1

蒙特卡洛方法与人工智能

艾俊 . 巴布, 朱松纯 著
魏平 译

2020 年 3 月 1 日

[1]艾俊 . 巴布是佛罗里达大学统计学教授，佛罗里达州塔拉哈西市, 32306。
朱松纯是加州大学洛杉矶分校计算机科学与统计学教授，加利福尼亚州洛杉矶市，90095。Emails: abarbu@stat.fsu.edu; sczhu@stat.ucla.edu

序言

科学（例如物理，化学和生物学）和工程（例如视觉，图形和机器人）中研究的现实世界系统涉及大量组件之间的复杂交互。对这种系统的表示是高维空间中定义在图上的概率模型，这种模型的解析解通常是难以获得的。因此，蒙特卡罗方法已经作为通用工具，用在了科学和工程的模拟，估计，推理和学习中。毫无疑问，Metropolis 算法是 20 世纪科学实践中最被经常使用的十大算法之一（Dongarra 和 Sullivan，2000）。随着计算能力的不断增长，研究人员正在处理更加复杂的问题并采用更加先进的模型。在 21 世纪科学和工程的发展中，蒙特卡罗方法将继续发挥重要的作用。Hamiltonian 蒙特卡洛和 Langevin 蒙特卡洛在近期发展深度学习中随机梯度下降法中的应用是这个趋势的另外一个例子。

在历史上，一些领域为蒙特卡罗方法的发展做出了贡献。

- *物理和化学*: 早期的 Metropolis 算法（Metropolis，Rosenbluth，Rosenbluth，Teller 和 Teller，1953），模拟退火（Kirkpatrick，Gelatt 和 Vecchi，1983），聚类采样（Swendsen 和 Wang，1987; Edwards 和 Sokal，1988），以及近期用于可视化旋转玻璃模型景观的断开图的工作（Becker 和 Karpus，1997）。

- *概率学和统计学*: 随机梯度（Robin 和 Monro，1951；Younes 1988），Hastings 动力学（Hastings，1970），数据增强（Tanner 和 Wong，1987），可逆跳跃（Green，1995），用于研究生物信息学的动态加权（Wong 和 Liang，1997），以及限制马尔可夫链蒙特卡罗收敛的数值分析（Diaconis 1988；Diaconis 和 Stroock，1991；Liu，1991）。

- *理论计算机科学*: 聚类采样的收敛率（Jerrum 和 Sinclair，1989；Cooper 和 Frieze，1999）。

- *计算机视觉和模式理论*: 用于图像处理的 Gibbs 采样器（Geman 和 Geman，1984），用于分割的 Jump-diffusion（Miller 和 Grenander，1994），用于目标跟踪的 condensation 或粒子滤波算法（Isard 和 Blake，1996），以及近期用于图像分割和解析的数据驱动的马尔可夫链蒙特卡罗（Tu 和 Zhu，2002）和广义 Swendsen-Wang 剪切（Barbu 和 Zhu，2005）。

考虑到这些多样化的领域使用不 同的方法语言，跨学科交流一直很稀少。这给想要使用蒙特卡罗方法的计算机科学与工程领域的从业者带来了一个巨大的挑战。

一方面，有效的蒙特卡罗算法必须探索问题领域的基础结构，因此它们是针对特定领域或问题而很难被领域外的人理解的。例如，物理学中的许多重要著作，如（Swendsen 和 Wang，1987），只有 2-3 页长，并且不包含背景或介绍，这使它们对计算机科学家和工程师显得十分神秘。

另一方面，统计学家发明的一般或领域无关的蒙特卡罗算法被很好地解释，但是当工程师以不依赖于底层模型和表示结构的通用方式来实现它们时，经常发现它们不是很有效。因此，科学家和研究生普

遍有一种误解，认为这些方法太慢并且通常效果不好。这对蒙特卡罗方法不公平，同时对初出茅庐的学生来说也是不幸的。

本书是建立在作者过去 10 年为加州大学洛杉矶分校（UCLA）统计系和计算机科学系学生授课的材料和草稿基础上，为统计学、计算机科学和工程领域的研究人员和研究生编著的。它涵盖了蒙特卡罗计算中广泛的主题，包括在上述四个领域发展的理论基础和直观思想，同时省略了在实践中较少使用或不起作用的小技巧。它使用计算机视觉、图形学和机器学习中的经典问题阐述了蒙特卡罗设计的艺术，因此可以被计算机视觉与模式识别、机器学习、图形学、机器人学和人工智能领域的研究人员用作参考书。

作者要感谢 UCLA 许多在读和已毕业的博士研究生，他们为本书做出了贡献。Mitchell Hill 以他的学位论文工作为基础，为第 9 章、第 10 章和第 11 章做出了贡献，这些工作丰富了本书的内容。Zachary Stokes 润色了手稿中的许多细节。Maria Pavlovskaia、Kewei Tu、Zhuowen Tu、Jacob Porway、Tianfu Wu、Craig Yu、Ruiqi Gao 和 Erik Nijkamp 贡献了作为例子的材料和图形。作者 UCLA 的两位同事 Ying Nian Wu 教授和 Qing Zhou 教授为本书的改进提供了非常宝贵的建议。

作者还要感谢 DARPA、ONR MURI 基金和 NSF 在完成本书过程中的支持。

佛罗里达塔拉哈西, 加利福尼亚洛杉矶　　艾俊 . 巴布

2018 年 9 月　　朱松纯

作者简介

艾俊 . 巴布 2000 年获得俄亥俄州立大学数学博士学位，2005 年获得加州大学洛杉矶分校计算机科学博士学位（师从朱松纯博士）。从 2005 年到 2007 年，他在西门子研究院从事医学成像研究工作，从开始担任研究科学家到后来升任项目经理。由于在边缘空间学习方面的工作，他与西门子的合作者获得了 2011 年 Thomas A. Edison 专利奖。2007 年，他加入佛罗里达州立大学统计系，从助理教授到副教授到 2019 年担任教授。他发表了 70 多篇关于计算机视觉、机器学习和医学成像方面的论文，并拥有超过 25 项与医学成像和图像去噪相关的专利。

朱松纯 1996 年获得哈佛大学计算机科学博士学位，是加州大学洛杉矶分校统计学与计算机科学教授，担任加州大学洛杉矶分校视觉、学习、认知与自主机器人中心主任。他一直以来的研究兴趣是为视觉和智能探寻一个统一的统计与计算框架，包括作为学习与推理的统一表达和数字蒙特卡洛方法的时空因果与或图（STC-AOG）。他在计算机视觉、统计学习、认知、人工智能和自主机器人领域发表了 260 多篇学术论文。他曾获得了多项荣誉，2003 年因图像解析的工作获 David Marr 奖，1999 年因纹理建模、2007 年因物体建模两次获得 David Marr 奖提名。2001 年，他获得了 NSF 职业奖、ONR 青年研究员奖和斯隆奖。因为在" 视觉模式的概念化、建模、学习和推理的统一基础方面的贡献"，他 2008 年获得了国际模式识别协会授予的 J.K. Aggarwal 奖。2013 年，他关于图像分割的论文获得了 Helmholtz Test-of-Time 奖。他 2011 年当选 IEEE Fellow。作为项目负责人，他领导了多个 ONR MURI 和 DARPA 团队，从事统一数学框架下的场景和事件理解以及认知机器人的工作。

目录

1 蒙特卡罗方法简介 **1**

1.1 动机和目标 . . . 1

1.2 蒙特卡罗计算中的任务 . . . 3

1.2.1 任务 1: 采样和模拟 . . . 3

1.2.2 任务 2: 通过蒙特卡罗模拟估算数量 . . . 5

1.2.3 任务 3: 优化和贝叶斯推理 . . . 7

1.2.4 任务 4: 学习和模型估计 . . . 8

1.2.5 任务 5: 可视化景观 . . . 9

2 顺序蒙特卡罗 **15**

2.1 采样一维密度 . . . 15

2.2 重要性抽样和加权样本 . . . 16

2.3 顺序重要性抽样 (SIS) . . . 19

2.3.1 应用：自避行走的次数 . . . 19

2.3.2 应用：用于视频中目标跟踪的粒子滤波 . . . 21

2.3.3 SMC 框架总结 . . . 24

2.4 应用：通过 SMC 进行光线追踪 . . . 25

2.4.1 示例：光泽高光 . . . 26

2.5 在重要性采样中保持样本多样性 . . . 28

2.5.1 Parzen 窗讨论 . . . 30

2.6 蒙特卡罗树搜索 . . . 32

2.6.1 纯蒙特卡罗树搜索 . . . 33

2.6.2 AlphaGo . . . 34

3 马尔可夫链蒙特卡罗 - 基础 **39**

3.1 马尔可夫链基础 . . . 39

3.2 转移矩阵的拓扑：相通与周期 . . . 42

3.3 Perron-Frobenius 定理 . . . 44

3.4 收敛措施 . . . 46

3.5 连续或异构状态空间中的马尔可夫链 . . . 48
3.6 各态历经性定理 . . . 49
3.7 通过模拟退火进行 MCMC 优化 . . . 49
3.7.1 网页排序示例 . . . 51

4 Metropolis 方法和变体 **57**
4.1 Metropolis-Hastings 算法 . . . 57
4.1.1 原始 Metropolis-Hastings 算法 . . . 58
4.1.2 Metropolis-Hastings 算法的另一版本 . . . 59
4.1.3 其他接受概率设计 . . . 60
4.1.4 Metropolis 设计中的关键问题 . . . 60
4.2 独立 Metropolis 采样 . . . 60
4.2.1 IMS 的特征结构 . . . 61
4.2.2 有限空间的一般首中时 . . . 62
4.2.3 IMS 击中时间分析 . . . 62
4.3 可逆跳跃和跨维 MCMC . . . 64
4.3.1 可逆跳跃 . . . 64
4.3.2 简单示例：1 维范围图像分割 . . . 65
4.4 应用：计算人数 . . . 68
4.4.1 标记点过程模型 . . . 68
4.4.2 MCMC 推理 . . . 69
4.4.3 结果 . . . 70
4.5 应用：家具布置 . . . 70
4.6 应用：场景合成 . . . 72
4.7 练习 . . . 74

5 吉布斯采样器及其变体 **79**
5.1 引言 . . . 79
5.2 吉布斯采样器 . . . 80
5.2.1 吉布斯采样器的一个主要问题 . . . 82
5.3 Gibbs 采样器泛化 . . . 83
5.3.1 击中逃跑 . . . 83
5.3.2 广义 Gibbs 采样器 . . . 83
5.3.3 广义击中逃跑 . . . 84
5.3.4 利用辅助变量采样 . . . 85
5.3.5 模拟退火 . . . 85
5.3.6 切片采样 . . . 86
5.3.7 数据增强 . . . 86
5.3.8 Metropolized 吉布斯采样器 . . . 87

5.4 数据关联和数据增强 . . . 89
5.5 Julesz 系综和 MCMC 纹理采样 . . . 90
5.5.1 Julesz 系综 - 纹理的数学定义 . . . 91
5.5.2 Gibbs 系综和系综等价性 . . . 93
5.5.3 Julesz 系综采样 . . . 93
5.5.4 实验：对 Julesz 系综进行采样 . . . 94

6 聚类采样方法 99
6.1 Potts 模型和 Swendsen-Wang . . . 100
6.2 SW 算法的解释 . . . 102
6.2.1 解释 1：Metropolis-Hastings 观点 . . . 102
6.2.2 解释 2: 数据增强 . . . 105
6.3 一些理论成果 . . . 108
6.4 任意概率的 Swendsen-Wang 切割 . . . 110
6.4.1 步骤 1: 数据驱动的聚类 . . . 110
6.4.2 Step 2: 颜色翻转 . . . 111
6.4.3 Step 3: 接受翻转 . . . 112
6.4.4 复杂性分析 . . . 113
6.5 集群抽样方法的变体 . . . 114
6.5.1 集群 Gibbs 采样 — "hit-and-run" 观点 . . . 114
6.5.2 多重翻转方案 . . . 115
6.6 应用: 图像分割 . . . 115
6.7 多重网格和多级 SW 切割 . . . 117
6.7.1 多重网格上的 SW-cuts . . . 119
6.7.2 多层次 SW-cuts . . . 120
6.8 在子空间聚类 . . . 121
6.8.1 由 Swendsen-Wang 切分的子空间聚类 . . . 122
6.8.2 应用：稀疏运动分割 . . . 124
6.9 C4: 聚类合作竞争约束 . . . 128
6.9.1 C^4 算法综述 . . . 130
6.9.2 图形, 耦合和聚类 . . . 131
6.9.3 平面图上的 C^4 算法 . . . 134
6.9.4 在平面图上的实验 . . . 137
6.9.5 棋盘 Ising 模型 . . . 138
6.9.6 分层图上的 C^4 . . . 141
6.9.7 C^4 分层实验 . . . 143

7 MCMC 的收敛性分析 **151**
7.1 关键融合主题 . . . 151
7.2 实用的监测方法 . . . 152
7.3 卡改组的耦合方法 . . . 154
7.3.1 拖到顶端 . . . 154
7.3.2 Riffle 洗牌 . . . 155
7.4 几何界限，瓶颈和电导 . . . 156
7.4.1 几何收敛 . . . 156
7.5 Peskun 的有序和遍历性定理 . . . 159
7.6 路径耦合和精确采样 . . . 160
7.6.1 从过去耦合 . . . 161
7.6.2 应用：对 Ising 模型进行采样 . . . 162

8 数据驱动的马尔可夫链蒙特卡罗 **167**
8.1 分割问题和 DDMCMC 简介 . . . 167
8.2 DDMCMC 简介 . . . 168
8.2.1 设计 MCMC--基本问题 . . . 170
8.2.2 计算原子空间中的提议概率--原子粒子 . . . 171
8.2.3 计算对象空间中的提议概率--对象粒子 . . . 173
8.2.4 计算多个不同的解--场景粒子 . . . 174
8.2.5 Ψ 世界实验 . . . 174
8.3 问题公式化和图像模型 . . . 175
8.3.1 用于分割的贝叶斯公式 . . . 175
8.3.2 先验概率 . . . 176
8.3.3 灰度图像的可能性 . . . 176
8.3.4 模型校准 . . . 178
8.3.5 彩色图像模型 . . . 179
8.4 解空间分析 . . . 179
8.5 利用遍历马尔可夫链探索解空间 . . . 180
8.5.1 五类马尔可夫链动力学 . . . 181
8.5.2 瓶颈 . . . 182
8.6 数据驱动方法 . . . 183
8.6.1 方法 I: 原子空间中的聚类 . . . 183
8.6.2 方法二：边缘检测 . . . 186
8.7 计算重要性提案概率 . . . 187
8.8 计算多种不同的解决方案 . . . 189
8.8.1 动机和数学原理 . . . 189
8.8.2 用于多种解决方案的 K-adventurers 算法 . . . 190
8.9 图像分割实验 . . . 191

8.10 应用：图像分析 192
8.10.1 自下而上和自上而下的处理 196
8.10.2 生成和判别方法 196
8.10.3 马尔可夫链内核和子内核 197
8.10.4 DDMCMC 和提案概率 198
8.10.5 马尔可夫链核 206
8.10.6 图像解析实验 212

9 汉密尔顿函数和拉文蒙特卡洛算法 221
9.1 哈密顿力学 221
9.1.1 汉密尔顿等式 221
9.1.2 HMC 的简单模型 222
9.2 哈密顿力学的性质 223
9.2.1 节约能源 223
9.2.2 可逆性 224
9.2.3 辛结构和体积保存 224
9.3 Hamilton 方程的 leapfrog 离散化 226
9.3.1 欧拉的方法 226
9.3.2 改进的欧拉方法 226
9.3.3 Leapfrog 积分器 227
9.3.4 Leapfrog 积分器的属性 227
9.4 汉密尔顿蒙特卡洛和朗格文蒙特卡洛 229
9.4.1 HMC 的公式 229
9.4.2 HMC 算法 230
9.4.3 LMC 算法 232
9.4.4 调整 HMC 234
9.4.5 HMC 的详细平衡证明 235
9.5 黎曼流形 HMC 236
9.5.1 HMC 中的线性变换 236
9.5.2 RMHMC 动态 239
9.5.3 RMHMC 算法和变体 240
9.5.4 RMHMC 中的协方差函数 242
9.6 实例中的 HMC 243
9.6.1 受约束正态分布的模拟实验 243
9.6.2 使用 RMHMC 对逻辑回归系数进行抽样 245
9.6.3 使用 LMC 采样图像密度：FRAME，GRADE 和 DeepFRAME 248

10 随机梯度学习 **255**
10.1 随机梯度: 动机和属性 . . . 255
10.1.1 激励案例 . . . 256
10.1.2 Robbins-Monro 定理 . . . 258
10.1.3 随机梯度下降和 Langevin 方程 . . . 259
10.2 马尔可夫随机场 (MRF) 模型的参数估计 . . . 261
10.2.1 学习具有随机梯度的 FRAME 模型 . . . 262
10.2.2 FRAME 的替代学习方法 . . . 263
10.2.3 FRAME 算法的四个变种 . . . 265
10.2.4 实验 . . . 268
10.3 用神经网络学习图像模型 . . . 271
10.3.1 对比发散与持续对比发散 . . . 271
10.3.2 使用深度网络学习图像的势能: DeepFRAME . . . 272
10.3.3 发生器网络和交替后向传播 . . . 275
10.3.4 协作网络和生成器模型 . . . 279

11 绘制能源景观 **287**
11.1 景观结构和任务 . . . 287
11.2 ELM 结构 . . . 290
11.2.1 空间分区 . . . 290
11.2.2 广义 Wang- - Landau 算法法 . . . 290
11.2.3 Constructing the ELM . . . 292
11.2.4 估计 ELM 中节点的质量和体积 . . . 293
11.2.5 表征学习任务的难度（或复杂性） . . . 295
11.2.6 MCMC 在模型空间中移动 . . . 295
11.2.7 ELM 收敛分析 . . . 295
11.3 实验 I：高斯混合模型的 ELM . . . 296
11.3.1 能量和梯度计算 . . . 297
11.3.2 限制 GMM 空间 . . . 298
11.3.3 合成数据的实验 . . . 299
11.3.4 对实际数据的实验 . . . 301
11.4 课程学习 . . . 302
11.4.1 学习依赖语法 . . . 302
11.4.2 能量函数 . . . 303
11.4.3 假设空间的离散化 . . . 303
11.4.4 实验 . . . 305
11.5 用景点 - 扩散映射景观 . . . 306
11.5.1 宏观景观结构和亚稳态 . . . 306
11.5.2 吸引-扩散简介 . . . 307

11.5.3 吸引 -扩散和 Ising 模型 308
11.5.4 吸引 -扩散 ELM 算法 308
11.6 应用：使用吸引 -扩散来映射图像空间 310
11.6.1 图像星系的结构 310
11.6.2 实验 311

第 1 章　蒙特卡罗方法简介

摩纳哥的蒙特卡洛赌场

“生活并不总是一个只有好牌的事情，有时候要把烂牌打好。”- 杰克伦敦

引言

蒙特卡罗方法是以摩纳哥的一个赌场命名的，它使用简单的随机事件模拟复杂的概率事件，例如抛掷一对骰子来模拟赌场的整体商业模型。在蒙特卡罗计算中，一个伪随机数生成器被不断调用并返回区间 $[0,1]$ 中的实数，该结果用于生成一个样本分布，其是所研究的目标概率分布的无偏表达。本章介绍蒙特卡罗的重要概念，包括两个主要类别（顺序和马尔可夫链）和五个目标（模拟，估计，优化，学习和可视化）。本章还给出了每个任务的例子，并且研究了近似计数、射线追踪和粒子滤波等应用。

1.1　动机和目标

一般来说，蒙特卡罗方法分为两类：

- 顺序蒙特卡罗：它通过顺序采样和重要性重新加权来保留和传播一组样本，通常在低维状态空间中。
- 马尔可夫链蒙特卡罗：它通过模拟马尔可夫链探索具有固定概率的状态空间，该固定概率被设计为收敛到一个给定的目标概率。

在工程应用中，例如计算机视觉、图形学和机器学习，目标函数是定义在图表达上的。研究人员在以下三种类型的建模和计算范式之间进行选择，权衡模型的精确度和计算的复杂性。

- 具有精确计算的近似模型：该类模型通过拆解循环连接或移除某些能量项来简化表达。一旦底层图成为树或链，动态规划等算法就可用于找到近似问题的精确解。这一类中还包括寻找能量凸近似且使用凸优化算法来搜索全局能量最优的问题。这样的例子包括 L_1-惩罚回归（lasso）[7] 和分类，这里非零模型权重数量的非凸 L_0 惩罚被替换为凸 L_1 惩罚。
- 具有局部计算的精确模型：该类模型保留原始表达和目标函数，但使用如梯度下降等近似算法先找到一个局部解决方案，然后依赖启发式搜索来指导初始状态。
- 具有渐近全局计算的精确模型：该类包含蒙特卡罗方法，它随时间模拟足够大的样本，并大概率收敛到全局最优解。

蒙特卡罗方法已经用于许多不同的任务中，我们将在下一节中以例子详细说明。

1. 模拟一个系统及其概率分布 $\pi(x)$

$$x \sim \pi(x); \tag{1.1}$$

2. 通过蒙特卡洛积分估算一个量

$$c = E_\pi(f(x)] = \int \pi(x) f(x) \mathrm{d}x; \tag{1.2}$$

3. 优化目标函数以找到其模式（最大量或最小量）

$$x^* = \arg\max \pi(x); \tag{1.3}$$

4. 从训练集中学习参数以优化一些损失函数，例如一组样本 $\{x_i, i = 1, 2, ..., M\}$ 的最大似然估计

$$\Theta^* = \arg\max \sum_{i=1}^{M} \log p(x_i; \Theta); \tag{1.4}$$

5. 可视化目标函数的能量景观，从而量化上述任务之一的难度和各种算法的效率。例如，生物学家对蛋白质折叠的能量景观感兴趣。不同的蛋白质具有不同的景观，能量景观的局部最小值可能与某些疾病（例如阿尔茨海默病）有关。在计算机视觉中，学习算法如卷积神经网络（CNN））的能量景观研究理解为什么它们似乎提供了独立于初始化的良好结果（所有局部最小值等效于滤波器的排列?），或者用于其他学习算法理解学习正确模型的困难以及能量景观如何随观察数据的数量而变化。

可以看出，蒙特卡罗方法可用于许多复杂的问题。

1.2 蒙特卡罗计算中的任务

科学（如物理、化学和生物学）和工程（如视觉、图形学、机器学习和机器人学）中研究的现实世界系统涉及大量组件之间的复杂交互。此类系统通常以图来表示，其中图的顶点表示组件，图的边表示交互关系。系统的行为由图上定义的概率模型控制。

例如，在统计物理学中，铁磁材料由经典的 Ising 和 Potts 模型表示 [6]。这些模型还用于计算机视觉中，以吉布斯分布和马尔可夫随机场表示相邻像素之间的依赖性。

一般意义上，我们的观察数据 $\{\mathbf{x}_1,...,\mathbf{x}_n\} \sim f(\mathbf{x})$ 表示来自“真实”概率模型 $f(\mathbf{x})$ 的样本。实际上，$f(\mathbf{x})$ 通常是未知的，只能通过经验样本 $\{\mathbf{x}_1,...,\mathbf{x}_n\}$ 来近似。

1.2.1 任务 1: 采样和模拟

很多时候我们对学习未知的“真实”模型 $f(\mathbf{x})$ 感兴趣，即用一个参数模型 $P(\mathbf{x};\theta)$ 来近似它。在许多情况下，学习一个模型甚至判断学习的参数模型 $P(\mathbf{x};\theta)$ 与真实模型的可比性意味着从参数模型中获取样本 $\mathbf{x} \sim P(\mathbf{x};\theta)$ 并在这些样本上计算某种充分统计量。因此，采样是蒙特卡罗计算的基本任务之一。

例如，我们将二维图像栅格表示为

$$\Lambda = \{(i,j) : 1 \leq i,j \leq N\}. \tag{1.5}$$

每个像素是一个图像强度为 $\mathbf{I}_{(i,j)} \in \{0,...,255\}$ 的顶点。由 $\mathbf{I}_\Lambda$ 表示的图像是由概率 $\pi(\mathbf{I}_\Lambda;\Theta)$ 支配的深层系统的微观态。换句话说，当系统达到动态平衡时，其状态遵循吉布斯分布

$$\mathbf{I}_\Lambda \sim \pi(\mathbf{I}_\Lambda;\Theta) \tag{1.6}$$

其中 Θ 是 K 个参数的向量。Gibbs 分布可写成以下形式，

$$\pi(\mathbf{I}_\Lambda;\Theta) = \frac{1}{Z}\exp\{- < \Theta, H(\mathbf{I}_\Lambda) >\}. \tag{1.7}$$

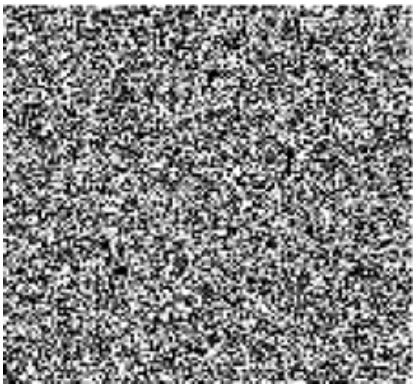
在上面的公式中，Z 是一个归一化常数，$H(\mathbf{I}_\Lambda)$ 是图像 $\mathbf{I}_\Lambda$ 的 K 个充分统计量的向量，内积部分被称为势函数 $U(\mathbf{I}) =< \Theta, H(\mathbf{I}_\Lambda) >$。

当栅格足够大时，概率质量 $\pi(\mathbf{I}_\Lambda;\theta)$ 将集中在一个子空间，在统计物理学中称为*微正则系综* [4]

$$\Omega_\Lambda(\mathbf{h}) = \{\mathbf{I}_\Lambda : H(\mathbf{I}_\Lambda) = \mathbf{h}\}. \tag{1.8}$$

这里，$\mathbf{h} = (h_1,...,h_k)$ 是一个常量向量，称为系统的宏观态。

因此，从分布 $\Omega_\Lambda(\mathbf{h}) \sim \pi(\mathbf{I}_\Lambda;\Theta)$ 中采样无偏样本等价于从系综 $\Omega_\Lambda(\mathbf{h}) \in \Omega_\Lambda(\mathbf{h})$ 中采样。通俗来说，采样过程旨在模拟系统的“典型”微观态。在计算机视觉中，这通常被称为合成 – 一种验证深层模型充分性的方式。



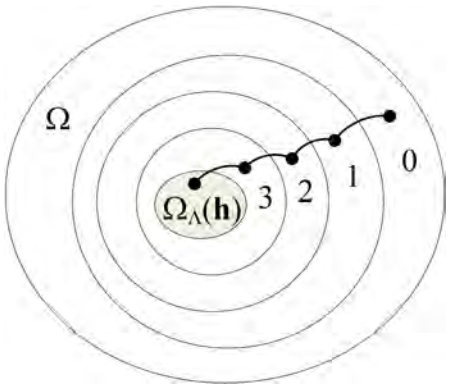


图 1.1: 左：从高斯模型中采样的一个典型图像。右：一组嵌套的系综空间 $\Omega_\Lambda(\mathbf{h})$，受到 $K = 0,1,2,3$ 越来越多的约束。

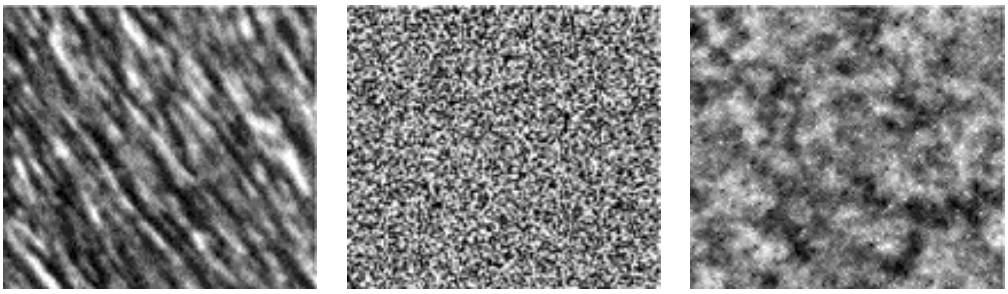
例 1.1 模拟高斯噪声图像。在一个大的栅格上，我们将“高斯噪声”模式定义为具有固定均值和方差的图像系综。

$$\text{高斯噪声} = \Omega_\Lambda(\mu,\sigma^2) = \{\mathbf{I}_\Lambda : \frac{1}{N^2}\sum_{(i,j)\in\Lambda} I(i,j) = \mu, \frac{1}{N^2}\sum_{(i,j)\in\Lambda} (I(i,j)-\mu)^2 = \sigma^2\}.$$

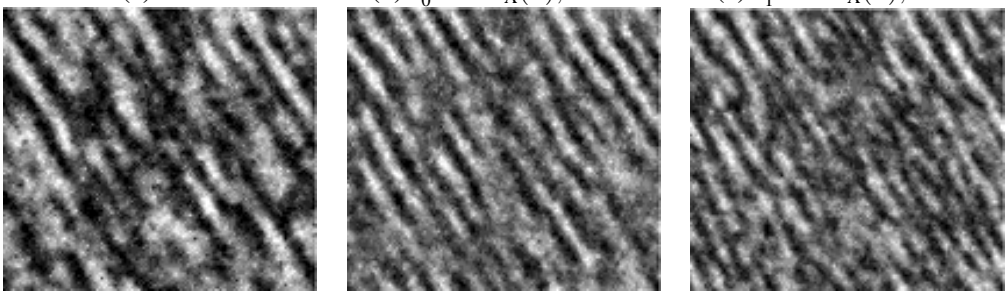
在这种情况下，该模型具有 $K = 2$ 个充分统计量。图 *1.1*显示了一个典型的噪声图像，它为此系综或分布的一个样本。

笔记

为什么最大概率图像 $\mathbf{I}_\Lambda$ 不是一个来自 $\Omega_\Lambda(\mu,\sigma^2)$ 的典型图像？



(a) $\mathbf{I}^{\text{obs}}$ (b) $\mathbf{I}_0^{\text{syn}} \sim \Omega_\Lambda(\mathbf{h}), K = 0$ (c) $\mathbf{I}_1^{\text{syn}} \sim \Omega_\Lambda(\mathbf{h}), K = 1$



(d) $\mathbf{I}_3^{\text{syn}} \sim \Omega_\Lambda(\mathbf{h}), K = 3$ (e) $\mathbf{I}_4^{\text{syn}} \sim \Omega_\Lambda(\mathbf{h}), K = 4$ (f) $\mathbf{I}_7^{\text{syn}} \sim \Omega_\Lambda(\mathbf{h}), K = 7$

图 1.2: 模拟 5 个不同等价类的纹理模式。©[1997] MIT 出版社。经许可重印，来自参考文献 [10]。

例 1.2 模拟纹理模式。正如我们将在后面章节 *5.5*中讨论的那样，每个纹理模式被定义为一个等价类，

$$\text{纹理} = \Omega_\Lambda(\mathbf{h}) = \{\mathbf{I}_\Lambda : H(\mathbf{I}_\Lambda) = \mathbf{h} = (h_1, ..., h_K)\}. \tag{1.9}$$

在这个例子中，充分统计量 $H_k(\mathbf{I}_\Lambda), k = 1, 2, ..., K$ 是 *Gabor* 滤波器的直方图。也就是说，如果任何两个纹理图像共享相同的 *Gabor* 滤波器直方图集，则它们是感知等价的。更详细的讨论参考章节 *5.5*和文献 *[9, 10]*。

图 *1.2*显示了一个纹理建模和模拟的例子，并展示了马尔可夫链蒙特卡罗（*MCMC*）方法的强大能力。自 *20* 世纪 *60* 年代以来，著名的心理物理学家 *Julesz* 研究了纹理感知，提出了一个后来被称为 Julesz 求索的经典问题：

> “这样的一组特征和统计量是什么，即如果两个纹理图像共享相同的特征和统计量，则不能通过前注意处理将这两个图像区分开?”

虽然心理学的兴趣是从一个图像 $\mathbf{I}_\Lambda$ 中找到充分统计量 $\mathbf{h}$，但 *Julesz* 求索提出了一个重大的技术挑战：我们如何为给定的统计量 $\mathbf{h}$ 生成无偏样本？*20* 世纪 *90* 年代后期，*Zhu*、*Wu* 和 *Mumford* 使用马尔可夫链蒙特卡罗方法（*MCMC*）*[10]* 回答了这个问题。图 *1.2*（*a*）是一个观察到的纹理图像 $\mathbf{I}^{\text{obs}}$，从中可以提取任何考虑到的充分统计量 $\mathbf{h}$。为了验证统计量，我们需要从系综或等价的一些 *Gibbs* 分布中抽取典型样本，它们满足 K 特征统计。图 *1.2 (b-f)* 是 $K = 0, 1, 3, 4, 7$ 的例子。每个统计量是汇集所有像素上的 *Gabor* 滤波响应的直方图，并在学习过程中顺序选择 *[10]*。如其所示，使用 $K = 7$ 选择的统计量，生成的纹理图像 $\mathbf{I}_7^{\text{syn}}$ 在感知上等价于观察图像 $\mathbf{I}^{\text{obs}}$，即

$$h_k(\mathbf{I}_7^{\text{syn}}) = h_k(\mathbf{I}^{\text{obs}}), \quad k = 1, 2, ..., 7. \tag{1.10}$$

因此，*MCMC* 方法在解决 *Julesz* 求索中起着关键作用。

1.2.2 任务 2: 通过蒙特卡罗模拟估算数量

在科学计算中，一个常见问题是在极高维空间 Ω 中计算一个函数的积分，

$$c = \int_\Omega \pi(\mathbf{x}) f(\mathbf{x}) d\mathbf{x}. \tag{1.11}$$

这通常通过蒙特卡洛积分来估计。从 $\pi(\mathbf{x})$ 中抽取 M 个样本，

$$x_1, x_2 ..., x_M \sim \pi(\mathbf{x}),$$

我们可以用样本均值来估算 c

$$\bar{c} = \frac{1}{M} \sum_{i=1}^{M} f(\mathbf{x}_i). \tag{1.12}$$

这通常通过顺序蒙特卡罗（SMC）方法完成。我们简要讨论 SMC 的三个例子。

例 1.3 近似计数。在化学中，一个有趣的问题是计算单位面积中的聚合物数量。在蒙特卡洛计算中，这被抽象为一个自避行走（*SAW*）问题。在一个 $N \times N$ 的栅格中，*SAW*$\mathbf{r}$ 是一条不经过任何地点两次的路

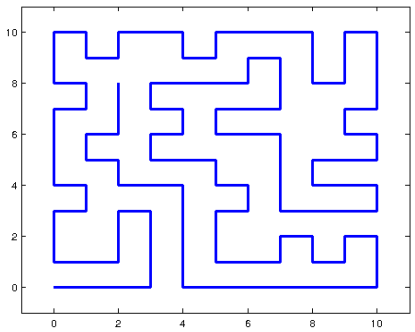


图 1.3: 一个长度为 115 的自避行走.

径。图 *1.3*给出了 *SAW* 的一个例子。我们将 *SAW* 的集合表示为

$$\Omega_{N^2} = \{\mathbf{r} : \mathrm{SAW}(\mathbf{r}) = 1\}. \tag{1.13}$$

其中 $SAW()$ 是一个逻辑指示器。正如我们将在章节 *2*中讨论的那样，Ω_{N^2} 的基数可以通过蒙特卡罗积分来估算，

$$|\Omega_{N^2}| = \sum_{\mathbf{r}\in\Omega_{N^2}} 1 = \sum_{\mathbf{r}\in\Omega_{N^2}} \frac{1}{p(r)} p(r) = E_p[\frac{1}{p(r)}] \approx \frac{1}{M}\sum_{i=1}^{M} \frac{1}{p(\mathbf{r}_i)}. \tag{1.14}$$

在上面的公式中，*SAW* 路径从参考模型 $p(\mathbf{r}_i)$ 通过随机行走来采样，这些随机行走顺序地增长链路。例如，当 $N = 10$ 时，从左下角 $(0,0)$ 到右上角 $(10,10)$ 的 *SAW* 路径的估计数量是 $(1.6 \pm 0.3) \times 10^{24}$。真实的数字是 1.56875×10^{24}。

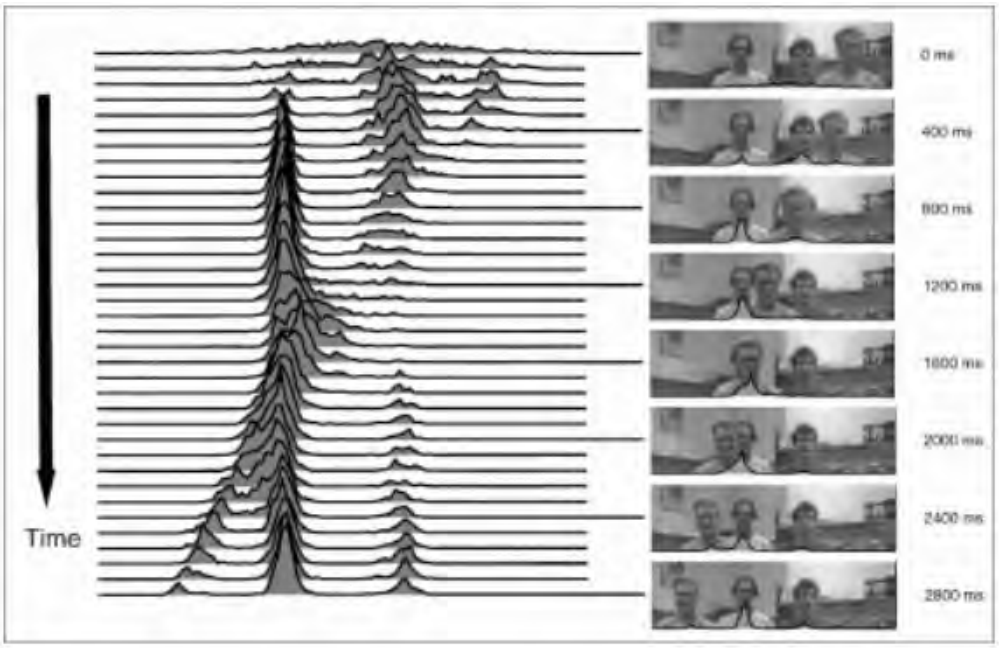


图 1.4: 顺序蒙特卡洛目标跟踪。©[1997] 斯普林格出版社。经许可重印，来自参考文献 [3].

例 1.4 粒子滤波。计算机视觉中一个众所周知的任务是跟踪视频序列中的目标。图 *1.4*是一个简化的例子，其中目标（即这里的人）的位置由水平轴 x 表示，每行是一个时间 t 处的视频帧 $\mathbf{I}(t)$。给定一个输入视频 $\mathbf{I}[0,t]$，在线跟踪的目的是用一组样本近似表示位置的后验概率，

$$S(t) = \{(x_i(t), \omega_i(t)) : i = 1,2,...,M\} \approx \pi(x(t)\,|\,\mathbf{I}[0,t]), \tag{1.15}$$

其中 $\omega_i(t)$ 是 $x_i(t)$ 的权重。如图 *1.4*中的每行所示，$S(t)$ 是对一个非参数分布的编码表示，并通过以下递归积分在时间上传播，

$$\pi(x(t+1)\,|\,\mathbf{I}[0,t+1]) = \int g(\mathbf{I}(t+1)\,|\,x(t+1))p(x(t+1)\,|\,x(t))\cdot\pi(x(t)\,|\,\mathbf{I}[0,t])dx(t). \tag{1.16}$$

在该积分中，$p(x(t+1)\,|\,x(t))$ 是目标运动的动态模型，$g(\mathbf{I}(t+1)\,|\,x(t+1))$ 是衡量位置 $x(t+1)$ 与观测的适应程度的图像似然模型。集合 $S(t)$ 中的每个样本称为粒子。通过表示整个后验概率，样本集 $S(t)$ 保持了实现鲁棒目标跟踪的灵活性。

例 1.5 蒙特卡洛光线追踪。在计算机图形学中，蒙特卡洛积分用来实现渲染图像的光线追踪算法。给定一个具有几何、反射和亮度的三维物理场景，从光源发射的光子将会在物体表面之间反弹，或者在它们撞击成像平面之前穿过透明物体。光线追踪方法通过在所有光源上求和（积分）来计算成像平面上每个像素的颜色和强度，对于这些光源，可以通过像素和各种目标将光线引回到光源。这个过程需要大量计算，可以通过蒙特卡罗积分来近似，我们将在章节中详细介绍 *2*。

1.2.3 任务 3: 优化和贝叶斯推理

自亥姆霍兹（Helmholtz，1860）以来，计算视觉中的一个基本假设是生物和机器视觉计算输入图像中最可能的解释。假设该解释表示为 W，对感知世界我们可以将其表达为一个最大化贝叶斯后验概率的优化问题，

$$W^* = \arg\max \pi(W\,|\,\mathbf{I}) = \arg\max p(\mathbf{I}\,|\,W)p(W), \tag{1.17}$$

其中 $p(W)$ 是真实世界场景如何组织的先验模型，$p(\mathbf{I}|W)$ 是从给定场景 W 生成图像 $\mathbf{I}$ 的似然。

有时图像有多种可能合理的解释，因此在更一般的背景下，我们需要保持多种不同的解释来近似代表后验

$$\{(W_i, \omega_i) : i = 1,2,...,M\} \approx p(W\,|\,\mathbf{I}). \tag{1.18}$$

马尔可夫链蒙特卡罗可用于从后验 $p(W\,|\,\mathbf{I})$ 获取样本; 然而，对后验进行采样与使其最大化并不是一回事。后验也可以通过模拟退火算法最大化，这意味着对 $p(W\,|\,\mathbf{I})^{1/T}$ 采样，其中 T 是在过程中改变的温度参数。在退火过程的开始阶段温度很高，这意味着 $p(W\,|\,\mathbf{I})^{1/T}$ 接近均匀，MCMC 可以自由探索解空间。在退火过程中，根据退火程式温度缓慢降低。随着温度的降低，概率 $p(W\,|\,\mathbf{I})^{1/T}$ 越来越集中在最大位置附近，MCMC 会更仔细地探索这些位置。当温度非常小时，MCMC 应当已经接近后验 $p(W\,|\,\mathbf{I})$ 的一个最大值。

例 1.6 图像分割和解析。图像分割和解析是计算机视觉中的一个核心问题。在此类任务中，由于底层场景复杂性未知，因此 W 中的变量数不定。因此，先验模型 $\pi(W)$ 分布在异构解空间上，该空间是不同

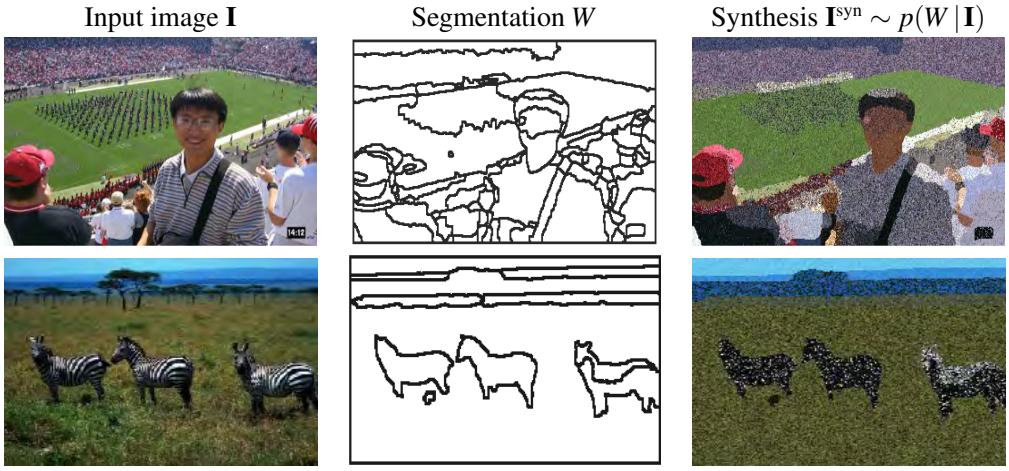


图 1.5: 数据驱动的马尔可夫链蒙特卡罗图像分割。©[1997] IEEE。经许可重印，来自参考文献 [8]。

维度的子空间的并集。当场景中的目标是组合的，W 是一个解析图，且解空间的结构变得更加复杂。在这样复杂的空间中寻找最优解可以通过蒙特卡罗方法来实现，该方法通过混合多种动力学模拟马尔可夫链遍历解空间，这些动力学如死亡与新生，分裂与合并，模型转换和边界扩散。为了提高计算效率，马尔可夫链由使用数据驱动方法计算的边缘分布指导。我们将在章节 *8*中详细介绍。

图 *1.5*展示了由数据驱动的马尔可夫链蒙特卡罗方法计算的两个实例□*[8]*。左列是两个输入图像，分割结果位于中间，每个区域都匹配某个似然模型。为了验证由计算机算法计算的世界 W^*，我们从似然 $\mathbf{I}^{syn} \sim p(W|\mathbf{I})$ 中采样一些典型图像。在上面的示例中，似然不包括面部模型，因此不构造人脸。

1.2.4 任务 4: 学习和模型估计

在统计和机器学习中，我们需要计算能优化某些损失函数的参数，这些函数通常是高度非凸的，尤其是涉及隐变量的时候。在下文中，我们简要讨论两个例子。

例 1.7 学习吉布斯分布。考虑我们在 *1.2.1* 节中提到的 *Gibbs* 模型。为清晰起见，我们省略了栅格符号 Λ

$$p(\mathbf{I};\Theta) = \frac{1}{Z}\exp\{- < \Theta, H(\mathbf{I}) >\}. \tag{1.19}$$

给定一组例子 $\{\mathbf{I}_i^{obs}, i = 1,2,...,M\}$，学习的目的是通过最大化数据的似然来估计参数，

$$\Theta^* = \arg\max \ell(\Theta), \; with \; \ell(\Theta) = \sum_{i=1}^{M} \log p(\mathbf{I}_i^{obs};\Theta). \tag{1.20}$$

损失函数 $\ell(\Theta)$ 相对于 Θ 是凸的。令 $\frac{\partial \ell}{\partial \Theta} = 0$，我们得出以下约束方程，

$$\int H(\mathbf{I})p(\mathbf{I};\Theta)d\mathbf{I} = \mathbf{h} = \frac{1}{M}\sum_{i=1^M} H(\mathbf{I}_i^{obs}). \tag{1.21}$$

Θ 通常须通过随机梯度求解。设 t 表示时间步长，如例子 *1.2* 中的一样，我们使用马尔可夫链蒙特卡罗从当前模型 $p(\mathbf{I};\Theta(t))$ 中采样一组典型样本 $\{\mathbf{I}_i^{syn}, i = 1,2,...,M\}$，并使用样本均值 $\hat{\mathbf{h}}(t) = \frac{1}{M}\sum_{i=1}^{M} H(\mathbf{I}_i^{syn})$ 来

估计期望（即蒙特卡洛积分）。参数通过梯度上升更新，

$$\frac{d\Theta}{dt} = \eta(\mathbf{h} - \hat{\mathbf{h}}(t)), \tag{1.22}$$

其中 η 是步长。

直观上，参数 Θ 被更新，这样根据 $H(\mathbf{I})$ 表示的一些充分统计量不能将观察数据上的分布和模型分布分开。

例 1.8 受限玻尔兹曼机。在深度学习中，受限玻尔兹曼机（*RBM*）是具有二值输入和输出的神经网络。它有一个权重矩阵（即参数）$W = (W_{ij})$ 连接一个可见单元的矢量（输入）$\mathbf{v}$ 和一个隐单元的矢量（输出）$\mathbf{h}$。请注意此表示法与前一个示例中的 $\mathbf{h}$ 含义不同。它还有可见单元和隐单元的偏移量 $\mathbf{a}, \mathbf{b}$。*RBM* 的概率是一个吉布斯分布

$$p(\mathbf{v}, \mathbf{h}; \Theta) = \frac{1}{Z}\exp(-E(\mathbf{v}, \mathbf{h}))$$

它的 *RBM* 能量函数为

$$E(\mathbf{v}, \mathbf{h}; \Theta) = -\mathbf{a}^T\mathbf{v} - \mathbf{b}^T\mathbf{h} - \mathbf{v}^T W\mathbf{h}.$$

使用一组样本 $\mathbf{v}_1, ..., \mathbf{v}_n$ 训练 *RBM* 通常意味着最大化对数似然：

$$\Theta^* = (W, \mathbf{a}, \mathbf{b})^* = \text{argmax}\sum_{i=1}^{n}\log\int p(\mathbf{v}_i, \mathbf{h}; \Theta)d\mathbf{h}$$

这个优化通过与前一个例子相同的蒙特卡罗方式完成。在 *[2]* 中使用的一个变体方法就是所谓的对比散度算法。

1.2.5 任务 5: 可视化景观

在之前的任务中，蒙特卡罗方法用于从目标分布中抽取无偏样本（任务 1），然后这些样本用于通过蒙特卡洛积分来估计未知量（任务 2），并优化状态空间中的一些后验概率（任务 3）或模型空间中的损失函数（任务 4）。使用蒙特卡罗方法的最雄心勃勃的任务是可视化整个能量景观。此能量函数可以是推理任务中 Ω_X 上的负对数后验概率 $-\log p(W|\mathbf{I})$，或者用于学习任务的参数空间中的损失函数 $L\,Theta|Data$。

在现实世界的应用中，这些函数是高度非凸的，有复杂且常常令人震惊的景观，其在高维空间中具有指数倍增数量的局部最小值。图 1.6展示了 K-means 聚类和学习问题中的一个简化的二维能量函数。该能量函数具有不同深度和宽度的多个局部最小值，由字母 $A, B, ...H$ 表示。红色曲线是由具有相同能量水平的点组成的水平集。

任务 5 的目标是使用有效的马尔可夫链蒙特卡罗方法从整个空间抽取有效样本，然后在定位连接相邻盆地的鞍点的同时绘制其能量盆中的所有局部最小值。结果由树形结构表示，物理学家在绘制自旋玻璃模型的景观时称其为非连通图 [1]。在该图中，每个叶节点表示局部最小值，其深度表示能量水平。两个相邻叶节点相遇的能量水平由其鞍点决定。

下文中我们展示了一个学习示例，其中景观在模型空间中而不是在状态空间，因此更难以计算。

例 1.9 数据聚类的景观。*K-means* 聚类是统计和机器学习中的经典问题。给定有限数量的点，其颜色表

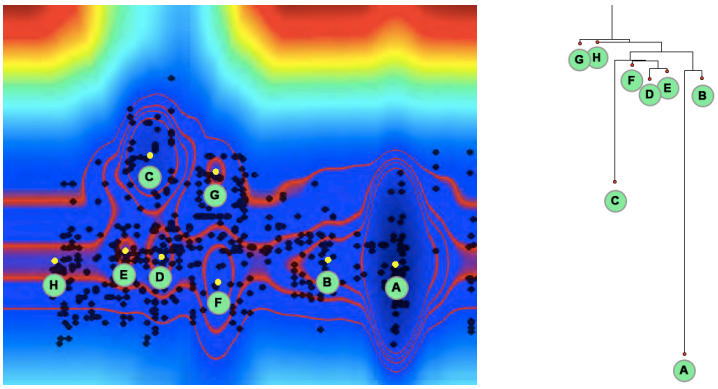


图 1.6: 可视化景观。（左）二维空间中的能量函数。（右）树形表示。深色代表更低的能量。©[2014] Maria Pavlovskaia。经许可重印，来自参考文献 [5]。

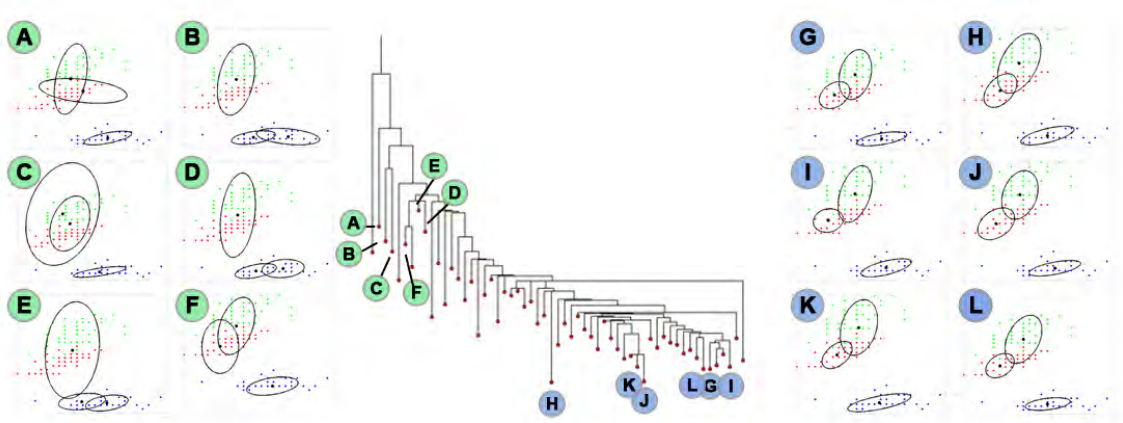


图 1.7: 可视化聚类问题的景观。©[2014] Maria Pavlovskaia。经许可重印，来自参考文献 [5]

示真实标签，学习问题是找到最匹配数据的参数 Θ。这里 Θ 包括 $K=3$ *Gaussian* 模型的均值、方差和权重。能量函数 $\ell(\Theta)$ 是后验概率似然和 Θ 的先验概率。在文献中，流行的算法是 *K-means* 和 *EM* 算法，它们只能找到局部最小值。通过探索每个点是模型 Θ 的空间，我们可以在图形 *1.7*中可视化景观。输入数据来自机器学习中的 *Iris* 数据集。两侧显示了十二个局部最小值 $A,B,...L$，其中每个高斯是一个椭圆。

通过这种景观，人们可以进一步可视化各种算法的行为，并量化目标函数的内在困难，无论是推理还是学习。人们也可以用它来研究影响景观复杂性的关键因素。

例 1.10 用于高斯混合模型的 SWC。设 $\{\mathbf{x}_i \in \mathbb{R}^d, i=1,...,n\}$ 是假定来自具有 k 个多元高斯的混合模型的数据点，对 $i=1,\ldots,K$，其混合权重 α_i 未知，均值为 $\mu_i \in \mathbb{R}^d$，协方差矩阵为 Σ_i。设 Θ 包含所有未知混合参数 $\alpha_i,\mu_i,\Sigma_i, i=1,\ldots,K$。

该高斯混合模型的对数似然（能量）是：

$$\log P(\Theta) = \sum_{i=1}^{n} \log \sum_{j=1}^{K} \alpha_j G(\mathbf{x}_i;\mu_j,\Sigma_j) - \log Z(\Theta), \tag{1.23}$$

这里 $G(\mathbf{x}_i;\mu_j,\Sigma_j) = \frac{1}{\sqrt{\det(2\pi\Sigma_j)}} \exp\left[-\frac{1}{2}(\mathbf{x}_i-\mu_j)^T \Sigma_j^{-1}(\mathbf{x}_i-\mu_j)\right]$ 是高斯分布，$Z(\theta)$ 是归一化常数。

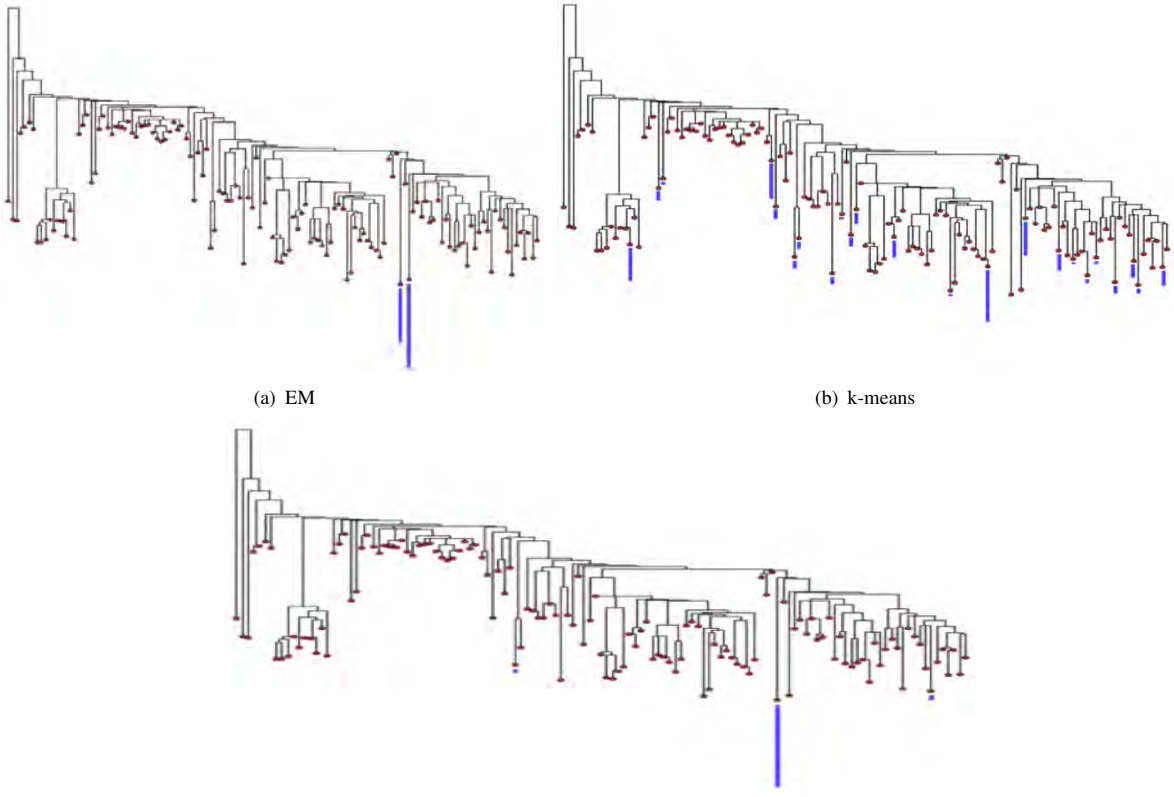
如果已知指向聚类的点的标签，即 $L=(l_1,\ldots,l_n)$，则对数似然是

$$\log P(L,\Theta)=\sum_{j=1}^{K}\sum_{i\in L_j}\log G(\mathbf{x}_i;\mu_j,\Sigma_j)$$

这里 $L_j=\{i,l_i=j\}$.

采样 $P(\Theta)$ 可以通过采样 $P(L,\Theta)$ 并取边缘分布 $P(\Theta)$ 来完成。采样 $P(L,\Theta)$ 可以通过交替采样 $P(L|\Theta)$ 和 $P(\Theta|L)$ 来完成。

对于 $P(L|\Theta)$ 的采样，我们可以使用 *SWC* 算法。我们将 *SWC* 图构造为 *k-NN* 图，并对所有边权重使用常数概率 q。



(a) EM (b) k-means

(c) SW-cut

图 1.8: ELM 上的 EM，k-means 和 SW-cut 算法的性能。由 [5] 提供。

采样 $P(\Theta|L)$ 则更加复杂。首先，我们应该观察到 $P(\Theta|L)=\prod_{j=1}^{K}\prod_{i\in L_j}G(\mathbf{x}_i;\mu_j,\Sigma_j)$ 分裂为独立部分：$P(\Theta|L)=\prod_{j=1}^{K}P(\Theta_j|L_j)$，这里 $\theta_j=(\alpha_j,\mu_j,\Sigma_j)$。因此，我们可以通过采样 $P(\mu_j|L_j,\Sigma_j)$ 和 $P(\mu_j|L_j,\Sigma_j)$ 为每个 j 独立地采样 $P(\Theta_j|L_j)$。现在

$$P(\mu_j|\Sigma_j,L_j)=\prod_{i\in L_j}G(\mathbf{x}_i;\mu_j,\Sigma_j)\propto G(\mu_j,\frac{1}{n_j}\sum_{i\in L_j}\mathbf{x}_i,\frac{1}{n_j}\Sigma_j)$$

只是一个高斯, 其中 $n_j = |L_j|$。还有,

$$P(\Sigma_j|\mu_j, L_j) = \det(\Sigma_j)^{-n_j/2} \exp(-\frac{1}{2}\sum_{i\in L_j}(\mu_j - \mathbf{x}_i)^T\Sigma_j^{-1}(\mu_j - \mathbf{x}_i)) = \det(\Sigma_j)^{-n_j/2}\exp(-\frac{1}{2}\mathrm{tr}(\hat{\Sigma}\Sigma_j^{-1}))$$

其中 $\hat{\Sigma} = \sum_{i\in L_j}(\mu_j - \mathbf{x}_i)(\mu_j - \mathbf{x}_i)^T$，这里我们用到 $\mathrm{tr}(AB) = \mathrm{tr}(BA)$ 且 $A = (\mu_j - \mathbf{x}_i)$，$B = (\mu_j - \mathbf{x}_i)^T\Sigma^{-1}$。由于 $\hat{\Sigma}$ 是对称且正定的，因此存在对称正定 S 使得 $\hat{\Sigma} = S^2$。令 $B = S\Sigma_j^{-1}S$，我们得到

$$P(\Sigma_j|\mu_j, L_j) = \det(\Sigma)^{-n_j/2}\exp(-\frac{1}{2}\mathrm{tr}(S\Sigma^{-1}S)) = \det(S)^{-n_j/2}\det(B)^{n_j/2}\exp(-\frac{1}{2}\mathrm{tr}(B)).$$

令 $B = UDU^T$，其中 $D = diag(\lambda_1, ..., \lambda_d)$ 是对角矩阵，我们得到

$$P(\Sigma_j|\mu_j, L_j) \propto \det(D)^{n_j/2}\exp(-\frac{1}{2}\mathrm{tr}(D)) = \prod_{i=1}^{d}\lambda_i^{n/2}e^{-\lambda_i/2}$$

因此，为了对 Σ_j 进行采样，我们首先从 *Gamma* 分布 $\Gamma(1+\frac{n_j}{2}, 2)$ 中独立地采样特征值 λ_i，以得到 $D = diag(\lambda_1, ..., \lambda_d)$，然后取任意旋转矩阵 U 得到 $B = UDU^T$ 和 $\Sigma_j = SUDU^TS$。图 *1.8*中显示了具有四个混合成分和低可分性的一维高斯混合模型的能量景观。我们可以看出 *k-means* 陷入许多局部最小值中而 *SWC* 总是找到全局最小值。

参考文献

[1] Oren M Becker and Martin Karplus. The topology of multidimensional potential energy surfaces: Theory and application to peptide structure and kinetics. *The Journal of chemical physics*, 106(4):1495–1517, 1997.

[2] Geoffrey Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[3] Michael Isard and Andrew Blake. Condensation: conditional density propagation for visual tracking. *International journal of computer vision*, 29(1):5–28, 1998.

[4] John T Lewis, Charles-Edouard Pfister, and Wayne G Sullivan. Entropy, concentration of probability and conditional limit theorems. *Markov Process. Relat. Fields*, 1(GR-PF-ARTICLE-1995-004):319–386, 1995.

[5] Maria Pavlovskaia, Kewei Tu, and Song-Chun Zhu. Mapping energy landscapes of non-convex learning problems. *arXiv preprint arXiv:1410.0576*, 2014.

[6] Renfrey Burnard Potts. Some generalized order-disorder transformations. In *Proceedings of the Cambridge Philosophical Society*, volume 48, pages 106–109, 1952.

[7] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[8] Zhuowen Tu and Song-Chun Zhu. Image segmentation by data-driven markov chain monte carlo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):657–673, 2002.

[9] Ying Nian Wu, Song Chun Zhu, and Xiuwen Liu. Equivalence of julesz and gibbs texture ensembles. In *ICCV*, volume 2, pages 1025–1032, 1999.

[10] Song Chun Zhu, Ying Nian Wu, and David Mumford. Minimax entropy principle and its application to texture modeling. *Neural Computation*, 9(8):1627–1660, 1997.

第 2 章　顺序蒙特卡罗

“给我一个立足之地，我能用杠杆移动地球" - 阿基米德

引言

顺序蒙特卡罗（SMC）用在目标分布是一维或多维的并可以进行分解的情况。设 $f(x)$ 表示调控一个过程的真实概率分布函数，$\pi(x)$ 表示基于模型的目标概率分布，我们的目标是找到一个使目标密度分布函数 $\pi(x)$ 收敛到 $f(x)$ 的模型。为了找到该模型，一个已知的试验概率密度函数 $g(x)$ 可能被用到。本章涵盖了与 SMC 的 $g(x)$ 选择相关的几个概念，包括样本加权和重要性采样。涵盖的应用包括自避行走，parzen 窗，光线追踪，粒子滤波和光泽高光。本章末尾讨论了蒙特卡罗树搜索。

2.1　采样一维密度

假设 $f(x):\mathbb{R}\to\mathbb{R}$ 是一维概率密度函数 (pdf). 累积密度函数 (cdf) $F(x):\mathbb{R}\to[0,1]$ 定义为

$$F(x)\stackrel{\text{def}}{=}\int_{-\infty}^{x} f(x)\mathrm{d}x$$

我们可以利用均匀样本 u 通过 cdf $F(x)$ 来获得 pdf $f(x)$ 中的样本 $x=F^{-1}(u)$。更严格地我们有

引理 2.1 *假设 $U\sim\text{Unif}[0,1]$ 且 F 是一个一维 pdf f 的 cdf. 则 $X=F^{-1}(U)$ 服从分布 f. 这里我们定义 $F^{-1}(u)=\inf\{x:F(x)\geq u\}$.*

证明 2.1.1

$$P(X\leqslant x)=P(F^{-1}(u)\leqslant x)=P(U\leqslant F(x))=F(x)=\int_{-\infty}^{x} f(x)\mathrm{d}x.$$

根据定义, 我们知道 $\frac{\mathrm{d}u}{\mathrm{d}x}=\frac{\mathrm{d}F(x)}{\mathrm{d}x}=f(x)$, 因此 $P\big(x\in(x_0,x_0+\mathrm{d}x)\big)=P\big(u\in(u_o,u_0+\mathrm{d}u)\big)=f(x)\cdot\mathrm{d}x$.

在更高维空间中，只要可以量化/排序所有数据序列，就可以将 $f(x)$ 简化为一维问题。但是当 $d\geq 3$ 时，我们通常不使用此方法，因为计算复杂度呈指数增长。

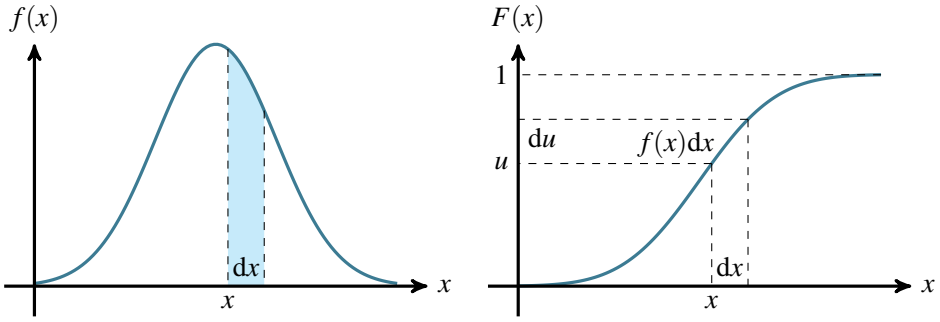


图 2.1: 左: 一个 pdf, $f(x)$. 右: 其对应的 cdf, $F(x)$. 左侧的阴影区域的面积 $f(x)\mathrm{d}x = \mathrm{d}u$.

2.2 重要性抽样和加权样本

假设我们要估计一个量

$$C = \int_{\Omega} \pi(x) \cdot h(x)\mathrm{d}x = E_{\pi}[h(x)] \tag{2.1}$$

其中 $\pi(x)$ 是概率密度函数.

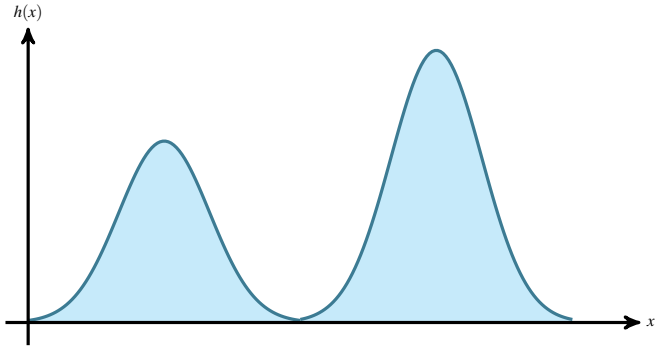


图 2.2: 多模 pdf $h(x)$.

如果我们可以从 $\pi(x)$, $D = \{x_1, x_2, \cdots, x_n\} \sim \pi(x)$ 中抽取样本, 那么就可以很容易的估计 C

$$\hat{C} = \frac{1}{n}\sum_{i=1}^{n} h(x_i)$$

因为 $\pi(x)$ 的信息是 D 中固有的, 我们不需要在公式中写出它。但是，如果难以直接从 $\pi(x)$ 抽取样本，我们可以从更简单的试验分布 $g(x)$ 中较容易地抽取样本，$D' = \{x'_1, x'_2, \cdots, x'_n\}$. 则公式(2.1)可以表达为

$$C = \int_{\Omega} \pi(x) \cdot h(x)\mathrm{d}x = \int_{\Omega} g(x) \cdot \left[\frac{\pi(x)}{g(x)} \cdot h(x)\right]\mathrm{d}x. \tag{2.2}$$

假设比例 $\frac{\pi(x)}{g(x)} \cdot h(x)$ 是可以计算的. 我们可以估计 C 为

$$\hat{C} = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(x_i')}{g(x_i')} \cdot h(x_i') = \frac{1}{n}\sum_{i=1}^{n} \omega(x_i')h(x_i'), \tag{2.3}$$

其中 $\omega(x_i')$ 是样本 i 的权重.

笔记

在公式 (2.3)中, 权重 $\{\omega(x_i'), i = 1,2,\ldots,m\}$ 依赖于分母中的 $g(x_i')$. 因此不论何时 $\pi(x) \neq 0$，我们都不能让 $g(x) = 0$.

设 $\pi(x) = \frac{1}{Z}\exp(-E(x)/T)$, 这里 Z 是归一化常数但是不可计算. 因此 $\pi(x)$ 由加权样本 $\{(x^{(i)}, \omega^{(i)}, i = 1,\cdots,m\}$ 表示.

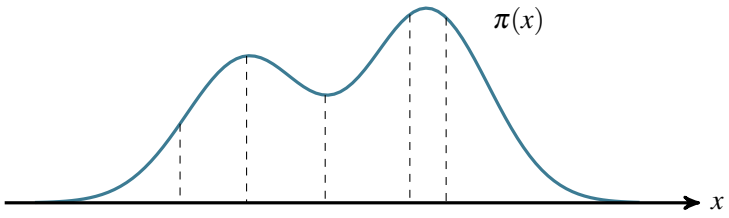


图 2.3: 概率密度函数 $\pi(x)$ 由加权样本近似.

一个特例: 如果

$$g(x) = \text{Unif}[a,b] = \begin{cases} \frac{1}{b-a} & x \in [a,b] \\ 0 & \text{otherwise}, \end{cases}$$

则

$$\hat{C} = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(x_i')}{g(x_i')} \cdot h(x_i') = \frac{b-a}{n}\sum_{i=1}^{n} \pi(x_i') \cdot h(x_i')$$

通常，我们将有以下三种情况:

1) 我们从均匀分布中采样，给每个样本一个特定的权重。

2) 我们从更简单的、$\pi(x)$ 的近似分布 $g(x)$ 中采样，给每个样本一个特定的频率和权重。

3) 我们直接从 $\pi(x)$ 采样，给每个样本一个特定的频率但相同的权重。

很容易证明 1) $\ll$ 2) $\ll$ 3), 其中 "$a \ll b$" 表示 a 比 b 差的多. 直观上，最好的情况是 $g(x) = \pi(x)$. 因为我们要求

$$\lim_{n\to\infty}(\hat{C} - C) = 0,$$

并且这三种情况都要满足这一点，它们之间的唯一区别在于估计器收敛或方差消失所需的样本数目。第二点是,

$$\lim_{n\to\infty} ||\hat{C} - C||^2 = 0.$$

近似分布 $g(x)$ 用作处理 $\pi(x)$ 的杠杆。希腊数学家阿基米德（公元前 212 年）因其著名的言论而闻名:

“给我一个立足之地，我能用杠杆移动地球”.

受他的启发，我们可以把 $g(x)$ 称为 $\pi(x)$ 的阿基米德杠杆。

例 2.1 对于上面的第二种情况，下面是阿基米德杠杆的一个例子。

$$\pi(x)=\frac{1}{Z}\exp\{-\sum_{i=1}^{K}\beta_i h_i(x)\},\qquad g(x)=\frac{1}{Z'}\exp\{-\sum_{i=1}^{K'<K}\beta_i h_i(x)\}$$

例 2.2 在高斯情形中，我们可以使用以下示例中所示的方案

$$\pi(x)=\frac{1}{Z}\exp\{-(ax^2+bx+c)\},\qquad g(x)=\frac{1}{Z'}\exp\{-ax^2\}$$

通常，我们使用来自“经验分布” $g(x)$ 的一组加权样本 $\{(x_i,\omega_i),i=1,2,\ldots,m\}$ 来表示 $\pi(x)$。当 $\mathbf{x}=(x_1,x_2,\ldots,x_n)\in\mathbb{R}^n$ 是高维时，我们可以用两种方式简化它。

$$\begin{aligned} g(\mathbf{x}) &= g(x_1,x_2,\ldots,x_n) \\ &\cong g(x_1)\cdot g(x_2)\cdot\cdots\cdot g(x_n) \quad \text{(通过分解和独立性假设)} &(2.4)\\ &\cong g(x_1)\cdot g(x_2)\cdot g(x_3|x_2)\cdot g(x_4|x_1,x_2)\cdot\cdots \quad \text{(通过分解)} &(2.5)\end{aligned}$$

在 (2.4), 我们假设 x_i 是独立的，我们可以单独地对每个维度进行采样；然而，事实上，它们总是相互依赖的。要更正这一点，我们需要像(2.5)使用条件依赖简化问题。

因为 $\hat{C}=\frac{1}{m}\sum_{i=1}^{m}w(x_i)h(x_i)$, 我们得到 $\mathrm{var}_m(\hat{C})=\frac{1}{m}\mathrm{var}_1(\hat{C})$. 这表明当我们有足够多的样本时，不管维数 n 如何，总的方差会趋于 0 且收敛速度是 $\frac{1}{m}$! 图 2.4中的三个图说明了这个想法。左边分布的收敛速度较快，而中间分布的收敛速度较慢。右边的分布可能会出现权重膨胀至 ∞ 的问题。

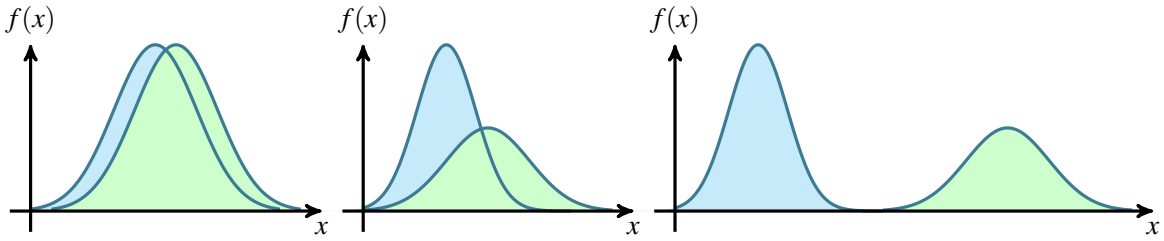


图 2.4: 左: 当 $g(x)$ 和 $\pi(x)$ 接近时, 收敛很快. 中: 当 $g(x)$ 和 $\pi(x)$ 相差很大时, 收敛很慢. 右: 一般情况下, $g(x)$ 在 $\pi(x)$ 是非零的情况下应该是非零的，这在这种情况下可能不会发生。

衡量 $g(x)$ 的样本有效性的启发式方法是测量权重的方差。一个有用的经验法则是使用有效样本量 (ESS) 来测量试验分布与目标分布的差异。假设从 $g(x)$ 生成 m 个独立样本。则 ESS 被定义为

$$\mathrm{ESS}(m)=\frac{m}{1+\mathrm{var}[\omega(x)]} \tag{2.6}$$

在 $g(x)=\pi(x)$ 的理想情况下, 则 $\omega(x)=1$, $\mathrm{var}_g[\omega(x)]=0$, 因此整个样本是有效的。由于目标分布 π

在许多问题中只有归一化常数是已知的，因此归一化权重的方差需要通过非归一化权重的变异系数来估计:

$$\mathrm{cv}^2(\omega) = \frac{1}{m-1}\sum_{i=1}^{m}\frac{(\omega_i - \overline{\omega})^2}{\overline{\omega^2}} \tag{2.7}$$

一般化: 分层抽样 – 一种降低 $\mathrm{var}(\hat{C})$ 的方法. 假设空间 Ω 是许多不相交子空间的并集 $\Omega = \cup_{j=1}^{M}\Omega_j$. 在每一个子空间 Ω_j 中, 我们可以将不同的 $g_j(x)$ 定义为试验分布。因此，我们得到

$$C = \sum_{j=1}^{M}\int_{\Omega_j} g_i(x)\cdot\frac{\pi(x)}{g_i(x)}\cdot h(x)\mathrm{d}x \tag{2.8}$$

在这个计算中，我们可以忽略 $g_i(x)$ 在高维空间中的重叠。

2.3 顺序重要性抽样 (SIS)

在高维空间中，通常很难找到有效的 $g(x)$。假设我们可以通过链式法则将 $\mathbf{x}$ 分解为 $\mathbf{x} = (x_1,\ldots,x_n)$ 。然后我们的试验密度函数可以构造为

$$g(\mathbf{x}) = g_1(x_1)\cdot g_2(x_2|x_1)\cdots g_n(x_n|x_1,\ldots,x_{n-1}). \tag{2.9}$$

通常这样做是不现实的，但在某些情况下，如果 $\pi(x)$ 可以被类似地分解，则可以这样做。对应于 $\mathbf{x}$ 的分解，我们可以将目标密度重写为

$$\pi(\mathbf{x}) = \pi_1(x_1)\cdot\pi_2(x_2|x_1)\cdots\pi_n(x_n|x_1,\ldots,x_{n-1}). \tag{2.10}$$

其重要性权重是

$$\omega(\mathbf{x}) = \frac{g(\mathbf{x}) = g_1(x_1)\cdot g_2(x_2|x_1)\cdots g_n(x_n|x_1,\ldots,x_{n-1})}{\pi(\mathbf{x}) = \pi_1(x_1)\cdot\pi_2(x_2|x_1)\cdots\pi_n(x_n|x_1,\ldots,x_{n-1})} \tag{2.11}$$

在下文中，我们将讨论两个例子。

1) 表达聚合物生长的自避行走

2) 目标跟踪的非线性/粒子滤波

2.3.1 应用：自避行走的次数

二维或三维网格空间中的自避随机行走（SAW）问题是计算一个给定域中存在多少个自避行走的问题。

我们可以使用硬核模型来描述这个问题。一连串的原子 $\mathbf{x} = (x_1, x_2, \cdots, x_N)$ 通过共价键连接。为清楚起见，我们假设每个分子是二维/三维空间/晶格中的一个点，并且键的长度是 1，该势被称为硬核的。在 2D 或 3D 空间中, 链不允许自身相交。

在本节中，我们将集中在二维空间 $\{0,1,...,n\}\times\{0,1,...,n\}$. 假设我们总是从位置 $(0,0)$ 即左下角开始, 如图 2.5所示. 我们分别用数字 1,2,3,4 表示左/右/上/下的移动，SAW 链的吉布斯/玻尔兹曼分布可以

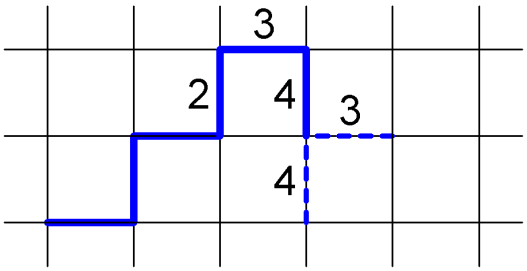


图 2.5: 自避行走的一个例子.

表达为：

$$\pi(x) = \text{unif}[\Omega], \qquad \Omega = \{x : \text{SAW}(x) = 1\}, \quad x \in \{1,2,3,4\}^n.$$

我们对自避随机行走（SAW）的总数感兴趣。为了估计这个值，我们使用蒙特卡罗积分。我们为 SAW x 设计了一个试验概率 $g(x)$，它更容易采样。然后我们从 $g(x)$ 中抽取 M 个 SAW 样本并通过以下方式估计总数

$$||\Omega|| = \theta = \sum_{x\in\Omega} 1 = \sum_{x\in\Omega} \frac{1}{g(x)} g(x) \cong \frac{1}{M} \sum_{i=1}^{M} \frac{1}{g(x_i)}, \tag{2.12}$$

其中 $\frac{1}{g(x_i)}$ 用作 x_i 的权重 $\omega(x_i)$.

试验概率涵盖了所有可能的路径，因此我们可以用它来计算 SAW 集合中许多子集的大小，例如从一个角落开始到另一个角落结束的 SAW 集合，或者长度为 n 的 SAW 集合。我们不需要担心这个新子集下的归一化常数。

因此，问题在于如何设计 $g(x)$, 并且有几种方法可以做到这一点。我们研究了在 $n = 10$ 的二维网格中 $g(x)$ 的三种不同模型表达，来生成 $M = 10^7$ 个样本。

a) 设计 1. 作为初始方法, 我们使用

$$g_1(x) = \prod_{j=1}^{m} \frac{1}{k_j}$$

其中 m 是路径的总长度, k_j 是第 j 次移动的可能选择数, 在步骤 j 我们从 k_j 个选择中均匀地采样. 使用 $M = 10^7$ 个样本, 估计的 SAW 数量是 $K_1 = 3.3844 \cdot 10^{25}$. 采样行走的长度分布如图 2.6所示。由于我们不限制行走的长度，所获得的分布类似于一个高斯分布。

b) 设计 2. 作为试验分布的另一种设计，我们在每个步骤引入提前终止概率 $\varepsilon = 0.1$ 并得到

$$g_2(x) = (1-\varepsilon)^m \prod_{j=1}^{m} \frac{1}{k_j}.$$

很明显，在这种情况下，我们期望得到比设计 1 更短的行走。采样行走的长度分布如图 2.6所示，估计的 SAW 数量是 $K_2 = 6.3852 \cdot 10^{25}$.

c) 设计 3. 对于第三种设计，我们调整设计 1 以支持更长的行走。对于任何超过 50 的行走，我们从该行走中分叉生成 5 个孩子，并以 $w_0 = w/5$ 对每个孩子进行重新加权。采样行走的长度分布如图 2.6所示，估计的 SAW 数量为 $K_3 = 7.3327 \cdot 10^{25}$. 三种设计中最长的 SAW 例子如图 2.7所示。

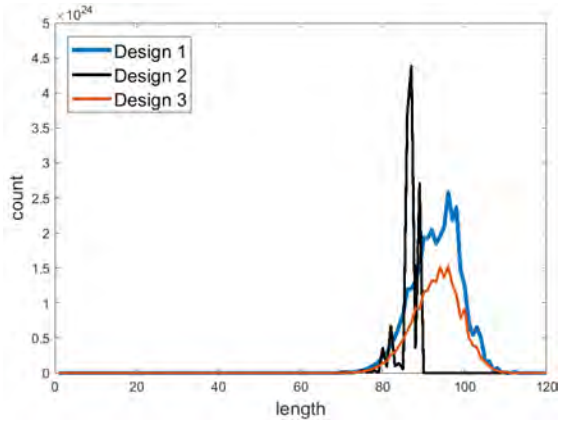


图 2.6: 三种设计的 SAW 长度分布.

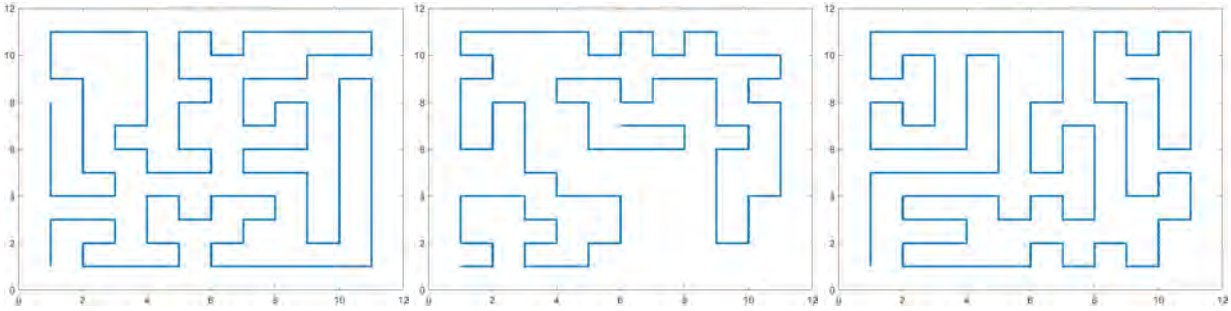


图 2.7: 设计 1 (长度 114, 左), 设计 2 (长度 90, 中间) and 设计 3 (长度 115, 右) 的最长 SAW.

估计的 SAW 数量 K 相对于样本大小 M 的对数-对数图如图 2.8所示。很明显，设计 3 收敛最快，设计 2 收敛最慢。

设计试验概率 $g(x)$ 的其他方法包括:

- 随时停止 (设计 2)
- 固定长度 N
- 丰富样本. 鼓励更长的样本 – 从一定长度开始 (设计 3)
- 全局引导？(即 $(0,0) \to (n,n)$)

我们也有兴趣计算从 $(0,0)$ 到 (n,n) 的 SAW。为了获得到达 (n,n) 的样本，我们重新采样直到获得一个这样的样本，然后给它重新赋予权重 $w_0 = w/u$，其中 u 是尝试的次数。这意味着我们尝试的次数越多，这个样本的权重越低。生成 10^6 个样本，我们估计的从 $(0,0)$ 到 (n,n) 的 SAW 总数约为 $1.7403 \cdot 10^{24}$（非常接近真实值 $1.5687 \cdot 10^{24}$）。

2.3.2 应用：用于视频中目标跟踪的粒子滤波

假设在一个目标跟踪问题中，时间 t 时的状态表示为 $\mathbf{x}_t$，图像的观察特征表示为 $\mathbf{z}_t$。$\mathcal{X}_t = \{\mathbf{x}_1, ..., \mathbf{x}_t\}$ 表示历史状态，$\mathcal{Z}_t = \{\mathbf{z}_1, ..., \mathbf{z}_t\}$ 为历史特征。

Michael Isard

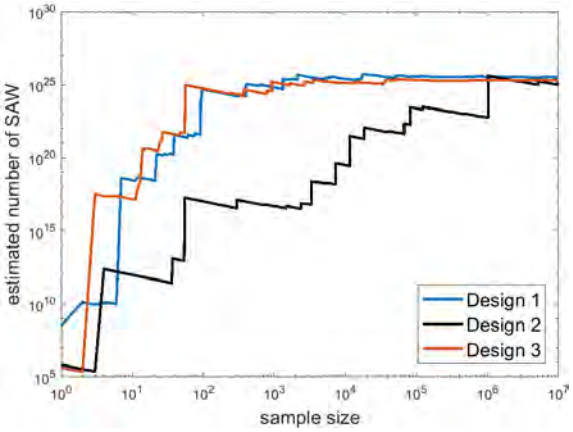


图 2.8: 三种设计的收敛速度比较。

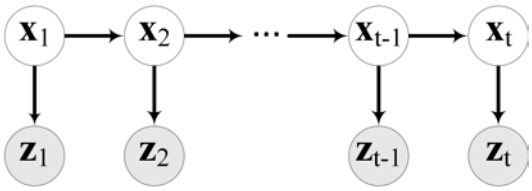


图 2.9: 相对于动态过程 $\mathbf{x}_t$，观测值 $\mathbf{z}_t$ 彼此独立。动态过程 $\mathbf{x}_t$ 仅依赖于先前状态 $\mathbf{x}_{t-1}$。

我们假设目标动态遵循 Markov 链，即当前状态仅取决于前一状态，独立于历史状态，如图 2.9所示。

$$p(\mathbf{x}_{t+1}|\mathcal{X}_t) = p(\mathbf{x}_{t+1}|\mathbf{x}_t)$$

假设相对于动态过程，$\mathbf{z}_t$ 彼此独立，如图 2.9所示。我们需要估计 $p(\mathbf{x}_{t+1}|\mathcal{Z}_{t+1})$，

即以到当前为止收到的数据为条件的状态 x_{t+1} 的分布。因为 $\mathbf{z}_{t+1}$ 与 $\mathcal{Z}_t$ 无关，我们有


Andrew Blake

$$\begin{aligned} p(\mathbf{x}_{t+1}|\mathcal{Z}_{t+1}) = p(\mathbf{x}_{t+1}|\mathbf{z}_{t+1},\mathcal{Z}_t) &= \frac{p(\mathbf{x}_{t+1},\mathbf{z}_{t+1}|\mathcal{Z}_t)}{p(\mathbf{z}_{t+1}|\mathcal{Z}_t)} \\ &\propto p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1},\mathcal{Z}_t)p(\mathbf{x}_{t+1}|\mathcal{Z}_t) = p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1})p(\mathbf{x}_{t+1}|\mathcal{Z}_t) \end{aligned}$$

我们可以计算

$$p(\mathbf{x}_{t+1}|\mathcal{Z}_t) = \int p(\mathbf{x}_{t+1},\mathbf{x}_t|\mathcal{Z}_t)d\mathbf{x}_t = \int p(\mathbf{x}_{t+1}|\mathbf{x}_t)p(\mathbf{x}_t|\mathcal{Z}_t)d\mathbf{x}_t,$$

所以我们得出结论

$$p(\mathbf{x}_{t+1}|\mathcal{Z}_{t+1}) \propto \int p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1})p(\mathbf{x}_{t+1}|\mathbf{x}_t)p(\mathbf{x}_t|\mathcal{Z}_t)d\mathbf{x}_t$$

概率 $p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1})$ 可以认为是自下而上的检测概率，而乘积 $p(\mathbf{x}_{t+1}|\mathbf{x}_t)p(\mathbf{x}_t|\mathcal{Z}_t)$ 是基于动态模型的预测。

条件密度传播 (Condensation) 算法 [4] 利用重要性采样将 $p(\mathbf{x}_t|\mathcal{Z}_t)$ 表示为一个权重为 $\pi_t^{(n)}$ 的加权样本集 $\{\mathbf{s}_t^{(n)}, n = 1,...,N\}$。图 2.10和图 2.11分别解释和描述了算法的一个步骤。

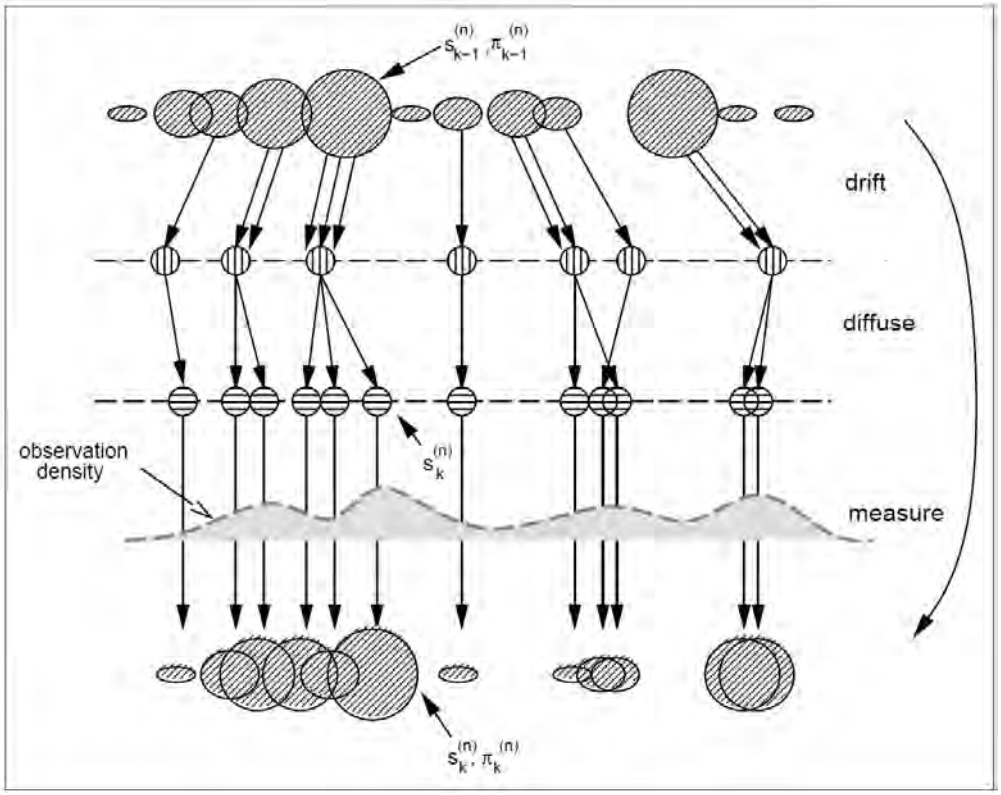


图 2.10: Condensation 算法的一个步骤。©[1998] Springer。经许可重印，来自参考文献 [4].

输入: 样本集合 $\{(\mathbf{s}_{t-1}^{(n)}, \pi_{t-1}^{(n)}), n = 1,...,N\}$
计算 累计分布值 $c_{t-1}^{(k)} = \sum_{i=1}^{k} \pi_{t-1}^{(i)}$。
for $n = 1,\ldots,N$ **do**
　漂移: 从累计分布中 $c_{t-1}^{(k)}, k = 1,...,N$ 采样 $\mathbf{s}_t^{\prime(n)}$。
　扩散: 从动态模型 $\mathbf{s}_t^{(n)} \sim p(\mathbf{x}_t | \mathbf{x}_{t-1} = \mathbf{s}_t^{\prime(n)})$ 中采样 $\mathbf{s}_t^{(n)}$。
　测量和重加权样本 $\mathbf{s}_t^{(n)}$ as $\pi_t^{(n)} = p(\mathbf{z}_t | \mathbf{x}_t = s_t^{(n)})$。
end for
归一化 $\pi_t^{(n)}$ 使 $\sum_{n=1}^{N} \pi_t^{(n)} = 1$.
输出: 样本集合 $\{(\mathbf{s}_t^{(n)}, \pi_t^{(n)}), n = 1,...,N\}$

图 2.11: Condensation 算法的一个步骤 [4]。

应用：曲线跟踪

一条曲线在时间 t 的值 $\mathbf{r}(s,t)$ 可以参数化为一个 B 样条:

$$\mathbf{r}(s,t) = (B(s)Q^x(t), B(s)Q^y(t)), s \in [0,L],$$

其中 $B(s) = (B_1(s),...,B_{N_B}(s))^T$ 是 B 样条基函数向量。向量 $X_t = (Q^x, Q^y)^T$ 包含样条控制点的坐标。

动态模型是一阶自回归：

$$\mathbf{x}_t - \bar{\mathbf{x}} = A(\mathbf{x}_{t-1} - \bar{\mathbf{x}}) + B\mathbf{w}_t,$$

其中 $\mathbf{w}_t$ 是独立同分布 $\mathcal{N}(0,1)$ 的独立向量，且 $\mathbf{x}_t = \begin{pmatrix} X_{t-1} \\ X_t \end{pmatrix}$。

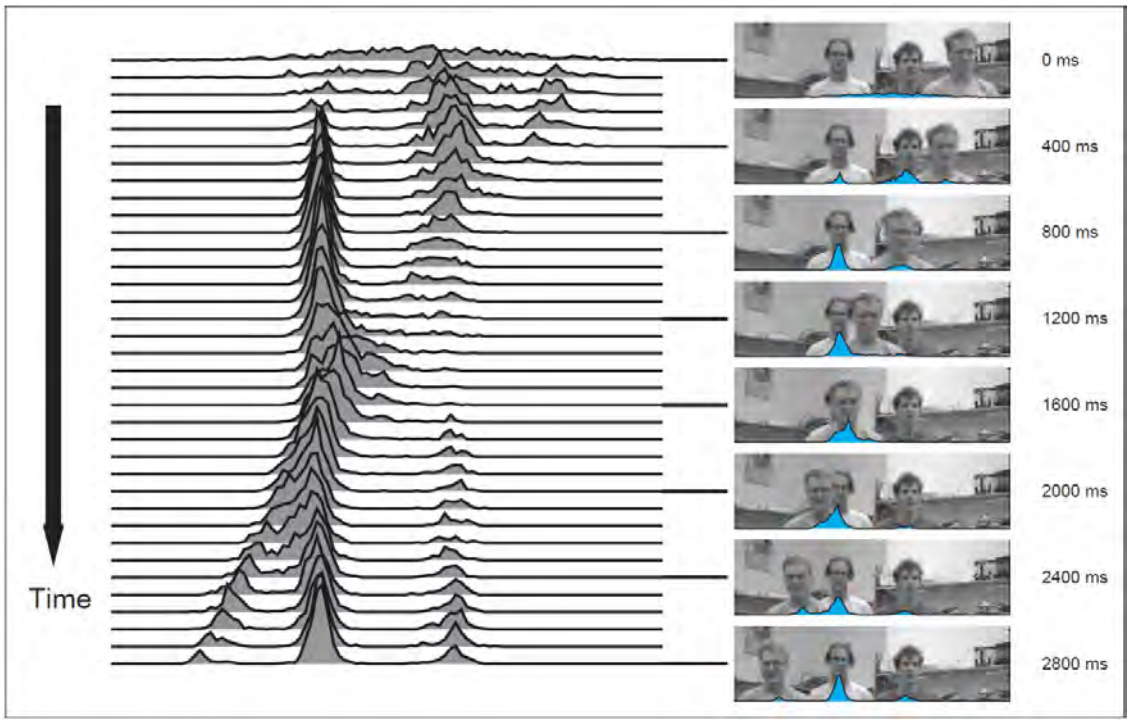


图 2.12: 来自 [4] 的视频多帧的状态密度的一维投影。

动态模型也可以表示为：

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) \propto \exp(-\frac{1}{2}\|(\mathbf{x}_t - \bar{\mathbf{x}}) - A(\mathbf{x}_{t-1} - \bar{\mathbf{x}})\|^2).$$

二维曲线的观测模型可以是：

$$p(\mathbf{z}|\mathbf{x}) \propto \exp\left(-\sum_{m=1}^{M} \frac{1}{2rM} f\left(\mathbf{z}_i(\frac{m}{M}) - \mathbf{r}(\frac{m}{M}); \mu\right)\right),$$

其中 r, μ 是常数, M 是曲线离散化的点数, $f(x;\mu) = \min(x^2, \mu^2)$, $\mathbf{z}_1(s)$ 是与 $\mathbf{r}(s)$ 最接近的图像特征:

$$\mathbf{z}_i(s) = \mathbf{z}(s') \text{ where } s' = \underset{s' \in g^{-1}(s)}{\operatorname{argmin}} |\mathbf{r}(s) - \mathbf{z}(s')|$$

使用该模型获得跟踪结果的一个例子如图 2.12所示。

2.3.3 SMC 框架总结

在顺序蒙特卡罗中，术语“顺序”有两个含义：

1. 将联合状态 $\mathbf{x} = (x_1, x_2, \cdots, x_d)$ 展开成组件，如 2.3.1节中的自避行走。
2. 像在 2.3.2节的粒子滤波器中那样随时间更新 X_t。

SMC/SIS 的设计中存在以下问题:

1. 试验概率的选择。例如，在粒子滤波中

$$p(\mathbf{x}_{t+1}|\mathcal{Z}_{t+1}) = \int_{\mathbf{x}_t} p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1})p(\mathbf{x}_{t+1}|\mathbf{x}_t)p(\mathbf{x}_t|\mathcal{Z}_t)d\mathbf{x}_t$$

我们可以利用以下方式产生粒子

a) 从 $p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1})$ 采样 (通过检测跟踪) 的数据驱动方法。当目标丢失时，这是很重要的。

b) 从 $p(\mathbf{x}_{t+1}|\mathbf{x}_t)p(\mathbf{x}_t|\mathcal{Z}_t)$ 采样并根据证据 $p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1})$ 重新加权的动态驱动方法。

更好的选择是同时使用数据驱动和动态驱动方法作为不同的通道生成粒子，这些通道可以根据每个时间步骤的数据质量共同完成。

2. 如何使样本恢复活力 – 巩固，丰富，重采样/重加权。

例 2.3 在自避行走中, 假设我们有一个长度为 n 的部分样本 $x^{(j)}$, 其中 n 足够大，试验概率为 $g_1(x^{(j)}) = \frac{1}{k_1}\frac{1}{k_2}\cdots\frac{1}{k_n}$，非常小。$w^{(j)} = k_1...k_n$ 参与最终求和，而最终的和非常大。一个想法是对 $x^{(j)}$ 产生 k 个副本, 每个副本具有 $\frac{1}{k}w^{(j)}$ 的权重。这等价于改变提议概率 $g(x)$ 使 $g(x^{(j)})$ 是原来的 k 倍。

例 2.4 类似地，在粒子滤波中，我们可以使用包含强样本的重复副本的等权重集 $\hat{S} = \{(\hat{x}^{(j)}, w^{(j)}), j = 1,...,m\}$ 重新采样加权样本集 $S = \{x^{(j)}, j = 1,...,m\}$，以便在下一步强样本产生更多子代。

在这两个示例中，通过此重采样方案，方法的性能显著提高。

重采样准则. 在 SMC 中，我们可以通过权重向量 $\mathbf{w} = (w^{(1)},\ldots,w^{(m)})$ 的方差或变异系数监控样本 $S = \{x^{(j)}, w^{(j)}), j = 1,...,m\}$。变异系数为

$$CV(\mathbf{w}) = \sqrt{\frac{\sum_{j=1}^{m}(w^{(j)} - \overline{w})^2}{(m-1)\overline{w}^2}}.$$

当 $CV(\mathbf{w})$ 太大时, $CV(\mathbf{w}) > c_0$, 重采样步骤是必要的。

重加权. 当重采样 $S = \{x^{(j)}, w^{(j)}), j = 1,...,m\}$ 时, 我们可能并不是必须使用权重向量 $\mathbf{w} = (w^{(1)},\ldots,w^{(m)})$ 按比例产生权重。相反，我们可以选择具有非零元素的任意向量 $\mathbf{a} = (a^{(1)},\ldots,a^{(m)}), a_i > 0$，并将样本重加权为 $w^{*(j)} = w^{(j)}/a^{(j)}$. $\mathbf{a}$ 的元素应设计为惩罚冗余样本并鼓励独特样本。

2.4 应用：通过 SMC 进行光线追踪

SMC 的另一个应用是光线追踪 [10], 它在给定作用于表面的光源描述下，计算表面的发光。

给定入射光函数 $L_i(\mathbf{x}, \omega_i)$, 在点 $\mathbf{x}$ 处反射光遵循反射方程：

$$L_r(\mathbf{x}, \omega_r) = \int_{S^2} f_r(\mathbf{x}, \omega_i \leftrightarrow \omega_r)L_i(x, \omega_i)|\cos(\theta_i)|d\sigma(\omega_i), \tag{2.13}$$

其中 f_r 是双向反射分布函数 (BRDF), S^2 是三维空间中的单位球, σ 是立体角测量, θ_i 是 ω_i 与 $\mathbf{x}$ 处的表面法线之间的角度。

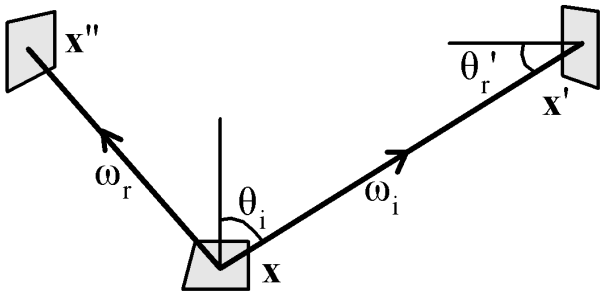


图 2.13: 反射方程的图示。©[1995] ACM。经许可重印，来自参考文献 [10].

如果我们只想在场景中使用点，我们也可以将反射方程写为：

$$L_r(\mathbf{x} \to \mathbf{x}") = \int_{\mathscr{M}} f_r(\mathbf{x}' \leftrightarrow \mathbf{x} \leftrightarrow \mathbf{x}'') L_i(\mathbf{x}' \to \mathbf{x}) G(\mathbf{x} \leftrightarrow \mathbf{x}') dA(\mathbf{x}'), \tag{2.14}$$

其中 $G(\mathbf{x} \leftrightarrow \mathbf{x}') = V(\mathbf{x} \leftrightarrow \mathbf{x}') \dfrac{\cos(\theta_r')\cos(\theta_i)}{\|\mathbf{x}-\mathbf{x}'\|2}$ 和 A 是表面积的度量, θ_r' 和 θ_i 是 $\mathbf{x} \leftrightarrow \mathbf{x}'$ 与 $\mathbf{x}$ 和 $\mathbf{x}'$ 的表面法线之间的角度，如图 2.13所示. 如果 $\mathbf{x}$ 和 $\mathbf{x}'$ 是相互可见的，则函数 $V(\mathbf{x} \leftrightarrow \mathbf{x}')$ 为 1，否则为 0 。

我们得到了找到平衡发光分布 L 的全局光照问题，该分布满足以下条件:

$$L(\mathbf{x} \to \mathbf{x}") = L_e(\mathbf{x} \to \mathbf{x}") + \int_{\mathscr{M}} f_r(\mathbf{x}' \leftrightarrow \mathbf{x} \leftrightarrow \mathbf{x}'') L(\mathbf{x}' \to \mathbf{x}) G(\mathbf{x} \leftrightarrow \mathbf{x}') dA(\mathbf{x}'),$$

其中发射的发光分布 L_e 是给定的。这是三点渲染方程 [5]。它可以简写为 $L = L_e + \mathcal{T}L$，其中 $\mathcal{T}$ 是光传输运算符。在弱假设下，解是 Neumann 级数：

$$L = \sum_{i=1}^{\infty} \mathcal{T}^i L_e.$$

2.4.1 示例：光泽高光

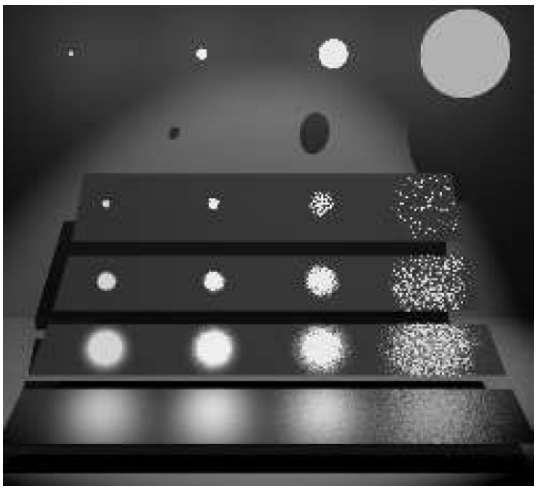
考虑光线追踪问题，即渲染由附近光泽表面上的区域光源 S 产生的高光，如图 2.14所示。有两种明显的策略可以使用蒙特卡罗方法来近似反射发光，即分别使用 eq. (2.13) 和 (2.14) 。对于这两种策略，我们使用重要性采样，其中样本 $x_1, ..., x_n$ 是从分布 $p(x)$ 获得的。积分近似为：

$$\int_{\Omega} f(x) d\mu(x) \approx \frac{1}{n} \sum_{i=1}^{n} \frac{f(x_i)}{p(x_i)}.$$

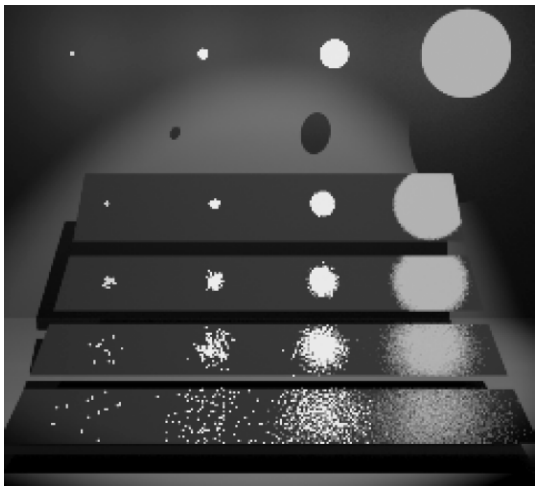
通过区域采样 我们随机选择表面上的点来近似 (2.14)。例如，可以相对于表面面积或发射功率在 S 上均匀地选择这些点。

通过定向采样 我们使用方向 ω_i 的随机样本来近似(2.13) 。通常选择 $p(\omega_i)d\sigma(\omega_i)$ 与 $f_r(\mathbf{x}\omega_i \leftrightarrow \omega_r)$ 或 $f_r(\mathbf{x}\omega_i \leftrightarrow \omega_r)|\cos(\theta_i)|$ 成比例。

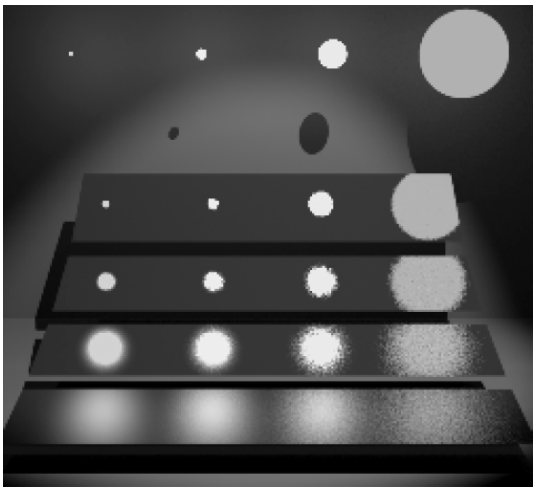
图 2.14 显示了使用不同采样方法得到渲染例子。场景包含四个不同半径和颜色的球形光源，以及顶上的聚光灯。所有球形光源发出相同的总功率。还有四个不同表面粗糙度的闪亮矩形板，这些矩形板



a) 区域抽样。



b) 定向采样。



c) 来自 (a) 和 (b) 的样本加权组合。

图 2.14: 从区域光源采样光泽高光。©[1995]ACM。经许可重印，来自参考文献 [10].

被倾斜以使得反射光源可见。给定一个观测光线照射到光滑的表面，图像 (a), (b),(c) 使用不同的技术进行高光计算。所有图像均为 500 x 450 像素。MC 技术是：

(a) 区域采样。每个像素 4 个样本，在每个光源的方向锥内均匀地（相对于立体角）选择样本方向 ω_i。

(b) 定向采样。每个像素 4 个样本，以与 BRDF $f_r(\mathbf{x}\omega_i \leftrightarrow \omega_r)d\sigma(\omega_i)$ 成比例的概率选择方向 ω_i。

(c) 使用 $\beta = 2$ 的幂启发式计算来自 (a) 和 (b) 的样本的加权组合。

光滑的 BRDF 是 Phong 模型的一个对称的、节能的变体。Phong 指数是 $n = 1/r - 1$, 其中 $r \in (0,1)$ 是表面粗糙度参数。光滑表面也具有小的扩散成分。

2.5 在重要性采样中保持样本多样性

为了简化表示，我们用 $p(\mathbf{x})$ 表示空间 Ω 中的一个任意分布。对于利用贝叶斯推理的图像分割问题，我们观察到 $p(\mathbf{x})$ 有两个重要的性质。

1. $p(\mathbf{x})$ 具有极多数量的局部最大值（统计学中称为众数）。一个重要众数对应于图像的不同解释，并且众数周围的云包含区域边界或模型参数的局部小扰动。这些 $p(\mathbf{x})$ 重要的众数 $\mathbf{x}_i, i = 1,2,...$，由于高维度而彼此很好地分离。

2. 每个众数 $\mathbf{x}_i$ 具有权重 $\omega_i = p(\mathbf{x}_i)$, 其能量被定义为 $E(\mathbf{x}_i) = -\log p(\mathbf{x}_i)$。这些众数的能量均匀分布在一个宽阔的范围 $[E_{\min}, E_{\max}]$, 如 $[1000, 10,000]$。例如，通常具有其能量差为 500 阶或更大的解（或局部最大值）。因此，它们的概率（权重）在 $\exp^{-500}$ 阶不同, 我们的感知对那些“平凡的”局部众数有兴趣！

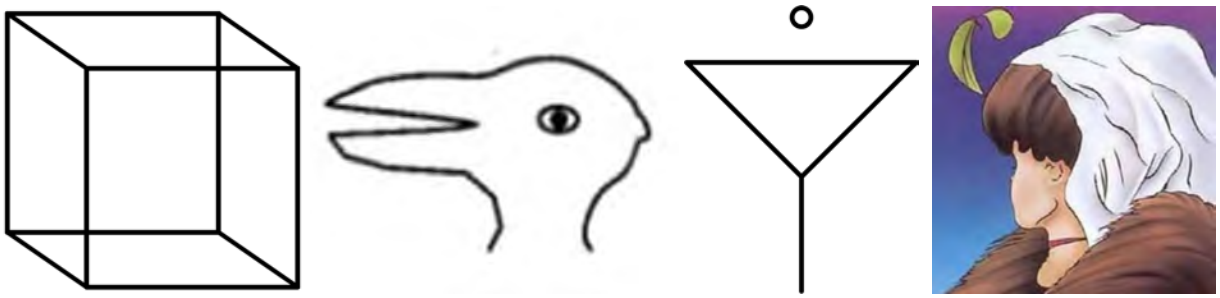


图 2.15: 不同的图像有多种解释。从左到右：内克尔立方体, 鸭子/兔子错觉 [6], 比基尼/马提尼模糊，老妇人与年轻女子。

保持样本多样性是维持一个概率分布众数的重要问题。维持概率众数对保持如图 2.15中图像解释的模糊性很重要。

直观上，将 Ω 中的 $p(\mathbf{x})$ 想象为像宇宙的质量一样分布是有帮助的。每颗星是质量密度的局部最大值。重要的和发达的星彼此很好地分开，它们的质量可以在许多数量级上不同。这个比喻将我们引向 $p(\mathbf{x})$ 的高斯混合表示。对于足够大的 N ，我们有，

$$p(\mathbf{x}) = \frac{1}{\omega}\sum_{j=1}^{N} \omega_j G(\mathbf{x} - \mathbf{x}_j, \sigma_j^2), \quad \omega = \sum_{j=1}^{N} \omega_j.$$

我们用

$$S_o = \{(\omega_j, \mathbf{x}_j),\ j = 1,2,...,N\},$$

表示加权粒子（或众数）的集合。因此，我们的任务是从 S_o 中选择一组 $K << N$ 粒子 S。我们定义 S 的索引到 S_o 的索引的映射为，

$$\tau : \{1,2,...,K\} \longrightarrow \{1,2,....,N\}.$$

因此，

$$S = \{(\omega_{\tau(i)}, \mathbf{x}_{\tau(i)});\ i = 1,2,...,K\}$$

通过

$$\hat{p}(\mathbf{x}) = \frac{1}{\alpha}\sum_{i=1}^{K}\omega_{\tau(i)}G(\mathbf{x}-\mathbf{x}_{\tau(i)},\sigma^2_{\tau(i)}), \quad \alpha = \sum_{i=1}^{K}\omega_{\tau(i)}.$$

表达了一个近似 $p(\mathbf{x})$ 的非参数模型。我们的目标是计算

$$S^* = \arg\min_{|S|=K} D(p||\hat{p}).$$

为了标记简洁，我们假设所有的高斯函数在近似 $p(\mathbf{x})$ 时具有相同的方差, 即 $\sigma_j = \sigma, j = 1,2,...,N.$ 按照我们的比喻，所有“星星”具有相同的体积，但重量不同。利用 $p(\mathbf{x})$ 的两个性质, 我们可以按照以下方式近似计算 $D(p||\hat{p})$ 。我们从观察高斯分布的 KL-散度开始。

设 $p_1(x) = G(\mathbf{x}-\mu_1;\sigma^2)$ 和 $p_2(x) = G(\mathbf{x}-\mu_2;\sigma^2)$ 是两个高斯分布, 我们很容易得到

$$D(p_1||p_2) = \frac{(\mu_1-\mu_2)^2}{2\sigma^2}.$$

我们将解空间 Ω 划分为不相交的域

$$\Omega = \cup_{i=1}^{N} D_i, \quad D_i \cap D_j = \emptyset, \forall i \neq j.$$

D_i 是一个域，在这个域中 $p(\mathbf{x})$ 由粒子 $(\omega_i,\mathbf{x}_i), i \in \{1,...,N\}$ 决定. 这种划分的原因是 S 中的粒子在高维空间中彼此远离，并且它们的能量基于本节开头描述的两个性质而显著变化。在每个域内，可以合理地假设 $p(\mathbf{x})$ 由混合物中的一个项支配，而其他 $N-1$ 项是可以忽略的。

$$p(\mathbf{x}) \approx \frac{\omega_i}{\omega}G(\mathbf{x}-\mathbf{x}_i;\sigma^2), \quad \mathbf{x} \in D_i,\ i = 1,2,...,N.$$

D_i 的大小远大于 σ^2。在空间中移除 $N-K$ 个粒子后，它由 S 中选择的附近粒子主导。

我们定义第二个映射函数

$$c : \{1,2,...,N\} \to \{1,2,...,K\},$$

使 D_i 中的 $\hat{p}(\mathbf{x})$ 由粒子 $\mathbf{x}_{\tau(c(i))} \in S_K$ 主导, 且

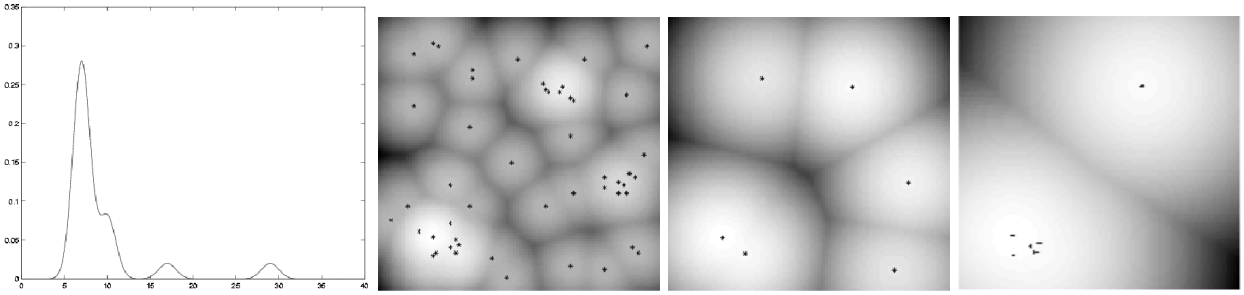
$$\hat{p}(\mathbf{x}) \approx \frac{\omega_{c(i)}}{\alpha}G(\mathbf{x}-\mathbf{x}_{\tau(c(i))};\sigma^2), \quad \mathbf{x} \in D_i, i = 1,2,...,N.$$

直观地, N 个区域被划分为 K 个组，每个组由 S_K 中的一个粒子支配。因此我们可以近似 $D(p||\hat{p})$，

$$\begin{aligned}
D(p||\hat{p}) &= \sum_{n=1}^{N}\int_{D_n} p(\mathbf{x})\log\frac{p(\mathbf{x})}{\hat{p}(\mathbf{x})}d\mathbf{x} \\
&= \sum_{n=1}^{N}\int_{D_n}\frac{1}{\omega}\sum_{i=1}^{N}\omega_i G(\mathbf{x}-\mathbf{x}_i;\sigma^2)\log\frac{\frac{1}{\omega}\sum_{i=1}^{N}\omega_i G(\mathbf{x}-\mathbf{x}_i;\sigma^2)}{\frac{1}{\alpha}\sum_{j=1}^{k}\omega_{\tau(j)}G(\mathbf{x}-\mu_{\tau(j)};\sigma^2)}d\mathbf{x} \\
&\approx \sum_{n=1}^{N}\int_{D_n}\frac{\omega_n}{\omega}G(\mathbf{x}-\mathbf{x}_n;\sigma^2)[\log\frac{\alpha}{\omega}+\log\frac{\omega_n G(\mathbf{x}-\mathbf{x}_n;\sigma^2)}{\omega_{\tau(c(n))}G(\mathbf{x}-\mathbf{x}_{\tau(c(n))};\sigma^2)}]d\mathbf{x}
\end{aligned}$$

$$= \sum_{n=1}^{N} \frac{\omega_n}{\omega}[\log\frac{\alpha}{\omega} + \log\frac{\omega_n}{\omega_{\tau(c(n))}} + \frac{(\mathbf{x}_n - \mathbf{x}_{\tau(c(n))})^2}{2\sigma^2}] \tag{2.15}$$

$$= \log\frac{\alpha}{\omega} + \sum_{n=1}^{N} \frac{\omega_n}{\omega}[(E(\mathbf{x}_{\tau(c(n))}) - E(\mathbf{x}_n)) + \frac{(\mathbf{x}_n - \mathbf{x}_{\tau(c(n))})^2}{2\sigma^2}] = \hat{D}(p||\hat{p}).$$



(a) (b) (c) (d)

图 2.16: (a) 具有四个粒子 $\mathbf{x}_i, i = 1,2,3,4$ 的一维分布 $p(\mathbf{x})$ 。(b) 具有 50 个粒子的二维分布 $p(\mathbf{x})$, 我们在图像中显示 $\log p(\mathbf{x})$ 用以可视化。(c) 具有 6 个粒子的 $\hat{p}_1(x)$，$D(p||\hat{p})$ 无变化。(d) 具有 6 个粒子的 $\hat{p}_2(x)$，这 6 个粒子使 $|p-\hat{p}|$ 最小化。©[2002]IEEE。经许可重印，来自参考文献 [9]。

公式 (2.15) 具有直观的含义。第二项表明每个选定的 $\mathbf{x}_{\tau(c(i))}$ 应具有较大的权重 $\omega_{\tau(c(i))}$. 第三项包含从 S_o 中的粒子到 S 中粒子的吸引力。因此，该项有助于在 S_k 中拉开粒子，并且还起到鼓励选择具有较大权重粒子的作用，如第二项一样。为了证明 $\hat{D}(p||\hat{p})$ 对 $D(p||\hat{p})$ 近似的好处, 我们给出了两个实验。

图 2.16.(a) 显示了一个一维分布 $p(\mathbf{x})$，它是 $N = 4$ 高斯（粒子）的混合模型。我们索引从左到右的中心 $\mathbf{x}_1 < \mathbf{x}_2 < \mathbf{x}_3 < \mathbf{x}_4$。假设我们想为 S_K 和 $\hat{p}(\mathbf{x})$ 选择 $K = 3$ 个粒子。

S_3:	$\{x_1,x_2,x_3\}$	$\{x_1,x_2,x_4\}$	$\{x_1,x_3,x_4\}$	$\{x_2,x_3,x_4\}$
$D(p\|\|\hat{p})$:	3.5487	1.1029	0.5373	2.9430
$\hat{D}(p\|\|\hat{p})$:	3.5487	1.1044	0.4263	2.8230
$\|p-\hat{p}\|$:	0.1000	0.1000	0.3500	1.2482

表 2.1: 左边图 2.16中一维分布的不同粒子集 S_3 的 $p(x)$ 和 $\hat{p}(x)$ 之间的距离。

表 2.1显示了四种可能的组合中 $p(\mathbf{x})$ 和 $\hat{p}(\mathbf{x})$ 之间的距离。第二行是 KL 散度 $D(p||\hat{p})$ ，第三行是估计值 $\hat{D}(p||\hat{p})$。如果粒子分离良好，近似是非常准确的。

两种方法都选择 (x_1,x_3,x_4) 作为最佳 S. 粒子 x_2 虽然比 x_3 和 x_4 具有更高的重量，但是由于它接近 x_1，所以它没有受到 KL 散度的支持。第四行显示了 $p(x)$ 和 $\hat{p}(x)$ 之间差的绝对值。这个距离支持 (x_1,x_2,x_3) 和 (x_1,x_2,x_4)。相比之下，KL 散度支持彼此分离的粒子，并在尾部获得显著的峰值。

这个想法在图 2.16中得到了更好的证明。图 2.16.(b) 显示了 $\log p(\mathbf{x}) = -E(\mathbf{x})$ ，为了显示，它被重新进行了归一化。$p(\mathbf{x})$ 由 $N = 50$ 个粒子组成，其中心由黑点表示。能量 $E(\mathbf{x}_i), i = 1,2,...,N$ 均匀地分布在区间 $[0,100]$ 中。因此它们的权重具有指数阶不同。图 2.16(c) 展示了 $\log\hat{p}(\mathbf{x})$，其中 $k = 6$ 个粒子使 $D(p||\hat{p})$ 和 $\hat{D}(p||\hat{p})$ 最小化。图 2.16(d) 显示了最小化绝对值 $|p-\hat{p}|$ 的 6 个粒子。很明显，图 2.16.(c) 具有更多分散的颗粒。

2.5.1 Parzen 窗讨论

利用来自 SMC 的 n 个样本估计密度 $p(\mathbf{x})$ 的一个类似方法是 Parzen 窗。对于这种方法，我们假设 $\mathbf{x}$ 周围区域的分布具有特定的形式。例如，假设与核函数相似的“窗口函数” $\phi(\mathbf{x})$ 是 d 维超立方体。此

函数是一个指示器，如果样本位于以原点为中心的单位超立方体内，则返回值 1，否则返回 0。则落入 $\mathbf{x}_i$ 周围边长为 ℓ 的立方体中的样本数 S_n 为

$$S_n = \sum_{i=1}^{n} \phi(\frac{\mathbf{x}-\mathbf{x}_i}{\ell}).$$

为了把这个计数转换成 $\mathbf{x}$ 处分布的估计，我们简单地取对窗口体积 V 和样本数量的均值，

$$p_n(\mathbf{x}) = \frac{1}{nV} S_n.$$

这一策略估计 $\mathbf{x}$ 处的分布, 它允许其他样本的影响随与 $\mathbf{x}$ 的接近而增加。

这是 Parzen 窗的一般框架，实际上我们可以选择任何积分为 1 的非负窗口函数 $\phi(\mathbf{x})$。此外，我们可以为不同的样本点选择不同的函数。这意味着可以将分布估计为窗口函数的线性组合。根据所选窗口的大小，分布将基本上成为焦点。

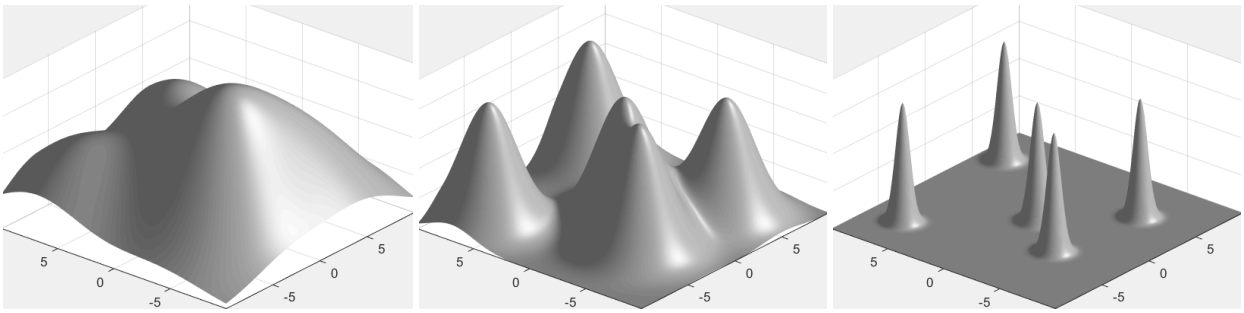


图 2.17: 使用 3 种不同窗口大小的 Parzen 窗口估计。窗口大小从左向右减小。

图 2.17显示了使用不同大小的高斯窗函数的相同数据的一系列图示。我们看到，窗口越大，概率函数越平滑。对于小窗口，景观呈现为一组分离的尖峰。

要理解这一点，考虑在山地景观中使用相机进行近景拍摄。在远景拍摄时，对应于较小的 ℓ，山是完全可见的，但与图像其余部分的其他特征（湖泊，田地等）相比，呈现为小而尖锐的峰。另一方面，当相机在山上放大进行近景拍摄时，它们将占据整个画框，并且可以看到整个范围中越来越小的部分。峰之间的山脊的具体细节可能成为焦点，但整体景观更难理解。有效地使用这种技术需要在这两种想法之间找到平衡点。

为了将这种想法与 SMC 结合起来，我们回想我们的目标概率 $p(\mathbf{x})$ 被分布 $g(\mathbf{x})$ 近似。样本从 g 中收集，且每个样本根据 $\omega(\mathbf{x}_i) = \frac{p(\mathbf{x}_i)}{g(\mathbf{x}_i)}$ 分配权重。对于每一个样本 $\mathbf{x}_i$, 我们有一个特定的窗函数，比如具有均值 $\mathbf{x}_i$ 和方差 v_i 的正态分布。则 $p(\mathbf{x})$ 的 Parzen 窗估计由下式给出

$$p_n(\mathbf{x}) = \sum_{i=1}^{n} w(\mathbf{x}_i) N(\mathbf{x}-\mathbf{x}_i, v_i).$$

上面的一般 Parzen 窗估计 $\bar{p}_n$ 的均值可以通过以下公式计算

$$\bar{p}_n(\mathbf{x}) = E[p_n(\mathbf{x})] = \frac{1}{n} E[\frac{1}{V} S_n] = \frac{1}{n} E[\sum_{i=1}^{n} \frac{1}{V} \phi(\frac{\mathbf{x}-\mathbf{x}_i}{\ell})] \xrightarrow{n\to\infty} \int \frac{1}{V} \phi(\frac{\mathbf{x}-\mathbf{z}}{\ell}) p(\mathbf{z}) d\mathbf{z} \tag{2.16}$$

正如前面所讨论而现在数学上表明，平均值为目标分布与窗口函数的卷积。从渐近的角度看，如果要求 $\lim_{n\to\infty} V = 0$，则均值将接近真值。对于方差，因为 $p(\mathbf{x})$ 是独立变量的和，所以我们简单地对每个变量的方差求和并推导得出

$$\begin{aligned} n\sigma_n^2(\mathbf{x}) &= n\sum_{i=1}^{n} E[(\frac{1}{nV}\phi(\frac{\mathbf{x}-\mathbf{x}_i}{\ell}) - \frac{1}{n}\bar{p}_n(\mathbf{x}))^2] \\ &= n\sum_{i=1}^{n} E[\frac{1}{n^2V^2}\phi^2(\frac{\mathbf{x}-\mathbf{x}_i}{\ell})] - \bar{p}_n^2(\mathbf{x}) \\ &\xrightarrow{n\to\infty} \frac{1}{V}\int \frac{1}{V}\phi^2(\frac{\mathbf{x}-\mathbf{z}}{\ell})p(\mathbf{z})d\mathbf{z} - \left(\int \frac{1}{V}\phi(\frac{\mathbf{x}-\mathbf{z}}{\ell})p(\mathbf{z})d\mathbf{z}\right)^2 \end{aligned} \tag{2.17}$$

这一结果与先前提出的山的类比一致。方差与窗口 V 的体积高度相关。对于较大体积或等价较大的 ℓ，每个点的窗函数都是平滑的，并且结果分布的方差是减少的。

2.6 蒙特卡罗树搜索

蒙特卡罗树搜索是马尔可夫决策过程中的一种随机决策程式，它通过预先搜索多个情况并利用它们对最有希望的即时行为积累支撑。

马尔可夫决策过程 (MDP) 用于强化学习，其中智能体在环境中执行动作，并且基于所执行的动作不时地得到奖励。在具有当前状态 s 的 MDP 中, 智能体执行动作 a，环境达到新状态 s'，其仅取决于当前状态 s 和执行的动作 a。同时，智能体得到奖励 $R(s')$。设 Ω 表示所有状态的空间，A 表示可能的动作的空间，当系统处于状态 $s \in \Omega$ 时智能体执行动作 $a \in A$，然后系统到达新状态 s'，该状态是来自分布 $p(s'|a,s)$ 的一个样本。

MDP 的例子诸如双陆棋、国际象棋和围棋之类的游戏，或者探索环境的机器人。极点平衡是另一个例子，其中每次极点下降时奖励为负，否则为零。

我们的目标是学习在每个状态采取什么动作以便使期望的奖励最大化。为此，人们希望学习一种策略 $\pi : \Omega \to A$，其中 $\pi(s)$ 表示对一个状态 s 为使期望奖励最大化而采取的最佳动作。在某些情况下使用非确定性策略 $\pi : A \times \Omega \to \mathbb{R}$，其中 $\pi(a|s)$ 表示在状态 s 中采取行为 a 的概率。

基于当前策略 π 从每个状态 s 开始的期望奖励由*状态-价值函数* $v_\pi : \Omega \to \mathbb{R}$ 表示。我们还可以考虑*动作-价值函数* $q_\pi : \Omega \times A \to \mathbb{R}$，其中 $q(s,a)$ 表示在状态 s 中采取动作 a 时总的期望奖励。

在动作状态空间 $\Omega \times A$ 是有限的且不是很大的特殊情况下，可以在智能体探索环境时估计状态-价值和动作-价值函数，并通过一些动态规划类型的算法，可以学习到越来越好的策略。然而，在大多数应用中，状态空间太大以至于实际上不可能记住价值函数 $v_\pi(s)$ 或动作价值函数 $q_\pi(s,a)$。在这些情况下，可以使用近似方法。

蒙特卡罗树搜索 (MCTS) 就是这样一种近似方法，它通过探索不同的动作、并基于同时获得的奖励获得对每个动作的支持，从而从当前状态 s_t 开始估计即时动作-价值函数 $q_\pi(s_t,a)$。请注意，对于所探索的每个状态 s_t 都运行单独的 MCTS，如图 2.18所示。

蒙特卡罗树搜索的不同变体已成功用于许多应用，包括特征选择 [3]，其中 MCTS 在 NIPS 2003 特征选择挑战的许多数据集上获得了当时最好的结果。另一个应用是解决量化约束满足问题（QCSP），其中 [1] 中描述的改进 MCTS 在性能表现上超过了大规模问题的现有最好的 $\alpha - \beta$ 搜索算法。

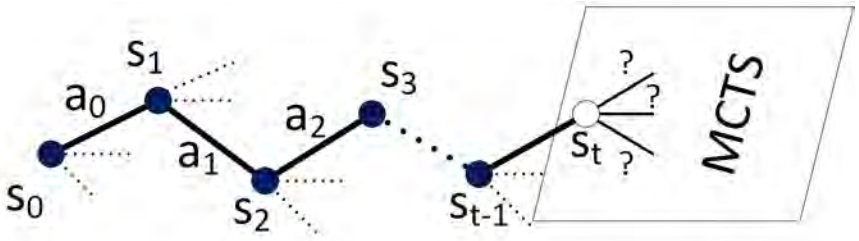


图 2.18: 蒙特卡罗树搜索通过估计每个可能的即时动作的期望奖励来决定在当前状态 s_t 中要采取的最佳动作。

2.6.1 纯蒙特卡罗树搜索

在当前状态 s_t，蒙特卡罗树搜索（MCTS）程式用于建立对最有希望的动作的支持，并决定可能采取的最佳动作。为此，树以当前状态为根进行生长，以最优的动作为直接分支。树是迭代构建的，在每个 MCTS 迭代中，一个叶子被添加到树中，并且树策略被更新。

树策略用于引导树到达叶节点。树策略平衡了对新分支的探索和对已存在的树分支的利用。可以使用的可能的树策略有很多，但有一种流行的策略是树的上限置信算法（UCT），将在本节末尾对其进行介绍。从叶节点开始，一种默认策略用于树搜索直到一段搜索结束（例如，当达到一个胜利/失败状态时）。

每个 MCTS 迭代过程可以分解成四个部分，如图 2.19所示。

a) 选择：在这个阶段，树策略基于学习的获胜机会随机采样最有希望的状态，直到到达具有未访问的子节点的节点 L。

b) 扩张：除非搜索在节点 L 处结束，否则用一个或多个子节点（基于可能的动作）扩张节点，并且随机选择一个子节点 C。

c) 模拟：使用从状态 C 开始的默认策略来玩搜索游戏，直到达到结果（例如游戏结束）。

d) 反向传播：结果用于更新从叶节点 C 到根的路径上的获胜计数（因此也即树策略）。

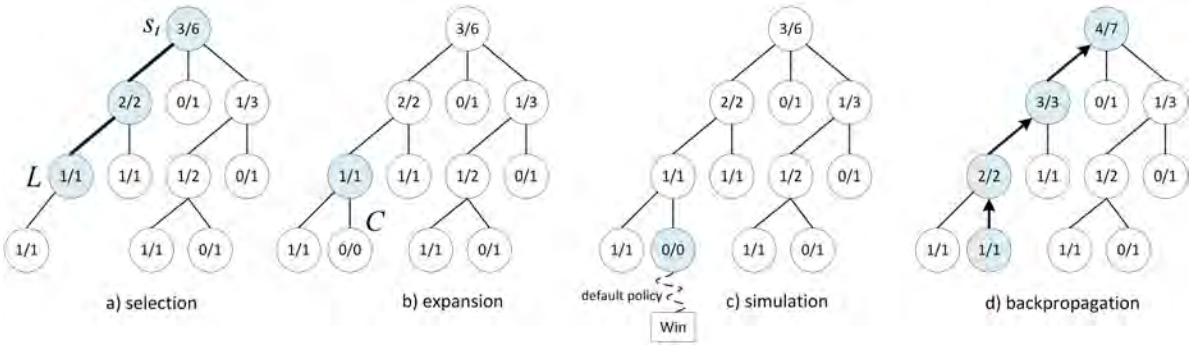


图 2.19: 蒙特卡罗树搜索迭代的四个部分的说明。

这四个步骤在 MCTS 的每次迭代中执行，并且迭代次数取决于计算预算。当达到计算预算时，搜索终止，并且使用最新的树策略来决定在根节点 s_t 处采取的最佳动作。在采取该动作之后，新状态为 s_{t+1}，并且以 s_{t+1} 作为根节点再次执行 MCTS。用 s_t 作为根节点生长的树可以被丢弃，或者更好的是，具有根 s_{t+1} 的子树可以被重新用于新的 MCTS。

观察到在每次 MCTS 迭代中，至少有一个节点被添加到树中，并且根节点处的总计数递增 1。通过这种方式，在多次迭代之后，根节点的子节点处的总计数很高，因此在估计这些子节点中的每一个的期望奖励时将获得越来越好的准确性。

对于每个树节点 v，两个值需要保持：节点已被访问的次数 $N(v)$ 和经过节点的播出的总奖励 $Q(v)$。然后，比率 $Q(v)/N(v)$ 是经过节点 v 播出的期望奖励的近似值。

$N(v)$ 和 $Q(v)$ 的值用于定义树策略，他们因此随着树的增长而变化。最受欢迎的树策略之一是*树的上限置信算法*（UCT）。它旨在平衡对未访问子节点的探索与对已访问子节点的利用（再访）。

根据 UCT，对于每个树节点 v，选择子节点 j 以最大化：

$$UCT(j) = \frac{Q(j)}{N(j)} + c\sqrt{\frac{2\ln N(v)}{N(j)}} \tag{2.18}$$

其中 $c > 0$ 是常数。注意，如果 $N(j) = 0$，则 $UCT(j) = \infty$，因此在进一步探索已访问的子节点之前，必须访问所有未访问的子节点。因此，UCT 是一种广度优先搜索策略。

当达到计算预算时，可根据 [7] 中描述的下列标准之一来选择要采取的最佳行为：

1. 最大的子节点。选择最高估计奖励的子节点。
2. 健壮的子节点。选择访问量最大的子节点。
3. 最大-健壮的子节点。选择同时访问最多和最高奖励的子节点。如果没有这样的子节点存在，MCTS 将继续进行，直到达到最高奖励的子节点的最低访问次数。
4. 安全的子节点。选择最大化置信下限的子节点。

MCTS 有许多变体，包括许多树策略、学习策略等。MCTS 方法和应用的全面综述可以在 [2] 中看到。

2.6.2 AlphaGo

AlphaGo [8] 是对 MCTS 的修改，以适用于玩围棋游戏。Go 是一种比国际象棋更具挑战性的游戏，因为所有可能的游戏空间大小具有 $250^{150} \approx 2^{1200}$（每种配置约 250 个可能的走法，总游戏长度约为 150）的数量级，而国际象棋的空间大小为 $35^{80} \approx 2^{410}$ 的数量级。搜索空间的庞大规模使的穷尽搜索最佳走法不可行。除了较大的搜索空间，一个可能更大的挑战是难以找到可以评估任何位置获胜机会的良好状态值函数。

由于纯 MCTS 的广度优先搜索特性以及从每个位置大量的可能走法，直接应用 MCTS 是不合适的。由于这些原因，作者采用了基于学习的策略来减小游戏的广度（从当前配置可能的走法的空间）和每个走法的评价深度。

通过使用树策略来减少游戏的广度，该策略在给定当前配置的情况下估计下一个棋子的最有希望的位置 a。通过利用一个学习到的价值函数 $V(s)$，它评估从状态 s 的获胜机会，每个可能的走法的评估深度也降低了。

为了描述策略和价值函数，我们首先需要描述在 AlphaGo MCTS 策略和价值函数中使用的三个学习策略网络 $p_\sigma(a|s), p_\rho(a|s), p_\pi(a|s)$ 和一个价值网络 $v_\theta(s)$。

- 首先，从 160,000 个游戏和大约 3000 万个棋盘走法中，策略网络 $p_\sigma(a|s)$ 以有监督方式学习得到。基于从当前配置 s 中提取的大量特征，该模型建立了一个 13 层 CNN（卷积神经网络）。
- 通过增加更多的训练数据，策略网络被进一步改进。这些训练数据是在当前策略网络与旧的策略网络自我博弈下产生的。这样就得到了一个改进的策略网络 $p_\rho(a|s)$ 。
- 默认（快速走子）策略 $p_\pi(a|s)$ 也被训练为扩展特征集上的线性模型。快速走子策略比策略网络快约 1000 倍，并用于蒙特卡罗树搜索的模拟步骤。
- 价值网络 $v_\theta(s)$ 也训练为 CNN，使用与策略网络 $p_\sigma(a|s)$ 相同的特征再加上一个表示当前玩家颜色的特征。为了避免过拟合，训练数据包括从单独的游戏中自玩而获得的 3000 万个配置。使用快速走子策略 $p_\pi(a|s)$，价值网络 $v_\theta(s)$ 获得比蒙特卡罗展开更好的位置评价准确度，并且与使用策略网络 $p_\rho(a|s)$ 的蒙特卡罗展开相当，但快 15,000 倍。

对于 MCTS，树的每个边 (s,a) 都存储了 MCTS 模拟上的访问计数 $N(s,a)$、动作价值 $Q(s,a)$ 和先验概率 $P(s,a)=p_\sigma(a|s)$。然后，MCTS 以如下步骤进行：

a) 选择：游戏通过选择配置 s 处的动作

$$a_t = \underset{a}{\operatorname{argmax}}[Q(s,a)+u(s,a)] \tag{2.19}$$

来玩，直到到达叶子节点 s_L，其中 $u(s,a)\propto P(s,a)/(1+N(s,a))$。

b) 扩张：除非游戏在叶子节点 s_L 处结束，否则叶子节点通过合理的走子 a 扩张，并计算、存储先验概率 $P(s_L,a)=p_\sigma(a|s_L)$。

c) 模拟：使用从状态 s_L 开始直到游戏结束的快速策略 $p_\pi(a|s)$，游戏通过蒙特卡罗展开来玩，获得输出结果 z_L。s_L 的值计算如下：

$$V(s_L)=(1-\lambda)v_\theta(s_L)+\lambda z_L \tag{2.20}$$

d) 反向传播：游戏结果 z_L 用于更新从叶片 s_L 到根部路径上的访问计数 $N(s,a)$ 和动作价值 $Q(s,a)$，如下式(2.21)描述。

在 n 次 MCTS 模拟之后，访问次数和动作价值为：

$$\begin{aligned} N(s,a) &= \sum_{i=1}^{n}\mathbf{1}(s,a,i), \\ Q(s,a) &= \frac{1}{N(s,a)}\sum_{i=1}^{n}\mathbf{1}(s,a,i)V(s_L^i) \end{aligned} \tag{2.21}$$

其中 s_L^i 是在模拟 i 中到达的叶子节点，$\mathbf{1}(s,a,i)$ 是二进制指示器，指示是否在模拟 i 中遍历了边 (s,a)。

我们注意到，与标准的 MCTS 不同，叶子的值 $V(S_L)$ 不完全由蒙特卡罗展开确定，而是展开结果和价值网络 $v_\theta(s_L)$ 预测的混合。比较 AlphaGo 仅使用价值网络（没有展开）或仅使用展开的性能，结果表明展开的性能优于价值网络，但是来自 Eq. (2.20)且 $\lambda=0.5$ 的组合比两者都好得多。

2015 年 10 月，AlphaGo 以 5-0 战胜欧洲冠军二段选手樊麾，并于 2016 年 3 月以 9-1 战胜九段职业选手李世石。然后在 2017 年，AlphaGo 以 3-0 战胜了世界排名第一的围棋选手柯洁，并被中国围棋协会授予专业九段。

练习

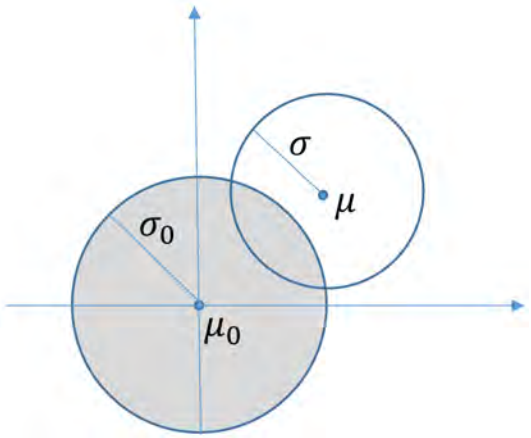


图 2.20: 问题 1 的图。

问题 1。2D 平面中的重要性采样和有效样本数。假设目标分布 $\pi(x,y)$ 是对称高斯分布，均值 $\mu=(2,2)$，标准差 $\sigma=1$。假设我们使用近似分布 $g(x,y)$ 作为试验密度，它是一个高斯分布，均值 $\mu_0=(0,0)$，标准偏差 σ_0。则

$$\pi(x,y)=\frac{1}{2\pi}e^{-1/2[(x-2)^2+(y-2)^2]},\quad g(x,y)=\frac{1}{2\pi\sigma_o}e^{-1/(2\sigma_o^2)[x^2+y^2]}$$

我们估计变量 $\theta=\int\sqrt{y^2+x^2}\pi(x,y)dxdy$。我们比较了三种重要性采样参考概率的有效性。

- 步骤 1, 计算 $\hat{\theta}_1$: 通过直接从 $\pi(x,y)$ 采样 n_1 个样本来估计 θ。由于这两个维度是独立的，因此可以从一维边缘高斯中对 x 和 y 进行采样。
- 步骤 2, 计算 $\hat{\theta}_2$: 通过从 gxy 采样 n_2 个样本来估计 θ，其中 $\sigma_0=1$。
- 步骤 3, 计算 $\hat{\theta}_3$: 通过从 gxy 采样 n_3 个样本来估计 θ，其中 $\sigma_0=4$。

i) 在一个图中，相对于 n 绘制 $\hat{\theta}_1,\hat{\theta}_2,\hat{\theta}_3$（增加 n 以使它们收敛）以比较收敛速率。在运行实验之前，请尝试猜测步骤 3 是否比步骤 2 更有效。[可以在几个点 $n=10,100,1000,10000,$ 使用对数图] ii) 估算“有效样本量”的值。我们建议式(2.6)中的估计器，

$$ess(n)=\frac{n}{1+\mathrm{var}_g[\omega]}$$

但我们不确定它有多好。由于步骤 1 中的样本都是直接从目标分布中抽取的“有效”样本，我们使用 $ess^*(n_1)=n_1$ 作为真实值，并比较步骤 2 和步骤 3 的有效样本大小，即真实 $ess^*(n_2)$ 和 $ess^*(n_3)$ 是估计

误差达到与步骤 1 的水平相同时的数目。绘制相对于 $ess^*(n_2)$ 的 $ess(n_2)$ 和相对于 $ess^*(n_3)$ 的 $ess(n_3)$，并讨论结果。

问题 2。估计 $(n+1)\times(n+1)$ 网格中自避行走的数量。假设我们总是从位置 $(0,0)$ 开始, 即左下角。我们为变化长度为 N 的 SAW $r=(r_1,r_2,,r_N)$ 设计了一个试验（参考）概率 $p(r)$。然后我们从 $p(r)$ 中采样了多个 M SAW，并且估计计算如下。本章举例说明一些结果。

在每一步，试验概率 $p(r)$ 可以选择停止 (中断路径) 或向左/右/上/下，只要它不与自己相交。每个选项都与（读者的设计）概率相关联，并且这些概率在每一点上和是 1。

1) 对于 $n=10$ [尝试 $M=107$ 到 108]，SAW 的总数 K 是多少？为了澄清：一个正方形是 2×2 的网格且 $n=1$。绘制 K 对 M（在对数 - 对数图中）的图示并监视顺序重要性采样（SIS）过程是否已收敛。尝试比较 $p(r)$ 的至少 3 种不同设计，看看哪种设计更有效。例如，只要正确计算 $p(r)$ ，就可以多次从之前找到的路径开始。

2) 从 $(0,0)$ 开始到 (n,n) 结束的 SAW 总数是多少？在这里，读者仍可以使用与上面相同的采样程式，但只记录成功到达 (n,n) 的 SAW。这个数字的真值是我们讨论过的：1.5687×10^{24}。

3) 对于 1）和 2）中的每个实验，在直方图中绘制 SAW 的长度 N 的分布（想一想：在计算直方图时是否需要对 SAW 进行加权?）并可视化（打印）你找到的最长 SAW。

参考文献

[1] Yongjoon Joe Baba Satomi, Atsushi Iwasaki, and Makoto Yokoo. Real-time solving of quantified csps based on monte-carlo game tree search. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence*, 2011.

[2] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1):1–43, 2012.

[3] Romaric Gaudel and Michele Sebag. Feature selection as a one-player game. In *International Conference on Machine Learning*, pages 359–366, 2010.

[4] Michael Isard and Andrew Blake. Condensation: conditional density propagation for visual tracking. *International journal of computer vision*, 29(1):5–28, 1998.

[5] James T Kajiya. The rendering equation. In *ACM Siggraph Computer Graphics*, volume 20, pages 143–150. ACM, 1986.

[6] Jacob Porway and Song-Chun Zhu. C^ 4: Exploring multiple solutions in graphical models by cluster sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1713–1727, 2011.

[7] Frederik Christiaan Schadd. Monte-carlo search techniques in the modern board game thurn and taxis. *M. sc, Maastricht University, Netherlands*, 2009.

[8] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[9] Zhuowen Tu and Song-Chun Zhu. Image segmentation by data-driven markov chain monte carlo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):657–673, 2002.

[10] Eric Veach and Leonidas J Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 419–428. ACM, 1995.

第 3 章　马尔可夫链蒙特卡罗 - 基础

松下问童子，言师采药去。只在此山中，云深不知处。
— 贾岛【779-843】

引言

马尔可夫

想象你进入了一个很大的国家公园（在上面的诗中是一座山），你的路径本质上是一个有界空间中的马尔可夫链。你驻足一个景点的频率正比于它的名气。你怎样才能预测你朋友在某一时刻 t 的位置 x? 该位置的不确定性是一个分布 $p_t(x)$。

马尔可夫链蒙特卡罗（MCMC）是一种在高维空间中从概率生成无偏样本通用技术，由从区间 $[a,b]$ 上的均匀分布中抽取的随机数驱动。马尔可夫链被设计为具有一个概率分布函数 $\pi(x)$ 作为其平稳（不变）概率。许多物理、化学和经济学中的随机系统可以用 MCMC 进行模拟。本章概述了马尔可夫链及其定义属性。此外，讨论了马尔可夫链唯一平稳分布的存在性定理，并给出了在模拟退火和网页流行度排序中的应用。

3.1　马尔可夫链基础

马尔可夫链是随机系统的一种数学模型，它的离散的或连续的状态由转移概率 P 控制。马尔可夫链中的当前状态仅取决于最近的先前状态，例如 1 阶马尔可夫链

$$X_t|X_{t-1},\ldots,X_0 \sim P(X_t|X_{t-1},\ldots,X_0) = P(X_t|X_{t-1})$$

马尔可夫性意味着空间或时间上的“局部性”，例如马尔可夫随机场和马尔可夫链。事实上，离散时间马尔可夫链可以看作是马尔可夫随机场（因果和一维）的一种特殊情况。

一个马尔可夫链通常表示为

$$\mathrm{MC} = (\Omega, \nu_0, K)$$

其中 Ω 是状态空间, $\nu_0 : \Omega \to \mathbb{R}$ 是状态上的初始概率分布，$K : \Omega \times \Omega \to \mathbb{R}$ 是转移概率，也称转移核。

假设 Ω 是可数的 (甚至更好，有限的), 则 K 是转移概率 $K(X_{t+1}|X_t)$ 的矩阵。在时间 n 处, 马尔可夫链状态将服从概率，

$$\nu_n = \nu_0 K^n.$$

例 3.1 假设有一个有限的状态空间，$|\Omega| = N \sim 10^{30}$，则转移概率 K 将由 $N \times N$ 转移矩阵表示

$$K(X_{t+1}|X_t) = \begin{bmatrix} k_{11} & \cdots & k_{N1} \\ \vdots & \ddots & \vdots \\ k_{1N} & \cdots & k_{NN} \end{bmatrix}_{(N\times N).}$$

转移矩阵通常是稀疏的，但并非总是如此。

因此，在 SMC 中我们尝试构造试验概率 $g(x)$, 而在 MCMC 中我们将构造转移矩阵 $K(X_{t+1}|X_t)$。因此,

$$X_n \sim \underbrace{(\cdots)_{(1\times N)}}_{\nu_n} = \underbrace{(\cdots)_{(1\times N)}}_{\nu_{n-1}} \begin{bmatrix} k_{11} & \cdots & k_{N1} \\ \vdots & \ddots & \vdots \\ k_{1N} & \cdots & k_{NN} \end{bmatrix}_{(N\times N).}$$

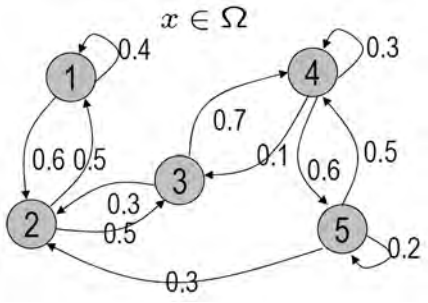


图 3.1: 五个家庭的贸易图示。

例 3.2 五个家庭。假设一个岛上有 *五*个家庭。有 *1,000,000* 个代币用作其货币，我们将财富归一化，使值为 *1* 意味着拥有所有代币。设 ν_t 表示 t 年之后 *5* 个家庭财富的 5×1 的向量。每个家庭都与其他家庭进行商品交易。例如，家庭 *1* 将花费 *60%* 的收入从家庭 *2* 中购买，并留下其余 *40%*，依此类推，如图 *3.1*所示。一个问题是：经过多年的发展，这些家庭间的财富将会怎样分配？以不同的方式提问，假设我们用特殊颜色（例如，红色）标记一个代币，在多年之后，我们想知道决定谁拥有这个代币的概率分布是什么。

我们将其转换为数学模型，用 $\Omega = \{1,2,3,4,5\}$ 表示红色代币的状态空间。则转移核为

$$K = \begin{pmatrix} 0.4 & 0.6 & 0.0 & 0.0 & 0.0 \\ 0.5 & 0.0 & 0.5 & 0.0 & 0.0 \\ 0.0 & 0.3 & 0.0 & 0.7 & 0.0 \\ 0.0 & 0.0 & 0.1 & 0.3 & 0.6 \\ 0.0 & 0.3 & 0.0 & 0.5 & 0.2 \end{pmatrix}.$$

从不同的初始条件开始计算财富分配，我们得到表 3.1中的结果。

表 3.1: 从不同的初始分布开始，在例 3.2中收敛后的最终财富分配。

Year	A					B				
1	1.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
2	0.4	0.6	0.0	0.0	0.0	0.0	0.3	0.0	0.7	0.0
3	0.46	0.24	0.30	0.0	0.0	0.15	0.0	0.22	0.21	0.42
4	⋯					⋯				
5	⋯					⋯				
6	0.23	0.21	0.16	0.21	0.17	0.17	0.16	0.16	0.26	0.25
...	⋯					⋯				
Final	0.17	0.20	0.13	0.28	0.21	0.17	0.20	0.13	0.28	0.21

在有限状态马尔可夫链的某些条件下，该状态收敛到一个不变概率

$$\lim_{n\to\infty} \nu_0 K^n = \pi.$$

在贝叶斯推理中，给一个目标概率 π，我们目的是构造一个马尔可夫链核 K，使 π 是 K 的唯一不变概率。

一般来说，有无数的 K' 具有相同的不变概率。

$$\begin{array}{ccccccccc} X_1 & \longrightarrow & X_2 & \longrightarrow & \cdots & \longrightarrow & X_n & \longrightarrow & \\ \wr & & \wr & & & & \wr & & \wr \\ \nu_1 & & \nu_2 & & \cdots & & \nu_n & & \pi \end{array}$$

假设给出 Ω 和目标概率 $\pi = (\pi_1, \cdots, \pi_N)_{(1\times N)}$，我们的目的是构造 ν_0 和 K 使得

1) $\pi K = \pi$; 这是马尔可夫链具有平稳概率 π 的必要条件。

2) 快速收敛。使用以下方式可以获得快速收敛：

   - 良好的初始概率。ν_0。
   - 良好的转移矩阵。K。

通常，由于局部连通性，转移矩阵是稀疏的（几乎处处为零），即，因为 MCMC 移动的新状态通常接近当前状态。但有些特例并非如此，例如章节 6中的 Swendsen-Wang 算法。

3.2 转移矩阵的拓扑：相通与周期

现在，我们核对马尔可夫链构造的条件。

1. 随机矩阵。核矩阵 K 应当是 一个随机矩阵, 即

$$\sum_{j=1}^{N} K_{ij} = 1, \qquad \forall i \in \Omega, \quad K_{ij} \geqslant 0,$$

或以矩阵形式：

$$K\mathbf{1} = \mathbf{1},$$

其中 $\mathbf{1}$ 是元素为 1 的 $N \times 1$ 向量: $\mathbf{1} = (1, ..., 1)^T$。

2. 全局平衡。另一个必要条件是全局平衡：

$$\pi K = \pi \qquad \rightarrow \qquad \sum_{i=1}^{N} \pi_i K_{ij} = \pi_j \qquad \forall j \in \Omega。$$

这种情况可以用*细致平衡* 条件（充分非必要条件）代替：

$$\pi(i) K_{ij} = \pi(j) K_{ji}, \ \forall\, i, j \in \Omega。 \tag{3.1}$$

实际上，细致平衡意味着平稳性：

$$\begin{aligned} \pi K &= \sum_i \pi(i) K_i = \sum_i \pi(i) \Big(K_{i1}, \ldots, K_{iN} \Big) \\ &= \sum_i \Big(\pi(1) K_{1i}, \ldots, \pi(N) K_{Ni} \Big) = \pi, \end{aligned}$$

特别是全局平衡

$$\sum_i \pi(i) K_{ij} = \sum_i \pi(j) K_{ji} = \pi(j) \sum_i K_{ji} = \pi(j)。$$

满足细致平衡条件的核称为*可逆的*。

回到例 3.2 我们可以推断，全局平衡方程表示总财富守恒。实际上，家庭 j 收到的总额是 $\sum_i \pi(i) K_{ij}$ ，它等于家庭 j 的财富 $\pi(j)$，即家庭 j 所花的金额。

在给定 π 的情况下，有非常多的方法来构造 K。在全局平衡中，我们有 $2N$ 个方程包含 $N \times N$ 个未知数，在细致平衡中我们有 $\frac{N^2}{2} + N$ 个方程包含 $N \times N$ 个未知数。

3. 不可约性。状态 j 可以从状态 i 可达，如果存在一个步骤 M，使 $(K^M)_{ij} > 0$,，其中

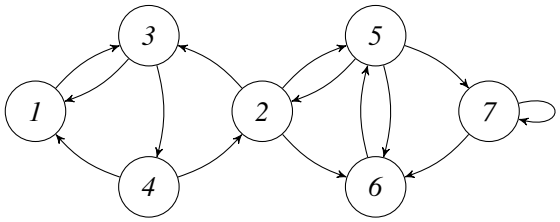
$$i \to j \qquad (K^M)_{ij} = \sum_{i_1, \ldots, i_{M-1}} K_{i i_1} \cdots K_{i_{M-1} j} > 0。$$

如果 i 和 j 相互可达，那么记 $i \leftrightarrow j$。相通关系 $\leftrightarrow$ 将状态空间划分为不相交的等价（相通）类，即

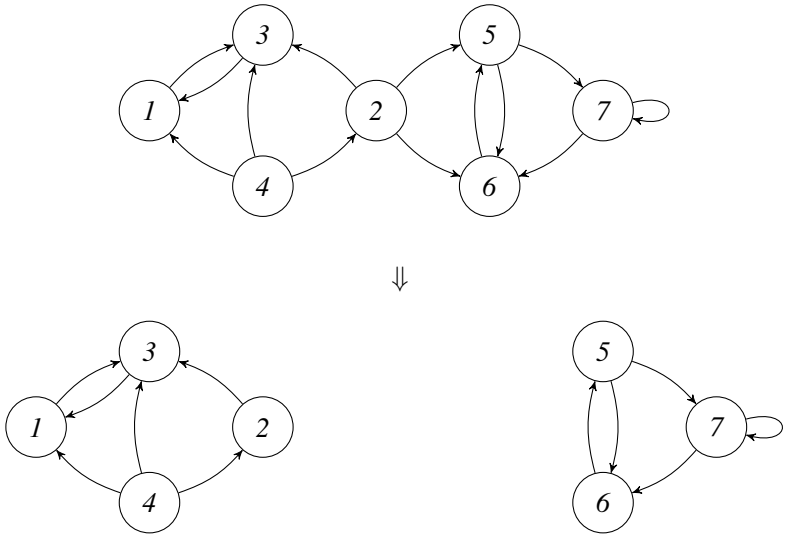
$$\Omega = \cup_{i=1}^{C} \Omega_i。$$

定义 3.1 *如果马尔可夫链的转移矩阵 K 只有 1 个相通类，则它是不可约的。*

例 3.3 *不可约的马尔可夫链:*



例 3.4 *可约的马尔可夫链:*



一般来说，贪心优化算法有一个可约链，并将陷入一个局部最优。

给定目标分布 π，理想的转换核是 $K = \begin{pmatrix} \pi \\ \pi \\ \cdots \\ \pi \end{pmatrix}$，无论它从哪里开始，它总是一步收敛。然而，通常很难直接对分布 π 进行采样，因此该核在实践中不是很有用。

4. 非周期性。为了定义非周期性，我们首先需要定义一个周期性马尔可夫链。

定义 3.2 *如果存在一个唯一的划分将图 G 分成 d 个循环类，则具有转移矩阵 K 的不可约马尔可夫链具有周期 d:*

$$C_1, \ldots, C_d, \qquad \sum_{j \in C_k} K_{ij} = 1, \qquad \forall i \in C_{k-1}.$$

备注 3.1 在周期性马尔可夫链中，每个单独循环类中的状态之间没有连接。转移矩阵是以下形式的分块矩阵。

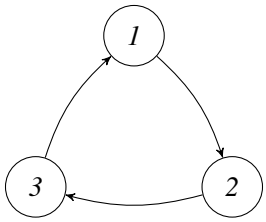
$$K = \begin{pmatrix} & & \\ & & \\ & & \end{pmatrix}$$

然后，K^d 的转移矩阵变为对角分块矩阵。

$$K^d = \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix}$$

这意味着 K 有一个相通类，但 K^d 有 d 个相通类。

例 3.5 考虑具有以下转移核的马尔可夫链。$K = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$.



它的周期是 *3* 并且在三个分布之间交替：$(1\,0\,0) \to (0\,1\,0) \to (0\,0\,1)$.

定义 3.3 如果具有转移矩阵 K 的不可约马尔可夫链的最大周期是 $d = 1$，则它是非周期的。

5. 平稳分布。具有转移核 K 的马尔可夫链具有平稳分布 π，如果满足 $\pi K = \pi$。

可能存在许多关于 K 的平稳分布。即使存在平稳分布，马尔可夫链也可能并不总收敛于它。

例 3.6 考虑具有转移核的马尔可夫链：$K = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$。$\pi = (\frac{1}{3}\ \frac{1}{3}\ \frac{1}{3})$ 是它的一个平稳分布，但可能永远不会收敛到它，如例 *3.5*所示。

3.3 Perron-Frobenius 定理

Theorem 3.4 (Perron-Frobenius) 对于任何原始（不可约和非周期）$N \times N$ 的随机矩阵 K，K 具有特征值

$$1 = \lambda_1 > |\lambda_2| > \cdots > |\lambda_r|$$

其具有多重性 $m_1,...,m_r$ 和左右特征向量 $(\mathbf{u}_i,\mathbf{v}_i)$。则 $\mathbf{u}_1=\pi,\mathbf{v}_1=\mathbf{1}$, 且

$$K^n=\mathbf{1}\cdot\pi+O(n^{m_2-1}|\lambda_2|^n).$$

定义 $\lambda_{\text{slem}}=|\lambda_2|$, 其是第二大特征值模，可以看到收敛速度取决于 λ_{slem}。

备注 3.2 如果 K 是可约的，并且具有 C 个相通类，则它是一个具有 C 块的分块对角矩阵。因此，特征值 1 具有至少 C 个不同的特征向量，且 K 不具有唯一不变概率。

备注 3.3 如果 K 是不可约的但是周期 $d>1$，那么它至少有 d 个模为 1 的不同的特征值，即 d^{th} 单位根。这是因为可以通过归纳证明它的特征多项式是 $\det(tI-K)=\det(t^dI-K_1...K_d)$，其中 $K_1,...,K_d$ 是非零块。$K_1,...,K_d$ 都具有特征值 1，所以 $U=K_1...K_d$ 具有特征值 1，因此其特征多项式 $\det(tI-U)$ 可被 $t-1$ 除尽。因此，$\det(tI-K)=\det(t^dI-U)$ 可被 t^d-1 除尽。

复习

假设 K 是 $N\times N$ 正非对称矩阵，并且具有 N 个特征值。

$$\begin{array}{ccccc}\lambda_1 & & \cdots\cdots & & \lambda_N\\ \wedge & & & & \wedge\\ u_1 & v_1 & & u_N & v_N\end{array}$$

每个特征值具有相应的左右特征向量。λ,u,v 都是复数。

$$\begin{aligned}u_iK&=\lambda_iu_i, &u_i&:1\times N\\ Kv_i&=\lambda_iv_i, &v_i&:N\times 1\end{aligned}$$

因此，

$$K=\lambda_1v_1u_1+\lambda_2v_2u_2+\cdots+\lambda_Nv_Nu_N$$

$$K\cdot K=\sum_{i=1}^N\lambda_iv_iu_i\cdot\sum_{j=1}^N\lambda_jv_ju_j=\sum_{\substack{i=1\\j=1}}^N\lambda_i\lambda_jv_iu_iv_ju_j,\qquad\begin{cases}\text{if } i\neq j & u_iv_j=0\\ \text{if } i=j & u_iv_j=1\end{cases}.$$

因此，

$$K^n=\lambda_1^nv_1u_1+\lambda_2^nv_2u_2+\cdots+\lambda_N^nv_Nu_N.$$

既然有全局平衡，

$$\left.\begin{array}{lll}\pi K=\pi & \Longrightarrow & \lambda_1=1,u_1=\pi\\ K\mathbf{1}=\mathbf{1} & \Longrightarrow & \lambda_1=1,v_1=\mathbf{1}\end{array}\right\}\Longrightarrow\lambda_1\cdot v_1\cdot u_1=\begin{pmatrix}\pi\\ \pi\\ \dots\\ \pi\end{pmatrix}.$$

因此，

$$K^n=\begin{pmatrix}\pi\\ \pi\\ \dots\\ \pi\end{pmatrix}+\underbrace{\varepsilon}_{\to 0}\qquad,\text{if } |\lambda_i|<1,\forall i>1,$$

所以当 $n \to \infty$ 时，K^n 逼近理想的核 $\begin{pmatrix} \pi \\ \pi \\ \cdots \\ \pi \end{pmatrix}$。

3.4 收敛措施

比较让人感兴趣的是状态 i，它是概率的全局最优，

$$i^* = \operatorname{argmax} \pi(x)。$$

定义 3.5 给定具有转移核 K 和不变概率 π 的马尔可夫链 $(x_0,...,x_n,...)$，我们定义

i) 状态 i 的首中时（在有限状态下）

$$\tau_{hit}(i) = \inf\{n \geqslant 1; x_n = i, x_0 \sim \nu_0\}, \qquad \forall i \in \Omega。$$

$E[\tau_{hit}(i)]$ 是由 K 控制的马尔可夫链的 i 的平均首中时。

ii) 状态 i 的首次返回时间

$$\tau_{ret}(i) = \inf\{n \geqslant 1; x_n = i, x_0 = i\}, \qquad \forall i \in \Omega。$$

iii) 混合时间

$$\tau_{mix} = \min_n \{\|\nu_0 K^n - \pi\|_{TV} \leqslant \varepsilon, \forall \nu_0\},$$

其中总变差定义为

$$\|\mu - \nu\|_{TV} = \frac{1}{2} \sum_{i \in \Omega} |\mu(i) - \nu(i)| = \sum_A \Big(\mu(i) - \nu(i)\Big), \qquad A = \{i : \mu(i) \geqslant \nu(i), i \in \Omega\}。$$

定义 3.6 K 的收缩系数是转移核中任意两行之间的最大总变差（TV）范数，它由下式计算

$$C(K) = \max_{x,y} \|K(x,\cdot) - K(y,\cdot)\|_{TV}。$$

例 3.7 考虑生活在岛上的五个家庭的马尔可夫核，其中他们的值与例 *3.2*中的值不同

$$K = \begin{pmatrix} 0.3, & 0.6, & 0.1, & 0.0, & 0.0 \\ 0.2 & 0.0, & 0.7, & 0.0, & 0.1 \\ 0.0, & 0.5, & 0.0, & 0.5, & 0.0 \\ 0.0, & 0.0, & 0.4, & 0.1, & 0.5 \\ 0.4, & 0.1, & 0.0, & 0.4, & 0.1 \end{pmatrix}$$

1) 在图 *3.2*左边，我们在 *2D* 平面中绘制了五个复特征值。

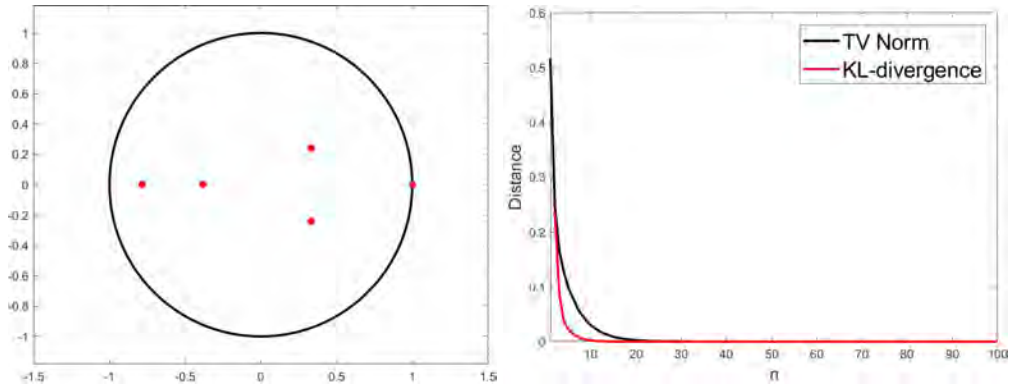


图 3.2: 例 3.7的五个家庭的核。左：五个复特征值。右：$\mu_n = \nu \cdot K^n$ 和不变概率 π 之间的 TV 范数和 KL 散度。

不变概率是 $\pi = (0.1488\ 0.2353\ 0.2635\ 0.2098\ 0.1427)$。第二大特征值具有 $\lambda_{slem} = \|\lambda_2\| = 0.7833$。

2) 假设我们从初始概率 $\nu = (1,0,0,0,0)$ 开始，即我们确定初始状态是在 $x_0 = 1$。因此, 在第 n 步, 马尔可夫链状态服从分布 $\mu_n = \nu \cdot K^n$。我们使用 *TV* 范数计算 μ_n 和 π 之间的距离，

$$d_{\mathrm{TV}}(n) = ||\pi - \mu_n||_{\mathrm{TV}} = \frac{1}{2}\sum_{i=1}^{5}|\pi(i) - \mu_n(i)|,$$

或者 *KL* 散度,

$$d_{\mathrm{KL}}(n) = \sum_{i=1}^{5}\pi(i)\log\frac{\pi(i)}{\mu_n(i)}。$$

图 *3.2*右边显示了前 *140* 步的两个距离 $d_{\mathrm{TV}}(n)$ 和 $d_{\mathrm{KL}}(n)$。

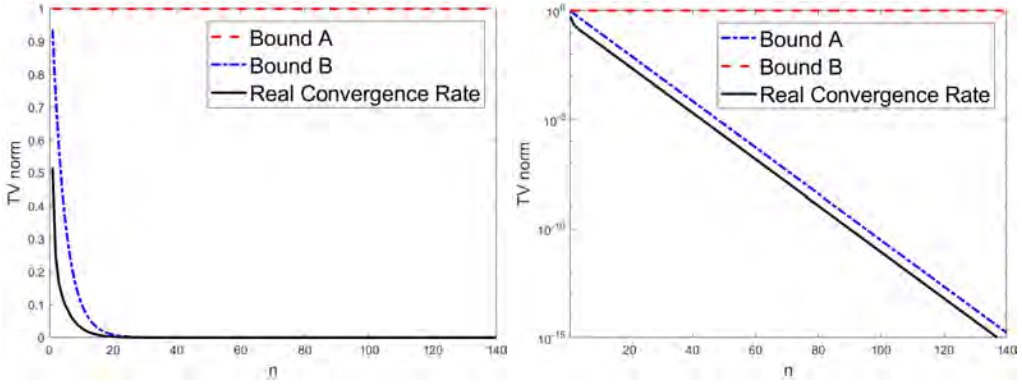


图 3.3: $\mu_n = \nu \cdot K^n$ 和不变概率 π 之间的 TV 范数以及来自方程(3.2)和(3.3)的两个界限 $A(n)$ 和 $B(n)$。左：原始比例。右：对数比例。

3) 我们计算 *K* 的收缩系数。注意收缩系数是转移核中任意两行之间的最大 *TV* 范数，

$$C(K) = \max_{x,y}||K(x,\cdot) - K(y,\cdot)||_{\mathrm{TV}}。$$

我们可以证明

$$||\nu_1 \cdot K - \nu_2 \cdot K||_{\mathrm{TV}} \leq C(K)||\nu_1 - \nu_2||_{\mathrm{TV}}。$$

因为 $||\nu_1 - \nu_2||_{\mathrm{TV}} \leq 1$, 如果 $C(K) < 1$, 那么收敛速度的上界为

$$A(n) = C^n(K) \geq C^n(K)||\nu_1 - \nu_2||_{\mathrm{TV}} \geq ||\nu_1 \cdot K^n - \nu_2 \cdot K^n||_{\mathrm{TV}}, \quad \forall \nu_1, \nu_2。 \tag{3.2}$$

对于这个例子，可以看到 $C(K) = 1$，所以约束不是很有用。

4) 另一个界限 – *Diaconis-Hanlon* 界限为

$$B(n) = \sqrt{\frac{1-\pi(x_0)}{4\pi(x_0)}}\lambda_{\mathrm{slem}}^n \geq ||\pi - \nu K^n||_{\mathrm{TV}}, \tag{3.3}$$

其中 $x_0 = 1$ 是初始状态，$\pi(x_0)$ 是 $x = 1$ 时的目标概率。在原始比例和对数比例上，图 *3.3*显示了实际收敛速度 $d_{\mathrm{TV}}(n)$ 与 $A(n)$ 和 $B(n)$ 的比较。该界限保持直到达到机器精度。

3.5 连续或异构状态空间中的马尔可夫链

在连续情况下，目标分布 $\pi : \Omega \to \mathbb{R}$ 是一个概率密度函数 $\pi(x)$，转移核是一个条件概率密度函数 $K(x,y) = K(y|x)$, 所以 $\int_\Omega K(x,y)\mathrm{d}y = 1$。

则对于任何事件 $A \subseteq \Omega$ 都一定满足全局平衡方程，

$$\pi K(A) = \int_A \int_\Omega \pi(x)K(x,y)\mathrm{d}x\mathrm{d}y = \int_A \pi(x)\mathrm{d}x = \pi(A),$$

连续情况下的细致平衡方程为

$$\int_A \int_B \pi(x)K(x,y)\mathrm{d}x\mathrm{d}y = \int_A \int_B \pi(y)K(y,x)\mathrm{d}x\mathrm{d}y。$$

在实践中，Ω 是由离散/有限和连续变量组成的混合/异构空间。

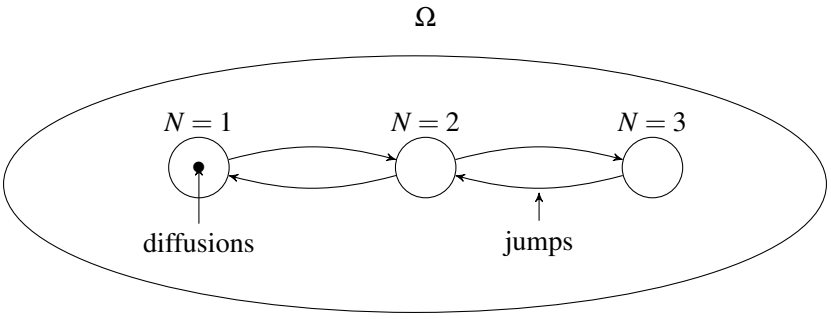


图 3.4: 例 3.8中异构空间的跳跃-扩散过程。

例 3.8 考虑 $X = \{N, (x_i, y_i), i = 1, \ldots, N\}$ 的异构空间, 其中 N 是一个图片中的人数，(x_i, y_i) 是他们的位置。

在这种情况下，有许多不同的马尔可夫链过程，如图 *3.4*所示。不可约的 *MCMC* 将具有许多动态过程（子链），例如：

$$\left\{\begin{array}{l} \text{跳跃}\left\{\begin{array}{l}\text{死亡/出生}\\ \text{分开/合并}\end{array}\right\}\text{过程} \\ \text{扩散}\end{array}\right. .$$

3.6 各态历经性定理

定义 3.7 状态 i 被称为常返的，如果有

$$P(\tau_{ret}(i) < \infty) = 1.$$

否则称为非常返的。$\tau_{ret}(i)$ 是返回时间, 即从 x 返回到 x 所需的总的步数。

定义 3.8 满足

$$E[\tau_{ret}(i)] < \infty$$

的状态 i 称为正常返的，否则为零常返的。如果一个马尔可夫链的所有状态都是正常返的，那么这个马尔可夫链就是正常返的。

通常，正常返是有无限状态的空间的一个条件。

Theorem 1 (各态历经性定理) 对于一个具有不可约的、正常返的、具有平稳概率 π 的马尔可夫链，在状态空间 Ω 中，设 $f(x)$ 为具有关于 π 的有限均值的任何实值函数，那么对于任意初始概率，几乎肯定有

$$\lim_{N\to\infty}\frac{1}{N}\sum_{i=1}^{N} f(x_i) = \sum_{x\in\Omega} f(x)\pi(x) = E_\pi[f(x)], \qquad \forall f$$

其中 x_i 是马尔可夫链状态 *(*但不需要是独立同分布*)*。

3.7 通过模拟退火进行 MCMC 优化

一个 MCMC 算法被设计从后验分布 $\pi, X \sim \pi$ 获得样本。我们看到在某些条件下（细致平衡，不可约性和非周期性），马尔可夫链不变概率将在老化期后收敛到平稳分布 π。

在运行马尔可夫链时通过缓慢改变平稳分布 π，MCMC 还可以运用于优化。假设我们想要最大化函数 $f(x) : \Omega \to \mathbb{R}$。我们考虑后验概率

$$\pi(x;T) = \frac{1}{Z(T)}\exp(-f(x)/T),$$

其依赖于温度参数 T。当 T 很大时，概率 $\pi(x,T)$ 将具有较小的峰值和局部最大值，使其更容易采样。当 T 非常小时，概率 $\pi(x,T)$ 将集中在其全局最大值，如图 3.5所示。

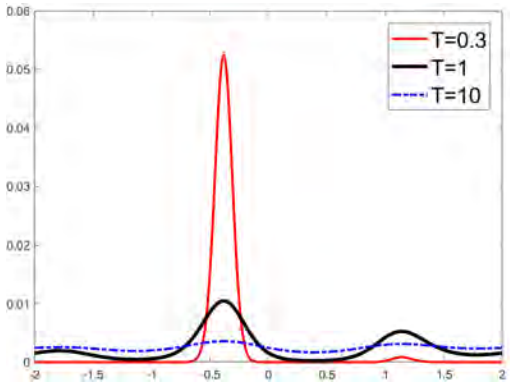


图 3.5: 温度对概率分布的影响。在温度 $T = 10$ 时，概率接近均匀分布，而在 $T = 0.3$ 时，全局最优值清楚地呈现。

Scott Kirkpatrick

退火程序 [1] 要求在高温下启动马尔可夫链并缓慢降低直到非常低的温度。该方法的灵感来自用于在金属或其他材料中产生晶体结构的退火方法。在该过程中，将材料加热至高温直至其熔化，然后缓慢降低温度以允许原子将其自身定位在低能量构型中，产生晶体结构。如果材料冷却太快，会产生裂缝或其他缺陷，如小晶体。如果冷却足够慢，则可以获得具有较少缺陷的较大晶体。

类似地，对于优化，需要选择*退火程式*，其指定了在马尔可夫链每一步使用的温度。该程式从较高的 T_0 开始，当 $k \to \infty$ 时减小到 0，即 $\lim_{k\to\infty} T_k = 0$。通过该退火程式，概率 $\pi(x,T)$ 变为 $\pi(x,T_k)$，一个时间相关的概率。下表描述模拟退火算法，对于任何 $x \in \Omega$, $N(x)$ 是在一个马尔可夫链步骤中可从状态 x 到达的可能状态的集合。

模拟退火

input: 初始解 $x \in \Omega$
input: 温度冷却程式, T_k
input: 初始温度 $T = T_0 > 0$
input: 重复程式 M_k-每个温度下执行的迭代次数 T_k
设置温度变化计数器 $k = 0$
repeat
  for $m = 0$ to M_k **do**
    生成一个解 $x' \in N(x)$
    计算 $\Delta_{x,x'} = f(x') - f(x)$
    if $\Delta_{x,x'} \leq 0$ **then**
      $x \leftarrow x'$
    else
      $x \leftarrow x'$，以概率 $\exp(-\Delta_{x,x'}/T_k)$
    end if
  end for
  k=k+1
until 满足停止标准

基于将上述算法建模为齐次马尔可夫链序列或单个非齐次马尔可夫链，存在两类收敛结果。以下是来自 Mitra [2] 的非齐次马尔可夫链结果。

Theorem 3.9 *(Mitra 1986)* 与具有以下更新函数的模拟退火相关的马尔可夫链，

$$T_k = \frac{\gamma}{\log(k+k_0+1)}$$

对于任何参数 $k_0 \geq 1$ 且足够大的 γ，无论初始解 $x \in \Omega$ 是什么，都会收敛到全局最优。

实际中，我们不能等太久才能找到解，实践中采用更快的退火程式，线性、甚至指数地递减 t。利用这些程式，优化算法找到局部最优，其好坏依赖于退火程式和使用的 MC 算法。一些 MCMC 算法，如 Gibbs 采样器，需要缓慢的退火程式来得到较好的解；而其他算法，如 Swendsen-Wang Cut，则允许更快的冷却程式来得到类似解。

备注 3.4 在许多计算机视觉问题中，寻找 $\pi(x)$ 的全局最优解可能是 *NP-hard* 问题，这意味着不会找到一个多项式优化算法。在这些情况下，退火程式必须以对数方式减少 t，从相对于问题大小（如 Ω 的维数）使全局最优在指数时间内被找到。

3.7.1 网页排序示例

作为 MCMC 应用的最后一个例子，我们考虑对一组网页的重要性进行排序。考虑一个有向图 $G = \langle V, E \rangle$，其中页面作为节点集合 V, 页面链接作为边缘集合 E。对于特定页面 x，考虑以下两个集合

$$out(x) = \{w | x \to w \in E\}, \quad in(x) = \{y | y \to x \in E\}。$$

为了创造一个页面重要性的充分度量，我们需要考虑其连接的页面的两个特征。

1. x 的链接是来自具有许多其他链接的页面，还是来自仅显示几个选项的页面？
2. x 的链接是来自著名的高流量的页面，还是来自个人网站或博客？

第一点表示，对于所有连接到 x 的页面 y，排序度量应该考虑 $|out(y)|$；而第二点表示页面的重要性 $\pi(x)$ 应该相对于连接到它的页面的重要性 $\pi(y)$ 递归地定义。考虑到这些，我们定义

$$\pi(x) = \sum_{y \in in(x)} \frac{\pi(y)}{|out(y)|}。$$

为了从排序网页的这种分布中采样，我们使用具有以下转移概率的 MCMC

$$K(y,x) = \frac{1}{|out(y)|}。$$

π 确实是这个马尔可夫链的一个平稳分布，因为 K 满足

$$\sum_{y \in V} \pi(y)K(y,x) = \sum_{y \in in(x)} \pi(y)K(y,x) = \sum_{y \in in(x)} \frac{\pi(y)}{|out(y)|} = \pi(x)。$$

虽然这个链的确具有 π 作为一个平稳分布，但它不是各态历经的，因为可能有几个页面不包含链接，除非 G 具有高连通性。出于这个原因，我们引入了用户在链接中键入并跳转到一个未连接页面的概率 α。如果有 $N = |V|$ 的网页，则新的转移概率为

$$K(x,y) = \begin{cases} \frac{1-\alpha}{N} & x \to y \notin E \\ \frac{1-\alpha}{N} + \frac{\alpha}{|out(y)|} & x \to y \in E \end{cases}.$$

因为无论图形如何连接，这个新链都是不可约的，所以它是各态历经的。

为了证明这一观点，图 3.6显示了一个个人网站的图表示，图中的转移概率由上面的式子计算。该网站包含 5 个页面：Homepage，About，Projects，Publications，以及 Contact。每页的链接也在下表 3.2中给出。

表 3.2: 示例网页中显示的链接。

网页	Homepage	About	Projects	Publications	Contact
链接	About Projects Publications Contact	Homepage Publications Contact	Homepage Publications	Homepage About Projects	Homepage About

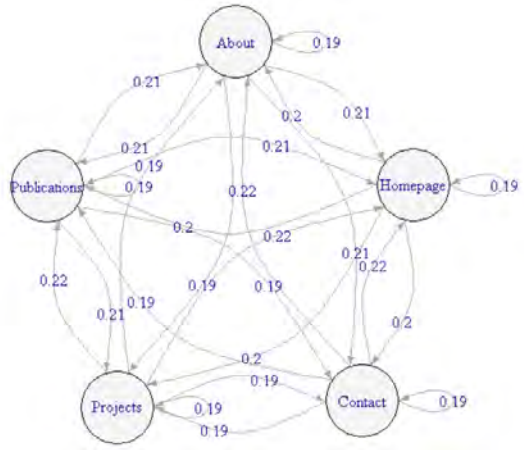


图 3.6: pagerank 应用的示例图。

在所有状态下，用户都可以点击后退按钮返回上一页并刷新当前页面。用户总是从主页开始，即初始概率为 $(1,0,0,0,0)$。运行此马尔可夫链，将会通过产生用户访问特定页面的平稳概率来生成页面的排序。

练习

问题 1. 考虑生活在一个岛上的五个家庭的马尔可夫核，其中数字发生了变化，

$$K = \begin{pmatrix} 0.3, & 0.6, & 0.1, & 0.0, & 0.0 \\ 0.2 & 0.0, & 0.7, & 0.0, & 0.1 \\ 0.0, & 0.5, & 0.0, & 0.5, & 0.0 \\ 0.0, & 0.0, & 0.4, & 0.1, & 0.5 \\ 0.4, & 0.1, & 0.0, & 0.4, & 0.1 \end{pmatrix}$$

1). 计算五个特征值，以及它们对应的左右特征向量（可以使用任何软件包）。

- 在 2D 平面中绘制 5 个特征值（复数），即将它们显示为单位圆中的点（绘制单位圆以供参考）。
- 它的不变概率 π 是多少？
- λ_{slem} 的值是多少？

2). 假设从初始概率 $\nu = (1,0,0,0,0)$ 开始，即确定初始状态是在 $x_0 = 1$。因此，在第 n 步，马尔可夫链状态服从分布 $\mu_n = \nu \cdot K^n$。通过 TV 范数计算 μ_n 和 π 之间的距离，

$$d_{\text{TV}}(n) = ||\pi - \mu_n||_{\text{TV}} = \frac{1}{2}\sum_{i=1}^{5} |\pi(i) - \mu_n(i)|;$$

或 KL 散度，

$$d_{\text{KL}}(n) = \sum_{i=1}^{5} \pi(i) \log \frac{\pi(i)}{\mu_n(i)}。$$

绘制前 1000 步的 $d_{\text{TV}}(n)$ 和 $d_{\text{KL}}(n)$。

3). 计算 K 的收缩系数。注意收缩系数是转移核中任意两行之间的最大 TV 范数，

$$C(K) = \max_{x,y} ||K(x,\cdot) - K(y,\cdot)||_{\text{TV}}。$$

可以证明

$$||\nu_1 \cdot K - \nu_2 \cdot K||_{\text{TV}} \leq C(K)||\nu_1 - \nu_2||_{\text{TV}}$$

因为 $||\nu_1 - \nu_2||_{\text{TV}} \leq 1$, 如果 $C(K) < 1$ 则收敛速度的上界为

$$A(n) = ||\nu_1 \cdot K^n - \nu_2 \cdot K^n||_{\text{TV}} \leq C^n(K)||\nu_1 - \nu_2||_{\text{TV}} \leq C^n(K), \quad \forall \nu_1, \nu_2。$$

在 $n = 1, ..., 1000$ 上绘制界限 $C^n(K)$ 。

4). 另一个界限 – Diaconis-Hanlon 界限为

$$B(n) = ||\pi - \nu K^n||_{\text{TV}} \leq \sqrt{\frac{1-\pi(x_0)}{4\pi(x_0)}}\lambda_{\text{slem}}^n$$

其中 $x_0 = 1$ 是初始状态，$\pi(x_0)$ 是 $x = 1$ 处的目标概率。与 $A(n)$ 和 $B(n)$ 相比，绘制实际收敛速度 $d_{\text{TV}}(n)$ 。

[在同一图中绘制三条曲线进行比较，然后绘制第二张图比较它们的对数图，因为它们是指数率。]

5). 我们定义了一个新的具有转移核 $P = K^n$ 的马尔可夫链。然后在 2 维复平面上绘制 P 的 5 个特征值，如在 1 所做的那样。显示这些特征值在三个阶段 $n = 10, 100, 1000$ 如何在平面上移动。最好画出 5 个点的轨迹（连接 5 个点的移动以显示它们的轨迹）。

打印 $n = 1000$ 的矩阵 P，看看它是否变成了“理想”的转移核。

问题 2. 现在考虑两个以上的转移矩阵

$$K_1 = \begin{pmatrix} 0.1, & 0.4, & 0.3, & 0.0, & 0.2 \\ 0.5 & 0.3, & 0.2, & 0.0, & 0.0 \\ 0.0, & 0.4, & 0.5, & 0.1, & 0.0 \\ 0.0, & 0.0, & 0.0, & 0.5, & 0.5 \\ 0.0, & 0.0, & 0.0, & 0.7, & 0.3 \end{pmatrix}, \quad K_2 = \begin{pmatrix} 0.0, & 0.0, & 0.0, & 0.4, & 0.6 \\ 0.0 & 0.0, & 0.0, & 0.5, & 0.5 \\ 0.0, & 0.0, & 0.0, & 0.9, & 0.1 \\ 0.0, & 0.2, & 0.8, & 0.0, & 0.0 \\ 0.3, & 0.0, & 0.7, & 0.0, & 0.0 \end{pmatrix}$$

1) K_1 和 K_2 是不可约的，非周期性的吗？2) 打印出两个矩阵的 5 个特征值和 5 个特征向量。3) 每个矩阵的不变概率是多少？

问题 3. 马尔可夫链的返回时间 $\tau_{\text{ret}}(i)$ 是马尔可夫链离开状态 i 后返回 i 的最小步数。假设我们在可数的非负数集合 $\Omega = \{0, 1, 2, ...,\}$ 中考虑随机行走。在一步中，马尔可夫链状态 $x_t = n$，有 α 的上升概率（即 $x_{t+1} = n+1$）和概率 $1-\alpha$ 返回到 $x_{t+1} = 0$。计算在有限步数内返回状态 0 的概率

$$Prob(\tau_{\text{ret}}(0) < \infty) = \sum_{\tau(0)=1}^{\infty} Prob(\tau(0))。$$

计算预期返回时间

$$E[\tau_{\text{ret}}(0)]。$$

问题 4. 设 Ω 为具有 $|\Omega| = N$ 个状态的有限状态空间，P 为 Ω 上具有不变概率 π 的 $N \times N$ 马尔可夫核。（注意 P 服从全局平衡方程，不一定是细致平衡方程）。我们定义一个具有核 Q 的反向链为满足以下方程的随机矩阵，

$$\pi(x)Q(x,y) = \pi(y)P(y,x), \quad \forall x, y。$$

证明 π 也是 Q 的不变概率。

问题 5. 在有限状态空间 Ω 中，假设在耦合模式中运行两个马尔可夫链 $\{X_t\}_{t\geq 0}$ 且 $\{Y_t\}_{r\geq 0}$，即两个链在每一步共享相同的转移核 P。假设第一个链是 $X_t \sim \pi$ 的平稳链，第二个链具有状态概率 $Y_t \sim \mu_t$。考虑联合概率 $Pr(X_t, Y_t)$，证明

$$||\pi - \mu_t||_{\text{TV}} \leq 1 - Pr(X_t = Y_t)$$

也就是说，在时间 t 的 TV 范数小于 1 减去两条链折叠（耦合）的概率。

提示：TV 范数可以写成其他形式：

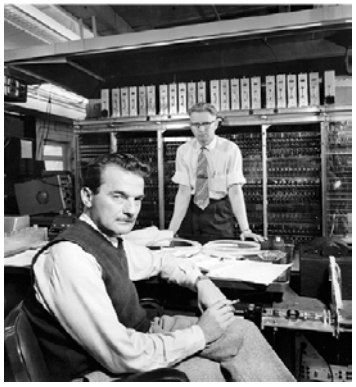
$$||\mu - \nu||_{\text{TV}} = \max_{A\subset\Omega}(\mu(A) - \nu(A)).$$

参考文献

[1] Scott Kirkpatrick, MP Vecchi, et al. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[2] D Mitra, F Romeo, and A Sangiovanni-Vincentelli. Convergence and finite-time behavior of simulated annealing. *Advances in applied probability*, 18(3):747–771, 1986.

第 4 章　Metropolis 方法和变体



Nicholas Metropolis 坐在 MANIAC 电脑前

“我们大多数人都对计算机的发展和能力（即使有一些是引人注目的）感到厌倦，很难相信或想象曾经有过一段时间我们忍受着嘈杂的、十分缓慢的、随穿孔卡片咯咯作响的机电设备 ” - Nicholas Metropolis

介绍

Metropolis 算法 [15, 16] 已被宣布为被 Dongarra 和 Sullivan 所引用的 20 世纪十大算法之一 [4]。原算法 [15] 被提议用于化学物理方程，并且已被 Hastings [10] 推广到当前形式。在本章中，我们讨论了原算法的几种变体，并讨论了可逆跳跃和扩散的概念。应用方面主要包括简单的图像分割，家具布置和人数统计。

4.1　Metropolis-Hastings 算法

Metropolis-Hastings 算法是一种处理从当前状态 X 跳转到新状态 Y 的任意算法的简单方法，它通过接受有概率的移动小幅修改它，以使得到的算法满足细致平衡方程(3.1)。

Wilfred Keith Hastings

例 4.1 *Metropolis-Hasting* 算法的想法如图 *4.1*所示。假设有一个“提议”算法试图根据图 *4.1*左显示的概率在 X 和 Y 之间移动。

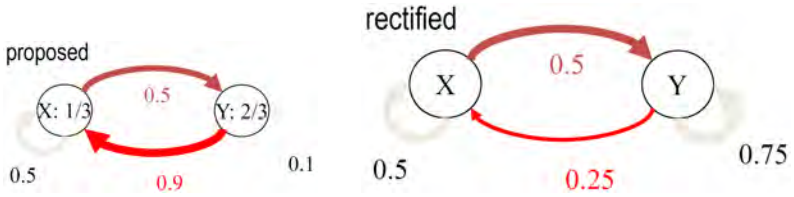


图 4.1: Metropolis-Hastings 算法图示。左：提议的状态 X 和 Y 之间的移动不满足细致平衡。右：纠正转移概率以满足细致平衡方程。

由于 $\pi(X) = 1/3$ 和 $\pi(Y) = 2/3$，细致平衡方程为

$$K(X,Y)\frac{1}{3} = K(Y,X)\frac{2}{3},$$

很容易检查在提议的转移概率下不满足。

X 和 Y 之间的移动用一个接受概率 $\alpha = \frac{0.5\times\frac{1}{3}}{0.9\times\frac{2}{3}} = \frac{5}{18}$ 进行纠正。只允许从 Y 到 X $\frac{5}{18}$ 的提议，允许从 X 到 Y 的所有提议。接受概率纠正了提议概率，因此 *MC* 服从目标分布。纠正的概率如图 *4.1*右所示。

4.1.1 原始 Metropolis-Hastings 算法

Metropolis Hastings 算法类似于重要性采样，因为它使用更简单的分布 $Q(x,y)$ 来产生提议样本，然后通过接受概率重新加权。一般来说，提议分布 $Q(x,y)$ 比较简单，可以轻松获得以 x 为条件的 y 的样本。

输入: 目标概率分布 $\pi(x)$、当前状态 $x^{(t)} \in \Omega$、和提议概率分布 $Q(x,y)$。
输出: 新状态 $x^{(t+1)} \in \Omega$
1. 通过从 $Q(x^{(t)},y)$ 采样提议一个新状态 y。
2. 计算接受概率:

$$\alpha(x,y) = \min\left(1, \frac{Q(y,x)}{Q(x,y)} \cdot \frac{\pi(y)}{\pi(x)}\right) \tag{4.1}$$

3. 以概率 $\alpha(x,y)$ 接受移动并使 $x^{(t+1)} = y$, 否则 $x^{(t+1)} = x^{(t)}$。

图 4.2: **Metropolis-Hastings 算法的一步**

Theorem 4.1 (Metropolis-Hastings) 图 *4.2*中的 *Metropolis-Hastings* 算法满足细致平衡方程。

证明. 有

$$\underbrace{K(x,y)}_{\text{转移概率}} = \begin{cases} \underbrace{Q(x,y)}_{\text{提议}} \cdot \underbrace{\alpha(x,y)}_{\text{接受率}} = Q(x,y) \cdot \min\left(1, \underbrace{\frac{Q(y,x)}{Q(x,y)}}_{\text{提议}} \cdot \underbrace{\frac{\pi(y)}{\pi(x)}}_{\text{验证}}\right), & \forall y \neq x。 \\ 1 - \sum_{y \neq x} Q(x,y)\alpha(x,y), & y = x \end{cases}$$

由于，

$$\alpha(x,y) = \min\Big(1, \frac{Q(y,x)}{Q(x,y)} \cdot \frac{\pi(y)}{\pi(x)}\Big),$$
$$\alpha(y,x) = \min\Big(1, \frac{Q(x,y)}{Q(y,x)} \cdot \frac{\pi(x)}{\pi(y)}\Big),$$

我们有 $\alpha(x,y) = 1$ 或 $\alpha(y,x) = 1$。因此，对细致平衡，左侧是

$$\pi(x)K(x,y) = \pi(x)Q(x,y)\alpha(x,y) = \pi(x)Q(x,y)\min\Big(1, \frac{Q(y,x)}{Q(x,y)} \cdot \frac{\pi(y)}{\pi(x)}\Big) = \min\Big(\pi(x)Q(x,y), \pi(y)Q(y,x)\Big)$$

, 右侧是

$$\pi(y)K(y,x) = \pi(y)Q(y,x)\alpha(y,x) = \pi(y)Q(y,x)\min\Big(1, \frac{Q(x,y)}{Q(y,x)} \cdot \frac{\pi(x)}{\pi(y)}\Big) = \min\Big(\pi(x)Q(x,y), \pi(y)Q(y,x)\Big)$$

;

因此细致平衡方程是满足的。□

4.1.2 Metropolis-Hastings 算法的另一版本

在很多情况下，目标概率写为一个 Gibbs 分布

$$\pi(x) = \frac{1}{Z}e^{-E(x)}$$

, 其归一化常数很难计算。假设提议概率是对称的 ($Q(x,y) = Q(y,x)$)，接受概率变为

$$\alpha(x,y) = \min\Big(1, \frac{\pi(x)}{\pi(y)}\Big) = \min\big(1, e^{-(E(x)-E(y))}\big) = \min(1, e^{-\Delta E})。$$

因此,

$$\alpha(x,y) = 1, \quad \text{如果}\Delta E < 0,\ y \text{ 是一个比}x\text{更低（好）的能量状态。}$$
$$\alpha(x,y) = e^{-\Delta E} < 1, \quad \text{如果}\Delta E > 0,\ y \text{ 是一个比}x\text{更高（差）的能量状态。}$$

因为两个状态 x 和 y 共享其大部分元素，ΔE 通常在局部计算。当提议被拒绝时（概率为 $1-\alpha$），马尔可夫链保持在 x 状态。

该过程如图 4.3所示。注意 $Q(y,x)$ 旨在给出可以在正确方向引导马尔可夫链的正确提议。

备注 4.1 我们必须注意此假设 $Q(x,y) = Q(y,x)$，因为它在 Ω 域的边界处通常无法满足。

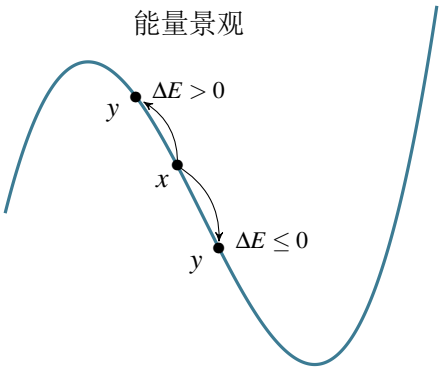


图 4.3: Gibbs 分布的 Metropolis-Hastings 算法变体图示

。

4.1.3 其他接受概率设计

存在能够保证细致平衡方程的其他接受率设计，如

$$\alpha(x,y) = \frac{\pi(y)Q(y,x)}{\pi(y)Q(y,x) + \pi(x)Q(x,y)}$$

或更一般地说

$$\alpha(x,y) = \frac{s(x,y)}{\pi(x)Q(x,y)}$$

, 其中 $s(x,y)$ 是一个任意对称函数。

备注 4.2 *回到例 3.2，可以认为 $\pi(x)$ 是经常相互交易的一些家庭的财富平衡分布。$Q(x,y)$ 可视为家庭之间的贸易提议。在这种情况下，Metropolis-Hastings 接受概率的选择在所有基于满足细致平衡的 $Q(x,y)$ 的设计中最大化所有家庭之间的贸易。*

4.1.4 Metropolis 设计中的关键问题

直觉上，Metropolis-Hastings 方法允许概率爬出局部最小值。设计 Metropolis 算法的关键问题是设计提议概率 $Q(x,y)$。$Q(x,y)$ 的一些希望的属性是：

i. 对于任意 x，可达状态集合 $\{y, Q(x,y) > 0\}$ 很大，因此 $K(x,y)$ 连接更紧密。

ii. 对于任意 x，概率 $Q(x,y)$ 远非均匀（即信息确切）。

4.2 独立 Metropolis 采样

独立 Metropolis 采样器 (IMS) 是提议独立于链的当前状态的一种 Metropolis-Hastings 算法。它也被称为 Metropolized 独立采样 (Liu [12])。其目标是模拟在 Ω 中取值且具有平稳分布 $\pi = (\pi_1, \pi_2, ..., \pi_N)$（目标概率）的马尔可夫链 $\{X_m\}_{m\geq 0}$，其具有非常大的 N，例如 $N = 10^{20}$，在这种情况下实际上不可能枚举

所有状态。在这种情况下，在每一步根据 $j \sim q_j$ 从提议概率 $q = (q_1, q_2, \ldots, q_N)$ 中采样得到一个新状态 $j \in \Omega$，其以概率

$$\alpha(i,j) = \min\{1, \frac{q_i}{\pi_i}\frac{\pi_j}{q_j}\}.$$

接受。因此，从 X_m 到 X_{m+1} 的转移由具有以下形式的转移核决定

$$\mathcal{K}(i,j) = \begin{cases} q_j\alpha(i,j), & j \neq i, \\ 1 - \sum_{k\neq i}\mathcal{K}(i,k), & j = i。 \end{cases}$$

初始状态可以是固定的，也可以从在这种情况下自然选择为 q 的分布生成。在 4.2.3部分，我们说明为什么从 q 生成初始状态而不是确定性地选择它更有效。

很容易证明 π 是链的不变（平稳）分布。换句话说，$\pi\mathcal{K} = \pi$。由于 $q > 0$，因此 $\mathcal{K}$ 是各态历经的，则 π 也是链的平衡分布。因此当 m 足够大时，在第 m 步，链的边缘分布近似为 pi。

然而相比于试图从目标分布 π 进行采样，我们可能更有兴趣搜索一个状态 i^* 有最大概率 $i^* = \arg\max_{i\in\Omega}\pi_i$。这是平均首中时可以起作用的地方。$E[\tau(i^*)]$ 通常是衡量搜索速度的一个好方法。作为一个特殊情况，我们想知道最优状态的 $E[\tau(i^*)]$。

此分析的一个关键量是概率比 $w_i = q_i/\pi_i$，这点会在之后详细介绍。概率比度量了启发式的 q_i 对 π_i 的了解程度，换句话说，对于状态 i，q 对于 pi 有多么知晓。因此，我们定义以下概念。

定义 4.2 *如果 $q_i > \pi_i$，则称状态 i 是过知的；如果 $q_i < \pi_i$，则称状态 i 是欠知的。*

下面定义了三种特殊状态。

定义 4.3 *如果 $q_i = \pi_i$，那么称状态 i 是确知的。如果它具有最高（或最低）比率 $w_i = q_i/\pi_i$，则称状态 i 是最高知的 (或最低知的): $i_{\max} = \arg\max_{i\in\Omega}\{w_i\}, i_{\min} = \arg\min_{i\in\Omega}\{w_i\}$.*

Liu [12] 注意到通过以信息性对状态进行升序排列，转移核能够以更简单的形式写出。因为对于 $i \neq j$，$\mathcal{K}_{ij} = q_j\min\{1, w_i/w_j\}$，如果 $w_1 \leq w_2 \leq \ldots \leq w_n$，那么

$$\mathcal{K}_{ij} = \begin{cases} w_i\pi_j & i < j, \\ 1 - \sum_{k<i} q_k - w_i\sum_{k>i}\pi_k & i = j, \\ q_j = w_j\pi_j & i > j。 \end{cases}$$

在不失一般性的情况下，可以假设状态被索引为 $w_1 \leq w_2 \leq \ldots \leq w_n$ 来考虑这种更易处理的转移核形式。

4.2.1 IMS 的特征结构

在过去的二十年中，大量工作致力于研究 IMS 的属性。不需要面面俱到，我们将简要回顾一些成果。对于有限状态空间，Diaconis、Hanlon [3] 和 Liu [12] 证明了 IMS 的更新和目标分布之间总变差距的各种上界。他们证明，马尔可夫链的收敛速度上界被一个依赖于第二大特征值的量约束：

$$\lambda_{slem} = 1 - \min_i\{\frac{q_i}{\pi_i}\}。$$

备注 4.3 在连续情况下，记 $\lambda^* = 1 - \inf_x\{\frac{q(x)}{p(x)}\}$，*Mengersen* 和 *Tweedie [14]* 证明如果 λ^* 严格小于 1，则链是均匀各态历经的，而如果 λ^* 等于 1，则收敛甚至不呈几何的。*Smith* 和 *Tierney [19]* 得到了类似的结果。这些结果表明，*IMS* 的马尔可夫链收敛速度受到最坏情况的影响。对于有限情况，与最小概率比 q_i/π_i 有关的状态决定收敛速率。也就是说，只有一个来自潜在巨大状态空间的状态决定马尔可夫链的收敛速度，这种状态甚至可能与 *MCMC* 的所有任务不相关！类似的情况发生在连续的空间中。

为了解释这种现象，考虑如下简单例子。

例 4.2 设 q 和 π 是两个具有相同方差的高斯分布，均值有略微偏移。则提议分布 q 将很好地近似目标 π。但是，很容易看出 $\inf_x\{q(x)/p(x)\} = 0$，因此 *IMS* 不会具有几何收敛速度。

这种令人沮丧的行为激发了人们对研究平均首中时作为衡量马尔可夫链“速度”的兴趣。在处理随机搜索算法时，重点可能是寻找单个状态而不是链的全局收敛，因此这一点尤其适用。例如，在计算机视觉问题中，人们经常搜索对场景最可能的解释，并且为此可以采用各种 Metropolis-Hastings 类型算法。有关示例和讨论，请参阅 Tu 和 Zhu 的工作 [20]。在这种背景下，我们感兴趣的是找到一些状态的首中时的行为，例如输入图像的场景的后验分布的众数。

4.2.2 有限空间的一般首中时

考虑有限空间 $\Omega = \{1,2\ldots n\}$ 上的各态历经马尔可夫链 $\{X_m\}_m$。设 $\mathcal{K}$ 是转移核，π 是唯一的平稳概率，q 是起始分布。对于每个状态 $i \in \Omega$，首中时 $\tau_{\text{hit}}(i)$ 已在 3.4节中定义。

对于任意 i，$\mathcal{K}_{-i}$ 表示 $\mathcal{K}$ 删除第 i 行和列得到的 $(n-1)\times(n-1)$ 矩阵，即 $\mathcal{K}_{-i}(k,j) = \mathcal{K}(k,j), \forall k \neq i,\ j \neq i$。设 $q_{-i} = (q_1,\ldots,q_{i-1},q_{i+1},\ldots,q_n)$，则 $P(\tau(i) > m) = q_{-i}\mathcal{K}_{-i}^{m-1}\mathbf{1}$，其中 $\mathbf{1} := (1,1,\ldots,1)'$。我们得到以下期望公式：

$$E_q[\tau(i)] = 1 + q_{-i}(\mathbf{I} - \mathcal{K}_{-i})^{-1}\mathbf{1}, \tag{4.2}$$

其中 $\mathbf{I}$ 表示单位矩阵。$\mathbf{I} - \mathcal{K}_{-i}$ 的逆的存在表明了 $\mathbf{I} - \mathcal{K}_{-i}$ 的亚随机性和 $\mathcal{K}$ 的不可约性（Bremaud[1]）。

更一般地说，Ω 的子集 A 的平均首中时（f.h.t）由下式给出

$$E_q[\tau(A)] = 1 + q_{-A}(\mathbf{I} - \mathcal{K}_{-A})^{-1}\mathbf{1}, \quad \forall A \subset \Omega。 \tag{4.3}$$

4.2.3 IMS 击中时间分析

这里，我们将充分利用之前的结果来计算 IMS 的平均首中时，并通过利用 IMS 核的特征结构为其提供界限。

Theorem 4.4 *(Maciuca and Zhu, 2006)* 根据具有提议 q 和目标概率 π 的 *IMS* 转移核，假设从 q 开始模拟马尔可夫链。然后，使用之前的符号：

i) $E[\tau(i)] = \dfrac{1}{\pi_i(1-\lambda_i)}, \forall i \in \Omega,$

ii) $\dfrac{1}{\min\{q_i,\pi_i\}} \leq E[\tau(i)] \leq \dfrac{1}{\min\{q_i,\pi_i\}}\dfrac{1}{1-\|\pi - q\|_{TV}},$

我们将 λ_n 定义为等于零，$\|\pi - q\|_{TV}$ 表示 π 和 q 之间的总变差距。对定义 *4.3*中的三个特殊状态，等号成立。

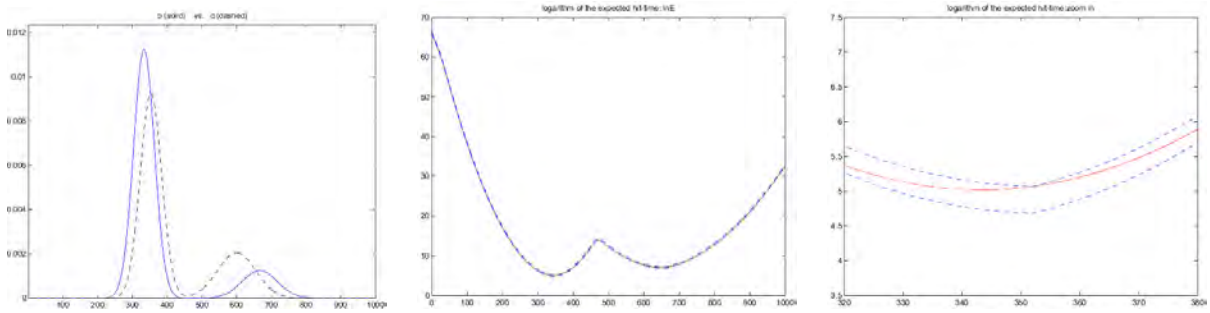
该定理的证明可参考 [13]。定理 4.4可以通过考虑某些特定集合的首中时来拓展。工作 [13] 证明以下推论成立。

推论 4.5 设 $A \subset \Omega$ 具有形式 $A = \{i+1, i+2, \ldots, i+k\}$，且 $w_1 \leq w_2 \leq \ldots \leq w_n$。记 $\pi_A := \pi_{i+1} + \pi_{i+2} + \ldots + \pi_{i+k}, q_A := q_{i+1} + q_{i+2} + \ldots + q_{i+k}, w_A := q_A/\pi_A$ 和 $\lambda_A := (q_{i+1} + \ldots + q_n) - (\pi_{i+1} + \ldots + \pi_n)w_A$。则定理 *4.4*的 *i* 和 *ii* 在将其中的 i 替换为 A 后也成立。

在本节引言中，我们提示为什么从 q 生成初始状态比从固定状态 $j \neq i$ 开始更好。下面的结果尝试说明这个问题。

命题 4.6 假设 $w_1 \leq w_2 \leq \ldots \leq w_n,$，则以下不等式成立：

$$E_1[\tau(i)] \geq E_2[\tau(i)] \geq \ldots \geq E_{i-1}[\tau(i)] \geq E_{i+1}[\tau(i)] = \ldots = E_n[\tau(i)] = E[\tau(i)], \forall i \in \Omega。$$



(a) π (实线) vs q (虚线)　(b) $\ln E[\tau(i)]$ and 边界　(c)(b) 放大后

图 4.4: 例 4.3的平均首中时和界限。©[2006] Springer。经许可重印，来自参考文献 [13]。

例 4.3 我们可以通过一个简单的例子来说明定理 *4.4*中的主要结果。考虑一个有 $n = 1000$ 个状态的空间。设 π 和 q 为两个离散高斯的混合，尾部被截断，然后归一化为一，如图 *4.4*（*a*）中实线（π）和虚线（q）所示。图 *4.4*显示了期望首中时的对数 $\ln E[\tau(i)]$。定理 *4.4*中的下界和上界相对于对数尺度绘制为虚线，其几乎与击中时间图重合。为了获得更好的分辨率，我们聚焦于众数周围的一部分图上，这三条曲线在图 *4.4*（*c*）中变得更加清晰可分。可以看到众数 $x^* = 333$ 有 $\pi(x^*) \approx 0.012$，且对 q 来说平均击中 $E[\tau_{x^*}] \approx 162$ 次。其远远小于穷举搜索的平均时间 $n/2 = 500$。相比之下，对于信息不足的（即均匀的）提议，结果是 $E[\tau_{x^*}] = 1000$。因此，可以看出“好”的提议 q 如何影响这种随机抽样的速度。

Theorem 4.7 *(Maciuca and Zhu, 2006)* 设 p 和 Q 分别为 *Metropolis-Hasting* 采样器的目标概率和提议矩阵。设 $M = \max_{ij} Q_{ij}/p_j$，$m = \min_{ij} Q_{ij}/p_j$，$m > 0$。对于任意初始分布 q，期望的首中时具有以下边界

$$p_i + \frac{1-q_i}{M} \leq p_i E_q^Q[\tau(i)] \leq p_i + \frac{1-q_i}{m}, \forall i.$$

如果 $Q_{ij} = p_j, \forall i, j$，等号成立。

该定理的证明在 [13] 中。

4.3 可逆跳跃和跨维 MCMC


Ulf Grenander

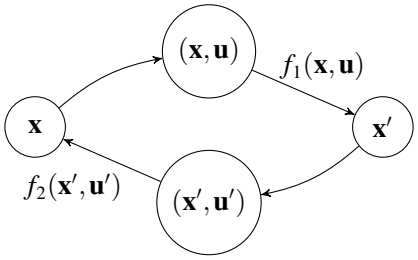
许多情况下，我们可能需要对在不同维度空间的并集上定义的后验概率进行采样。例如，我们可以为图像中的物体定义参数量可变的贝叶斯模型，并且可能对从这些模型中采样以估计给定图像最可能的观察感兴趣。

这个问题首先由 Grenander 和 Miller 1994[8] 提出进行图像分析，Green 1995[7] 提出进行贝叶斯模型选择。

4.3.1 可逆跳跃


Michael Miller

设 $\Omega = \cup_{i=1}^{\infty} \Omega_i$ 是一个解空间，它可以写为不同维度的子空间并集，$\dim(\Omega_i) = d_i$，且 π 是在 Ω 上定义的概率分布。可逆跳跃是从一个空间 Ω_i 中的状态到另一个 Ω_j 的 MCMC 移动，其满足关于 π 的细致平衡方程。



一般情况. 实现从 $\mathbf{x} \in \Omega_i$ 到 $\mathbf{x}' \in \Omega_j$ 的可逆跳跃移动 $q(\mathbf{x} \to \mathbf{x}')$，首先从概率 $q(j|i,\mathbf{x})$ 中采样 j，从概率密度函数 $q(\mathbf{u}|\mathbf{x})$ 中采样一个辅助向量 $\mathbf{u} \in \mathbb{R}^m$（对于某个维度 m 需要指定）采样，然后通过确定性函数 $\mathbf{x}' = f_1(\mathbf{x},\mathbf{u})$ 得到 $\mathbf{x}'$。反向移动 $q(\mathbf{x}' \to \mathbf{x})$ 可以以类似的方式定义，从概率 $q(i|j,\mathbf{x}')$ 采样 i 和从 pdf $q(\mathbf{u}'|\mathbf{x}')$ 采样辅助向量 $\mathbf{u}' \in \mathbb{R}^{m'}$。必有一个双射 $f : \Omega_i \times \mathbb{R}^m \to \Omega_j \times \mathbb{R}^{m'}$ 使得 $f(\mathbf{x},\mathbf{u}) = (\mathbf{x}',\mathbf{u}')$。因此，必须满足维度匹配条件 $d_i + m = d_j + m'$ 以及 $\dfrac{d\mathbf{x}'d\mathbf{u}'}{d\mathbf{x}d\mathbf{u}} = \dfrac{\partial f(\mathbf{x},\mathbf{u})}{\partial(\mathbf{x},\mathbf{u})}$。为了满足细致平衡，提议移动 $q(\mathbf{x} \to \mathbf{x}')$ 以以下概率被接受，

$$\alpha(\mathbf{x} \to \mathbf{x}') = \min\left(1, \frac{q(i|j,\mathbf{x}')q(\mathbf{u}'|\mathbf{x}')\pi(\mathbf{x}')}{q(j|i,\mathbf{x})q(\mathbf{u}|\mathbf{x})\pi(\mathbf{x})}\left|\det\frac{\partial f(\mathbf{x},\mathbf{u})}{\partial(\mathbf{x},\mathbf{u})}\right|\right)。 \tag{4.4}$$

膨胀收缩. 可逆跳跃的一个特例是膨胀-收缩移动，其中 $\Omega_j = \Omega_i \times Z$。从 $\mathbf{x} \in \Omega_i$ 开始，可以选择 $\mathbf{u} \in Z$ 且 f 作为恒等函数，从而得到膨胀移动 $q(\mathbf{x} \to \mathbf{x}') = (\mathbf{x},\mathbf{u})$。从 $\mathbf{x}' = (\mathbf{x},\mathbf{u}) \in \Omega_j$ 开始，收缩移动只会降低 $\mathbf{u}$，因此 $q(\mathbf{x}' \to \mathbf{x}) = \mathbf{x}$。膨胀移动的接受概率为

$$\alpha(\mathbf{x} \to \mathbf{x}') = \min\left(1, \frac{\pi(\mathbf{x}')}{\pi(\mathbf{x})q(\mathbf{u}|\mathbf{x})}\right), \tag{4.5}$$

对于收缩移动则是

$$\alpha(\mathbf{x}' \to \mathbf{x}) = \min\left(1, \frac{\pi(\mathbf{x})q(\mathbf{u}|\mathbf{x})}{\pi(\mathbf{x}')}\right)。 \tag{4.6}$$

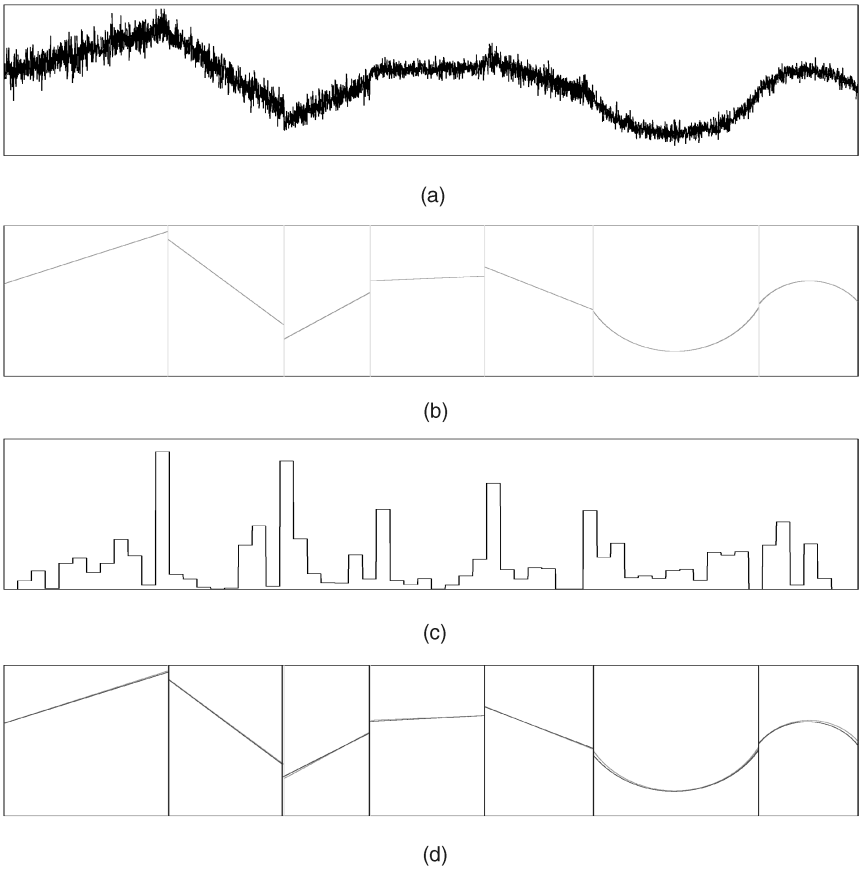


图 4.5: （a）1 维范围图像 $I(x), x \in [0,1]$ 。（b）真实分割，W_{th}。（c）边缘度量 $b(x), x \in [0,1]$。较大值的 $b(x)$ 表示 x 是变化点的概率很高。（d）相对于 W_{th}（浅灰色），算法找到的最佳解 W^*（深灰色）。©[2004] IEEE。经许可重印，来自参考文献 [9]。

4.3.2 简单示例：1 维范围图像分割

图 4.5显示了一个模拟 1 维图像 $I(x), x \in [0,1]$ 的例子。它是通过将高斯噪声 $N(0,\sigma^2)$ 加到图 2b 中的原始曲面 I_o 而生成的。I_o 由未知数量的 k 个曲面组成，可以是直线或圆弧，由 $k-1$ 个变化点分隔，

$$0 = x_0 < x_1 < ... < x_k = 1。$$

设 $l_i \in \{,\}$ 索引区间 $[x_{i-1}, x_i)$ 上的曲面类型，具有参数 $\theta_i, i = 1,...k$。对于一条直线，$\theta = (s,\rho)$ 表示斜率 s 和截距 ρ。对于一条圆弧，$\theta = (u,v,R)$ 表示圆心 (u,v) 和半径 R。因此，1 维“世界场景”由随机变量的向量表示，

$$W = (k, \{x_i, i = 1,...,k-1\}, \{(l_i, \theta_i), i = 1,...,k\})。$$

曲面 I_o 完全由 W 决定，有 $I_o(x) = I_o(x, l_i, \theta_i), x \in [x_{i-1}, x_i), i = 1,...,k$。

通过标准贝叶斯公式，我们得到后验概率

$$p(W|I) \propto \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{k}\int_{x_{i-1}}^{x_i}(I(x)-I_o(x,l_i,\theta_i))^2dx\right\}\cdot p(k)\prod_{i=1}^{k}p(l_i)p(\theta_i|l_i)。 \tag{4.7}$$

以上第一个因子是似然，其余的是先验概率 $p(k)\propto\exp(-\lambda_0 k)$ 和 $p(\theta_i|l_i)\propto\exp(-\lambda\#\theta_i$ ，其惩罚参数的数量 $\#\theta_i$。$p(l_i)$ 是线和弧上的均匀概率。因此，能量函数由下式定义

$$E(W)=\frac{1}{2\sigma^2}\sum_{i=1}^{k}\int_{x_{i-1}}^{x_i}(I(x)-I_o(x,l_i,\theta_i))^2dx+\lambda_o k+\lambda\sum_{i=1}^{k}\#\theta_i。 \tag{4.8}$$

与此相关一个问题是 W 没有固定的维度。因此概率 $p(W|I)$（或能量 $E(W)$）分布在可变维度的可数的子空间上。下一小节简要介绍了探索这种解空间的跳跃扩散过程。

跳跃-扩散


Peter Green

考虑一个解空间 $\Omega=\cup_{n=1}^{\infty}\Omega_n$，其中子空间索引 $n=(k,l_1,...,l_k)$ 包含模型的离散变量。为了穿越解空间，算法需要两种类型的移动：不同子空间之间的可逆跳跃和每个连续子空间内的随机扩散。

1. 可逆跳跃。设 $W=(i,\mathbf{x})$, 是马尔可夫链在 t 时间的状态，其中 $\mathbf{x}\in\Omega_i$ 表示解的连续变量。在无穷小的时间间隔 dt 中，马尔可夫链跳跃到另一个子空间 $\Omega_j, j\neq i$ 中的新状态 $W'=(j,\mathbf{x}')$。有三种类型的跳跃：1）从直线切换到圆弧或反之，2）两个相邻的区间合并为直线或圆，以及 3）一个区间分成两个区间（线或圆）。该跳跃通过一个 Metropolis 移动 [15] 来实现，它提出通过一个前向提议概率 $q(W'|W)=q(i\to j)q(\mathbf{x}'|j)$ 从 W 移动到 W'。反向提议概率是 $q(W|W')=q(j\to i)q(\mathbf{x}|i)$。前向提议以以下概率被接受，

$$\alpha(W\to W')=\min\left(1,\frac{q(j\to i)q(\mathbf{x}|i)\pi(W')}{q(i\to j)q(\mathbf{x}'|j)\pi(W)}\right)。 \tag{4.9}$$

在上述概率比中维度是匹配的。

2. 随机扩散. 在每个子空间 Ω_n 中，$n=(k,l_1,...,l_k)$ 固定，能量函数 $E(\mathbf{x})$ 为

$$E(\mathbf{x})=E(x_1,...,x_{k-1},\theta_1,...,\theta_k)=\frac{1}{2\sigma^2}\sum_{i=1}^{k}\int_{x_{i-1}}^{x_i}(I(x)-I_o(x,l_i,\theta_i))^2dx+const。$$

我们采用随机扩散（或 Langevin）方程来探索子空间。Langevin 方程是由温度为 T 的布朗运动 $dB(t)$ 驱动的最陡下降 PDE（偏微分方程）。设 $\mathbf{x}(t)$ 表示 t 时的变量，则

$$d\mathbf{x}(t)=-\frac{dE(\mathbf{x})}{d\mathbf{x}}dt+\sqrt{2T(t)}dw_t,\ dw_t\sim N(0,(dt)^2)。 \tag{4.10}$$

例如，变化点 x_i 的运动方程为

$$\frac{dx_i(t)}{dt}=\frac{1}{2\sigma^2}[(I(x)-I_o(x,l_{i-1},\theta_{i-1}))^2-(I(x)-I_o(x,l_i,\theta_i))^2]+\sqrt{2T(t)}N(0,1)。$$

这是区域竞争方程的一维版本 [23]。点 x_i 的移动是由数据 $I(x_i)$ 对两个相邻区间的曲面模型的适应性以及布朗运动驱动的。在实践中，布朗运动有助于避免局部陷阱。

为了计算参数 $\theta_i, i = 1, ..., k$，运行扩散比为每个区间 $[x_{i-1}, x_i)$ 确定性地拟合最佳 θ_i 更加健壮、更加快速，因为确定性拟合是“过度承诺”。当前区间包含多个对象时更是如此。

众所周知 [6]，式子(4.10)中的连续 Langevin 方程模拟具有平稳密度 $p(\mathbf{x}) \propto \exp(-E(\mathbf{x})/T)$ 的马尔可夫链。这是在温度 T 处子空间 Ω_n 内的后验概率。

3. 跳跃和扩散的协调. 作为泊松事件的时间实例 $t_1 < t_2 < \ldots < t_M \ldots$ 处的跳跃中断连续扩散。在实践中，扩散总是以离散的时间步长 δt 运行。因此，两次连续跳跃之间的离散等待时间 τ_j 是

$$w = \frac{t_{j+1} - t_j}{\delta_t} \sim p(w) = e^{-\tau} \frac{\tau^w}{w!},$$

其中期望等待时间 $E(w) = \tau$ 控制跳跃的频率。跳跃和扩散过程都应遵循退火方案逐步降低温度。

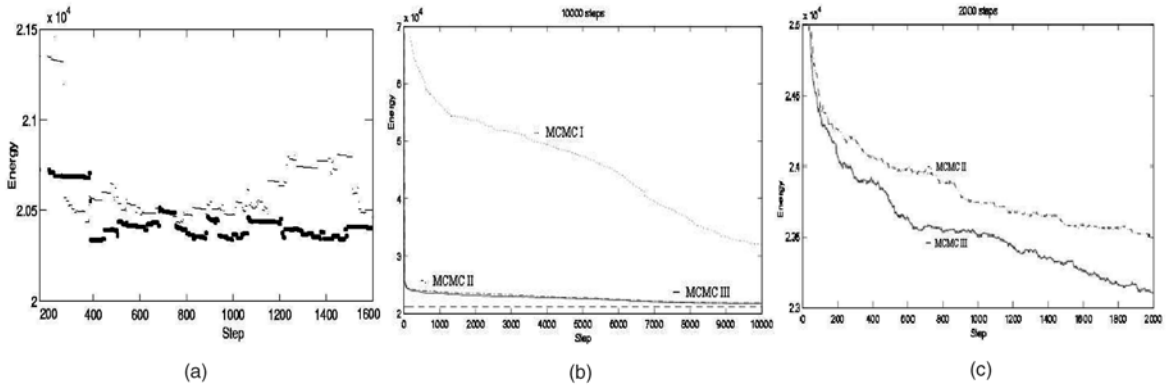


图 4.6: （a）跳跃扩散。两个试验（细的 MCMC II 和粗的 MCMC III）的跳跃扩散过程的能量图。扩散中的连续能量变化被能量跳跃中断。（b）平均能量图。三个马尔可夫链 MCMC I，II 和 III 的前 10,000 步中能量曲线的比较，100 个随机生成的信号的平均。（c）放大观察 MCMC II 和 III 的前 2000 步。注意，能量尺度与（b）不同。©[2004] IEEE。经许可重印，来自参考文献 [9]。

为了解释，图 4.6（a）显示了在图 14.5（a）中输入的 1 维范围数据上运行的跳跃扩散过程的两次试验。能量图上下变化（即算法不贪婪），连续能量曲线（扩散）被跳跃中断。

4. 可逆性和全局优化. 从工程角度看，跳跃扩散过程最重要的特性是它模拟穿越复杂解空间的马尔可夫链。这个属性将其与贪婪和局部方法区分开来。理论上，这个马尔可夫链从解空间 Ω 上的后验概率 $p(W|I)$ 采样 [8]。利用退火方案，理论上可以实现概率接近 1 的全局最优解。跳跃的可逆性可能不是必要条件；然而，它是在复杂解空间中实现马尔可夫链不可约的有效用工具。

5. 速度瓶颈. 传统的跳跃扩散设计受到其计算速度的限制。但是，通过设计更好的提议概率可以克服这个问题，这将在下一节中说明。我们观察到跳跃的瓶颈受到提议概率设计的影响。在式(4.9)中，区间 $[x_{i-1}, x_i)$ 中的提议概率 $q(\mathbf{x}'|j)$ 可分为三种情况：1）切换到 $\mathbf{x}' = \theta_i$ 的新模型，2）合并以形成新的类型为 l、参数 $\mathbf{x}$ 的区间 $[x_{i-2}, x_i)$，3）分开形成两个新的区间，分别对应模型 (l_a, θ_a) 和 (l_b, θ_b)。

$$q(\mathbf{x}|m)=\begin{cases} q(\theta_i|l_i,[x_{i-1},x_i)) & \text{把 } [x_{i-1},x_i) \text{ 转换到模型 } (l_i,\theta_i) \\ q(\theta|l,[x_{i-2},x_i)) & \text{合并到模型 } (l,\theta) \\ q(x|[x_{i-1},x_i))q(\theta_a|l_a,[x_{i-1},x))q(\theta_b|l_b,[x,x_i)) & \text{在 } x \text{ 处将 } [x_{i-1},x_i) \text{ 分为 } (l_a,\theta_a) \text{ and } (l_b,\theta_b)。 \end{cases}$$

4.4 应用：计算人数

文献 [5] 展示了将 Metropolis-Hastings 算法用于检测和统计拥挤场景中的人数。该问题表达在标记点过程框架下，其中每个人被表示为一个标记点 s，包含一个表示图像位置的空间过程 $p\in\mathbb{R}^2$ 和表示人的宽度、高度、方向和形状的标记过程 $m=(w,h,\theta,j)$。这些一起形成了标记点 $s=(p,(w,h,\theta,j))$。

4.4.1 标记点过程模型

该模型假设标记点依赖于标记过程的空间位置，因此对于每个标记点 $s=(p,(w,h,\theta,j))$，

$$\pi(s)=\pi(p)\pi(w,h,\theta,j|p)$$

。

点过程的先验 $\pi(p)$ 是一个齐次泊松点过程，即点的总数遵循泊松分布，且给定点数，它们在该区域内的位置是均匀的。先验模型的模拟如图 4.7所示。

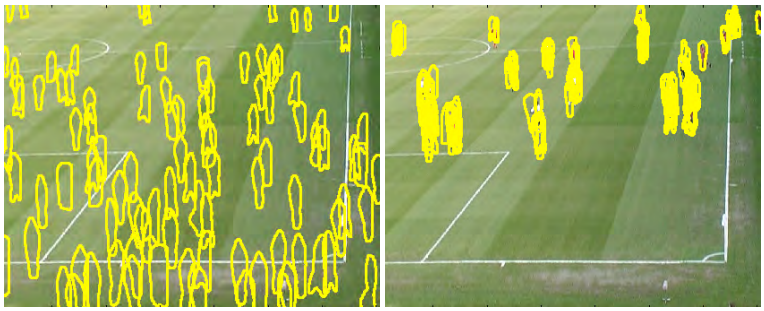


图 4.7: 来自泊松点过程先验 $\pi(s)$ 的样本。©[2009] IEEE。经许可重印，来自参考文献 [5]。

条件标记过程 $pi(w,h,\,theta,j|p)$ 用独立高斯表达宽度、高度和方向，其依赖于图像位置 p，并均匀分布表达来自一组可能的形状中的形状 j。空间相关的均值和方差存储为查找表。可能的形状集合通过 Bernoulli 模板混合模型的期望最大化从一组手动分割的边界框学习。

处理输入图像得到前景掩模数据 y，其中如果像素 i 是前景像素则 $y_i=1$，如果是背景则 $y_i=0$。给定当前点结构 $s_1,\ldots,s_n$，构造标签图像，其中如果对应于 n 个标记点的任何形状覆盖它，则将像素标记为前景，否则为背景。现实中，掩模和标签图像都有软标签，且包含区间 $[0,1]$ 中的值。

似然为

$$\log\mathcal{L}(Y|X)=\sum(x_i\log y_i+(1-x_i)\log(1-y_i))。$$

4.4.2 MCMC 推理

给定具有前景掩模 y 的输入图像，通过最大后验概率（MAP）估计可获得最可能的标记点结构，即后验概率函数 $\pi(s|y) = \pi(y|x(s))\pi(s)$ 的最大化。

这可以通过三种类型的可逆移动来实现：

- 出生提议。根据前景掩模在均匀位置处提议标记点。宽度、高度和方向从相应的以该点为条件的高斯分布中采样。形状的类型从学习到的形状原型集合中随机均匀地选择。这种模式的逆向就是死亡提议。
- 死亡提议。随机的一点以与出生提议相反的方式删除。
- 更新提议。随机选择一个标记点，并修改其位置或标记参数。该位置被修改为随机行走。修改标记通过选择三个参数中的一个、并从给定当前位置的条件分布中对其进行采样实现，或者从可能的形状中随机选择形状类型来实现。

这三种类型的移动分别使用概率 0.4,0.2,0.4。从一个空结构开始，一张图像需要大约 500-3000 次移动。越是拥挤的场景需要更多移动。

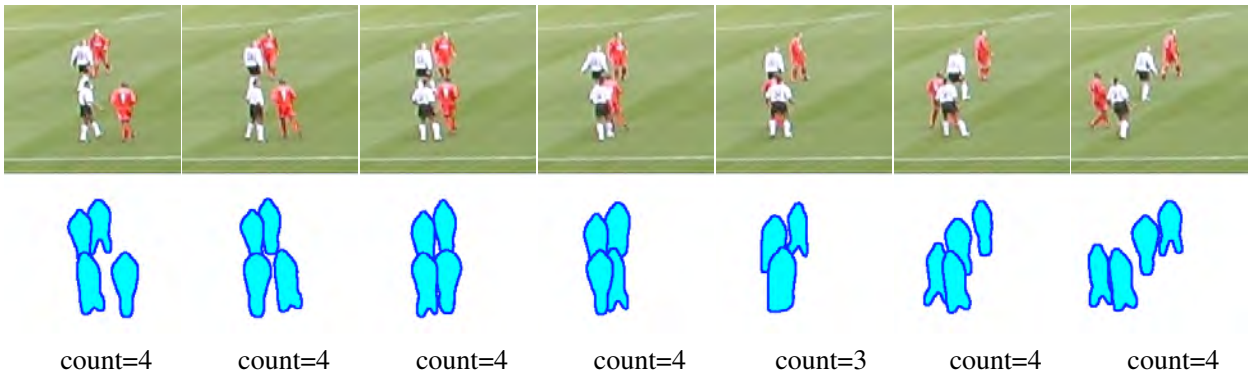


图 4.8: VSPETS 序列的七个帧上的结果。计数是精确的，直到有明显的重叠。©[2009] IEEE。经许可重印，来自参考文献 [5]。

图 4.9: CAVIAR 数据集图像上的结果。©[2009] IEEE。经许可重印，来自参考文献 [5]。

4.4.3 结果

MCMC 方法在两个具有真值标注的基准序列上进行了测试：EU CAVIAR 数据集[1]和 VSPETS 足球序列[2]。结果如图 4.8和 4.9所示。

4.5 应用：家具布置

Metropolis-Hastings 算法的一个应用是家具布置 [21]，如图 4.10所示。该过程包括两个阶段：（1）从正样本中提取空间、层次和成对关系，以及（2）通过优化合成新的家具布置。

图 4.10: 左：家具布局的任意初始化。中间和右侧：两个依据人类工效学标准（例如无障碍的可达性和可见性）进行优化合成的家具布置。©[2011] ACM。经许可重印，来自参考文献 [21]。

1. 物体表达. 将家具优化为现实的、功能的配置依赖于对各种交互因素的建模，例如成对家具关系、与房间相关的空间关系以及其他人为因素。

边界面: 场景中的每个物体都由一组边界面表示。除了顶面和底面之外，每个物体都有一个“背”面，这是最靠近墙壁的面。其他面标记为“非背”面。背面用作定义用于分配其他属性的参考平面。

中心和方向: 一个物体的关键属性是中心和方向，分别用 (p_i, θ_i) 表示，其中 p_i 表示 (x, y) 坐标、θ_i 表示相对于最近墙的角度（定义为最近墙和背面之间的角度）。

可达空间: 对于物体的每个表面，分配相应的可达空间。我们将 a_{ik} 定义为物体 i 的可达空间 k 的坐标中心。该区域的对角线由 ad_{ik} 度量，用于测量其他物体在优化期间能够渗透到该空间的深度。可达空间的大小由可用样本设置或作为与人体大小相关的输入给出。如果在所有样本中空间都非常靠近墙壁，则对应的表面不需要是可达的; 否则，如果没有给出这样的测量, 则将其设置为平均大小的成年人的尺寸。

视锥：对于某些物体，例如电视和绘画，正面必须是可见的。视锥被分配给该特定表面，对于一个物体 i，其由一系列具有中心坐标 v_{ik} 的矩形近似得到，其中 k 是矩形索引。vd_{ik} 是矩形的对角线，其在定义类似于可达空间的渗透损失方面十分有用。

其他属性: 优化过程中涉及的其他属性有从 p_i 到最近墙的距离，定义为 d_i，对角线 b_i，从 p_i 到边界框的角点。同时还有 z 位置，物体的 z_i。

2. 损失函数. 优化过程的目标是最小化刻画现实的、功能性的家具布置的损失函数。虽然通常很难量化家具布置的“现实性”或“功能性”，但不应违反以下基本准则。

可达性: 为了能实现功能，一个家具物体必须是可达的 [2, 17]。为了支持可达性，只要任何物体移动到另一个物体的可达空间中，损失就会增加。假设物体 i 与物体 j 的可达空间 k 重叠，则可达性损失定义

[1] http://homepages.inf.ed.ac.uk/rbf/CAVIAR/

[2] http://www.cvg.cs.rdg.ac.uk/VSPETS/vspets-db.html

为

$$C_{\mathrm{a}}(\phi) = \sum_i \sum_j \sum_k \max\left[0, 1 - \frac{\|p_i - a_{jk}\|}{b_i + ad_{jk}}\right]。 \tag{4.11}$$

还有其他损失项，如可见性、连接门的路径、某些家具物体（例如面向沙发的电视）之间的成对约束、激励训练样本中所见的结构的一个先验等。

3. 家具布置优化. 由于物体在优化过程中是相互依赖的，因此该问题的搜索空间非常复杂。家具的位置和方向依赖于许多因素，例如物体是否应该是可见的或可达的。很难获得一个全局优化方案或可以产生唯一优化的封闭形式的解。

为了处理这个问题，采用具有 Metropolis-Hastings 状态搜索步骤 [10, 15] 的模拟退火 [11] 来搜索全局最优的较好近似值。但请注意，给定一个房间，一组家具物体以及先验空间和层次关系，可以实现许多可接受的好的配置。这是在合理的短时间内找到较好近似的基本原理，而不是为了找到损失函数的全局最优在复杂的搜索空间上进行穷尽搜索。

为了有效地探索可能布置的空间，提议移动 $\phi \to \phi'$ 涉及修改当前布置的局部调整，以及交换物体的全局重新配置步骤，从而显著改变布置。有三种类型的移动：平移和旋转，交换物体和移动路径控制点。更多细节请参考 [21]。

Synthesis 1　　Synthesis 2　　Synthesis 3

图 4.11: 选择的合成结果的视图。从上到下：画廊，度假村，餐厅。©[2011] ACM。经许可重印，来自参考文献 [21]。

通过上述移动，给定一个平面图和定义解空间的家具物体的固定数量，家具物体 (p_i, θ_i) 的配置有可能移动到任何其他配置 (p_i', θ_i')。给定退火方案，在优化的早期阶段通过较大的移动更广泛地探索解空间，通过较小的移动最终精确调整家具配置。

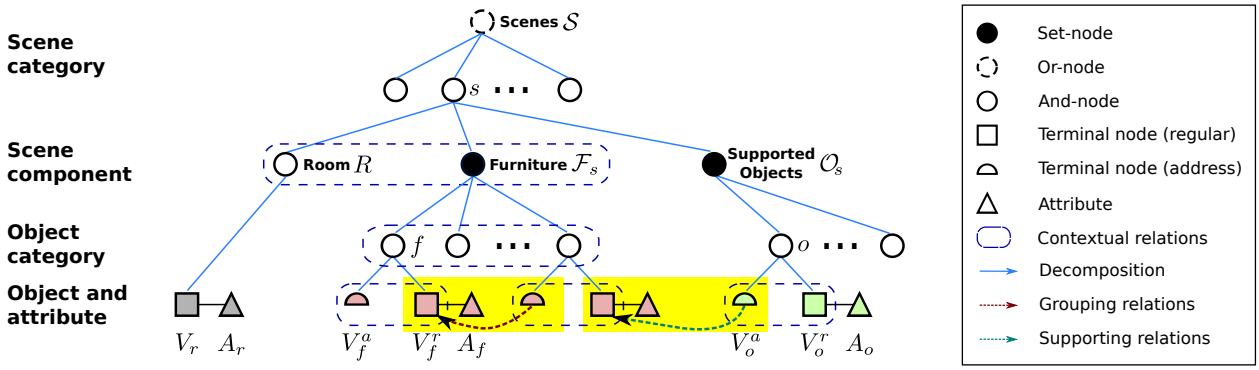


图 4.12: 场景语法的属性空间与或图。©[2018] IEEE。经许可重印，来自参考文献 [18]。

4.6 应用：场景合成

Metropolis-Hastings 的另一个应用是以人为中心的场景合成 [18]。这里，一个属性空间与或图（S-AOG）用于表示室内场景图，如图 4.12所示。模型分布从室内场景数据集中学习，新的布局可以使用马尔可夫链蒙特卡罗采样得到。

室内场景表达. 属性 S-AOG [22] 被用于表示室内场景。属性 S-AOG 是在终端节点上具有属性的概率语法模型。该属性结合了 i）概率上下文无关语法（PCFG），以及 ii）在马尔可夫随机场（MRF）上定义的上下文关系，即节点之间的水平连接。PCFG 表示由一组终端和非终端节点从场景（顶层）到物体（底层）的层次分解，而上下文关系通过水平连接刻画了空间和功能关系。S-AOG 的结构如图 **??**所示。

形式上，一个 S-AOG 被定义为一个 5 元组：$\mathcal{G} = \langle S, V, R, P, E \rangle$，其中 S 是场景语法的根节点，V 是顶点集合，R 是产生式规则，P 是定义在属性 S-AOG 上的概率模型，E 包含表达为同一层节点之间水平连接的上下文关系。[3]

顶点集 V 可以分解为一组有限的非终端节点和终端节点：$V = V_{NT} \cup V_T$。

- $V_{NT} = V^{And} \cup V^{Or} \cup V^{Set}$。非终端节点由三个子集组成。i）一组**与节点** V^{And}，其中每个节点表示将较大实体（例如卧室）分解为较小的组件（例如，墙壁，家具和支撑物体）。ii）一组**或节点** V^{Or}，其中每个节点分支到可选分解（例如，室内场景可以是卧室或起居室），使算法能够重新构造场景。iii）一组**集节点** V^{Set}，其中每个节点表示一个嵌套的与或关系：一组用作子分支的或节点被一个与节点组合，并且每个子分支可以包括不同数量的物体。

- $V_T = V_T^r \cup V_T^a$。终端节点由两个节点子集组成：常规节点和地址节点。i）一个 **常规终端节点** $v \in V_T^r$ 表示具有属性的场景（例如，卧室中的办公椅）中的空间实体。这里，属性包括物体尺寸 (w, l, h) 的内部属性 A_{int}，物体位置 (x, y, z) 和方向（$x-y$ plane）θ 的外部属性 A_{ext}，以及采样的人体位置 A_h。ii）为了避免图的过度密集，引入了**地址终端节点** $v \in V_T^a$ 来描述仅在特定语境存在但在所有其他语境中不存在的交互。它是指向常规终端节点的指针，从集合 $V_T^r \cup \{\text{nil}\}$ 中取值，表示支持或分组关系，如图 4.12所示。

上下文关系. 节点之间的上下文关系 E 由在终端节点上形成 MRF 的 S-AOG 中的水平连接表示。为了编码上下文关系，为不同的团定义了不同类型的势函数。上下文关系 $E = E_f \cup E_o \cup E_g \cup E_r$ 分为四个子

[3]我们使用术语“顶点”而不是“符号”（在传统的 PCFG 定义中）以便与图模型中的符号一致。

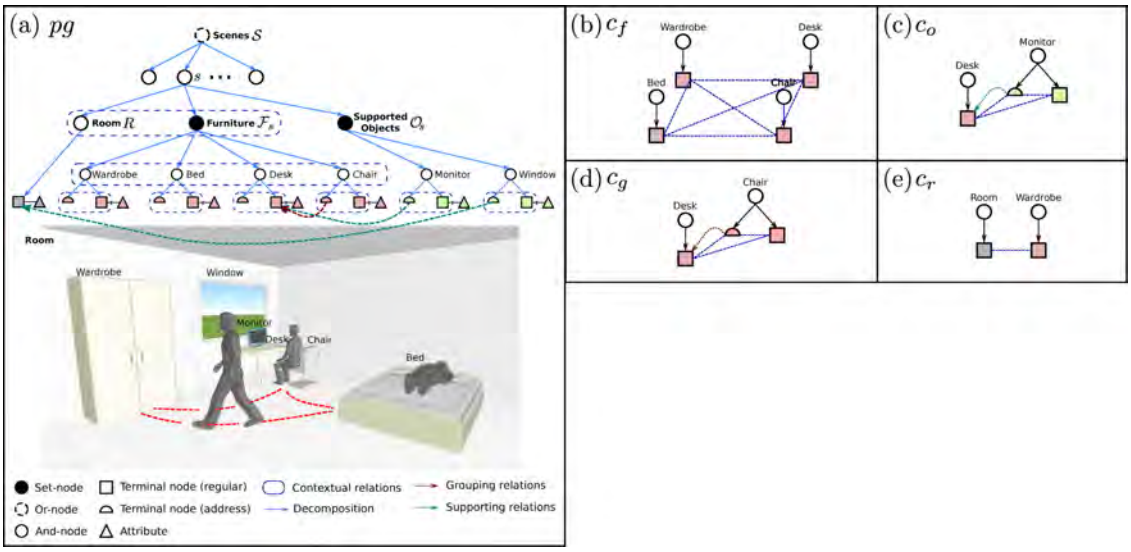


图 4.13: （a）卧室解析图的简化示例。解析图的终端节点在终端层中形成 MRF。团由投射到终端层的环境关系形成。（b）-（e）中显示了四种类型的团的示例，表示四种不同类型的环境关系。来自 [18]。

集：i）家具之间的关系 E_f; ii）被支撑物体与其支撑物体之间的关系 E_o（例如，桌面上的监视器）; iii）功能对 E_g（例如，椅子和桌子）的物体之间的关系; 和 iv）家具和房间之间的关系 E_r。因此，在终端层中形成的团也可以分成四个子集：$C = C_f \cup C_o \cup C_g \cup C_r$。使用动允性 (affordance) 作为表征物体-人-物体关系的桥梁来计算势，而不是直接捕捉物体-物体关系。

分层解析树 pt 是 S-AOG 的一个实现，通过为与节点选择子节点、以及确定集节点的每个子节点的状态来实现。解析图 pg 由解析树 pt 和解析树上的许多上下文关系 E 组成：$pg = (pt, E_{pt})$。图 4.13显示了一个解析图的简单例子和终端层中形成的四种团。

S-AOG 的概率公式化。一个场景构造由解析图 pg 表示，包括场景中的物体和相关属性。由 Θ 参数化的、S-AOG 生成的 pg 的先验概率为一个 Gibbs 分布

$$p(pg|\Theta) = \frac{1}{Z}\exp\{-\mathscr{E}(pg|\Theta)\} = \frac{1}{Z}\exp\{-\mathscr{E}(pt|\Theta) - \mathscr{E}(E_{pt}|\Theta)\}, \tag{4.12}$$

其中 $\mathscr{E}(pg|\Theta)$ 是解析图的能量函数，$\mathscr{E}(pt|\Theta)$ 是解析树的能量函数，$\mathscr{E}(E_{pt}|\Theta)$ 是上下文关系的能量项。

$\mathscr{E}(pt|\Theta)$ 可以进一步分解为不同类型的非终端节点的能量函数，以及常规终端节点和地址终端节点的内部属性的能量函数：

$$\mathscr{E}(pt|\Theta) = \underbrace{\sum_{v\in V^{Or}} \mathscr{E}_{\Theta}^{Or}(v) + \sum_{v\in V^{Set}} \mathscr{E}_{\Theta}^{Set}(v)}_{\text{非终端节点}} + \underbrace{\sum_{v\in V_T^f} \mathscr{E}_{\Theta}^{A_{in}}(v)}_{\text{终端节点}}, \tag{4.13}$$

其中，或节点 $v \in V^{Or}$ 的子节点的选择和集节点 $v \in V^{Set}$ 的子分支遵循不同的多项分布。由于与节点是确定性扩展的，因此在这里没有与节点的能量项。终端节点的内部属性 A_{in}（大小）遵循由核密度估计学习的非参数概率分布。

图 4.14: MCMC 采样的例子。从左到右: 模拟退火过程中得到的场景构造。©[2018] IEEE。经许可重印，来自参考文献 [18]。

$\mathscr{E}(E_{pt}|\Theta)$ 结合终端层中形成的四种团的势，整合人体属性和常规终端节点的外部属性：

$$p(E_{pt}|\Theta) = \frac{1}{Z}\exp\{-\mathscr{E}(E_{pt}|\Theta)\} = \prod_{c\in C_f}\phi_f(c)\prod_{c\in C_o}\phi_o(c)\prod_{c\in C_g}\phi_g(c)\prod_{c\in C_r}\phi_r(c)。 \tag{4.14}$$

合成场景构造. 合成场景构造是通过从 S-AOG 定义的先验概率 $p(pg|\Theta)$ 中采样解析图 pg 来完成的。解析树 pt 的结构（即或节点和集节点子分支的选择）和物体的内部属性（大小）可以容易地从封闭形式的分布或非参数分布中采样得到。然而物体的外部属性（位置和方向）受到多个势函数的约束，因此它们太复杂而无法直接采样。这里，马尔可夫链蒙特卡罗（MCMC）采样器用于提取分布中的典型状态。每次采样过程可分为两个主要步骤：

1. 直接采样 pt 的结构和内部属性 A_{in}:（i）为或节点采样子节点;（ii）确定集节点的每个子分支的状态;（iii）对于每个常规终端节点，从学习的分布中对大小和人体的位置进行采样。

2. 使用 MCMC 方案通过提议移动来采样地址节点 V^a 和外部属性 A_{ex} 的值。马尔可夫链收敛后将选择一个样本。

以概率随机使用两种简单类型的马尔可夫链动态过程 $q_i, i = 1,2$，以做出提议移动：

- 动态过程 q_1: 物体平移。此动态过程选择常规终端节点，并基于当前位置 x 对新位置进行采样: $x \to x+\delta x$，其中 δx 遵循二元正态分布。
- 动态过程 q_2: 物体旋转。此动态选择常规终端节点，并基于物体的当前方向对新方向进行采样：$\theta \to \theta+\delta\theta$，其中 $\delta\theta$ 遵循正态分布。

采用 Metropolis-Hastings 算法，新解析图 pg' 根据以下接受概率被接受:

$$\alpha(pg'|pg,\Theta) = \min(1, \frac{p(pg'|\Theta)p(pg|pg')}{p(pg|\Theta)p(pg'|pg)}) = \min(1, \exp(\mathscr{E}(pg|\Theta) - \mathscr{E}(pg'|\Theta))), \tag{4.15}$$

其中提议概率比被取消，因为提议移动在概率上是对称的。采用模拟退火方案得到高概率样本。该过程如图 4.14所示。图 4.15显示了高概率样本。

4.7 练习

问题 1. 考虑一个具有 $n = 1000$ 个状态 $1,2,...,1000$ 的空间，如例 4.3所示。概率 π 是一个混合高斯，其均值分别为 330 和 670，标准差分别为 40 和 50，权重分别为 0.9 和 0.1。提议概率 q 也是一个混合高斯，其均值分别为 350 和 600, 标准差分别为 40 和 50, 权重分别为 0.75 和 0.25。两个概率都离散化为状态 $1,...,1000$ 且归一化到 1。

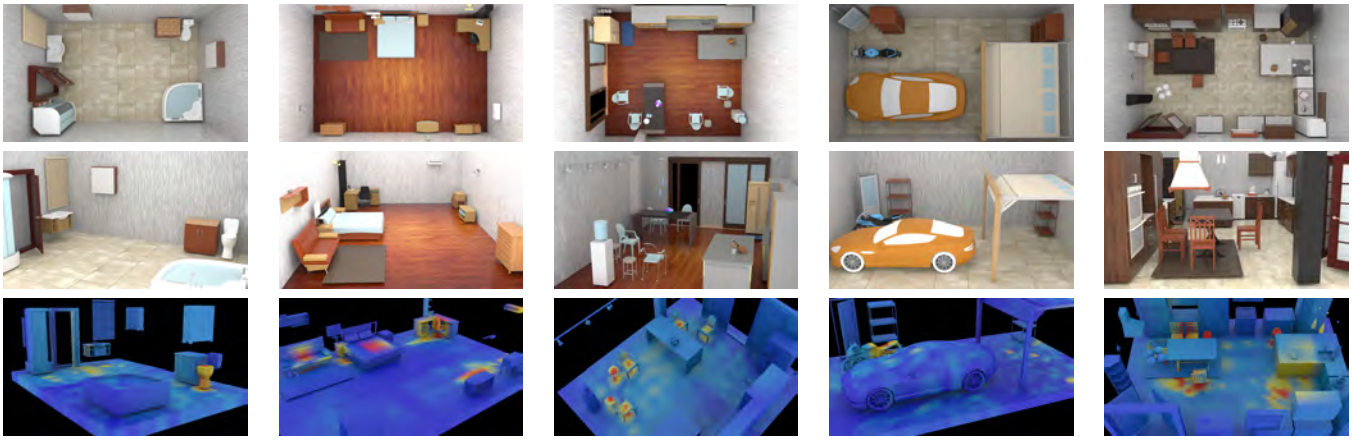


图 4.15: 五种不同类别的场景示例。上：顶视图。中：侧视图。下图：动允性热量图。©[2018] IEEE。经许可重印，来自参考文献 [18]。

a) 对于每一个 i，估计利用 100 马尔科夫链所有状态的首中时 $E[\tau(i)]$。

b) 计算所有状态的首中时，并将他们和估计的时间以及来自定理 4.4的边界画在一起。

问题 2. 假设我们有一个 1D 范围的图像由该式得到 $y(x) = \alpha|x| + \varepsilon, x \in \{-100, .-99, ..., 100\}$, 这里 α 控制信号强度且 $\varepsilon \sim \mathcal{N}(0,1)$。实现一个简单的可逆跳跃-扩散算法将该图像分割为最多三部分。可逆跳跃在一个分割和两个分割解之间。两个分割空间的扩散移动共同端点（断开）的位置。当分割的数目和断点的位置给定时，利用普通最小二乘法去匹配最好的分割。试一试不同的信号强度 $\alpha \in \{0.01, 0.003, 0.1, 0.3, 1\}$，画出 10 个独立运行的平均能量(**??**)相对于计算时间（秒）的图示。

参考文献

[1] Pierre Bremaud. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, volume 31. Springer, 1999.

[2] Francis DK Ching and Corky Binggeli. *Interior design illustrated*. John Wiley & Sons, 2012.

[3] Persi Diaconis and Phil Hanlon. Eigen-analysis for some examples of the metropolis algorithm. *Contemporary Mathematics*, 138:99–117, 1992.

[4] Jack Dongarra and Francis Sullivan. Guest editors introduction: The top 10 algorithms. *Computing in Science & Engineering*, 2(1):22–23, 2000.

[5] Weina Ge and Robert T Collins. Marked point processes for crowd counting. In *CVPR*, pages 2913–2920. IEEE, 2009.

[6] Stuart Geman and Chii-Ruey Hwang. Diffusions for global optimization. *SIAM Journal on Control and Optimization*, 24(5):1031–1043, 1986.

[7] Peter J Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[8] Ulf Grenander and Michael I Miller. Representations of knowledge in complex systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 549–603, 1994.

[9] Feng Han, Zhuowen Tu, and Song-Chun Zhu. Range image segmentation by an effective jump-diffusion method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1138–1153, 2004.

[10] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[11] Scott Kirkpatrick, MP Vecchi, et al. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[12] Jun S Liu. Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, 6(2):113–119, 1996.

[13] Romeo Maciuca and Song-Chun Zhu. First hitting time analysis of the independence metropolis sampler. *Journal of Theoretical Probability*, 19(1):235–261, 2006.

[14] Kerrie L Mengersen, Richard L Tweedie, et al. Rates of convergence of the hastings and metropolis algorithms. *The Annals of Statistics*, 24(1):101–121, 1996.

[15] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[16] Nicholas Metropolis and Stanislaw Ulam. The monte carlo method. *Journal of the American statistical association*, 44(247):335–341, 1949.

[17] Maureen Mitton and Courtney Nystuen. *Residential interior design: a guide to planning spaces*. John Wiley & Sons, 2011.

[18] Siyuan Qi, Yixin Zhu, Siyuan Huang, Chenfanfu Jiang, and Song-Chun Zhu. Human-centric indoor scene synthesis using stochastic grammar. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[19] Richard L Smith and Luke Tierney. Exact transition probabilities for the independence metropolis sampler. *Preprint*, 1996.

[20] Zhuowen Tu and Song-Chun Zhu. Image segmentation by data-driven markov chain monte carlo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):657–673, 2002.

[21] Lap-Fai Yu, Sai Kit Yeung, Chi-Keung Tang, Demetri Terzopoulos, Tony F Chan, and Stanley Osher. Make it home: automatic optimization of furniture arrangement. *ACM Trans. Graph.*, 30(4):86, 2011.

[22] Song-Chun Zhu and David Mumford. *A stochastic grammar of images*. Now Publishers Inc, 2007.

[23] Song Chun Zhu and Alan Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(9):884–900, 1996.

第 5 章　吉布斯采样器及其变体

“天下难事必作于易，天下大事必作于细。千里之行，始于足下。”- 老子

5.1　引言

Donald 和 Stuart Geman

吉布斯采样器 [9]，最初由 Geman 兄弟 Donald 和 Stewart 发明，是一种用于从难以采样的分布中获取样本的 MCMC 算法。通常，分布以吉布斯形式表示：

$$\pi(\mathbf{x}) = \frac{1}{Z} e^{-E(\mathbf{x})}, \quad \mathbf{x} = (x_1, ..., x_d) \in \Omega。$$

这种分布出现在解决约束 (硬、软) 满足问题 (如图像去噪) 或贝叶斯推理中。

例 5.1 八皇后问题是约束满足问题的一个例子。此问题是在 8×8 的国际象棋棋盘上放置 8 个皇后，使得它们都不会相互威胁：即任意两个皇后都没有处于同一行、列或对角线。用表示八皇后棋盘坐标的 $\mathbf{s} \in \{1, ..., 64\}^8$ 表示一个可能的解，这些解属于一个集合

$$\Omega^* = \{\mathbf{s} \in \{1, ..., 64\}^8, h_i(\mathbf{s}) \leq 1, i = 1, ..., 46\}$$

其中 $h_i(\mathbf{s})$ 是计算每行、每列和每个对角线上的皇后数的 $8+8+30$ 个约束。

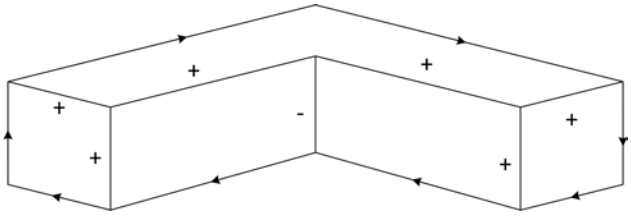


图 5.1: 线条画的一个例子和边缘的一个可能标记。

例 5.2 标记线条画图的边缘，使它们保持一致，如图 *5.1*所示。这是一个图 $G = \{V, E\}$ 上的约束满足问题。我们可以定义解集

$$\Omega^* = \{\mathbf{s} \in \{+, -, <, >\}^{|E|}, h_i(\mathbf{s}) = 1, i = 1, ..., |V|\}$$

其中 $h_i(\mathbf{s})$ 是在每个顶点的一致性硬（逻辑）约束。根据连接类型，这些约束如图 *5.2*所示。

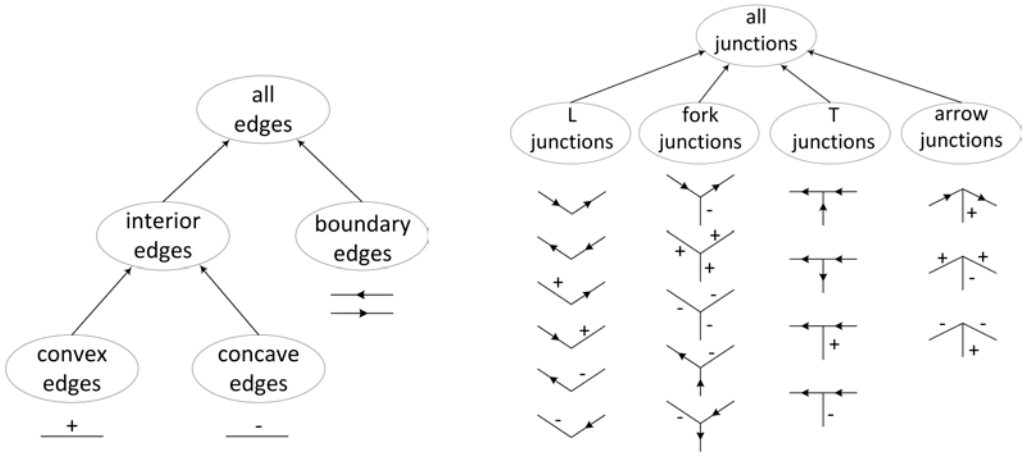


图 5.2: 这些允许的边缘标签和连接类型表示强约束和表示先验知识。

在这些情况下，人们可能想要找到分布的众数，或某些分布参数，如平均值、标准偏差等。在吉布斯采样器之前，使用松弛标记算法 [19] 找到众数，采用类似于下述算法。

松弛算法

Input: 能量函数 $E[\mathbf{x}]$, 当前状态 $\mathbf{x}^{(t)} = (x_1, ..., x_d) \in \Omega$

Output: 新状态 $\mathbf{x}^{(t+1)} \in \Omega$

1. 随机选择一个变量 $i \in \{1, ..., d\}$

2. 计算

$$u = \text{argmin}\Big(E[x_i = 1|x_{-i}], \cdots, E[x_i = L|x_{-i}]\Big).$$

3. 置

$$\mathbf{x}_{-i}^{(t+1)} = \mathbf{x}_{-i}^{(t)}, x_i^{(t+1)} = u.$$

这种贪婪算法的问题在于它无法保证找到全局最优。实际上，其经常陷入局部最优。吉布斯采样器作为松弛算法的随机版本引入，这就是为什么 Geman & Geman1984 的论文标题为“随机松弛”的原因。

在本章中，将讨论吉布斯采样器及其问题和一般化。数据增强的主题也进行介绍，并研究对 Julesz 系综的应用。

5.2 吉布斯采样器

吉布斯采样器的目标是对联合概率进行采样，

$$X = (x_1, x_2, \cdots, x_d) \sim \pi(x_1, x_2, \cdots, x_d)$$

根据条件概率在每个维度中采样，

$$x_i \sim \pi(x_i|\underbrace{x_{-i}}_{\text{fixed}}) = \frac{1}{Z}\exp(-E[x_i|x_{-i}]), \qquad \forall i。$$

这里 $\pi(x_i|x_{-i})$ 是一个在位点（变量）i 以其他变量为条件的的条件概率。

假设 Ω 是 d 维的，每个维度被离散化为 L 个有限状态。因此状态的总数是 L^d。吉布斯采样器程序如下算法所示。

吉布斯采样器

输入: 概率函数 $\pi(\mathbf{x})$, 当前状态 $\mathbf{x}^{(t)} = (x_1, ..., x_d) \in \Omega$
输出 t: 新状态 $\mathbf{x}^{(t+1)} \in \Omega$
1. 随机选择一个变量 $i \in \{1, ..., d\}$, 选择 L 个值 $y_1, ..., y_L$。
2. 使用下式计算条件概率向量 $\mathbf{u} = (u_1, ..., u_L)$

$$u_k = \pi(x_i = v_k|x_{-i})。$$

3. 采样 $j \sim \mathbf{u}$ 并置

$$\mathbf{x}_{-i}^{(t+1)} = \mathbf{x}_{-i}^{(t)}, x_i^{(t+1)} = y_j。$$

在上面的步骤 1 中选择变量的顺序可以是随机的也可以是遵循预定义的方案（例如 $1, 2, ..., d$）。

定义 5.1 *吉布斯采样器的一个扫描是对所有位点（变量）的一次顺序访问。*

虽然一个吉布斯每一步的转移矩阵 K_i 可能不是不可约和非周期性的，但很容易证明总转移矩阵 $K = K_1 \cdot K_2 \cdots K_d$ 在一次扫描后确实具有这些特征。因此，收缩系数满足 $C(K) < 1$。

如果在 t 时 $\mathbf{x}^{(t)} \sim \pi(\mathbf{x})$，且 $\mathbf{x}^{(t+1)} \sim \pi(\mathbf{x})$，那么 K 有 π 作为其不变概率，如下所示

$$\mathbf{x}^{(t)} = (x_1, \cdots, x_i, x_{i+1}, \cdots, x_d) \sim \pi(x)$$
$$\mathbf{x}^{(t+1)} = (x_1, \cdots, y_j, x_{i+1}, \cdots, x_d)。$$

在两个状态之间移动时发生的唯一变化是用 y_j 替换 x_i。然而，我们知道

$$\mathbf{x}^{(t+1)} \sim \pi(x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_d) \cdot \pi(y_j|x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_d) \implies \mathbf{x}^{(t+1)} \sim \pi(x)。$$

事实上，可以证明 [3] 周期性吉布斯采样器（使用预定义方案 $1, 2...d$ 访问位点）具有几何收敛率：

$$\|\mu K^n - \pi\|_{\text{TV}} \leqslant \frac{1}{2}(1 - e^{d\Delta})^n \|\mu - \pi\|_{\text{TV}}。$$

其中 $\Delta = \sup_i \delta_i$, 具有

$$\delta_i = \sup\{|E(\mathbf{x}) - E(\mathbf{y})|;\ x_j = y_j\ \forall j \neq i\}$$

这里我们使用 $E(\mathbf{x}), \pi(\mathbf{x})$ 的能量，即 $\pi(\mathbf{x}) = \frac{1}{Z}\exp\{-E(\mathbf{x})\}$。注意到我们只需要知道 $E(\mathbf{x})$ 和一个相加性常数。

5.2.1 吉布斯采样器的一个主要问题

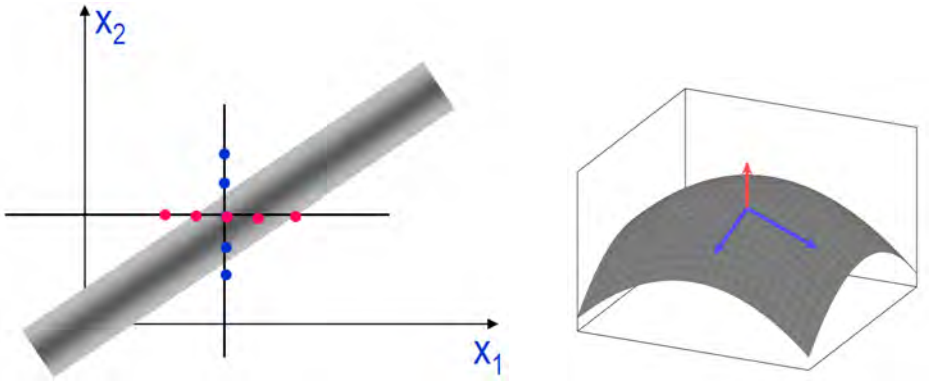


图 5.3: 左图：吉布斯采样器很难采样具有两个紧耦合变量的概率，如例 5.3中所述。右图：一般来说，吉布斯采样器很难采样集中在流形上的数据。

下例中说明了吉布斯采样器的一个主要问题。

例 5.3 *对于一个概率 $\pi(x_1,x_2)$，其概率质量集中在一维线段上，如图 5.3所示, 对这两个维度进行迭代采样显然是低效的，即链是“锯齿状”的。*

这个问题产生的原因是两个变量紧密耦合。最好是我们沿着线的方向移动。通常，当概率集中在 d 维空间中的较低维度流形中时会产生这个问题。马尔可夫链不允许在法线方向（离开流形）移动，而是仅能在切线方向上。

我们知道，吉布斯分布源自变量 $\underline{\mathrm{x}}$ 上的约束，因此它们是在一些隐式流形中定义的，

$$\Omega(H_0) = \{X : H_i(\mathbf{x}) = h_i, i = 1,2,\cdots,K\}, \qquad H_0 = (h_1,h_2,\cdots,h_K)。$$

使用吉布斯采样器难以采样的吉布斯分布的例子一般是马尔可夫随机场，特别是 Ising/Potts 模型。

设 $\mathbf{G} =< V,E >$ 是邻接图，例如一个具有 4 个最近邻连接的栅格。每个顶点 $v_i \in V$ 都有一个具有有限数量标签（或颜色）的状态变量 x_i，$x_i \in \{1,2,...,\mathrm{L}\}$。标签 L 的总数是预先定义的。

定义 5.2 *设 $\mathbf{x} = (x_1,x_2,...,x_{|V|})$ 表示图的标记，那么 Ising/Potts 模型是一个马尔可夫随机场，*

$$\pi_{\mathrm{PTS}}(\mathbf{x}) = \frac{1}{Z}\exp\{-\sum_{<s,t>\in E}\beta_{st}\mathbf{1}(x_s \neq x_t)\}, \tag{5.1}$$

其中 $\mathbf{1}(x_s \neq x_t)$ 是一个布尔函数，若满足条件 $x_s \neq x_t$ 则等于 1，否则等于 0。如果可能的标签数量是 $L = 2$，则 π 称为 Ising 模型；如果 $L \geq 3$，则称为 Potts 模型。

对于一个更倾向于相邻顶点的颜色相同的铁磁体系统，我们通常认为 $\beta_{st} > 0$。Potts 模型及其扩展在许多贝叶斯推理任务中用作先验概率。

图 5.4: Ising 模型具有平坦的难以采样的能量景观

例 5.4 对于定义在 *Ising* 模型(5.1)上的单位点吉布斯采样器，由于平的能量景观，如图(5.4)所示，边界节点以概率 $p = 1/2$ 翻转。翻转一串长度为 n 的字符串平均需要 $t \geq 1/p^n = 2^n$ 步！这意味着等待时间是指数的。

5.3 Gibbs 采样器泛化

本节介绍几个吉布斯采样器的修正和泛化，这些修正和泛化减轻了对 5.2.1节中强调的相关变量所使用的方法所带来的一些困难。

5.3.1 击中逃跑

该设计随机选择一个方向并在该方向上进行采样。

假设当前状态是 $\mathbf{x}^{(t)}$。

1) 选择一个方向或坐标轴 $\vec{e}_t$。

2) 沿坐标轴采样。

$$r \sim \pi(\mathbf{x}^{(t)} + r \cdot \vec{e}_t)$$

3) 更新状态

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + r \cdot \vec{e}_t$$

沿着坐标轴的采样是一个连续的吉布斯采样器，可以通过 Multi-Try Metropolis 实现。然而，这种设计仍然存在一个问题就是如何选择采样方向。

5.3.2 广义 Gibbs 采样器

作为进一步的一种泛化，我们可以不必在直线上移动。在更一般的情况下，只要移动保持不变概率，就可以将一组变换用于可能的移动。

Theorem 2 (Liu and Wu, 1999[12]) 设 $\Gamma = \{\gamma\}$ 是一个局部紧群，它作用于空间 Ω，每个元素乘法是一个可能的移动，由下式给出

$$\mathbf{x}^{(t)} \to \mathbf{x}^{(t+1)} = \gamma \cdot \mathbf{x}^{(t)}。$$

如果 $\mathbf{x} \sim \pi$ 且元素 $\gamma \in \Gamma$ 由下式选择

$$\gamma | \mathbf{x} \sim \pi(\gamma \cdot \mathbf{x}) | J_\gamma(\mathbf{x}) | H(\mathrm{d}\gamma),$$

其中 $J_\gamma(x)$ 是在 $\mathbf{x}$ 处计算的变换 $\mathbf{x} \to \gamma \cdot \mathbf{x}$ 的雅可比矩阵，$H(\mathrm{d}\gamma)$ 是左不变哈尔测度，

$$H(\gamma \cdot B) = H(B), \qquad \forall \gamma, B,$$

则新状态服从不变概率

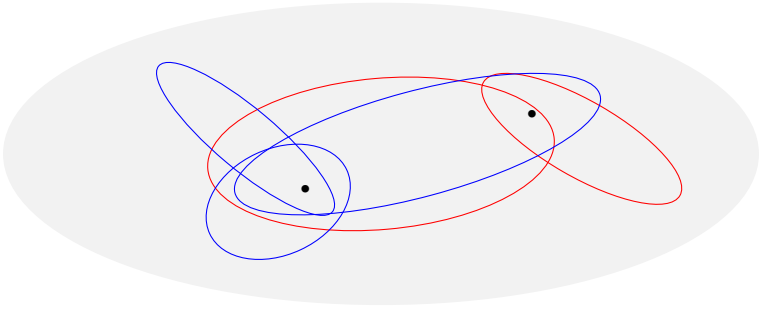
$$\mathbf{x}^{(t+1)} = \gamma \cdot \mathbf{x} \sim \pi。$$

5.3.3 广义击中逃跑

从概念上讲，将击中逃跑的想法推广到空间的任意分区尤其在有限状态空间中是有益的。这个概念是由 Persi Diaconis 在 2000 年提出的。

假设马尔可夫链由许多子链组成，转移概率是线性加权和，

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{N} \omega_i K_i(\mathbf{x}, \mathbf{y}), \qquad \omega_i = p(i), \quad \sum_{i=1}^{N} \omega_i = 1。$$



如果每个子核具有相同的不变概率，

$$\sum_{\mathbf{x}} \pi(\mathbf{x}) K_i(\mathbf{x}, \mathbf{y}) = \pi(\mathbf{y}), \qquad \forall \mathbf{y} \in \Omega,$$

则整个马尔可夫链服从 $\pi(\mathbf{x})$。通过第 i 个类型、核为 K_i 的移动连接到 $\mathbf{x}$ 的状态的集合表示为

$$\Omega_i(\mathbf{x}) = \{\mathbf{y} \in \Omega : K_i(\mathbf{x}, \mathbf{y}) > 0\}。$$

$\mathbf{x}$ 连接到集合

$$\Omega(\mathbf{x}) = \cup_{i=1}^{N} \Omega_i(\mathbf{x})。$$

这个方法的关键问题是:

1. 如何以一种系统的、有规则的方式，确定采样维度、方向、群变换、以及集合 $\Omega_i(\mathbf{x})$？

2. 如何调度由 $p(i)$ 控制的访问顺序？即如何选择移动方向、群和集合？

5.3.4 利用辅助变量采样

我们想从 $\pi(\mathbf{x})$ 中对 $\mathbf{x}$ 采样，但由于变量之间的相关性，可能很难从中采样。避开这些相关性的一个系统性方法是引入辅助随机变量 y，使得

$$\mathbf{x} \sim \pi(\mathbf{x}) \quad \rightarrow \quad (\mathbf{x}, y) \sim \pi^{+}(\mathbf{x}, y)。$$

辅助变量 y 的示例:

- T – 温度: 模拟退火 [11]
- S – 尺度: 多栅格采样
- w – 权重: 动态加权
- b – 边界: 聚类采样，Swendsen-Wang 方法 [7, 13]
- u – 能级：切片采样 [7]

5.3.5 模拟退火

设目标概率为

$$\pi(\mathbf{x}) = \frac{1}{Z} \exp\{-U(\mathbf{x})\},$$

对于 L 个温度水平，在 $\{1, 2, \ldots, \mathrm{L}\}$ 中增加一个变量 I

$$1 = T_1 < T_2 < \cdots < T_L。$$

然后对联合概率进行采样，在 $I = 1$ 保持 X's

$$(x, I) \sim \pi^{+}(x, I) = \frac{1}{Z^{+}} \exp\{-\frac{1}{T_I} U(\mathbf{x})\}。$$

在高温时采样器会更自由地移动，但很难在不同的温度水平之间穿越。假设在 L 个水平上并行地运行马尔可夫链。对所有链定义一个联合概率

$$\pi^{+}(\mathbf{x}_1, \ldots, \mathbf{x}_L) \propto \prod_{i=1}^{L} \exp\{-\frac{1}{T_i} U(\mathbf{x}_i)\}。$$

变换两条链

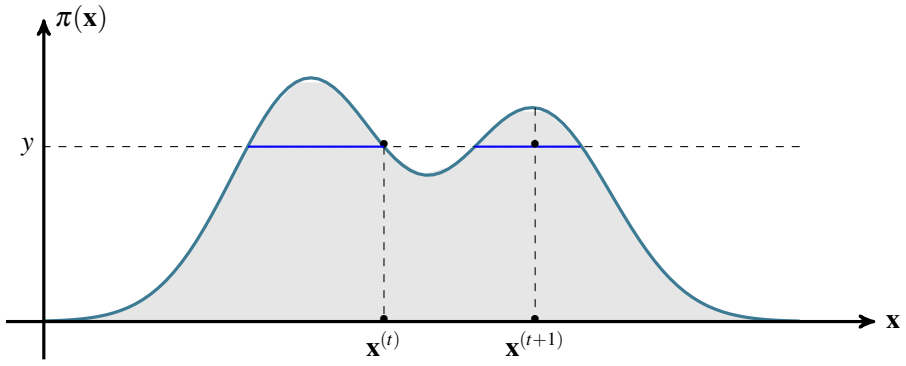
$$(\ldots, \mathbf{x}_i, \ldots, \mathbf{x}_j, \ldots) \rightarrow (\ldots, \mathbf{x}_j, \ldots, \mathbf{x}_i, \ldots)。$$

最后以 Metropolis-Hastings 接受

$$\alpha = \min\left(1, \exp\{\big(\frac{1}{T_j} - \frac{1}{T_i}\big)\big(U(\mathbf{x}_j) - U(\mathbf{x}_i)\big)\}\right)。$$

5.3.6 切片采样

假设 $\mathbf{x} \sim \pi(\mathbf{x})$ 在一个一维分布中。我们为概率水平引入一个辅助变量 $y \in [0,1]$。因此，采样 $\mathbf{x} \sim \pi(\mathbf{x})$ 相当于从 $(\mathbf{x}, y)$ 空间中的阴影区域均匀采样。



我们使满足条件

$$\sum_{y} \pi^{+}(\mathbf{x}, y) = \pi(\mathbf{x}),$$

但是

$$\begin{cases} y \sim \pi^{+}(y|\mathbf{x}) = \text{unif}(0, \pi(\mathbf{x})) \leftarrow \text{容易采样} \\ \mathbf{x} \sim \pi^{+}(\mathbf{x}|y) = \text{unif}(\overbrace{\{x; \pi(\mathbf{x}) \geqslant y\}}^{\text{level set}}) \leftarrow \text{难以采样。} \end{cases}$$

切片 $\{x; \pi(\mathbf{x}) \geqslant y\}$ 通常包含由水平集 $\pi(bx) = y$ 限定的多个部分，并且难以采样。这种情况如图 5.5所示。

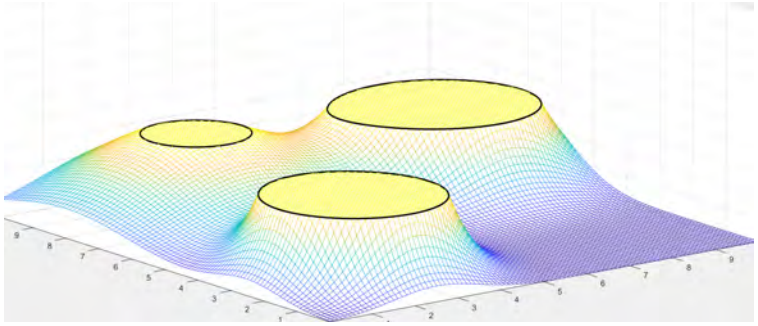


图 5.5: 切片 $\{x; \pi(\mathbf{x}) \geqslant y\}$ 通常包含多个部分，并且难以采样。

5.3.7 数据增强

对下式给出的辅助变量，切片采样方法提出两个一般条件

$$\mathbf{x} \sim \pi(\mathbf{x}) \quad \rightarrow \quad (\mathbf{x}, y) \sim \pi^{+}(\mathbf{x}, y)。$$

1) 边缘概率为

$$\sum_{y} \pi^{+}(\mathbf{x}, y) = \pi(\mathbf{x})。$$

2) 两个条件概率都可以分解，并且易于从下式中采样

$$\begin{cases} \mathbf{x} \sim \pi^{+}(\mathbf{x}|y) \\ y \sim \pi^{+}(y|\mathbf{x})。 \end{cases}$$

数据增强的直觉如下：
很多情况下概率集中在分离的众数（区域）上，并且在这些众数之间跳跃是很困难的，因为马尔可夫链通常在局部移动。良好的辅助变量将会:

1) 帮助选择移动方向/群/集合（在广义击中逃跑中）。

2) 扩大搜索范围。

5.3.8 Metropolized 吉布斯采样器

回到第 5.3.3节中的广义击中逃跑设置，其中核包含许多子核

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{N} \omega_i K_i(\mathbf{x}, \mathbf{y}), \qquad \omega_i = p(i), \quad \sum_{i=1}^{N} \omega_i = 1,$$

这些子核具有相同的不变概率，

$$\sum_{\mathbf{x}} \pi(\mathbf{x}) K_i(\mathbf{x}, \mathbf{y}) = \pi(\mathbf{y}), \qquad \forall \mathbf{y} \in \Omega。$$

通过第 i 类移动连接到 $\mathbf{x}$ 的状态集合为

$$\Omega_i(\mathbf{x}) = \{\mathbf{y} \in \Omega : K_i(\mathbf{x}, \mathbf{y}) > 0\}。$$

$\mathbf{x}$ 被连接到集合

$$\Omega(\mathbf{x}) = \cup_{i=1}^{N} \Omega_i(\mathbf{x})。$$

我们知道有两种一般设计: Gibbs 和 Metropolis.

1) Gibbs: 我们在每个集合中采样概率

$$y \sim [\pi]_i(\mathbf{y}), \qquad [\pi]_i(\mathbf{y}) \sim \begin{cases} \pi(\mathbf{y}) & y \in \Omega_i(\mathbf{x}), \\ 0, & y \notin \Omega_i(\mathbf{x})。 \end{cases}$$

在这种情况下，移动是对称的

$$\Omega_i(\mathbf{x}) = \Omega_i(\mathbf{y})。$$

2) Metropolis: 根据一个任意的 $\Omega_i(\mathbf{x})$ 移动，但提议分布 q 未知，其中

$$q_i(\mathbf{x},y)=\frac{\pi(\mathbf{y})}{\sum_{y'\in\Omega_i(\mathbf{x})}\pi(y')},\qquad \forall y'\in\Omega_i(\mathbf{x})。$$

但是需要检查

$$q_i(y,x)=\frac{\pi(\mathbf{x})}{\sum_{x'\in\Omega_i(\mathbf{y})}\pi(\mathbf{x}')},\qquad \forall x'\in\Omega_i(\mathbf{y})。$$

现在的问题是移动不再是对称的，即 $\Omega_i(\mathbf{x})\neq\Omega_i(\mathbf{y})$。虽然进行了归一化，但由于集合不同，细致平衡方程可能不满足。为了修正这种移动并得到正确的平衡，我们需要一个条件，

$$\mathbf{y}\in\Omega_i(\mathbf{x})\qquad\Longleftrightarrow\qquad\mathbf{x}\in\Omega_i(\mathbf{y})。$$

我们取接受概率为

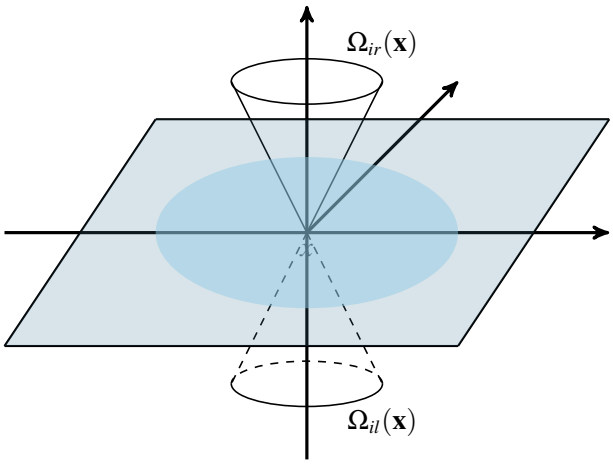
$$\alpha_i(\mathbf{x},\mathbf{y})=\min\left(1,\frac{q_i(\mathbf{y},\mathbf{x})\cdot\pi(\mathbf{y})}{q_i(\mathbf{x},\mathbf{y})\cdot\pi(\mathbf{x})}\right)=\min\left(1,\frac{\frac{\pi(\mathbf{x})}{\sum_{\mathbf{x}'\in\Omega_i(\mathbf{y})}\pi(\mathbf{x}')}\cdot\pi(\mathbf{y})}{\frac{\pi(\mathbf{y})}{\sum_{\mathbf{y}'\in\Omega_i(\mathbf{x})}\pi(\mathbf{y}')}\cdot\pi(\mathbf{x})}\right)$$

$$=\min\left(1,\frac{\overbrace{\sum_{\mathbf{y}'\in\Omega_i(\mathbf{x})}\pi(\mathbf{y}')}^{\Omega_i(\mathbf{x})\text{的总概率质量}}}{\underbrace{\sum_{\mathbf{x}'\in\Omega_i(\mathbf{y})}\pi(\mathbf{x}')}_{\Omega_i(\mathbf{y})\text{的总概率质量}}}\right)。$$

子核是成对设计的,

$$K_i(\mathbf{x},y)=\omega_{i1}K_{il}(\mathbf{x},y)+\omega_{ir}K_{ir}(\mathbf{x},y),$$

并有相应的空间 $\Omega_{il}(\mathbf{x})$ 和 $\Omega_{ir}(\mathbf{x})$。



在这种情况下，接受比是

$$\alpha_i(\mathbf{x},\mathbf{y}) = \min\left(1, \frac{\sum_{\mathbf{y}'\in\Omega_{il}(\mathbf{x})}\pi(\mathbf{y}')}{\sum_{\mathbf{x}'\in\Omega_{ir}(\mathbf{y})}\pi(\mathbf{x}')}\right), \qquad \mathbf{y}\in\Omega_{il}(\mathbf{x})。$$

如果集合是对称的, *i.e.* $\Omega_{il}(\mathbf{x}) = \Omega_{ir}(\mathbf{y})$, 则接受比为 1。如果集合是非对称的，则需要 Metropolis 接受步骤来重新平衡移动。

我们可以通过禁止马尔可夫链在条件概率中保持其当前状态来改进传统的吉布斯采样器。因此，这些集合肯定是不对称的，需要 Metropolis 接受步骤来重新平衡。该方法称为 Metropolized 吉布斯采样器（MGS）。提议矩阵中对角线的元素设为零。这是马尔可夫链的理想特性，因为它使链具有快速混合时间。

$$q(\mathbf{x},\mathbf{y}) = \frac{\pi(\mathbf{y})}{1-\pi(\mathbf{x})}, \qquad \mathbf{y}\in\Omega(\mathbf{x}), \mathbf{x}\notin\Omega(\mathbf{x}),$$

其中 1 表示归一化因子。因此，接受比变为

$$\alpha(\mathbf{x},y) = \min\left(1, \frac{1-\pi(\mathbf{x})}{1-\pi(\mathbf{y})}\right)。$$

此外，已经证明

$$K_{\text{MGS}}(\mathbf{x},\mathbf{y}) \geqslant K_{\text{Gibbs}}(\mathbf{x},\mathbf{y}), \qquad \forall \mathbf{x}\neq\mathbf{y}\in\Omega。$$

5.4 数据关联和数据增强

在许多情况下，当观测数据 $\mathbf{y}$ 增加了一些丢失（隐藏）数据 $\mathbf{h}$ 时，可以得到一个精确的模型 $f(\mathbf{y},\mathbf{h}|\theta)$。例如，如果观测的灰度图像增加一个包含面部位置、旋转、比例、3D 姿势以及其他变量（例如太阳眼镜、胡须等）的向量，则可以获得更精确的人脸模型 $f(\mathbf{y},\mathbf{h}|\theta)$（其中 $\theta\in\{0,1\}$ 可以表示人脸/非人脸）。

然后通过对隐变量积分，可以得到以观测数据为条件的参数 θ 的后验分布

$$p(\theta|\mathbf{y}) = \int p(\theta|\mathbf{y},\mathbf{h})p(\mathbf{h}|\mathbf{y})d\mathbf{h}。 \tag{5.2}$$

如果能对 $p(\mathbf{h}|\mathbf{y})$ 采样，那么我们可以使用方程(5.2)来得到 $p(\theta|\mathbf{y})$ 的蒙特卡罗近似。Tanner 和 Wong [14] 发现可以使用目标分布 $p(\theta|\mathbf{y})$ 的初始近似 $f(\theta)$，通过先对 $\theta_i\sim f(\theta)$ 采样，然后在对 $\mathbf{h}_i\sim p(\mathbf{h}|\theta_i,\mathbf{y})$ 采样，从

$$\tilde{p}(\mathbf{h}) = \int p(\mathbf{h}|\theta,\mathbf{y})f(\theta)d\theta,$$

中获得隐变量 $\mathbf{h}_1,...,\mathbf{h}_m$。这些隐样本也称为多重插补。我们可以使用他们来获得（有希望的）更好的目标分布近似

$$f(\theta) = \frac{1}{m}\sum_{i=1}^{m} p(\theta|\mathbf{y},\mathbf{h}_i)。$$

因此，原始数据增强算法以一组隐藏值 $\mathbf{h}_1^{(0)},...,\mathbf{h}_m^{(0)}$ 开始，以如下方式进行：

数据增强（DA）

初始化 $\mathbf{h}_1^{(0)},...,\mathbf{h}_m^{(0)}$
for 从 t=1 到 N^{iter} **do**
  for 从 i=1 到 m **do**
    从 $\{1,...,m\}$ 中随机抽取 k
    采样 $\theta' \sim p(\theta|\mathbf{y},\mathbf{h}_k^{(t-1)})$
    采样 $\mathbf{h}_k^{(t)} \sim p(\mathbf{h}|\mathbf{y},\theta')$
  end for
end for

一个重要的观测结果是 DA 算法等价于 $m=1$ 的版本，因为当前的每个元素 $\mathbf{h}_1^{(t)},...,\mathbf{h}_m^{(t)}$ 都可以追溯到它的起源。可以很容易地看出，当 t 足够大时，来自 t 代的所有样本都源于单个元素。由于父母被选择的纯随机方式，共同祖先的选择没有偏差。因此，DA 算法相当于 Gibbs 采样器类型的算法，该算法在对参数 θ 进行采样和对隐变量 $\mathbf{h}$ 进行采样之间进行交替：

简化的数据增强

初始化 $\mathbf{h}$
for 从 t=1 到 N^{iter} **do**
  采样 $\theta' \sim p(\theta|\mathbf{y},\mathbf{h}^{(t-1)})$
  采样 $\mathbf{h}^{(t)} \sim p(\mathbf{h}|\mathbf{y},\theta')$
end for

5.5 Julesz 系综和 MCMC 纹理采样

设 $\mathbf{I}$ 是定义在有限栅格 $\Lambda \subset Z^2$ 上的图像。对于每个像素 $v=(x,y) \in \Lambda$，在 v 处的灰度值表示为 $\mathbf{I}(v) \in S$，S 是实线上的有限区间或者是量化的灰度级的有限集合。我们用 $\Omega_\Lambda = S^{|\Lambda|}$ 表示 Λ 上所有图像的空间。

在对同质纹理图像建模时，我们对探索局部图像特征的有限统计量集合感兴趣。最初通过使用多边形和团上的共生矩阵对这些统计量进行了研究，但其被证明不足以描述现实世界的图像，以及与生物视觉系统无关。在 20 世纪 80 年代后期，人们认识到真实世界的意象能被空间/频率基更好地表示，如 Gabor 滤波器 [6]，小波变换 [5] 和滤波器金字塔。

给定一个滤波器的集合 $\{F^{(\alpha)},\alpha=1,2,...,K\}$，为每个滤波器 $F^{(\alpha)}$ 计算一个子带图像 $\mathbf{I}^{(\alpha)}=F^{(\alpha)}*\mathbf{I}$。然后统计量从子带图像或金字塔而不是强度图像中提取。从降维的角度看，滤波器表征局部纹理特征，从而子带图像的简单统计量可以捕获在高位空间中需要 k 百分度或团统计量的信息。

尽管 Gabor 滤波器在生物视觉 [4] 中具有良好的基础，但对于视觉皮层如何在图像中池化统计量知之甚少。文献中有四种流行的统计量的选择。

1. 单个滤波器响应的矩，如 $\mathbf{I}^{(\alpha)}$ 的均值和方差。2. 类似于“on/off”细胞响应的整流函数 [2]：

$$\mathbf{h}^{(\alpha,+)}(\mathbf{I})=\frac{1}{|\Lambda|}\sum_{v\in\Lambda}R^+(\mathbf{I}^{(\alpha)}(v)),\mathbf{h}^{(\alpha,-)}(\mathbf{I})=\frac{1}{|\Lambda|}\sum_{v\in\Lambda}R^-(\mathbf{I}^{(\alpha)}(v))。$$

3. $\mathbf{I}^{(\alpha)}$ 的经验直方图的一个单元区间。4. $(\mathbf{I}^{(1)},\cdots,\mathbf{I}^{(k)})$ 的完整联合直方图的一个区间。

在下文中，我们将研究基于这样的滤波器统计量的纹理的数学定义 - Julesz 系综 -以及从其中采样图像的算法。

5.5.1 Julesz 系综 - 纹理的数学定义

给定一个 K 个统计量 $\mathbf{h}=\{\mathbf{h}^{(\alpha)}:\alpha=1,2,...,K\}$ 的集合，其已经相对于栅格 $|\Lambda|$ 的大小进行了标准化，图像 $\mathbf{I}$ 被映射到统计空间中的一个点 $\mathbf{h}(\mathbf{I})=(\mathbf{h}^{(1)}(\mathbf{I}),...,\mathbf{h}^{(K)}(\mathbf{I}))$。设

$$\Omega_\Lambda(\mathbf{h}_0)=\{\mathbf{I}:\mathbf{h}(\mathbf{I})=\mathbf{h}_0\}$$

为共享相同统计量 $\mathbf{h}_o$ 的图像集合。图像空间 Ω_Λ 划分为等价类

$$\Omega_\Lambda=\cup_{\mathbf{h}}\Omega_\Lambda(\mathbf{h})。$$

由于有限栅格中的灰度量化，实际上需要减小对统计量的约束，并将图像集合定义为

$$\Omega_\Lambda(\mathcal{H})=\{\mathbf{I}:\mathbf{h}(\mathbf{I})\in\mathcal{H}\},$$

其中 $\mathcal{H}$ 是 $\mathbf{h}_0$ 周围的开集。$\Omega_\Lambda(\mathcal{H})$ 意味着一个均匀分布

$$q(\mathbf{I};\mathcal{H})=\begin{cases}\frac{1}{|\Omega_\Lambda(\mathcal{H})|} & 对\ \mathbf{I}\in\Omega_\Lambda(\mathcal{H}),\\ 0 & 其他,\end{cases}$$

其中 $|\Omega_\Lambda(\mathcal{H})|$ 是集合的容量。

定义 *给定一组标准化的统计量 $\mathbf{h}=\{\mathbf{h}^{(\alpha)}:\alpha=1,2,...,K\}$，一个 Julesz 系综是在某些边界条件下当 $\Lambda\to Z^2$ 且 $\mathcal{H}\to\{\mathbf{h}\}$ 时 $\Omega_\Lambda(\mathcal{H})$ 的极限。*

一个 Julesz 系综 $\Omega(\mathbf{h})$ 是在一个 $\mathcal{H}$ 接近 $\mathbf{h}$ 的大的栅格上 $\Omega_\Lambda(\mathcal{H})$ 的数学理想化。当 $\Lambda\to Z^2$ 时，使标准化统计量 $\mathcal{H}\to\{\mathbf{h}\}$ 是有意义的。我们在与 van Hove[10] 同样的意义上假设 $\Lambda\to Z^2$，即边界大小与 Λ 大小之间的比率变为 0，$|\partial\Lambda|/|\Lambda|\to 0$。事实上，我们通常会认为栅格足够大如果 $\frac{|\partial\Lambda|}{|\Lambda|}$ 非常小，如 1/15。因此，略微滥用一下这个概念并避免处理极限的专有性，我们将一个足够大的图像（如 256 × 256 像素）看做一个无限图像。更详细的说明请参考文献 [15]。

一个 Julesz 系综 $\Omega(\mathbf{h})$ 在 Z^2 上定义了一个纹理模式，并将纹理映射到特征统计空间 $\mathbf{h}$ 中。为了与颜色进行比较，当一个波长为 $\lambda\in[400,700]$ 纳米的电磁波定义了一种独特的可见颜色，统计值 $\mathbf{h}$ 定义了纹理模式！[1]

纹理的数学定义可能与人类的纹理感知不同。后者对统计量 $\mathbf{h}$ 具有非常粗糙的精确度，并且经常受到经验的影响。

在创建纹理的数学定义期间，建模被设为反问题。假设我们获得了一组观测的训练图像 $\Omega_{\text{obs}}=\{\mathbf{I}_{\text{obs},1},\mathbf{I}_{\text{obs},2},...,\mathbf{I}_{\text{obs},M}\}$，他们是从一个未知的 Julesz 系综 $\Omega_*=\Omega(\mathbf{h}_*)$ 中采样得到的。纹理建模的目的是搜索统计量 $\mathbf{h}_*$。

[1]我们用 Julesz 命名这个系综，以纪念他在纹理方面的先驱性工作。这并不是意味着 Julesz 用这种数学公式定义了纹理模式。

如上所述，我们首先从字典 B 中选择 K 个统计量的集合，然后利用下式计算观测图像的标准化统计量 $\mathbf{h}_{\text{obs}} = (\mathbf{h}_{\text{obs}}^{(1)}, ..., \mathbf{h}_{\text{obs}}^{(K)})$，

$$\mathbf{h}_{\text{obs}}^{(\alpha)} = \frac{1}{M}\sum_{i=1}^{M}\mathbf{h}^{(\alpha)}(\mathbf{I}_{\text{obs},i}), \quad \alpha = 1, 2, ..., K。 \tag{5.3}$$

然后使用 $\mathbf{h}_{\text{obs}}$ 定义纹理图像的系综，

$$\Omega_{K,\varepsilon} = \{\mathbf{I} : D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}_{\text{obs}}^{(\alpha)}) \leq \varepsilon, \quad \forall\alpha\}, \tag{5.4}$$

其中 D 是某种距离，例如直方图的 L_1 距离。如果 Λ 足够大到可以被认为是无限的，我们可以将 ε 实质设为 0，并将相应的 $\Omega_{K,\varepsilon}$ 表示为 Ω_K。系综 Ω_K 意味着 Ω_K 上的均匀概率分布 $q(\mathbf{I};\mathbf{h})$，其熵是 $\log|\Omega_K|$。

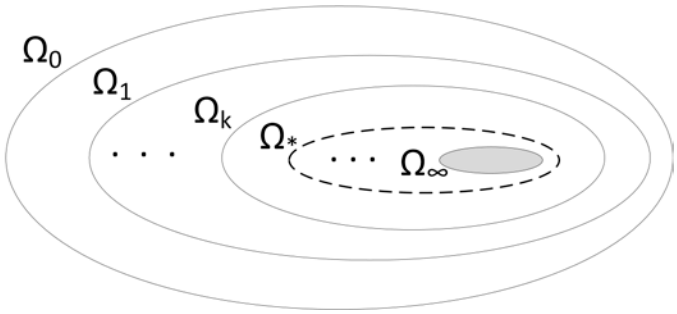


图 5.6: 随着加入更多的统计约束，Julesz 系综的容量（或熵）单调减少。©[2000] IEEE。经许可重印，来自参考文献 [16]。

为了搜索潜在的 Julesz 系综 Ω_*，我们可以采用 Zhu，Wu 和 Mumford 使用的搜寻策略 (1997)[17]。当 $k = 0$ 时，我们有 $\Omega_0 = \Omega_\Lambda$。假设在步骤 k 时，统计量 $\mathbf{h}$ 被选择。然后，在步骤 $k+1$ 时一个统计量 $\mathbf{h}^{(k+1)}$ 被加入以得到 $\mathbf{h}_+ = (\mathbf{h}, \mathbf{h}^{(k+1)})$。选择 $\mathbf{h}^{(k+1)}$ 为在字典 B 中的所有统计量中具有最大的熵减，

$$\mathbf{h}^{(k+1)} = \arg\max_{\beta\in B}[\text{entropy}(q(\mathbf{I};\mathbf{h})) - \text{entropy}(q(\mathbf{I};\mathbf{h}_+)] = \arg\max_{\beta\in B}[\log|\Omega_k| - \log|\Omega_{k+1}|]。 \tag{5.5}$$

熵减称为 $\mathbf{h}^{(k+1)}$ 的信息增益。

如图 5.6所示，随着增加更多统计量，Julesz 系综的熵或容量单调减少

$$\Omega_\Lambda = \Omega_0 \supseteq \Omega_1 \supseteq \cdots \supseteq \Omega_k \supseteq \cdots。$$

显然，引入太多统计量会导致过拟合。在极限 $k \to \infty$ 中，Ω_∞ 仅包括 Ω_{obs} 中的观测图像及其转化版本。

给定观测的有限图像，统计量 $\mathbf{h}$ 和 Julesz 系综 $\Omega(\mathbf{h})$ 的选择是一个模型复杂度问题，其已在统计学文献中进行了广泛研究。在最大最小熵模型 [17, 18] 中，AIC 准则 [1] 被采用进行模型选择。AIC 的直观想法很简单。给定有限图像，我们应该测量新统计量 $\mathbf{h}^{(k+1)}$ 对 Ω_{obs} 中训练图像的波动。因此，当添加一个新的统计量时，它会带来新的信息以及估计误差。当 $\mathbf{h}^{(k+1)}$ 带来的估计误差大于其信息增益时，应该停止特征搜寻过程。

5.5.2 Gibbs 系综和系综等价性

在本节中，我们将讨论 Gibbs 系综以及 Julesz 和 Gibbs 系综之间的等价性。文献 [15] 中给出了详细的讨论。给定一组观测图像 Ω_{obs} 和统计量 $\mathbf{h}_{\text{obs}}$，另一个研究方向是求索概率纹理模型，特别是在 Gibbs 分布或 MRF 模型中。MRF 模型的一个一般类型是由 Zhu, Wu, and Mumford 于 1997 年研究的 FRAME 模型 [17, 18]。源自最大熵准则的 FRAME 模型具有 Gibbs 形式

$$p(\mathbf{I};\beta) = \frac{1}{Z(\beta)} \exp\{-\sum_{\alpha=1}^{K} < \beta^{(\alpha)}, \mathbf{h}^{(\alpha)}(\mathbf{I}) >\} = \frac{1}{Z(\beta)} \exp\{< \beta, \mathbf{h}(\mathbf{I}) >\}。 \tag{5.6}$$

参数 $\beta = (\beta^{(1)}, \beta^{(2)}, ..., \beta^{(K)})$ 是拉格朗日乘数。通过使 $p(\mathbf{I};\beta)$ 重现观测统计量确定 β 的值，

$$E_{p(\mathbf{I};\beta)}[\mathbf{h}^{(\alpha)}(\mathbf{I})] = \mathbf{h}^{(\alpha)}_{\text{obs}} \quad \alpha = 1, 2, ..., K。 \tag{5.7}$$

统计量的选择遵循最小熵准则。

随着图像栅格变得足够大，标准化统计量的波动减小。因此，当 $\Lambda \to Z^2$ 时，FRAME 模型在没有相变的情况下收敛到一个有限随机场。有限随机场本质上将其所有概率质量均匀地集中在一组图像上，我们称其为 *Gibbs 系综*。[2]

在 [15] 中已经表明 $p(\mathbf{I};\beta)$ 给出的 Gibbs 系综等价于 $q(\mathbf{I};\mathbf{h}_{\text{obs}})$ 指定的 Julesz 系综。β 和 $\mathbf{h}_{\text{obs}}$ 之间的关系用等式 （5.7）表示。直观地，$q(\mathbf{I};\mathbf{h}_{\text{obs}})$ 由硬约束定义，而 Gibbs 模型 $p(\mathbf{I};\beta)$ 由软约束定义。两者都使用观测统计量 $\mathbf{h}_{\text{obs}}$，而当栅格 Λ 变得足够大时，模型 $p(\mathbf{I};\beta)$ 均匀集中在 Julesz 系综上。

上面的系综等价揭示了纹理建模中的两个重要事实。

1. 给定一组统计量 $\mathbf{h}$，我们可以通过从 Julesz 系综 $\Omega(\mathbf{h})$ 中采样来合成拟合 FRAME 模型的典型纹理图像，而无需在 FRAME 模型中学习参数 β[17]，其很浪费时间。因此，利用 Julesz 系综可以有效地完成特征追寻，模型选择和纹理合成。

2. 对于从 Julesz 系综采样的图像，给定其环境的图像局部分块服从由最小最大熵准则得出的 Gibbs 分布（或 FRAME 模型）。因此，Gibbs 模型 $p(\mathbf{I};\beta)$ 为小的图像分块上的 $q(\mathbf{I};\mathbf{h})$ 的条件分布提供了参数形式。$p(\mathbf{I};\beta)$ 应该用于纹理分类和分割等任务。

Julesz 系综的追寻也可以基于最小最大熵准则。首先，定义 $\Omega(\mathbf{h})$ 作为共享统计量 $\mathbf{h}$ 的图像的最大集合等价于最大熵准则。其次，方程 (5.5) 中的统计量追寻使用最小熵准则。因此，在最小最大熵理论下，一个纹理建模的统一框架出现了。

5.5.3 Julesz 系综采样

采样 Julesz 系综是一项很重要的任务！当 $|\Omega_K|/|\Omega_\Lambda|$ 呈指数减小时，Julesz 系综在图像空间中的容量几乎为零。因此拒绝采样方法是不合适的，我们改为使用马尔可夫链蒙特卡罗方法。

[2]在计算特征统计量 $\mathbf{h}(\mathbf{I})$ 时，我们需要定义边界条件，以使 Λ 中的滤波器响应被很好地定义。在相变的情况下，Gibbs 分布的极限不是唯一的，它取决于边界条件。然而，Julesz 系综和 Gibbs 系综之间的等价性甚至在相变时可以保持。相变的研究不在本书的范围。

首先我们定义一个函数

$$G(\mathbf{I}) = \begin{cases} 0, & \text{if } D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}^{(\alpha)}_{\text{obs}}) \le \varepsilon, \quad \forall \alpha。 \\ \sum_{\alpha=1}^{K} D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}^{(\alpha)}_{\text{obs}}), & 。 \end{cases}$$

当温度 T 变为 0 时，分布

$$q(\mathbf{I};\mathbf{h},T) = \frac{1}{Z(T)} \exp\{-G(\mathbf{I})/T\} \tag{5.8}$$

收敛到 Julesz 系综 Ω_K。$q(\mathbf{I};\mathbf{h},T)$ 可以由 Gibbs 采样器或其他 MCMC 算法采样。

采样 Julesz 系综

输入: 纹理图像 $\{\mathbf{I}_{\text{obs},i},\ i = 1,2,...,M\}$, K 个统计量 (滤波器) $\{F^{(1)}, F^{(2)}, ..., F^{(K)}\}$。
计算 $\mathbf{h}_{\text{obs}} = \{\mathbf{h}^{(\alpha)}_{\text{obs}}, \quad \alpha = 1,...,K\}$。
初始化一个合成图像 $\mathbf{I}$ (例如白噪声).
$T \leftarrow T_0$.
repeat
  随机选取一个位置 $v \in \Lambda$,
  for $\mathbf{I}(v) \in S$ **do**
    计算 $q(\mathbf{I}(v) \mid \mathbf{I}(-v);\mathbf{h},T)$.
  end for
  从 $q(\mathbf{I}(v) \mid \mathbf{I}(-v);\mathbf{h},T)$ 中随机抽取一个新的值 $\mathbf{I}(v)$.
  每次扫描后减少 T.
  当 $D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}^{(\alpha)}_{\text{obs}}) \le \varepsilon$，$\alpha = 1,2,...,K$ 时记录样本
until 收集到足够的样本.

在上面算法中，$q(\mathbf{I}(v) \mid \mathbf{I}(-v);\mathbf{h},T)$ 是像素值 $\mathbf{I}(v)$ 的条件概率，其余栅格灰度固定。在随机访问方案中，一次扫描翻转 $|\Lambda|$ 个像素，或在固定访问方案中翻转所有像素。

由于 Julesz 系综和 Gibbs 系综之间的等价性 [15]，来自 $q(\mathbf{I};\mathbf{h})$ 的采样图像和来自 $p(\mathbf{I};\beta)$ 的采样图像共享许多特征，它们不仅产生 bh 中的相同统计量，还产生由任何其他滤波器（线性或非线性）提取的统计量。有一个在一些计算机视觉工作中被误解的关键概念值得强调。Julesz 系综是 Gibbs 系综 $p(\mathbf{I};\beta)$ 的“典型”图像集合，而不是最小化 $p(\mathbf{I};\beta)$ 中 Gibbs 势（或能量）的“最可能”图像。

此算法可用于选择统计量 $\mathbf{h}$，如 [17] 中一样。也就是说，可以通过减少熵来追寻新的统计量，如在等式 (5.5) 中衡量的那样。[15] 中给出了深入的讨论。

5.5.4 实验：对 Julesz 系综进行采样

在这个实验中，我们选择了 [17] 中使用的所有 56 个线性滤波器（各种尺度和方向的 Gabor 滤波器和高斯滤波器的小拉普拉斯算子）。最大的滤波器窗口大小是 19×19 像素。我们选择 $\mathbf{h}$ 作为这些滤波器的边缘直方图，并使用算法 I 对 Julesz 系综进行采样。虽然每个纹理模式通常只需要一小部分滤波器，但我们还是使用通用滤波器组。通过集成所有这 56 个滤波器来学习 FRAME 模型几乎是不切实际的，因此使用更简单但等效的模型 $q(\mathbf{I};\mathbf{h})$ 计算更容易。

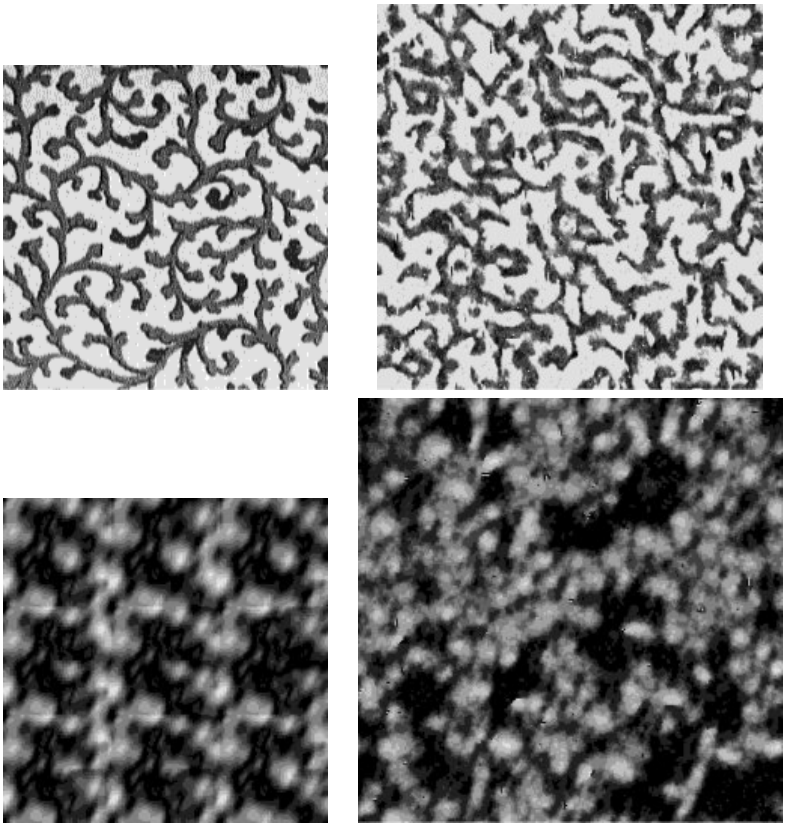


图 5.7: 左侧：观测纹理图像，右侧：与 56 个滤波器的观测直方图共享精确的直方图的合成纹理图像。©[2000] IEEE。经许可重印，来自参考文献 [16]。

我们在从各种源收集的一组大量的纹理图像上运行算法。结果显示在图 5.7中。左列表示观测纹理，右列显示大小为 256×256 像素的合成图像。对于这些纹理，从温度 $T_0 = 3$ 开始，在 20 到 100 次扫描之后，边缘统计量紧密匹配（每个直方图的误差小于 1%）。由于合成图像是有限的，匹配误差 ε 不能无限小。一般来说，我们设置 $\varepsilon \propto \frac{1}{|\Lambda|}$。

该实验表明 Gabor 滤波器和边缘直方图足以捕获各种均匀纹理模式。例如，图 5.7顶行中的布料图案具有非常规则的结构，在合成的纹理图像中可以很好地再现。这表明不同尺度的 Gabor 滤波器在不使用显式的联合直方图的情况下对齐。该对齐和高阶统计量可以通过滤波器的交互来说明。这个实验揭示了两个问题。

第一个问题在图 5.7底部的失败示例中得到证明。观测纹理模式具有较大的结构，其周期比滤波器组中最大的 Gabor 滤波器窗口还要长。结果，这些周期性模式在两个合成图像中蔓延，而基本纹理特征被很好地保留。

第二个问题与 Gibbs 采样器的有效性有关。如果我们放大棋盘图像以使棋盘的每个方格大小为 15×15 像素，那么必须选择具有窗口尺寸较大的滤波器来学习这种棋盘图案。使用算法 I 中的 Gibbs 采样器紧密匹配边缘统计量变得不可行，因为对于这种较大模式一次翻转一个像素是低效的。这建议我们应该寻找可以更新较大图像分块的更有效的采样方法。我们认为其他统计量匹配方法也会出现这个问题，例如最速下降 [2, 8]。Gibbs 采样器的低效也反映在其缓慢的混合速率上。在合成第一张图像之后，算法需要很长时间来生成与第一图像不同的图像。也就是说，马尔可夫链在 Julesz 系综中移动非常缓慢。

练习

问题 1. 在可数状态空间 Ω 中，考虑两个具有公共不变概率 π 的转移核 K_1 和 K_2。这里说*在 Pushin 顺序下*K_1 *主导*K_2，如果

$$K_1(x,y) \geq K_2(x,y), \forall x \neq y.$$

即 K_1 的非对角线元素不小于 K_2 对应的元素。（来自 K_1 的样本 $\{X_t^1\}_{t\geq 0}$ 的相关性低于来自 K_2 的样本 $\{X_t^2\}_{t\geq 0}$。证明 Metropolised Gibbs 采样器主导 Gibbs 采样器。（为了固定符号，假设我们用 $X=(x_1,x_2,...,x_n)$ 对概率 $\pi(X)$ 采样。两种情况都随机选择一个站点 x_i。）

问题 2. 我们考虑设计一个连续 Gibbs 采样器。为了简化问题，我们只研究一维概率 $\pi(x), x \in [a,b]$。对于高维空间，将应用相同的程序，因为每次我们仍然在实数区间 $[a,b]$ 中以一维条件概率进行采样。

我们将区间分成等长 $L=\frac{b-a}{K}$ 的 K 个单元区间。用 $B_1,B_2,...,B_K$ 表示这些单元区间。假定当前状态是 x，不失一般性，假设 $x \in B_1$。我们在其他 $K-1$ 个单元区间上均匀采样 $K-1$ 个中间点 $Z_2,Z_3,...,Z_K$，$Z_i \sim \text{Unif}(B_i)$。定义 $K-1$ 个点的总概率质量为

$$S=\pi(z_2)+\cdots+\pi(z_K)。$$

然后在 K 个点 $x,z_2,z_3,...,z_K$ 中通过离散的 Gibbs 采样器选择下一个状态 y。

$$y \in \{x,z_2,...,z_K\} \sim \frac{\pi(y)}{\pi(x)+S}$$

现在，当马尔可夫链处于 y 状态时，它可以以类似的方式回到状态 x。

1. 转移概率 $K(x,y)$ 是什么？[提示：将 $K-2$ 个其他变量视为需要积分的“阶梯石”或辅助变量。]

2. 如果我们将概率 $K(x,y)$ 视为提议概率，并应用 Metropolis-Hastings 步骤，证明该比率为

$$\frac{K(x,y)}{K(y,x)}=\frac{\pi(y)}{\pi(x)}$$

因此，接受比始终为 1。

3. 在上述结论中，容器数量 K 是否重要？如果不重要，那么你认为选择一个较大的 K 有什么好处？[想想这个，做个猜想] 我们应该在状态 x 和 y 时使用同一单元区间的集合吗？为什么？

参考文献

[1] Hirotugu Akaike. On entropy maximization principle. *Application of statistics*, 1977.

[2] CH Anderson and WD Langer. Statistical models of image texture. *Washington University Medical School*, 1997.

[3] Pierre Bremaud. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, volume 31. Springer, 1999.

[4] Charles Chubb and Michael S Landy. Orthogonal distribution analysis: A new approach to the study of texture perception. *Computational models of visual processing*, 12:394, 1991.

[5] Ingrid Daubechies et al. *Ten lectures on wavelets*, volume 61. SIAM, 1992.

[6] John G Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *JOSA A*, 2(7):1160–1169, 1985.

[7] Robert G Edwards and Alan D Sokal. Generalization of the fortuin-kasteleyn-swendsen-wang representation and monte carlo algorithm. *Physical review D*, 38(6):2009, 1988.

[8] A. Gagalowicz and S. D. Ma. Model driven synthesis of natural textures for 3d scenes. *Computers and Graphics*, 10(2):161–170, 1986.

[9] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 6:721–741, 1984.

[10] Hans-Otto Georgii. *Gibbs measures and phase transitions*, volume 9. Walter de Gruyter, 2011.

[11] Charles J Geyer and Elizabeth A Thompson. Annealing markov chain monte carlo with applications to ancestral inference. *Journal of the American Statistical Association*, 90(431):909–920, 1995.

[12] Jun S Liu and Ying Nian Wu. Parameter expansion for data augmentation. *Journal of the American Statistical Association*, 94(448):1264–1274, 1999.

[13] Robert H Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. *Physical Review Letters*, 58(2):86–88, 1987.

[14] Martin A Tanner and Wing Hung Wong. The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, 82(398):528–540, 1987.

[15] Ying Nian Wu, Song Chun Zhu, and Xiuwen Liu. Equivalence of julesz and gibbs texture ensembles. In *ICCV*, volume 2, pages 1025–1032, 1999.

[16] Song Chun Zhu, Xiu Wen Liu, and Ying Nian Wu. Exploring texture ensembles by efficient markov chain monte carlo-toward a "trichromacy" theory of texture. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(6):554–569, 2000.

[17] Song Chun Zhu, Ying Nian Wu, and David Mumford. Minimax entropy principle and its application to texture modeling. *Neural Computation*, 9(8):1627–1660, 1997.

[18] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126, 1998.

[19] Steven W. Zucker, Robert A. Hummel, and Azriel Rosenfeld. An application of relaxation labeling to line and curve enhancement. *IEEE Transactions on Computers*, 26(4):394–403, 1977.

第 5 章　聚类采样方法

‘‘不包含正/负关系的内部成分的形状在相同性质的形状下作用发挥的更好” - Keith Haring

介绍


Robert H. Swendsen

Swendsen-Wang 算法最初是为了解决在发生相变的临界温度或接近临界温度时，对下述 Ising 和 Potts 模型进行采样时的临界慢化问题而设计的。Fortuin and Kasteleyn [17] 已将 Potts 模型映射到了一种渗透模型 [7]。渗透模型是一种具有随机分布孔隙的多孔材料的模型，液体经过孔隙可以渗透。模型定义在一组节点 (例如在点阵上组织) 上，每个节点有一个从期望为 p 的伯努利随机分布独立采样的标签，其中标签 1 表示一个孔隙。两个具有标签 1 的相邻节点通过边自动连接。这样，通过对节点标签进行采样并自动连接具有标签 1 的相邻节点来获得随机节点聚类。


Jian-Sheng Wang

这是一种渗透模型。如果孔隙概率很大，那么会存在非零概率，形成将点阵左部的边缘连接到右部的边的聚类。在这种情况下，就认为是系统渗透。在本章中，我们回顾 Potts 模型和 Swendsen-Wang（SW）方法，还阐述了该方法的若干解释，定理和变体。本章最后讨论了子空间聚类以及目前最先进 C^4 算法。应用包括图像分割，稀疏动态分割和子空间聚类。

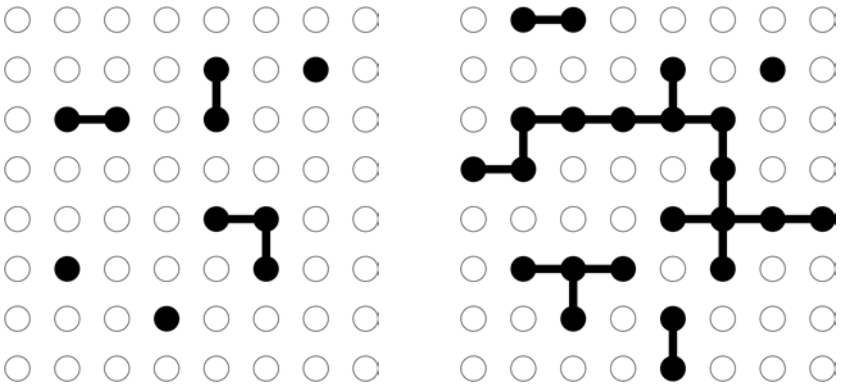


图 6.1: 渗透模型图解。左: 系统不渗透。右: 系统渗透。

6.1 Potts 模型和 Swendsen-Wang

设 $\mathbf{G} =< VE >$ 为一个邻接图，如有 4 个最近邻连接的栅格。每个顶点 $v_i \in V$ 有一个状态变量 x_i，其具有有限数量的标签（或颜色），$x_i \in \{12...\mathrm{L}\}$。标签 L 的总数是预先定义的。如果 $\mathbf{X} = (x_1, x_2, ..., x_{|V|})$ 表示图的标记，那么这个 Ising / Potts 模型是一个马尔可夫随机场，

$$\pi_{\mathrm{PTS}}(\mathbf{X}) = \frac{1}{Z} \exp\{- \sum_{<st>\in E} \beta_{st} \mathbf{1}(x_s \neq x_t)\}, \tag{6.1}$$

其中 $\mathbf{1}(x_s \neq x_t)$ 是一个布尔函数，如果观察到条件 $x_s \neq x_t$ 则等于 1，否则等于 0。如果可能的标签数量 $L = 2$，那么 π 就被称为 Ising 模型；如果 $L \geq 3$，那么就称为 Potts 模型。对于相邻顶点更喜欢相同颜色的铁磁体系统，通常我们认为 $\beta_{st} > 0$。Potts 模型及其扩展被用作许多贝叶斯推理任务中的先验概率。

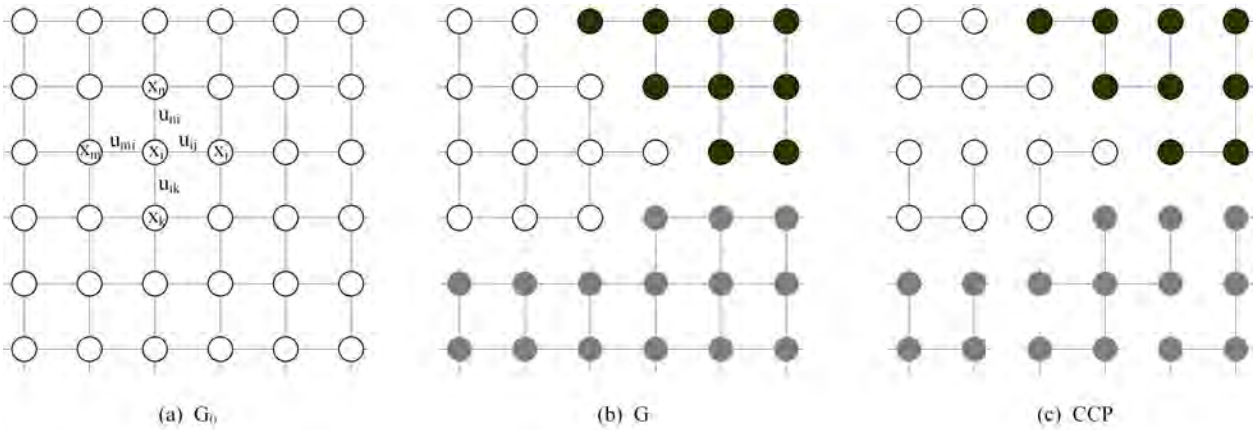


图 6.2: SW 方法解释。(a) 一个邻接矩阵 $\mathbf{G}$，其每条边 $e =< st >$ 有一个二进制变量 $\mu_e \in \{10\}$。(b) 图 $\mathbf{G}$ 的标记，其连接不同的颜色顶点的边会被移除。(c) 依概率关闭（b）中的一些边之后的许多连通分量。©[2000] Taylor & Francis。经许可重印，来自参考文献 [4]。

Swendsen-Wang（SW）算法在边中引入了一组辅助变量，如图 6.2 (a) 所示。

$$\mathbf{U} = \{\mu_e : \mu_e \in \{0,1\}, \forall e \in E\}。 \tag{6.2}$$

当且仅当 $\mu_e = 0$ 时，边 e 被断开（或关闭）。二进制变量 μ_e 服从以连接 $x_s x_t$ 的边 e 的标签为条件的伯努利分布。

$$\mu_e | (x_s x_t) \sim \mathrm{Bernoulli}(q_e \mathbf{1}(x_s = x_t)), \quad \text{with } q_e = 1 - e^{-\beta_{st}}, \forall e \in E。 \tag{6.3}$$

因此，如果 $x_s = x_t$，则以概率 q_e 有 $\mu_e = 1$；如果 $x_s \neq x_t$，则以概率 1 有 $\mu_e = 0$。在此设置下，SW 方法迭代以下两个步骤。

1. **聚类步骤**: 给定当前标签 $\mathbf{X}$，根据公式 (6.3) 对 $\mathbf{U}$ 中的辅助变量采样。首先依据 μ_e 关闭边 e 。也就是说，如果 $x_s \neq x_t$，则边 $e =< st >$ 确定关闭，如图 6.2 (b) 所示。现在边的全集由下式给出

$$E = E_{\mathrm{on}}(\mathbf{X}) \cup E_{\mathrm{off}}(\mathbf{X})。 \tag{6.4}$$

剩下的边以概率 $1-q_{st}=\exp(-\beta_{st})$ 关闭。边 e 根据 μ_e 分为“on”和“off”组。因此边的集和 $E_{\text{on}}(\mathbf{X})$ 进一步划分为，

$$E_{\text{on}}(\mathbf{X})=E_{\text{on}}(\mathbf{U},\mathbf{X})\cup E_{\text{off}}(\mathbf{U},\mathbf{X})。\tag{6.5}$$

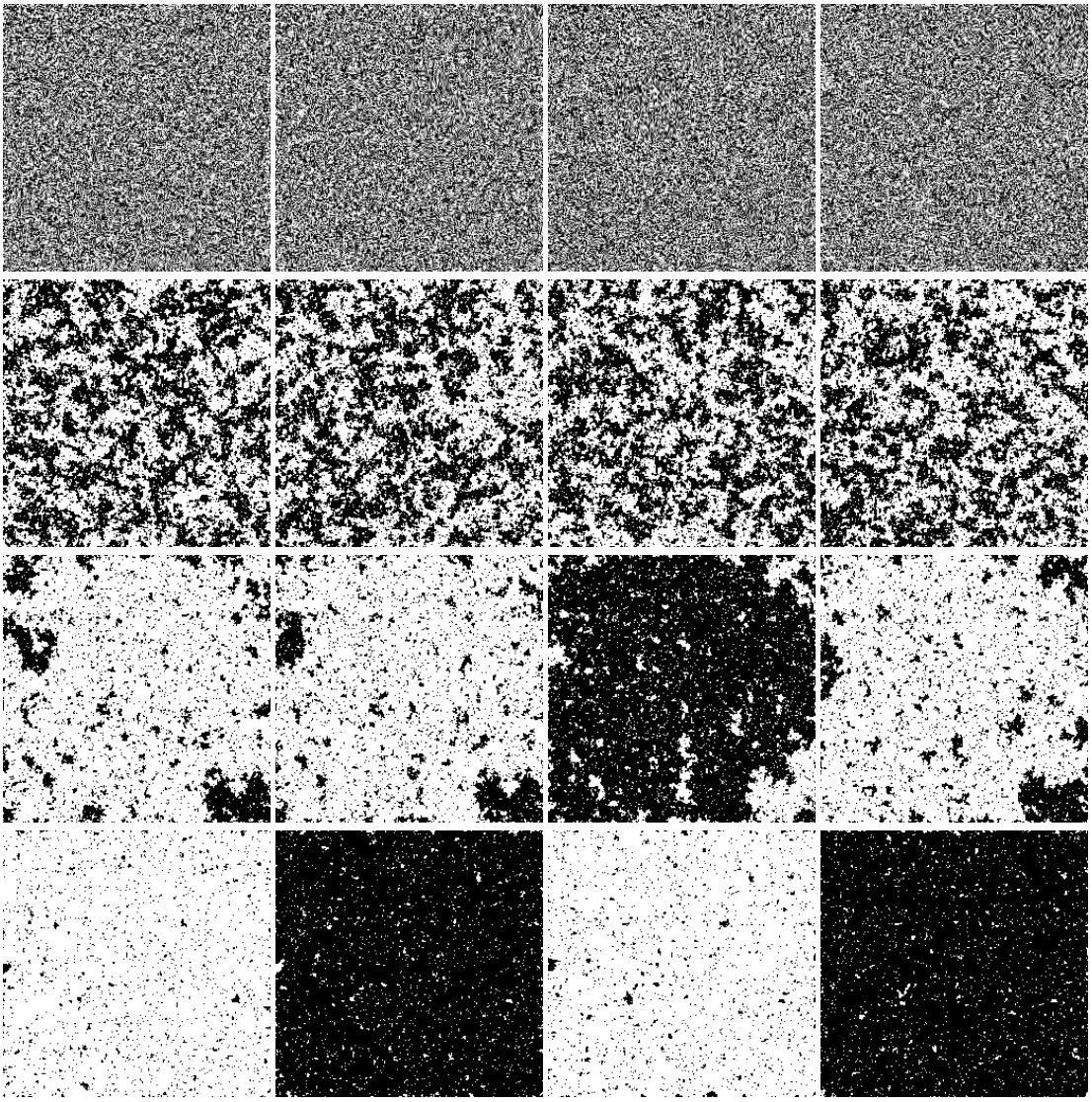


图 6.3: 对于 $\beta_{ij}=\beta$ 的不同值，通过 SW 算法在 Ising 模型上获得的连续样本。从上到下: β 分别等于 0.10.80.91.0。

如图 6.2 (c) 所示，$E_{\text{on}}(\mathbf{UX})$ 中的边形成许多连通分量。我们用下式表示在 $E_{\text{on}}(\mathbf{UX})$ 中连通分量的集合，

$$\text{CP}(\mathbf{U},\mathbf{X})=\{\text{cp}_i:\ i=1,2,...K,\ \text{with}\cup_{i=1}^{K}\text{cp}_i=V\}。\tag{6.6}$$

每个连通分量 cp_i 中的顶点保证具有相同的颜色。直观地，强耦合位点有更高概率被分组到一个连通分量。这些连通分量现在已经解耦。

2. **翻转步骤**: 随机选择一个连通分量 $V_o\in\text{CP}$，并为 V_o 中的所有顶点分配一个共同的颜色 ℓ。新标

签 ℓ 服从离散的均匀分布，

$$x_s = \ell \ \forall s \in V_o, \quad \ell \sim \text{uniform}\{1,2,...,\text{L}\}。 \tag{6.7}$$

在这步中，我们可以选择独立执行 CP(**U**) 中部分或全部连通分量的随机颜色翻转，因为他们是解耦的。通过这样做，图中所有可能的标记都在一步中连接，就像 Gibbs 采样器的一次扫描。

在 Wolff 的 [60] 修改版中，我们或者可以选择一个顶点 $v \in V$，并在 v 附近的边上进行伯努利试验后生成一个单连通分量。这在聚类步骤中节省了一些计算时间，但是较大的分量被选中的机会更大。

图 6.3显示了对于参数 $\beta_{ij} = \beta$ 的不同值，尺寸为 256×256 的栅格，在 Ising 模型上运行 SW 算法的结果。对于较小的 β 值，样本表现是随机的，而当 $\beta = 1$ 时，大多数节点具有相同的标记。存在 0.8 或 0.9 附近的值 β_0，在这里存在随机阶段和单色阶段之间的相变。值 $1/\beta_0$ 称为临界温度。

通过使用路径耦合技术，Cooper and Frieze [10] 已经证明，如果图 G 中的每个顶点连接到 $O(1)$ 个相邻点，则混合时间 τ(见公式 (6.40)) 是顶点数 N 的多项式，也就是说，每个顶点的连通性不会随着 V 的大小而增长。这通常会在计算机视觉问题中观察到，例如关于栅格或平面图。当图 G 完全连通时，混合时间在最坏情况下呈指数级变化 [23]。这种情况通常不会出现在视觉问题中。

在贝叶斯模型 $p(\mathbf{x}|I) \propto p(I|\mathbf{x})p(\mathbf{x})$ 中，Ising/Potts 模型 $p(\mathbf{x})$ 被用作先验模型，其中似然 $p(I|\mathbf{x})$ 衡量 $\mathbf{x}$ 对输入图像解释的如何。然而，在存在似然的情况下，SW 算法会减慢，这种情况也被称作外部场。这是因为聚类结果完全基于先验系数 β_{ij} 创建，而忽略了似然。Higdon 引入了一种名为部分解耦 [25] 的辅助变量方法，该方法考虑了增长聚类时的似然。然而，这种方法仍局限于具有 Ising / Potts 先验的模型。Huber[26] 为 Potts 模型(6.1)提出了一种边界链方法，其可以判断 SW 马尔可夫链何时收敛，从而获得精确或完美的采样 [45]。对于远低于或远高于临界水平的 $\frac{1}{\beta}$ 值，达到精确采样的步数为 $O(\log|E_o|)$ 的数量级。

6.2 SW 算法的解释

SW 算法有三种不同的解释，一个作为 Metropolis-Hastings 算法，一个作为具有辅助变量的数据增强方法，一个作为切片采样算法。简单起见，在本节中，我们假设我们正在使用 $\beta_{st} = \beta > 0 \forall < st > \in E$ 的同质 Potts 模型。

6.2.1 解释 1：Metropolis-Hastings 观点

SW 算法可以被解释为具有接受概率 1 的 Metropolis-Hastings 步骤。

图 6.4显示了在连通分量 V_o 的像素标签上不同的两个分区状态 A 和 B。假设当前状态是 A，其中 V_o 连接到作为剩余黑色顶点的 V_1。在 V_o 和 V_1 之间以某种概率被关闭的边形成一个切割，

$$C_{01} = C(V_0, V_1) = \{e =< s,t >: \ s \in V_0, t \in V_1\}。$$

切割由图 6.4十字符号表示。显然，通过 SW 聚类步骤到达连通分量 V_o 的方法有很多种。但是，每种方法都必须包括关闭 C_{01} 中的边。同样，如果马尔可夫链当前处于状态 B，它也有机会选择白色的连通分

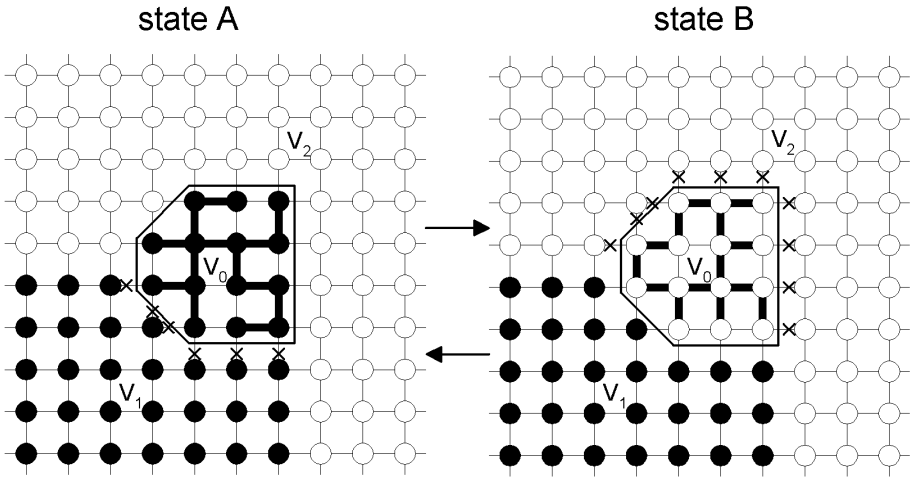


图 6.4: SW 算法在 Ising/Potts 模型的一个步骤中翻转了一块节点。©[2000]IEEE。经许可重印，来自参考文献 [3]。

量 V_o。我们将剩余的白色顶点表示为 V_2，并且 V_0 和 V_2 之间的切割是 V_0 and V_2 is

$$C_{02} = C(V_0, V_2) = \{e =< s,t >: \ s \in V_0, t \in V_2\}。$$

到目前为止，我们有两个对 V_o 有不同标记的状态 A 和 B。Metropolis-Hastings 算法用于在它们之间创建可逆移动。虽然难以计算提议概率 $Q(A \to B)$ 和 $Q(B \to A)$，但可以通过下式给出的化简来比较容易地计算它们的比率

$$\frac{Q(A \to B)}{Q(B \to A)} = \frac{(1-q)^{|C_{01}|}}{(1-q)^{|C_{02}|}} = (1-q)^{|C_{01}|-|C_{02}|}。 \tag{6.8}$$

在上式中，$|C_{01}|$ and $|C_{02}|$ 是集合的基数。换句话说，除了切口具有不同的尺寸，在状态 A 和 B 中选择 V_0 的概率是相同的。值得注意的是，概率比 $\pi(A)/\pi(B)$ 也由切口的大小决定，如下所示

$$\frac{\pi(A)}{\pi(B)} = \frac{e^{-\beta|C_{02}|}}{e^{-\beta|C_{01}|}} = e^{\beta(|C_{01}|-|C_{02}|)}。 \tag{6.9}$$

然后给出从 A 到 B 的接受概率

$$\alpha(A \to B) = \min(1, \frac{Q(B \to A)}{Q(A \to B)} \cdot \frac{\pi(B)}{\pi(A)}) = (\frac{e^{-\beta}}{1-q})^{|C_{01}|-|C_{02}|}。 \tag{6.10}$$

如果边缘概率选择为 $q = 1 - e^{-\beta}$，那么从 A 到 B 的提议总是被接受，$\alpha(A \to B) = 1$ 。由于 β 与温度倒数成正比，因此 q 在低温下倾向于 1，并且 SW 算法一次翻转一大块。因此，即使在临界温度下，SW 算法也可以快速融合。

证明 6.2.1 *(等式(6.8))*。设 $\mathbf{U}_A|(\mathbf{X} = A)$ 和 $\mathbf{U}_B|(\mathbf{X} = B)$ 分别是状态 A 和 B 中辅助变量的实现。根据翻转过程中的伯努利概率，分别得到两组连通分量 $\mathrm{CP}(\mathbf{U}_A|\mathbf{X} = A)$ 和 $\mathrm{CP}(\mathbf{U}_B|\mathbf{X} = B)$。我们将 $\mathbf{U}_A$ 分为两组，分别为 *on* 边和 *off* 边，

$$\mathbf{U}_A = \mathbf{U}_{A,\mathrm{on}} \cap \mathbf{U}_{A,\mathrm{off}}, \tag{6.11}$$

其中

$$\mathbf{U}_{A,\text{on}} = \{\mu_e \in \mathbf{U}_A : \mu_e = 1\},\ \mathbf{U}_{A,\text{off}} = \{\mu_e \in \mathbf{U}_A : \mu_e = 0\}$$

我们只对产生连通分量 V_o 的 $\mathbf{U}_A$'s（并且因此 $\text{CP}(\mathbf{U}_A|\mathbf{X}=A)$'s）感兴趣。我们在给定 A 的一个集合中搜集所有这样的 $\mathbf{U}_A$，

$$\Omega(V_o|A) = \{\mathbf{U}_A \ s。t。\ V_o \in \text{CP}(\mathbf{U}_A|\mathbf{X}=A)\}。 \tag{6.12}$$

为了使 V_o 成为 A 中的连通分量，必须切断（关闭）V_o 和 V_1 之间的所有边。我们表示一组关闭的边不是 ${}^{-}\mathbf{U}_{A,\text{off}}$ 切割的一部分，

$$\mathbf{U}_{A,\text{off}} = C(V_o,V_1) \cup {}^{-}\mathbf{U}_{A,\text{off}},\quad \forall \mathbf{U}_A \in \Omega(V_o|A)。 \tag{6.13}$$

同样地，我们在状态 B 中收集所有生成连通分量 V_o 的 $\mathbf{U}_B$，

$$\Omega(V_o|B) = \{\mathbf{U}_B \ s。t。\ V_o \in \text{CP}(\mathbf{U}_B|\mathbf{X}=B)\}。 \tag{6.14}$$

为了使 V_o 成为 $\mathbf{U}_B|B$ 中的连通分量，聚类步骤必须切断 V_o 和 V_2 之间的所有边。因此我们有

$$\mathbf{U}_B = \mathbf{U}_{B,\text{on}} \cup \mathbf{U}_{B,\text{off}},\ \text{其中} \tag{6.15}$$

$$\mathbf{U}_{B,\text{off}} = C(V_o,V_2) \cup {}^{-}\mathbf{U}_{B\text{off}},\quad \forall \mathbf{U}_B \in \Omega(V_o|B)。 \tag{6.16}$$

该公式中的一个关键是观察 $\Omega(V_o|A)$ 和 $\Omega(V_o|B)$ 存在的一对一映射。这是因为任何 $\mathbf{U}_A \in \Omega(V_o|A)$ 都具有一对一的相对应 $\mathbf{U}_B \in \Omega(V_o|B)$，其得到

$$\mathbf{U}_{B,\text{on}} = \mathbf{U}_{A,\text{on}},\ \mathbf{U}_{B,\text{off}} = {}^{-}\mathbf{U}_{A,\text{off}} \cup C(V_o,V_2)。 \tag{6.17}$$

也就是说，$\mathbf{U}_A$ 和 $\mathbf{U}_B$ 仅在切割 $C(V_o,V_1)$ 和 $C(V_o,V_2)$ 方面不同，其中所有辅助变量都关闭。因此，它们的连通分量都是相同的

$$\text{CP}(\mathbf{U}_A|\mathbf{X}=A) = \text{CP}(\mathbf{U}_B|\mathbf{X}=B) \tag{6.18}$$

类似地，任何 $\mathbf{U}_B \in \Omega(V_o|B)$ 都具有一一的相对应 $\mathbf{U}_A \in \Omega(V_o|A)$。

假设我们现在从 $\text{CP}(\mathbf{U}_A|\mathbf{X}=A)$ 中的所有连通分量中选择具有均匀概率的 $V_o \in \text{CP}(\mathbf{U}_A|\mathbf{X}=A)$。那么在状态 A 中选择 V_o 的概率为

$$q(V_o|A) = \sum_{\mathbf{U}_A \in \Omega(V_o|A)} \frac{1}{|\text{CP}(\mathbf{U}_A|\mathbf{X}=A)|} \prod_{e \in \mathbf{U}_{A,\text{on}}} q_e \prod_{e \in {}^{-}\mathbf{U}_{A,\text{off}}} (1-q_e) \prod_{e \in C(V_o,V_1)} (1-q_e)。 \tag{6.19}$$

类似地，在状态 B 中选择 V_o 的概率为

$$q(V_o|B) = \sum_{\mathbf{U}_B \in \Omega(V_o|B)} \frac{1}{|\text{CP}(\mathbf{U}_B|\mathbf{X}=B)|} \prod_{e \in \mathbf{U}_{B,\text{on}}} q_e \prod_{e \in {}^{-}\mathbf{U}_{B,\text{off}}} (1-q_e) \prod_{e \in C(V_o,V_2)} (1-q_e)。 \tag{6.20}$$

用等式 *(6.20)* 除以等式 *(6.19)*，通过 $\Omega(V_o|A)$ 和 $\Omega(V_o|B)$ 之间的一对一的对应关系化简，我们得到等式 *(6.8)* 中的比率。在 $C(V_o,V_1)=\emptyset$，$\prod_{e \in C(V_o,V_1)}(1-q_e)=1$ 的特殊情况下。

注意，这个证明对于任意设计的 q_e 都是成立的。

当连接两个状态的路径有两条时，会出现稍微复杂的情况，如图 6.5所示。

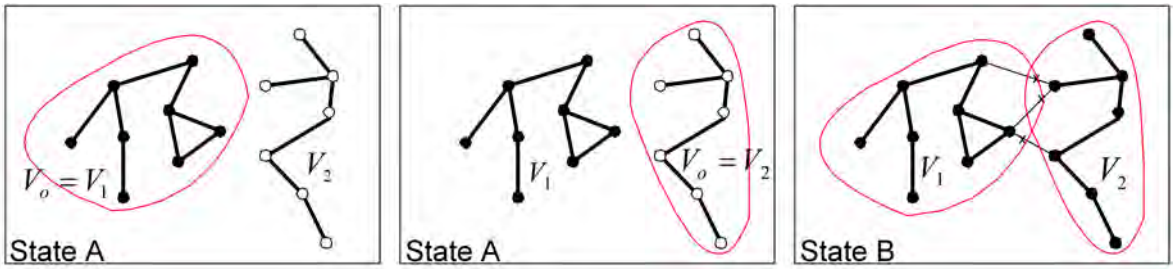


图 6.5: 状态 A 中具有两个子图 V_1 和 V_2，它们在状态 B 中合并。在这种情况下，从状态 A 到状态 B 有两条路径，一条通过选择 $V_o = V_1$，另一条通过选择 $V_o = V_2$。改编自 Barbu and Zhu [3]。

路径 1。选择 $V_o = V_1$。在状态 A 中，选择一个新标签 $\ell = 2$，即将 V_o 合并到 V_2 中，而在状态 B 中相反，选择一个新标签 $\ell = 1$，即从 V_2 中拆分 V_o。

路径 2。在状态 A 中，选择一个新标签 $\ell = 1$，即将 V_o 合并为 V_1，反之则合并到状态 B，选择 $\ell = 2$，即从 V_1 分割 V_o。在这种情况下，提议概率比为，

$$\frac{Q(B \to A)}{Q(A \to B)} = \frac{q(V_o = V_1|B)q(\mathbf{X}_{V_o} = 2|V_o, B) + q(V_o = V_2|B)q(\mathbf{X}_{V_o} = 1|V_o, B)}{q(V_o = V_1|A)q(\mathbf{X}_{V_o} = 1|V_o, A)) + q(V_o = V_2|A)q(\mathbf{X}_{V_o} = 2|V_o, A)}。 \tag{6.21}$$

在状态 A 中，两个路径的切割 $C(V_o, V_\ell \setminus V_o) = \emptyset$，并且在状态 B 中，切割为两个路径的 $C(V_1, V_2)$。在命题 6.6之后，选择 $V_o = V_1$ 和 $V_o = V_2$ 的概率比相等并由下式给出，

$$\frac{q(V_o = V_1|A)}{q(V_o = V_1|B)} = \frac{1}{\prod_{e \in C(V_1, V_2)}(1 - q_e)} = \frac{q(V_o = V_2|A)}{q(V_o = V_2|B}。 \tag{6.22}$$

一旦选择 V_o，使得 $V_o = V_1$ 或 $V_o = V_2$，A 和 B 的剩余划分是相同的，并且由 $\mathbf{X}_{V \setminus V_o}$ 表示。在提出 V_o 的新标签时，我们很容易观察到

$$\frac{q(\mathbf{X}_{V_o} = 2|V_o = V_1, B)}{q(\mathbf{X}_{V_o} = 1|V_o = V_2, A)} = \frac{q(\mathbf{X}_{V_o} = 1|V_o = V_2, B)}{q(\mathbf{X}_{V_o} = 2|V_o = V_1, A)}。 \tag{6.23}$$

在这种方法下，接受率仍为 1。

6.2.2 解释 2: 数据增强

第二种解释遵循 Edward 和 Sokal[14]，他将 Potts 模型扩展为 $\mathbf{X}$ 和 $\mathbf{U}$ 的联合概率，

$$p_{\text{ES}}(\mathbf{X}, \mathbf{U}) = \frac{1}{Z} \prod_{e=<s,t> \in E} [(1-\rho)\mathbf{1}(\mu_e = 0) + \rho \mathbf{1}(\mu_e = 1) \cdot \mathbf{1}(x_s = x_t)] \tag{6.24}$$

$$= \frac{1}{Z}[(1-\rho)^{|E_{\text{off}}(\mathbf{U})|} \cdot \rho^{|E_{\text{on}}(\mathbf{U})|}] \cdot \prod_{<s,t> \in E_{\text{on}}(\mathbf{U})} \mathbf{1}(x_s = x_t) \tag{6.25}$$

等式 (6.25) 中的第二个因素实际上是对 $\mathbf{X}$ 和 $\mathbf{U}$ 的硬约束。让 X 的空间为 Let the space of $\mathbf{X}$ be

$$\Omega = \{1,2,...,\mathrm{L}\}^{|V|}。 \tag{6.26}$$

在这种硬约束下，标记 $\mathbf{X}$ 被缩减为子空间 $\Omega_{\mathrm{CP}(\mathbf{U})}$，其中每个连通分量必须具有相同的标签，

$$\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})} \mathbf{1}(x_s = x_t) = \mathbf{1}(\mathbf{X} \in \Omega_{\mathrm{CP}(\mathbf{U})})。 \tag{6.27}$$

联合概率 $p_{\mathrm{ES}}(\mathbf{X},\mathbf{U})$ 观察到两个很好的属性，这两个属性都很容易验证。

命题 6.1 *Potts* 模型是联合概率的边际概率，

$$\sum_{\mathbf{U}} p_{\mathrm{ES}}(\mathbf{X},\mathbf{U}) = \pi_{\mathrm{PTS}}(\mathbf{X})。 \tag{6.28}$$

另一个边际概率是随机聚类模型 π_{RCM}，

$$\sum_{\mathbf{X}} p_{\mathrm{ES}}(\mathbf{X},\mathbf{U}) = \pi_{\mathrm{RCM}}(\mathbf{U}) = \frac{1}{Z}(1-\rho)^{|E_{\mathrm{off}}(\mathbf{U})|}\cdot\rho^{E_{\mathrm{on}}(\mathbf{U})}\mathrm{L}^{|\mathrm{CP}(\mathbf{U})|}。 \tag{6.29}$$

证明 6.2.2 记 $U = \{\mu_1,...,\mu_{|E|}\}$，且 $(1-\rho)\mathbf{1}(\mu_e=0)+\rho\mathbf{1}(\mu_e=1)\cdot\mathbf{1}(x_s=x_t) = f(\mu_e,x_s,x_t)$ 对于 $e=<s,t>$，我们有

$$\begin{aligned}\sum_{\mathbf{U}} p_{\mathrm{ES}}(\mathbf{X},\mathbf{U}) &= \frac{1}{Z}\sum_{\mu_1=0}^{1}...\sum_{\mu_{|E|}=0}^{1}\prod_{e=<s,t>\in E} f(\mu_e,x_s,x_t)\\ &= \frac{1}{Z}\sum_{\mu_1=0}^{1}...\sum_{\mu_{|E|}=0}^{1} f(\mu_1,x_{s_1},x_{t_1})f(\mu_2,x_{s_2},x_{t_2})\cdot...\cdot f(\mu_{|E|},x_{s_{|E|}},x_{t_{|E|}}),\end{aligned} \tag{6.30}$$

其中 $<s_1,t_1>$ 是对应于 μ_1 的边，$<s_2,t_2>$ 到 μ_2，等等。因此我们有

$$\begin{aligned}\sum_{\mathbf{U}} p_{\mathrm{ES}}(\mathbf{X},\mathbf{U}) &= \frac{1}{Z}\left[\sum_{\mu_1=0}^{1} f(\mu_1,x_{s_1},x_{t_1})\right]\cdot...\cdot\left[\sum_{\mu_{|E|}=0}^{1} f(\mu_{|E|},x_{s_{|E|}},x_{t_{|E|}})\right]\\ &= \frac{1}{Z}\prod_{e=<s,t>\in E}\sum_{\mu_e=0}^{1}[(1-\rho)\mathbf{1}(\mu_e=0)+\rho\mathbf{1}(\mu_e=1)\cdot\mathbf{1}(x_s=x_t)]\\ &= \frac{1}{Z}\prod_{e=<s,t>\in E}[(1-\rho)+\rho\cdot\mathbf{1}(x_s=x_t)] = \pi_{\mathrm{PTS}}(\mathbf{X})。\end{aligned} \tag{6.31}$$

对于第二个边际，我们观察到

$$\begin{aligned}\sum_{\mathbf{X}} p_{\mathrm{ES}}(\mathbf{X},\mathbf{U}) &= \sum_{\mathbf{X}}\frac{1}{Z}[(1-\rho)^{|E_{\mathrm{off}}(\mathbf{U})|}\cdot\rho^{|E_{\mathrm{on}}(\mathbf{U})|}]\cdot\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})}\mathbf{1}(x_s=x_t)\\ &= \frac{1}{Z}[(1-\rho)^{|E_{\mathrm{off}}(\mathbf{U})|}\cdot\rho^{|E_{\mathrm{on}}(\mathbf{U})|}]\cdot\sum_{\mathbf{X}}\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})}\mathbf{1}(x_s=x_t)\end{aligned} \tag{6.32}$$

连通分量的所有节点 $c_i \in \mathrm{CP}(\mathbf{U})$ 必须有相同的标签，否则乘积 $\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})} \mathbf{1}(x_s = x_t) = 0$。此外，每个连通分量可以用 L 标签之一独立标记，因此

$$\sum_{\mathbf{X}} \prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})} \mathbf{1}(x_s = x_t) = L^{|\mathrm{CP}(\mathbf{U})|}$$

命题 6.2 $p_{\mathrm{ES}}(\mathbf{X},\mathbf{U})$ 条件概率是

$$p_{\mathrm{ES}}(\mathbf{U}|\mathbf{X}) = \prod_{<s,t>\in E} p(\mu_e|x_s,x_t), \quad \text{with } p(\mu_e|x_s,x_t) = \mathrm{Bernoulli}(\rho \mathbf{1}(x_s = x_t)), \tag{6.33}$$

$$p_{\mathrm{ES}}(\mathbf{X}|\mathbf{U}) = \mathrm{unif}[\Omega_{\mathrm{CP}(\mathbf{U})}] = (\frac{1}{\mathrm{L}})^{|\mathrm{CP}(\mathbf{U})|} \text{ for } \mathbf{X} \in \Omega_{\mathrm{CP}(\mathbf{U})};\ 0 \text{ otherwise} \tag{6.34}$$

证明 6.2.3 我们得到 $p_{\mathrm{ES}}(\mathbf{U}|\mathbf{X}) \propto \prod_{<s,t>\in E} p(\mu_{st}|x_s,x_t)$ 与 $p(\mu_{st}|x_s,x_t) = (1-\rho)\mathbf{1}(\mu_{st}=0) + \rho\mathbf{1}(\mu_{st}=1)\cdot\mathbf{1}(x_s = x_t)$。因此

$$p(\mu_{st}|x_s,x_t) \propto (1-\rho)\mathbf{1}(\mu_{st}=0) + \rho\mathbf{1}(\mu_{st}=1)\cdot\mathbf{1}(x_s=x_t) = \begin{cases} (1-\rho)\mathbf{1}(\mu_{st}=0) & if\ x_s \neq x_t \\ (1-\rho)\mathbf{1}(\mu_{st}=0) + \rho\mathbf{1}(\mu_{st}=1) & if\ x_s = x_t \end{cases} \tag{6.35}$$

因此，在本例中，如果 $x_s \neq x_t$ 我们有 $p(\mu_{st}|x_s,x_t) \propto (1-\rho)\mathbf{1}(\mu_{st}=0)$ so $p(\mu_{st}=1|x_s,x_t) = 0$。如果 $x_s = x_t$ 我们有 $p(\mu_{st}|x_s,x_t) \propto (1-\rho)\mathbf{1}(\mu_{st}=0) + \rho\mathbf{1}(\mu_{st}=1)$ 所以在本例中 $p(\mu_{st}=1|x_s,x_t) = \rho$，$p(\mu_{st}=0|x_s,x_t) = 1-\rho$。这也就证明了

$$p_{\mathrm{ES}}(\mathbf{U}|\mathbf{X}) \propto \prod_{<s,t>\in E} \mathrm{Bernoulli}(\rho\mathbf{1}(x_s = x_t)) \tag{6.36}$$

由于右边是一个适当的的概率，我们有等式。(6.32)成立。

对于第二个条件概率，我们有

$$\begin{aligned} p_{\mathrm{ES}}(\mathbf{X}|\mathbf{U}) &= \frac{1}{Z_1}\prod_{e\in E}[(1-\rho)\mathbf{1}(\mu_e=0) + \rho\mathbf{1}(\mu_e=1)\cdot\mathbf{1}(x_s=x_t)] \\ &= \frac{1}{Z_1}\prod_{e\in E_{\mathrm{on}}(\mathbf{U})}[\rho\cdot\mathbf{1}(x_s=x_t)]\prod_{e\in E_{\mathrm{off}}(\mathbf{U})}(1-\rho) \\ &= \frac{\prod_{e\in E_{\mathrm{on}}(\mathbf{U})}(1-\rho)\prod_{e\in E_{\mathrm{off}}(\mathbf{U})}\rho}{Z_1}\prod_{e\in E_{\mathrm{on}}(\mathbf{U})}\mathbf{1}(x_s=x_t) = \frac{1}{Z_2}\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})}\mathbf{1}(x_s=x_t) \end{aligned} \tag{6.37}$$

连通分量 $c_i \in \mathrm{CP}(\mathbf{U})$ 的所有节点必须具有相同的标签，否则乘积 $\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})} \mathbf{1}(x_s = x_t) = 0$。此外，每个连通分量可以用 L 标签之一独立标记，因此

$$p_{\mathrm{ES}}(\mathbf{X}|\mathbf{U}) = \begin{cases} (1/L)^{|\mathrm{CP}(\mathbf{U})|} & if\ \mathbf{X} \in \Omega_{\mathrm{CP}(\mathbf{U})} \\ 0 & 否则 \end{cases}$$

因此，可以将两个 SW 步骤视为对两个条件概率进行采样。。

1. 聚类步骤: $\mathbf{U} \sim p_{\mathrm{ES}}(\mathbf{U}|\mathbf{X})$，即 $\mu_e|(x_s,x_t) \sim \mathrm{Bernoulli}(\rho\mathbf{1}(x_s=x_t))$。

2. 翻转步骤: $\mathbf{X} \sim p_{\mathrm{ES}}(\mathbf{U}|\mathbf{X})$，即 $\mathbf{X}(\mathrm{cp}_i) \sim \mathrm{Unif}\{1,2,...,\mathrm{L}\}, \forall \mathrm{cp}_i \in \mathrm{CP}(\mathbf{U})$。

由于 $(\mathbf{X},\mathbf{U}) \sim p_{\mathrm{ES}}(\mathbf{X},\mathbf{U})$，在舍弃辅助变量 $\mathbf{U}$ 之后，$\mathbf{X}$ 遵循 $p_{\mathrm{ES}}(\mathbf{X},\mathbf{U})$ 的边际分布。目标已实现，并且

$$\mathbf{X} \sim \pi_{\mathrm{PTS}}(\mathbf{X})。 \tag{6.38}$$

这种数据增强方法（Tanner 和 Wong[52]）的好处是，在给定辅助变量的情况下，连通分量的标签完全解耦（独立）。如 As $\rho = 1 - e^{-\beta}$，如果 Potts 模型中的温度较高则倾向于选择较小的簇，而如果温度较低则倾向于选择较大的簇。因此，它克服了单站点 Gibbs 采样器的耦合问题

6.3 一些理论成果

让马尔可夫链具有核 $\mathcal{K}$ 和初始状态 $\mathbf{X}_o$，在 t 步之后状态遵循概率 $p_t = \delta(\mathbf{X}-\mathbf{X}_o)\mathcal{K}^t$，其中 $\delta(\mathbf{X}-\mathbf{X}_o)$ 由下式给出

$$\delta(\mathbf{X}-\mathbf{X}_o) = \begin{cases} 1, & \text{if } \mathbf{X} = \mathbf{X}_o \\ 0, & \end{cases}$$

其是初始概率。马尔可夫链的收敛通常由整体差异来衡量

$$||p_t - \pi||_{\mathrm{TV}} = \frac{1}{2}\sum_{\mathbf{X}} |p_t(\mathbf{X}) - \pi(\mathbf{X})|。 \tag{6.39}$$

马尔可夫链的混合时间由下式定义

$$\tau = \max_{\mathbf{X}_o} \min\{t : \ ||p_t - \pi||_{\mathrm{TV}} \leq \varepsilon\}。 \tag{6.40}$$

τ 是 ε 的函数，就顶点数和连通性而言，图复杂度 $M = |\mathbf{G}_o|$ 。如果 $\tau(M)$ 是多项式或对数，则称马尔可夫链快速混合。

根据经验，发现 SW 方法能快速混合。最近，一些关于其性能的分析结果浮出水面。Cooper 和 Frieze[10] 使用路径耦合技术来证明 SW 在稀疏连接的图上快速混合。

Theorem 6.3 (Cooper 和 Frieze 1999) *设 $n = |V|$ 和 Δ 是任何单个顶点的最大边数，L 是 Potts 模型中的颜色数。如果 $\mathbf{G}$ 是树，则对于任何 β 和 L，SW 混合时间是 $O(n)$。如果 $\Delta = O(1)$，则存在 $\rho_o = \rho(\Delta)$，使得如果 $\rho \leq \rho_o$（即高于一定的温度），那么 SW 对所有 L 有多项式混合时间。*

Gore 和 JerrumA[23] 在完整图上构建了一个否定的案例。

Theorem 6.4 (Gore 和 Jerrum 1997) *如果 $\mathbf{G}$ 是完整图并且 $\mathrm{L} > 2$，那么对于 $\beta = \frac{2(\mathrm{L}-1)\ln(\mathrm{L}-1)}{n(\mathrm{L}-2)}$，SW 不会快速混合。*

在图像分析应用程序中，图像经常会观测到 Copper-Frieze 情况，且远未不够完整。

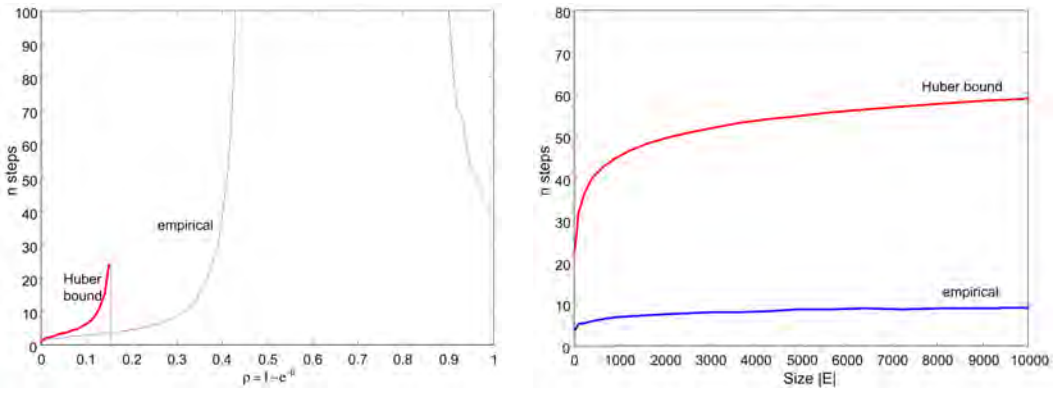
最近，在极端温度下，Huber[26] 为 Potts 模型开发了一种精确的采样技术。这种方法设计了一个边界链，它假设每个顶点 $s \in V$ 都有一组用全集 $|S_s| = \mathrm{L}$, $\forall s$ 初始化的颜色 S_s。辅助变量 μ_e 的伯努利概率

变为

$$\mathbf{U}^{\text{bd}} = \{\mu_e^{\text{bd}} : \mu_e^{\text{bd}} \in \{0,1\},\ \mu_e \sim \text{Bernoulli}(\rho \mathbf{1}(S_s \cap S_t \neq \emptyset))\}。 \tag{6.41}$$

因此，$\mathbf{U}^{\text{bd}}$ 在原始 SW 链中具有比 $\mathbf{U}$ 更多的边缘，即 $\mathbf{U} \subset \mathbf{U}^{\text{bd}}$。当 $\mathbf{U}^{\text{bd}}$ 折叠到 $\mathbf{U}$ 时，从任意初始状态开始的所有 SW 链都已折叠到当前单链中。因此，链必须收敛（精确采样）。折叠的步骤称为“耦合时间”。

Theorem 6.5 (Huber 2002) 设 $n = |V|$ *and* $m = |E|$。在较高的温度下，$\rho < \frac{1}{2(\Delta-1)}$，边界链完全按时间 $O(\ln(2m))$ 耦合，概率至少为 $1/2$。在较低的温度下，$\rho \geq 1 - \frac{1}{m\mathrm{L}}$，则偶和时间是 $O((m\mathrm{L})^2)$，概率至少为 $1/2$。



(a) (b)

图 6.6: 耦合时间经验图和 Ising 模型的 Huber 界限。

实际上，the Huber 界限并不像人们预期的那样严格。图。 6.6(a) 绘制了在 Ising 模型上具有环面边界条件的 5×5 点阵上的结果，经验耦合时间对 $\rho = 1 - e^{-\beta}$。在临界温度附近的耦合时间很长（未示出）。高温 Huber 边界从 $\rho_o = 0.16$ 开始，用短曲线表示。低温的界限以 $\rho_o > 0.99$ 开始，不可见。图。6.6。(b) 绘制了 $\rho = 0.15$ 时的耦合时间与图形尺寸 $m = |E|$ 的关系和 Huber 界限。

尽管上面讨论了令人鼓舞的成功，但 SW 方法在两个方面受到限制。

1. 它仅对 Ising 和 Potts 模型有效，而且还需要知道着色数 L。在许多应用中，例如图像分析，L 是对象（或图像区域）的数量，其必须从输入数据推断。

2. 它在存在外部字段（即输入数据）的情况下快速减速。例如，在图像分析问题中，我们的目标是从输入图像 $\mathbf{I}$ 推断出标签 $\mathbf{X}$，并且目标概率是贝叶斯后验概率，其中 $\pi_{\text{PTS}}(\mathbf{X})$ 被用作先验模型，

$$\pi(\mathbf{X}) = \pi(\mathbf{X}|\mathbf{I}) \propto \mathscr{L}(\mathbf{I}|\mathbf{X})\pi_{\text{PTS}}(\mathbf{X})。 \tag{6.42}$$

在这里，$\mathscr{L}(\mathbf{I}|\mathbf{X})$ 是似然模型，例如每个着色 $c = 1,2,...,\mathrm{L}$ 的独立高斯分布 $N(\bar{\mathbf{I}}_c, \sigma_c^2)$，

$$\mathscr{L}(\mathbf{I}|\mathbf{X}) \propto \prod_{c=1}^{\mathrm{L}} \prod_{x_i=c} \frac{1}{\sqrt{2\pi}\sigma_c} \exp\{-\frac{(\mathbf{I}(v_i) - \bar{\mathbf{I}}_c)^2}{2\sigma_c^2}\}。 \tag{6.43}$$

减速部分归因于以下事实：辅助变量的伯努利概率 $\rho = 1 - e^{-\beta}$ 是独立于输入图像计算的。

6.4 任意概率的 Swendsen-Wang 切割


Adrian Barbu

" 在本节中，我们将 SW 算法推广到 Metropolis-Hastings 方法 [24, 39] 的任意概率。Swendsen-Wang Cuts(SWC) 方法迭代三个步骤：(i) 由数据驱动的聚类步骤，(ii) 可以产生新标签的标签翻转步骤，以及 (iii) 所提出的标签的接受步骤。该算法的一个关键特征是计算接受概率的简单公式"。

我们在以下小节中描述了这三个步骤，然后展示了它如何简化为 Potts 模型的原始 SW 算法。

我们用图像分割示例说明了算法。图 6.7。(a) 是点阵 Λ 上的输入图像 $\mathbf{I}$，其在（b）中被分解成多个" 超像素" 以减小预处理阶段中的图形尺寸。每个超像素具有几乎恒定的强度并且是图中的顶点。如果它们的超像素共享边界，则连接两个顶点。图（c）是使用 SWC 算法优化贝叶斯概率 $\pi(\mathbf{X}) = \pi(\mathbf{X}|\mathrm{I})$ 的结果（详见第 (6.6) 节）。结果 $\mathbf{X}$ 为每个封闭区域中的所有顶点指定一致的颜色，这有希望与场景中的对象相对应。请注意对象或颜色的数量 L 是未知的，我们不区分标签的排列。

6.4.1 步骤 1: 数据驱动的聚类

我们首先使用边缘 $\mathbf{U} = \{\mu_e : e =< s,t >\in E\}$ 上的一组二元变量来扩充邻接图 $\mathbf{G}$，如在原始 SW 方法中的那样。每个 μ_e 遵循伯努利概率，取决于两个顶点 x_s 和 x_t 的当前状态，

$$\mu_e|(x_s,x_t) \sim \text{Bernoulli}(q_e\mathbf{1}(x_s = x_t)),\ \forall < s,t >\in E。 \tag{6.44}$$

这里，q_e 是边缘上的概率 $e =< s,t >$，它表示两个顶点 s 和 t 具有相同标签的可能性。在贝叶斯推理中，目标 $\pi(\mathbf{X})$ 是后验概率，q_e 可以更好的通知数据。

对于图像分割示例，基于 s 和 t（或其局部邻域）处的图像强度之间的相似性来计算 q_e，并且可以是 $\pi(\mathbf{X}|\mathrm{I})$ 的边际概率的近似值，

$$q_e = q(x_s = x_t|\mathbf{I}(s),\mathbf{I}(t)) \ \approx\ \pi(x_s = x_t|\mathrm{I})。 \tag{6.45}$$

使用所谓的判别方法计算 $q(x_s = x_t|\mathrm{I}(v_s),\mathrm{I}(v_t))$ 有很多种方法，但是详细讨论超出了本书的范围。

这种方法适用于任何 q_e，但较好的近似将告知聚类步骤，并根据经验实现更快的收敛。图。 6.8显

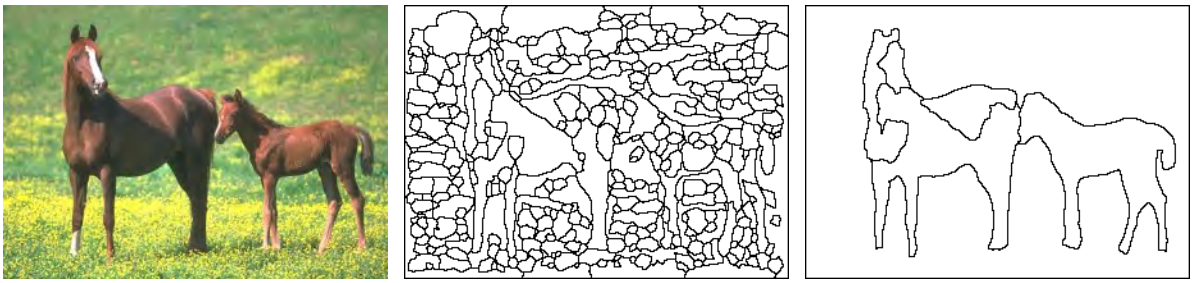

(a)。Input image　(b)。Superpixels　(c)。Segmentation

图 6.7: 图像分割示例。(a) 输入图像。(b) 通过边缘检测获得超像素，然后进行边缘跟踪和轮廓闭合。每个超像素都是图中的顶点。Each superpixel is a vertex in the graph。(c) 分割（标记）结果，其中每个封闭区域被分配颜色或标签。由 Barbu 和 Zhu[4] 提供。

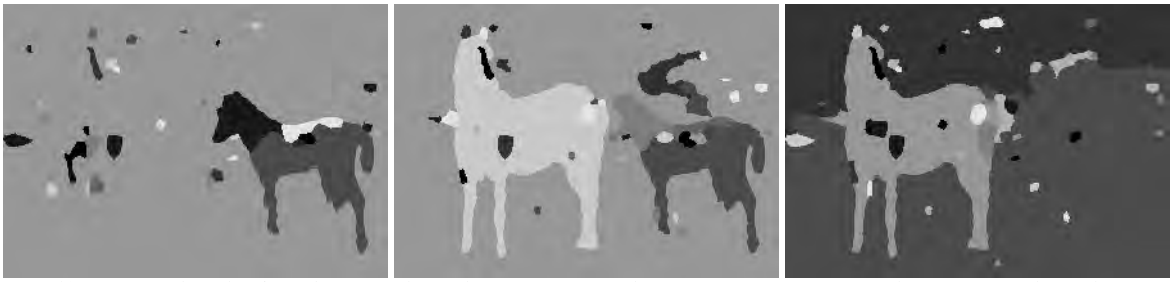


图 6.8: 使用判别边缘概率计算马图像的连通分量的三个示例。假设 $\mathbf{X}$ 是所有顶点的均匀颜色 $\mathbf{X} = c$ 。由 Barbu 和 Zhu[4]。

提供

示了马图像的几个聚类示例。在这些检查文件中，我们将所有顶点设置为相同的颜色（$\mathbf{X} = c$）并独立地采样边缘概率，

$$\mathbf{U}|\mathbf{X} = c \sim \prod_{<s,t>\in E} \text{Bernoulli}(q_e)。 \tag{6.46}$$

CP($\mathbf{U}$) 中的连接组件显示为不同的颜色。我们重复三次聚类步骤。我们可以看到，边缘概率导致有意义的聚类，这些聚类对应于图像中的不同对象。使用恒定边缘概率不能观察到这种效应。

6.4.2 Step 2: 颜色翻转

令 $\mathbf{X} = (x_1, x_2, ..., x_{|V|})$ 为当前着色状态。边缘变量 $\mathbf{U}$ 在 $\mathbf{X}$ 上有条件的采样，并将 $\mathbf{X}$ 分解为许多连通分量

$$\text{CP}(\mathbf{U}|\mathbf{X}) = \{\text{cp}_i : i = 1, 2, ..., N(\mathbf{U}|\mathbf{X})\}。 \tag{6.47}$$

假设我们选择一个颜色为 $\mathbf{X}_{V_o} = \ell \in \{1, 2, ..., L\}$ 的连通分量 $V_o \in \text{CP}(\mathbf{U}|\mathbf{X})$，并将其颜色指定为具有概率 $q(l'|V_o, \mathbf{X})$(稍后设计) 的 $\ell' \in \{1, 2, ..., L, L+1\}$。我们获得了一个新的状态 $\mathbf{X}'$。有三种情况，如图 6.9所示。

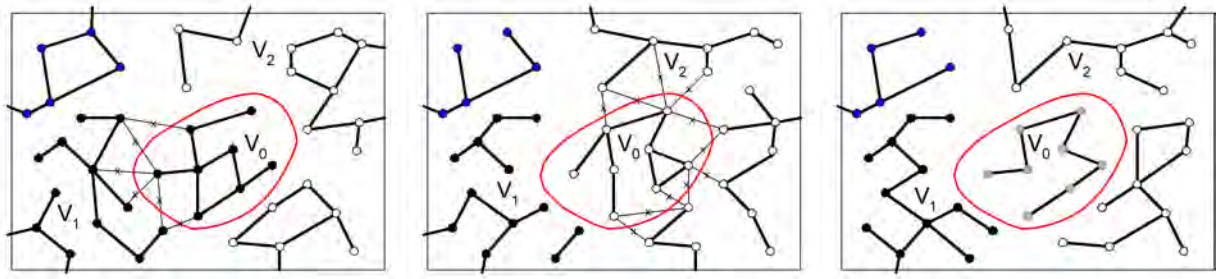


图 6.9: 三个分区状态 $\mathbf{X}_A$(左)，$\mathbf{X}_B$ (中) 和 $\mathbf{X}_C$(右) 之间的可逆移动，仅在集合 V_0 的颜色上有所不同。由粗边连接的顶点形成连通分量。标有叉的细线是 SW 切口的边。由 Barbu 和 Zhu [3]。

提供

1. 规范案例: $V_o \subset V_\ell$ 和 $\ell' \leq L$。即，将 V_ℓ 的一部分重新分组为现有颜色 $V_{\ell'}$，并且颜色的数量在 $\mathbf{X}'$ 中保持 $\text{L} = L$。这是图 6.9中状态 $\mathbf{X}_A$ 和 $\mathbf{X}_B$ 之间的移动。

2. 合并情况: $\mathbf{X}$ 中的 $V_o = V_\ell$ 是具有颜色 ℓ 的所有顶点的集合，$\ell' \leq L$，且 $\ell \neq \ell'$。V_ℓ 与 $V_{\ell'}$ 合并，不同颜色的数量在 $\mathbf{X}'$ 中减少到 $L-1$。这是图 6.9中从状态 $\mathbf{X}_C$ 到 $\mathbf{X}_A$ 或从 $\mathbf{X}_C$ 到 $\mathbf{X}_B$ 的移动。

3. 分割情况: $V_o \subset V_\ell$ 和 $\ell' = L+1$。V_ℓ 被分成两部分，并且不同颜色的数量在 $\mathbf{X}'$ 中增加到 $L+1$ 。这是图 6.9中从状态 $\mathbf{X}_A$ 到 $\mathbf{X}_C$ 或从 $\mathbf{X}_B$ 到 $\mathbf{X}_C$ 的移动。

请注意，此颜色翻转步骤也与使用 Potts 模型的原始 SW 不同，因为我们允许在每个步骤中使用新颜色。颜色数 L 不固定。

6.4.3 Step 3: 接受翻转

前两个步骤提出了两个状态 $\mathbf{X}$ 和 $\mathbf{X}'$ 之间的移动它们的连通分量 V_o 的颜色不同。在第三步中，我们接受概率移动

$$\alpha(\mathbf{X} \to \mathbf{X}') = \min\{1, \frac{q(\mathbf{X}' \to \mathbf{X})}{q(\mathbf{X} \to \mathbf{X}')} \cdot \frac{\pi(\mathbf{X}')}{\pi(\mathbf{X})}\}。 \tag{6.48}$$

$q(\mathbf{X}' \to \mathbf{X})$ 和 $q(\mathbf{X} \to \mathbf{X}')$ 是 $\mathbf{X}$ 和 $\mathbf{X}'$ 之间的提议概率。如果提议被拒绝，马尔可夫链将保持在状态 $\mathbf{X}$。转换核心是

$$\mathcal{K}(\mathbf{X} \to \mathbf{X}') = q(\mathbf{X} \to \mathbf{X}')\alpha(\mathbf{X} \to \mathbf{X}'), \quad \forall \mathbf{X} \neq \mathbf{X}'。 \tag{6.49}$$

对于规范案例，有一个独特的路径可以在一步地在 $\mathbf{X}$ 和 $\mathbf{X}'$ 之间移动 --选择 V_o 并改变其颜色。提议概率是在状态 $\mathbf{X}$ 和 $\mathbf{X}'$ 中选择 V_o 作为聚类步骤中的候选者的概率比，与在翻转步骤中选择 V_o 为新标签的概率比的乘积。该乘积由下式给出

$$\frac{q(\mathbf{X}' \to \mathbf{X})}{q(\mathbf{X} \to \mathbf{X}')} = \frac{q(V_o|\mathbf{X}')}{q(V_o|\mathbf{X})} \cdot \frac{q(\mathbf{X}_{V_o} = \ell|V_o, \mathbf{X}')}{q(\mathbf{X}_{V_o} = \ell'|V_o, \mathbf{X})}。 \tag{6.50}$$

对于拆分和合并情况，$\mathbf{X}$ 和 $\mathbf{X}'$ 之间有两条路径，但这并不会改变结论。现在我们计算提出 V_o 的概率比 $\frac{q(V_o|\mathbf{X}')}{q(V_o|\mathbf{X})}$。

Definition 6.1 *设 $\mathbf{X} = (V_1, V_2, ..., V_{\mathrm{L}})$ 是一个着色的状态，且 $V_o \in \mathrm{CP}(U|\mathbf{X})$ 是一个连通分量，V_o 和 V_k 之间的"cut" 是 V_o 和 $V_k \backslash V_o$ 之间的一组边，*

$$C(V_o, V_k) = \{< s,t >:\ s \in V_o, t \in V_k \backslash V_o\},\ \forall k$$

观察的关键之一是比率 $\frac{q(V_o|\mathbf{X}')}{q(V_o|\mathbf{X})}$ 仅取决于 V_o 与其余顶点之间的切割。

命题 6.6 *在上面的表示中，我们有*

$$\frac{q(V_o|\mathbf{X})}{q(V_o|\mathbf{X}')} = \frac{\prod_{<i,j> \in C(V_o, V_\ell)}(1-q_{ij})}{\prod_{<i,j> \in C(V_o, V_{\ell'})}(1-q_{ij})}, \tag{6.51}$$

其中 q_e's 是边缘概率。

因此，接受概率在以下定理中给出。

Theorem 6.7 *建议交换的接受概率是*

$$\alpha(\mathbf{X} \to \mathbf{X}') = \min\{1, \frac{\prod_{<i,j> \in C(V_0, V_{\ell'})}(1-q_{ij})}{\prod_{<i,j> \in C(V_0, V_\ell)}(1-q_{ij})} \cdot \frac{q(\mathbf{X}_{V_0} = \ell|V_0, \mathbf{X}')}{q(\mathbf{X}_{V_0} = \ell'|V_0, \mathbf{X})} \cdot \frac{\pi(\mathbf{X}')}{\pi(\mathbf{X})}\}。 \tag{6.52}$$

证明见 [3]。

例 6.1 在图像分析中，$\pi(\mathbf{X})$ 是贝叶斯后验 $\pi(\mathbf{X}|\mathbf{I}) \propto \mathscr{L}(\mathbf{I}|\mathbf{X})p_o(\mathbf{X})$，先验概率 $p_o(\mathbf{X})$ 是马尔可夫随机场模型（如公式 *(6.43*中的 *Potts)*）。可以计算 V_o 的局部邻域 ∂V_o 中的目标概率的比率

$$\frac{\pi(\mathbf{X}')}{\pi(\mathbf{X})} = \frac{\mathscr{L}(\mathbf{I}_{V_o}|\mathbf{X}_{V_o} = \ell')}{\mathscr{L}(\mathbf{I}_{V_o}|\mathbf{X}_{V_o} = \ell)} \cdot \frac{p_o(\mathbf{X}_{V_o} = \ell'|\mathbf{X}_{\partial V_o})}{p_o(\mathbf{X}_{V_o} = \ell|\mathbf{X}_{\partial V_o})}。 \tag{6.53}$$

从以上公式注意到，$\mathbf{X}_{\partial V_o} = \mathbf{X}'_{\partial V_o}$。

等式 (6.52) 中的第二个比率易于设计。例如，我们可以使其与可能性成比例，

$$q(\mathbf{X}_{V_o} = \ell|V_o, \mathbf{X}) = \mathscr{L}(\mathbf{I}_{V_o}|\mathbf{X}_{V_o} = \ell), \quad \forall \ell。 \tag{6.54}$$

因此，

$$\frac{q(\mathbf{X}_{V_o} = \ell|V_o, \mathbf{X}')}{q(\mathbf{X}_{V_o} = \ell'|V_o, \mathbf{X})} = \frac{\mathscr{L}(\mathbf{I}_{V_o}|\mathbf{X}_{V_o} = \ell)}{\mathscr{L}(\mathbf{I}_{V_o}|\mathbf{X}_{V_o} = \ell')}。 \tag{6.55}$$

现在它取消了等式（6.53）中的似然比。最后，我们得出以下命题。

命题 6.8 使用 *(6.54)* 中的提议进行提议集群翻转的接受概率为，

$$\alpha(\mathbf{X} \to \mathbf{X}') = \min\{1, \frac{\prod_{<s,t>\in C(R,V_{\ell'})}(1-q_e)}{\prod_{e\in C(V_o,V_\ell)}(1-q_e)} \cdot \frac{p_o(\mathbf{X}_{V_o} = \ell'|\mathbf{X}_{\partial V_o})}{p_o(\mathbf{X}_{V_o} = \ell|\mathbf{X}_{\partial V_o})}\}。 \tag{6.56}$$

上述结果具有以下特性：计算局限于由先前模型定义的 V_o 的局部邻域。如果使用 Wolff 修改并从顶点增长 V_o，则此结果也成立。在图像分析实验中，SWC 方法在经验上比单点 Gibbs 采样器快 $O(100)$ 倍。有关详细信息，请参考第 (6.6) 节中图 6.11，6.13中的图表和比较。

6.4.4 复杂性分析

本节介绍了对 SWC 算法计算复杂性的评估。
设 $N = |V|$ 是图 $\mathbf{G} =< V, E >$ 的节点数，N^{it} 是 SWC 算法的迭代次数。
每个 SWC 迭代涉及以下步骤:

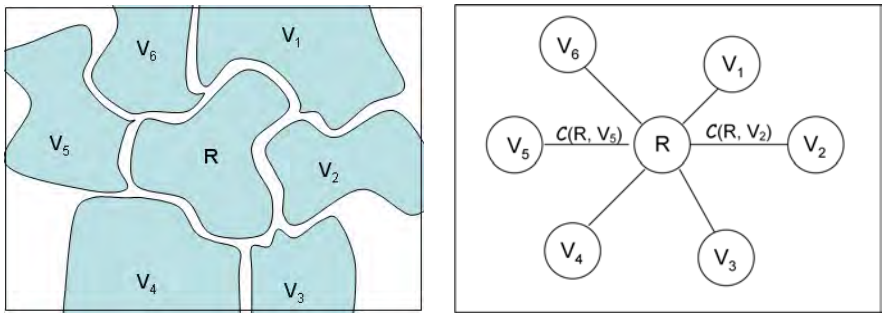
- 在数据驱动的聚类步骤中对边采样，即 $O(|E|)$。假设 $\mathbf{G} =< V, E >$ 稀疏，则 $O(|E|) = O(N)$。
- 使用不相交集森林数据结构 [18, 19] 构造连通分量，即 $O(|E|\alpha(|E|)) = O(N\alpha(N))$。函数 $\alpha(N)$ 是 $f(n) = A(n,n)$ 的倒数，其中 $A(m,n)$ 是快速增长的 Ackerman 函数 [1]。事实上，对于 N 的所有实际值，$\alpha(N) \leq 5$。
- 计算 $\pi(\mathbf{X})$，取决于问题，但通常是 $O(N)$。
- 翻转一个联通分量的标签，即 $O(N)$。

因此，一次迭代是 $O(N\alpha(N))$ ，所有迭代花费 $O(N^{it}N\alpha(N))$ 时间。

6.5 集群抽样方法的变体

在本节中，我们将简要讨论集群抽样方法的两种变体。

6.5.1 集群 Gibbs 采样 — "hit-and-run" 观点



(a) (b)

图 6.10: 解释集群 Gibbs 采样器。(a) 簇 V_o(此处为 R) 具有许多均匀颜色的相邻分量。(b) V_o 与其相邻颜色之间的切口。采样器遵循由切口上定义的边缘强度修改的条件概率。

稍作修改，我们就可以使集群抽样方法像广义的 Gibbs 抽样器一样。假定 $V_o \in \mathrm{CP}(U|\mathbf{X})$ 是在聚类步骤中选择的候选者，图 6.10显示了它与相邻集合的切割

$$C(V_o, V_k), k = 1, 2, ..., \mathrm{L}(\mathbf{X})$$

我们将数量 γ_k 算为 V_o 和 $V_k \backslash V_o$ 之间的连通强度，

$$\gamma_k = \prod_{e \in C(V_o, V_k)} (1 - q_e)。 \tag{6.57}$$

命题 6.9 *在上面的符号中，令 $\pi(\mathbf{X})$ 为目标概率。如果 V_o 是概率性重新标记的，用*

$$q(\mathbf{X}_{V_o} = k | V_o, \mathbf{X}) \propto \gamma_k \pi(\mathbf{X}_{V_o} = k | \mathbf{X}_{\partial V_o}),\ k = 1, 2, ..., N(\mathbf{X}), \tag{6.58}$$

然后在第三步中接受概率总是 1。

这产生了一个广义的 Gibbs 采样器，它根据修改的条件概率翻转一个簇的颜色。。

聚类 Gibbs 采样器

1。聚类步骤：选择顶点 $v \in V$，并通过伯努利边缘概率 μ_e 从 v 对聚类 V_o 进行分组。

2。翻转步骤：根据等式 (6.58) 重新标记 V_o 。

备注 6.2 *传统的单站点 Gibbs 采样器 [21] 是所有 e 的 $q_e = 0$ 时的特殊情况，导致所有 k 的 $V_o = \{v\}$ 和 $\gamma_k = 1$。*

人们也可以从 hit-and-run 的角度来看待上述方法。在连续状态空间中，hit-and-run 方法（参见 [22]）在时间 t 随机选择一个新方向，然后通过 $a \sim \pi(x + a\vec{e})$ 在这个方向上采样。Liu 和 Wu[36] 将此扩展到任何紧凑的行动组。在有限状态空间 Ω 中，可以选择有限集 $\Omega_a \subset \Omega$，然后在集合中应用 Gibbs 采样器。对于 hit-and-run 方法来说，很难选择好的方向或子集。在上面给出的聚类 Gibbs 采样器中，子集由边缘上的辅助变量选择。

6.5.2 多重翻转方案

在聚类步骤之后给定一组连通分量 CP($\mathbf{U}|\mathbf{X}$)(参见等式。 (6.47))，我们可以同时翻转所有（或任何选定数量）的连通分量，而不是翻转单个分量 V_o。可以独立地或联合地设计用于标记这些连接组件的提议概率。在下文中，我们假设通过从提议概率 $q(\mathbf{X}_{\text{cp}} = l|\text{cp})$ 采样，为每个连通分量 $\text{cp} \in \text{CP}(\mathbf{U}|\mathbf{X})$ 独立地选择标记。

假设我们在翻转后获得新标签 $\mathbf{X}'$。设 $E_{\text{on}}(\mathbf{X}) \subset E$ 和 $E_{\text{on}}(\mathbf{X}') \subset E$ 分别是连接 $\mathbf{X}$ 和 $\mathbf{X}'$ 中相同颜色顶点的边的子集。我们根据集合的不同来定义两个切割

$$C(\mathbf{X} \to \mathbf{X}') = E_{\text{on}}(\mathbf{X}') - E_{\text{on}}(\mathbf{X}), \text{ and } C(\mathbf{X}' \to \mathbf{X}) = E_{\text{on}}(\mathbf{X}) - E_{\text{on}}(\mathbf{X}')。 \tag{6.59}$$

我们用 $D(\mathbf{X}, \mathbf{X}') = \{\text{cp} : \mathbf{X}_{\text{cp}} \neq \mathbf{X}'_{\text{cp}}\}$ 表示在翻转之前和之后具有不同颜色的连通分量的集合。

命题 6.10 *多重翻转方案的接受概率是*

$$\alpha(\mathbf{X} \to \mathbf{X}') = \min\{1, \frac{\prod_{e \in C(\mathbf{X} \to \mathbf{X}')}(1 - q_e)}{\prod_{e \in C(\mathbf{X}' \to \mathbf{X})}(1 - q_e)} \frac{\prod_{\text{cp} \in D(\mathbf{X}, \mathbf{X}')} q(\mathbf{X}'_{\text{cp}}|\text{cp})}{\prod_{\text{cp} \in D(\mathbf{X}, \mathbf{X}')} q(\mathbf{X}_{\text{cp}}|\text{cp})} \cdot \frac{p(\pi')}{p(\pi)}\}。 \tag{6.60}$$

观察到当 $D = \{V_o\}$ 是单个连通分量时，这简化到到定理 6.56。值得一提的是，如果我们同时翻转所有连通分量，则 $\mathcal{K}(\mathbf{X}, \mathbf{X}')$ 的马尔可夫转移图完全连通，即。

$$\mathcal{K}(\mathbf{X}, \mathbf{X}') > 0, \ \forall \mathbf{X}, \mathbf{X}' \in \Omega。 \tag{6.61}$$

这意味着马尔可夫链可以一步内在任意两个分区之间移动。

6.6 应用: 图像分割

该实验在图像分割任务中测试聚类采样算法。目标是将图像划分为多个不相交的区域（如图 6.7和6.8所示），使得每个区域在适合某些图像模型的意义上具有一致的强度。最终结果应该优化贝叶斯后验概率 $\pi(\mathbf{X}) \propto \mathscr{L}(\mathbf{I}|\mathbf{X}) p_o(\mathbf{X})$。

在这些问题中，$\mathbf{G}$ 是邻接图，其顶点 V 是一组超像素。通常 $|V| = O(10^2)$。对于每个超像素 $v \in V$，我们计算一个 15-bin 强度直方图 h 归一化为 1。然后边缘概率计算为

$$q_{ij} = p(\mu_e = \text{on}|\mathbf{I}(v_i), \mathbf{I}(v_j)) = \exp\{-\frac{1}{2}(KL(h_i||h_j) + KL(h_j||h_i))\}, \tag{6.62}$$

其中 $KL()$ 是两个直方图之间的 Kullback-Leibler 差异。对于穿越物体边界的 e，通常 q_e 应接近零。在这些实验中，边缘概率导致较好的聚类，如图 6.8所示。

现在我们简要定义这个实验中的目标概率。令 $\mathbf{X} = (V_1, ..., V_{\mathrm{L}})$ 是图的着色，其中 L 是未知变量，并且每个组 V_k 中的图像强度在拟合到模型 θ_k 方面是一致的。假设不同的颜色是独立的。因此，我们有，

$$\pi(\mathbf{X}) = \pi(\mathbf{X}|\mathbf{I}) \propto \prod_{k=1}^{\mathrm{L}} [\mathscr{L}(\mathbf{I}(V_k); \theta_k) p_o(\theta_k)] p_o(\mathbf{X})。 \tag{6.63}$$

其中 $\mathscr{L}(\mathbf{I}(V_k); \theta_k)$ 是具有参数 $\theta - k$ 的似然模型，$p_o(\theta_k)$ 是 θ_k 之前的模型复杂度。这些量描述如下。

我们为似然模型选择了三种类型的简单模型来考虑不同的图像属性。第一个模型是非参数直方图 $\mathscr{H}$，其实际上由标准化为 1 的 B-bins $(\mathscr{H}_1, ..., \mathscr{H}_B)$ 的向量表示。它解释了杂乱的对象，如植被。

$$\mathbf{I}(x, y; \theta_0) \sim \mathscr{H} \text{ iid}, \forall (x, y) \in V_k。 \tag{6.64}$$

另外两个是二维图像平面 (x, y) 中强度平滑变化的回归模型，残差遵循经验分布 $\mathscr{H}$（即直方图）。

$$\mathbf{I}(x, y; \theta_1) = \beta_0 + \beta_1 x + \beta_2 y + \mathscr{H} \text{ iid}, \forall (x, y) \in V_k。 \tag{6.65}$$

$$\mathbf{I}(x, y; \theta_2) = \beta_0 + \beta_1 x + \beta_2 y + \beta_3 x^2 + \beta_4 xy + \beta_5 y^2 + \mathscr{H} \text{ iid}, \forall (x, y) \in V_k。 \tag{6.66}$$

在所有情况下，可能性以直方图 $\mathscr{H}$ 的熵表示

$$\mathscr{L}(\mathbf{I}(V_k); \theta_k) \propto \prod_{v \in V_k} \mathscr{H}(\mathbf{I}_v) = \prod_{j=1}^{B} \mathscr{H}_j^{n_j} = \exp(-|V_k| \text{entropy}(\mathscr{H}))。 \tag{6.67}$$

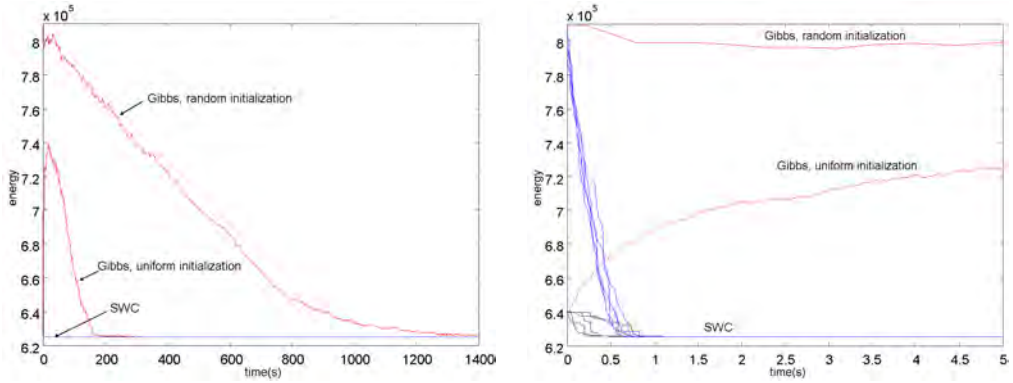
模型复杂度受先验概率 $p_o(\theta_k)$ 的影响，并且上述似然性中的参数 θ 在每个步骤确定性地计算为最好最小二乘拟合。确定性拟合可以由可逆跳跃和颜色翻转代替。这在 [56] 中完成，且将会在第 8章中介绍。

先验模型 $p_o(\mathbf{X})$ 鼓励具有少量颜色且大而紧凑的区域，如 [56] 中提到的那样。令 $r_1, r_2, ..., r_m$，$m \geq \mathrm{L}$ 为所有 $V_k, k = 1, ..., \mathrm{L}$ 的连通分量。然后先验为

$$p_o(\mathbf{X}) \propto \exp\{-\alpha_0 \mathrm{L} - \alpha_1 m - \alpha_2 \sum_{k=1}^{m} \text{Area}(r_k)^{0.9}\}。 \tag{6.68}$$

对于图 6.7和 6.8中所展示的图像分割示例（马），我们将簇采样方法与单站点 Gibbs 采样器进行比较，结果显示在图 6.11中。由于我们的目标是最大化后验概率 $\pi(\mathbf{X})$，我们必须添加具有高初始温度 T_o 的退火方案，然后降到低温（在我们的实验中是 0.05）。我们以秒为单位绘制了 $-\ln \pi(\mathbf{X})$/CPU 时间。Gibbs 采样器需要将初始温度提高（比如 $T_o \geq 100$））并使用慢退火计划来实现良好的解决方案。簇采样方法可以在低温下运行。我们通常将初始温度提高到 $T_o \leq 15$ 并使用快速退火方案。图 6.11（a）绘制了前 1,400 秒内的两种算法，图 6.11（b）是前 5 秒内的放大视图。

这两种算法两次初始化运行。一个是超像素的随机标记，因此具有更高的 $-\ln \pi(\mathbf{X})$，另一个初始化设置所有顶点为相同的颜色。在这两种情况下，聚类方法都运行五次。它们都在 1 秒内收敛到一个解（见图 6.7（c）），这比 Gibbs 采样器快 $O(10^2)$ 倍。



(a) 收敛 CPU 时间 (以秒为单位)　　(b) 前 5 秒的放大视图。

图 6.11: $-\ln \pi(X)$ 对 Gibbs 采样器和我们的对马图像算法的计算时间的图。（a）前 1400 秒内的情况。Gibbs 采样器需要高初始温度和慢退火步骤才能达到相同的能量水平。（b）前 5 秒的放大视图。Barbu 和 Zhu[4] 提供。

(a) 输入图像　　(b) 超像素　　(c) 分割结果

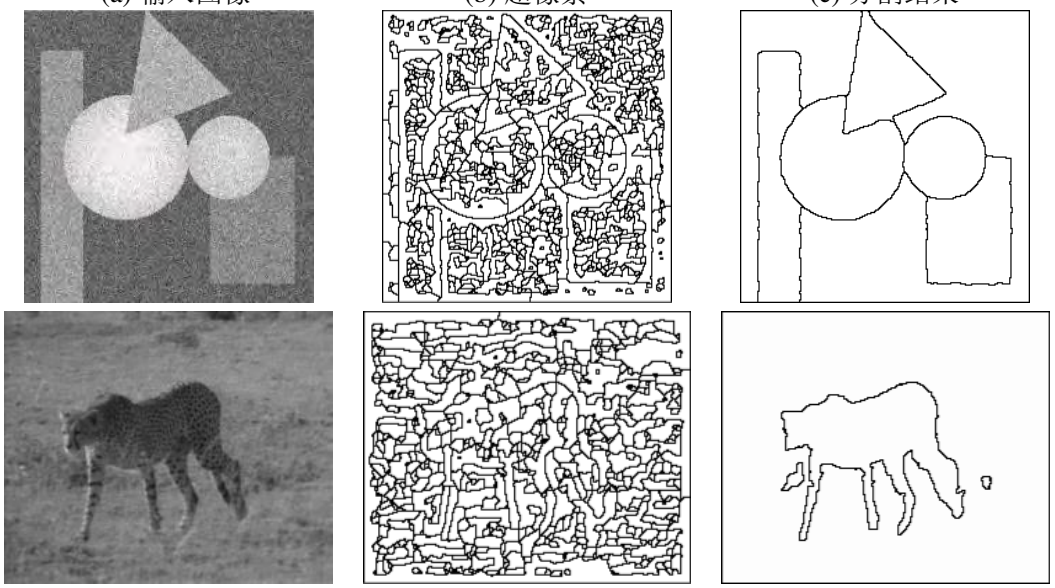


图 6.12: 更多图像分割的结果。Barbu 和 Zhu[4] 提供。

图 6.12显示了另外两张图像。使用马图像中的样本比较方法，我们在图 6.13中绘制了 $-\ln \pi(\mathbf{X})$ 与运行时间的关系。在实验中，我们还比较了边缘概率的影响。如果我们使用恒定边缘概率 $\mu_{ij} = c \in (0,1)$ 作为原始 SW 方法，则聚类算法慢了 $O(100)$ 倍。例如，单站点 Gibbs 采样器是 $q_{ij} = 0, \forall i, j$ 的示例。

6.7　多重网格和多级 SW 切割

SW 切割的本质是马尔可夫链 $\mathcal{MC} =< \nu, \mathcal{K}, p >$ 随着时间 t 访问分区空间 Ω 中的一系列状态,

$$\mathbf{X}(0), \mathbf{X}(1), ..., \mathbf{X}(t) \ \in \Omega.$$

上一节的结果确保 SW 切割设计遵循详细的平衡方程

$$p(\mathbf{X})\mathcal{K}(\mathbf{X}, \mathbf{X}') = p(\mathbf{X}')\mathcal{K}(\mathbf{X}', \mathbf{X}), \ \ \forall \mathbf{X}', \mathbf{X}. \tag{6.69}$$

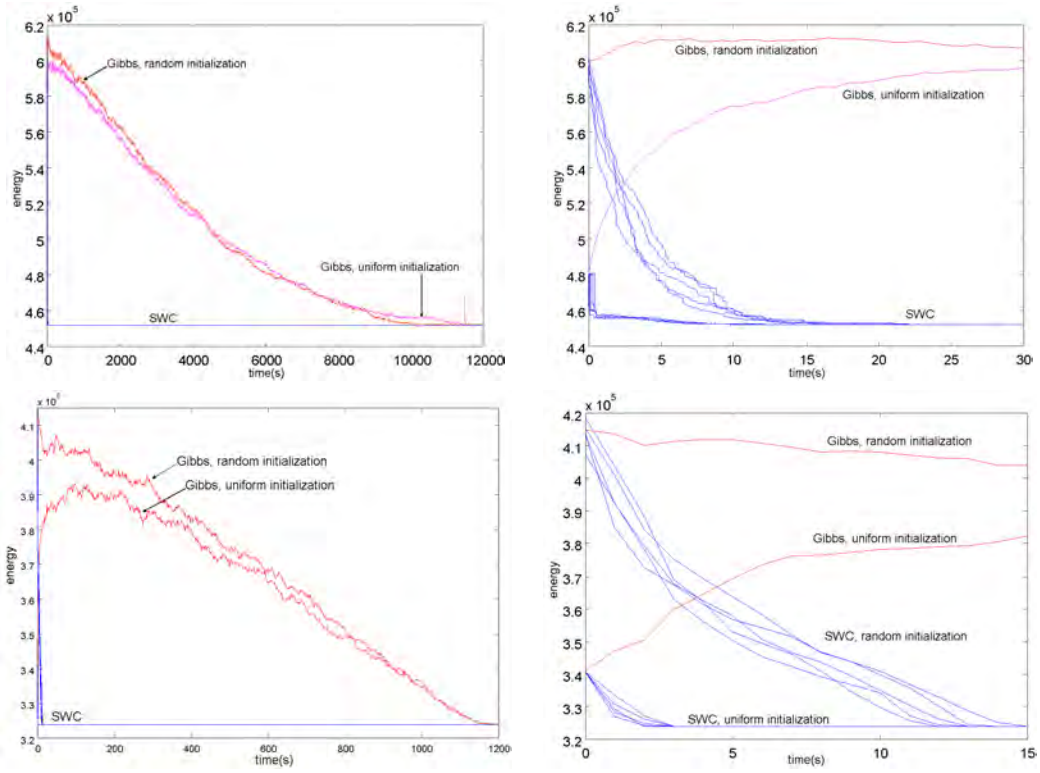


图 6.13: 对于图。6.12（左）的两个图像，聚类方法和 Gibbs 采样器在 CPU 时间（秒）内的收敛比较。前 1，200 秒。（右）放大前 15-30 秒的视图。对于随机和均匀初始化，聚类算法运行 5 次试验。Barbu 和 Zhu[4] 提供。

一旦收敛，SW 切割模拟来自 $p(\mathbf{X})$ 的均等样本

SW-cut 的特点是其设计中有三种选择.

1. 在邻接图 $G=<V,E>$ 的边上定义的判别提议概率。$q(\mathbf{X})=\prod_{e\in E} q_e$ 是对 $p(\mathbf{X})$ 的近似分解, 并且影响形连通分量 CP，因此影响候选分量 V_o.

2. 从连通分量 $V_o \in$ CP 中选择 V_o 的统一概率.

3. 连通分量 V_o 的新标签的重新分配概率 $Q(\ell_{\text{new}}(V_o)|V_o,\mathbf{X}_A)$.

我们通过引入多重网格和多级 SW 切割算法来扩展 SW 切割，这些算法为选择 V_o's 和 $q(\mathbf{X})$'s 提供了更灵活的方法。总而言之，这两个扩展是采样 $p(\mathbf{X})$ 的新方向.

1. 多重网格 SW-cut 通过对 $p(\mathbf{X})$ 的条件概率进行采样来模拟具有核心 $\mathcal{K}_{mg}$ 的马尔可夫链 $\mathcal{MC}_{mg}$.

2. 多级 SW 切割通过对较高水平的 $p(\mathbf{X})$ 的条件概率和较低水平的完全后验进行采样来模拟具有核 $\mathcal{K}_{mg}$ 的马尔可夫链 $\mathcal{MC}_{mg}$.

MCmg 和 MCml 都满足 (6.69) 中详细的平衡方程，它将在以下部分中显示。证明基于以下结果.

设 $p(x,y)$ 为二维概率，$\mathcal{K}$ 为采样条件概率 $p(x|y)$ (or $p(y|x)$ 的马尔可夫链内核。因此，$\mathcal{K}$ 观察到详细的平衡方程,

$$p(x|y)\mathcal{K}(x,x') \quad = \quad p(x'|y)\mathcal{K}(x',x),\ \forall x,x'. \tag{6.70}$$

内核 $\mathcal{K}$ 可以自然地增加到 (x,y) 上的内核

$$\mathcal{K}((x,y),(x',y')) = \begin{cases} \mathcal{K}(x,x') & \text{if } y = y' \\ 0 & \text{else} \end{cases}$$

Theorem 6.11 *在上面的表示中，$\mathcal{K}$ 在增加 y 之后观察一般的详细平衡方程，即*

$$p(x,y)\mathcal{K}((x,y),(x',y')) = p(x',y')\mathcal{K}((x',y'),(x,y)).$$

证明 6.7.1 *如果 $y = y'$, 那么就比较直接. 如果 $y \neq y'$, 那么 $\mathcal{K}((x,y),(x',y')) = \mathcal{K}((x',y'),(x,y)) = 0$.*

该定理的结论是，当从条件概率中采样时可逆的算法对于对全概率进行采样也是可逆的.

6.7.1 多重网格上的 SW-cuts

我们首先研究多重网格 SW 切割。回想一下，在每个步骤中，SW 切割在概率上关闭整个邻接图中的边，并且当 G 非常大时，这可能不太有效。多重网格 SW-cut 的概念, 是允许我们选择某些注意窗并在窗口内运行 SW 切割。因此，通过随时间选择各种尺寸和位置的窗口，它提供了设计“访问方案”的灵活性。例如，图 6.14显示了多网格排列中的窗口

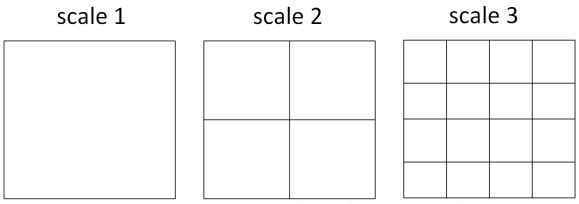


图 6.14: 在多重网格方案中选择窗口

令 $G =< V,E >$ 为邻接图，$\mathbf{X} = \{V_1,...,V_n\}$ 为当前分区，并且 Λ 是任意大小和形状的注意窗。Λ 分别将顶点划分为两个子集 $V = V_\Lambda \cup V_{\bar{\Lambda}}$，分别用于窗口内部和外部的顶点.

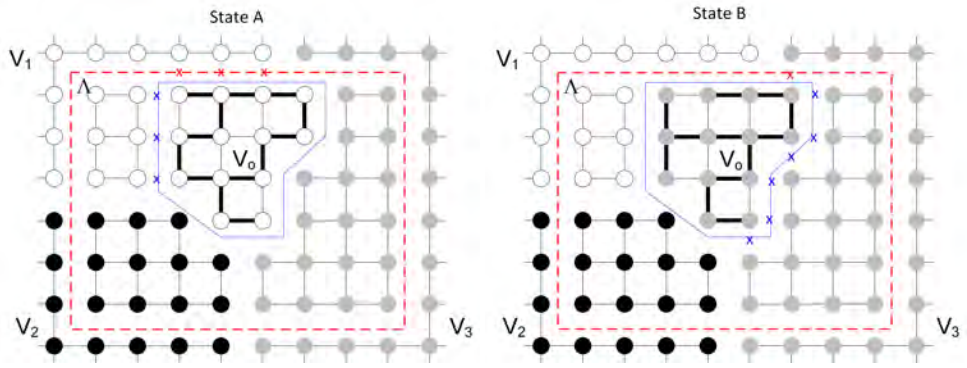


图 6.15: 多重网格 SW 切割：在注意窗口内运行 SW 切割 Λ，其余标签固定，并通过翻转 $V_o \subset V_\Lambda$ 的标签实现两个状态 $\mathbf{X}_A$ 和 $\mathbf{X}_B$ 之间的可逆移动. Barbu 和 Zhu[4] 提供.

例如，图 6.15显示了点阵 G 中的矩形窗口 Λ（红色）。窗口 Λ 进一步移除了每个子集 $V_i, i = 1,2,...,n$ 中的一些边，我们用下式表示它们,

$$\mathcal{C}(V_i|\Lambda) = \{e =< s,t >: s \in V_i \cap V_\Lambda,\ t \in V_i \cap V_{\bar{\Lambda}}\}.$$

例如，在图 6.15中，窗口 Λ 与三个子集 V_1（白色），V_2（黑色）和 V_3（灰色）相交，并且去除了与（红色）矩形窗口交叉的所有边。我们将顶点 V 的标记（着色或分区）分成两部分

$$\mathbf{X}(V) = (\mathbf{X}(V_\Lambda), \mathbf{X}(V_{\bar{\Lambda}})). \tag{6.71}$$

我们将 $\mathbf{X}(V_{\bar{\Lambda}})$ 固定为边界条件，并通过 SW-cut 对窗口内的顶点标签进行采样。总之，多重网格 SW-cut 迭代以下三个步骤:

1. 它按照概率 $\Lambda \sim q(\Lambda)$ 选择具有一定大小和形状的窗口 Λ.

2. 对于窗口内每个子图内的任何边,$e =< s,t >, s,t \in \Lambda$, $\ell_s = \ell_t$, 以概率 q_e 关闭边 e。因此，获得一组连通分量 $\mathrm{CP}(V_\Lambda)$.

3. 它选择 $V_o \in \mathrm{CP}(V_\Lambda)$ 作为连通分量，并根据概率翻转其标签

$$Q(\ell_{\text{new}}(V_o) = j | V_o, \mathbf{X}) = \frac{1}{C} \prod_{e \in \mathcal{C}_j} q_e \cdot p(\mathbf{X}^*_j), \ \forall j, \tag{6.72}$$

   其中 $\mathbf{X}^*_j$ 是分区，通过将 V_o 分配给标签 j, 且 $\mathcal{C}_j = \mathcal{C}(V_o, V_j) - \mathcal{C}(V_j|\Lambda)$.

例如，图 6.15示出了通过在两个状态 $\mathbf{X}_A$ 和 $\mathbf{X}_B$ 之间翻转连接分量 V_o（在蓝色多边形内）的可逆移动。$\mathcal{C}_1$ 和 $\mathcal{C}_3$ 由蓝色叉表示，通过随机程序去除。按照与前一个 SW-cut 相同的程序，我们可以推导出在 Λ 的两个状态中选择 V_o 的建议概率比.

Theorem 6.12 *在两个状态 $\mathbf{X}_A$ 和 $\mathbf{X}_B$ 处提出 V_o 作为窗口 Λ 内的候选子图的概率比是*

$$\frac{Q(V_o|\mathbf{X}_A, \Lambda)}{Q(V_o|\mathbf{X}_B, \Lambda)} = \frac{\prod_{e \in \mathcal{C}(V_o, V_1) - \mathcal{C}(V_1|\Lambda)} q_e}{\prod_{e \in \mathcal{C}(V_o, V_3) - \mathcal{C}(V_3|\Lambda)} q_e}.$$

该比率与定理 6.7中的比率之间的差异，在于一些边（参见图 6.15) 中的红色叉）不再参与计算。按照方程 (6.72) 中新标签的概率，我们可以证明它模拟了条件概率,

$$\mathbf{X}(V_\Lambda) \sim p(\mathbf{X}(V_\Lambda)|\mathbf{X}(V_{\bar{\Lambda}})).$$

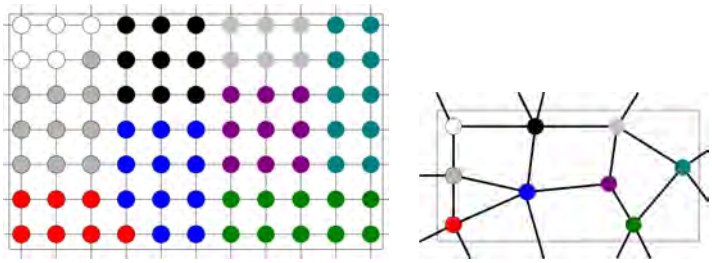
Theorem 6.13 *窗口 Λ 内的多重网格 SW 切割, 模拟了马尔可夫内核*

$$\begin{aligned} \mathcal{K}(\Lambda) &= \mathcal{K}(\mathbf{X}(V_\Lambda), \mathbf{X}'(V_\Lambda)|\mathbf{X}(V_{\bar{\Lambda}})), \\ p(\mathbf{X}(V_\Lambda)|\mathbf{X}(V_{\bar{\Lambda}}))\mathcal{K}(\mathbf{X}, \mathbf{X}') &= p(\mathbf{X}'(V_\Lambda)|\mathbf{X}(V_{\bar{\Lambda}}))\mathcal{K}(\mathbf{X}', \mathbf{X}). \end{aligned} \tag{6.73}$$

根据定理 6.11, 我们得到 $\mathcal{K}(\Lambda)$ 满足等式 (6.69) 中的一般详细平衡方程.

6.7.2 多层次 SW-cuts

现在我们添加一个多级 SW 切割机制。假设状态 $\mathbf{X} = \{V_1, V_2, ..., V_n\}$，我们冻结一些子集 $A_k, k \in \{1, ..., m\}$，使得对于任何 k，对于某些 i，$A_k \subset V_i$。这样，每个 A_k 中的顶点被锁定以具有相同的标签.



Level 1　　Level 2

图 6.16: 具有两个级别的多级 SW 切割. BarbuZhu[4] 提供.

子集 A_k 可以表示中间分割。例如，对于动作分割，获得强度分割 A 并将强度区域 A_k 分组为连贯地移动的对象，是很有用的。因此，$G = G^{(1)}$ 被缩减为较小的邻接图 $G^{(2)} =< U, F >$.U 是顶点集

$$U = \{u_1, ..., u_m\},\ u_k = A_k, k = 1, 2, ..., m.$$

F 是 G 中的子集 A_k 之间的邻接关系，由下式给出

$$F = \{f =< u_i, u_j >: \mathcal{C}(A_i, A_j) \neq \emptyset\}.$$

图 6.16展示了 $m = 9$ 的例子。我们在第 2 级上运行 SW 切割，基于新的判别性启发式 $q^{(2)}$，其测量 A_i, A_j, $q^{(2)}(\mathbf{X}(U)) = \prod_{f \in F} q_f^{(2)}$ 的相似性。一般来说，这些启发式方法比低级别更具信息性，因此 SW 切割移动更有意义，收敛速度更快.

图 $G^{(2)}$ 的分区空间是 Ω 的投影,

$$\Omega(G^{(2)}) = \{\mathbf{X} : x_s = x_t,\ \forall s, t \in A_i,\ i = 1, 2, ..., m\}.$$

显然，第 2 级上的 SW 切割模拟具有内核 $\mathcal{K}^{(2)}$ 的马尔可夫链，其具有不变概率 $p(\mathbf{X}(U)|A)$，$p(\mathbf{X})$ 的概率以所有 $s \in A_i$ 和 i 的关系 $x_s = x_{u_i}$ 为条件。根据定理 6.11，我们得到 $\mathcal{K}^{(2)}$ 满足一般详细平衡方程 (6.69).

假设我们设计了一个访问方案，用于随时间选择窗口 $\Lambda \sim q_w(\Lambda)$ 和水平 $\sigma \sim q_l(\sigma)$。那么广义 SW-cut 算法具有混合马尔可夫内核

$$\mathcal{K} = \sum_{\sigma} \sum_{\Lambda} q_l(\sigma) q_w(\Lambda) \mathcal{K}^{(\sigma)}(\Lambda).$$

由于每个 $\mathcal{K}^{(\sigma)}(\Lambda)$ 遵循详细平衡方程，$\mathcal{K}$ 也是如此。当窗口覆盖整个图形时，它也是不可约的，并且其状态在收敛时跟随 $p(\mathbf{X})$.

6.8　在子空间聚类

子空间聚类是将未标记的点集分组成对应于环境空间的子空间的多个聚类的问题。该问题在无监督学习和计算机视觉中具有应用。稀疏运动分割是一个这样的示例是，其中需要根据它们的共同运动模型将多个特征点轨迹分组为少量的簇。通过使用兴趣点检测器检测多个特征点，并使用特征点跟踪器或光流算法在许多帧中跟踪它们来获得特征点轨迹.

在目前最先进的稀疏运动分割方法 [15][32][34][57][63] 中，一种常用的方法是是将特征轨迹投影到

较低维空间，并使用基于谱聚类的子空间聚类方法对投影点进行分组并获得运动分割. 尽管这些方法在标准基准数据集上获得了非常好的结果，但是谱聚类算法需要在 $N \times N$ 密集矩阵上昂贵地计算特征向量和特征值，其中 N 是数据点的数量。以这种方式，这些子空间聚类/运动分割方法的计算时间缩放为 $O(N^3)$，因此它可能无法解决大规模问题 (例如 $N = 10^5 - 10^6$).

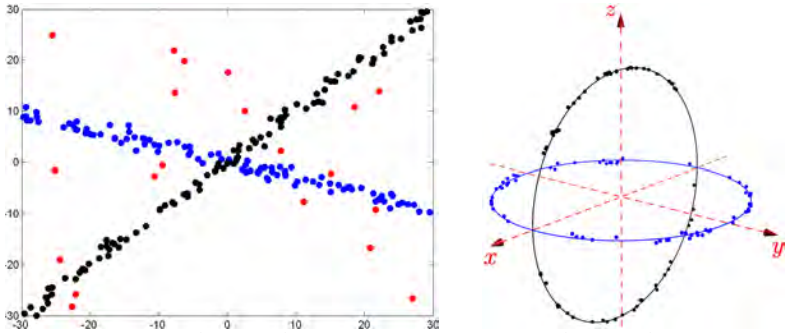


图 6.17: 子空间聚类示例。左：2D 中的两个 1D 子空间。右：3D 中的两个 2D 子空间，其中点已经标准化为单位长度。由于噪声，这些点可能不完全位于子空间上。可以观察到，除了平面交叉点之外，角距离在大多数地方找到正确的邻点。Ding 和 Barbu[13]. 提供

本节基于 SWC 算法提出了一种完全不同的子空间聚类方法。子空间聚类问题被公式化为贝叶斯框架中的最大后验（MAP）优化问题，具有 Ising / Potts 先验 [44] 和基于线性子空间模型的似然性。SWC 图被构建为来自亲和度矩阵的 k-NN 图。总体而言，该方法为解决子空间聚类问题提供了新的视角，并展示了 SWC 算法在聚类问题中的强大功能.

给定一组点 $\{\mathbf{x}_1, ..., \mathbf{x}_N\} \in \mathbb{R}^D$，子空间聚类问题是将点分组为对应于 $\mathbb{R}^D$ 的线性子空间的多个聚类，如图 6.17所示。一种流行的子空间聚类方法 [12][32][46] 基于谱聚类，其依赖于亲和度矩阵，该矩阵测量任何一对点属于同一子空间的可能性.

谱聚类 [40, 48] 是一种通用聚类方法，它根据连通性将一组点分组为聚类。点连通性被给定为 $N \times N$ 亲和度矩阵 A，如果点 i 接近点 j 则 A_{ij} 接近 1，且如果它们远离则接近零。亲和度矩阵的质量对于获得良好的聚类结果非常重要。用于谱子空间聚类的亲和度矩阵如下计算。首先，将这些点标准化为单位长度 [12, 32, 46]，然后接着在 [32] 中提出了基于向量之间角度的亲和度测量

$$A_{ij} = \left(\frac{x_i^T x_j}{\|x_i\|_2 \|x_j\|_2}\right)^{2\alpha}, \tag{6.74}$$

其中 α 是调整参数。在 [32] 中使用值 $\alpha = 4$。直观地看到，除了在子空间的交叉点附近的点之外，这些点倾向于与角度距离中的邻点位于相同的子空间中.

6.8.1 由 Swendsen-Wang 切分的子空间聚类

子空间聚类解决方法可以表示为输入点 $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathbb{R}^D$ 的分区 (标记)$\mathbf{X} : \{1, ..., N\} \to \{1, ..., M\}$。数量 $M \leq N$ 是允许的最大簇数。在本节中，假设给出了亲和度矩阵 A，表示任何一对点属于同一子空间的可能性。A 的一种形式在(6.74)中给出，另一种形式在下面给出.

可以使用后验概率来评估任何分区 $\mathbf{X}$ 的质量。然后可以通过最大化所有可能分区空间中的后验概

率来获得良好的分区。后验概率在贝叶斯框架中定义为

$$p(\mathbf{X}) \propto \exp[-E_{data}(\mathbf{X}) - E_{prior}(\mathbf{X})].$$

归一化常数在优化中是无关紧要的，因为它会抵消接受概率。数据项 $E_{data}(\mathbf{X})$ 基于子空间被假定为线性的事实。给定当前分区（标记）$\mathbf{X}$，对于每个标签 l，仿射子空间 L_l 以最小二乘意义拟合通过具有标签 l 的所有点。将具有标记 l 的点 x 与线性空间 L_l 的距离表示为 $d(\mathbf{x}, L_l)$. 然后数据项

$$E_{data}(\mathbf{X}) = \sum_{l=1}^{M} \sum_{i, \mathbf{X}(i)=l} d(\mathbf{x}_i, L_l). \tag{6.75}$$

先前的术语 $E_{prior}(\mathbf{X})$ 被设置为鼓励紧密连接的点保持在同一簇中.

$$E_{prior}(\mathbf{X}) = -\rho \sum_{<i,j> \in E, \mathbf{X}(i) \neq \mathbf{X}(j)} \log(1 - A_{ij}), \tag{6.76}$$

ρ 是控制先前项强度的参数。在下一节中将清楚地说明，该先验正好是 Potts 模型(6.1)，其将 A_{ij} 作为原始 SW 算法中的边权重。

SWC 算法可以自动决定簇的数量; 然而，在本章中，与大多数运动分割算法一样，假设子空间 M 的数量是已知的。因此，从子空间的数量 M 以均匀的概率对分量 V_o 的新标签进行采样

$$q(c_{V_o} = l' | V_o, \mathbf{X}) = 1/M.$$

谱聚类优化了归一化切割或比率切割 [58] 的近似，这是一种独特的测量方法。相反，基于 SWC 的子空间聚类方法优化了生成模型，其中可能性基于子空间是线性的假设。当违反线性假设时，判别措施可能更灵活并且更好地工作.

受 [32] 启发, 可以使用以下亲和度测量.

$$A_{ij} = \exp(-m \frac{\theta_{ij}}{\bar{\theta}}), \quad i \neq j, \tag{6.77}$$

其中 θ_{ij} 基于向量 x_i 和 x_j 之间的角度,

$$\theta_{ij} = 1 - (\frac{x_i^T x_j}{\|x_i\|_2 \|x_j\|_2})^2,$$

$\bar{\theta}$ 是所有 θ 的平均值. 参数 m 是调整参数，用于控制由 SWC 算法获得的连通分量的大小.

基于点之间的角度信息的亲和度测量使我们能够获得邻域图。获取之后，亲和度测量也用于获得边缘权重，用于在 SWC 算法中以及后验概率的先前项中，进行数据驱动的聚类提议。图 $G = (V, E)$ 具有需要聚集为其顶点的点集。边缘 E 基于等式 (6.77)的距离度量来构造。由于该距离度量在从相同子空间中找到最近邻居时更准确，因此该图被构造为 k-最近邻图，其中 k 是给定参数。获所获图的示例如 6.18所示.

SWC 算法被设计用于对后验概率 $p(\mathbf{X})$ 进行采样。要使用 SWC 进行优化，应采用模拟退火方案。

对于模拟退火，算法使用的概率是 $p(\mathbf{X})^{1/T}$，其中在优化开始时温度较大，且根据退火时间进度缓慢降低。如果退火时间进度足够慢，理论上保证 [27] 将找到概率 $p(\mathbf{X})$ 的全局最优值。实际上，我们使用更快的退火方案，最终的分区 $\mathbf{X}$ 只是局部最优。我们使用由三个参数控制的退火方案：起始温度 T_{start}，结束温度为 T_{end}，以及迭代次数 N^{it}。在步骤 i 处的温度计算为

$$T_i = \frac{T_{\text{end}}}{\log\left(\frac{i}{N}[e-\exp(\frac{T_{\text{end}}}{T_{\text{start}}})]+\exp(\frac{T_{\text{end}}}{T_{\text{start}}})\right)}, i = \overline{1, N^{it}}. \tag{6.78}$$

为了更好地探索概率空间，我们还使用具有不同随机初始化的多个执行。最终算法如下所示.

子空间聚类的 Swendsen-Wang 切割

输入: M 个子空间的 N 个点 $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$
使用等式 (6.77)将邻接图 G 构造为 k-NN 图.
for $r = 1, \ldots, Q$ **do**
  初始化分区 $\mathbf{X}$ 为 $\mathbf{X}(i) = 1, \forall i$.
  for $i = 1, \ldots, N^{it}$ **do**
    1. 使用公式 (6.78)计算温度 T_i.
    2. 使用公式(6.52)中的 $p(\mathbf{X}|I) \propto p^{1/T_i}(\mathbf{X})$ 执行 SWC 算法的一步.
  end for
  记录聚类结果 $\mathbf{X}_r$ 和最终的概率 $p_r = p(\mathbf{X}_r)$.
end for
输出: 具有最大 p_r 的聚类结果 $\mathbf{X}_r$.

设 N 是 $\mathbb{R}^D$ 中需要聚类的点数。SWC 子空间聚类方法的计算复杂度可以分解如下:

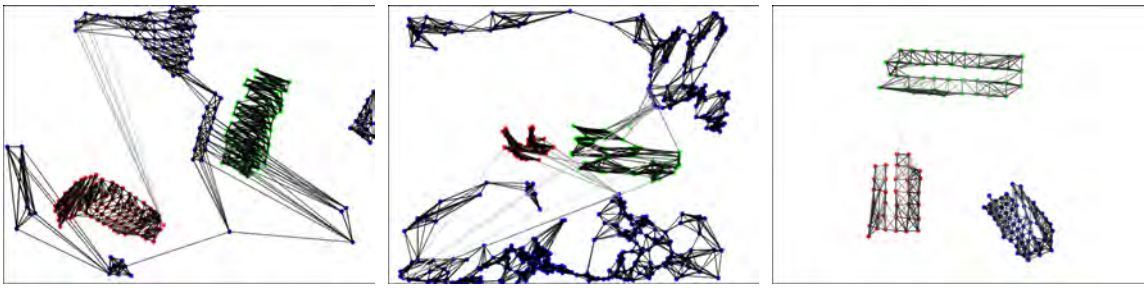
- 邻接图构造的复杂度是 $O(N^2 D \log k)$，其中 D 是空间维度。这是因为需要计算从每个点到其他 $N-1$ 个点的距离并保留其 k 个最近邻点.
- SWC 算法的 N^{it} 次迭代中的每一次都是 $O(N\alpha(N))$，如 6.4.4节中所讨论的。计算 $E_{data}(\mathbf{X})$ 涉及到拟合每个动态集群的线性子空间，即 $O(D^2N + D^3)$，而计算 $E_{prior}(\mathbf{X})$ 是 $O(N)$。迭代次数是固定的（例如 $N^{it} = 2000$），因此所有 SWC 迭代都需要 $O(N\alpha(N))$ 时间.

总之，整个算法复杂度为 $O(N^2)$，因此对于大问题，它比光谱聚类能更好地扩展.

6.8.2 应用：稀疏运动分割

本节介绍了基于 SWC 的子空间聚类算法在运动分割中的应用。最近大多关于运动分割的工作都使用了仿射相机模型，当物体远离相机时，该模型近似得到满足。在仿射相机模型下，图像平面上的点 (x, y) 与真实世界 3D 点 X 相关

$$\begin{bmatrix} x \\ y \end{bmatrix} = A \begin{bmatrix} X \\ 1 \end{bmatrix}, \tag{6.79}$$



(a) 1RT2TC (b) cars3 (c) articulated

图 6.18: checkerboard（a），traffic（b）和 articulated（c）序列的 SWC 加权图的示例。显示了第一帧中的特征点位置。边缘强度表示从 0（白色）到 1（黑色）的权重. Ding 和 Barbu[13] 提供.

其中 $A \in \mathbb{R}^{2\times 4}$ 是仿射运动矩阵.

设 $t_i = (x_i^1, y_i^1, x_i^2, y_i^2, \ldots, x_i^F, y_i^F)^T, i = 1, \ldots .N$ 是 F 帧中跟踪特征点的轨迹（2D 图像），其中 N 是轨迹的数量。让度量矩阵$W = [t_1, t_2, \ldots, t_N]$ 通过将轨迹组合为列来构造。如果所有轨迹都经历相同的刚体运动，则方程 (6.79) 意味着 W 可以被分解为运动矩阵 $M \in \mathbb{R}^{2F\times 4}$ 和结构矩阵 $S \in \mathbb{R}^{4\times N}$，如

$$W = MS$$

$$\begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_N^1 \\ y_1^1 & y_2^1 & \cdots & y_N^1 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^F & x_2^F & \cdots & x_N^F \\ y_1^F & y_2^F & \cdots & y_N^F \end{bmatrix} = \begin{bmatrix} A^1 \\ \vdots \\ A^F \end{bmatrix} \begin{bmatrix} X_1 & \cdots & X_N \\ 1 & \cdots & 1 \end{bmatrix},$$

其中 A^f 是帧 f 处世界变换矩阵的仿射对象。这意味着 $\text{rank}(W) \leq 4$。由于 S 的最后一行的输入总是 1，所以在仿射相机模型下，来自刚性运动对象的特征点的轨迹，位于维度最多为 3 的仿射子空间中.

(a) Checkerboard (b) Traffic (c) Articulated

图 6.19: 来自 Hopkins155 数据库上三个类别的一些序列的样本图像，其中叠加了 ground-truth. Ding 和 Barbu[13] 提供.

通常，我们给出一个测量矩阵 W，其包含来自多个可能的非刚性运动的轨迹。运动分割的任务是将来自每个运动的所有轨迹聚集在一起。一种流行的方法 [12][32][46][57] 是使用如前面所述的光谱聚类将轨迹投影到较低维空间，并在该空间中执行子空间聚类。这些方法在投影尺寸 D 和用于光谱聚类的亲和度测量 A 中不同.

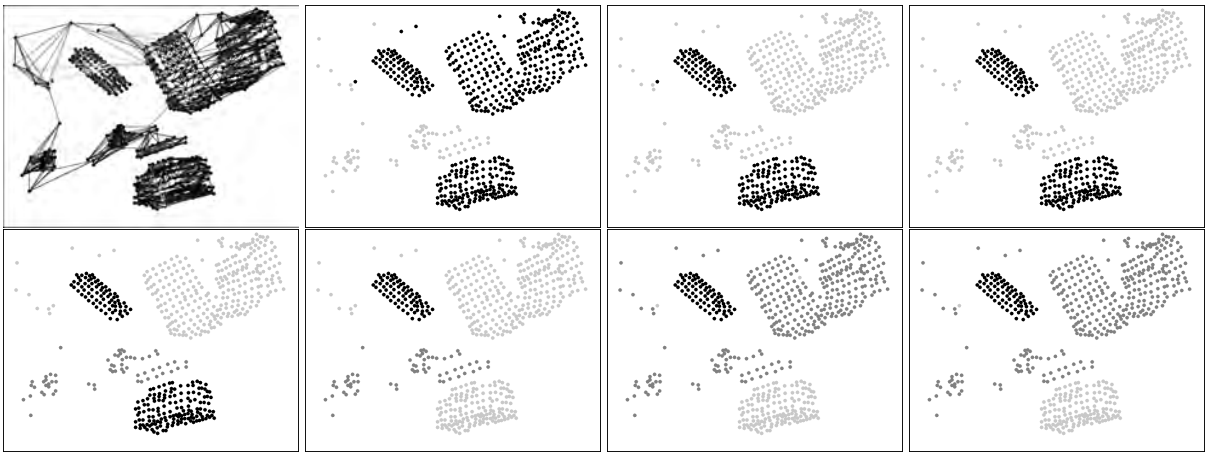


图 6.20: Hopkins155 序列 1R2TCR 的 SWC 聚类，包含 $M = 3$ 个运动。上图显示了第一帧中的特征点位置，其颜色为从初始状态（左上角）到最终状态（右下角）运行 SWC 算法时获得的标记状态 $\mathbf{X}$. Ding 和 Barbu[13]. 提供

降维是获得良好运动分割的必要预处理步骤。为了执行它，通常应用截断的 SVD [12, 32, 46, 57]。为了将测量矩阵 $W \in \mathbb{R}^{2F\times N}$ 投影到 $X = [x_1,...,x_N] \in \mathbb{R}^{D\times N}$，其中 D 是期望的投影维数，矩阵 W 被分解为 $W = U\Sigma V^T$，选择矩阵 V 的前 D 列作为 X^T。降维的 D 值也是运动分割中的主要问题。此值对最终结果的速度和准确性有很大影响，因此选择最佳维度以执行分割非常重要. 运动的维度不是固定的，而是可以随序列而变化，并且当存在多个运动时难以确定混合空间的实际维度，因此不同的方法可以具有不同的投影维度。一些方法 [12, 32] 使用穷举搜索策略，在具有一系列可能维度的空间中执行分割并选择最佳结果。在本节中，我们使用 $D = 2M+1$，它可以在这个应用中很好地工作.

当 $m >> n$ 时，计算 $m \times n$ 矩阵 U 的 SVD 的计算复杂度为 $O(mn^2 + n^3)$ [54]。如果 $n >> m$，则计算 U^T 的 SVD 更快，需要 $O(nm^2 + m^3)$。假设 $2F << N$ 意味着可以在 $O(NF^2 + F^3)$ 运算内计算 W 的 SVD。投影到维度 $D = 2M+1$ 的子空间后，应用第 6.8.1节中的 SWC 子空间聚类算法，聚类结果给出最终的运动分割结果.

本节介绍了 Hopkins155 运动数据库 [55] 中基于 SWC 的运动分割算法的实验。该数据库由 155 个两个和三个运动的序列组成。还提供了 ground-truth 分割以用于评估目的。根据视频的内容，序列可以是分为三大类：checkerboard，traffic 和 articulated 序列，示例图如 6.19所示。轨迹由跟踪器自动提取，因此它们会被噪声轻微破坏.

如上所述，在应用 SWC 算法之前，数据的维度从 $2F$ 减少到 $D = 2M+1$，其中 M 是运动的数量。在投影之后，SWC 算法中的初始标记状态，使所有点具有相同的标记.

参数设置. 运动分割算法具有许多调节参数，这些参数保持恒定为以下值。用于图构造的最近邻的数量是 k = 7，亲和度量度(6.77)中的参数 $m = 10$，并且(6.76)中的先验系数 $\rho = 2.2$。退火参数为 $T_{\text{start}} = 1$，$T_{\text{end}} = 0.01$，$N^{it} = 2000$. 获得最可能分区的独立运行次数 $Q = 10$.SWC 运行期间所有分区状态的示例如图 6.20所示.

结果. 使用由下式给出的误分类错误率来评估运动分割结果

$$\text{误分类率} = \frac{\#\text{ 误分类的点}}{\text{全部 }\#\text{ 点}}. \tag{6.80}$$

表 6.1列出了平均和中位误分类错误。为了准确起见，表 6.1中 SWC 算法的结果在 10 次运行中取平均值，标准偏差显示在括号中。为了将 SWC 方法与目前最先进的方法进行比较，我们还列出 ALC [46], SC [32], SSC [15] 和 VC [12] 的结果.

表 6.1: 在 Hopkins155 数据库上不同运动分割的误分类率 (百分比).

Method	ALC	SC	SSC	VC	SWC (std)	SC^4	SC^{4k}	KASP
All (2 motion)								
Average	2.40	0.94	0.82	0.96	1.49 (0.19)	11.50	7.82	4.76
Median	0.43	0.00	0.00	0.00	0.00 (0.00)	2.09	0.27	0.00
All (3 motion)								
Average	6.69	2.11	2.45	1.10	2.62 (0.13)	19.55	11.25	9.00
Median	0.67	0.37	0.20	0.22	0.81 (0.00)	18.88	1.42	1.70
All sequences combined								
Average	3.37	1.20	1.24	0.99	1.75 (0.15)	13.32	8.59	5.72
Median	0.49	0.00	0.00	0.00	0.00 (0.00)	6.46	0.36	0.31

基于 SWC 的算法获得的平均误差小于其他方法误差的两倍。在我们的实验中，我们观察到最终状态的能量通常小于 ground truth 状态的能量。这一事实表明 SWC 算法在优化模型方面做得很好，而贝叶斯模型在当前形态上不够准确，需要改进.

表 6.1中还显示了标记为 SC^4 和 SC^{4k} 的列，其表示 SC 方法 [32] 的误分类错误，并分别具有 4 个最近邻和 4k 个最近邻的亲和度矩阵。这些误差分别是 13.32 和 8.59，表明了谱聚类确实需要密集的亲和度矩阵才能正常工作，并且无法使用稀疏矩阵运算加速.

最后，将基于 SWC 的算法的性能与 KASP 算法 [62] 进行了比较，后者是一种快速近似谱聚类，用于代替 SC 方法中的谱聚类步骤 [32]。使用的数据约简参数是 $\gamma = 10$，结果仍然是，聚类算法的复杂度为 $O(N^3)$。总的误分类错误是 5.72，大约是 SWC 方法的三倍.

图 6.21: 序列 car10 的选定帧具有 1000 个跟踪特征点. Ding 和 Barbu[13]. 提供

为了评估不同算法的可扩展性，需要具有大量轨迹的序列。轨迹可以通过一些光流算法生成，但很难获得 ground truth 分割并消除不良轨迹。Brox 等 [8] 为 Hopkins155 数据集[1]中的 12 个序列的某些帧提供了密集分割。从他们那里，我们选择了 cars10 序列并使用 Classic+NL 方法 [50] 跟踪第一帧的所有像素。选择 car10 有两个原因 10。首先，它有三个动作，两个移动的车和背景。其次，两个移动的车在视频中相对较大，因此可以从每个动作中获得大量轨迹.

序列中有 30 帧，其中 3 帧具有所有像素的密集手动分割。我们删除了 3 个 ground truth 帧上具有不同标签的轨迹。接近动作边界的轨迹也被移除，我们只保留了聚类的完整轨迹。这样我们获得了大

[1] http://lmb.informatik.uni-freiburg.de/resources/datasets/

约 48,000 个轨迹作为池。从池中，对不同数量 N 的轨迹进行二次采样以进行评估。对于每个给定的 N，从池中随机选择轨迹，使得三个动作中每一个的轨迹数量大致相同。例如，为了生成 $N = 1000$ 个轨迹，我们将从两个动作的池中随机选取 333 个轨迹，从第三个动作的池中随机选取 334 个轨迹。如果一个动作中没有足够的轨迹，我们将从具有最多轨迹的运动中添加更多轨迹.

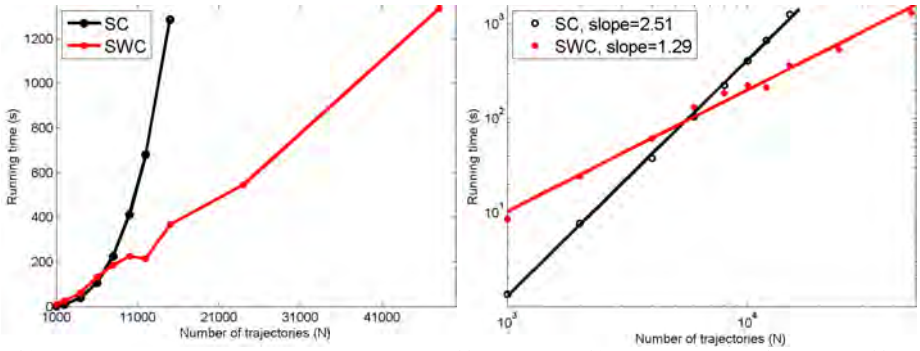


图 6.22: 左. 计算时间（秒）与 SC 和 SWC 的轨迹数 N. 右：使用拟合回归线的相同数据的 log-log 图. Ding 和 Barbu[13]. 提供

在图 6.22中，以原始比例和对数标度显示了计算时间与谱聚类（SC）和 SW 切割（SWC）算法的轨迹数量 N 的关系。从图中可以发现，对于少量轨迹，SC 比 SWC 快，但是对于超过 $N = 6,000$ 个轨迹，SC 的计算时间大于 SWC 的计算时间，并且增加得更快。在图 6.22中，右侧显示了 log（时间）与 log（N），并且线性回归通过两种方法的数据点拟合。如果线的斜率是 α，则计算复杂度按 $O(N^{\alpha})$ 缩放。我们观察到 SC 的斜率为 2.52，而 SWC 的斜率为 1.29，这与 6.8.1节的复杂性分析一致.

6.9 C4: 聚类合作竞争约束

Jake Porway

" 许多视觉任务，如场景标记 [30, 43, 47]，物体检测/识别 [16, 53]，分割 [11, 56] 和图形匹配 [9, 33] 被制定为能量最小化（或最大后验概率）在图形模型上定义的问题 - 马尔可夫随机场 [5, 21]，条件随机场 [30, 31] 或层次图 [20, 64]。当存在多种解决方案时，这些优化问题变得非常困难，即具有高概率的不同模式，或者在某些情况下，具有相同的概率."

图 6.23显示了在没有进一步的上下文情况下，具有多个同样可能的解决方案的典型场景的示例。第一行显示了著名的 Necker Cube，它有两个有效的 3D 解释。中间一行是 Wittgenstein 幻觉，其中的绘画可能看起来像是鸭子或兔子。没有进一步的背景，我们无法确定正确的标签。底行显示航拍图像。它可以解释为带有通风口的屋顶或包含汽车的停车场.

计算多个解决方案对于保持内在模糊性和避免对单个解决方案的早期承诺非常重要，即使它当前是全局最优解，也可能在后来的上下文到来时变得不那么有利。然而，使算法爬出局部最优，并在状态空间中相隔很远的解之间跳跃是一个持久的挑战。流行的能量最小化算法，如迭代条件模式 (ICM) [5]，Loopy Belief Propagation(LBP) [29, 59] 和图形切割 [6, 28] 计算单个解决方案，并没有解决这个问题。现有的 MCMC 算法，如各种 Gibbs 采样器 [21, 35]，DDMCMC [56] 和 Swendsen-Wang 切割 [3, 51]，有希

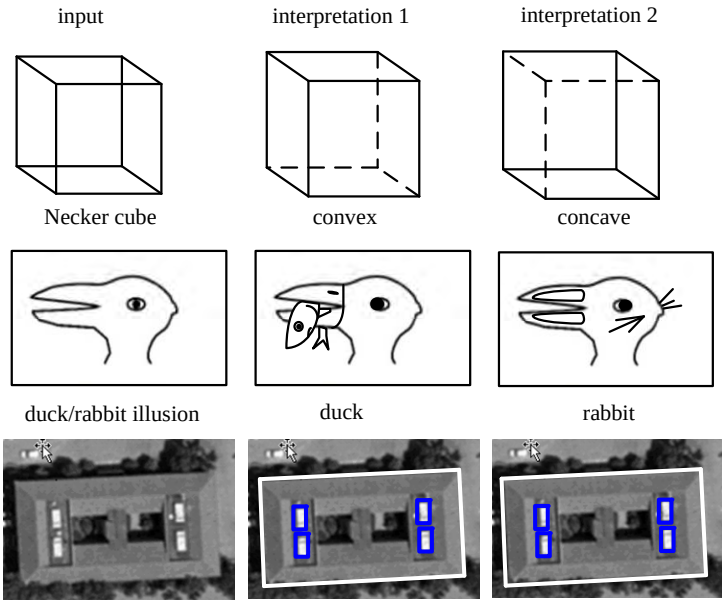


图 6.23: 多种解决方案的问题:(顶部）Necker Cube; （中）Wittgenstein 错觉; （底部）航拍图像被解释为带有通风口的车顶或带有汽车的停车场。在进一步的上下文到来之前，应保持歧义。Porway 和 Zhu[42] 提供.

望在状态空间中进行全局优化和遍历，但通常需要很长的等待时间才能在不同的模式之间移动，这需要一系列幸运的动作才能逃离景观中的能量井.

在本节中，我们将讨论一种算法，该算法可以通过在相等概率状态之间跳转来发现多个解决方案，从而保留相对一般设置的模糊性:

1. 图形可以是平面的，例如 MRF 或 CRF，或者是分层的，例如解析图.
2. 对于硬约束或软约束，该图可以具有正（合作）和负（竞争或冲突）边缘.

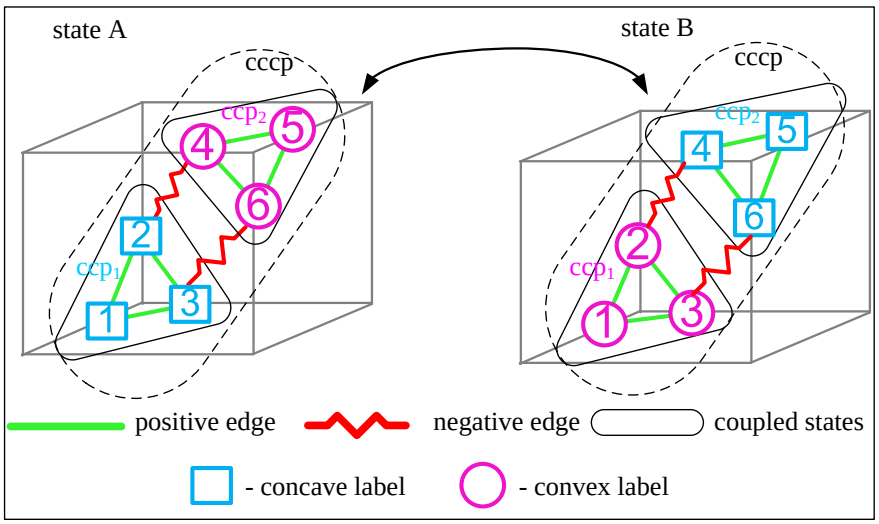


图 6.24: 在 Necker Cube 的两种解释之间进行转换。局部耦合标签与备用标签转换以强制实现全局一致性。请参阅正文以获取解释. Porway 和 Zhu [42] 提供.

3. 即使能量项涉及两个以上的节点，图上定义的概率（能量）也可能非常普遍.

在视觉中，可以安全地假设图形是局部连接的，且我们不考虑图形完全连接的最坏情况.

在 20 世纪 70 年代，许多问题，包括线条绘制和场景标记，被提出作为约束满足问题（CSPs）。通过启发式搜索方法 [41] 或约束传播方法 [2, 37] 解决了 CSP。前者保留了一个开放节点列表，以寻找合理的替代方案，并可以回溯以探索多种解决方案。但是，当图很大时，开放列表可能会变得太长而无法维护。后者基于其邻居迭代地更新节点的标签。一种众所周知的约束传播算法是 Rosenfeld，Hummel 和 Zucker 在 1976 年 [47] 提出的松弛标记方法.

在 20 世纪 80 年代， [21] 中提出了 Gibbs 采样器。标签的更新在固体 MCMC 和 MRF 框架中是合理的，因此保证从后验概率中进行采样。在特殊情况下，Gibbs 采样器等于多边形的信任传播 [41] 和链中的动态规划。当图中的多个节点强耦合时，发现 Gibbs 采样器严重减速.

图 6.24说明了使用 Necker Cube 与强耦合图相关的难点示例。该图的六条内部线分为两个耦合组：(1-2-3) 和 (4-5-6)。每组中的线必须具有相同的标签（凹面或凸面）才能形成有效的立方体，因为它们共享两个'Y' 形连接点。因此，除非我们一起更新整个组的标签，即一步所有六个标签，否则更新耦合组中单行的标签不会产生任何差别。问题是，对于具有大图的一般情况，我们不知道图中的哪些节点是耦合的，以及在何种程度上耦合.

6.9.1 C^4 算法综述

在本节中，我们提出了一种概率聚类算法，称为*聚类协同和竞争约束* (C^4)，用于计算图形模型中的多个解决方案。我们考虑两种类型的图. 邻接图将每个节点视为一个实体，例如像素，超像素，线条或物体，它们必须用 K-classes(或颜色) 标记。计算机视觉中使用的大多数 MRFs 和 CRFs 都是邻接图。候选图将每个节点视为候选或假设，例如实体的潜在标签，或窗口中被检测的对象实例，其必须被确认（打开）或拒绝（关闭）。换句话说，该图标记有 $K = 2$ 种颜色.

正如我们将在 6.9.2节中所示，邻接图总是可以转换为更大的候选图。在这两种情况下，任务在 MRF,CRF 或层次图上作为图形着色问题提出。有两种类型的边表示节点之间的硬约束或软约束（或耦合）。正边是合作约束，有利于两个节点在邻接图中具有相同的标签，或者在候选图中同时关闭和打开。负边是竞争性或冲突约束，要求两个节点在邻接图中具有不同的标签，或者一个节点打开而另一个节点在候选图中关闭.

在图 6.24中，我们显示了 Necker 立方体可以在邻接图中表示，每条线都是一个节点。六条内部线由六条正边（绿色）和两条负边（锯齿状的红色）连接。第 2 和第 4 行在它们相互交叉时具有负边，第 3 和第 6 行也是如此。为了清楚起见，我们省略了六条外线的标记.

边起计算作用，用于动态分组强耦合的节点。在每个正边或负边上，我们使用自下而上的判别模型来定义耦合强度的边缘概率。然后，我们设计了一个协议，用于根据每次迭代的边缘概率分别关闭和打开这些边。该协议对于所有问题都是通用的，而边缘概率是特定于问题的。这个概率过程关闭了一些边，剩下的所有边将图形划分为一些连通分量 (*ccp*'s).

A ccp 是一组通过正边连接的节点。例如，图 6.24有两个 *ccp*'s: ccp_1 包含节点 1-2-3，ccp_2 包含节点 4-5-6。每个 *ccp* 都是一个局部耦合的子解。*A cccp* 是一个复合连通分量，由多个负边连接的 *ccp*'s 组成。例如，图 6.24有一个包含 ccp_1 和 ccp_2 的 *cccp*。每个 *cccp* 包含一些冲突的子解.

在每次迭代中，C^4 选择一个 *cccp* 并同时更新 *cccp* 中所有节点的标签，以便（i）每个 *cccp* 中的节点保持相同的标签以满足正约束或耦合约束，（ii）*cccp* 中的不同 *ccp*'s 被分配不同的标签来遵循负面约

束。由于 C^4 可以在一个步骤中更新大量节点，因此它可以移出局部模式并在多个解之间有效跳转。协议设计动态地对 *cccp*'s 进行分组，并保证每个步骤都遵循 MCMC 要求，例如详细的平衡方程，因此它从后验概率中采样.

6.9.2 图形, 耦合和聚类

从平面图 G 开始，我们将扩展到 6.9.6节中的层次图,

$$G =< V, E >, \quad E = E^+ \cup E^-. \tag{6.81}$$

这里, $V = \{v_i, i = 1,2,...,n\}$ 一组顶点或节点，其上定义了变量 $X = (x_1,...,x_n)$, $E = \{e_{ij} = (v_i, v_j)\}$ 是一组边，对于正（合作）和负（竞争或冲突）约束，它分别分为 E^+ 和 E^-。我们考虑 G 的邻接和候选图.

通过将每个节点 v_i 转换为 K_i 个节点 x_{ij}，可以始终将邻接图转换到更大的候选图。$x_{ij} \in \{'\text{on}','\text{off}'\}$ 表示邻接图中的 $x_i = j$。这些节点遵循一个互斥约束，以防止模糊赋值给 x_i。图 6.25显示了这种转换。邻接图 $G_{\text{adj}} =< V_{\text{adj}}, E_{\text{adj}} >$ 具有六个节点 $V_{\text{adj}} = \{A,B,C,D,E,F\}$，并且每个节点具有 3 到 5 个潜在标签。变量是 $X_{\text{adj}} = (x_A,...,x_F)$，$x_A \in \{1,2,3,4,5\}$ 等等。我们将其转换为具有 24 个节点 $V_{\text{can}} = \{A_1,...,A_5,...,F_1,...,F_4\}$ 的候选图 $G_{\text{can}} =< V_{\text{can}}, E_{\text{can}} >$。节点 A_1 表示分配 $x_A = 1$ 的候选假设.$X_{\text{can}} = (x_{A_1},...,x_{F_4})$ 是布尔变量.

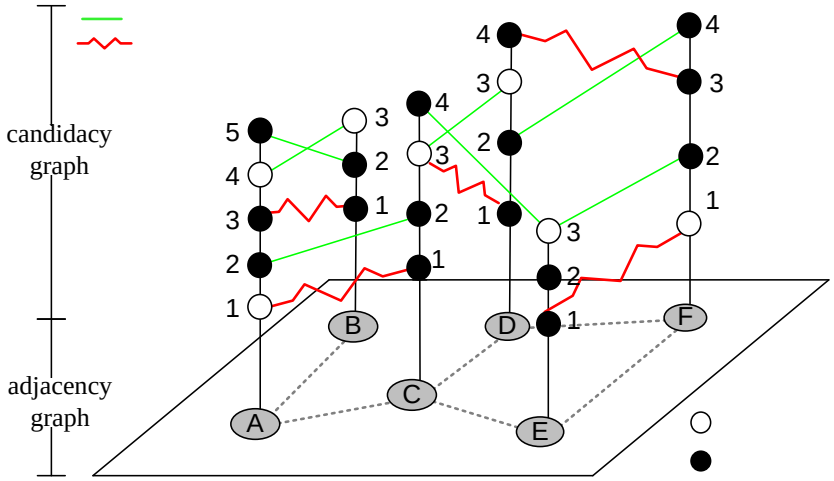


图 6.25: 将邻接图转换为候选图。候选图表具有正（绿色直线）和负（红色锯齿状线条）边，具体取决于在邻接图中分配给节点的值. Porway 和 Zhu [42] 提供.

由图 G 所表示，视觉任务被提出作为优化问题，用后验概率 $p(X|\mathbf{I})$ 或能量函数 $\mathcal{E}(X)$ 计算最可能的解释.

$$X^* = \arg\max p(X|\mathbf{I}) = \arg\min \mathcal{E}(X). \tag{6.82}$$

为了保持模糊性和不确定性，我们可以用权重 $\{\omega_i\}$ 计算多个不同的解 $\{X_i\}$ 来表示后验概率.

$$(X_i, \omega_i) \sim p(X|\mathbf{I}), \quad i = 1,2,...,K. \tag{6.83}$$

在传统的视觉公式中，图中的边是代表性的概念，$\mathcal{E}$ 中的能量项在边上定义，用来表示节点之间的相互作用。相比之下，Swendsen-Wang [51] 和 Edward-Sokal [14] 在他们的聚类采样方法中为边添加了

一个新的计算角色。边概率性地关闭和打开，来动态地形成强耦合节点的组（或簇）。在下面的示例后不久，我们将介绍聚类过程。在本节中，我们采用这一概念，图 G 中的边有三个方面的特征:

正 *vs.* 负. 正边表示在邻接图中具有相同标签或在候选图中同时打开（或关闭）的两个节点的协作约束。负边要求两个节点在邻接图中具有不同的标签，或者需要打开一个节点而在候选图中关闭另一个节点.

刚性 *vs.* 柔性. 一些边表示必须满足的刚性约束，而其他边约束是柔性的, 并且可以用概率表示.

位置依赖 *vs.* 值相关. 邻接图中的边通常取决于位置。例如，在 Ising 模型中，两个相邻节点之间的边构成柔性约束，它们应具有相同或相反的标签。相反，候选图中的边是依赖于值的，因此具有更强的表达能力。这对于视觉任务很常见，例如场景标注，线图解释和图形匹配。如图 6.25所示，候选图中节点之间的边可以是正的也可以是负的，这取决于在邻接图中分配给节点 A 和 B 的值.

我们将在后面的小节中说明，正边和负边对于生成连通分量和解决节点耦合问题至关重要.

图 6.26显示了用于解释 Necker 立方体的候选图 G 的结构。为了清楚起见，我们假设外部线被标记，并且任务是为六个内部线分配两个标签（凹和凸），以便满足所有局部约束和全局约束。因此，我们在 G 中总共有 12 个候选分配或节点.

基于线图解释理论 [38, 49]，两个'Y' 结构形成正约束，因此线 1-2-3 具有相同的标记，线 4-5-6 具有相同的标记。我们在 G 中有 12 个正边（绿色）来表示这些约束。线 2 和线 4 的交叉点构成负约束，要求线 2 和线 4 具有相反的标签，如图 6.26中红色锯齿状的边所示。线 3 和线 6 也是如此。每条线的两个不同的赋值也应该通过负边连接。为清楚起见，未示出这些负边.

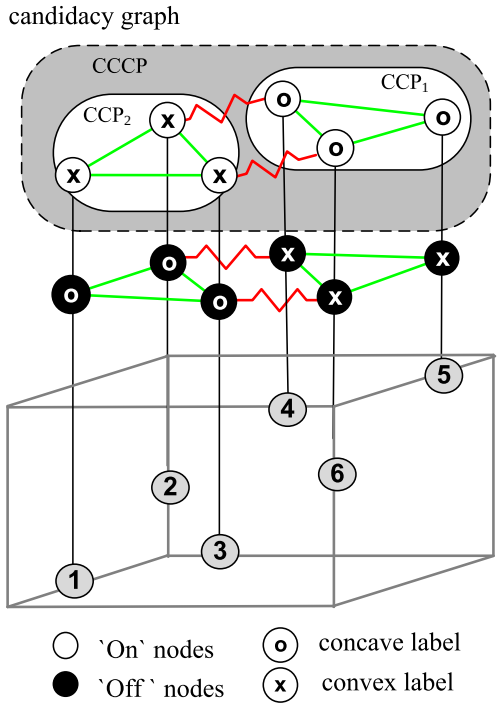


图 6.26: Necker 立方体的示例。具有 6 个节点（底部）的邻接图被分别转换为 12 个节点（顶部）的候选图，用于凹凸标签分配。在这些候选分配之间放置 12 个正边和 2 个负边以确保一致性. Porway 和 Zhu[42] 提供.

在这个候选图中，满足所有约束的两个解由图 6.26中的 2 种颜色表示。第一种解具有标记为凸（x）

的所有节点 1,2 和 3 以及标记为凹 (o) 的所有节点 4,5 和 6。该解目前处于开启状态。这将创建一个有效的 3D 解释，其中立方体将从这一页伸出来。替代解具有相反的标签，并创建对于下沉到页面中的立方体的 3D 解释.

要从一个解切换到另一个解，我们必须交换连接标签。每组节点 1-2-3 和 4-5-6 构成了 Necker Cube 的一个角，并且它们之间都有正约束。这表明我们应该同时更新所有的这些值, 我们创建两个连通分量 ccp_1 和 ccp_2, 分别由耦合节点 1-2-3 和节点 4-5-6 组成, 如果我们只是简单地反转 ccp_1 或 ccp_2 的标签，我们将创建一个不一致的解释，其中整个图中的所有边现在都具有相同的标签。我们要做的是同时交换 ccp_1 和 ccp_2.

注意到，我们在节点 2 和 4 之间以及节点 3 和 6 之间有负边。负边可以被认为是多个竞争解的指标，因为它们必然要求边任一端的组可以是 (on, off) 或 (off, on)，而创造两种可能的结果。此负边连接 ccp_1 和 ccp_2 中的节点，因此表明两个 ccp's 中的节点必须具有不同的标签。我们构造一个复合连通分量 $cccp_{12}$，包含节点 1-6，现在有一个包含所有相关约束的完整分量。从解 1 到解 2 的移动现在就像同时翻转所有节点一样简单，或者等效地满足所有约束。在下一小节中，我们将解释如何正式地形成 ccp's 和 ccp's.

在每个正边或负边上, 我们定义了耦合强度的边缘概率. 也就是说，在每个边 $e \in E$ 上，我们定义一个辅助概率 $u_e \in \{0,1\}$，它遵循一个独立的概率.

在 Swendsen 和 Wang [51] 中, q_e 的定义由 Potts 模型 $q_e = e^{-2\beta}$ 中的能量项决定，作为所有 e 的常数。Barbu 和 Zhu[3] 将 q_e 与能量函数分开并将其定义为自下而上的概率，$q_e = p(l(x_i) = l(x_j)|F(x_i),F(x_j)) = p(e = on|F(x_i),F(x_j))$，其中 $F(x_i)$ 和 $F(x_j)$ 是在节点 x_i 和 x_j 处提取的局部特征。这可以通过区分性训练来学习，例如通过逻辑回归和助推,

$$\frac{p(l(x_i) = l(x_j)|F(x_i),F(x_j))}{p(l(x_i) \neq l(x_j)|F(x_i),F(x_j))} = \sum_n \lambda_n h_n(F(x_i),F(x_j)).$$

在正边 $e = (i, j) \in E^+$ 上, $u_e = on$ 遵循伯努利概率,

$$u_e \sim \text{Bernoulli}(q_e \cdot 1(x_i = x_j)).$$

因此, 在当前状态 X，如果两个节点具有相同的颜色，即 $x_i = x_j$，则边 e 以概率 q_e 连通。如果 $x_i \neq x_j$，则 $u_e \sim \text{Bernoulli}(0)$ 和 e 以概率 1 关闭。因此，如果两个节点强耦合，则 q_e 应该具有更高的值以确保它们具有更高的概率来保持相同颜色。类似地，对于负边 $e \in E^-$，$u_e =$ 'on' 也遵循伯努利概率,

$$u_e \sim \text{Bernoulli}(q_e 1(x_i \neq x_j)).$$

在当前状态 X，如果两个节点具有相同的颜色 $x_i = x_j$，则边 e 以概率 1 关闭，否则以概率 e 打开 e，以强制 x_i 和 x_j 保持不同的颜色.

在独立地对所有 $e \in E$ 进行 u_e 采样后，我们分别表示保留在 $E^+_{on} \subset E_+$ 和 $E^-_{on} \subset E^-$ 上正边和负边的集合。现在我们给出 ccp 和 $cccp$ 的正式定义.

Definition 6.3 一个ccp 是一组顶点 $\{v_i; i = 1,2,...,k\}$，其中每个顶点都可以通过 E^+_{on} 中的正边从每个其他顶点到达.

Definition 6.4 一个 *cccp* 是一组 $ccp's\{ccp_i; i = 1,2,...,m\}$，其中每个 *ccp* 都可以通过 E^-_{on} 中的负边从每个其他 *ccp* 到达.

没有两个 *ccp*'s 可以通过正边到达，否则它们将是单个 *ccp*。因此，*cccp* 是一组由负边连接的分离的 *ccp*'s。分离的 *ccp* 也被视为 *cccp*。在 6.9.6节中，我们通过将 *ccp* 转换为 *cccp* 来处理 *ccp* 包含负边的无效情况。为了观察 MCMC 设计中的详细平衡方程，我们需要计算选择一个由边缘概率 q_e 决定的 *ccp* 或 *cccp* 的概率。为此我们定义他们的切割。通常，切割是连接两个节点集之间节点的所有边的集合.

Definition 6.5 在当前状态 X 下，*ccp* 切割是 *ccp* 中节点与具有相同标签的周围节点之间的所有正边的集合,

$$Cut(ccp|X) = \{e : e \in E^+, x_i = x_j, i \in ccp, j \notin ccp\}.$$

这些是必须概率地（概率为 $1-q_e$）关闭以形成 *ccp* 的边。切割取决于状态 X.

Definition 6.6 状态 XX 的 *cccp* 切割是连接 *cccp* 及其相邻节点中具有不同（或相同）标签节点的所有负（或正）边的集合,

$$Cut(cccp|X) = \{e : e \in E^-, i \in cccp, j \notin cccp, x_i \neq x_j\} \cup \{e : e \in E^+, i \in cccp, j \notin cccp, x_i = x_j\}.$$

所有这些边必须概率地关闭（概率为 $1-q_e$），以便在状态 X 处形成 *cccp*。由于 E^+_{on} 中的边仅连接具有相同标签的节点，因此 *ccp* 中的所有节点必须具有相同的标签。相比之下，E^-_{on} 中的所有边仅连接具有不同标签的节点，因此 *cccp* 中的相邻 *ccp*'s 必须具有不同的标签.

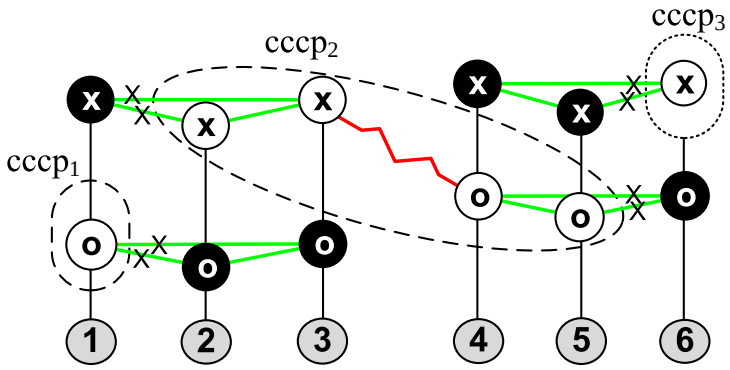


图 6.27: Necker 立方体候选图表不处于解决状态. Porway 和 Zhu[42] 提供.

为了解释这些概念，我们在图 6.27中显示了 Necker 立方体的非解状态 X. 通过关闭一些边（用叉标记），我们获得了当前打开的节点的三个 *cccp*'s。在此示例中，$q_e = 1$，因为这些是不可改变的刚性约束。$cccp_1$ 和 $cccp_3$ 只有 1 个节点，而 $cccp_2$ 有具有 4 个节点的两个 *ccp*'s。该算法将任意选择一个 *cccp* 并根据约束更新其值。如果它选择 $cccp_1$ 或 $cccp_3$，那么我们距离解更近了一步。如果它选择 $cccp_2$，则将交换所有 4 个顶点标签，并且我们将达到解状态。在这种情况下，我们将继续在两种解之间来回切换.

6.9.3 平面图上的 C^4 算法

平面图上的 C^4 算法遵循 MCMC 设计迭代地工作。在每次迭代中，它生成 *cccp*'s，以一定概率选择 $cccp_o$，并将标签重新分配给其 *ccp*'s，以便满足所有内部负约束。随着 $cccp_o$ 中 *ccp*'s 的数量增加，潜在

标签的数量也将增加。可以通过两种方式处理这种状况:

1. 使用约束满足问题求解器（CSP 求解器）来解决 $cccp_o$ 中更小，更简单的约束满足问题.

2. 使用随机或启发式采样来查找新的有效标签.

我们将在本节中使用第二种方法，并且 $cccp_o$ 中的 ccp's 数量通常很小，因此标签分配不是问题。C^4 算法可以被视为将大的约束满足问题分解为可以在局部满足的较小片段的方法。然后通过迭代传播解决方法。此赋值表示 MCMC 中的移动，MetropolisHastings 步骤以一个接受概率接受该移动。该接受概率考虑了生成 $cccp$'s，选择 $cccp_o$，分配新标签和后验概率的可能性.

总而言之，下面给出了该算法的伪代码版本.

The C^4 Algorithm

输入: 图 $G =< V,E >$ 和后验概率. $p(X|\mathbf{I})$.

计算边缘概率 $q_e, \forall e \in E$.

q_e 是特定于问题的判别概率.

初始化状态 $X = (x_1, x_2, ..., x_n)$.

例如在候选图中关闭所有节点.

for $s = 1$ to N^{iter} **do**

用状态 A 表示当前 X.

步骤 1: 在 A 生成一个 $cccp_o$.

$\forall e = (i, j) \in E^+$, 样本 $u_e \sim \text{Bernoulli}(q_e 1(x_i = x_j))$.

$\forall e = (i, j) \in E^-$, 样本 $u_e \sim \text{Bernoulli}(q_e 1(x_i \neq x_j))$.

基于 E^+_{on} 和 E^-_{on} 生成 $\{ccp\}$ and $\{cccp\}$

依概率从 $\{cccp\}$ 选择一个 $cccp_o$

用 $q(cccp_o|A)$ 表示选择 $cccp_o$ 的概率.

步骤 2: 按概率: $q(l(cccp_o = L|cccp_o, A))$ 将标签分配给 $cccp$ 中的 ccp's.

将新 X 表示为状态 B.

Step 3: 计算接受概率:

$$\alpha(A \to B) = \min(1, \frac{q(B \to A)}{q(A \to B)} \cdot \frac{p(X = B|\mathbf{I})}{p(X = A|\mathbf{I})}).$$

end for

输出: 具有最高概率的不同状态 $\{X^*\}$.

在马尔可夫链设计中, 两个状态 A 和 B 之间的每个移动都是可逆的, 观察详细的平衡方程, In Markov chain design, each move between two states A and B is made reversible and observes the detailed balance equation,

$$p(X = A|\mathbf{I})\mathcal{K}(A \to B) = p(X = B|\mathbf{I})\mathcal{K}(B \to A). \tag{6.84}$$

$\mathcal{K}(A \to B)$ 是马尔可夫链内核或从 A 到 B 的转换概率。在 Metropolis-Hastings 设计中,

$$\mathcal{K}(A \to B) = q(A \to B)\alpha(A \to B), \ \forall A \neq B. \tag{6.85}$$

$q(A \to B)$ 是从状态 A 提议状态 B 的概率，$\alpha(A \to B)$ 是接受概率,

$$\alpha(A \to B) = \min(1, \frac{q(B \to A)}{q(A \to B)} \cdot \frac{p(X = B|\mathbf{I})}{p(X = A|\mathbf{I})}). \tag{6.86}$$

很容易查验, 方程 (6.86) 中提议概率的设计和方程 (6.85) 中的接受概率使得内核满足 (6.84) 中的详细平衡方程，反过来又满足不变性条件,

$$p(X = A|\mathbf{I})\mathcal{K}(A \to B) = p(X = B|\mathbf{I}). \tag{6.87}$$

因此，$p(X|\mathbf{I})$ 是具有内核 $\mathcal{K}$ 的马尔可夫链的不变概率。现在我们详细说明提议和接受概率的设计。接受概率由两个比值确定。.

(i) 比值 $\frac{p(X=B|\mathbf{I})}{p(X=A|\mathbf{I})}$ 是特定问题，不属于我们的设计。后验概率可以是一般形式，不必修改或近似以适合 C^4 算法。由于状态 A 和 B 仅在 $cccp_o$ 中的节点标签上不同，如果后验概率是 MRF 或 CRF，则该比值通常可以在局部计算.

(ii) 提议概率完全取决于我们的设计，其包括两部分,

$$\frac{q(B \to A)}{q(A \to B)} = \frac{q(cccp_o|B)}{q(cccp_o|A)} \cdot \frac{q(l(cccp_o) = L_A|cccp_o, B)}{q(l(cccp_o) = L_B|cccp_o, A)}.$$

$q(cccp_0|A)$ 和 $q(cccp_0|B)$ 分别是在状态 A 和 B 选择 $cccp_o$ 的概率。给定选择的 $cccp_o$，新标签的分配独立于 $cccp_o$ 的周围邻居，并且通常在 CSP 求解器的所有有效赋值中以相等的概率分配. 因此，他们约掉了，我们有 $\frac{q(l(cccp_o)=L_A|cccp_o,B)}{q(l(cccp_o)=L_B|cccp_o,A)} = 1$.

总之，算法设计的关键是比值 $\frac{q(cccp_o|B)}{q(cccp_o|A)}$.。在单站点采样中，例如 Gibbs 采样器，每个节点都是 $cccp_o$，选择只是一个访问方案。在 C^4 中，在一个状态下选择 $cccp_o$ 的概率取决于两个概率：（a）通过对伯努利概率之后的边缘概率 q_e 进行采样来生成 $cccp_o$ 的可能性有多大;（b）在状态 A 和 B 中形成的 $\{cccp\}$ 集合中选择 $cccp_o$ 的可能性有多大。这些概率很难计算，因为通过打开/关闭边，图表中有大量分区包含某个 $cccp_o$。关闭一些边后，分区是一组 $cccp$'s.

有趣的是，状态 A 中所有可能分区的集合与状态 B 中的相同，并且所有这些分区必须共享相同的切口 $Cut(cccp_o)$。也就是说，为了使 $cccp_o$ 成为复合连通分量，必须关闭其与相邻节点的连接。尽管概率的形式复杂，但由于约分，它们的比值简单干净。此外，给定分区，即可从所有可能的 $cccp$'s 中以均匀的概率选择 $cccp_o$.

Proposition 1 *在状态 A 和 B 选择 $cccp_o$ 的提议概率比是*

$$\frac{q(cccp_0|B)}{q(cccp_0|A)} = \frac{\prod_{e \in Cut(cccp_o|B)}(1 - q_e)}{\prod_{e \in Cut(cccp_o|A)}(1 - q_e)}. \tag{6.88}$$

为了解释 C^4, 我们更详细地推导出了具有正边和负边的 Potts 模型. 设 X 是在具有离散状态 $x_i \in$

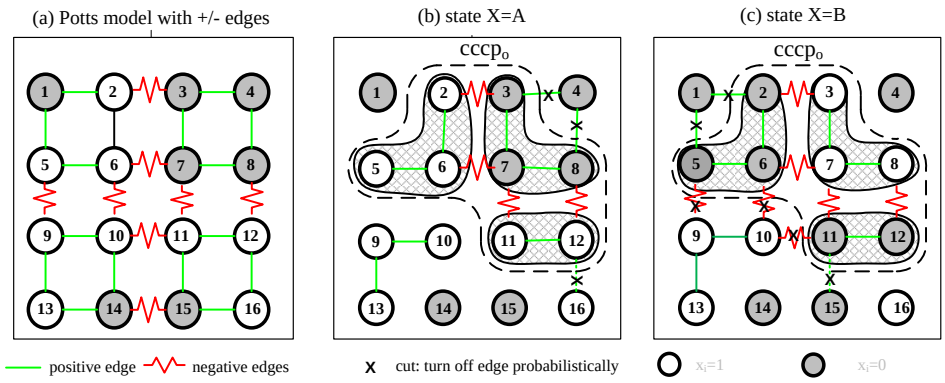


图 6.28: 具有负边的 Potts 模型。（a）在棋盘图案中找到最小能量。（b）形成 cccps。（c）$cccp_0$ 由通过负边连接的子 ccps 的正边组成. Porway 和 Zhu[42] 提供.

$\{0,1,2,...,L-1\}$ 的 2D 点阵上定义的随机场. 其概率由下式指定

$$p(X) = \frac{1}{Z}\exp\{-\mathcal{E}(X)\};\ \text{with}\ \mathcal{E}(X) = \sum_{<i,j>\in E^+} \beta\delta(x_i = x_j) + \sum_{<i,j>\in E^-} \beta\delta(x_i \neq x_j), \tag{6.89}$$

其中 $\beta > 0$ 是一个常量. 对于所有边, 边缘概率是 $q_e = 1 - e^{-\beta}$.

图 Figure 6.28(a) 显示了具有 $L = 2$ 个标签的小点阵上的示例，其是具有位置相关边的邻接图。具有棋盘图案的状态将具有最高概率。图 6.28(b) 和 (c) 通过一步翻转 $cccp_o$ 的标签显示了两个可逆状态 A 和 B. 在这个例子中, $cccp_o$ 有三个 ccp's, $cccp_o = \{\{2,5,6\};\{3,7,8\};\{11,12\}\}$. 8 个节点的标签以均匀的概率重新分配，这导致两个状态下 $cccp_o$ 的切口差异, $Cut(cccp_o|A) = \{(3,4),(4,8),(12,16)\}$ 以及 $Cut(cccp_o|B) = \{(1,2),(1,5),(5,9),(6,10),(10,11),(11,15)\}$.

Proposition 2 *对于在 $cccp_o$ 中具有不同标记的任何两个状态，Potts 模型上 C^4 的接受概率是 $\alpha(A \to B) = 1$。因此，此举始终被接受.*

证明遵循两个观察结果。首先，$cccp_o$ 内外的能量项对于 A 和 B 都是相同的，它们仅在 $cccp_o$ 的切口上有所不同. 更确切的说, 设 $c = |Cut(cccp_o|B)| - |Cut(cccp_o|A)$ 为两个切口大小的差异（即在我们的例子中 $c = 3$）。表明这一点并不难

$$\frac{p(X = B|\mathbf{I})}{p(X = A|\mathbf{I})} = e^{-\beta c}. \tag{6.90}$$

其次，我们有提议概率比，遵循等式 (6.88),

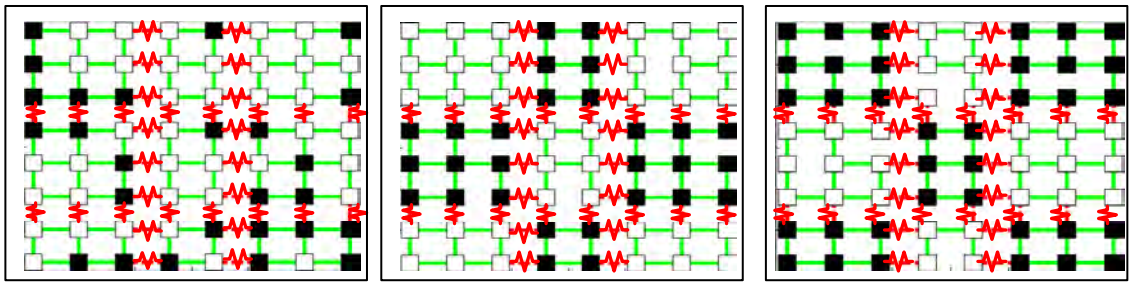
$$\frac{q(cccp_0|B)}{q(cccp_0|A)} = \frac{(1-q_e)^{|Cut(cccp_o|B)|}}{(1-q_e)^{|Cut(cccp_o|A)|}} = e^{\beta c}. \tag{6.91}$$

在等式 6.86中插入两个比值，我们得到 $\alpha(A \to B) = 1$.

6.9.4 在平面图上的实验

在本节中，我们与 Gibbs 采样器 [21]，SW 方法 [51]，迭代条件模式（ICM），图形分割 [6] 和循环置信度传播（LBP） [29] 一起, 测试 C^4 在某些平面图（MRF 和 CRF）上的性能。我们选择了一些经典

的案例：（i）MRF 的 Ising/Potts 模型；（ii）使用候选图的约束满足问题的线描解释；（iii）使用 CRF 进行场景标记；（iv）航拍图像的场景解释.



(a) initial state (b) solution state 1 (c) solution state 2

图 6.29: 具有棋盘约束的 Ising / Potts 模型和由 C^4 计算的两个最小能量状态. Porway 和 Zhu[42] 提供.

6.9.5 棋盘 Ising 模型

我们首先考虑具有正边和负边的 9×9 点阵上的 Ising 模型。我们用两个参数设置测试了 C^4:（i）$\beta = 1, q_e = 0.632$;（ii）$\beta = 5, q_e = 0.993$。在这个点阵中，我们创建了一个棋盘图案。我们已经分配了负边和正边，以便节点块想要具有相同的颜色，但是这些块想要与它们的邻居具有不同的颜色.

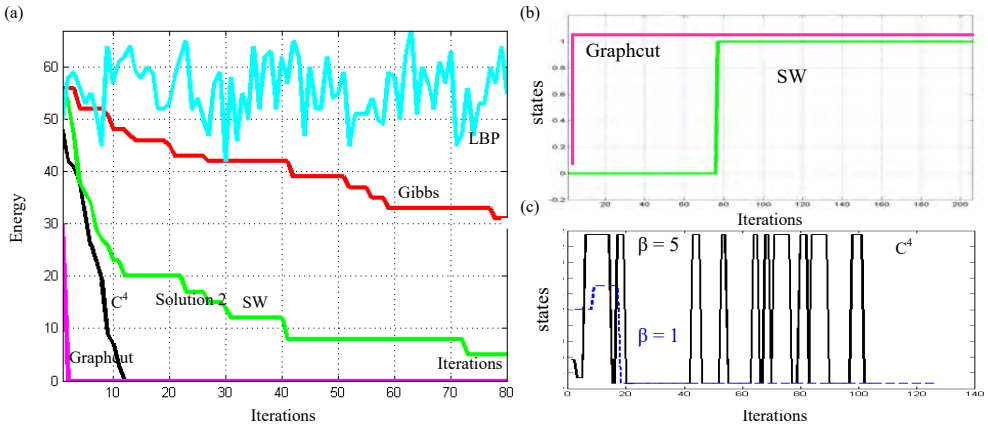


图 6.30:（a）在 Ising 模型 vs 时间上，C^4，SW，Gibbs 采样器，图形分割和 LBP 的能量图。（b），（c）状态（由算法访问）及时进行图形分割，SW 和 C^4。一旦 SW 和图形分割达到第一个解，它们就会卡住，同时 C^4 保持在两个最小能量状态之间交换。C^4 结果显示 $\beta = 1$ 且 $\beta = 5$. Porway 和 Zhu[42] 提供.

图 6.29显示了启动算法的典型初始状态，以及两个具有最小（即 0）能量的解。图 6.30(a) 显示了 C^4，Gibbs 采样器，SW，图形分割和 LBP 的能量对时间的曲线图。在约 10 次迭代中,C^4 在所有的五种算法中收敛地第二快，落后于图形分割。由于图的循环性，置信度传播无法收敛，而 Gibbs 采样器和传统的 Swendsen-Wang 无法快速满足约束，因为它们在每次迭代时都没有更新足够的空间。这表明 C^4 具有非常低的老化时间.

图 6.30(b) 和 (c) 显示了每次迭代时访问的状态。我们以 3 个级别表示状态：曲线分别在两个最小能量状态下到顶或到底，而在所有其他状态下到达中间。这里我们只比较图形分割，SW 和 C^4，因为它们是在合理的时间内收敛到解决方案的唯一算法。C^4 清楚地交换解，同时 SW 和图形分割卡在他们的第一个解中。这是因为 C^4 可以沿着负边和正边分组以同时更新系统的大部分，而 Swendsen-Wang 被阻止提出在解空间的较小部分上低概率移动.

我们还比较了 $\beta = 1$ 和 $\beta = 5$ 的实验结果。图 6.30(c) 显示了采样器随时间变化所访问的状态。在 $\beta = 1$ 的情况下，C^4 收敛需要更长的时间，因为它不能形成具有高概率的大分量。然而，随着 β 变大，C^4 在空间中非常快速地朝向解迈出步骤并且可以在解状态之间快速移动。我们已经发现退火策略，其中 $q_e = 1 - e^{-\beta/T}$ 且 T 被调整，使得 q_e 在实验过程中从 0 移动到 1 也很有效.

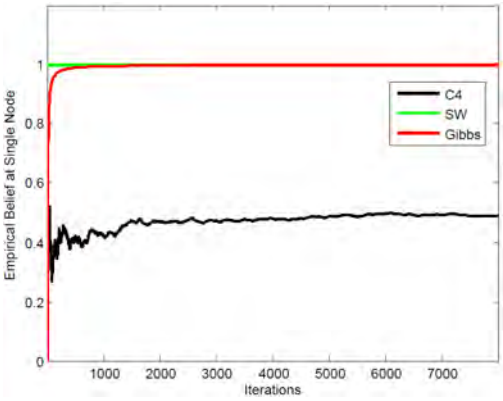


图 6.31: 对 Gibbs，SW 和 C^4Ising 模型的单个位置的边际置信度进行比较。C^4 正确地收敛到 0.5，而其他算法只找到单个解状态。LBP 不会收敛，因此我们没有在这个图中显示出不稳定的置信度. Porway 和 Zhu[42] 提供.

我们最后比较每个算法计算每个节点的估计边际置信度.LBP 直接计算这些置信度，但我们可以通过运行每个算法来估计 Gibbs 采样，SW 和 C^4，并在给定先前状态的每个节点的每次迭代中记录经验均值。。图 6.31显示了 4 种算法中，每种算法对 Ising 模型站点之一随时间的置信度。LBP 不收敛，因此随着时间的推移有一个噪声估计，为了清楚起见没有绘制，Gibbs 和 SW 收敛到概率为 1，因为它们陷入单一解状态，而 C^4 接近 0.5，因为它在两个状态之间不断翻转.

我们在上面进行了相同的实验，但这次每个站点可以采用七种可能的颜色之一 ($L = 7$)。在这个例子中，我们有大量具有最小能量的相等状态（棋盘图案）.

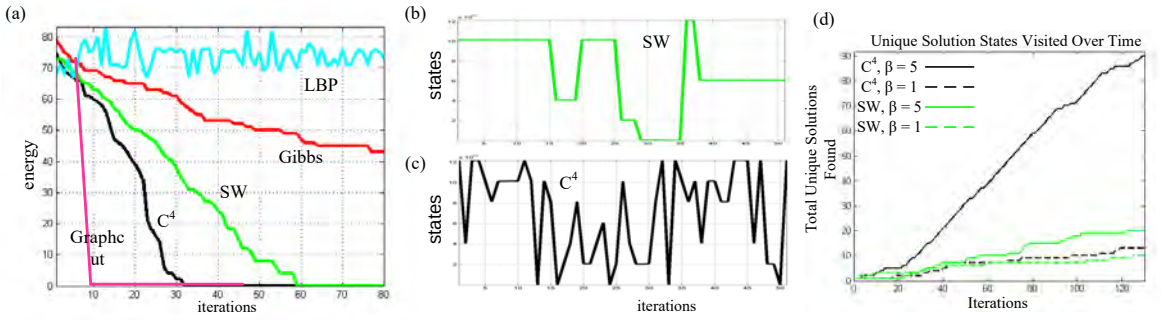


图 6.32: （a）Potts 模型 ($L = 7$) 与时间的 C^4，SW，Gibbs 采样器和 LBP 的能量图。(b) (c)SW 和 C^4 算法随时间访问的最小能量状态。（d）发现的唯一解总数与 SW 和 C^4 的时间对比，$\beta = 1$ 且 $\beta = 5$. Porway 和 Zhu[42] 提供.

图 6.32(a) 描绘了每种算法随时间的能量收敛。图形分割再次收敛到众多解中的一个。与 $L = 2$ 模型的情况不同，SW 这次能够找到多个解，如图 6.32(b) 所示。图 6.32(c) 显示了 SW 和 C^4 随时间访问能量最小的不同状态的数量。我们看到 C^4 在给定的时间限制内探测了更多的状态，这再次表明 C^4 更具动态性，因此具有快速的混合时间--这是 MCMC 算法效率的关键指标。我们还比较了 $\beta = 1$ 与 $\beta = 5$ 的情况。

再一次，我们看到 $\beta = 1$ 并没有为 C^4 提供足够强的连接以移除局部最小值，因此它找到与 Swendsen 一样多的唯一解（约 13）。然而，当 β 增加到 5 时，数量从 13 增加到 90. 因此，当 β 很高时，C^4 可以比其他方法更快地在解空间中移动，并且可以发现大量唯一解状态.

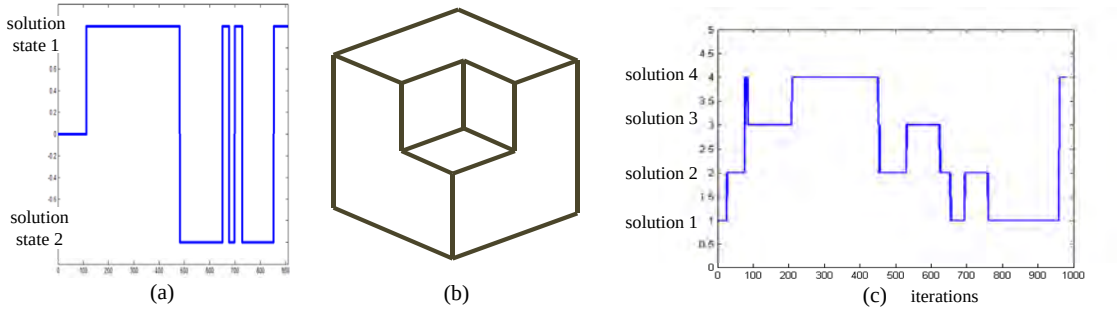


图 6.33: 在解释之间交换状态的实验结果:(a)C4 访问 Necker 立方体的状态。(b) 带有外立方体和内立方体的线条图。(c)C^4 访问双重立方体的状态. Porway 和 Zhu[42] 提供.

前两个示例基于 MRF 模型，其边取决于位置。现在我们考虑候选图上的线条图解释。我们使用两个具有多种稳定解释的经典实例或解：(i) 图 6.23中的 Necker 立方体有两种解释，(ii) 图 6.33中带有双重立方体的线条图，有四种解释。在这些状态之间交换涉及到同时翻转 3 或 12 行。我们的目标是测试算法是否可以随时间计算多个不同的解.

我们在候选图上采用类似 Potts 的模型。线条图中的每一行都是 Potts 模型中的一个节点，该节点可以采用八个线条绘制标签中的一个来指示边缘是凹面，凸面还是深度边界。有关连贯的线条图标签的深入讨论，请参见 [49]。我们在共享交汇点的任意两条线之间的候选图中添加了边。在每个交汇点，每条线只有一小组有效标签可在 3D 世界中实现。我们在成对的线标签之间添加正边，这些线标签与这些结点类型之一相一致，而线标签之间的负边则不是。因此，我们根据它们形成的连接类型来模拟相邻线标签的成对相容性.

在这些实验中，我们设定 $\beta = 2$，得到 $q_e = 0.865$。图 6.33(a) 和 (c) 绘制了算法随时间访问的状态。我们再次看到 C^4 可以在 CSP 解算器或其他 MCMC 方法可能卡住的情况下快速切换解.

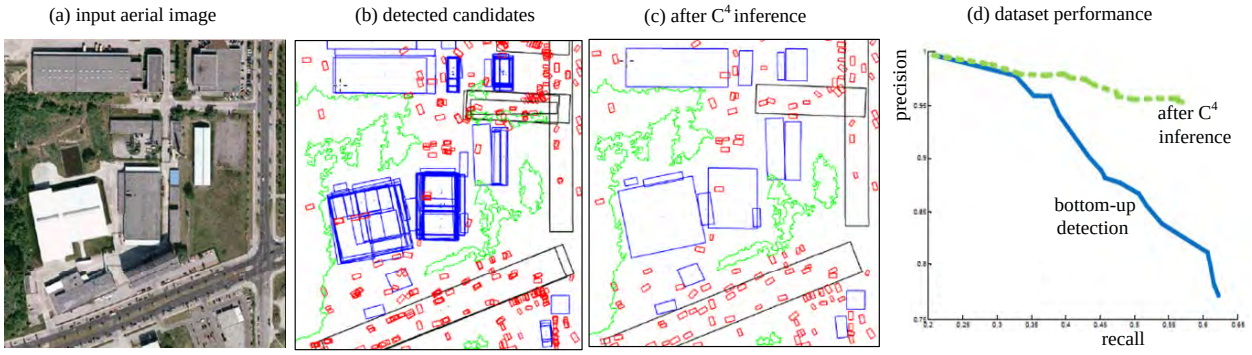


图 6.34: C^4 在航拍图像分析中的应用。（a）谷歌地球的一部分航拍图像。（b）一组自下而上的目标检测，每个目标都是候选对象，即候选图中的一个节点。请注意需要关闭的大量假阳性。（b）C^4 选择的最终提议子集代表场景。C^4 删除了与先前不一致的候选对象。（c）航拍图像数据集上像素级性能的精确召回曲线. PorwayZhu[42] 提供.

在下一个实验中，我们使用 C^4 来解析航拍图像。这个实验是 [43] 工作的延伸。在 [43] 中，航空图像表示为目标组的集合，通过统计外观约束相关联。在推断之前，在离线阶段自动学习这些约束.

Method	False Positives per image	Detection Rate (%)
LBP	85.32	0.668
ICM	82.11	0.768
SW	87.91	0.813
C^4	**83.04**	**0.875**

表 6.2: 每个图像的假阳性和使用 Loopy BP，SW，ICM 和 C^4 进行航拍图像分析的检测率.

我们通过让每个自下而上检测到的窗口成为图中的顶点来创建我们的候选图，通过边连接，其概率与这些目标的相容性成比例。每个候选都可以打开或关闭，指示它是否在场景的当前解释中.

通过检查其两个节点之间的能量 $\varepsilon = \phi(x_i, x_j)$，将每个边分配为正或负，并给出打开的概率 q_e。如果 $\varepsilon > t$，则边标记为负，如果 $\varepsilon < t$，则边标记为正，其中 t 是用户选择的阈值。在我们的实验中，我们让 $t = 0$. 这样我们创建数据驱动的边缘概率并确定 C^4 的正边和负边类型.

在这个实验中，我们使用标记的航拍图像学习了可能的目标配置的先验模型。在一组共 50 多个图像中，每个图像中标记了目标边界。我们测试了从谷歌地球收集的五个大型航拍图像的结果，这些图像也是手工标记的，这样我们就可以测量 C^4 提高了最终检测结果的程度。虽然我们只使用五个图像，但每个图像大于 1000×1000 像素并包含数百个目标，因此人们还可以将评估视为跨越 200×200 像素的 125 个图像.

图 6.34显示了解析的空中场景的示例。自下而上检测到的窗口被视为候选，许多是假阳性。使用 C^4 后最小化全局能量函数; 但是，我们保留了最能满足系统约束的子集。在 C^4 排除不相容的提议后，假阳性率大大降低。图 6.34(d) 显示了使用 C^4 与仅自下而上提示的航拍图像目标检测的精确召回曲线。我们可以看到，以绿色虚线绘制的 C^4 曲线具有比自下而上检测更高的精度，即使召回增加也是如此。我们还比较了使用 C^4 而不是 LBP，ICM 和 SW 的类似误报率的结果。结果如表 6.2所示

6.9.6 分层图上的 C^4

在本节中，我们将讨论平面图的一致性，并将 C^4 从平面图扩展到层次图。然后，我们解决涉及两个以上站点的高阶约束.

在 C^4 算法的每次迭代中，假设我们已经概率地打开了边，原始图形 $G = < V, E >$ 变为 $G_{on} = < V, E_{on} >$ with $E = E_{on} \cup E_{off}$, $E_{on} = E_{on}^{+} \cup E_{on}^{-}$, and $E_{off} = E_{off}^{+} \cup E_{off}^{-}$。正如我们在 6.9.2节中讨论的那样，每个 ccp 的图 G_{on} 中的所有节点共享相同的标签，并且期望形成耦合的部分解。但是，如果图 G 中的约束不一致，则 ccp 中的某些节点可以通过 E_{off}^{-} 中的边连接。尽管在 ccp 中没有打开这样的负边，但它们表明 ccp 中的某些节点可能彼此冲突。这可能不是一个严重的问题，例如，负边可能只是表示柔性约束，例如重叠窗口，这在最终解中是可以接受的.

图 6.35显示了负边是刚性约束的示例。如果我们尝试使用平面候选图解鸭/兔幻觉，则 ccp 可能包含 $\{'eye','nose','head'\}$，这是不一致的。我们称这个分组为 *l*ove triangle.

Definition 6.7 在图 G 中，如果在 i，j 之间存在由所有正边组成的路径，则称由负边连接的两个节点 i, j 属于 love triangle.

Definition 6.8 如果在连接 ccp 中的两个节点的 E 没有负边, 即 $\{e : i, j \in ccp\} \cap E^{-} = \emptyset$，则认为 ccp 在图

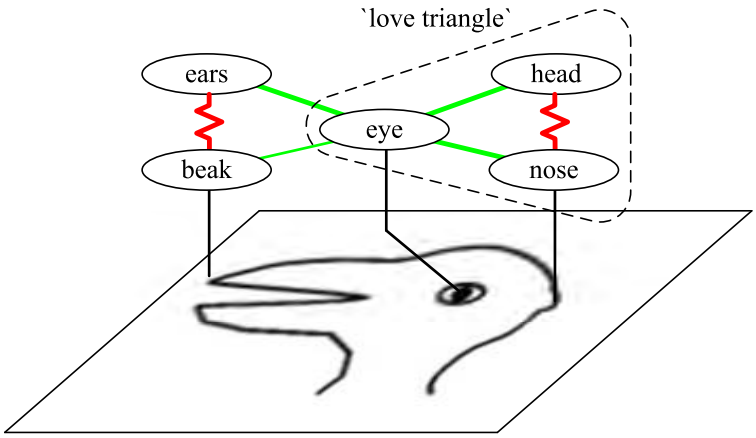


图 6.35: 尝试用平面 C^4 解决鸭/兔幻觉。我们看到很有可能在图的左侧和右侧形成爱三角，这使得约束满足非常困难. Porway 和 Zhu[42] 提供.

G 中是一致的。如果图 G 的所有 ccp 在 C^4 中始终一致，则称图 G 是一致的.

当图一致时，我们确保获得有效的解。所谓 love triangles 的存在是产生不一致的 *ccp*'s 的唯一原因。为此我们可以很容易地证明以下命题.

Proposition 3 *在没有爱三角的情况下，图 G 将是一致的.*

在图生成爱三角的主要原因，主要是在图中, 大多数是在候选图中，是某些节点被多个标签重载，因此它们与冲突的节点耦合。例如，节点' 眼' 应该是' 兔眼' 或' 鸭眼'，它应该分成两个由负边连接的相互冲突的候选节点。这样它就可以消除爱三角。图 6.36说明了我们可以通过将节点 1 分成节点 1 和 1′ 来移除爱三角，因此我们将具有一致的 *ccp*.

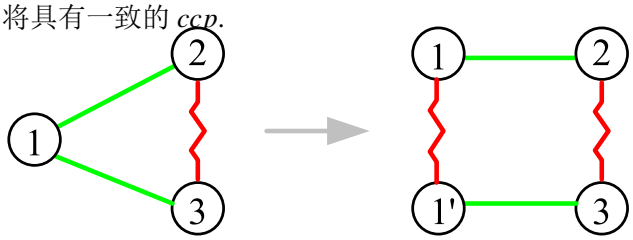


图 6.36: 在候选图中破坏爱三角. Porway 和 Zhu[42] 提供.

我们需要解决的另一个常见问题涉及到 2 个以上节点的高阶约束。图 6.37显示了鸭/兔幻觉的分层图表示。这是一个有两层的候选图。顶层包含两个隐藏的候选假设：鸭子和兔子。这两个节点分别在第 1 层中分解为三个部分，因此在它们之间施加高阶约束。现在，部分假设专门针对鸭眼，兔眼等。连接两个目标节点的负边是从它们重叠的子节点继承的.

这种分层候选图是在运行中构建的，其中的节点由多个自下而上检测和绑定过程以及自上而下的预测过程生成。我们将 Wu 和 Zhu [61] 称为目标解析中的各种自下而上/自上而下的过程。在该图中，以与平面候选图相同的方式在相同层上的节点之间添加正边和负边，而父子节点之间的垂直连结是确定性的.

通过在每一层概率地打开/关闭正边和负边，C^4 获得 *ccp*'s 和 *cccp*'s，如平面候选图中所示。在这种情况下，一个 *ccp* 包含一组在水平和垂直方向上耦合的节点，因此表示部分解析树。一个 *cccp* 包含多个竞争解析树，它们将在一个步骤中交换。例如，图 6.37中的左侧面板分别显示了鸭子和兔子的两个

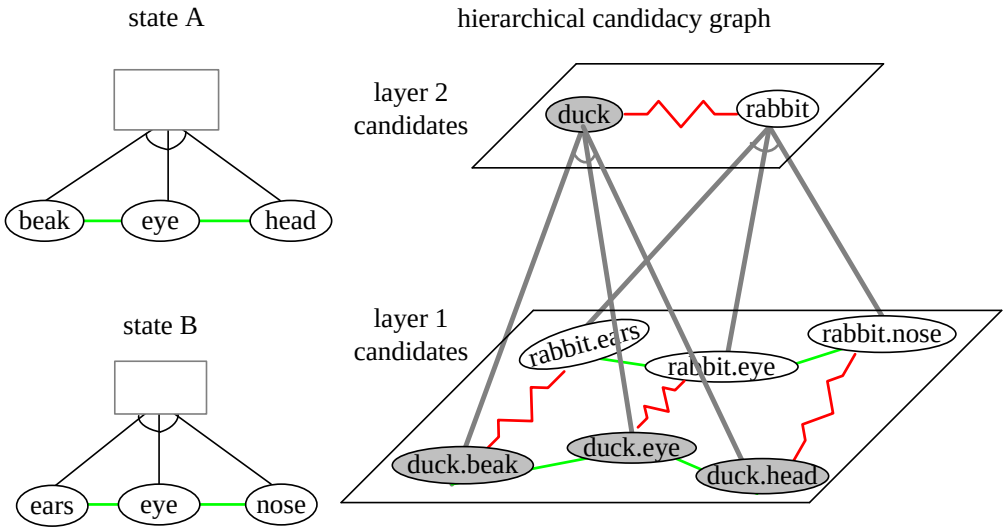


图 6.37: 尝试使用分层 C^4 解决鸭/兔幻觉。树定义了包含每个对象的部分。根据这些树对节点进行分组，从而创建更高级别的节点。较高级别的节点继承负面约束. Porway 和 Zhu[42] 提供.

ccp's，它们与候选图中的负边相连。此分层表示还可以消除由于重载标签导致的不一致。也就是说，如果某个部分由多个目标或目标实例共享，我们需要在分层候选图中创建多个实例作为节点.

6.9.7 C^4 分层实验

为了证明分层 C^4 比平面 C^4 的优势，我们提出了一个解释鸭/兔幻觉的实验.

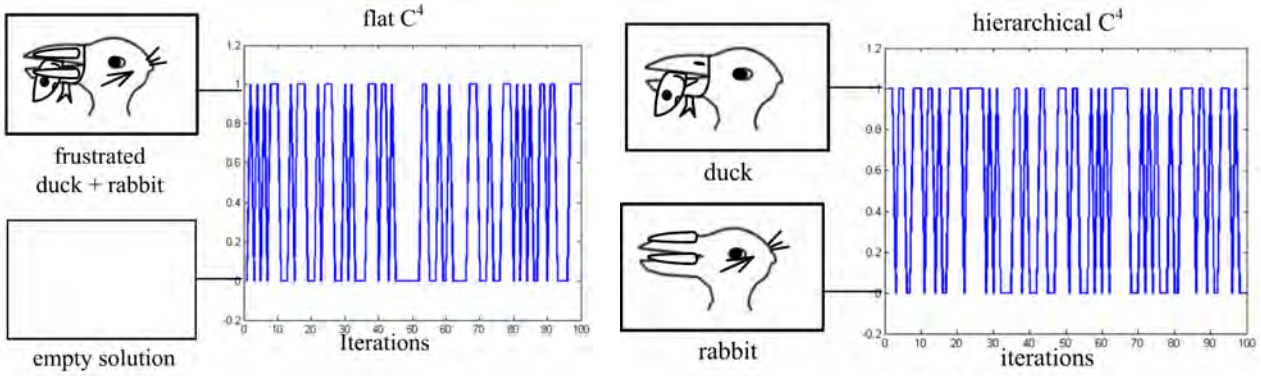


图 6.38: 左：鸭/兔幻觉的平面 C4 结果。由于爱三角，C^4 在两个不可能状态之间交换。右：鸭/兔幻觉的分层 C^4 结果。C^4 现在在两个正确的解之间均匀交换. 改编自 Porway 和 Zhu [42].

如上所述，图 6.35中平面候选图上的 C^4 创建了两个爱三角。图 6.38的左侧显示了鸭/兔幻觉上平面 C^4 的结果。C^4 在两个状态之间连续交换，但是这两个状态要么所有节点都打开，要么所有节点关闭，这两个状态都不是有效的解。图 6.38的右侧显示了将分层 C^4 应用于鸭/兔幻觉的结果。我们为鸭/兔幻觉定义了一棵树，包括鸭，{,,}, 或兔子 {,,}。结果是，算法立即找到两个解，然后继续统一交换它们。这些结果表明，分层 C^4 可以帮助引导算法得到更稳健的解，并消除爱三角的影响.

练习

问题 1。具有耦合马尔可夫链的 Ising/Potts 模型的精确采样。我们考虑 Ising 模型在 $n \times n$ 点阵中（$n = 64$，如果你有一台快速计算机，或许你可以尝试 $n = 128$）与 4 最近邻。X 是在点阵上定义的图像，每个位置的变量 X_s 取 0，1 中的值。模型是

$$p(X) = \frac{1}{Z} \exp\{\beta \sum_{<s,t>} 1(X_s = X_t)\}$$

我们用 Gibbs 采样器模拟两个 Markov 链：:

- MC1 以所有站点为 1（称为白链）开始，其状态由 X^1 表示;
- MC2 以所有站点为 0（称为黑链）开始，其状态由 X^2 表示。

在每一步中，Gibbs 采样器在两个图像中拾取一个位置 s，并计算条件概率，

$$p(X_s^1 | X_{\partial s}^1) \text{ and } p(X_s^2 | X_{\partial s}^2)$$

它根据上述两个条件概率更新变量 X_s^1 和 X_s^2，并共享相同的随机数 $r = rand[0,1]$。两个马尔可夫链被认为“耦合”。

1) 证明在任一步中 $X_s^1 \geq X_s^2, \forall s$。也就是说，白链总是在黑链上方。

2) 经过多次扫描后，当两条链相遇时，即 $X_s^1 = X_s^2 \forall s$，他们被认为“合并”。它们将永远保持在相同的状态，因为每一步它们都由相同的随机数驱动。我们用 τ 表示合并时间（扫描次数）。时间 τ 后的图像被称为是来自 Ising 模型的精确样本。

在扫描中绘制两个链状态（使用它们的总和 $\sum_s X_s^1$ 和 $\sum_s X_s^2$），如图 7.8所示。并在两条链合并时显示图像。尝试使 $\beta = 0.6, 0.7, 0.8, 0.83, 0.84, 0.85, 0.9, 1.0$。

3) 绘制 τ 与 β 的曲线（使用上述参数），以确定在 $\beta = 0.84$ 附近是否存在临界减速。

问题 2。Ising/Potts 模型的聚类采样。为简单起见，我们将 Ising 模型视为 $n \times n$ 点阵（n 介于 64 和 256 之间）与 4 最近邻。X 是在点阵上定义的图像（或状态），每个位置的变量 X_s 取 $\{0,1\}$ 中的值。模型是

$$p(X) = \frac{1}{Z} \exp\{\beta \sum_{<s,t>} 1(X_s = X_t)\} = \frac{1}{Z} \exp\{-\beta \sum_{<s,t>} 1(X_s \neq X_t)\}$$

当 $n \times n$ 足够大时，我们从物理学中得知 $\pi(X)$ 的概率质量集中在下面的集合上，集合外的概率为零:

$$\Omega(h) = \{X : H(X) = h\}, \quad H(X) = \frac{1}{2n^2} \sum_{<s,t>} 1(X_s \neq X_t)$$

$H(X)$ 是 X 的“充分统计”。直观地，它测量 X 中的总边界（裂缝）的长度，并且通过边的数量来标准化。如果 $H(X_1) = H(X_2)$，则两个图像 X_1 和 X_2 具有相同的概率。理论上，在没有相变的情况下，β 和 h 之间存在一一对应关系，即 $h = h(\beta)$。因此，根据经验，我们可以通过监测 $H(X)$ 是否随时间收敛到恒定值 h 来判断收敛。

我们选择 3 个 β 值: $\beta_1 = 0.6, \beta_2 = 0.8, \beta_3 = 0.84$。我们在合并时间 t_1, t_2, t_3(扫描) 处有三个图像 X_1, X_2, X_3。从这些图像中，我们分别计算其充足的统计量 h_1^*, h_2^*, h_3^*。

对每个 $\beta_i, i = 1,2,3$，我们使用聚类采样运行马尔可夫链。

- MC1 从恒定图像（黑色或白色）开始—$h = 0$ 最小;
- MC2 从棋盘图像开始 —$h = 1$ 最大。

因此，当在 h_i^* 处相遇时，我们认为他们已经收敛到了 $\Omega(h_i^*)$。
1) 绘制当前状态 $X(t)$ 随时间 t 的充足统计量 $H(X)$ 并且当 h 在距离 h_i^* 的 epsilon 距离内时停止。
2) 在图中标记 Gibbs 收敛时间 t_1, t_2, t_3（扫描），以便在问题 1 中对三个参数和 Gibbs 采样器收敛进行比较。（这种比较可能对 Gibbs 采样器有点不公平，因为它可能在合并之前已收敛到 $\Omega(h_i^*)$。）
3) 绘制 CPs(在每个步骤（扫描）一起翻转的像素数量) 的平均大小，并在三个 $\beta_i, i = 1,2,3$ 设置下比较它们。

使用两种版本的聚类采样算法重复实验:

版本 1: 在整个图像上形成 CP’ s 并随机翻转它们。所以每一步都是一次扫描。

版本 2: 随机挑选一个像素，从中生成一个 CP，并仅翻转此 CP。累计已翻转的像素数量，并将数字除以 n^2 以获得扫描数。

测试总数为：3 个温度 $\times$ 2 个初始状态 $\times$ 2 个 SW 版本 = 12 次试验。你也可以为 $\beta_3 = 1.0$ 运行两个 MC，看它快速收敛（两个 MC 在 $H(X)$ 中相遇）。

参考文献

[1] Wilhelm Ackermann. Zum hilbertschen aufbau der reellen zahlen. *Mathematische Annalen*, 99(1):118–133, 1928.

[2] Krzysztof R Apt. The essence of constraint propagation. *Theoretical computer science*, 221(1):179–210, 1999.

[3] Adrian Barbu and Song-Chun Zhu. Generalizing swendsen-wang to sampling arbitrary posterior probabilities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1239–1253, 2005.

[4] Adrian Barbu and Song-Chun Zhu. Generalizing swendsen–wang for image analysis. *Journal of Computational and Graphical Statistics*, 16(4), 2007.

[5] Julian Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 259–302, 1986.

[6] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, 2001.

[7] Simon R Broadbent and John M Hammersley. Percolation processes: I. crystals and mazes. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 53, pages 629–641. Cambridge University Press, 1957.

[8] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, pages 282–295. 2010.

[9] Haili Chui and Anand Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2):114–141, 2003.

[10] Colin Cooper and Alan M Frieze. Mixing properties of the swendsen-wang process on classes of graphs. *Random Structures and Algorithms*, 15(3-4):242–261, 1999.

[11] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein, et al. *Introduction to algorithms*, volume 2. MIT press Cambridge, 2001.

[12] L. Ding, A. Barbu, and A. Meyer-Baese. Motion segmentation by velocity clustering with estimation of subspace dimension. In *ACCV Workshop on Detection and Tracking in Challenging Environments*, 2012.

[13] Liangjing Ding and Adrian Barbu. Scalable subspace clustering with application to motion segmentation. *Current Trends in Bayesian Methodology with Applications*, page 267, 2015.

[14] Robert G Edwards and Alan D Sokal. Generalization of the fortuin-kasteleyn-swendsen-wang representation and monte carlo algorithm. *Physical review D*, 38(6):2009, 1988.

[15] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *CVPR*, 2009.

[16] Pedro F Felzenszwalb and Joshua D Schwartz. Hierarchical matching of deformable shapes. In *CVPR*, pages 1–8, 2007.

[17] Cornelis Marius Fortuin and Piet W Kasteleyn. On the random-cluster model: I. introduction and relation to other models. *Physica*, 57(4):536–564, 1972.

[18] Michael Fredman and Michael Saks. The cell probe complexity of dynamic data structures. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 345–354, 1989.

[19] Bernard A Galler and Michael J Fisher. An improved equivalence algorithm. *Communications of the ACM*, 7(5):301–303, 1964.

[20] Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. CRC press, 2013.

[21] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 6:721–741, 1984.

[22] Walter R Gilks and Gareth O Roberts. Strategies for improving mcmc. In *Markov chain Monte Carlo in practice*, pages 89–114. Springer, 1996.

[23] Vivek K Gore and Mark R Jerrum. The swendsen–wang process does not always mix rapidly. *Journal of statistical physics*, 97(1-2):67–86, 1999.

[24] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[25] David M Higdon. Auxiliary variable methods for markov chain monte carlo with applications. *Journal of the American Statistical Association*, 93(442):585–595, 1998.

[26] Mark Huber. A bounding chain for swendsen-wang. *Random Structures & Algorithms*, 22(1):43–59, 2003.

[27] Scott Kirkpatrick, MP Vecchi, et al. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[28] Vladimir Kolmogorov and Carsten Rother. Minimizing nonsubmodular functions with graph cuts-a review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(7):1274–1279, 2007.

[29] M Pawan Kumar and Philip HS Torr. Fast memory-efficient generalized belief propagation. In *Computer Vision–ECCV 2006*, pages 451–463. Springer, 2006.

[30] Sanjiv Kumar and Martial Hebert. Man-made structure detection in natural images using a causal multi-scale random field. In *CVPR*, volume 1, pages I–119. IEEE, 2003.

[31] John D Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289. Morgan Kaufmann Publishers Inc., 2001.

[32] F. Lauer and C. Schnörr. Spectral clustering of linear subspaces for motion segmentation. In *ICCV*, 2009.

[33] Liang Lin, Kun Zeng, Xiaobai Liu, and Song-Chun Zhu. Layered graph matching by composite cluster sampling with collaborative and competitive interactions. In *CVPR*, pages 1351–1358, 2009.

[34] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *ICML*, 2010.

[35] Jun S Liu, Wing H Wong, and Augustine Kong. Covariance structure and convergence rate of the gibbs sampler with various scans. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 157–169, 1995.

[36] Jun S Liu and Ying Nian Wu. Parameter expansion for data augmentation. *Journal of the American Statistical Association*, 94(448):1264–1274, 1999.

[37] Alan K Mackworth. Consistency in networks of relations. *Artificial intelligence*, 8(1):99–118, 1977.

[38] AK Macworth. Interpreting pictures of polyhedral scenes. *Artificial Intelligence*, 4(2):121–137, 1973.

[39] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[40] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *NIPS*, 14:849–856, 2001.

[41] Judea Pearl. Heuristics. intelligent search strategies for computer problem solving. *The Addison-Wesley Series in Artificial Intelligence, Reading, Mass.: Addison-Wesley, 1985, Reprinted version*, 1, 1985.

[42] Jacob Porway and Song-Chun Zhu. C^ 4: Exploring multiple solutions in graphical models by cluster sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1713–1727, 2011.

[43] Jake Porway, Qiongchen Wang, and Song Chun Zhu. A hierarchical and contextual model for aerial image parsing. *International journal of computer vision*, 88(2):254–283, 2010.

[44] Renfrey Burnard Potts. Some generalized order-disorder transformations. In *Proceedings of the Cambridge Philosophical Society*, volume 48, pages 106–109, 1952.

[45] James Gary Propp and David Bruce Wilson. Exact sampling with coupled markov chains and applications to statistical mechanics. *Random structures and Algorithms*, 9(1-2):223–252, 1996.

[46] S. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. *IEEE Trans. on PAMI*, 32(10):1832–1845, 2010.

[47] Azriel Rosenfeld, Robert A Hummel, and Steven W Zucker. Scene labeling by relaxation operations. *Systems, Man and Cybernetics, IEEE Transactions on*, (6):420–433, 1976.

[48] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.

[49] Kokichi Sugihara. *Machine interpretation of line drawings*, volume 1. MIT press Cambridge, 1986.

[50] Deqing Sun, Stefan Roth, and Michael J Black. Secrets of optical flow estimation and their principles. In *CVPR*, pages 2432–2439, 2010.

[51] Robert H Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. *Physical Review Letters*, 58(2):86–88, 1987.

[52] Martin A Tanner and Wing Hung Wong. The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, 82(398):528–540, 1987.

[53] Antonio Torralba, Kevin P Murphy, and William T Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *CVPR*, volume 2, pages II–762. IEEE.

[54] Lloyd N Trefethen and David Bau III. *Numerical linear algebra*. Number 50. 1997.

[55] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *CVPR*, pages 1–8. IEEE, 2007.

[56] Zhuowen Tu and Song-Chun Zhu. Image segmentation by data-driven markov chain monte carlo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):657–673, 2002.

[57] R. Vidal and R. Hartley. Motion segmentation with missing data using powerfactorization and gpca. In *CVPR*, pages II–310, 2004.

[58] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[59] Yair Weiss. Correctness of local probability propagation in graphical models with loops. *Neural computation*, 12(1):1–41, 2000.

[60] Ulli Wolff. Collective monte carlo updating for spin systems. *Physical Review Letters*, 62(4):361, 1989.

[61] Tianfu Wu and Song-Chun Zhu. A numerical study of the bottom-up and top-down inference processes in and-or graphs. *International journal of computer vision*, 93(2):226–252, 2011.

[62] Donghui Yan, Ling Huang, and Michael I Jordan. Fast approximate spectral clustering. In *SIGKDD*, pages 907–916, 2009.

[63] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *ECCV*, pages 94–106, 2006.

[64] Song-Chun Zhu and David Mumford. *A stochastic grammar of images*. Now Publishers Inc, 2007.

第 6 章　MCMC 的收敛性分析

“在一个非瓶颈中省下的一个小时是海市蜃楼。”- Eliyahu Goldratt

简介

许多从业者在使用 MCMC 时遇到的主要问题之一是收敛速度慢。虽然许多 MCMC 方法已经证明会收敛到目标分布，但整个收敛很大程度上取决于转移矩阵 λ_{slem} 的第二大特征值的大小。因此，基于此数量，νK^n 的收敛率有很多限制。在本章中，派生和实现了一些最有用的边界。通过随机改组一副纸牌来研究这些界限。另外，为了加速收敛过程，解释了交易图，瓶颈和电导的概念。最后，介绍了路径耦合和精确采样的主题，并将这些方法应用于 Ising 模型。

7.1　关键融合主题

设 (ν, K, Ω) 成为马尔可夫链，初始分配 ν，转换内核 K 在 Ω 空间。这个链在某个时间 n 获得的样本遵循分布 $X(t) \sim \nu \cdot K^n \xrightarrow[n]{\infty} \pi$。$\nu K^n$ 的收敛是使用总变差 $\|\nu K^n - \pi\|_{TV}$ 来衡量的。如果此数量接近 0 作为 $n \to \infty$，则该链接收到 π。

$$K^n = \sum_{i=1}^{n} \lambda_i v_i u_i$$

回想一下章节中定义的链的这些有用特性 3.4:

i) 第一次命中状态 i（在有限状态下）

$$\tau_{\text{hit}}(i) = \inf\{n \geqslant 1; x_n = i, x_0 \sim \nu_0\}, \qquad \forall i \in \Omega$$

ii) 状态 i 的首次返回时间

$$\tau_{\text{ret}}(i) = \inf\{n \geqslant 1; x_n = i, x_0 = i\}, \qquad \forall i \in \Omega$$

iii) 混合时间

$$\tau_{\text{mix}}(i) = \min_n\{\|\nu K^n - \pi\|_{\text{TV}} \leqslant \varepsilon, \forall \nu_0\}$$

我们还可以定义以下概念来表征链。

定义 7.1 *老化期是马尔可夫链进入典型状态子空间之前的预期步数。典型状态的子空间是 Ω 的子空间，其中 $\pi(x)$ 是集中的。*

老化概念不是很精确，因为很难估计马尔可夫链 νK^n 的分布何时足够接近目标分布 π。在高维空间中尤其如此。

定义 7.2 *马尔可夫链 $\mathbf{x} = (x_0, ...,)$ 的状态之间的自相关被定义为*

$$\mathrm{Corr}(\tau) = \frac{1}{T}\sum_{t=t_0+1}^{t_0+T}(x_t - \overline{\mathbf{x}})(x_{t+\tau} - \overline{\mathbf{x}}), \qquad \forall \tau \geq 0.$$

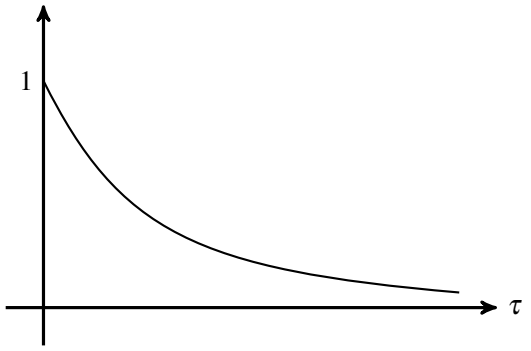


图 7.1: 通常样本之间的自相关性随着滞后 τ 而减少。

高自相关意味着收敛慢，而低自相关意味着快速收敛。我们可以使用 MC 样本进行整合和获取

$$\theta = \int f(x)\pi(x)\mathrm{d}x \cong \frac{1}{T}\sum_{t=t_0+1}^{t+T} f(x_t) = \hat{\theta},$$

$$\mathrm{var}(\hat{\theta}) = E_{\text{samples}}\left[(\hat{\theta} - \theta)^2\right] = \frac{1}{m}\cdot \text{const},$$

其中 m 是独立样本的有效数量。

7.2 实用的监测方法

确定性算法收敛到一个点，通常可以监视到该点的收敛。例如，在最大似然估计（MLE）中，我们可以检查似然函数 $f(x)$ 以查看算法是否已收敛。相比之下，MCMC 算法是随机的，很难确定是否发生了收敛。但是，有几种方法可以让我们监控融合过程，包括

1. 监测 $\pi(x)$ 的足够统计数据，例如边界长度，总能量等。统计数据是空间平均值（样本的扰动）或时间（样本数量）。这种方法将状态空间 Ω 投射到足够的统计数据 H 的空间中。对于 H 的任何特定值，我们有反向空间

$$\Omega_{(h)} = \{x \in \Omega : H(x) = h\}.$$

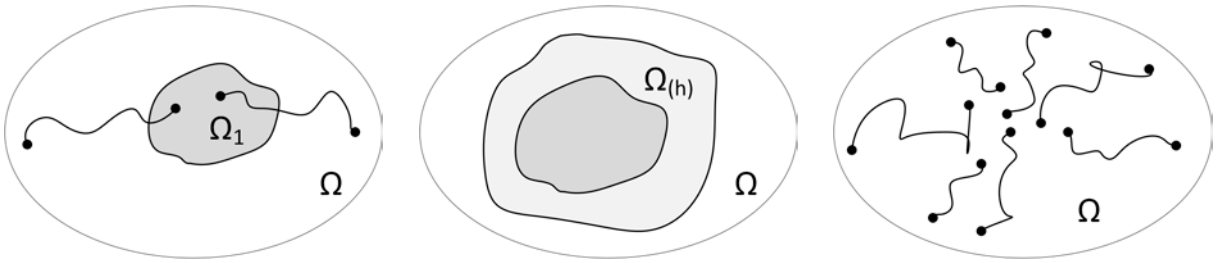


图 7.2: 左图：在老化期后，不同路径到达概率高的区域 Ω_1。中：足够的统计量 H（例如，能量）为我们提供了反向空间（水平集）$\Omega_{(h)}$，用于获取不同的值 h。右：运行多个马尔可夫链可以更好地探索空间 Ω。

2. 从广泛分散的初始状态开始并行运行许多马尔可夫链，并且如果可能的话，从极端状态开始。如 1. 中的监视可以同时进行。例如，在 Ising/Potts 模型中，我们从常数 0/1（白/黑）或白噪声网格开始。

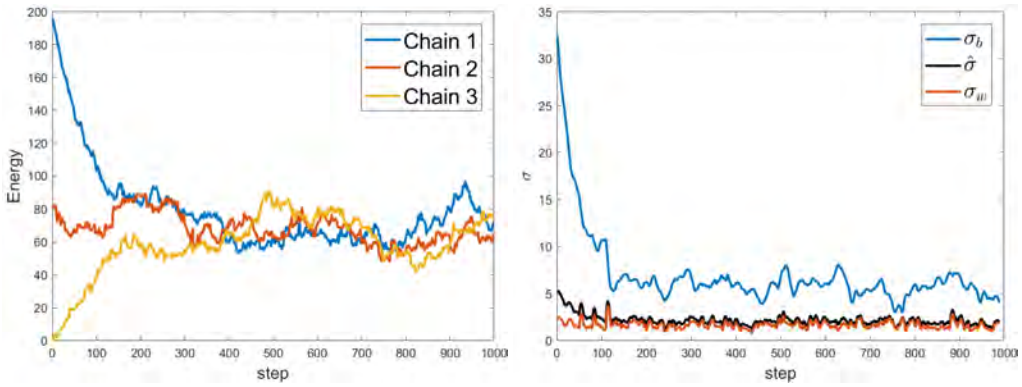


图 7.3: 左：通过检查足够的统计数据 H，监测马尔可夫链已经忘记了初始点。能源。右：σ_w 和 σ_b 和 $\hat{\sigma}$ 近似于真正的 σ。

3. 监视马尔可夫链是否忘记了过去或初始点，如左图 7.3所示。

4. 监测采样温度 T。通过这种方法，我们可以监测 Metropolis-Hastings 算法的拒绝率以及 Gibbs 采样器中 $\pi(x_i|x_{-i})$ 的熵。

5. 模拟 M 不同的马尔可夫链序列，$\{\text{MC}_i; i = 1,2,\cdots,M\}$。我们可以计算单个意味着 ψ_i，以及所有链的总平均值，$\overline{\psi}$。因此，我们有链间方差

$$\sigma_b^2 = \frac{1}{M-1}\sum_{i=1}^{M}(\psi_i - \overline{\psi})^2$$

和链内差异

$$\sigma_w^2 = \frac{1}{M}\sum_{i=1}^{M}\sigma_i^2,\ \sigma_i^2 = \frac{1}{T}\sum_{i=t_0}^{t_0+T}\left(x_i(t) - \psi_i\right)^2,$$

这低估了真正的方差 *sigma*。

然后我们可以估计马尔可夫链的方差为

$$\hat{\sigma} = \frac{T-1}{T}\sigma_{\mathrm{w}} + \frac{1}{T}\sigma_{\mathrm{b}}.$$

这些数量如图 7.3，右图所示。

7.3 卡改组的耦合方法

洗牌一副牌也可以用马尔可夫链代表。我们可以使用卡改组来研究马尔可夫链的耦合方法，其中两个或更多个链独立地开始并且在多个步骤之后缓慢地聚结（相同地移动）。

假设我们有一张 $n = 52$ 牌的套牌。我们可以使用马尔可夫链来回答一些问题，例如：这些卡什么时候彻底洗牌？所有卡都是随机分布的吗？有三种方法可以理解这些问题。

1. 收敛是相对于一个程序，例如洗牌过程，因为在每次改组之后我们得到一个新订单。通过重复这个过程 N 次，我们得到 N decks（安排），并且可以回答问题

   a) 测试出现在给定位置的卡片的分布，i，并将其与 i 的统一卡片分布进行比较。

   b) 跟踪从 i 位置开始的卡片，并检查其位置分布。

2. 检查新牌组是否已忘记其历史记录，因此玩家不能通过记忆原始订单来作弊。

3. 监控牌之间的一些边际统计数据，以便玩家无法根据已经玩过的牌预测下一张牌。

有很多方法可以洗牌，其中两种是在这里展示的。

7.3.1 拖到顶端

拖曳到顶部是一种易于理论研究的简单方法。在每一步中，随机选择卡 i 并放置在牌组的顶部。经过多次移动后，牌组将完全随机。为了连接与甲板 1 相关联的马尔可夫链和与甲板 2 相关联的马尔可夫链，我们在甲板 2 中找到甲板 1 的顶部卡，并将其放在顶部。因此，顶层牌在牌组 1 和 2 之间是相同的。重复该过程直到所有 52 张牌被挑选至少一次。这也被称为“优惠券收藏家的问题”。经过一段时间 T，两个甲板将是相同的，据说已经合并。合并时间 T 具有以下特征：

$$E[T] = n(\frac{1}{n} + \frac{1}{n-1} + \cdots + 1) \cong n\log n;$$
$$\mathrm{var}[T] \cong 0.72n.$$

备注 7.1 *在每一步我们必须在所有 n 卡中进行选择，以便甲板 1 上的洗牌是无偏的，否则甲板 2 不再是随机的。*

备注 7.2 *这两个套牌在每次移动时都用同一张卡 i 连接。*

7.3.2 Riffle 洗牌

对于浅滩洗牌，我们根据二项式 $(n, \frac{1}{2})$ 将 52 张卡分成两个牌组，然后我们将它们洗牌。这样，卡片 1 和 $n-k$ 的卡片数量在甲板 2 中。k 的数量跟随分发

$$K \sim P(k) = \frac{1}{2^n}\binom{n}{k} = \frac{1}{2^n} \cdot \frac{n!}{k!(n-k)!}.$$

这可以更好地理解为反向混洗过程，类似于倒带视频。这意味着我们随机挑选属于甲板 1 的牌，然后我们将它们放在顶部。

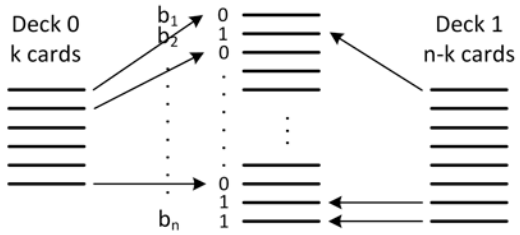


图 7.4: Illustration of rifle shuffling

为此，在每次洗牌时，我们模拟每张卡的二进制位，$b_1, b_2, \cdots, b_n \sim$ 伯努利 $(\frac{1}{2})$，n 次总计，如图 ref fig：ch7：rifle 所示。然后我们回去把所有 0 放在所有 1 的顶部。在 t 洗牌操作之后，原始订单中的每张卡 i 与 $x_i = b_{1i}b_{2i}\cdots b_{ti}$ 的 t 位相关联，如图**??**所示。

当没有进行足够的 shuffle 时，存在具有相同二进制代码 $x_i = x_j$ 的卡，如图 7.5所示。在这种情况下，这些卡之间的顺序与原始套牌中的顺序相同

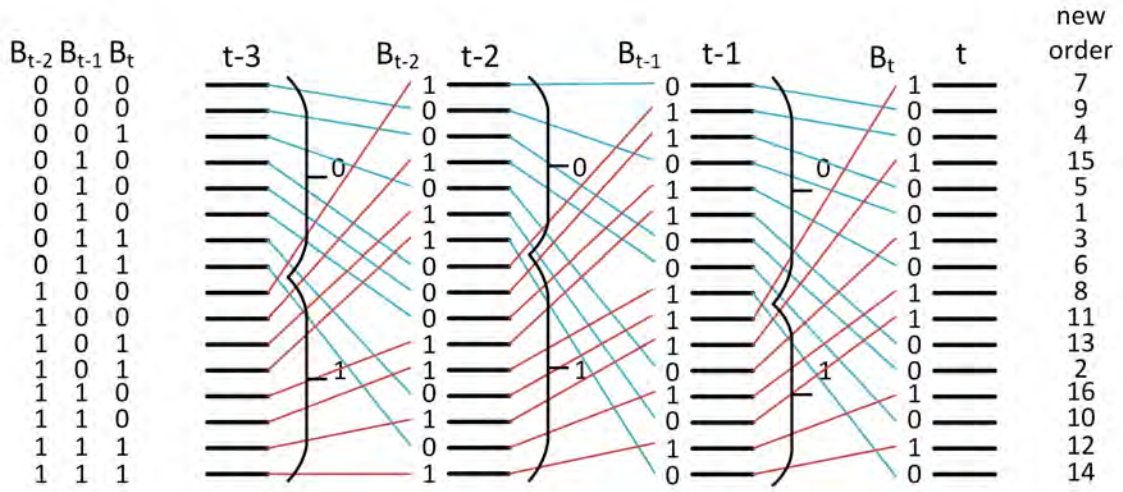


图 7.5: Example of rifle shuffling and the coding bits.

但是，当 t 足够大以至于所有 $\{x_i\}$ 都是不同的时，然后卡的顺序对应于 x_i 的值，因为具有 0 last 的卡在牌组中的另一张牌之上。然后，在 t 之后的排序是完全随机的，因为该顺序仅由位 x_i 决定。

计算 t 这样所有 $\{x_i\}$ 是不同的是与一个经典的统计问题有关：在 2^t 箱子中丢弃 n 球的概率是多少，所以每个 bin 最多只有 1 个球？这些箱子对应于 2^t 位组合 $(b_1, b_2, \cdots, b_t)$ 没有重复，并且球对应于 n 卡。

例 7.1 *为了解决这个问题，请考虑我们有 520 张卡片以相同的方式开始完全分类，每张卡片从上到下编*

号为 *1* 到 *52*。对于每个牌组，我们独立地从伯努利（*.5*）中抽取 *52* 个值。为了洗牌，我们计算 *52* 个值的向量中的 *0* 的数量，并从牌组的顶部取出那么多张牌。然后我们按顺序将它们放在一个“新”牌组中，使它们准确地定位在每个 *0* 的位置。其余的牌同样放置在 *1* 的位置。

为了测试这种方法如何有效地创建随机洗牌的牌组，对于 *520* 个牌组中的每一个，该过程重复 t 次。在每次洗牌之后，我们创建 *52* 个直方图，每个位置对应于牌组中的每个位置。在上下文中，这相当于排列所有 *520* 个甲板并从所有甲板上翻转顶部卡片，然后根据相关联的数字对它们进行分类并堆叠以获得该位置处的卡片分布。理想情况下，我们希望看到这 *52* 个分布中的每一个都大致统一，以代表完美的混乱套牌。电视规范用于衡量分布的接近均匀程度，误差由

$$err(t) = \frac{1}{52}\sum_{i=1}^{52} ||H_{t,i} - \frac{1}{52}||_{TV}.$$

给出

在这个表达式中，$H_{t,i}$ 是甲板位置 i 时 t 的概率分布。预计每次迭代都会缩小所有位置的平均误差，但它不太可能收敛到 *0*。

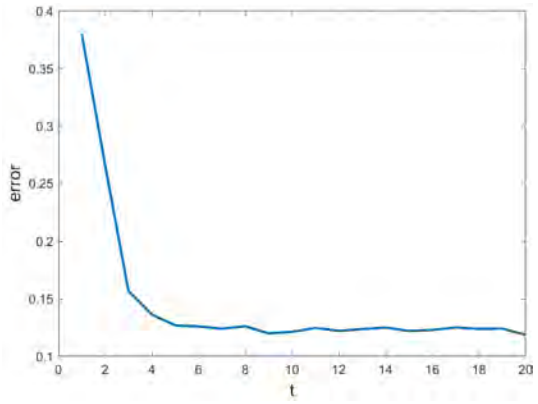


图 7.6: 在 t riffle 洗牌之后，作为电视规范差异的均匀分布偏离。

如上图 *7.6*所示，仅在 *5* 次洗牌之后，错误减少到*.12* 并稳定下来。用于获得随机牌组的传统 *riffle shuffle* 的推荐数量在 *7* 到 *11* 之间。然而，在实践中，与上面使用的肯定随机伯努利过程相比，浅滩洗牌可能不是完全随机的。

7.4 几何界限，瓶颈和电导

在本节中，我们将介绍有关 MCMC 收敛速度的更多关键概念。

7.4.1 几何收敛

让 (ν, K, Ω) 成为马尔可夫链。如果 K 是不可简化的和非周期性的，那么

$$\|\nu \cdot K^n - \pi\|_{\mathrm{TV}} \leqslant c \cdot r^n,$$

其中 c 是常量，$0<r<1$ 是几何率。对于这样的马尔可夫链，存在 $n_0>0$ 这样

$$K^{n_0}(x,y)>0, \qquad \forall x,y\in\Omega.$$

有许多方法可以证明这一点，因为没有指定 r。

i) 使用收缩系数。K 的收缩系数是转换内核中任意两行之间的最大 TV 范数，并通过计算得出

$$C(K)=\max_{x,y}\|K(x,\cdot)-K(y,\cdot)\|_{\mathrm{TV}}.$$

然后对于任何两个概率 ν_1,ν_2 和在 x 上定义的函数 h 使得 $Kh=\sum_y h(y)K(x,y)$ 以下成立：

$$\begin{aligned}\|\nu_1K-\nu_2K\|_{\mathrm{TV}} &= \max\{\|(\nu_1K)h-(\nu_2K)h\|:\|h\|\le 1\}=\max\{\|\nu_1(Kh)-\nu_2(Kh)\|:\|h\|\le 1\} &(7.1)\\ &\le \max\{\frac{1}{2}\max_{x,y}\|Kh(x)-Kh(y)\|:\|h\|\le 1\}\|\nu_1-\nu_2\| &(7.2)\\ &= \frac{1}{2}\max_{x,y}\max\{\|Kh(x)-Kh(y)\|:\|h\|\le 1\}\|\nu_1-\nu_2\|=c(K)\|\nu_1-\nu_2\|. &(7.3)\end{aligned}$$

有关此证明的更多详细信息，请参阅 [3]。

ii) 使用 Perron-Frobenius 定理。召回定理 3.4：

Theorem 3 (Perron-Frobenius 定理) *对于任何原始（不可约和非周期）$N\times N$ 随机矩阵 K，K 具有分别具有特征值 $1=\lambda_1>|\lambda_2|>\cdots>|\lambda_r|$ 与多重性 $m_1...m_r$ 和左右特征向量 $(\mathbf{u}_i,\mathbf{v}_i)$。然后 $\mathbf{u}_1=\pi,\mathbf{v}_1=\mathbf{1}$，和*

$$K^n=\mathbf{1}\cdot\pi'+O(n^{m_2-1}|\lambda_2|^n).$$

从这个结果我们得到了每个起始状态 $\mathbf{x}_0\in\Omega$ 的界限

$$\|\nu K^n-\pi\|_{TV}\leqslant\sqrt{\frac{1-\pi(\mathbf{x}_0)}{4\pi(\mathbf{x}_0)}}\cdot\lambda_{\mathrm{slem}}^n$$

这个界限被称为 Diaconis-Hanlon 边界 [1]。


Persi Diaconis

现在我们需要分析哪些因素约束 λ_{slem}。λ_{slem} 通常与链的特征相关，该链最能阻碍收敛过程，即最坏的情况。这可以是状态，顶点或边缘。有几个与理解 λ_{slem} 有关的重要概念。

交易图（转换图）。回到马尔可夫链内核 K 的比喻作为人与人之间的交易（如例子 3.2），我们定义一个图 $G=<V,E>$ 其中 $V=\Omega$ 是有限状态集，$E=\{<x,y>;x,y\in V,K(x,y)>0\}$ 是边的集合。每个边缘 $e=<x,y>$ 由 $Q(e)=\pi(x)\cdot K(x,y)$ 加权。

此映射的几个属性可用于诊断收敛。通过不可减少性，$\forall x\neq y\in\Omega$，x 和 y 通过许多路径连接。我们定义加权路径 $\Gamma_{xy}\stackrel{\mathrm{def}}{=}(x,\cdots,y)$ 并进一步要求此路径最多包含一次每条边。在这个约束条件下，我们说 Γ_{xy} 的有效长度由

$$\gamma_{xy}\stackrel{\mathrm{def}}{=}\sum_{<s,t>\in\Gamma_{xy}}\frac{1}{\pi(s)K(s,t)}.$$

因此，从 x 到 y 的概率较低意味着更长的有效长度，并且从 x 到 y 需要很长的等待时间。

Bottleneck. G 的瓶颈指标表示图表的整体连通性，并由下式给出

$$\kappa = \max_{e \in E} \sum_{\Gamma_{xy} \ni e} \gamma_{xy} \cdot \pi(x)\pi(y),$$

其中总和是在包含边缘 e 的所有有效路径（无论它们从哪里开始或结束）上获取的。瓶颈本身就是产生最大值的边缘 e^*。直观地说，e^* 可以被认为是图表的金门大桥或巴拿马运河。两个人口稠密的城市/水体/节点体系通过一条高交通路径相连。

通过计算建立的图的瓶颈的过程，有一个结果给出了基于 *kappa* 的收敛的界限。庞加莱 é 不等式意味着

$$\lambda_{\text{slem}} \leqslant 1 - \kappa^{-1}. \tag{7.4}$$

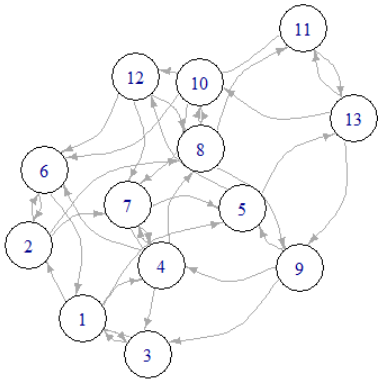


图 7.7: 左图：五个家庭的贸易图。右：未加权的有向图，其顶点集等于五个族图的边集。

例 7.2 为了解决问题，我们考虑在岛上交易的五个家庭的情况下找到瓶颈的例子。假设我们认为瓶颈是边缘（*3,2*）。有许多路径包含边缘（*3,2*）。有一步路径，六条两步路径和二十一条三步路径。其中，仅允许（*2,3,2,3*），因为它重复边缘。还有四步路径，五步路径，等等。我们可以看到，最佳边缘有超过 *10,000* 个与之相关的路径。

为了降低此问题的复杂性，我们改为计算 κ，如下所示。设 G 是有向图的未加权版本。也就是说，G 以矩阵形式表示，将所有非零 *Kxy* 组件转换为 *1*。现在根据 G 的边集完全是 G_0 的顶点的规则，将未加权（有向）图形 G 转换为新的（未加权和定向）图形 G_0。因此，v_1 和 v_2 是 G_0 的边缘，当且仅当 v_1 = （s_1t_1）和 $v_2 = s_2t_2$ 时，某些顶点 $s_1s_2t_1$ 和 t_2 为 G。最后，构建 G_0，这样当且仅当 $t_1 = s_2$ 时，才会有 v_1 到 v_2 的有向边。这在右图 *7.7*中说明。

现在，考虑这个新图形上的所有简单路径 G_0。一条简单的路径是一次触及任何顶点的路径。因此，G_0 上的简单路径沿着 G 的边缘追踪可行路径，使用 G 的每个边缘最多一次。因此，我们可以轻松搜索 G_0 以查找所有简单路径，然后丢弃那些没有我们正在考虑的边缘的路径，计算 $\gamma_{x,y}\pi(x)\pi(y)$ 并对剩余路径求和。对 G_0 中的每个顶点重复此操作（这正是我们最大化的每个可能的参数）并选择最大化总和的顶点。

使用此算法，我们获得边缘 $<3,4>$ 的电导 $\kappa = 90508.08$。由于 $\lambda_{slem} = |.5443 + .1824i| = .57405$，我们观察到 *Poincar é* 满足不等式 $\lambda_{slem} \leqslant 1 - \kappa^{-1}$。

电导. 假设我们将状态空间 Ω 分成两个子空间，例如 $\Omega = S \cup S^c$。然后我们定义子空间之间的转换概率

$$K(S,S^c) =: \sum_{x\in S}\sum_{y\in S^c} K(x,y).$$

设 $\pi(S) = \sum_{x\in S}\pi(x)$ 是 S 的容量并定义

$$Q(S,S^c) =: \sum_{\substack{x\in S\\ y\in S^c}} \pi(x)\cdot K(x,y)$$

作为流出 S。然后给出 G 的电导

$$h =: \min_{S:\pi(S)\leqslant\frac{1}{2}} \frac{Q(S,S^c)}{\pi(S)}.$$

如果电导很小，则存在 S，其中 $\pi(S)$ 很大但是 $Q(S,S^c)$ 很小。这种电导的定义产生了 Cheeger 的不等式 [2]，其中指出

$$1-2h \leqslant \lambda_{\text{slem}} \leqslant 1-\frac{h^2}{2}.$$

这些界限很直观，但在实践中并没有真正指导设计。在实践中，使用启发式算法，例如数据驱动的 MCMC（第 8 章）或 SW 切割（第 6 章）算法来加速马尔可夫链。

7.5 Peskun 的有序和遍历性定理

现在，我们回到设计 MCMC 的早期动机。回想一下 3.6部分中的遍历性定理。请记住，这允许我们执行蒙特卡洛积分来计算参数 θ

$$\theta = \int f(x)\pi(x)\mathrm{d}x \cong \hat{\theta} = \frac{1}{n}\sum_{t=1}^{n} f(x^{(t)}),$$

通过使用从 MCMC 获得的样本 $\{x^{(1)},\cdots,x^{(n)}\} \sim \pi(X)$。马尔可夫链的效率最终通过方差来衡量

$$\mathrm{var}(\hat{\theta}) = \lim_{n\to\infty}\frac{1}{n}\mathrm{var}\Big\{\sum_{t=1}^{n} f(x^{(t)})\Big\}.$$

假设两个 Markov 内核 K_1 和 K_2 具有相同的不变概率 π。我们在集合中的所有这些 K 中引入了部分顺序

$$\Omega_\pi = \{K : \pi K = \pi,\ K\text{不可约和非周期性的}\}.$$

我们说 K_1 主导 K_2，写 $K_1 \succeq K_2$，如果 $K_1(x,y) \geqslant K_2(x,y), \forall x \neq y$。

Theorem 7.3 (Peskun) *如果 $K_1 \succeq K_2$, 然后 $\mathrm{var}(\hat{\theta}_1) \leqslant \mathrm{var}(\hat{\theta}_2)$.*

例 7.3 *考虑以下两个马尔可夫链：*

MC1: $K_1(x,y) = Q(x,y)\cdot\alpha(x,y) = Q(x,y)\cdot\min\left(1,\frac{Q(y,x)\pi(y)}{Q(x,y)\pi(x)}\right)$ *– Metropolis-Hastings 设计。*

MC2: $K_2(x,y) = Q(x,y)\cdot\frac{\pi(y)Q(y,x)}{\pi(x)Q(x,y)+\pi(y)Q(y,x)}$ *– 贝克的设计。*

可以证明 $K_1 \succeq K_2$。

例 7.4 *Metropolized Gibbs* 采样器 $\succeq$ *Gibbs* 采样器。

7.6 路径耦合和精确采样

例 7.5 我们考虑 $n \times n$ 晶格及其由 *4* 个最近晶格组成的图。我们在其原始物理环境中使用 *Ising* 模型，其中它在带电粒子的晶格上模拟磁性材料。每个粒子可以具有两种可能的旋转状态之一，-1（向下）或 1（向上）。设 X 是晶格的自旋状态，因此每个网站 s 的变量 X_s 是自旋状态，取值为 $\{-1,1\}$。我们考虑 $n \times n$ 晶格及其由 *4* 个最近邻居组成的图。我们在其原始物理环境中使用 *Ising* 模型，其中它在带电粒子的晶格上模拟磁性材料。每个粒子可以具有两种可能的旋转状态之一，-1（向下）或 1（向上）。设 X 是晶格的自旋状态，因此每个网站 s 的变量 X_s 是自旋状态，取值为 $\{-1,1\}$。自旋相互作用的模型将正能量分配给相反方向的自旋。形式上，系统的能量由下式给出

$$H(X) = -\sum_{<s,t>\in C} \beta X_s X_t,$$

其中 C 是格子的 *4* 个最近邻居，β 是交互强度。这导致了晶格的每种可能状态的概率测量：

$$P(X) = \frac{1}{Z}\exp^{-H(X)}.$$

我们可以使用 *Gibbs* 采样器模拟两个 *Markov* 链：

1. 白色链从所有站点为 *1* 开始，其状态由 X^1 表示;
2. 黑链从所有站点为 *0* 开始，其状态由 X^2 表示。

在每一步中，*Gibbs* 采样器在两个图像中选取一个站点 s 并计算条件概率，$p(X_s^1|X_{\partial s}^1)$ 和 $p(X_s^2|X_{\partial s}^2)$。它根据上述两个条件概率更新变量 X_s^1 和 X_s^2，并使用相同的随机数 $r \in [0,1]$ 来对两个变量进行采样。在这个过程中，据说两个马尔可夫链是耦合的。

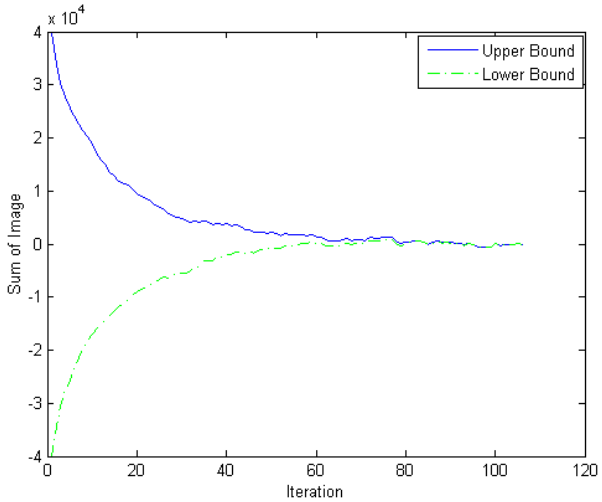


图 7.8: Ising 模型上的总磁化 $\sum_s X_s$，白链和黑链的 $\beta = 0.35$，合并为 $\tau = 105$。

可以在任何步骤中显示 $X_s^1 \geq X_s^2, \forall s$。也就是说，白链总是具有比黑链更大的总和。$\beta = 0.35$ 的示例如图 *7.8*所示。

合并。当这两条链彼此相遇时，即 $X_s^1 = X_s^2, \forall s$，经过多次扫描后，它们被称为已经合并。它们将永远保持在相同的状态，因为它们是相同的，并且在每一步都由相同的随机数驱动。我们用 τ 表示合并时间（扫描）。τ 扫描后的图像是来自 *Ising* 模型的精确样本。

7.6.1 从过去耦合

精确采样的主要概念之一是过去的耦合。我们的想法是从每个状态及时向后运行模拟，跟踪每个链最终的状态。直观的是，一旦两个状态在从时间 $-t$ 到时间 0 的模拟之后映射到单个状态，它们将保持相同。如果使用相同的随机数，则从 $-t-1$ 模拟。从过去耦合确保在有限数量的模拟 M 之后，我们最终得到的状态 i 的 $\rho(i)$ 足够接近链的均衡分布 $\pi(i)$，即小量 $||\rho(i) - \pi(i)|| < \varepsilon$。固定时间向后模拟的输出由 $F_{-M}^0(i)$ 给出，其中 $F_{t_1}^{t_2}$ 定义为组合 $f_{t_2-1} \circ f_{t_2-2} \circ \cdots \circ f_{t_1+1} \circ f_{t_1}$。此输出的几个重要功能包括：

1. 每个 $f_t(i)$ 将状态空间映射到自身，$-M \; leqt \; leq 1$。

2. F_t^0 通过 $F_t^0 = F_{t+1}^0 \circ f_t$ 更新。

3. 合并是在 F_t^0 成为常量映射的时间点，$F_t^0(i) = F_t^0(i'), \forall i, i'$。

Theorem 7.4 *在概率为 1 的情况下，来自过去过程的耦合返回根据马尔可夫链的平稳分布分布的值。*

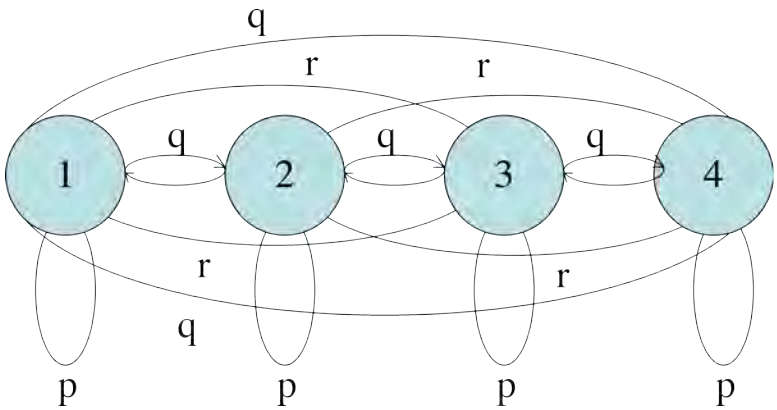


图 7.9: 具有四种状态的马尔可夫链，其中 $p = \frac{1}{3}, q = \frac{1}{4}, r = 1 - p - 2q$。

例 7.6 *考虑图 7.9中所示的四状态马尔可夫链。我们模拟过去的所有状态。在 5 次迭代模拟之后发生了聚结，如图 7.10所示。*

状态空间 S 具有自然的部分排序，这样

$$x \leq y \Rightarrow \phi(x,u) \leq \phi(y,u),$$

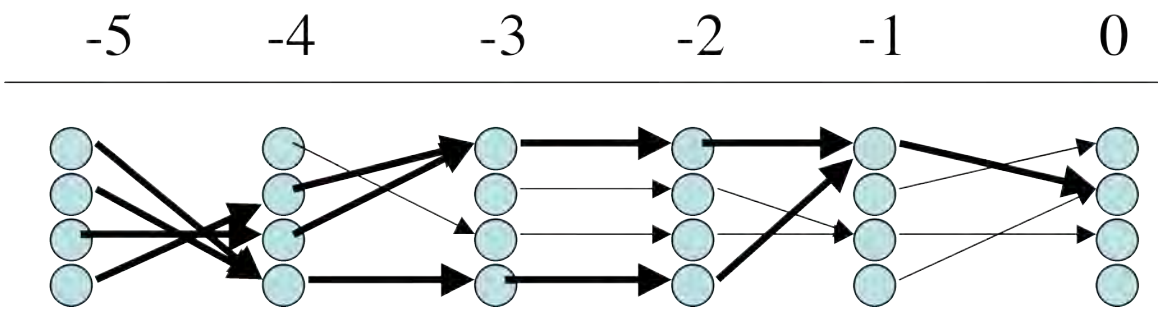
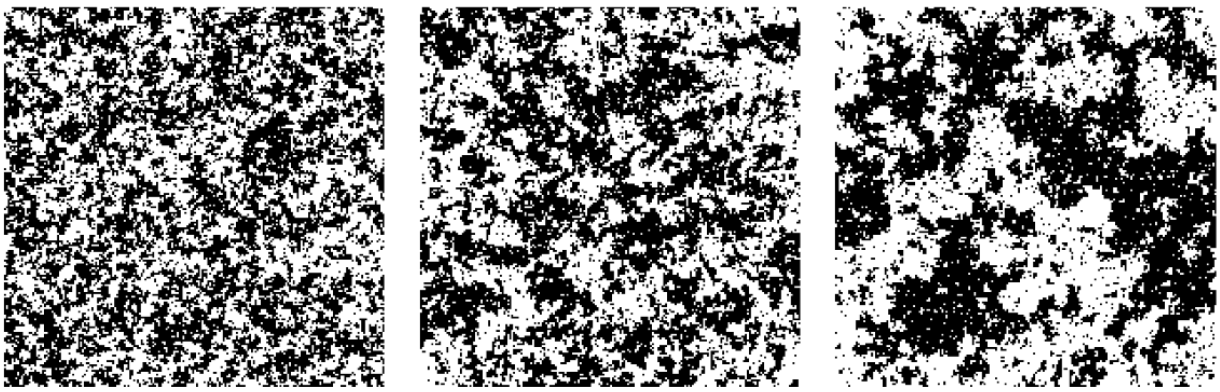


图 7.10: 从图 7.9耦合过去的马尔可夫链。在 5 次迭代模拟后发生聚结。

其中 ϕ 是更新规则，u 是随机源。可以通过跟踪 S 的最大和最小元素来验证合并。

7.6.2 应用：对 Ising 模型进行采样

我们现在从示例 7.5返回到 Ising 模型。由于其高维度，对 Ising 模型进行采样并非易事，因此我们使用 Gibbs 采样器根据晶格的每个特定旋转的条件分布来更新链，$P(s/\partial_s)$，其中 ∂_s 是 s 的相邻系统。直接从此分布中进行采样非常容易。



$\beta = 0.35, \tau = 105$　　$\beta = 0.40, \tau = 780$　　$\beta = 0.43, \tau = 5265$

图 7.11: 二维伊辛模型在不同温度下的样本。格子尺寸：200×200。

已经表明，如果采用用 Gibbs 采样器更新晶格中所有点的确定性（或半确定性）方案，则诱导马尔可夫链将收敛到晶格的联合分布，$P(I)$。在图 7.11中显示来自 Ising 模型的样本，在合并时具有不同的 β 值。每个图像下方显示 β 的值和合并时间 τ。在图 7.8中显示了 MC1（白链）和 MC2（黑链）的总磁化 $\sum_s X_s$，$\beta = 0.35$。对于不同的随机序列，聚结时的能量可能不同，如图 7.12所示。图 7.13显示每次 i 和时间 $i+t$ 给出的状态之间的相关性

$$R(t) = \frac{1}{N}\sum_{i=1}^{N} < X^{(i)}, X^{(i+t)} > .$$

如果将外部字段 X^{obs} 添加到模型中，则潜在的 $H(X)$ 变为

$$H(X) = -\sum_{<s,t>\in C} \alpha X_s X_t - \sum_{<s>\in I} X_s X_s^{obs}.$$

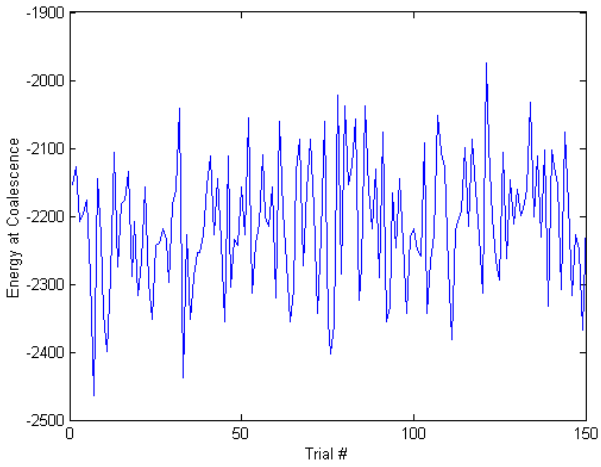


图 7.12: 150 次试验的聚结能量。$\beta = 0.40$，晶格尺寸：50×50。

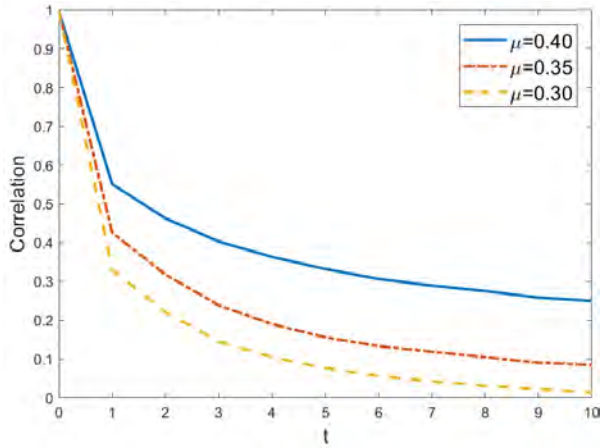


图 7.13: 不同温度下平衡态的相关性。

通过使用观察到的噪声图像作为外部场 X^{obs} 和采样图像 X 作为去噪图像，可以将具有外部场的 Ising 模型用于图像去噪。图 7.14显示了一个采样图像 X，它来自过去与左上角显示的外部字段的耦合，以及交互强度参数 β 的不同值。$\beta = 1$ 的上限和下限链的总磁化强度如图 7.15所示。

练习

问题 1. 考虑生活在太平洋岛屿的五个家庭的 Markov 核心，数字略有变化。

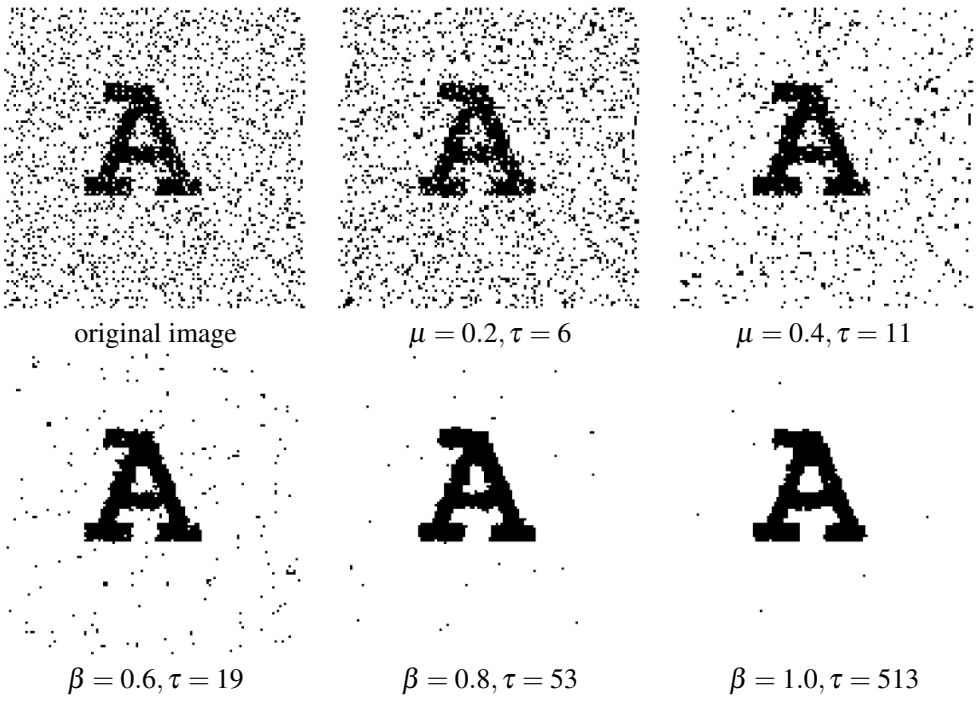


图 7.14: 具有外场的耦合马尔可夫链的降噪仿真。

$$K = \begin{pmatrix} 0.1, & 0.8, & 0.1, & 0.0, & 0.0 \\ 0.3 & 0.0, & 0.7, & 0.0, & 0.0 \\ 0.1, & 0.6, & 0.0, & 0.3, & 0.0 \\ 0.0, & 0.0, & 0.1, & 0.6, & 0.3 \\ 0.0, & 0.0, & 0.2, & 0.4, & 0.4 \end{pmatrix}$$

此转换矩阵定义有向图 $G =< V, E >$ 其中 $V = \{1,2,3,4,5\}$ 是状态集，$E = \{e = (x,y) : K(x,y) > 0\}$ 是一组有向边。

1. 绘制图形 G 并计算五个状态 $x \in \{1,2,3,4,5\}$ 的不变概率 $\pi(x)$; 并计算 λ_{slem} 的值。

   现在我们将尝试通过本章研究的以下两个概念验证 λ_{slem} 的界限 ---瓶颈和电导。

2. 哪个边缘 $e = (x,y)$ 是 G 的瓶颈？（您可以先根据图形连接进行猜测，然后根据其定义进行计算）；并计算图形 G 的瓶颈 κ。验证 Poincare 不等式：

$$\lambda_{\text{slem}} \leq 1 - \frac{1}{\kappa}.$$

3. 计算图形 G 的电导 h。验证 Cheeger 的不平等：

$$1 - 2h \leq \lambda_{\text{slem}} \leq 1 - \frac{h^2}{2}.$$

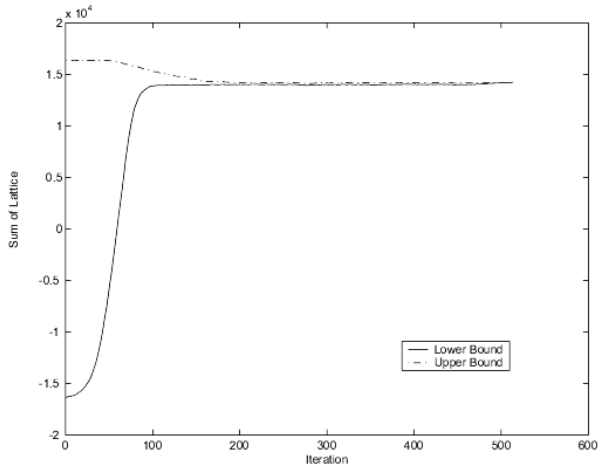


图 7.15: 上限和下限合并在 $\tau = 513, \beta = 1$。

4. 现在我们知道 π，我们可以设计一个“梦想”矩阵 K^*，它一步收敛。然后对于 K^*，$\lambda_{\text{slem}} = 0$。重新运行上面的代码来计算 K^* 的电导 h。验证 Cheeger 的不平等。

问题 2. 在卡片的洗牌中，我们提到了两个界限：7 和 11 作为预期的洗牌次数，使 52 卡随机。在证明边界之前，通过经验绘制收敛曲线通常是个好主意。

假设我们将 52 张卡片标记为 1,2，...，52 并以甲板（或状态）X_0 开始，从 1 到 52 排序。然后我们迭代地模拟以下的 riffle shuffling 过程从 X_{t-1} 到 X_t。

模拟 52 个独立的伯努利试验，概率 1/2 为 0 或 1. 因此，我们得到二元向量 0,1,1,0 .... 0。假设有 n 个零和 52 个 n。我们从牌组 X_{t-1} 中取出前 n 张牌，然后将它们顺序放入零位置，其余 52-n 牌顺序放入一个位置。

现在，让我们检查甲板 X_t 是否随机增加 t。您可以设计自己的方法来测试随机性。如果您没有更好的想法，则下面是默认方法。我们总是从排序的牌组 X_0 开始，并重复洗牌过程 K 次。因此，每次 t 我们记录 K 甲板的人口：

$$\{X_t^k : k = 1, 2, ..., K\}$$

对于每个卡位置 $i = 1, 2..., 52$，我们计算 K 套牌中 i 位置的 K 卡的直方图（边际分布）。用 $H_{t,i}$ 表示它并将其标准化为 1。这个直方图有 52 个箱子，所以我们可以选择 $K = 52 \times 10$。然后我们将这个 52-bin 直方图与 TV-norm 的均匀分布进行比较，并将它们在 52 个位置上进行平均，作为 t 时随机性的度量。

$$\text{err}(t) = \frac{1}{52} \sum_{i=1}^{52} ||H_{t,i} - \text{uniform}||_{\text{TV}}.$$

绘制 $\text{err}(t)$ 随着时间的推移 t 来验证收敛步骤。根据你的情节，我们真的需要多少次洗牌？

问题 3. 在有限状态空间 Ω 中，假设在步骤 t 中，马尔可夫链 MC 在 ν 概率之后具有状态 X。通过应用

马尔可夫内核 P 一次，其在 $t+1$ 中的状态是 Y，其遵循概率 $\mu = \nu \cdot P$。我们知道 P 观察具有不变概率 π 的详细平衡方程，即

$$\pi(x)P(x,y) = \pi(y)P(y,x), \quad \forall x,y \in \Omega.$$

证明 Kullback-Leibler 散度单调递减，

$$KL(\pi||\nu) - KL(\pi||\mu) = E[KL(P(Y,X)||Q(Y,X))] \geq 0.$$

其中 $P(Y,X)$ 是条件概率和

$$Q(Y,X) = \frac{P(X,Y)\nu(X)}{\mu(Y)}$$

是 P 的反向步骤。

（请注意，KL-发散是 $KL(p_1||p_2) = \sum_x p_1(x) \log \frac{p_1(x)}{p_2(x)}$.

提示：根据两个链计算两个连续状态 (X,Y) 的联合概率：固定链 $\pi(X,Y)$ 和当前链。然后计算两者之间的 KL-分歧。）

问题 4. 让 K 成为有限空间中的随机矩阵 Ω，让 μ 和 ν 成为 Ω 的两个初始概率，表明

$$||\mu K - \nu K||_{\text{TV}} \leq C(K)||\mu - \nu||_{\text{TV}},$$

其中 $C(K) \leq 1$ 是收缩系数，即转换内核中任意两行之间的最大 TV 范数，

$$C(K) = \max_{x,y} ||K(x,\cdot) - K(y,\cdot)||_{\text{TV}}.$$

TV 范数是 $||\mu - \nu||_{\text{TV}} = \frac{1}{2}\sum_{x\in\Omega} |\mu(x) - \nu(x)|$.

参考文献

[1] Persi Diaconis and Phil Hanlon. Eigen-analysis for some examples of the metropolis algorithm. *Contemporary Mathematics*, 138:99–117, 1992.

[2] Persi Diaconis and Daniel Stroock. Geometric bounds for eigenvalues of markov chains. *The Annals of Applied Probability*, pages 36–61, 1991.

[3] G. Winkler. *Image Analysis, Random Fields and Dynamic Monte Carlo Methods*. Springer, 2003.

第 7 章　数据驱动的马尔可夫链蒙特卡罗

‘‘数据就是新的石油。” - Clive Humby

简言


Zhuowen Tu

" 数据驱动的马尔可夫链蒙特卡罗（DDMCMC）提供了一种原则性方法，可以使用来自边缘检测和聚类等过程的低级信息，通过在解空间中进行已知跳跃来指导 MCMC 搜索，在后验概率模式的收敛中实现显着的加速。从边缘检测和强度聚类获得的数据驱动信息表示为加权样本 (粒子)，并使用 Metropolis-Hastings 方法用作 MCMC 跳跃的重要性提议概率。

在图像分割应用程序中，这些数据驱动的跳跃用于拆分合并操作，与区域增长，蛇/气球和区域竞争等类似扩散的操作一起，实现对解空间的有效遍历探索。"

8.1　分割问题和 DDMCMC 简介

由于两大挑战的持续存在，图像分割是计算机视觉领域的一个长期问题。


Song-Chun Zhu

第一个挑战是与一般图像中出现的大量视觉模式建模相关的基本复杂性。图像分割的目的是将图像解析为其组成部分。它们由各种随机过程组成，例如归属点，线，曲线，纹理，光照变化和可变形对象。因此，分割算法必须包含许多图像模型族，并且其性能受所选模型的精确度限制。

第二个挑战是图像感知的内在模糊性，特别是当没有特定的任务来引导注意力时。真实世界的图像基本上是模棱两可的，我们对图像的感知会随着时间而变化。此外，图像通常以多种尺度展示细节。因此，观看图像越多，人们看到的就越多。因此，认为分割算法仅输出一个结果一定是错误的。图像分割应被视为计算过程而非视觉任务。它应该动态且无限地输出多个不同的解，以便这些解最好地保持内在模糊性。

在上述两个观察的启发下，一种称为*数据驱动的马尔可夫链蒙特卡罗*（DDMCMC）的随机计算范式，被创建用于图像分割。该算法分五步进行。

1. 在贝叶斯/MDL 框架 [31, 33, 68] 中提出这个问题，用七个图像模型族竞争解释图像中的各种视觉形态，例如平坦区域、杂波区域、纹理、光滑阴影等。

2. 把解空间分解成许多不同维度的子空间的联合, 每个子空间都是图像分割和图像模型的多个子空间的乘积 (空间结构见图 8.10)。贝叶斯后验概率分布在这样的非均匀结构空间上。

3. 设计遍历马尔可夫链，探索解空间，并对后验概率进行采样。马尔可夫链由跳跃和扩散两类动力学组成。跳跃动力学模拟了可重复的拆分和合并以及模型切换。扩散动力学模拟了边界变形、区域增长、区域竞争 [68] 和模型适应。分裂和合并过程是可重复的，因此遍历性和可逆性使算法能够独立于初始分割条件获得几乎全局最优解。

4. 利用数据驱动技术来指导马尔可夫链搜索，以便与其他 MCMC 算法 [24, 26, 27] 相比能够获得极大加速。在文献中，有各种提高马尔可夫链速度的技术，如多分辨率方法 [7, 63] 、因果马尔科夫模型 [7, 46] 和聚类 [3, 21, 53, 63]。在 DDMCMC 范式中，使用了边缘检测 [10] 和跟踪以及数据聚类 [12, 13] 等数据驱动方法。这些算法的结果表示为加权样本 (或粒子)，在不同的子空间中编码非参数概率。这些概率分别近似于贝叶斯后验概率的边际概率，并用于设计驱动马尔科夫链的重要性提议概率。

5. 实现数学原理和 *K*-adventurers 算法，用于从马尔可夫链序列中和多尺度细节上选择和修剪一组重要且不同的解。该组解编码贝叶斯后验概率的近似值。计算多个解以使 Kullback-Leibler 从近似后验到真正后验的发散最小化，并且它们保留图像分割中的模糊性。

总之，DDMCMC 范式是关于有效地创建粒子 (通过自下而上的聚类/边缘检测)、组成粒子 (通过重要性建议)、和修剪粒子 (通过 K-adventurers 算法)，这些粒子表示了解空间中不同粒度级别的假设。从概念上讲，DDMCMC 范式也揭示了一些著名分割算法的作用。分割合并、区域增长、蛇 [51] 和气球/气泡 [51]、区域竞争 [68]、变分方法 [31] 等算法, PDEs[48] 可以被看作是经过微小修改的各种 MCMC 跳跃扩散动力学。边缘检测 [10] 和聚类 [13, 18] 等其他算法计算重要性提议概率。

8.2 DDMCMC 简介

我们可能用最简单的例子 – “Ψ-世界”来说明 DDMCMC 的概念。“Ψ-世界”仅由四种类型的对象组成: 背景像素、线段、弧线和希腊符号 Ψ，分别用 *BLCP* 标记。在点阵 Λ 上观察到的图像 $\mathbf{I}$ 是通过在背景上用加性高斯噪声叠加 n 个对象产生的。并且服从泊松分布，物体的大小和位置服从一些均匀分布。图 8.1显示了指定分布下的两个典型图像。

用向量来描述 Ψ-世界,

$$W = (n, \{(\ell_i, \theta_i); i = 0, 1, ..., n, \ \alpha).$$

$n \in \{0, 1, 2, .., |\Lambda|\}$ 是该对象以外的对象数量，$|\Lambda|$ 是图像中的像素数量。$\ell_i \in \{B, L, C, P\}$ 是标签，θ_i 是描述第 i 个对象的矢量值参数。我们只有一个背景对象 $\ell_0 = B$。

参数定义如下。

1. B 类型: θ_0 类型 B：对于像素的灰度级，0 仅为 0。

2. L 类型: θ_i 包括 $(\rho_i, \tau_i, s_i, e_i, \mu_i)$。$(\rho_i, \tau_i)$ 描绘一条直线，s_i, e_i 是起点和终点。μ_i 是线的强度等级。

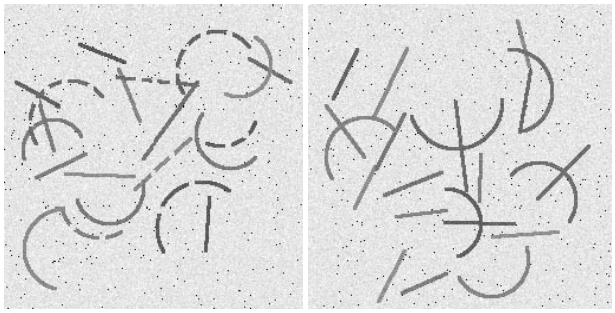


图 8.1: 随机生成的对象数量不同的图像示例。Zhu，Zhang 和 Tu 提供 [69].

3. C 类型: θ_i 包括 $(x_i, y_i, r_i, s_i, e_i, \mu_i)$，表示弧对象的中心，半径，起点，终点，终点和强度等级。

4. P 类型: θ_i 包括 $(x_i, y_i, r_i, \tau_i, \mu_i)$，表示半圆的中心和半径以及线段的角度和强度等级。根据定义，Ψ 中的弧必须是半圆。

W 中另一个重要变量是遮挡的 α 图。

$$\alpha : \Lambda \to \{0, 1, 2, ..., n\}, \quad \alpha \in \Omega_\alpha.$$

对于像素 $(x, y) \in \Lambda$，$\alpha(x, y)$ 索引最顶部的对象，该对象是该像素处唯一的可见对象。

我们用 $\varpi_g = [0, 255]$ 表示图像强度等级的空间，Ψ 世界的解空间是，

$$\Omega = \Omega_\alpha \times \varpi_g \times \cup_{n=0}^{|\Lambda|} \Omega_o^n,$$

其中 Ω_o^n 是具有 n 个对象（不包括背景）的子空间。

$$\Omega_o^n = \cup_{k+l+m=n} \Omega_{k,l,m}, \; k, l, m \geq 0,$$

其中 $\Omega_{k,l,m}$ 分别是正好具有 k 行，l 个弧和 $m\Psi$ 对象的子空间。

$$\Omega_{k,l,m} = \underbrace{\Omega_L \times \cdots \times \Omega_L}_{k} \times \underbrace{\Omega_C \times \cdots \times \Omega_C}_{l} \times \underbrace{\Omega_\Psi \times \cdots \times \Omega_\Psi}_{m}.$$

我们将 Ω_L，Ω_C 和 Ω_Ψ 称为对象空间。

这些对象空间进一步分解为五个*原子空间*，用小写的希腊符号表示。

1. ϖ_g: 像素强度空间 μ.
2. ϖ_c: 圆变量空间 x, y, r.
3. ϖ_l: 线变量空间 ρ, τ.
4. ϖ_e: 起点和终点空间 s, e.
5. ϖ_τ: Ψ 的方向空间。

因此，对象空间是原子空间的产物。

$$\Omega_l = \varpi_l \times \varpi_e \times \varpi_g, \quad \Omega_c = \varpi_c \times \varpi_e \times \varpi_g, \quad \Omega_\Psi = \varpi_c \times \varpi_\tau \times \varpi_g.$$

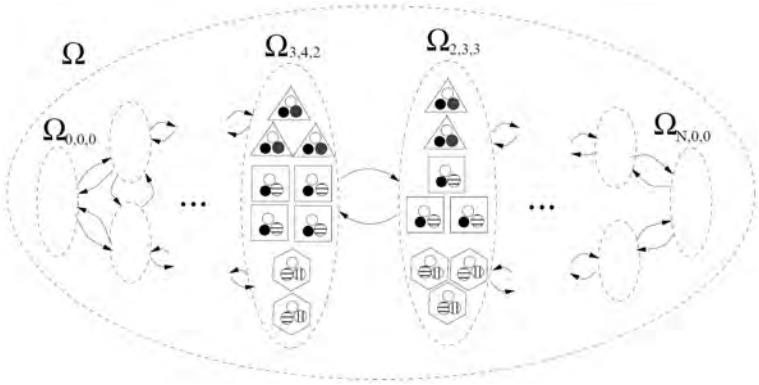


图 8.2: Ψ 世界的解空间 Ω。Zhu, Zhang and Tu 提供 [69]。

图 8.2说明了解空间 Ω 的结构。三角形，正方形和六边形分别代表三个对象空间 Ω_L，Ω_C 和 Ω_Ψ。各种阴影和尺寸的小圆圈代表五个原子空间。箭头表示下一节中讨论的子空间之间的可逆跳跃。

Ψ 世界中的对象识别被认为是贝叶斯推理问题，

$$W \sim p(W|\mathbf{I}) \propto p(\mathbf{I}|W)p(W),\ W \in \Omega.$$

$p(W)$ 是对象数量 n 上的泊松分布和对象参数 $\theta_i, i = 1,...,n$. Each W 上的一些均匀密度的乘积。具有所有参数 θ_i 和 α 图的每个 W 确定地指定清晰图像 $\mathbf{I}_o$，并且 $p(\mathbf{I}|W) = p(\mathbf{I}|\mathbf{I}_o)$ 仅仅是独立同分布高斯噪声的乘积。由于篇幅限制，我们选择不去详细地定义概率。

注意$p(W|\mathbf{I})$ 的概率质量分布在不同维度的多个子空间中，在下一节中，我们模拟了随机马尔可夫链动力学，它可以在这样的非均匀结构空间中传播并实现两个一般目标。

1. 计算 Ω 中的全局最优 W^*。

2. 为了鲁棒性，计算 M 个不同的解 (或解释)$S = \{W_1, W_2, ..., W_M\}$。

8.2.1 设计 MCMC--基本问题

我们分两步设计 MCMC。

首先，为了遍历和非周期，我们选择五种类型的 MCMC 动力学。

类型 I: 扩散过程。该过程改变参数 θ_i，例如，移动和延长线段等等。

类型 II: 死亡过程。此过程消除现有对象，并跳转到较低维度的子空间。

类型 III: 出生过程。此过程添加一个新对象，并跳转到更高维度的子空间。

类型 IV: 组合过程。此过程将两个现有对象组合为一个新对象，并跳转到另一个子空间。例如，将两条短线组成一条长线，或将一条线与一个圆组合成一个 Ψ。

类型 V: 分解过程。此过程将现有对象分解为两个。

例如，在图 8.2中，从 $\Omega_{3,4,2}$ 到 $\Omega_{2,3,3}$ 的移动将一个线对象和一个圆对象组合成一个 Ψ 对象。

通过投掷骰子决定，随机顺序应用五种类型的动力学。很容易证明马尔可夫链具有五种类型的运动是可逆的，可遍历的，和非周期性。

其次，我们讨论如何平衡马尔可夫链动力学。扩散过程可以通过随机 Langevin 方程来实现，该方

程是相对于 θ_i 加上布朗运动，使得 $p(W|\mathbf{I})$ 最大化的陡峭上升动力学。它也可以通过连续的 Gibbs 采样器实现。

由于篇幅限制，我们只讨论 II 型和 III 型之间的平衡。IV 型和 V 型的动力学可以以类似的方式完成。假设在某个时间步长 t 时，我们提议消除由 $\theta_i = (x_i, y_i, r_i, s_i, e_i, \mu_i)$ 指定的现有弧对象：

$$W = (n, \theta_i, w) \longrightarrow (n-1, w) = W'.$$

w 表示在此移动期间保持不变的对象。为了实现死亡动作，我们必须计算同一个对象立即出生的可能性 - 这是出生过程的逆向移动。请注意，这是一对在不同维度的两个子空间之间跳转的移动。我们使用 Metropolis-Hastings 方法。设 $G(W \to dW')$ 和 $G(W' \to dW)$ 分别为两次移动的提议概率，然后以概率接受死亡移，

$$A(W \to dW') = \min(1, \frac{G(W' \to dW)p(W'|\mathbf{I})dW'}{G(W \to dW')p(W|\mathbf{I})dW}). \tag{8.1}$$

转换概率 $P(W \to dW') = G(W \to dW')A(W \to dW')$ for $W \neq W'$. 后验概率的比率通常是支配后验影响的主要因素，以平衡提议概率的可能偏差。

死亡提议概率为

$$G(W \to dW') = q(\text{II})q_o(i)dw. \tag{8.2}$$

$q(\text{II}) \in (0,1)$ 是在时间 t 使用 II 型动力学的概率，$q_o(i)$ 是选择圆对象 θ_i 的概率。出生提议为

$$G(W' \to dW) = q(\text{III})q(\theta_i)d\theta_i dw. \tag{8.3}$$

首先选择概率为 $q(\text{III})$ 的 III 型，然后以概率 $q(\theta_i)d\theta_i$ 选择一个新的圆对象 θ_i。

由于 $dW = d\theta_i dw$，$dW' = dw$，因此等式 (8.1) 中分子和分母的维数是匹配的。设计 $q(\text{II})$ 和 $q(\text{III})$ 很容易，并且通常对速度并不重要。例如，可以在开始时更频繁地使用类型 II。这里的关键问题是计算 $q(\theta_i)$!

在统计文献中，跳跃动力学首先在 [26, 27] 中进行了研究，其中新维度中的变量由先验模型提出。在我们的例子中，选择 $q(\theta_i)$ 为盲目搜索的均匀分布。显然，这些提议最有可能被拒绝。这是 MCMC 效率低下的主要原因！直观地，$q(\theta_i)$s 应该能够预测新对象可能在对象空间中的位置。这是数据驱动（自下而上）技术的用武之地。

8.2.2 计算原子空间中的提议概率--原子粒子

图 8.3显示了 Ψ 世界中对象的层次结构。终端（圆圈）节点表示特征元素：条形，终点和交叉，箭头表示构图关系。我们使用三种类型的特征检测器：3 个交叉检测器，6 个条形检测器和 12 个不同方向的终点检测器。

使用条形检测图，我们计算线条和圆形的霍夫变换。使用均值偏移算法 [13] 计算局部最大值。我们用 $(\rho_l^{(i)}, \tau_l^{(i)}), i = 1, 2, ..., n_l$ 表示这些局部最大值。

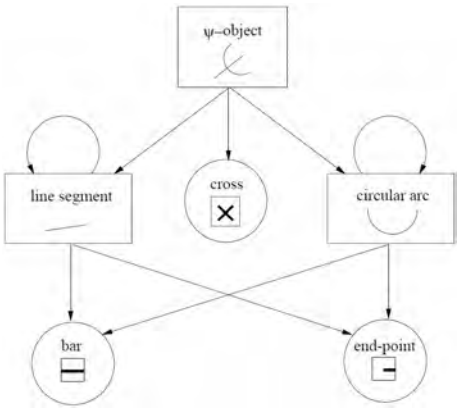


图 8.3: Ψ 世界中的层次结构。Zhu，Zhang 和 Tu 提供 [69].

因此，我们计算原子空间 ϖ_l 上的经验密度，表示为一组加权样本。

$$q_l(\rho,\tau)=\sum_{i=1}^{n_l} w_l^{(i)}\delta(\rho-\rho_l^{(i)},\tau-\tau_l^{(i)}),\sum_{i=1}^{n_l} w_l^{(i)}=1.$$

$\delta(\rho-\rho_l^{(i)},\tau-\tau_l^{(i)})$a 是以 $(\rho_l^{(i)},\tau_l^{(i)})$ 为中心的窗函数。我们将 $q_l(\rho,\tau)$ 称为原子空间 ϖ_l 中的重要性提议概率，将 $\{(\rho_l^{(i)},\tau_l^{(i)}),i=1,2,...,n_l\}$ 称为原子粒子。

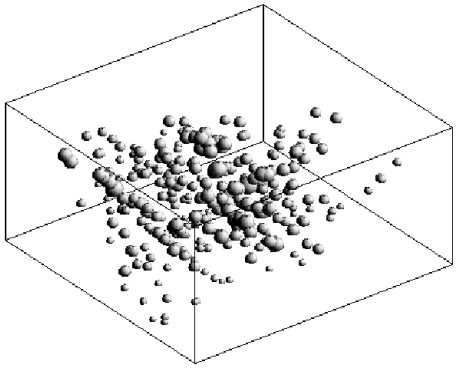


图 8.4: 293 加权样本 (x,y,r) 用于原子空间 ϖ_c 中的圆粒子。Zhu, Zhang 和 Tu 提供 [69]。

类似地，图 8.4显示了空间 ϖ_c 中的原子粒子。它们是圆的霍夫变换结果的局部最大值。球体的大小代表权重 $w_c^{(i)}$。

所以我们在 ϖ_c 上有一个原子提议概率，

$$q_c(x,y,r)=\sum_{i=1}^{n_c} w_c^{(i)}\delta(x-x_c^{(i)},y-y_c^{(i)},r-r_c^{(i)}),\ \sum_{i=1}^{n_c} w_l^{(i)}=1.$$

以类似的方式，可以计算其他原子空间中的提议概率。1）在 ϖ_g 中，我们计算强度直方图 $q_g(\mu)$。2）在 ϖ_e 中，我们计算了终点图。3）在 ϖ_τ 中，我们可以简单地将 $q_l()$ 投影到 τ 轴上。对于 鲁棒性和可逆性，原子空间中的原子提议概率是连续且非零的。

8.2.3 计算对象空间中的提议概率--对象粒子

因为对象空间是原子空间的乘积，所以三个对象空间 $\Omega_l\Omega_c$ 和 Ω_Ψ 中的提议概率可以通过五个原子空间中的概率来计算。我们讨论三种方法。

方法 I: 条件绑定。通过顺序使用原子特征来组成对象粒子。例如，对于线对象 $\theta=(\rho,\tau,s,e,\mu)$，我们计算

$$q(\theta)=q_l(\rho,\tau)q_e(s|\rho,\tau)q_e(e|\rho,\tau,s)q_g(\mu|\rho,\tau,s,e). \tag{8.4}$$

从 $q(\theta)$ 采样一组线段 $\theta_l^{(i)}=(\rho_l^{(i)},\tau_l^{(i)},s_l^{(j)},e_l^{(k)})$,

$$\{\theta_l^{(i)}\ :\ i=1,2,...,n_L\}.$$

称它们为 Ω_l 中的对象粒子。以类似的方式，我们可以在 Ω_c 中生成对象粒子。

这些对象粒子在思想上与工程方法中的假设相似。但是，有一个至关重要的区别。每个对象粒子代表对象空间中的窗口域而不仅仅是一个点。对象粒子的窗口的并集覆盖整个对象空间。为了使马尔可夫链可逆，每次我们通过抽样提议概率提出一个新的对象，而不仅仅是从粒子集中选择。

对象粒子也应该通过递归组合对象粒子来生成，如图 8.3中的箭头所示。

方法 II: 离线组合。如果后者是相容的，则可以通过合并两个其他粒子来组成一个粒子。该组合离线发生，即在我们开始运行 MCMC 之前。但由于可能的组合数量呈指数级，这会非常昂贵。

方法 III: 在线组合。当两个兼容对象在当前 W 中出现（或“活着”）时，这与方法 II 的不同之处在于 MCMC 计算期间的绑定对象。

原子粒子在自下而上的过程中计算一次，而对象粒子在 MCMC 过程中动态组合。图 8.5显示了三行对象粒子。一个用于出生候选者，一个用于死亡候选者和分解候选者，一个用于在 W 中存活的兼容组合对。我们还必须通过 α 图捕获的遮挡效应来考虑对象之间的相互作用。例如，假设当前图像中的长线段被两个短线对象 $\theta_l^{(i)}$ 和 $\theta_l^{(j)}$ 覆盖。然后，由于遮挡效应，在拟合图像时添加长线对象 $\theta_l^{(k)}$ 将获得非常小的增益。因此，必须在线更新这些对象颗粒的权重。

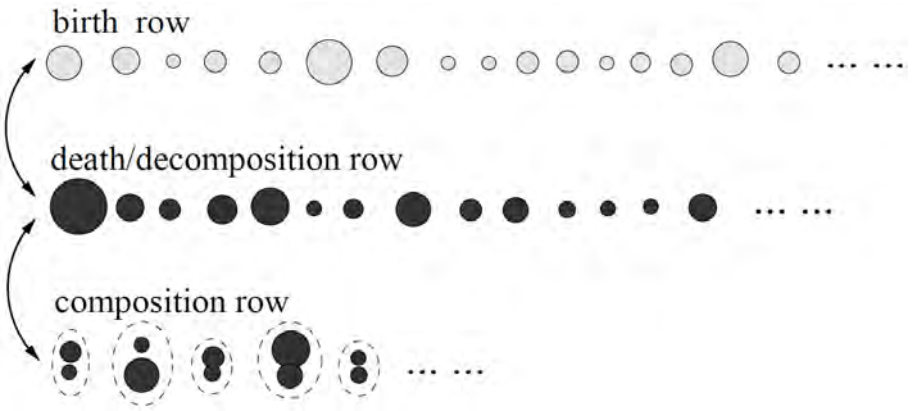


图 8.5: 对象粒子分为三行，驱动 MCMC 动态学。黑暗的粒子还存在。Zhu，Zhang 和 Tu 提供 [69]。

8.2.4 计算多个不同的解--场景粒子

为了构建用于对象识别的强大视觉系统，传统的 MAP（最大后验）估计器

$$W^* = \arg\max_{W\in\Omega} p(W|\mathbf{I})$$

是不够的. 相反，我们应该采样 $p(W|\mathbf{I})$ 并计算一组具有代表性的解。然而，当 Ω 复杂且高维时，简单地采样 $p(W|\mathbf{I})$ 仅产生全部来自单一模式并且彼此具有微小差异的解。因此，必须导出数学标准以保留重要的，独特的局部模式。

设 $S=\{W_1,W_2,...,W_M\}$ 为权重是 $\omega_i \propto p(W|\mathbf{I}),\forall i$ 的 M 个解。M 个加权样本以非参数形式编码 $p(W|\mathbf{I})$。

$$\hat{p}(W|\mathbf{I}) = \sum_{i=1}^{M} \omega_i G(W-W_i), \quad \sum_{i=1}^{M} \omega_i = 1.$$

通过一些高斯窗函数 G. 在本节中，我们使用以下标准来扩展传统的 MAP，

$$S^* = \{W_1,W_2,...,W_M\} = \arg\min_{|S|=M} D(p||\hat{p}). \tag{8.5}$$

我们把 $\{W_1,W_2,...,W_M\}$ 称为*场景粒子*. 选择它们以最小化 Kullback-Leibler 散度 $D(p||\hat{p})$，使得 $\hat{p}$ “最佳”保留 p--在复杂度 M 的约束下的真实后验分布。

实际上，p 表示为在 MCMC 过程中记录的大量 $N>>M$ 个粒子的高斯模型如 $\hat{p}$ 的混合。因此，我们在 MCMC 计算期间选择 M 个不同的解，以使 $D(p||\hat{p})$ 最小化。

通过 $D(p||\hat{p})$ 的数学推导，我们得出三个引导场景粒子选择的较好原则。

1. 其中一个粒子必须是全局最优 W^*。缺少 W^* 会导致 $D(p||\hat{p})$ 大幅增加。

2. 场景粒子应该最小化能量的总和（或最大化概率的乘积）。

3. 场景粒子还应最大化彼此的距离之和。

最后两个是冲突的，因此粒子必须“占据”解空间 Ω 中的不同模式。

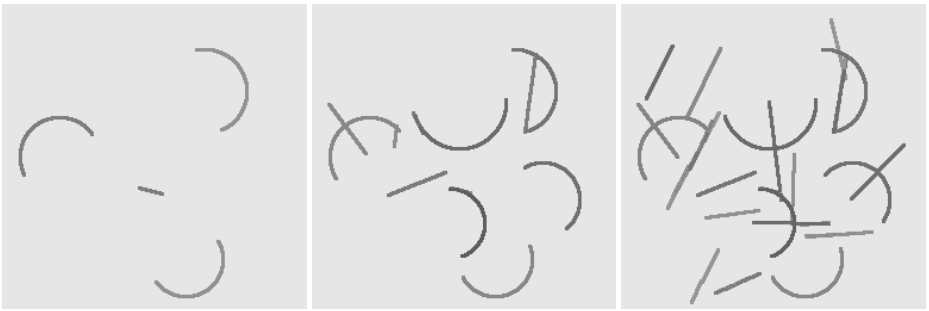


图 8.6: Solution W visited by MCMC at three time steps. Courtesy of Zhu, Zhang and Tu [69].

8.2.5 Ψ 世界实验

我们使用随机生成的图像集合进行实验，其中两个如图 8.1所示。图 8.6显示了在第 t 步，由 W 决定的 $\mathbf{I}_o$ 的 MCMC 算法中的步骤。

我们主要关注，通过比较四个马尔可夫链来研究 MCMC 效率如何提高。

MCMC I：马尔可夫链使用统一的提议概率，如文献 [26, 27]。

MCMC II：马尔可夫链使用原子提议概率与霍夫变换而没有终点检测。

MCMC III：马尔可夫链使用霍夫变换和终点检测，并从粒子集中随机采样新对象。

MCMC IV：马尔可夫链使用霍夫变换和终点检测。但它与 MCMC III 的不同之处在于它可以在线评估对象粒子的权重。

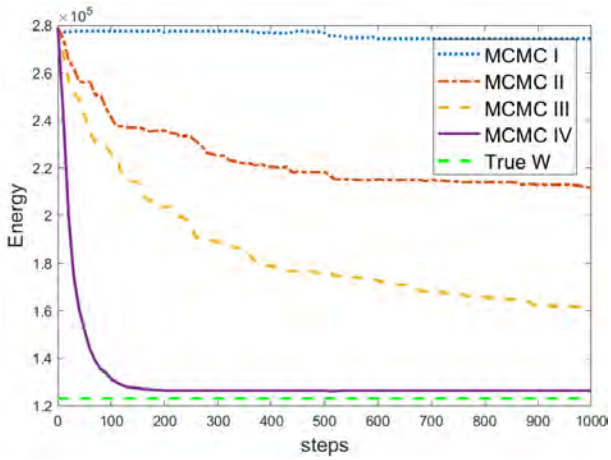


图 8.7: 解 W 的能量与 MCMC 迭代的关系。Zhu，Zhang 和 Tu 提供 [69]。

图 8.7绘制了步骤 t 中 MCMC 状态的能级，即 $-\log p(W|\mathbf{I})$。对于 8 张图像中的每一个，四个马尔可夫链中的每一个运行 10 次，因此能量曲线平均超过 80 次运行。虚线，点划线，虚线和实线曲线分别用于 MCMC I，II，III 和 IV。底部的水平线是真实 W^*s 的平均能量。很明显，使用更多的数据驱动方法，马尔可夫链可以更快的速度接近解。MCMC IV 在 200 步时达到具有 2.3% 相对误差的解。该误差主要是通过扩散进行微调的问题。相比之下，MCMC III 需要大约 5,000 步。MCMC I 在经过数百万步之后也不会下降。我们还通过测量所获得的解 S 以 Ω 分布的宽度来比较“混合”速率。

8.3 问题公式化和图像模型

在本节中，问题是在贝叶斯框架中制定的，并讨论了先验和可能性模型。

8.3.1 用于分割的贝叶斯公式

设 $\Lambda = \{(i,j) : 1 \le i \le L, 1 \le j \le H\}$ 为一个图像点阵，$\mathbf{I}_\Lambda$ 是定义在 Λ 上的图像。对于任何点 $v \in \Lambda$，$\mathbf{I}_v \in \{0,...,G\}$ 是灰度图像的像素强度，或者对于彩色图像，$\mathbf{I}_v = (L_v, U_v, V_v)$。图像分割问题是指将点阵划分为未知数量 K 个不相交区域。

$$\Lambda = \cup_{i=1}^{K} R_i, \quad R_i \cap R_j = \emptyset, \quad \forall i \neq j. \tag{8.6}$$

不需要连接每个区域 $R \subset \Lambda$。设 $\Gamma_i = \partial R_i$ 表示 R_i 的边界。微微有点复杂，文献中可互换地使用两种符号。一个把区域 $R \in \Lambda$ 认为是一个离散的标签图，另一个把区域边界 $\Gamma(s) = \partial R$ 认为是由 s 参数化的连续轮廓。连续表示对于扩散是方便的，而标签映射表示更适合于维持拓扑。水平集方法 [47, 48] 在两者之间提供了良好的折衷。

假设 $\mathbf{I}_R$ 是来自概率模型 $p(\mathbf{I}_R;\Theta)$ 的实现，每个图像区域 $\mathbf{I}_R$ 被认为是相干的。Θ 代表一个随机过程，其类型由 ℓ 索引。因此，分割由隐变量 W 的一个向量表示，其描述了用于生成图像 $\mathbf{I}$ 的世界状态，由其下式给出

$$W = (K, \{(R_i, \ell_i, \Theta_i);\ i = 1,2,...,K\}).$$

在贝叶斯框架中，我们在解空间 Ω 上从 $\mathbf{I}$ 推断 W，如下

$$W \sim p(W|\mathbf{I}) \propto p(\mathbf{I}|W)p(W),\ W \in \Omega.$$

正如我们之前提到的，分割中的第一个挑战是获得逼真的图像模型。在下面的小节中，我们将简要讨论先验和可能性模型。

8.3.2 先验概率

先验概率 $p(W)$ 是以下四个概率的乘积。

1. 区域数量 $p(K) \propto e^{-\lambda_0 K}$ 的指数模型。
2. 区域边界 $p(\Gamma) \propto e^{-\mu \oint_\Gamma ds}$ 的一般平滑 Gibbs 先验。
3. 促使大区域形成 $p(A) \propto e^{-\gamma A \cdot 9}$ 的模型，其中 γ 是控制分割比例的比例因子。
4. 一个图像模型参数 Θ 的先验值，它惩罚模型的复杂度 $p(\Theta|\ell) \propto e^{-\nu|\Theta|}$。

总之，我们有以下先验模型

$$p(W) \propto p(K)\prod_{i=1}^{K} p(R_i)p(\ell_i)p(\Theta_i|\ell_i) \propto \exp\{-\lambda_0 K - \sum_{i=1}^{K}[\mu \oint_{\partial R_i} ds + \gamma|R_i|^c + \nu|\Theta_i|]\}. \tag{8.7}$$

8.3.3 灰度图像的可能性

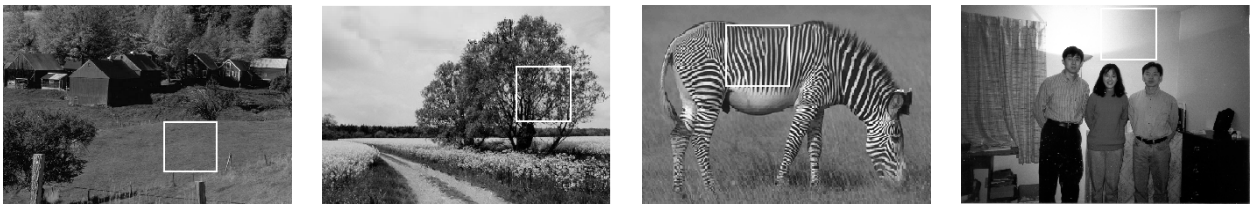
假设不同区域中的视觉模式是由 $(\Theta_i, \ell_i), i = 1,2,...,K$ 指定的独立随机过程。因此可能性是，[1]

$$p(\mathbf{I}|W) = \prod_{i=1}^{K} p(\mathbf{I}_{R_i};\Theta_i,\ell_i).$$

模型的选择需要平衡模型的充分性和计算效率。在实际图像中最常出现四种类型的区域。图 8.8显示了以下各项的示例：a）没有明显图像结构的平坦区域，b）杂乱区域，c）具有均匀纹理的区域，以及 d）具有全局平滑阴影变化的不均匀区域。

我们对四种类型的区域采用以下四个模型族。该算法可以通过马尔可夫链跳转，在它们之间切换。四个族由 $\ell \in \{g_1, g_2, g_3, g_4\}$ 索引，并分别由 ϖ_{g_1}，ϖ_{g_2}，ϖ_{g_3}，和 ϖ_{g_4} 表示。设 $G(0;\sigma^2)$ 为以 0 为中心的高斯密度，方差为 σ^2。

[1]符号有点微微复杂，$\Theta\ell$ 可以被视为 W 中的参数或隐变量。为简单起见，我们在两种情况下都使用 $p(\mathbf{I};\Theta,\ell)$。



(a) 均匀 (b) 杂波 (c) 纹理 (d) 阴影

图 8.8: 在实际图像中，窗口中的四种区域是典型的. Tu 和 Zhu 提供 [58].

1. 高斯模型 $\ell = g_1$: ϖ_{g_1}. 这假设区域 R 中的像素强度受独立同分布（iid）的高斯分布，

$$p(\mathbf{I}_R;\Theta,g_1) = \prod_{v\in R} G(\mathbf{I}_v - \mu;\sigma^2), \quad \Theta = (\mu,\sigma) \in \varpi_{g_1}. \tag{8.8}$$

2. 直方图模型 $\ell = g_2$: ϖ_{g_2}. 这是非参数强度直方图 $h()$。实际上，$h()$ 被离散化为由向量 $(h_0,h_1,...,h_G)$. n_j 表示的阶梯函数。n_j 是强度等级为 j 的 R 中的像素数量。

$$p(\mathbf{I}_R;\Theta,g_2) = \prod_{v\in R} h(\mathbf{I}_v) = \prod_{j=0}^{G} h_j^{n_j}, \quad \Theta = (h_0,h_1,...,h_G) \in \varpi_{g_2}. \tag{8.9}$$

3. 伪模型 $\ell = g_3$: ϖ_{g_3}. 这是纹理 FRAME 模型 [67]，其中像素交互由一组 Gabor 滤波器捕获。为了便于计算，我们选择一组 8 个滤波器，并以伪似然形式 [66] 表示该模型。该模型由长矢量 $\Theta = (\beta_1,\beta_2,...,\beta_m) \in \varpi_{g_3}$ 指定，其中 m 是 8 个 Gabor 滤波图像直方图中的区间总数。令 ∂v 表示 $v \in R$ 的马尔可夫邻域，$\mathbf{h}(\mathbf{I}_v|\mathbf{I}_{\partial v})$ 表示像素 v 附近，滤波器响应的 8 个局部直方图的矢量。每个滤波器直方图都会对滤波器窗口覆盖 v 的滤波器响应进行计数。因此我们有

$$p(\mathbf{I}_R;\Theta,g_3) = \prod_{v\in R} p(\mathbf{I}_v|\mathbf{I}_{\partial v};\Theta) = \prod_{v\in R} \frac{1}{Z_v} \exp\{- < \Theta, \mathbf{h}(\mathbf{I}_v|\mathbf{I}_{\partial v}) >\}, \tag{8.10}$$

可以精确地计算归一化常数，并且可以很容易地从图像中估计 Θ。我们参考 [66] 讨论该模型的计算及其变化，例如小块可能性。

4. 表面模型 g_4: ϖ_{g_4}. 前三个模型是同质的，在表征具有阴影效应的区域时失败，例如天空，湖泊，墙壁，透视纹理等。在文献中，这样的平滑区域通常由低阶马尔可夫随机场建模，其也不会再次对空间上的非均匀图案建模，并经常导致过度分割。可以替代地在 Λ 上采用具有十六个等间隔控制点的 2D 贝兹曲线模型（即，我们解决这个了问题）。这是一种生成模型。设 B（x，y）为贝塞尔曲面，对任意 $v = (x,y) \in \Lambda$,

$$B(x,y) = U_{(x)}^T \times M \times U_{(y)}, \tag{8.11}$$

其中 $U_{(x)} = ((1-x)^3, 3x(1-x)^2, 3x^2(1-x), x^3))^T$，$M = (m_{11},m_{12},m_{13},m_{14};...;m_{41},...,m_{44})$. 因此，区域 R 的图像模型是，

$$p(\mathbf{I}_R;\Theta,g_4) = \prod_{v\in R} G(\mathbf{I}_v - B_v;\sigma^2), \quad \Theta = (M,\sigma) \in \varpi_{g_4}. \tag{8.12}$$

这四种模型相互竞争解释灰度强度区域。无论哪个更适合该地区，都有更高的可能性。灰度模型空

间由 ϖ_Θ^g 表示，并由下式给出

$$\Theta \in \varpi_\Theta^g = \varpi_{g_1} \cup \varpi_{g_2} \cup \varpi_{g_3} \cup \varpi_{g_4}.$$

8.3.4 模型校准

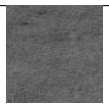
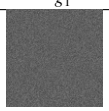
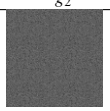
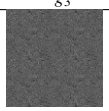
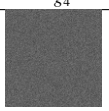
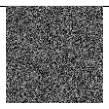
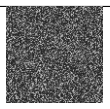
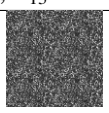
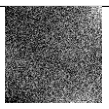
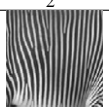
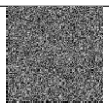
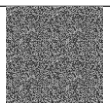
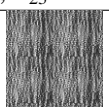
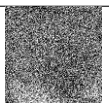
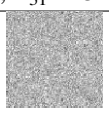
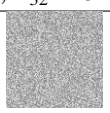
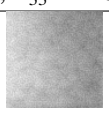
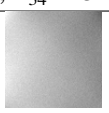
observed	ϖ_{g_1}	ϖ_{g_2}	ϖ_{g_3}	ϖ_{g_4}
				
$\mathbf{I}_1^{\text{obs}}$	$\mathbf{I}_{11}^{\text{syn}}, L_{11} = 1.957$	$\mathbf{I}_{12}^{\text{syn}}, L_{12} = 1.929$	$\mathbf{I}_{13}^{\text{syn}}, L_{13} = 1.680$	$\mathbf{I}_{14}^{\text{syn}}, L_{14} = 1.765$
				
$\mathbf{I}_2^{\text{obs}}$	$\mathbf{I}_{21}^{\text{syn}}, L_{21} = 3.503$	$\mathbf{I}_{22}^{\text{syn}}, L_{22} = 3.094$	$\mathbf{I}_{23}^{\text{syn}}, L_{23} = 2.749$	$\mathbf{I}_{24}^{\text{syn}}, L_{24} = 3.422$
				
$\mathbf{I}_3^{\text{obs}}$	$\mathbf{I}_{31}^{\text{syn}}, L_{31} = 3.852$	$\mathbf{I}_{32}^{\text{syn}}, L_{32} = 3.627$	$\mathbf{I}_{33}^{\text{syn}}, L_{33} = 2.514$	$\mathbf{I}_{34}^{\text{syn}}, L_{34} = 3.658$
				
$\mathbf{I}_4^{\text{obs}}$	$\mathbf{I}_{41}^{\text{syn}}, L_{41} = 3.121$	$\mathbf{I}_{42}^{\text{syn}}, L_{42} = 3.050$	$\mathbf{I}_{43}^{\text{syn}}, L_{43} = 1.259$	$\mathbf{I}_{44}^{\text{syn}}, L_{44} = 0.944$

图 8.9: 四类模型的比较研究。第一列包含从图 8.8中所示的四个实际图像中裁剪的原始图像区域。第 2-5 列是在 MLE 拟合之后分别从四个族采样的合成图像 $\mathbf{I}_{ij}^{\text{syn}} \sim p(\mathbf{I}_R; \Theta_{ij}^*)$。每个合成图像下方的数字表示使用每个模型族的每个像素编码位。Tu 和 Zhu 提供 [58]。

四个图像模型应该被校准，主要有两个原因。首先，为了计算效率，优先选取具有较少参数的简单模型。然而，实际上惩罚参数的数量是不够的。当一个区域大小超过 ~ 100 像素时，数据项已满足了之前要求并且需要更复杂的模型。其次，族 ϖ_{g_3} 中的伪似然模型不是真实可能性，因为它们依赖于相当大的邻域，因此不能与其他三种类型的模型直接比较。

为了校准似然概率，可以使用实证研究。我们从自然图像中收集了一组典型区域，并手动将它们分为四类。例如，图 8.9显示了第一列中的四个典型图像，这些图像是从图 8.8中的图像裁剪出来的。四个图像由点阵 Λ_o 上的 $\mathbf{I}_i^{\text{obs}}, i = 1,2,3,4$ 表示。对于每个图像 $\mathbf{I}_i^{\text{obs}}$，我们根据族 ϖ_{g_j} 内的最佳模型计算其每个像素编码长度（减去对数似然），该最佳模型通过下式给出的 $j = 1,2,3,4$ 的极大似然估计来计算。

$$L_{ij} = \min_{\varpi_{g_j} \ni \Theta} -\frac{\log p(\mathbf{I}_i^{\text{obs}}; \Theta, g_j)}{|\Lambda_o|}, \quad \text{for } 1 \le i, j \le 4. \tag{8.13}$$

如果 $\Theta_{ij}^* \in \varpi_{gj}$ 是每个族中的最佳拟合，那么我们可以从每个拟合模型中绘制一个典型样本（合成），

$$\mathbf{I}_{ij}^{\text{syn}} \sim p(\mathbf{I}; \Theta_{ij}^*, g_j), \quad \text{for } 1 \le i, j \le 4.$$

$\mathbf{I}_i^{\text{obs}}$，$\mathbf{I}_{ij}^{\text{syn}}$，和 L_{ij} 如图 8.9所示，$1 \le ij \le 4$。

图 8.9中的结果表明，样条模型具有最短的着色区域编码长度，而纹理模型最适合其他三个区域。

我们可以通过每个像素 v 的常数因子 e^{-c_j} 来修正这些模型，

$$\hat{p}(\mathbf{I}_v;\Theta,g_j)=p(\mathbf{I}_v;\Theta,g_j)e^{-c_j},\quad \text{for } j=1,2,3,4.$$

选择 c_j 使得当 $i=j$ 时修正编码长度 $\hat{L}_{ij}$ 达到最小值。有效地，均匀区域，杂波区域，纹理区域和阴影区域最好分别通过 ϖ_1，ϖ_2，ϖ_3，和 ϖ_4 中的模型拟合。

8.3.5 彩色图像模型

实际上，我们同时处理灰度和彩色图像。对于彩色图像，我们采用 (L^*,u^*,v^*) 颜色空间和由 $\ell\in\{c_1,c_2,c_3\}$ 索引的三个模型族。设 $G(0;\Sigma)$ 表示 3D 高斯密度。

1. 高斯模型 c_1: ϖ_{c_1}. 这是一个独立同分布。(L^*,u^*,v^*) 空间中的高斯模型.

$$p(\mathbf{I}_R;\Theta,c_1)=\prod_{v\in R}G(\mathbf{I}_v-\mu;\Sigma),\quad \Theta=(\mu,\Sigma)\in\varpi_{c_1}. \tag{8.14}$$

2. 混合模型 c_2: ϖ_{c_2}. 这是两个高斯的混合，用于建模纹理颜色区域，

$$p(\mathbf{I}_R;\Theta,c_2)=\prod_{v\in R}[\alpha_1 G(\mathbf{I}_v-\mu_1;\Sigma_1)+\alpha_2 G(\mathbf{I}_v-\mu_2;\Sigma_2)].$$

因此 $\Theta=(\alpha_1,\mu_1,\Sigma_1,\alpha_2,\mu_2,\Sigma_2)\in\varpi_{c_2}$ 是参数。

3. 贝塞尔模型 c_3: ϖ_{c_3}. 我们分别使用三个贝塞尔样条曲面（见公式 (8.11)）来表示 L^*，u^* 和 v^*，以表征颜色逐渐变化的区域，如天空，墙壁等。设 $\mathbf{B}(x,y)$ 为任何 $v=(x,y)\in\Lambda$ 的 (L^*,u^*,v^*) 空间中的颜色值，

$$\mathbf{B}(x,y)=(U_{(x)}^T\times M_L\times U_{(y)},U_{(x)}^T\times M_u\times U_{(y)},U_{(x)}^T\times M_v\times U_{(y)})^T.$$

因此模型为

$$p(\mathbf{I}_R;\Theta,c_3)=\prod_{v\in R}G(\mathbf{I}_v-\mathbf{B}_v;\Sigma),$$

其中 $\Theta=(M_L,M_u,M_v,\Sigma)$ 是参数.

这三种模型竞争解释颜色区域。无论哪个更适合该地区，都有更高的可能性。我们用 ϖ_Θ^c 表示颜色模型空间，使得

$$\varpi_\Theta^c=\varpi_{c_1}\cup\varpi_{c_2}\cup\varpi_{c_3}.$$

8.4 解空间分析

在我们了解算法的细节之前，我们需要研究解空间 Ω 的结构，其中后验概率 $p(W|\mathbf{I})$ 是分布的。我们从点阵 Λ 的所有可能分区的分区空间开始。当点阵 Λ 被分割成 k 个不相交的区域时，我们称之为 k 分区，用 π_k 表示，

$$\pi_k=(R_1,R_2,...,R_k),\quad \cup_{i=1}^k R_i=\Lambda,\quad R_i\cap R_j=\emptyset,\ \forall i\neq j. \tag{8.15}$$

如果连接每个区域中的所有像素，则 π_k 是连通分量分区 [63]。由 ϖ_{π_k} 表示的所有 k 分区的集合，是所有可能的 k 种着色集合的商空间除以下式给出的标签的置换组 $\mathcal{PG}$。

$$\varpi_{\pi_k} = \{(R_1, R_2, ..., R_k) = \pi_k; \quad |R_i| > 0, \forall i = 1, 2, ..., k\}/\mathcal{PG}. \tag{8.16}$$

因此，我们具有一般分区空间 ϖ_π，区域数量 $1 \leq k \leq |\Lambda|$，

$$\varpi_\pi = \cup_{k=1}^{|\Lambda|} \varpi_{\pi_k}.$$

W 的解空间是子空间 Ω_k 的并集，每个 Ω_k 是图像模型的一个 k 分区空间 ϖ_{π_k} 和 k 空间的乘积

$$\Omega = \cup_{k=1}^{|\Lambda|} \Omega_k = \cup_{k=1}^{|\Lambda|} [\, \varpi_{\pi_k} \times \underbrace{\varpi_\Theta \times \cdots \times \varpi_\Theta}_{k} \,], \tag{8.17}$$

其中对于灰度图像，$\varpi_\Theta = \cup_{i=1}^{4} \varpi_{gi}$，对于彩色图像，$\varpi_\Theta = \cup_{i=1}^{3} \varpi_{ci}$。

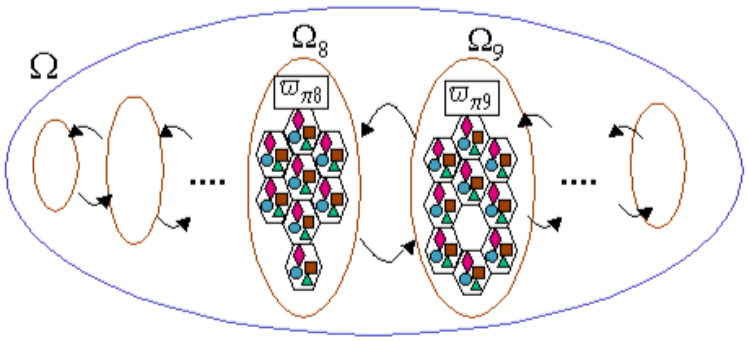


图 8.10: 解空间分析。箭头代表马尔可夫链跳跃。两个子空间 Ω_8 和 Ω_9 之间的可逆跳跃实现了区域的分裂和合并。Tu 和 Zhu 提供 [58].

图 8.10说明了解空间的结构。四个图像族 $\varpi_\ell, \ell = g_1, g_2, g_3, g_4$ 分别由三角形，正方形，菱形和圆形表示。$\varpi_\Theta = \varpi_\Theta^g$ 由包含四种形状的六边形表示。分区空间 ϖ_{π_k} 由矩形表示。每个子空间 Ω_k 由矩形和 k 个六边形组成，并且每个点 $W \in \Omega_k$ 表示 k 分区加上 k 个区域的 k 个图像模型。我们称 Ω_k 为场景空间。ϖ_{π_k} 和 $\varpi_\ell, \ell = g_1, g_2, g_3, g_4$（或 $\ell = c_1, c_2, c_3$）是构造 Ω 的基本分量，因此被称为原子空间。有时我们将 $\varpi_\pi \varpi_\pi$ 称为分区空间，将 $\varpi_\ell, \ell = g_1, g_2, g_3, g_4, c_1, c_2, c_3$ 称为线索空间。

8.5 利用遍历马尔可夫链探索解空间

图 8.10中的解空间是视觉问题中典型的。后验概率 $p(W|\mathbf{I})$ 不仅具有大量的局部最大值，而且分布在不同维度的子空间上。为了在这样的空间中搜索全局最优解，我们采用马尔可夫链蒙特卡罗（MCMC）技术。

8.5.1 五类马尔可夫链动力学

我们采用五类马尔可夫链动力学，它们分别随机地用于概率 $p(1),...,p(5)$。动力学 1-2 是扩散，动力学 3-5 是可逆跳跃。

动力学 1：边界扩散/竞争。为了运算方便，我们转换到区域 $R_i, i = 1,...,K$ 的连续边界表示。这些曲线通过区域竞争方程 [68] 演变为最大化后验概率。设 Γ_{ij} 是 $R_i, R_j, \forall i, j$ 之间的边界，Θ_i, Θ_j 分别是两个区域的模型。点的运动 $\Gamma_{ij}(s) = (x(s), y(s))$ 遵循 $\log p(W|\mathbf{I})$ 的最陡上升方程加上沿曲线法线方向的布朗运动 dB。通过变分法，得到的方程是 [68]，

$$\frac{d\Gamma_{ij}(s)}{dt} = [f_{prior}(s) + \log \frac{p(\mathbf{I}(x(s), y(s)); \Theta_i, \ell_i)}{p(\mathbf{I}(x(s), y(s)); \Theta_j, \ell_j)} + \sqrt{2T(t)}dB]\, \vec{n}(s).$$

前两个项分别来自先验和似然。布朗运动是正态分布，其大小由温度 $T(t)$ 控制，温度 $T(t)$ 随时间减小。布朗运动有助于避免局部缺陷。对数似然比要求图像模型具有可比性。动力学 1 实现原子（或分区）空间 ϖ_{π_k} 内的扩散（即在图 8.10的矩形内移动）。

动力学 2: 模型自适应. 通过最陡上升简单地拟合一个区域的参数。可以添加布朗运动，但实际上并没有太大的区别。动力学由下式给出

$$\frac{d\Theta_i}{dt} = \frac{\partial \log p(\mathbf{I}_{R_i}; \Theta_i, \ell_i)}{\partial \Theta_i}.$$

这实现了原子（或线索）空间 $\varpi_\ell, \ell \in \{g_1, g_2, g_3, g_4, c_1, c_2, c_3\}$ 中的扩散（在图 8.10的三角形，正方形，菱形或圆形内移动）。

动力学 3-4: 分拆和合并. 假设在某个时间步长，具有模型 Θ_k 的区域 R_k 被分成具有模型 Θ_i 和 Θ_j 的两个区域 R_i 和 R_j，反之亦然。这实现了两个状态 W 到 W' 之间的跳跃，如图 8.10中的箭头所示。

$$W = (K,\, (R_k, \ell_k, \Theta_k),\, W_-) \longleftrightarrow (K+1,\, (R_i, \ell_i, \Theta_i), (R_j, \ell_j, \Theta_j),\, W_-) = W',$$

其中 W_- 表示在移动过程中未改变的剩余变量。通过 Metropolis-Hastings 方法 [42]，我们需要两个提议概率 $G(W \to dW')$ 和 $G(W' \to dW)$。$G(W \to dW')$ 是马尔可夫链在状态 W 处，提议移动到 W0 的可能性的条件概率，$G(W' \to dW)$ 是返回的提议概率。然后以概率接受提议分割

$$\alpha(W \to dW') = \min(1, \frac{G(W' \to dW)p(W'|\mathbf{I})dW'}{G(W \to dW')p(W|\mathbf{I})dW}).$$

有两种途径计算分拆提议 $G(W \to dW')$。在途径 1 中，首先选择具有概率 $q(3)$ 的分割运动，然后随机地从总共 K 个区域中选择区域 R_k。我们用 $q(R_k)$ 表示这个概率。给定 R_k，以概率 $q(\Gamma_{ij}|R_k)$ 在 R_k 内选择候选分裂边界 Γ_{ij}。然后，对于两个新区域 R_i 和 R_j，分别以概率 $q(\ell_i)$ 和 $q(\ell_j)$ 选择两个新的模型类型 ℓ_i 和 ℓ_j。然后用概率 $q(\Theta_i|R_i, \ell_i)$ 选择 $\Theta_i \in \varpi_{\ell_i}$，以概率 $q(\Theta_j|R_j, \ell_j)$ 选择 Θ_j。从而，

$$G(W \to dW') = q(3)q(R_k)q(\Gamma_{ij}|R_k)q(\ell_i)q(\Theta_i|R_i, \ell_i)q(\ell_j)q(\Theta_j|R_j, \ell_j)dW'. \tag{8.18}$$

在途径 2 中，首先选择两个新的区域模型 Θ_i 和 Θ_j，然后确定边界 Γ_{ij}。从而，

$$G(W \to dW') = q(3)q(R_k)q(\ell_i)q(\ell_j)q(\Theta_i,\Theta_j|R_k,\ell_i,\ell_j)q(\Gamma_{ij}|R_k,\Theta_i,\Theta_j)dW'. \tag{8.19}$$

我们将在后面的小节中讨论，根据区域 R_k，两条途径中的任何一条都比另一条更有效。

同样地，我们有合并提议概率，

$$G(W' \to dW) = q(4)q(R_i,R_j)q(\ell_k)q(\Theta_k|R_k,\ell_k)dW, \tag{8.20}$$

其中 $q(R_i,R_j)$ 是随机选择合并两个区域 R_i 和 R_j 的概率。

动力学 5: 切换图像模型. 这将区域 R_i 中的四个族（三个用于彩色图像）内的图像模型切换。例如，从纹理描述到样条曲面我们都有

$$W = (\ell_i,\Theta_i,\, W_-) \longleftrightarrow (\ell_i',\Theta_i',\, W_-) = W'.$$

提议概率为

$$\begin{aligned} G(W \to dW') &= q(5)q(R_i)q(\ell_i')q(\Theta_i'|R_i,\ell_i')dW', & (8.21)\\ G(W' \to dW) &= q(5)q(R_i)q(\ell_i)q(\Theta_i|R_i,\ell_i)dW. & (8.22) \end{aligned}$$

8.5.2 瓶颈

马尔可夫链的速度关键取决于其在跳跃中提议概率的设计。在我们的实验中，提议概率，如 $q(1),...,q(5)$，$q(R_k)$，$q(R_i,R_j)$，$q(\ell)$ 很容易指定，不会显著影响收敛。真正的瓶颈是由跳跃动态中的两个提议概率引起的。

1. 等式（8.18）中的 $q(\Gamma|R)$：对于给定区域 R 的划分，什么是一个较好的 Γ？$q(\Gamma|R)$ 是原子空间 ϖ_π 中的概率。

2. 等式（8.18），（8.20）和（8.22）中的 $q(\Theta|R,\ell)$：对于给定区域 R 和模型族 $\ell \in \{g_1,...,g_4,c_1,c_2,c_3\}$，什么是一个较好的 Θ？$q(\Theta|R,\ell)$ 是原子空间 ϖ_ℓ 中的概率。

值得一提的是，概率 $q(\Gamma|R)$ 和 $q(\Theta|R,\ell)$ 都不能被像区域竞争 [68] 和其他 [33] 中使用的确定性决策所取代。否则，马尔可夫链将不可逆，因此减少为贪婪算法。另一方面，如果我们选择均匀分布，它相当于盲搜索，并且马尔可夫链将在每次跳跃之前经历指数级的等待时间。实际上，等待时间的长短与线索空间的体积成比例。这些概率的设计需要在速度和鲁棒性之间取得平衡（非贪婪）。

虽然很难分析推导出这些复杂算法的收敛速度，但在一个简单的例子中观察以下定理 [41] 是显而易见的。

Theorem 8.1 *通过具有提议概率 $q(x)$ 的独立 Metropolis-Hastings 算法，给定要采样的目标密度 $p(x)$，令 $Pn(xo, y)$ 为随机行走最多 n 步到达 y 点的概率。如果存在 $\rho > 0$，那么，*

$$\frac{q(x)}{p(x)} \geq \rho, \quad \forall x,$$

那么收敛可以通过 L_1 范式距离来测量

$$||P^n(x_o,\cdot)-p|| \le (1-\rho)^n.$$

该定理表明，提议概率 $q(x)$ 应非常接近 $p(x)$ 以便快速收敛。在本例中，$q(\Gamma|R)$ 和 $q(\Theta|R,\ell)$ 应该分别等于原子空间 ϖ_π 和 ϖ_ℓ 中后验 $p(W|\mathbf{I})$ 的一些边际概率的条件概率。也就是，

$$q^*_\Gamma(\Gamma_{ij}|R_k) = p(\Gamma_{ij}|\mathbf{I},R_k), \quad q^*_\Theta(\Theta|R,\ell) = p(\Theta|\mathbf{I},R,\ell),\ \forall \ell. \tag{8.23}$$

不幸的是，q^*_Γ 和 q^*_Θ 必须整合来自整个图像 $\mathbf{I}$ 的信息，因此是很难处理的。我们必须寻求近似值，这就是数据驱动方法的用武之地。

在下节中，我们将讨论每个原子空间 ϖ_ℓ, $\ell \in \{c_1,c_2,c_3,g_1,g_2,g_3,g_4\}$ 的数据聚类和 ϖ_π 中的边缘检测。聚类和边缘检测的结果表示为非参数概率，分别在这些原子空间中逼近理想边际概率 q^*_Γ 和 q^*_Θ。

8.6 数据驱动方法

8.6.1 方法 I: 原子空间中的聚类

给定点阵 Λ 上的一张图像 $\mathbf{I}$（灰色或彩色），我们在每个像素 $v \in \Lambda$ 处提取特征向量 F^ℓ_v。F^ℓ_v 的维度取决于由 ℓ 索引的图像模型。然后我们有一组向量

$$\mho^\ell = \{F^\ell_v :\ v \in \Lambda\}.$$

实际上，为了便于计算，可以对 v 进行二次采样。该组向量通过 EM 方法 [17] 或均值偏移聚类 [12, 13] 算法到聚簇到 $\mho^\ell$。EM 聚类通过 m 个高斯混合来近似 $\mho^\ell$ 中的点密度，并且通过软聚类分配从 m 均值聚类延伸到每个向量 F_v。均值偏移算法假设 $\mho^\ell$ 的一个非参数分布并且在其密度中寻找模式（局部最大值）（在一些高斯窗口平滑之后）。两种算法都返回 m 加权簇的列表 $\Theta^\ell_1,\Theta^\ell_2,...,\Theta^\ell_m$，权重 $\omega^\ell_i, i=1,2,...,m$，我们用下式表示

$$\mathcal{P}^\ell = \{\ (\omega^\ell_i,\Theta^\ell_i) :\ i=1,2,...,m.\ \}. \tag{8.24}$$

对于 $\ell \in \{c_1,c_3,g_1,g_2,g_3,g_4\}$，我们将 $(\omega^\ell_i,\Theta^\ell_i)$ 称为 ϖ_ℓ 中的加权原子（或线索）粒子。m 大小的选择是保守的，或者它可以在粗到精策略中以极限 $m=|\mho^\ell|$ 来计算。更多细节见 [12, 13]。

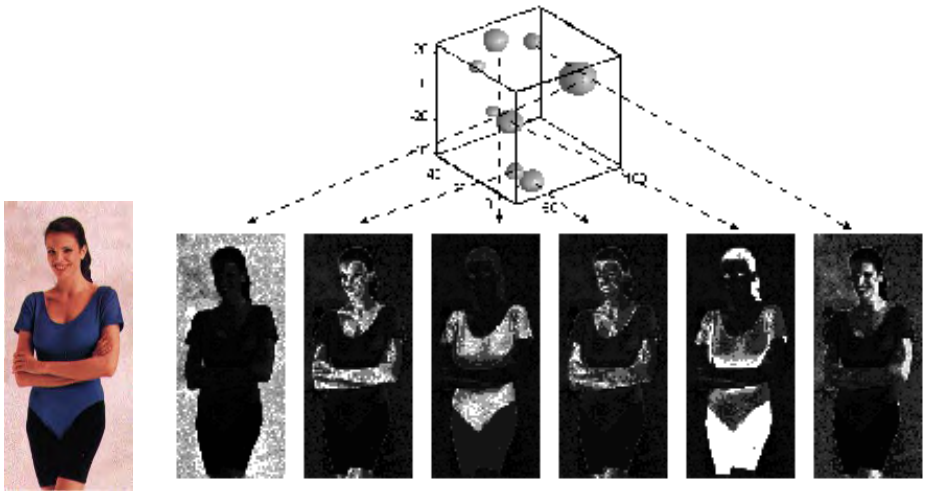
在聚类算法中，每个特征 F^ℓ_v 及其位置 v 被分类为一个簇 Θ^ℓ_i，使得

$$S^\ell_{i,v} = p(F^\ell_v;\Theta^\ell_i), \qquad \text{with } \sum_{i=1}^m S^\ell_{i,v} = 1, \quad \forall v \in \Lambda,\ \forall \ell.$$

这是一个软分配，可以通过从 F_v 到聚类中心的距离来计算。我们称

$$S^\ell_i = \{S^\ell_{i,v} :\ v \in \Lambda\}, \quad \text{for } i=1,2,...,m,\ \forall \ell \tag{8.25}$$

是与线索粒子 Θ^ℓ_i 相关联的显著图。



Input $\mathbf{I}$ Color clusters and their saliency maps $S_i^{c_1}, i = 1, ..., 6$

图 8.11: (L^*, u^*, v^*) 空间中 ϖ_{c_1} 的彩色图像及其簇 (L^*, u^*, v^*)，第二行包含与彩色簇相关联的六个显著图。Tu 和 Zhu 提供 [58].

计算 ϖ_{c_1} 中的线索粒子. 对于彩色图像，我们采用 $F_v = (L_v, U_v, V_v)$ 并应用均值偏移算法 [12,13] 来计算 ϖ_{c_1} 中的彩色簇。例如，图 8.11显示了立方体空间中的一些颜色簇（球），用于简单的彩色图像。球的大小代表权重 $\omega_i^{c_1}$。每个簇与显著图 $S_i^{c_1}$，$i = 1, 2, ..., 6$ 相关联，其中明亮区域表示较高概率。从左到右，地图分别是背景，皮肤，衬衫，阴影皮肤，裤子和头发，突出显示的皮肤。

(a) (b) (c) (d) (e) (f) (g) (h)

图 8.12: （a）-（d）是与 ϖ_{c_3} 中的四个聚类相关的显著图。（e）-（h）是四个聚类的颜色样条曲面。Tu 和 Zhu 提供 [58].

计算 ϖ_{c_3} 中的线索粒子。每个点 v 将其颜色 $\mathbf{I}_v = (L_v, U_v, V_v)$ 作为“表面高度”，并且我们应用 EM 聚类来找到样条曲面模型。图 8.12显示了女性图像的聚类结果。图 8.12（a-d）是显著图 $S_i^{c_3}$，$i = 1, 2, 3, 4$。图 8.12（e-h）是依据拟合样条曲面的四个重建图像，其反映了一些全局照明变化。

计算 ϖ_{g_1} 中的线索粒子。在该模型中，特征空间 $F_v = \mathbf{I}_v$ 仅是强度，υ^{g_1} 是图像强度直方图。我们简单地应用均值偏移算法来获得直方图模型（峰值），并且每个峰值的宽度决定其方差。

图 8.13显示了图 8.21（a）中所示的斑马图像的六个显著图 $S_i^{g_1}, i = 1, 2, ..., 6$。在左侧的聚类图中，每个像素都分配给其最可能的粒子。

计算 ϖ_{g_2} 中的线索粒子。对于 ϖ_{g_2} 中的聚类，在每个子采样像素 $v \in \Lambda$ 处，我们计算 F_v，作为在以 v 为中心的局部窗口上累积的局部强度直方图 $F_v = (h_{v0}, ..., h_{vG})$。然后应用 EM 聚类计算线索粒子。每个粒子 $\Theta_i^{g_2}, i = 1, ..., m$ 是直方图。该模型用于杂波区域。

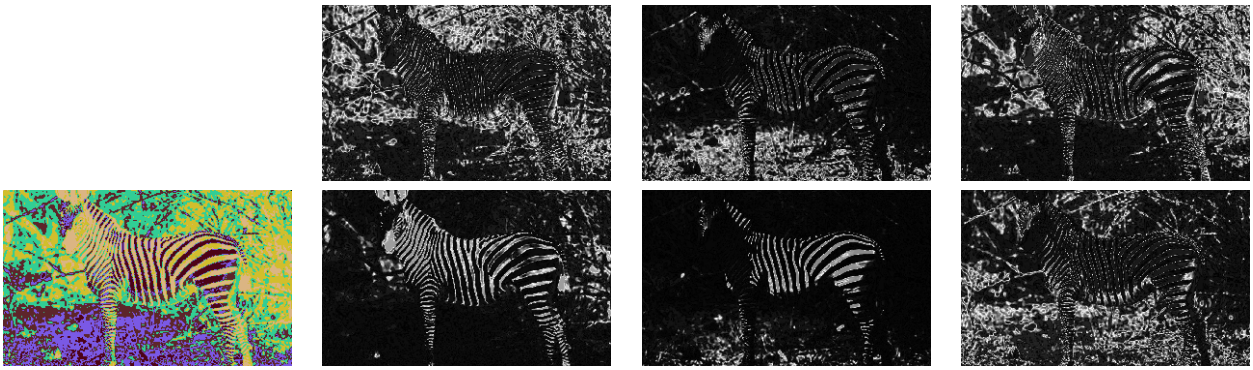


图 8.13: ϖ_{g_1} 的聚类图（左）和斑马图像的六个显著图 $S_i^{g_1}, i = 1, ..., 6$。Tu 和 Zhu 提供 [58].

图 8.14显示了同一斑马图像上的聚类结果。

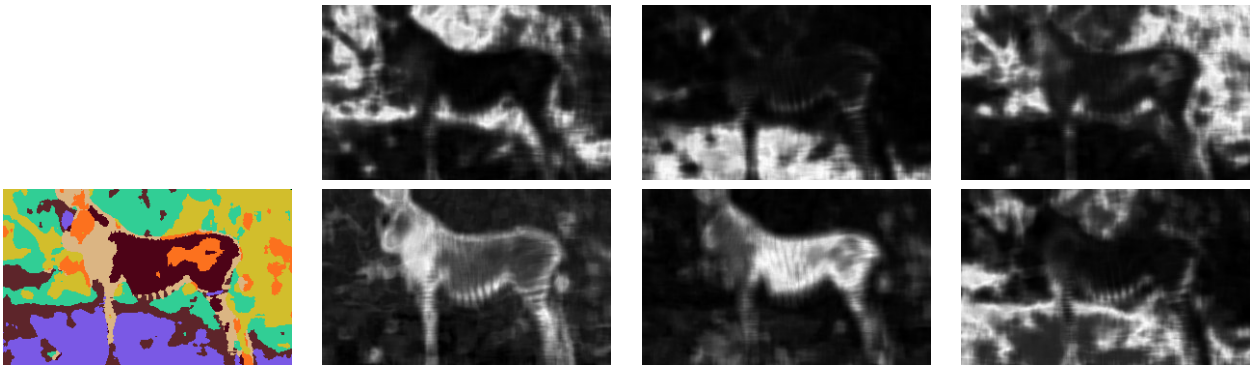


图 8.14: ϖ_{g_2} 的聚类图（左）和斑马图像的六个显著图 $S_i^{g_2}, i = 1..6$。Tu 和 Zhu 提供 [58]。

计算 ϖ_{g_3} 中的线索粒子。在每个子采样像素 $v \in \Lambda$ 处，我们在 12×12 像素的局部窗口上计算一组 8 个局部直方图，用于 8 个滤波器。为计算方便，我们选择 8 个滤波器：一个 δ 滤波器，两个梯度滤波器，一个拉普拉斯高斯滤波器和四个 Gabor 滤波器。每个直方图有 9 个箱。那么 $F_v^{g_3} = (h_{v,1,1}, ..., h_{v,8,9})$ 就是这个特征。应用 EM 聚类来找到 m 个平均直方图 $\bar{\mathbf{h}}_i, i = 1, 2, ..., m$。我们可以从 $\bar{\mathbf{h}}_i$，$i = 1, 2, ..., m$ 计算纹理模型 $\Theta_i^{g_3}$ 的线索粒子。[66] 详细说明了这种变换。

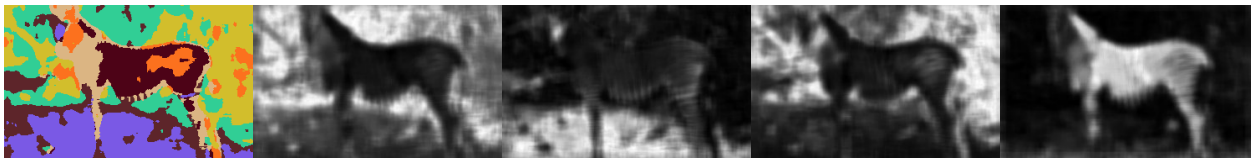


图 8.15: 纹理聚类。聚类图（左）和四个粒子的四个显著图 $\Theta_i^{g_3}, i = 1, 2, ..., 4$。Tu 和 Zhu 提供 [58].

图 8.15显示斑马图像上的纹理聚类结果，左侧是一个聚类图，四个粒子的四个显著图 $\Theta_i^{g_3}, i = 1, 2, ..., 4$。**计算 ϖ_{g_4} 中的线索粒子。**每个点 v 将其强度 $\mathbf{I}_v = F_v$ 作为“表面高度”，并且我们应用 EM 聚类来找到样条曲面模型。图 8.16显示了具有四个表面的斑马图像的聚类结果。第二行显示了恢复一些全局照明变化的四个表面。与将斑马条纹捕捉为整个区域的纹理聚类结果不同，表面模型将黑色和白色条纹分离为两个区域--另一种有效感知。有趣的是，斑马皮肤中的黑色和白色条纹都有阴影变化，这些变化由样条模型拟合。

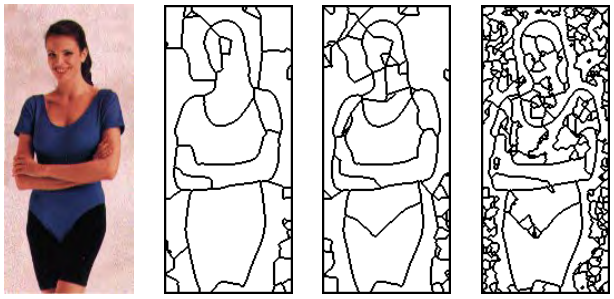
聚类图　　四个显著图和表面

图 8.16: 贝塞尔曲面模型下斑马图像的聚类结果。左图是聚类图。右侧的第一行图像是显著图。第二行显示使用表面高度作为强度的拟合表面。Tu 和 Zhu 提供 [58].

8.6.2　方法二：边缘检测

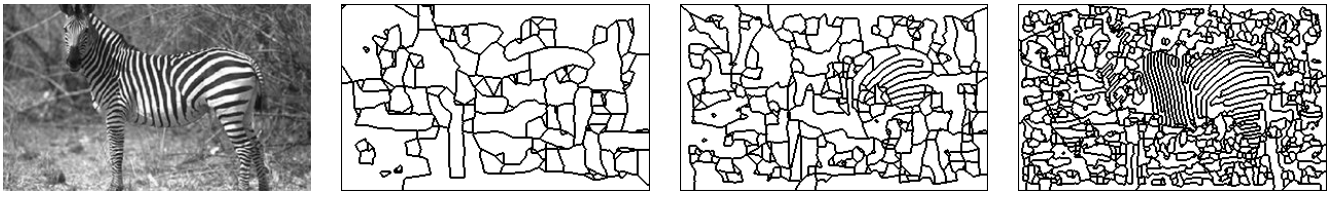
我们使用 Canny 边缘检测器 [10] 和颜色边缘 [34] 中的方法检测强度边缘。然后跟踪边缘以形成图像晶格的分区。我们根据边缘强度选择三个尺度的边缘，从而以三个从粗到细的尺度计算分割图。我们的选择不考虑细节，但使用两个运行的例子显示一些结果：女人和斑马图像。

图 8.17显示了彩色图像和三个分区比例。由于此图像具有强烈的颜色提示，因此边缘图可以非常有用地了解区域边界的位置。相比之下，斑马图像的边缘图非常混乱，如图 8.18所示。



(a)　(b)　(c)　(d)

图 8.17: 彩色图像的三个细节尺度的分区图。(a) 输入图像。(b) 比例 1. (c) 比例 2. (d) 比例 3. Courtesy of Tu and Zhu [58].



(a)　(b)　(c)　(d)

图 8.18: 灰度图像和三个尺度的三个分区图。(a) 输入图像。(b) 比例 1. (c) 比例 2. (d) 比例 3. Courtesy of Tu and Zhu [58].

8.7 计算重要性提案概率

人们普遍认为，聚类和边缘检测算法有时可以为某些图像产生良好的分割或完美的结果，通常它们对于通用图像来说没那么可靠，如图 8.11-8.18 中的实验所示。同样，有时其中一个图像模型和边缘检测标尺在分割某些区域方面比其他模型和标度更好，但我们不知道先验图像中存在哪些类型的区域。因此，我们计算多个尺度的所有模型和边缘检测，然后概率地利用聚类和边缘检测结果。MCMC 理论提供了一个框架，用于在全局定义的贝叶斯后验概率的指导下以原则方式整合该概率信息。

计算重要性提议概率 $q(\Theta|R,\ell)$**.** 原子空间 ϖ_ℓ 中的聚类方法输出一组加权提示粒子 $\mathcal{P}^\ell$。$\mathcal{P}^\ell$ 以 ϖ_ℓ 编码非参数概率，

$$q(\Theta|\Lambda,\ell)=\sum_{i=1}^{m}\omega_i^\ell G(\Theta-\Theta_i^\ell),\quad \text{with }\sum_{i=1}^{m}\omega_i^\ell=1, \tag{8.26}$$

其中 $G(x)$ 是以 0 为中心的 Parzen 窗口。事实上，在提示空间 $\varpi_\ell,\ell\in\{g_1,g_2,g_3,g_4,c_1,c_3\}$ 中，因为分区 π 在 EM 聚类中被积分，$q(\Theta|\Lambda,\ell)=q(\Theta|\mathbf{I})$) 是后验 $p(W|\mathbf{I})$ 的边际概率的近似值。

对于整个图像计算 $q(\Theta|\Lambda,\ell)$ 一次，并且在运行时针对每个 R 计算 $q(\Theta|R,\ell)$。该方法以下列方式进行。每个簇 $\Theta_i^\ell,i=1,2,...,m$ 从区域 R 中的像素 $v\in R$ 接收实值投票，并且累积投票是与 Θ_i^ℓ 相关联的显著图 S_i^ℓ 的总和。象征性地，p_i 由下式给出

$$p_i=\frac{1}{|R|}\sum_{v\in R}S_{i,v}^\ell,\quad i=1,2,...,m,\ \forall\ell.$$

显然，获得更多选票的集群应该有更高的选择机会。因此，我们为区域 R 采样新的图像模型 Θ，

$$\Theta\sim q(\Theta|R,\ell)=\sum_{i=1}^{m}p_i G(\Theta-\Theta_i^\ell). \tag{8.27}$$

公式 (8.27) 解释了我们如何选择（或提出）区域 R 的图像模型。我们首先根据概率 $p=(p_1,p_2,...,p_m)$ 随机绘制一个聚类 i 然后取一个 Θ_i^ℓ 的随机扰动。因此，任何 $\Theta\in\varpi_\ell$ 具有非零概率以被选择用于稳健性和遍历性。直观地，利用本地投票的聚类结果以概率方式提出空间的“最热”部分以指导跳跃动态。实际上，可以在较小的本地窗口上实现多分辨率聚类算法。在这种情况下，簇 $\Theta_i^\ell,i=1,2,...,m$ 将以某些开销计算为代价更有效。

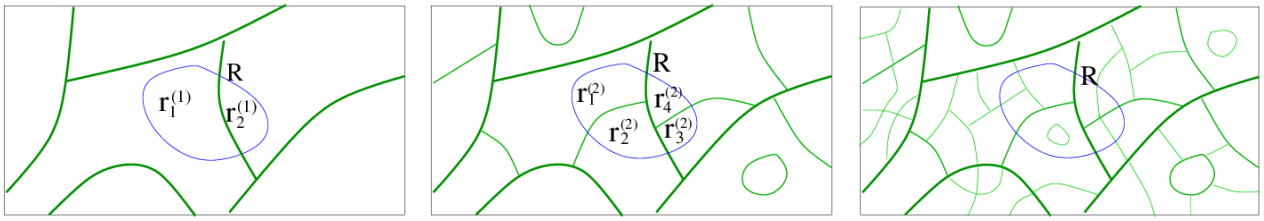
计算重要性提议概率 $q(\Gamma|R)$**.** 通过边缘检测和跟踪，我们获得了在多个尺度 $s=1,2,3$ 处由 $\Delta^{(s)}$ 表示的分区图。实际上，每个分区图 $\Delta^{(s)}$ 由一组“元区域” $r_i^{(s)}$ 组成

$$\Delta^{(s)}(\Lambda)=\{r_i^{(s)}:\ i=1,2,...,n,\cup_{i=1}^{n}r_i^{(s)}=\Lambda\},\quad \text{for } s=1,2,3.$$

这些元区域组合使用以形成 $K\le n$ 区域 $R_1^{(s)},R_2^{(s)},...,R_K^{(s)}$，

$$R_i^{(s)}=\cup_j r_j^{(s)},\quad \text{with } r_j^{(s)}\in\Delta^{(s)},\ \forall i=1,2,...,K.$$

可以进一步要求区域 $R_i^{(s)}$ 中的所有元区域连接。令 $\pi_k^{(s)}=(R_1^{(s)},R_2^{(s)},...,R_k^{(s)})$ 表示基于 $\Delta^{(s)}$ 的 k 分区。$\pi_k^{(s)}$ 与一般 k 分区 π_k 不同，因为 $\pi_k^{(s)}$ 中的区域 $R_i^{(s)},i=1,...,K$ 限于元区域。我们基于分区映射 $\Delta^{(s)}$ 表示所有



scale 1 scale 2 scale 3

图 8.19: 候选区域 R_k 以三个比例叠加在分区图上，用于计算未决分割的候选边界 Γ_{ij}。Courtesy of Tu and Zhu [58].

k 分区的集合

$$\Pi_k^{(s)} = \{(R_1^{(s)}, R_2^{(s)}, ..., R_k^{(s)}) = \pi_k^{(s)} : \quad \cup_{i=1}^{k} R_i^{(s)} = \Lambda\}. \tag{8.28}$$

我们将 $\Pi_k^{(s)}$ 中的每个 $\pi_k^{(s)}$ 称为原子（分区）空间中的 k 分区粒子 ϖ_{π_k}。与提示空间中的聚类一样，$\Pi_k^{(s)}$ 是 ϖ_{π_k} 的稀疏子集，并且它将搜索范围缩小到最有希望的部分。因此，每个分区映射 $\Delta^{(s)}$ 编码原子（分区）空间中的概率 ϖ_{π_k}，并且

$$q^{(s)}(\pi_k) = \frac{1}{|\Pi_k^{(s)}|} \sum_{j=1}^{|\Pi_k^{(s)}|} G(\pi_k - \pi_{k,j}^{(s)}), \quad \text{for } s = 1,2,3. \ \forall k. \tag{8.29}$$

$G()$ 是一个以 0 为中心的平滑窗口，其平滑度考虑了边界变形并在每个分区粒子周围形成一个簇，$\pi_k - \pi_{k,j}^{(s)}$ 测量两个分区图 π_k 和 $\pi_{k,j}^{(s)}$ 之间的差异。

在最精细的分辨率中，所有元区域都减少为像素，然后 $\Pi_k^{(s)}$ 等于原子空间 ϖ_{π_k}。总之，所有尺度的分区图都以 ϖ_{π_k} 编码非参数概率，

$$q(\pi_k) = \sum_s q(s) q^{(s)}(\pi_k), \quad \forall k.$$

该 $q(\pi_k)$ 可以被认为是边际后验概率 $p(\pi_k|\mathbf{I})$ 的近似值。

对整个图像计算一次分区映射 $\Delta^{(s)}, \forall s$ (或者 $q(\pi_k), \forall k$ ，然后从 $q(\pi_k)$ 计算重要性提议概率 $q(\Gamma|R)$ 每个区域作为运行时的条件概率，就像在提示空间中一样。图 8.19说明了一个例子。我们在三个尺度上标出了分区图 $\Delta^{(s)}(\Lambda)$，并且对于 $s = 1,2,3$，边缘分别在宽度 $3,2,1$ 处示出。提议候选区域 R 分裂。$q(\Gamma|R)$ 是提出分裂边界 Γ 的概率。我们将 R 叠加在三个分区图上。R 和元区域之间的交叉点产生三组

$$\Delta^{(s)}(R) = \{r_j^{(s)} : \ r_j^{(s)} = R \cap r_j \text{ for } r_j \in \Delta^{(s)}(\Lambda), \ \text{and } \cup_i r_i^{(s)} = R\}, \quad s = 1,2,3.$$

例如，图 8.19, $\Delta^{(1)}(R) = \{r_1^{(1)}, r_2^{(1)}\}$, $\Delta^{(2)}(R) = \{r_1^{(2)}, r_2^{(2)}, r_3^{(2)}, r_4^{(2)}\}$, 等等.

因此，我们可以基于 $\Delta^{(s)}(R)$ 将 $\pi_c^{(s)}(R) = (R_1^{(s)}, R_2^{(s)}, ..., R_c^{(s)})$ 定义为区域 R 的 c 分区。并将 R 的 c 分区空间定义为

$$\Pi_c^{(s)}(R) = \{(R_1^{(s)}, R_2^{(s)}, ..., R_c^{(s)}) = \pi_c^{(s)}(R) : \quad \cup_{i=1}^{c} R_i^{(s)} = R\}, \quad \forall s. \tag{8.30}$$

我们可以将 $\Pi_c^{(s)}(R)$ 上的分布定义为

$$q^{(s)}(\pi_c(R)) = \frac{1}{|\Pi_c^{(s)}(R)|} \sum_{j=1}^{|\Pi_c^{(s)}(R)|} G(\pi_c - \pi_{c,j}^{(s)}(R)), \quad \text{for } s = 1,2,3, \ \forall c. \tag{8.31}$$

因此，在一般情况下，可以建议将 R 分成 c 个部分，

$$\pi_c(R) \sim q(\pi_c(R)) = \sum_s q(s) q^{(s)}(\pi_c(R)).$$

也就是说，我们首先选择概率为 $q(s)$ 的尺度 s。$q(s)$ 取决于 R. 例如，对于大区域 R，我们可以选择具有较高概率的粗尺度，并为小区域选择精细尺度。然后我们从集合 $\Pi_c^{(s)}(R)$ 中选择一个 c 分区。在我们的实现中，选择 $c = 2$ 作为易于实现的特殊情况。显示通过多次组合 $\pi_2(R)$ 可以产生区域 R，$\pi_c(R)$ 的任意 c 分区是微不足道的。显然，选择大型 c 会有很大的开销。

计算 $q(\Theta_i, \Theta_j | R, \ell_i, \ell_j)$ 和 $q(\Gamma_{ij} | R, \Theta_i, \Theta_j)$. 在某些情况下，我们找到了第二条路线，我们讨论了设计 MCMC 动力学 3-4（见公式 (8.19)），这对于分割区域非常有用。例如，图 8.21中有两种方法可以感知斑马。人们将斑马视为一个纹理区域（通过 ϖ_{g_3}) 中的模型）。另一个将其视为黑色条纹的一个区域加上一个白色条纹区域，因此使用 ϖ_{g_1} 或 ϖ_{g_4} 中的模型。马尔可夫链应该能够有效地在两种感知之间切换（见图 8.21.b-d 中的结果）。对于任何纹理区域和强度区域之间的过渡，这是必要且典型的。

因为这种纹理中的条纹数量很大，所以第一个分割过程（路径 1）非常无效，并且它一次在一个条带上工作。这激发了分裂动力学的第二条路径。对于候选区域 R，我们首先提出两个新的区域模型（我们总是假设相同的标签 $\ell_i = \ell_j$），这可以通过对重要性提议概率 $q(\Theta | R, \ell)$ 进行两次采样来完成，所以

$$(\Theta_i, \Theta_j) \sim q(\Theta_i, \Theta_j | R, \ell_i, \ell_j) = q(\Theta_i | R, \ell_i) q(\Theta_j | R, \ell_j).$$

显然，当我们选择 Θ_j 时，我们从候选集中排除 Θ_i。然后，我们通过根据显著图的概率随机标记 R 中的像素来决定边界 Γ $q(\Gamma_{ij} | R, \Theta_i, \Theta_j)$。

一个统一的框架。 总结本节，DDMCMC 范例为理解许多现有图像分割算法的作用提供了统一的框架。首先，边缘检测和跟踪方法 [10, 34] 隐含地计算分区空间 上的边际概率 $q(\pi | \mathbf{I})$ 。其次，聚类算法 [12, 13] 计算模型空间的边际概率 ϖ_ℓ 用于各种模型 ℓ。第三，分裂 - 合并和模型切换 [3] 实现了跳跃动态。第四，区域增长和竞争方法 [48, 68] 实现了扩展区域边界的扩散动力学。

8.8 计算多种不同的解决方案

8.8.1 动机和数学原理

DDMCMC 算法从后 $W \sim p(W | \mathbf{I})$ 无限地采样解。为了提取最佳结果，退火策略可以与传统的最大后验（MAP）估计器一起使用

$$W^* = \arg\max_{W \in \Omega} p(W | \mathbf{I}).$$

计算多个不同解决方案的理想且通常是关键的原因详述如下。

1. 自然场景本质上是模糊的，对于图像 **I** 在视觉感知中存在许多竞争组织和解释。

2. . 对于稳健性，当分段过程与特定任务集成时，决策应留在计算的最后阶段。因此，最好保持一套典型的解决方案。

3. 当先验概率模型和似然模型不完美时，保留多个解是必要的，因为全局最优解可能在语义上不比其他一些较低的局部最大值更有意义。

然而，仅仅保留马尔可夫链序列中的一组样本是不够的，因为它经常收集一组彼此平凡不同的分段。相反，可以使用计算空间 Ω 中重要和独特解决方案的数学原理，它依赖于第 2.5节中提出的技术。用于保留重要性抽样中的样本多样性。

设 $S = \{(\omega_i, W_i) : i = 1, ..., K\}$ 是一组 K 加权解，我们称之为场景粒子，其权重是它们的后验概率 $\omega_i = p(W|\mathbf{I}), i = 1, 2, ..., K$.（注意，有一点点乱用符号，我们使用 K 表示之前 W 中的区域数。这里是不同的 K）。S 编码非参数概率通过 Ω 和

$$\hat{p}(W|\mathbf{I}) = \sum_{i=1}^{K} \frac{\omega_i}{\omega} G(W - W_i), \quad \sum_{i=1}^{K} \omega_i = \omega,$$

其中 G 是 Ω 中的高斯窗口。

由于所有图像模糊都是在贝叶斯后验概率中捕获的，为了反映内在的模糊性，我们应该计算出最能保持后验概率的解 S 集。因此，我们让 $\hat{p}(W|\mathbf{I})$ 通过在复杂性约束 $|S| = K$ 下最小化 Kullback-Leibler 散度 $D(p||\hat{p})$ 来逼近 $p(W|\mathbf{I})$，

$$S^* = \arg\min_{|S|=K} D(p||\hat{p}) = \arg\min_{|S|=K} \int p(W|\mathbf{I}) \log \frac{p(W|\mathbf{I})}{\hat{p}(W|\mathbf{I})} dW. \tag{8.32}$$

该标准扩展了传统的 MAP 估计器。

8.8.2 用于多种解决方案的 K-adventurers 算法

幸运的是，由于多模态后验概率 $p(W|\mathbf{I})$ 的两个观测值，可以通过距离度量 $\hat{D}(p||\hat{p})$ 相当精确地估计 KL-散度 $D(p||\hat{p})$，这是可计算的。我们总是可以用高斯的混合物来表示 $p(W|\mathbf{I})$，即具有足够大的 N 的一组 N 个粒子。通过遍历性，马尔可夫链应该随着时间的推移访问这些重要的模式！因此，我们的目标是从马尔可夫链采样过程中提取 K 个不同的解决方案。

为了实现这一目标，可以使用称为 K-adventurers 的算法。[2] 假设我们在步骤 t 有一组 K 粒子 S. 在时间 $t+1$，我们通常在成功跳跃之后通过 MCMC 获得新粒子（或多个粒子）。我们通过添加新增加集合 S 到 S_+ 粒子）。然后，我们通过最小化近似 KL 发散 $\hat{D}(p_+||p_{\text{new}})$ 从 S_+ 中消除一个粒子（或多个粒子）以获得 S_{new}。

K-adventurers 算法

[2]The name follows a statistics metaphor told by Mumford to S.C. Zhu. A team of K adventurers want to occupy K largest islands in an ocean while keeping apart from each other's territories.

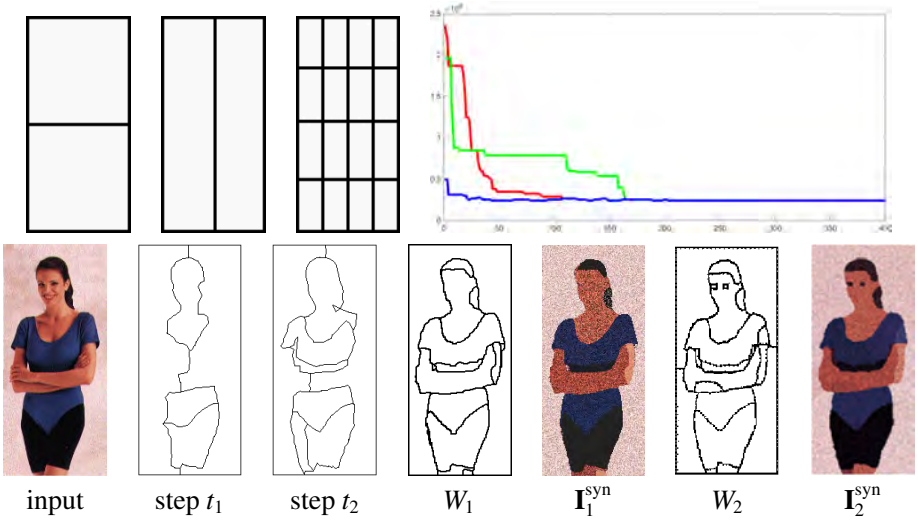


图 8.20: DDMCMC 用两种解决方案分割彩色图像。请参阅文本以获取解释。Courtesy of Tu and Zhu [58].

1. Initialize S and $\hat{p}$ by repeating one initial solution K times.
2. Repeat
3. Compute a new particle $(\omega_{K+1}, \mathbf{x}_{K+1})$ by DDMCMC after a successful jump.
4. $S_+ \leftarrow S \bigcup \{(\omega_{K+1}, \mathbf{x}_{K+1})\}$.
5. $\hat{p} \leftarrow S_+$.
6. For $i = 1, 2, \ldots, K+1$ do
7. $S_{-i} \leftarrow S_+ / \{(\omega_i, \mathbf{x}_i)\}$.
8. $\hat{p}_{-i} \leftarrow S_{-i}$.
9. $d_i = D(p||\hat{p}_{-i})$.
10. $i^* = \arg\min_{i \in \{1,2,\ldots,K+1\}} d_i$.
11. $S \leftarrow S_{-i^*}$, $\hat{p} \leftarrow \hat{p}_{-i^*}$.

在实践中，我们运行多个马尔可夫链并以批量方式将新粒子添加到集合 S.

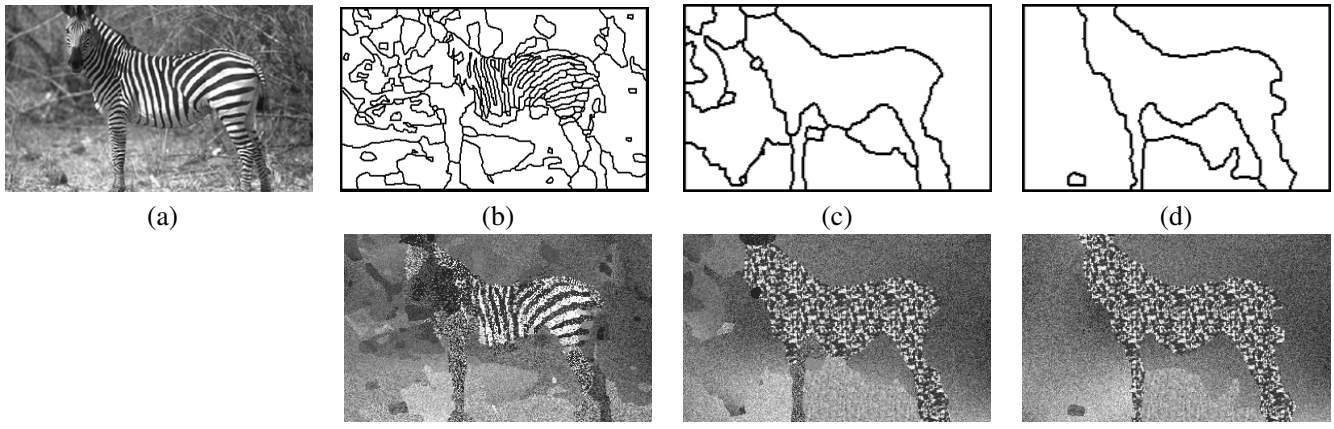
8.9 图像分割实验

DDMCMC 范例在许多灰度，彩色和纹理图像上进行了广泛测试。本节介绍了一些示例，我们的网站[3]上提供了更多示例。它还在 Berkeley group [4]的 50 个自然图像的基准数据集中进行了测试 [40]，其中 DDMCMC 和其他方法如 [52] 的结果与多个方法相比较。人类受试者。每个测试算法对所有基准图像使用相同的参数设置，因此结果纯粹自动获得。

我们首先在彩色女性形象上展示我们的工作实例。按照图 8.17中边缘的重要性提议概率和图 8.11中的颜色聚类，我们模拟了具有三个不同初始分割的三个马尔可夫链，如图 8.20所示（顶行）。三个 MCMC 的能量变化（$-\log p(W|\mathbf{I})$）绘制在图 8.20中，相对于时间 s。图 8.20显示了使用 K-adventurers 算法通

[3]See http://vcla.stat.ucla.edu/old/Segmentation/Segment.htm
[4]See https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/

过马尔可夫链获得的两种不同解 W_1, W_2。为了验证计算的解 W_i，我们通过从似然 $\mathbf{I}_i^{\text{syn}} \sim p(\mathbf{I}|W_i), i = 1,2$ 中采样来合成图像。综合是检查分割中模型充分性的好方法。



(a) (b) (c) (d)

(e) (f) (g)

图 8.21: 用三种解决方案对灰度斑马纹图像进行的实验。（a）输入图像。（b）-（d）是斑马图像的三个解，$W_i, i = 1,2,3$。（e）-（g）是用于验证结果的合成图像 $\mathbf{I}_i^{\text{syn}} \sim p(\mathbf{I}|W_i^*)$。Courtesy of Tu and Zhu [58].

图 8.21显示了灰度级斑马图像上的三个分割。如前所述，本节中的 DDMCMC 算法只有一个自由参数 γ，它是先前模型中的“杂波因子”（见公式 (8.7)）。它控制分段的范围。大 γ 鼓励大区域的粗分割。我们通常通过分别设置 $\gamma = 1.0, 2.0, 3.0$ 来提取三个等级的结果。在我们的实验中，K-adventurers 算法仅对在一定范围内计算不同解决方案有效。我们期望如果我们形成具有多个尺度的图像金字塔并且在每个尺度下使用 K-adventurers 算法进行分割，然后将结果传播并精细化到下一个更精细的尺度，则可以将参数 γ 固定为常数。

对于斑马图像，W_1 分割出黑白条纹，而 W_2 和 W_3 将斑马视为 a 纹理区域。合成图像 $\mathbf{I}_i^{\text{syn}} \sim p(\mathbf{I}|W_i), i = 1,2,3$ 表明纹理模型不充分，因为我们仅选择 8 个小滤波器以便于计算。此外，样条曲面模型在分割地面和背景草中起着重要作用，这通过 $\mathbf{I}_2^{\text{syn}}$ and $\mathbf{I}_3^{\text{syn}}$ 中的全局阴影变化得到验证。

图 8.22 和 8.23 使用相同的算法显示一些其他灰度和彩色图像。我们显示输入，从任意初始条件开始的分割和从似然 $\mathbf{I}^{\text{syn}} \sim p(\mathbf{I}|W)$ 绘制的合成图像。这些图像的 γ 值大多设定为 1.5，少数在 1.0-3.5 处获得。在一开始学习伪似然纹理模型之后，在 Pentium III PC 上花费大约 10-30 分钟（取决于图像内容的复杂性）来分割具有中等尺寸的图像，例如 350×250 像素。

合成图像显示我们需要使用更多的随机模型，如点和曲线过程，以及像面等对象。例如，在图 8.23的第一行。足球场中的乐队形成了一个未被捕获的点过程。合成中也缺少面部。

图 8.24 显示了基准研究中 50 个自然图像中的三个灰度图像，包括颜色和灰度。输入（左），DDMCMC 的分割结果（中），以及人类主体的手动分割（右）。

8.10 应用：图像分析

我们将图像解析定义为将图像 $\mathbf{I}$ 分解为其组成视觉图案的任务。输出由分层图表 W 表示 - 称为“解析图”。目标是优化贝叶斯后验概率 $p(W|\mathbf{I})$。图 8.25 示出了一个典型的例子，其中足球场景首先被分成粗略的三个部分：前景中的人，运动场和观众。这三个部分在第二级进一步分解为九个视觉模式：一

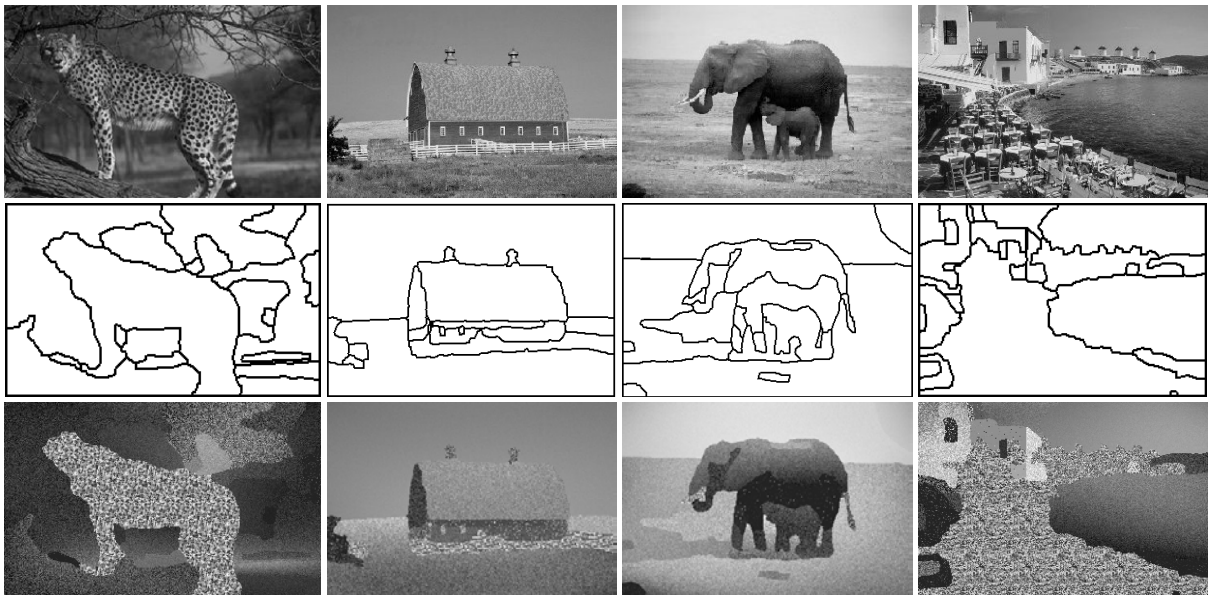


图 8.22: DDMCMC 的灰度图像分割。顶部：输入图像，中间：分割结果 W，底部：合成图像 $\mathbf{I}^{\text{syn}} \sim p(\mathbf{I}|W)$ 与分割结果 W。Courtesy of Tu and Zhu [58].

个面，三个纹理区域，一些文本，一个点过程（场上的带），一个曲线过程（场上的标记），一个颜色区域，以及附近人的地区。原则上，我们可以继续分解这些部分，直到达到分辨率标准。解析图在精神上类似于语音和自然语言处理中使用的解析树 [38]，除了它可以包括水平连接（参见图 8.25中的虚线曲线），用于指定不同视觉模式之间的空间关系和边界共享。

与自然语言处理一样，解析图不是固定的，取决于输入图像。图像解析算法必须在运行中构造解析图[5]。我们的图像解析算法由一组可逆的马尔可夫链跳 [26] 组成，每种类型的跳转对应于一个运算符，用于重新配置解析图（即创建或删除节点或改变节点属性的值）。这些跳跃在可能的解析图空间中组合形成遍历和可逆的马尔可夫链。马尔可夫链概率保证收敛到不变概率 $p(W|\mathbf{I})$，马尔可夫链将模拟来自

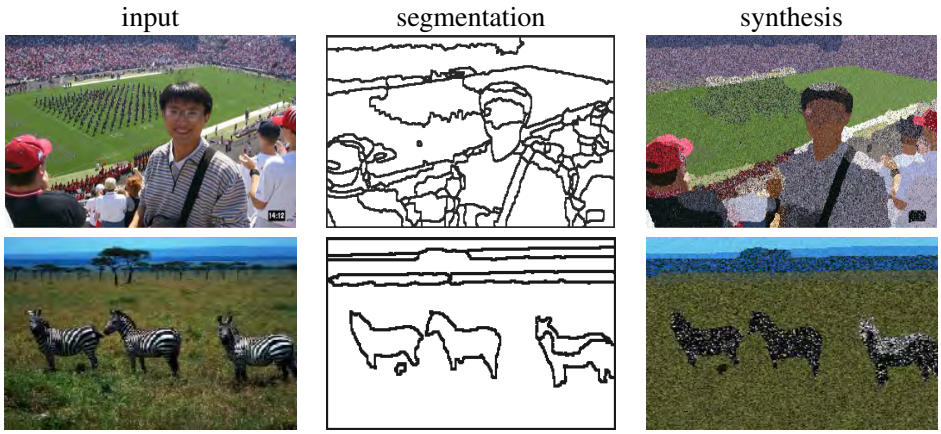


图 8.23: DDMCMC 的彩色图像分割。左：输入图像，中间：分割结果 W，右：合成图像 $\mathbf{I}^{\text{syn}} \sim p(\mathbf{I}|W)$ 与分割结果 W。Courtesy of Tu and Zhu [58].

[5]Unlike most graphical inference algorithms in the literature which assume fixed graphs, see belief propagation [65].

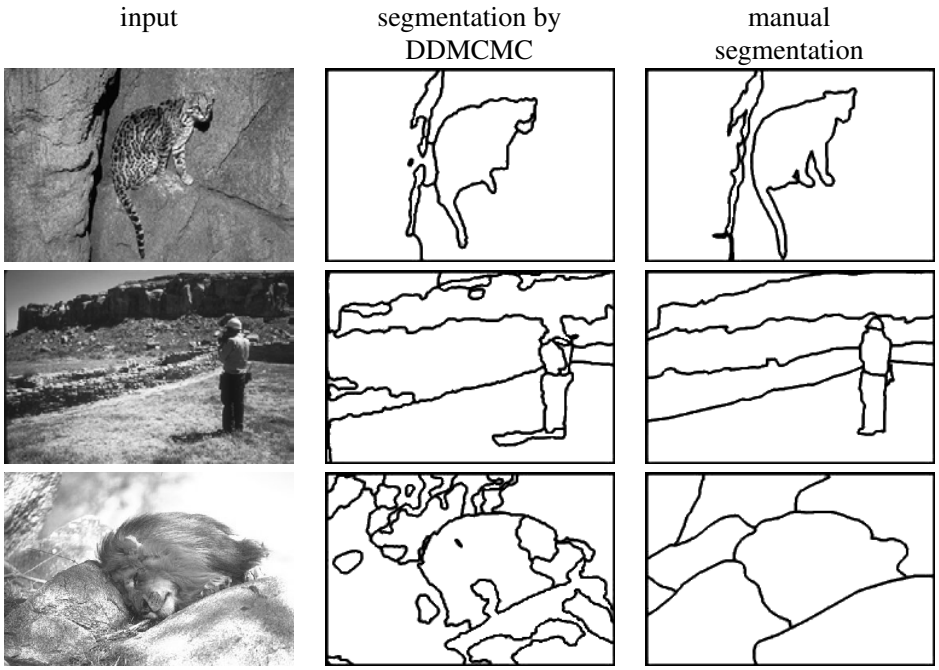


图 8.24: DDMCMC 对 Martin 进行基准测试的一些细分结果。DDMCMC（中）与人类受试者（右）的结果相比，上述结果的误差分别为 0.1083, 0.3082, 和 0.5290。Courtesy of Tu and Zhu [58].

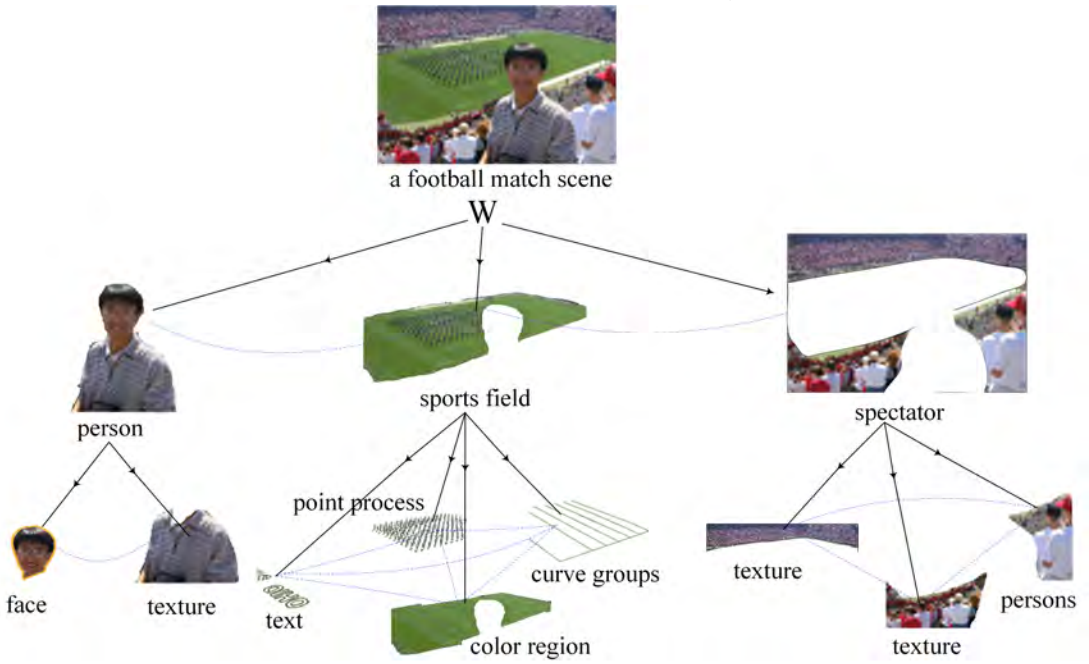


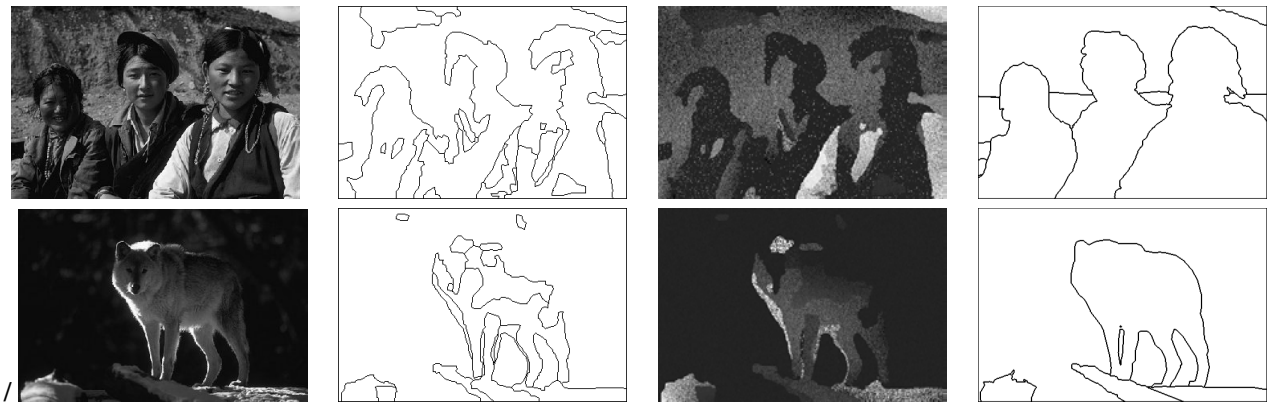
图 8.25: 图像解析示例。解析图是分层的，并结合了生成模型（向下箭头）具有水平连接（虚线），其指定视觉图案之间的空间关系。有关包含节点属性变量的更抽象的表示，请参见图 8.28。Courtesy of Tu et al. [56].

该概率的公平样本。[6]这种方法建立在以前的数据驱动马尔可夫链蒙特卡罗（DDMCMC）的工作基础上，

[6]For many natural images the posterior probabilities $P(W|\mathbf{I})$ are strongly peaked and so stochastic samples are close to the maxima of the posterior. So in this section we do not distinguish between sampling and inference (optimization).

用于识别 [69]，分割 [58]，分组 [59] 和图分区 [1, 2]。

图像解析根据生成模型 $p(\mathbf{I}|W)$ 和 $p(W)$ 寻找输入图像的完整生成性解释，用于自然图像中出现的各种视觉模式，见图 8.25。这与其他计算机视觉任务不同，例如分段，分组和识别。这些通常由孤立的视觉模块执行，其仅寻求解释图像的部分。图像解析方法使这些不同的模块能够协作和竞争，以对整个图像进行一致的解释。



a. Input image　b. Segmentation　c. Synthesized image　d. Manual segmentation

图 8.26: DDMCMC 的图像分割失败示例，它仅使用通用视觉模式（即仅使用低级视觉提示）。结果（b）表明，低级视觉线索不足以获得良好的直观分割。仅使用通用视觉模式的限制在合成图像（c）中也是清楚的，其在通过 DDMCMC 估计参数之后通过生成模型的随机采样获得。右侧面板（d）显示了人类受试者获得的分割，与算法相比，他们在进行分割时似乎使用了对象特定的知识（尽管他们没有被指示）[40]。我们得出结论，要在这些类型的图像上实现良好的分割，需要将分割与对象检测和识别相结合。Courtesy of Tu et al. [56].

但是，组合不同的可视模块需要一个确保一致性的通用框架尽管用于计算场景组件（例如对象标签和类别）的判别方法的有效性，它们也可以生成冗余且冲突的结果。数学家认为 [6] 歧视方法必须遵循更复杂的过程，以（i）消除错误警报，（ii）通过全局背景信息修正缺失的对象，以及（iii）通过模型比较协调冲突（重叠）解释。在本节中，我们通过使用整个图像的生成模型来强制执行此类过程。

正如我们将要展示的那样，图像解析算法能够整合判别和生成方法，以便利用它们的互补优势。此外，诸如分割和对象检测的模块可以通过选择用于解析图像的一组视觉图案来耦合。在本节中，我们将重点放在两种类型的模式上: – 低/中等水平视觉的通用视觉模式，如纹理和阴影，以及高级视觉的对象模式，如正面人脸和文本。

这两种类型的模式说明了构造解析图的不同方式（参见图 8.40和相关讨论）。对象模式（面部和文本）具有相对较小的可变性，因此它们通常可以通过自下而上的测试作为整体有效地检测，并且它们的部分可以按顺序定位。因此，它们的解析子图可以从整体到部分以“分解”方式构造。相比之下，通用纹理区域具有任意形状，并且其强度图案具有高熵。通过自下而上的测试来检测这样的区域将需要大量的测试来处理所有这些可变性，因此在计算上是不切实际的。相反，解析子图应该通过以“组合”方式对小元素进行分组来构建 [5]。

我们在复杂城市场景的自然图像上说明了图像解析算法，并给出了通过允许对象特定知识消除低级别线索歧义来改善图像分割的示例，相反，通过使用通用视觉模式来解释阴影，可以改善对象检测和遮挡。

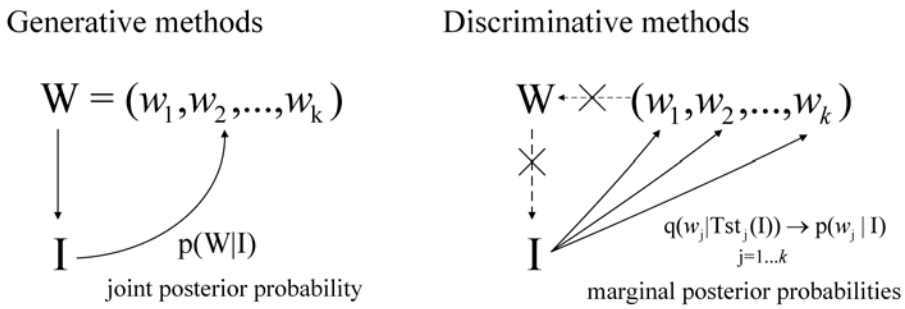


图 8.27: 两种推理范式的比较：自上而下的生成方法与自下而上的判别方法。生成方法指定如何从场景表示 W 合成图像 $\mathbf{I}$. 另一方面，辨别方法通过执行测试 $Tst_j(\mathbf{I})$ 来工作并且不保证产生一致的解决方案。Crosses are explained in the text.

8.10.1 自下而上和自上而下的处理

DDMCMC 的一个主要元素是整合判别和生成方法进行推理。自上而下和自下而上的程序可以大致分为两种流行的推理范式 – 自上而下的生成方法和自下而上的判别方法，如图 8.27所示。从这个角度来看，整合生成模型和判别模型相当于组合自下而上和自上而下的处理。

自下而上和自上而下处理在视觉中的作用已经得到了极大的讨论。越来越多的证据（ [35, 54]）表明人类可以像低水平纹理辨别和其他所谓的预注意视觉任务那样快速地执行高级场景和对象分类任务。这表明人类可以在视觉处理的早期阶段检测到低水平和高水平的视觉模式。这与传统的自下而上前馈架构 [39] 形成对比，传统的自下而上前馈架构从边缘检测开始，然后是分段/分组，然后再进行对象识别和其他高级视觉任务。这些实验还涉及关于自下而上/自上而下环在视觉皮层区 [44, 61] 中的作用的长期猜想，视觉例程和途径 [60]，视觉线索的结合 [55]，以及神经网络模型，如亥姆霍兹机器 [16]。DDMCMC 通过使用判别方法来统一这两种方法，以快速推断生成模型的参数。从计算机视觉的角度来看，DDMCMC 结合了自下而上的处理，由判别模型实现，以及生成模型的自上而下处理。本节的其余部分提供了有关此算法功能的更多详细信息。

8.10.2 生成和判别方法

生成方法指定如何从场景表示 $W \in \Omega$ 生成图像 $\mathbf{I}$. 它结合先验 $p(W)$ 和似然函数 $p(\mathbf{I}|W)$ 来给出联合后验概率 $p(W|\mathbf{I})$。这些可以表示为图上的概率，其中输入图像 $\mathbf{I}$ 在叶节点上表示，W 表示图的剩余节点和节点属性。图的结构，例如节点的数量是未知的，并且必须针对每个输入图像估计。可以通过来自后验的随机抽样 W 来执行推断

$$W \sim p(W|\mathbf{I}) \propto p(\mathbf{I}|W)p(W). \tag{8.33}$$

这使我们能够估计 $W^* = \arg\max P(W|\mathbf{I})$.[7] 但样本空间 Ω 的维数非常高，因此标准采样技术的计算成本很高。

相比之下，判别方法很容易计算。它们没有指定图像生成方式。相反，它们基于在图像上执行的自下而上测试序列 $\mathrm{Tst}_j(\mathbf{I})$ 给出 W 的分量 $\{w_j\}$ 的判别（条件）概率 $q(w_j|\mathrm{Tst}_j(\mathbf{I}))$。测试基于局部图像特征

[7] We are assuming that there are no known algorithms for estimating W^* directly.

$\{F_{j,n}(\mathbf{I})\}$，可以从图像中以级联方式计算（例如 AdaBoost 过滤器，参见章节 (8.10.5)），

$$\mathrm{Tst}_j(\mathbf{I}) = (F_{j,1}(\mathbf{I}), F_{j,2}(\mathbf{I}), ..., F_{j,n}(\mathbf{I})), \quad j = 1, 2, ..., K. \tag{8.34}$$

以下定理表明，使用测试 $\mathrm{Tst}(\mathbf{I})$ 的真实边际后验 $p(w_j|\mathbf{I})$ 和最佳判别近似 $q(w_j|\mathrm{Tst}(\mathbf{I}))$ 之间的 KL-发散将随着新测试的增加[8]而单调减小。

Theorem 8.1 *通过新测试获得的变量 w 的信息 $\mathrm{Tst}_+(\mathbf{I})$ 是 $p(w|\mathbf{I})$ 与其最佳判别估计 $q(w|\mathrm{Tst}_t(\mathbf{I}))$ 之间的 Kullback-Leibler 散度的减少或者增加 w 和测试之间的互信息。*

$$\begin{aligned} & E_{\mathbf{I}}[KL(p(w|\mathbf{I}) \,||\, q(w|\mathrm{Tst}(\mathbf{I})))] - E_{\mathbf{I}}[KL(p(w|\mathbf{I}) \,||\, q(w|\mathrm{Tst}(\mathbf{I}), \mathrm{Tst}_+(\mathbf{I})))] \\ = \quad & MI(w \,||\, \mathrm{Tst}, \mathrm{Tst}_+) - MI(w \,||\, \mathrm{Tst}) = E_{\mathrm{Tst},\mathrm{Tst}_+} KL(q(w \,|\, \mathrm{Tst}_t, \mathrm{Tst}_+) \,||\, q(w \,|\, \mathrm{Tst}_t) \geq 0, \end{aligned}$$

其中 $E_{\mathbf{I}}$ 是关于 $P(\mathbf{I})$ 的期望，并且 $E_{\mathrm{Tst},\mathrm{Tst}_+}$ 是关于由 $P(\mathbf{I})$ 诱导的测试响应 $(\mathrm{Tst}, \mathrm{Tst}_+)$ 的概率的期望。当且仅当 $\mathrm{Tst}(\mathbf{I})$ 相对于 w 是足够的统计量时，Kullback-Leibler 散度的减小等于零。

在实践中，辨别方法，特别是标准计算机视觉算法 - 参见第 (8.10.4) 小节，通常仅使用少量特征来计算实用性。鉴别概率 $q(w_j|\mathrm{Tst}(\mathbf{I}))$ 通常也不是最优的。幸运的是，本节中的图像解析算法仅要求判别概率 $q(w_j|\mathrm{Tst}(\mathbf{I}))$ 是对 $p(w_j|\mathbf{I})$ 的粗略近似。

判别模型和生成模型之间的区别如图 8.27所示。判别模型易于计算，并且可以并行运行，因为不同的组件是独立计算的（参见图 8.27中的箭头）。但是组件 $\{w_i\}$ 可能不会产生一致的解 $\mathbf{W}$，而且，W 可能没有指定用于生成观察图像 $\mathbf{I}$ 的一致模型。这些不一致性由图 8.27中的十字表示。生成模型确保一致性，但需要解决困难的推理问题。是否可以设计判别模型来推断我们正在处理的复杂生成模型的整个状态 W，这是一个悬而未决的问题。数学家 [6] 认为这不实用，而且歧视性模型总是需要额外的后处理。

8.10.3 马尔可夫链内核和子内核

形式上，DDMCMC 图像解析算法模拟马尔可夫链 $\mathcal{MC} = < \Omega, \nu, \mathcal{K} >$，其中核 $\mathcal{K}$ 在空间 Ω 中并且具有用于起始状态的概率 ν。$W \in \Omega$ 是一个解析图，我们让解析图集 Ω 是有限的，因为图像具有有限像素和有限灰度级。我们继续定义一组用于重新配置图形的移动。其中包括创建节点，删除节点和更改节点属性的移动。我们根据转换内核[9]指定这些移动的随机动态。

对于每次移动，我们通过转移矩阵 $\mathcal{K}_a(W'|W:\mathbf{I})$ 定义马尔可夫链子内核，其中 $a \in \mathcal{A}$ 是索引。这表示当应用子内核 a 时系统从状态 W 转换到状态 W' 的概率（即，$\sum_{W'} \mathcal{K}_a(W'|W:\mathbf{I}) = 1, \forall\, W$）。改变图结构的内核被分组为可逆对。例如，用于节点创建的子内核 $\mathcal{K}_{a,r}(W'|W:\mathbf{I})$ 与用于节点删除的子内核 $\mathcal{K}_{a,l}(W'|W:\mathbf{I})$ 配对。这可以组合成成对的子核 $\mathcal{K}_a = \rho_{ar}\mathcal{K}_{a,r}(W'|W:\mathbf{I}) + \rho_{al}\mathcal{K}_{a,l}(W'|W:\mathbf{I})$ ($\rho_{ar} + \rho_{al} = 1$。该配对确保 $\mathcal{K}_a(W'|W:\mathbf{I}) = 0$，并且仅当 $\mathcal{K}_a(W|W':\mathbf{I}) = 0$ 时对于所有状态 $W, W' \in \Omega$。构造子核（配对后）以遵守详细的平衡方程

$$p(W|\mathbf{I})\mathcal{K}_a(W'|W:\mathbf{I}) = p(W'|\mathbf{I})\mathcal{K}_a(W|W':\mathbf{I}). \tag{8.35}$$

[8]The optimal approximation occurs when $q(w_j|\mathrm{Tst}(\mathbf{I}))$ equals the probability $p(w_j|\mathrm{Tst}(\mathbf{I}))$ induced by $P(\mathbf{I}|W)P(W)$.

[9]We choose stochastic dynamics because the Markov chain probability is guaranteed to converge to the posterior $P(W|\mathbf{I})$. The complexity of the problem means that deterministic algorithms for implementing these moves risk getting stuck in local minima.

完整转换内核表示为

$$\mathcal{K}(W'|W:\mathbf{I}) = \sum_a \rho(a:\mathbf{I})\mathcal{K}_a(W'|W:\mathbf{I}),\ \sum_a \rho(a:\mathbf{I}) = 1,\ \rho(a:\mathbf{I}) > 0. \tag{8.36}$$

为了使用该内核，在每个时间步骤，算法选择概率为 $\rho(a:\mathbf{I})$ 的移动 a，然后使用内核 $\mathcal{K}_a(W'|W;\mathbf{I})$ 来选择从状态 W 到状态 W' 的转换。注意，概率 $\rho(a:\mathbf{I})$ 和 $\mathcal{K}_a(W'|W;\mathbf{I})$ 都取决于输入图像 $\mathbf{I}$. 这将 DDMCMC 方法与传统的 MCMC 计算区分开来 [9, 36]。

完整的内核遵循详细的平衡（等式 (8.35)），因为每个子内核都有。如果移动组足够（即，使得我们可以使用这些移动在任何两个状态 $W, W' \in \Omega$ 之间转换），它也将是遍历的。这两个条件确保 $p(W|\mathbf{I})$ 是有限空间 Ω 中马氏链 [9] 的不变（目标）概率。应用核 $\mathcal{K}_{a(t)}$ 将步骤 t 处的马尔可夫链状态概率 $\mu_t(W)$ 更新为 $t+1$ 处的 $\mu_{t+1}(W')$ ，

$$\mu_{t+1}(W') = \sum_W \mathcal{K}_{a(t)}(W'|W:\mathbf{I})\mu_t(W). \tag{8.37}$$

总之，DDMCMC 图像解析器模拟具有唯一不变概率 $p(W|\mathbf{I})$ 的马尔可夫链 $\mathcal{MC}$。在时间 t，马尔可夫链状态（即，解析图）W 遵循概率 μ_t，概率 t 是在时间 t 之前选择的子内核的乘积，

$$W \sim \mu_t(W) = \nu(W_o)\cdot[\mathcal{K}_{a(1)} \circ \mathcal{K}_{a(2)} \circ \cdots \circ \mathcal{K}_{a(t)}](W_o, W) \longrightarrow p(W|\mathbf{I}), \tag{8.38}$$

其中 $a(t)$ 索引在时间 t 选择的子内核。随着时间 t 的增加，$\mu_t(W)$ 以几何速率 [19] 单调地接近后验 $p(W|\mathbf{I})$ [9]。以下收敛定理对于图像解析很有用，因为它有助于量化不同子内核的有效性。

Theorem 8.2 *当应用子核 $\mathcal{K}_{a(t)}, \forall\, a(t) \in \mathcal{A}$ 时，后 $p(W|\mathbf{I})$ 和马尔可夫链状态概率之间的 Kullback-Leibler 散度单调减小，*

$$KL(p(W|\mathbf{I})\,||\,\mu_t(W)) - KL(p(W|\mathbf{I})\,||\,\mu_{t+1}(W)) \geq 0. \tag{8.39}$$

KL-发散的减小是严格正向的，并且仅在马尔可夫链变为静止之后等于零，即 $\mu = p$。

这个结果的证明可以在 [56] 中找到。该定理与热力学第二定律有关 [15]，其证明利用了详细的平衡方程（8.35）。这个 *KL* 分歧给出了每个子内核 $\mathcal{K}_{a(t)}$ 的“功率”的度量，因此它提出了在每个时间步骤选择子内核的有效机制。相比之下，经典的收敛性分析表明马尔可夫链的收敛速度呈指数级增长，但没有给出子核的功率测量。

8.10.4 DDMCMC 和提案概率

我们现在描述如何使用提议概率和判别模型来设计子内核。这是 DDMCMC 的核心。每个子内核[10]被设计成具有 Metropolis-Hastings [29, 42] 的形式

$$\mathcal{K}_a(W'|W:\mathbf{I}) = Q_a(W'|W:\mathrm{Tst}_a(\mathbf{I}))\min\{1, \frac{p(W'|\mathbf{I})Q_a(W|W':\mathrm{Tst}_a(\mathbf{I}))}{p(W|\mathbf{I})Q_a(W'|W:\mathrm{Tst}_a(\mathbf{I}))}\},\ W' \neq W, \tag{8.40}$$

[10]Except for one that evolves region boundaries.

通过提议概率 $Q_a(W'|W:\text{Tst}_a(\mathbf{I}))$ 提出（随机地）从 W 到 W' 的转换并且通过接受概率接受（随机地）

$$\alpha(W'|W:\mathbf{I}) = \min\{1, \frac{p(W'|\mathbf{I})Q_a(W|W':\text{Tst}_a(\mathbf{I}))}{p(W|\mathbf{I})Q_a(W'|W:\text{Tst}_a(\mathbf{I}))}\}. \tag{8.41}$$

Metropolis-Hastings 确保子内核遵守详细的平衡（配对后）。

提议概率 $Q_a(W'|W:\text{Tst}_a(\mathbf{I}))$ 是在 W 和 W' 之间的移动中改变的各个元素 w_j 的判别概率 $q(w_j|\text{Tst}_j(\mathbf{I}))$ 的乘积。$\text{Tst}_a(\mathbf{I})$ 是在提议概率 $Q_a(W'|W:\text{Tst}_a(\mathbf{I}))$ 和 $Q_a(W|W':\text{Tst}_a(\mathbf{I}))$ 中使用的自下而上测试的集合。提议概率必须易于计算（因为它们应该针对子内核 a 可以达到的所有可能状态 W' 进行评估）并且应该建议转换到状态 W'，其中后验 $p(W'|\mathbf{I})$ 可能是高。由于它们依赖于 $p(W'|\mathbf{I})$，接受概率在计算上更加昂贵，但是它们仅需要针对所提出的状态进行评估。

提案的设计涉及权衡。理想情况下，提案将从后验 $p(W'|\mathbf{I})$ 中采样，但这是不切实际的。相反，权衡需要在快速计算和具有高后验概率的状态的大运动之间进行选择。更正式地，我们将范围 $\Omega_a(W) = \{W' \in \Omega : \mathcal{K}_a(W'|W:\mathbf{I}) > 0\}$ 定义为可以使用一次子内核 a 从 W 到达的状态集。我们希望范围 $S_a(W)$ 很大，以便我们可以在每个时间步进行大的移动（即跳转到解决方案，而不是爬行）。如果可能，范围还应包括具有高后验 $p(W'|\mathbf{I})$ 的状态 W'（即，范围也应该在 Ω 的右侧部分）。

应该选择提议 $Q_a(W'|W:\text{Tst}_a(\mathbf{I}))$ 以便近似

$$\frac{p(W'|\mathbf{I})}{\sum_{W''\in\Omega_a(W)} p(W''|\mathbf{I})} \text{ if } W' \in \Omega_a(W), \text{ or 0 otherwise.} \tag{8.42}$$

它们将是 W' 的分量和当前状态 W 的生成模型的判别模型的函数（因为评估当前状态的生成模型在计算上是便宜的）。模型 $p(W|\mathbf{I})$ 的细节将决定提案的形式以及我们在保持提案易于计算和能够近似等式 (8.42) 的同时制定范围的程度。请参阅第 (8.10.5) 节中给出的详细示例。

本概述简要介绍了 DDMCMC。我们参考 [59] 从 MCMC 的角度对这些问题进行更复杂的讨论。

生成模型和贝叶斯公式

本节描述了解析图和用于图像解析算法的生成模型。

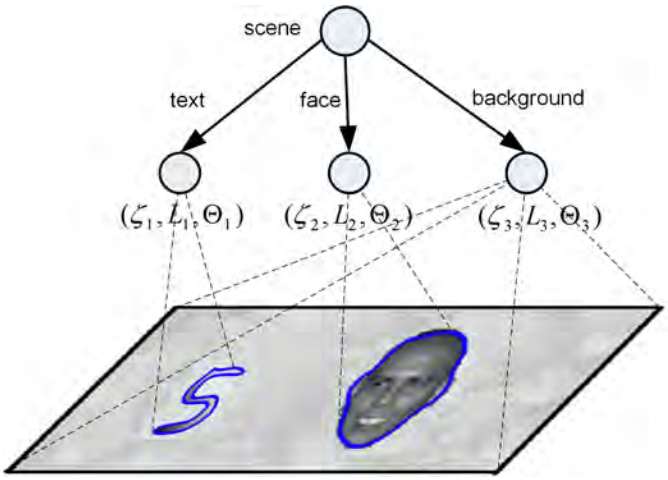


图 8.28: 本节中使用的解析图的抽象表示。中间节点表示视觉模式。它们的子节点对应于图像中的像素。Courtesy of Tu et al. [56].

图 8.25 说明了解析图的一般结构。在本节中，我们考虑图 8.28中所示的简化的两层图，它在生成意义上完全指定。图的顶部节点（“根”）表示整个场景（带有标签）。它具有用于视觉图案（面部，文本，纹理和阴影）的 K 个中间节点。每个视觉图案底部都有许多像素（“叶子”）。在该图中，除了它们共享边界并形成图像点阵的分区之外，在视觉图案之间不考虑水平连接。

中间节点的数量 K 是随机变量，并且每个节点 $i = 1,...,K$ 具有如下定义的一组属性 (L_i, ζ_i, Θ_i)。L_i 是形状描述符，并且确定由中间节点的视觉图案覆盖的图像像素的区域 $R_i = R(L_i)$。从概念上讲，R_i 内的像素是中间节点 i 的子节点。（区域可能包含孔，在这种情况下，形状描述符将具有内部和外部边界）。剩余的属性变量 (ζ_i, Θ_i) 指定用于在区域 $R(L_i)$ 中生成子图像 $\mathbf{I}_{R(L_i)}$ 的概率模型 $p(\mathbf{I}_{R(L_i)}|\zeta_i, L_i, \Theta_i)$。变量 $\zeta_i \in \{1,...,66\}$ 表示视觉模式类型（3 种通用视觉模式，1 种面部模式和 62 种文本字符模式），Θ_i 表示模型相应视觉模式的参数。完整的场景描述可以概括为：

$$W = (K, \{(\zeta_i, L_i, \Theta_i) : i = 1,2,...,K\}).$$

形状描述符 $\{L_i : i = 1,...,K\}$ 需要是一致的，以便图像中的每个像素是一个且仅有一个中间节点的子像素。形状描述符必须提供图像点阵的分区 $\Lambda = \{(m,n) : 1 \le m \le Height(\mathbf{I}), 1 \le n \le Width(\mathbf{I})\}$ 因此满足条件

$$\Lambda = \cup_{i=1}^{K} R(L_i), \quad R(L_i) \cap R(L_j) = \emptyset, \ \ \forall i \neq j.$$

从场景描述 W 到 $\mathbf{I}$ 的生成过程由似然函数控制

$$p(\mathbf{I}|W) = \prod_{i=1}^{K} p(\mathbf{I}_{R(L_i)}|\zeta_i, L_i, \Theta_i).$$

先验概率 $p(W)$ 由下式定义

$$p(W) = p(K) \prod_{i=1}^{K} p(L_i) p(\zeta_i|L_i) p(\Theta_i|\zeta_i).$$

在贝叶斯公式下，解析图像对应于计算 W^*，其最大化 a 后验概率超过 Ω，即 W 的解空间，

$$W^* = \arg\max_{W \in \Omega} p(W|\mathbf{I}) = \arg\max_{W \in \Omega} p(\mathbf{I}|W) p(W). \tag{8.43}$$

它主要规定了先验 $p(W)$ 和似然函数 $p(\mathbf{I}|W)$。将初始值 $p(K)$ 和 $p(\Theta_i|\zeta_i)$ 设为均匀概率。术语 $p(\zeta_i|L_i)$ 用于惩罚高模型复杂度，并且根据 [58] 中的训练数据估计三种通用视觉模式。

形状模型

我们使用两种形状描述符。第一个用于定义通用视觉图案和面部的形状。第二个定义文本字符的形状。在第一种情况下，形状描述符通过像素列表 $L_i = \partial R_i$ 表示图像区域的边界[11] 。先验定义为

$$p(L_i) \propto \exp\{-\gamma |R(L_i)|^{\alpha} - \lambda |L_i|\}. \tag{8.44}$$

通常 $\alpha = 0.9$. 出于计算原因，我们将此先验用于面部形状，即使可以应用更复杂的先验 [14]。

在文本字符的情况下，我们通过对应于十个数字的六十二个模板和在大写和小写的情况下的二十六个字母对字符进行建模。这些模板由六十二个原型字符和一组变形定义。原型由外边界表示，最多由两个内边界表示。每个边界由 B 样条建模，使用 25 个控制点。原型字符由 $c_i \in \{1,...,62\}$ 索引，并且它们的控制点由矩阵 $TP(c_i)$ 表示。

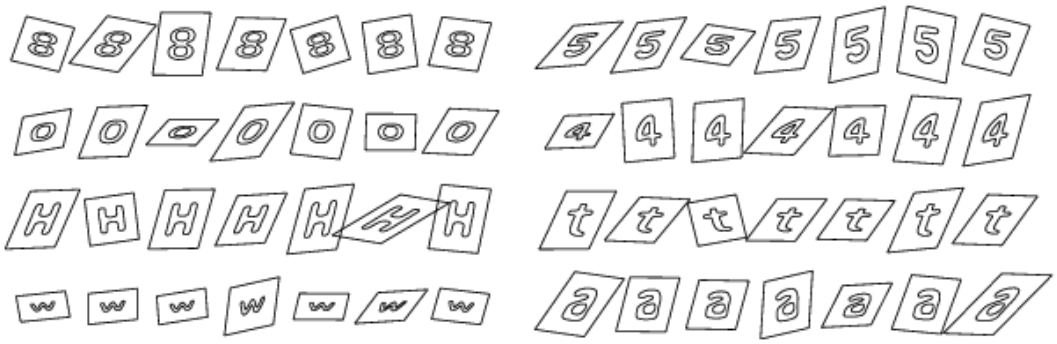


图 8.29: 从文本字符的形状描述符中抽取的随机样本。Courtesy of Tu et al. [56].

我们现在在模板上定义两种类型的变形。一个是全局仿射变换，另一个是局部弹性变形。首先，我们允许字母通过仿射变换 M_i 变形。我们先放置一个 $p(M_i)$ 来惩罚严重的旋转和扭曲。这是通过将 M_i 分解为得到的

$$M_i = \begin{pmatrix} \sigma_x & 0 \\ 0 & \sigma_y \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix},$$

其中 θ 是旋转角度，σ_x and σ_y 表示缩放，h 表示剪切。先前的 M_i 是

$$p(M_i) \propto \exp\{-a|\theta|^2 + b(\frac{\sigma_x}{\sigma_y} + \frac{\sigma_y}{\sigma_x})^2 + ch^2\},$$

其中 $a, b,$ 和 c 是参数。

接下来，我们通过调整 B 样条控制点的位置来允许局部变形。对于数字/字母 c_i 和仿射变换 M_i，模板的轮廓点由 $G_{TP}(M_i, c_i) = U \times M_s \times M_i \times TP(c_i)$ 给出。类似地，具有控制点 S_i 的形状上的轮廓点由 $G_S(M_i, c_i) = U \times M_s \times S_i$（$U$ 和 M_s 是 B 样条矩阵）给出。我们定义了由 S_i 给出的弹性变形的概率分布 $p(S_i|M_i, c_i)$

$$p(S_i|M_i, c_i) \propto \exp\{-\gamma |R(L_i)|^{\alpha} - D(G_S(M_i, c_i)||G_{TP}(M_i, c_i))\},$$

[11]The boundary can include an “internal boundary” if there is a hole inside the image region explained by a different visual pattern.

其中 $D(G_S(M_i,c_i)||G_{TP}(M_i,c_i))$ 是轮廓模板和变形轮廓之间的总距离。这些变形很小，因此曲线上的点之间的对应关系可以通过最近邻匹配来获得。参见 [57] 我们如何改进这一点。图 8.29显示了从上述模型中抽取的一些样本。

总之，每个可变形模板由 $c_i \in \{1,...,62\}$ 索引并具有形状描述符 $L_i = (c_i, M_i, S_i)$，其中 L_i 上的先验分布由 $p(L_i) = p(c_i)p(M_i)p(S_i|M_i,c_i)$ 指定。这里，$p(c_i)$ 是所有数字和字母的均匀分布（我们不会在文本字符串上放置先验分布，尽管可以这样做 [30]）。

生成强度模型

我们使用四个生成强度模型族来描述具有（近似）恒定强度，杂波/纹理，阴影和面部的模式。前三个特征类似于 [58] 和本章前面定义的特征。

1. 恒定强度模型 $\zeta = 1$.

该模型假设区域 R 中的像素强度是 i.i.d. 高斯分布由下式给出

$$p_1(\mathbf{I}_{R(L)}|\zeta = 1, L, \Theta) = \prod_{v \in R(L)} G(\mathbf{I}_v - \mu; \sigma^2), \quad \Theta = (\mu, \sigma).$$

2. 杂波/纹理模型 $\zeta = 2$.

这是非参数强度直方图 $h()$ 离散化以取 G 值（即，它表示为矢量 $(h_1, h_2, ..., h_G)$）。设 n_j 为强度值 j 的 $R(L)$ 中的像素数

$$p_2(\mathbf{I}_{R(L)}|\zeta = 2, L, \Theta) = \prod_{v \in R(L)} h(\mathbf{I}_v) = \prod_{j=1}^{G} h_j^{n_j}, \quad \Theta = (h_1, h_2, ..., h_G).$$

3. 着色模型 $\zeta = 3$ 和 $\zeta = 5, ..., 66$.

该系列模型用于描述通用着色模式和文本字符。我们使用二次型

$$J(x, y; \Theta) = ax^2 + bxy + cy^2 + dx + ey + f,$$

参数 $\Theta = (a, b, c, d, e, f, \sigma)$. 因此，像素 (x, y) 的生成模型是

$$p_3(\mathbf{I}_R(L)|\zeta \in \{3, (5, ..., 66)\}, L, \Theta) = \prod_{v \in R(L)} G(\mathbf{I}_v - J_v; \sigma^2), \quad \Theta = (a, b, c, d, e, f, \sigma).$$

4. PCA 面部模型 $\zeta = 4$.

面部的生成模型很简单，并使用主成分分析（PCA）来获得面部的主成分 $\{B_i\}$ 和协方差 Σ 的表示。也可以添加由 PCA 建模的低级功能 [43]。我们还可以添加其他功能，例如 Hallinan 等 [28] 中描述的功能。然后给出模型

$$p_4(\mathbf{I}_R(L)|\zeta = 4, L, \Theta) = G(\mathbf{I}_{R(L)} - \sum_i \lambda_i B_i; \Sigma), \quad \Theta = (\lambda_1, .., \lambda_n, \Sigma).$$

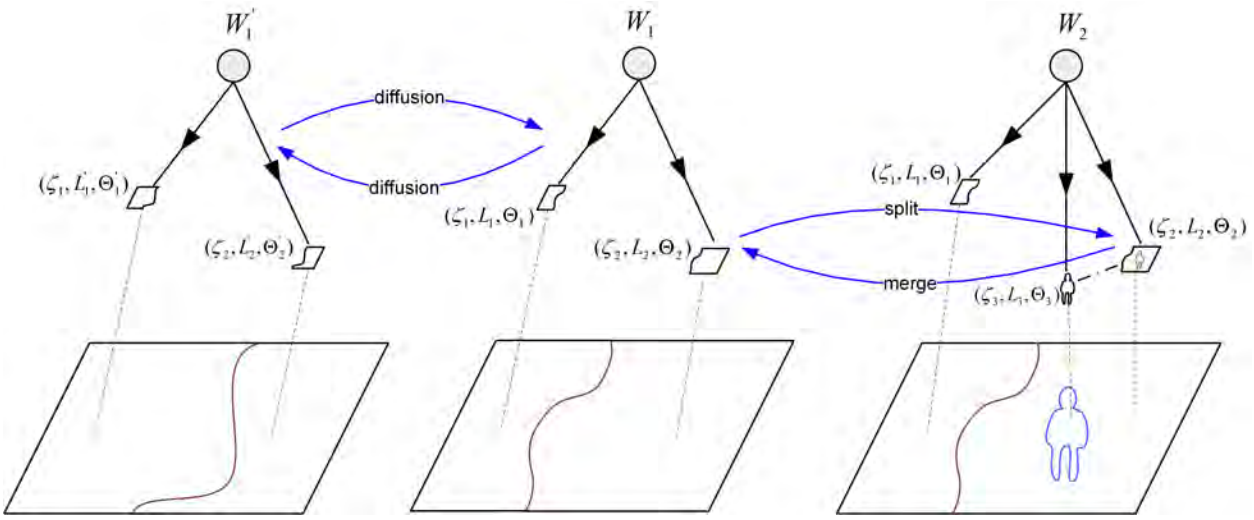


图 8.30: 马尔可夫链动力学的例子，它改变了图形结构或图形的节点属性，从而产生了解析图像的不同方法。Courtesy of Tu et al. [56].

算法概述

本节给出了图像解析算法的控制结构。图 8.31显示了算法图。我们的算法必须在运行中构造解析图并估计场景解释 W.

图 8.30说明了算法如何通过改变图形结构（通过删除或添加节点）以及通过更改节点属性来选择马尔可夫链移动（动态或子内核）来搜索图像的可能解析图的空间。可视化算法的等效方式是通过解空间 Ω 的搜索。有关此观点的更多详细信息，请参见 [58, 59]。

我们首先定义一组动作以重新配置图形。这些是面部节点的诞生或死亡，文本字符的诞生或死亡，区域的分裂或合并，切换节点属性（区域类型 ζ_i 和模型参数 Θ_i）和边界演化（用节点改变节点的形状描述符 L_i）邻近地区）。这些移动由子内核实现。前四个动作是可逆跳跃 [26]，并将由 Metropolis-Hastings 等式 (8.40) 实现。第五步，边界演化，由随机偏微分方程实现。

这些移动的子内核需要由基本判别方法驱动的提议概率，我们将在下一小节中进行讨论。提案概率是使用第 (8.10.4) 小节中的标准设计的，完整的细节在第 (8.10.5) 节中给出。该算法的控制结构在 (8.10.4) 节中描述。如第 (8.10.3) 小节和图 8.31所述，通过组合子内核构建图像解析器的完整转换内核。该算法（随机地）通过选择子内核，选择图中的应用位置，然后决定是否接受该操作来进行（随机）。

判别方法

判别方法给出 W 的基本分量 w_j 的近似后验概率 $q(w_j|\text{Tst}_j(\mathbf{I}))$。对于计算效率，这些概率仅基于少量简单测试 $\text{Tst}_j(\mathbf{I})$。我们简要概述和分类此实现中使用的判别方法。第（ (8.10.5) 节）部分显示了如何将这些判别方法结合起来给出在解析图中进行移动的建议。

1. 边缘线索. 这些基于边缘检测器 [10],[8],[32]。它们用于给出区域边界的提议（即节点的形状描述符属性）。具体来说，我们以三个刻度运行 Canny 检测器，然后进行边缘链接以给出图像点阵的分区。这给出了分配权重的候选分区的有限列表，参见 [58]。辨别概率由该加权粒子列表表示。原则上，统计边缘检测器 [32] 优于 Canny，因为它们给出了从训练数据中学习的判别概率 $q(w_j|\text{Tst}_j(\mathbf{I}))$。

2. 二值化线索. 这些是使用 Niblack 算法 [45] 的变体计算的。它们用于提出文本字符的边界（即文本节点的形状描述符），并与文本检测的提议结合使用。二值化算法及其输出示例在第 (8.10.5) 节中给出。与边缘提示一样，该算法针对不同的参数设置运行并表示辨别力通过指示候选边界位置的加权粒子列表的概率。

3. 面部区域提示. 这些是通过 AdaBoost [50],[62] 的变体学习的，它输出了判别概率 [23]。他们建议在图像的子区域中存在面部。这些提示与边缘检测相结合，以提出图像中面部的定位。

4. 文本区域提示. 这些也是由 AdaBoost 的概率版本学习的。该算法适用于各种尺度的图像窗口。它输出每个窗口中存在文本的判别概率。文本区域提示与二值化结合以提出文本字符的边界。

5. 形状亲和力提示. 这些作用于通过二值化产生的形状边界，以提出文本字符。他们使用形状上下文线索 [4] 和信息特征 [57] 来提出形状边界和文本字符的可变形模板模型之间的匹配。

6. 区域亲和线索. 这些用于估计两个区域 R_i, R_j 是否可能由相同的视觉图案族和模型参数生成。他们使用强度属性 $\mathbf{I}_{R_i}, \mathbf{I}_{R_j}$ 的亲和力相似性度量 [52]。

7. 模型参数和视觉模式族提示. 这些用于提出模型参数和视觉模式家族身份。它们基于聚类算法，例如均值漂移 [13]。聚类算法取决于模型类型，并在 [58] 中进行了描述。

通常自下而上的测试 $\mathrm{Tst}_j(\mathbf{I}), j = 1, 2, ..., K$ 在所有判别模型 $q_j(w_j|\mathrm{Tst}_j(\mathbf{I}))$ 的早期阶段进行。然后将结果组合以形成方程 (8.40,8.41) 中的每个子内核 $\mathcal{K}_a$ 的复合测试 $\mathrm{Tst}_a(\mathbf{I})$。

算法的控制结构

图像解析器使用的控制策略如图 8.31所示。图像解析器通过 MCMC 采样算法探索解析图的空间。该算法使用转换内核 K，转换内核 $\mathcal{K}$ 由子内核 $\mathcal{K}_a$ 组成，对应于不同的配置稀疏图。这些子核以可逆对[12]（例如出生和死亡）形式出现，并且被设计成使得核的目标概率分布是生成后验 $p(W|\mathbf{I})$。在每个时间步，随机选择子内核。子内核使用 Metropolis-Hasting 抽样算法，方程 (8.40)，分两个阶段进行。首先，它建议通过从提议概率中抽样来重新配置图。然后，它通过对接受概率进行抽样来接受或拒绝此重新配置。

总而言之，我们概述了下面的算法。在每个时间步，它（随机地）指定哪个移动到选择（即哪个子内核），在图中应用它的位置，以及是否接受移动。选择移动 $\rho(a:\mathbf{I})$ 的概率首先被设置为独立于 $\mathbf{I}$，但是通过使用判别线索来估计图像中的面部和文本字符的数量，我们获得了更好的性能。选择移动的位置由子内核指定。对于一些子内核，它是随机选择的，并且对于其他子内核是基于适应度因子来选择的，该适应度因子测量当前模型与图像数据的拟合程度。由于在该实施方式中移动的范围有限（如果使用 [2]) 中描述的组成技术）将减少退火的需要），则需要一些退火来启动算法。

我们可以通过使移动选择适应图像（即通过使 $\rho(a:\mathbf{I})$ 取决于 $\mathbf{I}$）来提高算法的有效性。特别是，如果自下而上（AdaBoost）建议表明场景中有很多物体，我们可以增加面部和文本的出生和死亡概率，$\rho(1)$ a 和 $\rho(2)$。例如，让 $N(\mathbf{I})$ 是高于阈值 T_a 的面部或文本的提议数量。然后我们通过 $\rho(a_1) \mapsto \{\rho(a_1) + kg(N(\mathbf{I}))\}/Z$, $\rho(a_2) \mapsto \{\rho(a_2) + kg(N)\}/Z$, $\rho(a_3) \mapsto \rho(a_3)/Z$, $\rho(a_4) \mapsto \rho(a_4)/Z$ 来修改表中的概率, 其中 $g(x) = x$, $x \leq T_b$ $g(x) = T_b$, $x \geq T_b$ 和 $Z = 1 + 2k$ 被选择来规范化概率。

图像解析算法的基本控制策略

[12]Except for the boundary evolution sub-kernel which will be described separately. See Section 8.10.5.

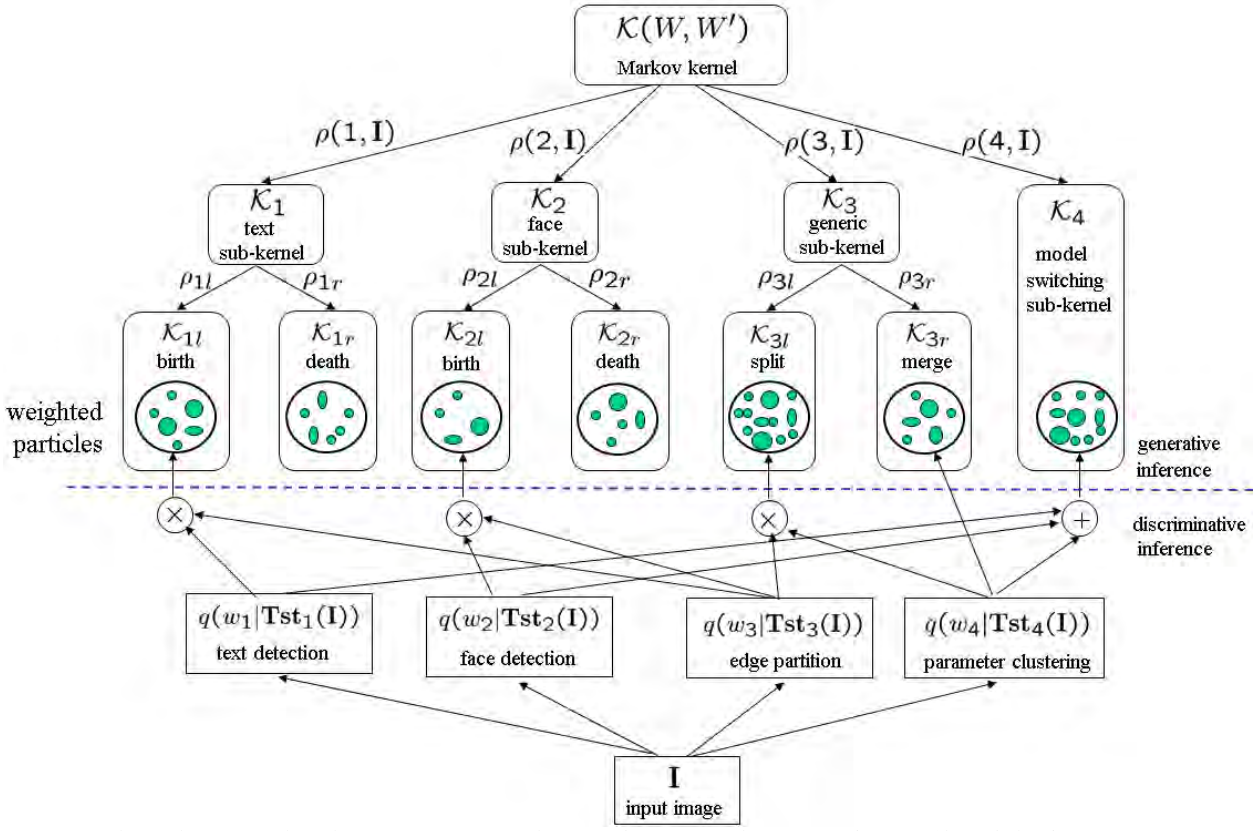


图 8.31: 此图说明了图像解析器的要点。动力学由遍历马尔科夫链 $\mathcal{K}$ 实现，其不变概率是后验 $p(W|\mathbf{I})$，并且由可逆子核 $\mathcal{K}_a$ 组成，用于在解析图中进行不同类型的移动。在每个时间步，算法随机选择子内核。所选择的子内核提出特定移动（例如，创建或删除特定节点），然后随机评估和接受该移动，参见等式 (8.40)。这些提案基于自下而上（歧视性）和自上而下（生成性）流程。自下而上的过程基于特征测试 $\text{Tst}_j(\mathbf{I})$ 从输入图像 $\mathbf{I}$ 计算判别概率 $q(w_j|\text{Tst}_j(\mathbf{I})), j = 1,2,3,4$ 。Courtesy of Tu et al. [56].

1. 初始化 W（例如，将图像划分为四个区域），设置其形状描述符，并随机分配剩余的节点属性。
2. 将温度设置为 T_{init}.
3. 通过从概率 $\rho(a)$ 中采样来选择移动 a 的类型，对于面，$\rho(1) = 0.2$，对于文本，$\rho(2) = 0.2$，对于分裂和合并，$\rho(3) = 0.4$，$\rho(4) = 0.15$ 用于切换区域模型（类型或模型参数），$\rho(5) = 0.05$ 用于边界演化。
4. 如果选定的移动是边界演变，则随机选择相邻区域（节点）并应用随机最速下降。
5. 如果选择了跳跃移动，则按如下方式随机采样新的解 W'：If the jump moves are selected, then a new solution W' is randomly sampled as follows:
   - 对于面部或角色的出生或死亡，我们创建或删除现有的角色。这包括有关在何处执行此操作的建议。
   - 对于区域划分，基于其适应度因子随机选择区域（节点）。有关于在何处拆分它以及生成的两个节点的属性的建议。
   - 对于区域合并，基于提议概率选择两个相邻区域（节点）。存在关于结果节点的属性的提议。

– 对于区域切换，根据其适应度因子随机选择区域，并提出新的区域类型和/或模型参数集。

- 计算完整的提议概率 $Q(W|W:\mathbf{I})$ 和 $Q(W'|W:\mathbf{I})$ 。
- Metropolis-Hastings 算法适用于接受或拒绝建议的移动。

6. 降低温度 $T = 1 + T_{init} \times \exp(-t \times c|R|)$, 其中 t 是当前迭代步骤，c 是常数且 $|R|$ 是图像的大小。

7. 重复上述步骤，直到通过达到允许步数的最大数量或通过负对数后验的减少来满足收敛标准。

8.10.5 马尔可夫链核

边界演化

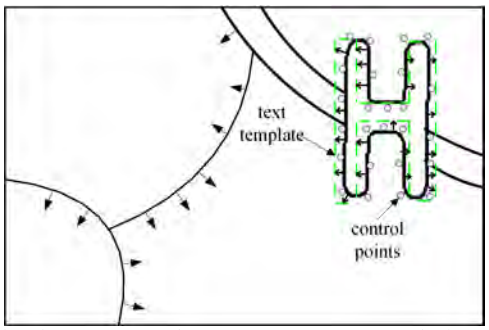


图 8.32: 区域边界的演化是由随机偏微分方程实现的，这些方程由竞争区域所有权的模型驱动。Courtesy of Tu et al. [56].

这些移动演变了区域边界的位置，但保留了图形结构。它们由布朗噪声驱动的随机偏微分方程（Langevin 方程）实现，可以从马尔可夫链中导出 [25]。PDE 的确定性分量是通过在负对数后验上进行最速下降而获得的，如 [68] 中所推导的。

我们通过导出 PDE 的确定性分量来说明这种方法，以用于字母 T_j 和通用视觉图案区域 $\mathbf{R}_i$ 之间的边界的演变。边界将以字母形状描述符的控制点 $\{S_m\}$ 表示。设 v 表示边界上的一个点，即对 $v(s) = (x(s), y(s))$ on $\Gamma(s) = \partial R_i \cap \partial R_j$。演化方程的确定性部分是通过相对于控制点取负 log-posterior，$-\log p(W|\mathbf{I})$ 的导数得到的。

负对数后验的相关部分由 $E(\mathbf{R}_i)$ 和 $E(T_j)$ 给出

$$E(\mathbf{R}_i) = \int\int_{R_i} -\log p(\mathbf{I}(x,y)|\theta_{\zeta_i})dxdy + \gamma|R_i|^{\alpha} + \lambda|\partial R_i|,$$

和

$$E(T_j) = \int\int_{L_j} \log p(\mathbf{I}(x,y)|\theta_{\zeta_j})dxdy + \gamma|R(L_j)|^{\alpha} - \log p(L_j).$$

相对于控制点 $\{S_m\}$ 区分 $E(R_i) + E(T_j)$ 产生进化 PDE

$$\frac{dS_m}{dt} = -\frac{\delta E(\mathbf{R}_i)}{\delta S_m} - \frac{\delta E(T_j)}{\delta S_m}$$

$$= \int [-\frac{\delta E(\mathbf{R}_i)}{\delta v} - \frac{\delta E(T_j)}{\delta v}] \frac{1}{|\mathbf{J}(s)|} ds$$
$$= \int \mathbf{n}(v)[\log \frac{p(\mathbf{I}(v);\theta_{\zeta_i})}{p(\mathbf{I}(v);\theta_{\zeta_j})} + \alpha\gamma(\frac{1}{|D_j|^{1-\alpha}} - \frac{1}{|D_i|^{1-\alpha}}) - \lambda\kappa + D(G_{S_j}(s)||G_T(s))] \frac{1}{|\mathbf{J}(s)|} ds,$$

其中 $\mathbf{J}(s)$ 是样条函数的雅可比矩阵。（回想一下，在实现中 $\alpha = 0.9$）。
对数似然比项实现了字母和通用区域模型之间对边界像素所有权的竞争。

马尔可夫链子内核

通过由四个不同子内核实现的马尔可夫链跳跃来实现图结构的变化。

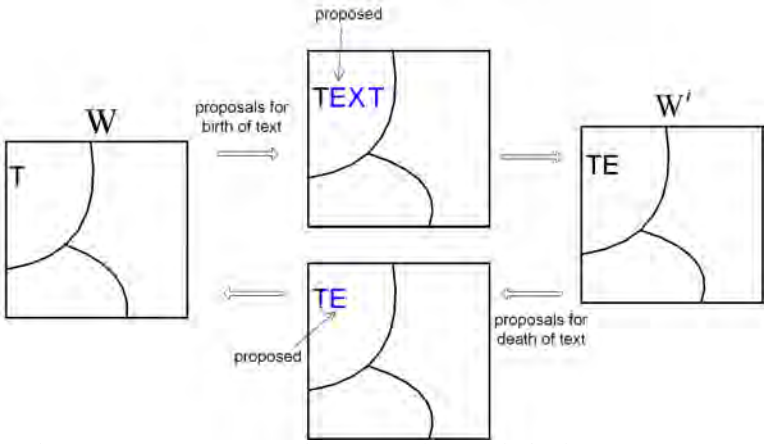


图 8.33: 文本生灭的一个例子。状态 W 由三个通用区域和一个字符“T”组成。通过 AdaBoost 和二值化方法获得的 3 个候选字符“E”，“X”和“T”的计算结果。选择一个，参见箭头，将状态更改为 W'。相反，在状态 W' 中有 2 个候选者，并且选择的一个，参见箭头，将系统返回到状态 W. Courtesy of Tu et al. [56].

子内核 I：文本的诞生和死亡。这对跳转用于创建或删除文本字符。我们从解析图 W 开始，并通过创建一个字符转换到解析图 W'。相反，我们通过删除一个字符从 W' 过渡到 W.

创建和删除文本字符的提议旨在接近等式 (8.42) 中的术语。我们通过使用 AdaBoost 检测文本区域，然后进行二值化以检测文本区域内的候选文本字符边界，获得候选文本字符形状列表。该列表由一组粒子表示，这些粒子通过与文本字符的可变形模板的相似性加权

$$S_{1r}(W) = \{ (z_{1r}^{(\mu)}, \omega_{1r}^{(\mu)}) : \mu = 1, 2, ..., N_{1r} \}.$$

同样，我们指定另一组加权粒子来删除文本字符

$$S_{1l}(W') = \{ (z_{1l}^{(\nu)}, \omega_{1l}^{(\nu)}) : \nu = 1, 2, ..., N_{1l} \}.$$

在这里，$\{z_{1r}^{(\mu)}\}$ 和 $\{z_{1l}^{(\nu)}\}$ 表示可能的（离散的）形状位置和文本字符可变形的模板，用于创建或删除文本，$\{\omega_{1r}^{(\mu)}\}$ 和 $\{\omega_{1l}^{(\nu)}\}$ 是他们相应的权重。然后使用粒子计算提议概率

$$\mathbf{Q_{1r}}(\mathbf{W}'|\mathbf{W}:\mathbf{I}) = \frac{\omega_{\mathbf{1r}}(\mathbf{W}')}{\sum_{\mu=1}^{\mathbf{N_{1r}}} \omega_{\mathbf{1r}}^{(\mu)}}, \quad \mathbf{Q_{1l}}(\mathbf{W}|\mathbf{W}', \mathbf{I}) = \frac{\omega_{\mathbf{1l}}(\mathbf{W})}{\sum_{\nu=1}^{\mathbf{N_{1l}}} \omega_{\mathbf{1l}}^{(\nu)}}.$$

用于创建新文本字符的权重 $\omega_{1r}^{(\mu)}$ 和 $\omega_{1l}^{(\nu)}$ 由形状亲和度测量指定，例如，形状上下文 [4] 和信息特征 [57]。为了删除文本字符，我们直接从中计算 $\omega_{1l}^{(\nu)}$ 文本字符的可能性和先验。理想情况下，这些权重将接近比率 $\frac{p(W'|\mathbf{I})}{p(W|\mathbf{I})}$ 和 $\frac{p(W|\mathbf{I})}{p(W'|\mathbf{I})}$.

子内核 II：面部的出生和死亡。面部出生和死亡的子内核非常类似于文本诞生和死亡的子内核。我们使用第 (8.10.5) 节中讨论的 AdaBoost 方法来检测候选面。候选面部边界直接从边缘检测获得。提议概率的计算类似于子内核 I 的概率。

子内核 III：区域的分裂和合并。这对跳转用于通过拆分和合并区域（节点）来创建或删除节点。我们从解析图 W 开始，并通过将节点 i 分成节点 j 和 k 来转换到解析图 W'。相反，我们通过将节点 j 和 k 合并到节点 i 中而转换回 W. 选择要分割的区域 i 是基于 $p(\mathbf{I}_{R_i}|\zeta_i,L_i,\Theta_i)$（即. 区域 R_i 的模型越适合数据，我们就越有可能将其拆分）上的鲁棒函数。对于合并，我们使用区域亲和度度量 [52] 并提出具有高亲和力的区域之间的合并。形式上，我们定义 W,W'：

$$W=(K,(\zeta_k,L_k,\Theta_k),W_-) \;\rightleftharpoons\; W'=(K+1,(\zeta_i,L_i,\Theta_i),(\zeta_j,L_j,\Theta_j),W_-),$$

其中 W_- 表示图中剩余 $K-1$ 个节点的属性。

我们通过如下求近似等式 (8.42) 来获得建议。我们首先获得三个边缘图。这些是由不同尺度的 Canny 边缘探测器 [10] 给出的（详见 [58]）。我们使用这些边缘图来创建用于分割 $S_{3r}(W)$ 的粒子列表。用于合并的粒子列表由 $S_{3l}(W')$ 表示。从形式上看，

$$S_{3r}(W)=\{\,(z_{3r}^{(\mu)},\omega_{3r}^{(\mu)}):\;\mu=1,2,...,N_{3r}\},\quad S_{3l}(W')=\{\,(z_{3l}^{(\nu)},\omega_{3l}^{(\nu)}):\;\nu=1,2,...,N_{3l},\}$$

其中 $\{z_{3r}^{(\mu)}\}$ 和 $\{z_{3l}^{(\nu)}\}$ 表示用于分裂和合并的可能（离散化）位置，并且将很快定义它们的权重 $\{\omega_{3r}\},\{\omega_{3l}\}$。换句话说，我们只能沿着轮廓 z_{3r}^{μ} 将区域 i 分割成区域 j 和 k（即，z_{3r}^{μ} 形成新的边界）。类似地，我们只能通过去除边界轮廓 z_{3l}^{μ} 将区域 j 和 k 合并到区域 i 中。

我们现在定义权重 $\{\omega_{3r}\},\{\omega_{3l}\}$。这些权重将用于确定拆分和合并的概率

$$\mathbf{Q_{3r}}(\mathbf{W}'|\mathbf{W}:\mathbf{I})=\frac{\omega_{\mathbf{3r}}(\mathbf{W}')}{\sum_{\mu=\mathbf{1}}^{\mathbf{N_{3r}}}\omega_{\mathbf{3r}}^{(\mu)}},\;\mathbf{Q_{3l}}(\mathbf{W}|\mathbf{W}':\mathbf{I})=\frac{\omega_{\mathbf{3l}}(\mathbf{W})}{\sum_{\nu=\mathbf{1}}^{\mathbf{N_{3l}}}\omega_{\mathbf{3l}}^{(\nu)}}.$$

同样，我们希望 ω_{3r}^{μ} 和 ω_{3l}^{ν} 来近似比率 $\frac{p(W|\mathbf{I})}{p(W'|\mathbf{I})}$ 和 $\frac{p(W'|\mathbf{I})}{p(W|\mathbf{I})}$。$\frac{p(W'|\mathbf{I})}{p(W|\mathbf{I})}$ 由下式给出

$$\frac{p(W'|\mathbf{I})}{p(W|\mathbf{I})}=\frac{p(\mathbf{I}_{R_i}|\zeta_i,L_i,\Theta_i)p(\mathbf{I}_{R_j}|\zeta_j,L_j,\Theta_j)}{p(\mathbf{I}_{R_k}|\zeta_k,L_k,\Theta_k)}\cdot\frac{p(\zeta_i,L_i,\Theta_i)p(\zeta_j,L_j,\Theta_j)}{p(\zeta_k,L_k,\Theta_k)}\cdot\frac{p(K+1)}{p(K)}.$$

计算起来很昂贵，所以我们用 $\frac{p(W'|\mathbf{I})}{p(W|\mathbf{I})}$ 和 $\frac{p(W|\mathbf{I})}{p(W'|\mathbf{I})}$ 来近似

$$\omega_{3r}^{(\mu)}=\frac{q(R_i,R_j)}{p(\mathbf{I}_{R_k}|\zeta_k,L_k,\Theta_k)}\cdot\frac{[q(L_i)q(\zeta_i,\Theta_i)][q(L_j)q(\zeta_j,\Theta_j)]}{p(\zeta_k,L_k,\Theta_k)},\tag{8.45}$$

$$\omega_{3l}^{(\nu)}=\frac{q(R_i,R_j)}{p(\mathbf{I}_{R_i}|\zeta_i,L_i,\Theta_i)p(\mathbf{I}_{R_j}|\zeta_j,L_j,\Theta_j)}\cdot\frac{q(L_k)q(\zeta_k,\Theta_k)}{p(\zeta_i,L_i,\Theta_i)p(\zeta_j,L_j,\Theta_j)}.\tag{8.46}$$

这里，$q(R_i,R_j)$ 是两个区域 R_i 和 R_j 的相似度的亲和度度量 [52]（它是强度差 $|\bar{I}_i-\bar{I}_j|$ 和 chi 的加权和。强度直方图之间的平方差），$q(L_i)$ 由形状描述符上的先验给出，$q(\zeta_i,\Theta_i)$ 通过参数空间中的聚类获得（见

[58]）。

子核 IV：模型切换。这些移动切换节点 i 的属性。这涉及改变区域类型 ζ_i 和模型参数 Θ_i。移动在两个状态之间转换

$$W = ((\zeta_i, L_i, \Theta_i), W_-) \rightleftharpoons W' = ((\zeta_i', L_i', \Theta_i'), W_-).$$

该提议，见方程式 (8.42)，应近似

$$\frac{p(W'|\mathbf{I})}{p(W|\mathbf{I})} = \frac{p(\mathbf{I}_{R_i}|\zeta_i', L_i', \Theta_i')p(\zeta_i', L_i', \Theta_i')}{p(\mathbf{I}_{R_i}|\zeta_i, L_i, \Theta_i)p(\zeta_i, L_i, \Theta_i)}.$$

我们通过权重 $\omega_4^{(\mu)}$ 来估计它

$$\omega_4^{(\mu)} = \frac{q(L_i')q(\zeta_i', \Theta_i')}{p(\mathbf{I}_{R_i}|\zeta_i, L_i, \Theta_i)p(\zeta_i, L_i, \Theta_i)},$$

其中 $q(L_i')q(\zeta_i', \Theta_i')$ 与分割和合并移动中使用的函数相同。提议概率是候选集中归一化的权重，$\mathbf{Q_4}(\mathbf{W}'|\mathbf{W}:\mathbf{I}) = \frac{\omega_4(\mathbf{W}')}{\sum_{\mu=1}^{\mathbf{N_4}} \omega_4^{(\mu)}}$.

AdaBoost 用于面部和文本的判别概率

本节描述了我们如何使用 AdaBoost 技术来计算用于检测面部和文本（字母串）的判别概率。我们还描述了用于检测文本字符边界的二值化算法。

通过 Adaboost 计算判别概率。标准 AdaBoost 算法用于例如区分人脸和非人脸 [62]，通过组合一组 n 个二元值弱分类器或特征测试 $\text{Tst}_{\text{Ada}}(\mathbf{I}) = (h_1(\mathbf{I}), ..., h_n(\mathbf{I}))$ 来学习二进制值强分类器 H_{Ada}，使用一组权重 $\alpha_{\text{Ada}} = (\alpha_1, ..., \alpha_n)$[22] 使得

$$H_{\text{Ada}}(\text{Tst}_{\text{Ada}}(\mathbf{I})) = \text{sign}(\sum_{i=1}^{n} \alpha_i h_i(\mathbf{I})) = \text{sign} < \alpha_{\text{Ada}}, \text{Tst}_{\text{Ada}}(\mathbf{I}) > . \tag{8.47}$$

这些特征选自预先设计的字典 Δ_{Ada}。特征的选择和权重的调整被提出作为监督学习问题。给定一组标记的例子，$\{(\mathbf{I}_i, \ell_i) : i = 1, 2, ..., M\}$ ($\ell_i = \pm 1$)，AdaBoost 学习可以表达为贪婪地优化以下函数 [50]：

$$(\alpha_{\text{Ada}}^*, \text{Tst}_{\text{Ada}}^*) = \arg \min_{\text{Tst}_{\text{Ada}} \subset \Delta_{\text{Ada}}} \arg \min_{\alpha_{\text{Ada}}} \sum_{i=1}^{M} \exp^{-\ell_i < \alpha_{\text{Ada}}, \text{Tst}_{\text{Ada}}(\mathbf{I}_i) >}. \tag{8.48}$$

为了获得判别概率，我们使用一个定理 [23]，该定理指出由 AdaBoost 学习的特征和测试给出（渐近）对象标签（例如，面部或非面部）的后验概率。AdaBoost 强分类器可以重新作为对数后验比率测试。

Theorem 8.3 (Friedman et al 1998) *具有足够的训练样本 M 和特征 n，AdaBoost learning 选择权重 α_{Ada}^* 并测试 $\text{Tst}_{\text{Ada}}^*$ 以满足*

$$q(\ell = +1|\mathbf{I}) = \frac{e^{\ell < \alpha_{\text{Ada}}, \text{Tst}_{\text{Ada}}(\mathbf{I}_i) >}}{e^{< \alpha_{\text{Ada}}, \text{Tst}_{\text{Ada}}(\mathbf{I}_i) >} + e^{- < \alpha_{\text{Ada}}, \text{Tst}_{\text{Ada}}(\mathbf{I}_i) >}}.$$

此外，强分类器渐近收敛于后验概率比测试

$$H_{\text{Ada}}(\text{Tst}_{\text{Ada}}(\mathbf{I})) = \text{sign}(< \alpha_{\text{Ada}}, \text{Tst}_{\text{Ada}}(\mathbf{I}) >) = \text{sign}(\frac{q(\ell = +1|\mathbf{I})}{q(\ell = -1|\mathbf{I})}).$$

实际上，AdaBoost 分类器以不同的比例应用于图像中的窗口。每个窗口被评估为面部或非面部（或文本与非文本）。对于大多数图像，对于图像的几乎所有部分，面部或文本的后验概率可以忽略不计。因此，我们使用一系列测试 [62, 64]，这使我们能够通过将其边际概率设置为零来快速拒绝许多窗口。当然，AdaBoost 只会收敛到真实后验概率 $p(\ell|\mathbf{I})$ 的近似值，因为只能使用有限数量的测试（并且只有有限数量的训练数据）。请注意，AdaBoost 只是学习后验概率的一种方法（参见定理 (8.1)）。已经发现对于具有相对刚性结构的物体图案（例如面和文本）非常有效（字母的形状是可变的，但序列的图案是相当结构的 [11]）。

AdaBoost 训练. 标准的 AdaBoost 训练方法 [22, 23] 可以使用并与使用不对称加权的级联方法相结合 [62, 64]。级联使得算法能够通过少量测试将大部分图像排除为面部或文本位置，并允许计算资源集中在图像的更具挑战性的部分上。

图 8.34: 从中提取训练文本补丁的一些场景. Courtesy of Tu et al. [56].

AdaBoost for text 旨在检测文本段。测试数据是从旧金山的 162 张图像中手工提取的（见图 8.34），包含 561 个文本图像。超过一半的图像是由盲人志愿者拍摄的（这减少了偏见）。我们将每个文本图像分成几个重叠的文本段，其中宽度与高度的比例固定为 2：1（通常包含两到三个字母）。共有 7,000 个文本段用作积极训练集。负面例子是通过类似于 Drucker 等人 [20] 的自助程序获得的。首先，通过从图像数据集中的窗口随机采样来选择负面示例。在对这些样本进行训练之后，AdaBoost 算法应用于一系列尺度，以对训练图像中的所有窗口进行分类。那些错误分类为文本的内容随后被用作 AdaBoost 下一阶段的反面例子。最容易与文本混淆的图像区域是植被和重复结构，如栏杆或建筑物外墙。用于 AdaBoost 的功能是与基本过滤器的统计相对应的图像测试。选择这些特征来检测对于各个字母或数字的形状相对不变的文本片段的属性。它们包括平均图像窗口内的强度，以及边数的统计。我们参考 [11] 了解更多细节。

面部的 AdaBoost 后卫以类似的方式进行训练。这次我们使用 Haar 基函数 [62] 作为基本特征。我们使用 FERET [49] 数据库作为我们的正面例子，通过允许小的旋转和平移转换，我们有 5,000 个正面例子。我们使用与上述文本相同的策略来获得负面示例。

在这两种情况下，我们使用许多不同的阈值评估测试数据集的对数后验比率测试（参见 [62]）。与面部工作一致 [62]，AdaBoost 给出了非常高的性能，很少有误报和漏报。但是由于每张图像中有大量的窗口，这些低错误率有些误导（见表 (8.1)）。较小的误报率可能意味着任何常规图像的大量误报。通

Object	False Positive	False Negative	Images	Subwindows
Face	65	26	162	355,960,040
Face	918	14	162	355,960,040
Face	7542	1	162	355,960,040
Text	118	27	35	20,183,316
Text	1879	5	35	20,183,316

表 8.1: AdaBoost 在不同阈值下的表现。

过改变阈值，我们可以消除误报或误报，但不能同时消除两者。我们通过显示 AdaBoost 在图 8.35中提出的面部区域和文本区域来说明这一点。如果我们通过设置阈值来尝试分类，那么我们只能以误报为代价正确地检测所有面部和文本。

图 8.35: 这些框显示了 AdaBoost 对数后验比率测试检测到的具有固定阈值的面部和文本。观察由植被，树木结构和随机图像模式引起的误报。不可能为该图像选择没有误报和漏报的阈值。正如我们后来的实验所示，生成模型将消除误报并恢复丢失的文本。Courtesy of Tu et al. [56].

当 Adaboost 与图像解析器中的通用区域模型集成时，通用区域提议可以消除误报并找到 AdaBoost 未命中的文本。例如，未检测到图 8.35右侧面板中的“9”，因为我们的 AdaBoost 算法是在文本段上训练的。相反，它被检测为通用着色区域，后来被识别为字母’9’，见图 8.37。图 8.37 和 8.39中删除了图 8.35中的一些误报文本和面部。

二值化算法. 用于文本的 AdaBoost 算法需要用下面描述的二值化算法来补充，以确定文本字符位置。然后将形状上下文 [4] 和信息特征 [57] 应用于二值化结果，以便为特定字母和数字的存在提出建议。在许多情况下，二值化的结果是如此之好以至于可以立即检测到字母和数字（即，二值化阶段提出的提议被自动接受），但情况并非总是如此。我们注意到二值化比边缘检测等替代方法提供了更好的结果 [10]。

图 8.36: 检测到的文本的二值化示例。Courtesy of Tu et al. [56].

二值化算法是 Niblack [45] 提出的算法的变体。我们使用基于自适应窗口大小的自适应阈值来对图像强度进行二值化。需要自适应方法，因为包含文本的图像窗口通常具有阴影，阴影和遮挡。我们的二值化方法通过其局部窗口 $r(v)$ 的强度分布（以 v 为中心）确定每个像素 v 的阈值 $T_b(v)$。该阈值由下式给出

$$T_b(v) = \mu(\mathbf{I}_{r(v)}) + k \cdot std(\mathbf{I}_{r(v)}),$$

其中 $\mu(\mathbf{I}_{r(v)})$ 和 $std(\mathbf{I}_{r(v)})$ 是局部窗口内的强度均值和标准差。选择局部窗口的大小作为其强度方差高于固定阈值的最小可能窗口。参数 $k = \pm 0.2$，其中 $\pm$ 允许前景比背景更亮或更暗的情况。

8.10.6 图像解析实验

图像解析算法应用于许多室外/室内图像。PC（Pentium IV）的速度可与分割方法（如标准化切割 [37] 或 [58] 中的 DDMCMC 算法）相媲美。它通常运行大约 10-40 分钟。计算时间的主要部分用于分割通用区域和边界扩散 [68]。

a. Input image　b. Segmentation　c. Object recognition　d. Synthesized image

图 8.37: 两幅图像的分割和识别结果。与图 8.35中显示的纯自下而上（AdaBoost）结果相比，结果得到了改善。Courtesy of Tu et al. [56].

a. Input image　b. Synthesis 1　c. Synthesis 2

图 8.38: 图 8.37中图像的特写外观。深色眼镜由通用着色模型解释，因此面部模型不必适合这部分数据。否则面部模型会有困难，因为它会试图将眼镜贴在眼睛上。标准 AdaBoost 仅以误报为代价正确分类这些面，请参见图 8.35。我们展示了两个合成面的例子，一个（合成 1）用深色眼镜（用阴影区域建模）和另一个（合成 2）用深色眼镜去除，即使用生成面部模型来取样被墨镜遮盖住的面部（即. 眼睛）的部分。Courtesy of Tu et al. [56].

图 8.37, 8.38, 和 8.39显示了一些具有重杂波和阴影效果的具有挑战性的例子。我们将结果分为两部分。一个显示通用区域和对象的分割边界，另一个显示用文本符号检测的文本和面以指示文本识别，即字母由算法正确读取。然后我们合成从似然模型 $p(\mathbf{I}|W^*)$ 中采样的图像，其中 W^* 是通过解析算法获得的解析图（面，文本，区域参数和边界）。合成图像用于可视化解析图 W^*，即计算机“理解”的图像内容。

在实验中，我们观察到，与我们之前的工作 [58] 相比，面部和文本模型改进了图像分割结果，后者仅使用通用区域模型。相反，通用区域模型通过去除一些误报并恢复最初未检测到的对象来改进对象检测。我们现在讨论具体的例子。

在图 8.35中，我们展示了两个图像，其中使用 AdaBoost 纯粹自下而上检测文本和面部。选择阈值是不可能的，因此我们的 AdaBoost 算法没有误报或漏报。为了确保没有假阴性，除了'9' 之外，我们不得不降低阈值并允许由植被和重阴影引起的误报（例如标志“HEIGHTS OPTICAL”中的阴影）。

在任何阈值都没有检测到字母'9'。这是因为我们的 AdaBoost 算法经过训练以检测文本段，因此没有响应单个数字。

相比之下，图 8.37显示了这两个图像的图像解析结果。我们看到 AdaBoost 提出的误报被删除了，因为通用区域模型可以更好地解释它们。通用着色模型通过解释文本“HEIGHTS OPTICAL”上的重阴影和女性的深色眼镜来帮助检测物体，见图 8.38。此外，现在可以正确检测到丢失的数字“9”。该算法首先将其检测为通用着色区域，然后使用切换节点属性的子内核将其重新分类为数字。

从解析图 W^* 合成图像的能力是贝叶斯方法的优点。综合有助于说明生成模型的成功，有时还有弱点。此外，合成图像显示模型已捕获有关图像的信息量。在表 (8.2) 中给出了在表示 W^* 中使用的变量的数量，并表明它们与 jpeg 字节大致成比例。W^* 中的大多数变量用于表示分割边界上的点，并且目前它们是独立计数的。我们可以通过有效地编码边界点来减少 W^* 的编码长度，例如，使用空间接近度。然而，图像编码不是我们当前工作的目标，并且需要更复杂的生成模型来合成非常逼真的图像。

Image	Stop	Soccer	Parking	Street	Westwood
jpg bytes	23,998	19,563	23,311	26,170	27,790
$\|W^*\|$	4,886	3,971	5,013	6,346	9,687

表 8.2: 与 JPG 字节相比，每个图像的 W^* 变量数。

在本节中，我们描述了图像解析的两个具有挑战性的技术问题。首先，可以在组合和分解模式中构造解析图。组合模式通过对小元素进行分组来进行，而分解方法涉及检测整个对象然后定位其部分。

对于图 8.40(a)，组合模式似乎最有效。通过自下而上的测试检测猎豹，例如 AdaBoost 学到的那些，由于猎豹的形状和光度特性的大变化，似乎很难。相比之下，使用 Swendsen-Wang Cuts（[1] 和第 6 章）来分割图像并使用自下而上的组合方法和具有多个级别的解析树来获取猎豹的边界是非常实用的。从像素作为叶子开始构造解析图（图 8.40 (a) 中有 46,256 个像素）。使用局部图像纹理相似性来获得图的下一级，以构建对应于图像的原子区域的图节点（其中的 113 个）。然后，算法通过对原子区域进行分组（即，每个原子区域节点将是纹理区域节点的子节点），在下一级别为“纹理区域”构建节点（其中 4 个）。在每个级别，我们计算相邻节点（例如像素或原子区域）有多大可能属于同一对象或模式的判别（提议）概率。然后，我们应用实现拆分和合并动态的转换内核（使用提议）。我们参考第 6 章或 [1] 进行更详细的讨论。

对于具有较小变异性的对象，如图 8.40（b）所示，我们可以使用底部 - 对立面（例如 AdaBoost）来激活代表整个面部的节点。然后可以通过扩展面部节点来向下构建解析图（即，在分解模式中）以为面

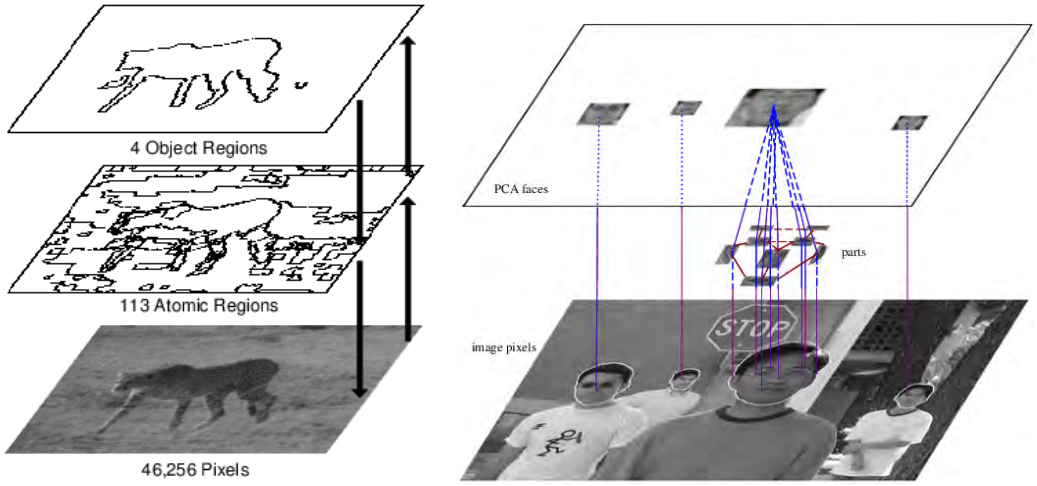
a. Input image　b. Segmentation　c. Object recognition　d. Synthesized image

图 8.39: 室外图像的分割和识别结果。观察以多种比例检测面部和文本的功能。Courtesy of Tu et al. [56].

部的部分创建子节点。反过来，这些子节点可以扩展到代表更精细比例部分的孙子节点。可以使节点扩展量适应于取决于图像的分辨率。例如，图 8.40 （b）中最大的面被扩展为子节点，但没有足够的分辨率来扩展对应于三个较小面的面节点。主要技术问题是开发一种数学标准，对于哪种类型的对象和模式，模式最有效。这将使算法能够相应地调整其搜索策略。

第二个挑战涉及最佳排序。图像解析算法的控制策略不以最佳方式选择测试和子内核。在每个时间步，子内核的选择独立于当前状态 W（尽管选择应用子内核的图中的位置将取决于 W）。此外，算法永远不会使用一些自下而上的测试。如果选择过程需要低计算成本，那么具有自适应地选择子内核和测试的控制策略将更有效。我们寻求一种最佳的选择控制策略，它对大量图像和视觉模式都有效。选择标准应该选择那些最大化信息增益的测试和子内核。

我们可以使用这两个信息标准。第一个在定理 8.1中说明，并通过执行新的测试 Tst_+ 来测量在解析图中获得的变量 w 的信息。信息增益为 $\delta(w||\text{Tst}_+) = KL(p(w|\mathbf{I}) \,||\, q(w|\text{Tst}(\mathbf{I}))) - KL(p(w|\mathbf{I}) \,||\, q(w|\text{Tst}_t(\mathbf{I}), F_+))$，其中 $\text{Tst}(\mathbf{I})$ 表示先前的测试（并且 KL 是 Kullback-Leibler 发散）。第二个在定理 8.2中说明。它通过 KL-发散 $\delta(\mathcal{K}_a) = KL(p \,||\, \mu_t) - KL(p \,||\, \mu_t \mathcal{K}_a)$ 的减小来测量子核 $\mathcal{K}_a$ 的功率。当通过 $\text{Tst}_t(\mathbf{I})$ 通知时，减小量 δ_a 给出子核 $\mathcal{K}_a$ 的功率的度量。我们还需要考虑选择程序的计算成本。有关如何在考虑计算成本的情况下最佳选择测试的案例研究，请参见 [6]。



(a) “composition” (b) “decomposition”

图 8.40: 构造解析图的两种机制。请参阅文本以获取解释。Courtesy of Tu et al. [56].

参考文献

[1] Adrian Barbu and Song-Chun Zhu. Multigrid and multi-level swendsen-wang cuts for hierarchic graph partition. In *CVPR*, volume 2, pages II–731, 2004.

[2] Adrian Barbu and Song-Chun Zhu. Generalizing swendsen-wang to sampling arbitrary posterior probabilities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1239–1253, 2005.

[3] Simon A Barker, Anil C Kokaram, and Peter JW Rayner. Unsupervised segmentation of images. In *SPIE's International Symposium on Optical Science, Engineering, and Instrumentation*, pages 200–211. International Society for Optics and Photonics, 1998.

[4] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, 2002.

[5] Elie Bienenstock, Stuart Geman, and Daniel Potter. Compositionality, mdl priors, and object recognition. *NIPS*, pages 838–844, 1997.

[6] Gilles Blanchard and Donald Geman. Hierarchical testing designs for pattern recognition. *Annals of Statistics*, pages 1155–1202, 2005.

[7] Charles Bouman and Bede Liu. Multiple resolution segmentation of textured images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(2):99–113, 1991.

[8] Kevin Bowyer, Christine Kranenburg, and Sean Dougherty. Edge detector evaluation using empirical roc curves. *Computer Vision and Image Understanding*, 84(1):77–103, 2001.

[9] Pierre Bremaud. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, volume 31. Springer, 1999.

[10] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):679–698, 1986.

[11] Xiangrong Chen and Alan L Yuille. Detecting and reading text in natural scenes. In *CVPR*, volume 2, pages II–366. IEEE, 2004.

[12] Yizong Cheng. Mean shift, mode seeking, and clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(8):790–799, 1995.

[13] Dorin Comaniciu and Peter Meer. Mean shift analysis and applications. In *ICCV*, volume 2, pages 1197–1203. IEEE, 1999.

[14] Timothy F Cootes, Gareth J Edwards, and Christopher J Taylor. Active appearance models. *IEEE Transactions on pattern analysis and machine intelligence*, 23(6):681–685, 2001.

[15] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[16] Peter Dayan, Geoffrey E Hinton, Radford M Neal, and Richard S Zemel. The helmholtz machine. *Neural computation*, 7(5):889–904, 1995.

[17] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[18] Yining Deng, B Shin Manjunath, and Hyundoo Shin. Color image segmentation. In *CVPR*, volume 2, 1999.

[19] Persi Diaconis and Phil Hanlon. Eigen-analysis for some examples of the metropolis algorithm. *Contemporary Mathematics*, 138:99–117, 1992.

[20] Harris Drucker, Robert Schapire, and Patrice Simard. Boosting performance in neural networks. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):705–719, 1993.

[21] David A Forsyth. Sampling, resampling and colour constancy. In *CVPR*, volume 1. IEEE, 1999.

[22] Yoav Freund, Robert E Schapire, et al. Experiments with a new boosting algorithm. In *ICML*, volume 96, pages 148–156, 1996.

[23] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.

[24] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 6:721–741, 1984.

[25] Stuart Geman and Chii-Ruey Hwang. Diffusions for global optimization. *SIAM Journal on Control and Optimization*, 24(5):1031–1043, 1986.

[26] Peter J Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[27] Ulf Grenander and Michael I Miller. Representations of knowledge in complex systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 549–603, 1994.

[28] Peter W Hallinan, Gaile G Gordon, Alan L Yuille, Peter Giblin, and David Mumford. *Two-and three-dimensional patterns of the face*. AK Peters, Ltd., 1999.

[29] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[30] Dan Klein and Christopher D Manning. A generative constituent-context model for improved grammar induction. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 128–135. Association for Computational Linguistics, 2002.

[31] Georges Koepfler, Christian Lopez, and Jean-Michel Morel. A multiscale algorithm for image segmentation by variational method. *SIAM journal on numerical analysis*, 31(1):282–299, 1994.

[32] Scott Konishi, Alan L. Yuille, James M. Coughlan, and Song Chun Zhu. Statistical edge detection: Learning and evaluating edge cues. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(1):57–74, 2003.

[33] Yvan G Leclerc. Constructing simple stable descriptions for image partitioning. *International journal of computer vision*, 3(1):73–102, 1989.

[34] Hsien-Che Lee and David R Cok. Detecting boundaries in a vector field. *Signal Processing, IEEE Transactions on*, 39(5):1181–1194, 1991.

[35] Fei Fei Li, Rufin VanRullen, Christof Koch, and Pietro Perona. Rapid natural scene categorization in the near absence of attention. *Proceedings of the National Academy of Sciences*, 99(14):9596–9601, 2002.

[36] Jun S Liu. *Monte Carlo strategies in scientific computing*. springer, 2008.

[37] Jitendra Malik, Serge Belongie, Thomas Leung, and Jianbo Shi. Contour and texture analysis for image segmentation. *International journal of computer vision*, 43(1):7–27, 2001.

[38] Christopher D Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT press, 1999.

[39] D Marr. Vision, 1982. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, 1982.

[40] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 416–423, 2001.

[41] Kerrie L Mengersen, Richard L Tweedie, et al. Rates of convergence of the hastings and metropolis algorithms. *The Annals of Statistics*, 24(1):101–121, 1996.

[42] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[43] Baback Moghaddam and Alex Pentland. Probabilistic visual learning for object representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):696–710, 1997.

[44] David Mumford. *Neuronal architectures for pattern-theoretic problems*. Large-Scale Theories of the Cortex. Cambridge, MA: MIT Press, 1994.

[45] W Niblack. An introduction to digital image processing. 1986.

[46] Shunichiro Oe. Texture segmentation method by using two-dimensional ar model and kullback information. *Pattern recognition*, 26(2):237–244, 1993.

[47] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[48] Nikos Paragios and Rachid Deriche. Coupled geodesic active regions for image segmentation: A level set approach. In *ECCV*, pages 224–240. 2000.

[49] P Jonathon Phillips, Harry Wechsler, Jeffery Huang, and Patrick J Rauss. The feret database and evaluation procedure for face-recognition algorithms. *Image and vision computing*, 16(5):295–306, 1998.

[50] Robert E Schapire. The boosting approach to machine learning: An overview. In *Nonlinear estimation and classification*, pages 149–171. Springer, 2003.

[51] Stan Sclaroff and John Isidoro. Active blobs. In *Computer Vision, 1998. Sixth International Conference on*, pages 1146–1153. IEEE, 1998.

[52] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.

[53] Robert H Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. *Physical Review Letters*, 58(2):86–88, 1987.

[54] Simon Thorpe, Denis Fize, Catherine Marlot, et al. Speed of processing in the human visual system. *Nature*, 381(6582):520–522, 1996.

[55] Anne Treisman. Features and objects in visual processing. *Scientific American*, 255(5):114–125, 1986.

[56] Zhuowen Tu, Xiangrong Chen, Alan L Yuille, and Song-Chun Zhu. Image parsing: Unifying segmentation, detection, and recognition. *International Journal of computer vision*, 63(2):113–140, 2005.

[57] Zhuowen Tu and Alan L Yuille. Shape matching and recognition–using generative models and informative features. In *Computer Vision-ECCV 2004*, pages 195–209. Springer, 2004.

[58] Zhuowen Tu and Song-Chun Zhu. Image segmentation by data-driven markov chain monte carlo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):657–673, 2002.

[59] Zhuowen Tu and Song-Chun Zhu. Parsing images into regions, curves, and curve groups. *International Journal of Computer Vision*, 69(2):223–249, 2006.

[60] Shimon Ullman. Visual routines. *Cognition*, 18(1):97–159, 1984.

[61] Shimon Ullman. Sequence seeking and counter streams: a computational model for bidirectional information flow in the visual cortex. *Cerebral cortex*, 5(1):1–11, 1995.

[62] Paul Viola and Michael Jones. Fast and robust classification using asymmetric adaboost and a detector cascade. *Proc. of NIPS01*, 2001.

[63] Jia-Ping Wang. Stochastic relaxation on partitions with connected components and its application to image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(6):619–636, 1998.

[64] Jianxin Wu, James M Rehg, and Matthew D Mullin. Learning a rare event detection cascade by direct feature selection. In *NIPS*, page None, 2003.

[65] Jonathan S Yedidia, William T Freeman, and Yair Weiss. Generalized belief propagation. In *NIPS*, pages 689–695, 2001.

[66] Song Chun Zhu and Xiuwen Liu. Learning in gibbsian fields: How accurate and how fast can it be? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):1001–1006, 2002.

[67] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126, 1998.

[68] Song Chun Zhu and Alan Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(9):884–900, 1996.

[69] Song-Chun Zhu, Rong Zhang, and Zhuowen Tu. Integrating bottom-up/top-down for object recognition by data driven markov chain monte carlo. In *CVPR*, volume 1, pages 738–745. IEEE, 2000.

第 8 章　汉密尔顿函数和拉文蒙特卡洛算法

“假设我们采取一定的热量并将其转化为工作。在这样做的过程中，我们没有摧毁热量，我们只将它转移到另一个地方，或者可能将其转换为另一种能量形式。”- 艾萨克 • 阿西莫夫

介绍

Hamiltonian Monte Carlo（HMC）是一个强大的高维连续分布采样框架。Langevin Monte Carlo（LMC）是 HMC 的一个特例，广泛用于深度学习应用程序。给定 n -维连续密度 $P(X)$，实现 HMC 的唯一要求是能量 $U(X) = -\log P(X)$ 是可微的。与其他 MCMC 方法（例如切片采样，Swendsen-Wang 切割）一样，HMC 引入辅助变量以促进原始空间中的移动。在 HMC 中，原始变量代表位置，辅助变量代表动量。每个位置维度都有一个相应的动量变量，因此原始变量和辅助变量的联合空间的维数为 $2n$，是原始空间大小的两倍。一旦引入动量变量，Hamilton 方程用于模拟具有势能 U 的物理系统的时间演化。Hamilton 方程的性质确保了关节空间中的运动保留了原始空间中 P 的分布。

9.1　哈密顿力学

9.1.1　汉密尔顿等式

哈密顿力学原理是 HMC 采样方法的基础。哈密顿力学最初是作为拉格朗日力学的替代但等价的公式而开发的，两者都等同于牛顿力学。在哈密顿力学中，物理系统的状态由一对 n-维变量 q 和 p 表示。变量 q 表示系统中的位置，p 表示动量。联合状态 (qp) 在一个时刻提供物理系统的完整描述。HMC 框架中的位置和动量可以解释为简单动力学系统中熟悉的位置和动量概念的高维扩展。

随着时间的推移，状态 (qp) 的演化由表示系统能量的标量值函数 $H(qp)$ 和一对称为 Hamilton 方程的偏微分方程控制:

$$\frac{dq}{dt} = \frac{\partial H}{\partial p}, \tag{9.1}$$

$$\frac{dp}{dt} = -\frac{\partial H}{\partial q}. \tag{9.2}$$

$H(qp)$ 通常被称为系统的汉密顿量。根据汉密尔顿方程更新 q 和 p 可确保系统属性的许多属性（包括能量）的守恒。换句话说，$H(qp)$ 保持不变因为 (qp) 随时间变化。

在许多情况下，包括标准 HMC 和 LMC，哈密顿量可以用可分离形式表示

$$H(q,p) = U(q) + K(p), \tag{9.3}$$

其中 $U(q)$ 代表系统的潜在能量，$K(p)$ 代表动能。哈密顿量是可分的时候

$$\frac{dq}{dt} = \frac{\partial H}{\partial p} = \frac{\partial K}{\partial p} \qquad \text{和} \qquad \frac{dp}{dt} = -\frac{\partial H}{\partial q} = -\frac{\partial U}{\partial q}, \tag{9.4}$$

汉密尔顿方程有简化形式

$$\frac{dq}{dt} = \frac{\partial K}{\partial p}, \tag{9.5}$$

$$\frac{dp}{dt} = -\frac{\partial U}{\partial q}. \tag{9.6}$$

9.1.2 HMC 的简单模型

考虑在无摩擦的 2D 表面上移动的点质量（图 9.1）。每个瞬时系统状态可以用 (qp) 对来描述，其中 q 是一个 2D 变量，给出点质量的位置（纬度和经度坐标），p 是一个在每个方向给出动量的 2D 变量。系统哈密顿量的形式为 $H(qp) = U(q) + K(p)$，其中 $U(q)$ 是点质量的高度（相对于固定参考点）和动力学能量 K 的格式为 $K(p) = ||p||^2/(2m)$，其中 m 为质量。

点质量的运动由汉密尔顿方程确定。直观地，汉密尔顿方程表示在穿越景观时发生的动能和势能之间的权衡。例如，在斜坡一侧具有零动量的状态将被向下拉动，并且其势能将在最陡下降的方向上转换为运动（动能）。另一方面，如果点质量沿平坦平面向前移动然后遇到障碍物，则动能转移到势能，如果障碍物足够陡峭，质量将减缓甚至反向到朝向平原的方向。

这个简单的模型非常类似于汉密尔顿蒙特卡罗的“物理系统”。在 HMC 中，q 表示目标密度的 n-维空间中的一个点

$$P(q) = \frac{1}{Z}\exp\{-U(q)\}. \tag{9.7}$$

与其他 MCMC 方法一样，不需要难以处理的标准化常量 Z。HMC 只要求 $U(q)$ 是可微分的，因为汉密尔顿的方程涉及 U 的导数。在实践中，动量 p 遵循多元正态分布，其具有能量函数

$$K(p) = \frac{1}{2} p^\intercal \Sigma^{-1} p \tag{9.8}$$

对于正定协方差矩阵 Σ。最简单的选择是 $\Sigma = \sigma^2 I_n$，当 $\Sigma = \sqrt{m}$ 时，它给出标准动能方程 $K(p) = ||p||^2/(2m)$。HMC 不需要 p 的特定分布，但是从正态分布中采样的简单性以及与信息几何的重要连接（参见章节 9.5）使得高斯动量成为自然选择。

在每次 HMC 迭代开始时，采样新动量 p。然后使用汉密尔顿方程更新联合状态 (q,p) 以达到提议状态 (q^*,p^*)。与随机游动方法不同，其中位移与 $\sqrt{t}$ 成比例，HMC 轨迹期间的位移可以在 t 中线性缩

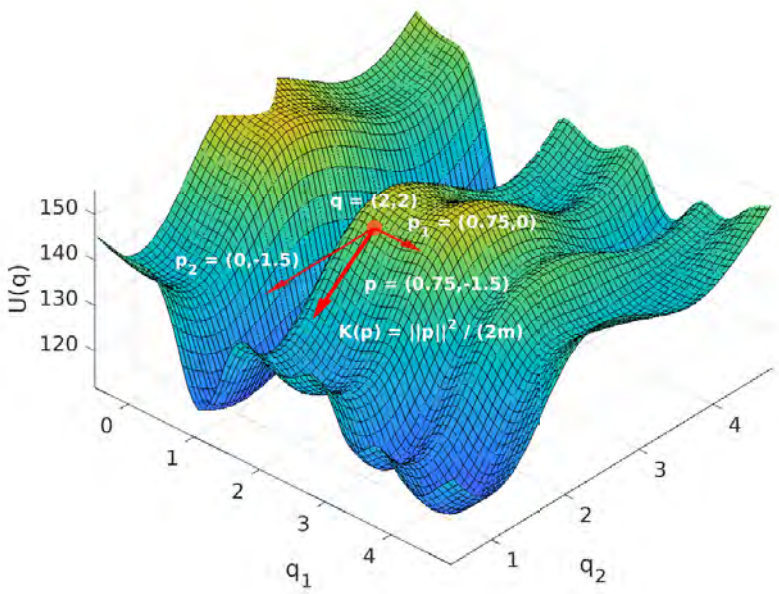


图 9.1: 二维哈密顿力学图在一个瞬间。位置 q 代表原始变量，动量 p 是一个辅助变量，其维度与 q 相同。目标能量 $U(q)$ 是 $\mathbb{R}^3$ 中的 2D 表面。对 (qp) 完全描述系统状态的一个瞬间。系统具有哈密顿量 $H(qp) = U(q) + K(p)$，Hamilton 的等式（9.1）和（9.2）定义 (qp) 随时间变化。引入允许在 q 中移动的 p，保留分配 $P(q) = \frac{1}{Z}\exp\{-U(q)\}$。

放。因此，HMC 可以在每次迭代中制定有效的全局提案。在实践中，汉密尔顿方程无法准确求解，并且必须包括 Metropolis-Hastings 接受步骤。至关重要的是，从 HMC 获得的 q 的边际分布具有固定分配 $P(q)$。整个轨迹中的位置变量 q 保留为 $P(q)$ 的样本，并且动量 p 被丢弃。

9.2 哈密顿力学的性质

哈密顿力学有几个重要的属性，可确保 HMC 定义有效的抽样方法。本节介绍 Hamilton 方程的理想连续时间解的性质。实际上，Hamilton 方程不能明确求解，必须使用离散数值近似。除了用于保存能量，9.3表示可以在离散时间实现中保留相同的属性。

9.2.1 节约能源

使用 Hamilton 方程更新物理系统会保留 Hamiltonian $H(q,p)$ 的值，因此即使 q 和 p 会有所不同,$H(q,p)$ 的值应该随时间保持不变。换句话说，汉密尔顿方程定义的路径沿哈密顿量 $H(q,p)$ 的水平曲线移动（见图 9.2）。

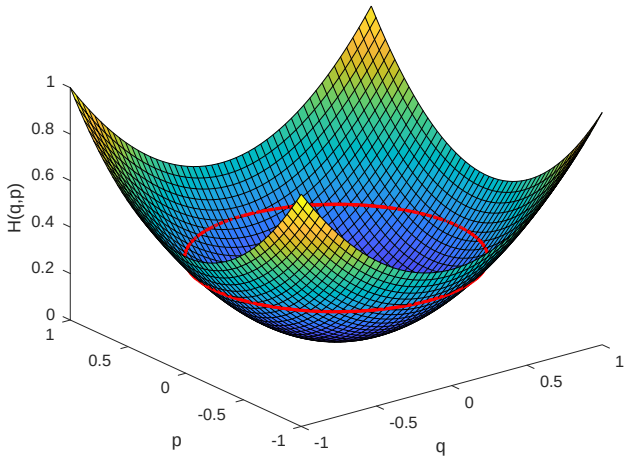


图 9.2: 对于 $q \sim \mathrm{N}(0,1)$ 和 $p \sim \mathrm{N}(0,1)$，汉密尔顿函数 $H(q,p) = U(q) + K(p) = (q^2 + p^2)/2$. 单个 HMC 轨迹的路径限制为 H 的水平曲线。可能的水平曲线以红色显示。

能量守恒的证据很简单：

$$\frac{dH}{dt} = \sum_{i=1}^{n} \left[\frac{\partial H}{\partial q_i} \frac{dq_i}{dt} + \frac{\partial H}{\partial p_i} \frac{dp_i}{dt} \right] = \sum_{i=1}^{n} \left[\frac{\partial H}{\partial q_i} \frac{\partial H}{\partial p_i} - \frac{\partial H}{\partial q_i} \frac{\partial H}{\partial p_i} \right] = 0.$$

此属性在 HMC 中很重要，因为它确保 $H(q,p) = H(q^*,p^*)$，其中 (q,p) 是关节空间中的先前状态和 (q^*,p^*) 是建议的状态。结合哈密顿力学的其他性质，能量守恒可以用来表明理想的 HMC 定义了一个接受概率为 1 的 Metropolis-Hastings 提议。实际上，因为必须使用离散数值近似来求解 Hamilton 方程，因此 $H(q,p)$ 可能与 $H(q^*,p^*)$ 不同，所以这个属性只是近似为真。如果近似是准确的，则该差异应该相对较小，并且仍然可以实现高接受概率。

9.2.2 可逆性

由哈密顿力学定义的 $(q(t),p(t))$ 到 $(q(t+s),p(t+s))$ 的映射是唯一的，因此是可逆的。如果 $H(q,p) = U(q) + K(p)$ 和 $K(p) = K(-p)$，在具有高斯动量的标准 HMC 中为真，那么在路径末尾对 p 取反可以明确地给出逆映射，同时将系统演化为 s，然后再次取反 p。在这种情况下因为 $T(q(t+s),-p(t+s)) = (q(t),p(t))$ (见图 9.3)，所以映射 $T : (p(t),q(t)) \mapsto (q(t+s),-p(t+s))$ 是完全可逆的。可逆性将用于表明 HMC 满足详细的平衡，这是证明 MCMC 方法具有所需静态分布的最简单方法。在 Hamilton 方程的离散实现中可以精确地保留可逆性。

9.2.3 辛结构和体积保存

对于任何平滑函数 $H : \mathbb{R}^{2n} \to \mathbb{R}$，Hamilton 方程在流形上定义一种特殊类型的向量场和一个辛结构 $\mathbb{R}^{2n}$。辛的流形是一个光滑的流形 M（实际上，通常是 $\mathbb{R}^{2n}$），差分 2 形式 ω 称为辛形式。与 $\mathbb{R}^{2n}$ 中的

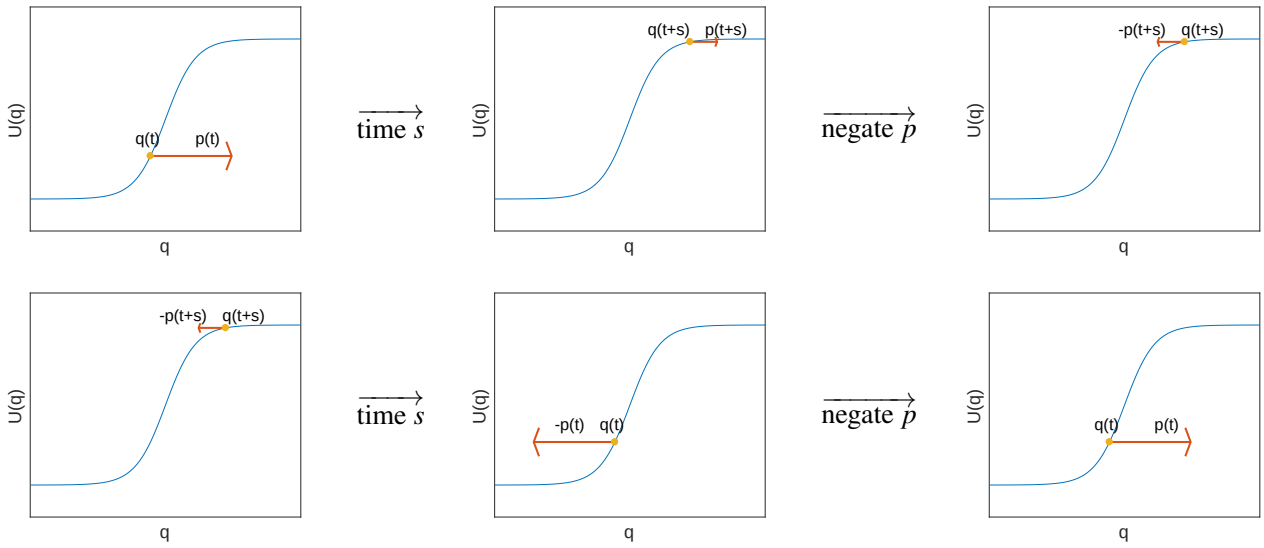


图 9.3: 具有可分离哈密顿量的 1D HMC 的可逆性 $H(q,p) = U(q) + K(p)$ 和 $K(p) = K(-p)$。通过根据 Hamilton 方程为时间 s 更新状态 (q,p) 定义的坐标变化，并且在轨迹末尾（顶行）取反 p 可以通过相同的过程完全颠倒（底行））。

Hamilton 方程相关的标准辛形式是 $\omega = \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix}$, 因为

$$\frac{d}{dt}(q,p) = \omega \frac{dH}{d(q,p)}.$$

一般来说，ω 只需要在 M 上关闭且不退化的 2-形式。辛形式可以直观地理解为从哈密顿量 $H(q,p)$ 的差分 1 形式 dH 生成向量场的方式。

通过积分 Hamilton 方程得到的解（或等效地流动在由汉密尔顿算子 H 上导致的矢量场上的辛流形 M）具有保留辛形式 ω 的重要特性。换句话说，映射 $(q(t),p(t)) \mapsto (q(t+s),p(t+s))$ 超过 $(q,p) \in M$ 定义了从 M 到自身的差异性，尊重 ω 的结构。哈密顿量流下 ω 的不变性是哈密顿力学的许多守恒性质的数学基础，包括能量守恒。

保留 2 形式 ω 的一个重要结果是 Hamilton 方程下的体积守恒，这一结果被称为路易斯维尔定理。使用辛几何，这个定理的证明非常简单。M 上的非简并 2-form ω 可以提升到 n^{th} 能量来定义一个非简并卷形式 ω^n（ω^n 是 $2n$-形式，因为 ω 是 2-形式），并且在 Hamiltonian 流量下保留 ω 意味着保留 ω^n 的音量。此属性对于 HMC 很重要，因为它确保从 Hamilton 方程获得的坐标 $(q,p) \mapsto (q^*,p^*)$ 的变化具有绝对值为 1 的行列式的雅可比行列式。如果没有体积保存，计算雅可比行列式的决定因素以在每个提议之后重新调整密度的难度将使 HMC 在实践中不可行。如果使用正确的更新方案，则体积保持可以精确地保持在 Hamilton 方程的离散实现中。

可以给出一个简单的只使用 Hamilton 方程而不参考辛几何的体积保持证明。设可以给出一个简单

的是由 Hamilton 方程定义的向量场。那么 V 的分歧到处都是 0，因为

$$\operatorname{div}(V)=\sum_{i=1}^{n}\left(\frac{\partial}{\partial q_i}\frac{dq_i}{dt}+\frac{\partial}{\partial p_i}\frac{dp_i}{dt}\right)=\sum_{i=1}^{n}\left(\frac{\partial}{\partial q_i}\frac{\partial H}{\partial p_i}-\frac{\partial}{\partial p_i}\frac{\partial H_i}{\partial q_i}\right)=\sum_{i=1}^{n}\left(\frac{\partial^2 H}{\partial q_i\partial p_i}-\frac{\partial^2 H}{\partial q_i\partial p_i}\right)=0,$$

并且可以显示具有偏差 0 的矢量场来保持体积。

9.3 Hamilton 方程的 leapfrog 离散化

除了最简单的系统外，不可能精确地求解 Hamilton 方程，因此哈密顿力学的数值实现必须依赖于对真实连续解的离散近似。在讨论最有效和广泛使用的离散化之前，引入了两种不太有效但具有指导性的方法，它们被称为 Leapfrog 积分器。

9.3.1 欧拉的方法

在汉密尔顿方程下离散哈密顿量 H 的时间演化的最直接的方法是通过一些小步长 ε 同时更新 q 和 p，如下所示：

$$\begin{aligned}p(t+\varepsilon)&=p(t)+\varepsilon\frac{dp}{dt}(q(t),p(t))=p(t)-\varepsilon\frac{\partial H}{\partial q}(q(t),p(t)),\\ q(t+\varepsilon)&=q(t)+\varepsilon\frac{dq}{dt}(q(t),p(t))=q(t)+\varepsilon\frac{\partial H}{\partial p}(q(t),p(t)).\end{aligned}$$

这种离散化被称为欧拉方法。它不会保留音量，只需几步就可能导致不准确的近似值。

9.3.2 改进的欧拉方法

Euler 方法的改进是改进的 Euler 方法，它使用 q 和 p 的交替 ε-size 更新。当能量函数具有可分离形式 $H(q,p)=U(q)+K(p)$，（9.5）和（9.6）保持并更新 q 仅取决于 p（反之亦然）。根据这一观察结果，Modified Euler 方法包括使用步长 ε 更新当前 q 的当前 p，然后使用*更新后的* p 更新当前 q 相同的步长 ε 如下：

$$p(t+\varepsilon)=p(t)-\varepsilon\frac{\partial U}{\partial q}(q(t)), \tag{9.9}$$

$$q(t+\varepsilon)=q(t)+\varepsilon\frac{\partial K}{\partial p}(p(t+\varepsilon)). \tag{9.10}$$

颠倒 p 和 q 的更新顺序同样有效。交替更新是剪切变换，这保留了体积。通过保持 Hamilton 方程的真实连续解的体积保持性质，改进的欧拉方法提供了比朴素欧拉方法更好的离散化。当哈密顿量具有可分离形式 $H(q,p)=U(q)+K(p)$ 时，Modified Euler 方法仅保留体积。

与汉密尔顿方程的真正解法不同，由于更新的顺序所以修正的欧拉方法是不可逆的。假设 $K(p)=K(-p)$ 并考虑一个从 (q,p) 开始的提议，更新 p 然后 q 使用 Modified Euler 以达到新状态 (q^*,p^*)，最后取反动量达到 $(q^*,-p^*)$。可以在 $(q^*,-p^*)$ 开始一个新链，更新 q^* 然后 $-p^*$ 用并用 Modified Euler 达到 $q-p$ 并且取反达到 (q,p) 的动量。另一方面，从 $(q^*,-p^*)$ 开始并将更新顺序 $-p^*$ 加入到原始提议并应

用，这时 q^* 可能会导致状态关闭，但是由于离散化错误，它不等于 (q,p)。理想的积分器应该能够在不改变积分器本身的情况下精确地倒转所有更新（即，倒转更新顺序）。

9.3.3 Leapfrog 积分器

Leapfrog 积分器是 Modified Euler 方法的近似，它是 HMC 中使用的标准离散积分方案。当哈密顿量具有可分离形式 $H(q,p)=U(q)+K(p)$ 时，(9.5) 和 (9.6) 保持并且 Leapfrog 积分器满足体积保持和可逆性，这是 Hamilton 方程的真正连续解的理想特性。下面给出了 Leapfrog 积分器大小的一个步骤，其中 ε 是步长的参数：

$$p(t+\varepsilon/2)=p(t)-(\varepsilon/2)\frac{\partial U}{\partial q}(q(t)), \tag{9.11}$$

$$q(t+\varepsilon)=q(t)+\varepsilon\frac{\partial K}{\partial p}(p(t+\varepsilon/2)), \tag{9.12}$$

$$p(t+\varepsilon)=p(t+\varepsilon/2)-(\varepsilon/2)\frac{\partial U}{\partial q}(q(t+\varepsilon)). \tag{9.13}$$

一个 Leapfrog 更新包含一个 $(\varepsilon/2)$-大小更新的 p 和旧的 q，接着是一个 ε-大小的更新 q 和新的 p，其次通过 $(\varepsilon/2)$-大小更新 p 和新的 q。当执行多个 Leapfrog 步骤时，上述方案相当于仅在轨迹的开始和结束时执行 p 的半步更新，并且在 q 和 p 之间的全步更新之间交替，因为 p 的两个 $(\varepsilon/2)$-大小更新和旧步骤的结束以及新步骤的开始相当于 p 的单个 ε-大小的更新。

修改的 Euler 方法和 Leapfrog 方法之间的唯一区别是将 P 的初始 ε-大小的更新在修改的 Euler 轨迹中拆分为 p 两个 $(\varepsilon/2)$-步的更新步骤分别在在 Leapfrog 轨迹的开头和结尾。

9.3.4 Leapfrog 积分器的属性

Leapfrog 积分器的对称性确保了可逆性，因为可以通过取反 p，应用 Leapfrog 积分器并再次否定 p 来反转单个 Leapfrog 步骤。在 HMC 的一般情况下，在轨迹末端的 p 的否定是可逆的，但在实践中当使用高斯辅助变量时可以忽略它，因为 $K(p)=K(-p)$。

Leapfrog 积分器的体积保持与修改的 Euler 方法的原因相同：当哈密顿量具有可分离形式 $H(q,p)=U(q)+K(p)$ 时，q 的更新仅取决于 p，反之亦然。因此，由 Leapfrog 方程（9.11）定义的坐标 $(q(t),p(t))\mapsto(q(t+\varepsilon),p(t+\varepsilon))$ 的变化，（9.12）和（9.13）是三个剪切变换的组合，每个剪切变换都有一个雅可比行列式 1。该组合定义了雅可比行列式 1 的单个坐标变化，因为组合的雅可比行列式是各个坐标变化的雅可比行列式的乘积。

由（9.11）定义的映射 $(q(t),p(t))\mapsto(q(t),p(t+\varepsilon/2))$ 的非正式证明是如下所示的切变转换。用几乎相同的证明可以表示（9.12）和（9.13），以及（9.9）和（9.10），也是切变转换。

设 $J_p=\begin{pmatrix}\frac{\partial q^*}{\partial q} & \frac{\partial q^*}{\partial p}\\ \frac{\partial p^*}{\partial q} & \frac{\partial p^*}{\partial p}\end{pmatrix}$ 是对应于 $(q(t),p(t))\mapsto(q(t),p(t+\varepsilon/2))$ 坐标变化的雅可比行数 $(q,p)\mapsto(q^*,p^*)$。考虑一些初始状态 $(q(0),p(0))$ 及其近邻 $(q'(0),p'(0))=(q(0)+\delta u_q,p(0)+\delta u_p)$ 为某些单位向量 $u=$

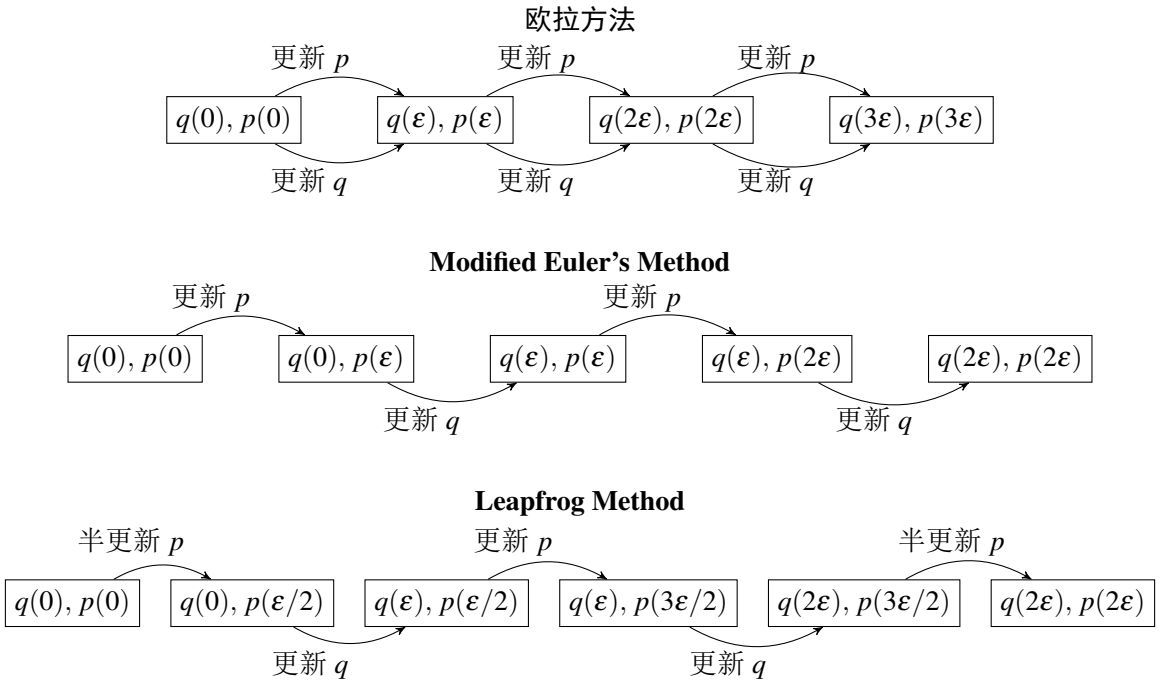


图 9.4: 三种 Hamilton 方程积分方法的可视化。顶部：欧拉的方法。同时更新 p 和 q 会产生较差的近似值。*Middle*：改进的 Euler 方法。当哈密顿量可分离形式为 $H(q,p) = U(q) + K(p)$ 时，p 和 q 的交替更新可以保留交易量。由于更新顺序所以它不可逆。*Bottom*：Leapfrog 方法。Leapfrog 方法与 Modified Euler 方法相同，只是在轨迹的开头和结尾处有半步 p 更新。可逆性需要半更新。

(u_q, u_p) 和一些小 $\delta > 0$。这两个状态的 Leapfrog 更新的第一步由下式给出

$$p(\varepsilon/2) = p(0) - (\varepsilon/2)\frac{\partial U}{\partial q}(q(0)),$$

$$p'(\varepsilon/2) = p'(0) - (\varepsilon/2)\frac{\partial U}{\partial q}(q'(0)) = p(0) + \delta u_p - (\varepsilon/2)\frac{\partial U}{\partial q}(q(0) + \delta u_q),$$

和 $q(\varepsilon/2) = q(0)$，因为 q 在此步骤期间未更新所以 $q'(\varepsilon/2) = q'(0) = q(0) + \delta u_q$。使用泰勒展开，对于小的 δ，$\frac{\partial U}{\partial q}(q(0) + \delta u_q) \approx \frac{\partial U}{\partial q}(q(0)) + \delta[\frac{\partial^2 U}{\partial q^2}(q(0))]u_q$。因此

$$\begin{pmatrix} q'(\varepsilon/2) - q(\varepsilon/2) \\ p'(\varepsilon/2) - p(\varepsilon/2) \end{pmatrix} \approx \delta \begin{pmatrix} I_n & 0 \\ -(\varepsilon/2)\frac{\partial^2 U}{\partial q^2}(q(0)) & I_n \end{pmatrix} \begin{pmatrix} u_q \\ u_p \end{pmatrix}$$

让 δ 变为 0 意味着

$$J_p = \begin{pmatrix} \frac{\partial q^*}{\partial q} & \frac{\partial q^*}{\partial p} \\ \frac{\partial p^*}{\partial q} & \frac{\partial p^*}{\partial p} \end{pmatrix} = \begin{pmatrix} I_n & 0 \\ -(\varepsilon/2)\frac{\partial^2 U}{\partial q^2} & I_n \end{pmatrix}$$

这是一个带有行列式 1 的剪切矩阵。注意 ε 是任意的且在此证明中是固定的，并且该限制仅在空间扰动 δ 中进行。Leapfrog Integrator 精确保留任何 ε 的空间。通过使用导数 $\frac{\partial U}{\partial q}$ or $\frac{\partial K}{\partial p}$ 相同方法的泰勒展开，可以发现 Leapfrog 的另外两个更新步骤是剪切变换。

如果在 HMC 中使用高斯辅助变量，则由公式（9.12）给出的 q-更新具有形式的雅可比行列式

$$J_q = \begin{pmatrix} I_n & \varepsilon\Sigma^{-1} \\ 0 & I_n \end{pmatrix}$$

其中 Σ 是 p 的协方差矩阵。使用带有 $\Sigma \approx \frac{\partial^2 U}{\partial q^2}$ 的高斯提议可以显着改善通过 q-空间的移动，尤其是在受最大和最小线性方向的受限宽度之间具有高比率的分布进行采样时，即局部协方差的最大和最小特征值之间的较大比率。不幸的是，如果能量函数 U 没有恒定曲率，那么 Σ 的理想选择随着 q 的位置而变化，在这种情况下，$H(q,p)$ 不再是可分离的，Leapfrog 积分器不能保留空间，解决汉密尔顿方程变得更加困难。有关详细讨论，请参见章节 9.5。

9.4 汉密尔顿蒙特卡洛和朗格文蒙特卡洛

本节介绍 HMC 方法，以及称为 Langevin Monte Carlo（LMC，也称为 Metropolis-Adjusted Langevin 算法或 MALA）的 HMC 的特例。讨论了 HMC 调优，该部分以 HMC 满足详细平衡的证据结束。

9.4.1 HMC 的公式

为了与前面章节中使用的符号一致，在 HMC 期间要采样的目标密度将写为

$$P(q) = \frac{1}{Z}\exp\{-U(q)\} \tag{9.14}$$

对于 $q \in \mathbb{R}^n$ 和平滑势能函数 $U : \mathbb{R}^n \to \mathbb{R}$ 与规范化常数 Z。有关在 HMC 中处理约束的讨论，以便 q 可以限制为集合 $U \subset \mathbb{R}^n$，请参阅 [7]。在贝叶斯推断中，U 是一组参数 q 和数据集 X 的后验分布的负对数，具有先前的 π 和对数似然 l，即

$$U(q) = -\log[\pi(q)] - l(X|q).$$

HMC 是辅助变量方法，标准辅助变量是 $p \sim \mathrm{N}(0,\Sigma)$，负对数密度

$$K(p) = \frac{1}{2}p^\intercal\Sigma^{-1}p \tag{9.15}$$

对于一些 $n \times n$ 正定的协方差矩阵 Σ。$(q,p) \in \mathbb{R}^{2n}$ 对具有联合密度

$$P(q,p) = \frac{1}{Z}\exp\{-H(q,p)\} = \frac{1}{Z}\exp\left\{-U(q) - \frac{1}{2}p^\intercal\Sigma^{-1}p\right\} \tag{9.16}$$

和联合能量函数

$$H(q,p) = U(q) + K(p) = U(q) + \frac{1}{2}p^\intercal\Sigma^{-1}p. \tag{9.17}$$

联合概率密度 $P(q,p) = \frac{1}{Z}e^{-H(q,p)}$ 具有边缘分布 $q \sim \frac{1}{Z}\exp^{-U(q)}$ 因为

$$\int_{\mathbb{R}^n} P(q,p)dp = \frac{1}{Z_q}e^{-U(q)} \int_{\mathbb{R}^n} \frac{1}{Z_p}e^{-\frac{1}{2}p^T\Sigma^{-1}p}dp = \frac{1}{Z_q}e^{-U(q)}. \tag{9.18}$$

因此，从联合密度 $P(q,p)$ 中抽样将提供跟随目标密度 $P(q)$ 的样本 q。

本书仅讨论正态分布的辅助变量，由于多种原因，正常的辅助变量是自然选择。物理中使用的标准能量函数 $K(p) = ||p||^2/m$ 在 $\Sigma = mI_n$ 时与 (9.15) 相等。正太分可以准确有效得进行模拟更重要的是，具有高斯动量的 HMC 必须与信息几何有重要联系，这有助于正确调整动量协方差 Σ（参见章节 9.5）。

9.4.2 HMC 算法

让 U 成为目标能量函数，让 q 成为当前状态。首先，从 $N(0,\Sigma)$ 中抽取正常的辅助变量 p。然后，在状态 (q,p) 上执行步长 ε 的 LLeapfrog 更新。最后，Metropolis-Hastings 步骤用于接受或拒绝提议 (q^*,p^*) 以纠正来自 Leapfrog Integrator 的离散化错误。在接受步骤之后，丢弃 p^* 并为下一个 HMC 步骤生成新的 p。标准 HMC 算法如下。

HMC 算法

输入: 可微能量函数 $U(q)$，初始状 $q_0 \in \mathbb{R}^n$, $n \times n$ p.d. 协方差矩阵 Σ，步长 ε，Leapfrog 步数 L，迭代次数 N

输出: 马尔可夫链样本 $\{q_1, \ldots, q_N\}$，固定分配 U

对于 $i = 1 : N$,

1. 产生动量 $p_{i-1} \sim \mathrm{N}(0, \Sigma)$.

2. 设 $(q_0', p_0') = (q_{i-1}, p_{i-1})$。从 (q_0', p_0') 开始执行 LLeapfrog 更新，以达到提议状态 (q_0', p_0')，如下所示：

   (a) 做 p 的前半步更新，

   $$p'_{\frac{1}{2}} = p_0' - (\varepsilon/2)\frac{\partial U}{\partial q}(q_0'). \tag{9.19}$$

   (b) 对于 $l = 1 : (L-1)$, 执行 q 和 p 的交替全步更新：

   $$q_l' = q_{l-1}' + \varepsilon\Sigma^{-1}p'_{l-\frac{1}{2}}, \tag{9.20}$$

   $$p'_{l+\frac{1}{2}} = p'_{l-\frac{1}{2}} - \varepsilon\frac{\partial U}{\partial q}(q_l'). \tag{9.21}$$

   如果 $L = 1$，这是 LMC 算法，请跳过此步骤。

   (c) 计算最后的全步 q-更新和最后的半步 p-更新

   $$q_L' = q_{L-1}' + \varepsilon\Sigma^{-1}p'_{L-\frac{1}{2}}, \tag{9.22}$$

   $$p_L' = p'_{L-\frac{1}{2}} - (\varepsilon/2)\frac{\partial U}{\partial q}(q_L'). \tag{9.23}$$

   然后建议的状态是 $(q^*, p^*) = (q_L', p_L')$。

3. 根据 Metropolis-Hastings 接受概率接受建议的状态 (q^*, p^*)

   $$\alpha = \min\left(1, \exp\left\{-\left(U(q^*) + \frac{1}{2}(p^*)^{\intercal}\Sigma^{-1}p^*\right) + \left(U(q_{i-1}) + \frac{1}{2}p_{i-1}^{\intercal}\Sigma^{-1}p_{i-1}\right)\right\}\right). \tag{9.24}$$

   如果提议被接受，那么 $q_i = q^*$。否则，$q_i = q_{i-1}$。在提案之后，可以丢弃 p_{i-1} 和 p^*。

注释 1:HMC 算法中步骤 2 结束时的建议状态应该是 $(q^*, p^*) = (q_L', -p_L')$ 需要在 Leapfrog 轨迹的最后对动量取反来确保 HMC 的可逆性和详细平衡，如下一节所示。由于高斯分布的 $K(p) = K(-p)$，步骤 3 中的计算不会改变 $p^* = p_L'$ 或 $p^* = -p_L'$，以及否定可以安全地忽略。

注释 2: 可以使用不同的协方差矩阵 Σ 来生成每个 p_i。但是，必须在单个提案的持续时间内使用相同的 Σ。在 Leapfrog 迭代之间更改 Σ 会破坏 Leapfrog 更新的剪切结构，并且无法再保证详细的平衡。这是 RMHMC 方法的主要障碍，其通过允许基于当前位置的依赖性 $\Sigma(q)$ 来考虑局部流形结构。

步骤 3 中的 Metropolis-Hastings 接受概率对应于 $P(q,p)$ 的联合密度的比率：

$$\begin{aligned}\alpha &= \min\left(1, \frac{P(q^*,p^*)}{P(q_{i-1},p_{i-1})}\right) \\ &= \min\left(1, \frac{\exp\{-H(q^*,p^*)\}}{\exp\{-H(q_{i-1},p_{i-1})\}}\right) \\ &= \min\left(1, \frac{\exp\{-U(q^*)-K(p^*)\}}{\exp\{-U(q_{i-1})-K(p_{i-1})\}}\right).\end{aligned}$$

Leapfrog 更新是确定性的，体积保持不变，并且完全可逆，因此没有转换概率 $Q((q,p)\mapsto(q^*,p^*))$ 出现在 Metropolis-Hastings 比率中，只有密度 Pq,p。汉密尔顿方程的真正连续解完全满足 $H(q,p)=H(q^*,p^*)$ 的建议 (q^*,p^*) 根据由自初始状态 (q,p) 生成 Hamilton 等式。因此，如果 Hamilton 方程的精确解可用，则 Metropolis-Hastings 接受概率总是等于 1。

由于必须使用 Leapfrog 离散化，因此 H 的值不完全保守，并且需要 Metropolis-Hastings 步骤来纠正此错误。为了获得 Hamilton 方程的精确近似和高接受概率，有必要正确调整采样变量 Σ，ε 和 L。理论上，对于任何参数设置，HMC 都具有静态分布 $\frac{1}{Z}e^{-U(q)}$，但是与任何 MCMC 方法一样，良好混合需要良好的调整。有关调整 HMC 参数的详细信息，请参见 9.4.4部分。

需要在每次迭代中采样新的动量 p_i 以满足 HMC 中的遍历性。回想一下 $H(q,p)$ 保持（近似）不变，因为 (q,p) 被更新。如果 $H(q,p)=U(q)+K(p)$ and $K(p)=(1/2)p^\intercal\Sigma^{-1}p$，那么显然是 $U(q)\le h=H(q,p)$ 用于轨迹中的所有 q，这限制了可以访问的可能状态（特别是 $\{q:U(q)>h\}$ 不能到达）。每个新动量都将探索限制在哈密顿量 $H(q,p)$ 的单级曲线上，这可能无法涵盖 q 的整个空间。刷新动量允许沿着 $H(q,p)$ 的不同水平曲线移动，这对于 HMC 的遍历性是必要的。

9.4.3 LMC 算法

Langevin Monte Carlo 或者说 LMC 只是 HMC 算法，只执行 $L=1$ Leapfrog 更新。LMC 相当于 Langevin 方程

$$q(t+\varepsilon)=q(t)-\frac{\varepsilon^2}{2}\Sigma^{-1}\frac{\partial U}{\partial q}(q(t))+\varepsilon\sqrt{\Sigma^{-1}}z \tag{9.25}$$

在额外 p-更新的优化中和 Metropolis-Hastings 接受步骤中使用 $z\sim\mathrm{N}(0,I_n)$），将得到 $\frac{1}{Z}e^{-U(q)}$ 上的采样算法。LMC 算法可以使用上一节中的 HMC 算法实现，其中 $L=1$，但通常以稍微更紧凑的方式完成，只有一个 q-更新和一个 p-更新，如下所示。

LMC 算法

输入: 离散能量函数 $U(q)$, 初始状态 $q_0 \in \mathbb{R}^n$, $n \times n$ P.D. 协方差矩阵 Σ, 步长 ε, 迭代次数 N

输出: 固定分布 U 的马尔可夫链样本 $\{q_1, \ldots, q_N\}$

对于 $i = 1 : N$,

1. 产生动量 $p_{i-1} \sim \mathrm{N}(0, \Sigma)$.

2. 设 $(q_0', p_0') = (q_{i-1}, p_{i-1})$. 根据 Langevin 方程更新 q:

$$q_1' = q_0' - \frac{\varepsilon^2}{2}\Sigma^{-1}\frac{\partial U}{\partial q}(q_0') + \varepsilon\Sigma^{-1}p \tag{9.26}$$

   并根据 Leapfrog 更新更新 p

$$p_1' = p_0' - \frac{\varepsilon}{2}\frac{\partial U}{\partial q}(q_0') - \frac{\varepsilon}{2}\frac{\partial U}{\partial q}(q_1'). \tag{9.27}$$

   那么建议的状态就是 $(q^*, p^*) = (q_1', p_1')$.

3. 根据 Metropolis-Hastings 的接受概率接受提议状态 (q^*, p^*)

$$\alpha = \min\left(1, \exp\left\{-\left(U(q^*) + \frac{1}{2}(p^*)^\intercal\Sigma^{-1}p^*\right) + \left(U(q_{i-1}) + \frac{1}{2}p_{i-1}^\intercal\Sigma^{-1}p_{i-1}\right)\right\}\right). \tag{9.28}$$

   如果提议被接受那么 $q_i = q^*$. 否则, $q_i = q_{i-1}$. 动量 p_{i-1} 可以在提案后丢弃。

在（9.26）中的 q-更新的表达式显示了作用于 LMC 中原始空间的两个竞争力，并且在 HMC 的每个 Leapfrog 更新中都有相同的原则。$-\frac{\varepsilon^2}{2}\Sigma^{-1}\frac{\partial U}{\partial q}(q_0')$ 由一个 p.d. 矩阵重新调整简单梯度下降，大致相当于能源领域的“引力”。当动量协方差是 Fisher 信息 $\Sigma(\theta) = \mathrm{E}_{X|\theta}\left[\frac{\partial^2 U}{\partial\theta^2}(X|\theta)\right]$ 为贝叶斯推理问题给出观察 X，这个词成为自然梯度 $\Sigma^{-1}(\theta)\frac{\partial U}{\partial\theta}$（Amari,[1]），它适应参数空间的局部曲率。$\frac{\partial U}{\partial\theta}$.Fisher 信息总是正定的，自然梯度比天真梯度 $\frac{\partial U}{\partial\theta}$ 具有更好的性能和不变性。

术语 $\varepsilon\Sigma^{-1}p = \varepsilon\sqrt{\Sigma^{-1}}z$ 粗略对应于随机“风”。梯度项的引力应该超过扩散项沿能量斜率的随机力，但是一旦达到局部最小值并且梯度消失，扩散项就变得占主导地位。需要明智地选择 Σ，以确保一旦链条到达能源盆地的底部，随机扩散力可以提出有意义的提议。如果 $\Sigma(q) \approx \frac{\partial^2 U}{\partial q^2}$, 然后 $\sqrt{\Sigma(q)^{-1}}z \sim \mathrm{N}(0, \Sigma(q)^{-1})$ 并且扩散力遵循局部协方差结构 $[\frac{\partial^2 U}{\partial q^2}]^{-1}$，以便“风”主要沿着局部流形吹动。想象一下，当地的景观是一个峡谷（见图 9.5）。如果风垂直吹向峡谷的墙壁，峡谷的陡峭边缘将阻止任何有意义的运动。但是，如果风向平行于峡谷，则可以通过峡谷移动。

LMC 具有与“完整”HMC 不同的属性，LMC 使用了大量的 Leapfrog 更新 L。由于在 LMC 中只有一步之后就丢弃了动量，因此不鼓励连续的提议向同一方向移动，LMC 在随机游走中探索景观 $U(q)$。与不使用局部几何的 Random-Walk Metropolis-Hastings 不同，LMC 更新中使用的梯度信息有助于优化阶段的采样，但 HMC 在扩散阶段具有更好的理论属性，因为重复更新相同的动量 p 可以导致关节空间中更长的轨迹。但是，在某些情况下，LMC 比完整 HMC 更有用。实现 LMC 的 q-dependent 动力 $p \sim \mathrm{N}(0, \Sigma(q))$ 的动态近似比 HMC 更实际和准确，将在下面 RMHMC 部分讨论。在复杂的景观中，HMC 的好处受到

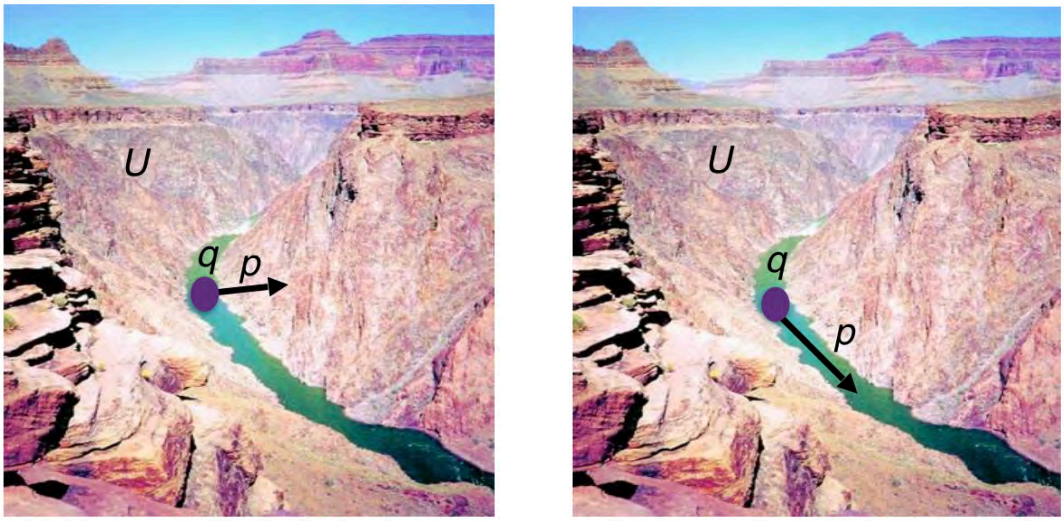


图 9.5: 在“峡谷”潜在的 U 山谷中扩散。*左*：当来自 HMC 的动量 p 指向陡峭的峡谷壁时，q 的移动将很快被逆转并且粒子被卡住。*右*：当来自 HMC 的势头指向峡谷谷时，因为没有遇到障碍,q 是可以自由移动的。在 U 的局部最小值附近进行有效采样需要一个反映局部缩放的动量协方差（参见章节 9.5）。

Leapfrog 动力学的不稳定性的限制，并且通常必须保持 Leapfrog 步数 L 的数量以实现合理的接受率，在这种情况下，HMC 和 LMC 非常相似。

9.4.4 调整 HMC

本节讨论使用固定的 Σ 调整标准 HMC 设置中的 ε 和 L 参数。因为通过重新调整 q-空间可以自然地将结果扩展到任意 Σ，如 Lemma9.1，所以考虑 $\Sigma = I_n$ 就足够了。调整 Σ 是 9.5部分的主要主题。

为了使 Leapfrog Integrator 能够准确地模拟 Hamilton 方程以便 H 在整个 Leapfrog 更新中保持近似恒定，步长 ε 必须足够小。由于 Metropolis-Hastings 接受度取决于原始装填与提案之间的 H 差异，较小的步长往往具有较高的接受率，因为 H 不会改变那么多。另一方面，如果 ε 太小，那么链条将保持几乎静止，有效采样变得不可能。在新环境中调整 HMC 的一种简单方法是设置 $\Sigma = I_n$，$L = 1$，并改变 ε，直到获得 40% - 85% 的接受率。该范围在高接受度和良好运动之间提供了良好的平衡，并且在任一方向上更极端的 ε 都不可能显着改善 HMC 性能。在状态空间的不同区域可能需要不同的 ε 值。

当 $\Sigma = I_n$ 时，理想步长 ε^* 应该在能量范围中局部区域的最受约束的线性方向上大致等于 $U(q)$ 的宽度。如果景观是高斯或近似高斯，则 ε^* 应该接近局部协方差矩阵的最小特征值的平方根。当 ε 远大于 $U(q)$ 的最小边际标准偏差时，（9.20）中的 q-更新将导致低接受率，因为球形辅助变量将作出沿着最受限制的方向的不太可能的提议。另一方面，当 ε 和最小标准差大致相等时，应该以相当高的概率接受建议，因为任何方向上的局部偏差将给出与当前状态具有大致相同能量的状态。

由于每个 Leapfrog 更新在 q-空间移动约 ε 的距离，忽略渐变的影响，ε 最多限制为 q 的最小边际标准差，在单个 HMC 步骤中达到几乎独立状态所需的 Leapfrog 步数 $L^* \approx \sqrt{\lambda_{\max}}/\varepsilon^*$，其中 $\lambda_{\max}$ 是 q-空间的局部协方差的最大特征值。记住除非遇到能量景观中的障碍物，汉密尔顿轨迹不是随机行走并且往往沿着相同的方向移，因此位移与步数 L 成线性比例。在简单的景观中，因为 Leapfrog 动态非常准

确 L 可能非常大（超过 100），但在更复杂的景观中，当 L 变得太大时，HMC 轨迹的接受率会迅速下降。如果 εL 不是最大标准差的量级，HMC 将表现出自相关性，并且无法有效地在空间中移动。有关这些原则的实验演示，请参见章节 9.6。

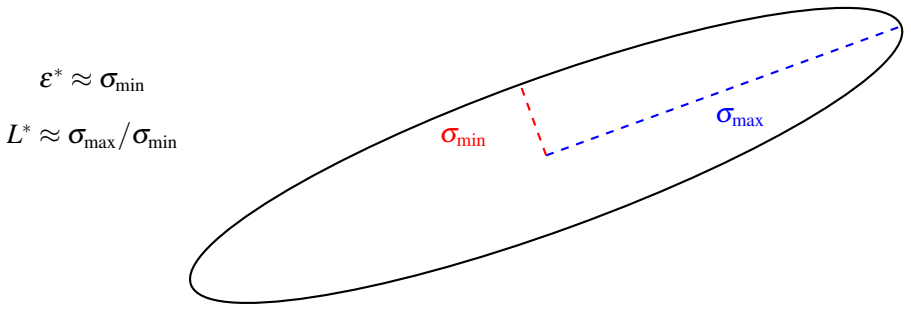


图 9.6: 在 $\Sigma = U \begin{pmatrix} \sigma_{\min}^2 & 0 \\ 0 & \sigma_{\max}^2 \end{pmatrix} U^\intercal$ 时，$q \sim \mathrm{N}(0, \Phi)$ 的等级曲线（一个标准差）. 设 $p \sim \mathrm{N}(0, I_n)$. 优化步长为 $\varepsilon^* \approx \sigma_{\min}$, 因为较大的提议将导致可能被拒绝的 U 的高能区域，而较小的提案效率较低。另一方面，设置 $L^* \approx \sigma_{\max}/\sigma_{\min}$ 需要在每次 HMC 迭代中制作全局提案。当 $\sigma_{\max}/\sigma_{\min}$ 很大时可能会出现问题，因为在经过多次更新后，Leapfrog 动态通常会变得不稳定。在重新缩放部分之后，相同的原则适用于动量协方差 $\Sigma \neq I_n$9.5.1.

局部相关性可能在整个状态空间中剧烈变化，而且在一种景观模式中有效的参数设置可能在另一模式中表现不佳。但是，使用 RMHMC（9.5可以缓解这些问题，因为 ε 变为“无量纲”数量，少量的 Leapfrog 步骤（甚至 $L = 1$）仍然可以提供良好的空间移动。

9.4.5 HMC 的详细平衡证明

最简单的方法表明 MCMC 采样方法保留分布 P 是为了表明该方法对于 MCMC 方法定义的提议密度 T 满足详细的平衡关系

$$P(x)T(x \mapsto x^*) = P(x^*)T(x^* \mapsto x) \tag{9.29}$$

对于 MCMC 方法定义的提议密度 T。下面给出了 HMC 满足详细平衡的证据。还可以证明 HMC 是遍历的并且保证探索整个 q-空间，前提是在每次 HMC 更新开始时从一个小的随机间隔中选择 ε。需要随机选择 ε 以确保理论上的遍历性，因为在 HMC 期间可能出现完全或近乎完全的周期性的轨道，但这种现象仅存在于 ε 的窄带中。有关详细信息，请参阅 [5]。

Theorem 9.1 *HMC 算法满足详细的平衡并且具有静态分布* $P(q) = \frac{1}{Z} e^{-U(q)}$。

证明 9.4.1 *足以证明为了表明 HMC 过程中 q 的固定分布遵循 $P(q,p) = \frac{1}{Z} e^{-U(q)-K(p)}$，HMC 步骤满足联合分布的详细平衡 $P(q,p) = \frac{1}{Z} e^{-U(q)-K(p)}$，如（9.18）所示。For this proof, $p \sim \frac{1}{Z} e^{-K(p)}$ 对于这个证明，$p \sim \frac{1}{Z} e^{-K(p)}$ 用于平滑能量函数 $\mathbb{R}^n$ 中的 K，在 $K(p) = K(-p)$ 时（不一定是高斯）。*

设 $q \sim \frac{1}{Z} e^{-U(q)}$. 在 HMC 算法的第 1 步中生成 $p \sim \frac{1}{Z} e^{-K(p)}$ 后，很明显 $(q,p) \sim P(q,p)$ 因为联合密度的可分解形式所暗示的 q 和 p 的独立性。设提议 (q^*, p^*) 是从 (q,p) 状态执行大小 ε 的 LLeapfrog 并且在轨迹的终点取反 p。如章节 9.3.4所示，每个 Leapfrog 步骤是带有行列式 1 的坐标的变化，并且轨迹

末尾的 p 的取反是坐标的变化，雅可比行列式的绝对值为 *1*。因此，$(q,p)\mapsto(q^*,p^*)$ 是坐标的变化，雅可比行列式的绝对值为 *1*，因为坐标变化的组成的决定因素是决定因素每个变化的乘积。通过改变概率密度的坐标规则

$$g(y)=f(x)\left|det\left(\frac{dx}{dy}\right)\right|$$

其中 $f(x)$ 是原始密度，而 $g(y)$ 是映射 $x\mapsto y$ 的新密度，因此 (q^*,p^*) 具有与 (q,p) 相同的密度函数，因为 $|det(dx/dy)|=1$。因为将 *L Leapfrog* 步骤的大小 ε 应用于 (q^*,p^*) 并在轨迹末尾否定 p^* 将给出原始状态 (q,p)，该提议也是完全可逆的。

由于映射 $(q,p)\mapsto(q^*,p^*)$ 是确定性且可逆的，因此 *HMC* 算法定义的提议密度 T 是

$$T((q,p)\mapsto(q^*,p^*))=min\left(1,\exp\{-(U(q^*)+K(p^*))+(U(q)+K(p))\}\right),$$

$$T((q,p)\mapsto(q',p'))=0\quad if\quad (q',p')\neq(q^*,p^*).$$

类似地，从 (q^*,p^*) 开始的转换密度仅对于提案 (q,p) 非零，并且具有形式

$$T((q^*,p^*)\mapsto(q,p))=min\left(1,\exp\{-(U(q)+K(p))+(U(q^*)+K(p^*))\}\right).$$

HMC 的详细平衡方程（*9.29*）是

$$\frac{1}{Z}e^{-U(q)-K(p)}min\left(1,\frac{\exp\{-U(q^*)-K(p^*)\}}{\exp\{-U(q)-K(p)\}}\right)=\frac{1}{Z}e^{-U(q^*)-K(p^*)}min\left(1,\frac{\exp\{-U(q)-K(p)\}}{\exp\{-U(q^*)-K(p^*)\}}\right)$$

这显然是真的。因此，*HMC* 满足详细的平衡并保留联合分布 $\frac{1}{Z}e^{-U(q)-K(p)}$。

9.5 黎曼流形 HMC

Riemann Manifold HMC（或 RMHMC）通过允许辅助动量变量 p 的协方差矩阵在能源领域的当前位置 q 上具有依赖性 $\Sigma(q)$ 来扩展标准 HMC 方法。这可以极大地改善 HMC 的采样特性，特别是在 q 的分布集中在状态空间中的低维流形上的情况下。

具有 $Sigma=I_n$ 的传统 HMC 在这些情况下无效，因为接受所需的步长必须是 q 的最小标准差的量级，这将比沿着主要复杂维度的标准偏差小几个数量级。使用大量的 Leapfrog 步骤 L 只能部分弥补这种差异，而在复杂的景观中，当 L 太大时，轨迹会变得不稳定。

另一方面，RMHMC 使用局部几何图形沿着局部流形在有意义的方向上提出建议，从而仅通过少量的 Leapfrog 步骤实现更好的采样。依赖性 $\Sigma(q)$ 使 RMHMC 的动态变得复杂，并且需要额外的计算考虑，其中一些是非常有问题的。虽然在许多实际情况下确切的 RMHMC 实现是不可行的，但是近似实现可以在灵活且通用的框架中提供 RMHMC 的许多益处。

9.5.1 HMC 中的线性变换

下面的引理说明了 HMC 在某种线性变换下的一个重要的不变性，阐明了提议协方差 *Sigma* 在 HMC 中的作用。

Lemma 9.1 设 $U(q)$ 是一个平滑的能量函数，设 $p \sim N(0,\Sigma)$ 为 *p.d* 矩阵的 *HMC* 辅助变量的分布。因为对于任何 *Leapfrog* 步骤 $t \geq 1$，$(Aq_t,(A^\intercal)^{-1}p_t) = (q'_t,p'_t)$，所以矩阵 Σ，设 A 是可逆矩阵。在 (q_0,p_0) 初始化的 (q,p) 的 *HMC* 动态等价于 $(q',p') = (Aq,(A^\intercal)^{-1}p)$ 的 *HMC* 动态初始化为 $(q'_0,p'_0) = (Aq_0,(A^\intercal)^{-1}p_0)$。

证明 9.5.1 设 $(q',p') = (Aq,(A^\intercal)^{-1}p)$。改变概率密度的变量公式，$P'(q') = P(q)/|det(A)|$，并且由于 A 是常数，新分母被吸收到归一化常数，所以 q 和 q 的能量函数只有一个加性常数不同：$U'(q') = U(A^{-1}q') + c$。使用链规则，

$$\frac{\partial U'}{\partial q}(q^*) = (A^\intercal)^{-1}\frac{\partial U}{\partial q}(A^{-1}q^*)$$

对于任何向量 q^*。变换后的动量有一个分布 $p' \sim N(0,(A^\intercal)^{-1}\Sigma A^{-1})$ 和能量函数 $K'(p') = A\Sigma^{-1}A^\intercal p'$ 的一次 *Leapfrog* 更新由下式给出

$$p'_{1/2} = p'_0 - \frac{\varepsilon}{2}\frac{\partial U'}{\partial q}(q'_0) = (A^\intercal)^{-1}p_0 - \frac{\varepsilon}{2}(A^\intercal)^{-1}\frac{\partial U}{\partial q}(q_0) \tag{9.30}$$

$$q'_1 = q'_0 + \varepsilon A\Sigma^{-1}A^\intercal p'_{1/2} = Aq_0 - \frac{\varepsilon^2}{2}A\Sigma^{-1}\frac{\partial U}{\partial q}(q_0) + \varepsilon A\Sigma^{-1}p_0 \tag{9.31}$$

$$p'_1 = p'_{1/2} - \frac{\varepsilon}{2}\frac{\partial U'}{\partial q'}(q'_1) = (A^\intercal)^{-1}p_0 - \frac{\varepsilon}{2}(A^\intercal)^{-1}\frac{\partial U}{\partial q}(q_0) - \frac{\varepsilon}{2}(A^\intercal)^{-1}\frac{\partial U}{\partial q}(A^{-1}q'_1). \tag{9.32}$$

将（*9.30*）和（*9.32*）乘以 $A^\intercal$ 并乘以（*9.31*）A^{-1} 给出原始对 (q_0,p_0) 的 *Leapfrog* 更新，很明显 $(q'_1,p'_1) = (Aq_1,(A^\intercal)^{-1}p_1)$。通过归纳，这种关系必须适用于任何数量的 *Leapfrog* 步骤。

备注: 在实践中，引理 9.1中的等价是不准确的，这是用 A 执行的矩阵运算引起的计算不准确引起的。但是，如果 A 条件良好，两个链的数值实现应该可以给出非常相似的结果。

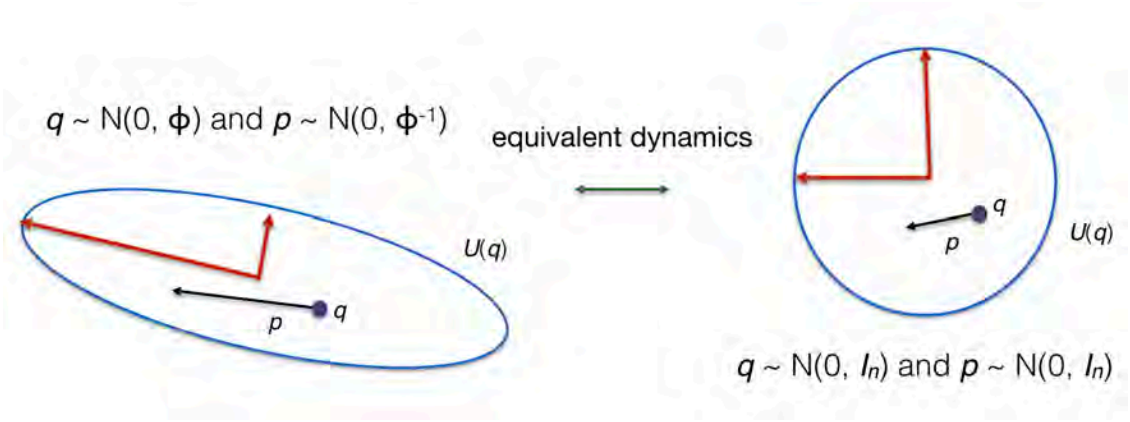


图 9.7: 引理 9.1的可视化。当 q 具有协方差 Φ 时，具有动量 $p \sim \mathrm{N}(0,\Phi^{-1})$ 的 HMC 动态等同于理想的 HMC 动态，其中 q 和 p 都具有各向同性协方差。这表明 HMC 中理想的动量协方差是当前位置 q 的 U 的局部协方差的倒数。

这个引理提供了关于调整 HMC 的关键见解。假设 q 的分布在 p.d. 协方差 Σ_{q^*} 中 q^* 点附近的某个区域近似为高斯分布。协方差可以通过几种可能的方法（例如 Choelsky 或特征值分解）分解为 $\Sigma_{q^*} = AA^\intercal$。

考虑 $p \sim \mathrm{N}(0, \Sigma_{q^*}^{-1})$ 中 (q,p) 链的 HMC 动态。因为 Lemma 引理 9.1，(q,p) 的动态等价于 $(A^{-1}q, A^{\intercal}p)$ 的动态。现在 $A^{\intercal}p \sim \mathrm{N}(0, I_n)$ 并且在 q^* 附近的区域 $\mathrm{Var}(A^{-1}q) = I_n$，因此，变换后的位置和动量变量近似独立，每个维度的方差为 1。由于变换后的空间很容易采样，因此 $(A^{-1}q, A^{\intercal}p)$ 的 HMC 动力学应该在少数 Leapfrog 步骤中导致几乎独立的状态（甚至 $L = 1$），相同的采样属性适用于原始 (q,p) 的等效动态。

与 RMHMC 相同，这种观察是使用局部曲率信息来改善 HMC 动力学性能的几种动机之一。设 q^* 成为能源领域的一个位置，假设 $\frac{\partial U^2}{\partial^2 q}(q^*)$ 是正定的，所以 $\Sigma_{q^*} = [\frac{\partial U^2}{\partial^2 q}(q^*)]^{-1}$ 给出了 q^* 邻域内的局部相关和缩放结构 U。一般来说，$\frac{\partial U^2}{\partial^2 q}(q^*)$ 可能不是正定的，但是通过对 $\frac{\partial U^2}{\partial^2 q}(q^*)$ 和反转的特征值进行阈值处理获得的 p.d. 相对 Σ'_{q^*} 可以提供相同的好处。

通过上面的讨论，使用动量 $p \sim \mathrm{N}(0, \Sigma_{q^*}^{-1})$ 应该导致少量的 Leapfrog 步骤中 q-space 中的一个几乎独立的提议，所以 $\Sigma_{q^*}^{-1}$ 是 q^* 点的理想提议协方差。如章节 9.4.3中所述，使用 $\Sigma_{q^*}^{-1}$ 作为动量的协方差促进沿局部流形的运动并允许链条沿着能量盆底部的水平曲线移动，单独使用梯度信息是不可能的。为了使 HMC 成为有效的采样方法，而不仅仅是优化方法，有必要使用信息动量协方差。如果可以获得边际标准差的估计 s_i，则对角协方差 Λ，其中 $\lambda_i = 1/s_i^2$ 可以解释变量之间的比例差异。但是 s_i 可能会在整个状态空间内变化，而对角线协方差 Λ 无法解释维度之间的相关性，这在现实世界问题中通常很强。

$p \sim \mathrm{N}(0, \Sigma)$. 下面的引理给出了三个等价的方法来实现一个可逆矩阵 C 的链 (q, Cp) 的 HMC 动力学，其中原始动量分布是 $p \sim \mathrm{N}(0, \Sigma)$。在 RMHMC 中 $\Sigma = I_n$ and $C = \sqrt{\partial U^2 / \partial^2 q}$，假设曲率是正定的。尽管等效，但这些实现的计算成本可以根据所需的矩阵分解和反转而变化。当使用大型矩阵在高维度工作时，矩阵操作的成本迅速增加，并且需要注意确保链在合理的时间内更新。

Lemma 9.2 *设 $U(q)$ 成为一个平滑的能量函数，Σ 是 p.d. 矩阵，C 是一个可逆矩阵。以下 HMC 链的动态是等效的：*

1. *动量从 $p \sim N(0, C^{\intercal}\Sigma C)$ 中采样，并根据 (q,p) 的标准 HMC 动态更新链。*

2. *动量从 $p \sim N(0, \Sigma)$ 中采样，并根据 $(q, C^{\intercal}p)$ 的标准 HMC 动态更新链，即*

3. *动量从 $p \sim N(0, \Sigma)$ 中采样，并根据由此定义的改变的 HMC 动态更新链。*

$$\frac{dq}{dt} = C^{-1}\frac{\partial K}{\partial p}, \tag{9.33}$$

$$\frac{dp}{dt} = -[C^{-1}]^{\intercal}\frac{\partial U}{\partial q}. \tag{9.34}$$

此外，（2）和（3）都可以使用改变的 Leapfrog 更新来实现

$$p_{t+1/2} = p_t - \frac{\varepsilon}{2}[C^{-1}]^{\intercal}\frac{\partial U}{\partial q}(q_t) \tag{9.35}$$

$$q_{t+1} = q_t + \varepsilon C^{-1}\Sigma^{-1}p_{t+1/2} \tag{9.36}$$

$$p_{t+1} = p_{t+1/2} - \frac{\varepsilon}{2}[C^{-1}]^{\intercal}\frac{\partial U}{\partial q}(q_{t+1}). \tag{9.37}$$

证明 9.5.2 *考虑（2）的动态。采样 $p_0 \sim N(0,\Sigma)$，并且让 $p_0' = C^\intercal p_0$，这意味着 p_0' 分配 $N(0, C^\intercal \Sigma C)$ 的 Leapfrog 更新由下式给出*

$$p'_{t+1/2} = p'_t - \frac{\varepsilon}{2}\frac{\partial U}{\partial q}(q_t) \tag{9.38}$$

$$q_{t+1} = q_t + \varepsilon C^{-1}\Sigma^{-1}[C^{-1}]^\intercal p'_{t+1/2} \tag{9.39}$$

$$p'_{t+1} = p'_{t+1/2} - \frac{\varepsilon}{2}\frac{\partial U}{\partial q}(q_{t+1}) \tag{9.40}$$

这与（1）的标准 Leapfrog 更新相同，证明了（1）和（2）之间的等价性。另一方面，将（9.38）和（9.40）乘以 $[C^{-1}]^\intercal$ 给出与（9.35）通过（9.37），因为在每一步 t 中 $p_t = [C^{-1}]^\intercal p'_t$。从（9.35）到（9.37）的更新很容易被识别为改变后的汉密顿方程（9.33）和（9.34）的跳跃动力学，它们表明（2）和（3）等价。

上述引理很重要，有两个原因。首先，它表明在 $M = C^\intercal C$ 时 $p \sim \mathrm{N}(0, M)$ 的 HMC 动态可以解释为由动量 $p \sim \mathrm{N}(0, I_n)$ 产生的 HMC 动态 $p \sim \mathrm{N}(0, I_n)$，它在（9.33）和（9.34）中改变了 Hamilton 方程的形式。这提供了 RMHMC 与其他“偏斜对称”HMC 方法之间的重要联系，这些方法以类似的方式改变 Hamilton 方程，其中最重要的是随机梯度 HMC。

其次，引理提供了一种只需要计算 $\sqrt{M^{-1}}$，而不是 M 本身 de 替代方法来实现 $p \sim \mathrm{N}(0, M)$ 的动态。这是因为让 $\Sigma = I_n$，$C = \sqrt{M}$，并观察到（9.35）到（9.37）的更新只需要 C^{-1}。理想的动量协方差是 $\frac{\partial U^2}{\partial^2 q}$，并且在凸区域中 $\sqrt{[\frac{\partial U^2}{\partial^2 q}]^{-1}}$ 可以使用 LBFGS 算法的变体从局部位置的样本近似。该计算不需要矩阵求逆或分解，并且提供了实现近似 RMHMC 算法的计算上有效的方式，这会在稍后讨论。尽管如此，在复杂景观中获得根逆 Hessian 的准确估计是所有 RMHMC 实施的重大障碍。

9.5.2 RMHMC 动态

适应局部曲率的线性变换具有明显的理论优势，并且在少数 Leapfrog 步骤中允许几乎独立的采样。然而，包括局部曲率信息的 HMC 动力学比标准 HMC 动力学更难以离散化，并且需代价很高的的计算方法。

在标准 HMC 中，相同的矩阵 Σ 用作整个单一提案中的动量的协方差。在 RMHMC 中，动量协方差 $\Sigma(q)$ 依赖于能源领域的当前位置 q。目前，$\Sigma(q)$ 是 q 的返回 p.d. 对称矩阵的任何平滑矩阵函数，但实际上这个矩阵应该反映能量景观中位置 q 附近的局部曲率。有关 $\Sigma(q)$ 在实践中的选择的讨论可以在本节后面找到。RMHMC 动量分布为 $p \sim \mathrm{N}(0, \Sigma(q))$，具有能量函数

$$K(q,p) = \frac{1}{2}\log((2\pi)^n|\Sigma(q)|) + \frac{1}{2}p^\intercal\Sigma(q)^{-1}p, \tag{9.41}$$

联合汉密尔顿是

$$H(q,p) = U(q) + \frac{1}{2}\log((2\pi)^n|\Sigma(q)|) + \frac{1}{2}p^\intercal\Sigma(q)^{-1}p. \tag{9.42}$$

动量能量函数必须包含在标准 HMC 中找不到的 $\frac{1}{2}\log((2\pi)^n|\Sigma(q)|)$ 额外的项，并且评估此术语的衍生物

是计算难度的来源。观察到

$$\int_{\mathbb{R}^n} \frac{1}{Z} e^{-H(q,p)} dp = \frac{1}{Z} e^{-U(q)} \int_{\mathbb{R}^n} \frac{1}{\sqrt{(2\pi)^n |\Sigma(q)|}} e^{-p^\intercal \Sigma(q)^{-1} p} dp = \frac{1}{Z} e^{-U(q)}, \tag{9.43}$$

因此 q 的边际分布是目标分布，并且通过更新 (q,p) 获得的 q-采样将像在标准 HMC 中一样遵循正确的分布。控制 RMHMC 的汉密尔顿方程是

$$\frac{dq}{dt} = \frac{\partial H}{\partial p} = \Sigma(q)^{-1} p, \tag{9.44}$$

$$\frac{dp}{dt} = -\frac{\partial H}{\partial q} = -\frac{\partial U}{\partial q} - \frac{1}{2}\mathrm{Tr}\left[\Sigma(q)^{-1}\frac{\partial \Sigma(q)}{\partial q}\right] + \frac{1}{2} p^\intercal \Sigma(q)^{-1} \frac{\partial \Sigma(q)}{\partial q} \Sigma(q)^{-1} p. \tag{9.45}$$

q 和 p 的更新不再可分，因为 p 的更新取决于 q 和当前的 p。因此，如果使用 Leapfrog 积分器，坐标变化将不再具有切变结构，因此无法保证 Leapfrog 坐标变化对 RMHMC 动态具有决定因素 1。这扰乱了 HMC 的详细平衡，并且在 RMHMC 设置中简单地实现 Leapfrog 集成器并不会保留 $\frac{1}{Z}e^{-U(q)}$ 的分布。

为了克服 RMHMC 更新方程的不可分离性，更新值由必须使用定点迭代求解的隐式方程组定义。用于离散不可分离关节 Hamiltonian$H$ 的动力学的广义 Leapfrog 积分器的一次迭代由下式给出：

$$p_{t+1/2} = p_t - \frac{\varepsilon}{2}\frac{\partial H}{\partial q}(q_t, p_{t+1/2}), \tag{9.46}$$

$$q_{t+1/2} = q_t + \frac{\varepsilon}{2}\left[\frac{\partial H}{\partial p}(q_t, p_{t+1/2}) + \frac{\partial H}{\partial p}(q_{t+1}, p_{t+1/2})\right], \tag{9.47}$$

$$p_{t+1} = p_{t+1/2} - \frac{\varepsilon}{2}\frac{\partial H}{\partial q}(q_t, p_{t+1/2}). \tag{9.48}$$

隐含地定义了前两个步骤中的更新，允许模拟不可分离的 H 的动态。在标准 HMC 的情况下，$H(q,p) = U(q) + K(p)$ 并且广义 Leapfrog 更新与标准 Leapfrog 方案相同。当 H 不可分时，必须使用定点迭代来求解（9.46）和（9.47）。固定点更新的详细信息在本节后面的 RMHMC 算法中。

可以看出，广义 Leapfrog 更新是保持体积的，RMHMC 保持详细的平衡并保留目标分布。该证明类似于标准 HMC 的详细平衡证明，并根据广义 Leapfrog 更新对体积保存和可逆性的证明进行了适当调整。

9.5.3 RMHMC 算法和变体

RMHMC 算法有几种变体。首先是完整的 RMHMC 算法，该算法需要定点迭代和 $\Sigma(q)$ 的导数的计算。由于 $\Sigma(q)$ 在实践中是局部曲率，因此完整的 RMHMC 需要计算目标能量 U 的三阶导数，这是一个很大的计算负担，在许多实际情况下是不可能的。$L = 1$Leapfrog 更新有一个 RMHMC 变体，它需要在原始状态和建议状态下计算 U 的三阶导数，但不涉及定点迭代。该算法的细节与完整的 RMHMC 略有不同，感兴趣的读者可以参考 [3]。

完整的 RMHMC 算法

输入: 可微能量函数 $U(q)$, 初始 $q_0 \in \mathbb{R}^n$, $n \times n$ 可微 p.d. 协方差 $\Sigma(q)$, 步长 ε, 迭代次数 N,Leapfrog 步数 L, 固定点步数 K

输出: 固定分配 U 的马尔可夫链样本 $\{q_1, \ldots, q_N\}$

对于 $i = 1:N$,

1. 产生动量 $p_{i-1} \sim \mathrm{N}(0, \Sigma(q_{i-1}))$.

2. Let $(q'_0, p'_0) = (q_{i-1}, p_{i-1})$. 对于 $l = 1:L$, 使用广义 Leapfrog 积分器达到提议状态 $(q^*, p^*) = (q'_L, p'_L)$ 来更新，过程如下:

   (a) 设 $\hat{p}_0 = p'_{l-1}$. 对于 $k = 1:K$, 根据不动点方程更新 $\hat{p}_{k-1}$

   $$\hat{p}_k = \hat{p}_{k-1} - \frac{\varepsilon}{2} \Sigma(q'_{l-1})^{-1} \hat{p}_{k-1} \tag{9.49}$$

   来获得半步动量更新 $p'_{l-1/2} = \hat{p}_K$.

   (b) Let $\hat{q}_0 = q'_{l-1}$. 对于 $k = 1:K$, 根据定点方程更新 $\hat{q}_{k-1}$

   $$\hat{p}_k = \hat{p}_{k-1} - \frac{\varepsilon}{2} \Sigma(q'_{l-1})^{-1} \hat{p}_{k-1} \tag{9.50}$$

   其中 $\partial H / \partial p$ 在（9.45）中给出，以获得全步位置更新 $q'_l = \hat{q}_K$。

   (c) 根据如下公式更新 $p'_{l-1/2}$

   $$p'_l = p'_{l-1/2} - \frac{\varepsilon}{2} \Sigma(q'_l)^{-1} p'_{l-1/2} \tag{9.51}$$

   来获得全步动力更新 p'_l。

3. 根据 Metropolis-Hastings 接受概率接受建议的状态 (q^*, p^*)

   $$\alpha = \min\left(1, \exp\{-H(q^*, p^*) + H(q_{i-1}, p_{i-1})\}\right) \tag{9.52}$$

   其中 $H(q,p)$ 是（9.42）的联合汉密尔顿分布。如果提议被接受，那么 $q_i = q^*$。否则，$q_i = q_{i-1}$。提案后可以丢弃动量 p_{i-1}。

备注: 步长 ε 是一个“无量纲”数量，因为 RMHMC 动态应该在本地对应于普通的 HMC 分布，其中 q 和 p 都是 $\mathrm{N}(0 I_n)$。RMHMC 的规模隐含地是标准正态尺度，因此将 ε 设置为略小于 1 的值，最小（和最大）重新缩放的标准偏差应该对任何 RMHMC 算法都能产生良好的结果。

为了减轻完整 RMHMC 的困难，可以使用近似 RMHMC 算法，其中动量采样之前链的先前状态可以决定协方差 $\Sigma(q_{t-J+1}, q_{t-J+2}, \ldots, q_t)$，但在整个 Leapfrog 更新中是固定的。此变体本质上是标准 HMC

算法，具有在提议间更改 Σ 的原理方式。

在提案之间更改 Σ 并不与保留目标分配矛盾，因为如果 q 在使用协方差 $Sigma_0$ 更新后具有正确的分布，则在具有任何协方差 $Sigma_1$ 的 HMC 更新后，q 仍将遵循正确的分布，不一定等于 $Sigma_0$。起初可能看起来使用先前的状态来获得 $\Sigma(q_{t-J+1}, q_{t-J+2}, \ldots, q_t)$ 可能违反了 HMC 的马尔可夫结构，但事实并非如此，因为任何 $\Sigma(x_1, \ldots, x_J)$ 都有详细的余量，特别是 $(x_1, \ldots, x_J)$ 不需要具有目标分布。

虽然简化的 RMHMC 算法无法捕捉到 RMHMC 动态所暗示的完全依赖性，因为 $Sigmaq$ 没有通过 Leapfrog 迭代更新，所以它在计算上与标准 HMC 相同，并且通过使用逆 Hessian 的拟牛顿估计提供了合并曲率信息的有效方式。即使这些信息是近似的，它仍然可以显着改善链条在能源领域的运动。

简化的 RMHMC 算法

输入: 可微能量函数 $U(q)$, 初始状态 $q_0 \in \mathbb{R}^n$, $n \times n$ p.d. 协方差函数 $\Sigma(q_1, \ldots, q_J)$, 步长 ε, 迭代次数 N, 内存中先前状态的数量 J

输出: 固定分配 U 的马尔可夫链样本 $\{q_1, \ldots, q_N\}$，

对于 $i = 1 : N$,

1. 计算当前协方差矩阵 $\Sigma^* = \Sigma(q_{i-1-J}, q_{i-J}, \ldots, q_{i-1})$.
2. 根据使用提议协方差 Σ^* 的标准 HMC 动态更新 $q_{i-1} p_{i-1}$。

注释: 由于 Σ^* 在整个更新过程中是固定的，因此在简化的 RMHMC 算法中只使用少量的 Leapfrog 步骤（通常为 $L = 1$），这是因为局部曲率随每次更新而变化。

9.5.4 RMHMC 中的协方差函数

RMHMC 算法及其简化保留了任何可微分的 p.d. 协方差函数 $\Sigma(q)$ 的目标，但要实际改进采样 $\Sigma(q)$ 必须反映空间的局部曲率。一般来说，$\partial^2 U / \partial q^2$ 不一定是 pd，所以简单地的选择 $\Sigma(q) = \frac{\partial U^2}{\partial q^2}(q)$ 在实践中有时是不可行的。

RMHMC 的原始作者将注意力限制在一个概率模型 $p(\theta|X)$ 族的后验概率 $p(X|\theta)$ 中。在这种情况下，Fisher 信息提供的提议协方差有一个自然的选择

$$\Sigma(\theta) = -\mathrm{E}_{X|\theta}\left[\frac{\partial^2}{\partial \theta^2} \log p(X|\theta)\right] \tag{9.53}$$

这肯定是正定的。在简单的情况下，Fisher 信息可以通过分析获得。如果这是不可能的，则可以通过对观察到的数据取曲率的期望并对所得矩阵的特征值进行阈值处理来估计稳定性。Fisher 信息的正定结构在理论上是一个很好的性质，但在实践中必须估计矩阵，而且仍然可以遇到非常小甚至负的特征值。当使用 Fisher 信息时，Langevin 方程中出现的梯度项 $\Sigma(\theta)^{-1} \frac{\partial U}{\partial \theta}(\theta)$ 对应于 Amari 等人的自然梯度度 [1].

Fisher 信息不是一个完整的解决方案，因为它只能在为具有一组观测数据 X 的分布系列采样参数 θ 时使用。当仅从概率分布 $P(q) = \frac{1}{Z} e^{-U(q)}$ 进行采样时，没有办法通过期望来制作曲率 p.d.。然而 $\frac{\partial U}{\partial q}$ 的最大特征值应该是 HMC 动力学中最重要的。这是因为曲率的最大特征值表示分布的最受约束的线性维度。当接近局部最小值时，0 附近的负特征值或特征值不成问题，因为在这些方向上的移动使 H 近似恒

定或减少 H。阈值 $\frac{\partial U}{\partial q}$ 的特征值可以给出当曲率本身不是 p.d 时保留最重要的局部几何信息的 p.d. 协方差。

另一个选择是像在准牛顿方法一样估计局部曲率 $\Sigma(q)$。这种类型的方法使用过去状态序列 $q_{t+1-J}, q_{t+2-J}, \ldots, q_t$ 来估计当前状态 q_t 处的逆 Hessian。如引理 9.2所示，只需要根逆 Hessian 来模拟 $\Sigma(q) = \frac{\partial U}{\partial q}$ 的 HMC 动态，并且存在直接估计根逆 Hessian 的 LBFGS 算法的变体。有关详细信息，请参阅 [2] 和 [9]。

9.6 实例中的 HMC

在本节中，介绍了 HMC 和相关算法的两个应用。第一个应用是高斯分布的玩具实验，除了几个方向外，它们都受到高度约束。该实验对于理解调整 HMC 参数背后的基本原理非常有用。接下来，检查 HMC 的后验取样和逻辑回归模型中的变体。因为 Fisher 信息以封闭形式提供，所以此设置是在实践中可以实现完整 RMHMC 算法的少数情况之一。Logistic 回归是不同 HMC 模型之间直接比较的良好设置。最后，提出了交替反向传播算法，该算法使用 LMC 作为从高维参数和图像定义的分布采样时的关键步骤。

9.6.1 受约束正态分布的模拟实验

在本节中，HMC 和变体用于从正常分布中进行采样，这些分布在除了几个方向之外的所有方向上都受到高度约束。对于使用基于能量景观中的当前位置的局部更新的任何 MCMC 方法，这种分布是具有挑战性的，因为难以有效地采样无约束的维度，同时仍然保持在景观的紧密约束区域中。

考虑两种不同的分布：$\mathrm{N}(0,\Sigma_1)$ 和 $\mathrm{N}(0,\Sigma_2)$。Σ_1 和 Σ_2 均为 100×100 对角矩阵。Σ_1 和 Σ_2 的前 15 个条目为 1，表示无约束的采样方向。Σ_1 的最后 85 个条目是 0.01^2，Σ_2 的最后 85 个条目是 0.0001^2。Σ_1 是一个更容易采样的情况，因为最大和最小标准偏差之间的比率大约为 100，因此使用标准 HMC 进行有效采样时需要大约 100 个 Leapfrog 步骤。Σ_2 是一个更困难的情况，因为最大和最小标准差之间的比率是 10,000。当局部协方差在尺度上表现出如此极大的差异时，HMC 不再是一种有效的采样方法，因为当使用非常大量的步骤时，Leapfrog 近似变得非常不稳定，如可从章节**??**中的澳大利亚信用数据中看到的那样。这可以通过用准牛顿 HMC 变体包括近似曲率信息来解决。所有实验都使用 5000 次老化迭代和 10,000 次采样迭代。

两个目标分布的能量函数的形式为 $U_1(q) = \frac{1}{2}q^\intercal\Sigma_1^{-1}p$ 和 $U_2 = \frac{1}{2}q^\intercal\Sigma_2^{-1}q$。与所有正态分布一样，目标分布具有恒定曲率 Σ_1^{-1} and Σ_2^{-1}。完整的 RMHMC 算法可以通过简单地为每次更新提供 $p\sim\mathrm{N}(0,\Sigma_1^{-1})$ 或 $p\sim\mathrm{N}(0,\Sigma_2^{-1})$ 来实现，又因为曲率 $\Sigma(q)$ 在整个状态空间中是恒定的，所以也可以使用标准动态，因此 $\Sigma(q)$ 的导数为 0 且广义 Leapfrog 更新变得与标准 Leapfrog 更新相同。在该实验中，仅具有一个 Leapfrog 更新的 RMHMC 算法可以在每次迭代中从目标分布获得几乎独立的样本。

考虑目标分布 $\mathrm{N}(0,\Sigma_1)$。使用三种不同的方法来对此分布进行采样：随机游走 Metropolis Hastings，LMC 和 HMC，L= 150 Leapfrog 更新。HMC 和 LMC 的动量协方差设定为 I_{100}。RW Metropolis 使用了 0.04 的步长，LMC 和 HMC 使用了 0.008 的步长，因为 0.008 略小于 0.01 的最小边际标准差。前两种方法无法有效地对目标分布进行采样，因为这两种方法都将被迫以小步长随机游走来探索分布的无约束方向。LMC 在这种情况下遇到困难，因为在单次更新后动态瞬间被刷新，而 HMC 使用相同的动量进行大量更新，因此 HMC 就像其他两种方法一样，没有通过随机漫游来实现探索分布。由于 HMC 的

$\varepsilon L = 1.5$，并且目标分布的最大边际标准差为 1，因此 HMC 在每次迭代中都获得几乎独立的样本。为了公平比较这些方法，150 RW Metropolis 和 LMC 更新在下图中计为单次迭代。

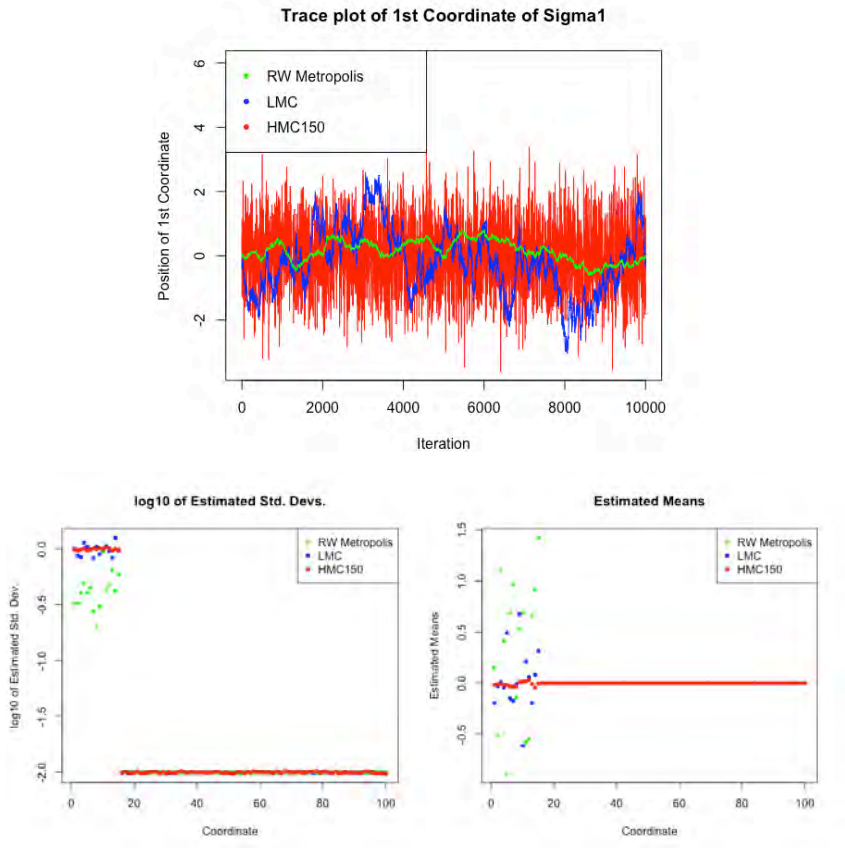


图 9.8: 使用 Σ_1 进行模拟研究。采样器的性能在约束维度上是相似的，即使这些方法中的每一个的 150 次迭代被计为图中的单个更新，但 RW Metropolis 和 LMC 仍难以有效地对无约束维度进行采样。另一方面，通过在大量跨越迭代中重复使用相同的动量，HMC 在样本空间中移动的能力足以进行非常有效的采样。

接下来，考虑目标分布 $\mathrm{N}(0,\Sigma_2)$。该分布的最大和最小标准偏差之间的比率为 10,000，因此在标准 HMC 中将需要大约 10,000 个 Leapfrog 步骤，其具有身份协方差以获得每次 HMC 更新的独立样本。即使在相当规则的景观中，例如来自 9.6.2部分的逻辑回归景观，在超过几百个步骤之后 Leapfrog 近似的准确性会降低，在实际问题中，通过使用标准 HMC 使用 $L = 10,000$ Leapfrog 更新来补偿目标分布中的比例差异是很简单的。为了有效地进行抽样，有必要考虑二阶信息。

假设真实位置协方差 Σ_2 是未知的并且不能直接计算，这在大多数实际情况中都是如此。在这种情况下，仍然可以从先前采样的位置估计 Σ_2，并且这种近似信息仍然可以促进相当有效的采样。首先，使用标准 HMC 对 40 个位置进行采样，其中 $\varepsilon = 0.000075$ 和动量协方差 I_100。获得一些初始点后，使用 LBFGS 估计 Σ_2 实现简化的 RMHMC 算法。使用过去的 40 个采样位置从初始矩阵 $H_0 = \gamma I_{100}$ 开始 LBFGS 递归。

不幸的是，原始 LBFGS 估计不能显着改善采样，因为真正的矩阵 Σ_2 太大而不能仅使用 40 个点进

行精确估计。然而，在分解和调整之后可以获得有用的近似。当从 LBFGS 估计中观察特征值时，观察到最小特征值的估计非常接近 0.0001^2 的真实值，并且 LBFGS 估计可以识别具有大特征值的几个无约束方向。对于最大特征值的估计往往非常不准确，并且在很大程度上取决于所选择的 γ 的值。在最大和最小特征值之间，剩余的特征值大部分由 LBFGS 保持不变并保持 γ，因为正在使用非常少量的数据来估计非常大的矩阵，所以这并不令人感到意外。

虽然真正的协方差 Σ_2 是未知的，但在某些情况下，假设只有 Σ_2 的前几个最大特征值很重要，并且大多数其他特征值接近 0 是合理的。在高维空间中对低维流形进行采样时会出现这种情况，这在采样在图像或其他复杂数据结构上定义的分布时很常见。最大特征值对应于目标分布中相对较少数量的无约束维度。给定关于特征值的局部结构的一些知识，可以调整原始 LBFGS 估计以提供更有用的协方差。

设 H^* 是从过去的 $J = 40$HMC 样本中获得的估计的原始 LBFGS。设 $U\Lambda U^\intercal$ 是 H^* 的对称特征值分解，其中 Λ 是一个对角矩阵，特征值 λ_1, ..., λ_{100} 按降序排序。设 $\lambda_i^* = \lambda_{100}$ for $i = K+1, \ldots, 100$ 用于某些参数 K，这是目标分布中无约束方向的估计数量。K 的真实值是 15，但在实验中使用保守估计 $K = 10$。λ_i^* 的第一个 K 等于原始值。然后动量协方差由 $\Sigma^* = U\Lambda^* U^\intercal$ 给出。理论上，对于任何 RMHMC 方法，ε 应设置为略小于 1。但是，由于最大标准偏差的估计值不准确（往往太大），因此应设置 ε 以使 $\varepsilon\lambda_1 \approx 1$。$\gamma$ 的值对采样没有太大影响，只要相应地设置 ε，从 0.000001 到 1 的值给出大致相同的结果。只需 $L = 1$ Leapfrog 步骤就可以获得良好的效果。在实现过程中使用了引理 9.2的第三种方法，其中 $\sqrt{C}^{-1} = U(\Lambda^*)^{1/2}U^\intercal$。

有两种方法用于采样 Σ_2：具有 $L = 150$ 的 HMC，以及上述简化的 RMHMC 算法。使用准牛顿信息的简化 RMHMC 算法优于标准 HMC，只需 $L = 1$ Leapfrog 步骤。在简化的 RMHMC 算法的每次迭代中所需的特征值分解计算代价很大，但是对于良好的结果是必要的，因为 LBFGS 估计根本无法估计具有如此有限数量的数据的 Σ_2 的所有真特征值，因此需要对特征值进行一些调整。

9.6.2 使用 RMHMC 对逻辑回归系数进行抽样

通过逻辑回归定义的分布的采样系数是应用完整 RMHC 方法的理想情况，因为状态空间中任何点处的 Fisher 信息可以以闭合形式给出。在许多实际情况中（例如，L_1-正则化回归），情况并非如此。给定 $N \times P$ 观察 X（每行给出一个案例）和二进制 0 或 1 响应 Y 和正则化系数 λ（视为给定常数），P-length 系数 β 的能量由下式给出出

$$U(\beta) = -\log[L(X,Y|\beta,\lambda)p(\beta|\lambda)] = -\beta^\intercal X^\intercal Y + \sum_{j=1}^{N} \log(1 + e^{\beta^\intercal X_n^\intercal}) + \frac{\lambda}{2}\beta^\intercal\beta \tag{9.54}$$

其中 X_n 是矩阵 X 的行 n。能量函数的导数是

$$\frac{dU}{d\beta}(\beta) = -X^\intercal Y + X^\intercal S + \lambda\beta \tag{9.55}$$

S 是一个长度为 P 的向量，其中 $S_n = \sigma(\beta^\intercal X_n^\intercal)$,$\sigma(*)$ 是 sigmoid 函数，并且有 Fisher 信息

$$I(\beta) = \mathrm{E}_{Y|X,\beta,\lambda}\left[\frac{d^2U}{d\beta^2}(\beta)\right] = X^\intercal\Lambda X + \lambda I \tag{9.56}$$

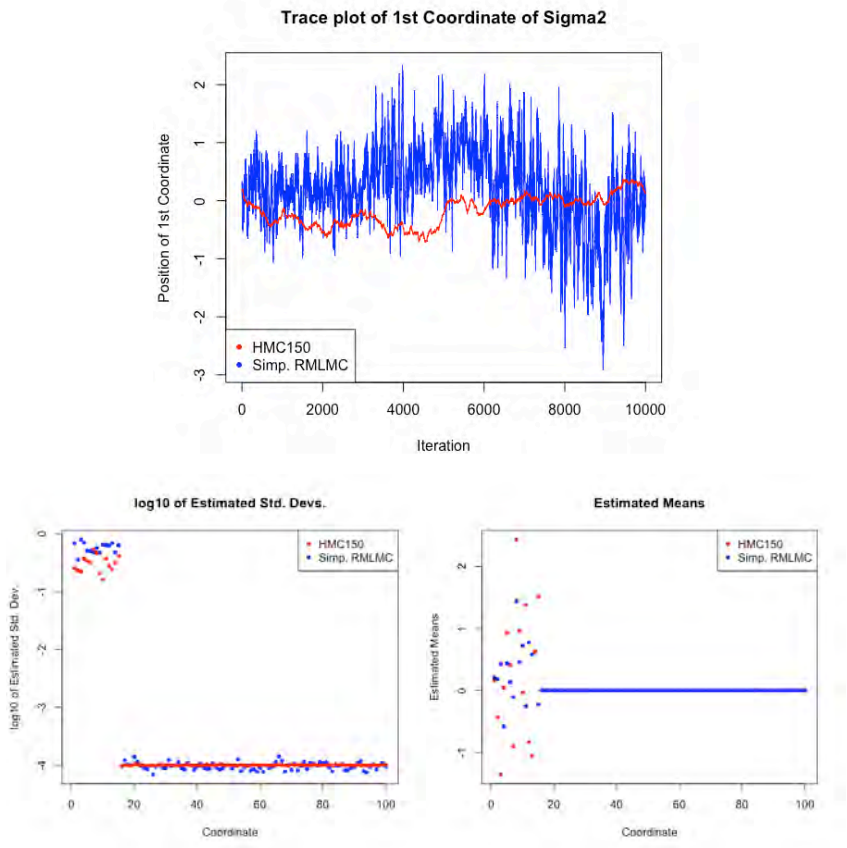


图 9.9: 使用 Σ_2 进行模拟研究。$L = 150$ 美元的 HMC 跳跃式更新无法再在无约束的维度中进行有效采样。然而，使用具有 $L = 1$ 的简化 RMHMC 算法和如上所述计算的协方差可以更有效地进行采样。

其中 Λ 是 $N \times N$ 对角矩阵，元素为 $\Lambda_{n,n} = \sigma(\beta^\intercal X_n^\intercal)(1 - \sigma(\beta^\intercal X_n^\intercal))$。完整的 RMHMC 还需要 $I(\beta)$ 的衍生物，由下式给出

$$\frac{dI(\beta)}{d\beta_i} = X^\intercal \Lambda V_i X \tag{9.57}$$

其中 V_i 是对角矩阵，元素为 $V_{i,(n,n)} = (1 - 2\sigma(\beta^\intercal X_n^\intercal))X_{n,i}$。

以下是 Giorlami 和 Calderhead 在 [3] 中进行的一项研究的结果，该研究比较了采样回归系数时 RMHMC，传统 HMC 和其他常用方法的表现。作者使用了 6 个不同的数据集和二元响应，并提供了各种大小的矩阵，这里给出了 4 个数据集的结果。我们给出了作者研究的 6 种采样算法的结果：组件式 Metropolis-Hastings，LMC，HMC，RMHMC，RMLMC 和简化的 RMLMC。

对于 LMC 和 HMC 采样器，使用动量协方差 $\Sigma = I_n$。对于所有采样器，设置步长 ε，使接受率约为 70%。RMHMC 和 HMC 的步长 L 设置为 $\varepsilon L \approx 3$，略大于最大边际标准差，因此每次 HMC 迭代应获得大致独立的样本。逻辑回归定义的哈密顿动力学表现相对较好，所以 L 可以设置得相当大（几百）而接受率没有显着下降。对于 RMHMC，越级步骤 L 的数量通常相对较小，因为每次越级更新都可以获得几

乎独立的点数。对于 HMC，需要大量的越级更新来补偿最小和最大边际标准偏差之间的比例差异。简化的 RMLMC 对应于简化的 RMHMC 算法，其中 $\Sigma(\beta_t) = I(\beta_t)$，当前点的 Fisher 信息（$J = 1$）和 $L = 1$ Leapfrog 更新。RMLMC 是 RMHMC 算法的略微变体，$L = 1$。有关详细信息，请参阅 [3]。

所有回归模型都包含一个截距，因此系数 P 的数量比数据矩阵的列数多一个。每次采样运行包括 5000 次老化迭代和 5000 次采样迭代，并且每种采样方法运行 10 次试验。

Pima 印度数据集, $N = 532$, $P = 8$

Method	Time (sec)	ESS(Min,Med,Max)	s/ESS(Min)	Rel. Speed
Metropolis	4.1	(14, 37, 201)	0.29	× 1.9
LMC	1.63	(3, 10, 39)	0.54	× 1
HMC	1499.1	(3149,3657,3941)	0.48	× 1.1
RMLMC	4.4	(1124,1266,1409)	0.0039	× 138
Simp. RMLMC	1.9	(1022,1185,1312)	0.0019	× 284
RMHMC	50.9	(5000,5000,5000)	0.01	× 54

澳大利亚信贷数据集, $N = 690$, $P = 14$

Method	Time (sec)	ESS(Min,Med,Max)	s/ESS(Min)	Rel. Speed
Metropolis	9.1	(15, 208, 691)	0.61	× 1
LMC	No Conv.	-	-	-
HMC	No Conv.	-	-	-
RMLMC	11.8	(730, 872, 1033)	0.0162	× 37
Simp. RMLMC	2.6	(459, 598, 726)	0.0057	× 107
RMHMC	145.8	(4940,5000,5000)	0.023	× 26

德国信贷数据集, $N = 1000$, $P = 24$

Method	Time (sec)	ESS(Min,Med,Max)	s/ESS(Min)	Rel. Speed
Metropolis	20.9	(10, 82, 601)	2.09	× 1
LMC	2.7	(3, 5, 130)	0.9	× 2.6
HMC	3161.6	(2707, 4201, 5000)	1.17	× 2
RMLMC	36.2	(616, 769, 911)	0.059	× 39.6
Simp. RMLMC	4.1	(463, 611, 740)	0.0009	× 260
RMHMC	287.9	(4791, 5000,5000)	0.06	× 39

Caravan 数据集, $N = 5822$, $P = 86$

Method	Time (sec)	ESS(Min,Med,Max)	s/ESS(Min)	Rel. Speed
Metropolis	388.7	(3.8, 23.9, 804)	101.9	× 3.7
LMC	17.4	(2.8, 5.3, 17.2)	6.2	× 59
HMC	12,519	(33.8, 4032, 5000)	369.7	× 1
RMLMC	305.3	(7.5, 21.1, 50.7)	305.3	× 1.2
Simp. RMLMC	48.9	(7.5, 18.4, 44)	6.5	× 56
RMHMC	45,760	(877, 1554, 2053)	52.1	× 7.1

最大边缘标准偏差与最小边缘标准偏差之比

Dataset	Pima	Australian	German	Caravan
Ratio	225	6404	303	236

从这些实验的结果中可以得出各种有用的指导性观察。在原始作者之后，算法的速度和相对速度基于 10 次试验中的最小 ESS 给出。

虽然速度很慢，但 HMC 确实能够在 Pima，German 和 Caravan 数据集中实现 ESS 的理想 ESS 为 5000 的重要比例。然而，作者发现 HMC 和 LMC 都没有收敛到 Australian 数据集中的静态分布。

通过参考表给出最大与最小边际标准偏差的比率，可以理解这一点。回想一下，最大和最小边际标准偏差之间的比率是在单个 HMC 更新中达到独立状态所需的最小跳跃步数。在 Pima，德国和 Caravan 数据集中，这个比率大约是 200 到 300，这意味着使用一个简单的协方差矩阵 $\Sigma = I_n$ 需要 200 到 300 个越级步骤来达到一个独立的状态。然而，澳大利亚数据集在其最大和最小约束方向的长度之间的比率超过 6000，因此每个 HMC 更新需要数千个跳跃步骤以达到独立状态。Leapfrog 离散化不够准确，无法为如此大的 L 提供高接受率，LMC 采样器的更新太小，无法进行有效采样。逻辑回归格局相对良好; 在更复杂的景观中，甚至可能无法使用 200 到 300 个 Leapfrog 步骤。

值得注意的是，一般来说，LMC 的表现要比所有其他方法要差得多，除了在高维 Caravan 数据集中的中它优于所有其他方法。已经发现 LMC 是对图像模式的非常高维的生成模型进行采样的有效方法。LMC 在这种情况下是一个很有吸引力的选择，因为它可以很好地扩展并且可以快速实现。

现在考虑 RMHMC 方法。RMHMC 和 RMLMC 在数据集上具有相似的性能，表明在考虑局部曲率后 RMHMC 的额外采样功率由 RMLMC 的更快速度均衡平衡。简化的 RMLMC 优于所有数据集中的完整 RMHMC 实现，提供证据表明仅考虑局部曲率信息而不改变 HMC 动态可以是实现和 RMHMC 近似的有效方式。

对于较小的数据集，完整的 RMHMC 方法优于标准 HMC 方法，但对于 Caravan 数据集则相反。计算完全 RMHMC 中 Fisher 信息的导数需要在 9.56中沿对角线评估 N^2 表达式，这解释了不良的缩放。在如从高斯过程中采样特殊情况下，Fisher 信息可以具有稀疏结构，但是通常它不可用或者计算非常昂贵，并且完整的 RMHMC 不是在复杂的高维景观中采样的实际解决方案。

9.6.3 使用 LMC 采样图像密度：FRAME，GRADE 和 DeepFRAME

随机模型是一种强大而统一的图像数据表示和理解方式，近年来随机图像模型的能力大大提高。通过从一组训练图像 $\{I_k\}_{k=1}^K$ 中学习密度 $P(I)$，可以合成新颖逼真的图像，甚至可以探索图像空间的结构。但是，从图像密度 PI 中采样通常会产生问题。即使是相对较小的图像也是高维数据，因为图像尺寸与图像宽度的平方成比例。此外，图像尺寸之间存在强相关性，尤其是在附近像素之间。诸如 Metropolis-Hastings 和 Gibbs 采样之类的方法在这种实际情况下使用太慢。LMC 是克服这些困难的流行方式。

给定形式的图像密度

$$P(I) = \frac{1}{Z}\exp\{-U(I)\},$$

其中 I 是具有连续像素强度的 $n \times n$ 图像（通常每个像素强度位于有界区间内，如 [0,1]，[- 1 -1] 或

[0,255])，LMC 可用于从密度 P 获得 MCMC 样本。图像更新具有标准的 Langevin 形式

$$I_{t+1} = I_t - \frac{\varepsilon^2}{2}\frac{dU}{dI}(I_t) + \varepsilon Z_t \tag{9.58}$$

其中 $Z_t \sim \mathrm{N}(0, \sigma^2 I_{n\times n})$。渐变项 $\frac{dU}{dI}(I_t)$ 比随机行走 Metropolis-Hastings 或 Gibbs 采样更快地收敛。由于 Leapfrog 积分器在应用于复合能量 U 时不稳定，LMC 经常被用来代替 HMC 而且 $L > 1$Leapfrog 步骤。

LMC 提供了一种从学习能量中采样的强大方法，但能量形式 $U(I)$ 本身是图像建模的核心问题。函数 U 必须具有足够的表现力，以捕捉现实世界图像模式中存在的共同结构和特殊变异。特别是 U 必须捕获在观察图像中发现的像素强度之间的相关性的复杂结构，以成功地合成逼真图像。本节介绍两种图像模型的中心配方。学习参数图像密度 $P(I;\theta)$ 的方法将在 10章节中讨论。

例 9.1 ***FRAME*（过滤器，随机区域和最大熵）模型。（朱等人，*1998 [12])*** *FRAME* 模型是一系列图像密度的开创性公式，能够合成逼真的纹理。对于一组超参数 $(\mathscr{F},\Lambda)$，*FRAME* 模型的密度具有吉布斯形式

$$P(I;\mathscr{F},\Lambda) = \frac{1}{Z(\mathscr{F},\Lambda)}\exp\{-U(I;\mathscr{F},\Lambda)\}. \tag{9.59}$$

超参数 $\mathscr{F}$ 是一组预定义的滤波器 $\mathscr{F} = \{F^{(1)},\ldots,F^{(K)}\}$，它们通常是 *Gabor* 滤波器和不同大小/方向的高斯滤波器的拉普拉斯滤波器，用于测量 I 中每个空间位置的卷积响应。超参数 $\Lambda = \{\lambda^{(1)}(\cdot),\ldots,\lambda^{(K)}(\cdot)\}$ 鼓励过滤器 $F^{(i)}$ 对合成图像的响应，以匹配观察到的 $F^{(i)}$ 对训练图像的响应。通常，预先选择过滤器集 $\mathscr{F}$，并通过章节 *10*中讨论的随机梯度方法学习潜在 Λ。

FRAME 能量具有形式

$$U(I;\mathscr{F},\Lambda) = \sum_{i=1}^{K}\sum_{(x,y)\in\mathscr{L}}\lambda^{(i)}(F^{(i)} * I_{(x,y)}) \tag{9.60}$$

其中 $\mathscr{L}$ 是像素点阵。符号 $F^{(i)} * I_{(x,y)}$ 指的是过滤器 $F^{(i)}$ 的卷积和图像晶格中位置 (x,y) 的图像 I。为了简化学习过程，每个 $(x,y)\in\mathscr{L}$ 的回复 $F^{(i)} * I_{(x,y)}$ 被放入离散区间 $\{B_1^{(i)},\ldots,B_L^{(i)}\}$ 并且潜在函数 $\lambda^{(i)}(\cdot)$ 被潜在的向量 $\lambda^{(i)} = (\lambda_1^{(i)},\ldots,\lambda_L^{(i)})$ 取代。这导致 *FRAME* 能量更容易处理

$$U(I;\mathscr{F},\Lambda) = \sum_{i=1}^{K}\langle\lambda^{(i)},H^{(i)}(I)\rangle = \langle\Lambda,H(I)\rangle \tag{9.61}$$

其中 $H^{(i)}(I) = (H_1^i(I),\ldots,H_L^{(i)}(I))$ 是图像 I 的响应频率直方图，用于在所有空间位置过滤 $F^{(i)}$，$H(I) = (H^{(1)}(I),\ldots,H^{(K)}(I))$。在本节的其余部分，我们将讨论表单的 *FRAME* 能量（*9.60*），并且（*9.61*）具有相同的属性。

在一个 *FRAME* 能量 U 中检查学到的电位 $\{\lambda^{(i)}(\cdot)\}$ 在一个 *Langevin* 方程（*9.58*）的梯度项 $\frac{\partial U}{\partial I}$ 中显示出两个相互作用的力。学习潜力 $\lambda^{(i)}$ 分为两个通用的潜在类型：

$$\phi(\xi) = a\left(1 - \frac{1}{1+(|\xi-\xi_0|/b)^\gamma}\right) \quad with\ a > 0, \tag{9.62}$$

or

$$\psi(\xi) = a\left(1 - \frac{1}{1+(|\xi-\xi_0|/b)^{\gamma}}\right) \quad with\ a<0, \tag{9.63}$$

其中 ξ_0 是转换常量，b 是缩放常量，$|a|$ 是过滤器 $F^{(i)}$ 的贡献权重。这两个潜在的家族在图 *9.10*中可视化。

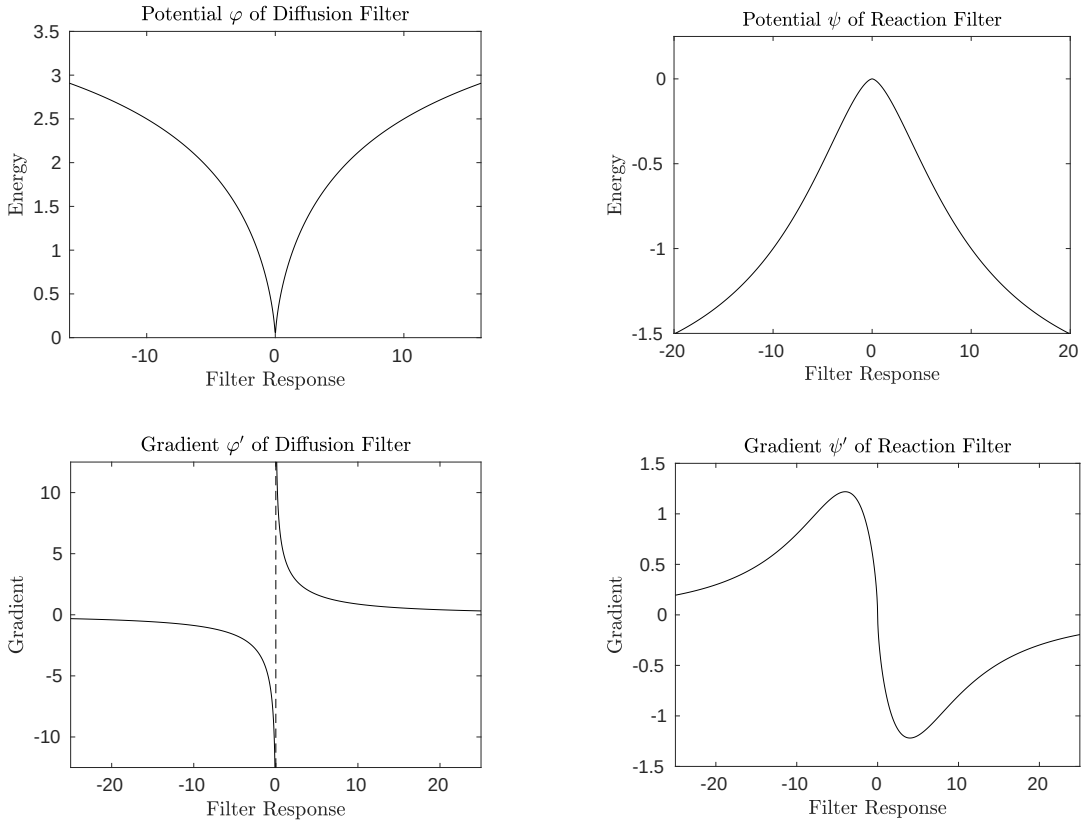


图 9.10: *左:* 扩散滤波器的电位和梯度作为滤波器响应的函数。潜在的参数是 $(a=5,,b=10,\gamma=0.7,\xi_0=0)$. 当滤波器响应接近 $xi_0=0$ 时，扩散滤波器的能量很低。因为高斯和梯度滤波器的拉普拉斯算子通控制图像的平滑度，所以它们常用作扩散滤波器。由于 $\gamma<1$，$\xi=\xi_0$ 形成一个不可微分的尖点。*右*: 反应过滤器的能量和梯度作为过滤器响应的函数。潜在的参数是 $(a=-2,,b=10,\gamma=1.6,\xi_0=0)$。当过滤器响应接近极端时，反应过滤器具有低能量。因为 Gabor 滤波器编码突出的图案特征（例如条和条纹）形成，所以它们通常充当反应滤波器。因为 $\gamma>1$ 所以 $\xi=\xi_0$ 没尖端。

这两个势系引起的动力学在模式形成过程中起着相反的作用。扩散潜力 $\phi(\xi)$ 指定最低能量（或最高概率）来过滤接近平移常数 ξ 的响应。这种类型的潜力是早期图像模型中研究的主要目标，并且它引起了各向异性扩散，其中像素强度在邻域中以与经典热方程相当的过程进行扩散。但是，单独的 $\phi(\xi)$ 无法合成逼真的图像模式。由 $\phi(\xi)$ 单独控制的 *MCMC* 过程最终会退化为恒定图像，就像封闭系统中的热量浓度最终会扩散并达到热平衡一样。

潜在的反应$\psi(\xi)$，扩散潜力的反转，与所有早期的图像模型有很大的不同。反应电位尾端的低能量促使 $\mathcal{F}$ 对滤波器产生高幅度响应。因为与高幅度滤波器响应相关联的高概率导致图案特征（例如边

缘和纹理）的主动形成，所以来自由 $\psi(\xi)$ 控制的过程的 *MCMC* 样本不会退化为恒定图像。

两组过滤器 $\mathscr{F}_d = \{F_d^{(1)}, \ldots, F_d^{(K_d)}\}$ 和 $\mathscr{F}_r = \{F_r^{(1)}, \ldots, F_r^{(K_r)}\}$ 自然来自两个潜在家族 $\phi(\xi)$ *and* $\psi(\xi)$，其中 $\mathscr{F} = \mathscr{F}_d \cup \mathscr{F}_r$。类似地，*FRAME* 能量（*9.60*）可以通过分离扩散和反应电位来重写：

$$U(I;\mathscr{F},\Lambda) = \sum_{i=1}^{K_d} \sum_{(x,y)\in\mathscr{L}} \phi^{(i)}(F_d^{(i)} * I_{(x,y)}) + \sum_{i=1}^{K_r} \sum_{(x,y)\in\mathscr{L}} \psi^{(i)}(F_r^{(i)} * I_{(x,y)}). \tag{9.64}$$

$\mathscr{F}_d$ 中的滤波器倾向于是高斯滤波器的梯度或拉普拉斯滤波器，因为这些类型的滤波器捕获的特征与图像的平滑度有关。平滑滤波器的电位通常会促进附近像素组之间的均匀性。另一方面，$\mathscr{F}_r$ 中的过滤器通常是 *Gabor* 过滤器，其表征显着特征，例如不同方向的边缘。

通过梯度下降最小化（*9.64*）得到图像 $I(x,y,t)$ 的偏微分方程：

$$\frac{\partial I}{\partial t} = \sum_{i=1}^{K_d} F_d^{(i)'} * \phi'(F_d^{(i)} * I) + \sum_{i=1}^{K_r} F_r^{(i)'} * \psi'(F_r^{(i)} * I), \tag{9.65}$$

其中 $F^{(i)'}(x,y) = -F^{(i)}(-x,-y)$。第一项减少扩散滤波器的响应梯度 $\mathscr{F}_d$，鼓励平滑度和均匀性，而第二项增加反应滤波器的梯度 $\mathscr{F}_r$，鼓励形成模式特征。(*9.65*) 被称为吉布斯反应和扩散方程（*GRADE*）*[10, 11]*。要从 $U(I;\mathscr{F},\Lambda)$ 进行采样而不是简单地最小化 $U(I;\mathscr{F},\Lambda)$，可以添加各向同性噪声以获得 *Langevin* 方程

$$I_{s+1} = I_s + \frac{\varepsilon^2}{2}\frac{\partial I}{\partial t}(I_s) + \varepsilon Z_s \tag{9.66}$$

对于 *MCMC* 迭代 s，其中 $Z_s \sim N(0, \sigma^2 I_{n\times n})$。起始图像 I_0 是任意的。有关应用于 *FRAME* 势的 *Langevin* 动力学合成的图像示例，请参见图 *9.11*。

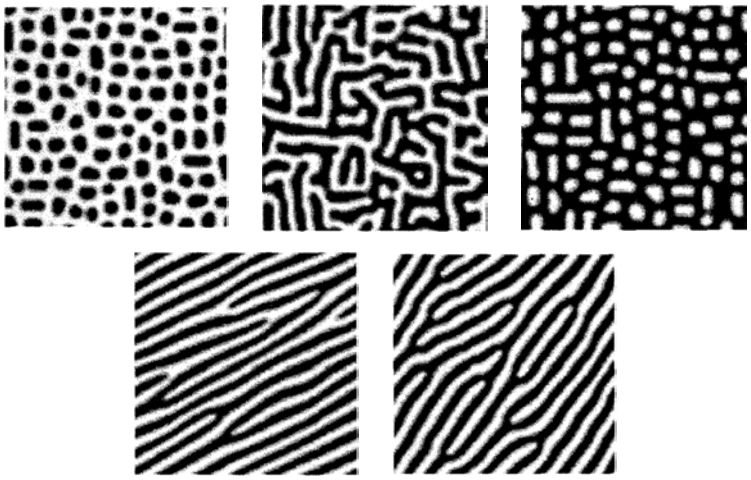


图 9.11: 使用 Langevin Dynamics 在 GRADE 电位参数化的 FRAME 密度上合成的图像的示例。*Top Row*：使用一个拉普拉斯高斯扩散滤波器和一个拉普拉斯高斯反应滤波器合成的图像。中间图像参数 $\xi_0 = 0$ 的反应过滤器，左图像的 $\xi_0 < 0$ 和右图像的 $\xi_0 > 0$。在这种情况下，ξ_0 控制 blob 颜色。*Bottom Row*：使用一个拉普拉斯高斯扩散滤波器和一个或多个 Gabor 反应滤波器合成的图像。左图像使用角度为 30° 的单个余弦 Gabor 滤镜，而右图像使用角度为 30 ° 和 60 ° 的两个余弦 Gabor 滤镜。

例 9.2 ***DeepFRAME* 模型.** *[8]* 在 *FRAME* 模型中，过滤器是从预定义的过滤器库中选择的，它限制了可以表示的模式种类。无法保证滤波器组能够有效地表示训练图像，并且当滤波器无法捕获重要图像特征时，合成结果很差。神经网络的最新趋势表明，在训练期间学习过滤器本身可以产生灵活和逼真的图像模型。包括多层滤波器卷积也可以导致复杂数据的明显更好的表示。*DeepFRAME* 模型 *[4, 8]* 扩展了 *FRAME* 模型以包含这些新功能。*DeepFRAME* 密度具有形式

$$p(I;W) = \frac{1}{Z}\exp\{F(I;W)\}q(I) \tag{9.67}$$

其中 q 是高斯白噪声的先前分布 $N(0,\sigma^2 I_N)$，而得分函数 $F(\cdot;W)$ 由权重为 W 的 *ConvNet* 定义，必须学习。相关的能量函数具有形式

$$U(I;W) = -F(I;W) + \frac{1}{2\sigma^2}||I||_2^2. \tag{9.68}$$

我们可以将 $p(I;W)$ 解释为 q 的指数倾斜，其具有均值变换的效果。由网络层之间的激活函数引起的非线性对于成功表示真实图像是必不可少的。

当激活函数是经过整流的线性单位（*ReLU*）时，$F(I;W)$ 在 I 中是分段线性的，线性区域之间的边界由网络中的激活来控制 *[6]*。设 $\Omega_{\delta,W} = \{I : \sigma_k(I;W) = \delta_k, 1 \le k \le K\}$ 其中 W 是网络权重，K 是整个网络中的激活函数数量，$\sigma_k(I;W) \in \{0,1\}$ 表明是否激活函数 k 是否对图片 I 开启，并且 $\delta = (\delta_1,\ldots,\delta_K) \in \{0,1\}^K$。由于 $F(I;W)$ 对所有 δ 的 $\Omega_{\delta,W}$ 是线性的，因此能量可写为

$$U(I;W) = -(\langle I, B_{\delta,W}\rangle + a_{\delta,W}) + \frac{1}{2\sigma^2}||I||_2^2 \tag{9.69}$$

对于一些常量 $a_{\delta,W}$ 和 $B_{\delta,W}$，这表明在 $\Omega_{\delta,W}$ 上的 $I \sim N(\sigma^2 B_{\delta,W}, \sigma^2 I_N)$ 和 $p(I;W)$ 在图像空间上是分段高斯。此分析还表征了 $U(I;W)$ 的局部最小值，它们只是 $\Omega_{\delta,W}$ 上的 $I \sim N(\sigma^2 B_{\delta,W}, \sigma^2 I_N)$。但是，无法保证高斯片段 $\Omega_{\delta,W}$ 包含其模式 $\sigma^2 B_{\delta,W}$，而且高斯片段的数量非常大，所以直接列举当地最低标准是不可行的。

与几乎所有深度学习应用程序一样，可以通过向后传播有效地计算权重 W 和图像 I 的 $F(I;W)$ 的梯度。给定一组学习权重 W，很容易将 *Langevin Dynamics* 应用于（*9.58*）来从图像密度 $P(I;W)$ 中进行采样。有关使用 *Langevin* 动态在训练的 *DeepFRAME* 密度上合成的图像的示例，请参见图 *9.12*。

图 9.12: *左:* 来自 DeepFRAME 模型的 Langevin 样本在未对齐的纹理图像上进行训练。左图像是训练图像，右图像是合成样本。合成图像再现主要纹理特征（花头和草地区域）。*右*：来自 DeepFRAME 模型的 Langevin 样本在对齐图像上训练。训练图像位于顶行，并且从 DeepFRAME 密度合成的示例图像位于底行。

练习

题 1. 从非各向同性高斯分布的采样。考虑目标分布 $(X,Y) \sim N(0,\Phi)$，其中 $\Phi = \begin{pmatrix} 1 & 0.9998 \\ 0.9998 & 1 \end{pmatrix}$

a) 目标分布的能量函数是什么？制作能量函数的等高线图。

b) 假设您使用单位矩阵 I_2 作为动量协方差从目标分布中进行采样。在单个 HMC 迭代中从目标分布获取独立样本所需的最大步长 ε^* 和最小跳跃步数 L^* 是多少？

c) 动量协方差矩阵 Σ_{ideal} 的理想选择是什么？当使用理想动量协方差时，获得独立样本所需的最大步长 ε^*_{ideal} 和最小跳跃步数 L^*_{ideal} 是多少？

d) 对于下面列出的每种方法，从状态 $(X,Y) = (0,-10)$ 开始一个链，运行 $1{,}000 \cdot K$ 老化迭代，以及来自目标分布的 $10{,}000 \cdot K$ 抽样迭代（为了方法之间的公平比较，需要并给出了 K）。对于每种方法，可视化老化路径，在采样阶段的迭代中绘制 X 和 Y 坐标的值，并计算最终 X 和 Y 样本的 ESS。对于图和 ESS 计算，每 K 迭代使用一个采样点。评论结果之间的差异。ε^* 和 L^* 参考 b）的答案。

1. 从 $N(0,\Phi), K = 1$ 直接采样.
2. 高斯提议 $N(0,(\varepsilon^*)^2), K = L^*$ 的 Metropolis-Hastings。
3. 随机行走: $p \sim N(0,I_2), \varepsilon = \varepsilon^*, L = L^* K = 1$ 的 HMC。
4. $p \sim N(0,I_2), \varepsilon = \varepsilon^*, L = L^*/2, K = 1$ 的 HMC。
5. $p \sim N(0,I_2), \varepsilon = \varepsilon^*, K = L*$ 的 LMC。
6. $p \sim N(0,\Sigma_{ideal}), \varepsilon = \varepsilon^*_{ideal}, L = L^*_{ideal}, K = 1$ 的 HMC。(使用上面 c) 的答案)
7. $p \sim N(0,\Sigma_{ideal}), \varepsilon = \varepsilon^*_{ideal}, K = L^*_{ideal}$ 的 LMC。

题 2. 从“香蕉”分布中抽样。考虑 $\theta = (\theta_1,\theta_2)$ 的后验分布与之前的 θ $N(0,I_2)$ 和 $Y|\theta \sim N(\theta_1 + (\theta_2)^2, 2)$。

a) 后密度 $P(\theta|Y)$ 的能量函数是多少？制作能量函数的等高线图。

b) 在新设置中，可以通过调整接受率来找到步长 ε^*。对于 θ 的网格，从原点开始运行具有动量协方差 I_2 的 2000LMC 迭代。选择拒绝率介于 10%-35% 之间的步长，并报告该值。

c) 对于下面列出的每种方法，从状态 $(\theta_1,\theta_2) = (0,0)$ 开始一个链，运行 $1{,}000 \cdot K$ 老化迭代，以及来自目标分布的 $10{,}000 \cdot K$ 抽样迭代（方法之间的公平比较，需要并已给出 K）。对于每种方法，在采样阶段的迭代中绘制 θ_1 和 θ_2 坐标的值，并计算最终 θ_1 和 θ_2 样本的 ESS。对于图和 ESS 计算，每 K 迭代使用一个采样点。对于 HMC 方法 4 和 5，可视化已接受路径和被拒绝路径的 Leapfrog 步骤。评论结果之间的差异。ε^* 指的是上面 b）的值。

1. 高斯提议 $N(0,(\varepsilon^*)^2), K = 25$ 的 Metropolis-Hastings。
2. 随机行走: LMC with $p \sim N(0,I_2), \varepsilon = \varepsilon^*, K = 1$。
3. $p \sim N(0,I_2), \varepsilon = \varepsilon^*, K = 25$ 的 LMC。
4. $p \sim N(0,I_2), \varepsilon = \varepsilon^*, L = 5, K = 1$ 的 HMC。
5. $p \sim N(0,I_2), \varepsilon = \varepsilon^*, L = 25, K = 1$ 的 HMC。

参考文献

[1] S. Amari and H. Nagaoka. *Methods of Information Geometry*. Oxford Univ Press, 2000.

[2] K. Brodlie, A. Gourlay, and J. Greenstadt. Rank-one and rank-two corrections to positive definite matrices expressed in product form. *IMA Journal of Applied Mathematics*, 11(1):73–82, 1973.

[3] Mark Girolami and Ben Calderhead. Riemann manifold langevin and hamiltonian monte carlo. *Journal ofthe Royal Statistical Society: B*, 73(2):123–214, 2011.

[4] Yang Lu, Song Chun Zhu, and Ying Nian Wu. Learning frame models using cnn filters. *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[5] P.B. Mackenzie. An improved hybrid monte carlo method. *Physics Letters B*, 226:369–371, 1989.

[6] Guido F Montufar, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio. On the number of linear regions of deep neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2924–2932, 2014.

[7] Radford M Neal. Mcmc using hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo Chapter 5*, 2011.

[8] Jianwen Xie, Wenze Hu, Song Chun Zhu, and Ying Nian Wu. A theory of generative convnet. *International Conference on Machine Learning*, 2016.

[9] Y. Zhang and C. Sutton. Quasi-newton methods for markov chain monte carlo. *Advances in Neural Information Processing Systems*, pages 2393–2401, 2011.

[10] Song Chun Zhu and David Mumford. Prior learning and gibbs reaction-diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(11):1236–1250, 1997.

[11] Song Chun Zhu and David Mumford. Grade: Gibbs reaction and diffusion equations. *ICCV*, pages 847–854, 1998.

[12] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126, 1998.

第 9 章　随机梯度学习

‘‘在线学习数学研究的起点必须是我们对学习系统主观理解的数学陈述。.” - Léon Bottou

引言

统计学习通常涉及最小化目标函数以找到模型参数的合适值。用于最小化可微分目标函数的简单且普遍存在的方法是梯度下降（Gradient Descent），其使用最陡下降方向上的迭代参数更新，来最小化目标。然而，存在目标函数梯度的计算在分析上难以处理或在计算上不可行的情况。两个重要的例子是 Gibbs 模型的参数估计和深度神经网络中的权重优化。随机梯度方法使用随机但无偏的全梯度估计，能够成为克服完全梯度不可用情况的有用工具。在本章的前半部分，提出了几个关于每个观测梯度逼近真实梯度的定理，并讨论了随机梯度和 Langevin 动力学之间的重要联系。第二部分介绍马尔可夫随机场模型的参数估计，最后一部分介绍了深度图像模型的 MCMC 学习方法.

10.1　随机梯度: 动机和属性

最小化目标函数 f 是统计学习的最常见框架。在判别模型中，目标是损失函数 $f(w)$，其测量具有参数 w 模型的预测误差，例如真实类别和预测类别之间的交叉熵。在生成模型中，目标是真实数据分布 q 与来自族 $\mathscr{P} = \{p_\theta : \theta \in \Theta\}$ 分布之间的差异的度量 $f(\theta) = D(q, p_\theta)$。KL 散度是一种流行的概率分布间的分离度量。

当统计模型的参数 x 是连续的并且目标函数 $f(x)$ 是可微分的时候，用于最小化 f 的简单且有效的方法是梯度下降 [2]。从初始点 x_0 开始，根据以下规则更新参数

$$x_{t+1} = x_t - \gamma_t \nabla f(x_t), \tag{10.1}$$

其中 $\gamma_t > 0$ 是迭代 t 的步长，直到找到局部最小值。当真实梯度 $\nabla f(x)$ 由于分析或计算原因而不可用时，可以使用满足 $\mathrm{E}\left[\widetilde{\nabla} f(x)\right] = \nabla f(x)$ 的真实梯度的随机近似 $\widetilde{\nabla} f(x)$。更新规则

$$x_{t+1} = x_t - \gamma_t \widetilde{\nabla} f(x_t), \tag{10.2}$$

被称为随机梯度下降 *(SGD)*。下面讨论两个典型案例.

10.1.1 激励案例

例 10.1 ***Gibbs*** **模型的极大似然参数估计.** 当 f 的梯度具有分析上难以处理的期望的形式时，随机梯度是有用

$$\nabla f(\theta) = E_{p_\theta}[g(X;\theta)] \tag{10.3}$$

对于分布族 $\mathscr{P} = \{p_\theta\}_{\theta\in\Theta}$ 和随机变量 $X \in \mathscr{X}$。当极大似然被用来估计 *Gibbs* 模型的参数时会遇到这种情况

$$p_\theta(X) = \frac{1}{Z(\theta)} \exp\{-U(X;\theta)\} \tag{10.4}$$

具有偏函数 $Z(\theta) = \int_{\mathscr{X}} \exp\{-U(X;\theta)\}dX$. 给出独立同分布的 p_θ 的对数似然。观测数据 $\{X_i\}_{i=1}^n$ 时

$$l(\theta) = \frac{1}{n}\sum_{i=1}^{n} \log p_\theta(X_i) = -\log Z(\theta) - \frac{1}{n}\sum_{i=1}^{n} U(X_i;\theta), \tag{10.5}$$

并且最大化 $l(\theta)$ 产生模型参数的最大似然估计（*MLE*）θ^*.

最大化对数似然 *(10.5)* 找到 *MLE* 等效于找到 的值，其最小化 p_θ 与真实数据分布 q 之间的 *KL* 散度。观察

$$\begin{aligned} KL(q||p_\theta) &= E_q\left[\log \frac{q(X)}{p_\theta(X)}\right] \\ &= E_q[\log q(X)] - E_q[\log p_\theta(X)], \end{aligned}$$

且 $E_q[\log q(X)]$ 不依赖于 θ。最小化 $KL(q||p_\theta)$ 只需最小化 $-E_q[\log p_\theta(X)]$。给定 q 之后的数据集 $\{X_i\}_{i=1}^n$，我们可以使用大数定律来获得近似 $E_q[\log p_\theta(X)] \approx \frac{1}{n}\sum_{i=1}^n \log p_\theta(X_i)$。因此，最小化 $KL(q||p_\theta)$ 相当于最大化 *(10.5)*，并且 p_{θ^*} 可以被解释为族 $\mathscr{P}$ 中与 q 最接近的近似.

难以理解的偏函数 $Z(\theta)$ 是评估 $\nabla l(\theta)$ 时的主要障碍。幸运的是，$\log Z(\theta)$ 的梯度可以用封闭的形式表示:

$$\frac{d}{d\theta}\log Z(\theta) = -E_{p_\theta}\left[\frac{\partial}{\partial\theta}U(X;\theta)\right]. \tag{10.6}$$

推导如下:

$$\begin{aligned} \frac{d}{d\theta}\log Z(\theta) &= \frac{1}{Z(\theta)}\left[\frac{d}{d\theta}Z(\theta)\right] \\ &= \frac{1}{Z(\theta)}\left[\frac{d}{d\theta}\int_{\mathscr{X}} \exp\{-U(X;\theta)\}dX\right] \\ &= \frac{1}{Z(\theta)}\int_{\mathscr{X}} \frac{\partial}{\partial\theta}\exp\{-U(X;\theta)\}dX \\ &= -\frac{1}{Z(\theta)}\int_{\mathscr{X}} \exp\{-U(X;\theta)\}\left[\frac{\partial}{\partial\theta}U(X;\theta)\right]dX \end{aligned}$$

$$= -\int_{\mathscr{X}} p_\theta(X)\left[\frac{\partial}{\partial\theta}U(X;\theta)\right]dX = -E_{p_\theta}\left[\frac{\partial}{\partial\theta}U(X;\theta)\right].$$

在温和的规律性条件下，第三行中的积分和微分运算的交换是合理的。该分析表明似然梯度可以写成

$$\nabla l(\theta) = E_{p_\theta}\left[\frac{\partial}{\partial\theta}U(X;\theta)\right] - \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\theta}U(X_i;\theta), \tag{10.7}$$

这是形式 *(10.3)* 的特殊情况, 其中

$$g(X;\theta) = \frac{\partial}{\partial\theta}U(X;\theta) - \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\theta}U(X_i;\theta).$$

除了最简单外的所有情况中，期望 $E_{p_\theta}\left[\frac{\partial}{\partial\theta}U(X;\theta)\right]$ 不能精确计算。然而，通过从分布 p_θ 获得 *MCMC* 样本 $\{Y_i\}_{i=1}^m$，我们可以使用大数定律近似 $E_{p_\theta}\left[\frac{\partial}{\partial\theta}U(X;\theta)\right] \approx \frac{1}{m}\sum_{i=1}^m \frac{\partial}{\partial\theta}U(Y_i;\theta)$ 并计算真实梯度 $\nabla l(\theta)$ 的 $\widetilde{\nabla} l(\theta)$*:*

$$\widetilde{\nabla} l(\theta) = \frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial\theta}U(Y_i;\theta) - \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\theta}U(X_i;\theta). \tag{10.8}$$

来自 p_θ 的 *MCMC* 样本 $\{Y_i\}_{i=1}^m$ 有时被称为负样本，而不是跟随真实分布 q 的"正"样本 $\{X_i\}_{i=1}^n$。在计算近似梯度后，可以应用随机梯度下降更新 *(10.2)* 来迭代求解 $\hat{\theta}$，其中 $\widetilde{\nabla} f(\theta) = -\widetilde{\nabla} l(\theta)$，因为我们最大化$l(\theta)$。直观地，*(10.8)* 中的梯度激励 p_θ 的 *MCMC* 样本, 去匹配由 $U(X;\theta)$ 编码的特征中的训练数据 $\{X_i\}_{i=1}^n$.

例 10.2 大规模学习. 当目标是大量可微分子函数的总和时，随机梯度下降也很有用*:*

$$L(w) = \frac{1}{n}\sum_{i=1}^{n}L_i(w). \tag{10.9}$$

$L(w)$ 的倒数具有简单的形式

$$\nabla L(w) = \frac{1}{n}\sum_{i=1}^{n}\nabla L_i(w). \tag{10.10}$$

在实际中经常遇到目标函数形式 *(10.9)*。将每个观察损失 $L(X;w)$ 应用于一组观察数据 $\{X_i\}_{i=1}^n$ 时，可能出现附加目标，其中 $L_i(w) = L(X_i;w)$。在监督学习中，观察结果为 $X_i = (Z_i, Y_i)$ 对，个体损失项为 $L(X_i;w) = L(Z_i,Y_i;w)$。通常，不同的观察 X_i 被视为 i 独立同分布样本。当 $L(X;w)$ 是无监督学习中的负对数似然 $-\log p(X;w)$ 或监督学习中的条件负对数似然 $-\log p(Z|Y;w)$ 时，观察之间的独立性导致整个数据集的损失分为各个损失项的总和*:*

$$-\log p(\{X_i\}_{i=1}^n;w) = -\log\left[\prod_{i=1}^{n}p(X_i;w)\right] = \sum_{i=1}^{n}-\log p(X_i;w) = \sum_{i=1}^{n}L(X_i;w) \propto L(w).$$

在对复杂数据进行建模时，单个梯度 $\frac{\partial}{\partial w}L(X_i;w)$ 的计算成本可能很高。如果数据集非常大，则完全梯度 $\nabla L(w)$ 的计算可能很快地变得非常昂贵。在深度学习应用中昂贵的梯度计算和大型数据集都是典

型的。另一方面，每个观测梯度 $\frac{\partial}{\partial w}L(X_i;w)$ 可以被认为是真实梯度 $\nabla L(w)$ 的噪声版本，并且在大样本量的限制下，少量梯度的预测能力保持不变。这促使了对于 $|B| = n_B$ 大小的随机样本 $B \subset \{1,\ldots,n\}$ 使用小批量 $\{X_i : i \in B\}$。在每次迭代中，仅针对小批量观察计算梯度，产生随机梯度

$$\nabla_B L(w) = \frac{1}{n_B}\sum_{i\in B}\frac{\partial}{\partial w}\nabla L(X_i;w). \tag{10.11}$$

显示 $E[\nabla_B L(w)] = \nabla L(w)$ 是直接的，*10.1.3*节中讨论了 *(10.11)* 中更深层的属性。*(10.11)* 的一个特例是在线学习，每批 *(*即 $n_B = 1$*)* 使用一次观测.

在使用 MCMC 的深度学习应用程序中 (参见第 10.3节)，需要来自例 10.1例 10.2的随机梯度。当用梯度 (10.8) 学习深层网络函数的 MLE 参数时，需要 MCMC 样本以在第一项中获得近似梯度，而在第二项中使用小批量的观测数据以降低计.

10.1.2 Robbins-Monro 定理

SGD 算法 (10.2) 的收敛性分析仅限于 1 维情况 $d = 1$.

Theorem 10.1 (Robbins-Monro) 如果满足以下条件，则序列 w_n 会收敛于 L^2*(*因此概率*)* 为 θ*:*

1. $G(w)$ 在存在 $C < \infty$ 使得 $P(|G(w)| \le C) = 1$ 的意义上时均匀有界的.
2. $F(w)$ 不减少. 可微且 $F'(\theta) > 0$.
3. 序列 γ_n 满足 $\sum_{n=0}^{\infty}\gamma_n = \infty$ 以及 $\sum_{\gamma=0}^{\infty}\gamma_n^2 < \infty$.

定理 10.1的多维版本受到许多约束性假设的限制。最近的显式版本 [16] 考虑了噪声观, 因此 $G(\theta_n) = \nabla f_n(\theta_n)$.

Theorem 10.2 (Moulines, 2011) 假设满足一下条件

H1) 存在可微函数 $f : \mathbb{R}^d \to \mathbb{R}$ 使得

$$E[\nabla f_n(\theta)] = \nabla f(\theta), \quad \forall n \ge 1 \quad \text{概率为 } 1$$

H2) 对所有的 $n \ge 1$ f_n 几乎肯定时凸的, 可微且

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, E(\|\nabla f_n(\theta_1) - \nabla f_n(\theta_2)\|^2) \le L^2\|\theta_1 - \theta_2\|^2 \quad \text{概率为 } 1$$

H3) 函数 f 相对于范数 $\|\cdot\|$ 是强凸的，其中一些常数 $\mu > 0$*:*

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, f(\theta_1) \ge f(\theta_2) + (\theta_1 - \theta_2)\nabla f(\theta_2) + \frac{\mu}{2}\|\theta_1 - \theta_2\|^2$$

H4) 存在 $\sigma > 0$ 使得 $\forall n \ge 1, E(\|\nabla f_n(\theta)\|^2) \le \sigma$.

设 $\delta_0 = \|\theta_0 - \theta\|^2$, $\varphi_\beta(t) = \frac{t^\beta - 1}{\beta}$ 且对于一些 $\alpha \in [0,1], \gamma_n = Cn^{-\alpha}$. 来自于等式. (10.2)的序列满足

$$E\|\theta_n - \theta\|^2 \leq 2\exp\left[4L^2C^2\varphi_{1-2\alpha}(n) - \frac{\mu C}{2}n^{1-\alpha}\right]\left(\delta_0 + \frac{\sigma^2}{L^2}\right) + \frac{4C\sigma^2}{\mu n^\alpha}$$

当 $\alpha < 1$ 时. 如果 $\alpha = 1$, 那么:

$$E\|\theta_n - \theta\|^2 \leq \frac{\exp(2L^2C^2)}{n^{\mu C}}\left(\delta_0 + \frac{\sigma^2}{L^2}\right) + 2\frac{C^2\sigma^2}{n^{\mu C/2}}\varphi_{\mu C/2-1}(n).$$

在最小处 [13] 收敛时间的 Hessian 矩阵 $H_{ij} = \frac{\partial^2 L(\theta)}{\partial w_i \partial w_j}$ 的条件数 $\kappa = \lambda_{max}/\lambda_{min}$ 成正比。此行为与第 9.5节中讨论的 HMC 属性相当。完全梯度迭代 (批量大小等于 n) 在 $L(\theta^k) - L(\theta) = O(\rho^k)$ 的意义上具有线性收敛，其中 $\rho < 1$ 取决于条件数 κ(来自 [17], 定理 2.1.15)。在线版本具有子线性收敛 $E[L(\theta^k)] - L(\theta) = O(1/k)$[18] 但每次迭代的速度要快 n 倍。

10.1.3 随机梯度下降和 Langevin 方程

考虑 (10.11) 的随机小批量梯度。小批量选择的采样过程定义了梯度的多变量分布。可以使用梯度的第一和第二时刻来分析 SGD，以 Langevin 方程的形式定义 SGD 的连续时间模拟（参见第 9.4.3节）。令人惊讶的是，出现在 SGD Langevin 方程中的扩散矩阵是传统牛顿优化中使用的矩阵的逆。SGD 与 Langevin 方程之间的联系揭示了 SGD 的重要性质，包括批量大小的作用，稳态分布和一般化.

考虑使用小批量梯度 $\nabla_B f(x) = \frac{1}{n_B}\sum_{i\in B}\nabla f_i(x)$ 最小化额外损失 $f(x) = \sum_{i=1}^n f_i(x)$. 继继 Hu 等人 [11] 之后, 我们首先找到随机梯度 $\nabla_B f(x)$ 的期望和方差。矩阵

$$D(x) = \left(\frac{1}{N}\sum_{i=1}^N \nabla f_i(x)\nabla f_i(x)^\intercal\right) - \nabla f(x)\nabla f(x)^\intercal \tag{10.12}$$

将在分析中发挥重要作用，我们将其称为扩散矩阵，其原因将变得很清楚。设 $B = \{i_1, \ldots, i_{n_B}\}$ 是小批量成员的索引。假设为了便于分析,B 是一个简单的随机样本，替换 $\{1, \ldots, n\}$。单样本期望和方差是

$$\mathrm{E}[\nabla f_{i_j}(x)] = \frac{1}{n}\sum_{i=1}^n \nabla f_i(x) = \nabla f(x),$$

$$\mathrm{Var}[\nabla f_{i_j}(x)] = \mathrm{E}\left[\nabla f_{i_j}(x)\nabla f_{i_j}(x)^\intercal\right] - \mathrm{E}\left[\nabla f_{i_j}(x)\right]\mathrm{E}\left[\nabla f_{i_j}(x)\right]^\intercal = D(x).$$

关于替换的简单随机样本的均值和方差的标准结果显示:

$$\mathrm{E}[\nabla_B f(x)] = \frac{1}{n_B}\sum_{j=1}^{n_B}\mathrm{E}[\nabla f_{i_j}(x)] = \frac{1}{n_B}\sum_{j=1}^{n_B}\nabla f(x) = \nabla f(x) \tag{10.13}$$

$$\mathrm{Var}[\nabla_B f(x)] = \mathrm{Var}\left[\frac{1}{n_B}\sum_{j=1}^{n_B}\nabla f_{i_j}(x)\right] = \frac{1}{n_B^2}\sum_{j=1}^{n_B}\mathrm{Var}\left[\nabla f_{i_j}(x)\right] = \frac{D(x)}{n_B} \tag{10.14}$$

其中方差计算中的第二个相等利用小批量梯度是独立随机变量的事实.

我们可以将均值与小批量梯度分开，并将 SGD 更新 (10.2) 写为

$$X_{t+1} = X_t - \eta \nabla_B f(X_t) = X_t - \eta \nabla f(X_t) + \sqrt{\eta} V_t$$

其中 $V_t = \sqrt{\eta}(\nabla f(X_t) - \nabla_B f(X_t))$. SGD 扩散项 V_t 明显具有均值 $\mathrm{E}[V_t] = 0$ 且方差 $\mathrm{Var}[V_t] = \frac{\eta}{|B|} D(X_t)$. Li 等人.[14] 分析 V_t 以证明 SGD 和 Langevin 动力学之间的以下联系. 见 Hu 等人. [11] 进行其它分析.

Theorem 10.3 (Li et al., 2017 [14]) *假设 f 和 $\{f_i\}_{i=1}^n$ 是 Lipschitz 连续的，具有至多线性渐进增长，并且具有足够高的导数属于具有多项式增长的函数集。那么*

$$dX_t = -\nabla f(X_t) dt + \sqrt{\frac{\eta}{n_B} D(X_t)}\, dW_t, \tag{10.15}$$

其中 dW_t 布朗运动, 是 (10.2) 中具有小批量梯度 (10.11) 的 SGD 过程的一阶随机近似。

SGD 和 Langevin 动力学 (10.15) 之间的等价揭示了 SGD 的几个重要特性。一个观测结果是 SGD 批量大小起反温作用，因为噪声 $\frac{\eta}{n_B} D(X_t)$ 的大小与批量大小 n_B 成反比变化。使用较小的批量大小可以直观地理解为在较高温度下探索损失情况，因为当使用较小批次时，采样过程中存在更多变化。大批量 SGD 与完全梯度优化类似，就像低温采样一样。有趣的是，最近的工作 [12] 表明，使用小批量 SGD 发现的深度网络参数比使用大批量 SGD 的参数具有更好的泛化属性。这证明小批量 SGD 产生的" 高温" 动态比完全梯度方法更不容易陷入局部最小值。虽然 SGD 最初是出于计算效率的原因而采用的，但它也具有自然适合在高度过度参数化设置中找到具有良好泛化参数的属性。

Zhang 等人. [23] 观察到扩散矩阵 $D(x)$ 的结构也与 SGD 的泛化特性有关。考虑在局部最小值附近的 SGD 过程。在这种情况下，$\nabla f(x) \approx 0$ 且扩散矩阵 $D(x)$ 近似等于经验费希尔信息:

$$D(x) \approx \frac{1}{N} \sum_{i=1} \nabla f_i(x) \nabla f_i(x)^\intercal \approx \mathrm{E}[\nabla f_i(x) \nabla f_i(x)]^\intercal. \tag{10.16}$$

有较大特征值的费希尔信息的特征向量具有较大的曲率，这意味着沿着这些方向的小扰动可以导致 $f(x)$ 的输出大的变化。相反，有较小特征值的费希尔信息的特征向量对应于抵抗输入的小扰动的" 平坦" 方向。直观地," 平坦" 最小值具有更好的泛化属性，因为对输入扰动的鲁棒性表明该模型不会过度拟合.

比较 (10.15) 中的 SGD Langevin 动力学与 Riemann 流形 Langevin 动力学 (见第 9.5节) 揭示了令人惊讶的事实，(10.16) 中的噪声矩阵 $D(x)$ 是出现在 RMLMC 中的噪声矩阵的逆。回想一下，RMLMC 噪声协方差是逆费希尔信息，它重新调整局部景观几何，使每个方向的单位方差不相关。换句话说，RMLMC 动态调整加入 LMC 中梯度的各向同性噪声，以便沿着局部景观的更受约束的方向（具有更大特征值的方向）采取更小的步骤，而沿着局部景观的更平坦的方向采取更大的步骤（具有较小特征值的方向）。RMLMC 动态与传统二阶优化技术的理念一致.

另一方面，SGD 中的噪声矩阵 $D(x)$ 是未反转的费希尔信息的近似值。与传统的二阶优化不同，SGD 在具有高特征值的方向上采取较大的步长，而在具有低特征值的方向上采取较小的步长。SGD 的扩散动态主动探测局部协方差的最受约束的方向。受约束的方向可以直观地理解为" 不信任的" 方向，因为沿着这些方向, 模型中的微小变化会导致模型性能大的变化。有趣的是，SGD 扩散项似乎集中在不值得

信任的方向上。通过在每次迭代中寻找最受约束的方向，SGD 最终定位" 平坦的" 局部最小值，其中几乎所有方向都是不受约束的。因此，SGD 可以在训练早期逃离狭窄的盆地并避免过度拟合。SGD 动态与经典优化技术有很大不同。局部几何中的紧密约束方向不需要被遵守--相反，刚性方向的主动扰动有助于找到具有更好泛化的平坦最小值。深度学习中高度复杂函数的优化不仅有利于 SGD 的计算效率，还有利于自然适应损失景观几何的 SGD 属性。

鉴于 SGD 可以通过连续时间 Langevin 方程近似，很自然地就能分析 Langevin 动力学的稳态分布。天真的直觉可能表明稳态是 $\frac{1}{Z}\exp\{-f(x)/T\}$，是目标函数 f 的 Gibbs 分布。然而，SGD 近似的真实稳态更复杂。直觉上，当 $D(x)$ 是一个常数矩阵 D 时，考虑 (10.15) 的动态。在这种情况下，动态 (10.15) 可以写成

$$dX_t = -\frac{\eta}{2n_B} D\left[\frac{2n_B}{\eta} D^{-1}\nabla f(X_t)\right] dt + \sqrt{2\left(\frac{\eta}{2n_B}\right) D}\, dW_t. \tag{10.17}$$

第 9.4.3节中 Langevin 方程的讨论表明, 如果有一个函数 g 满足 $\nabla g(x) = D^{-1}\nabla f(x)$, 那么 $\frac{1}{Z}\exp\{-\frac{2n_B}{\eta} g(x)\}$ 必定是 (10.17) 的稳态. 很容易看出，当 $D = c\,\mathrm{Id}$ 时, (10.17) 的稳态具有 $\frac{1}{Z}\exp\{-f(x)/T\}$ 的形式, 标量乘以单位矩阵.

更一般地，Chaudhari 和 Soatto[3] 在简化 $\nabla f(x)$ 结构的假设下表明 SGD Langevin (10.15) 的稳态等于 $\frac{1}{Z}\exp\{-f(x)/T\}$, 当且仅当 $D(x) = c\,\mathrm{Id}$ 时。在实际中，在 SGD 轨迹中观察到的扩散矩阵是高度非各向同性的。鉴于先前的观察结果,$D(x)$ 的结构与 f 的非欧几里德几何相关，这并不令人惊讶。因此 [3] 中的作者得出结论，SGD 不遵循目标函数的 Gibbs 分布。另一方面，自然梯度 $D^{-1}\nabla f(X)$ 和 (10.17) 中的扩散矩阵 D 的出现表明 SGD 的动态自然地适应 f 的能量景观。进一步了解 SGD 的稳态和几何特性可以继续为大规模优化和深度学习的复杂行为提供有价值的见解。

10.2 马尔可夫随机场 (MRF) 模型的参数估计

本节介绍了学习马尔可夫随机场（MRF）模型参数的方法。这种密度族具有这种形式

$$p_\beta(x) = \frac{1}{Z(\beta)}\exp\left\{-\sum_{k=1}^{K}\langle\beta^{(k)}, U_k(x)\rangle\right\} = \frac{1}{Z(\beta)}\exp\{-\langle\beta, U(x)\rangle\} \tag{10.18}$$

其中 $\beta = (\beta^{(1)}, \ldots, \beta^{(K)})$，$\beta^{(k)} \in \mathbb{R}^{d_k}$，是要学习的模型参数, $x \in \Omega \subset \mathbb{R}^d$ 是 MRF 空间中的观测值，$U_k(x) : \Omega \to \mathbb{R}^{d_k}$ 是足够的统计量，$Z(\beta)$ 是偏函数 $Z(\beta) = \int_\Omega \exp(-\langle\beta, U(x)\rangle)dx$。MRF 密度是 Gibbs 密度 (10.4) 的特殊情况，其中势能具有线性分解 $U(x;\beta) = \langle\beta, U(x)\rangle$。

可以将 MRF 分布推导为满足给定期望集的最大熵分布。考虑足够的统计量 $U_k(x) : \Omega \to \mathbb{R}^{d_k}$ for $k = 1, \ldots, K$，并且假设对于每个 U_k，想要找到具有特定期望值 $\mathrm{E}_p[U_k(X)] = a_k$ 的概率分布 p。为了避免在强制预期之外指定模型，应该寻找仅满足充分统计要求的最一般分布。使用 $-\mathrm{E}_p[\log p(X)]$ 作为分布一般性的度量产生约束优化问题

$$p(x) = \mathrm{argmax}_p\left\{-\int_\Omega p(x)\log p(x)\,dx\right\} \quad \text{服从} \quad \mathrm{E}_p[U_k(X)] = a_k \ \text{ for } k = 1, \ldots, K. \tag{10.19}$$

使用拉格朗日乘子法,可以证明约束最大化问题 (10.20) 的解是 MRF 分布 (10.18),其中 $\beta = (\beta^{(1)}, \ldots, \beta^{(K)})$

是拉格朗日乘数。

虽然对于给定的一组期望的最大熵密度的形式总是 MRF 密度，但是学习 MRF 模型仍然需要找到超参数 β。拉格朗日乘数不能明确计算，必须使用梯度下降的迭代估计。如例 10.1中所述，通过最大化 i.i.d 的对数似然来找到 MLEβ^*。观测数据相当于最小化 MRF 模型 p_β 与真实数据分布之间的 KL 散度，这进一步地相当于在真实分布下最小化 p_β 的熵。因此，对于 MRF 模型的 MLE 学习可以被解释为极小化熵学习：MRF 密度给出最大熵模型的形式，而 MLE 学习在 MRF 族中找到最小熵模型。

给定一组独立同分布观察 $\{X_i\}_{i=1}^n$，可以按照例 10.1，通过根据 (10.8) 中的梯度优化 β 来找到对数似然 (10.5) 的 MLE β^*。由于 MRF 密度的势能在 β 中是线性的，因此得分函数是 $\frac{\partial}{\partial\beta}\nabla\log p_\beta(x) = -\nabla\log Z(\beta) - U(x)$ 并且对数似然 $l(\beta)$ 的梯度有更简单的形式

$$\nabla l(\beta) = \mathrm{E}_{p_\beta}[U(X)] - \frac{1}{n}\sum_{i=1}^{n} U(X_i) \approx \frac{1}{m}\sum_{i=1}^{m} U(Y_i) - \frac{1}{n}\sum_{i=1}^{n} U(X_i) \tag{10.20}$$

其中 $\{Y_i\}_{i=1}^m$ 是从 MCMC 获取的 p_β 的样本. 需要蒙特卡罗模拟来估计难以处理的期望 $\mathrm{E}_{p_\beta}[U(X)]$ 以获得 β 的近似梯度。由于潜在的 $-\langle\beta, U(x)\rangle$ 的线性，对数似然 $l(\beta)$ 是凸的，MLE β^* 对于 MRF 模型是唯一的。观察到随机梯度 (10.20) 激励来自当前分布 p_β 的样本去匹配来自真实数据分布样本的充分统计。

10.2.1 学习具有随机梯度的 FRAME 模型

作为 MRF 模型的随机梯度学习的一个具体例子，我们讨论了第 9.6.3节例 9.1中介绍的 FRAME 模型。设 $\mathbf{I}_\Lambda$ 是在点阵 Λ 上定义的图像和 $\mathbf{I}_{\partial\Lambda}$ 是放射性点阵 Λ 的邻域 $\partial\Lambda$ 的固定边界条件。设 $\mathbf{h}(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda})$ 是边界条件 $\mathbf{I}_{\partial\Lambda}$ 下 $\mathbf{I}_\Lambda$ 的特征统计量。通常 $\mathbf{h}(\cdot)$ 是一组直方图 $\{\mathbf{h}_k(\cdot)\}_{k=1}^K$，其测量 $\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda}$ 上卷积滤波器 F_k 的响应。FRAME 密度具有形式 (见 [25])

$$p(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda};\beta) = \frac{1}{Z(\mathbf{I}_{\partial\Lambda},\beta)}\exp\left\{-\sum_{k=1}^{K}\langle\beta^{(k)}, \mathbf{h}_k(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda})\rangle\right\} = \frac{1}{Z(\mathbf{I}_{\partial\Lambda},\beta)}\exp\{-\langle\beta, \mathbf{h}(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda})\rangle\}. \tag{10.21}$$

在原始 FRAME 应用中，单个纹理图像 $\mathbf{I}^{\text{obs}}$ 用作模型的训练数据。假设纹理图像的分布是空间不变的，则 $\mathbf{I}^{\text{obs}}$ 的不同块（滤波器卷积非零的局部区域）可以被视为独立同分布观测结果。因此，来自单个大纹理图像的随机贴片具有与独立贴片相同的统计量 $\mathbf{h}(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda})$。当建模对齐的数据 (如对象) 时，必须更加小心。

最大化 FRAME 对数似然

$$\mathcal{G}(\beta) = \log p(\mathbf{I}_\Lambda^{\text{obs}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}};\beta) \tag{10.22}$$

在形式中使用梯度 (10.20)

$$\widetilde{\nabla}\mathcal{G}(\beta) = \mathbf{h}(\mathbf{I}_\Lambda^{\text{syn}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}}) - \mathbf{h}(\mathbf{I}_\Lambda^{\text{obs}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}}) \tag{10.23}$$

可用于迭代求解唯一的 MLE β^*。合成图像 $\mathbf{I}^{\text{syn}}$ 由来自当前模型 $p(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda};\beta)$ 的 MCMC 样本生成。下面给出 FRAME 学习算法的草图。有关 FRAME 合成的图像示例，请参阅第 10.2.4节。

FRAME 算法

输入: 观察到的纹理图像 $\mathbf{I}^{\text{obs}}$, 滤波组件 $\mathscr{F} = \{F_1, \ldots, F_K\}$, Gibbs 扫描量 S, 步长 $\delta > 0$, 收敛误差 $\varepsilon > 0$.
输出: MLE $\beta^* = ((\beta^{(1)})^*, \ldots, (\beta^{(K)})^*)$ 以及合成图像 $\mathbf{I}^{\text{syn}}$.

1. 通过滤波器与 $\{F_k\}_{k=1}^K$ 在每个位置 $(x,y) \in \Lambda$ 的卷积计算 $\mathbf{h}(\mathbf{I}_\Lambda^{\text{obs}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}}) = \{\mathbf{h}_k(\mathbf{I}_\Lambda^{\text{obs}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}})\}_{k=1}^K$.
2. 初始化 $\beta_0^{(k)} = 0$ for $k = 1, \ldots, K$ 和 $\mathbf{I}^{\text{syn}}$ 为均匀的白噪声.
3. 重复:
   (a) 计算 $\mathbf{h}(\mathbf{I}_\Lambda^{\text{syn}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}}) = \{\mathbf{h}_k(\mathbf{I}_\Lambda^{\text{syn}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}})\}_{k=1}^K$.
   (b) 更新 β，根据
   $$\beta_t = \beta_{t-1} + \delta\widetilde{\nabla}\mathcal{G}(\beta_{t-1})$$
   其中 $\widetilde{\nabla}\mathcal{G}(\beta)$ 是 (10.23) 中的随机梯度.
   (c) 应用 Gibbs 采样器，使用 (10.21) 中密度 $p(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda};\beta_t)$ 对 S 次扫描更新 $\mathbf{I}^{\text{syn}}$.

直到: $\frac{1}{2}\sum_{k=1}^K ||\mathbf{h}_k(\mathbf{I}_\Lambda^{\text{obs}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}}) - \mathbf{h}_k(\mathbf{I}_\Lambda^{\text{syn}}|\mathbf{I}_{\partial\Lambda}^{\text{obs}})||_1 < \varepsilon$.

10.2.2 FRAME 的替代学习方法

训练 FRAME 模型时的主要障碍是用于在每次更新 之后合成图像的 MCMC 步骤的计算成本。由于直方图特征在 **I** 中不可区分，因此 HMC 不是可行的选择，必须使用单站点 Gibbs 采样。这种为随机梯度学习生成负样本的方法可能非常慢。通过减少评估 $\text{E}_{p(I_\Lambda|I_{\partial\Lambda};\beta)}[\mathbf{h}(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda})]$ 的计算负担，可以使用原始 FRAME 算法的变体来提高训练速度。

在原始 FRAME 算法中使用 MCMC 采样来绕过难以处理的偏函数 $Z(\mathbf{I}_{\partial\Lambda},\beta)$ 的计算。FRAME 算法的变体利用纹理图像的局部结构和 FRAME 势的线性结构 $-\langle\beta,\mathbf{h}(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda})\rangle$ 来找到用于近似 $Z(\mathbf{I}_{\partial\Lambda},\beta)$ 的替代方法。本节介绍了设计 FRAME 算法替代方案的两个原则，下一节介绍了四种变体算法。

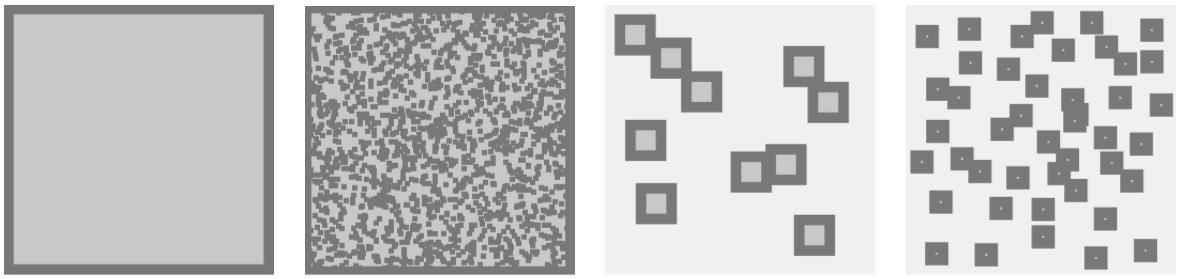
设计准则 I: 数量, 大小, 和前景形状 patches.

第一个设计准则是将完整图像点阵 Λ 分成一组较小的，可能重叠的子点阵 $\{\Lambda_i\}_{i=1}^M$。子点阵的大小和形状可以是任意的。然后将联合对数似然定义为 patch 对数似然的总和:

$$\mathcal{G}_{\text{patch}}(\beta) = \sum_{j=1}^M \log p(\mathbf{I}_{\Lambda_i}^{\text{obs}}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta). \tag{10.24}$$

将图像点阵分成较小的块有助于提高训练期间的采样效率。固定背景像素可以显着提高 MCMC 样本的混合速率，而不会大大降低合成图像的精确度。

图 **??**显示了 $\{\Lambda_i\}_{i=1}^M$ 的四种典型选择。较亮的像素位于前景Λ_i 中，其被背景 $\partial\Lambda_i$ 中的较暗像素围绕。在情况（a），（c）和（d）中，Λ_i 是具有 $m \times m$ 个像素的方形贴片。在一个极端情况下，图 **??**(a) 选择一个由 Λ_1 表示的最大贴片，即 $M = 1$，$m = N - 2w$，其中 w 是边界的宽度。在这种情况下，$\mathcal{G}_{\text{patch}}$ 对



a b c d

图 10.1: $\Lambda_i, i = 1,2,...,M$ 的各种选择. 较亮的像素在前景 Λ_i 中被 $\partial\Lambda_i$ 中较暗的背景像素围绕。$\partial\Lambda_i$ 的宽度取决于最大卷积滤波器的大小。示例相应为：a）似然（原始 FRAME 算法），b）部分似然（算法 I），c）patch 似然（算法 II）或卫星似然（算法 III），以及 d）伪似然（算法 IV）。Zhu 和 Liu[24] 提供.

应于标准对数似然，它采用随机梯度 MLE 方法 ([22, 25] [4, 5, 6]) 在例 10.1中讨论。在另一个极端情况下，图 **??**(d) 选择最小贴片大小 m = 1. 这里 $\mathcal{G}_{\text{patch}}$ 是来自最大伪似然估计 (MPLE)[1] 的对数伪似然。在 MPLE 中，假设图像的联合密度是每个给定其余输入的单变量维数的条件密度的乘积，所以每个像素定义一个独立的贴片。图 **??**(c) 是极端情况 (a) 和 (d) 之间的一个例子，其中 $\mathcal{G}_{\text{patch}}$ 是 *log-patch-likelihood* 或对数-卫星-似然。在第四种情况下，图 **??**(b) 只选择一个 ($M = 1$) 不规则形状的贴片 Λ_1，这是一组随机选择的像素, 其余的像素在背景 $\partial\Lambda_1$ 中。在这种情况下，$\mathcal{G}_{\text{patch}}$ 被称为对数-局部-似然。在图 **??**(c) 和 (d) 中，前景像素可以作为不同贴片的背景。直接证明最大化 $\mathcal{G}(\beta)$ 导致所有四种选择 [7] 的一致估计。

设计准则 II: 用于估计偏函数的参考模型.

不是从每个步骤中的更新模型中采样以获得随机梯度，而是可以使用固定参考模型并使用重要性采样来评估期望 $\mathrm{E}_{p(I_\Lambda|I_{\partial\Lambda};\beta)}[\mathbf{h}(\mathbf{I}_\Lambda|\mathbf{I}_{\partial\Lambda})]$. FRAME 势的线性结构促进了这种方法. 设 β_o 为 FRAME 参数 β 的参考值，因此 $p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i};\beta_o)$ 为参考模型. 假设 $\{\Lambda_i\}_{i=1}^M$ 是前景贴片，并且 $\mathbf{I}_{ij}^{\text{syn}}, j = 1,2,...,L$ 是来自参考模型 $p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta_o)$ 对每个贴片 Λ_i 的典型样本。使用 FRAME 势的线性结构，我们可以通过蒙特卡罗积分与参考模型 β_o 近似分割函数 $Z(\mathbf{I}_{\partial\Lambda_i}^{\text{obs}},\beta)$，如下所示:

$$\begin{aligned} Z(\mathbf{I}_{\partial\Lambda_i}^{\text{obs}},\beta) &= \int \exp\{-\langle\beta,\mathbf{h}(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\rangle\}d\mathbf{I}_{\Lambda_i} \\ &= Z(\mathbf{I}_{\partial\Lambda_i},\beta_o)\int \exp\{-\langle\beta-\beta_o,\mathbf{h}(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\rangle\}p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta_o)d\mathbf{I}_{\Lambda_i} \\ &\approx \frac{Z(\mathbf{I}_{\partial\Lambda_i}^{\text{obs}},\beta_o)}{L}\sum_{j=1}^{L}\exp\{-\langle\beta-\beta_o,\mathbf{h}(\mathbf{I}_{ij}^{\text{syn}}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\rangle\}. \end{aligned} \tag{10.25}$$

我们可以使用它来获得直方图期望的替代近似值:

$$\begin{aligned} \mathrm{E}_{p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta)}\left[\mathbf{h}(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\right] &= \mathrm{E}_{p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta_o)}\left[\mathbf{h}(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\frac{p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta)}{p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta_o)}\right] \\ &= \mathrm{E}_{p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta_o)}\left[\mathbf{h}(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\exp\{-\langle\beta-\beta_o,\mathbf{h}(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\rangle\}\frac{Z(\mathbf{I}_{\partial\Lambda_i}^{\text{obs}},\beta_o)}{Z(\mathbf{I}_{\partial\Lambda_i}^{\text{obs}},\beta)}\right] \end{aligned}$$

$$\approx \mathrm{E}_{p(\mathbf{I}_{\Lambda_i}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i};\beta_o)}\left[L\,\omega_{ij}\,\mathbf{h}(\mathbf{I}_{\Lambda_i}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i})\right]$$
$$\approx \sum_{j=1}^{L}\omega_{ij}\,\mathbf{h}(\mathbf{I}^{\mathrm{syn}}_{ij}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i})$$

其中 ω_{ij} 是样本 $\mathbf{I}^{\mathrm{syn}}_{ij}$ 的权重:

$$\omega_{ij}=\frac{\exp\{-\langle\beta-\beta_o,\mathbf{h}(\mathbf{I}^{\mathrm{syn}}_{ij}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i})\rangle\}}{\sum_{j'=1}^{L}\exp\{-\langle\beta-\beta_o,\mathbf{h}(\mathbf{I}^{\mathrm{syn}}_{ij'}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i})\rangle\}}.$$

然后可以用随机梯度完成优化 (10.24)

$$\widetilde{\nabla}\mathcal{G}_{\mathrm{patch}}(\beta)=\sum_{i=1}^{M}\left\{\sum_{j=1}^{L}\omega_{ij}\mathbf{h}(\mathbf{I}^{\mathrm{syn}}_{ij}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i})-\mathbf{h}(\mathbf{I}^{\mathrm{obs}}_{\Lambda_i}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i})\right\}. \tag{10.26}$$

参考模型 $p(\mathbf{I}_{\Lambda_i}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i};\beta_o)$ 的选择取决于贴片 Λ_i 的大小。通常，重要性采样仅在两个分布 $p(\mathbf{I}_{\Lambda_i}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i};\beta_o)$ 和 $p(\mathbf{I}_{\Lambda_i}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i};\beta)$ 重叠时才有效。在极端情况 $m=1$ 的情况下，MPLE 方法 [1] 选择 $\beta_o=0$ 并将 $p(\mathbf{I}_{\Lambda_i}|\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i};\beta_o)$ 设置为均匀分布。在这种情况下，可以精确地计算 $Z(\mathbf{I}^{\mathrm{obs}}_{\partial\Lambda_i},\beta)$。在大前景 $m=N-2w$ 的另一种极端情况下，随机梯度 MLE 方法必须选择 $\beta_o=\beta$ 以便获得合理的近似。因此，两种方法必须从 $\beta_0=0$ 开始迭代地采样 $p(\mathbf{I};\beta)$。

10.2.3 FRAME 算法的四个变种

算法 I: 最大化偏似然.

我们通过随机选择一定百分比（比如 30%）的像素作为前景 Λ_1 来选择图 **??**中所示的点阵，其余的被视为背景 Λ/Λ_1。我们定义了一个对数偏似然

$$\mathcal{G}_1(\beta)=\log p(\mathbf{I}^{\mathrm{obs}}_{\Lambda_1}|\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1};\beta).$$

通过梯度下降最大化 $\mathcal{G}_1(\beta)$, 我们迭代地更新 β:

$$\nabla\mathcal{G}_1(\beta)=\mathrm{E}_{p(\mathbf{I}_{\Lambda_1}|\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1};\beta)}[\mathbf{h}(\mathbf{I}_{\Lambda_1}|\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1})]-\mathbf{h}(\mathbf{I}^{\mathrm{obs}}_{\Lambda_1}|\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1})\approx\mathbf{h}(\mathbf{I}^{\mathrm{syn}}_{\Lambda_1}|\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1})-\mathbf{h}(\mathbf{I}^{\mathrm{obs}}_{\Lambda_1}|\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1}), \tag{10.27}$$

其中 $\mathbf{I}^{\mathrm{syn}}$ 是来自 $p(\mathbf{I}_{\Lambda_1}|\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1};\beta)$ 的一个 MCMC 样本。

该算法遵循与 FRAME[25] 中的原始方法相同的过程。它以一种比 FRAME[25] 中的原始算法更好的方式在准确度和速度之间进行折衷。对数似然可能性具有比对数似然更低的费希尔信息，但是我们的实验证明它比原始的最大最小化学习方法快 25 倍而不会失去很多准确性。我们观察到这种加速的原因是, 原始采样方法 [25] 花费其大部分时间在白色噪声图像开始的“非典型”边界条件下合成 $\mathbf{I}^{\mathrm{syn}}_{\Lambda_1}$。相反，新算法适用于典型的边界条件 $\mathbf{I}^{\mathrm{obs}}_{\Lambda/\Lambda_1}$，其中 Gibbs 模型 $p(\mathbf{I};\beta)$ 的概率质量被聚焦。速度似乎由前景点阵的直径决定，其直径由可以适合前景点阵的最大圆测量。

算法 II: 最大化贴片似然.

算法 II 从 $\mathbf{I}_{\Lambda}^{\text{obs}}$ 中选择一组 M 个重叠的贴片，并在每个贴片上" 挖" 一个孔 Λ_i，如图 **??**.b 所示。因此我们定义一个贴片对数-似然

$$\mathcal{G}_2(\beta) = \sum_{i=1}^{M} \log p(\mathbf{I}_{\Lambda_i}^{\text{obs}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}; \beta).$$

通过随机梯度最大化 $\mathcal{G}_2(\beta)$, 我们按照算法 1 迭代地更新 β:

$$\nabla \mathcal{G}_2(\beta) = \sum_{i=1}^{M} \mathrm{E}_{p(\mathbf{I}_{\Lambda_i} | \mathbf{I}_{\Lambda/\Lambda_i}^{\text{obs}}; \beta)} [\mathbf{h}(\mathbf{I}_{\Lambda_i}^{\text{syn}} | \mathbf{I}_{\Lambda/\Lambda_i}^{\text{obs}})] - \sum_{i=1}^{M} \mathbf{h}(\mathbf{I}_{\Lambda_i}^{\text{obs}} | \mathbf{I}_{\Lambda/\Lambda_i}^{\text{obs}}) \approx \sum_{i=1}^{M} \mathbf{h}(\mathbf{I}_{\Lambda_i}^{\text{syn}} | \mathbf{I}_{\Lambda/\Lambda_i}^{\text{obs}}) - \sum_{i=1}^{M} \mathbf{h}(\mathbf{I}_{\Lambda_i}^{\text{obs}} | \mathbf{I}_{\Lambda/\Lambda_i}^{\text{obs}}). \tag{10.28}$$

与算法 I 相比，点阵的直径被均匀地控制了。算法 II 具有和算法 I 类似的性能。

算法 III: 最大化卫星似然.

算法 I 和 II 仍然需要为每个参数更新合成图像，这是计算密集型任务。现在我们提出第三种算法，它能够以几秒的速度近似计算 β，而无需在线合成图像。我们选择 Gibbs 模型 $p(\mathbf{I};\beta)$ 所属的指数族 Ω 中的一组参考模型 $\mathscr{R}$:

$$\mathscr{R} = \{p(\mathbf{I};\beta_j) \,:\, \beta_j \in \Omega, j = 1,2,...,s\}.$$

我们使用 MCMC 离线采样（或合成）每个参考模型的一个大的典型图像 $\mathbf{I}_j^{\text{syn}} \sim p(\mathbf{I};\beta_j)$。这些参考模型从 Ω 中的不同“视角”估计 β。通过类比全球定位系统，我们将参考模型称为“卫星”。参考模型是允许估计任意系统的已知系统，就像已知的卫星位置可用于估计任意位置一样。

The 对数-卫星-似然

$$\mathcal{G}_3(\beta) = \sum_{i=1}^{M} \log p(\mathbf{I}_{\Lambda_i}^{\text{obs}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}; \beta).$$

与对数-贴片-似然相同,但我们将在参考模型 $p(\mathbf{I}_{\Lambda_i} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}; \beta_j)$ 上使用重要性采样来评估期望值 $\mathrm{E}_{p(\mathbf{I}_{\Lambda_i} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}; \beta)} \left[\mathbf{h}(\mathbf{I}_{\Lambda_i} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\right]$. 正如在第 10.2.2节中所讨论的一样, 我们可以使用近似值:

$$\mathrm{E}_{p(\mathbf{I}_{\Lambda_i} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}; \beta)} \left[\mathbf{h}(\mathbf{I}_{\Lambda_i} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}})\right] \approx \sum_{\ell=1}^{L} \frac{\exp\{-\langle \beta - \beta_o, \mathbf{h}(\mathbf{I}_{ij\ell}^{\text{syn}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}) \rangle\}}{\sum_{\ell'=1}^{L} \exp\{-\langle \beta - \beta_o, \mathbf{h}(\mathbf{I}_{ij\ell'}^{\text{syn}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}) \rangle\}} \mathbf{h}(\mathbf{I}_{ij\ell}^{\text{syn}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}})$$

其中 $\mathbf{I}_{ij\ell}^{\text{syn}}$ 是来自 $p(\mathbf{I}_{\Lambda_i} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}; \beta_j)$ 的一个 MCMC 样本,ω_{ij} 是样本 $\mathbf{I}_{ij\ell}^{\text{syn}}$ 的权重:

$$\omega_{ij\ell} = \frac{\exp\{-\langle \beta - \beta_j, \mathbf{h}(\mathbf{I}_{ij\ell}^{\text{syn}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}) \rangle\}}{\sum_{\ell'=1}^{L} \exp\{-\langle \beta - \beta_j, \mathbf{h}(\mathbf{I}_{ij\ell'}^{\text{syn}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}) \rangle\}}.$$

按照 10.2.2节中的方法，我们用随机梯度最大化 $\mathcal{G}_3(\beta)$

$$\widetilde{\nabla} \mathcal{G}_3(\beta) = \sum_{j=1}^{s} \left\{ \sum_{i=1}^{M} \left[\sum_{\ell=1}^{L} \omega_{ij} \mathbf{h}(\mathbf{I}_{ij\ell}^{\text{syn}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}) \; - \; \mathbf{h}(\mathbf{I}_{\Lambda_i}^{\text{obs}} | \mathbf{I}_{\partial\Lambda_i}^{\text{obs}}) \right] \right\}. \tag{10.29}$$

对于每个前景贴片 Λ_i 和每个参考模型 $p(\mathbf{I};\beta_j)$，我们需要生成一组 L 个合成贴片 $\mathscr{I}_{ij}^{\text{syn}} = \{\mathbf{I}_{ij\ell}^{\text{syn}}; \ell =$

$1,2,...,L,\forall i,j\}$ 以填充 Λ_i 直方图频率的计算。有两种生成 $\mathcal{I}_{ij}^{\text{syn}}$ 的方法:

1. 从条件分布 $\mathbf{I}_{ij\ell}^{\text{syn}} \sim p(\mathbf{I}_{\Lambda_i}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}};\beta_j)$ 采样. 这很昂贵，必须在线计算.

2. 从边缘分布 $\mathbf{I}_{ij\ell}^{\text{syn}} \sim p(\mathbf{I}_{\Lambda_i};\beta_j)$ 采样. 实际上, 这只是用从离线计算的合成图像 $\mathbf{I}_j^{\text{syn}}$ 中随机选择的贴片案来填充孔。

在实验中，我们尝试了两种情况，发现中等大小 $m\times m$ 点阵的差异非常小，比如 $4\le m\le 13$。第二种情况引出了一种有用的训练算法，因为它允许我们绕过 MCMC 采样同时学习 β。当预合成图像用于参数更新时，等式 (10.29) 以秒速收敛于平均纹理模型。

但是，我们应该意识到对数卫星似然 $\mathcal{G}_3(\beta)$ 可能不是上限的风险。当 $\mathbf{h}(\mathbf{I}_{\Lambda_i}^{\text{obs}}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})$ 不能通过采样的贴片 $\sum_{\ell=1}^{L}\omega_{ij}\mathbf{h}(\mathbf{I}_{ij\ell}^{\text{syn}}|\mathbf{I}_{\partial\Lambda_i}^{\text{obs}})$ 的统计量的线性组合来描述时，会发生这种情况。当发生时，β 不会收敛。我们可以通过在 H_{ij}^{syn} 中包含观察到的贴片 $\mathbf{I}_{\Lambda_i}^{\text{obs}}$ 来处理这个问题，这样可以确保卫星似然总是在上界。直观地说，让 $\mathbf{I}_{ij1}^{\text{syn}}=\mathbf{I}_{\Lambda_i}^{\text{obs}}$。然后学习 β，因此 $\omega_{ij1}\to 1$，$\omega_{ij\ell}\to 0,\forall\ell\neq 1$. 由于 L 通常相对较大，比如说 $L=64$，因此添加一个额外的样本不会污染样本集。

算法 IV: 最大化伪似然.

最大伪似然估计（MPLE）假设目标密度 $p(X;\theta)$ 可以被考虑为

$$p(X;\theta)=\prod_{i=1}^{N}p(X_i|X_{\partial\{i\}};\theta)$$

其中 $\partial\{i\}$ 时单独 $\{i\}$ 的边界. 因此对于前景点阵 Λ，FRAME 对数-伪似然具有形式

$$\mathcal{G}_4(\beta)=\sum_{(x,y)\in\Lambda}\log p(\mathbf{I}_{(x,y)}^{\text{obs}}|\mathbf{I}_{\partial\{(x,y)\}}^{\text{obs}};\beta)$$

. 换句话说, 贴片时每个单一的像素 $\Lambda_i=(x_i,y_i)$.

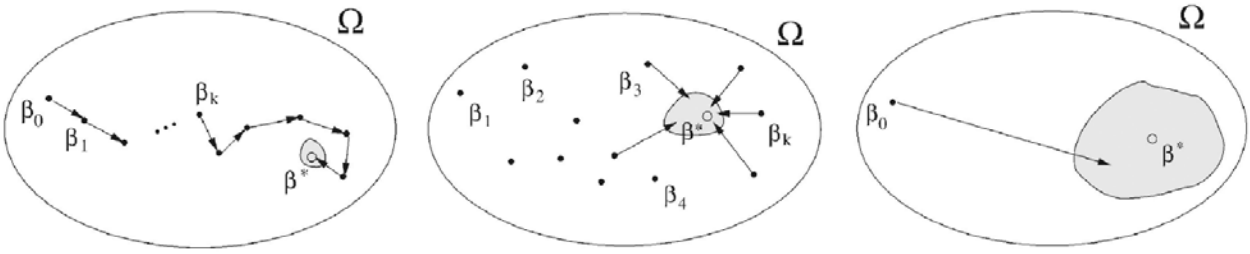
即使在 MPLE 因子分解之后，项 $\log p\left(\mathbf{I}_{(x,y)}^{\text{obs}}|\mathbf{I}_{\partial\{(x,y)\}}^{\text{obs}};\beta\right)$ 也难以评估任意 β，而且如算法 III 那样要求参考分布。MRF 模型的简单选择是平凡的参考模型 $\beta_o=0$. 显然，参考模型的密度 $p_{\beta_o}(I)$ 在图像空间上是均匀的，因此生成参考分布的样本是微不足道的。可以使用 $s=1$ 和 $\beta_1=0$ 的梯度（1.29）来找到最大化伪似然 $\mathcal{G}_4(\beta)$。MPLE 通过使用简单的参考分布来绕开在线图像合成的负担，但计算增益通常以降低采样图像中的真实性为代价。

总之，我们比较了用于估计 MRF 模型的 $\beta^*\in\Omega$ 的不同算法，并将它们分成三组。图 10.2说明了对比。椭圆代表空间 Ω，每个 Gibbs 模型由单个点表示。

第 1 组表示最大似然估计（原始 FRAME 算法）和最大偏/贴片似然估计。如图 10.2(a) 所示，第 1 组方法在线生成并采样一系列“卫星”$\beta_0,\beta_1,...,\beta_k$。这些卫星越来越接近 β^*（假设的真实值）。β^* 周围的阴影区域表示计算 β 的不确定性，其大小可以通过费希尔信息测量。

第 2 组是最大卫星似然估计。该估计器使用一组预先计算并离线采样的卫星，如图 10.2(b) 所示。为了节省时间，可以选择一小部分卫星来计算给定模型。可以根据差异 $\mathbf{h}(\mathbf{I}_j^{\text{syn}})$ 和 $\mathbf{h}(\mathbf{I}^{\text{obs}})$ 选择卫星。差异

越小，卫星越接近估计模型，近似值越好。卫星应均匀分布在 β^* 周围以获得较好的估计。第 3 组是最大伪似然方法，这是第 2 组的特例。如图 10.2(c) 所示，伪似然使用均匀模型 $\beta_o = 0$ 作为“卫星”来估计任何模型，因此具有较大的方差。



Group 1　　Group 2　　Group 3

图 10.2: β^* 周围的阴影区域表示了对数似然函数的估计 β 或效率的方差。第 1 组：原始 FRAME 算法和算法 I 和 II 在线生成一系列卫星以紧密接近 β。贴片尺寸可大可小。第 2 组：最大卫星似然估计器使用普通的一组离线计算的卫星，并可逐步更新。这可以用于小贴片。第 3 组：MPLE 使用单个卫星：$\beta_o = 0$。Zhu 和 Liu[24] 提供.

10.2.4 实验

在本节中，我们在学习 FRAME 纹理模型的上下文中评估各种算法的性能。我们使用 12 个滤波器，包括一个强度滤波器，两个梯度滤波器，三个拉普拉斯高斯滤波器和六个固定比例和不同方向的 Gabor 滤波器。因此，$\mathbf{h}(\mathbf{I})$ 包括 12 个直方图的滤波器响应，并且每个直方图具有 12 个区间，所以 β 具有 12×11 个自由参数。我们选择 15 种自然图像。随机梯度 MLE 估计来自原始 FRAME 算法的 β_j^*（参见第 10.2.1节和 [25]）被视为 ground truth 以进行比较。在将 FRAME 算法应用于每个纹理之后，我们获得了 15 个合成图像 $\mathbf{I}_j^{\text{syn}}$，其可以在算法 III 中用作离线卫星图像。

实验 I: 四种算法的比较.

在第一个实验中，我们比较了纹理合成中五种算法的性能。图 10.3显示了 3 个 128×128 像素的纹理图案。对于每一行，第一列是来自原始 FRAME 模型中使用随机梯度 MLE 方法的“ground-truth”合成图像。其他四个图像使用 10.2.3节中的方法合成。对于算法 I 到 III，我们将前景像素的总数固定为 5,000。对于贴片可能性和卫星可能性，贴片大小固定为 5×5 像素。我们从每个纹理的 14 个可用预先计算的模型中选择 5 个卫星。

对于不同的纹理，模型 $p(\mathbf{I};\beta)$ 可能对 β 的一些元素（例如尾区）比对其余参数更敏感，并且 β 向量在组分之间高度相关。因此，使用误差测量 $|\beta - \beta^*|$ 来比较学习 β 的准确性并不是很有意义。相反，我们对每个学习模型 $\mathbf{I}^{\text{syn}} \sim p(\mathbf{I};\beta)$ 进行采样，并比较合成图像与观察到的直方图误差。损失度量是 $||\mathbf{h}(\mathbf{I}^{\text{syn}}) - \mathbf{h}(\mathbf{I}^{\text{obs}})||_1$，超过 12 对归一化直方图的总和。下表显示了图 10.3中合成图像的每种算法的误差。这些数字受到采样过程中一些计算波动的影响。

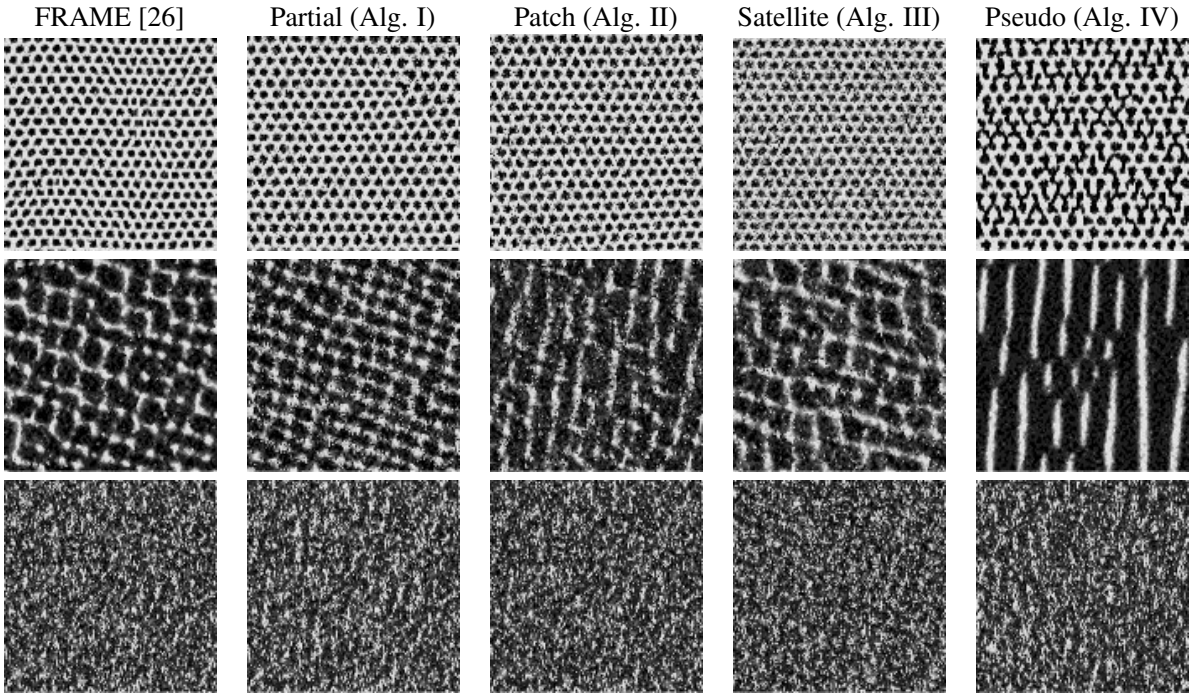


图 10.3: 使用从各种算法学习到的 β 合成纹理图像。从左到右的每一列。1. 使用完全似然作为 ground truth 的随机梯度算法，2。伪似然，3。卫星似然，4。贴片似然，5。偏似然。Zhu 和 Liu [24] 提供.

Fig.4.	FRAME	Pseudo	Satellite	Patch	Partial
Fig.4.a	0.449	2.078	1.704	1.219	1.559
Fig.4.b	0.639	2.555	1.075	1.470	1.790
Fig.4.c	0.225	0.283	0.325	0.291	0.378

实验结果表明，这四种算法运行良好。卫星方法通常接近贴片和偏似然方法，虽然它有时可能比其他方法产生稍好的结果，这取决于参考模型和要学习的模型之间的相似性。伪似然方法还可以捕获一些较大的图像特征。特别地是，它适用于随机性质的纹理，如图 Figure 10.3(c) 中的纹理。

就计算复杂度而言，卫星算法是最快的，并且它在 HP 工作站中以 10 秒的量级计算 β。第二快的是伪似然，需要几分钟。然而，伪似然方法消耗大量内存，因为它需要记住 $N \times N$ 像素中的 g 个灰度级的所有 k 个直方图。空间复杂度为 $O(N^2 \times g \times k \times B)$，$B$ 为区间数，通常需要超过 1 千兆字节的内存。偏似然和贴片似然算法与具有完整 MLE 的随机梯度算法非常相似。由于初始边界条件是典型的，因此这两个估计器通常仅占收敛扫描数的 $1/10^{\text{th}}$。另外，仅需要合成像素点阵的一部分，这可以进一步节省计算。贴片和偏似然算法的合成时间仅仅约为完全似然算法的 $1/20^{\text{th}}$。

实验 II: 分析最大卫星似然估计.

在第二个实验中，我们研究了卫星算法的性能如何受到 1）边界条件，以及 2）贴片尺寸 $m \times m$ 的影响。

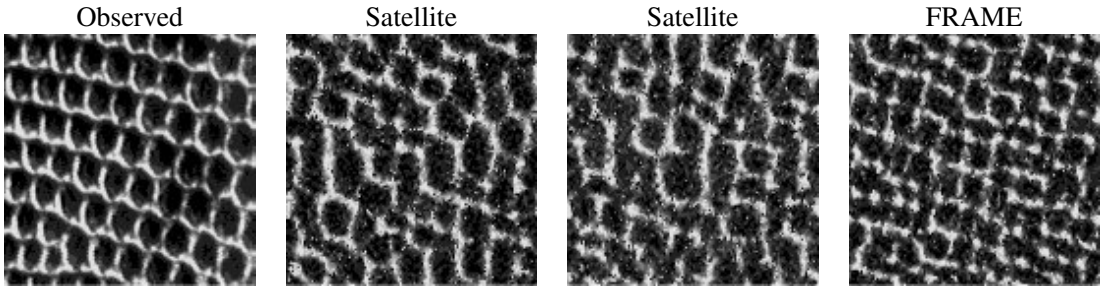


图 10.4: 卫星算法的性能评估。（a）观察到的纹理图像。（b）使用 β 学习的无边界条件的合成图像。（c）使用边界条件学习的 β 的合成图像。（d）使用随机梯度学习的 β 的合成图像。Zhu 和 Liu [24] 提供.

1) 边界条件的影响.

图 10.4.a 将纹理图像显示为 $\mathbf{I}^{\text{obs}}$。我们运行三种算法进行比较。图 10.4(d) 是 FRAME（随机梯度法）的结果。图 10.4(b) 和 (c) 是使用两个版本的卫星算法的结果: 在线和离线。如 10.2.3节算法 III 中所述，生成图 10.4(c) 的算法使用每个贴片的观察边界条件并进行在线采样，而生成图 10.4(b) 的算法忽略边界条件并离线合成图像。结果非常相似，这证明与离线采样的计算增益相比，算法 III 的离线版本的精度损失可以忽略不计。

2) 贴片尺寸 $m \times m$ 的影响.

我们修正前景像素的总数 $\sum_i |\Lambda_i|$ 并且研究不同贴片尺寸 m 的卫星算法的性能。图 10.5(a)-(c) 显示了使用通过卫星算法学习的 β 的三个合成图像，其中不同的贴片尺寸分别为 $m = 2, 6, 9$。从图 10.5(a)-(c) 可以清楚地看出，6×6 像素的贴片尺寸给出了更好的结果。

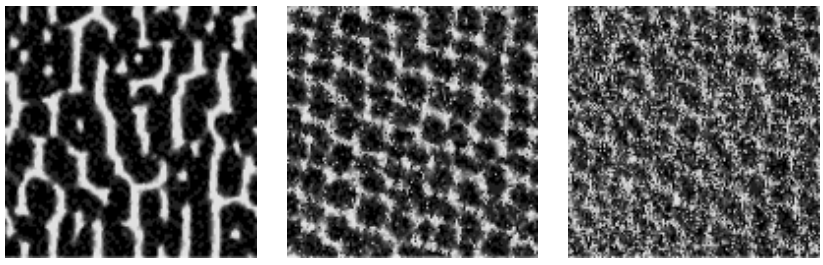


图 10.5: 使用通过具有不同贴片尺寸的卫星方法学习到的 的合成图像。(a) $m = 2$. (b) $m = 6$. (c) $m = 9$。Zhu 和 Liu [24] 提供.

为了解释为什么 $m = 6$ 的贴片大小给出了更好的卫星近似，我们计算了决定性能的两个关键因素，如图 10.6(a) 所示。当贴片尺寸较小时，可以精确估计分割函数，如图 10.6中实线，点划线和虚曲线中小的 $E_p[(\hat{Z} - Z)^2]$ 所示。然而，对于小贴片，方差 $E_f[(\hat{\beta} - \beta^*)^2]$ 较大，如图 10.6(a) 中的虚线所示。因此，贴片尺寸的最佳选择大约是两条曲线的交点。由于我们使用的参考模型接近图 10.6(a) 所示的虚线，我们预测最佳贴片大小在 5×5 和 6×6 之间。图 10.6(b) 显示了合成图像 $\mathbf{I}^{\text{syn}} \sim p(\mathbf{I}; \beta)$ 的统计量与观察到的统计误差之间的平均误差，其中 β 是对 $m = 1, 2, ..., 9$ 使用卫星方法学习的，这里 6×6 像素的贴片尺

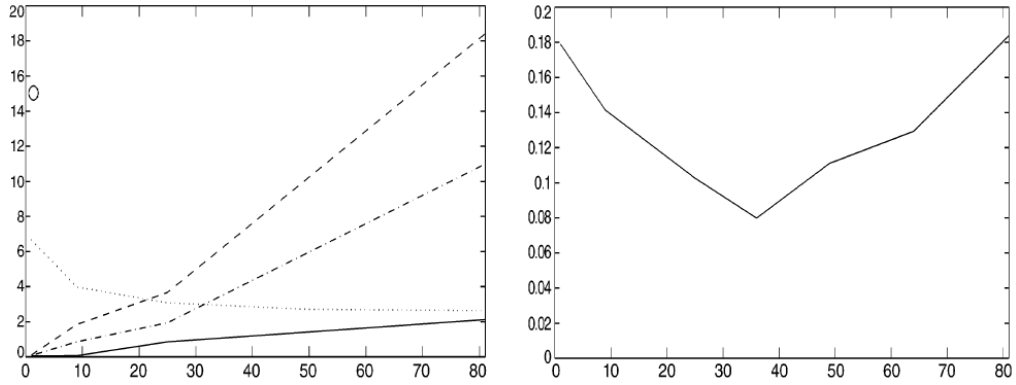


图 10.6: x 轴时贴片尺寸 m^2. a). 针对贴片尺寸 m^2 绘制虚曲线 $E_f[(\hat{\beta}-\beta^*)^2]$. 三个不同的参考模型的实线，点划线和虚曲线是 $E_p[(\hat{Z}-Z)^2]$ .b). 每个滤波器相对于贴片尺寸 m^2 的平均合成误差. Zhu 和 Liu [24] 提供.

寸给出了最好的结果。

10.3 用神经网络学习图像模型

本节介绍了用于学习图像数据深度网络模型权重的随机梯度方法。所有方法的共同点是使用 MCMC 来评估最大似然估计所需的难处理的梯度。由于深度网络函数几乎总是可微分的，因此 Langevin Dynamics（参见第 9.4.3节）通常是生成 MCMC 样本的首选方法。在本节的第一部分，我们介绍了对比发散和持续对比发散，这是用于加速顺序更新的模型中蒙特卡罗样本的收敛两种重要技术。然后，我们将介绍势能图像模型，图像生成器网络以及能量和发生器模型的协作系统的学习技术。

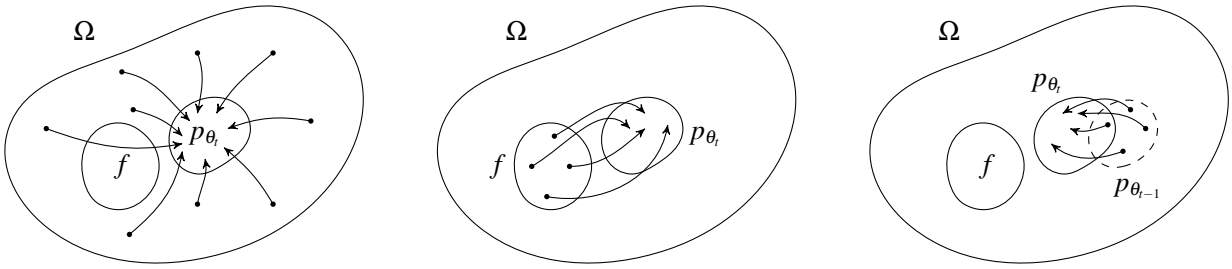
10.3.1 对比发散与持续对比发散

随机梯度 MLE 方法（见例 10.1）使用来自电流分布 $p_\theta(x)$ 的样本 $\{Y_i\}_{i=1}^m$，用等式 (10.8) 中的对数似然梯度更新 θ。由于每次更新 θ 时 $p_\theta(x)$ 都会发生变化，因此每次新的梯度计算都需要新的 MCMC 样本。生成这些样本可能非常耗时。MCMC 样本通常表现出高自相关性且具有很长的混合时间，并且在大多数情况下，对于每个梯度更新，从 $p_\theta(x)$ 产生真正独立的 MCMC 样本是不可行的。另一方面，接近目标分布初始化的 MCMC 样本可以快速收敛，因此用于随机梯度学习的合理初始化方案可以显着节省计算量。

对比发散（CD）[10] 和持续对比发散（PCD）[19] 是初始化 MCMC 样本的两种常用方法。因为它们使用“热启动”初始化，所以 CD 和 PCD 仅需要少量 MCMC 迭代来达到每个梯度计算的近似收敛。甚至在将 CD 和 PCD 正式引入机器学习著作之前，随机梯度方法（如 FRAME 算法 [25]）已经使用了 PCD 而没有给该技术一个明确的名称。

在 CD 学习中，观察到的数据 $\{X_i\}_{i=1}^n$（或是一小批量观察结果）被用作负 MCMC 样本的初始点 $\{Y_i^{(0)}\}_{i=1}^n$。在 PCD 学习中，先前参数更新的最终负样本 $\{\tilde{Y}_i\}_{i=1}^m$ 被用作当前参数更新的初始点 $\{Y_i^{(0)}\}_{i=1}^m$。CD 和 PCD 在初始点上仅使用少量 MCMC 更新 k（甚至 $k=1$）以获得用于等式 (10.8) 中梯度计算的样

本 $\{Y_i^{(k)}\}_{i=1}^m$。从理论角度看，应该使用大量的马尔可夫变换来获得可靠的稳态样本 $\{Y_i^{(\infty)}\}_{i=1}^m$。从计算角度看，来自有意义的初始化的少量转换仍然可以提供用于参数估计的准确梯度信息。有关完全随机梯度 MLE，CD 和 PCD 的视觉比较，请参见图 10.7。



随机梯度 MLE　　CD　　PCD

图 10.7: 在所有图中，Ω 是数据 x 的状态空间，p_{θ_t} 是训练步骤 t 的学习分布，f 是真实数据分布，点是 MCMC 样本的初始点。*左*：完全随机梯度 MLE。MCMC 样本从 Ω 中的随机点初始化，许多马尔可夫更新用于获得可靠的稳态样本。*中*：使用 CD 的近似 MLE。MCMC 样本从训练数据初始化，当 p_{θ_t} 接近真实分布 f 时，仅需要一些更新来进行收敛。*右*：使用 PCD 近似 MLE。从先前学习迭代的 $p_{\theta_{t-1}}$ 的样本，初始化 MCMC 样本。如果 θ_t 和 θ_{t-1} 之间的差异很小，则 MCMC 样本应该仅在几次马尔可夫更新后就会收敛。

由于 CD 的初始图像来自真实分布，因此在训练结束时，当学习的分布接近真实分布时，仅应使用少量 MCMC 更新。另一方面，PCD 初始图像来自先前学习模型的分布。只要参数更新很小，先前的模型应该接近当前模型，并且 PCD 更新假设应该在整个训练过程中被证明是合理的。在某些情况下，PCD 似乎具有优于 CD 的性质，因为初始样本更接近当前模型，且初始样本在整个训练过程中变化。在随机梯度学习中使用 CD 或 PCD 几乎是不可避免的，但这两种方法都会在训练过程中引入难以分析的额外误差。

10.3.2 使用深度网络学习图像的势能: DeepFRAME

DeepFRAME 模型（见第 9.6.3节例 9.2和 [15, 20]) 扩展了 FRAME 模型，以融入深度学习的见解。这些模型之间存在两个主要差异。首先，DeepFRAME 势是线性层和激活函数的非线性组合。深度网络电势比原始 FRAME 中使用的线性电势更具表现力。其次，DeepGRAME 模型的过滤器是在训练期间学习的，与 FRAME 模型中使用的固定手选过滤器相反。在训练期间自己学习滤波器对于灵活的图像表示是必不可少的，因为手工设计捕获复杂图像的相关特征滤波器是非常困难的。

具有权重参数 w 的 DeepFRAME 密度具有形式

$$p_w(I) = \frac{1}{Z}\exp\{F(I;w)\}q(I), \tag{10.30}$$

其中 q 是一个高斯先验分布 $\mathrm{N}(0, \sigma^2 \mathrm{Id})$. DeepFRAME 势能函数为

$$U(I;w) = -F(I;w) + \frac{||I||^2}{2\sigma^2}. \tag{10.31}$$

学习 DeepFRAME 密度的权重 w 可以通过遵循例 10.1中所概述的随机梯度 MLE 方法来完成。实际上，学习 DeepFRAME 模型几乎与学习 FRAME 模型完全相同。两种方法通常使用 PCD 进行训练，这意味着来自先前参数更新的 MCMC 样本，用作当前参数更新的 MCMC 样本的初始化。原始 FRAME 模型在离散图像上使用 Gibbs 采样进行 MCMC 更新。在训练 DeepFRAME 模型时，Langevin 动力学用于在连续空间中更新 MCMC 图像样本。图像更新具有形式

$$I_{t+1} = I_t + \frac{\varepsilon^2}{2}\left(\frac{\partial}{\partial I} F(I;w) - \frac{I}{\sigma^2}\right) + \varepsilon Z_t \tag{10.32}$$

其中 $Z_t \sim \mathrm{N}(0, \tau^2 \mathrm{Id})$ 是动量. 权重 w 可以使用等式 (10.8) 更新:

$$\widetilde{\nabla} l(w) = \frac{1}{n}\sum_{i=1}^{n} \frac{\partial}{\partial w} F(X_i;w) - \frac{1}{m}\sum_{i=1}^{m} \frac{\partial}{\partial w} F(Y_i;w) \tag{10.33}$$

其中 $\{X_i\}_{i=1}^n$ 是一组训练图像，$\{Y_i\}_{i=1}^m$ 是 $p_w(I)$ 的负样本. 下面给出 DeepFRAME 学习算法的概述。

DeepFRAME 算法

输入: 观测的图像 $\{X_i\}_{i=1}^n$, Langevin 迭代次数 S, 步长 $\delta > 0$, 学习迭代次数 T.
输出: MLE w^*, 合成图像 $\{Y_i\}_{i=1}^m$.

1. 初始化权重 w_0 和负样本 $\{Y_i\}_{i=1}^m$ 为白噪声.
2. *For* $t = 1 : T$*:*
   (a) 使用等式 (10.32), 将 S Langevin 更新应用于当前模型 $p_{w_{t-1}}(I)$ 下的持久负样本 $\{Y_i\}_{i=1}^m$.
   (b) 更新 w，根据
   $$w_t = w_{t-1} + \delta \widetilde{\nabla} l(w_{t-1})$$
   其中 $\widetilde{\nabla} l(w)$ 是 (10.33) 中的随机梯度.

下面介绍了显示 DeepFRAME 模型能力的四个实验。

DeepFRAME 实验 1: 合成纹理图像 (Figure 10.8)

在第一个实验中，由单个训练纹理图像合成新的纹理图像。网络中较低级别的过滤器学习合成局部纹理特征，而较高级别过滤器确定训练纹理中观察到的局部特征的组成。DeepFRAME 电势的非线性结构提高了原始 FRAME 模型的综合能力。

图 10.8: DeepFRAME 实验 1. 在每组三个图像中，左边的图像是训练纹理，中间和右边的图像是从 DeepFRAME 电势合成的。

DeepFRAME 实验 2: 合成目标图像 (Figure 10.9)

只要训练图像具有一致的对齐，DeepFRAME 模型就可以学习合成目标图像。ConvNet 评分函数 $F(I;w)$ 的最后一层应该完全连接，应完全连接以强制执行能量函数的几何一致性。合成的目标图像与训练数据的外观和对齐一致。

DeepFRAME 实验 3: 合成混合图像 (Figure 10.10)

当来自不同类别的对齐图像用作训练数据时，DeepFRAME 模型学习合成混合图像，其组合训练类别的不同局部特征。虽然整体图像形状与训练数据的对齐一致，但是在训练图像中看不到新奇的特征组合出现在合成图像中。

DeepFRAME 实验 4: 图像重构 (Figure 10.11)

由于 DeepFRAME 函数的能量函数是真实图像密度的近似值，因此来自真实分布的未见图像应接近学习密度的局部模式。像 Hopfield[] 最初描述的那样，图像空间上的势能可以被解释为存储器。在 DeepFRAME 电势上的 MCMC 过程下的图像演变将被吸引到最近的局部模式。如果初始图像类似于训练数据，则 MCMC 过程应该收敛到看起来类似于初始图像的模式。另一方面，从与训练数据不同的图像初始化的 MCMC 样本，在样本中将发生很大变化，直到它类似于训练图像。

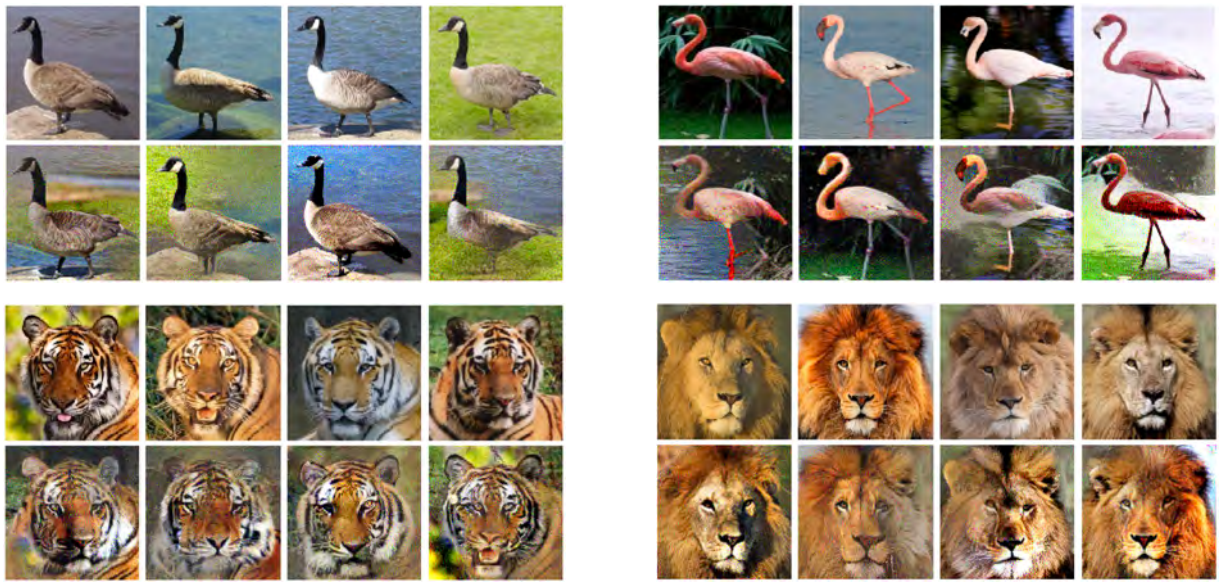


图 10.9: DeepFRAME 实验 2. 在每个图像组中，顶行显示训练图像，底行显示合成图像。

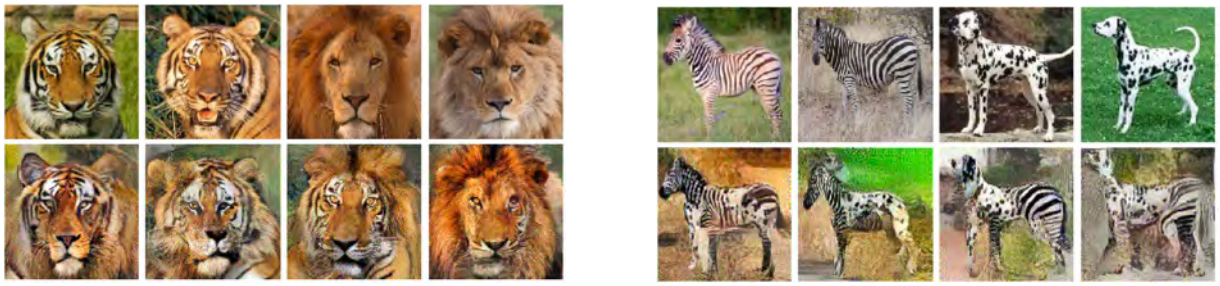


图 10.10: DeepFRAME 实验 3. 在每个图像组中，顶行显示训练图像，底行显示合成图像。

10.3.3 发生器网络和交替后向传播

势能函数，如 DeepFRAME，是几个家庭图像模型之一。发生器网络是另一种重要的图像模型。这些网络概括了经典因子分析模型，可用于从简单信号生成逼真的图像。与需要 MCMC 进行图像合成的势能模型相比，发生器网络可以直接从潜在输入合成图像。

设 $Z \in \mathbb{R}^d$ 为具有平反分布的潜在因素 (即一致或 $\mathrm{N}(0, I_d)$)。实际上，Z 的尺寸 d 通常为 100 或更小。具有权重 w 的生成器网络 $G(Z;w)$ 定义从 Z 的潜在空间到高维图像空间 $\mathbb{R}^D$ 的变换。可以学习生成器网络的权重 w，使得合成图像 $G(Z;w) \in \mathbb{R}^D$ 匹配 z 的潜在分布中训练数据集的外观。

经典因子分析模型使用线性发生器 $G(Z;w) = WZ + \varepsilon$。深度生成器网络通过在线性层之间包含激活函数，来在 G 中引入非线性。生成器网络可以被理解为具有以下关系的递归因子分析模型

$$Z_{l-1} = f_l(W_l Z_l + b_l) \tag{10.34}$$

图 10.11: DeepFRAME 实验 4. 顶行是训练期间未使用的真实图像，底行是单次 Langevin 更新后的重建图像。

其中 f_l 是一个非线性激活 (通常是 ReLU), (W_l, b_l) 是来自 l 层的权重和偏差, $w = \{(W_l, b_l) : l = 1, \ldots, L\}$, $Z_L = Z, Z_0 = G(Z; w)$. 隐藏因子层 Z_{l-1} 是 W_l 列与系数 Z_l 的线性组合，加上变换和激活。如果使用 ReLu 激活，则 $G(Z; w)$ 是分段线性函数，并且线性区域之间的边界对应于 f_l 的激活边界。G 的非线性结构对于生成逼真的图像是必不可少的。

交替后向传播（ABP）算法 [9] 是用于学习发生器网络的权重 w 的一种方法。顾名思义，ABP 算法有两个阶段。在第一个阶段，使用 Langevin 动力学推断该组训练图像的潜在因素。在第二阶段中，基于新的潜在因素更新将潜在因素变换为图像的权重。由于在训练过程中推断出潜在因素，因此 ABP 算法执行无监督学习。ABP 算法与 EM 算法密切相关：第一阶段对应于 EM 算法的 E 阶段，其中期望值基于当前参数评估，第二阶段对应于 M 阶段，其中参数被调整解释预期因子。

在 ABP 学习中，给定 d 维潜在因素 Z，其 D 维图像 X 的条件分布被定义为

$$X|Z \sim \mathrm{N}(G(Z; w), \sigma^2 \mathrm{Id}_D)$$

对于具有权重 w 的生成器网络 $G(Z; w)$. 由于 Z 和 $X|Z$ 都是多元正态变量, 他们的联合能量函数具有形式

$$U(X, Z; W) = \frac{1}{2\sigma^2}||X - G(Z; w)||^2 + \frac{1}{2}||Z||^2$$

这只是 Z 和 $X|Z$ 的高斯能量函数之和。条件变量 $Z|X$ 的能量函数是 $U_{Z|X=x;w}(z) = U(z, x; w)$，因为后验分布 $Z|Y$ 与 Y 和 Z 的联合分布成比例。给定一组完整的观察值 $\{X_i, Z_i\}_{i=1}^n$，可以通过梯度上升最大化对数似然

$$l(w, \{Z_i\}) = \frac{1}{n}\sum_{i=1}^{n} \log p(X_i, Z_i; w) = -\frac{1}{n}\sum_{i=1}^{n} U(X_i, Z_i) + \text{常量}$$

来估计权重 w。由于归一化常数不依赖于 w，因此完整数据对数似然不需要随机梯度。然而，如在 EM 算法中，潜在因素 $\{Z\}_{i=1}^n$ 是未知的，并且必须通过最大化观察到的数据对数似然来学习 w，这对应于最

大化函数。

$$l(w) = \sum_{i=1}^{n} \log p(X_i; w) = \sum_{i=1}^{n} \log \int p(X_i, Z; w) dZ$$

将潜在因素整合到联合分布中。这种损失不能直接计算，但是对数似然的梯度可以改写为

$$\begin{aligned}
\frac{\partial}{\partial W} \log p(X; w) &= \frac{1}{p(X; w)} \frac{\partial}{\partial w} p(X; w) \\
&= \frac{1}{p(X; w)} \frac{\partial}{\partial w} \int p(X, Z; w) dZ \\
&= \int \left(\frac{1}{p(X; w)} \frac{\partial}{\partial w} p(X; w) \right) p(Z|X; w) dZ \\
&= \int \left(\frac{\partial}{\partial w} \log p(X; w) \right) p(Z|X; w) dZ \\
&= -\mathrm{E}_{Z|X;w} \left[\frac{\partial}{\partial w} U(X, Z; w) \right],
\end{aligned}$$

因此，可以通过使用当前权重 w 绘制 $Z|X$ 的 MCMC 样本来估计对数似然梯度，所述潜在因子以观察到的数据为条件。Langevin 动力学可用于 $Z|X_i; w$ 中采样，Langevin 更新方程式为

$$Z_{t+1} = Z_t + \frac{\varepsilon^2}{2} \left(\frac{1}{\sigma^2} (X_i - G(Z_t; w)) \frac{\partial}{\partial Z} G(Z_t; w) - Z_t \right) + \varepsilon U_t \tag{10.35}$$

对于 $U_t \sim \mathrm{N}(0, I_d)$ 和步长 ε, 对于 $t = 1, \ldots, T$ 迭代. 对于每个观察到的图像 X_i 推断出一个 Z_i。在训练期间使用 PCD，因此在每个新推论阶段中 MCMC 采样从先前推论阶段的 Z_i 开始。一旦从 $Z|X_i; w$ 采样了 Z_i，就可以在算法的第二阶段更新权重 w

$$\widetilde{\nabla} l(w) = -\frac{1}{n} \sum_{i=1}^{n} \frac{\partial}{\partial w} U(X_i, Z_i; w) = -\frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sigma^2} (X_i - g(Z_i; w)) \frac{\partial}{\partial w} G(Z_i; w) \tag{10.36}$$

推理阶段使用反向传播梯度 $\frac{\partial}{\partial Z} G(Z; w)$，而学习阶段使用反向传播梯度 $\frac{\partial}{\partial w} G(Z; w)$。要求获得 $\frac{\partial}{\partial Z} G(Z; w)$ 所需的计算作为计算 $\frac{\partial}{\partial w} G(Z; w)$ 的一部分，因此两个阶段都可以以类似的方式实现。下面给出算法草图。

交替反向传播算法

输入: 观察到的图像 $\{X_i\}_{i=1}^n$, Langevin 迭代次数 S, 步长 $\delta > 0$, 学习步数 T.
输出: 生成器网络 $G(Z;w)$ 的 MLE w^*.

1. 初始化权重 w_0 为白噪声以及从潜在分布中采样推断出的潜在因素 $\{Z_i\}_{i=1}^m$.

2. *For* $t = 1 : T$*:*

   (a) 使用方程 (10.35), 使用当前权重 w_{t-1} 将 S Langevin 更新应用于持久潜在因素 $\{Z_i\}_{i=1}^n$ 中

   (b) 更新 w, 根据

$$w_t = w_{t-1} + \delta \widetilde{\nabla} l(w_{t-1})$$

   其中 $\widetilde{\nabla} l(w)$ 是 (10.36) 中的随机梯度.

提出了三个不同的实验来显示 ABP 算法的能力。

ABP 实验 1: 生成纹理图案 (Figure 10.12)

设输入 Z 为 $\sqrt{d} \times \sqrt{d}$ 维图像，每个像素遵循标准正态分布。每层的权重由卷积滤波器给出，每层的上采样因子为 2。一旦学习了滤波器，只需增加 Z 的大小并在较大的输入上运行滤波器卷积，就可以直接扩展网络并生成更大的纹理模式。在下面的示例中，Z 是在训练期间重建 224×224 图像的 7×7 图像，而在测试期间 Z 被扩展为 14×14 图像，使用完全相同的权重生成 448×448 图像。

ABP 实验 2: 生成目标模式 (Figure 10.13)

除了潜在因子层必须完全连接，生成目标模式类似于生成纹理模式，因此输入 Z 是 d 维向量而不是 $\sqrt{d} \times \sqrt{d}$ 矩阵。下图显示了 ABP 算法生成的两种不同的目标模式：狮子/老虎脸和人脸。学习网络的潜在空间中的点之间的插值，在沿着训练数据的流形图像空间中给出非线性插值。

ABP 实验 3: 从不完整的数据中学习 (Figure 10.14)

在某些情况下，损坏或遮挡可能导致训练图像丢失一些像素。ABP 算法可以在给定训练集的情况下学习完整图像的生成器模型，其中某些像素被标记为丢失。唯一需要进行的调整是将能量定义为仅观察到的像素总和 Λ^{obs} of X:

$$U_{\text{obs}}(X, Z; w) = \frac{1}{2\sigma^2} \sum_{(x,y) \in \Lambda^{\text{obs}}} \left(X_{(x,y)} - g(Z;w)_{(x,y)}\right)^2 + \frac{1}{2}||Z||^2. \tag{10.37}$$

那么 $Z|X \sim \frac{1}{Z(w)} \exp\{-U_{\text{obs}}(X, Z; w)\}$ 并且可以使用 Langevin 动力学在 $Z|X$ 上推断出被遮挡图像的潜在向量。然后，学习的模型可以完成三个任务:（1）从训练图像中恢复缺失的像素;（2）从测试图像中恢复缺失的像素;（3）从模型中合成新图像。

图 10.12: ABP 实验 1. 原始图像大小为 224×224，旁边的合成图像大小为 448×448.

10.3.4 协作网络和生成器模型

10.3.2节中的 DeepFRAME 模型和 10.3.3 节中的 ABP 模型可以集成到协作学习方案中，其中能量函数 $-F(I;w_F)$ 的权重 w_F 和生成器网络 $G(Z;w_G)$ 的权重 w_G 是联合学习的。用于协作训练 DeepFRAME 模型和 ABP 模型的等式与用于单独训练模型的等式相同。协作学习的创新是在串联训练网络时初始化 MCMC 样本的方式。由每个模型创建的合成图像可用于在下一个学习迭代中跳转启动伙伴模型的采样阶段。在训练 DeepFRAME 和 ABP 模型时，协作学习 [21] 是 CD 和 PCD 的替代方案。

ABP 网络（称为生成器网络）在协作学习中扮演学生的角色，而 DeepFRAME 网络（称为描述符网络）扮演教师的角色。由于生成器模型可以有效地合成接近真实分布的图像，因此来自生成器的样本可以在训练描述符网络时用作 Langevin 更新的初始图像。另一方面，当训练发生器网络时，来自描述符能量的 MCMC 更新可被视为来自“真实”分布的样本。由于描述符修正类似于由生成器网络创建的原始图像，因此从原始潜在向量初始化的 Langevin 样本应该快速收敛以给出描述符修正的良好潜在近似。生成器网络从描述符网络接收指导，而描述符网络将生成器的工作与实际数据进行比较以确定生成器需要学习什么。融合学习算法提供了一种天然的初始化方案，允许两个网络看到更多样的初始样本。除了 MCMC 初始化之外，每个网络的协作训练过程与单独训练相同。下面给出算法草图。

协作训练算法

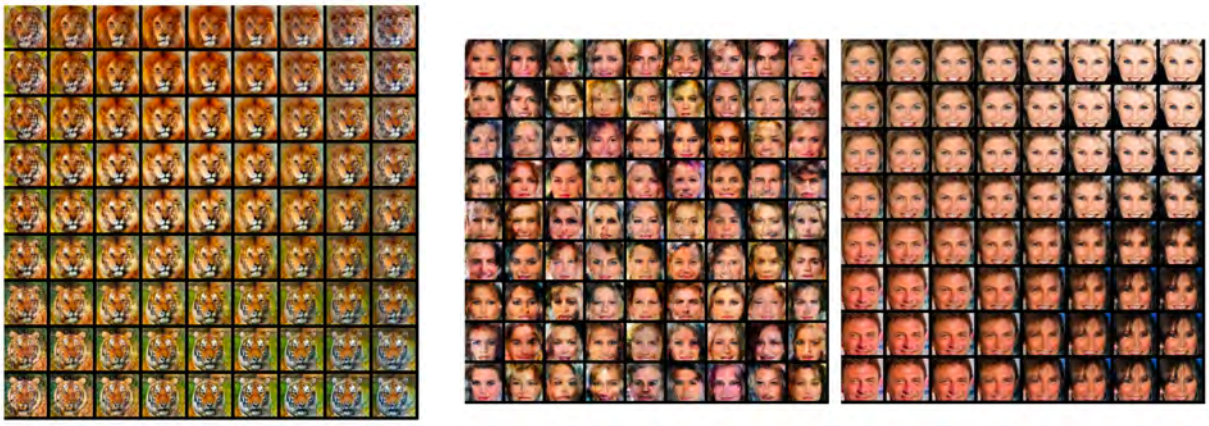


图 10.13: ABP 实验 2. *左:* 狮子/老虎脸生成模型的二维潜在空间的 9×9 离散化。潜伏空间具有分隔狮子和老虎的可识别区域，这些区域之间的插值可以平滑地将狮子脸变成老虎脸. *右:* 来自具有 100 个潜在因子的生成模型的合成人脸。左图显示从学习模型中采样的 81 个脸，右图显示图像四个角上的脸之间潜在空间中的线性插值.

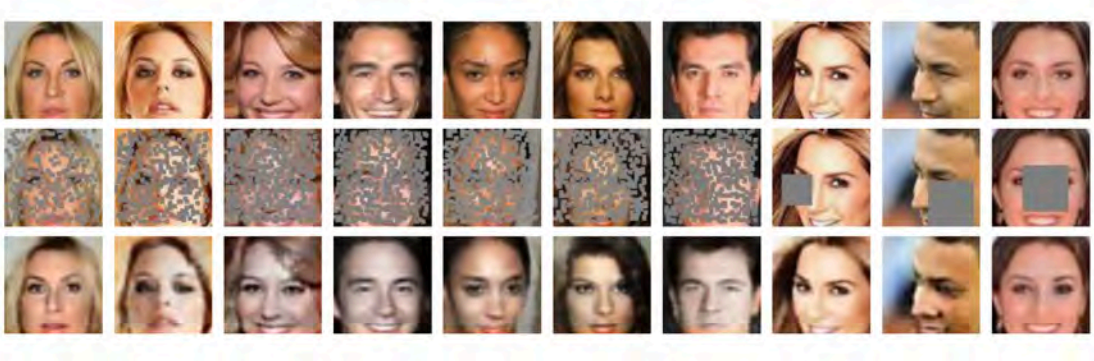


图 10.14: ABP 实验 3. *Top:* 原始图像. *Middle:* 闭塞的训练图像. *Bottom:* 重构的训练图像.

输入: 观察到的图像 $\{X_i\}_{i=1}^n$, 潜在样本数 m, Langevin 迭代次数 S, 描述符步长 $\delta_F > 0$, 生成器步长 $\delta_G > 0$, 学习步数 T.

输出: 描述符网络 F 的权重 w_F^* 和生成器网络 G 的权重 w_G^*.

1. 初始化权重 $w_{F,0}$ 和 $w_{G,0}$.

2. *For* $t = 1 : T$*:*

   (a) 从生成器网络 $G(Z; w_{G,t-1})$ 的潜在分布 $\mathrm{N}(0, I_d)$ 中抽取独立同分布样本 $\{Z_i\}_{i=1}^m$. 计算图像 $\{Y_i\}_{i=1}^m$, 其中 $Y_i = G(Z_i; w_{G,t-1})$.

   (b) 使用等式 (10.32), 将 S Langevin 更新应用于当前能量为 $F(X; w_{F,t-1})$ 的图像 $\{Y_i\}_{i=1}^m$ 中.

   (c) 使用等式 (10.35), 将 S Langevin 更新应用于当前权重为 $w_{G,t-1}$ 的潜在因子 $\{Z_i\}_{i=1}^n$ 中, 其中上一步修正后的 Y_i 是每个 Z_i 的条件图像.

   (d) 对小批量 $\{X_i\}_{i=1}^m$ 的训练数据采样并更新 w_F，根据

$$w_{F,t} = w_{F,t-1} + \delta_F \widetilde{\nabla} l_F(w_{F,t-1})$$

每个网络的缺点由协作伙伴网络的能力弥补。描述符可以轻松修改接近真实数据分布的图像，使其看起来更逼真。然而，描述符很难从白噪声开始生成逼真的图像，因为收敛需要许多次 MCMC 迭代。另一方面，生成器很容易通过简单的正向传递穿过网络合成接近真实数据分布的图像。然而，生成器很难推断出新图像的潜在向量，因为这又需要长时间的 MCMC 推断过程。协作学习解决了这两个缺点。生成器向描述符提供接近真实分布的初始 MCMC 样本，因此描述符修正版快速收敛。由于原始生成器图像类似于描述符修正版，因此对于由生成器执行的潜在因子推断，原始潜在因子是初始化的良好点。协作管道自然地促进了 MCMC 采样的两个阶段。

协作网络与生成对抗网络（GAN）[8] 有关，这是另一种与伙伴网络协同训练生成器网络的方法。如名称所示，GAN 模型的两个网络是在竞争性而非协作性方案中训练的。具体地，训练 GAN（通常称为鉴别器）的描述符 $F(X;w)$ 以区分发生器网络的样本和真实数据的样本。当训练 GAN 时，生成器网络学习欺骗鉴别器网络，而鉴别器网络试图正确地将生成的图像与真实图像分离。GAN 的目标函数是

$$\min_{w_G}\max_{w_D} L(w_G,w_F) = \mathrm{E}_q[\log F(X;w_F)] + \mathrm{E}_{\mathrm{N}(0,I_d)}[\log(1-F(G(Z;w_G);w_F))],$$

其中 q 是真实数据分布，$F(X;w_F)\in(0,1)$ 是判别分类器。人们可以迭代地求解 w_G 和 w_F，其中关于 wG 的梯度上升和关于 wG 的梯度下降的交替更新。第一个期望可以通过采用一小批观察数据来近似，而第二个期望可以通过从潜在正态分布中抽取 Z 样本来近似。

在协作学习中，描述符网络学习训练数据上的潜在能量，该能量指导发生器的合成以匹配训练数据的特征。因此，来自协作学习的能量函数比从 GAN 学习的鉴别器更有意义，因为它是训练数据而不是分类器的非标准化密度。我们可以使用描述符能量来映射协作模型的潜在空间与第**??**节中的能量景观映射技术。相同的技术不能应用于 GAN 模型，因为鉴别器网络在训练后是无用的。图 10.15给出了 DeepFRAME，ABP，Cooperative 和 GAN 模型的视觉比较。

下面介绍了测试协作学习能力的三个实验.

协作学习实验 1: 合成图像 (Figure 10.16)

协作网络可以通过 MCMC 或通过直接前向传递从发生器网络合成来自描述符能量的图像。通常优先选择第二种方法，因为它更有效。可以训练协作网络以合成纹理图像和对齐的目标图像.

协作学习实验 2: 潜在空间插值 (Figure 10.17)

与在 ABP 模型中一样，发生器网络的潜在空间中的线性插值对应于图像空间中的图像之间的非线性但在视觉上直观的插值。生成器网络在图像空间中定义了低维流形，其近似于真实数据分布的流形。移动生成器空间而不是原始图像空间使得映射描述符能量的结构更容易。更多细节参见章节**??**。

协作实验 3: 图像实现 (Figure 10.18)

生成器网络可用于重建具有遮挡或缺失像素的图像。遮挡图像 X_0 被用作分布 $Z|X=X_0$ 的条件图像，并

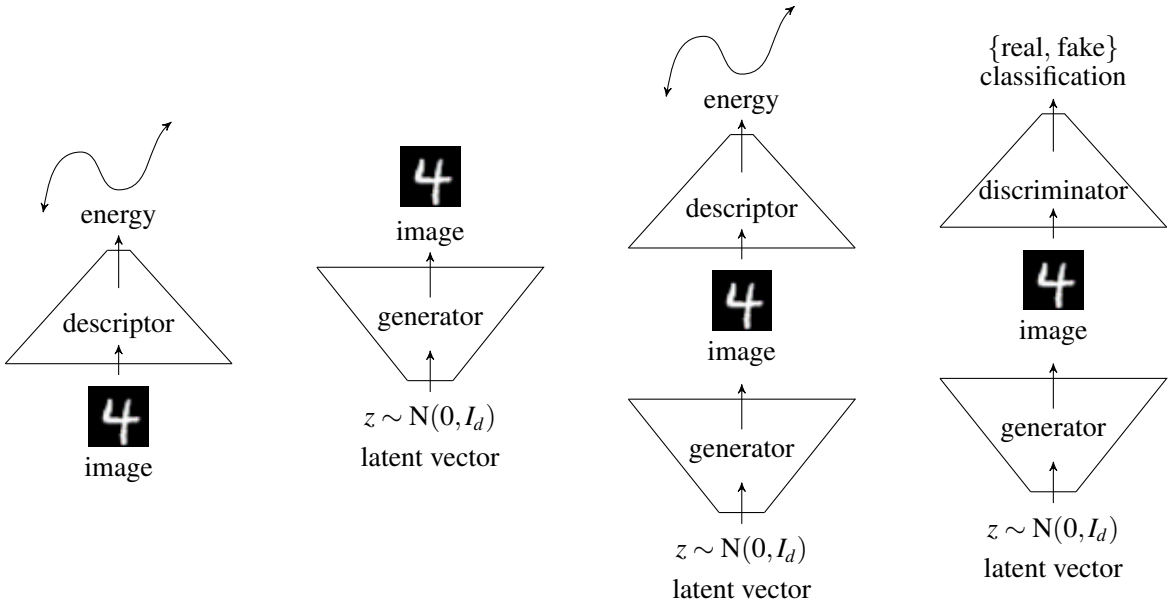


图 10.15: 四种不同图像模型的比较. DeepFRAME 模型将图像作为输入并返回标量能量值。ABP 模型将低维潜在信号转换为逼真图像。协同学习将 DeepFRAME 和 ABP 模型融合在一起，共同训练描述符和生成器网络。GAN 模型类似于协作模型，除了描述符被鉴别器替换，鉴别器将图像分类为真实或伪造而不是返回非标准化密度。协作模型的能量函数比 GAN 的鉴别器更有用，因为它可用于映射图像空间的结构，如第**??**节所述。

且可以通过 Langevin 动力学推断出，给出 X_0 的最佳重建的潜在向量 Z，其具有等式 (10.37) 中的势。

参考文献

[1] J Besag. Efficiency of pseudo-likelihood estimation for simple gaussian fields. *Biometrika*, 64:616–618, 1977.

[2] Augustin Louis Cauchy. Methode generale pour la resolution des systemes d'equations simultanees. *C. R. Acad. Sci. Paris*, (25):536–538, 1847.

[3] Pratik Chaudhari and Soatto Stefano. Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks. *ICLR*, 2018.

[4] Xavier Descombes, Robin Morris, Josiane Zerubia, and Marc Berthod. Maximum likelihood estimation of markov random field parameters using markov chain monte carlo algorithms. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 133–148. Springer, 1997.

图 10.16: 协作学习实验 1. *左:* 景观图像训练后的生成器网络合成. *右:* 在每组图像中, 最左边的图像是训练纹理，其他图像是从生成器网络合成的.

[5] Charles J Geyer. On the convergence of monte carlo maximum likelihood calculations. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 261–274, 1994.

[6] Charles J Geyer and Elizabeth A Thompson. Constrained monte carlo maximum likelihood for dependent data. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 657–699, 1992.

[7] Basilis Gidas. Consistency of maximum likelihood and pseudo-likelihood estimators for gibbs distributions. In *Stochastic differential systems, stochastic control theory and applications*, pages 129–145. 1988.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherijl Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[9] Tian Han, Yang Lu, Song-Chun Zhu, and Ying Nian Wu. Alternating back-propagation for generator network. In *AAAI*, volume 3, page 13, 2017.

[10] Geoffrey Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[11] Wenqing Hu, Chris J. Li, Lei Li, and Jian-Guo Liu. On the diffusion approximation of nonconvex stochastic gradient descent. *arXiv preprint arXiv:1705.07562*, 2017.

[12] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *ICLR*, 2017.

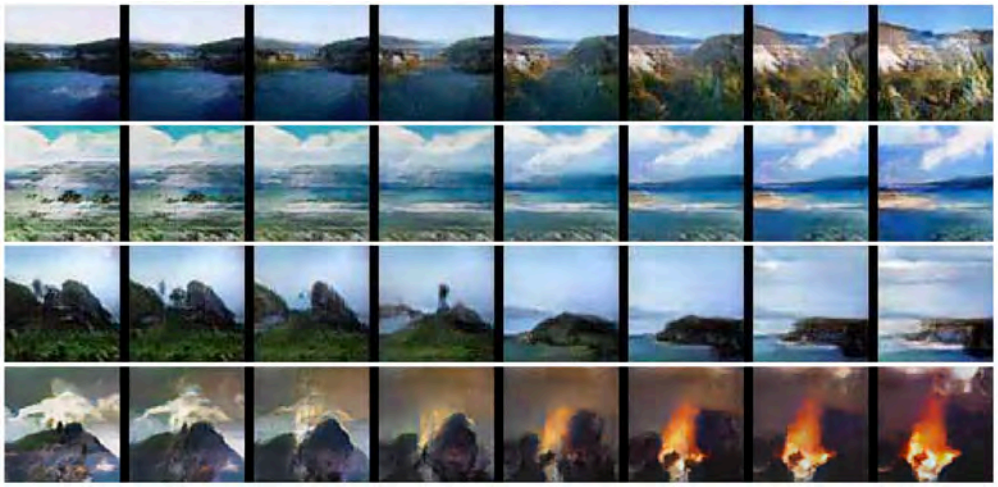


图 10.17: 协作学习实验 2. 潜在空间中的线性插值对应于图像空间中的直观非线性插值。

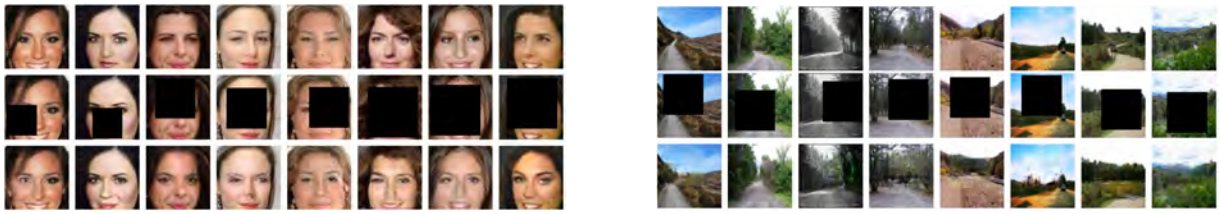


图 10.18: 协作学习实验 3. 在每组中, 顶行显示原始图像，中间一行显示原始图像的遮挡观察，底行显示生成器潜在空间的遮挡图的重建像.

[13] Yann A LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012.

[14] Qianxiao Li, Cheng Tai, and Weinan E. Stochastic modified equations and adaptive stochastic gradient algorithms. *ICML*, pages 2101–2110, 2017.

[15] Yang Lu, Song Chun Zhu, and Ying Nian Wu. Learning frame models using cnn filters. *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[16] Eric Moulines and Francis R Bach. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Advances in Neural Information Processing Systems*, pages 451–459, 2011.

[17] Yurii Nesterov. *Introductory lectures on convex optimization*, volume 87. Springer Science & Business Media, 2004.

[18] Nicolas L Roux, Mark Schmidt, and Francis R Bach. A stochastic gradient method with an exponential convergence _rate for finite training sets. In *Advances in Neural Information Processing Systems*, pages 2663–2671, 2012.

[19] Tijmen Tieleman. Training restricted boltzmann machines using approximations to the likelihood gradient. *ICML*, 2008.

[20] Jianwen Xie, Wenze Hu, Song Chun Zhu, and Ying Nian Wu. A theory of generative convnet. *International Conference on Machine Learning*, 2016.

[21] Jianwen Xie, Yang Lu, and Ying Nian Wu. Cooperative learning of energy-baed model and latent variable model via mcmc teaching. *AAAI*, 2018.

[22] Laurent Younes. Estimation and annealing for gibbsian fields. *Ann. Inst. H. Poincaré Probab. Statist*, 24:269–294, 1988.

[23] Yao Zhang, Andrew M. Saxe, Madhu S. Advani, and Alpha A. Lee. Entropy-energy competition and the effectiveness of stochastic gradient descent in machine learning. *arXiv preprint arXiv:1803.0192*, 2018.

[24] Song Chun Zhu and Xiuwen Liu. Learning in gibbsian fields: How accurate and how fast can it be? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):1001–1006, 2002.

[25] Song Chun Zhu, Ying Nian Wu, and David Mumford. Minimax entropy principle and its application to texture modeling. *Neural Computation*, 9(8):1627–1660, 1997.

[26] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126, 1998.

第 10 章　绘制能源景观

‘‘通过可视化信息，我们将其转化为你可以用眼睛探索的景观：一种信息地图。当你迷失在信息中时，信息地图就会很有用。” - David McCandless

引言

在许多统计学习问题中，要优化的能量函数是高度非凸的。大量的研究致力于通过凸优化来近似目标函数，例如在回归中用 L_0 范数替换 L_1 范数，或者设计算法以找到良好的局部最优，例如 EM 算法集群。在分析这种非凸能量景观的特性方面已经做了很少的工作。在本章中，受到 [2] 和 [32] 可视化 Ising 和 Spin-glass 模型景观成功的启发，我们计算了高维模型空间中的能量景观图（ELM）（即机器学习文献）用于一些经典的统计学习问题 - 聚类和双聚类。

11.1　景观结构和任务

ELM 是树结构，如图 11.1所示，其中每个叶节点代表局部最小值，每个非叶节点代表相邻能量盆之间的屏障。ELM 通过以下信息描述能源格局。

- 当地最小值及其能量水平;
- 相邻局部最小值之间的能量障碍及其能量水平;

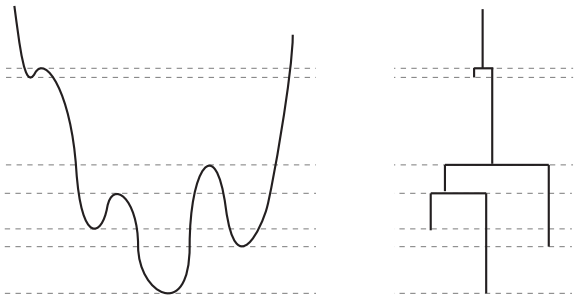


图 11.1: 能量函数和相应的能量景观图（ELM）。ELM 的 y 轴是能级，每个叶节点是局部最小值，叶节点在其能量盆的脊处连接。

- 每个局部最小值的概率质量和体积 (如图 11.3所示)。

此类信息在以下任务中很有用。

1. 分析优化问题的内在困难（或复杂性），用于推理或学习任务。例如，在双聚类中，我们将问题分解为在不同条件下的简单, 困难, 和不可能的条件。

2. 分析各种条件对 ELM 复杂性的影响，例如，聚类中的可分离性，训练样本的数量，监督的级别（即标记示例的百分比），以及正则化的强度（即先前模型）。

3. 通过显示访问各种最小值的频率来分析各种算法的行为。例如，在多高斯聚类问题中，我们发现当高斯分量高度可分时，K 均值聚类比 EM 算法效果更好 [9]，而当组分可分离性较差时则相反。与通过 K-means 和 EM 频繁访问局部最小值相比，Swendsen-Wang 切割方法 [1] 在所有可分离条件下收敛于全局最小值。

4. 分析蛋白质折叠，其中能量景观具有漏斗状外观 [21]，具有相变。在到达漏斗底部之前，折叠很容易在不同的吸引盆之间移动，但是底部可能有一些局部最优，其中只有一个是原生状态（全局最优），而其他的产生稳定的错误折叠蛋白质。大量此类错误折叠的蛋白质与神经退行性疾病有关，如阿尔茨海默病，帕金森病，疯牛病等。

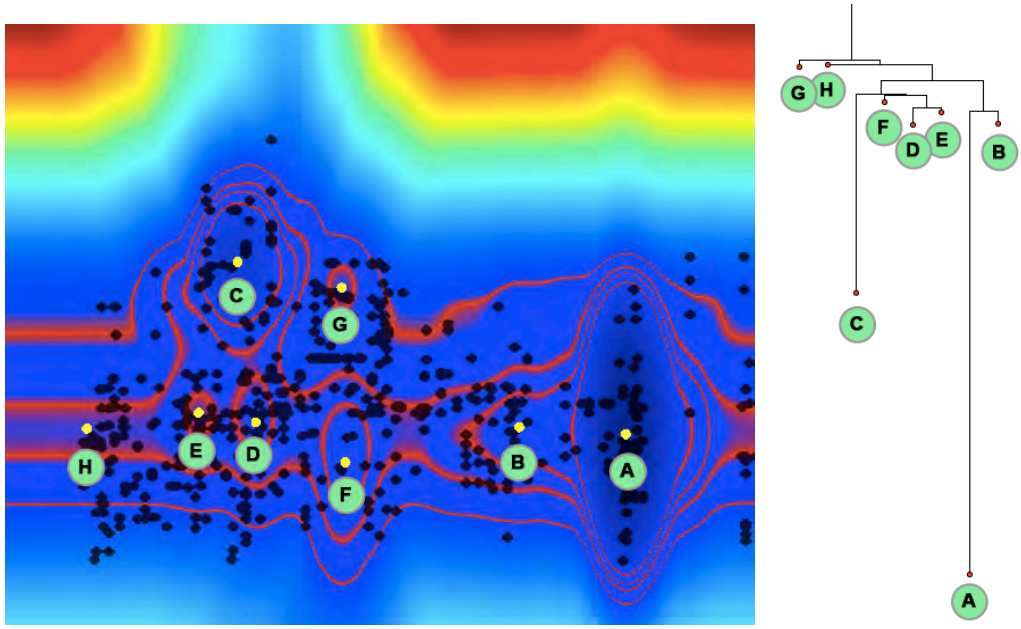
我们从图 11.2 和 11.3中的一个简单说明性示例开始。假设基础概率分布是 1D 空间中的 4 分量高斯混合模型（GMM），并且组件很好地分离。模型空间是 11 维的，参数 $\{(\mu_i, \sigma_i, \alpha_i) : i = 1,2,3,4\}$ 表示每个分量的均值，方差和权重。我们从 GMM 中抽取了 70 个数据点，并在模型空间中构建了 ELM。我们将模型空间限制在由样本定义的有限范围内。

由于我们只能可视化 2D 地图，我们将所有参数设置为等于真值，除了保持 μ_1 和 μ_2 为未知数。图 11.2(a) 显示了 $0 \leq \mu_1, \mu_2 \leq 5$ 范围内的能量图。景观中的不对称性是由于真实模型在第一和第二组分之间具有不同的权重。一些浅的局部最小值，如 E，F，G，H，是由有限数据样本引起的小“凹痕”。

图 11.2 (a) 显示了所有局部最小值。此外，它显示了我们将讨论的算法接受的前 200 个 MCMC 样本。样本聚集在局部最小值周围，覆盖所有能量盆。如期望的那样，它们不存在于远离局部最小值的高能区域中。图 11.2 (b) 显示了由此产生的 ELM 以及叶片与能量景观中局部最小值之间的对应关系。此外，图 11.3 (a) 和 (b) 显示了这些能量盆的概率质量和体积。

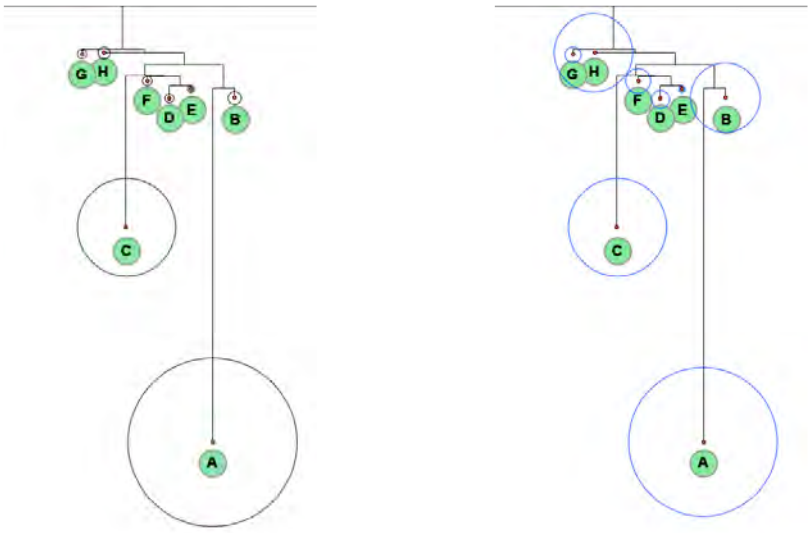
在文献中，[2] 提出了可视化旋转玻璃模型的多维能量景观的第一项工作。从那以后，统计学家开发了一系列 MCMC 方法，用于提高遍历状态空间的采样算法的效率。最值得注意的是，[17] 将 Wang-Landau 算法 [29] 推广到状态空间中的随机游走。[32] 使用广义 Wang-Landau 算法绘制具有数百个局部最小值的 Ising 模型的断开图，并提出了估算能垒的有效方法。此外，[33] 通过聚类蒙特卡罗样本构建贝叶斯推断 DNA 序列分割的能量景观。

与上述计算推理问题的“状态”空间中的景观的工作相反，我们的工作主要集中在“模型”空间（所有模型的集合，也称为机器学习社区中的假设空间）中的景观用于统计学习和模型估计问题。在绘制模型空间景观时存在一些新问题。i）许多盆地有一个平底，例如图 11.2.(a) 中的盆 A。这可能导致大量错误的局部最小值。ii）可能存在约束参数，例如，权重必须总和为 $\sum_i \alpha_i = 1$. 因此我们可能需要在流型上运行我们的算法。



(a) (b)

图 11.2: (a) 4 组分 1-d GMM 的能量景观，除两种方法外，所有参数均已固定。级别集以红色突出显示。局部最小值以黄点显示，前 200 个 MCMC 样本以黑点显示。(b) 由此产生的 ELM 以及叶子与能量景观中的局部最小值之间的对应关系。



(a) probability mass (b) volume

图 11.3: 二维景观的能量盆的概率质量和体积如图 11.2所示。

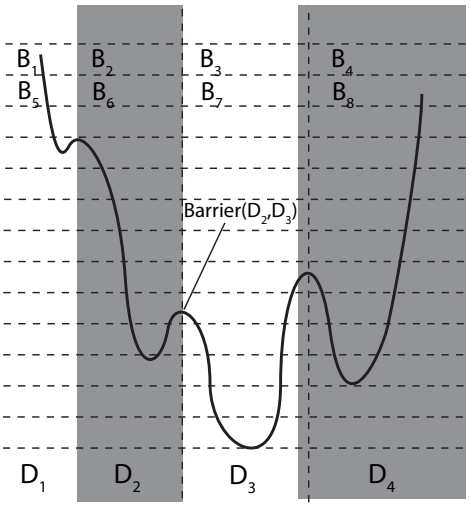


图 11.4: 模型空间 Ω 被划分为能量盆 D_i（沿 x 轴），能量 $\mathbb{R}$（y 轴）被划分为均匀间隔 $[u_{j+1}, u_j)$ 。

11.2 ELM 结构

在本节中，我们将介绍构建 ELM 的基本思想，并估算其性质 --质量，体积和复杂性。

11.2.1 空间分区

设 Ω 是定义概率分布 $\pi(x)$ 和能量 $E(x)$ 的模型空间。在本文中，我们假设 Ω 是使用样本的属性限制的。Ω 被分成 K 个不相交的子空间，代表能量盆

$$\Omega = \cup_{i=1}^{K} D_i, \quad \cap_{i=1}^{K} D_i = \emptyset \ \ \forall i \neq j. \tag{11.1}$$

也就是说，任何点 $x \in D_i$ 将通过梯度下降收敛到相同的最小值。

如图 11.4 所示, 能量也被划分为区间 $[u_{j+1}, u_j), j = 1, 2, ..., L$。因此，我们在产品空间 $\Omega \times \mathbb{R}$ 中获得了一组区间作为量子化的原子元素,

$$B_{ij} = \{x : x \in D_i, \ E(x) \in [u_{j+1}, u_j)\}. \tag{11.2}$$

盆地 K 的数量和间隔数量 L 是未知的，并且必须在计算过程中以自适应和迭代的方式估计。

11.2.2 广义 Wang- - Landau 算法法

广义 Wang-Landau (GWL) 算法的目的是模拟以相等概率访问所有二元组 $\{B_{ij}, \forall i, j\}$ 的马尔科夫链，从而有效地揭示景观的结构。

设 $\phi : \Omega \to \{1, \ldots, K\} \times \{1, ..., L\}$ 是模型空间和 bin 索引之间的映射: $\phi(x) = (i, j)$ 如果 $x \in B_{ij}$. 给定任何 x, 通过梯度下降或其变体，我们可以找到并记录它所属的盆 D_i ，计算其能级 $E(x)$，从而找到指数 $\phi(x)$。

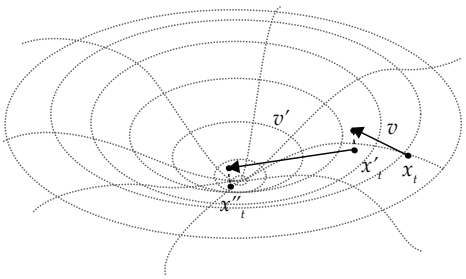


图 11.5: 投影梯度下降的前两步。该算法用 MCMC 样本 x_t 初始化。v 是点 x_t 处 $E(x)$ 的梯度。Armijo 线搜索用于确定沿矢量 v 的步长 α，x'_t 是投影 $T(x_t+\alpha v)$ 到子空间 Γ 上。然后 x''_t 是投影 $T(x_t+\alpha' v')$，依此类推。

我们将 $\beta(i,j)$ 定义为 bin 的概率质量

$$\beta(i,j)=\int_{B_{i,j}}\pi(x)\,dx. \tag{11.3}$$

然后，我们可以定义一个新的概率分布，它在所有的 bin 中具有相同的概率，

$$\pi'(x)=\frac{1}{Z}\pi(x)/\beta(\phi(x)), \tag{11.4}$$

Z 是一个缩放常数。

为了从 $\pi'(x)$ 采样, 可以通过变量 γ_{ij} 估计 $\beta(i,j)$。我们定义概率函数 $\pi_\gamma:\Omega\to\mathbb{R}$ 为

$$\pi_\gamma(x)\propto\frac{\pi(x)}{\gamma_{\phi(x)}}=\sum_{i,j}\frac{\pi(x)}{\gamma_{ij}}\mathbb{1}(x\in B_{ij})\ \text{ st. }\int_\Omega\pi_\gamma(x)dx=1.$$

我们从初始 γ^0 开始, 并使用随机近似 [18] 迭代地更新 $\gamma^t=\{\gamma^t_{ij},\forall i,j\}$ 。假设 x_t 是时间 t 的 MCMC 状态，然后 γ^t 以指数速率更新，

$$\log\gamma^{t+1}_{ij}=\log\gamma^t_{ij}+\eta_t\mathbb{1}(x_t\in B_{ij}),\quad\forall i,j. \tag{11.5}$$

η_t 是时间 t 的步长。步长随着时间的推移而减小，减少的时间表要么像 [18] 中那样预先确定，要么自适应地确定，如 [31] 所示。

给定 γ^t 的每次迭代使用 Metropolis 步骤。设 $Q(x,y)$ 为移动的提议概率 x 到 y，则接受概率为

$$\begin{aligned}\alpha(x,y)&=\min\left(1,\tfrac{Q(y,x)\pi_\gamma(y)}{Q(x,y)\pi_\gamma(x)}\right)\\&=\min\left(1,\tfrac{Q(y,x)}{Q(x,y)}\tfrac{\pi(y)}{\pi(x)}\tfrac{\gamma^t_{\phi(x)}}{\gamma^t_{\phi(y)}}\right).\end{aligned} \tag{11.6}$$

直观地, 如果 $\gamma^t_{\phi(x)}<\gamma^t_{\phi(y)}$, 那么访问 y 的概率就会降低。以探索能源景观为目的，GWL 算法改进了传统方法，例如模拟退火 [13] 和回火 [19] 过程。后者从 $\pi(x)^{\frac{1}{T}}$ 采样，即使在高温下也不会以相同的概率访问 bin。

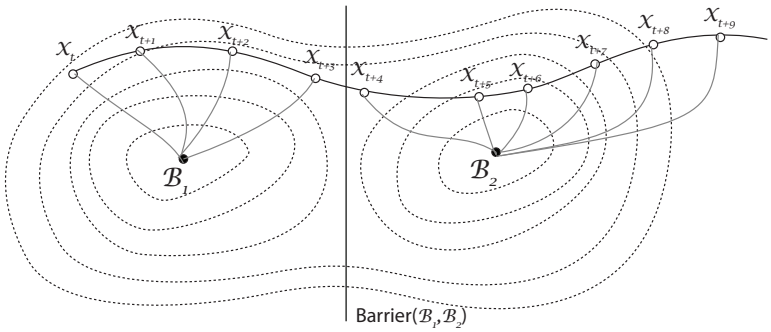


图 11.6: 顺序 MCMC 样本 $x_t, x_{t+1}, \ldots, x_{t+9}$. 对于每个样本，我们执行梯度下降以确定样本属于哪个能量盆。如果两个连续样本落入不同的盆地 (本例中为 x_{t+3} 和 x_{t+4}), 我们估计或更新各自盆地之间能垒的上限 (本例中为 B_1 和 B_2).

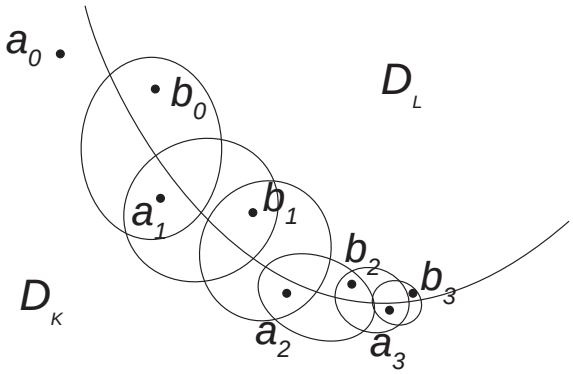


图 11.7: 脊下降算法用于估算在连续 MCMC 样本 $a_0 = x_t, b_0 = x_{t+1}$ 处初始化的盆地 D_k 和 D_l 之间的能垒 $a_0 \in D_k$ 和 $b_0 \in D_l$.

在进行梯度下降时，我们采用 Armijo 线搜索来确定步长; 如果模型空间 Ω 是 $\mathbb{R}^n$ 中的流形，我们执行投影梯度下降，如图 11.5所示。为了避免错误地识别同一盆地内的多个局部最小值（特别是当存在大的平坦区域时），我们基于以下标准合并通过梯度下降识别的局部最小值：（1）两个局部最小值之间的距离小于常数 ε; 或（2）沿两条局部最小值之间的直线没有障碍物。

图 11.6 (a) 说明了两个能量盆地上的马尔可夫链状态 $x_t, ..., x_{t+9}$ 的序列。虚线曲线是能量函数的水平集。

11.2.3 Constructing the ELM

假设我们收集了一系列来自 GWL 算法的样本 $x_1, \ldots, x_N$ 。ELM 结构包括以下两个过程。

1, 找到相邻盆地之间的能量障碍。我们收集跨越两个盆地 D_k 和 D_l 的所有连续 MCMC 状态，

$$X_{kl} = \{(x_t, x_{t+1}) : x_t \in D_k, x_{t+1} \in D_l\} \tag{11.7}$$

我们选择具有最低能量的 $(a_0, b_0) \in X_{kl}$

$$(a_0, b_0) = \text{argmin}_{(a,b)\in\Omega_{kl}} [\min(E(a), E(b))] .$$

接下来，我们迭代以下步骤，如图 11.7所示

$$a_i = \text{argmin}_a \{E(a) : a \in \text{Neighborhood}(b_{i-1}) \cap D_k\}$$
$$b_i = \text{argmin}_b \{E(b) : b \in \text{Neighborhood}(a_i) \cap D_l\}$$

直到 $b_{i-1} = b_i$。邻域由自适应半径定义。然后 b_i 是能垒，$E(b_i)$ 是势垒的能级。在 [32] 中使用了这种脊下降法的离散形式。

2, 构造树形结构. ELM 的树结构由能量盆组和它们之间的能垒构成，通过从分层凝聚聚类算法修改的迭代算法。最初，能量盆由未连接的叶节点表示，其 y 坐标由盆的局部最小值确定。在每次迭代中，表示具有最低屏障的能量盆 D_1, D_2 的两个节点通过新的父节点连接，其父坐标是屏障的能级; 然后将 D_1 和 D_2 视为合并，合并盆地与任何其他盆地 D_i 之间的能垒仅仅是 D_1/D_2 之间的能垒和 D_i 中较低的一个。当所有能量盆合并时，我们获得完整的树结构。为清楚起见，我们可以从深度小于常数 ε 的树盆中移除。

11.2.4 估计 ELM 中节点的质量和体积

在 ELM 中，我们可以估计每个能量盆地的概率质量和体积。当算法收敛时，γ_{ij} 的归一化值接近 bin B_{ij} 的概率质量：

$$\hat{P}(B_{ij}) = \frac{\gamma_{ij}}{\sum_{kl}\gamma_{kl}} \to \beta(i,j), \quad .$$

因此盆地 D_i 的概率质量可以通过估算

$$\hat{P}(D_i) = \sum_j \hat{P}(B_{ij}) = \frac{\sum_j \gamma_{ij}}{\sum_{kl}\gamma_{kl}} \tag{11.8}$$

假设能量 $E(x)$ 被划分为足够小的大小为 du 的区间。基于概率质量，我们可以估计模型空间 Ω 中的区间和盆地的大小[1] 。具有能量区间 $[u_j, u_j + du)$ 的 bin B_{ij} 可被视为具有能量 u_j 和概率密度 αe^{-u_j}（α 是归一化因子）。bin B_{ij} 的大小可以通过估算

$$\hat{A}(B_{ij}) = \frac{\hat{P}(B_{ij})}{\alpha e^{-u_j}} = \frac{\gamma_{ij}}{\alpha e^{-u_j}\sum_{kl}\gamma_{kl}}$$

盆地 D_i 的大小可以通过估算

$$\hat{A}(D_i) = \sum_j \hat{A}(B_{ij}) = \frac{1}{\sum_{kl}\gamma_{kl}} \sum_j \frac{\gamma_{ij}}{\alpha e^{-u_j}} \tag{11.9}$$

[1]Note that the size of a bin/basin in the model space is called its volume by [33], but here we will use the term "volume" to denote the capacity of a basin in the energy landscape.

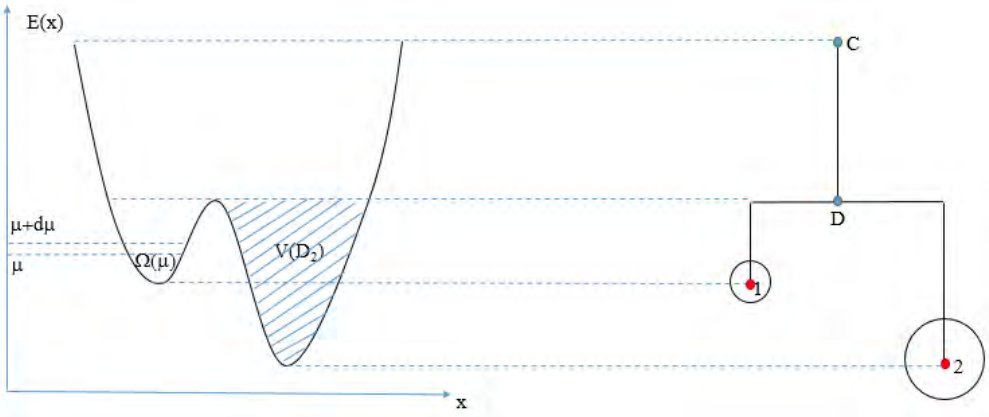
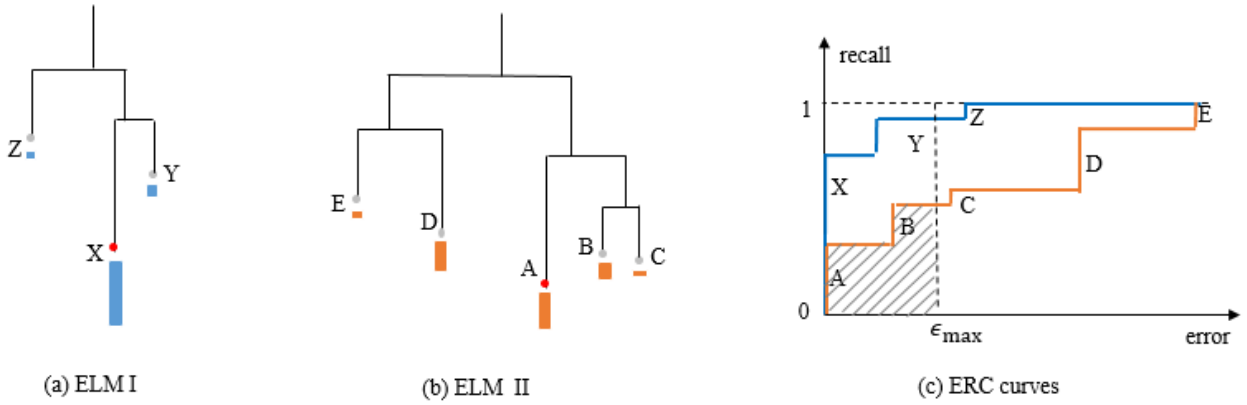


图 11.8: 盆地的体积。假设 du 足够小，能量盆的体积可以通过每个能量区间的估计体积的总和来近似。



(a) ELM I　　(b) ELM II　　(c) ERC curves

图 11.9: 描述 ELM 学习的难度。对于使用 ELM I 和 ELM II 的两个学习任务，彩色条显示学习算法收敛到盆地的频率，从中绘制两条误差回忆曲线。关于该算法，学习任务的难度可以通过曲线下面积在可接受的最大误差内来测量。

此外，我们可以估算能量景观中盆地的体积，其定义为 $\Omega \times \mathbb{R}$ 空间中盆地中所含的空间量。

$$\hat{V}(D_i) = \sum_j \sum_{k:u_k \leq u_j} \hat{A}(B_{ik}) \times du = \frac{du}{\sum_{lm} \gamma_{lm}} \sum_j \sum_{k:u_k \leq u_j} \frac{\gamma_{ik}}{\alpha e^{-u_k}} \tag{11.10}$$

其中 j 的范围取决于盆地的定义。在限制性定义中，盆地仅包括最近屏障下的体积，如图 11.8所示。盆地 1 和 2 上方的体积由两个盆地共享，并且位于两个能量屏障 C 和 D 之间。因此，我们将 ELM 中非叶节点的体积定义为其子体积加上体积的总和。障碍之间。例如，节点 C 具有体积 $V(A)+V(B)+V(AB)$。

如果我们的目标是通过反复平滑景观来开发 ELM 的尺度空间表示，那么盆地 A 和 B 将以一定的比例合并到一个盆地中，并且两个盆地上方的体积也将被添加到这个新的合并盆地中。

注意，将空间划分为箱而不是盆，有助于计算能垒，盆的质量和体积。

11.2.5 表征学习任务的难度（或复杂性）

通常希望通过单个数字来测量学习任务的难度。例如，我们比较图 11.9中的两个 ELM。在 ELM 的环境中学习我看起来比 ELM II 更容易。但是，困难还取决于学习算法。因此，我们可以多次运行学习算法，并记录它收敛到每个盆地或最小的频率。频率由叶节点下的彩色条的长度表示。

假设 Θ^* 是要学习的真实模型。在图 11.9中, Θ^* 对应于 ELM I 和 ELM II X 的节点 A 。通常, Θ^* 可能不是全局最小值或甚至不是最小值。然后我们测量 Θ^* 和任何其他局部最小值之间的距离（或误差）。随着误差的增加，我们累积频率绘制曲线。我们称之为误差 - 回忆曲线（ERC），因为水平轴是误差，垂直轴是调用解的频率。这就像贝叶斯决策理论，模式识别和机器学习中的 ROC（受体 - 算子特征）曲线一样。通过滑动最大容许误差的阈值 $\varepsilon_{\max}$，曲线表征 ELM 相对于算法的难度。

表征难度的单个数字可以是给定 $\varepsilon_{\max}$ 的曲线下面积（AUC）。这是由 11.9的阴影区域说明的。（c）ELM II。当 AUC 接近 1 时，任务很容易，当 AUC 接近 0 时，学习是不可能的。

在学习问题中，我们可以设置与一系列 ELM 相对应的不同条件。这些 ELM 的难度测量可以在参数空间中可视化为难度图。我们将在实验 III 中展示这样的地图。

11.2.6 MCMC 在模型空间中移动

为了设计马尔可夫链在模型空间 $\mathbb{R}$ 中的移动，我们在方程 (11.6) 中的 metropolis-Hastings 设计中使用两种类型的方法。

1, 当前模型 x 附近的随机提议概率 $Q(x,y)$。

2, 数据增强。很大一部分非凸优化问题涉及潜在变量。例如，在聚类问题中，每个数据点的类标签都是潜在的。对于这些问题，我们使用数据增加 [26] 来提高采样效率。为了提出一个新的模型 $y = x_{t+1}$，我们首先根据 $p(Z_t|x_t)$ 对潜在变量 Z_t 的值进行采样，然后基于样本对新模型 x_{t+1} 进行采样 $p(x_{t+1}|Z_t)$。然后，基于相同的接受概率，在公式 11.6，提议 $y = x_{t+1}$ 被接受或拒绝。

但请注意，我们在 ELM 构造中的目标是遍历模型空间而不是从概率分布中进行采样。当收集到足够的样本并因此重量 γ_{ij} 变大时，重新加权的概率分布将与原始分布 $\pi(x)$ 显著不同，并且通过数据增加提出的模型的拒绝率将变高。因此，我们在开始时更频繁地使用基于数据增加的提议概率，并且当权重变大时越来越依赖于随机提议。

11.2.7 ELM 收敛分析

GWL 算法与静止分布的收敛是 ELM 收敛的必要但不充分的条件。如图 11.10所示，由于两个因素，构造的 ELM 可能有微小的变化：（i）当我们在障碍下绘制分支时的左右模糊; （ii）能量障碍的精确度将影响树木的内部结构。

在实验中，首先我们监控模型空间中 GWL 的收敛。我们使用随机起始值运行多个 MCMC 初始化。在老化期后，我们使用多维缩放收集样本并在 2-3 维空间中投影。我们使用 Gelman 和 Rubin 准则 [12][5] 的多变量扩展来检查链是否已收敛到静止分布。

一旦认为 GWL 已经收敛，我们可以通过检查随后时间 t 的以下两组的收敛来监视 ELM 的收敛。

1. 树 S_L^t 的叶子标记组，其中每个点 x 是具有能量 $E(x)$ 的局部最小值。随着 t 的增加, S_L^t 单调增长，直到找不到局部最小值，如图 11.11.(a) 所示。

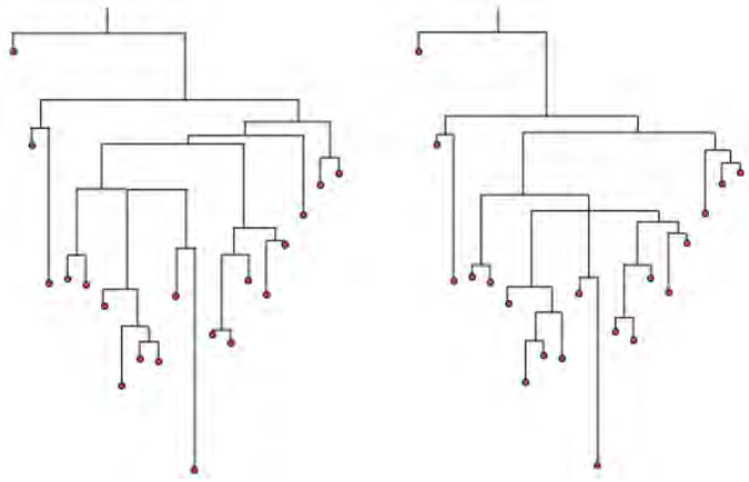
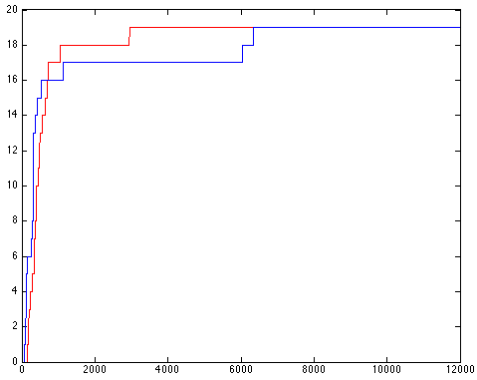
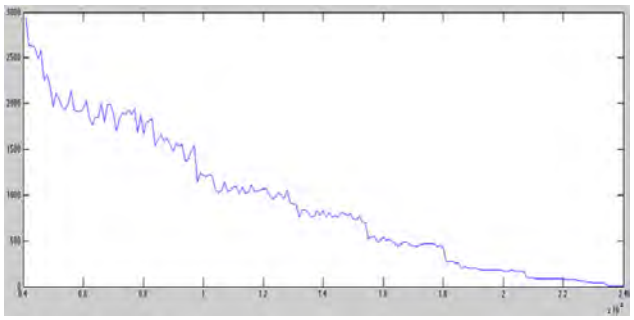


图 11.10: 两个 MCMC 链 C_1 和 C_2 产生的两个 ELM 在 24,000 次迭代收敛后在不同的起始点初始化。



(a)



(b)

图 11.11: 监测从不同起始点初始化的两个 MCMC 链 C_1 和 C_2 产生的 ELM 的收敛性。(a) 找到的局部最小值的数量与 C_1 和 C_2 的迭代次数之比。(b) 两个 ELM 之间的距离与迭代次数。

2. 树 S_N^t 的内部节点集合，其中每个点 y 是级别 $E(y)$ 的能量势垒。随着 t 的增加，随着 t 的增加，因为马尔可夫链穿过盆地之间的不同脊，我们可能会发现较低的障碍。从而 $E(y)$ 单调减小，直到在某一时间段内 S_N^t 中没有障碍被更新。

我们进一步计算由两个具有不同初始化的 MCMC 构造的两个 ELM 之间的距离度量。为此，我们计算两棵树之间匹配的最佳节点，然后距离是根据匹配的叶节点和障碍的差异以及对不匹配节点的惩罚来定义的。我们省略了这个定义的细节，因为它对这项工作并不重要。图 11.11.(b) 显示随着生成更多样本，距离会减小。

11.3 实验 I：高斯混合模型的 ELM

在本节中，我们计算了用于学习两种类型的高斯混合模型的 ELM：（i）研究不同条件的影响，例如可分离性和监督水平; ii）比较流行算法的行为和表现，包括 K 均值聚类，EM（期望最大化），两步 EM

和 Swendsen-Wang 切割。我们将在实验中使用合成数据和实际数据。

11.3.1 能量和梯度计算

具有 d 维中的 n 个分量的高斯混合模型 Θ 具有权重 α_i，μ_i 和协方差矩阵 Σ_i 其中 $i = 1,\ldots,n$。给定一组观测数据点 $\{z_i, i = 1,...,m\}$，我们将能量函数写为

$$\begin{aligned} E(\Theta) &= -\log P(z_i : i = 1 \ldots m|\Theta) - \log P(\Theta) && (11.11)\\ &= -\sum_{i=1}^{m} \log f(z_i|\Theta) - \log P(\Theta). && (11.12) \end{aligned}$$

$P(\Theta)$ 是 Dirichlet 先验和 NIW 先验的乘机。它的偏导数很容易计算。$f(z_i|\Theta) = \sum_{j=1}^{n} \alpha_j G(z_i;\mu_j,\Sigma_j)$ 是数据 z_i 的可能性, 其中 $G(z_i;\mu_j,\Sigma_j) = \frac{1}{\sqrt{\det(2\pi\Sigma_j)}} \exp\left[-\frac{1}{2}(z_i-\mu_j)^T \Sigma_j^{-1}(z_i-\mu_j)\right]$ 是高斯模型。在标记数据点的情况下（即，从其采样的组件是已知的），可能性仅为 $G(z_i;\mu_j,\Sigma_j)$.

对于样本 z_i, 我们有以下对数似然的偏导数来计算能量景观中的梯度。

a) 关于每个权重 α_j 的偏导数:

$$\frac{\delta \log f(z_i)}{\delta \alpha_j} = \frac{G(z_i;\mu_j,\Sigma_j)}{\sum_{k=1}^{K} \alpha_k G(z_i,\mu_k,\Sigma_k)}.$$

b) 关于每个平均 μ_j 的偏导数:

$$\frac{\delta \log f(z_i)}{\delta \mu_j} = \frac{\alpha_j G(z_i;\mu_j,\Sigma_j)}{\sum_{k=1}^{K} \alpha_k G(z_i;\mu_k,\Sigma_k)} \Sigma_j^{-1}(\mu_j - z_i).$$

c) 关于每个协方差 Σ_j 的偏导数:

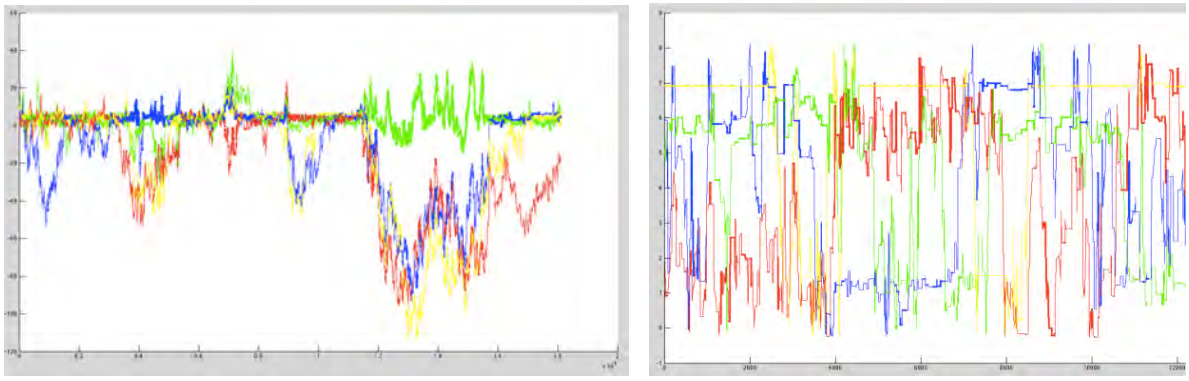
$$\begin{aligned} \frac{\delta \log f_{\text{mm}}(z_i)}{\delta \Sigma_j} &= \frac{\alpha_j G(z_i;\mu_j,\Sigma_j)}{\sum_{k=1}^{K} \alpha_k G(z_i;\mu_k,\Sigma_k)} \frac{1}{2}\left[\frac{\delta}{\delta \Sigma_j} \log \alpha_j G(z_i;\mu_j,\Sigma_j)\right] \\ &= \frac{\alpha_j G(z_i;\mu_j,\Sigma_j)}{\sum_{k=1}^{K} \alpha_k G(z_i;\mu_k,\Sigma_k)} \frac{1}{2}\left[-\Sigma_j^{-T} + \Sigma_j^{-T}(z_i-\mu_j)(z_i-\mu_j)^T \Sigma_j^{-T}\right] \end{aligned}$$

在计算过程中，我们需要限制 Σ_j 矩阵，使每个逆 Σ_j^{-1} 存在，以便具有定义的梯度。每个 Σ_j 是半正定的，因此每个特征值大于或等于零。因此，对于某些 $\varepsilon > 0$，我们只需要对 Σ_j 的每个特征值 λ_i，$\lambda_i > \varepsilon$ 的次要限制。但是，在一个梯度下降步骤之后，新的 GMM 参数可能会在有效 GMM 空间之外，即步骤 $t+1$ 处的新 Σ_j^{t+1} 矩阵将不是对称正定。因此，我们需要将每个 Σ_j^{t+1} 投影到具有投影的对称正定空间中

$$P_{\text{symm}}(P_{\text{pos}}(\Sigma_j^{t+1})).$$

函数 $P_{\text{symm}}(\Sigma)$ 将矩阵投影到对称矩阵的空间中

$$P_{\text{symm}}(\Sigma) = \frac{1}{2}(\Sigma + (\Sigma)^T).$$



(a) unbounded GMM space (b) bounded GMM space

图 11.12: 我们从一维 4 分量 GMM 中采样了 70 个数据点，并在（a）无界（b）有界 GMM 空间中运行 MCM 随机游走以获得 ELM 构造算法。这些图显示了 4 种成分中心位置随时间的演变。线的宽度表示相应组件的重量。

假设 Σ 是对称的, 函数 $P_{\text{pos}}(\Sigma)$ 将 Σ 投射到对称矩阵的空间中，其特征值大于 ε. 因为 Σ 是对称的，所以它可以被分解成 $\Sigma = Q\Lambda Q^T$，其中 Λ 是对角特征值矩阵 $\Lambda = diag\{\lambda_1, \ldots, \lambda_n\}$, Q 是标准正交特征向量矩阵。然后功能

$$P_{\text{pos}}(\Sigma) = Q \begin{pmatrix} \max(\lambda_1, \varepsilon) & 0 & \ldots & 0 \\ 0 & \max(\lambda_2, \varepsilon) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \max(\lambda_n, \varepsilon) \end{pmatrix} Q^T$$

确保 $P_{\text{pos}}(\Sigma)$ 是对称正定。

11.3.2 限制 GMM 空间

从 m 个数据点 $\{z_i, i = 1, \ldots, m\}$, 我们可以估计可能参数 Θ 的空间边界，如果 m 足够大。

设 μ_o 和 Σ_o 为所有 m 个点的样本均值和样本协方差矩阵。我们设置了高斯分量的均值 μ_j 的范围，

$$||\mu_j - \mu_o||_2 < \max_i ||z_i - \mu_o||_2 + \varepsilon_m.$$

ε_m 是我们将在实验中选择的常数。为了约束协方差矩阵 Σ_j, 让 $\Sigma_o = Q\Lambda Q^T$ 为 Σ_o 的特征值分解，其中 $\Lambda = diag\{\lambda_1, \cdots, \lambda_n\}$。我们用 $L = \max(\lambda_1, \ldots, \lambda_n) + \varepsilon_m$ 表示特征值的上界, 并将 Σ_j 的所以特征值绑定为 L。

图 11.12 (a,b) 比较了无界和有界空间中的 MCMC。我们从一维四分量 GMM 中采样 $m = 70$ 个数据点，并运行 MCMC 随机游走以进行 ELM 构建算法。该图显示了 $\mu_1, ..., \mu_4$ 随时间位置的演变。请注意，在图 11.12 (a) 中, MCMC 链可以远离中心移动，并且大部分时间都在有界子空间之外。在图 11.12 (b) 中, 通过强制链保持在边界内，我们能够更有效地探索相关子空间。

11.3.3 合成数据的实验

我们从 2 维空间上具有 $n = 3$ 分量 GMM 的合成数据开始，绘制 m 个样本并运行我们的算法以在不同设置下绘制 ELM。

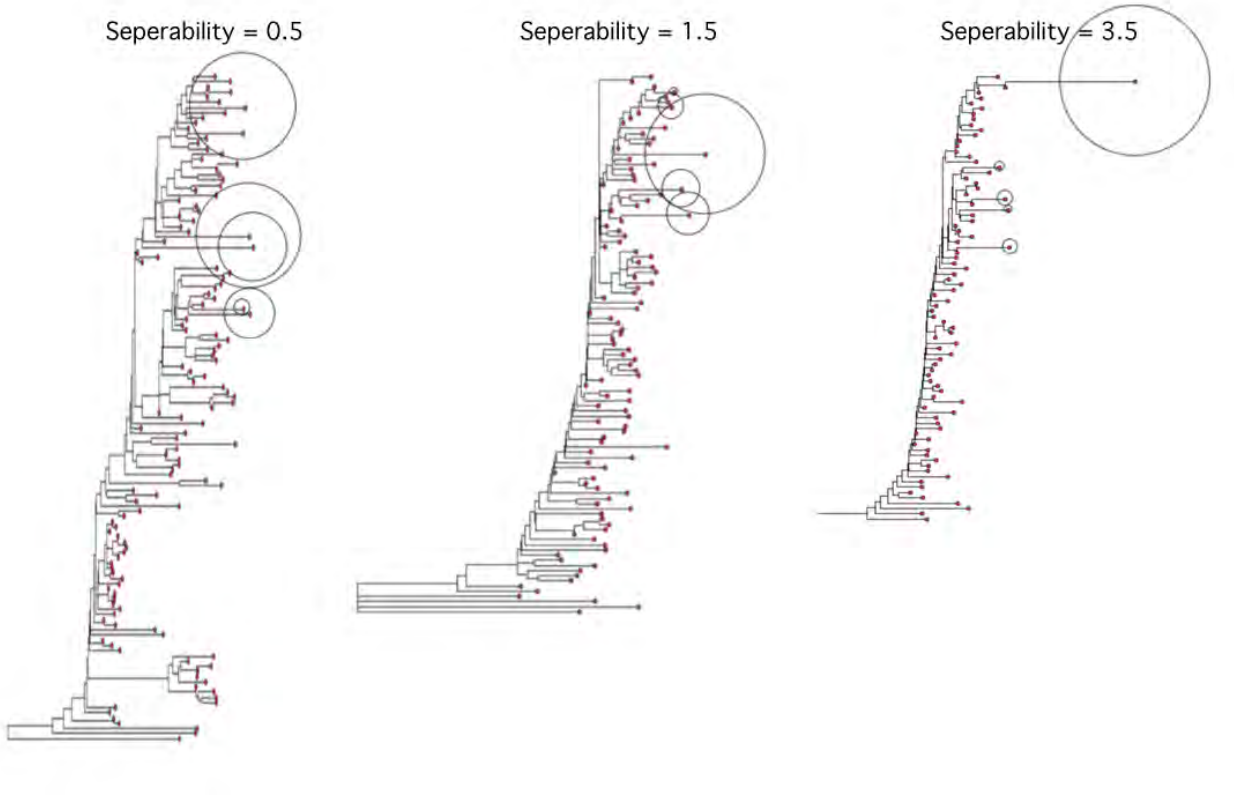


图 11.13: 从具有低，中和高可分离性的 GMM 中抽取的 $m = 100$ 个样品的 ELM 分别为 $c = 0.5, 1.5, 3.5$。圆圈代表盆地的概率质量。

1) 可分离性的影响.GMM 的可分离性表示组件之间的重叠模型定义为 $c = \min\left(\frac{||\mu_i - \mu_j||}{\sqrt{n}\max(\sigma_1, \sigma_2)}\right)$. 这通常用于文献中以衡量学习真实 GMM 模型的难度。

图 11.13 显示了三个代表性的 ELM，其中 $m = 100$ 个数据点的可分离性分别为 $c = 0.5, 1.5, 3.5$。这清楚地表明，在 $c = 0.5$ 时，很难识别模型，许多局部最小值达到相似的能量水平。随着可分离性的增加，能源格局变得越来越简单。当 $c = 3.5$ 时，突出的全球最小值占主导地位。

2) 部分监督的影响. 我们将地面实况标签分配给 m 个数据点的一部分。对于 z_i，其标签 ℓ_i 表示它属于哪个组件。我们设置 $m = 100$，可分性 $c = 1.0$。图 11.14 显示了具有 0%, 5%, 10%, 50%, 90% 数据点标签的 ELM。虽然无监督学习 (0%) 非常具有挑战性，但当标记 5% 或 10% 的数据时，它变得更加简单。标记 90% 的数据时，ELM 只有一个最小值。图 11.15 显示了标记 $1, \ldots, 100$ 样本时 ELM 中的局部最小值。这表明前 10$ 个标签的景观复杂性显着下降，并且在最初的 10% 之后监督输入的收益递减。

3) 学习算法的行为. 我们比较了不同可分性条件下的以下算法的行为。

- 期望最大化（EM）是在统计学中学习 GMM 的最流行的算法。

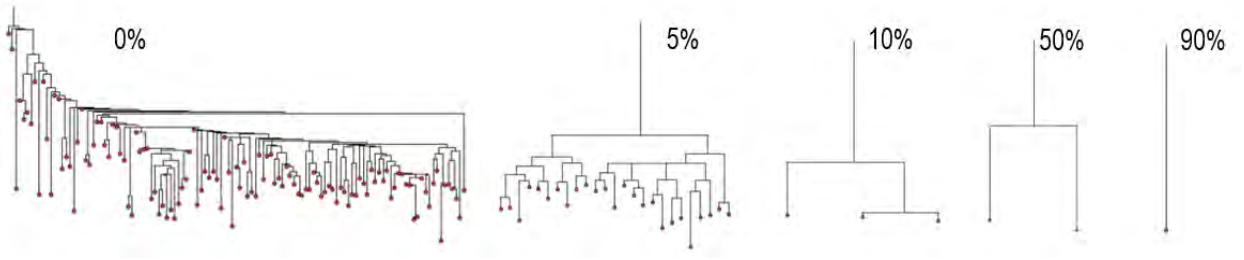


图 11.14: 具有合成 GMM (可分离性 $c = 1.0$, nSamples = 100) 的 ELM $\{0\%, 5\%, 10\%, 50\%, 90\%\}$ 标记的数据点。

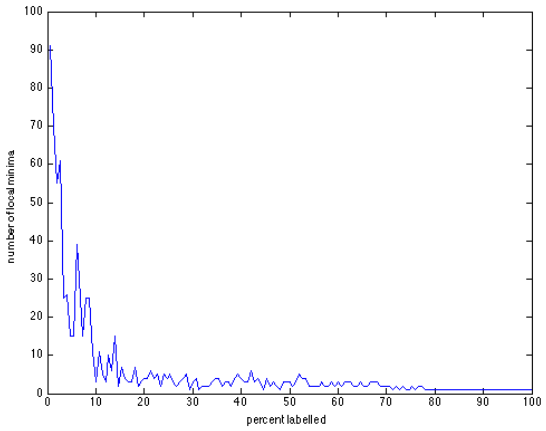


图 11.15: 具有可分离性 $c = 1.0$ 的 GMM 的局部最小值与标记数据点的百分比。

- K 均值聚类是机器学习和模式识别中的一种流行算法。
- 两步 EM 是 [8] 中提出的 EM 的变体，已证明在某些可分离条件下具有性能保证。它从过多的组件开始，然后修剪它们。
- [1] 中提出的 Swedsen-Wang Cut（SW-cut）算法。这将 SW 方法 [25] 从 Ising / Potts 模型推广到任意概率。

我们在实验中修改了 EM，两步 EM 和 SW 切割，以便最小化公式 11.11中定义的能量函数。K-means 不优化我们的能量函数，但它经常被用作学习 GMM 的近似算法，因此我们将其包括在我们的比较中。

对于实验中的每个合成数据集，我们首先构造 ELM，然后运行每个算法 200 次并记录算法所针对的能量盆中的哪一个。因此，我们通过每种算法获得盆地的访问频率，其在图 11.16 和 11.17中的叶节点处显示为不同长度的条。

图 11.16显示了 $n = 10$ 个样本的 K-means，EM 和两步 EM 算法之间的比较从低 ($c = 0.5$) 可分性 GMM 中提取。无论如何，结果分散在不同的局部最小值算法。这说明了从具有大型能量障碍隔开的许多局部最小值的景观中学习模型的困难。

图 11.17 显示了从低 ($c = 0.5$) 和高 ($c = 3.5$) 可分离性 GMM 中抽取的 $m = 100$ 个样本的 EM，k 均值和 SW 切割算法的比较。SW-cut 算法在每种情况下表现最佳，始终收敛于全局最优解。在低可分性

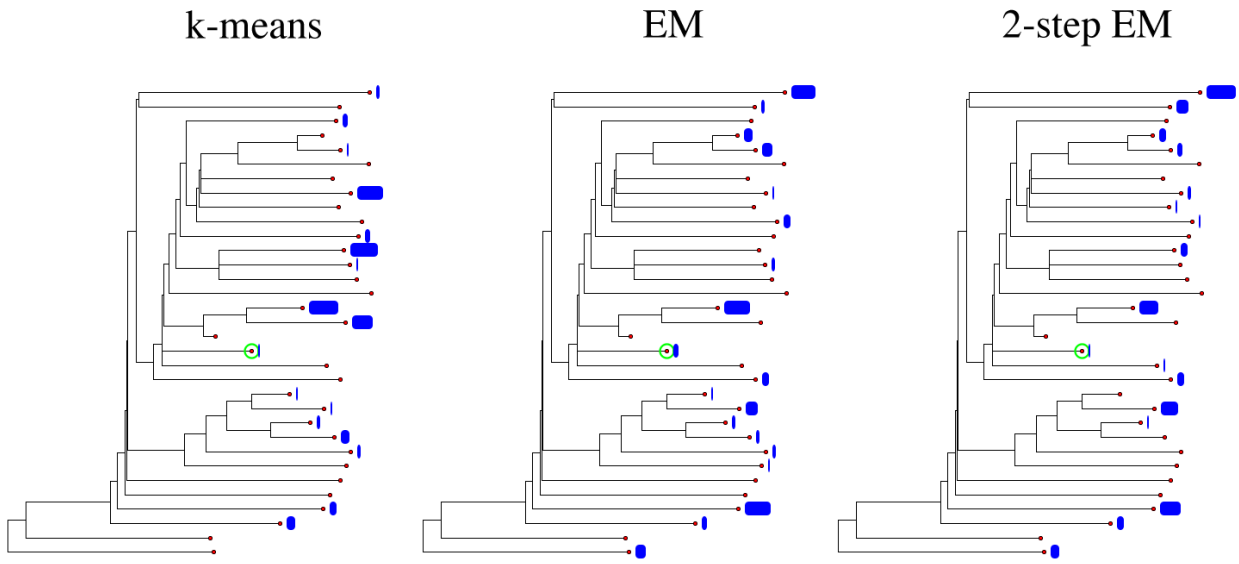


图 11.16: ELM 上 k-means，EM 和 2 步 EM 算法的性能，其中 10 个样本来自 GMM，具有低可分性 ($c = 0.5$)

的情况下，k 均值算法是非常随机的，而 EM 算法几乎总是找到全局最小值，因此优于 k 均值。然而，在高可分性的情况下，k-means 算法在大多数时间收敛到真实模型，而 EM 几乎总是收敛到具有比真实模型更高能量的局部最小值。这一结果证实了最近的理论结果，表明硬 EM 的目标函数（以 k 均值为特例）包含有利于高可分性模型的归纳偏差 [23, 28]。具体来说，我们可以证明 hard-EM 的实际能量函数是：

$$E(\Theta) = -\log P(\Theta|Z) + \min_q \left(\mathbf{KL}(q(L)||P(L|Z,\Theta)) + H_q(L)\right)$$

其中 Θ 是模型参数, $Z = z_1, \ldots, z_m$ 是可观察数据点的集合, L 是潜在变量的集合 (GMM 中的数据点标签), q 是 L 的辅助分布, 而 H_q 是用 $q(L)$ 测量的 L 的熵。上式中的第一项是用 GMM 聚类的标准能量函数。第二项称为后正则化项 [11]，它促使分布 $P(L|Z,\Theta)$ 具有低熵。在 GMM 的情况下，很容易看出 $P(L|Z,\Theta)$ 中的低熵意味着高斯分量之间的高可分离性。

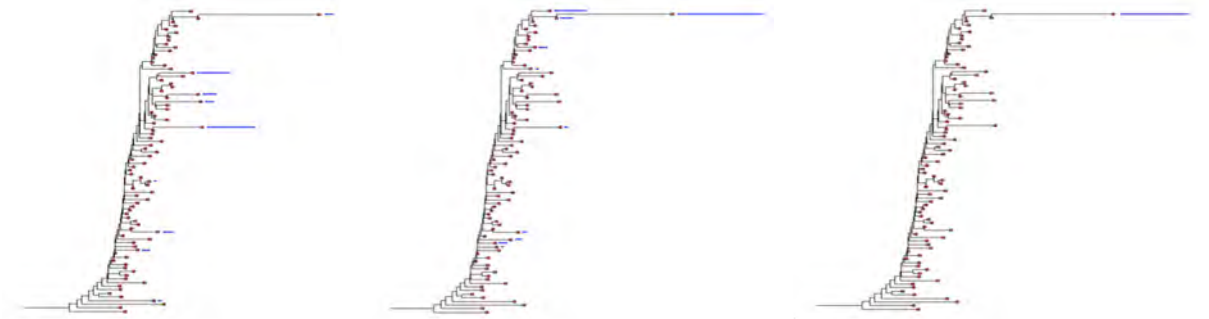
11.3.4 对实际数据的实验

我们运行算法来绘制来自 UCI 存储库的着名 Iris 数据集的 ELM [4]。Iris 数据集包含 4 个维度的 150 个点，可以通过 3 个组件 GMM 进行建模。这三个组件各自代表一种虹膜植物，并且真正的组分标签是已知的。对应于第一组分的点可与其它组分线性分离，但对应于其余两组分的点不可线性分离。

图 11.18显示了 Iris 数据集的 ELM。我们通过绘制以 4 个维度中的 2 个中的每个分量的均值为中心的协方差矩阵的椭球来可视化局部最小值。

6 个最低能量局部最小值显示在右侧，6 个最高能量局部最小值显示在左侧。高能量局部最小值是低能量局部最小值的精确模型。局部最小值（E）（B）和（D）将第一个分量分成两个，剩余的两个（不可分离）分量合并为一个。局部最小值（A）和（F）在第二和第三组分之间具有显着的重叠，并且（C）

(a) EM (b) k-means (c) SW-cut



(d) EM (e) k-means (f) SW-cut

图 11.17: ELM 上 EM，k-means 和 SW-cut 算法的性能。(a-c) 低可分性 $c = 0.5$. (d-f) 高可分性 $c = 3.5$.

具有完全重叠的组分。低能量局部最小值（GL）都具有相同的第一组分和略微不同的第二和第三组分的位置。

我们使用分配了基本实况标签的 $0, 5, 10, 50, 90, 100$ 的点运行算法。图 11.19 显示了这些情况下能源格局的全局最小值。

*

11.4 课程学习

11.4.1 学习依赖语法

依赖语法通过句子单词之间的一组依赖关系来模拟句子的句法结构（图 11.20）。依赖语法已被广泛用于自然语言句法分析，特别是对于具有自由词序的语言 [7, 16, 20]。依赖语法包含一个特殊的根节点和一组代表语言单词的 n 个其他节点。该语法包含以下参数：1. 从根节点到字节点的转移概率的向量;2. 字节点之间的转移概率矩阵; 3. 每个节点在左右方向上继续或停止生成子节点的概率。因此，具有 n 节点的依赖语法的空间具有 $n^2 + n + 2 * 2 * n$ 维度。由于每个概率向量被约束为总和为 1，因此有效依赖语法形成维度 $n^2 + 2n - 1$ 的子空间。要使用依赖语法生成句子，从根节点开始，递归地从每个节点生成子节点; 每个节点处的子节点生成过程由连续/停止概率（是否生成新的子节点）以及转移概率（要生

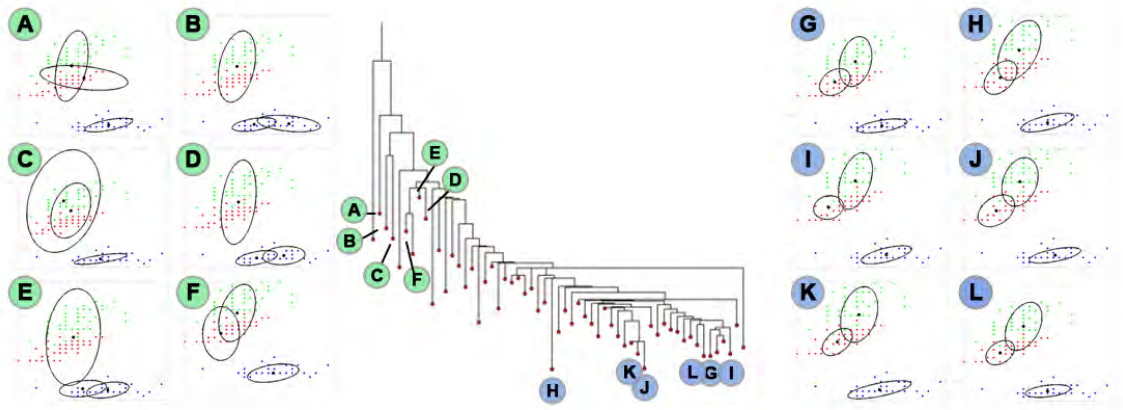


图 11.18: ELM 和 Iris 数据集的一些局部最小值。

成的子节点）控制。生成过程可以用解析树表示，如图 11.20所示。解析树的概率是生成期间所有选择的概率的乘积。句子的概率是句子的所有可能的解析树的概率的总和。

人们越来越关注以监督的方式（例如 [6, 7]）或无监督的方式（例如，[14, 15]）从数据中学习依赖语法。学习问题通常是非凸的，特别是在无人监督的情况下训练句子的依赖性解析是潜在的情况。大多数学习算法试图确定局部最优，并且关于这种局部最优质量的理论分析很少。

大多数现有的学习依赖语法的自动方法都是从训练语料库的所有句子开始，并尝试学习整个语法。另一方面，人类以一种非常不同的方式学习母语的语法：他们接触到非常简单的句子作为婴儿，然后随着他们的成长逐渐变得越来越复杂。这种学习策略被称为课程学习 [3]。早期对语法课程学习的研究产生了积极的 [10] 和消极的结果 [22]。最近，[24] 经验证明，课程有助于无监督的依赖语法学习。

为了解释课程的好处，[27] 建议理想课程逐渐强调数据样本，帮助学习者连续发现目标语法的新语法规则，这有利于学习。[3] 给出了可能与前一个相容的另一种解释，他假设一个好的课程对应于从平滑的目标函数开始学习，并逐渐降低课程阶段的平滑程度，从而引导学习者更好能量函数的局部最小值。

11.4.2 能量函数

用于无监督学习依赖语法的能量函数是 $E(\theta) = -\log P(\theta|D)$ 其中 θ 是语法的参数向量而 D 是训练句子的集合。$\log P(\theta|D)$ 是语法的对数后验概率，定义如下：

$$logP(\theta|D) = \sum_{x \in D} \log P(x|\theta) + \log P(\theta)$$

其中 $P(x|\theta)$ 是前一节中定义的句子 x 的概率，$P(\theta)$ 是 Dirichlet 先验。

11.4.3 假设空间的离散化

从我们的实验中，我们发现即使节点数 n 很小，WL 算法也无法有效地遍历依赖语法的连续假设空间，因为

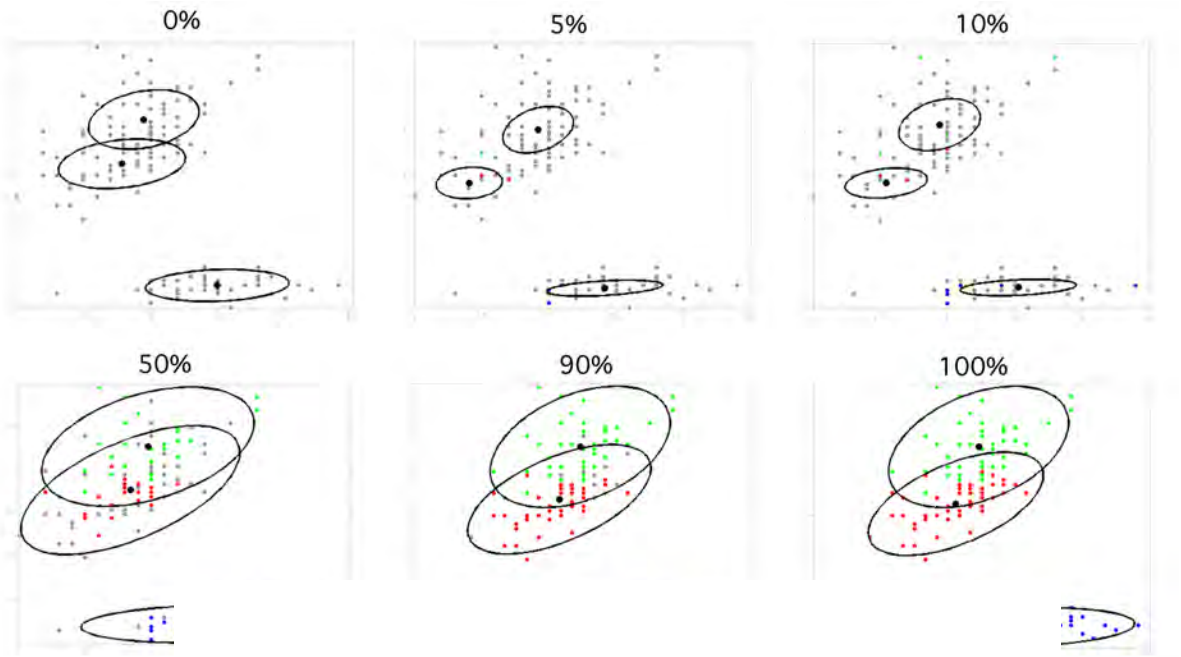


图 11.19: 从 Iris 数 值标记。未标记的点以灰色绘

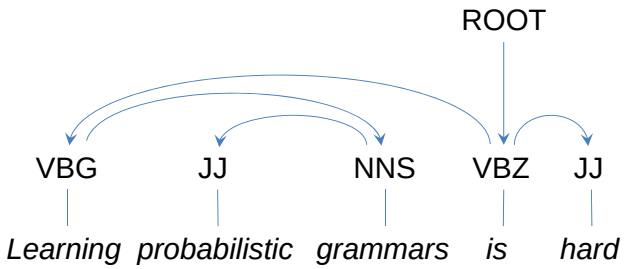


图 11.20: 依赖语法生成的语法结构。

- 空间中局部最小值的数量太大（局部最小值的数量在 100,000 次迭代后仍然线性增长）；
- 梯度计算很慢，特别是对于长句子，梯度通常每次迭代计算超过 100 次;
- 拒绝率超过 90%，因此不到 10%的 MCMC 移动被接受。

为了解决或缓解这些问题（特别是前两个），我们建议对参数空间进行离散化。离散化减少了局部最小值的数量，并且以最陡的下降取代了梯度下降，这在计算上更有效。离散化的 ELM 是原始 ELM 的近似值，仍然传达有关景观的有用信息。

我们通过以下方式对参数空间进行离散化：让 Ω_r 为离散化参数空间，离散化分辨率为 $r > 4$：

$$\Omega_r = \{\vec{\theta} = [\theta_1, \ldots, \theta_{n^2+n+4n}] | \theta_i \in \{0, \frac{1}{r}, \frac{2}{r}, \ldots, \frac{r-1}{r}, 1\} \text{ and } \sum_{j \in I_k} \theta_j = 1\}.$$

其中索引集 I_k 的范围超过 $\vec{\theta}$ 中的所有概率向量。

在离散空间中，我们执行最陡下降（代替梯度下降）以找到局部最小值。给定 $\theta_t = [\theta_1, \ldots, \theta_{n^2+n+4n}] \in \Omega_r$, 设 $\theta_t^{(i,j)} = [\theta_1, \ldots, \theta_i - \frac{1}{r}, \ldots, \theta_j + \frac{1}{r}, \ldots, \theta_{n^2+n+4n}]$ 对于每个有序对 (i, j) 在同一概率向量 $i, j \in I_k$ 中的索引概率。一个最陡的下降步骤为

$$\theta_{t+1} = \text{argmin}_{(i,j)} \left(E\left(\theta_t^{(i,j)}\right) | i, j \in I_k \text{ for some } k \right).$$

对于一些 t，当 $\theta_t \leq \theta_{t+1}$ 时, 最终下降算法终止，表明 θ_t 是离散空间中的局部最小值。

对于广义 Wang-Landau 算法中的提议分布 $Q(\theta_t, \theta')$, 我们在 θ' 中从相同概率向量中选择两个概率所产生的所有 θ_t 的空间上使用均匀分布, 将 $\frac{1}{r}$ 添加到第一个，并从第二个中减去 $\frac{1}{r}$。

当我们尝试运行离散化算法的简单实现时，会出现两个问题（1）发现多个离散局部最小值属于连续空间中的同一能量盆地（2）如果梯度在离散局部最小值处陡峭，则离散空间中局部最小值的能量可能是连续空间中相应局部最小值的能量的差近似值。因此，我们采用混合离散连续方法。我们在离散中运行主算法循环在每个样本 θ_t 被接受之后，我们（1）在用 θ_t 初始化的离散空间中进行最速下降，以找到离散局部最小值 θ_t^*；（2）在用 θ_t^* 初始化的连续空间中进行梯度下降，以找到更精确的局部最小 θ_t。离散空间的使用限制了本地的数量最小值和梯度下降计算的数量以及随后使用的连续空间合并属于同一连续能量盆地的离散局部最小值。为了改善能量边界估计，我们重复以下两个步骤直到收敛：在离散网格上运行脊下降并将离散化精细化 2 倍。

11.4.4 实验

我们通过简化从 Penn Treebank 的 WSJ 语料库中学习的树库语法，构建了几个简单英语语法的依赖语法。依赖语法中的每个节点代表一个词性标签，如名词，动词，形容词和副词。通过删除 WSJ 语料库中出现频率最低的节点来完成简化。

我们首先根据样本句子长度探索了一个课程。我们使用 3 节点依赖语法并使用离散化因子 $r = 10$ 对假设空间进行离散化。用 θ_e 表示该语法。接下来我们采样 $m = 200$ 个句子 $D = \{x_j | j = 1, \ldots, 200\}$，来自 θ_e。我们将 $D_i \subset D$ 定义为所有句子的集合 x_j 包含 i 或更少的单词。设 $w(x_j)$ 为句子 x_j 的字数，则 $D_i = \{x_j | w(x_j) \leq i\}$。集合 D_i 嵌套 ($D_i \subseteq D_{i+1}$) 和 $\cup_i^\infty D_i = D$. 在课程学习过程中，第 i 阶段使用 D_i 训练。图 11.21 (a-g) 显示了课程阶段 1 到 7 的能源景观地图。

接下来，我们根据语法中的节点数 n 来探索课程。我们使用 5 节点依赖语法及其简化为 $n = 4, 3, 2, 1$ 节点，离散因子 $r = 10$. 我们采样 $m = 200$ 个句子 $D_i = \{x_j | j = 1, \ldots, 200\}$ 来自每个语法 $\theta_i, i = 1, \ldots, 5$。再次，课程学习的第 i 阶段使用 D_i 训练。图 11.22 (a-d) 显示了课程阶段 2 到 5 的能源景观地图。阶段 1 的 ELM 被省略，因为它与图 11.21 (a) 中的 ELM 相同，因为景观的凸起。

对于这两个课程（基于句子长度和基于语法中节点的数量），我们观察到 ELM 在课程的后期阶段变得更加复杂; 后期的景观更平坦，更具局部极小。在图中所示的每个 ELM 中，全局最小值突出显示在红色和与上一课程阶段的全球最小值相近的最小局部最小值以蓝色突出显示。很明显，对于基于句子长度（图 11.21 c-g ）的课程的第 3-7 阶段和基于节点数量的课程的阶段 3-5（图和 11.22 b-d），课程阶段 i 的全球最小值是接近阶段 $i+1$ 的全局最小值。这为课程学习的性能优势提供了解释：早期阶段（可以更容易地学习）为后期阶段提供了良好的起始猜测，这允许后期阶段收敛到更好的局部最小值，这也

导致总体上更少的计算时间。

最后，我们对训练数据运行了期望最大化学习算法，以确认课程学习的优势。在实验中使用第二课程（基于节点数）。为了加快课程学习，我们为每次运行分配了 18,000 秒的总运行时间，并为每个连续阶段分配两倍于前一阶段的时间。选择课程设计的指数级增长时间是因为后期阶段的复杂性需要更多时间来收敛。我们运行了 1,000 次学习算法，找到了学习模型所属 ELM 的能量盆。因此，我们在 ELM 的叶节点上获得了学习模型的直方图，如图 11.23 (b) 所示。为了比较，图 11.23 (a) 显示了在不使用课程的情况下学习的模型的直方图。课程的使用导致更频繁地融合到全球最低点以及全球最低点附近的能源盆地。

11.5 用景点 - 扩散映射景观

11.5.1 宏观景观结构和亚稳态

ELM 应用的一个主要障碍是在实践中遇到的非凸面景观中可以找到的天文数量的局部最小值。当不可能完全枚举所有本地模式时，映射无法收敛，而过多的信息可能使映射无用。

高度非凸的景观可以具有简单且可识别的全局结构。一个众所周知的例子是与蛋白质折叠相关的潜在能量表面的“漏斗”结构。漏斗形状适合于将未折叠或部分折叠的蛋白质引导至其天然状态。沿着折叠路径可能发生弱稳定的中间状态，但是来自环境的随机扰动足以扰乱这些浅层最小值并允许折叠过程继续进行。一旦蛋白质达到其天然状态，其构型稳定并且抵抗小扰动。宏观景观具有单一的全局盆地，尽管沿漏斗“两侧”的弱稳定中间态的天文数。

在这种观察的启发下，人们可以为 ELM 定义一个新的框架，该框架旨在识别非凸面景观中的*宏观*结构，同时忽略嘈杂的局部结构。直觉上，人们可以将高维状态空间想象成一个巨大且大部分为空的宇宙，非凸能量 E 作为引力势能，而 E 作为密集恒星的局部最小值。由低能垒（在系统能量方面具有相似性质的状态）分隔的相关局部最小值组形成连通的低能量，即图像宇宙中的“星系”（见图 11.25）。

可以使用*亚稳态*的概念来定义非凸面景观中局部最小值的“星系”。不是将状态空间划分为每个局部最小值的吸引盆，而是可以将图像空间划分为亚稳区域，使得 E 上的扩散过程在区域内的短时间尺度上混合，而混合在长时间尺度上发生。区域之间。换句话说，从图像星系发起的局部 MCMC 样本 p 将在相同的星系中传播很长一段时间，因为随机波动足以克服星系内的小能量障碍，而更大的能量障碍限制星系之间的运动。

形式上，星系是满足属性的不相交集 $\{G_i\} \subset \Omega$

$$\frac{\sup_{x \notin \cup_i B_i} \mathrm{E}[\tau(x, \cup_i B_i)]}{\inf_{x \in \cup_i B_i} \mathrm{E}[\tau(x, \cup_{k \neq \varphi(x)} B_k)]} = o(1) \tag{11.13}$$

其中 $\tau(x, \mathscr{D})$ 是一组 $\mathscr{D}$ 的命中时间，而 $\varphi(x)$ 给出了指数 i，其中 $x \in G_i$. 分子项要求星系是吸引子，所以 MCMC 样本在星系将在短时间内访问其中一个星系。分母要求星系之间的混合时间很慢。至关重要的是，击中时间 $\tau(x, \mathscr{D})$ 的定义隐含地取决于所使用的 MCMC 采样器。当检测亚稳态时，实际上希望使用混合不良而不是良好混合的采样器，因为这里提出的亚稳态的定义实际上依赖于 MCMC 样本在强模式下捕获时表现出的高自相关性。

11.5.2 吸引-扩散简介

吸引-扩散是一种表征高度非凸面景观中局部最小值的相对稳定性的方法。给定能量函数 E 和两个局部最小值，将一个最小值指定为起始位置 X_0，将另一个最小值指定为目标位置 X^*。使用改变的密度从 X_0 启动 MCMC 样本

$$p_{T,\alpha,X^*}(X) = \frac{1}{Z_{T,\alpha,X^*}} \exp\{-(E(X)/T + \alpha||X - X^*||_2)\} \tag{11.14}$$

其能量函数是原始能量 E 和惩罚当前状态与目标位置之间距离的“磁化”项的总和。T 给出系统的温度，而 α 是“磁场”的强弱，与目标最小值的距离有所不同。起始位置和目标位置的作用是任意的，并且可以在两个方向上扩散。X 的空间可以是连续的或离散的。


Mitchell K. Hill

" 通过调整 α 和 T 的值，可以调整改变的景观，使得扩散路径可以克服原始景观中的小障碍物，同时保留在强大的盆地中。如果马尔可夫链在目标状态的近距离内，那么起始状态属于与目标状态相同的能量盆，其能量分辨率由磁化强度隐式定义。如果链条在链条的先前状态和目标状态之间的最小距离上不能改进 M 连续迭代，那么在起始位置和目标位置之间必须存在比磁化力更强的能垒。图 11.27 演示了具有两个全局盆地的简单一维景观中 AD 的基本原理。"

AD 旨在加速亚稳星系内的混合，同时保留分离星系的长时间尺度。亚稳态是由国家是否在“慢”或“快”时间尺度上混合来定义的，但所谓的“快速”星系内混合时间尺度对于有效模拟来说仍然太长。两个马尔可夫链很可能不会在计算上可行的时间内在高维空间中相遇，即使它们在同一个星系中.AD 中的吸引项使动力学偏向于增加混合速度。通过正确的调整，这种加速仅发生在相同的星系，星系之间混合的长时间尺度将保持不变。

AD 也可以用于估计最小值之间的能量势垒，因为沿着成功的扩散路径的最大能量是最小势垒高度的上限。可以通过将 α 设置在扩散路径未能到达目标的阈值之上来改进该估计。通过使用局部 MCMC 方法，如 Random-Walk Metropolis-Hastings，Component-Wise Metropolis Hastings，Gibbs 采样或 Hamiltonian Monte Carlo，可以限制扩散路径中各点之间的最大欧几里德距离，并确保步长小足以使连续图像之间的一维景观表现良好。AD 链根据磁化景观中的测地距离移动，只要磁化强度不太强，就应该与原始景观中的测地距离类似。

选择 L_2- 范数作为磁化罚分是由观察到 $\frac{d}{dX}||X||_2 = X/||X||_2$，这意味着 AD 磁化力在整个能量范围内指向均匀强度 α 的目标最小值。这可以在 Langevin 方程中看出

$$dX(t) = -\left(\nabla E(X(t))/T + \alpha \frac{X(t) - X^*}{||X(t) - X^*||_2}\right) dt + \sqrt{2}\, dW(t) \tag{11.15}$$

与磁化动力学相关联。L_1 惩罚可能会产生类似的结果。惩罚 $\alpha||X - X^*||_2^2$ 不具有期望的特性，因为磁化强度将取决于点之间的距离，并且改变的幅度将在整个景观中变化。

11.5.3 吸引 -扩散和 Ising 模型

AD 惩罚项与统计物理学中能量函数中的磁化项密切相关。考虑 N 态磁化 Ising 能量函数

$$E_{T,H}(\sigma) = -\frac{1}{T}\sum_{(i,j)\in\mathscr{N}}\sigma_i\sigma_j - H\sum_{i=1}^{N}\sigma_i \tag{11.16}$$

其中 $\sigma_i = \pm 1$, $\mathscr{N}$ 是相邻节点的集合，$T > 0$ 给出温度,H 给出外部磁场的强度。这个能量函数有时通过稍微不同的形式 $E_{T,H}(\sigma) = \frac{1}{T}\left(-\sum\sigma_i\sigma_j - H\sum\sigma_i\right)$ 参数化, 但是相同的属性和图表都有两种方式。第一项 $-\frac{1}{T}\sum\sigma_i\sigma_j$ 是标准 Ising 模型的能量函数，$-H\sum\sigma_i$ 代表均匀磁场，强度为 H ，在每个节点上。当 $H > 0$ 时，该磁场具有正磁化，鼓励每个节点处于 +1 状态。在这种情况下，$E_{T,H}$ 可以重写为

$$\begin{aligned} E^*_{T,H}(\sigma) &= E_{T,H}(\sigma) + NH \\ &= -\frac{1}{T}\sum_{(i,j)\in\mathscr{N}}\sigma_i\sigma_j + H\sum_{i=1}^{N}(1-\sigma_i) \\ &= -\frac{1}{T}\sum_{(i,j)\in\mathscr{N}}\sigma_i\sigma_j + H||\sigma-\sigma^+||_1 \end{aligned}$$

其中 σ^+ 是所有节点的 $\sigma_i^+ = 1$ 的状态。由 $E^*_{T,H}$ 定义的概率分布与 $E_{T,H}$ 定义的分布相同，因为它们仅由常数不同。类似地，当 $H < 0$ 且磁场为负时，能量函数可以重写为

$$E^*_{T,H}(\sigma) = -\frac{1}{T}\sum_{(i,j)\in\mathscr{N}}\sigma_i\sigma_j + |H|\,||\sigma-\sigma^-||_1$$

其中 σ^- 是所有 $\sigma_i^- = -1$ 的状态。这表明 H 在磁化的 Ising 模型中的作用与 α 在 (11.14) 中的作用相同, 应为 $E^*_{T,H}$ 是未磁化的 Ising 能量和一个惩罚距离 σ^+ 或 σ^- 项的总和, 镜像全局最小值。引入磁化项会扰乱标准 Ising 能量函数的对称性，并导致 σ^+ 或 σ^- 成为唯一的全局最小值，具体取决于 H 的符号。

系统相对于参数 (T,H) 的行为可以用图 11.28中的简单相图表示。点是系统的临界温度，实线是一阶相变边界。当系统的参数扫过一阶过渡边界时，当系统从主要为正状态翻转为主要为负状态时，状态空间发生不连续变化，反之亦然。另一方面，将磁场 H 扫过高于临界温度的 0 导致平滑过渡，其中正节点和负节点共存。

令 $H > 0$ 为弱磁场, 并假设温度 T 低于临界温度 T_c。在这种情况下，可能发生称为亚稳态的现象。如果系统是从随机配置初始化的（每个节点 $+1$ 或 -1 ，概率为 1/2），磁场的影响将导致系统崩溃到 σ^+ ，或附近的主要正区域国家空间，概率很高。然而，如果系统从 σ^- 初始化，并且如果 H 足够小，则系统将表现出亚稳态，因为磁力 H 将无法克服 σ^- 中的键强度，这非常强低于临界温度。尽管能量景观的全局最小值是 σ^+，但系统将长时间保持稳定的，主要为负的状态，因为磁场力无法克服原始 Ising 能源景观中 σ^+ 和 σ^- 之间的障碍。

11.5.4 吸引 -扩散 ELM 算法

ELM 方法有三个基本的探索步骤：

1. 获取状态 X 作为最小值搜索的起点。

2. 从 X 开始查找当地最小 Y。

3. 确定 Y 是否与先前找到的最小盆地组合，或者 Y 是否开始新的最小盆地。

重复这些步骤，直到在一定次数的迭代中没有找到新的局部最小值。确定局部最小值后，估计最小值之间的障碍。

步骤 2 可以使用标准梯度下降方法完成，GWL 算法提供了在步骤 1 中提出 X 的原则方法。以前的 ELM 方法缺乏可靠的方法来处理步骤 3. 传统上，ELM 研究试图列举所有吸引盆地能量景观（或 N 最低能量最小值），无论多浅。如果 Minima 在离散空间中相同，或者它们在连续空间中非常接近，则它们仅被组合在一起。除了最简单的情况之外，这种方法注定要失败，因为不同的局部最小值的数量随景观复杂性和/或维度呈指数增长。另一方面，对于一些能量函数族，随着景观复杂性/维度的增加，宏观结构可能保持不变。例如，无论邻域结构或节点数量如何，伊辛能源景观将始终具有两个全球盆地。

吸引 - 扩散 ELM 不是根据梯度流下的吸引盆地划分状态空间，而是根据在可逆 MCMC 过程引起的流动下亚稳态的不相交区域来划分状态空间。这产生了对景观的更简单的描述，因为亚稳区域将合并仅由小障碍隔开的吸引盆地。如果 AD 中使用的磁化 α 较弱，则改变的横向的亚稳区域应该大致对应于原始景观的亚稳区域，并且 AD 试验的成功或失败可以用作给定亚稳区域的成员资格的指标。在 MCMC 过程中映射亚稳定区域而不是梯度流下的吸引盆地对于 ELM 在复杂景观中的成功至关重要。

MCMC 采样器 S 应该是局部的，在一个步骤之后的位移相对于具有高概率的景观特征是小的。具有步长参数 ε 的 MCMC 方法（例如具有高斯提议的 Metropolis-Hastings 或 HMC/Langevin 动力学）是局部采样器，因为可以调整 ε 以限制位移。Gibbs 采样也是本地的，因为每次更新只改变一个维度。需要 S 是本地的，以确保使用 S 更新的马尔可夫链在低温下不能脱离本地模式。换句话说，在映射 Ising 模型的局部最小值时，必须使用 Gibbs 采样而不是 Swendsen-Wang Cuts，因为 SWC 方法可以轻松地从局部模式中出来，从而破坏整个映射过程。通常，高自相关被认为是 MCMC 方法的不希望的特征，但是在 AD 中，马克托夫样本必须在没有磁化的情况下被捕获。通过引入磁场来扰乱这种基线行为，可以找到景观特征。

在 ADELM 算法中，每个盆地的全局最小值 Z_j 被用作 AD 试验的目标。这种选择的一个原因是直觉，对于相同的强度 α，AD 链应该更可能成功地从较高能量最小值行进到较低能量最小值，反之亦然。虽然一般情况下并非如此，但在实践中，直觉在大多数情况下都适用，特别是对于非常深的最小值。更细微的实施可以考虑来自同一流域的多个候选人作为扩散目标，而不仅仅是全局最小值。

正确调整 T 和 α 对于获得良好结果至关重要。温度 T 必须设置得足够低，以便将运动限制在当前模式，但不能太低以至于链条变得完全冻结。磁化强度 α 必须足够强，以克服景观中嘈杂的浅层障碍，同时保证大规模障碍。调整 T 和 α 的一种简单方法需要两个最小 X 和 Y，它们被认为是在单独的盆中。通过扰动最小值，可以找到两个相对的 X' 和 Y'，它们应分别与 X 和 Y 在同一盆地中。然后可以调整参数 T 和 α，使得仅原始和扰动副本之间的扩散成功。选择 α 通常是最重要的调整决策。

理想情况下，在 ADELM 算法的每个步骤中，仅扩散到一个盆地代表 Z_j 应该是成功的。成功扩散到大量先前发现的盆地是调整不良的标志 – 特别是 T 或 α（或两者）的值太高，导致盆地之间的泄漏。

另一方面，最小值之间的一些泄漏通常是不可避免的，因为通常存在位于较强的全局盆地之间的高原区域。只要流域代表保持分离，这不是一个太大的问题。应定期检查全球流域代表 $\{Z_j\}$，以确保它们在当前参数设置下保持良好分离。如果 AD 链成功地在两个 $\{Z_j\}$ 之间移动，则这些最小值应合并为一个组。这在绘图的早期阶段尤其重要，因为还没有找到好的流域代表。如果早期代表不是整个流域的有效吸引状态，那么单个流域可以分成多个群。在巩固最小值时，较低能量最小值保留为组代表。

ADELM 算法有两个计算瓶颈：步骤 2 中的局部最小值搜索和步骤 3 中的 AD 分组。步骤 2 的计算成本对于任何 ELM 方法都是不可避免的，并且步骤 3 中的 MCMC 采样不是不合理的，只要它具有可比较的运行时间。在我们的实验中，我们发现局部最小搜索和单个 AD 试验的运行时间大致相同。ADELM 算法的第 3 步涉及新的最小候选和几个已知候选之间的 AD 试验，并且通过并行运行 AD 试验可以大大提高 ADELM 的效率。

11.6 应用：使用吸引 -扩散来映射图像空间

11.6.1 图像星系的结构

本节研究了训练以模拟未知图像密度 f 的学习 Gibbs 密度 p（或等效地，能量 $E = -\log p$）的结构。在训练过程中，密度学习在 f 的样本周围形成模式，E 的局部最小值可以解释为训练数据的“记忆”。图像密度的星系代表图像模式中的不同概念，并通过寻找星系在图像密度方面，我们可以将巨大的高维图像空间减少到几个总结主要模式外观的群体。我们也对测量星系之间的能量障碍感兴趣，因为它们编码了星系间的相似性。学习图像密度的结构对模式流形的存储进行编码，但此信息隐藏在 p 中，必须通过映射来恢复。

景观结构根据建模的图像模式而有所不同（见图 11.30）。特别是，图像比例应该对图像存储器的结构具有强烈影响。在模式表示的核心范式之一，Julesz 确定了两种主要的图像尺度：纹理和纹元。*纹理*是高熵模式，被定义为在附近像素之间共享相同统计数据的图像组。另一方面，*纹元*是低熵模式，可以理解为原子建筑元素或局部的，显著的特征，如条形，斑点或角落。

如图 11.29所示，文本尺度图像具有易于识别的显式结构，并且该结构允许人类将文本图像可靠地分类为连贯的组。纹理尺度图像具有隐式结构，并且通常难以或不可能在相同纹理的图像中找到组，因为在纹理集合内不能识别区别特征。随着图像比例增加，可识别图像组的数量趋于增加，直到达到可感知性阈值，其中文本尺度图像转变为纹理尺度图像，并且人类开始失去识别区别特征的能力。超出可感知性的阈值，纹理图像不能被分开或可靠地分组。图像比例的变化导致图像的统计特性发生变化，我们将这种现象称为*信息缩放*。

信息缩放反映在图像景观的结构中，并且图案图像之间差异的可感知性与局部最小图像的稳定性/深度之间存在联系。当景观模拟文本尺度图像时，可以很容易地区分图像中的组，人们期望在景观中找到许多单独的，稳定的盆地，这些盆地编码各组的单独外观。另一方面，对纹理尺度图像进行建模的景观应该表现出与人类感知类似的行为，并形成单个宏观的吸引盆地，其中许多浅的局部最小值用于编码纹理。通过在多个尺度上映射来自相同图案的图像，可以表明在尺度之间发生的可感知性的转变导致图像存储器的横向结构的转变（参见图 11.30和图 11.34）。

11.6.2 实验

整个部分的目标能量函数是

$$E_{W_1,W_2}(Z) = E(g(Z|W_2)|W_1) \tag{11.17}$$

其中 $E(\cdot|W_1)$ 和 $g(\cdot|W_2)$ 是根据 Co-Op 网络算法 [30] 学习的。ADELM 的步骤 1 中的提议可以通过从发电机网络的潜在分布中取样来获得. (11.17) 的公式提供了一种使用 ADELM 有效地映射在实际大小的图像上定义的 DeepFRAME 函数的方法。

潜在空间中的数字 0-9 ELM

我们应用 ADELM 来映射 Co-Op Networks 的能量 (11.17)，模拟 MNIST 的所有数字。我们使用 MNIST 测试集的前半部分作为我们的训练数据（每个数字约 500 个例子）。这次，我们将图像大小增加到 64×64 像素。由于我们只会在具有低维度的潜在空间中进行采样，因此我们可以在训练期间使用逼真大小的图像。

描述符网络有三层：两个卷积层，大小为 5×5 的 200 个滤波器和 100 个滤波器，以及一个完全连接的 10 个滤波器层。每层后面都有一个 ReLu 激活功能。潜在的发电机分布是 8 维法线 $\mathrm{N}(0, I_8)$。发电机网络具有三层尺寸 4×4, 7×7, 和 7×7，分别具有 200,100 和 10 个滤波器。ReLu 用作前两层的激活，tanh 用作最后一层的激活。在每个发生器层之后使用上采样因子 4。AD 参数为 $T = 1200$ 和 $\alpha = 230$. 使用的其他 ADELM 参数与斑点/条纹潜在空间 ELM 中的相同。对于映射，使用了 500 次老化迭代和 5000 次测试迭代，结果如图 11.31所示。

图 11.31中的 ELM 具有许多强大，分离良好的能量盆。仔细观察 DG 表明，所有 10 个数字都至少由一个强大的极小盆地代表。盆地成员和 DG 的全局结构都与人类视觉直觉紧密相关。

潜在空间中的 Ivy Texton ELM

我们现在映射一个 Co-Op 网络，该网络是通过常春藤纹理训练的。在近距离范围内，常春藤补丁具有独特且可识别的结构，并且映射的目标是识别在常春藤纹理中重现的主要模式。图 11.32显示整个常春藤纹理图像以及从四个不同比例拍摄的纹理中的图像块。本实验中的网络经过训练，可以从 Scale 2 中对 1000 个图像块进行建模。

图 11.33中用于常春藤纹理映射的 DG 表明景观由 3 个或 4 个全局盆地控制。盆地内的图像非常一致，盆地之间的障碍代表了极小图像之间的视觉相似性。与数字映射不同，最小值分组没有基本事实，因此探索不同能量分辨率的景观以识别不同视觉相似度的图像分组是有用的。ADELM 的一个主要优点是能够简单地通过改变 AD 试验期间使用的磁化强度 α 来执行不同能量分辨率的映射。图 11.33 在不同的能量分辨率下呈现两个相同景观的映射。相同的景观特征出现在两个映射中，具有或多或少的子结构取决于磁化强度。

潜在空间中的多尺度常春藤 ELM

我们继续研究上一节中的常春藤纹理图像，方法是绘制一个 Co-Op 网络，该网络是从图 11.32所示的四个比例尺中的每一个进行训练的。在这个实验中，我们想要研究不同尺度之间记忆形成的差异。特别是，我们有兴趣确定景观中局部极小值的亚稳态与感知之间的关系。最小的视觉差异的能力。我们期望在极端尺度上找到更少的结构。尺度 1 的图像块大多是纯色图像，几乎没有变化，应该在景观中形成一些强大的盆地。来自 Scale 4 的图像块没有明显的特征，并且不能被人类分开，所以我们期望这些图像将形成没有太多子结构的宽盆。对于中间尺度，我们期望找到更丰富的稳定局部最小值，因为中间尺度包含比尺度 1 更多的变化，但是与尺度 4 图像相比，仍然可以在视觉上区分变化。

景观的结构确实在图像尺度之间不同。正如预期的那样，Scale 1 的记忆形成了一些强大的盆地。Scale 2 占据了景观中的大部分盆地，因为该尺度包含多种可感知的图像外观。Scale 2 盆地与 DG 可视化中的 Scale 1 盆地合并，表明景观的这些区域之间存在可访问的低能量连接。来自 Scale 3 和 Scale 4 的图像各自形成能量景观的单独区域，具有很少的子结构。该映射表明常春藤纹理图像的感知阈值（至少在 Co-Op 网络学习的记忆方面）位于 Scale 2 和 Scale 3 之间。在可感知阈值之上，网络不能可靠地区分图像之间的变化，并且景观形成单一区域，没有明显的子结构。人类很难区分 Scale 3 中的图像组，因此网络的可感知阈值看起来与人类相似。

潜在空间中的猫脸 ELM

对于我们的最终实验，我们绘制了一个 Co-Op 网络，该网络通过互联网收集的猫脸图像进行训练。我们的映射结果如图 11.35所示。DG 有一个分支，能量障碍很浅。局部最小值的主要特征是猫脸的几何形状和颜色，但是这些在插值期间可以平滑地变形而不会遇到不可能的图像，与诸如数字的图像相反，图像必须沿着插值路径进入不可能的几何配置。出于这个原因，整个猫的景观中的能量障碍非常低。尽管如此，ADELM 发现的全球盆地一致地识别出猫脸的主要群体。即使大多数流域成员的能量高于流域合并的障碍，AD 也能有效地识别景观结构。

参考文献

[1] Adrian Barbu and Song-Chun Zhu. Generalizing swendsen-wang to sampling arbitrary posterior probabilities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1239–1253, 2005.

[2] Oren M Becker and Martin Karplus. The topology of multidimensional potential energy surfaces: Theory and application to peptide structure and kinetics. *The Journal of chemical physics*, 106(4):1495–1517, 1997.

[3] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, pages 41–48, 2009.

[4] C L Blake and C J Merz. Uci repository of machine learning databases, 1998. Robustness of maximum boxes.

[5] Stephen P. Brooks and Andrew Gelman. General Methods for Monitoring Convergence of Iterative Simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, December 1998.

[6] Eugene Charniak. Immediate-head parsing for language models. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 124–131, 2001.

[7] Michael Collins. *Head-driven statistical models for natural language parsing*. PhD thesis, University of Pennsylvania, 1999.

[8] Sanjoy Dasgupta and Leonard J. Schulman. A two-round variant of em for gaussian mixtures. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, UAI'00, pages 152–159, 2000.

[9] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[10] Jeffrey L Elman. Learning and development in neural networks: The importance of starting small. *Cognition*, 48(1):71–99, 1993.

[11] Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*, 11:2001–2049, 2010.

[12] Andrew Gelman and Donald B Rubin. Inference from iterative simulation using multiple sequences. *Statistical science*, pages 457–472, 1992.

[13] Charles J Geyer and Elizabeth A Thompson. Annealing markov chain monte carlo with applications to ancestral inference. *Journal of the American Statistical Association*, 90(431):909–920, 1995.

[14] William P Headden III, Mark Johnson, and David McClosky. Improving unsupervised dependency parsing with richer contexts and smoothing. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 101–109, 2009.

[15] Dan Klein and Christopher D Manning. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 478, 2004.

[16] Sandra Kübler, Ryan McDonald, and Joakim Nivre. Dependency parsing. *Synthesis Lectures on Human Language Technologies*, 1(1):1–127, 2009.

[17] Faming Liang. A generalized wang-landau algorithm for monte carlo computation. *Journal of the American Statistical Association*, 100(472):1311–1327, 2005.

[18] Faming Liang, Chuanhai Liu, and Raymond J Carroll. Stochastic approximation in monte carlo computation. *Journal of the American Statistical Association*, 102(477):305–320, 2007.

[19] Enzo Marinari and Giorgio Parisi. Simulated tempering: a new monte carlo scheme. *EPL (Europhysics Letters)*, 19(6):451, 1992.

[20] Igor' Aleksandrovič Mel'čuk. *Dependency syntax: theory and practice*. SUNY Press, 1988.

[21] José Nelson Onuchic, Zaida Luthey-Schulten, and Peter G Wolynes. Theory of protein folding: the energy landscape perspective. *Annual review of physical chemistry*, 48(1):545–600, 1997.

[22] Douglas LT Rohde and David C Plaut. Language acquisition in the absence of explicit negative evidence: How important is starting small? *Cognition*, 72(1):67–109, 1999.

[23] Rajhans Samdani, Ming-Wei Chang, and Dan Roth. Unified expectation maximization. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 688–698. Association for Computational Linguistics, 2012.

[24] Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *NAACL*, 2010.

[25] Robert H Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. *Physical Review Letters*, 58(2):86–88, 1987.

[26] Martin A Tanner and Wing Hung Wong. The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, 82(398):528–540, 1987.

[27] Kewei Tu and Vasant Honavar. On the utility of curricula in unsupervised learning of probabilistic grammars. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1523, 2011.

[28] Kewei Tu and Vasant Honavar. Unambiguity regularization for unsupervised learning of probabilistic grammars. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Natural Language Learning (EMNLP-CoNLL 2012)*, 2012.

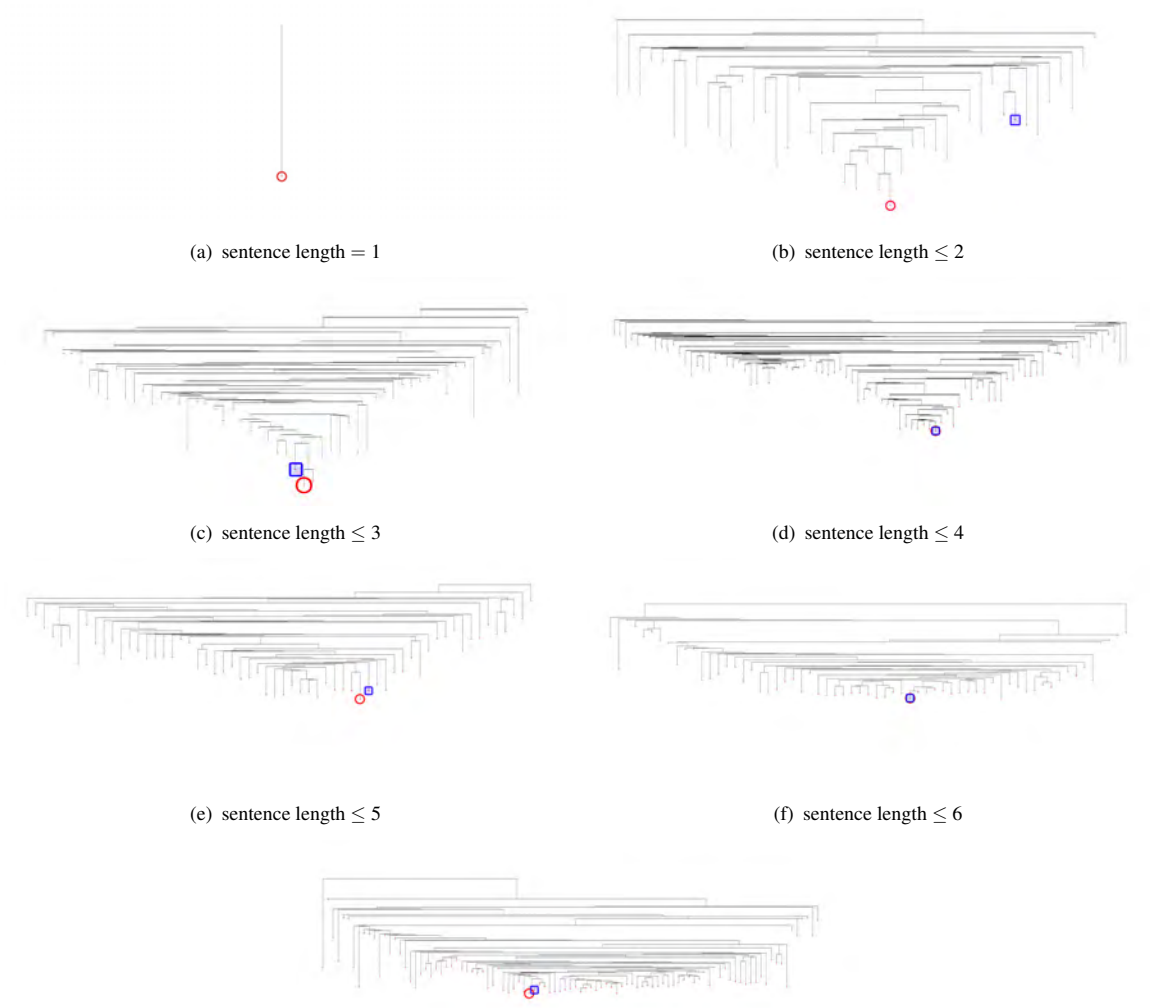
[29] Fugao Wang and David P Landau. Efficient, multiple-range random walk algorithm to calculate the density of states. *Physical review letters*, 86(10):2050, 2001.

[30] Jianwen Xie, Yang Lu, and Ying Nian Wu. Cooperative learning of energy-baed model and latent variable model via mcmc teaching. *AAAI*, 2018.

[31] Qing Zhou. Multi-domain sampling with applications to structural inference of bayesian networks. *Journal of the American Statistical Association*, 106(496):1317–1330, 2011.

[32] Qing Zhou. Random walk over basins of attraction to construct ising energy landscapes. *Physical review letters*, 106(18):180602, 2011.

[33] Qing Zhou and Wing Hung Wong. Reconstructing the energy landscape of a distribution from monte carlo samples. *The Annals of Applied Statistics*, 2:1307–1331, 2008.



(a) sentence length $= 1$

(b) sentence length ≤ 2

(c) sentence length ≤ 3

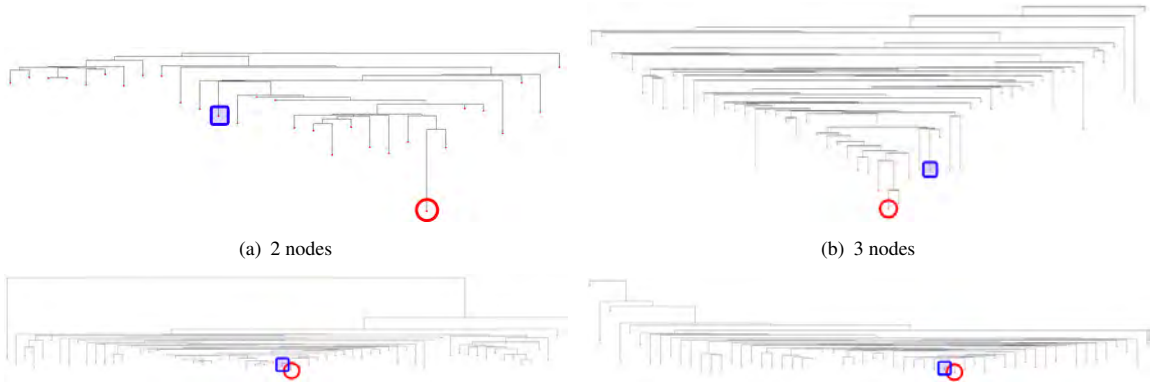
(d) sentence length ≤ 4

(e) sentence length ≤ 5

(f) sentence length ≤ 6

(g) sentence length ≤ 7

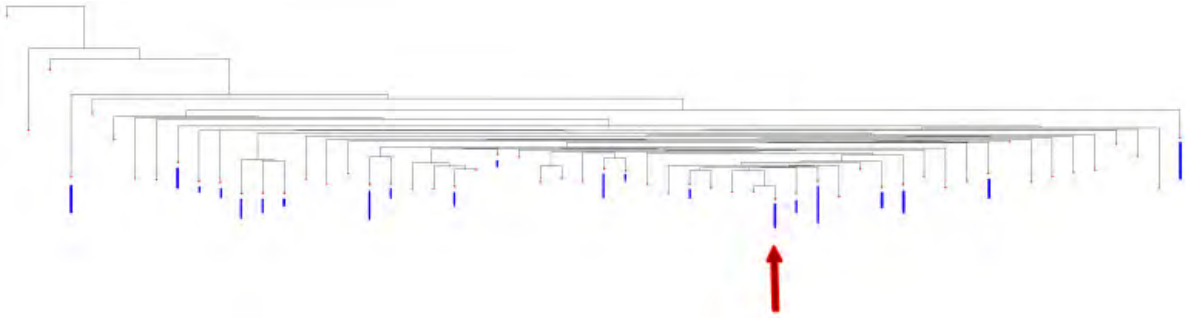
图 11.21: 基于训练样本句子长度的课程。



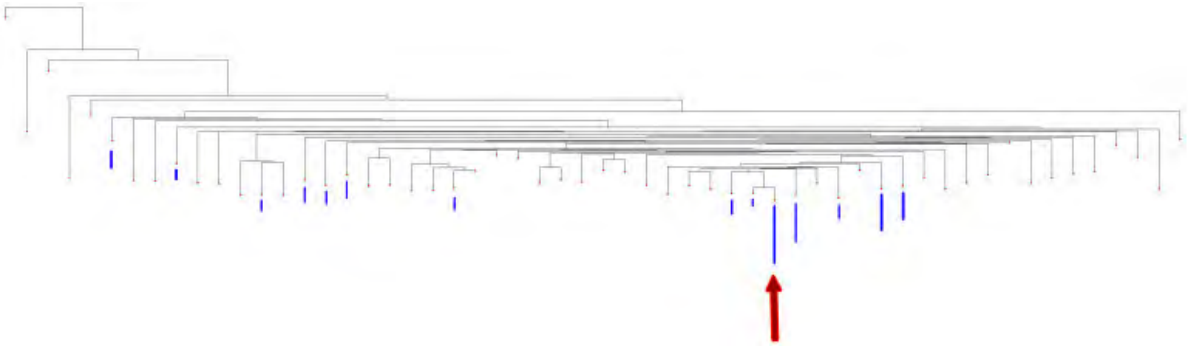
(a) 2 nodes (b) 3 nodes

(c) 4 nodes (d) 5 nodes

图 11.22: 基于节点数的课程



(a) no curriculum



(b) exponentially decreasing curriculum

图 11.23: 学习语法的分布（a）没有学习课程（b）与时间有限的课程。蓝条直方图表示属于每个能量盆地的学习语法的数量，红色箭头表示地面真实解的能量盆地。

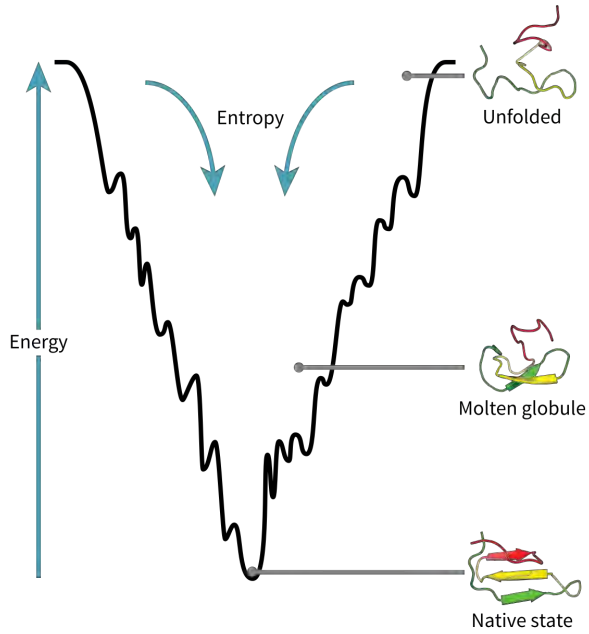


图 11.24: 蛋白质折叠的势能图。通过景观的漏斗结构将未折叠的蛋白质引导至其天然状态。景观有大量的局部最小值，但它们很浅。景观的宏观结构只有一个全局盆地。

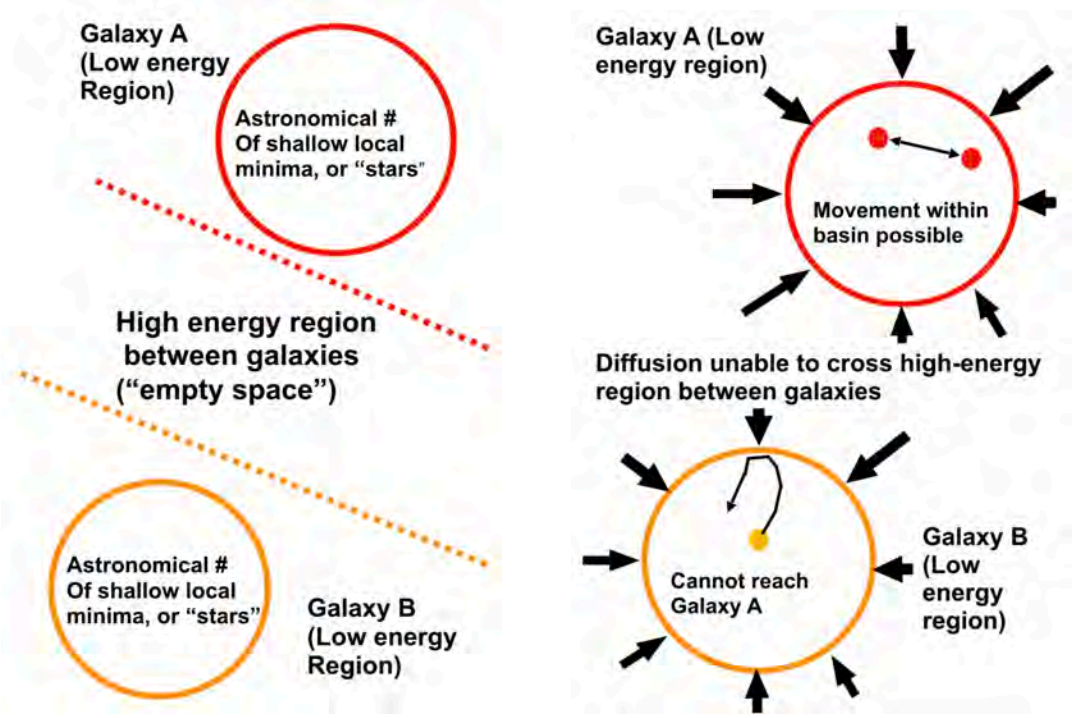


图 11.25: *左图:* 局部极小“星系”的简化图。圆是具有高密度的低维歧管。在星系之间是高能量，“空”区域（实际上，与星系大小相比，空的空间是巨大的）。*右图:* 图像星系中亚稳行为图。吸引扩散算法用于检测这种行为。

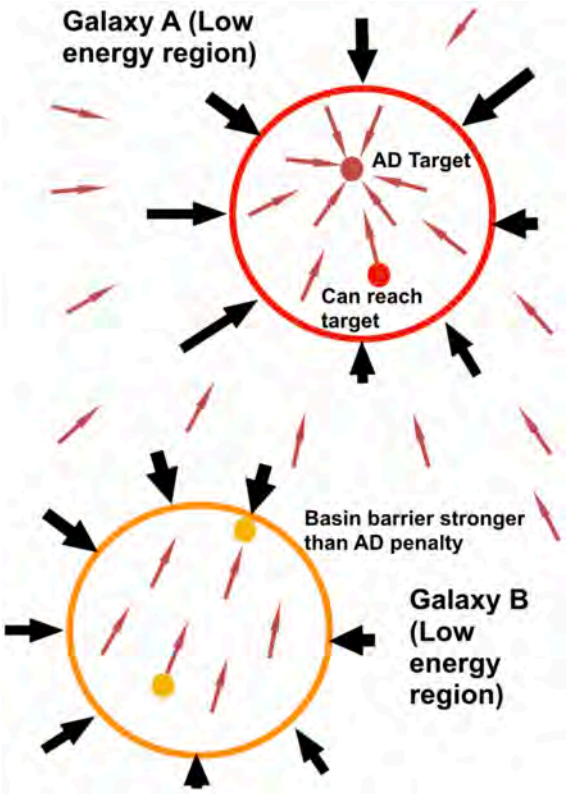


图 11.26: 用 AD 检测亚稳行为的可视化。AD 惩罚指向整个景观中具有恒定强度 α 的目标。从与目标相同的图像星系初始化的 MCMC 样本将快速传播到目标。从不同星系发起的 MCMC 样本可以在短时间内接近目标，但最终会被分隔星系的强大障碍所困。

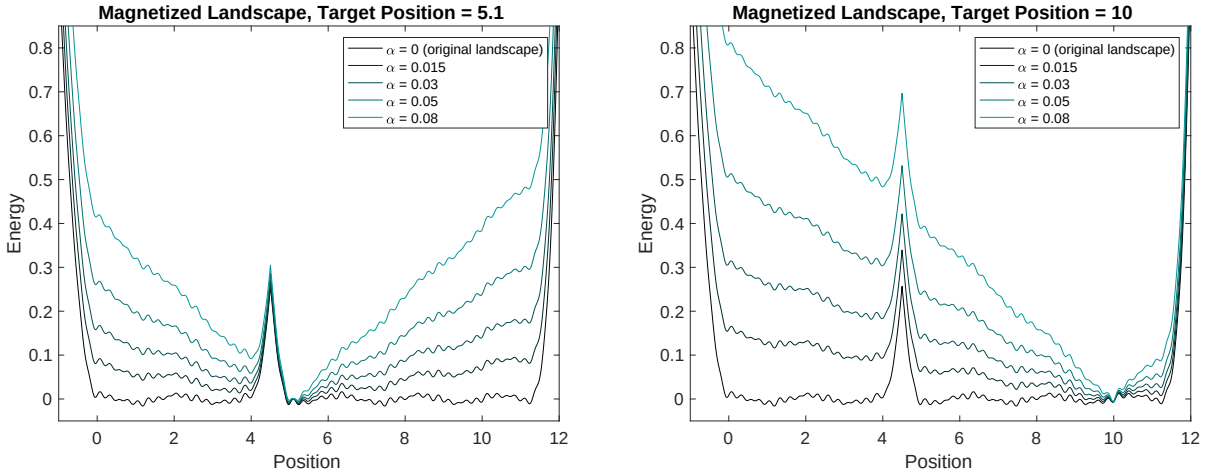


图 11.27: 工具 1D 景观的磁化，目标位置 $X = 5.1$ （左）和 $X = 10$ （右）。原始景观有两个平坦而嘈杂的盆地。两个目标位置都属于同一个盆地，即使它们在欧几里德空间中很远。磁化景观具有易于识别的最小值，并保留分隔两个盆地的大屏障。由于从 $X = 10$ 开始的左手景观中的扩散将达到 $X = 5.1$ ，反之亦然，在右手景观中，这些点属于同一盆地。从屏障左侧开始的低温扩散将无法到达目标位置景观。

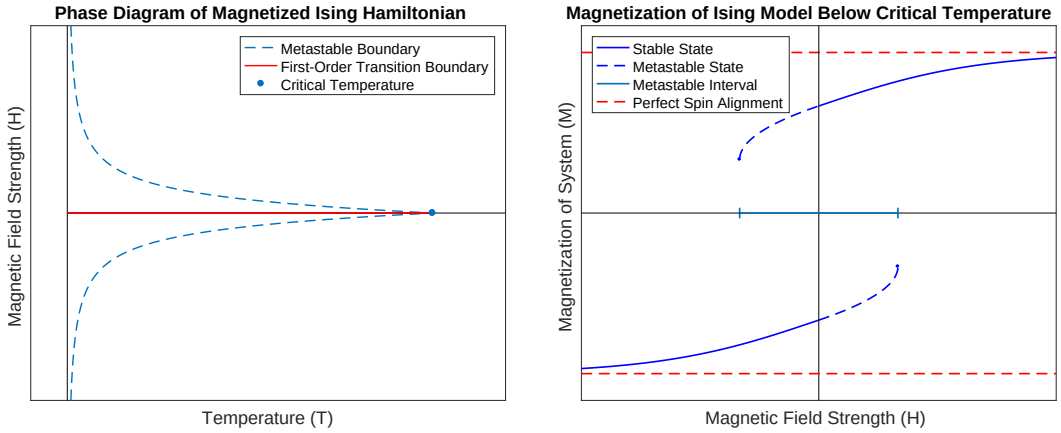


图 11.28: *左图:* 磁化 Ising 模型的相图。低于临界温度，将磁场 H 从正向扫描到负向（或反之亦然），导致 σ^+ 和 σ^- 的盆地之间发生跳跃。但是，如果磁化力较弱，则相对盆地中的状态可以长时间保持稳定。*右图:* 对于固定的 $T^* < T_c$，磁化 $M = \sum_i \sigma_i$ 作为 H 的函数。亚稳间隔是沿左图中垂直线 $T = T^*$ 的虚线之间的区域。

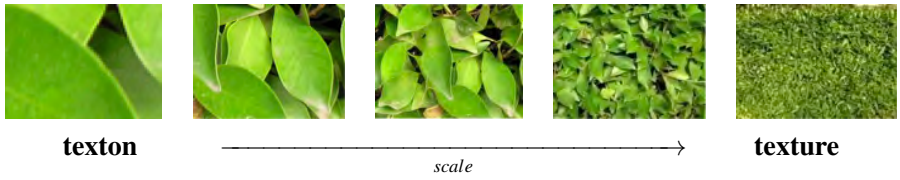


图 11.29: 常春藤叶子在不同的尺度。随着图像比例从左向右增加，可以识别出越来越多种类型的图像组，直到达到可感知性的阈值，之后变得难以区分图像。第四个尺度接近人类的感知阈值，而第五个尺度超出人类可感知性。当超越阈值时，会发生从显式，稀疏结构到隐式密集结构的制度转变。能源领域也发生了类似的转变 (见图 11.34).

input : Target energy E, local MCMC sampler S, temperature $T > 0$, magnetization force $\alpha > 0$, distance resolution $\delta > 0$, improvement limit M, number of iterations N

output: States $\{X_1, \ldots, X_N\}$ with local minima $\{Y_1, \ldots, Y_N\}$, minima group labels $\{l_1, \ldots, l_N\}$, and group global minima $\{Z_1, \ldots, Z_L\}$, where $L = \max\{l_n\}$

for $n = 1 : N$ **do**

1. Get proposal state X_n for minima search. (Random initialization, or a GWL MCMC proposal)
2. Start a local minimum search from X_n and find a local minimum Y_n.
3. **if** $n = 1$ **then**
   Set $Z_1 = Y_1$ and $l_1 = 1$.

end

else

Determine if Y_n can be grouped with a known group using AD. Let $L_n = \max\{l_1, \ldots, l_{n-1}\}$, and let minimum group membership set $G_n = \emptyset$.

for $j = 1 : L_n$ **do**

a) Set $C = Y_n$, $X^* = Z_j$, $d_1 = ||C - X^*||_2$, $d^* = d_1$, and $m = 0$.

while $(d_1 > \delta)$ & $(m < M)$ **do**

Update C with a single step of sampler S using the density

$$P(X) = \frac{1}{Z} \exp\{-(E(X)/T + \alpha||X - X^*||_2)\}$$

and find the new distance to the target minimum: $d_1 \leftarrow ||C - X^*||_2$.

If $d_1 \geq d^*$ **then** $m \leftarrow m + 1$, **else** $m \leftarrow 0$ and $d^* \leftarrow d_1$.

end

b) Set $C = Z_j$, $X^* = Y_n$, $d_2 = ||C - X^*||_2$, $d^* = d_1$, and $m = 0$, and repeat the loop in Step a).

c) If $d_1 \leq \delta$ or $d_2 \leq \delta$, then add j to the set G_n, and let B_j be the barrier along the successful path. If both paths are successful, let B_j be the smaller of the two barriers.

end

if G_n *is empty* **then**

Y_n starts a new minima group. Set $l_n = \max\{l_1, \ldots, l_{n-1}\} + 1$, and $Z_{l_n} = Y_n$.

end

else

Y_n belongs to a previous minima group. Set $l_n = \text{argmin}_j B_j$.

if $E(Y_n) < E(Z_{l_n})$ **then**

Update the group global minimum: $Z_{l_n} \leftarrow Y_n$.

end

end

end

end

Algorithm 1: Attraction-Diffusion ELM (ADELM)

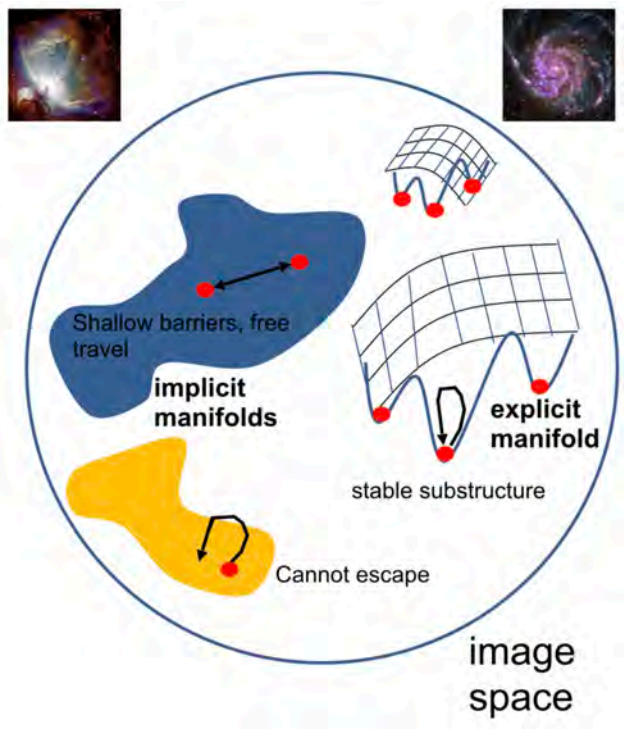


图 11.30: 纹理（隐式流形）和纹理（显式流形）的图像星系图。纹理适用于没有内部结构的宽大和阴影区域，而纹理形成具有稳定子结构的星系，以编码不同典型的纹理外观。

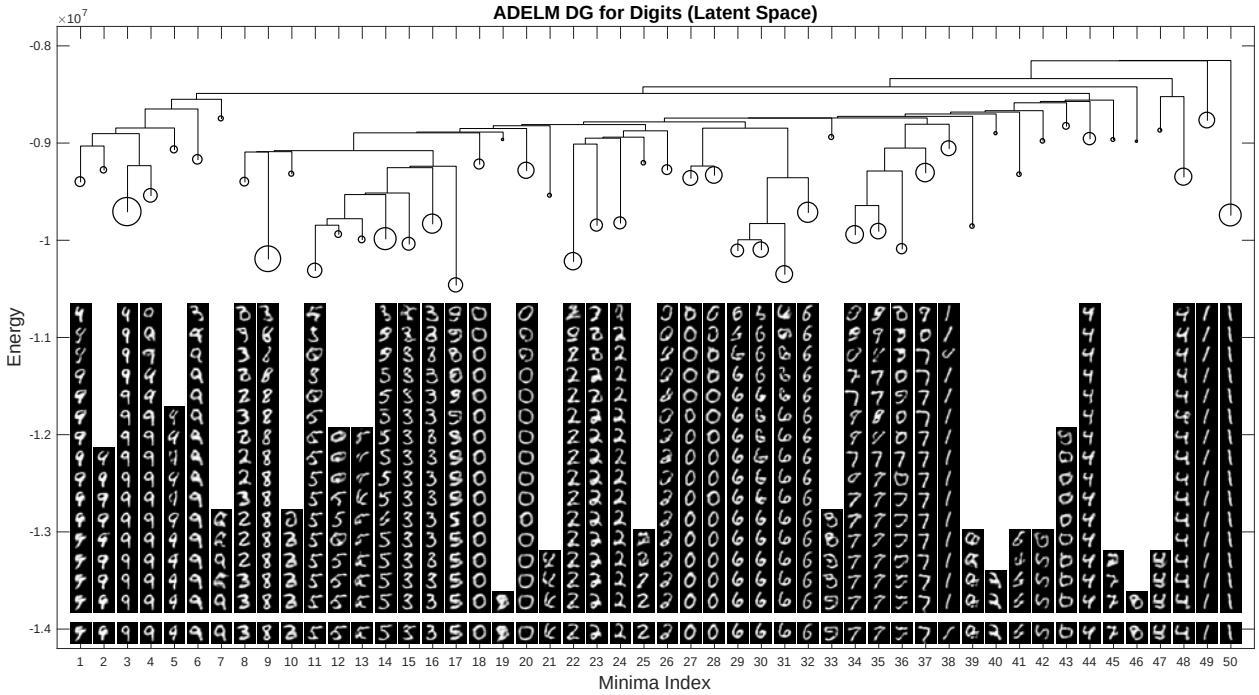


图 11.31: 潜伏空间中数字 0-9 ELM 的 DG。描述符网络超过 64×64 个图像，但生成器潜在空间只有 8 个维度，允许有效的映射。值得注意的是，所有 10 个数字在 DG 中至少有一个分离良好的分支。代表相同数字的最小值通常以低能量水平合并。

图 11.32: 常春藤纹理图像和四个尺度的图像补丁

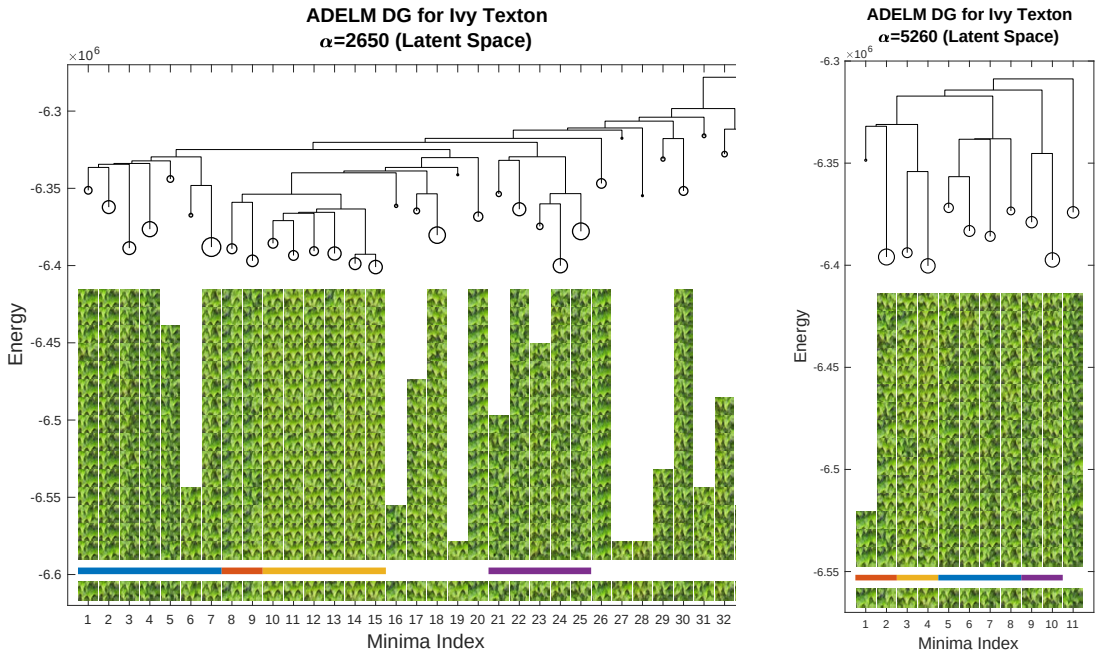


图 11.33: 常春藤纹理的 DG，用于两种不同的磁化强度 α。这两个映射都显示出这些盆地中有 3 个强大的全球盆地和子结构，这些盆地在不同的磁化条件下是稳定的。对于纹理图像块没有地面真实分组，因此以多种分辨率映射图像结构以识别不同视觉相似度下的“概念”是有用的。盆地代表下方的颜色表示出现在两个映射中的区域。

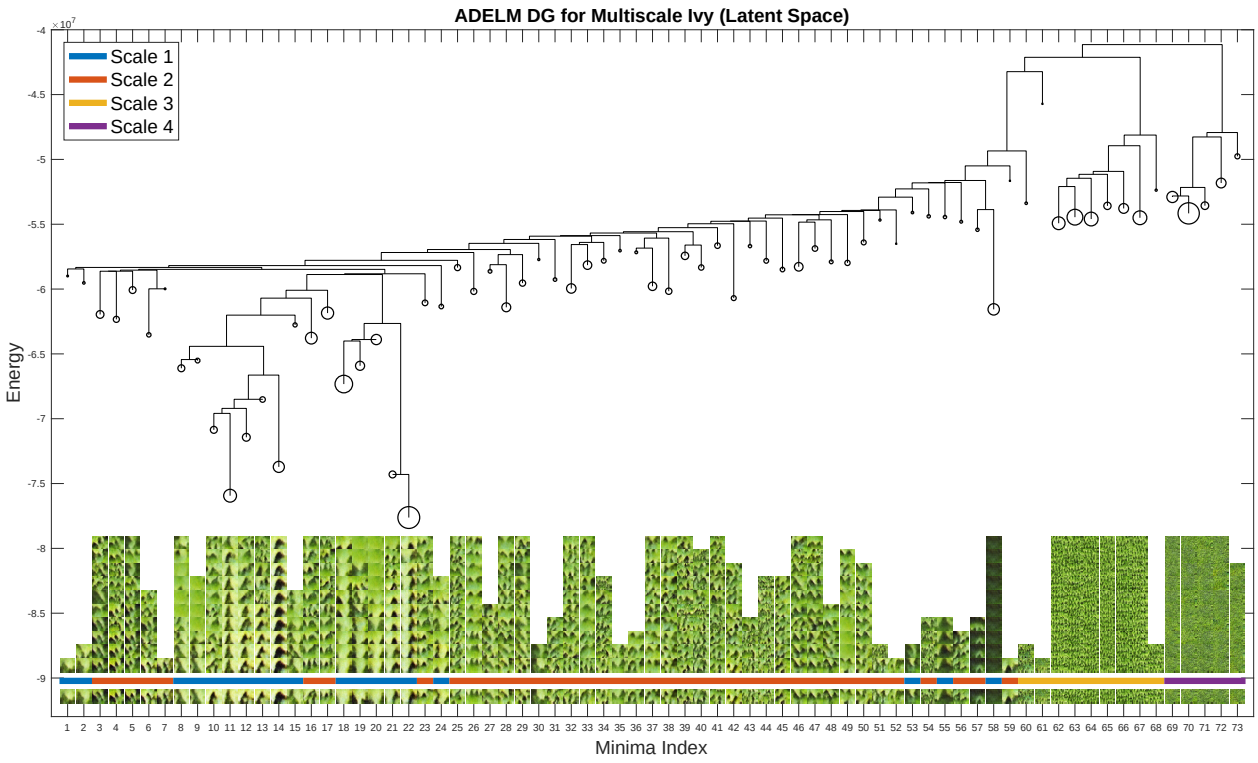


图 11.34: 四种不同尺度的常春藤图像斑块的景观。标尺 1 和标尺 2 的图像是纹理，而标尺 3 和标尺 4 的图像是纹理。纹理尺度图像占景观中的大部分盆地。标尺 2 比标尺 1 更多的盆地被识别，因为标尺 2 具有更丰富多样的不同外观，而标尺 1 最小值具有更低的能量，因为来自该标尺的外观更可靠。纹理尺度图像形成具有很少子结构的独立盆地。每个尺度的盆地成员如图 **??**。参见章节 **??** 完整的解释。

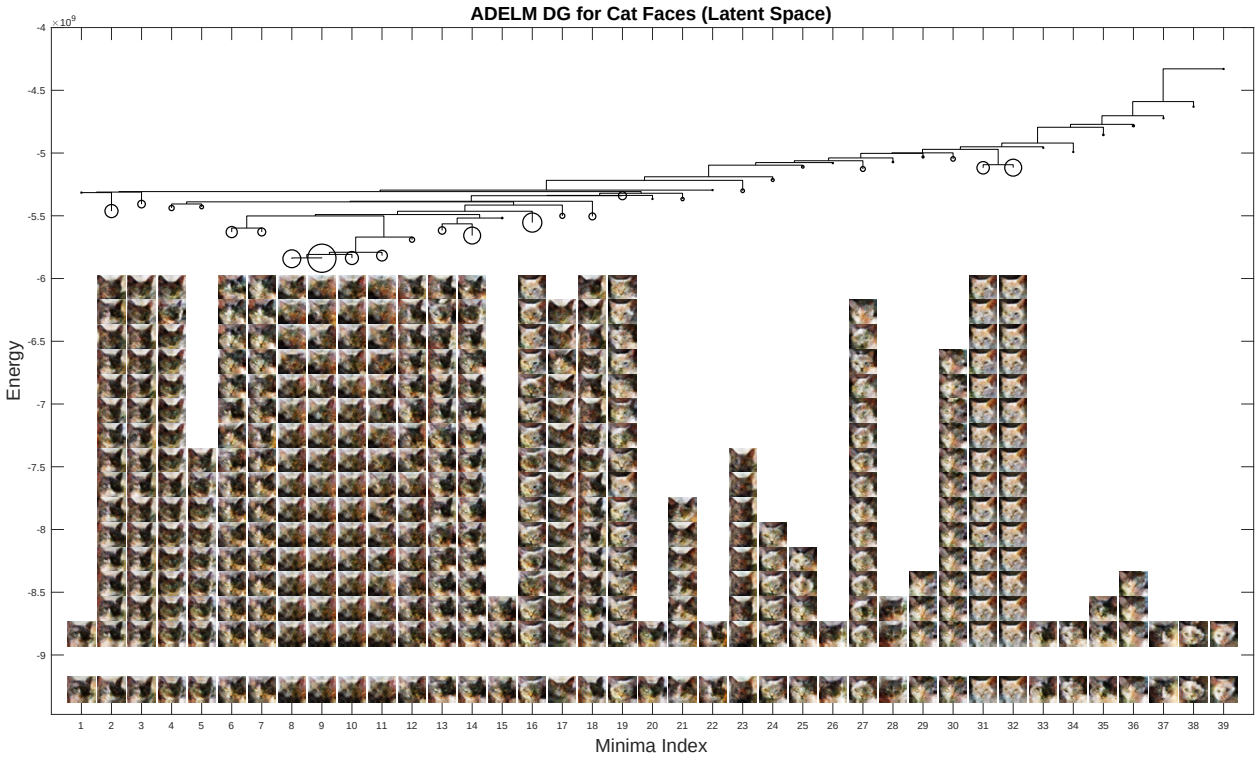


图 11.35: 猫脸在潜伏空间中的 DG。景观有一个单一的全球盆地，可能是因为很容易找到猫脸之间关于几何和颜色约束的插值，不像数字之间的插值，它必须沿着路径通过高能几何配置。尽管缺乏整体景观结构，但 AD 能够找到显示各种猫脸的有意义的图像盆。