uc(ret_out, hist_roll_var$VaR_95, 0.05, "Hist 95%"),
193 gau99 = uc(ret_out, gauss_roll_var$VaR_99, 0.01, "Normal 99%"),

```

```

194 gau95 = uc(ret_out, gauss_roll_var$VaR_95, 0.05, "Normal 95%"),
195 gar99 = uc(ret_out, garch_t_var_loss$VaR_99, 0.01, "GARCH-t 99%"),
196 gar95 = uc(ret_out, garch_t_var_loss$VaR_95, 0.05, "GARCH-t 95%"),
197 evt99 = uc(ret_out, evt_var_loss$VaR_99, 0.01, "EVT 99%"),
198 evt95 = uc(ret_out, evt_var_loss$VaR_95, 0.05, "EVT 95%")
199)
200
201 # A6 输出今日 99% VaR (近似百分比跌幅)
202 last_99 <- c(
203 Historical = as.numeric(last(hist_roll_var$VaR_99)),
204 Gaussian = as.numeric(last(gauss_roll_var$VaR_99)),
205 GARCH_t = as.numeric(last(garch_t_var_loss$VaR_99)),
206 EVT_POT = as.numeric(last(evt_var_loss$VaR_99))
207)
208 VaR_to_pct <- function(v) 100*(1 - exp(-v))
209 cat("\n[A] 最近 1 步 99% VaR (正损失阈值, log-return 单位) \n")
210 print(round(last_99, 5))
211 cat("\n[A] 约当百分比跌幅 (%) \n")
212 print(round(VaR_to_pct(last_99), 2))
213
214 # A7 作图: 最近一年收益与 99% VaR
215 message("[A] Plotting last ~1y returns & 99% VaR ...")
216 op <- par(no.readonly=TRUE); on.exit(par(op), add=TRUE)
217 par(mar=c(4,4,2,1))
218 plot_ret <- tail(ret_out, 252)
219 plot(index(plot_ret), coredata(plot_ret), type="h",
220 main=paste0(ticker, " Returns & 99% VaR (last ~1y)"),
221 xlab="", ylab="log-return", col="grey40")
222 lines(index(plot_ret), -coredata(hist_roll_var$VaR_99[index(plot_ret)]),
223 lwd=2)
224 lines(index(plot_ret), -coredata(gauss_roll_var$VaR_99[index(plot_ret)]),
225 lwd=2, lty=2)
226 lines(index(plot_ret), -coredata(garch_t_var_loss$VaR_99[index(plot_ret)]),
227 lwd=2, lty=3)
228 lines(index(plot_ret), -coredata(evt_var_loss$VaR_99[index(plot_ret)]),
229 lwd=2, lty=4)
230 legend("bottomleft",
231 c("Returns", "Hist 99%", "Normal 99%", "GARCH-t 99%", "EVT 99%"),
232 lty=c(1,1,2,3,4), lwd=c(1,2,2,2,2), bty="n")
233 }
234
235 # ----- B) 半参数尾厚度 -----
236 if (RUN_B){
237 message("B) Semi-parametric tail thickness: Hill/DEM/diagnostics ...")
238
239 M_grid <- seq(50, max(2000, round(0.1*T_est)), by=10)
240 hill_path <- sapply(M_grid, function(M) hill_once(y_est, M)$kappa)
241 roll_sd <- zoo::rollapply(hill_path, width=5, sd, align="right", fill=NA)
242 M_star <- M_grid[which.min(roll_sd)]
243 HILL <- hill_once(y_est, M_star)
244 DEM <- dem_once(y_est, M_star)

```

```

241 cat(sprintf("[B] Hill: ^ =%.3f (se %.3f) @ M*=%d\n", HILL$kappa, HILL$se,
242 M_star))
243 cat(sprintf("[B] DEM: ^ =%.3f (^ =%.3f) @ M*=%d\n", DEM$kappa, DEM$gamma,
244 M_star))
245
246 # 诊断与外推作图
247 par(mfrow=c(2,2))
248 plot(M_grid, hill_path, type="l", lwd=2,
249 xlab="M", ylab="Hill ^ ", main="Hill 曲线 (阈值选择)")
250 abline(v=M_star, lty=2); abline(h=HILL$kappa, lty=3)
251
252 # Rank-Size (=1/2)
253 xsort <- order_desc(y_est)
254 i <- 1:M_star; delta <- 0.5
255 y_rs <- log(i - delta); X_rs <- log(xsort[i])
256 fit_rs <- lm(y_rs ~ X_rs)
257 b_hat <- -coef(fit_rs)[2]
258 plot(X_rs, y_rs, pch=16, cex=0.7,
259 xlab=expression(log~X[(i)]), ylab=expression(log(i-1/2)),
260 main="Rank-Size 回归 (=1/2)")
261 abline(fit_rs, lwd=2, lty=2)
262 legend("bottomleft", sprintf("斜率 b %.2f ()", b_hat), bty="n")
263
264 # CCDF 双对数
265 Yp <- sort(y_est[is.finite(y_est) & y_est>0])
266 emp_ccdf <- 1 - (1:length(Yp))/length(Yp)
267 plot(log(Yp), log(pmax(emp_ccdf, 1e-10)), type="l", lwd=2,
268 xlab=expression(log~y), ylab=expression(log~bar(F)(y)),
269 main="CCDF 双对数图")
270
271 # Mean Excess
272 u_grid <- quantile(Yp, probs=seq(0.8, 0.99, by=0.01), names=FALSE)
273 e_u <- sapply(u_grid, function(u){
274 exc <- Yp[Yp>u]; if (length(exc)>0) mean(exc - u) else NA
275 })
276 plot(u_grid, e_u, type="b", pch=16, xlab="阈值 u", ylab="均超额 e(u)",
277 main="Mean Excess Plot (Pareto 下上升)")
278 par(mfrow=c(1,1))
279
280 # 极端分位外推
281 alphas <- c(1/100, 1/250, 1/1000, 1/5000)
282 q_hat <- sapply(alphas, tail_quantile_uncond, x=y_est, M=M_star, kappa_
283 hat=HILL$kappa)
284 names(q_hat) <- paste0(" =", alphas)
285 cat("[B] 外推 q_{1- } (单位=亏损阈值, % 收益的近似) :\n")
286 print(round(q_hat, 3))
287 }
288
289 # ----- C) 动态 VaR -----
290 if (RUN_C){
291 message("C) Dynamic VaR: GARCH-norm / quasi-nonparam / CaViaR (SAV) ...")
```

```

289
290 # C1 GARCH(1,1)-norm (先在估计期拟合, 再固定参数 filter 全样本)
291 spec_n <- ugarchspec(
292 variance.model=list(model="sGARCH", garchOrder=c(1,1)),
293 mean.model =list(armaOrder=c(0,0), include.mean=TRUE),
294 distribution.model="norm"
295)
296 fit_n <- ugarchfit(spec_n, ret_est, solver="hybrid")
297 spec_fix <- spec_n; setfixed(spec_fix) <- as.list(coef(fit_n))
298 filt <- ugarchfilter(spec_fix, data=ret_all)
299 mu_all <- as.numeric(fitted(filt)); sig_all <- as.numeric(sigma(filt))
300 mu_t <- mu_all[(T_est+1):T_total]; sig_t <- sig_all[(T_est+1):T_total]
301
302 # 选择 =1% 进行展示 (可改)
303 alpha_v <- 0.01
304 VaR_garch_norm <- -(mu_t + sig_t * qnorm(alpha_v))
305
306 # C2 准非参: 估计期残差的左尾分位 w_ (极端 则 Hill 外推)
307 eps_est <- residuals(fit_n, standardize=TRUE)
308 W <- -as.numeric(na.omit(eps_est)) # 右尾
309 M_eps <- max(50, min(if (RUN_B) M_star else floor(0.1*length(W))-1, length(W)-10))
310 H_eps <- hill_once(W, M_eps)
311 w_alpha_hat <- function(alpha){
312 T_eps <- length(W)
313 if (alpha >= 1/T_eps){
314 unname(quantile(-W, probs=alpha, type=7)) # 的左尾
315 } else {
316 -tail_quantile_uncond(alpha, x=W, M=M_eps, kappa_hat=H_eps$kappa)
317 }
318 }
319 VaR_garch_qnpar <- -(mu_t + sig_t * w_alpha_hat(alpha_v))
320
321 # C3 CaViaR (SAV) : VaR_t = b0 + b1 VaR_{t-1} + b2 |X_{t-1}|
322 x_eval <- as.numeric(ret_eval)
323 caviar_fit <- function(x, alpha, VaR0_init){
324 loss <- function(par, x, alpha, VaR0){
325 b0 <- par[1]; b1 <- par[2]; b2 <- par[3]
326 H <- length(x); VaR <- numeric(H)
327 VaR[1] <- b0 + b1*VaR0 + b2*abs(0)
328 for (t in 2:H) VaR[t] <- b0 + b1*VaR[t-1] + b2*abs(x[t-1])
329 sum(rho_alpha(x + VaR, alpha))
330 }
331 opt <- optim(c(0.1,0.9,0.1), loss, x=x, alpha=alpha, VaR0=VaR0_init,
332 method="Nelder-Mead", control=list(maxit=5000))
333 par <- opt$par; H <- length(x); VaR <- numeric(H)
334 VaR[1] <- par[1] + par[2]*VaR0_init + par[3]*abs(0)
335 for (t in 2:H) VaR[t] <- par[1] + par[2]*VaR[t-1] + par[3]*abs(x[t-1])
336 list(VaR=VaR, par=par)
337 }
338 VaR0_init <- as.numeric(VaR_garch_norm[1])

```

```

339 cav <- caviar_fit(x_eval, alpha=alpha_v, VaR0_init=VaR0_init)
340 VaR_caviar <- cav$VaR
341
342 # C4 回测
343 bt_gn <- backtest_basic(as.numeric(ret_eval), VaR_garch_norm, alpha_v)
344 bt_gq <- backtest_basic(as.numeric(ret_eval), VaR_garch_qnpar, alpha_v)
345 bt_cav <- backtest_basic(as.numeric(ret_eval), VaR_caviar, alpha_v)
346 cat("\n[C] 回测 (alpha=", alpha_v, ") \n", sep="")
347 print_bt("GARCH-norm", bt_gn)
348 print_bt("GARCH-qnpar", bt_gq)
349 print_bt("CaViaR(SAV)", bt_cav)
350
351 # C5 作图 (评估期)
352 op <- par(mar=c(4,4,2,2))
353 plot(index(ret_eval), as.numeric(ret_eval), type="h",
354 main=paste0("评估期收益与 VaR (= ", 100*alpha_v, "%) "),
355 xlab="", ylab="日收益(%)", col="grey40")
356 lines(index(ret_eval), -VaR_garch_norm, lwd=2)
357 lines(index(ret_eval), -VaR_garch_qnpar, lwd=2, lty=2)
358 lines(index(ret_eval), -VaR_caviar, lwd=2, lty=3)
359 abline(h=0, col="darkgrey")
360 legend("bottomleft",
361 c("收益", "GARCH-norm", "GARCH-qnpar", "CaViaR(SAV)"),
362 lwd=c(1,2,2,2), lty=c(1,1,2,3), col=c("grey40", "black", "black",
363 "black"),
364 bty="n")
364 par(op)
365 }
366
367 message("==== Done ====")
368

```

## 11.6 多变量情形

前文从单变量角度讨论了尾部厚度与风险度量；而在实践中，投资者更关注由多类风险资产构成的组合。常用思路是将“边际分布”与“依赖结构”分离建模：先将各维随机变量通过分位映射统一到  $[0, 1]$  尺度，再用 Copula 刻画联合依赖关系。

设  $X_1, X_2$  连续分布，记

$$Y_1 = F_{X_1}(X_1), \quad Y_2 = F_{X_2}(X_2),$$

则  $Y_1, Y_2 \sim \text{Unif}[0, 1]$ 。定义

$$C(u_1, u_2) = \Pr(Y_1 \leq u_1, Y_2 \leq u_2),$$

称为  $(X_1, X_2)$  的 Copula；它是  $[0, 1]^2$  上的二维分布函数。由此， $(X_1, X_2)$  的联合分布可以由  $C$  与两个边际  $F_{X_1}, F_{X_2}$  等价表述，实现“边际—依赖”的解耦：边际由各自的合适模型拟合，依赖结构由  $C$  单独建模（见 Sklar 定理）。

**定理 11.3 (Sklar 定理 (Sklar 1959)):** 若  $X_1, X_2$  连续，则存在唯一的  $C : [0, 1]^2 \rightarrow [0, 1]$  使

$$\Pr(X_1 \leq x_1, X_2 \leq x_2) = C(F_{X_1}(x_1), F_{X_2}(x_2)).$$

当  $C(u_1, u_2) = u_1 u_2$  时两变量独立；其他  $C$  则允许更丰富的依赖（含尾部依赖与不对称性）。

**例 11.6 (高斯 Copula):**

$$C(u_1, u_2; \rho) = \Phi_2(\Phi^{-1}(u_1), \Phi^{-1}(u_2); \rho),$$

其中  $\Phi$  为标准正态分布函数， $\Phi_2(\cdot, \cdot; \rho)$  为相关系数为  $\rho$  的标准二维正态分布函数。对应联合分布与密度可写为

$$F(x_1, x_2) = C(F_1(x_1), F_2(x_2); \rho), \quad f(x_1, x_2) = c(F_1(x_1), F_2(x_2); \rho) f_1(x_1) f_2(x_2),$$

其中  $c = \partial_{u_1} \partial_{u_2} C$  为 Copula 密度。高斯 Copula 在信用风险中广泛使用，因其对边际分布要求宽松（不必为正态分布）；但其尾依赖为零，在极端情况下可能低估“同涨同跌”的聚集性，曾引发广泛反思。

**定义 11.9 (对称 Joe–Clayton (SJC) Copula (Patton 2006)):** 设

$$C_{\text{SJC}}(u, v) = \frac{1}{2} \left( C_{\text{JC}}(u, v | \tau^U, \tau^L) + C_{\text{JC}}(1 - u, 1 - v | \tau^L, \tau^U) + u + v - 1 \right),$$

$$1 - C_{\text{JC}}(u, v | \tau^U, \tau^L) = \left( 1 - \left[ (1 - (1 - u)^\kappa)^{-\gamma} + (1 - (1 - v)^\kappa)^{-\gamma} - 1 \right]^{-1/\gamma} \right)^{1/\kappa},$$

其中  $\kappa = 1 / \log_2(2 - \tau^U)$ ,  $\gamma = -1 / \log_2(\tau^L)$ ,  $\tau^L, \tau^U$  为尾依赖系数（下、上）：

$$\tau^L = \lim_{\varepsilon \rightarrow 0+} \frac{C(\varepsilon, \varepsilon)}{\varepsilon}, \quad \tau^U = \lim_{\varepsilon \rightarrow 1-} \frac{C(\varepsilon, \varepsilon)}{\varepsilon}.$$

SJC 允许上下尾依赖不对称。实证上，[Patton \(2006\)](#) 发现欧元—美元与日元—美元汇率在贬值期相关性更强（上、下尾不对称）。估计与推断参见 [Chen & Fan \(2006\)](#), [Patton \(2012\)](#)。

**定义 11.10 (系统性风险的 CoVaR(Adrian & Brunnermeier 2016)):** 机构  $j$  的条件 VaR 定义为

$$\Pr\left(X^j \leq \text{CoVaR}_{\alpha}^{j|\mathcal{C}(X^i)} \mid \mathcal{C}(X^i)\right) = \alpha,$$

其中  $\mathcal{C}(X^i)$  表示对机构  $i$  的状态施加的约束（条件）。进一步定义“边际贡献”

$$\Delta \text{CoVaR}_{\alpha}^{ji} = \text{CoVaR}_{\alpha}^{j|X^i=\text{VaR}_{\alpha}^i} - \text{CoVaR}_{\alpha}^{j|X^i=\text{Median}(X^i)}.$$

### 11.6.1 多变量依赖与系统性风险：Copula 与 CoVaR 的实证演示

本小节使用 Yahoo Finance 的 ^GSPC 与 ^IXIC 日对数收益。首先用经验分布函数 (empirical CDF) 将各边际变量映射为伪观测  $U \in (0, 1)$ ；随后分别拟合两类 Copula：

- 高斯 Copula (仅刻画线性相关, 尾依赖为零);
- Joe–Clayton/BB7 Copula (允许不对称的上下尾依赖)。

拟合后报告上下尾依赖系数  $\lambda_L, \lambda_U$ , 并基于 Copula 的条件分布逆 (Hinv) 计算 CoVaR 与  $\Delta$ CoVaR:

$$\text{CoVaR}_{\alpha}^{j|i} : \Pr(X^j \leq \text{CoVaR}_{\alpha}^{j|X^i=\text{VaR}_{\alpha}^i} \mid X^i = \text{VaR}_{\alpha}^i) = \alpha,$$

$$\Delta\text{CoVaR}_{\alpha}^{ji} = \text{CoVaR}_{\alpha}^{j|X^i=\text{VaR}_{\alpha}^i} - \text{CoVaR}_{\alpha}^{j|X^i=\text{Median}(X^i)}.$$

脚本对不同版本的 VineCopula 做了兼容 (自适应 BiCopHinv1 的参数签名), 并提供  $U$ -空间与原始收益空间的可视化, 以展示“对触发资产施加极端条件  $\Rightarrow$  另一资产条件分位点下移”的系统性效应。读者可更换标的资产与  $\alpha$  (如 1% 或 5%), 或使用 BiCopSelect 进行 Copula 自动选择。

```

1 # 0) 环境准备 ----
2 req <- c("quantmod", "VineCopula", "copula", "xts", "zoo")
3 inst <- setdiff(req, rownames(installed.packages()))
4 if (length(inst)) install.packages(inst, repos = "https://cloud.r-project.org")
5 suppressPackageStartupMessages(invisible(lapply(req, library, character.only = TRUE)))
6
7 ticker_i <- "^GSPC" # 资产 i (触发方)
8 ticker_j <- "^IXIC" # 资产 j (响应方)
9 start_date <- "2000-01-01"
10 alpha <- 0.05 # CoVaR 目标分位 (例如 0.05 或 0.01)
11
12 set.seed(1)
13
14 # 1) 数据: 复权收盘价与对数收益 ----
15 x_i <- suppressWarnings(getSymbols(ticker_i, src="yahoo", from=start_date,
16 auto.assign=FALSE))
16 x_j <- suppressWarnings(getSymbols(ticker_j, src="yahoo", from=start_date,
17 auto.assign=FALSE))
17
18 px_i <- Ad(x_i)
19 px_j <- Ad(x_j)
20
21 px <- na.omit(merge(px_i, px_j))
22 colnames(px) <- c("Pi", "Pj")
23
24 ret <- na.omit(diff(log(px))) # 日度对数收益
25 colnames(ret) <- c("ri", "rj")
26
27 stopifnot(NROW(ret) > 500)
28
29 # 2) 伪观测 (经验 CDF 秩) ----
30 pobs_emp <- function(x) rank(x, ties.method="average") / (length(x) + 1)
31 Ui <- pobs_emp(as.numeric(ret$ri))
32 Uj <- pobs_emp(as.numeric(ret$rj))
33 U <- cbind(Ui, Uj)
34

```

```

35 # 3) Copula 拟合 ----
36 # 3.1 高斯 Copula (family = 1)
37 fit_gauss <- VineCopula::BiCopEst(u1=U[,1], u2=U[,2], family=1, method="mle")
38 rho_hat <- fit_gauss$par
39 cat(sprintf("Gaussian Copula: rho_hat = %.3f\n", rho_hat))
40
41 # 3.2 Joe - Clayton / BB7 (family = 9) , 支持不对称尾依赖
42 fit_bb7 <- VineCopula::BiCopEst(u1=U[,1], u2=U[,2], family=9, method="mle")
43 par1 <- fit_bb7$par
44 par2 <- fit_bb7$par2
45 td <- VineCopula::BiCopPar2TailDep(family=9, par=par1, par2=par2)
46 lambdaL <- td$lower; lambdaU <- td$upper
47 cat(sprintf("BB7 (Joe - Clayton): par1=%.3f, par2=%.3f | _L=%.3f, _U=%.3f\n",
48 ",
49 par1, par2, lambdaL, lambdaU))
50
51 # 信息准则 (可选)
52 ll_g <- fit_gauss$logLik; k_g <- 1
53 ll_b <- fit_bb7$logLik; k_b <- 2
54 n <- nrow(U)
55 AIC_g <- -2*ll_g + 2*k_g; BIC_g <- -2*ll_g + log(n)*k_g
56 AIC_b <- -2*ll_b + 2*k_b; BIC_b <- -2*ll_b + log(n)*k_b
57 cat(sprintf("IC: Gauss AIC=%.1f BIC=%.1f | BB7 AIC=%.1f BIC=%.1f\n",
58 AIC_g, BIC_g, AIC_b, BIC_b))
59
60 # 4) 版本兼容的 H-inverse 包装器 (BiCopHinv1) ----
61 hinv1 <- function(alpha, u1, family, par, par2 = 0) {
62 fn <- get("BiCopHinv1", asNamespace("VineCopula"))
63 fml <- names(formals(fn))
64 # 先尝试按参数名 (h 或 t) , 否则使用位置参数
65 if ("h" %in% fml) return(fn(h = alpha, u1 = u1, family = family, par = par,
66 , par2 = par2))
67 if ("t" %in% fml) return(fn(t = alpha, u1 = u1, family = family, par = par,
68 , par2 = par2))
69 # 位置参数调用顺序: (h, u1, family, par, par2)
70 fn(alpha, u1, family, par, par2)
71 }
72
73 # 5) CoVaR 与 ΔCoVaR 的辅助函数 ----
74 # 将 u(0,1) 映回到原始 X 空间 (经验分位)
75 q_emp <- function(x, u) unname(quantile(x, probs = min(max(u, 1e-6), 1-1e-6),
76 ,
77 type=7, na.rm=TRUE))
78
79 # 计算给定 Copula 的 CoVaR_{alpha}^{j | X_i = VaR_alpha^i} 与 ΔCoVaR (方向 i
80 # -> j)
81 covar_from_copula <- function(family, par, par2 = 0, ri, rj, alpha){
82 VaR_i <- q_emp(ri, alpha) # 资产 i 的左尾 VaR
83 ui_star <- alpha # 在伪观测下, Xi=VaR 近似对应 Ui=
84 uj_star <- hinv1(alpha, ui_star, family = family, par = par, par2 = par2)

```

```

80 CoVaR <- q_emp(rj, uj_star)
81 # 基线: Xi=median (Ui = 0.5)
82 uj_med <- hinv1(alpha, 0.5, family = family, par = par, par2 = par2)
83 CoVaR_med <- q_emp(rj, uj_med)
84 list(CoVaR = CoVaR, CoVaR_med = CoVaR_med, Delta = CoVaR - CoVaR_med,
85 u_i = ui_star, u_j = uj_star, u_j_med = uj_med, VaR_i = VaR_i)
86 }
87
88 ri <- as.numeric(ret$ri)
89 rj <- as.numeric(ret$rj)
90
91 # i -> j 的 CoVaR
92 covar_g_ij <- covar_from_copula(family=1, par=rho_hat, ri=ri, rj=rj, alpha
93 =alpha)
94 covar_bb7_ij <- covar_from_copula(family=9, par=par1, par2=par2, ri=ri, rj=
95 rj, alpha=alpha)
96
97 cat(sprintf("\nCoVaR (i→j), =%.2f:\n", alpha))
98 cat(sprintf(" Gaussian: CoVaR=%.4f, baseline=%.4f, ΔCoVaR=%.4f\n",
99 covar_g_ij$CoVaR, covar_g_ij$CoVaR_med, covar_g_ij$Delta))
100 cat(sprintf(" BB7 (JC): CoVaR=%.4f, baseline=%.4f, ΔCoVaR=%.4f\n",
101 covar_bb7_ij$CoVaR, covar_bb7_ij$CoVaR_med, covar_bb7_ij$Delta))
102
103 # j -> i 的 CoVaR (交换角色)
104 covar_g_ji <- covar_from_copula(family=1, par=rho_hat, ri=rj, rj=ri, alpha
105 =alpha)
106 covar_bb7_ji <- covar_from_copula(family=9, par=par1, par2=par2, ri=rj, rj=
107 ri, alpha=alpha)
108
109 cat(sprintf("\nCoVaR (j→i), =%.2f:\n", alpha))
110 cat(sprintf(" Gaussian: CoVaR=%.4f, baseline=%.4f, ΔCoVaR=%.4f\n",
111 covar_g_ji$CoVaR, covar_g_ji$CoVaR_med, covar_g_ji$Delta))
112 cat(sprintf(" BB7 (JC): CoVaR=%.4f, baseline=%.4f, ΔCoVaR=%.4f\n",
113 covar_bb7_ji$CoVaR, covar_bb7_ji$CoVaR_med, covar_bb7_ji$Delta))
114
115 # 6) 作图 ----
116 op <- par(no.readonly=TRUE); on.exit(par(op), add=TRUE)
117
118 par(mfrow=c(1,2), mar=c(4,4,2,1))
119 # U 空间散点 (i→j 条件点)
120 plot(U[,1], U[,2], pch=16, cex=0.5, col=rgb(0,0,0,0.25),
121 xlab=expression(U[i]), ylab=expression(U[j]),
122 main="U 空间 (伪观测) ")
123 abline(v=alpha, col="grey60", lty=2)
124 points(alpha, covar_g_ij$u_j, pch=19, col="steelblue")
125 points(alpha, covar_bb7_ij$u_j, pch=19, col="tomato")
126 legend("topleft",
127 c("样本", "Ui= ", "u_j* (Gauss)", "u_j* (BB7)"),
128 pch=c(16,NA,19,19), lty=c(NA,2,NA,NA),
129 col=c(rgb(0,0,0,0.25), "grey60", "steelblue", "tomato"), bty="n", cex
130 =0.85)

```

```

126
127 # X 空间散点与 CoVaR 水平线 (i+j)
128 plot(ri, rj, pch=16, cex=0.5, col=rgb(0,0,0,0.25),
129 xlab=expression(X^i), ylab=expression(X^j),
130 main = bquote(CoVaR~"(" * i %->% j * ")" ~~~ alpha == .(alpha)))
131 abline(v = covar_g_ij$VaR_i, col="grey60", lty=2)
132 abline(h = covar_g_ij$CoVaR, col="steelblue", lty=1)
133 abline(h = covar_bb7_ij$CoVaR, col="tomato", lty=1)
134 legend("bottomright",
135 c(expression(X^i==VaR[alpha]^i),"CoVaR (Gauss)","CoVaR (BB7)"),
136 lty=c(2,1,1), col=c("grey60","steelblue","tomato"), bty="n", cex=0.9)
137
138 par(mfrow=c(1,1))

```

## 11.7 一致性风险度量

何谓“好的”风险度量? Artzner et al. (1999) 提出四条被广泛接受的公理。令  $R$  为与价值  $W$  关联的风险度量:

- |                 |                                                                                                                                                                                                                                                                                                                                                                                                   |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 定义 11.11 (四公理): | <ul style="list-style-type: none"> <li>• <b>单调性:</b> 若 <math>W</math> 在一阶随机占优意义上劣于 <math>W^*</math>, 则 <math>R(W) \geq R(W^*)</math>;</li> <li>• <b>现金不变性 (平移不变性):</b> <math>\forall c, R(W + c) = R(W) - c</math>;</li> <li>• <b>齐次性:</b> <math>\forall \lambda \geq 0, R(\lambda W) = \lambda R(W)</math>;</li> <li>• <b>次可加性 (分散化):</b> <math>R(W + W^*) \leq R(W) + R(W^*)</math>。</li> </ul> |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

上述四公理将“好的风险度量”刻画为一种**货币化**的资本需求:  $R(W)$  越大, 头寸越“危险”。单调性说明: 若  $W$  的分布在一阶随机占优意义上劣于  $W^*$  (即各分位点表现同等或更差), 所需资本不应更低。现金不变性表示: 往头寸中确定性地注入现金  $c$  会把风险恰好降低  $c$ ——现金一元等价地抵消一元风险。齐次性表达“规模效应线性”: 在不考虑规模不经济或交易成本的情况下, 把头寸放大  $\lambda$  倍, 风险也应放大  $\lambda$  倍。次可加性体现“**分散化不吃亏**”: 组合的总体风险不应超过分拆后各自风险之和, 从而鼓励风险汇总与分散。实际中, 标准差不满足现金不变性, VaR 可能违背次可加性, 而 ES (期望损失) 满足全部四条公理。

- |                           |
|---------------------------|
| 例 11.7 (标准差满足齐次性与次可加性): 由 |
|---------------------------|

$$\text{var}(X + Y) = \text{var}(X) + \text{var}(Y) + 2 \text{cov}(X, Y) \leq (\text{sd}(X) + \text{sd}(Y))^2$$

|                                                                                              |
|----------------------------------------------------------------------------------------------|
| 可知 $\text{sd}(X + Y) \leq \text{sd}(X) + \text{sd}(Y)$ 。但标准差对常数平移不满足 $R(W + c) = R(W) - c$ 。 |
|----------------------------------------------------------------------------------------------|

VaR 满足部分公理, 但不必然满足次可加性。

- |                                            |
|--------------------------------------------|
| 例 11.8 (VaR 的非次可加性反例): 设 $X, Y$ 相互独立同分布, 且 |
|--------------------------------------------|

$$\Pr(X = -100) = 0.04, \Pr(X = 0) = 0.96.$$

则 5% VaR 对单资产均为 0。但

$$X + Y = \begin{cases} -200, & 0.0016 \\ -100, & 0.0768 \\ 0, & 0.9216 \end{cases} \Rightarrow \text{VaR}_{5\%}(X + Y) = 100,$$

从而  $\text{VaR}_{5\%}(X + Y) > \text{VaR}_{5\%}(X) + \text{VaR}_{5\%}(Y)$ 。

另一方面, Ibragimov (2009) 证明: 在包括  $\alpha$ -稳定分布 ( $\alpha > 1$ ) 等广泛分布类中, VaR 满足次可加性。例如若  $X_j \sim N(\mu_j, \sigma_j^2)$  独立, 则

$$\text{VaR}_\alpha(X_1 + X_2) = \mu_1 + \mu_2 + \sqrt{\sigma_1^2 + \sigma_2^2} z_\alpha \leq \mu_1 + \sigma_1 z_\alpha + \mu_2 + \sigma_2 z_\alpha = \text{VaR}_\alpha(X_1) + \text{VaR}_\alpha(X_2).$$

当稳定指数  $\alpha < 1$  时, VaR 甚至超可加 (多样化有害); 见 Danielsson et al. (2013) 在正则变化条件下的相关结果。

## 11.8 期望损失 (ES)

**定义 11.12 (ES 的定义):** 记  $q_\alpha$  为  $X_t$  的  $\alpha$  分位数, 则

$$ES_\alpha(X_t) = -E(X_t | X_t \leq q_\alpha) = -\frac{E[X_t \mathbf{1}_{\{X_t \leq q_\alpha\}}]}{\Pr(X_t \leq q_\alpha)} = -\frac{1}{\alpha} E[X_t \mathbf{1}_{\{X_t \leq q_\alpha\}}].$$

若随机变量  $X$  连续,

$$ES_\alpha(X) = \frac{1}{\alpha} \int_0^\alpha \text{VaR}_\tau(X) d\tau.$$

**定理 11.4 (ES 的次可加性):** 令  $Z = X + Y$ , 则

$$\alpha[ES_\alpha(X) + ES_\alpha(Y) - ES_\alpha(Z)] = E[Z \mathbf{1}_{\{Z \leq q_\alpha(Z)\}} - X \mathbf{1}_{\{X \leq q_\alpha(X)\}} - Y \mathbf{1}_{\{Y \leq q_\alpha(Y)\}}] \geq 0,$$

其中用到

$$\mathbf{1}_{\{Z \leq q_\alpha(Z)\}} - \mathbf{1}_{\{X \leq q_\alpha(X)\}} \begin{cases} \geq 0, & X \geq q_\alpha(X), \\ \leq 0, & X \leq q_\alpha(X). \end{cases}$$

故  $ES_\alpha$  满足次可加性。

ES 的估计可用

$$\widehat{ES}_\alpha = -\frac{\sum_{t=1}^T X_t \mathbf{1}_{\{X_t \leq \widehat{q}_\alpha\}}}{\sum_{t=1}^T \mathbf{1}_{\{X_t \leq \widehat{q}_\alpha\}}}, \quad \widetilde{ES}_\alpha = -\frac{1}{\alpha} \sum_{t=1}^T X_t \mathbf{1}_{\{X_t \leq \widehat{q}_\alpha\}}.$$

其中  $\widehat{q}_\alpha$  为分位数/VaR 的估计值。两者在  $T \rightarrow \infty$  时具有相合性且渐近正态 (Scaillet 2004, 2005)。在 GARCH(1, 1) 等条件异方差模型中,

$$ES_\alpha(X_t | \mathcal{F}_{t-1}) = \mu + \sigma_t \cdot ES_\alpha(\varepsilon),$$

其中  $ES_\alpha(\varepsilon)$  为创新项的 ES，随条件波动率  $\sigma_t$  同步起伏。

### 11.8.1 期望损失 (ES): 历史模拟法与条件 GARCH 模型的实证计算

本小节给出一段可直接运行的 R 代码，对标普 500 指数 (^GSPC) 的日对数收益率计算与展示期望损失 (ES)。代码覆盖两条主线：其一是基于样本分布的无条件度量（历史模拟法），其二是基于时间序列模型的条件度量（GARCH 模型）。具体而言：

- **历史模拟法（无条件）**: 按定义  $ES_\alpha(X) = -\alpha^{-1}E[X \mathbf{1}_{\{X \leq q_\alpha\}}] = -E[X | X \leq q_\alpha]$ ，分别给出全样本 ES 与滚动窗口（如 250 日）的一步 VaR/ES 路径，便于观察极端损失的时间变化。
- **条件 GARCH（时变）**: 设  $X_t = \mu_t + \sigma_t \varepsilon_t$ ，在正态与学生  $t$  创新分布下，利用闭式条件尾期望  $m_\alpha := E(\varepsilon_t | \varepsilon_t \leq z_\alpha)$  计算  $ES_{t,\alpha} = -(\mu_t + \sigma_t m_\alpha)$ 。其中正态情形  $m_\alpha = -\phi(z_\alpha)/\alpha$ ；学生  $t_\nu$  情形可由标准化  $t$  密度/分位函数给出相应闭式。

实作细节与读图说明：

1. 代码输出“正的损失阈值”形式的 VaR/ES（与风险管理习惯一致）；作图时将其取负，与收益序列同轴呈现，便于对比“收益跌破阈值”的事件。
2. 条件 ES 的构造流程为：先用 GARCH(1,1) 模型估计  $\mu_t, \sigma_t$ ，再代入所选创新分布的  $m_\alpha$  得到  $ES_{t,\alpha}$ 。
3. 若需回测 ES 的校准性，可参照 ? 的 ES 回测统计量实现；本节代码留作扩展练习。

```

1 # 0) 环境准备 ----
2 req <- c("quantmod", "xts", "zoo", "rugarch")
3 inst <- setdiff(req, rownames(installed.packages()))
4 if (length(inst)) install.packages(inst, repos = "https://cloud.r-project.org")
5 suppressPackageStartupMessages(invisible(lapply(req, library, character.only = TRUE)))
6 set.seed(1)
7
8 ticker <- "^GSPC"
9 start_date <- "1990-01-01"
10 alpha_set <- c(0.01, 0.05) # 1% 与 5%
11 roll_win <- 250 # 滚动窗口 (约一年交易日)
12
13 # 1) 数据: 价格 -> 日对数收益 ----
14 x <- suppressWarnings(getSymbols(ticker, src="yahoo", from=start_date, auto.assign=FALSE))
15 px <- Ad(x)
16 ret <- na.omit(diff(log(px))) # 日对数收益 (可以为正/负)
17 colnames(ret) <- "r"
18 stopifnot(NROW(ret) > roll_win + 20)
19
20 r_num <- as.numeric(ret)
21
22 # 2) 无条件 ES (样本整体 + 滚动窗口) ----
23 # 定义: ES_alpha(X) = - E[X * 1{X <= q_alpha}] / alpha

```

```

24 es_empirical <- function(x, alpha){
25 q <- unname(quantile(x, probs=alpha, type=7, na.rm=TRUE))
26 # 两个等价写法 (总体与样本比例差异忽略时相同)
27 es1 <- - mean(x[x <= q]) # - E[X | X <= q]
28 es2 <- - (mean(x * (x <= q)) / alpha) # - (1/alpha) E[X 1{...}]
29 c(VaR = -q, ES_via_cond_mean = es1, ES_via_tail_avg = es2)
30 }
31
32 # 样本整体 (全期)
33 cat("== 无条件 ES: 整体样本 ==\n")
34 for (a in alpha_set){
35 out <- es_empirical(r_num, a)
36 cat(sprintf("alpha=% .2f -> VaR(loss)=%.4f, ES1=%.4f, ES2=%.4f\n",
37 a, out["VaR"], out["ES_via_cond_mean"], out["ES_via_tail_avg"]
38)))
39 }
40 # 滚动窗口 1 日期 VaR/ES 路径 (历史法)
41 roll_es <- function(x, alpha, win){
42 n <- length(x); res <- matrix(NA_real_, n, 3)
43 colnames(res) <- c("VaR", "ES1", "ES2")
44 for (t in seq_len(n)){
45 if (t < win) next
46 w <- x[(t-win+1):t]
47 tmp <- es_empirical(w, alpha)
48 res[t,] <- tmp
49 }
50 xts(res, order.by=index(ret))
51 }
52
53 roll_res_list <- lapply(alpha_set, function(a) roll_es(r_num, a, roll_win))
54 names(roll_res_list) <- paste0("alpha_", alpha_set)
55
56 # 3) 条件 ES: GARCH(1,1) ----
57 # 规格: X_t = mu + sigma_t * eps_t
58 # 正态: m_alpha := E[eps | eps <= z_alpha] = - dnorm(z_alpha) / alpha
59 # t (对称, 不偏) : 使用 rugarch 的 "std", 通过 qdist/ddist 计算:
60 # 取 q = qdist("std", alpha, mu=0, sigma=1, skew=1, shape=nu)
61 # f = ddist("std", q, mu=0, sigma=1, skew=1, shape=nu)
62 # m_alpha = - ((nu + q^2)/(nu - 1)) * f / alpha (nu > 1)
63
64 # 拟合 GARCH(1,1) (正态 & 学生t)
65 spec_norm <- ugarchspec(
66 variance.model=list(model="sGARCH", garchOrder=c(1,1)),
67 mean.model =list(armaOrder=c(0,0), include.mean=TRUE),
68 distribution.model="norm"
69)
70 fit_norm <- ugarchfit(spec_norm, ret, solver="hybrid")
71
72 spec_t <- ugarchspec(
73 variance.model=list(model="sGARCH", garchOrder=c(1,1)),

```

```

74 mean.model =list(armaOrder=c(0,0), include.mean=TRUE),
75 distribution.model="std"
76)
77 fit_t <- ugarchfit(spec_t, ret, solver="hybrid")
78
79 # 提取条件均值、波动 & t 的自由度
80 mu_n <- as.numeric(fitted(fit_norm))
81 sig_n <- as.numeric(sigma(fit_norm))
82
83 mu_t <- as.numeric(fitted(fit_t))
84 sig_t <- as.numeric(sigma(fit_t))
85 coef_t <- coef(fit_t)
86 nu <- as.numeric(coef_t[grep("shape", names(coef_t), ignore.case=TRUE)])
87 if (!is.finite(nu)) stop("未能从 GARCH-t 中识别自由度参数 shape/df。")
88
89 # 辅助：给定 alpha，返回 m_alpha (创新在左尾的条件期望，注意是负数)
90 m_alpha_norm <- function(alpha){
91 z <- qnorm(alpha)
92 - dnorm(z) / alpha
93 }
94 m_alpha_std <- function(alpha, nu){
95 q <- qdist("std", alpha, mu=0, sigma=1, skew=1, shape=nu)
96 f <- ddist("std", q, mu=0, sigma=1, skew=1, shape=nu)
97 - ((nu + q^2)/(nu - 1)) * f / alpha
98 }
99
100 # 生成条件 VaR/ES 路径 ("正的损失阈值")
101 cond_paths <- lapply(alpha_set, function(a){
102 # 正态
103 z <- qnorm(a)
104 mn <- m_alpha_norm(a)
105 VaR_loss_norm <- -(mu_n + sig_n * z)
106 ES_loss_norm <- -(mu_n + sig_n * mn)
107 # t
108 qt_ <- qdist("std", a, mu=0, sigma=1, skew=1, shape=nu)
109 mt_ <- m_alpha_std(a, nu)
110 VaR_loss_t <- -(mu_t + sig_t * qt_)
111 ES_loss_t <- -(mu_t + sig_t * mt_)
112 xts(cbind(VaR_norm = VaR_loss_norm,
113 ES_norm = ES_loss_norm,
114 VaR_t = VaR_loss_t,
115 ES_t = ES_loss_t),
116 order.by = index(ret))
117 })
118 names(cond_paths) <- paste0("alpha_", alpha_set)
119
120 # 4) 示例输出 ----
121 cat("\n== 条件 ES (GARCH) 示例：最近一个交易日 ==\n")
122 last_row <- function(X) as.numeric(tail(X, 1))
123 for (nm in names(cond_paths)){
124 a <- sub("alpha_", "", nm)

```

```

125 v <- cond_paths[[nm]]
126 vals <- round(last_row(v), 6)
127 names(vals) <- colnames(v)
128 cat(sprintf("alpha=%s -> %s\n", a,
129 paste(sprintf("%s=%g", names(vals), vals), collapse=", ")))
130 }
131
132 cat("\n== 滚动历史法 ES 示例: 最近一个交易日 ==\n")
133 for (nm in names(roll_res_list)){
134 a <- sub("alpha_", "", nm)
135 v <- roll_res_list[[nm]]
136 vals <- round(as.numeric(tail(v, 1)), 6)
137 names(vals) <- colnames(v)
138 cat(sprintf("alpha=%s -> %s\n", a,
139 paste(sprintf("%s=%g", names(vals), vals), collapse=", ")))
140 }
141
142 # 5) 简单可视化: 最近一年 (以 alpha=1% 为例) ----
143 a_plot <- "alpha_0.01"
144 if (a_plot %in% names(cond_paths)){
145 series_c <- cond_paths[[a_plot]]
146 series_h <- roll_res_list[[a_plot]]
147 one_year <- 252
148 r_tail <- tail(ret, one_year)
149 c_tail <- tail(series_c, one_year)
150 h_tail <- tail(series_h, one_year)
151
152 op <- par(no.readonly=TRUE); on.exit(par(op), add=TRUE)
153 par(mfrow=c(1,2), mar=c(4,4,2,1))
154
155 # 历史法: VaR/ES (将“正的损失阈值”取负画在收益轴上)
156 plot(index(r_tail), as.numeric(r_tail), type="h",
157 main="历史法 (滚动窗口) VaR/ES (alpha=1%) ",
158 xlab="", ylab="日对数收益", col="grey40")
159 lines(index(h_tail), -h_tail$VaR, lwd=2)
160 lines(index(h_tail), -h_tail$ES1, lwd=2, lty=2)
161 legend("bottomleft", c("收益", "-VaR (loss)", "-ES (loss)"),
162 lty=c(1,1,2), lwd=c(1,2,2), col=c("grey40", "black", "black"), bty="n"
163)
164
165 # 条件 GARCH (t)
166 plot(index(r_tail), as.numeric(r_tail), type="h",
167 main="条件 GARCH-t VaR/ES (alpha=1%) ",
168 xlab="", ylab="日对数收益", col="grey40")
169 lines(index(c_tail), -c_tail$VaR_t, lwd=2)
170 lines(index(c_tail), -c_tail$ES_t, lwd=2, lty=2)
171 legend("bottomleft", c("收益", "-VaR_t (loss)", "-ES_t (loss)"),
172 lty=c(1,1,2), lwd=c(1,2,2), col=c("grey40", "black", "black"), bty="n"
173)
174
175 par(mfrow=c(1,1))

```

174 }

175

176

## 11.9 黑天鹅、龙王与灰犀牛

**定义 11.13 (黑天鹅 (Taleb 2007)):** 黑天鹅指先验难以预见、影响巨大且事后被合理化的低概率事件。其要点：一是超出常用模型与经验外推的“认知支集”（模型外部性）；二是可计算性受限——样本与机制都不足以支持可靠的先验概率；三是人类倾向于在事后构造因果叙事（“看后视镜开车”）。典型治理思路：承认不可知，做鲁棒化与反脆弱设计（资本缓冲、限仓、熔断、尾部对冲等）。

**定义 11.14 (龙王 (Sornette 2012)):** 龙王指在统计上远超幂律尾部、且由不同于常态的内生机制（自激式正反馈、临界转变、网络连锁）所驱动的特异极端事件。与黑天鹅不同，龙王可能可预警：在危机前往往出现临界减速（自相关上升、恢复变慢）、方差/偏度上升、加速增长或对数周期等信号。典型治理思路：监测早期预警信号，针对性削弱正反馈与耦合（降杠杆、限集中度、隔离关键节点）。

**定义 11.15 (灰犀牛 (Wucker 2016)):** 指大概率、影响大、肉眼可见却被系统性忽视或拖延的风险（债务滚雪球、期限错配、气候与流动性错配等）。它并非“不可知”，而是“看见但不行动”导致的治理失败。典型治理思路：把显性风险纳入硬约束与考核（逆周期缓冲、压力测试情景、红线指标与问责机制），避免温水煮青蛙。

气候风险被归入“灰犀牛事件”，是因为气候相关风险符合其三要素：概率不低、影响巨大、且长期可见却常被拖延应对。这里讨论的不是“气候变化是否发生”的科学判断，而是由此引致的金融与宏观风险：一是物理风险（极端天气增多、海平面上升、慢性干旱、热浪等），会以较高概率、可预期地冲击资产与产能（如保险赔付激增、沿海不动产估值下调、供应链中断、粮价波动）；二是转型风险（从高碳向低碳的政策、技术、偏好切换），会触发资产重估与搁浅资产（高排放行业融资成本上升、碳价上行、合规成本抬升、绿色替代冲击旧产能）。这些并非难以预测的小概率意外，而是有时间表和指标的变化（如减排路径、披露规则、碳市场建设等）。从监管与风控视角看，若不及早开展情景分析、敞口识别与资本/流动性缓冲，相关冲击在集中爆发时往往被放大为系统性问题（如再保险保费跳升、抵押品折价、产业链信用收缩等）。

表 11.1: 黑天鹅、龙王与灰犀牛：对比要点

| 维度     | 黑天鹅             | 龙王             | 灰犀牛            |
|--------|-----------------|----------------|----------------|
| 先验可计算性 | 极弱 / 不可得        | 中等：依赖机制与预警信号   | 较强：风险公开且量化可得   |
| 可预警性   | 低（避免过度自信）       | 中高（临界征兆可监测）    | 高（但常被忽视 / 博弈掉） |
| 驱动机制   | 模型外部性 / 信息缺失    | 非线性正反馈、临界转变    | 累积性、显性失衡       |
| 治理策略   | 提升鲁棒性 / 缓冲、尾部对冲 | 监测信号、降耦合 / 降杠杆 | 硬约束与执行、逆周期管理   |

“黑天鹅” (Taleb 2007) 指那些出人意料、影响巨大、事后易被合理化的罕见事件：它强调 (i) 历史、科技、金融中罕见事件的非对称影响；(ii) 由于小概率的本质，可计算性受限；(iii) 人类对不确定性的系统性偏见。相关隐喻还包括“龙王”（规模与成因皆异于常态的极端事件）与“灰犀牛”（显而易见却被忽视的风险）。

2008 年金融危机可作为一个熟悉的范例：在危机前的“灰犀牛”阶段，房地产泡沫、影子银行链条、高杠杆证券化与短融长投等结构性失衡长期累积，相关脆弱性可量化、可观测却被忽视；当房价拐头、违约率上升并引发抵押品价值下跌、回购融资收缩时，“去杠杆—抛售—价格下跌—进一步去杠杆”的流动性螺旋迅速形成，并通过担保链与同质化头寸在机构间蔓延，系统进入“明斯基时刻”的非线性爆发阶段。危机爆发后，诸如评级下调、单笔违约、操作失误等原本局部且偶发的事件，因处于系统脆弱状态而被显著放大，表面上看像“黑天鹅事件”遍地出现；事后叙事若将整体危机简单归为“不可预见”，就会遮蔽此前多年显性失衡的累积与可治理空间。

以上证综指为例，2007–2020 年间多次出现“极端日”。其中：2007 年 2 月 27 日收盘下跌 8.84%<sup>1</sup>；2015 年 7 月 27 日下跌 8.48%<sup>2</sup>，2015 年 8 月 24 日下跌 8.49%（“黑色星期一”）<sup>3</sup>；2020 年 2 月 3 日（春节后复市首日）下跌约 7.7%<sup>4</sup>。虽然 2016–2019 年间此类单日大跌相对稀少，但这段“平静”容易让人误以为风险已消退；实际上，尾部事件并未消失，只是呈现低频、间歇的特征。政策与实务层面，估算 VaR/ES 时宜显式考虑厚尾与尾部依赖，并配合压力情景，更加关注尾端风险。

## 11.10 本章小结

本章围绕“尾部风险如何识别、如何量化、如何落地”展开，先说明了仅依赖均值一方差与正态近似的局限：金融收益常见尖峰厚尾、波动聚集与时变依赖，持有期风险也并非总能用“平方根法则”缩放。随后以 VaR 作为共同语言，串起三条主线：用正态与稳定分布理解时间缩放与厚尾现象；用非参数分位数稳健刻画中等尾部；以及在极端尾部引入极值理论与半参数尾模型，解决“样本之外”的外推问题。

在尾部建模方面，章节系统介绍了块极大 (GEV) 与高阈超额 (POT/GPD) 两条路径，并给出“正则变化尾”的实务做法：通过 Hill / DEM 等估计尾指数，配合 Hill 曲线、CCDF 双对数与平均超额图选择阈值并检验稳健性，再据此外推极端分位数。时间维度上，引入条件风险度量：使用 GARCH 模型刻画时变波动，结合经验或分布假设得到条件 VaR/ES；同时以 CaViaR 模型将 VaR 本身作为状态变量直接建模。评估环节强调“建模—回测—再校准”的闭环，通过无条件覆盖与独立性检验等方法检视度量角度的有效性。

在多资产情境下，本章阐明了 Copula 的“边际—依赖”解耦思想，指出线性相关无法准确刻画尾部依赖；据此给出 CoVaR 与  $\Delta$ CoVaR 的计算方法，用于度量系统性风险溢出与“同跌”风险。度量选型方面，对比了 VaR 与 ES：VaR 便于沟通但可能不满足次可加性，ES 为一致性风险度量，更适合监管与资本计提；无论采用何种度量方式，都应配套进行回测检验、模型漂移监控与压力测试。

本章最后提供了可复现的脚手架：滚动比较历史模拟法 / 高斯分布 / GARCH- $t$  / EVT-POT 的 VaR；半参数尾部厚度的诊断与外推；条件 VaR/ES 与 CaViaR 的轨迹与回测。并在“黑天鹅—灰犀牛”的框架下给出治理要点：承认不可知、监测临界信号、对显性失衡设硬约束，将度量、回测、压力测试与披露整合为“可检验、可实施、可沟通”的尾部风险管理闭环。

<sup>1</sup><https://finance.sina.cn/sa/2007-02-27/detail-ikftssap0992591.d.html?from=wap>

<sup>2</sup><https://finance.china.com/stock/pmx/20150727/3254296.shtml>

<sup>3</sup><https://www.cncfin.com/stock-xh08/a/20150824/1543213.shtml>

<sup>4</sup><https://www.xinhuanet.com/fortune/caiyuan/ksh/298.htm>

## 11.11 习题

### 1. VaR 的时间尺度与尾指数

(a) 设日收益  $X_D \sim N(\mu_D, \sigma_D^2)$  且独立同分布。证明  $T$  日持有期的 VaR 满足

$$\text{VaR}_\alpha(T) = T\mu_D + T^{1/2}\sigma_D z_\alpha,$$

并在  $\mu_D \approx 0$  时推出“平方根法则”  $\text{VaR}_\alpha(T) \approx \sqrt{T} \text{VaR}_\alpha(1)$ 。

(b) 若  $X_t$  服从稳定分布族, 特征指数为  $\theta \in (0, 2]$ , 说明并推导

$$\text{VaR}_\alpha(T) = T^{1/\theta} \text{VaR}_\alpha(1),$$

并比较  $\theta = 2$  与  $\theta < 2$  时 VaR 的随期限放大速度差异。

(c) 思考: 当  $\theta < 2$  而你仍使用  $\sqrt{T}$  缩放, 会对远期 VaR 产生怎样的偏误 (方向与直觉解释)?

### 2. von Mises 条件与 Pareto 尾部

(a) 对 Pareto 分布  $1 - F(x) = L^\kappa x^{-\kappa}$  ( $x \geq L > 0$ ), 证明

$$\lim_{x \rightarrow \infty} \frac{x f(x)}{1 - F(x)} = \kappa,$$

从而满足冯·米塞斯 (von Mises) 条件。

(b) 举一个不满足 von Mises 条件的轻尾分布 (如指数分布或对数正态分布), 并说明不满足的原因。

(c) 思考: von Mises 条件为何能保证 Hill 估计量的渐近正态性? 用“尾部风险率  $h(x) = f(x)/(1 - F(x))$  的一阶正则性”直观解释。

### 3. Hill / DEM 估计与阈值选择

(a) 设样本  $\{X_t\}_{t=1}^T$  按降序  $X_{(1)} \geq \dots \geq X_{(T)}$  排序。推导 Hill 估计

$$\hat{\kappa} = \frac{1}{M} \sum_{j=1}^M \log \frac{X_{(j)}}{X_{(M+1)}}.$$

(b) 参照 DEM 估计的定义, 写出  $\hat{\gamma}_{\text{DEM}}$  的显式表达式, 并说明其与 Hill 估计量的关系。

(c) 实践: 用你选择的股票 / 指数目收益率数据, 取若干  $M$  值绘制 Hill 曲线, 并讨论“平台段”的选择标准。

### 4. 极值理论: 块极大与 POT

(a) 设  $M_T = \max_{1 \leq t \leq T} X_t$ , 在适当条件下存在常数  $a_T > 0$ ,  $b_T$  使  $a_T(M_T - b_T) \Rightarrow G_\gamma$ 。写出  $G_\gamma$  的统一形式并指出  $\gamma > 0, = 0, < 0$  的三个子类。

- (b) **POT 模型:** 若超额  $Y = X - u \mid X > u$  近似服从 GPD, 写出 GPD 的累积分布函数 (CDF), 并给出阈上尾分位数 (无条件) 的闭式解:

$$\text{VaR}_p = u + \frac{\beta}{\xi} \left[ \left( \frac{\lambda}{1-p} \right)^\xi - 1 \right], \quad (\xi \neq 0),$$

并解释  $\lambda = \Pr(X > u)$  的估计方法。

- (c) **实践:** 用 95% 阈值拟合 GPD, 报告  $\hat{\xi}, \hat{\beta}$ , 并据此外推 99% VaR; 与经验分位数比较差异。

## 5. GARCH 条件 VaR/ES 与回测检验

- (a) 设  $X_t = \mu_t + \sigma_t \varepsilon_t$ ,  $\varepsilon_t \sim N(0, 1)$ 。写出条件 VaR 与 ES 的闭式表达式:

$$\text{VaR}_{t,\alpha} = -(\mu_t + \sigma_t z_\alpha), \quad \text{ES}_{t,\alpha} = -(\mu_t + \sigma_t m_\alpha), \quad m_\alpha = -\phi(z_\alpha)/\alpha.$$

- (b) 若  $\varepsilon_t \sim t_\nu$  (对称分布), 给出  $m_\alpha = E(\varepsilon_t \mid \varepsilon_t \leq q_\alpha)$  的闭式表达式 (可用密度函数与分位数函数表述)。

- (c) **回测:** 推导 Kupiec's UC 统计量

$$LR_{uc} = 2 \left[ (n-x) \ln \frac{1-\hat{p}}{1-\alpha} + x \ln \frac{\hat{p}}{\alpha} \right],$$

并说明其在  $H_0: p = \alpha$  下的极限分布; 补充一阶马尔可夫链的独立性检验思路。

## 6. Copula 与 CoVaR

- (a) 给定边际分布函数  $F_X, F_Y$  与 Copula  $C(u, v)$ , 写出联合密度函数

$$f_{X,Y}(x, y) = c(F_X(x), F_Y(y)) f_X(x) f_Y(y), \quad c(u, v) = \frac{\partial^2}{\partial u \partial v} C(u, v).$$

- (b) 在高斯 Copula 下 (相关系数  $\rho$ ), 解释为何上 / 下尾依赖系数为零; 与 BB7 (Joe-Clayton) Copula 的尾依赖进行对比。

- (c) **CoVaR:** 用 U-空间条件分布的逆映射  $H^{-1}$  形式, 表示

$$\text{CoVaR}_\alpha^{Y|X} = F_Y^{-1}(H^{-1}(\alpha \mid U_X = \alpha)),$$

并定义  $\Delta\text{CoVaR}$ 。

## 7. ES 的性质与估计

- (a) 证明 (或给出证明思路) Expected Shortfall (ES) 的次可加性:  $ES_\alpha(X+Y) \leq ES_\alpha(X) + ES_\alpha(Y)$ 。

- (b) 设  $\hat{q}_\alpha$  为样本分位数, 写出

$$\widehat{\text{ES}}_\alpha = -\frac{\sum_{t=1}^T X_t \mathbf{1}_{\{X_t \leq \hat{q}_\alpha\}}}{\sum_{t=1}^T \mathbf{1}_{\{X_t \leq \hat{q}_\alpha\}}}, \quad \widetilde{\text{ES}}_\alpha = -\frac{1}{\alpha} \sum_{t=1}^T X_t \mathbf{1}_{\{X_t \leq \hat{q}_\alpha\}},$$

并简述二者在  $T \rightarrow \infty$  下的一致性与渐近正态性要点。

(c) **实践：**在同一数据集上比较 VaR 与 ES 的突破事件特征曲线，讨论 ES 相较于 VaR 在稳健性和一致性方面的优势。

#### 8. 稳定分布与分位统计

- (a) 设  $X$  服从对称稳定分布  $S(\theta, \beta = 0, \gamma, \mu)$ 。说明当  $\theta \leq 1$  时为何均值可能不存在，但分位数与 VaR / ES 仍可定义。
- (b) **模拟：**生成不同  $\theta \in \{1.2, 1.5, 2.0\}$  的稳定分布样本（可用 R 的 `stabledist` 包或自定义函数），对比  $T$  日 VaR 的缩放关系： $T^{1/\theta}$  vs.  $\sqrt{T}$ 。

#### 9. 极端相关与尾部依赖（开放题）

- (a) 选取一个危机时期（如 2008 年、2020 年），估计两大类资产（股指 / 信用债或股指 / 大宗商品）在常态期与危机期的上下尾相关性（可采用非参数尾部相关性测度或 Copula 参数估计法）。
- (b) 讨论：尾依赖上升对“分散化有效性”的含义；给出一条面向风险控制的量化建议。

## 12 AI 与金融计量：模型、应用与实践

面向金融问题的计量建模，长期以来以 OLS、VAR/ARMA、GARCH、Logit/Probit 等 {传统计量} 为基石：它们结构清晰、检验体系成熟、推断逻辑完备，能够围绕参数的显著性、区间估计与假设检验形成“可复核、可审计”的证据链。然而，现代金融数据与业务场景呈现出三类新的张力：其一，{非线性与高维交互} 的普遍存在，使线性或准线性关系难以完整刻画；其二，{非平稳与制度切换}（如零利率下限（ZLB）、疫情冲击、监管与微观结构变化）导致参数不稳定、外推风险上升；其三，{数据模态多样化}（文本、图像、逐笔成交与层级账务等）要求模型具备跨模态的表征与融合能力。围绕这些痛点，树集成、核方法与深度学习等 {AI 方法} 提供了更强的函数逼近与表征学习能力，并且可以在稳健的交叉验证与正则化框架内，围绕“预测—解释—合规”重塑建模流程。

从目标函数看，预测型学习以**广义风险最小化**为准则：

$$R(f) = \mathbb{E}[\ell(Y, f(X))], \quad \hat{f} = \arg \min_{f \in \mathcal{F}} \widehat{R}(f),$$

强调在给定 {当前分布} 下的泛化误差；而**结构性 / 因果性**计量则关注数据生成机制

$$Y = g(X; \theta) + \varepsilon, \quad \mathbb{E}[\varepsilon | X] = 0,$$

以对  $\theta$  的可解释推断为核心。AI 方法可被视作对函数类  $\mathcal{F}$  的扩展：当 {预测} 是首要目标（择时、排序、违约检测）时，非线性模型在“弱信号—高噪声—高维协变量”下往往具备更强表达力；当 {解释} 与 {政策敏感度分析} 是首要目标（政策评估、规制口径）时，应将机器学习纳入正交化 / 双重稳健等框架（例如 Double Machine Learning），以保持推断有效性与因果可读性。

从时变环境看，金融系统很少满足“独立同分布”。协变量漂移与概念漂移可形式化为

$$p_t(X) \neq p_s(X) \quad \text{或} \quad p_t(Y | X) \neq p_s(Y | X),$$

它们对模型的影响分别体现为“外推偏差”与“策略失效”。因此，本书将**时间序列交叉验证**作为底层约束：滚动 / 扩展窗口、purged / embargo K-fold（清除与验证标签区间重叠的训练样本，并在边界设置“禁运区”），以及**外层一次性报告**的嵌套交叉验证，都是为了将调参与选择的“研究者自由度”与样本外评估严格分离，避免信息泄漏与回测过拟合。

从治理与合规角度看，金融领域的落地应用要求“可解释—可追溯—可监控”。AI 模型的“黑箱性”并非不可化解：全局 / 局部 SHAP、置换重要度、PDP / ALE 与 ICE 等工具可以回答“哪些特征重要、在哪些区间敏感、模型是否稳定”等问题；概率输出需通过 Platt / 保序回归进行 {校准}（参考 Brier / ECE 指标），并与阈值 / 成本对齐；漂移监控可使用 PSI / JS 散度或基于分类器的两样本检验，触发再训练或阈值重定标；公平性可按机会均等（TPR 一致）、均衡误差（TPR / FPR 同时一致）或人口比例等口径衡量，

并通过阈值分组 / 后处理进行调整。文档化（包括数据血缘、特征字典、切分与随机种子、阈值与版本管理）、三道防线（开发—独立验证—内部审计）与人工干预停机开关 / 兜底机制，是将“算法正确性”提升为“业务可用性”的制度性前提。

基于上述原则，{本章定位} 是：不把 AI 作为传统计量的替代品，而是作为 {补全者与增强器}。在方法论层面，我们在**树模型**（随机森林 / XGBoost）与**深度神经网络**（MLP / LSTM / Transformer）中展开，提供从超参数调优、时序交叉验证、早停与正则化，到概率校准、阈值-成本对齐、风险-收益评估的 {可复现管线}；在解释与治理层面，我们结合 SHAP / PDP / ALE、稳定性与漂移监控、现实检验 / SPA 与 FDR 控制，给出“好看亦能用”的样本外证据；在风险视角，我们以 VaR / ES 与场景分析串联因子抽取（PCA / AE / VAE）、暴露估计与不确定性量化（自助法区间 / 因子 MC），展示 AI 在风险聚合与金融稳定中的可操作一面。最终目标是把“强表达力的学习器”嵌入“因果正确的评估—合规完整的治理”闭环，在 {预测—解释—合规} 三角中找到可复用的均衡点。

## 12.1 树模型（随机森林与 XGBoost）的原理及适用性

### 12.1.1 决策树模型

**决策树**通过对特征空间进行递归的二叉划分，学习一组“如果……则……”的判别 / 预测规则。其核心思想是在某个特征的某一阈值上划分数据，使得划分后子集在目标变量上的“纯度”更高：分类问题常用熵或基尼指数衡量不纯度，回归问题常用残差平方和 (SSE) 或均方误差 (MSE) 作为分裂标准。

#### 12.1.1.1 分类树：基尼指数与信息增益

**定义 12.1 (基尼指数 (Gini Index))**：设样本集合  $D$  中共有  $K$  个类别，第  $k$  类的样本数量占总体的比例为  $p_k$ ，则

$$\text{Gini}(D) = 1 - \sum_{k=1}^K p_k^2 = \sum_{k=1}^K p_k(1 - p_k).$$

若按特征  $A$  的某个阈值（如  $A \leq a$  与  $A > a$ ）将  $D$  划分为  $D_1, D_2$ ，则加权基尼指数为

$$\text{Gini}_A(D) = \frac{|D_1|}{|D|} \text{Gini}(D_1) + \frac{|D_2|}{|D|} \text{Gini}(D_2).$$

训练时选择使  $\text{Gini}_A(D)$  最小的特征与阈值。

**定义 12.2 (熵与信息增益)**：数据集  $D$  的熵定义为

$$\text{Entropy}(D) = - \sum_{k=1}^K p_k \log_2 p_k,$$

其中约定  $0 \log 0 := 0$ ；对数底数不影响特征选择的相对次序。按特征  $A$  将  $D$  划分为若干子集  $D_i$  ( $i = 1, \dots, n$ )，信息增益定义为

$$\text{Gain}(D, A) = \text{Entropy}(D) - \sum_{i=1}^n \frac{|D_i|}{|D|} \text{Entropy}(D_i).$$

**定义 12.3 (信息增益率 (避免对多值特征的偏好) ):**

$$\text{GainRatio}(D, A) = \frac{\text{Gain}(D, A)}{\text{IV}(A)}, \quad \text{IV}(A) = -\sum_{i=1}^n \frac{|D_i|}{|D|} \log_2 \frac{|D_i|}{|D|}.$$

### 12.1.1.2 回归树：最小化残差平方和 / 最大化方差减小

设按特征  $A$  在阈值  $a$  处分割将  $D$  分为  $D_1 = \{x \mid A(x) \leq a\}$  与  $D_2 = \{x \mid A(x) > a\}$ ，对应损失为

$$\text{SSE}(A, a) = \sum_{x_i \in D_1} (y_i - \bar{y}_1)^2 + \sum_{x_i \in D_2} (y_i - \bar{y}_2)^2,$$

其中  $\bar{y}_1, \bar{y}_2$  分别为两个子集中  $y$  的均值。最小化 SSE 等价于 {最大化方差减小}（与分类中最大化纯度下降相对应）。在对异常值更敏感的场景中，也可采用绝对误差 (MAE) 或 Huber 损失。

### 12.1.1.3 统一的不纯度下降视角

记节点不纯度为  $I(D)$ ：分类时取 Gini 或 Entropy，回归时取方差或 SSE 的归一化形式。一次二分划分  $(A, a)$  的不纯度下降为

$$\Delta I(A, a \mid D) = I(D) - \frac{|D_1|}{|D|} I(D_1) - \frac{|D_2|}{|D|} I(D_2).$$

训练时等价于在候选阈值上贪心地选择使  $\Delta I$  最大的划分。对连续特征，通常在排序后的相邻取值中点处枚举候选阈值。

### 12.1.1.4 预测与规则形式

生成的树对应一组可解释的规则。分类叶节点输出多数类或类别概率；回归叶节点的预测值通常取叶内  $y$  的均值（有时也取中位数以增强鲁棒性）。示例：

1. 分类：若  $A \leq a$  且  $B > b$ ，则预测为类别  $C$ 。
2. 回归：若  $A(x) \leq a$  且  $B(x) > b$ ，则

$$\hat{y}(x) = \frac{1}{|D'|} \sum_{x_i \in D'} y_i, \quad D' = \{x_i \in D \mid A(x_i) \leq a, B(x_i) > b\}.$$

### 12.1.1.5 正则化与剪枝：超参数的含义与作用

实践中，决策树的复杂度既可通过 {预剪枝}（训练时限制生长）控制，也可通过 {后剪枝}（先长满再回退）控制。以下超参数是对同一思想的不同实现方式，独立于具体实现库。

**定义 12.4 (常见预剪枝超参数):** `max_depth` 最大树深度。限制根到叶的最长路径长度。值越小，模型越简单（偏差↑、方差↓）。常与 `min_samples_leaf` 联调以避免“深而窄”的枝条。

`min_samples_split` 继续分裂一个内部节点所需的最小样本数。若节点样本数小于此阈值则停止分裂。适当增大可抑制微小数据扰动导致的过拟合。

`min_samples_leaf` 叶节点允许的最小样本数。保证每个叶子“足够大”，降低方差，尤其在噪声较大或类别极不平衡时更稳健。

`max_leaf_nodes` 直接限制叶子总数（即决策树的分段数）。是对模型容量的硬约束，常与 `max_depth` 二选一使用。

`min_impurity_decrease` 最小“不纯度下降”阈值。记一次候选划分  $(A, a)$  的不纯度下降为：

$$\Delta I(A, a | D) = I(D) - \frac{|D_1|}{|D|} I(D_1) - \frac{|D_2|}{|D|} I(D_2),$$

其中  $I(\cdot)$  为节点不纯度（分类取 **Gini 指数 / 熵**，回归取方差或 SSE 的归一化形式）。仅当  $\Delta I(A, a | D) \geq \delta$  (`min_impurity_decrease = delta`) 时才允许分裂，从而过滤“收益很小”的分裂。

**定义 12.5 (后剪枝：代价复杂度剪枝与 `ccp_alpha`)：**先训练一棵较大的树  $T$ 。回归情形的经验风险定义为

$$R(T) = \sum_{t \in \text{leaves}(T)} \sum_{x_i \in t} (y_i - \bar{y}_t)^2,$$

并加入对叶子数量的惩罚项

$$R_\alpha(T) = R(T) + \alpha |T|, \quad \alpha \geq 0.$$

随着  $\alpha$  增大，可得到一条嵌套的“最优子树”序列。实际选择可结合交叉验证最小化验证误差，或采用“{一倍标准误差原则}”（在最小点附近选更简单的那棵）。实现中常将该惩罚系数以 `ccp_alpha` 形式暴露。

#### 12.1.1.6 如何使用与调参建议

- 先定容量，再微调阈值：**优先用 `max_depth/max_leaf_nodes/min_samples_leaf` 控制整体复杂度，再用 `min_impurity_decrease` 去除“边际收益小”的分裂。
- 时间序列交叉验证：**金融数据应按时间滚动 / 扩展窗口做验证，避免信息泄露；在此框架下选取最小化样本外误差且 {尽量简单} 的组合。
- 不平衡 / 噪声场景：**适当增大 `min_samples_leaf` 与 `min_samples_split` 可显著降低方差；必要时配合样本权重 / 代价敏感学习。
- 后剪枝备用：**当预剪枝难以事先把握时，可先放宽约束训练大树，再通过 `ccp_alpha` 后剪枝，并用交叉验证确定惩罚强度。

**定义 12.6 (代价复杂度剪枝 (Cost-Complexity Pruning))：**设一棵已生长的树为  $T$ ，叶节点数为  $|T|$ 。回归情形的经验风险为

$$R(T) = \sum_{t \in \text{leaves}(T)} \sum_{x_i \in t} (y_i - \bar{y}_t)^2,$$

其正则化目标为

$$R_\alpha(T) = R(T) + \alpha |T|, \quad \alpha \geq 0.$$

通过调节  $\alpha$  可得到一条最优子树序列，并结合交叉验证选择合适的  $\alpha$ 。

### 12.1.1.7 金融数据中的建模要点

- 时间序列评估与防泄露：**采用时间序列交叉验证（滚动 / 扩展窗口），严格保证训练集时间早于验证 / 测试集。设观测按时间顺序编号为  $\{1, \dots, T\}$ ；第  $m$  折使用训练集  $\{1, \dots, t_m\}$ 、验证集  $\{t_m + 1, \dots, t_{m+1}\}$ ，其中  $1 < t_1 < \dots < t_M < T$ ，最终指标取各折加权平均值。
- 指标体系：**除分类准确率 / 回归误差外，关注 {收益-风险} 指标（年化收益率、波动率、最大回撤、夏普比率 / 索提诺比率），并计入 {交易成本与滑点} 后的净表现。
- 类别不平衡 / 稀有事件：**如违约、欺诈、极端收益。采用加权损失函数、阈值优化方法，优先使用 PR-AUC、F1 分数、召回率等指标。
- 概率校准与可解释性：**对评分 / 概率输出进行 Platt 校准或保序回归 (Isotonic Regression)；使用 {置换重要度} 与基于树的 SHAP (TreeSHAP) 解释特征贡献。
- 非平稳与漂移：**监控分布 / 概念漂移，制定再训练与稳定性监控策略；必要时加入 {单调性约束} 以符合经济学常识（如利率上升  $\Rightarrow$  违约概率上升）。
- 缺失值与高基数类别：**缺失值可单独作为一类或使用 {替代划分 (surrogate splits)}；高基数类别可用目标编码 / 频数编码，并按时间顺序处理以避免信息泄露。

### 12.1.1.8 与集成学习的承接

单棵树方差较高，实际应用中常配合集成方法使用：

- 随机森林（装袋法）：**对样本与特征做随机子采样，训练多棵树并通过投票 / 平均以降低方差。关键超参数：树的数量、最大深度、特征子采样比例、样本子采样比例。
- 梯度提升树（GBDT / XGBoost / LightGBM）：**以深度受限的决策树为弱学习器，逐步拟合残差或负梯度，偏差小、精度高。关键超参数：学习率、弱学习器数量、最大深度、最小叶节点样本数、行 / 列采样率。

### 12.1.1.9 示例（量化 / 风控）

在量化选股中，树模型可学习“低利率且高估值  $\Rightarrow$  卖出，否则买入”等交互式规则；在信贷风控中，可结合 AUC / KS 指标与校准曲线评估模型性能，并配合阈值策略以满足业务约束。

**示例 A：量化选股（滚动回测 + 决策树）。**以下示例构造了一个“200 只股票  $\times$  120 个月”的模拟面板数据，特征包含利率（宏观）、市盈率（估值）、动量与波动率，并内嵌“低利率且高估值时收益更差”的交互式真值生成机制。代码采用 {时间序列滚动回测}（训练窗口 60 个月）训练决策树模型，逐月样本外预测“下月上涨概率”，在每个横截面上计算 AUC，并按概率从高到低构建“做多前 20%、做空后 20%”的多空组合（扣除双边交易成本），输出累计净值曲线以及年化收益率、年化波动、夏普比率等绩效指标，同时展示最新一棵树的结

构以便解释规则。该示例演示了树模型如何在不平稳的时间序列环境中进行 {样本外评估、信号转化与组合构建} 的全流程；实务中可用真实因子 / 行情数据替换模拟数据，并根据交易制度与成本校准窗口长度、分位数阈值与风险约束。

```

1 # =====
2 # 量化选股：树模型 + 时间序列滚动回测 + 多空组合
3 # =====
4 # 依赖包（如未安装请先 install.packages(c("rpart","rpart.plot","pROC",
5 "ggplot2","dplyr","tidyR"))）
6 library(dplyr); library(tidyR); library(ggplot2)
7 library(rpart); library(rpart.plot); library(pROC)
8
9 set.seed(123)
10
11 # 1) 模拟面板数据（200只股票 x 120个月），含宏观利率、估值、动量、波动等特征
12 n_stocks <- 200; n_months <- 120
13 dates <- seq(as.Date("2010-01-31"), by="month", length.out=n_months)
14
15 # 宏观“利率”（标准化的 AR(1) 序列）
16 rate_series <- as.numeric(stats::arima.sim(list(ar=0.8), n=n_months, sd=0.2))
17 rate_series <- scale(rate_series)[,1]
18
19 panel <- expand.grid(id = sprintf("S%03d", 1:n_stocks), date = dates) |>
20 as_tibble() |>
21 mutate(rate = rate_series[match(date, dates)],
22 pe = pmax(rnorm(n(), mean=15, sd=5), 1), # 估值：越高越贵
23 mom = rnorm(n()), # 动量
24 vol = exp(rnorm(n(), sd=0.3))) # 波动（对数正态）
25
26 # 构造“真实”信号并生成下一期收益（含交互：低利率 & 高估值→偏负）
27 panel <- panel |>
28 mutate(signal = -0.02*pe + 0.5*(-rate) + 0.3*mom - 0.1*vol + ifelse(rate
29 <0 & pe>18, -0.2, 0),
30 ret_fwd = 0.02*signal + rnorm(n(), sd=0.05),
31 y = factor(ifelse(ret_fwd>0, "Up", "Down")))
32
33 # 2) 时间序列滚动回测（训练窗=60个月，逐月样本外预测）
34 uniq_dates <- sort(unique(panel$date))
35 train_win <- 60
36 q <- 0.2 # 多 top 20%，空 bottom 20%
37 tc <- 0.0005 # 单边交易成本（示例设为 5bp=0.0005，可按需调整）
38
39 oos_list <- list()
40 for (t in (train_win+1):length(uniq_dates)) {
41 trng <- uniq_dates[(t-train_win):(t-1)]
42 test <- uniq_dates[t]
43
44 train_df <- filter(panel, date %in% trng)
45 test_df <- filter(panel, date == test)
46
47 fit <- rpart(

```

```

46 y ~ pe + rate + mom + vol,
47 data=train_df, method="class",
48 control = rpart.control(cp=0.0, maxdepth=4, minsplit=200, minbucket=50)
49)
50
51 test_df$prob <- predict(fit, newdata=test_df, type="prob")[, "Up"]
52
53 # 截面AUC (每月)
54 roc_obj <- pROC::roc(response=test_df$y, predictor=test_df$prob, quiet=
 TRUE)
55 auc_val <- as.numeric(pROC::auc(roc_obj))
56
57 # 多空组合: 多 top-q, 空 bottom-q, 净回报扣除双边交易成本
58 thr_long <- quantile(test_df$prob, 1-q, na.rm=TRUE)
59 thr_short <- quantile(test_df$prob, q, na.rm=TRUE)
60 long_ret <- mean(test_df$ret_fwd[test_df$prob>=thr_long], na.rm=TRUE)
61 short_ret <- mean(test_df$ret_fwd[test_df$prob<=thr_short], na.rm=TRUE)
62 ls_ret <- (long_ret - short_ret) - 2*tc
63
64 oos_list[[length(oos_list)+1]] <- tibble(date=test, auc=auc_val, long=long_
 _ret, short=short_ret, ls=ls_ret)
65 }
66 oos <- bind_rows(oos_list) |> arrange(date) |> mutate(nav = cumprod(1+ls))
67
68 # 3) 绩效汇总 (年化)
69 ann_factor <- 12
70 ann_ret <- prod(1 + oos$ls)^{(ann_factor/nrow(oos))} - 1
71 ann_vol <- sd(oos$ls, na.rm=TRUE) * sqrt(ann_factor)
72 sharpe <- mean(oos$ls, na.rm=TRUE) / sd(oos$ls, na.rm=TRUE) * sqrt(ann_
 factor)
73 print(round(c(AnnReturn=ann_ret, AnnVol=ann_vol, Sharpe=sharpe), 3))
74
75 # 4) 可视化: 累计净值与月度AUC
76 ggplot(oos, aes(date, nav)) + geom_line() +
 labs(title="多空策略累计净值 (含交易成本)", y="净值", x=NULL)
77
78 ggplot(oos, aes(date, auc)) + geom_line() + geom_hline(yintercept=0.5,
 linetype=2) +
 labs(title="截面AUC (每月)", y="AUC", x=NULL)
79
80
81
82 # (可选) 查看最新一棵树的结构
83 rpart.plot::rpart.plot(fit, type=2, extra=104, under=TRUE, fallen.leaves=
 TRUE)

```

**示例 B: 信贷风控 (AUC / KS / 校准曲线 + 阈值策略)**。本示例基于 ISLR::Default 数据集, 将“是否违约”设为二分类标签, 按 70% / 30% 划分训练集与测试集, 训练一棵 {预剪枝} 决策树并输出样本外违约概率。随后计算 AUC 与 KS 衡量 {区分度}; 按预测概率分位构建 {校准曲线} 检验概率可靠性; 基于 ROC 曲线选取 {Youden J} 最优阈值并给出混淆矩阵; 同时提供一套 {成本敏感阈值} 优化 (假设“好客户 +100、坏客户 -500”) 以最大化单笔期望利润, 进而得到相应的放款率与收益评估。该示例覆盖了风控常见的“区分

度—校准—阈值策略”链路；在真实业务中，建议先做概率校准（Platt / Isotonic），结合样本权重 / 代价敏感学习应对类不平衡，并在监管 / 业务约束下联动阈值策略与拒绝推断等治理环节。

```

1 # =====
2 # 信贷风控：树模型 + AUC/KS/校准曲线 + 阈值策略
3 # 使用 ISLR::Default 数据集（信用卡违约示例）
4 # =====
5 # 依赖包 install.packages(c("ISLR","rpart","pROC","ggplot2","dplyr","rpart.
6 plot")))
7 library(ISLR); library(dplyr); library(ggplot2)
8 library(rpart); library(rpart.plot); library(pROC)
9
10 data(Default)
11 df <- as_tibble(Default) |>
12 mutate(default = factor(default, levels=c("No","Yes"))) # 标签：是否违约
13
14 set.seed(42)
15 idx <- sample.int(nrow(df), size = 0.7*nrow(df))
16 train <- df[idx,]
17 test <- df[-idx,]
18
19 # 决策树（分类）
20 fit <- rpart(default ~ balance + income + student, data=train, method="class"
21 ,
22 control=rpart.control(cp=0.0, maxdepth=4, minsplit=100,
23 minbucket=50))
24 rpart.plot(fit, type=2, extra=104, under=TRUE)
25
26 # 样本外预测概率
27 test$prob <- predict(fit, newdata=test, type="prob")[, "Yes"]
28
29 # AUC
30 roc_obj <- pROC::roc(response=test$default, predictor=test$prob,
31 levels=c("No","Yes"), direction "<", quiet=TRUE)
32 auc_val <- as.numeric(pROC::auc(roc_obj)); print(auc_val)
33
34 # KS (= max(TPR - FPR) = Youden J)
35 roc_df <- tibble(tpr = roc_obj$sensitivities, fpr = 1-roc_obj$specificities)
36 ks_val <- max(roc_df$tpr - roc_df$fpr, na.rm=TRUE); print(ks_val)
37
38 # 阈值1：Youden J 最大（兼顾召回与特异度）
39 coords_best <- pROC::coords(roc_obj, "best", best.method="youden",
40 ret=c("threshold","sensitivity","specificity"))
41 thr1 <- as.numeric(coords_best["threshold"])
42
43 # 基于 thr1 的分类与混淆矩阵
44 test <- test |>
45 mutate(pred = factor(ifelse(prob >= thr1, "Yes","No"), levels=c("No","Yes"
46)))
47 print(table(Predicted=test$pred, Actual=test$default))
48

```

```

45 # 简易校准曲线（按十分位分箱）
46 cal <- test |>
47 mutate(bin = ntile(prob, 10)) |>
48 group_by(bin) |>
49 summarise(pred = mean(prob), obs = mean(default=="Yes"), .groups="drop")
50
51 ggplot(cal, aes(pred, obs)) + geom_point() + geom_line() +
52 geom_abline(slope=1, intercept=0, linetype=2) +
53 labs(title="概率校准曲线（十分位）", x="平均预测违约概率", y="实际违约率")
54
55 # 阈值2：成本敏感（示例：好客户利润 +100，坏客户损失 -500）
56 profit_good <- 100; loss_bad <- -500
57 grid <- tibble(thr = seq(0,1,by=0.01)) |>
58 mutate(exp_profit = sapply(thr, function(th){
59 approve <- test$prob < th # 通过阈值则放款
60 p <- test$prob[approve] # 违约概率近似校准后使用更稳健
61 if(length(p)==0) return(-Inf)
62 mean((1 - p)*profit_good + p*loss_bad) # 单笔期望利润
63 }))
64
65 thr2 <- grid$thr[which.max(grid$exp_profit)]
66 print(thr2)
67
68 # 使用 thr2 的放款率与期望利润
69 approve2 <- test$prob < thr2
70 cat("放款率: ", mean(approve2), "\n",
71 "单笔期望利润: ", mean((1-test$prob[approve2])*profit_good + test$prob[
72 approve2]*loss_bad), "\n")
73 # 备注：若用于利润驱动的阈值设定，请先做概率校准（如 Platt/Isotonic校准），以免偏差放大导致成本评估误差。

```

总之，决策树以“最大化不纯度下降”为统一目标，既能表达非线性与交互效应，又保持良好可解释性；通过合适的正则化与时间序列验证，并结合随机森林 / 梯度提升树等集成方法，可在金融任务中实现更稳健、贴近业务约束的建模与部署。

## 12.1.2 随机森林

### 12.1.2.1 构建机制与集成原理

为提升单棵决策树的样本外表现，**随机森林** (Random Forest (Breiman 2001)) 在训练阶段对“样本”和“特征”同时注入随机性。具体做法是：对包含  $n$  个样本的数据集  $D$ ，第  $b$  棵树使用有放回抽样 (bootstrap 采样) 得到训练子集

$$D_b = \{(x_i, y_i)\}_{i=1}^n, \quad x_i \sim D \text{ (bootstrap, 有放回)},$$

并在每个节点分裂时，从全部  $p$  个特征中随机抽取  $m$  个候选特征（常记作  $mtry$ ），只在这  $m$  个特征中寻找最佳划分。记节点  $t$  的候选集合为  $A_t$  且  $|A_t| = m$ ，则在  $A_t$  中选择使 {不

纯度下降} 最大的特征及其对应阈值:

$$j^* = \arg \max_{j \in A_t} \Delta I_t(j), \quad \Delta I_t(j) = I(D_t) - \left( \frac{|D_{t,L}|}{|D_t|} I(D_{t,L}) + \frac{|D_{t,R}|}{|D_t|} I(D_{t,R}) \right),$$

其中  $I(\cdot)$  为不纯度度量函数。为便于统一理解，分类任务常用

$$I_{\text{Gini}}(D) = 1 - \sum_{k=1}^K p_k^2, \quad I_{\text{Entropy}}(D) = - \sum_{k=1}^K p_k \log p_k,$$

回归任务常用

$$I_{\text{Var}}(D) = \frac{1}{|D|} \sum_{x_i \in D} (y_i - \bar{y}_D)^2,$$

也可用按样本量归一化的 SSE 表示<sup>1</sup>。当训练出  $B$  棵彼此差异化的树  $\{T_b\}_{b=1}^B$  后，回归以平均、分类以多数表决进行集成：

$$\hat{f}_{\text{RF}}(x) = \frac{1}{B} \sum_{b=1}^B T_b(x) \quad (\text{回归}), \quad \hat{y}(x) = \text{mode}\{T_b(x)\}_{b=1}^B \quad (\text{分类}).$$

这种“双重随机化 + 集成”的关键作用在于 {降低方差}。若各树预测的方差近似相同为  $\sigma^2$ ，两两相关系数近似为  $\rho$ ，则集成平均的方差满足

$$\text{Var}\left(\frac{1}{B} \sum_{b=1}^B T_b(x)\right) = \frac{\sigma^2}{B} (1 + (B-1)\rho) = \rho \sigma^2 + \frac{1-\rho}{B} \sigma^2.$$

由此可见：减小树与树之间的相关性  $\rho$ （通过特征子采样实现）与增加树的数量  $B$  都能显著降低总体方差。

### 12.1.2.2 袋外误差与时间序列评估

由于 Bootstrap 抽样的性质，每棵树对某个样本“不被抽中”的概率约为

$$\Pr\{x_i \notin D_b\} = \left(1 - \frac{1}{n}\right)^n \approx e^{-1} \approx 0.368,$$

相应地，每个自助样本中“被抽中的独立样本比例”约为  $1 - e^{-1} \approx 0.632$ （即“0.632 自助法”）。因此，对每个样本，可以用“未使用该样本”的那部分树做 {袋外 (out-of-bag, OOB)} 预测。设  $\mathcal{B}^{(-i)}$  表示未包含样本  $i$  的树索引集合，袋外集成预测定义为

$$\hat{f}^{(-i)}(x_i) = \frac{1}{|\mathcal{B}^{(-i)}|} \sum_{b \in \mathcal{B}^{(-i)}} T_b(x_i),$$

据此可得到无需额外验证集的近似样本外误差估计：

$$\hat{\varepsilon}_{\text{OOB}} = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \hat{f}^{(-i)}(x_i)),$$

其中  $\ell$  为任务对应的损失函数（分类任务可取 0-1 损失或对数损失，回归任务可取平方损失）。在时间序列与金融面板数据中，评估必须遵守时间因果顺序：推荐在滚动 / 扩展窗口

<sup>1</sup>下文为统一叙述，将 impurity 译为“不纯度”。在回归情形，可用方差或以样本量归一化的 SSE 表示。

框架下进行“训练—验证”分割，并据此计算 OOB（Out-of-Bag）或直接采用时间序列交叉验证与样本外留出期（out-of-time）评估，避免信息泄漏与生存偏差。

### 12.1.2.3 特征重要性与可解释性

随机森林常见的“基于不纯度下降”的重要性度量方法：将特征  $j$  在其参与分裂的所有节点上的不纯度下降量累加，即得到全局贡献值

$$\text{VI}_j^{\text{imp}} = \sum_{t \in \mathcal{T}_j} \Delta I_t.$$

该指标直观，但对高基数类别或取值范围宽的连续特征可能存在偏好。更稳妥的评估可采用 {置换重要度}，其思想是在验证集中对某一特征  $A$  随机打乱，并观测模型性能的退化幅度：

$$\Delta \mathcal{E}(A) = \mathcal{E}\left(\hat{f}; \pi_A(X); Y\right) - \mathcal{E}\left(\hat{f}; X; Y\right),$$

其中  $\pi_A(\cdot)$  表示仅对特征  $A$  进行置换的算子。为揭示非线性与交互作用，还可使用 {部分依赖图}：

$$\text{PD}_j(z) = \frac{1}{n} \sum_{i=1}^n \hat{f}(z, x_{i,-j}),$$

其中  $x_{i,-j}$  表示除第  $j$  个特征外的其余分量；配合 {ICE（个体条件期望）} 曲线，可观察到个体层面的响应差异。

### 12.1.2.4 金融场景下的优势与局限

在高维、噪声较多且变量相关性强的金融计量分析中，随机森林表现稳健：每次分裂只查看  $m$  个特征，许多无效变量会被随机化“排除在外”，形成类似于隐含的变量筛选；OOB（袋外评估）能快速给出样本外误差，便于调参和监控。然而，随机森林的集成结构使得经济含义的直接归纳并不总是容易；在近似线性的场景下，线性或稀疏正则化模型可能以更低复杂度获得相当甚至更优的效果；此外，树模型本质上以分段常数方式外推，对远离训练支持集的外推预测更需谨慎，并建议在概率输出用于阈值 / 成本决策前做校准（如 Platt 缩放或保序回归）。若将概率用于仓位或阈值决策，可显式写作： $p_t = \Pr(\text{Up}_{t+1} | x_t)$ 、 $w_t = \mathbf{1}\{p_t > \tau\}$ ；含交易成本  $c$  的策略收益为  $r_{t+1}^\pi = w_t r_{t+1} - c|w_t - w_{t-1}|$ ，由此可定义年化 {夏普比率} 为

$$\text{Sharpe} = \frac{\mathbb{E}[r_t^\pi - r_t^f]}{\sqrt{\text{Var}(r_t^\pi - r_t^f)}} \times \sqrt{12},$$

其中  $r_t^f$  为无风险收益率（以上年化因子  $\sqrt{12}$  以月度收益为例；其他频率相应调整）。

## 12.1.3 梯度提升树与 XGBoost

### 12.1.3.1 提升思想与可加性模型

梯度提升决策树（GBDT）以浅树为弱学习器，按可加性方式逐轮逼近目标函数。令  $F_{m-1}(x)$  为前  $m-1$  轮的模型，第  $m$  轮学习器  $h_m(x)$  在损失函数的 {负梯度方向} 上修正

前一轮的残差。设总体损失为  $L = \sum_{i=1}^n \ell(y_i, F_{m-1}(x_i))$ , 则第  $m$  轮的伪残差为

$$r_{im} = -\left. \frac{\partial \ell(y_i, F(x_i))}{\partial F(x_i)} \right|_{F=F_{m-1}}.$$

在样本点上拟合  $h_m$  使其逼近  $\{r_{im}\}$ , 并通过线搜索得到步长  $\gamma_m$ , 更新

$$F_m(x) = F_{m-1}(x) + \eta \gamma_m h_m(x),$$

其中  $\eta \in (0, 1]$  为学习率。以二分类对数损失为例, 若  $p_{m-1}(x) = \sigma(F_{m-1}(x)) = \frac{1}{1+e^{-F_{m-1}(x)}}$ , 则

$$\ell(y, F) = -[y \log p + (1-y) \log(1-p)], \quad r_{im} = y_i - p_{m-1}(x_i),$$

说明每一轮都在“用浅树拟合残差”。

### 12.1.3.2 XGBoost 的目标函数与正则化

**XGBoost** 是 GBDT 的代表性实现之一 (Chen & Guestrin 2016)。每一轮通过二阶泰勒展开近似整体损失, 并加入结构正则项:

$$\min_{f_m} \sum_{i=1}^n \left[ g_i f_m(x_i) + \frac{1}{2} h_i f_m(x_i)^2 \right] + \Omega(f_m), \quad \Omega(f) = \gamma T + \frac{\lambda}{2} \sum_{j=1}^T w_j^2,$$

其中  $g_i = \partial_F \ell(y_i, F)|_{F=F_{m-1}}$ 、 $h_i = \partial_F^2 \ell(y_i, F)|_{F=F_{m-1}}$  为梯度与海森 (二阶) 信息,  $T$  为叶节点数,  $w_j$  为第  $j$  个叶节点的权重。记每个叶区域  $R_j$  内的梯度 / 海森聚合为

$$G_j = \sum_{x_i \in R_j} g_i, \quad H_j = \sum_{x_i \in R_j} h_i,$$

则对应的最优叶权重与近似最优目标值为

$$w_j^* = -\frac{G_j}{H_j + \lambda}, \quad \tilde{\mathcal{L}}(f_m) = -\frac{1}{2} \sum_{j=1}^T \frac{G_j^2}{H_j + \lambda} + \gamma T.$$

对任意一次二分操作, 若左右子区的聚合量为  $(G_L, H_L)$  与  $(G_R, H_R)$ , 则其 {分裂增益} (gain) 为

$$\text{Gain} = \frac{1}{2} \left[ \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma,$$

当  $\text{Gain} > 0$  时才值得分裂。该形式清楚地体现了二阶信息 ( $H$ )、正则项 ( $\lambda$ ) 与结构惩罚项 ( $\gamma$ ) 对“是否继续生长”的共同约束。

### 12.1.3.3 可定制损失函数与业务约束

在金融任务中, XGBoost 的一个重要优势是可定制目标函数。例如, 风险度量中可采用 {分位数损失} 来拟合 VaR 和 ES:

$$L_\tau(y, \hat{y}) = (\tau - \mathbf{1}_{\{y < \hat{y}\}})(y - \hat{y}).$$

量化选股可采用配对 / 排序损失（通过分数差拉开排名）：

$$\sum_{\substack{(i,j) \\ y_i > y_j}} \log \left( 1 + \exp \left[ - (F(x_i) - F(x_j)) \right] \right).$$

信用评分的类不平衡可用加权对数损失

$$\ell_\alpha(y, p) = -\alpha y \log p - (1 - \alpha)(1 - y) \log(1 - p),$$

或直接设置正负样本权重。为满足经济学上的单调关系（如“利率上升  $\Rightarrow$  违约概率上升”），可施加 {单调性约束}；为在阈值决策中获得更可靠的收益评估，建议在样本外验证集上进行概率校准（如 Platt 校准或保序回归）。

#### 12.1.3.4 与随机森林的差异与选型

从机制上看，随机森林通过“并行训练的深树 + 投票/平均”降低方差，调参成本低，袋外误差能直接提供近似样本外评价；XGBoost 通过“串行叠加的浅树 + 梯度驱动”降低偏差，往往能在表格型数据上取得更高精度，但对学习率、结构惩罚与采样率更为敏感，需要验证集与早停共同保障稳定性。在缺失值处理上，随机森林多依赖替代划分或显式插补，而 XGBoost 自带默认方向；在可解释性上，两者均可配合置换重要度与 SHAP 实现全局与局部解释。选型上，若需要一个稳健、调参成本低的强基线并快速得到样本外评估，随机森林是自然起点；若追求更高的精度与更灵活的目标定制，在严格的时间序列评估框架下采用 XGBoost 更具潜力。

#### 12.1.3.5 与金融计量的衔接与注意事项

本节方法以 {预测} 为目标，适用于高维特征的非线性整合与截面排序。将其纳入“金融计量学”的语境，需要明确：第一，评估体系应以 {样本外表现} 为中心，遵循时间顺序，采用滚动 / 扩展窗口与样本外预留期；第二，除 AUC / MSE 外，应报告与任务相称的 {收益—风险} 指标（年化收益率、波动率、最大回撤、{夏普比率 / 索提诺比率}），并在信用 / 欺诈等任务中引入阈值—成本分析与放款率约束；第三，应关注 {数据漂移与模型稳定性}，可用人口稳定性指数（PSI）度量分布漂移：

$$\text{PSI} = \sum_b (p_b - q_b) \ln \frac{p_b}{q_b},$$

其中  $p_b, q_b$  分别为基准期与当前期在第  $b$  个分箱中的占比。部署阶段应监控特征重要性漂移与再训练节奏。第四，若研究重在 {推断或因果}，可在样本分割与正交化框架下，将机器学习模型嵌入 Double Machine Learning 或 Causal Forest 等方法中，以获得更稳健的边际效应估计与置信区间。同时，在概率评估上可报告 {Brier 分数}：

$$\text{BS} = \frac{1}{n} \sum_{i=1}^n (y_i - p_i)^2,$$

并在成本敏感阈值选择时利用

$$\tau^* = \frac{c_{\text{FP}}}{c_{\text{FP}} + c_{\text{FN}}}$$

作为最优分类阈值的基准 ( $c_{FP}, c_{FN}$  分别为误报与漏报的单位成本)，将模型概率与业务成本直接对齐。

### 12.1.3.6 与随机森林的差异与选型

随机森林与 XGBoost 同为树模型的集成方法：前者通过并行装袋降低方差，后者通过梯度提升降低偏差。二者在金融数据应用中各有侧重：当需要稳健基线与快速样本外评估时，可优先采用随机森林；当需要更高精度与可定制目标时，可采用 XGBoost，并以时间序列交叉验证、早停法与概率校准保证可用性。通过置换重要性、SHAP 值分析与部分依赖图等工具，可以在模型精度与经济学解释之间取得更好的平衡。

### 12.1.3.7 案例：随机森林 vs 逻辑回归（固定窗口，滚动样本外预测）

本节配套代码以 `quantmod::getSymbols` 从 Yahoo Finance 自动下载价格数据，优先获取 ETF SPY (失败时回退至指数 `^GSPC`)，并统一使用复权收盘价 (`Adjusted`)。为保证可复现性，代码将样本区间固定在 `start_date = "2010-01-01"` 至 `end_date = "2024-12-31"`，设置随机种子 `set.seed(2025)`，并在下载时额外向前抓取一段历史数据用于技术指标的“预热”。价格序列经 `to.monthly` 聚合为“月末”频率，随后计算月度对数收益率  $r_t = \Delta \log P_t$ 。在此基础上构造一组仅依赖过去信息的技术特征：6 / 12 个月动量、3 个月波动率、10 月均线偏离率、RSI(14)、以及 MACD 直方图。所有特征在预测前统一滞后一周期，即使用时点  $t$  的特征预测  $t+1$  时刻的方向，从而消除前视偏差。标签定义为下月方向  $y_{t+1} = \mathbf{1}\{r_{t+1} > 0\}$ ，并将因子水平设为 {Up, Down} 以便 `glm` 按“Up 为成功事件”返回概率。

样本划分遵循时间顺序。代码自适应地选择训练窗口长度：记有效月数为  $\ell$ ，在保证至少 6 个月样本外评估的前提下，令训练窗口长度  $W = \min(120, \max(18, \ell - 6))$ 。随后按“滚动窗口”的方式进行样本外预测：第  $i$  次迭代使用最近  $W$  个月作为训练集，预测下一个月的单点测试样本。比较的模型包括传统 {逻辑回归} (`glm`, 二项族) 与 {随机森林} (`randomForest`,  $B = 500$  棵树, `mtry = \lfloor \sqrt{p} \rfloor`, `nodesize = 5`, 并开启 `importance=TRUE` 以便后续分析)。对每个月份，分别得到“上涨概率”  $\hat{p}_t^{\text{GLM}}$  与  $\hat{p}_t^{\text{RF}}$ 。

评估分为两层：第一层是区分度指标，即 AUC 与基于阈值 0.5 的准确率。AUC 使用 `pROC` 计算，确保阳性类顺序为 {Down, Up} 以获得方向一致的曲线；准确率则将  $\hat{p}_t > 0.5$  视为“看涨”。第二层是择时策略检验，以便把概率信号转化为收益—风险指标。仓位规则为  $w_t = \mathbf{1}\{\hat{p}_t > 0.5\}$ ，策略收益写作

$$r_{t+1}^\pi = w_t r_{t+1} - c |w_t - w_{t-1}|,$$

其中  $c = 5\text{bp}$  为单边交易成本 (代码中记为 `tc=0.0005`)，换手率由  $|w_t - w_{t-1}|$  表示。对两种模型分别得到月度策略收益序列，并与买入并持有 (BH) 基准对比。年化收益率、年化波动率与夏普比率按月频换算，最终绘制三条净值曲线用于直观比较。需要强调的是，这里故意保留了“最简单”的阈值与二元仓位，用于展示“模型概率——交易信号——收益表现”的全链路；在长期上行的单资产环境中，买入并持有往往占优，这一基准有助于提醒读者：择时要想胜出，必须通过阈值、成本与风险约束的联合优化来克服“现金拖累”和“误切换”的影响。

代码还考虑了短样本的可用性。当研究者将起始日期改到近年 (例如 2020 年) 时，有效月数  $\ell$  变短，程序会自动缩短训练窗口但保留至少 6 个月的样本外期；若  $\ell < 18 + 6$  则提示扩大区间或放宽最小训练 / 样本外长度。实践中如需提高策略可用性，可在此框架上继续引入概率校准 (Platt 缩放或保序回归)、连续仓位映射  $w_t = \text{clip}(2(\hat{p}_t - 0.5), 0, 1)$ 、阈

值的样本外选择（例如在滚动验证窗口内直接最大化样本外夏普比率），以及更贴近经济含义的额外特征（期限利差、信用利差、VIX 指数等）。上述改动都可以在不改变“固定窗口 + 滚动样本外预测”的大框架下逐项加入，从而得到更贴近真实投研流程的实验结果。

```

1 # =====
2 # 随机森林 vs 逻辑回归（固定时间区间 + 自适应训练窗）
3 # 数据源：Yahoo Finance (quantmod::getSymbols)
4 # 任务：预测下月方向（Up/Down），滚动样本外比较
5 # 说明：支持短样本（如 2020-01-01 到 2024-12-31）
6 # =====
7
8 # ----- 固定时间区间（可按需修改） -----
9 start_date <- as.Date("2010-01-01")
10 end_date <- as.Date("2024-12-31")
11
12 # ---- 缺包自动安装并加载 ----
13 pkgs <- c("quantmod", "TTR", "dplyr", "lubridate", "randomForest", "pROC",
14 "ggplot2", "tidyverse", "zoo")
15 for (p in pkgs) if (!requireNamespace(p, quietly = TRUE)) install.packages(p)
16 invisible(lapply(pkgs, library, character.only = TRUE))
17 options(stringsAsFactors = FALSE, scipen = 99, timeout = max(300,getOption(
18 "timeout")))
19 set.seed(2025) # 固定随机种子，保证可复现
20
21 # -----
22 # 1) 下载价格（优先 SPY，失败回退 ^GSPC），时间区间固定
23 # -----
24 symbols <- c("SPY", "^GSPC")
25 px <- NULL
26 for (sym in symbols) {
27 message("尝试下载：", sym)
28 ok <- try({
29 x <- suppressWarnings(
30 getSymbols(sym, src = "yahoo",
31 from = start_date - 400, # 提前取历史，供技术指标“热身”
32 to = end_date,
33 auto.assign = FALSE)
34 if (!is.null(x)) { px <- x; attr(px, "symbol") <- sym }
35 }, silent = TRUE)
36 if (!is.null(px)) break
37 }
38 if (is.null(px)) stop("数据下载失败（检查网络或代理）。")
39
40 adj <- Ad(px) # 复权收盘价
41
42 # -----
43 # 2) 聚合到月频 + 技术特征（仅用过去信息）
44 # -----
45 adj_m <- to.monthly(adj, indexAt = "lastof", OHLC = FALSE)

```

```

46 adj_m <- adj_m[paste0(format(start_date, "%Y-%m"), "/", format(end_date, "%Y
 -%m"))]
47
48 ret_m <- diff(log(adj_m)) # 月度对数收益 r_t
49
50 # 特征：动量(6/12)、波动(3)、均线偏离(SMA10)、RSI(14)、MACD直方
51 mom6 <- log(adj_m / lag(adj_m, 6))
52 mom12 <- log(adj_m / lag(adj_m, 12))
53 vol3 <- runSD(ret_m, n = 3)
54 sma10 <- SMA(adj_m, n = 10)
55 sma_ratio <- adj_m / sma10 - 1
56 rsi14 <- RSI(adj_m, n = 14)
57 macd <- MACD(adj_m, nFast = 12, nSlow = 26, nSig = 9)
58 macd_m <- macd[, "macd"]
59 macd_h <- macd_m - macd[, "signal"]
60
61 df <- tibble(
62 date = as.Date(index(adj_m)),
63 price = as.numeric(adj_m),
64 r1 = as.numeric(ret_m),
65 mom6 = as.numeric(mom6),
66 mom12 = as.numeric(mom12),
67 vol3 = as.numeric(vol3),
68 sma_ratio = as.numeric(sma_ratio),
69 rsi14 = as.numeric(rsi14),
70 macd = as.numeric(macd_m),
71 macd_hist = as.numeric(macd_h)
72) |>
73 arrange(date) |>
74 mutate(
75 ret_fwd = dplyr::lead(r1, 1), # 预测目标：
76 下一期(月)收益
77 y = factor(ifelse(ret_fwd > 0, "Up", "Down"),
78 levels = c("Up", "Down")), # GLM 将第一
79 across(c(mom6, mom12, vol3, sma_ratio, rsi14, macd, macd_hist),
80 ~ dplyr::lag(.x, 1), .names = "{.col}_lag") # 特征统一滞
81 后 1 期
82) |>
83 select(date, y, ret_fwd, ends_with("_lag")) |>
84 tidyr::drop_na()
85
86 # -----
87 # 3) 自适应训练窗口(避免“时间不够”)
88 # -----
89 len <- nrow(df) # 可用月数(已扣除指标热身与滞后)
90 min_train <- 18 # 最小训练窗(个月)，短样本时放宽到 18
91 min_oos <- 6 # 最少样本外月数
92 if (len < (min_train + min_oos)) {
93 stop(sprintf("有效月数不足：当前%d，至少需要%d(最小训练%d + 最小样本外%d
94)。请适当延长区间。",

```

```
92 len, min_train + min_oos, min_train, min_oos))
93 }
94 train_win <- min(120, max(min_train, len - min_oos)) # 自适应：不超过120，且保留至少 min_oos
95 oos_months <- len - train_win
96 message(sprintf("可用月数=%d; 训练窗=%d; 样本外=%d。", len, train_win, oos_
97 months))
98 features <- names(df)[grepl("_lag$", names(df))]
99
100 # -----
101 # 4) 滚动样本外：随机森林 vs 逻辑回归
102 # -----
103 oos <- list()
104 for (i in (train_win + 1):nrow(df)) {
105 train <- df[(i - train_win):(i - 1),]
106 test <- df[i,]
107
108 # 逻辑回归
109 fit_glm <- glm(
110 reformulate(term.labels = features, response = "y"),
111 data = train, family = binomial()
112)
113 p_glm <- as.numeric(predict(fit_glm, newdata = test, type = "response"))
114 # 概率 P(Up)
115
116 # 随机森林
117 fit_rf <- randomForest(
118 x = train[, features],
119 y = train$y,
120 ntree = 500,
121 mtry = max(1, floor(sqrt(length(features)))),
122 nodesize = 5,
123 importance = TRUE
124)
125 p_rf <- as.numeric(predict(fit_rf, newdata = test[, features, drop = FALSE
126], type = "prob")[, "Up"])
127
128 oos[[length(oos) + 1]] <- tibble(
129 date = test$date,
130 ret_fwd = test$ret_fwd,
131 p_glm = p_glm,
132 p_rf = p_rf
133)
134
135 # -----
136 # 5) 指标与策略：AUC / ACC + 阈值择时（含交易成本）
137 # -----
138 resp <- factor(ifelse(oos$ret_fwd > 0, "Up", "Down"), levels = c("Down", "Up"))
```

```

)) # Up 为阳性
139
140 auc_glm <- as.numeric(pROC::auc(pROC::roc(resp, oos$p_glm, quiet = TRUE)))
141 auc_rf <- as.numeric(pROC::auc(pROC::roc(resp, oos$p_rf , quiet = TRUE)))
142
143 pred_glm <- ifelse(oos$p_glm > 0.5, "Up", "Down")
144 pred_rf <- ifelse(oos$p_rf > 0.5, "Up", "Down")
145 acc_glm <- mean(pred_glm == as.character(resp), na.rm = TRUE)
146 acc_rf <- mean(pred_rf == as.character(resp), na.rm = TRUE)
147
148 # 简单择时: p>0.5 则满仓, 否则持有现金; 计入换手成本 (单边 5bp)
149 tc <- 0.0005
150 sig_glm <- ifelse(oos$p_glm > 0.5, 1, 0)
151 sig_rf <- ifelse(oos$p_rf > 0.5, 1, 0)
152 turn_glm <- c(0, abs(diff(sig_glm)))
153 turn_rf <- c(0, abs(diff(sig_rf)))
154
155 ret_glm <- sig_glm * oos$ret_fwd - turn_glm * tc
156 ret_rf <- sig_rf * oos$ret_fwd - turn_rf * tc
157 ret_bh <- oos$ret_fwd # 买入并持有
158
159 nav <- tibble(
160 date = oos$date,
161 GLM = cumprod(1 + ret_glm),
162 RF = cumprod(1 + ret_rf),
163 BH = cumprod(1 + ret_bh)
164) |>
165 tidyr::pivot_longer(-date, names_to = "Strategy", values_to = "NAV")
166
167 # 年化统计 (按月频)
168 ann_factor <- 12
169 stat <- tibble(
170 Strategy = c("GLM", "RF", "BH"),
171 AnnRet = c(prod(1 + ret_glm)^(ann_factor/length(ret_glm)) - 1,
172 prod(1 + ret_rf)^(ann_factor/length(ret_rf)) - 1,
173 prod(1 + ret_bh)^(ann_factor/length(ret_bh)) - 1),
174 AnnVol = c(sd(ret_glm, na.rm = TRUE) * sqrt(ann_factor),
175 sd(ret_rf , na.rm = TRUE) * sqrt(ann_factor),
176 sd(ret_bh , na.rm = TRUE) * sqrt(ann_factor))
177) |>
178 mutate(Sharpe = AnnRet / AnnVol)
179
180 stat_rounded <- stat |> dplyr::mutate(dplyr::across(where(is.numeric), ~
181 round(.x, 3)))
182 cat("\n===== 固定区间: ", format(start_date), " 至 ", format(end_date),
183 "; 标的: ", attr(px, "symbol"), " =====\n", sep = "")
184 cat("样本外指标 (AUC 与 ACC) : \n")
185 cat(sprintf(" AUC (GLM, RF): %.3f, %.3f\n", auc_glm, auc_rf))
186 cat(sprintf(" ACC (GLM, RF): %.3f, %.3f\n\n", acc_glm, acc_rf))
187

```

```

188 cat("年化统计 (AnnRet / AnnVol / Sharpe) : \n")
189 print(stat_rounded)
190
191 # -----
192 # 6) 可视化: 三种策略累计净值
193 #
194 ggplot(nav, aes(date, NAV, color = Strategy)) +
195 geom_line(linewidth = 0.9) +
196 labs(
197 title = paste0("随机森林 vs 逻辑回归 (",
198 attr(px, "symbol"),
199 ", 月频, 滚动样本外; ",
200 format(start_date), "—",
201 format(end_date), ")"),
202 x = NULL, y = "净值 (起点=1)"
203) +
204 theme_minimal() +
205 theme(legend.position = "bottom")

```

### 12.1.3.8 代码说明: XGBoost (固定区间, 滚动样本外, 早停)

本节配套代码用于展示 XGBoost 在单资产“下月方向”预测中的完整建模—评估流程。数据通过 `quantmod::getSymbols` 从 Yahoo Finance 自动下载，优先使用含分红复权的 ETF SPY (失败时回退指数 `^GSPC`)，并固定样本区间为 `start_date = 2010-01-01` 至 `end_date = 2024-12-31`。为保证可复现性，设置随机种子 `set.seed(2025)` 且令 `xgboost` 的 `nthread=1`。价格序列按“月末”聚合得到月频复权价  $P_t$ ，月度对数收益定义为

$$r_t = \Delta \log P_t = \log P_t - \log P_{t-1}.$$

在此基础上构造仅依赖过去信息的技术特征：6/12 个月动量、3 个月波动率、10 月均线偏离、RSI(14)、以及 MACD 与其直方。所有特征在预测前统一滞后一周期，即用  $t$  时点的特征去预测  $t+1$  的方向，从而避免前视偏差。标签定义为下月是否上涨

$$y_{t+1} = \mathbf{1}\{r_{t+1} > 0\} \in \{0, 1\},$$

并以 `binary:logistic` 目标学习概率  $p_t = \Pr(y_{t+1} = 1 | x_t)$ 。

样本划分严格遵守时间顺序，采用“固定训练窗 + 单点滚动测试”的样本外评估框架。设有效月数为  $\ell$ ，程序自适应选择训练窗

$$W = \min(120, \max(24, \ell - 6)),$$

确保至少 6 个月样本外期。每次迭代使用最近  $W$  个月作为训练集，并在其末尾切出一段验证窗（约为  $W$  的 20%，至少 6 个月）用于早停；训练目标为 AUC，基础超参数取浅树 (`max_depth = 3`)、学习率  $\eta = 0.05$ 、行 / 列子采样 0.8，并以

$$\text{scale\_pos\_weight} = \frac{\#\{y = 0\}}{\#\{y = 1\}}$$

处理可能的类别不平衡问题。每轮训练得到对下一月的样本外上涨概率  $\hat{p}_t$ ，并存入时间序列用于汇总评估。

评估包含“区分度”和“策略化”两条线索。区分度方面，采用样本外 AUC 与基于阈

值 0.5 的准确率 (ACC)，反映概率排序与二元判别的基本能力。策略化方面，将概率映射为二元仓位  $w_t = \mathbf{1}\{\hat{p}_t > 0.5\}$ ，并计入单边交易成本  $c = 5 \text{ bp}$ ，策略月度收益记为

$$r_{t+1}^{\pi} = w_t r_{t+1} - c |w_t - w_{t-1}|.$$

据此计算年化收益率、年化波动率与夏普比率（按月频换算，乘以  $\sqrt{12}$ ），并与买入持有策略 (BH) 进行并列对比，绘制两条净值曲线以直观展示样本外表现。需要强调的是，这里刻意保持最简单的阈值与二元仓位设置，以便将“模型概率——仓位——收益”的链条清晰呈现；在长期上行的单资产环境中，BH 通常具备天然优势，因而该基准可作为检验择时信号经济价值的下限。

代码最后给出一次基于 `xgb.importance` 的特征重要度报告 (“Gain”)，展示哪些特征在分裂中贡献了更多损失下降，并绘制前若干名的重要度条形图以便说明模型“依赖了什么信息”。需要注意，“Gain” 反映的是“贡献大小”而非因果方向；若要观察方向与非线性，可在此框架上进一步输出 SHAP 值或进行样本外置换重要度检验。整体流程不依赖具体标的与特征选择：研究者只需在固定区间、滚动评估与早停的框架内替换或扩展特征（例如加入期限利差、信用利差、VIX 指数等宏观风险因子），即可得到可复现、可解释且贴近实务的样本外结果。

```

1 # =====
2 # XGBoost 案例 (固定区间 + 滚动样本外 + 早停)
3 # 数据源: Yahoo Finance (quantmod::getSymbols)
4 # 任务: 预测下月方向 (Up/Down)，并做简易择时评价
5 # =====
6
7 # ----- 可复现的固定时间区间 (按需修改) -----
8 start_date <- as.Date("2010-01-01")
9 end_date <- as.Date("2024-12-31")
10
11 # ---- 缺包自动安装并加载 ----
12 pkgs <- c("quantmod", "TTR", "dplyr", "lubridate", "pROC", "ggplot2", "tidyverse", "zoo",
13 "xgboost", "Matrix")
14 for (p in pkgs) if (!requireNamespace(p, quietly = TRUE)) install.packages(p)
15
16 invisible(lapply(pkgs, library, character.only = TRUE))
17
18 options(stringsAsFactors = FALSE, scipen = 99, timeout = max(300, getOption(
19 "timeout")))
20 set.seed(2025) # 固定随机种子，配合 xgboost 的 nthread=1，保证可复现
21
22 # -----
23 # 1) 下载价格 (优先 SPY，失败回退 ^GSPC)，时间区间固定
24 #
25 symbols <- c("SPY", "^GSPC")
26 px <- NULL
27 for (sym in symbols) {
28 message("尝试下载: ", sym)
29 ok <- try({
30 x <- suppressWarnings(
31 getSymbols(sym, src = "yahoo",
32 from = start_date - 400, # 提前抓历史，给技术指标“热身”
33 to = end_date)
34 })
35 if (ok) px <- rbind(px, px[order(px$Date),])
36 }
37
38 # -----
```

```

30 to = end_date,
31 auto.assign = FALSE)
32)
33 if (!is.null(x)) { px <- x; attr(px, "symbol") <- sym }
34 }, silent = TRUE)
35 if (!is.null(px)) break
36 }
37 if (is.null(px)) stop("数据下载失败（检查网络或代理）。")
38
39 adj <- Ad(px) # 复权收盘价（ETF含分红复权）
40
41 # -----
42 # 2) 聚合为月频 + 特征工程（仅用过去信息）
43 # -----
44 adj_m <- to.monthly(adj, indexAt = "lastof", OHLC = FALSE)
45 adj_m <- adj_m[paste0(format(start_date, "%Y-%m"), "/",
46 format(end_date, "%Y-%m"))]
47 ret_m <- diff(log(adj_m)) # 月度对数收益 r_t
48
49 # 技术特征：动量(6/12)、波动(3)、SMA10 偏离、RSI(14)、MACD 直方
50 mom6 <- log(adj_m / lag(adj_m, 6))
51 mom12 <- log(adj_m / lag(adj_m, 12))
52 vol3 <- runSD(ret_m, n = 3)
53 sma10 <- SMA(adj_m, n = 10)
54 sma_ratio <- adj_m / sma10 - 1
55 rsi14 <- RSI(adj_m, n = 14)
56 macd <- MACD(adj_m, nFast = 12, nSlow = 26, nSig = 9)
57 macd_m <- macd[, "macd"]
58 macd_h <- macd_m - macd[, "signal"]
59
60 # 组装数据框，并将特征统一滞后 1 期（用 t 特征预测 t+1）
61 df <- tibble(
62 date = as.Date(index(adj_m)),
63 price = as.numeric(adj_m),
64 r1 = as.numeric(ret_m),
65 mom6 = as.numeric(mom6),
66 mom12 = as.numeric(mom12),
67 vol3 = as.numeric(vol3),
68 sma_ratio = as.numeric(sma_ratio),
69 rsi14 = as.numeric(rsi14),
70 macd = as.numeric(macd_m),
71 macd_hist = as.numeric(macd_h)
72) |>
73 arrange(date) |>
74 mutate(
75 ret_fwd = dplyr::lead(r1, 1), # 目标：下一
76 # 期（月）收益
77 y_fac = factor(ifelse(ret_fwd > 0, "Up", "Down"),
78 levels = c("Down", "Up")), # 方便 AUC 的
79 # 阳性类顺序（Up 为阳性）
80)

```

```

78 y_bin = as.numeric(y_fac == "Up"), # XGBoost 的
79 0/1 标签
80 across(c(mom6, mom12, vol3, sma_ratio, rsi14, macd, macd_hist),
81 ~ dplyr::lag(.x, 1), .names = "{.col}_lag") # 特征统一滞后 1 期
82) |>
83 select(date, y_fac, y_bin, ret_fwd, ends_with("_lag")) |>
84 tidyr::drop_na()
85
86 # -----#
87 # 3) 自适应训练窗口 + 验证窗 (用于早停)
88 # -----
89 len <- nrow(df)
90 min_train <- 24 # 至少 24 个月训练
91 min_oos <- 6 # 至少 6 个月样本外
92 if (len < (min_train + min_oos)) {
93 stop(sprintf("有效月数不足：当前%d，至少需要%d (训练 %d + 样本外 %d)。",
94 len, min_train + min_oos, min_train, min_oos))
95 }
96 train_win <- min(120, max(min_train, len - min_oos)) # 训练窗不超过 120 个月
97 oos_months <- len - train_win
98 message(sprintf("可用月数=%d；训练窗=%d；样本外=%d。", len, train_win, oos_months))
99 features <- names(df)[grepl("_lag$", names(df))]
100
101 # -----
102 # 4) 滚动训练 + 早停：每次用训练窗的末尾做验证（例如 20% 或至少 6 个月）
103 # -----
104 oos <- list()
105 last_model <- NULL # 用于训练结束后展示一次重要度
106 for (i in (train_win + 1):nrow(df)) {
107 train_df <- df[(i - train_win):(i - 1),]
108 test_df <- df[i,]
109
110 # 验证窗长度：train_win 的 20%，至少 6 个月，且留出 >=12 个月给纯训练
111 val_win <- min(max(6, round(0.2 * nrow(train_df))), max(6, nrow(train_df) - 12))
112 core_idx <- 1:(nrow(train_df) - val_win)
113 val_idx <- (nrow(train_df) - val_win + 1):nrow(train_df)
114
115 X_core <- as.matrix(train_df[core_idx, features])
116 y_core <- train_df$y_bin[core_idx]
117 X_val <- as.matrix(train_df[val_idx, features])
118 y_val <- train_df$y_bin[val_idx]
119 X_test <- as.matrix(test_df[, features])
120
121 dtrain <- xgb.DMatrix(data = X_core, label = y_core, missing = NA)
122 dval <- xgb.DMatrix(data = X_val, label = y_val, missing = NA)
123 dtest <- xgb.DMatrix(data = X_test, missing = NA)

```

```
124
125 # 类不平衡权重（负/正）
126 pos <- max(1L, sum(y_core == 1))
127 neg <- max(1L, sum(y_core == 0))
128 spw <- as.numeric(neg / pos)
129
130 params <- list(
131 objective = "binary:logistic",
132 eval_metric = "auc",
133 eta = 0.05, # 学习率
134 max_depth = 3, # 浅树更稳健
135 min_child_weight = 5,
136 subsample = 0.8,
137 colsample_bytree = 0.8,
138 lambda = 1.0, # L2
139 alpha = 0.0, # L1
140 gamma = 0.0, # 结构惩罚
141 scale_pos_weight = spw,
142 nthread = 1 # 为了复现性
143)
144
145 watch <- list(train = dtrain, val = dval)
146 bst <- xgb.train(
147 params = params,
148 data = dtrain,
149 nrounds = 2000,
150 watchlist = watch,
151 early_stopping_rounds = 50, # 早停
152 verbose = 0
153)
154 last_model <- bst
155
156 p_xgb <- as.numeric(predict(bst, dtest, ntreelimit = bst$best_ntreelimit))
157 oos[[length(oos) + 1]] <- tibble(
158 date = test_df$date,
159 ret_fwd = test_df$ret_fwd,
160 p_xgb = p_xgb
161)
162 }
163 oos <- bind_rows(oos) |> arrange(date)
164
165 # -----
166 # 5) 指标与策略：AUC / ACC + 阈值择时（含交易成本）
167 # -----
168 resp <- factor(ifelse(oos$ret_fwd > 0, "Up", "Down"), levels = c("Down", "Up"))
169
170 auc_xgb <- as.numeric(pROC::auc(pROC::roc(resp, oos$p_xgb, quiet = TRUE)))
171 pred_xgb <- ifelse(oos$p_xgb > 0.5, "Up", "Down")
172 acc_xgb <- mean(pred_xgb == as.character(resp), na.rm = TRUE)
173
```

```

174 # 简单择时：p>0.5 则满仓，否则持有现金；计入换手成本（单边 5bp）
175 tc <- 0.0005
176 sig_xgb <- ifelse(oos$p_xgb > 0.5, 1, 0)
177 turn_xgb <- c(0, abs(diff(sig_xgb)))
178 ret_xgb <- sig_xgb * oos$ret_fwd - turn_xgb * tc
179 ret_bh <- oos$ret_fwd # 买入并持有基准
180
181 # 累计净值
182 nav <- tibble(
183 date = oos$date,
184 XGB = cumprod(1 + ret_xgb),
185 BH = cumprod(1 + ret_bh)
186) |>
187 tidyr::pivot_longer(-date, names_to = "Strategy", values_to = "NAV")
188
189 # 年化统计（按月频）
190 ann_factor <- 12
191 stat <- tibble(
192 Strategy = c("XGB", "BH"),
193 AnnRet = c(prod(1 + ret_xgb)^{(ann_factor/length(ret_xgb))} - 1,
194 prod(1 + ret_bh)^{(ann_factor/length(ret_bh))} - 1),
195 AnnVol = c(sd(ret_xgb, na.rm = TRUE) * sqrt(ann_factor),
196 sd(ret_bh, na.rm = TRUE) * sqrt(ann_factor))
197) |>
198 mutate(Sharpe = AnnRet / AnnVol)
199 stat_rounded <- stat |> dplyr::mutate(dplyr::across(where(is.numeric), ~
200 round(.x, 3)))
201
202 cat("\n==== 固定区间：", format(start_date), " 至 ", format(end_date),
203 "; 标的：", attr(px, "symbol"), " =====\n", sep = "")
204 cat(sprintf("AUC_XGB = %.3f, ACC_XGB = %.3f\n\n", auc_xgb, acc_xgb))
205 cat("年化统计 (AnnRet / AnnVol / Sharpe) : \n")
206 print(stat_rounded)
207
208 # -----
209 # 6) 可视化：净值曲线与特征重要度（稳健写法，避免 slice 冲突）
210 # -----
211
212 # 净值
213 p1 <- ggplot(nav, aes(date, NAV, color = Strategy)) +
214 geom_line(linewidth = 0.9) +
215 labs(
216 title = paste0("XGBoost 择时 vs 买入并持有 (", attr(px, "symbol"),
217 ", 月频, 滚动样本外; ", format(start_date), "—", format(
218 end_date), ")"),
219 x = NULL, y = "净值 (起点=1)"
220) +
221 theme_minimal() +
222 theme(legend.position = "bottom")
223 print(p1)

```

```

223 # 特征重要度 (基于最后一轮模型)
224 imp <- xgb.importance(model = last_model, feature_names = features)
225 imp_df <- as.data.frame(imp, stringsAsFactors = FALSE)
226
227 # 按 Gain 从高到低排序, 取前 10 (不使用 slice/n(), 跨版本最稳)
228 ord <- order(-imp_df$Gain)
229 top_n <- min(10L, nrow(imp_df))
230 imp_top <- imp_df[ord[seq_len(top_n)], , drop = FALSE]
231
232 print(imp_top) # 在控制台查看前 10 个重要特征
233
234 # 绘图
235 p2 <- ggplot(imp_top, aes(x = reorder(Feature, Gain), y = Gain)) +
236 geom_col() +
237 coord_flip() +
238 labs(title = "XGBoost 特征重要度 (Gain, 最后一轮模型)", x = NULL, y = "Gain") +
239 theme_minimal()
240 print(p2)

```

## 12.2 神经网络与深度学习方法概述及金融场景应用

### 12.2.1 背景与动机

人工神经网络 (Artificial Neural Network, ANN) 通过堆叠线性变换与非线性激活来逼近复杂函数。与传统线性 / 广义线性模型相比, 神经网络不预设函数形式, 而是以数据驱动的方式自动学习非线性与高阶交互。在金融计量的语境下, 这一点尤为重要: 横截面选股、文本情绪、盘口微结构、高维宏观合成指标等场景往往存在结构性非线性与异质性。深度学习 (多层神经网络) 依靠多级非线性映射构建 {表示}, 为这些问题提供了统一的函数逼近框架; 相应地, 金融数据的高噪声、弱信号与非平稳性, 使得模型选择、正则化与 {严格的样本外评估} 成为落地的关键。

### 12.2.2 基本模型与经验风险最小化

以单隐藏层前馈网络 (多层感知机, MLP) 为例, 输入  $\mathbf{x} \in \mathbb{R}^p$ 、隐藏层宽度为  $H$ , 前向传播

$$\mathbf{a}^{(1)} = \sigma(W^{(1)}\mathbf{x} + \mathbf{b}^{(1)}) \in \mathbb{R}^H, \quad (12.1)$$

$$\hat{y} = \sigma_{\text{out}}(W^{(2)}\mathbf{a}^{(1)} + \mathbf{b}^{(2)}), \quad (12.2)$$

其中  $W^{(1)} \in \mathbb{R}^{H \times p}$ ,  $W^{(2)} \in \mathbb{R}^{1 \times H}$  (或多维输出),  $\sigma$  常取 ReLU / tanh, 输出层映射  $\sigma_{\text{out}}$  为恒等 (回归)、sigmoid (两类) 或 softmax (多类,  $p_k = \exp z_k / \sum_j \exp z_j$ )。训练遵循 {经验风险最小化} 并配正则:

$$\min_{\theta} \mathcal{J}(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, f_{\theta}(\mathbf{x}_i)) + \lambda \Omega(\theta), \quad (12.3)$$

典型损失包括均方误差

$$\ell_{\text{MSE}}(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2,$$

二分类交叉熵（最大似然）

$$\ell_{\text{CE}}(y, p) = -[y \log p + (1 - y) \log(1 - p)], \quad p = \sigma(z),$$

以及 {分位数损失}（用于 VaR 等）

$$\ell_{\tau}(y, \hat{q}) = \left(\tau - \mathbf{1}\{y < \hat{q}\}\right)(y - \hat{q}), \quad \tau \in (0, 1).$$

参数以随机梯度类方法更新

$$\theta \leftarrow \theta - \eta \nabla_{\theta} \mathcal{J}(\theta), \quad (12.4)$$

梯度由反向传播算法计算；为缓解“梯度消失或爆炸”问题，参数初始化常用 Xavier 或 He 方法，激活函数倾向选用 ReLU 或 GELU。

### 12.2.3 优化与训练细节

实践中常用 Adam (Adaptive Moment Estimation) / AdamW (Adaptive Moment Estimation with decoupled Weight decay)、学习率调度与梯度裁剪。以 Adam 为例，记小批次梯度  $g_t = \nabla_{\theta} \mathcal{J}(\theta_t)$ ，一阶 / 二阶动量为

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \quad v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2.$$

经偏差校正  $\hat{m}_t = m_t / (1 - \beta_1^t)$ 、 $\hat{v}_t = v_t / (1 - \beta_2^t)$ ，更新

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon} - \eta \lambda_{\text{wd}} \theta_t \quad (\text{AdamW}).$$

梯度裁剪常取

$$g_t \leftarrow g_t \cdot \min\left(1, \frac{\tau}{\|g_t\|_2}\right),$$

以稳定训练。学习率可采用余弦退火或分段下降；早停以验证集损失为准，避免过拟合。

### 12.2.4 正则化与归一化

过拟合是金融应用的首要风险。 $L_2$  权重衰减在目标函数中加入  $\Omega(\theta) = \frac{1}{2}\|\theta\|_2^2$ ；Dropout 在训练期以概率  $p$  屏蔽神经元、测试期按保留率缩放，近似集成多子网络；批归一化 (Batch-Norm) 在层内做标准化并学习缩放平移参数：

$$\text{BN}(h) = \gamma \cdot \frac{h - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \varepsilon}} + \beta.$$

时间序列场景中，任何标准化 / 降维操作都必须在 {训练窗口内} 拟合参数  $\mu, \sigma$ ，随后 {外推} 到验证 / 测试集，以杜绝信息泄露：

$$\tilde{x}_{t,j} = \frac{x_{t,j} - \mu_j^{(\mathcal{T})}}{\sigma_j^{(\mathcal{T})}}, \quad t > \max(\mathcal{T}).$$

### 12.2.5 序列建模：RNN、LSTM 与 GRU

递归神经网络通过隐藏状态传递记忆。LSTM 的门控机制如下：

$$\mathbf{i}_t = \sigma(W_i \mathbf{x}_t + U_i \mathbf{h}_{t-1} + \mathbf{b}_i), \quad \mathbf{f}_t = \sigma(W_f \mathbf{x}_t + U_f \mathbf{h}_{t-1} + \mathbf{b}_f), \quad (12.5)$$

$$\mathbf{o}_t = \sigma(W_o \mathbf{x}_t + U_o \mathbf{h}_{t-1} + \mathbf{b}_o), \quad \tilde{\mathbf{c}}_t = \tanh(W_c \mathbf{x}_t + U_c \mathbf{h}_{t-1} + \mathbf{b}_c), \quad (12.6)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \quad \mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \quad (12.7)$$

金融序列往往非平稳、突变多，窗口化训练与强正则化是必要前提；在样本有限时，浅层 RNN / 一维卷积往往更稳健。

### 12.2.6 自注意力与 Transformer

自注意力通过相似度加权方式聚合信息。单头注意力机制为

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK'}{\sqrt{d_k}}\right)V, \quad (12.8)$$

可并行建模长依赖。因  $O(T^2)$  的时间 / 内存开销，金融高频 / 长文本常需稀疏注意力或窗口化注意力。时间因果约束通过上三角 mask 实现，确保  $t$  时刻不访问未来信息。

### 12.2.7 表示学习与生成模型

自动编码器（AE）通过重建任务学习压缩表示：

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n \ell(x_i, \psi_{\theta}(\phi_{\theta}(x_i))) + \lambda \Omega(\theta).$$

变分自编码器（VAE）通过最大化证据下界（ELBO）进行优化：

$$\log p_{\theta}(x) \geq \mathbb{E}_{q_{\phi}(z|x)}[\log p_{\theta}(x|z)] - D_{\text{KL}}(q_{\phi}(z|x) \| p(z)).$$

生成对抗网络（GAN）通过极小极大博弈学习数据分布。AE / VAE 适用于异常检测、{特征压缩}；GAN 可用于极端场景生成与压力测试，但训练稳定性与 {评估标准} 需谨慎。

### 12.2.8 概率预测、校准与风险度量

许多金融决策需要 {概率} 而非硬二元分类。对于二分类问题，交叉熵是 {合适的评分规则}；概率校准可采用 Platt 缩放法或保序回归。常用的概率质量评估指标包括 Brier 分数

$$\text{Brier} = \frac{1}{n} \sum_{i=1}^n (y_i - p_i)^2,$$

与期望校准误差（ECE）

$$\text{ECE} = \sum_{b=1}^B \frac{|B_b|}{n} |\text{acc}(B_b) - \text{conf}(B_b)|,$$

其中  $B_b$  为按预测置信度划分的分箱。风险度量方面，分位数网络直接最小化  $\ell_\tau$  拟合 VaR；若将概率用于阈值  $w_t = \mathbf{1}_{\{p_t > \tau\}}$  的仓位决策，含交易成本  $c$  的策略收益为

$$r_{t+1}^\pi = w_t r_{t+1} - c |w_t - w_{t-1}|.$$

年化夏普比率

$$\text{Sharpe} = \frac{\text{E}[r_t^\pi - r_t^f]}{\sqrt{\text{Var}(r_t^\pi - r_t^f)}} \sqrt{T}$$

作为 {经济性可用性} 的核心指标（按使用的数据频率取  $T$  为一年内的期数，如月度数据  $T = 12$ 、周度  $T = 52$ ）。

### 12.2.9 金融任务与建模要点

在单资产择时中，可令网络直接输出  $p_t = \Pr(r_{t+1} > 0 | x_t)$  或  $\hat{r}_{t+1}$ ；在横截面选股中，可将“股票—期限”等特征输入共享网络，输出横截面打分，损失采用排序 / 配对形式以稳定名次；在信用 / 欺诈中，关注极端不平衡与阈值—成本对齐；在文本情绪分析中，建议以预训练语言模型（BERT 类）提取向量 / 情感特征，再与结构化因子拼接后输入下游网络。任何场景都应在 {严格时序化} 的数据处理过程中避免信息泄露，并以滚动 / 扩展窗口进行样本外评估与阈值选择。

### 12.2.10 样本外评估与非平稳性

评估必须按时间顺序：若以  $1 < t_1 < \dots < t_M < T$  为折点，第  $m$  折训练集为  $\{1, \dots, t_m\}$ ，验证 / 测试为  $\{t_m + 1, \dots, t_{m+1}\}$ 。特征标准化、PCA / AE 等变换均需在训练 {窗口} 拟合后外推。部署阶段需监控分布漂移与稳定性，可用人口稳定性指数 (PSI)：

$$\text{PSI} = \sum_b (p_b - q_b) \ln \frac{p_b}{q_b},$$

当 PSI 超过阈值时触发模型再训练或规则回退。必要时可采用 {区块自举法} 评估夏普比率差异的统计显著性。

### 12.2.11 可解释性与经济含义

为缓解“黑箱”担忧，可采用积分梯度（Integrated Gradients, IG）度量边际贡献：

$$\text{IG}_j(x) = (x_j - \tilde{x}_j) \int_0^1 \frac{\partial f(\tilde{\mathbf{x}} + \alpha(\mathbf{x} - \tilde{\mathbf{x}}))}{\partial x_j} d\alpha.$$

配合 SHAP 进行全局 / 局部解释，结合 {部分依赖图} 与 {个体条件期望 (ICE) 曲线} 揭示非线性效应与交互作用。解释的目标并非替代经济学推理，而是在保证样本外稳定性的前提下，验证模型是否捕捉到合理的经济机制。

### 12.2.12 模型治理与合规落地

合规落地要求从数据血缘、特征字典、训练集 / 验证集 / 测试集划分到超参数与随机种子均可复现；评估除统计指标外，应报告收益-风险、回撤、换手率与成本；对违约 / 欺

诈等评估分类任务，需满足公平性与可解释性要求；生产环境中应建立数据质量、漂移与性能退化{监控机制}，并设置应急回退与再训练流程。

不难看出，神经网络提供了强大的非线性函数逼近工具箱：MLP 适合结构化高维数据的非线性整合；LSTM / GRU 聚焦时间依赖；Transformer 擅长长序列与文本；AE / VAE / GAN 支持表示学习与场景生成。将其纳入金融计量框架的关键，是把建模、正则与样本外评估串成闭环，并以经济目标与解释性作为约束。数据越丰富、非线性越强、结构越复杂的子领域（高频、文本、图结构风险传播），深度学习的优势越明显；而在信号稀薄、样本有限的传统任务中，简洁的线性或树模型仍是坚实基线，深度方法应在严格治理与清晰经济假说的约束下使用。

## 12.3 超参数调优与交叉验证：金融计量视角

本节讨论机器学习模型在金融场景中的超参数（hyperparameters）调优与交叉验证（CV）策略。与传统计量方法不同，复杂模型的性能对超参数高度敏感，而金融数据又存在时间依赖、弱信号、微观结构噪声、成本与约束等特点，因此需要将调参过程置于“样本外评估—经济目标—防泄露”的统一框架下。

### 12.3.1 问题形式化与风险估计

设训练样本  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ ，学习器由训练算子  $\mathcal{A}_\lambda$  和超参数向量  $\lambda \in \Lambda \subset \mathbb{R}^d$  决定，得到模型  $f_\lambda = \mathcal{A}_\lambda(\mathcal{D})$ 。令损失函数为  $\ell(\cdot, \cdot)$ ，广义风险为

$$R(\lambda) = \mathbb{E}_{(X, Y)} [\ell(Y, f_\lambda(X))],$$

经验风险为

$$\hat{R}(\lambda; \mathcal{D}) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, f_\lambda(x_i)).$$

交叉验证给出样本外风险估计。以  $K$  折交叉验证为例，将索引集划分为互不相交的验证折  $V_k$  与其互补的训练折  $T_k$ ，对每个  $k$  训练  $f_\lambda^{(-k)} = \mathcal{A}_\lambda(\mathcal{D}_{T_k})$ ，CV 风险估计为

$$\hat{R}_{\text{CV}}(\lambda) = \frac{1}{K} \sum_{k=1}^K \frac{1}{|V_k|} \sum_{i \in V_k} \ell(y_i, f_\lambda^{(-k)}(x_i)).$$

模型选择取  $\hat{\lambda} = \arg \min_{\lambda \in \Lambda} \hat{R}_{\text{CV}}(\lambda)$ ；性能报告应在独立测试集或外层交叉验证（CV）上进行。

### 12.3.2 嵌套交叉验证与时序滚动验证

若在同一验证数据上反复调参会产生验证集过拟合。嵌套交叉验证（nested CV）通过外层折评估泛化性能、内层折进行参数调优：

$$\hat{R}_{\text{nested}} = \frac{1}{K_{\text{out}}} \sum_{k=1}^{K_{\text{out}}} \frac{1}{|V_k^{\text{out}}|} \sum_{i \in V_k^{\text{out}}} \ell(y_i, f_{\hat{\lambda}_k}(x_i)),$$

其中  $\hat{\lambda}_k = \arg \min_{\lambda} \hat{R}_{\text{CV,in}}(\lambda; \mathcal{D}_{T_k^{\text{out}}})$ 。

**时间序列交叉验证（滚动 / 扩展窗口）。** 金融时间序列必须保持因果顺序：训练集来自过去，验证集来自未来。令折点  $1 < t_1 < \dots < t_M < T$ ，第  $m$  折训练集与验证集取

$$\mathcal{D}_m^{\text{train}} = \{1, \dots, t_m\}, \quad \mathcal{D}_m^{\text{val}} = \{t_m+1, \dots, t_{m+1}\}.$$

扩展窗用全部历史数据直至  $t_m$  训练；{滚动窗} 以长度  $W$  的最近历史数据训练。指标取各折的加权平均。

**Purged / Embargo K-fold (防泄露)。** 若标签由未来窗口构造（如  $r_{t:t+h}$ ），训练样本可能与验证标签窗口重叠，需在验证边界两侧清除（purge）重叠样本，并施加禁运（embargo）长度  $e$ ：

$$\mathcal{D}_m^{\text{train}} \leftarrow \mathcal{D}_m^{\text{train}} \setminus \mathcal{O}(\mathcal{D}_m^{\text{val}}), \quad \mathcal{D}_m^{\text{train}} \leftarrow \mathcal{D}_m^{\text{train}} \setminus \text{Embargo}_e(\mathcal{D}_m^{\text{val}}).$$

这里  $\mathcal{O}(\cdot)$  表示与验证标签窗口重叠的训练样本索引集合， $\text{Embargo}_e$  将靠近验证期的长度- $e$  区间从训练中移除，显著降低 {标签重叠—信息泄露} 偏差。

### 12.3.3 超参数搜索：网格、随机、贝叶斯与多臂策略

**网格与随机。** 在高维空间，随机搜索往往更高效（重要维度优先探索，(Bergstra & Bengio 2012)）。实践上常“随机粗搜  $\Rightarrow$  局部细化”。

**贝叶斯优化（高斯过程贯模型优化）。** 设目标为最小化噪声函数  $f(\lambda) = \hat{R}_{\text{CV}}(\lambda)$ 。以高斯过程先验  $\mathcal{GP}(m(\lambda), k(\lambda, \lambda'))$  建模  $f$ ，给定观测  $\mathcal{D}_t = \{(\lambda_j, f_j)\}_{j=1}^t$ ，后验为

$$\mu_t(\lambda) = \mathbf{k}_t(\lambda)' (K_t + \sigma^2 I)^{-1} \mathbf{y}_t, \quad \sigma_t^2(\lambda) = k(\lambda, \lambda) - \mathbf{k}_t(\lambda)' (K_t + \sigma^2 I)^{-1} \mathbf{k}_t(\lambda),$$

其中  $[K_t]_{ij} = k(\lambda_i, \lambda_j)$ 、 $[\mathbf{k}_t(\lambda)]_j = k(\lambda, \lambda_j)$ 。以 {期望改进} (EI) 作为采集函数（最小化情形）：

$$\text{EI}_t(\lambda) = (f^* - \mu_t(\lambda) - \xi) \Phi(z) + \sigma_t(\lambda) \phi(z), \quad z = \frac{f^* - \mu_t(\lambda) - \xi}{\sigma_t(\lambda)},$$

其中  $f^*$  为当前最优值， $\phi$   $\Phi$  为标准正态 {密度函数 / 分布函数}， $\xi \geq 0$  控制“探索—利用”权衡。每轮取  $\lambda_{t+1} = \arg \max \text{EI}_t(\lambda)$  并评估  $f(\lambda_{t+1})$ 。

**Successive Halving / Hyperband** 将预算  $B$  在不同“初始配置数—训练步数”的轨迹上分配，逐轮淘汰劣势配置，保留少数 {优势配置} 进入更大预算。对深网 / 提升树这类“可逐步增加预算”的模型尤其高效。

### 12.3.4 金融目标与自定义损失

金融任务往往不以 {准确率或 MSE} 为最终目标，而是以 {经济指标} 为核心。典型包括：

**成本敏感阈值（信用 / 欺诈）：**若误报 / 漏报成本为  $(c_{\text{FP}}, c_{\text{FN}})$ ，则阈值基线

$$\tau^* = \frac{c_{\text{FP}}}{c_{\text{FP}} + c_{\text{FN}}}.$$

**风险调整收益（择时 / 配置）：**由概率  $p_t$  或预测  $\hat{r}_{t+1}$  映射至仓位  $w_t$ ，且考虑交易成本

$c$  的策略收益

$$r_{t+1}^\pi = w_t r_{t+1} - c |w_t - w_{t-1}|, \quad \text{Sharpe Ratio} = \frac{\mathbb{E}[r_t^\pi - r_t^f]}{\sqrt{\text{Var}(r_t^\pi - r_t^f)}} \sqrt{A},$$

其中  $A$  为年化因子（如  $A = 12$ ）。调参时可直接以 {样本外 Sharpe / Sortino / 回撤约束} 为目标（或将之转化为可微代理目标）。概率用于阈值或连续仓位前，建议做 {校准}（如 Platt 缩放或保序回归）。

### 12.3.5 早停、正则与模型选择

**早停** (early stopping) 将“训练轮数 / 弱学习器数 / 树数”等视为容量控制的超参数，通过验证集选择样本外表现最佳的迭代点并终止训练。本质上，早停属于隐式正则化：在训练误差单调下降的同时，样本外误差先降后升，选择“拐点之前”的模型以抑制方差。实现上，应与内层交叉验证联动：每个内折独立选择早停步数及对应权重，再把外层折的性能作为唯一报告口径，以避免验证集“过拟合”。

**线性 / 平方损失情形：谱滤波视角。** 以最小二乘为例，目标为  $\min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2$ 。令  $X = U\Sigma V'$  为奇异值分解，满步长梯度下降（步长  $\eta < 2/\|X\|_2^2$ ）第  $t$  次迭代解可写为

$$\hat{\beta}^{(t)} = \sum_{i=1}^r \left(1 - (1 - \eta\sigma_i^2)^t\right) \frac{u_i'y}{\sigma_i} v_i,$$

其中  $r = \text{rank}(X)$ 。上式显示早停对第  $i$  个主方向的收缩因子为  $g_t(\sigma_i^2) = 1 - (1 - \eta\sigma_i^2)^t$ ；而岭回归  $\hat{\beta}_\lambda = \arg \min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2 + \frac{\lambda}{2} \|\beta\|_2^2$  的收缩因子为  $g_\lambda(\sigma_i^2) = \sigma_i^2 / (\sigma_i^2 + \lambda)$ 。二者都是随谱值单调递增的“低通滤波器”，因此早停可视为一种谱正则化；在适当的  $t$  与  $\eta$  下， $g_t(\cdot)$  在形状上逼近岭回归的  $g_\lambda(\cdot)$ 。这也解释了“平方损失 + 线性模型时，早停近似  $L_2$  正则”的经验事实。

**梯度提升 / GBDT：逐步加性建模的容量控制。** 梯度提升 (Gradient Boosting Decision Tree, GBDT) 以弱学习器的逐步叠加逼近目标函数，

$$F_M(x) = \sum_{m=1}^M \eta h_m(x), \quad h_m \in \mathcal{H},$$

其中  $\eta$  为学习率、 $M$  为迭代数（树数）。在固定弱学习器类  $\mathcal{H}$  的前提下， $M$  本身刻画了函数类的复杂度（类似  $\ell_1$  型路径选择：更小的  $M$  趋向更稀疏的加性模型）。用验证集选择  $M^*$  的 {早停}，等价于在经验风险最小化上附加对“树数”的结构惩罚。配合较小学习率 (shrinkage) 时，取“大  $M$  + 小  $\eta$  + 早停”通常比“{小  $M$  + 大  $\eta$ ”更稳定，且泛化更好；实现时需记录最佳迭代  $M^*$  并据此导出  $F_{M^*}$ （例如 XGBoost 的 `best_ntree_limit`）。

**神经网络：早停与显式正则的配合。** 深度网络训练中，早停常与权重衰减 ( $L_2$ )、 $L_1$  稀疏化、Dropout、BatchNorm 共同使用。典型做法是在训练集内划分验证集，设置“容忍步数” (patience) 与最小改进阈值  $\delta$ ：若验证指标在连续  $P$  轮内未改善超过  $\delta$  则终止训练，并恢复至最佳权重。当验证曲线平台期较长或噪声较大时，可结合学习率调度（如 `ReduceLROnPlateau`）先降低学习率，若仍无改善再触发早停。早停点应当作为超参数选择的结果记录下来，不能在外层循环中继续“观望”，以免信息泄露。

**一标准误原则 (one-standard-error rule)**。深度网络训练中，早停常与权重衰减 ( $L_2$ )、 $L_1$  稀疏化、Dropout、BatchNorm 共同使用。典型做法是在训练集内划分验证集，设置“容忍步数”(patience) 与最小改进阈值  $\delta$ ：若验证指标在连续  $P$  轮内未改善超过  $\delta$  则终止训练，并恢复至最佳权重。当验证曲线平台期较长或噪声较大时，可结合学习率调度（如 ReduceLROnPlateau）先降低学习率，若仍无改善再触发早停。早停点应当作为超参数选择的结果记录下来，不能在外层循环中继续“观望”，以免信息泄露。

**时序金融的实施细节。**在时间序列或“横截面-时间”面板中，早停与模型选择必须在因果正确的切片内完成：外层滚动 / 扩展窗口仅用于最终评估；内层窗口完成所有调参与早停；若标签涉及未来窗口（如  $t : t + h$  的收益），训练样本需对验证期施加 purge / embargo（清除重叠样本并施加禁运期）；阈值、成本与校准也应在内层确定。建议将“早停触发逻辑、最佳轮数  $t^*$ 、随机种子、学习率轨迹”一并固化在管道中，便于复现与审计。

**与显式正则的协同。**早停与显式正则化（岭回归 / Lasso 回归、权重衰减、树结构惩罚参数  $\gamma$ 、最小叶节点权重 min\_child\_weight、样本 / 特征子采样、最大深度等）并不互斥。经验表明：

- 线性与广义线性： $L_2$  控制范数，早停作为 {谱正则化方法} 补充； $L_1$  引入稀疏性。
- GBDT / XGBoost：在较小  $\eta$  下，配合子采样 (subsample, colsample\_bytree)、深度 / 叶大小与结构惩罚 ( $\gamma$ ) 先稳定单步方差，再用早停选  $M^*$ 。
- 深度网络：权重衰减与 Dropout 稳定表示学习，早停决定 {拟合到何种程度}；学习率调度与早停顺序：先调度再停止。

**度量与选择的匹配。**早停的判定指标应与最终经济目标一致：用于二分类的阈值决策应以校准后的 AUC / PR-AUC / KS 或成本敏感损失为准；择时 / 配置任务可直接以样本外 Sharpe / Sortino / 最大回撤控制为判据（并在内层优化中计入交易成本与滑点）。若训练目标与判据不同（如训练最小化对数损失但模型选择依据 Sharpe 比率），需在管道中显式区分“训练损失”与“早停指标”。

**双降现象与风险提示。**在某些模型与数据尺度下，测试误差可能出现“双降”曲线——即在插值阈值附近先上升后下降。早停通常停止在第一谷附近，但未必达到全局最优；因此，建议将早停纳入内层搜索（例如与学习率、深度、正则化强度等超参数联合搜索），并基于外层折的分布与区块自助法 (block bootstrap) 评估差异显著性，而非仅报告单点改进结果。

**实现准则 (实践小抄)。**训练时把“步数-验证指标”曲线记录为时间序列，并使用 {滑动窗口} 平滑噪声后判定是否改善；设定最小改进  $\delta$  与容忍步数  $P$ ，防止误触发；对小样本内层折增大  $P$ 、减小  $\delta$ ；为避免“反复窥视”验证集，外层严格一次性评估并冻结早停点。对于提升树 / 深度网络，推荐“小学习率 + 大最大轮数 + 早停”的组合；对于需要概率用作阈值或连续仓位的任务，在内层先做概率校准 (Platt / 保序回归) 再判定早停。

综上，早停既是一个容量控制器，也是一种模型选择机制：它通过验证集在“训练—泛化”的权衡曲线上选择拐点前的模型；对平方损失—线性模型，它等价于谱滤波（近似岭回归）；对提升树与深度网络，它与显式正则共同决定函数类的有效复杂度。只要将早停放在嵌套时序交叉验证的内层，并与经济度量、成本与校准保持一致，就能在金融数据的弱信号与非平稳环境下获得稳健的样本外表现。

### 12.3.6 避免过度搜索与检验偏差

在高维模型与大规模调参的语境下，回测过拟合与多重比较是两类最常见且最隐蔽的偏差来源。所谓回测过拟合，是指研究者在同一数据集上反复试参、筛选与再筛选，最终挑出一个在历史样本上看似“最佳”的配置，但该优势并不具有可复现的样本外有效性；多重比较则是指在比较众多候选模型或策略时，单个显著性检验的标称显著性水平不再可靠，容易把本应归因于偶然波动的“胜出者”误判为真信号。下述做法可以系统地缓解这些问题。

首先，**外层（嵌套）评估**是防止验证集被“用坏”的首要手段。实操时将时间轴切成若干外层滚动/扩展窗口：每个外层窗口内，仅在训练子窗上完成所有调参（含早停、阈值、正则强度等），随后在外层验证子窗上只评估一次并记录结果；合并所有外层验证窗即可得到稳健的样本外分布。对时间序列，应配合 `purge/embargo`（清除与验证标签窗口重叠的训练样本，并在验证期边界施加禁运期），从而打断由标签构造引起的区间重叠与信息泄露。嵌套评估的核心是把“选型”（内层）与“报告”（外层）分离：外层不再做任何选择，只做一次性打分。

其次，**CSCV/PBO 方法** (Combinatorially Symmetric Cross-Validation / Probability of Backtest Overfitting) 可量化“最佳策略是否仅为运气所致”。做法是把全样本划分为  $S$  个等长子段，枚举所有对称的“训练/测试”划分（例如每次选取  $S/2$  个子段训练、其余测试），在每个划分上用训练段选出“样本内最佳”策略，并比较它在对应测试段的相对排名。将这些相对排名映射为对数几率后得到一组统计量，其落在负半轴（样本外表现劣于中位数）的比例即为 PBO。若 PBO 接近  $1/2$  或更高，表明“样本内最佳策略在样本外经常垫底”，回测过拟合风险很大；若 PBO 明显低于  $1/2$ ，则支持存在可复现的样本外优势。CSCV/PBO 的优势在于：它不依赖单一切片的偶然性，而是在大量对称切分上考察“最佳者”的稳定性。

再次，**现实检验（White's Reality Check）与 SPA** (Superior Predictive Ability) 可在多模型同时比较时给出经“数据窥探”调整的显著性。其基本框架是假设检验

$$H_0 : \sup_{k \leq m} E[d_{t,k}] \leq 0,$$

其中  $d_{t,k}$  是第  $k$  个候选相对某基准（如买入并持有、零权重）的损失差/效用差。统计量通常取最大平均差异，并用（平稳）区块自助法在时间依赖下近似其分布，从而得到针对“最好者”的家族层面  $p$  值。White 的现实检验在原理上稳健但保守，SPA 在此基础上对“明显劣势者”进行截尾与重心调整，改善有限样本下的尺寸与功效。在落地时，应明确：基准是什么（零收益、基准组合或业务基线）、损失差如何定义（负收益、负 Sharpe 的估计对立项、成本后净收益等）、以及自助法的区块长度如何选取（可用自动带宽）。

此外，当候选数量众多时，**FDR 控制**（假发现率）比“逐一检验 + Bonferroni 校正”更切实可用。Benjamini–Hochberg (BH) 过程的做法是：将  $m$  个  $p$  值从小到大排序为  $p_{(1)} \leq \dots \leq p_{(m)}$ ，找到最大的  $k$  使

$$p_{(k)} \leq \frac{k}{m} q,$$

则拒绝  $p_{(1)}, \dots, p_{(k)}$  并保证在独立或正相关条件下  $FDR \leq q$ 。若检验间相关性更复杂，可用 Benjamini–Yekutieli (BY) 修正（其中  $q/(mc_m)$ ,  $c_m = \sum_{i=1}^m 1/i$ ）或直接报告 q-value。当模型家族存在分层结构（如若干算法  $\times$  多种特征集  $\times$  多阈值），可采用层级 FDR：先控制家族层显著性，再在进入下一层时局部控制 FDR，既维持发现率又抑制“假阳性瀑布”。

最后，从研究流程的角度，还可以通过预注册/冻结规则来抑制“研究者自由度”。在项目伊始明确：目标指标与业务约束、时间切片与数据保护期、搜索预算（配置数或评估次数上限）、候选家族与可调整的超参数边界；将所有试验与版本纳入“模型清单”（含时间戳与随机种子），避免只报告胜出的那个；在最终报告中给出外层折的全分布（而非单一均值）与区块自助法的置信区间/显著性检验结果，并说明尝试过的候选总数与筛选规则。上述做法与嵌套评估、CSCV/PBO、现实检验/SPA、FDR 控制相结合，能够将“好看”的回测优势尽量转化为可复现、可审计、可落地的样本外证据。

### 12.3.7 随机森林、XGBoost、深度网络的调参要点

**随机森林：**控制容量的关键是 `mtry/max_depth/min_samples_leaf`；优先用 OOB 误差做粗筛，再用时序 CV（滚动/扩展）微调。

**XGBoost/GBDT：**以 `eta`(学习率)+`n_rounds`(早停)为骨架，配合 `max_depth/min_child_weight/subsample` 控制方差；`scale_pos_weight` 处理类不平衡数据。

**深度网络：**通过层数/宽度、学习率调度、权重衰减、Dropout、BatchNorm 与早停 联动控制容量；时序任务优先采用滚动/扩展窗口 + `purge/embargo` 策略；小样本优先使用浅层网络与强正则化。

### 12.3.8 一个面向时序的调参与评估流程（建议）

1. 固定外层滚动/扩展窗口切片（含 `purge/embargo`），作为最终报告口径。
2. 在每个外层训练窗内，运行内层滚动交叉验证 (CV): (a) 随机 / 贝叶斯 / Hyperband 搜索超参数；(b) 模型自带早停在内层确定；(c) 以经济目标（夏普比率 / 成本敏感）或校准后指标作为模型选型准则。
3. 将内层最优超参数在外层训练窗上重新训练一次，外层验证窗上仅评估一次并记录经济与统计指标（含回撤、换手率、交易成本）。
4. 合并所有外层折的样本外结果，报告均值 / 分布，并（可选）采用块自举法评估差异显著性。

### 12.3.9 实践附注：评价指标与报告规范

除 AUC/MSE 外，应报告：年化收益率、波动率、Sharpe/Sortino、最大回撤、平均仓位与年化换手（成本暴露）、阈值/成本设定与校准方法、时间切片与 `embargo` 参数、搜索预算与采集策略、随机种子与版本，确保可复现与可审计。

金融中的调参与交叉验证不只是“寻找最小验证误差”，而是要在因果正确的时序评估中、围绕经济目标、在受控搜索预算下完成模型选择与不确定性度量。网格 / 随机 / 贝叶斯 / 多臂赌博机策略各有侧重；早停与正则化共同控制容量；`purge/embargo`、嵌套交叉验证与外层一次性评估则是防泄露与防过度搜索的根本手段。遵循以上流程，才能让高维机器学习在金融计量中“既好看又能用”。

## 12.4 案例：AI 模型的典型应用场景

### 12.4.1 收益率预测：线性回归 vs. XGBoost 与 LSTM

基准思路是从“可解释性—稳健性”的线性族出发，再逐步引入能够刻画非线性与时序记忆的树集成与序列模型。在线性端，取 OLS/岭回归（Ridge）作为基线，以宏观与期限结构因子（如 VIX 水平/变动、10Y–3M 期限利差的水平/斜率、信用利差替代变量）与简单的技术指标（6/12 个月动量、3 个月波动率、均线偏离度等）为解释变量，构造下一期（月度）对数收益率的点预测；岭回归通过  $L_2$  约束在“小样本-弱信号”下提升稳健性。非线性端，XGBoost 通过分段常数函数 + 剪枝/结构惩罚自然捕捉交互项与非线性效应，在不改变特征组合的前提下提高 RMSE 与方向预测准确率；序列端，LSTM/Transformer 借助门控/自注意力机制显式建模长记忆与状态依赖，对“环境切换、慢变量-快反应”的市场结构尤为敏感。评估口径坚持滚动/前进（walk-forward）时序验证与样本外误差（RMSE）/方向正确率，并将现实检验（简单仓位、交易成本、基点敏感度）纳入同一管道，避免“纸面优异、交易失效”。实现上，所有特征统一滞后一期以杜绝前视偏差，训练—验证—测试严格按时间推进：岭回归在内层验证选  $\lambda$ ；XGBoost 以小学习率 + 早停法选最佳迭代数；LSTM 以 `patience`（耐心值）控制容量；阈值/成本及（若适用）概率校准亦在内层确定，外层仅一次性报告样本外指标与经成本调整后的业绩。

为将上述口径落实到可复现的工作流，下面给出一份 R 实证脚手架的衔接说明与关键片段。脚本从 Yahoo Finance 抓取 SPY（标的）、^VIX（波动代理）、^TNX/^IRX（期限结构）、HYG/LQD（信用利差代理），按月末对齐并构造动量（6/12 个月）、波动（3 个月）、均线偏离、VIX 水平与变动、期限利差水平与变动、信用利差变动等特征；目标为下一个月对数收益，所有特征滞后一月。评估采用滚动窗口步进（默认训练窗口 120 个月）；岭回归用内层验证选择  $\lambda$ ；XGBoost 用早停法（小  $\eta$ 、`max_depth`、`min_child_weight`、`subsample`/`colsample_bytree`）确定最佳迭代；LSTM 为可选项（需 `keras` 环境，样本少时可缩短序列长度）。现实检验以“预测  $> 0$  则持有、否则持有现金”为简易仓位，在 {0, 5, 10} bp 单边成本下生成月度净值与年化收益/波动/夏普比率（`nav_all/sum_all`）。为确保汇总不被“空模型”污染，代码在合并样本外结果后利用 `keep_models` 自动剔除无有效预测的模型，并以 `metric()` 在存在缺失时稳健计算 RMSE/方向准确率；同时将 XGBoost 预测接口更新为 `iteration_range`。这一处理与本节提出的“滚动窗口步进验证 + 样本外误差 + 方向正确率 + 含成本现实检验”一致，亦便于后续加入阈值内层优化（直接最大化样本外夏普比率）、概率校准与连续仓位映射，或引入 `purge/embargo` 与嵌套交叉验证，获得更稳健与可重复的样本外证据。

```

1 # 在 oos <- bind_rows(oos) 之后插入：
2 pred_cols <- c("OLS", "RIDGE", "XGB", "LSTM")
3 avail <- sapply(pred_cols, function(cn) sum(!is.na(oos[[cn]])))
4 keep_models <- names(avail[avail > 0]) # 只保留有预测的模型
5
6 # 计算指标时也要更稳健
7 metric <- function(pred){
8 ok <- !is.na(pred) & !is.na(oos$y)
9 if (!any(ok)) return(c(RMSE = NA, DirAcc = NA))
10 rmse <- sqrt(mean((oos$y[ok] - pred[ok])^2))
11 dir <- mean(sign(oos$y[ok]) == sign(pred[ok]))
12 c(RMSE = rmse, DirAcc = dir)
13 }
14
15 # 汇总表
16 res_tab <- do.call(rbind, lapply(keep_models, function(m) metric(oos[[m]])))
17 rownames(res_tab) <- keep_models

```

```

18 print(round(res_tab, 4))
19
20 # 现实检验循环里把模型列表替换为有预测的模型
21 nav_list <- list(); sum_list <- list()
22 for (mdl in keep_models) {
23 for (tc in bp_set) {
24 nav <- make_nav(oos[[mdl]], tc)
25 nav$Model <- mdl; nav$TC <- paste0(tc*1e4, " bp")
26 nav_list[[length(nav_list)+1]] <- nav
27
28 s <- round(summ(nav), 4)
29 sum_list[[length(sum_list)+1]] <- tibble(
30 Model = mdl, TC = paste0(tc*1e4, " bp"),
31 AnnRet = s[["AnnRet"]], AnnVol = s[["AnnVol"]], Sharpe = s[["Sharpe"]]
32)
33 }
34 }
35 nav_all <- dplyr::bind_rows(nav_list)
36 sum_all <- dplyr::bind_rows(sum_list)
37 print(sum_all)

```

### 12.4.2 波动率建模：GARCH vs. 注意力机制

本节从“结构可解释—统计可检验”的 GARCH 家族出发，过渡到能够表达长记忆与非线性的注意力/深度序列方法，并给出一套可复现的比较脚手架。在线性—条件异方差端，GARCH/EGARCH/GJR 通过对收益的条件方差建模捕捉“冲击—记忆—杠杆”机制，参数具有明确的经济含义且便于残差诊断与稳健性检验；但当目标切换为实现波动率（由高频价格构造）或存在多尺度记忆与异质行为时，单一的条件异方差结构可能偏刚性。此时，一方面可以采用 HAR-RV (log) 将日/周/月实现波动的多尺度信息以线性形式汇总，既贴合“长记忆”又保持可解释性；另一方面，可引入 LSTM/注意力直接在序列层面学习非线性与跨尺度依赖，或做混合：先用 GARCH 抽取条件波动作为“可解释的主干”，再由注意力去学习剩余结构，从而在解释与拟合之间取得平衡。

为将上述思路落地，本节的 R 脚本统一以对数实现波动为预测目标：先用 OHLC 计算 Garman-Klass 日度实现波动 (GK)，再取 log 以稳定方差；比较对象包含三类——(1) GARCH(1,1) 采用 ugarchroll 做逐日滚动预测并设 refit.every=5 (低频重训、日内状态递推)，避免“台阶式”预测并显著提高样本外相关；(2) HAR-RV (log) 以日/周/月三尺度的 log RV 为解释变量，滚动估计与一步预测，作为长记忆线性基线；(3) LSTM/注意力/混合提供“快速模式”（默认关闭）：仅用最近 keep\_last 天、短序列 L\_seq 与低频重训 retrain\_every，在有限算力下演示序列模型或叠加 GARCH 条件波动特征的混合方案。评估口径坚持滚动样本外 RMSE 与相关系数，并绘制“最近 500 天”的真实 vs 预测曲线直观比对；速度—精度的旋钮（如 keep\_last、n\_out、refit.every、L\_seq）在代码中显式给出，便于读者按需调整。需要更贴近极端波动分布时，可将 GARCH 替换为 EGARCH/GJR 并采用学生  $t$  分布；需要更强的线性基线时，可扩展为 HAR-X (在 d/w/m 上叠加 VIX/信用利差等)；当拥有更丰富的高频实现特征与状态变量，并配合规范的内层调参与早停，注意力/混合模型的增益会更稳定地体现。

```

1 # =====
2 # 波动率建模: GARCH(roll) vs HAR-RV(log) vs LSTM(+Attention, 可选)

```

```

3 # 数据: Yahoo Finance (SPY, ^VIX), 频率: 日度
4 # 目标: 下一期“对数”GK 实现波动 (log-Vol)
5 # 评估: 样本外 RMSE / 相关系数; 可视化最近 500 天
6 # 备注: GARCH 使用 ugarchroll (每天更新, 5 天重训), HAR 为 d/w/m 线性基线。
7 # LSTM 默认关闭 (教材快速版); 需要可设 run_lstm <- TRUE。
8 # =====
9
10 # ----- 参数 -----
11 start_date <- as.Date("2010-01-01")
12 end_date <- as.Date("2024-12-31")
13 set.seed(2025)
14
15 # 深度部分 (需 keras/tensorflow; 教材默认关闭)
16 run_lstm <- FALSE # 设 TRUE 演示 LSTM
17 run_hybrid <- FALSE # LSTM + GARCH 特征, 可与 run_lstm 搭配
18
19 # LSTM 快速模式
20 L_seq <- 20
21 train_win <- 800
22 val_frac <- 0.2
23 epochs <- 40
24 patience <- 8
25 retrain_every <- 30
26
27 # GARCH 快速模式: 仅保留最近 keep_last 天, 以加速
28 keep_last <- 1200
29 # ugarchroll 设置
30 n_out <- 600 # 仅评估最后 600 天
31 refit_every <- 5 # 每 5 天重训一次 (速度/精度折中)
32
33 # ----- 依赖 -----
34 pkgs <- c("quantmod", "rugarch", "dplyr", "tidyverse", "lubridate", "ggplot2", "zoo")
35 for (p in pkgs) if (!requireNamespace(p, quietly = TRUE)) install.packages(p)
36 invisible(lapply(pkgs, library, character.only = TRUE))
37
38 if (run_lstm || run_hybrid) {
39 if (!requireNamespace("keras", quietly = TRUE)) install.packages("keras")
40 library(keras)
41 # 首次: keras::install_keras()
42 }
43
44 options(stringsAsFactors = FALSE, scipen = 99, timeout = max(300,getOption(
45 "timeout")))
46
47 # ----- 1) 获取日度数据 & 目标 (GK → log-Vol) -----
48 get_ok <- function(sym){
49 tryCatch({
50 suppressWarnings(getSymbols(sym, src = "yahoo",
51 from = start_date - 30, to = end_date, auto.
52 assign = FALSE)))

```

```

51 }, error = function(e) NULL)
52 }
53
54 spy <- get_ok("SPY"); vix <- get_ok("^VIX")
55 if (is.null(spy) || is.null(vix)) stop("下载数据失败。")
56
57 # 对齐交易日
58 idx <- sort(intersect(index(spy), index(vix)))
59 spy <- spy[idx]; vix <- vix[idx]
60
61 # 日度对数收益
62 ret <- diff(log(Ad(spy)))
63
64 # Garman - Klass 日方差与波动
65 O <- Op(spy); H <- Hi(spy); L <- Lo(spy); C <- Cl(spy)
66 GKvar <- 0.5 * (log(H/L))^2 - (2*log(2)-1) * (log(C/O))^2
67 GKvar <- GKvar[index(ret)]
68 GKvol <- sqrt(pmax(as.numeric(GKvar), 0))
69
70 # VIX 收盘 (近似隐含波动水平)
71 vix_cl <- Cl(vix)[index(GKvol)]
72 vix_cl <- zoo::na.locf(vix_cl)
73
74 # 基础数据框
75 df <- tibble(
76 date = as.Date(index(GKvol)),
77 r = as.numeric(ret),
78 abs_r = abs(as.numeric(ret)),
79 r2 = as.numeric(ret)^2,
80 GKvol = as.numeric(GKvol),
81 VIX = as.numeric(vix_cl)
82) |>
83 drop_na() |>
84 mutate(
85 y_lin = dplyr::lead(GKvol, 1),
86 y_log = dplyr::lead(log(pmax(GKvol, 1e-8)), 1)
87) |>
88 drop_na()
89
90 # 教材快速：仅用最近 keep_last 天
91 df_q <- tail(df, keep_last)
92 n_q <- nrow(df_q)
93
94 # ----- 2) GARCH(1,1): ugarchroll 日度滚动 + 低频重训 -----
95 spec_garch <- rugarch::ugarchspec(
96 variance.model = list(model = "sGARCH", garchOrder = c(1,1)),
97 mean.model = list(armaOrder = c(0,0), include.mean = TRUE),
98 distribution.model = "norm"
99)
100 roll <- ugarchroll(

```

```

102 spec = spec_garch, data = df_q$r,
103 n.ahead = 1, forecast.length = n_out,
104 refit.every = refit_every, # 每 refit_every 天重训
105 refit.window = "moving",
106 solver = "hybrid", solver.control = list(trace = 0),
107 calculate.VaR = FALSE
108)
109 roll_df <- as.data.frame(roll) # 包含 Sigma 等列 (长度 = n_out)
110
111 pred_garch <- rep(NA_real_, n_q)
112 garch_idx <- (n_q - n_out + 1):n_q
113 pred_garch[garch_idx] <- log(pmax(roll_df$Sigma, 1e-8)) # 与 y_log 量纲一致
114
115 # ----- 3) HAR-RV(log) 基线 (d / w / m) 滚动 -----
116 har_df <- df_q |>
117 mutate(
118 RVd = log(pmax(GKvol, 1e-8)),
119 RVw_tmp = zoo::rollapply(GKvol, 5, mean, align = "right", fill = NA),
120 RVm_tmp = zoo::rollapply(GKvol, 22, mean, align = "right", fill = NA),
121 RVw = log(pmax(dplyr::lag(RVw_tmp), 1e-8)),
122 RVm = log(pmax(dplyr::lag(RVm_tmp), 1e-8))
123) |>
124 select(date, y_log, RVd, RVw, RVm) |>
125 drop_na()
126
127 pred_har <- rep(NA_real_, nrow(har_df))
128 start_h <- 600 # 起始滚动点 (可调); 也可用 train_win+1
129 for (i in start_h:(nrow(har_df)-1)) {
130 tr <- (i - 500):(i - 1) # HAR 训练窗 (约两年), 教材取小窗加速
131 fit <- lm(y_log ~ RVd + RVw + RVm, data = har_df[tr,])
132 pred_har[i] <- as.numeric(predict(fit, newdata = har_df[i,]))
133 }
134
135 # ----- 4) LSTM / 注意力 (快速模式, 可选) -----
136 has_mha <- run_lstm && "layer_multi_head_attention" %in% ls(getNamespace("keras"))
137
138 scale_fit <- function(M) list(mu = colMeans(M), sd = pmax(apply(M, 2, sd),
139 1e-8))
140 scale_apply<- function(M, stat) sweep(sweep(M, 2, stat$mu, "-"), 2, stat$sd,
141 "/")
142 make_seq <- function(X, y, L){
143 n <- nrow(X); p <- ncol(X)
144 if (n <= L) return(NULL)
145 Xs <- array(0, dim = c(n - L, L, p))
146 ys <- y[(L+1):n]
147 for (k in 1:(n - L)) Xs[k,,] <- X[k:(k+L-1),]
148 list(X = Xs, y = ys)
149 }
150 build_lstm_model <- function(L, p, use_mha = FALSE){
151 inputs <- layer_input(shape = c(L, p))

```

```

150 x <- inputs |>
151 layer_lstm(units = 32, return_sequences = TRUE) |>
152 layer_dropout(0.1)
153 if (use_mha) {
154 x <- layer_multi_head_attention(num_heads = 4, key_dim = 16,
155 dropout = 0.0)(list(x, x, x)) |>
156 layer_layer_normalization()
157 }
158 x <- x |>
159 layer_lstm(units = 16) |>
160 layer_dropout(0.1) |>
161 layer_dense(units = 1)
162 model <- keras_model(inputs = inputs, outputs = x)
163 model |>
164 compile(optimizer = optimizer_adam(learning_rate = 0.005), loss = "mse")
165 model
166 }
167
168 pred_lstm <- rep(NA_real_, n_q)
169 if (run_lstm || run_hybrid) {
170 base_feats <- c("GKvol", "abs_r", "r2", "VIX")
171 need_train <- TRUE
172 start_i <- max(train_win + 1, L_seq + 1)
173 for (i in start_i:(n_q-1)) {
174 need_train <- need_train || ((i - start_i) %% retrain_every == 0)
175 tr_idx <- (i - train_win):(i - 1)
176
177 X_tr <- as.matrix(df_q[tr_idx, base_feats])
178 y_tr <- df_q$y_log[tr_idx]
179
180 if (run_hybrid) {
181 fit_h <- try(ugarchfit(spec = spec_garch, data = df_q$r[tr_idx],
182 solver="hybrid", solver.control=list(trace=0)),
183 silent = TRUE)
184 if (!inherits(fit_h, "try-error")) {
185 X_tr <- cbind(X_tr, GARCH = as.numeric(sigma(fit_h)))
186 }
187 stat <- scale_fit(X_tr)
188 Xs_tr <- scale_apply(X_tr, stat)
189
190 n_tr <- nrow(Xs_tr)
191 val_n <- max(50, round(n_tr * val_frac))
192 core_n<- n_tr - val_n
193 X_core<- Xs_tr[1:core_n, , drop = FALSE]
194 y_core<- y_tr[1:core_n]
195 X_val <- Xs_tr[(core_n+1):n_tr, , drop = FALSE]
196 y_val <- y_tr[(core_n+1):n_tr]
197
198 seq_core <- make_seq(X_core, y_core, L = L_seq)
199 seq_val <- make_seq(rbind(X_core[(nrow(X_core)-L_seq+1):nrow(X_core), ,

```

```

drop=FALSE], X_val),
200 c(tail(y_core, L_seq), y_val),
201 L = L_seq)

202
203 X_te <- as.matrix(df_q[(i-L_seq):(i-1), base_feats])
204 if (run_hybrid && exists("fit_h") && !inherits(fit_h, "try-error")) {
205 gsig_te <- as.numeric(sigma(ugarchforecast(fit_h, n.ahead = 1)))
206 X_te <- cbind(X_te, GARCH = c(tail(as.numeric(sigma(fit_h)), L_seq-1),
207 gsig_te))
208 }
209 X_te <- scale_apply(X_te, stat)
210 X_te <- array(X_te, dim = c(1, L_seq, ncol(X_te)))

211 if ((run_lstm || run_hybrid) && !is.null(seq_core) && !is.null(seq_val)) {
212 if (need_train || !exists("model_lstm_q")) {
213 model_lstm_q <- build_lstm_model(L_seq, ncol(Xs_tr), use_mha = (run_lstm && has_mha))
214 }
215 fit <- try(
216 model_lstm_q |> fit(
217 x = seq_core$X, y = seq_core$y,
218 validation_data = list(seq_valX, seq_valy),
219 epochs = epochs, batch_size = 32, verbose = 0,
220 callbacks = list(callback_early_stopping(monitor="val_loss",
221 patience=patience,
222 restore_best_weights=TRUE
223))
224), silent = TRUE
225)
226 if (!inherits(fit, "try-error")) {
227 pred_lstm[i] <- as.numeric(predict(model_lstm_q, X_te))
228 }
229 need_train <- FALSE
230 }
231 }
232
233 # ----- 5) 指标与可视化 -----
234 oos_df <- df_q |>
235 select(date, y_true = y_log) |>
236 left_join(tibble(date = df_q$date, GARCH = pred_garch), by = "date") |>
237 left_join(tibble(date = har_df$date, HAR = pred_har), by = "date") |>
238 left_join(tibble(date = df_q$date, LSTM = pred_lstm), by = "date") |>
239 drop_na(y_true)

240 mdl_cols <- c("GARCH", "HAR", "LSTM")
241 mdl_cols <- mdl_cols[colSums(!is.na(oos_df[mdl_cols])) > 0]
242
243 metric <- function(y, yhat){
244 ok <- !is.na(y) & !is.na(yhat)

```

```

246 if (!any(ok)) return(c(RMSE = NA, Corr = NA))
247 rmse <- sqrt(mean((y[ok] - yhat[ok])^2))
248 corr <- suppressWarnings(cor(y[ok], yhat[ok]))
249 c(RMSE = rmse, Corr = corr)
250 }
251
252 res_tab <- do.call(rbind, lapply(mdl_cols, function(m) metric(oos_df$y_true,
253 oos_df[[m]])))
254 rownames(res_tab) <- mdl_cols
255 print(round(res_tab, 4))
256
257 # 可视化：真实 vs 预测（最近 500 天）
258 plt_df <- oos_df |>
259 filter(row_number() >= n() - 500) |>
260 tidyr::pivot_longer(all_of(c("y_true", mdl_cols)),
261 names_to = "Series", values_to = "Value") |>
262 dplyr::group_by(Series) |>
263 tidyr::drop_na(Value) |>
264 dplyr::ungroup()
265
266 ggplot(plt_df, aes(date, Value, color = Series)) +
267 geom_line(linewidth = 0.8) +
268 labs(title = "日度 log-Vol: 真实 (GK) 与模型预测 (最近 500 天)",
269 x = NULL, y = "log(Vol)") +
270 theme_minimal() +
271 theme(legend.position = "bottom")

```

### 12.4.3 信用评分：Logit vs. 随机森林 / 神经网络

信用评分是将个人或企业的可观察信息（人口属性、交易行为、负债与资产、征信记录等）映射为违约风险度量的统计/计量框架。狭义上，它产出“是否违约”的概率 (PD)，并常通过评分卡把对数赔率映射为分值以便运营；广义上，PD 与损失给定违约 (LGD)、违约时风险暴露 (EAD) 共同决定预期损失  $EL = PD \times LGD \times EAD$ 。在监管与会计口径下，PD 直接影响资本与减值：巴塞尔 IRB 框架下资本需求随 PD 单调变化；IFRS 9 以全期限预期信用损失为基准，需要一条随时间与宏观情景变化的 PD 曲线。微观层面，评分支撑授信审批、额度与利率定价、欺诈拦截、催收分流与账户监控；宏观层面，评分体系的稳健性与可解释性关乎银行资产质量、逆周期缓冲与宏观审慎。

在模型选择上，Logit 是最常用、最易治理的基线。设  $y_i \in \{0, 1\}$  表示是否违约， $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})'$  为特征，假定  $y_i | \mathbf{x}_i \sim \text{Bernoulli}(p_i)$ ，并令

$$\text{logit}(p_i) = \log \frac{p_i}{1 - p_i} = \eta_i = \beta_0 + \mathbf{x}'_i \boldsymbol{\beta}, \quad p_i = \frac{1}{1 + \exp(-\eta_i)}.$$

系数含义直观： $x_{ij}$  每增加 1 单位，对数赔率增加  $\beta_j$ ，对应的赔率比为  $\exp(\beta_j)$ 。参数通常用极大似然估计；评分卡把  $\hat{p}$  映射为分值  $\text{Score} = A - B \cdot \text{logit}(\hat{p})$ ，并用 PDO 约束确定  $A, B$ 。工程实现中，可配合哑变量/WOE 与单调分箱增强稳健性与可解释性；出现完全分离时可采用正则化或 Firth 修正；样本不平衡可设类权重或再抽样，但需在校准阶段做先验纠偏。尽管理论上 Logit 直接输出概率，受再抽样、类权重或漂移影响，概率校准 (Platt / 保序回归) 在实务中仍是必要环节。

在非线性与交互较强、或类别变量基数较大时，随机森林/浅层神经网络可以在保持速度与可控复杂度的前提下提升排序能力（AUC/KS）与召回。生产上常见的折中是“Logit 主体 + 树模型辅助”的两段式架构：用树模型做变量筛选与衍生（分裂点启发、交互候选），再用 Logit 固化为“可解释评分卡”；在边缘群体或灰名单引入二级 AI 复核，形成“稳态评分 + 局部提升”。全流程除排序外，还需关注概率质量（Brier、校准曲线）、成本敏感阈值（按误报/漏报成本选  $\tau$ ）、稳定性/漂移（PSI/CSI）与公平性（分组 AUC/TPR 差异与阈值策略），以及拒绝推断（仅有通过样本时的选择偏差修正）。

为把上述思路落地，本节提供一份“下载—建模—校准—评估”的 R 脚本。脚本优先在线获取 UCI German Credit，失败则回退 `caret::GermanCredit`；按 60/20/20 分层切分训练/验证/测试，对类别变量做哑变量、对数值做标准化；训练三类模型——Logit（基线，可映射为评分卡雏形）、随机森林（`ranger`，概率输出）与浅层神经网络（`nnet`，小隐层与  $L_2$  衰减）。在验证集进行概率校准（默认 Platt，亦预留保序接口），将校准器应用于测试集，报告样本外 AUC/KS、PR-AUC、Brier 与校准曲线；依据给定的误报/漏报成本计算成本敏感阈值并输出混淆矩阵；同时计算 PSI（训练 → 测试的分布漂移）与按年龄分组的 AUC/TPR 差异以检视稳定性与公平性。脚本中的关键旋钮（树的数量、`min.node.size`、神经网络的 `size` 与 `decay`、校准方式、成本比与阈值等）均显式给出，便于复现实验或迁移到本地信贷特征集。需要强调的是，示例数据属于“已通过样本”，未包含拒绝记录；若用于生产迁移，应在建模阶段嵌入拒绝推断（如分层再权重、EM/半监督、parceling 等），并在模型治理层面固定分箱/评分映射、阈值与监控指标，以满足审计与合规要求。

```

1 # =====
2 # 信用评分：Logit vs 随机森林 / 神经网络（含下载、校准、稳定性/公平性）
3 # 数据：优先在线下载 UCI German Credit，失败回退 caret::GermanCredit
4 # 评估：AUC/KS/PR-AUC/Brier；概率校准（Platt/保序）；PSI；年龄分组公平性
5 # =====
6
7 # ----- 依赖 -----
8 pkgs <- c("data.table", "dplyr", "tidyverse", "caret", "pROC", "PRROC",
9 "ranger", "nnet", "ggplot2")
10 for (p in pkgs) if (!requireNamespace(p, quietly = TRUE)) install.packages(p)
11 invisible(lapply(pkgs, library, character.only = TRUE))
12 set.seed(2025)
13
14 # ----- 1) 下载/读取数据 -----
15 # 优先 UCI: https://archive.ics.uci.edu/ml/machine-learning-databases/
16 # statlog/german/german.data
17 uci_url <- "https://archive.ics.uci.edu/ml/machine-learning-databases/
18 # statlog/german/german.data"
19 tmp <- tempfile(fileext = ".data")
20 uci_ok <- try(utils::download.file(uci_url, tmp, quiet = TRUE), silent =
21 TRUE)
22
23 if (!inherits(uci_ok, "try-error")) {
24 message("使用 UCI German Credit (german.data) ")
25 # 按文档列名
26 cols <- c("chk_acc", "duration", "cred_hist", "purpose", "credit_amt", "savings",
27 "",
28 "emp_since", "install_rate", "pers_status_sex", "debtors", "
29 "
30)
31
32 # 读取文件
33 german <- read.table(tmp, header = TRUE, sep = " ", na.strings = "?",
34 col.names = cols)
35
36 # 去除空行
37 german <- german[!is.na(german$duration),]
38
39 # 将类别变量转换为哑变量
40 german <- as.data.table(german)
41 dummies <- lapply(german[, -c("duration", "purpose")], function(x) dummy(x))
42 dummies <- rbindlist(dummies)
43 dummies <- dummies[, -c("duration", "purpose")]
44 german <- cbind(german, dummies)
45
46 # 标准化数值变量
47 german$duration <- scale(german$duration)
48 german$credit_amt <- scale(german$credit_amt)
49 german$savings <- scale(german$savings)
50 german$emp_since <- scale(german$emp_since)
51 german$install_rate <- scale(german$install_rate)
52
53 # 重新设置列顺序
54 german <- german[, c("duration", "purpose", "credit_amt", "savings",
55 "emp_since", "install_rate", "pers_status_sex", "debtors",
56 "chk_acc", "cred_hist", "debtors", "over18", "self-employed",
57 "unemployed", "male", "female", "other", "yes", "no", "no", "no",
58 "no", "no",
59 "no", "no",
60 "no", "no",
61 "no", "no",
62 "no", "no",
63 "no", "no",
64 "no", "no",
65 "no", "no",
66 "no", "no",
67 "no", "no",
68 "no", "no",
69 "no", "no",
70 "no", "no",
71 "no", "no",
72 "no", "no",
73 "no", "no",
74 "no", "no",
75 "no", "no",
76 "no", "no",
77 "no", "no",
78 "no", "no",
79 "no", "no",
80 "no", "no",
81 "no", "no",
82 "no", "no",
83 "no", "no",
84 "no", "no",
85 "no", "no",
86 "no", "no",
87 "no", "no",
88 "no", "no",
89 "no", "no",
90 "no", "no",
91 "no", "no",
92 "no", "no",
93 "no", "no",
94 "no", "no",
95 "no", "no",
96 "no", "no",
97 "no", "no",
98 "no", "no",
99 "no", "no",
100 "no", "no",
101 "no", "no",
102 "no", "no",
103 "no", "no",
104 "no", "no",
105 "no", "no",
106 "no", "no",
107 "no", "no",
108 "no", "no",
109 "no", "no",
110 "no", "no",
111 "no", "no",
112 "no", "no",
113 "no", "no",
114 "no", "no",
115 "no", "no",
116 "no", "no",
117 "no", "no",
118 "no", "no",
119 "no", "no",
120 "no", "no",
121 "no", "no",
122 "no", "no",
123 "no", "no",
124 "no", "no",
125 "no", "no",
126 "no", "no",
127 "no", "no",
128 "no", "no",
129 "no", "no",
130 "no", "no",
131 "no", "no",
132 "no", "no",
133 "no", "no",
134 "no", "no",
135 "no", "no",
136 "no", "no",
137 "no", "no",
138 "no", "no",
139 "no", "no",
140 "no", "no",
141 "no", "no",
142 "no", "no",
143 "no", "no",
144 "no", "no",
145 "no", "no",
146 "no", "no",
147 "no", "no",
148 "no", "no",
149 "no", "no",
150 "no", "no",
151 "no", "no",
152 "no", "no",
153 "no", "no",
154 "no", "no",
155 "no", "no",
156 "no", "no",
157 "no", "no",
158 "no", "no",
159 "no", "no",
160 "no", "no",
161 "no", "no",
162 "no", "no",
163 "no", "no",
164 "no", "no",
165 "no", "no",
166 "no", "no",
167 "no", "no",
168 "no", "no",
169 "no", "no",
170 "no", "no",
171 "no", "no",
172 "no", "no",
173 "no", "no",
174 "no", "no",
175 "no", "no",
176 "no", "no",
177 "no", "no",
178 "no", "no",
179 "no", "no",
180 "no", "no",
181 "no", "no",
182 "no", "no",
183 "no", "no",
184 "no", "no",
185 "no", "no",
186 "no", "no",
187 "no", "no",
188 "no", "no",
189 "no", "no",
190 "no", "no",
191 "no", "no",
192 "no", "no",
193 "no", "no",
194 "no", "no",
195 "no", "no",
196 "no", "no",
197 "no", "no",
198 "no", "no",
199 "no", "no",
200 "no", "no",
201 "no", "no",
202 "no", "no",
203 "no", "no",
204 "no", "no",
205 "no", "no",
206 "no", "no",
207 "no", "no",
208 "no", "no",
209 "no", "no",
210 "no", "no",
211 "no", "no",
212 "no", "no",
213 "no", "no",
214 "no", "no",
215 "no", "no",
216 "no", "no",
217 "no", "no",
218 "no", "no",
219 "no", "no",
220 "no", "no",
221 "no", "no",
222 "no", "no",
223 "no", "no",
224 "no", "no",
225 "no", "no",
226 "no", "no",
227 "no", "no",
228 "no", "no",
229 "no", "no",
230 "no", "no",
231 "no", "no",
232 "no", "no",
233 "no", "no",
234 "no", "no",
235 "no", "no",
236 "no", "no",
237 "no", "no",
238 "no", "no",
239 "no", "no",
240 "no", "no",
241 "no", "no",
242 "no", "no",
243 "no", "no",
244 "no", "no",
245 "no", "no",
246 "no", "no",
247 "no", "no",
248 "no", "no",
249 "no", "no",
250 "no", "no",
251 "no", "no",
252 "no", "no",
253 "no", "no",
254 "no", "no",
255 "no", "no",
256 "no", "no",
257 "no", "no",
258 "no", "no",
259 "no", "no",
260 "no", "no",
261 "no", "no",
262 "no", "no",
263 "no", "no",
264 "no", "no",
265 "no", "no",
266 "no", "no",
267 "no", "no",
268 "no", "no",
269 "no", "no",
270 "no", "no",
271 "no", "no",
272 "no", "no",
273 "no", "no",
274 "no", "no",
275 "no", "no",
276 "no", "no",
277 "no", "no",
278 "no", "no",
279 "no", "no",
280 "no", "no",
281 "no", "no",
282 "no", "no",
283 "no", "no",
284 "no", "no",
285 "no", "no",
286 "no", "no",
287 "no", "no",
288 "no", "no",
289 "no", "no",
290 "no", "no",
291 "no", "no",
292 "no", "no",
293 "no", "no",
294 "no", "no",
295 "no", "no",
296 "no", "no",
297 "no", "no",
298 "no", "no",
299 "no", "no",
300 "no", "no",
301 "no", "no",
302 "no", "no",
303 "no", "no",
304 "no", "no",
305 "no", "no",
306 "no", "no",
307 "no", "no",
308 "no", "no",
309 "no", "no",
310 "no", "no",
311 "no", "no",
312 "no", "no",
313 "no", "no",
314 "no", "no",
315 "no", "no",
316 "no", "no",
317 "no", "no",
318
```

```

 residence_since",
25 "property", "age", "inst_plans", "housing", "num_credits", "job",
26 "num_people_maint", "telephone", "foreign_worker", "class")
27 df <- data.table::fread(tmp, header = FALSE, col.names = cols, data.table
28 = FALSE)
29 # 目标: class = 1(好) / 2(坏) -> bad = 1 表示违约
30 df$bad <- ifelse(df$class == 2, 1L, 0L)
31 df$class <- NULL
32 # 类型: 数值列
33 num_cols <- c("duration", "credit_amt", "install_rate", "residence_since",
34 "age", "num_credits", "num_people_maint")
35 for (cn in num_cols) df[[cn]] <- as.numeric(df[[cn]])
36 # 类别列转因子
37 cat_cols <- setdiff(names(df), c("bad", num_cols))
38 for (cn in cat_cols) df[[cn]] <- as.factor(df[[cn]])
39 } else {
40 message("UCI 下载失败, 回退到 caret::GermanCredit")
41 if (!requireNamespace("caret", quietly = TRUE)) install.packages("caret")
42 data("GermanCredit", package = "caret")
43 df <- as.data.frame(caret::GermanCredit)
44 # caret 版本用 "Class" (Bad/Good) ; 统一为 bad=1
45 df$bad <- ifelse(df$Class == "Bad", 1L, 0L)
46 df$Class <- NULL
47 # 简化列名
48 names(df) <- make.names(names(df))
49 }
50 # ----- 2) 分层切分: 60/20/20 -----
51 idx_tr <- caret::createDataPartition(df$bad, p = 0.6, list = FALSE)
52 train <- df[idx_tr,]
53 rest <- df[-idx_tr,]
54 idx_va <- caret::createDataPartition(rest$bad, p = 0.5, list = FALSE) # 0.2
55 /0.2
56 valid <- rest[idx_va,]
57 test <- rest[-idx_va,]
58 # ----- 3) 预处理 (哑变量 + 数值标准化, 为 nnet/Logit 一致口径)
59 -----
60 dummy <- caret::dummyVars(~ ., data = train[, setdiff(names(train), "bad")]
61], fullRank = TRUE)
62 X_tr <- predict(dummy, newdata = train)
63 X_va <- predict(dummy, newdata = valid)
64 X_te <- predict(dummy, newdata = test)
65 # 数值标准化 (基于训练集)
66 center <- colMeans(X_tr); scale <- apply(X_tr, 2, sd)
67 scale[scale == 0 | is.na(scale)] <- 1
68 X_tr <- sweep(sweep(X_tr, 2, center, "-"), 2, scale, "/")
69 X_va <- sweep(sweep(X_va, 2, center, "-"), 2, scale, "/")
70 X_te <- sweep(sweep(X_te, 2, center, "-"), 2, scale, "/")

```

```

71 y_tr <- train$bad; y_va <- valid$bad; y_te <- test$bad
72
73 # ----- 4) 训练三类模型 -----
74 # 4.1 Logit (基线, 可扩展为 WOE/评分卡)
75 fit_logit <- glm(y_tr ~ ., data = data.frame(y_tr = y_tr, X_tr), family =
 binomial())
76
77 # 4.2 随机森林 (ranger, 输出概率; 可加 class.weights 处理不平衡)
78 fit_rf <- ranger::ranger(
 dependent.variable.name = "bad",
 data = dplyr::bind_cols(bad = factor(y_tr), as.data.frame(X_tr)),
 probability = TRUE, num.trees = 500, mtry = max(1, floor(sqrt(ncol(X_tr))))
),
 min.node.size = 10, seed = 2025
83)
84
85 # 4.3 浅层神经网络 (nnet), 隐层10、L2衰减
86 set.seed(2025)
87 fit_nn <- nnet::nnet(x = X_tr, y = y_tr, size = 10, decay = 1e-4,
88 maxit = 200, linout = FALSE, trace = FALSE)
89
90 # ----- 5) 验证集概率 + 校准 (Platt 默认; 可选保序) -----
91 clip01 <- function(p) pmin(pmax(p, 1e-6), 1 - 1e-6)
92
93 p_logit_va <- clip01(predict(fit_logit, newdata = as.data.frame(X_va), type
 = "response"))
94 p_rf_va <- clip01(predict(fit_rf, data = as.data.frame(X_va))$predictions[, "1"])
95 p_nn_va <- clip01(predict(fit_nn, newdata = X_va, type = "raw"))
96
97 # Platt 校准函数 (对 logit(score) 回归)
98 platt_fit <- function(p, y) {
99 z <- qlogis(clip01(p))
100 glm(y ~ z, family = binomial())
101 }
102 platt_pred <- function(p, mod) {
103 z <- qlogis(clip01(p))
104 as.numeric(plogis(cbind(1, z) %*% coef(mod)))
105 }
106 cal_logit <- platt_fit(p_logit_va, y_va)
107 cal_rf <- platt_fit(p_rf_va, y_va)
108 cal_nn <- platt_fit(p_nn_va, y_va)
109
110 # (可选) 保序校准: 把等分箱均值作为单调映射 (简单稳定)
111 iso_cal <- function(p, y, bins = 20) {
112 dt <- data.frame(p = p, y = y) |>
113 dplyr::arrange(p) |>
114 dplyr::mutate(bin = cut_number(p, bins))
115 map <- dt |>
116 dplyr::group_by(bin) |>
117 dplyr::summarise(px = mean(p), py = mean(y), .groups = "drop") |>

```

```

118 dplyr::arrange(px)
119 function(pnew) {
120 approx(x = map$px, y = map$py, xout = pnew, rule = 2)$y
121 }
122 }
123 # 如需改用保序：把 platt_pred(...) 换成 iso_fun...
124 # iso_logit <- iso_cal(p_logit_va, y_va); iso_rf <- iso_cal(p_rf_va, y_va);
125 iso_nn <- iso_cal(p_nn_va, y_va)
126
127 # ----- 6) 测试集预测（含校准） -----
128 p_logit_te_raw <- clip01(predict(fit_logit, newdata = as.data.frame(X_te),
129 type = "response"))
130 p_rf_te_raw <- clip01(predict(fit_rf, data = as.data.frame(X_te))$

131 predictions[, "1"])
132 p_nn_te_raw <- clip01(predict(fit_nn, newdata = X_te, type = "raw"))
133
134 p_logit_te <- platt_pred(p_logit_te_raw, cal_logit)
135 p_rf_te <- platt_pred(p_rf_te_raw, cal_rf)
136 p_nn_te <- platt_pred(p_nn_te_raw, cal_nn)
137
138 # ----- 7) 评估工具 -----
139 ks_from_roc <- function(y, p){
140 roc <- pROC::roc(y, p, quiet = TRUE, direction = "<")
141 coords <- pROC::coords(roc, x = "all", ret = c("sensitivity", "specificity"

142))
143 max(coords["sensitivity",] - (1 - coords["specificity",]))
144 }
145 brier <- function(y, p) mean((y - p)^2)
146
147 report_model <- function(name, y, p){
148 roc <- pROC::roc(y, p, quiet = TRUE, direction = "<")
149 auc <- as.numeric(pROC::auc(roc))
150 ks <- ks_from_roc(y, p)
151 pr <- try(PRROC::pr.curve(scores.class0 = p[y==1], scores.class1 = p[y

152 ==0])$auc.integral,

153 silent = TRUE)
154 if (inherits(pr, "try-error")) pr <- NA_real_
155 cat(sprintf("[%s] AUC=% .4f KS=% .4f PR-AUC=% .4f Brier=% .4f\n",
156 name, auc, ks, pr, brier(y, p)))
157 invisible(list(auc=auc, ks=ks, pr_auc=pr, brier=brier(y,p)))
158 }
159
160 cat("== 测试集（校准后） ==\n")
161 m1 <- report_model("Logit", y_te, p_logit_te)
162 m2 <- report_model("RF", y_te, p_rf_te)
163 m3 <- report_model("NN", y_te, p_nn_te)
164
165 # ----- 8) 成本敏感阈值 + 混淆矩阵 -----
166 # 例：误报成本=1, 漏报成本=5 → tau* = c_fp/(c_fp+c_fn)
167 c_fp <- 1; c_fn <- 5; tau_star <- c_fp/(c_fp + c_fn)
168 cmat <- function(y, p, tau){

```

```

164 pd <- ifelse(p >= tau, 1L, 0L)
165 table(Pred = pd, True = y)
166 }
167 cat(sprintf("\n成本敏感阈值 tau*=% .3f 下的混淆矩阵 (Logit) \n", tau_star))
168 print(cmat(y_te, p_logit_te, tau_star))
169
170 # ----- 9) 校准曲线 (十分位) -----
171 cal_curve <- function(y, p, bins=10){
172 d <- data.frame(y=y, p=p) |>
173 dplyr::mutate(bin = cut(p, breaks = quantile(p, probs = seq(0,1,length.out=bins+1)),
174 include.lowest = TRUE)) |>
175 dplyr::group_by(bin) |>
176 dplyr::summarise(pred = mean(p), obs = mean(y), n = dplyr::n(), .groups=
177 "drop")
178 d
179 }
180 cal_logit_df <- cal_curve(y_te, p_logit_te, bins = 10)
181 ggplot(cal_logit_df, aes(pred, obs)) +
182 geom_point() + geom_line() +
183 geom_abline(slope=1, intercept=0, linetype=2) +
184 labs(title="校准曲线 (Logit, 测试集)", x="平均预测概率", y="实际违约率") +
185 theme_minimal()
186
187 # ----- 10) PSI (稳定性, 训练 vs 测试的分布漂移) -----
188 psi <- function(x_train, x_test, bins=10){
189 cuts <- quantile(x_train, probs = seq(0,1,length.out=bins+1), na.rm=TRUE)
190 cuts[1] <- -Inf; cuts[length(cuts)] <- Inf
191 bt <- table(cut(x_train, cuts)); be <- table(cut(x_test, cuts))
192 pt <- as.numeric(bt)/sum(bt); pe <- as.numeric(be)/sum(be)
193 sum((pt - pe) * log((pt + 1e-8) / (pe + 1e-8)))
194 }
195 psi_score <- psi(clip01(p_logit_te_raw), clip01(p_logit_te)) # 例：未校准
196 vs 校准
197 cat(sprintf("\nPSI (原始Logit概率 vs 校准后概率) : %.4f\n", psi_score))
198
199 # ----- 11) 公平性 (例：按年龄分组的 AUC/TPR 差异) -----
200 # 若数据集中有 age，则做一个简单分组 (年轻<35 vs 其余)
201 fair_summary <- function(y, p, age, tau){
202 grp <- ifelse(age < 35, "young", "old")
203 df <- data.frame(y=y, p=p, grp=grp)
204 auc_y <- as.numeric(pROC::auc(pROC::roc(df$y[df$grp=="young"], df$p[df$grp
205 == "young"], quiet=TRUE)))
206 auc_o <- as.numeric(pROC::auc(pROC::roc(df$y[df$grp=="old"], df$p[df$grp
207 == "old"], quiet=TRUE)))
208 tpr_y <- with(df[df$grp=="young",], mean(p>=tau & y==1)/max(mean(y==1), 1
209 e-8))
210 tpr_o <- with(df[df$grp=="old",], mean(p>=tau & y==1)/max(mean(y==1), 1
211 e-8))
212 data.frame(AUC_young=auc_y, AUC_old=auc_o, AUC_gap=auc_o-auc_y,

```

```

207 TPR_young=tpr_y, TPR_old=tpr_o, TPR_gap=tpr_o-tpr_y)
208 }
209 if ("age" %in% tolower(names(df))) {
210 # 尝试自动识别 age 列名
211 age_col <- names(df)[tolower(names(df))== "age"] [1]
212 age_te <- test[[age_col]]
213 fs <- fair_summary(y_te, p_logit_te, age_te, tau_star)
214 cat("\n按年龄分组的公平性 (Logit) : \n"); print(fs)
215 } else {
216 cat("\n未找到 age 列，跳过年龄分组的公平性示例。 \n")
217 }
218
219 # ----- 12) 小结打印 -----
220 cat("\n== 指标小结 (测试集, 校准后) ==\n")
221 print(rbind(
222 Logit = unlist(m1),
223 RF = unlist(m2),
224 NN = unlist(m3)
225))

```

#### 12.4.4 因子提取与风险度量：自编码器与深度贝叶斯

在高维资产空间中，把众多标的的共同波动“压缩”为少数可解释的因子，是资产定价与风险管理的核心步骤。线性主成分（PCA）给出了一条稳健的起点，但当共动结构存在非线性或异质性时，自编码器（AE / 变分 AE）能以端到端的方式学习到更灵活的“非线性因子”，并与后续的线性风险模型自然衔接。因子得到后，可用横截面回归估计各资产对因子的暴露（ $\beta$  与截距  $\alpha$ ），据此构造因子协方差与特质风险，进而进行 VaR / ES 等组合层面的风险聚合与情景评估。

为将上述流程落地，本节提供一套“下载—提取—暴露—聚合—不确定性”的可复现 R 脚本。脚本从 Yahoo Finance 抓取多只 ETF 的日度价格，构造对数收益率矩阵；默认以 PCA 提取  $k$  个因子（可选将 `run_ae` 设为 TRUE 使用 Keras-AE 提取“非线性因子”），随后在训练窗口对每只资产做 OLS 回归得到  $\beta/\alpha$  与特质波动率，再以训练窗口的因子协方差聚合组合风险。考虑到不同阶段的波动水平可能差异显著，脚本在因子 Monte-Carlo 环节默认采用 EWMA 协方差 (`use_ewma, λ = 0.97`)，并提供“波动率目标化” (`vol_target`) 选项，将模型无条件波动率缩放到测试期真实水平，避免 MC VaR 因尺度错配而偏大。除历史法 VaR/ES 与因子 MC VaR/ES 外，脚本还通过对训练窗口“重采样—重估暴露”的自助法构造 VaR/ES 的分位数区间，用以刻画估计不确定性并做情景沟通——这是一种“贝叶斯式不确定性”的轻量近似，便于工程化集成（本例未做真正的深度贝叶斯后验推断，如 VI/MCMC 对潜变量和网络权重）。输出包括测试期因子时间序列、资产暴露热力图，以及“历史法/因子 MC/自助法中位数”的 VaR 对比图；关键参数 (`k_factors`、`use_ewma/lambda_ewma`、`vol_target`、`B_boot` 等) 均显式给出，读者可据此快速复现实验、切换口径并扩展到自有资产池或更高频的实现波动率特征。

```

1 # -----
2 # 因子提取与风险度量：PCA/自编码器因子 + OLS 暴露 + VaR/ES
3 # 数据：Yahoo Finance (多资产ETF)，频率：日度
4 # 亮点：
5 # 1) PCA 为基线 (run_ae=TRUE 时用 Keras-AE 提取“非线性因子”)
6 # 2) 稳健“暴露估计”与“自助法”实现 (严格按因子名对齐，避免维度错配)

```

```

7 # 3) 因子 MC 使用 EWMA 协方差（可选波动目标化），修正“MC VaR 偏大”问题
8 # 4) 输出：历史法 VaR/ES、因子MC VaR/ES、自助法 VaR/ES 分位区间 + 可视化
9 # =====
10
11 ## ----- 参数 -----
12 start_date <- as.Date("2015-01-01")
13 end_date <- as.Date("2024-12-31")
14 tickers <- c(# 风格/行业/大类资产（可按需增删）
15 "SPY", "QQQ", "IWM", "EFA", "EEM",
16 "TLT", "IEF", "LQD", "HYG",
17 "GLD", "SLV", "USO",
18 "XLK", "XLF", "XLY", "XLP", "XLV", "XLI", "XLE", "XLU", "XLB"
19)
20 k_factors <- 3 # 因子数
21 run_ae <- FALSE # TRUE 启用自编码器（需 keras / tensorflow）
22 alpha <- 0.95 # VaR/ES 置信水平
23 B_boot <- 200 # 自助法次数
24 # 因子 MC 的“状态对齐”设置
25 use_ewma <- TRUE # TRUE: 用 EWMA 协方差；FALSE: 用样本协方差
26 lambda_ewma <- 0.97 # EWMA 衰减
27 vol_target <- TRUE # TRUE: 将 MC 无条件波动缩放到测试期真实波动
28 set.seed(2025)
29
30 ## ----- 依赖 -----
31 pkgs <- c("quantmod", "dplyr", "tidyverse", "zoo", "ggplot2", "scales", "MASS", "
32 matrixStats")
33 for (p in pkgs) if (!requireNamespace(p, quietly = TRUE)) install.packages(p
34)
35 invisible(lapply(pkgs, library, character.only = TRUE))
36
37 if (run_ae) {
38 if (!requireNamespace("keras", quietly = TRUE)) install.packages("keras")
39 library(keras) # 首次需: keras::install_keras()
40 }
41
42 ## ----- 1) 下载 & 构造收益矩阵 -----
43 get_ok <- function(sym){
44 tryCatch({
45 suppressWarnings(getSymbols(sym, src = "yahoo",
46 from = start_date - 30, to = end_date,
47 auto.assign = FALSE))
48 }, error = function(e) NULL)
49 }
50
51 raw <- lapply(tickers, get_ok); names(raw) <- tickers
52 ok <- vapply(raw, Negate(is.null), logical(1))
53 if (!all(ok)) message("以下代码下载失败并被剔除：", paste(tickers[!ok],
54 collapse = ", "))

```

```

54 tickers <- tickers[ok]; raw <- raw[ok]
55
56 Adj <- lapply(raw, Ad)
57 idx_all <- Reduce(intersect, lapply(Adj, index))
58 Adj <- lapply(Adj, function(x) x[idx_all])
59 R <- do.call(cbind, lapply(Adj, function(p) diff(log(p))))
60 R <- R[complete.cases(R),]
61 colnames(R) <- tickers
62
63 # 可选：剔除缺失/极端资产
64 na_rate <- colMeans(is.na(R))
65 keep <- na_rate < 0.01
66 R <- R[, keep, drop = FALSE]; tickers <- colnames(R)
67
68 ## ----- 2) 训练/测试切分 & 标准化 -----
69 n <- nrow(R)
70 split_tr <- floor(0.7 * n) # 70% 训练，其余测试
71 R_tr <- R[1:split_tr, , drop = FALSE]
72 R_te <- R[(split_tr+1):n, , drop = FALSE]
73
74 mu_tr <- colMeans(R_tr); sd_tr <- apply(R_tr, 2, sd)
75 sd_tr[sd_tr == 0 | is.na(sd_tr)] <- 1
76
77 scale_apply <- function(X, mu, sd) sweep(sweep(X, 2, mu, "-"), 2, sd, "/")
78 Xs_tr <- scale_apply(R_tr, mu_tr, sd_tr)
79 Xs_te <- scale_apply(R_te, mu_tr, sd_tr)
80
81 ## ----- 3) 因子提取：PCA / Keras-AE -----
82 # 3.1 PCA
83 pca <- prcomp(Xs_tr, center = FALSE, scale. = FALSE)
84 Z_tr_pca <- pca$x[, 1:k_factors, drop = FALSE]
85 Z_te_pca <- scale(Xs_te, center = FALSE, scale = FALSE) %*% pca$rotation[, 1:k_factors, drop = FALSE]
86
87 # 3.2 AE (可选)
88 if (run_ae) {
89 p <- ncol(Xs_tr)
90 inputs <- layer_input(shape = p)
91 x <- inputs |>
92 layer_dense(units = 64, activation = "relu") |>
93 layer_dropout(rate = 0.1) |>
94 layer_dense(units = k_factors, activation = "linear", name = "latent")
95 y <- x |>
96 layer_dense(units = 64, activation = "relu") |>
97 layer_dropout(rate = 0.1) |>
98 layer_dense(units = p, activation = "linear")
99 ae <- keras_model(inputs = inputs, outputs = y) |>
100 compile(optimizer = optimizer_adam(learning_rate = 0.01), loss = "mse")
101 ae |>
102 fit(x = as.matrix(Xs_tr), y = as.matrix(Xs_tr),
103 validation_split = 0.1, epochs = 80, batch_size = 64,

```

```

104 callbacks = list(callback_early_stopping(monitor="val_loss",
105 patience=8,
106 restore_best_weights=TRUE))
107 , verbose = 0)
108 encoder <- keras_model(inputs = inputs, outputs = get_layer(ae, "latent")$output)
109 } else {
110 Z_tr <- Z_tr_pca; Z_te <- Z_te_pca
111 }
112
113 colnames(Z_tr) <- paste0("F", 1:k_factors)
114 colnames(Z_te) <- paste0("F", 1:k_factors)
115
116 ## ----- 4) 暴露 (beta/alpha/sigma) 估计: 稳健对齐版
117 -----
118 if (is.null(colnames(Z_tr)) || anyDuplicated(colnames(Z_tr)) > 0) {
119 colnames(Z_tr) <- paste0("F", seq_len(ncol(Z_tr)))
120 }
121 fac_names <- colnames(Z_tr) # k
122 assets <- colnames(R_tr) # N
123 k <- length(fac_names); N <- length(assets)
124
125 B_mat <- matrix(NA_real_, nrow = k, ncol = N,
126 dimnames = list(fac_names, assets))
127 A_vec <- setNames(numeric(N), assets)
128 S_eps <- setNames(numeric(N), assets)
129
130 ols_fit <- function(y, Zdf) {
131 fit <- lm(y ~ ., data = cbind(y = y, Zdf))
132 cf <- coef(fit)
133 beta <- cf[-1]; names(beta) <- gsub(" ", "", names(beta))
134 alpha<- unname(cf[1])
135 yhat <- as.numeric(alpha + as.matrix(Zdf) %*% beta)
136 list(alpha=alpha, beta=beta, yhat=yhat)
137 }
138 ridge_fit <- function(y, Zdf, lambda = 1e-4) {
139 if (!requireNamespace("MASS", quietly = TRUE)) install.packages("MASS")
140 X <- as.matrix(Zdf)
141 rr <- MASS::lm.ridge(y ~ X, lambda = lambda)
142 beta <- setNames(as.numeric(rr$coef), colnames(X))
143 yhat <- as.numeric(X %*% beta)
144 alpha<- mean(y - yhat); yhat <- alpha + yhat
145 list(alpha=alpha, beta=beta, yhat=yhat)
146 }
147 for (j in seq_along(assets)) {
148 y <- R_tr[, j]
149 Zdf <- as.data.frame(Z_tr)
150 fit <- try(ols_fit(y, Zdf), silent = TRUE)

```

```

151 if (inherits(fit, "try-error")) fit <- ridge_fit(y, Zdf, lambda = 1e-4)
152
153 beta <- fit$beta; names(beta) <- gsub(" ", "", names(beta))
154 common <- intersect(fac_names, names(beta))
155 if (length(common) > 0) B_mat[common, j] <- as.numeric(beta[common])
156
157 A_vec[j] <- fit$alpha
158 S_eps[j] <- sd(y - fit$yhat)
159 }
160
161 # 可选：把缺失暴露当作 0
162 # B_mat[is.na(B_mat)] <- 0
163
164 cat("B_mat 维度: ", paste(dim(B_mat), collapse = " x "), "\n")
165 stopifnot(nrow(B_mat) == k, ncol(B_mat) == N,
166 all(rownames(B_mat) == fac_names),
167 all(colnames(B_mat) == assets))
168
169 ## ----- 5) 组合暴露与风险聚合 (VaR/ES) -----
170 w <- rep(1/length(assets), length(assets)); names(w) <- assets # 等权 (按实际资产集)
171
172 b_port <- as.numeric(B_mat %*% w) # k×1
173 # 因子协方差: EWMA or 样本
174 if (use_ewma) {
175 wts <- lambda_ewma^(rev(seq_len(nrow(Z_tr))) - 1); wts <- wts / sum(wts)
176 Sigma_F <- cov.wt(Z_tr, wt = wts, center = rep(0, ncol(Z_tr)))$cov
177 } else {
178 Sigma_F <- cov(Z_tr)
179 }
180 Sigma_eps <- diag(S_eps^2, nrow = N)
181 sigma_idio2 <- as.numeric(t(w) %*% Sigma_eps %*% w)
182
183 Rte_mat <- as.matrix(R_te); Zte_mat <- as.matrix(Z_te)
184 pnl_true_te <- as.numeric(Rte_mat %*% w) # 真实组合 (测试期)
185 # 诊断: _factor / _idio / _model / _true
186 sigma_factor <- sqrt(t(b_port) %*% Sigma_F %*% b_port)
187 sigma_model <- sqrt(sigma_factor^2 + sigma_idio2)
188 sigma_true <- sd(pnl_true_te)
189 print(c(sigma_factor = sigma_factor, sigma_idio = sqrt(sigma_idio2),
190 sigma_model = sigma_model, sigma_true = sigma_true))
191
192 # 历史法 VaR/ES (测试期真实收益)
193 VaR_hist <- -quantile(pnl_true_te, probs = 1 - alpha, na.rm = TRUE)
194 ES_hist <- -mean(pnl_true_te[pnl_true_te <= quantile(pnl_true_te, probs = 1
195 - alpha, na.rm = TRUE)])
196 cat(sprintf("历史法 (真实组合): VaR@%.0f%%=% .4f, ES@%.0f%%=% .4f\n", alpha*
197 100, VaR_hist, alpha*100, ES_hist))
198
199 # 因子 MC: 高斯抽样 (可选 vol_target 把无条件波动对齐到测试期)

```

```

198 mc_sims <- 10000
199 F_draw <- MASS::mvrnorm(n = mc_sims, mu = rep(0, k), Sigma = Sigma_F)
200 raw_mc <- as.numeric(F_draw %*% b_port) + rnorm(mc_sims, 0, sqrt(sigma_
 idio2))
201 pnl_mc <- if (vol_target) as.numeric(sigma_true / sigma_model) * raw_mc
 else raw_mc
202 VaR_mc <- -quantile(pnl_mc, probs = 1 - alpha, na.rm = TRUE)
203 ES_mc <- -mean(pnl_mc[pnl_mc <= quantile(pnl_mc, probs = 1 - alpha, na.rm
 = TRUE)])
204 cat(sprintf("因子MC（模型）: VaR@%.0f%%=% .4f, ES@%.0f%%=% .4f\n", alpha*
 100, VaR_mc, alpha*100, ES_mc))
205
206 ## ----- 6) 自助法：暴露不确定性 → VaR/ES 区间 -----
207 boot_expo_safe <- function(R_tr, Z_tr, w, B = 200, alpha = 0.95, lambda_
 ridge = 1e-4) {
208 stopifnot(nrow(R_tr) == nrow(Z_tr))
209 if (is.null(colnames(Z_tr)) || anyDuplicated(colnames(Z_tr)) > 0)
210 colnames(Z_tr) <- paste0("F", seq_len(ncol(Z_tr)))
211 fac_names <- colnames(Z_tr); assets <- colnames(R_tr)
212 k <- length(fac_names); N <- length(assets)
213 VaR_b <- numeric(B); ES_b <- numeric(B)
214
215 ols_fit <- function(y, Zdf) {
216 fit <- lm(y ~ ., data = cbind(y = y, Zdf))
217 cf <- coef(fit); alpha <- unname(cf[1]); beta <- cf[-1]
218 names(beta) <- gsub("`", "", names(beta))
219 yhat <- as.numeric(alpha + as.matrix(Zdf) %*% beta)
220 list(alpha = alpha, beta = beta, yhat = yhat)
221 }
222 ridge_fit <- function(y, Zdf, lambda = 1e-4) {
223 if (!requireNamespace("MASS", quietly = TRUE)) install.packages("MASS")
224 X <- as.matrix(Zdf); rr <- MASS::lm.ridge(y ~ X, lambda = lambda)
225 beta <- setNames(as.numeric(rr$coef), colnames(X))
226 yhat <- as.numeric(X %*% beta); alpha <- mean(y - yhat); yhat <- alpha +
227 yhat
228 list(alpha = alpha, beta = beta, yhat = yhat)
229 }
230
231 for (b in seq_len(B)) {
232 idx <- sample(nrow(R_tr), nrow(R_tr), replace = TRUE)
233 Rb <- R_tr[idx, , drop = FALSE]
234 Zb <- Z_tr[idx, , drop = FALSE]
235 colnames(Zb) <- fac_names
236
237 Bmat_b <- matrix(0, nrow = k, ncol = N, dimnames = list(fac_names,
238 assets))
239 Sb2 <- numeric(N)
240
241 for (j in seq_len(N)) {
242 y <- Rb[, j]; Zdf <- as.data.frame(Zb)
243 fit <- try(ols_fit(y, Zdf), silent = TRUE)

```

```

242 if (inherits(fit, "try-error")) fit <- ridge_fit(y, Zdf, lambda =
243 lambda_ridge)
244 beta <- fit$beta; names(beta) <- gsub(" ", "", names(beta))
245 common <- intersect(fac_names, names(beta))
246 if (length(common) > 0) Bmat_b[common, j] <- as.numeric(beta[common])
247 Sb2[j] <- stats::sd(y - fit$yhat)^2
248 }
249
250 b_port_b <- as.numeric(Bmat_b %*% w)
251 mcs <- 5000L
252 # 为了一致性，使用自举样本的因子协方差（也可用 EWMA）
253 SigmaF_b <- cov(Zb)
254 F_draw <- MASS::mvrnorm(mcs, mu = rep(0, k), Sigma = SigmaF_b)
255 eps_sd <- sqrt(as.numeric(t(w) %*% diag(Sb2, nrow = N) %*% w))
256 pnl_b <- as.numeric(F_draw %*% b_port_b) + stats::rnorm(mcs, 0, eps_sd)
257
258 VaR_b[b] <- -stats::quantile(pnl_b, probs = 1 - alpha, na.rm = TRUE)
259 ES_b[b] <- -mean(pnl_b[pnl_b <= stats::quantile(pnl_b, probs = 1 -
260 alpha, na.rm = TRUE)])
261 }
262
263 list(VaR = VaR_b, ES = ES_b)
264 }
265
266 boot_res <- boot_expo_safe(R_tr, Z_tr, w, B = B_boot, alpha = alpha)
267 VaR_ci <- quantile(boot_res$VaR, probs = c(0.05, 0.50, 0.95))
268 ES_ci <- quantile(boot_res$ES, probs = c(0.05, 0.50, 0.95))
269 cat(sprintf(" 自助 VaR 区间 [5%,50%,95%]: %s\n", paste(round(VaR_ci, 4),
270 collapse = " / ")))
271 cat(sprintf(" 自助 ES 区间 [5%,50%,95%]: %s\n", paste(round(ES_ci, 4),
272 collapse = " / ")))
273
274 ## ----- 7) 可视化：因子、暴露、VaR 对比 -----
275 df_fac <- data.frame(date = index(R_te), Z_te) |>
276 tidyr::pivot_longer(-date, names_to = "Factor", values_to = "Value")
277 g1 <- ggplot(df_fac, aes(date, Value, color = Factor)) +
278 geom_line() + theme_minimal() +
279 labs(title = " 测试期因子时间序列 ", x = NULL, y = "Factor")
280
281 df_exp <- as.data.frame(t(B_mat))
282 df_exp$Asset <- rownames(df_exp)
283 df_exp <- tidyr::pivot_longer(df_exp, -Asset, names_to = "Factor", values_to
284 = "Beta")
285 g2 <- ggplot(df_exp, aes(Factor, Asset, fill = Beta)) +
286 geom_tile() + scale_fill_gradient2(low = "steelblue", high = "firebrick",
287 mid = "white") +
288 theme_minimal() + labs(title = " 资产对因子暴露 (训练期 OLS) ", x = NULL, y
289 = NULL)
290
291 df_risk <- data.frame(
292 Method = c("Boot-VaR-50%", "Hist-VaR", "MC-VaR"),
293 Value = c(VaR_ci[2], VaR_hist, VaR_mc)

```

```

286)
287 g3 <- ggplot(df_risk, aes(Method, Value, fill = Method)) +
288 geom_col(width = 0.6) + theme_minimal() +
289 labs(title = sprintf("组合 VaR@%.0f%% 对比", alpha*100), y = "VaR")
290
291 print(g1); print(g2); print(g3)
292
293 ## ----- 8) 小结 -----
294 cat("\n==== 小结 ====\n")
295 cat("* PCA/AE 将高维收益压缩为少数因子; OLS 回归得到资产暴露, 因子协方差由训
 练期估计 (本例默认 EWMA)。\\n")
296 cat("* 历史法 VaR/ES 基于测试期真实收益; 因子 MC VaR/ES 使用 Σ_F 高斯抽样,
 并可做波动目标化以匹配当前状态。\\n")
297 cat("* 自助法给出 VaR/ES 的不确定性分位区间, 便于做“深度贝叶斯”风格的风险
 沟通与情景分析。\\n")

```

## 12.5 模型可解释性与金融稳定性

### 12.5.1 AI 模型的可解释性挑战

在金融计量的语境中, 模型不仅要“准”, 还要“可说清楚”和“可治理”。黑箱模型之所以难落地, 根源在于它们通常只能给出  $\hat{y} = f(x)$  的点预测, 却难以回答监管与业务最关心的四个问题: 第一, “为什么是这个结果”(重要变量、方向、相对贡献); 第二, “在什么区间最敏感”(局部斜率、形状与边界); 第三, “换个时点/子样本是否仍然成立”(稳定性与漂移); 第四, “是否存在系统性偏差”(数据泄漏、选择偏、群体公平)。

从数据与评估角度看, 数据泄漏会直接破坏样本外有效性: 若标签窗口为  $[t, t+h]$ , 则训练集中任何在  $[t, t+h]$  才能观测到的特征都不得出现(避免 look-ahead / 目标衍生); 横截面—时间面板的切分应采用净化的时序交叉验证第  $k$  折验证区间为  $[T_k^{\text{start}}, T_k^{\text{end}}]$ , 则训练集需剔除与其发生时间重叠的样本, 并在验证区间两端各空出  $\delta$  的禁运带; 这样可以把“隐性泄漏”转化为显式的样本外评估。另一方面, 漂移指数据分布随时间发生变化: 协变量漂移可记为  $p_t(x) \neq p_s(x)$ , 概念漂移可记为  $p_t(y | x) \neq p_s(y | x)$ 。常用的稳定性度量如人口稳定性指数 (PSI):

$$\text{PSI} = \sum_b (p_b - q_b) \log \frac{p_b}{q_b},$$

其中  $p_b, q_b$  分别为基准期与监测期在第  $b$  个分箱的占比; 当 PSI 超过阈值(如 0.1 / 0.25)时, 应触发再训练或阈值重校准。更精细的漂移检验还可采用 Jensen-Shannon 散度或基于分类器的两样本检验(把时期标签当作二分类目标, 检验可分性)。

### 12.5.2 可解释性工具: SHAP / LIME / PDP / ICE

可解释性工具的核心思想, 是用“可加性解释”或“局部线性近似”将黑箱输出分解到各特征上, 同时保持与原模型的一致性。

SHAP 基于 Shapley 值给出“局部—全局一致”的特征贡献。令特征集合为  $\mathcal{F}$  且  $M = |\mathcal{F}|$ 。对给定样本  $x$ , 特征  $j$  的 Shapley 归因为

$$\phi_j(f, x) = \sum_{S \subseteq \mathcal{F} \setminus \{j\}} \frac{|S|! (M - |S| - 1)!}{M!} \left( f_{S \cup \{j\}}(x) - f_S(x) \right),$$

其中  $f_S$  表示将非  $S$  特征“积分掉”的条件期望，实际实现中以背景分布近似。SHAP 满足局部准确性 ( $f(x) = \phi_0 + \sum_j \phi_j$ )、一致性与缺省性等公理；树模型可用 TreeSHAP 在多项式时间内精确计算。全局层面可聚合  $E|\phi_j|$  得到特征重要性排序，也可计算交互项  $\Phi_{ij}$  识别二阶交互作用。

LIME 则在  $x$  的邻域内用一个稀疏线性模型逼近黑箱模型：生成邻域样本  $\{z_i\}$ ，以  $\pi_x(z_i)$  衡量邻近度，求解

$$\hat{g} = \arg \min_{g \in \mathcal{G}} \sum_i \pi_x(z_i) (f(z_i) - g(z_i))^2 + \Omega(g),$$

其中  $\mathcal{G}$  为可解释模型族（如线性模型）， $\Omega$  为稀疏惩罚项。LIME 的优点是直观、模型无关；缺点是邻域与权重的选择会影响稳定性，适合与 SHAP 交叉验证使用。

部分依赖 (PDP) 与个体条件期望 (ICE) 用于刻画边际效应与个体异质性。对特征子集  $S$  与补集  $C$ ，经验 PDP 定义为

$$\hat{f}_S(x_S) = \frac{1}{n} \sum_{i=1}^n f(x_S, x_C^{(i)}).$$

ICE 则保留个体  $i$  的  $x_C^{(i)}$  不变，描绘  $x_S$  扫描时  $f$  的轨迹。若特征高度相关，PDP 可能落在数据支撑之外，建议改用累积局部效应 (ALE)，其一维形式为

$$ALE_j(x) = \int_{x_0}^x E \left[ \frac{\partial f(u, \mathbf{x}_{-j})}{\partial u} \mid x_j = u \right] du,$$

以条件导数的积分替代边际插值，显著降低外推偏差。全局重要性可用置换重要度：打乱特征  $j$  得到评分劣化  $\Delta_j = Score(D) - Score(D_{\pi(j)})$ ，并与 SHAP 排序互相验证。

### 12.5.3 稳定性与合规

在时变金融环境中，稳定性与合规性与可解释性同等重要。稳定性首先源于评估设计：采用滚动/扩展窗口与 purged K-fold（对每折验证区间  $[T_k^{\text{start}}, T_k^{\text{end}}]$  的左右两侧各设置禁运带  $\delta$ ），所有特征仅以“过去信息”构造，严禁跨窗口信息共享；其次源于形状与符号约束：对评分卡或树/提升模型，可在训练中加入单调性约束  $\partial f / \partial x_j \geq 0$ （或  $\leq 0$ ），以及对关键变量施加平滑/分段单调先验，必要时用保序回归对输出概率进行后校准，从而将经济学先验融入模型结构。针对参数或贡献的时变性，可定义“符号一致性/排名一致性”指标，例如对滚动窗口  $w = 1, \dots, W$  的 SHAP 排序  $r_j^{(w)}$ ，考察 Kendall  $\tau$  或 Spearman  $\rho$  的稳定性。

漂移监控除 PSI 外，还可区分协变量漂移与概念漂移：前者可通过重要性加权修正损失

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^n w(x_i) \ell(y_i, f_{\theta}(x_i)), \quad w(x) = \frac{p_t(x)}{p_s(x)},$$

权重可由核密度比或类判别模型估计；后者则需要再训练或门限重定标。概率输出的质量建议同时报告 Brier 分数

$$\text{Brier} = \frac{1}{n} \sum_{i=1}^n (\hat{p}_i - y_i)^2,$$

以及“校准斜率/截距”（在验证集上用  $\text{logit}(\hat{p})$  回归  $y$  的斜率与截距），并在必要时采用 Platt/保序回归做校准。对于区间预测，可采用分位数回归或共形预测构造  $1 - \alpha$  的预测区间，后者在独立同分布假设下满足有限样本覆盖率保证。

合规层面，模型风险管理要求“开发—独立验证—内审”三道防线清晰分工；文档化需覆盖数据来源与处理过程、特征字典、训练与评估数据集划分、超参数与随机种子、阈值与策略、漂移监测与再训练计划；可追溯性要求完整的版本与环境封装；公平性与隐私方面，应明确公平性定义（如机会均等  $TPR_{g_1} = TPR_{g_2}$ 、均衡误差 ( $TPR, FPR$ ) 同时匹配或人口比例  $\Pr(\hat{y} = 1 | A = g)$  不依赖分组  $g$ ），并在必要时通过阈值分组、约束学习或后处理调整实现；隐私保护可采用差分隐私  $((\epsilon, \delta)\text{-DP})$  或联邦学习在合规边界内训练。最后，关键业务流程需设置“停机开关”与人工兜底机制，并将极端场景的压力测试纳入上线前与周期性检验：例如对输入施加情境扰动  $\Delta x$  检查  $\Delta f$  的上界，或对收益/损失序列计算  $ES_\alpha = E[L | L \geq VaR_\alpha]$  的敏感性，以确保在尾部风险下模型仍然可控。

综上，面向金融计量的可解释性实践是一个“从设计到治理”的闭环：以严谨的评估与数据设计守住外推边界，以 SHAP/LIME/PDP/ICE 等工具回答“谁重要、怎么变”，以形状约束和单调约束注入经济学先验，以漂移监控与校准确保概率输出的可靠性，并在公平与隐私的约束下交付可追溯、可复现、可审计的模型证据。这些环节共同支撑“模型·稳定性·合规”的三重目标。

## 12.6 本章小结

本章围绕“AI 与金融计量”的结合展开，从动机—方法—评估—治理四个维度系统梳理了树模型与深度学习在金融中的可复现应用路径。首先，在方法层，给出了决策树的统一“不纯度下降”视角，推导了基尼/熵、信息增益/增益率与回归树的方差下降；进一步将单树扩展到随机森林（装袋降方差）与梯度提升树/XGBoost（二阶近似与结构惩罚），明确了关键超参数（学习率、树深、最小叶权重、行/列采样率等）与正则化作用。配套的 SPY 月频案例展示了“固定样本区间 + 滚动样本外 + 早停/校准”的完整流水线，并将概率信号转换为含交易成本的简易仓位以报告年化收益、波动率与 Sharpe（夏普比率）。

其次，在深度学习部分，从 MLP 的经验风险最小化出发，给出交叉熵/分位数损失、Adam/AdamW 与早停、权重衰减/Dropout/BatchNorm 等训练细节；在时序建模上，阐述了 LSTM 的门控结构与 Transformer 的自注意力机制，并强调全部数据预处理（标准化、PCA/AE 等）必须在训练窗内拟合、对验证/测试外推以杜绝信息泄漏。面向收益率预测与波动率建模，提供了“OLS/岭回归—XGBoost—LSTM（可选）”与“滚动 GARCH—HAR-RV（取对数）—注意力（可选）”的可复现实验，说明了在样本有限、弱信号与非平稳背景下，滚动/扩展窗口 + 早停 + 校准 + 成本约束是获得稳健样本外证据的必要前提。

再次，在调参与评估层面，本章给出了金融时序下的嵌套交叉验证与 Purged/Embargo K-fold 方案，讨论了网格/随机/贝叶斯优化与 Hyperband 等搜索策略，并用谱滤波视角解释了“平方损失 + 线性模型中，早停近似  $L_2$  正则”的结论。针对回测过拟合与多重比较风险，介绍了 CSCV/PBO、White 的现实检验与 SPA，以及 BH/BY 假发现率 (FDR) 控制，并建议以外层一次性报告固化口径，配合区块自举报告经济指标差异的显著性。

随后，本章从解释与治理角度，系统介绍了 SHAP (TreeSHAP)、LIME、PDP/ALE 与 ICE 的使用场景与优缺点，给出 SHAP + PDP/ALE 的组合实践以回答“谁重要、怎么变、在何区间敏感”。在稳定性与合规方面，提出 PSI/JS 散度做漂移监控、Platt/保序做概率校准，阈值以  $\tau^* = c_{FP}/(c_{FP} + c_{FN})$  对齐成本；并强调三道防线、文档化与可追溯（数据血缘、特征字典、切分与随机种子、版本与阈值），以及公平性（机会均等/均衡误差/人口比例）与隐私（差分隐私/联邦学习）的落地要求。

最后，在应用案例上，信用评分部分以 UCI German Credit 为例，构建“Logit（评分卡雏形）—随机森林—浅层神经网络”的对比，并在验证集完成概率校准后于测试集报

告 AUC/PR-AUC/Brier 与成本敏感阈值的混淆矩阵，同时给出 PSI 与分组 AUC/TPR 公平性诊断；因子与风险聚合部分则以“PCA/AE 因子—OLS 暴露—EWMA 协方差—因子 MC/历史法 VaR/ES”为主线，加入 vol-target 与自助法区间以修正状态尺度并量化估计不确定性。总体而言，本章把“强表达的学习器”嵌入到“因果正确的时序评估—合规完整的治理”闭环中，给出了在预测—解释—合规三目标下可复用的建模与评估范式。

## 12.7 习题

### 1. 树模型与不纯度下降

- (a) 证明分类树在基尼指数下的一次二分划分增益等价于“类间方差”提升：设二分类场景中正类比例为  $p$ ，分裂后左右子节点的正类比例分别为  $(p_L, p_R)$ ，证明

$$\Delta I_{\text{Gini}} = \text{Gini}(D) - \frac{|D_L|}{|D|} \text{Gini}(D_L) - \frac{|D_R|}{|D|} \text{Gini}(D_R) = 2 \frac{|D_L|}{|D|} \frac{|D_R|}{|D|} (p_L - p_R)^2.$$

并用此解释“好的划分让两侧更纯”的几何直觉。

- (b) 回归树中，设父节点方差为  $s^2$ ，子节点方差为  $s_L^2, s_R^2$ 。证明最大化不纯度下降

$$\Delta I_{\text{Var}} = s^2 - \left( \frac{|D_L|}{|D|} s_L^2 + \frac{|D_R|}{|D|} s_R^2 \right)$$

等价于最小化 SSE，并说明为何“均值作叶值”是最优的。

- (c) 实践：随机生成一维连续特征与二分类标签（可通过预设阈值关系生成），在所有相邻取值的中点处扫描阈值，绘制  $\Delta I_{\text{Gini}}(\tau)$  曲线，确定最优阈值并可视化分裂前后的类别分布比例。

### 2. 随机森林的方差与 OOB

- (a) 设森林包含  $B$  棵同分布基学习器，单棵方差为  $\sigma^2$ ，两两相关系数为  $\rho$ 。推导集成平均的方差

$$\text{Var}\left(\frac{1}{B} \sum_{b=1}^B T_b(x)\right) = \rho \sigma^2 + \frac{1-\rho}{B} \sigma^2,$$

并讨论  $B \rightarrow \infty$  与  $\rho \rightarrow 0$  的极限。

- (b) 证明自助采样下，样本  $i$  未被抽入某棵树的概率为  $(1 - 1/n)^n \rightarrow e^{-1}$ ，并给出袋外（OOB）预测的无偏性直觉：为什么 OOB 可近似样本外预测？
- (c) 实践：在一个小面板数据集上绘制袋外（OOB）误差随树数量  $B$  的变化曲线，观察是否出现“平台段”，并给出选择  $B$  的经验性标准。

### 3. XGBoost 的二阶目标与分裂增益

- (a) 在二阶泰勒近似下，给定每个叶的梯度/海森聚合  $(G_j, H_j)$ ，推导最优叶权重  $w_j^* = -\frac{G_j}{H_j + \lambda}$  与近似目标值  $\tilde{\mathcal{L}} = -\frac{1}{2} \sum_{j=1}^T \frac{G_j^2}{H_j + \lambda} + \gamma T$ 。

- (b) 对一次候选分裂，计算分裂增益

$$\text{Gain} = \frac{1}{2} \left( \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right) - \gamma,$$

并给出“仅当  $\text{Gain} > 0$  才分裂”的充要条件。

(c) **思考：**解释单调性约束 (monotonicity constraints) 如何在分裂搜索中被实现；它与经济学先验知识（如“利率上升  $\Rightarrow$  违约概率上升”）的关系是什么？

#### 4. 时序交叉验证与 Purged/Embargo 切分

- (a) 设标签为未来  $h$  期收益  $r_{t+1:t+h}$  的函数。说明为什么普通 K 折交叉验证会发生“标签重叠—信息泄漏”。给出清除 (purge) 与禁运 (embargo) 的形式化定义。
- (b) 对总样本长度  $T$ 、标签窗口长度  $h$ 、外层验证窗口长度  $v$  与禁运长度  $e$ ，写出第  $m$  折的训练索引集合，保证训练标签窗与验证标签窗不重叠。
- (c) **实践：**在您的收益率预测框架上，实现带清除/禁运机制的 K 折交叉验证（或滚动窗口清除），比较启用与禁用该机制时的样本外 AUC/夏普比率是否出现显著差异。

#### 5. 早停与岭回归的谱滤波等价

- (a) 在线性最小二乘  $y = X\beta + \varepsilon$  的梯度下降算法中，令  $X = U\Sigma V'$ 。证明第  $t$  步解可写作

$$\hat{\beta}^{(t)} = \sum_{i=1}^r (1 - (1 - \eta\sigma_i^2)^t) \frac{\mathbf{u}_i' \mathbf{y}}{\sigma_i} \mathbf{v}_i,$$

并解释  $g_t(\sigma^2) = 1 - (1 - \eta\sigma^2)^t$  是“谱收缩因子”。

- (b) 比较岭回归的收缩因子  $g_\lambda(\sigma^2) = \sigma^2 / (\sigma^2 + \lambda)$ ，说明为何适当选择  $(\eta, t)$  可模拟岭回归效应（早停  $\approx L_2$  正则化）。
- (c) **思考：**结合“双降”现象谈谈：为何将早停纳入内层搜索而非仅凭单一验证曲线更稳妥？

#### 6. 概率校准与评分规则

- (a) 给出 Brier 分数的分解：不确定性 (uncertainty)、分辨率 (resolution)、可靠性 (reliability) 三部分，并解释“良好校准”对应于哪一部分的减小。
- (b) 证明对数损失与 Brier 分数都是恰当评分规则 (proper scoring rule)：在真实概率  $p^*$  下，预测  $p^*$  使期望损失最小。
- (c) **实践：**对信用评分或违约预测的验证集概率进行 Platt 与保序校准，分别在测试集报告 AUC、Brier、ECE；对比校准前后的阈值策略收益与混淆矩阵。

#### 7. 阈值—成本对齐与交易成本

- (a) (分类) 在误报成本  $c_{FP}$ 、漏报成本  $c_{FN}$  下，推导最优阈值  $\tau^* = \frac{c_{FP}}{c_{FP} + c_{FN}}$ 。
- (b) (择时) 设仓位  $w_t = \mathbf{1}\{p_t > \tau\}$ ，含单边成本  $c$  的策略收益  $r_{t+1}^\pi = w_t r_{t+1} - c |w_t - w_{t-1}|$ 。给出“阈值偏移  $\Delta\tau$ ”对换手率与收益的一阶近似影响（可用小扰动展开），解释为何成本越高，最优阈值越偏离 0.5。
- (c) **实践：**在收益率预测框架上，把阈值作为内层窗口的优化变量（直接最大化样本外 Sharpe），比较固定阈值与自适应阈值的净值曲线与换手率。

#### 8. 波动率建模：GARCH(rolling) vs HAR-RV(log)

- (a) 解释“逐日滚动 + 低频重训”如何缓解 GARCH 模型的“台阶式”预测问题；在您的实现中报告对数波动率 (log-Vol) 的 RMSE 与相关系数。

- (b) 定义并计算  $\sigma_{\text{factor}}, \sigma_{\text{idiosyncratic}}, \sigma_{\text{model}}, \sigma_{\text{true}}$ , 检查模型层与真实层的尺度一致性; 说明为何需要波动率目标 (vol-target)。
- (c) **实践:** 更换波动分布假设 (Student- $t$  分布)、EGARCH、GJR 结构, 比较极端波动阶段的样本外预测效果是否改善。

#### 9. 因子—暴露—VaR/ES 与自助法置信区间

- (a) 用 PCA (或 AE) 提取  $k$  个因子, 在训练窗内对每只资产进行 OLS 回归得到  $\beta, \alpha$  与特质波动, 写出组合对因子暴露  $b_{\text{port}} = \sum_j w_j \beta_j$  及组合方差分解

$$\sigma_{\text{port}}^2 = \mathbf{b}'_{\text{port}} \boldsymbol{\Sigma}_F \mathbf{b}_{\text{port}} + \sum_j w_j^2 \sigma_{\varepsilon,j}^2.$$

- (b) 因子 MC: 在高斯因子模型下给出投资组合 VaR/ES 的估计流程, 并说明 EWMA 协方差估计与波动率目标 (vol-target) 的作用。
- (c) **实践:** 实现“重采样—重估暴露”的自助法, 报告 VaR 与 ES 的分位数区间; 比较 Boot-VaR 中位数、历史法 VaR 与因子 MC VaR 的相对大小, 并给出状态解释。

#### 10. 可解释性与稳定性: SHAP / PDP(ALE) / 漂移与公平

- (a) 写出 SHAP 的局部加和性与一致性公理, 解释“全局重要性 =  $E[|\phi_j|]$ ”与“交互 SHAP”各自回答的问题。
- (b) 在相关特征场景下, 对比 PDP 与 ALE 的差异: 为何 ALE 更不易受低密度区域影响? 给出一维 ALE 的积分表达式并解释“局部梯度积分”的直观意义。
- (c) **实践:** 针对您的分类/回归模型, 报告全局 SHAP 重要性排序及顶层特征 PDP/ALE 曲线; 监控 PSI (或 JS 散度) 并设定再训练触发阈值; 按分组 (如年龄/行业) 报告 AUC/TPR 差异, 并进行基于阈值策略的公平性调整。

#### 11. 现实检验与多重比较: RC/SPA 与 FDR 控制

- (a) White 的现实检验: 设  $d_{t,k}$  为第  $k$  个候选相对基准的损失差, 写出最优者统计量与区块自助法的基本步骤; 说明为何该检验在“模型集合较大”时更保守。
- (b) SPA 检验的改进思想是什么? 简述其对“明显劣质策略”的截尾处理与重心调整, 并说明有限样本功效提升的原因。
- (c) **实践:** 在一组策略/模型家族上, 报告 (i) RC 或 SPA 的  $p$  值; (ii) 对单项指标检验应用 BH 程序的 FDR 控制结果; 讨论“搜索预算—显著性—可重复性”的权衡关系。

## 参考文献

- Adrian, T. & Brunnermeier, M. K. (2016), ‘Covar’, *American Economic Review* **106**(7), 1705–1741.
- Aït-Sahalia, Y. (1996), ‘Testing continuous-time models of the spot interest rate’, *The Review of Financial Studies* **9**(2), 385–426.
- Aït-Sahalia, Y. (2002), ‘Maximum likelihood estimation of discretely sampled diffusions: a closed-form approximation approach’, *Econometrica* **70**(1), 223–262.
- Aït-Sahalia, Y., Fan, J. & Peng, H. (2009), ‘Nonparametric transition-based tests for jump diffusions’, *Journal of the American Statistical Association* **104**(487), 1102–1116.
- Aït-Sahalia, Y., Mykland, P. A. & Zhang, L. (2011), ‘Ultra high frequency volatility estimation with dependent microstructure noise’, *Journal of Econometrics* **160**(1), 160–175.
- Amisano, G. & Giannini, C. (2012), *Topics in structural VAR econometrics*, Springer Science & Business Media.
- Andersen, T. G., Bollerslev, T., Diebold, F. X. & Labys, P. (2003), ‘Modeling and forecasting realized volatility’, *Econometrica* **71**(2), 579–625.
- Anderson, R. L. (1942), ‘Distribution of the serial correlation coefficient’, *The Annals of Mathematical Statistics* **13**(1), 1–13.
- Arrow, K. J. (1965), ‘Aspects of the theory of risk-bearing’.
- Artzner, P., Delbaen, F., Eber, J.-M. & Heath, D. (1999), ‘Coherent measures of risk’, *Mathematical Finance* **9**(3), 203–228.
- Asquith, P. & Mullins Jr, D. W. (1986), ‘Equity issues and offering dilution’, *Journal of Financial Economics* **15**(1-2), 61–89.
- Bachelier, L. (1900), Théorie de la spéculation, in ‘Annales Scientifiques de l’École Normale Supérieure’, Vol. 17, pp. 21–86.
- Backus, D. K., Foresi, S. & Telmer, C. I. (1998), ‘Discrete-time models of bond pricing’, *NBER Macroeconomics Annual* **13**, 57–121.
- Ball, R. & Brown, P. (1968), ‘An empirical evaluation of accounting income numbers’, *Journal of Accounting Research* **6**(2), 159–178.
- URL:** <http://www.jstor.org/stable/2490232>

- Barber, B. M. & Odean, T. (2000), 'Trading is hazardous to your wealth: The common stock investment performance of individual investors', *The Journal of Finance* **55**(2), 773–806.
- Barndorff-Nielsen, O. E., Hansen, P. R., Lunde, A. & Shephard, N. (2008), 'Designing realized kernels to measure the ex post variation of equity prices in the presence of noise', *Econometrica* **76**(6), 1481–1536.
- Barndorff-Nielsen, O. E., Hansen, P. R., Lunde, A. & Shephard, N. (2011), 'Multivariate realised kernels: consistent positive semi-definite estimators of the covariation of equity prices with noise and non-synchronous trading', *Journal of Econometrics* **162**(2), 149–169.
- Barndorff-Nielsen, O. E. & Shephard, N. (2002), 'Estimating quadratic variation using realized variance', *Journal of Applied Econometrics* **17**(5), 457–477.
- Bartlett, M. S. (1950), 'Periodogram analysis and continuous spectra', *Biometrika* **37**(1/2), 1–16.
- Basel Committee on Banking Supervision (2012), Fundamental review of the trading book, Technical report, Bank for International Settlements.
- Becker, R. A. & Chambers, J. M. (1984), *S: an interactive environment for data analysis and graphics*, CRC Press.
- Becker, R. A. & Chambers, J. M. (1985), *Extending the S Systems*, Pacific Grove, CA, USA: Wadsworth & Brooks Cole.
- Bergstra, J. & Bengio, Y. (2012), 'Random search for hyper-parameter optimization', *The journal of machine learning research* **13**(1), 281–305.
- Berman, S. M. (1964), 'Limit theorems for the maximum term in stationary sequences', *Annals of Mathematical Statistics* **35**(2), 502–516.
- Black, F. (1972), 'Capital market equilibrium with restricted borrowing', *The Journal of Business* **45**(3), 444–455.
- Blanchard, O. J. & Quah, D. (1989), 'The dynamic effects of aggregate demand and supply disturbances', *American Economic Review* **79**(4), 655–673.
- Bliss, R. R. (1997), Testing term structure estimation methods, Working Paper 97-12, Federal Reserve Bank of Atlanta.
- Bollerslev, T. (1986), 'Generalized autoregressive conditional heteroskedasticity', *Journal of Econometrics* **31**(3), 307–327.
- Bollerslev, T. & Wooldridge, J. M. (1992), 'Quasi-maximum likelihood estimation and inference in dynamic models with time-varying covariances', *Econometric reviews* **11**(2), 143–172.
- Box, G. E. & Jenkins, G. M. (1976), *Time series analysis. Forecasting and control*.
- Bradley, R. C. (2005), 'Basic properties of strong mixing conditions. a survey and some open questions', *Probability Surveys* **2**, 107–144.

- Brealey, R. A., Myers, S. C. & Allen, F. (2014), *Principles of corporate finance*, McGraw-hill.
- Breiman, L. (2001), ‘Random forests’, *Machine learning* **45**, 5–32.
- Campbell, J. Y., Lo, A. W. & MacKinlay, A. C. (1997), *The econometrics of financial markets*, Princeton University press.
- Campbell, J. Y. & Shiller, R. J. (1991), ‘Yield spreads and interest rate movements: A bird’s eye view’, *Review of Economic Studies* **58**(3), 495–514.
- Campbell, J. Y. & Thompson, S. B. (2008), ‘Predicting excess stock returns out of sample: Can anything beat the historical average?’, *The Review of Financial Studies* **21**(4), 1509–1531.
- Carhart, M. M. (1997), ‘On persistence in mutual fund performance’, *The Journal of Finance* **52**(1), 57–82.
- Carlton, D. W. & Fischel, D. R. (1982), ‘The regulation of insider trading’, *Stan. L. Rev.* **35**, 857.
- Chamberlain, G. (1984), Panel data, in Z. Griliches & M. D. Intriligator, eds, ‘Handbook of Econometrics’, Vol. 2, Elsevier.
- Chambers, J. M. (1998), *Programming with data: A guide to the S language*, Springer Science & Business Media.
- Chambers, J. M. & Hastie, T. J. (2017), Statistical models, in ‘Statistical models in S’, Routledge, pp. 13–44.
- Chaussé, P. (2010), ‘Computing generalized method of moments and generalized empirical likelihood with r’, *Journal of Statistical Software* **34**, 1–35.
- Chen, B. & Hong, Y. (2012), ‘Testing for smooth structural changes in time series models via nonparametric regression’, *Econometrica* **80**(3), 1157–1183.
- Chen, S. X., Gao, J. & Tang, C. Y. (2008), ‘A test for model specification of diffusion processes’, *The Annals of Statistics* **36**(1).
- Chen, T. & Guestrin, C. (2016), Xgboost: A scalable tree boosting system, in ‘Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining’, pp. 785–794.
- Chen, X. & Fan, Y. (2006), ‘Estimation and model selection of semiparametric copula-based multivariate dynamic models’, *Journal of Econometrics* **130**, 307–335.
- Choi, I. (1999), ‘Testing the random walk hypothesis for real exchange rates’, *Journal of Applied Econometrics* **14**(3), 293–308.
- Christensen, K., Kinnebrock, S. & Podolskij, M. (2010), ‘Pre-averaging estimators of the ex-post covariance matrix in noisy diffusion models with non-synchronous data’, *Journal of Econometrics* **159**(1), 116–133.

- Constantinides, G. M. (1992), 'A theory of the nominal term structure of interest rates', *The Review of Financial Studies* **5**(4), 531–552.
- Courtadon, G. (1982), 'The pricing of options on default-free bonds', *Journal of Financial and Quantitative Analysis* pp. 75–100.
- Cox, D. R. (1975), 'Partial likelihood', *Biometrika* **62**(2), 269–276.
- Cox, J. C., Ingersoll Jr, J. E. & Ross, S. A. (1985), 'An intertemporal general equilibrium model of asset prices', *Econometrica* pp. 363–384.
- Dai, Q. & Singleton, K. J. (2000), 'Specification analysis of affine term structure models', *The Journal of Finance* **55**(5), 1943–1978.
- Danielsson, J., Jorgensen, B. N., Samorodnitsky, G., Sarma, M. & de Vries, C. G. (2013), 'Fat tails, var and subadditivity', *Journal of Econometrics* **172**(2), 283–291.
- Darling, D. A. & Siegert, A. J. (1953), 'The first passage problem for a continuous markov process', *The Annals of Mathematical Statistics* pp. 624–639.
- Davison, A. C. & Hinkley, D. V. (1997), *Bootstrap methods and their application*, number 1, Cambridge university press.
- Dekkers, A. L. M., Einmahl, J. H. J. & de Haan, L. (1989), 'A moment estimator for the index of an extreme-value distribution', *The Annals of Statistics* **17**(4), 1833–1855.
- Diebold, F. X. & Mariano, R. S. (1995), 'Comparing predictive accuracy', *Journal of Business & Economic Statistics* **13**(3), 253–263.
- Drost, F. C., Klaassen, C. A. & Werker, B. J. (1997), 'Adaptive estimation in time-series models', *The Annals of Statistics* **25**(2), 786–817.
- Duffie, D. & Kan, R. (1996), 'A yield-factor model of interest rates', *Mathematical Finance* **6**(4), 379–406.
- Efron, B. (1998), 'R. a. fisher in the 21st century', *Statistical Science* **13**(2), 95–122.
- Embrechts, P., Klüppelberg, C. & Mikosch, T. (1997), *Modelling Extremal Events: for Insurance and Finance*, Springer.
- Engle, R. F. (1982), 'Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation', *Econometrica* pp. 987–1007.
- Engle, R. F. & Gonzalez-Rivera, G. (1991), 'Semiparametric arch models', *Journal of Business & Economic Statistics* **9**(4), 345–359.
- Engle, R. F. & Kroner, K. F. (1995), 'Multivariate simultaneous generalized arch', *Econometric Theory* **11**(1), 122–150.
- Engle, R. F. & Manganelli, S. (2004), 'Caviar: Conditional autoregressive value at risk by regression quantiles', *Journal of Business & Economic Statistics* **22**(4), 367–381.
- Epanechnikov, V. A. (1969), 'Non-parametric estimation of a multivariate probability density', *Theory of Probability & Its Applications* **14**(1), 153–158.

- Estrella, A. & Mishkin, F. S. (1997), 'The predictive power of the term structure of interest rates in europe and the united states: Implications for the european central bank', *European Economic Review* **41**(7), 1375–1401.
- Fama, E. (1963), 'Mandelbrot and the stable paretian hypothesis', *Journal of Business* **36**, 420–429.
- Fama, E. (1965a), 'The behavior of stock market prices', *Journal of Business* **38**, 34–105.
- Fama, E. (1965b), 'Random walks in stock market prices', *Financial Analysts Journal* **21**, 55–59.
- Fama, E. (1970), 'Efficient capital markets: a review of theory and empirical work', *Journal of Finance* **25**, 383–417.
- Fama, E. & Blume, M. (1966), 'Filter rules and stock market trading profits', *Journal of Business* **39**, 226–241.
- Fama, E. F. (1965c), 'The behavior of stock-market prices', *Journal of Business* **38**(1), 34–105.
- Fama, E. F. & Bliss, R. R. (1987), 'The information in long-maturity forward rates', *American Economic Review* **77**(4), 680–692.
- Fama, E. F. & French, K. R. (1993), 'Common risk factors in the returns on stocks and bonds', *Journal of Financial Economics* **33**(1), 3–56.
- Fama, E. F. & French, K. R. (2015), 'A five-factor asset pricing model', *Journal of Financial Economics* **116**(1), 1–22.
- Feller, W. (1991), *An introduction to probability theory and its applications, Volume 2*, Vol. 81, John Wiley & Sons.
- Fisher, M., Nychka, D. W. & Zervos, D. (1995), 'Fitting the term structure of interest rates with smoothing splines', *Finance and Stochastics* **1**(2), 107–134.
- Fisher, R. A. & Tippett, L. H. C. (1928), 'Limiting forms of the frequency distribution of the largest or smallest member of a sample', *Proceedings of the Cambridge Philosophical Society* **24**, 180–190.
- Foster, G. (1973), 'Stock market reaction to estimates of earnings per share by company officials', *Journal of Accounting Research* pp. 25–37.
- Fox, J. (1997), *Applied regression analysis, linear models, and related methods.*, Sage Publications, Inc.
- Fox, J. (2009), 'Aspects of the social organization and trajectory of the r project', *The R Journal* **1**(2), 5.
- Gabaix, X. & Ibragimov, R. (2011), 'Rank- 1/2: a simple way to improve the ols estimation of tail exponents', *Journal of Business & Economic Statistics* **29**(1), 24–39.

- Glosten, L. R., Jagannathan, R. & Runkle, D. E. (1993), ‘On the relation between the expected value and the volatility of the nominal excess return on stocks’, *The journal of finance* **48**(5), 1779–1801.
- Gnedenko, B. (1943), ‘Sur la distribution limite du terme maximum d’une série aléatoire’, *Annals of Mathematics* **44**, 423–453.
- Gourioux, C., Monfort, A. & Polimenis, M. (2006), ‘Affine models for discrete-time interest rates’, *Journal of Econometrics* **131**(1–2), 39–73.
- Greene, W. H. (2000), *Econometric Analysis (4th Edition)*, Prentice Hall, New Jersey.
- Hald, A. (1998), *A History of Mathematical Statistics*, Wiley, New York.
- Hall, P. & Heyde, C. C. (2014), *Martingale limit theory and its application*, Academic press.
- Hamilton, J. D. (1994), *Time Series Analysis*, Princeton University Press, Princeton, NJ.
- Hamilton, J. D. & Lin, G. (1996), ‘Stock market volatility and the business cycle’, *Journal of applied econometrics* **11**(5), 573–593.
- Hamilton, J. D. & Susmel, R. (1994), ‘Autoregressive conditional heteroskedasticity and changes in regime’, *Journal of econometrics* **64**(1–2), 307–333.
- Hansen, L. P. (1982), ‘Large sample properties of generalized method of moments estimators’, *Econometrica* pp. 1029–1054.
- Hansen, P. R. & Lunde, A. (2006), ‘Realized variance and market microstructure noise’, *Journal of Business & Economic Statistics* **24**(2), 127–161.
- Härdle, W. & Linton, O. (1994), ‘Applied nonparametric methods’, *Handbook of Econometrics* **4**, 2295–2339.
- Harris, F. H. d., McInish, T. H., Shoesmith, G. L. & Wood, R. A. (1995), ‘Cointegration, error correction, and price discovery on informationally linked security markets’, *Journal of Financial and Quantitative Analysis* **30**(4), 563–579.
- Hart, J. D. (1996), ‘Some automated methods of smoothing time-dependent data’, *Journal of Nonparametric Statistics* **6**(2–3), 115–142.
- Hautsch, N. & Podolskij, M. (2013), ‘Preaveraging-based estimation of quadratic variation in the presence of noise and jumps: theory, implementation, and empirical evidence’, *Journal of Business & Economic Statistics* **31**(2), 165–183.
- Hayn, C. (1995), ‘The information content of losses’, *Journal of accounting and economics* **20**(2), 125–153.
- Herrndorf, N. (1984), ‘A functional central limit theorem for weakly dependent sequences of random variables’, *The Annals of Probability* pp. 141–153.
- Hill, J. B. (2006), ‘An estimator for the asymptotic variance of the hill estimator’, *Econometric Theory* **22**(3), 578–592.

- Hill, J. B. (2010), ‘Robust estimation of tail thickness under dependence’, *Journal of Econometrics* **158**(1), 68–75.
- Hong, Y. & Li, H. (2005), ‘Nonparametric specification testing for continuous-time models with applications to term structure of interest rates’, *The Review of Financial Studies* **18**(1), 37–84.
- Hong, Y., Linton, O., McCabe, B., Sun, J. & Wang, S. (2024), ‘Kolmogorov–smirnov type testing for structural breaks: A new adjusted-range based self-normalization approach’, *Journal of Econometrics* **238**(2), 105603.
- Hornik, K. (2012), ‘The comprehensive r archive network’, *Wiley Interdisciplinary Reviews: Computational Statistics* **4**(4), 394–398.
- Ibragimov, R. (2009), ‘Portfolio diversification and value at risk under thick-tailedness’, *Operations Research Letters* **37**(2), 162–168.
- Ihaka, R. (1998), ‘R: Past and future history’, *Computing Science and Statistics* **39**2396.
- Jacod, J., Li, Y., Mykland, P. A., Podolskij, M. & Vetter, M. (2009), ‘Microstructure noise in the continuous case: the pre-averaging approach’, *Stochastic Processes and Their Applications* **119**(7), 2249–2276.
- Jacod, J. & Protter, P. (1998), ‘Asymptotic error distributions for the euler method for stochastic differential equations’, *The Annals of Probability* **26**(1), 267–307.
- James, G., Witten, D., Hastie, T. & Tibshirani, R. (2013), *An Introduction to Statistical Learning: with Applications in R*, Springer, New York.
- Jensen, B. & Poulsen, R. (1999), *A comparison of approximation techniques for transition densities of diffusion processes*, Citeseer.
- Jensen, M. C. (1969), ‘Risk, the pricing of capital assets, and the evaluation of investment portfolios’, *The Journal of Business* **42**(2), 167–247.
- Jorion, P. (1997), *Value at Risk: The New Benchmark for Managing Financial Risk*, McGraw–Hill.
- Keown, A. J. & Pinkerton, J. M. (1981), ‘Merger announcements and insider trading activity: An empirical investigation’, *The Journal of Finance* **36**(4), 855–869.
- Kilian, L. (2009), ‘Not all oil price shocks are alike: Disentangling demand and supply shocks in the crude oil market’, *American Economic Review* **99**(3), 1053–1069.
- Kilian, L. & Park, C. (2009), ‘The impact of oil price shocks on the us stock market’, *International Economic Review* **50**(4), 1267–1287.
- Koenker, R. (2005), *Quantile Regression*, Cambridge University Press.
- Lehmann, E. L. & Casella, G. (2006), *Theory of Point Estimation*, Springer Science & Business Media.

- Lim, K.-P. & Brooks, R. (2011), ‘The evolution of stock market efficiency over time: A survey of the empirical literature’, *Journal of Economic Surveys* **25**(1), 69–108.
- Lintner, J. (1965), ‘Security prices, risk, and maximal gains from diversification’, *The Journal of Finance* **20**(4), 587–615.
- Linton, O. (1993), ‘Adaptive estimation in arch models’, *Econometric Theory* **9**(4), 539–569.
- Linton, O. (2019), *Financial econometrics*, Cambridge University Press.
- Lo, A. W. (2008), Efficient markets hypothesis, in S. N. Durlauf & L. E. Blume, eds, ‘The New Palgrave Dictionary of Economics Online’, 2 edn, Palgrave Macmillan, New York.
- Lo, A. W. & MacKinlay, A. C. (1988), ‘Stock market prices do not follow random walks: Evidence from a simple specification test’, *The Review of Financial Studies* **1**(1), 41–66.
- Lütkepohl, H. (2005), *New introduction to multiple time series analysis*, Springer Science & Business Media.
- Mandelbrot, B. (1963), ‘The variation of certain speculative prices’, *Journal of Business* **36**(4), 394–419.
- Manne, H. (1966), *Insider Trading and the Stock Market*, The Free Press, New York.
- Markowitz, H. M. (1959), *Portfolio Selection: Efficient Diversification of Investments*, John Wiley & Sons, New York.
- Marsh, T. A. & Rosenfeld, E. R. (1983), ‘Stochastic processes for interest rates and equilibrium bond prices’, *The Journal of Finance* **38**(2), 635–646.
- McCulloch, J. H. (1971), ‘Measuring the term structure of interest rates’, *Journal of Business* **44**(1), 19–31.
- McCulloch, J. H. (1986), ‘Simple consistent estimators of stable distribution parameters’, *Communications in Statistics – Simulation and Computation* **15**(4), 1109–1136.
- McCulloch, J. H. & Kwon, H.-C. (1993), U.s. term structure data, 1947–1991, Working paper, Ohio State University.
- McLeod, A. I. & Li, W. K. (1983), ‘Diagnostic checking arma time series models using squared-residual autocorrelations’, *Journal of Time Series Analysis* **4**(4), 269–273.
- McQueen, G. & Thorley, S. (1993), ‘Asymmetric business cycle turning points’, *Journal of Monetary Economics* **31**(3), 341–362.
- Mehra, R. & Prescott, E. C. (1985), ‘The equity premium: A puzzle’, *Journal of Monetary Economics* **15**(2), 145–161.
- Merton, R. C. (1973), ‘An intertemporal capital asset pricing model’, *Econometrica* pp. 867–887.
- Meulbroek, L. K. (1992), ‘An empirical analysis of illegal insider trading’, *The Journal of Finance* **47**(5), 1661–1699.

- Morgan, J. et al. (1997), ‘Creditmetrics-technical document’, *JP Morgan, New York* **1**, 102–127.
- Mykland, P. A. & Zhang, L. (2012), The econometrics of high-frequency data, in M. Kessler, A. Lindner & M. Sørensen, eds, ‘Statistical Methods for Stochastic Differential Equations’, Chapman and Hall/CRC, Boca Raton.
- Nadaraya, E. A. (1964), ‘On estimating regression’, *Theory of Probability & Its Applications* **9**(1), 141–142.
- Nelson, C. R. & Siegel, A. F. (1987), ‘Parsimonious modeling of yield curves’, *Journal of Business* **60**(4), 473–489.
- Nelson, D. B. (1990a), ‘Arch models as diffusion approximations’, *Journal of econometrics* **45**(1-2), 7–38.
- Nelson, D. B. (1990b), ‘Stationarity and persistence in the GARCH (1, 1) model’, *Econometric theory* **6**(3), 318–334.
- Nelson, D. B. (1991), ‘Conditional heteroskedasticity in asset returns: A new approach’, *Econometrica* **59**(2), 347–370.
- Newey, W. K. & Steigerwald, D. G. (1997), ‘Asymptotic bias for quasi-maximum-likelihood estimators in conditional heteroskedasticity models’, *Econometrica* pp. 587–599.
- Newey, W. K. & West, K. D. (1987), ‘A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix’, *Econometrica* **55**(3), 703–708.
- Newey, W. K. & West, K. D. (1994), ‘Automatic lag selection in covariance matrix estimation’, *The Review of Economic Studies* **61**(4), 631–653.
- Parzen, E. (1957), ‘On consistent estimates of the spectrum of a stationary time series’, *The Annals of Mathematical Statistics* pp. 329–348.
- Patton, A. J. (2006), ‘Modelling asymmetric exchange rate dependence’, *International Economic Review* **47**(2), 527–556.
- Patton, A. J. (2012), ‘A review of copula models for economic time series’, *Journal of Multivariate Analysis* **110**, 4–18.
- Pedersen, A. R. (1995), ‘A new approach to maximum likelihood estimation for stochastic differential equations based on discrete observations’, *Scandinavian Journal of Statistics* pp. 55–71.
- Poterba, J. M. & Summers, L. H. (1988), ‘Mean reversion in stock prices: Evidence and implications’, *Journal of Financial Economics* **22**(1), 27–59.
- Pratt, J. W. (1964), Risk aversion in the small and in the large, Vol. 32, pp. 122–136.
- Resnick, S. I. & Stărică, C. (1998), ‘Tail index estimation for dependent data’, *Annals of Applied Probability* **8**(4), 1156–1183.

- Ross, S. A. (1976), 'The arbitrage theory of capital asset pricing', *Journal of Economic Theory* **13**(3), 341–360.
- Samuelson, P. A. (1965), 'Proof that properly anticipated prices fluctuate randomly', *Industrial Management Review* **6**, 41–50.
- Scaillet, O. (2004), 'Nonparametric estimation and sensitivity analysis of expected shortfall', *Mathematical Finance* **14**(1), 115–129.
- Scaillet, O. (2005), 'Nonparametric estimation and sensitivity analysis of expected shortfall', *Mathematical Finance* **15**(3), 283–306.
- Schaefer, S. (1981), 'A note on nonparametric estimation of the term structure using bernstein polynomials', *Journal of Finance* **36**(3), 917–930.
- Schuster, E. F. (1985), 'Incorporating support constraints into nonparametric estimators of densities', *Communications in Statistics-Theory and Methods* **14**(5), 1123–1136.
- Schwert, G. W. (1989), 'Why does stock market volatility change over time?', *The journal of finance* **44**(5), 1115–1153.
- Sewell, M. (2011), 'History of the efficient market hypothesis', *Rn* **11**(04), 04.
- Sharpe, W. F. (1964), 'Capital asset prices: A theory of market equilibrium under conditions of risk', *The Journal of Finance* **19**(3), 425–442.
- Silverman, B. W. (1986), *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London.
- Sklar, A. (1959), 'Fonctions de répartition à n dimensions et leurs marges', *Publications de l'Institut de Statistique de l'Université de Paris* **8**, 229–231.
- Solnik, B. H. (1973), 'Note on the validity of the random walk for european stock prices', *The Journal of Finance* **28**(5), 1151–1159.
- Sornette, D. (2012), 'Dragon-kings: Mechanisms, statistical methods and empirical evidence', *The European Physical Journal Special Topics* **205**(1), 1–26.
- Stock, J. H. & Watson, M. W. (2001), 'Vector autoregressions', *Journal of Economic Perspectives* **15**(4), 101–115.
- Svensson, L. E. O. (1994), Estimating and interpreting forward interest rates: Sweden 1992–1994, NBER Working Paper 4871, National Bureau of Economic Research.
- Taleb, N. N. (2007), *The Black Swan: The Impact of the Highly Improbable*, Random House.
- Tsay, R. S. (2013), *Analysis of Financial Time Series*, 3rd edn, Wiley, Hoboken, NJ.
- Tukey, J. W. (1961), Curves as parameters, and touch estimation, in 'Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics', Vol. 4, University of California Press, pp. 681–695.
- Turner, A. (2009), The turner review: A regulatory response to the global banking crisis, Technical report, UK Financial Services Authority.

- Vasicek, O. (1977), ‘An equilibrium characterization of the term structure’, *Journal of Financial Economics* **5**(2), 177–188.
- Vasicek, O. & Fong, H.-L. (1982), ‘Term structure modeling using exponential splines’, *Journal of Finance* **37**(2), 339–348.
- Watson, G. S. (1964), ‘Smooth regression analysis’, *Sankhyā: The Indian Journal of Statistics, Series A* pp. 359–372.
- Whittle, P. (1951), *Hypothesis Testing in Time Series Analysis*, Almquist and Wicksell.
- Wu, C.-F. J. (1986), ‘Jackknife, bootstrap and other resampling methods in regression analysis’, *The Annals of Statistics* **14**(4), 1261–1295.
- Wucker, M. (2016), *The Gray Rhino: How to Recognize and Act on the Obvious Dangers We Ignore*, St. Martin’s Press.
- Zakoian, J.-M. (1994), ‘Threshold heteroskedastic models’, *Journal of Economic Dynamics and Control* **18**(5), 931–955.
- Zhang, L. (2006), ‘Efficient estimation of stochastic volatility using noisy observations: A multi-scale approach’, *Bernoulli* **12**(6), 1019–1043.
- Zhang, L. (2011), ‘Estimating covariation: Epps effect, microstructure noise’, *Journal of Econometrics* **160**(1), 33–47.
- Zhang, L., Mykland, P. A. & Aït-Sahalia, Y. (2005), ‘A tale of two time scales: Determining integrated volatility with noisy high-frequency data’, *Journal of the American Statistical Association* **100**(472), 1394–1411.
- Zhou, B. (1996), ‘High-frequency data and volatility in foreign-exchange rates’, *Journal of Business & Economic Statistics* **14**(1), 45–52.