第十七章　蒙特卡罗方法

随机算法可以粗略地分为两类：Las Vegas算法和蒙特卡罗算法。Las Vegas算法总是精确地返回一个正确答案（或者返回算法失败了）。这类方法通常需要占用随机量的计算资源（一般指内存或运行时间）。与此相对的，蒙特卡罗方法返回的答案具有随机大小的错误。花费更多的计算资源（通常包括内存和运行时间）可以减少这种错误。在任意固定的计算资源下，蒙特卡罗算法可以得到一个近似解。

对于机器学习中的许多问题来说，我们很难得到精确的答案。这类问题很难用精确的确定性算法如Las Vegas算法解决。取而代之的是确定性的近似算法或蒙特卡罗近似方法。这两种方法在机器学习中都非常普遍。本章主要关注蒙特卡罗方法。

17.1　采样和蒙特卡罗方法

机器学习中的许多重要工具都基于从某种分布中采样以及用这些样本对目标量做一个蒙特卡罗估计。

17.1.1　为什么需要采样?

有许多原因使我们希望从某个分布中采样。当我们需要以较小的代价近似许多项的和或某个积分时，采样是一种很灵活的选择。有时候，我们使用它加速一些很费时却易于处理的求和估计，就像我们使用小批量对整个训练代价进行子采样一样。在其他情况下，我们需要近似一个难以处理的求和或积分，例如估计一个无向模型中配分函数对数的梯度时。在许多其他情况下，抽样实际上是我们的目标，例如我们想训练一个可以从训练分布采样的模型。

17.1.2 蒙特卡罗采样的基础

当无法精确计算和或积分（例如，和具有指数数量个项，且无法被精确简化）时，通常可以使用蒙特卡罗采样来近似它。这种想法把和或者积分视作某分布下的期望，然后通过估计对应的平均值来近似这个期望。令

$$s = \sum_{\boldsymbol{x}} p(\boldsymbol{x}) f(\boldsymbol{x}) = E_p[f(\mathbf{x})] \tag{17.1}$$

或者

$$s = \int p(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x} = E_p[f(\mathbf{x})] \tag{17.2}$$

为我们所需要估计的和或者积分，写成期望的形式，p 是一个关于随机变量 $\mathbf{x}$ 的概率分布（求和时）或者概率密度函数（求积分时）。

我们可以通过从 p 中抽取 n 个样本 $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}$ 来近似 s 并得到一个经验平均值

$$\hat{s}_n = \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}^{(i)}). \tag{17.3}$$

下面几个性质表明了这种近似的合理性。首先很容易观察到 $\hat{s}$ 这个估计是无偏的，由于

$$\mathbb{E}[\hat{s}_n] = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[f(\boldsymbol{x}^{(i)})] = \frac{1}{n} \sum_{i=1}^{n} s = s. \tag{17.4}$$

此外，根据 **大数定理**（Law of large number），如果样本 $\boldsymbol{x}^{(i)}$ 是独立同分布的，那么其平均值几乎必然收敛到期望值，即

$$\lim_{n \to \infty} \hat{s}_n = s, \tag{17.5}$$

只需要满足各个单项的方差 $\mathrm{Var}[f(\boldsymbol{x}^{(i)})]$ 有界。详细地说，我们考虑当 n 增大时 $\hat{s}_n$ 的方差。只要满足 $\mathrm{Var}[f(\mathbf{x}^{(i)})] < \infty$，方差 $\mathrm{Var}[\hat{s}_n]$ 就会减小并收敛到 0：

$$\mathrm{Var}[\hat{s}_n] = \frac{1}{n^2} \sum_{i=1}^{n} \mathrm{Var}[f(\mathbf{x})] \tag{17.6}$$

$$= \frac{\mathrm{Var}[f(\mathbf{x})]}{n}. \tag{17.7}$$

这个简单有用的结果启迪我们如何估计蒙特卡罗均值中的不确定性，或者等价地说是蒙特卡罗估计的期望误差。我们计算了 $f(\boldsymbol{x}^{(i)})$ 的经验均值和方差[1]，然后将估计的方差除以样本数 n 来得到 $\mathrm{Var}[\hat{s}_n]$ 的估计。**中心极限定理**（central limit theorem）告诉我们 $\hat{s}_n$ 的分布收敛到以 s 为均值以 $\frac{\mathrm{Var}[f(\mathbf{x})]}{n}$ 为方差的正态分布。这使得我们可以利用正态分布的累积函数来估计 $\hat{s}_n$ 的置信区间。

以上的所有结论都依赖于我们可以从基准分布 $p(\mathbf{x})$ 中轻易地采样，但是这个假设并不是一直成立的。当我们无法从 p 中采样时，一个备选方案是用第 17.2 节讲到的重要采样。一种更加通用的方式是构建一个收敛到目标分布的估计序列。这就是马尔可夫链蒙特卡罗方法（见第 17.3 节）。

17.2 重要采样

如方程 (17.2)所示，在蒙特卡罗方法中，对积分（或者和）分解，确定积分中哪一部分作为概率分布 $p(\boldsymbol{x})$ 以及哪一部分作为被积的函数 $f(\boldsymbol{x})$（我们感兴趣的是估计 $f(\boldsymbol{x})$ 在概率分布 $p(\boldsymbol{x})$ 下的期望）是很关键的一步。$p(\boldsymbol{x})f(\boldsymbol{x})$ 不存在唯一的分解，因为它总是可以被写成

$$p(\boldsymbol{x})f(\boldsymbol{x}) = q(\boldsymbol{x})\frac{p(\boldsymbol{x})f(\boldsymbol{x})}{q(\boldsymbol{x})}, \tag{17.8}$$

在这里，我们从 q 分布中采样，然后估计 $\frac{pf}{q}$ 在此分布下的均值。许多情况中，我们希望在给定 p 和 f 的情况下计算某个期望，这个问题既然是求期望，那么很自然地 p 和 f 是一种分解选择。然而，如果考虑达到某给定精度所需要的样本数量，这个问题最初的分解选择不是最优的选择。幸运的是，最优的选择 q^* 可以被简单地推导出来。这种最优的采样函数 q^* 对应所谓的最优重要采样。

从式 (17.8) 所示的关系中可以发现，任意蒙特卡罗估计

$$\hat{s}_p = \frac{1}{n}\sum_{i=1,\boldsymbol{x}^{(i)}\sim p}^{n} f(\boldsymbol{x}^{(i)}) \tag{17.9}$$

可以被转化为一个重要采样的估计

$$\hat{s}_q = \frac{1}{n}\sum_{i=1,\boldsymbol{x}^{(i)}\sim q}^{n} \frac{p(\boldsymbol{x}^{(i)})f(\boldsymbol{x}^{(i)})}{q(\boldsymbol{x}^{(i)})}. \tag{17.10}$$

[1] 通常我们会倾向于计算方差的无偏估计，它由偏差的平方和除以 $n-1$ 而非 n 得到。

我们可以容易地发现估计的期望与 q 分布无关：

$$\mathbb{E}_q[\hat{s}_q] = \mathbb{E}_p[\hat{s}_p] = s. \tag{17.11}$$

然而，重要采样的方差可能对 q 的选择非常敏感。这个方差可以表示为

$$\mathrm{Var}[\hat{s}_q] = \mathrm{Var}\left[\frac{p(\mathbf{x})f(\mathbf{x})}{q(\mathbf{x})}\right]/n. \tag{17.12}$$

方差想要取到最小值，q 需要满足

$$q^*(\boldsymbol{x}) = \frac{p(\boldsymbol{x})|f(\boldsymbol{x})|}{Z}, \tag{17.13}$$

在这里 Z 表示归一化常数，选择适当的 Z 使得 $q^*(\boldsymbol{x})$ 之和或者积分为 1。一个更好的重要采样分布会把更多的权重放在被积函数较大的地方。事实上，当 $f(\boldsymbol{x})$ 的正负符号不变时，$\mathrm{Var}[\hat{s}_{q^*}] = 0$，这意味着当使用最优的 q 分布时，只需要一个样本就足够了。当然，这仅仅是因为计算 q^* 时已经解决了原问题。所以在实践中这种只需要采样一个样本的方法往往是无法实现的。

对于重要采样来说任意 q 分布都是可行的（从得到一个期望上正确的值的角度来说），q^* 指的是最优的 q 分布（从得到最小方差的角度上考虑）。从 q^* 中采样往往是不可行的，但是其他仍然能降低方差的 q 的选择还是可行的。

另一种方法是采用 **有偏重要采样**（biased importance sampling），这种方法有一个优势，即不需要归一化的 p 或 q 分布。在处理离散变量时，有偏重要采样估计可以表示为

$$\hat{s}_{\mathrm{BIS}} = \frac{\sum_{i=1}^{n} \frac{p(\boldsymbol{x}^{(i)})}{q(\boldsymbol{x}^{(i)})} f(\boldsymbol{x}^{(i)})}{\sum_{i=1}^{n} \frac{p(\boldsymbol{x}^{(i)})}{q(\boldsymbol{x}^{(i)})}} \tag{17.14}$$

$$= \frac{\sum_{i=1}^{n} \frac{p(\boldsymbol{x}^{(i)})}{\tilde{q}(\boldsymbol{x}^{(i)})} f(\boldsymbol{x}^{(i)})}{\sum_{i=1}^{n} \frac{p(\boldsymbol{x}^{(i)})}{\tilde{q}(\boldsymbol{x}^{(i)})}} \tag{17.15}$$

$$= \frac{\sum_{i=1}^{n} \frac{\tilde{p}(\boldsymbol{x}^{(i)})}{\tilde{q}(\boldsymbol{x}^{(i)})} f(\boldsymbol{x}^{(i)})}{\sum_{i=1}^{n} \frac{\tilde{p}(\boldsymbol{x}^{(i)})}{\tilde{q}(\boldsymbol{x}^{(i)})}}, \tag{17.16}$$

其中 $\tilde{p}$ 和 $\tilde{q}$ 分别是分布 p 和 q 的未经归一化的形式，$\boldsymbol{x}^{(i)}$ 是从分布 q 中抽取的样本。这种估计是有偏的，因为 $\mathbb{E}[\hat{s}_{\mathrm{BIS}}] \neq s$，只有当 $n \to \infty$ 且方程 式 (17.14) 的分母收敛到 1 时，等式才渐近地成立。所以这一估计也被称为渐近无偏的。

一个好的 q 分布的选择可以显著地提高蒙特卡罗估计的效率，而一个糟糕的 q 分布选择则会使效率更糟糕。我们回过头来看看方程 式 (17.12) 会发现，如果存在一个 q 使得 $\frac{p(\boldsymbol{x})f(\boldsymbol{x})}{q(\boldsymbol{x})}$ 很大，那么这个估计的方差也会很大。当 $q(\boldsymbol{x})$ 很小，而 $f(\boldsymbol{x})$ 和 $p(\boldsymbol{x})$ 都较大并且无法抵消 q 时，这种情况会非常明显。q 分布经常会取一些简单常用的分布使得我们能够从 q 分布中容易地采样。当 $\boldsymbol{x}$ 是高维数据时，q 分布的简单性使得它很难与 p 或者 $p|f|$ 相匹配。当 $q(\boldsymbol{x}^{(i)}) \gg p(\boldsymbol{x}^{(i)})|f(\boldsymbol{x}^{(i)})|$ 时，重要采样采到了很多无用的样本（很小的数或零相加）。另一种相对少见的情况是 $q(\boldsymbol{x}^{(i)}) \ll p(\boldsymbol{x}^{(i)})|f(\boldsymbol{x}^{(i)})|$，相应的比值会非常大。正因为后一个事件是很少发生的，这种样本很难被采到，通常使得对 s 的估计出现了典型的欠估计，很难被整体的过估计抵消。这样的不均匀情况在高维数据屡见不鲜，因为在高维度分布中联合分布的动态域可能非常大。

尽管存在上述的风险，但是重要采样及其变种在机器学习的应用中仍然扮演着重要的角色，包括深度学习算法。例如，重要采样被应用于加速训练具有大规模词表的神经网络语言模型的过程中（见第 12.4.3.3 节）或者其他有着大量输出结点的神经网络中。此外，还可以看到重要采样应用于估计配分函数（一个概率分布的归一化常数），详见第 18.7 节，以及在深度有向图模型比如变分自编码器中估计对数似然（详见第 20.10.3 节）。采用随机梯度下降训练模型参数时重要采样可以用来改进对代价函数梯度的估计，尤其是分类器这样的模型，其中代价函数的大部分代价来自于少量错误分类的样本。在这种情况下，更加频繁地抽取这些困难的样本可以减小梯度估计的方差 (Hinton *et al.*, 2006a)。

17.3 马尔可夫链蒙特卡罗方法

在许多实例中，我们希望采用蒙特卡罗方法，然而往往又不存在一种简单的方法可以直接从目标分布 $p_{\text{model}}(\mathbf{x})$ 中精确采样或者一个好的（方差较小的）重要采样分布 $q(\boldsymbol{x})$。在深度学习中，当分布 $p_{\text{model}}(\mathbf{x})$ 表示成无向模型时，这种情况往往会发生。在这种情况下，为了从分布 $p_{\text{model}}(\mathbf{x})$ 中近似采样，我们引入了一种称为 **马尔可夫链**（Markov Chain）的数学工具。利用马尔可夫链来进行蒙特卡罗估计的这一类算法被称为 **马尔可夫链蒙特卡罗**（Markov Chain Monte Carlo, MCMC）方法。Koller and Friedman (2009) 花了大量篇幅来描述马尔可夫链蒙特卡罗算法在机器学习中的应用。MCMC 技术最标准、最一般的理论保证只适用于那些各状态概率均不为零的模型。因此，这些技术最方便的使用方法是用于从 **基于能量的模型**（Energy-based model）即 $p(\boldsymbol{x}) \propto \exp(-E(\boldsymbol{x}))$ 中采样，见第 16.2.4 节。在 EBM 的公式表述中，每

一个状态所对应的概率都不为零。事实上，MCMC 方法可以被广泛地应用在包含 0 概率状态的许多概率分布中。然而，在这种情况下，关于 MCMC 方法性能的理论保证只能依据具体不同类型的分布具体分析证明。在深度学习中，我们通常依赖于那些一般的理论保证，其在所有基于能量的模型都能自然成立。

为了解释从基于能量的模型中采样困难的原因，我们考虑一个包含两个变量的 EBM 的例子，记 $p(\mathrm{a}, \mathrm{b})$ 为其分布。为了采 a，我们必须先从 $p(\mathrm{a} \mid \mathrm{b})$ 中采样；为了采 b，我们又必须从 $p(\mathrm{b} \mid \mathrm{a})$ 中采样。这似乎成了棘手的先有鸡还是先有蛋的问题。有向模型避免了这一问题因为它的图是有向无环的。为了完成 **原始采样**（Ancestral Sampling），在给定每个变量的所有父结点的条件下，我们根据拓扑顺序采样每一个变量，这个变量是确定能够被采样的（详见第 16.3 节）。原始采样定义了一种高效的、单遍的方法来抽取一个样本。

在 EBM 中，我们通过使用马尔可夫链来采样，从而避免了先有鸡还是先有蛋的问题。马尔可夫链的核心思想是从某个可取任意值的状态 $\boldsymbol{x}$ 出发。随着时间的推移，我们随机地反复更新状态 $\boldsymbol{x}$。最终 $\boldsymbol{x}$ 成为了一个从 $p(\boldsymbol{x})$ 中抽出的（非常接近）比较一般的样本。在正式的定义中，马尔可夫链由一个随机状态 x 和一个转移分布 $T(\boldsymbol{x}' \mid \boldsymbol{x})$ 定义而成，$T(\boldsymbol{x}' \mid \boldsymbol{x})$ 是一个概率分布，说明了给定状态 $\boldsymbol{x}$ 的情况下随机地转移到 $\boldsymbol{x}'$ 的概率。运行一个马尔可夫链意味着根据转移分布 $T(\mathbf{x}' \mid \boldsymbol{x})$ 采出的值 $\boldsymbol{x}'$ 来更新状态 $\boldsymbol{x}$。

为了给出 MCMC 方法为何有效的一些理论解释，重参数化这个问题是很有用的。首先我们关注一些简单的情况，其中随机变量 $\mathbf{x}$ 有可数个状态。我们将这种状态简单地记作正整数 x。不同的整数 x 的大小对应着原始问题中 $\boldsymbol{x}$ 的不同状态。

接下来我们考虑如果并行地运行无穷多个马尔可夫链的情况。不同马尔可夫链的所有状态都采样自某一个分布 $q^{(t)}(x)$，在这里 t 表示消耗的时间数。开始时，对每个马尔可夫链，我们采用一个分布 q^0 来任意地初始化 x。之后，$q^{(t)}$ 与所有之前运行的马尔可夫链有关。我们的目标是 $q^{(t)}(x)$ 收敛到 $p(x)$。

因为我们已经用正整数 x 重参数化了这个问题，我们可以用一个向量 $\boldsymbol{v}$ 来描述这个概率分布 q，

$$q(\mathrm{x} = i) = v_i. \tag{17.17}$$

然后我们考虑更新单一的马尔可夫链，从状态 x 到新状态 x'。单一状态转移到

x' 的概率可以表示为

$$q^{(t+1)}(x') = \sum_x q^{(t)}(x)T(x' \mid x). \tag{17.18}$$

根据状态为整数的参数化设定，我们可以将转移算子 T 表示成一个矩阵 $\boldsymbol{A}$。矩阵 $\boldsymbol{A}$ 的定义如下：

$$\boldsymbol{A}_{i,j} = T(\mathbf{x}' = i \mid \mathbf{x} = j). \tag{17.19}$$

使用这一定义，我们可以改写式 (17.18)。不同于之前使用 q 和 T 来理解单个状态的更新，我们现在可以使用 $\boldsymbol{v}$ 和 $\boldsymbol{A}$ 来描述当我们更新时（并行运行的）不同马尔可夫链上整个分布是如何变化的：

$$\boldsymbol{v}^{(t)} = \boldsymbol{A}\boldsymbol{v}^{(t-1)}. \tag{17.20}$$

重复地使用马尔可夫链更新相当于重复地与矩阵 $\boldsymbol{A}$ 相乘。换言之，我们可以认为这一过程就是关于 $\boldsymbol{A}$ 的幂乘：

$$\boldsymbol{v}^{(t)} = \boldsymbol{A}^t\boldsymbol{v}^{(0)}. \tag{17.21}$$

矩阵 $\boldsymbol{A}$ 有一种特殊的结构，因为它的每一列都代表一个概率分布。这样的矩阵被称为 **随机矩阵**（Stochastic Matrix）。如果对于任意状态 x 到任意其他状态 x' 存在一个 t 使得转移概率不为 0，那么 Perron-Frobenius 定理 (Perron, 1907; Frobenius, 1908) 可以保证这个矩阵的最大特征值是实数且大小为 1。我们可以看到所有的特征值随着时间呈现指数变化：

$$\boldsymbol{v}^{(t)} = (\boldsymbol{V}\mathrm{diag}(\boldsymbol{\lambda})\,\boldsymbol{V}^{-1})^t\boldsymbol{v}^{(0)} = \boldsymbol{V}\mathrm{diag}(\boldsymbol{\lambda})^t\,\boldsymbol{V}^{-1}\boldsymbol{v}^{(0)}. \tag{17.22}$$

这个过程导致了所有不等于 1 的特征值都衰减到 0。在一些额外的较为宽松的假设下，我们可以保证矩阵 $\boldsymbol{A}$ 只有一个对应特征值为 1 的特征向量。所以这个过程收敛到 **平稳分布**（Stationary Distribution），有时也被称为 **均衡分布**（Equilibrium Distribution）。收敛时，我们得到

$$\boldsymbol{v}' = \boldsymbol{A}\boldsymbol{v} = \boldsymbol{v}, \tag{17.23}$$

这个条件也适用于收敛之后的每一步。这就是特征向量方程。作为收敛的稳定点，$\boldsymbol{v}$ 一定是特征值为 1 所对应的特征向量。这个条件保证收敛到了平稳分布以后，再重

复转移采样过程不会改变所有不同马尔可夫链上状态的分布（尽管转移算子自然而然地会改变每个单独的状态）。

如果我们正确地选择了转移算子 T，那么最终的平稳分布q 将会等于我们所希望采样的分布 p。我们会将第 17.4 节介绍如何选择 T。

可数状态马尔可夫链的大多数性质可以被推广到连续状态的马尔可夫链中。在这种情况下，一些研究者把这种马尔可夫链称为 **哈里斯链**（Harris Chain），但是我们将这两种情况都称为马尔可夫链。通常在一些宽松的条件下，一个带有转移算子 T 的马尔可夫链都会收敛到一个不动点，这个不动点可以写成如下形式：

$$q'(\mathbf{x}') = \mathbb{E}_{\mathbf{x}\sim q} T(\mathbf{x}' \mid \mathbf{x}), \tag{17.24}$$

这个方程的离散版本就相当于重新改写方程 式 (17.23)。当 $\mathbf{x}$ 是离散值时，这个期望对应着求和，而当 $\mathbf{x}$ 是连续值时，这个期望对应的是积分。

无论状态是连续的还是离散的，所有的马尔可夫链方法都包括了重复、随机地更新直到最后状态开始从均衡分布中采样。运行马尔可夫链直到它达到均衡分布的过程通常被称为马尔可夫链的 **磨合**（Burning-in）过程。在马尔可夫链达到均衡分布之后，我们可以从均衡分布中抽取一个无限多数量的样本序列。这些样本服从同一分布，但是两个连续的样本之间会高度相关。所以一个有限的序列无法完全表达均衡分布。一种解决这个问题的方法是每隔 n 个样本返回一个样本，从而使得我们对于均衡分布的统计量的估计不会被 MCMC 方法的样本之间的相关性所干扰。所以马尔可夫链的计算代价很高，主要源于达到均衡分布前需要磨合的时间以及在达到均衡分布之后从一个样本转移到另一个足够无关的样本所需要的时间。如果我们想要得到完全独立的样本，那么我们可以同时并行地运行多个马尔可夫链。这种方法使用了额外的并行计算来减少时延。使用一条马尔可夫链来生成所有样本的策略和（使用多条马尔可夫链）每条马尔可夫链只产生一个样本的策略是两种极端。深度学习的从业者们通常选取的马尔可夫链的数目和小批量中的样本数相近，然后从这些固定的马尔可夫链集合中抽取所需要的样本。马尔可夫链的数目通常选为 100。

另一个难点是我们无法预先知道马尔可夫链需要运行多少步才能到达均衡分布。这段时间通常被称为 **混合时间**（Mixing Time）。检测一个马尔可夫链是否达到平衡是很困难的。我们并没有足够完善的理论来解决这个问题。理论只能保证马尔可夫链会最终收敛，但是无法保证其他。如果我们从矩阵 $\boldsymbol{A}$ 作用在概率向量 $\boldsymbol{v}$ 上的角度来分析马尔可夫链，那么我们可以发现当 $\boldsymbol{A}^t$ 除了单个 1 以外的特征值都趋于 0 时，马尔可夫链混合成功（收敛到了均衡分布）。这也意味着矩阵 $\boldsymbol{A}$ 的第二大特征值决

定了马尔可夫链的混合时间。然而，在实践中，我们通常不能真的将马尔可夫链表示成矩阵的形式。我们的概率模型所能够达到的状态是变量数的指数级别，所以表达 $\boldsymbol{v}$，$\boldsymbol{A}$ 或者 $\boldsymbol{A}$ 的特征值是不现实的。由于以上在内的诸多阻碍，我们通常无法知道马尔可夫链是否已经混合成功。作为替代，我们只能运行一定量时间马尔可夫链直到我们粗略估计这段时间是足够的，然后使用启发式的方法来判断马尔可夫链是否混合成功。这些启发性的算法包括了手动检查样本或者衡量前后样本之间的相关性。

17.4 Gibbs 采样

目前为止我们已经了解了如何通过反复更新 $\boldsymbol{x} \leftarrow \boldsymbol{x}' \sim T(\boldsymbol{x}' \mid \boldsymbol{x})$ 从一个分布 $q(\boldsymbol{x})$ 中采样。然而我们还没有介绍过如何确定 $q(\boldsymbol{x})$ 是否是一个有效的分布。本书中将会描述两种基本的方法。第一种方法是从已经学习到的分布 p_{model} 中推导出 T，下文描述了如何从基于能量的模型中采样。第二种方法是直接用参数描述 T，然后学习这些参数，其平稳分布隐式地定义了我们所感兴趣的模型 p_{model}。我们将在第 20.12 节和第 20.13 节中讨论第二种方法的例子。

在深度学习中，我们通常使用马尔可夫链从定义为基于能量的模型的分布 $p_{\text{model}}(\boldsymbol{x})$ 中采样。在这种情况下，我们希望马尔可夫链的 $q(\boldsymbol{x})$ 分布就是 $p_{\text{model}}(\boldsymbol{x})$。为了得到所期望的 $q(\boldsymbol{x})$ 分布，我们必须选取合适的 $T(\boldsymbol{x}' \mid \boldsymbol{x})$。

Gibbs 采样（Gibbs Sampling）是一种概念简单而又有效的方法。它构造一个从 $p_{\text{model}}(\boldsymbol{x})$ 中采样的马尔可夫链，其中在基于能量的模型中从 $T(\mathbf{x}' \mid \mathbf{x})$ 采样是通过选择一个变量 x_i，然后从 p_{model} 中该点关于在无向图 $\mathcal{G}$（定义了基于能量的模型结构）中邻接点的条件分布中采样。只要一些变量在给定相邻变量时是条件独立的，那么这些变量就可以被同时采样。正如在第 16.7.1 节中看到的 RBM 示例一样，RBM 中所有的隐藏单元可以被同时采样，因为在给定所有可见单元的条件下它们相互条件独立。同样地，所有的可见单元也可以被同时采样，因为在给定所有隐藏单元的情况下它们相互条件独立。以这种方式同时更新许多变量的 Gibbs 采样通常被称为 **块吉布斯采样**（block Gibbs Sampling）。

设计从 p_{model} 中采样的马尔可夫链还存在其他备选方法。比如说，Metropolis-Hastings 算法在其他领域中广泛使用。不过在深度学习的无向模型中，我们主要使用 Gibbs 采样，很少使用其他方法。改进采样技巧也是一个潜在的研究热点。

17.5 不同的峰值之间的混合挑战

使用MCMC方法的主要难点在于马尔可夫链的**混合**（Mixing）通常不理想。在理想情况下，从设计好的马尔可夫链中采出的连续样本之间是完全独立的，而且在 $\boldsymbol{x}$ 空间中，马尔可夫链会按概率大小访问许多不同区域。

然而，MCMC方法采出的样本可能会具有很强的相关性，尤其是在高维的情况下。我们把这种现象称为慢混合甚至混合失败。具有缓慢混合的MCMC方法可以被视为对能量函数无意地执行类似于带噪声的梯度下降的操作，或者说等价于相对于链的状态（被采样的随机变量）依据概率进行噪声爬坡。（在马尔可夫链的状态空间中）从 $\boldsymbol{x}^{(t-1)}$ 到 $\boldsymbol{x}^{(t)}$ 该链倾向于选取很小的步长，其中能量 $E(\boldsymbol{x}^{(t)})$ 通常低于或者近似等于能量 $E(\boldsymbol{x}^{(t-1)})$，倾向于向较低能量的区域移动。当从可能性较小的状态（比来自 $p(\boldsymbol{x})$ 的典型样本拥有更高的能量）开始时，链趋向于逐渐减少状态的能量，并且仅仅偶尔移动到另一个峰值。一旦该链已经找到低能量的区域（例如，如果变量是图像中的像素，则低能量的区域可以是同一对象所对应图像的一个连通的流形），我们称之为峰值，链将倾向于围绕着这个峰值游走（按某一种形式随机游走）。它时不时会走出该峰值，但是结果通常会返回该峰值或者（如果找到一条离开的路线）移向另一个峰值。问题是对于很多有趣的分布来说成功的离开路线很少，所以马尔可夫链将在一个峰值附近抽取远超过需求的样本。

当我们考虑 Gibbs 采样算法（见第17.4节）时，这种现象格外明显。在这种情况下，我们考虑在一定步数内从一个峰值移动到一个临近峰值的概率。决定这个概率的是两个峰值之间的“能量障碍”的形状。隔着一个巨大“能量障碍”（低概率的区域）的两个峰值之间的转移概率是（随着能量障碍的高度）指数下降的，如图17.1所示。当目标分布有多个高概率峰值并且被低概率区域所分割，尤其当Gibbs采样的每一步都只是更新变量的一小部分而这一小部分变量又严重依赖其他的变量时，就会产生问题。

举一个简单的例子，考虑两个变量 a，b 的基于能量的模型，这两个变量都是二值的，取值 $+1$ 或者 -1。如果对某个较大的正数 w，$E(\mathrm{a},\mathrm{b}) = -w\mathrm{ab}$，那么这个模型传达了一个强烈的信息，a 和 b 有相同的符号。当 $\mathrm{a} = 1$ 时用 Gibbs 采样更新 b。给定 b 时的条件分布满足 $p(\mathrm{b} = 1 \mid \mathrm{a} = 1) = \sigma(w)$。如果 w 的值很大，sigmoid 函数趋近于饱和，那么 b 也取到 1 的概率趋近于 1。同理，如果 $\mathrm{a} = -1$，那么 b 取到 -1 的概率也趋于 1。根据模型 $p_{\mathrm{model}}(\mathrm{a},\mathrm{b})$，两个变量取一样的符号的概率几乎相等。根据 $p_{\mathrm{model}}(\mathrm{a} \mid \mathrm{b})$，两个变量应该有相同的符号。这也意味着 Gibbs 采样很难会

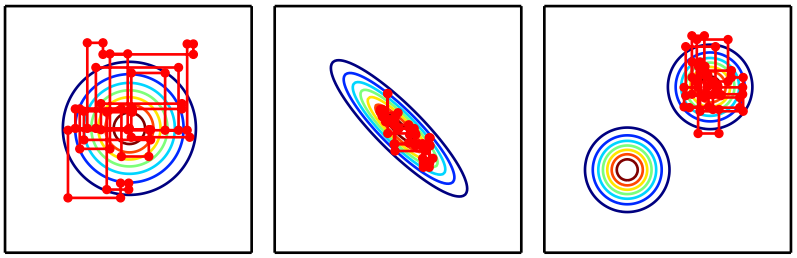

图 17.1: 对于三种分布使用 Gibbs 采样所产生的路径，所有的分布马尔可夫链初始值都设为峰值。*(左)* 一个带有两个独立变量的多维正态分布。由于变量之间是相互独立的，Gibbs 采样混合得很好。*(中)* 变量之间存在高度相关性的一个多维正态分布。变量之间的相关性使得马尔可夫链很难混合。因为每一个变量的更新需要相对其他变量求条件分布，相关性减慢了马尔可夫链远离初始点的速度。*(右)* 峰值之间间距很大且不在轴上对齐的混合高斯分布。Gibbs 采样混合得很慢，因为每次更新仅仅一个变量很难跨越不同的峰值。

改变这些变量的符号。

在更实际的问题中，这种挑战更加艰巨因为在实际问题中我们不能仅仅关注在两个峰值之间的转移，更要关注在多个峰值之间的转移。如果由于峰值之间混合困难，而导致某几个这样的转移难以完成，那么得到一些可靠的覆盖大部分峰值的样本集合的计算代价是很高的，同时马尔可夫链收敛到它的平稳分布的过程也会非常缓慢。

通过寻找一些高度依赖变量的组以及分块同时更新块（组）中的变量，这个问题有时候是可以被解决的。然而不幸的是，当依赖关系很复杂时，从这些组中采样的过程从计算角度上说是难以处理的。归根结底，马尔可夫链最初就是被提出来解决这个问题，即从大量变量中采样的问题。

在定义了一个联合分布 $p_{\text{model}}(\boldsymbol{x}, \boldsymbol{h})$ 的潜变量模型中，我们经常通过交替地从 $p_{\text{model}}(\boldsymbol{x} \mid \boldsymbol{h})$ 和 $p_{\text{model}}(\boldsymbol{h} \mid \boldsymbol{x})$ 中采样来达到抽 $\boldsymbol{x}$ 的目的。从快速混合的角度上说，我们更希望 $p_{\text{model}}(\boldsymbol{h} \mid \boldsymbol{x})$ 有很大的熵。然而，从学习一个 $\boldsymbol{h}$ 的有用表示的角度上考虑，我们还是希望 $\boldsymbol{h}$ 能够包含 $\boldsymbol{x}$ 的足够信息从而能够较完整地重构它，这意味 $\boldsymbol{h}$ 和 $\boldsymbol{x}$ 要有非常高的互信息。这两个目标是相互矛盾的。我们经常学习到能够将 $\boldsymbol{x}$ 精确地

编码为 $\boldsymbol{h}$ 的生成模型，但是无法很好混合。这种情况在玻尔兹曼机中经常出现，一个玻尔兹曼机学到的分布越尖锐，该分布的马尔可夫链采样越难混合得好。这个问题在图 17.2 中有所描述。

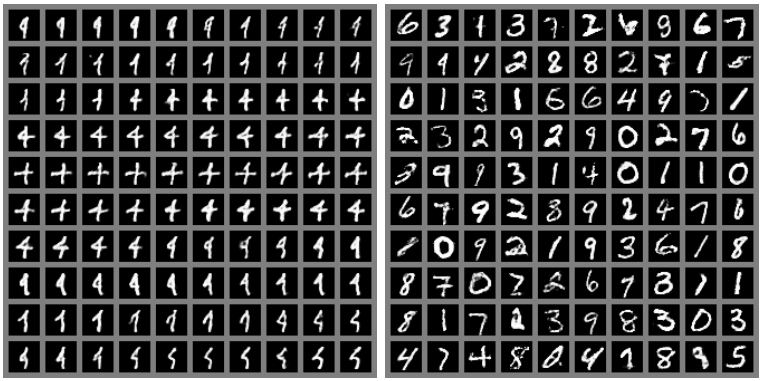

图 17.2: 深度概率模型中一个混合缓慢问题的例证。每张图都是按照从左到右从上到下的顺序的。*(左)* Gibbs 采样从 MNIST 数据集训练成的深度玻尔兹曼机中采出的连续样本。这些连续的样本之间非常相似。由于 Gibbs 采样作用于一个深度图模型，相似度更多地是基于语义而非原始视觉特征。但是对于吉布斯链来说从分布的一个峰值转移到另一个仍然是很困难的，比如说改变数字。*(右)* 从生成式对抗网络中抽出的连续原始样本。因为原始采样生成的样本之间互相独立，所以不存在混合问题。

当感兴趣的分布对于每个类具有单独的流形结构时，所有这些问题都使 MCMC 方法变得不那么有用：分布集中在许多峰值周围，并且这些峰值由大量高能量区域分割。我们在许多分类问题中遇到的是这种类型的分布，由于峰值之间混合缓慢，它将使得 MCMC 方法非常缓慢地收敛。

17.5.1 不同峰值之间通过回火来混合

当一个分布有一些陡峭的峰并且被低概率区域包围时，很难在分布的不同峰值之间混合。一些加速混合的方法是基于构造一个概率分布替代目标分布，这个概率分布的峰值没有那么高，峰值周围的低谷也没有那么低。基于能量的模型为这个想法提供一种简单的做法。目前为止，我们一直将基于能量的模型描述为定义一个概率分布：

$$p(\boldsymbol{x}) \propto \exp(-E(\boldsymbol{x})). \tag{17.25}$$

基于能量的模型可以通过添加一个额外的控制峰值尖锐程度的参数 β 来加强：

$$p_\beta(\boldsymbol{x}) \propto \exp(-\beta E(\boldsymbol{x})). \tag{17.26}$$

β 参数可以被理解为 **温度**（temperature）的倒数，反映了基于能量的模型的统计物理学起源。当温度趋近于 0 时，β 趋近于无穷大，此时的基于能量的模型是确定性的。当温度趋近于无穷大时，β 趋近于零，基于能量的模型（对离散的 $\boldsymbol{x}$）成了均匀分布。

通常情况下，在 $\beta = 1$ 时训练一个模型。但我们也可以利用其他温度，尤其是 $\beta < 1$ 的情况。**回火**（tempering）作为一种通用的策略，它通过从 $\beta < 1$ 模型中采样来实现在 p_1 的不同峰值之间快速混合。

基于 **回火转移**（tempered transition） (Neal, 1994) 的马尔可夫链临时从高温度的分布中采样使其在不同峰值之间混合，然后继续从单位温度的分布中采样。这些技巧被应用在一些模型比如RBM中 (Salakhutdinov, 2010)。另一种方法是利用 **并行回火**（parallel tempering） (Iba, 2001)。其中马尔可夫链并行地模拟许多不同温度的不同状态。最高温度的状态混合较慢，相比之下最低温度的状态，即温度为 1 时，采出了精确的样本。转移算子包括了两个温度之间的随机跳转，所以一个高温度状态分布槽中的样本有足够大的概率跳转到低温度分布的槽中。这个方法也被应用到了RBM中 (Desjardins *et al.*, 2010; Cho *et al.*, 2010a)。尽管回火这种方法前景可期，现今它仍然无法让我们在采样复杂的基于能量的模型中更进一步。一个可能的原因是在 **临界温度**（critical temperatures）时温度转移算子必须设置得非常慢（因为温度需要逐渐下降）来确保回火的有效性。

17.5.2 深度也许会有助于混合

当我们从潜变量模型 $p(\boldsymbol{h}, \boldsymbol{x})$ 中采样时，我们可以发现如果 $p(\boldsymbol{h} \mid \boldsymbol{x})$ 将 $\boldsymbol{x}$ 编码得非常好，那么从 $p(\boldsymbol{x} \mid \boldsymbol{h})$ 中采样时，并不会太大地改变 $\boldsymbol{x}$，那么混合结果会很糟糕。解决这个问题的一种方法是使得 $\boldsymbol{h}$ 成为一种将 $\boldsymbol{x}$ 编码为 $\boldsymbol{h}$ 的深度表示，从而使得马尔可夫链在 $\boldsymbol{h}$ 空间中更容易混合。在许多表示学习算法如自编码器和RBM中，$\boldsymbol{h}$ 的边缘分布相比于 $\boldsymbol{x}$ 上的原始数据分布，通常表现为更加均匀、更趋近于单峰值。或许可以说，这是因为利用了所有可用的表示空间并尽量减小重构误差。因为当训练集上的不同样本之间在 $\boldsymbol{h}$ 空间能够被非常容易地区分时，我们也会很容易地最小化重构误差。Bengio *et al.* (2013a) 观察到这样的现象，堆叠越深的正则化自编码