第三章　概率与信息论

本章我们讨论概率论和信息论。

概率论是用于表示不确定性声明的数学框架。它不仅提供了量化不确定性的方法，也提供了用于导出新的不确定性**声明**（statement）的公理。在人工智能领域，概率论主要有两种用途。首先，概率法则告诉我们 AI 系统如何推理，据此我们设计一些算法来计算或者估算由概率论导出的表达式。其次，我们可以用概率和统计从理论上分析我们提出的 AI 系统的行为。

概率论是众多科学学科和工程学科的基本工具。我们提供这一章，是为了确保那些背景偏软件工程而较少接触概率论的读者也可以理解本书的内容。

概率论使我们能够提出不确定的声明以及在不确定性存在的情况下进行推理，而信息论使我们能够量化概率分布中的不确定性总量。

如果你已经对概率论和信息论很熟悉了，那么除了第 3.14 节以外的整章内容，你都可以跳过。而在第 3.14 节中，我们会介绍用来描述机器学习中结构化概率模型的图。即使你对这些主题没有任何的先验知识，本章对于完成深度学习的研究项目来说也已经足够，尽管如此我们还是建议你能够参考一些额外的资料，例如 Jaynes (2003)。

3.1　为什么要使用概率？

计算机科学的许多分支处理的实体大部分都是完全确定且必然的。程序员通常可以安全地假定 CPU 将完美地执行每条机器指令。虽然硬件错误确实会发生，但它们足够罕见，以致于大部分软件应用在设计时并不需要考虑这些因素的影响。鉴于许多计算机科学家和软件工程师在一个相对干净和确定的环境中工作，机器学习对

于概率论的大量使用是很令人吃惊的。

这是因为机器学习通常必须处理不确定量，有时也可能需要处理随机 (非确定性的) 量。不确定性和随机性可能来自多个方面。至少从 20 世纪 80 年代开始，研究人员就对使用概率论来量化不确定性提出了令人信服的论据。这里给出的许多论据都是根据 Pearl (1988) 的工作总结或启发得到的。

几乎所有的活动都需要一些在不确定性存在的情况下进行推理的能力。事实上，除了那些被定义为真的数学声明，我们很难认定某个命题是千真万确的或者确保某件事一定会发生。

不确定性有三种可能的来源：

1. 被建模系统内在的随机性。例如，大多数量子力学的解释，都将亚原子粒子的动力学描述为概率的。我们还可以创建一些我们假设具有随机动态的理论情境，例如一个假想的纸牌游戏，在这个游戏中我们假设纸牌被真正混洗成了随机顺序。

2. 不完全观测。即使是确定的系统，当我们不能观测到所有驱动系统行为的变量时，该系统也会呈现随机性。例如，在 Monty Hall 问题中，一个游戏节目的参与者被要求在三个门之间选择，并且会赢得放置在选中门后的奖品。其中两扇门通向山羊，第三扇门通向一辆汽车。选手的每个选择所导致的结果是确定的，但是站在选手的角度，结果是不确定的。

3. 不完全建模。当我们使用一些必须舍弃某些观测信息的模型时，舍弃的信息会导致模型的预测出现不确定性。例如，假设我们制作了一个机器人，它可以准确地观察周围每一个对象的位置。在对这些对象将来的位置进行预测时，如果机器人采用的是离散化的空间，那么离散化的方法将使得机器人无法确定对象们的精确位置：因为每个对象都可能处于它被观测到的离散单元的任何一个角落。

在很多情况下，使用一些简单而不确定的规则要比复杂而确定的规则更为实用，即使真正的规则是确定的并且我们建模的系统可以足够精确地容纳复杂的规则。例如，“多数鸟儿都会飞” 这个简单的规则描述起来很简单很并且使用广泛，而正式的规则——“除了那些还没学会飞翔的幼鸟，因为生病或是受伤而失去了飞翔能力的鸟，包括食火鸟 (cassowary)、鸵鸟 (ostrich)、几维 (kiwi，一种新西兰产的无翼鸟)

等不会飞的鸟类……以外，鸟儿会飞”，很难应用、维护和沟通，即使经过这么多的努力，这个规则还是很脆弱而且容易失效。

尽管我们的确需要一种用以对不确定性进行表示和推理的方法，但是概率论并不能明显地提供我们在人工智能领域需要的所有工具。概率论最初的发展是为了分析事件发生的频率。我们可以很容易地看出概率论，对于像在扑克牌游戏中抽出一手特定的牌这种事件的研究中，是如何使用的。这类事件往往是可以重复的。当我们说一个结果发生的概率为 p，这意味着如果我们反复实验 (例如，抽取一手牌) 无限次，有 p 的比例可能会导致这样的结果。这种推理似乎并不立即适用于那些不可重复的命题。如果一个医生诊断了病人，并说该病人患流感的几率为 40%，这意味着非常不同的事情——我们既不能让病人有无穷多的副本，也没有任何理由去相信病人的不同副本在具有不同的潜在条件下表现出相同的症状。在医生诊断病人的例子中，我们用概率来表示一种 **信任度**（degree of belief），其中 1 表示非常肯定病人患有流感，而 0 表示非常肯定病人没有流感。前面那种概率，直接与事件发生的频率相联系，被称为 **频率派概率**（frequentist probability）；而后者，涉及到确定性水平，被称为 **贝叶斯概率**（Bayesian probability）。

关于不确定性的常识推理，如果我们已经列出了若干条我们期望它具有的性质，那么满足这些性质的唯一一种方法就是将贝叶斯概率和频率派概率视为等同的。例如，如果我们要在扑克牌游戏中根据玩家手上的牌计算她能够获胜的概率，我们使用和医生情境完全相同的公式，就是我们依据病人的某些症状计算她是否患病的概率。为什么一小组常识性假设蕴含了必须是相同的公理控制两种概率？更多的细节参见 Ramsey (1926)。

概率可以被看作是用于处理不确定性的逻辑扩展。逻辑提供了一套形式化的规则，可以在给定某些命题是真或假的假设下，判断另外一些命题是真的还是假的。概率论提供了一套形式化的规则，可以在给定一些命题的似然后，计算其他命题为真的似然。

3.2 随机变量

随机变量（random variable）是可以随机地取不同值的变量。我们通常用无格式字体 (plain typeface) 中的小写字母来表示随机变量本身，而用手写体中的小写字母来表示随机变量能够取到的值。例如，x_1 和 x_2 都是随机变量 x 可能的取值。对

于向量值变量，我们会将随机变量写成 $\mathbf{x}$，它的一个可能取值为 $\boldsymbol{x}$。就其本身而言，一个随机变量只是对可能的状态的描述；它必须伴随着一个概率分布来指定每个状态的可能性。

随机变量可以是离散的或者连续的。离散随机变量拥有有限或者可数无限多的状态。注意这些状态不一定非要是整数；它们也可能只是一些被命名的状态而没有数值。连续随机变量伴随着实数值。

3.3 概率分布

概率分布（probability distribution）用来描述随机变量或一簇随机变量在每一个可能取到的状态的可能性大小。我们描述概率分布的方式取决于随机变量是离散的还是连续的。

3.3.1 离散型变量和概率质量函数

离散型变量的概率分布可以用**概率质量函数**（probability mass function, PMF）[1]来描述。我们通常用大写字母 P 来表示概率质量函数。通常每一个随机变量都会有一个不同的概率质量函数，并且读者必须根据随机变量来推断所使用的 PMF，而不是根据函数的名称来推断；例如，$P(\mathrm{x})$ 通常和 $P(\mathrm{y})$ 不一样。

概率质量函数将随机变量能够取得的每个状态映射到随机变量取得该状态的概率。$\mathrm{x} = x$ 的概率用 $P(x)$ 来表示，概率为 1 表示 $\mathrm{x} = x$ 是确定的，概率为 0 表示 $\mathrm{x} = x$ 是不可能发生的。有时为了使得PMF的使用不相互混淆，我们会明确写出随机变量的名称：$P(\mathrm{x} = x)$。有时我们会先定义一个随机变量，然后用 $\sim$ 符号来说明它遵循的分布：$\mathrm{x} \sim P(\mathrm{x})$。

概率质量函数可以同时作用于多个随机变量。这种多个变量的概率分布被称为**联合概率分布**（joint probability distribution）。$P(\mathrm{x} = x, \mathrm{y} = y)$ 表示 $\mathrm{x} = x$ 和 $\mathrm{y} = y$ 同时发生的概率。我们也可以简写为 $P(x, y)$。

如果一个函数 P 是随机变量 x 的 PMF，必须满足下面这几个条件：

- P 的定义域必须是 x 所有可能状态的集合。

[1] 译者注：国内有些教材也将它翻译成概率分布律。

- $\forall x \in \mathrm{x}, 0 \leq P(x) \leq 1$. 不可能发生的事件概率为 0，并且不存在比这概率更低的状态。类似的，能够确保一定发生的事件概率为 1，而且不存在比这概率更高的状态。
- $\sum_{x \in \mathrm{x}} P(x) = 1$. 我们把这条性质称之为**归一化的**（normalized）。如果没有这条性质，当我们计算很多事件其中之一发生的概率时可能会得到大于 1 的概率。

例如，考虑一个离散型随机变量 x 有 k 个不同的状态。我们可以假设 x 是**均匀分布**（uniform distribution）的（也就是将它的每个状态视为等可能的），通过将它的PMF设为

$$P(\mathrm{x} = x_i) = \frac{1}{k} \tag{3.1}$$

对于所有的 i 都成立。我们可以看出这满足上述成为概率质量函数的条件。因为 k 是一个正整数，所以 $\frac{1}{k}$ 是正的。我们也可以看出

$$\sum_i P(\mathrm{x} = x_i) = \sum_i \frac{1}{k} = \frac{k}{k} = 1, \tag{3.2}$$

因此分布也满足归一化条件。

3.3.2 连续型变量和概率密度函数

当我们研究的对象是连续型随机变量时，我们用**概率密度函数**（probability density function, PDF）而不是概率质量函数来描述它的概率分布。如果一个函数 p 是概率密度函数，必须满足下面这几个条件：

- p 的定义域必须是 x 所有可能状态的集合。
- $\forall x \in \mathrm{x}, p(x) \geq 0$. 注意，我们并不要求 $p(x) \leq 1$。
- $\int p(x)dx = 1$.

概率密度函数 $p(x)$ 并没有直接对特定的状态给出概率，相对的，它给出了落在面积为 δx 的无限小的区域内的概率为 $p(x)\delta x$。

我们可以对概率密度函数求积分来获得点集的真实概率质量。特别地，x 落在集合 $\mathbb{S}$ 中的概率可以通过 $p(x)$ 对这个集合求积分来得到。在单变量的例子中，x 落在区间 $[a, b]$ 的概率是 $\int_{[a,b]} p(x)dx$。

为了给出一个连续型随机变量的 PDF 的例子，我们可以考虑实数区间上的均匀分布。我们可以使用函数 $u(x; a, b)$，其中 a 和 b 是区间的端点且满足 $b > a$。符号 ";" 表示 "以什么为参数"；我们把 x 作为函数的自变量，a 和 b 作为定义函数的参数。为了确保区间外没有概率，我们对所有的 $x \notin [a, b]$，令 $u(x; a, b) = 0$。在 $[a, b]$ 内，有 $u(x; a, b) = \frac{1}{b-a}$。我们可以看出任何一点都非负。另外，它的积分为 1。我们通常用 $\mathrm{x} \sim U(a, b)$ 表示 x 在 $[a, b]$ 上是均匀分布的。

3.4 边缘概率

有时候，我们知道了一组变量的联合概率分布，但想要了解其中一个子集的概率分布。这种定义在子集上的概率分布被称为 **边缘概率分布**（marginal probability distribution）。

例如，假设有离散型随机变量 x 和 y，并且我们知道 $P(\mathrm{x}, \mathrm{y})$。我们可以依据下面的 **求和法则**（sum rule）来计算 $P(\mathrm{x})$：

$$\forall x \in \mathrm{x}, P(\mathrm{x} = x) = \sum_{y} P(\mathrm{x} = x, \mathrm{y} = y). \tag{3.3}$$

"边缘概率" 的名称来源于手算边缘概率的计算过程。当 $P(\mathrm{x}, \mathrm{y})$ 的每个值被写在由每行表示不同的 x 值，每列表示不同的 y 值形成的网格中时，对网格中的每行求和是很自然的事情，然后将求和的结果 $P(x)$ 写在每行右边的纸的边缘处。

对于连续型变量，我们需要用积分替代求和：

$$p(x) = \int p(x, y) dy. \tag{3.4}$$

3.5 条件概率

在很多情况下，我们感兴趣的是某个事件，在给定其他事件发生时出现的概率。这种概率叫做条件概率。我们将给定 $\mathrm{x} = x$，$\mathrm{y} = y$ 发生的条件概率记为 $P(\mathrm{y} = y \mid \mathrm{x} = x)$。这个条件概率可以通过下面的公式计算：

$$P(\mathrm{y} = y \mid \mathrm{x} = x) = \frac{P(\mathrm{y} = y, \mathrm{x} = x)}{P(\mathrm{x} = x)}. \tag{3.5}$$

条件概率只在 $P(\mathrm{x} = x) > 0$ 时有定义。我们不能计算给定在永远不会发生的事件上的条件概率。

这里需要注意的是，不要把条件概率和计算当采用某个动作后会发生什么相混淆。假定某个人说德语，那么他是德国人的条件概率是非常高的，但是如果随机选择的一个人会说德语，他的国籍不会因此而改变。计算一个行动的后果被称为 **干预查询**（intervention query）。干预查询属于 **因果模型**（causal modeling）的范畴，我们不会在本书中讨论。

3.6 条件概率的链式法则

任何多维随机变量的联合概率分布，都可以分解成只有一个变量的条件概率相乘的形式：

$$P(\mathrm{x}^{(1)}, \ldots, \mathrm{x}^{(n)}) = P(\mathrm{x}^{(1)}) \Pi_{i=2}^{n} P(\mathrm{x}^{(i)} \mid \mathrm{x}^{(1)}, \ldots, \mathrm{x}^{(i-1)}). \tag{3.6}$$

这个规则被称为概率的 **链式法则**（chain rule）或者 **乘法法则**（product rule）。它可以直接从式 (3.5) 条件概率的定义中得到。例如，使用两次定义可以得到

$$\begin{aligned} P(\mathrm{a}, \mathrm{b}, \mathrm{c}) &= P(\mathrm{a} \mid \mathrm{b}, \mathrm{c}) P(\mathrm{b}, \mathrm{c}) \\ P(\mathrm{b}, \mathrm{c}) &= P(\mathrm{b} \mid \mathrm{c}) P(\mathrm{c}) \\ P(\mathrm{a}, \mathrm{b}, \mathrm{c}) &= P(\mathrm{a} \mid \mathrm{b}, \mathrm{c}) P(\mathrm{b} \mid \mathrm{c}) P(\mathrm{c}). \end{aligned}$$

3.7 独立性和条件独立性

两个随机变量 x 和 y，如果它们的概率分布可以表示成两个因子的乘积形式，并且一个因子只包含 x 另一个因子只包含 y，我们就称这两个随机变量是 **相互独立的**（independent）：

$$\forall x \in \mathrm{x}, y \in \mathrm{y}, p(\mathrm{x} = x, \mathrm{y} = y) = p(\mathrm{x} = x) p(\mathrm{y} = y). \tag{3.7}$$

如果关于 x 和 y 的条件概率分布对于 z 的每一个值都可以写成乘积的形式，那么这两个随机变量 x 和 y 在给定随机变量 z 时是 **条件独立的**（conditionally

independent）：

$$\forall x \in \mathrm{x}, y \in \mathrm{y}, z \in \mathrm{z}, p(\mathrm{x} = x, \mathrm{y} = y \mid \mathrm{z} = z) = p(\mathrm{x} = x \mid \mathrm{z} = z)p(\mathrm{y} = y \mid \mathrm{z} = z). \tag{3.8}$$

我们可以采用一种简化形式来表示独立性和条件独立性：$\mathrm{x} \perp \mathrm{y}$ 表示 x 和 y 相互独立，$\mathrm{x} \perp \mathrm{y} \mid \mathrm{z}$ 表示 x 和 y 在给定 z 时条件独立。

3.8 期望、方差和协方差

函数 $f(x)$ 关于某分布 $P(\mathrm{x})$ 的**期望**（expectation）或者**期望值**（expected value）是指，当 x 由 P 产生，f 作用于 x 时，$f(x)$ 的平均值。对于离散型随机变量，这可以通过求和得到：

$$\mathbb{E}_{\mathrm{x}\sim P}[f(x)] = \sum_x P(x)f(x), \tag{3.9}$$

对于连续型随机变量可以通过求积分得到：

$$\mathbb{E}_{\mathrm{x}\sim p}[f(x)] = \int p(x)f(x)dx. \tag{3.10}$$

当概率分布在上下文中指明时，我们可以只写出期望作用的随机变量的名称来进行简化，例如 $\mathbb{E}_{\mathrm{x}}[f(x)]$。如果期望作用的随机变量也很明确，我们可以完全不写脚标，就像 $\mathbb{E}[f(x)]$。默认地，我们假设 $\mathbb{E}[\cdot]$ 表示对方括号内的所有随机变量的值求平均。类似的，当没有歧义时，我们还可以省略方括号。

期望是线性的，例如，

$$\mathbb{E}_{\mathrm{x}}[\alpha f(x) + \beta g(x)] = \alpha\mathbb{E}_{\mathrm{x}}[f(x)] + \beta\mathbb{E}_{\mathrm{x}}[g(x)], \tag{3.11}$$

其中 α 和 β 不依赖于 x。

方差（variance）衡量的是当我们对 x 依据它的概率分布进行采样时，随机变量 x 的函数值会呈现多大的差异：

$$\mathrm{Var}(f(x)) = \mathbb{E}\left[(f(x) - \mathbb{E}[f(x)])^2\right]. \tag{3.12}$$

当方差很小时，$f(x)$ 的值形成的簇比较接近它们的期望值。方差的平方根被称为**标准差**（standard deviation）。

协方差（covariance）在某种意义上给出了两个变量线性相关性的强度以及这些变量的尺度：

$$\mathrm{Cov}(f(x), g(y)) = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])(g(y) - \mathbb{E}[g(y)])]. \tag{3.13}$$

协方差的绝对值如果很大则意味着变量值变化很大并且它们同时距离各自的均值很远。如果协方差是正的，那么两个变量都倾向于同时取得相对较大的值。如果协方差是负的，那么其中一个变量倾向于取得相对较大的值的同时，另一个变量倾向于取得相对较小的值，反之亦然。其他的衡量指标如**相关系数**（correlation）将每个变量的贡献归一化，为了只衡量变量的相关性而不受各个变量尺度大小的影响。

协方差和相关性是有联系的，但实际上是不同的概念。它们是有联系的，因为两个变量如果相互独立那么它们的协方差为零，如果两个变量的协方差不为零那么它们一定是相关的。然而，独立性又是和协方差完全不同的性质。两个变量如果协方差为零，它们之间一定没有线性关系。独立性比零协方差的要求更强，因为独立性还排除了非线性的关系。两个变量相互依赖但具有零协方差是可能的。例如，假设我们首先从区间 $[-1, 1]$ 上的均匀分布中采样出一个实数 x。然后我们对一个随机变量 s 进行采样。s 以 $\frac{1}{2}$ 的概率值为 1，否则为-1。我们可以通过令 $y = sx$ 来生成一个随机变量 y。显然，x 和 y 不是相互独立的，因为 x 完全决定了 y 的尺度。然而，$\mathrm{Cov}(x, y) = 0$。

随机向量 $\boldsymbol{x} \in \mathbb{R}^n$ 的**协方差矩阵**（covariance matrix）是一个 $n \times n$ 的矩阵，并且满足

$$\mathrm{Cov}(\mathbf{x})_{i,j} = \mathrm{Cov}(\mathrm{x}_i, \mathrm{x}_j). \tag{3.14}$$

协方差矩阵的对角元是方差：

$$\mathrm{Cov}(\mathrm{x}_i, \mathrm{x}_i) = \mathrm{Var}(\mathrm{x}_i). \tag{3.15}$$

3.9 常用概率分布

许多简单的概率分布在机器学习的众多领域中都是有用的。

3.9.1 Bernoulli 分布

Bernoulli 分布（Bernoulli distribution）是单个二值随机变量的分布。它由单个参数 $\phi \in [0,1]$ 控制，ϕ 给出了随机变量等于 1 的概率。它具有如下的一些性质：

$$P(\mathrm{x}=1)=\phi \tag{3.16}$$

$$P(\mathrm{x}=0)=1-\phi \tag{3.17}$$

$$P(\mathrm{x}=x)=\phi^{x}(1-\phi)^{1-x} \tag{3.18}$$

$$\mathbb{E}_{\mathrm{x}}[\mathrm{x}]=\phi \tag{3.19}$$

$$\mathrm{Var}_{\mathrm{x}}(\mathrm{x})=\phi(1-\phi) \tag{3.20}$$

3.9.2 Multinoulli 分布

Multinoulli 分布（multinoulli distribution）或者 **范畴分布**（categorical distribution）是指在具有 k 个不同状态的单个离散型随机变量上的分布，其中 k 是一个有限值。[2] Multinoulli 分布由向量 $\boldsymbol{p} \in [0,1]^{k-1}$ 参数化，其中每一个分量 p_i 表示第 i 个状态的概率。最后的第 k 个状态的概率可以通过 $1-\mathbf{1}^{\top}\boldsymbol{p}$ 给出。注意我们必须限制 $\mathbf{1}^{\top}\boldsymbol{p} \leq 1$。Multinoulli 分布经常用来表示对象分类的分布，所以我们很少假设状态 1 具有数值 1 之类的。因此，我们通常不需要去计算 Multinoulli 分布的随机变量的期望和方差。

Bernoulli 分布和 Multinoulli 分布足够用来描述在它们领域内的任意分布。它们能够描述这些分布，不是因为它们特别强大，而是因为它们的领域很简单；它们可以对那些，能够将所有的状态进行枚举的离散型随机变量进行建模。当处理的是连续型随机变量时，会有不可数无限多的状态，所以任何通过少量参数描述的概率分布都必须在分布上加以严格的限制。

[2] "multinoulli" 这个术语是最近被 Gustavo Lacerdo 发明、被Murphy (2012) 推广的。Multinoulli 分布是 **多项式分布**（multinomial distribution）的一个特例。多项式分布是 $\{0,\ldots,n\}^{k}$ 中的向量的分布，用于表示当对 Multinoulli 分布采样 n 次时 k 个类中的每一个被访问的次数。很多文章使用 "多项式分布" 而实际上说的是 Multinoulli 分布，但是他们并没有说是对 $n=1$ 的情况，这点需要注意。

3.9.3 高斯分布

实数上最常用的分布就是**正态分布**（normal distribution），也称为**高斯分布**（Gaussian distribution）：

$$\mathcal{N}(x;\mu,\sigma^2)=\sqrt{\frac{1}{2\pi\sigma^2}}\exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right). \tag{3.21}$$

图 3.1 画出了正态分布的概率密度函数。

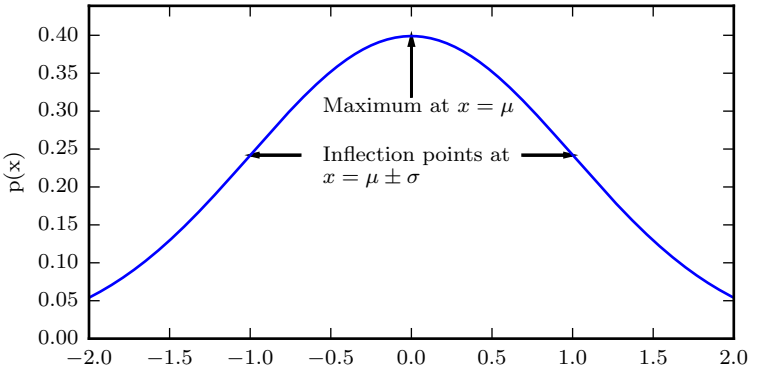

图 3.1: 正态分布。正态分布 $\mathcal{N}(x;\mu,\sigma^2)$ 呈现经典的“钟形曲线”的形状，其中中心峰的 x 坐标由 μ 给出，峰的宽度受 σ 控制。在这个示例中，我们展示的是**标准正态分布**（standard normal distribution），其中 $\mu=0,\sigma=1$。

正态分布由两个参数控制，$\mu\in\mathbb{R}$ 和 $\sigma\in(0,\infty)$。参数 μ 给出了中心峰值的坐标，这也是分布的均值：$\mathbb{E}[\mathrm{x}]=\mu$。分布的标准差用 σ 表示，方差用 σ^2 表示。

当我们要对概率密度函数求值时，我们需要对 σ 平方并且取倒数。当我们需要经常对不同参数下的概率密度函数求值时，一种更高效的参数化分布的方式是使用参数 $\beta\in(0,\infty)$，来控制分布的**精度**（precision）(或方差的倒数)：

$$\mathcal{N}(x;\mu,\beta^{-1})=\sqrt{\frac{\beta}{2\pi}}\exp\left(-\frac{1}{2}\beta(x-\mu)^2\right). \tag{3.22}$$

采用正态分布在很多应用中都是一个明智的选择。当我们由于缺乏关于某个实数上分布的先验知识而不知道该选择怎样的形式时，正态分布是默认的比较好的选择，其中有两个原因。

第一，我们想要建模的很多分布的真实情况是比较接近正态分布的。**中心极限定理**（central limit theorem）说明很多独立随机变量的和近似服从正态分布。这意味着在实际中，很多复杂系统都可以被成功地建模成正态分布的噪声，即使系统可以被分解成一些更结构化的部分。

第二，在具有相同方差的所有可能的概率分布中，正态分布在实数上具有最大的不确定性。因此，我们可以认为正态分布是对模型加入的先验知识量最少的分布。充分利用和证明这个想法需要更多的数学工具，我们推迟到第 19.4.2 节进行讲解。

正态分布可以推广到 $\mathbb{R}^n$ 空间，这种情况下被称为**多维正态分布**（multivariate normal distribution）。它的参数是一个正定对称矩阵 $\boldsymbol{\Sigma}$：

$$\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \sqrt{\frac{1}{(2\pi)^n \det(\boldsymbol{\Sigma})}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right). \tag{3.23}$$

参数 $\boldsymbol{\mu}$ 仍然表示分布的均值，只不过现在是向量值。参数 $\boldsymbol{\Sigma}$ 给出了分布的协方差矩阵。和单变量的情况类似，当我们希望对很多不同参数下的概率密度函数多次求值时，协方差矩阵并不是一个很高效的参数化分布的方式，因为对概率密度函数求值时需要对 $\boldsymbol{\Sigma}$ 求逆。我们可以使用一个**精度矩阵**（precision matrix）$\boldsymbol{\beta}$ 进行替代：

$$\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\beta}^{-1}) = \sqrt{\frac{\det(\boldsymbol{\beta})}{(2\pi)^n}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\beta}(\boldsymbol{x}-\boldsymbol{\mu})\right). \tag{3.24}$$

我们常常把协方差矩阵固定成一个对角阵。一个更简单的版本是**各向同性**（isotropic）高斯分布，它的协方差矩阵是一个标量乘以单位阵。

3.9.4 指数分布和 Laplace 分布

在深度学习中，我们经常会需要一个在 $x = 0$ 点处取得边界点 (sharp point) 的分布。为了实现这一目的，我们可以使用**指数分布**（exponential distribution）：

$$p(x;\lambda) = \lambda \mathbf{1}_{x\geq 0} \exp(-\lambda x). \tag{3.25}$$

指数分布使用指示函数(indicator function)$\mathbf{1}_{x\geq 0}$ 来使得当 x 取负值时的概率为零。

一个联系紧密的概率分布是**Laplace 分布**（Laplace distribution），它允许我们在任意一点 μ 处设置概率质量的峰值

$$\text{Laplace}(x;\mu,\gamma) = \frac{1}{2\gamma} \exp\left(-\frac{|x-\mu|}{\gamma}\right). \tag{3.26}$$

3.9.5 Dirac 分布和经验分布

在一些情况下，我们希望概率分布中的所有质量都集中在一个点上。这可以通过 **Dirac delta 函数**（Dirac delta function）$\delta(x)$ 定义概率密度函数来实现：

$$p(x) = \delta(x - \mu). \tag{3.27}$$

Dirac delta 函数被定义成在除了 0 以外的所有点的值都为 0，但是积分为 1。Dirac delta 函数不像普通函数一样对 x 的每一个值都有一个实数值的输出，它是一种不同类型的数学对象，被称为 **广义函数**（generalized function），广义函数是依据积分性质定义的数学对象。我们可以把 Dirac delta 函数想成一系列函数的极限点，这一系列函数把除 0 以外的所有点的概率密度越变越小。

通过把 $p(x)$ 定义成 δ 函数左移 $-\mu$ 个单位，我们得到了一个在 $x = \mu$ 处具有无限窄也无限高的峰值的概率质量。

Dirac 分布经常作为 **经验分布**（empirical distribution）的一个组成部分出现：

$$\hat{p}(\boldsymbol{x}) = \frac{1}{m} \sum_{i=1}^{m} \delta(\boldsymbol{x} - \boldsymbol{x}^{(i)}) \tag{3.28}$$

经验分布将概率密度 $\frac{1}{m}$ 赋给 m 个点 $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}$ 中的每一个，这些点是给定的数据集或者采样的集合。只有在定义连续型随机变量的经验分布时，Dirac delta 函数才是必要的。对于离散型随机变量，情况更加简单：经验分布可以被定义成一个 Multinoulli 分布，对于每一个可能的输入，其概率可以简单地设为在训练集上那个输入值的 **经验频率**（empirical frequency）。

当我们在训练集上训练模型时，我们可以认为从这个训练集上得到的经验分布指明了我们采样来源的分布。关于经验分布另外一种重要的观点是，它是训练数据的似然最大的那个概率密度函数 (见第 5.5 节)。

3.9.6 分布的混合

通过组合一些简单的概率分布来定义新的概率分布也是很常见的。一种通用的组合方法是构造 **混合分布**（mixture distribution）。混合分布由一些组件 (component) 分布构成。每次实验，样本是由哪个组件分布产生的取决于从一个 Multinoulli 分布中采样的结果：

$$P(\mathrm{x}) = \sum_{i} P(\mathrm{c} = i) P(\mathrm{x} \mid \mathrm{c} = i), \tag{3.29}$$

这里 $P(\mathrm{c})$ 是对各组件的一个 Multinoulli 分布。

我们已经看过一个混合分布的例子了：实值变量的经验分布对于每一个训练实例来说，就是以 Dirac 分布为组件的混合分布。

混合模型是组合简单概率分布来生成更丰富的分布的一种简单策略。在第十六章中，我们更加详细地探讨从简单概率分布构建复杂模型的技术。

混合模型使我们能够一瞥以后会用到的一个非常重要的概念——**潜变量**（latent variable）。潜变量是我们不能直接观测到的随机变量。混合模型的组件标识变量 c 就是其中一个例子。潜变量在联合分布中可能和 x 有关，在这种情况下，$P(\mathrm{x}, \mathrm{c}) = P(\mathrm{x} \mid \mathrm{c})P(\mathrm{c})$。潜变量的分布 $P(\mathrm{c})$ 以及关联潜变量和观测变量的条件分布 $P(\mathrm{x} \mid \mathrm{c})$，共同决定了分布 $P(\mathrm{x})$ 的形状，尽管描述 $P(\mathrm{x})$ 时可能并不需要潜变量。潜变量将在第 16.5 节中深入讨论。

一个非常强大且常见的混合模型是**高斯混合模型**（Gaussian Mixture Model），它的组件 $p(\mathbf{x} \mid \mathrm{c} = i)$ 是高斯分布。每个组件都有各自的参数，均值 $\boldsymbol{\mu}^{(i)}$ 和协方差矩阵 $\boldsymbol{\Sigma}^{(i)}$。有一些混合可以有更多的限制。例如，协方差矩阵可以通过 $\boldsymbol{\Sigma}^{(i)} = \boldsymbol{\Sigma}, \forall i$ 的形式在组件之间共享参数。和单个高斯分布一样，高斯混合模型有时会限制每个组件的协方差矩阵为对角的或者各向同性的 (标量乘以单位矩阵)。

除了均值和协方差以外，高斯混合模型的参数指明了给每个组件 i 的**先验概率**（prior probability）$\alpha_i = P(\mathrm{c} = i)$。“先验”一词表明了在观测到 $\mathbf{x}$ 之前传递给模型关于 c 的信念。作为对比，$P(\mathrm{c} \mid \boldsymbol{x})$ 是**后验概率**（posterior probability），因为它是在观测到 x 之后进行计算的。高斯混合模型是概率密度的**万能近似器**（universal approximator），在这种意义下，任何平滑的概率密度都可以用具有足够多组件的高斯混合模型以任意精度来逼近。

图 3.2 演示了某个高斯混合模型生成的样本。

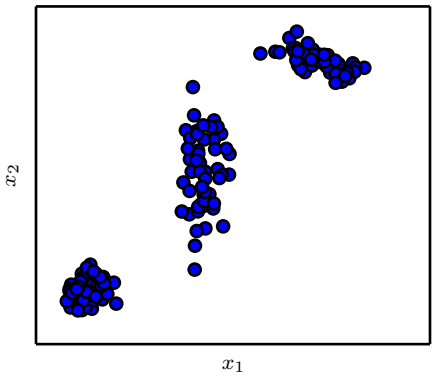

图 3.2: 来自高斯混合模型的样本。在这个示例中，有三个组件。从左到右，第一个组件具有各向同性的协方差矩阵，这意味着它在每个方向上具有相同的方差。第二个组件具有对角的协方差矩阵，这意味着它可以沿着每个轴的对齐方向单独控制方差。该示例中，沿着 x_2 轴的方差要比沿着 x_1 轴的方差大。第三个组件具有满秩的协方差矩阵，使它能够沿着任意基的方向单独地控制方差。

3.10 常用函数的有用性质

某些函数在处理概率分布时经常会出现，尤其是深度学习的模型中用到的概率分布。

其中一个函数是 **logistic sigmoid** 函数：

$$\sigma(x) = \frac{1}{1 + \exp(-x)}. \tag{3.30}$$

logistic sigmoid 函数通常用来产生 Bernoulli 分布中的参数 ϕ，因为它的范围是 $(0, 1)$，处在 ϕ 的有效取值范围内。图 3.3 给出了 sigmoid 函数的图示。sigmoid 函数在变量取绝对值非常大的正值或负值时会出现 **饱和**（saturate）现象，意味着函数会变得很平，并且对输入的微小改变会变得不敏感。

另外一个经常遇到的函数是 **softplus 函数**（softplus function）(Dugas *et al.*, 2001)：

$$\zeta(x) = \log(1 + \exp(x)). \tag{3.31}$$

softplus 函数可以用来产生正态分布的 β 和 σ 参数，因为它的范围是 $(0, \infty)$。当处理包含 sigmoid 函数的表达式时它也经常出现。softplus 函数名来源于它是另外一个

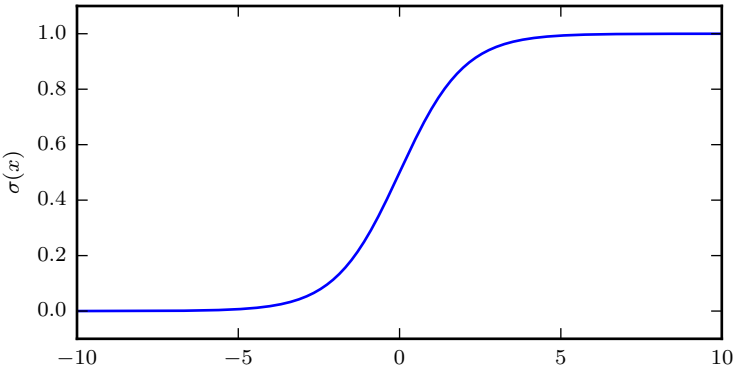

图 3.3: logistic sigmoid函数。

函数的平滑（或“软化”）形式，这个函数是

$$x^+ = \max(0, x). \tag{3.32}$$

图 3.4 给出了 softplus 函数的图示。

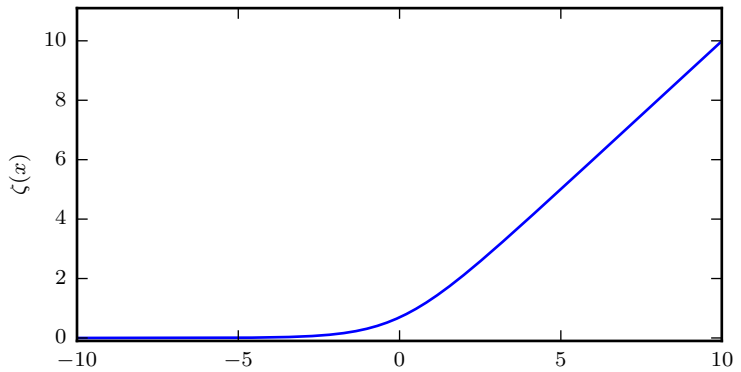

图 3.4: softplus 函数。

下面一些性质非常有用，你可能要记下来：

$$\sigma(x) = \frac{\exp(x)}{\exp(x) + \exp(0)} \tag{3.33}$$

$$\frac{d}{dx}\sigma(x) = \sigma(x)(1 - \sigma(x)) \tag{3.34}$$

$$1 - \sigma(x) = \sigma(-x) \tag{3.35}$$

$$\log \sigma(x) = -\zeta(-x) \tag{3.36}$$

$$\frac{d}{dx}\zeta(x) = \sigma(x) \tag{3.37}$$

$$\forall x \in (0, 1), \sigma^{-1}(x) = \log\left(\frac{x}{1-x}\right) \tag{3.38}$$

$$\forall x > 0, \zeta^{-1}(x) = \log(\exp(x) - 1) \tag{3.39}$$

$$\zeta(x) = \int_{-\infty}^{x} \sigma(y)dy \tag{3.40}$$

$$\zeta(x) - \zeta(-x) = x \tag{3.41}$$

函数 $\sigma^{-1}(x)$ 在统计学中被称为 **分对数**（ logit ），但这个函数在机器学习中很少用到。

式 (3.41) 为函数名 “softplus’’ 提供了其他的正当理由。softplus 函数被设计成 **正部函数**（ positive part function ）的平滑版本，这个正部函数是指 $x^+ = \max\{0, x\}$。与正部函数相对的是 **负部函数**（ negative part function ）$x^- = \max\{0, -x\}$。为了获得类似负部函数的一个平滑函数，我们可以使用 $\zeta(-x)$。就像 x 可以用它的正部和负部通过等式 $x^+ - x^- = x$ 恢复一样，我们也可以用同样的方式对 $\zeta(x)$ 和 $\zeta(-x)$ 进行操作，就像式 (3.41) 中那样。

3.11 贝叶斯规则

我们经常会需要在已知 $P(\mathrm{y} \mid \mathrm{x})$ 时计算 $P(\mathrm{x} \mid \mathrm{y})$。幸运的是，如果还知道 $P(\mathrm{x})$，我们可以用 **贝叶斯规则**（ Bayes’ rule ）来实现这一目的：

$$P(\mathrm{x} \mid \mathrm{y}) = \frac{P(\mathrm{x})P(\mathrm{y} \mid \mathrm{x})}{P(\mathrm{y})}. \tag{3.42}$$

注意到 $P(\mathrm{y})$ 出现在上面的公式中，它通常使用 $P(\mathrm{y}) = \sum_x P(\mathrm{y} \mid x)P(x)$ 来计算，所以我们并不需要事先知道 $P(\mathrm{y})$ 的信息。

贝叶斯规则可以从条件概率的定义直接推导得出，但我们最好记住这个公式的名字，因为很多文献通过名字来引用这个公式。这个公式是以牧师 Thomas Bayes 的名字来命名的，他是第一个发现这个公式特例的人。这里介绍的一般形式由 Pierre-Simon Laplace 独立发现。

3.12 连续型变量的技术细节

连续型随机变量和概率密度函数的深入理解需要用到数学分支 **测度论**（measure theory）的相关内容来扩展概率论。测度论超出了本书的范畴，但我们可以简要勾勒一些测度论用来解决的问题。

在第 3.3.2 节中，我们已经看到连续型向量值随机变量 $\mathbf{x}$ 落在某个集合 $\mathbb{S}$ 中的概率是通过 $p(\boldsymbol{x})$ 对集合 $\mathbb{S}$ 积分得到的。对于集合 $\mathbb{S}$ 的一些选择可能会引起悖论。例如，构造两个集合 $\mathbb{S}_1$ 和 $\mathbb{S}_2$ 使得 $p(\boldsymbol{x} \in \mathbb{S}_1) + p(\boldsymbol{x} \in \mathbb{S}_2) > 1$ 并且 $\mathbb{S}_1 \cap \mathbb{S}_2 = \emptyset$ 是可能的。这些集合通常是大量使用了实数的无限精度来构造的，例如通过构造分形形状 (fractal-shaped) 的集合或者是通过有理数相关集合的变换定义的集合。[3] 测度论的一个重要贡献就是提供了一些集合的特征使得我们在计算概率时不会遇到悖论。在本书中，我们只对相对简单的集合进行积分，所以测度论的这个方面不会成为一个相关考虑。

对于我们的目的，测度论更多的是用来描述那些适用于 $\mathbb{R}^n$ 上的大多数点，却不适用于一些边界情况的定理。测度论提供了一种严格的方式来描述那些非常微小的点集。这种集合被称为“**零测度**（measure zero）”的。我们不会在本书中给出这个概念的正式定义。然而，直观地理解这个概念是有用的，我们可以认为零测度集在我们的度量空间中不占有任何的体积。例如，在 $\mathbb{R}^2$ 空间中，一条直线的测度为零，而填充的多边形具有正的测度。类似的，一个单独的点的测度为零。可数多个零测度集的并仍然是零测度的 (所以所有有理数构成的集合测度为零)。

另外一个有用的测度论中的术语是“**几乎处处**（almost everywhere）”。某个性质如果是几乎处处都成立的，那么它在整个空间中除了一个测度为零的集合以外都是成立的。因为这些例外只在空间中占有极其微小的量，它们在多数应用中都可以被放心地忽略。概率论中的一些重要结果对于离散值成立但对于连续值只能是“几乎处处”成立。

[3] Banach-Tarski 定理给出了这类集合的一个有趣的例子。译者注：我们这里把“the set of rational numbers”翻译成“有理数相关集合”，理解为“一些有理数组成的集合”，如果直接用后面的翻译读起来会比较拗口。

连续型随机变量的另一技术细节，涉及到处理那种相互之间有确定性函数关系的连续型变量。假设我们有两个随机变量 $\mathbf{x}$ 和 $\mathbf{y}$ 满足 $\boldsymbol{y} = g(\boldsymbol{x})$，其中 g 是可逆的、连续可微的函数。可能有人会想 $p_y(\boldsymbol{y}) = p_x(g^{-1}(\boldsymbol{y}))$。但实际上这并不对。

举一个简单的例子，假设我们有两个标量值随机变量 x 和 y，并且满足 $\mathrm{y} = \frac{\mathrm{x}}{2}$ 以及 $\mathrm{x} \sim U(0,1)$。如果我们使用 $p_y(y) = p_x(2y)$，那么 p_y 除了区间 $[0, \frac{1}{2}]$ 以外都为 0，并且在这个区间上的值为 1。这意味着

$$\int p_y(y)dy = \frac{1}{2}, \tag{3.43}$$

而这违背了概率密度的定义 (积分为 1)。这个常见错误之所以错是因为它没有考虑到引入函数 g 后造成的空间变形。回忆一下，$\boldsymbol{x}$ 落在无穷小的体积为 $\delta\boldsymbol{x}$ 的区域内的概率为 $p(\boldsymbol{x})\delta\boldsymbol{x}$。因为 g 可能会扩展或者压缩空间，在 $\boldsymbol{x}$ 空间内的包围着 $\boldsymbol{x}$ 的无穷小体积在 $\boldsymbol{y}$ 空间中可能有不同的体积。

为了看出如何改正这个问题，我们回到标量值的情况。我们需要保持下面这个性质：

$$|p_y(g(x))dy| = |p_x(x)dx|. \tag{3.44}$$

求解上式，我们得到

$$p_y(y) = p_x(g^{-1}(y)) \left|\frac{\partial x}{\partial y}\right| \tag{3.45}$$

或者等价地，

$$p_x(x) = p_y(g(x)) \left|\frac{\partial g(x)}{\partial x}\right|. \tag{3.46}$$

在高维空间中，微分运算扩展为**Jacobian 矩阵**（Jacobian matrix）的行列式——矩阵的每个元素为 $J_{i,j} = \frac{\partial x_i}{\partial y_j}$。因此，对于实值向量 $\boldsymbol{x}$ 和 $\boldsymbol{y}$，

$$p_x(\boldsymbol{x}) = p_y(g(\boldsymbol{x})) \left|\det\left(\frac{\partial g(\boldsymbol{x})}{\partial \boldsymbol{x}}\right)\right|. \tag{3.47}$$

3.13 信息论

信息论是应用数学的一个分支，主要研究的是对一个信号包含信息的多少进行量化。它最初被发明是用来研究在一个含有噪声的信道上用离散的字母表来发送消息，例如通过无线电传输来通信。在这种情况下，信息论告诉我们如何对消息设计最优编码以及计算消息的期望长度，这些消息是使用多种不同编码机制、从特定

的概率分布上采样得到的。在机器学习中，我们也可以把信息论应用于连续型变量，此时某些消息长度的解释不再适用。信息论是电子工程和计算机科学中许多领域的基础。在本书中，我们主要使用信息论的一些关键思想来描述概率分布或者量化概率分布之间的相似性。有关信息论的更多细节，参见 Cover and Thomas (2006) 或者 MacKay (2003)。

信息论的基本想法是一个不太可能的事件居然发生了，要比一个非常可能的事件发生，能提供更多的信息。消息说：“今天早上太阳升起” 信息量是如此之少以至于没有必要发送，但一条消息说：“今天早上有日食” 信息量就很丰富。

我们想要通过这种基本想法来量化信息。特别地，

- 非常可能发生的事件信息量要比较少，并且极端情况下，确保能够发生的事件应该没有信息量。
- 较不可能发生的事件具有更高的信息量。
- 独立事件应具有增量的信息。例如，投掷的硬币两次正面朝上传递的信息量，应该是投掷一次硬币正面朝上的信息量的两倍。

为了满足上述三个性质，我们定义一个事件 $\mathrm{x} = x$ 的 **自信息**（self-information）为

$$I(x) = -\log P(x). \tag{3.48}$$

在本书中，我们总是用 log 来表示自然对数，其底数为 e。因此我们定义的 $I(x)$ 单位是 **奈特**（nats）。一奈特是以 $\frac{1}{e}$ 的概率观测到一个事件时获得的信息量。其他的材料中使用底数为 2 的对数，单位是 **比特**（bit）或者 **香农**（shannons）；通过比特度量的信息只是通过奈特度量信息的常数倍。

当 x 是连续的，我们使用类似的关于信息的定义，但有些来源于离散形式的性质就丢失了。例如，一个具有单位密度的事件信息量仍然为 0，但是不能保证它一定发生。

自信息只处理单个的输出。我们可以用 **香农熵**（Shannon entropy）来对整个概率分布中的不确定性总量进行量化：

$$H(\mathrm{x}) = \mathbb{E}_{\mathrm{x}\sim P}[I(x)] = -\mathbb{E}_{\mathrm{x}\sim P}[\log P(x)], \tag{3.49}$$

也记作 $H(P)$。换言之，一个分布的香农熵是指遵循这个分布的事件所产生的期望信息总量。它给出了对依据概率分布 P 生成的符号进行编码所需的比特数在平均意义

上的下界 (当对数底数不是 2 时，单位将有所不同)。那些接近确定性的分布 (输出几乎可以确定) 具有较低的熵；那些接近均匀分布的概率分布具有较高的熵。图 3.5 给出了一个说明。当 x 是连续的，香农熵被称为 **微分熵**（differential entropy）。

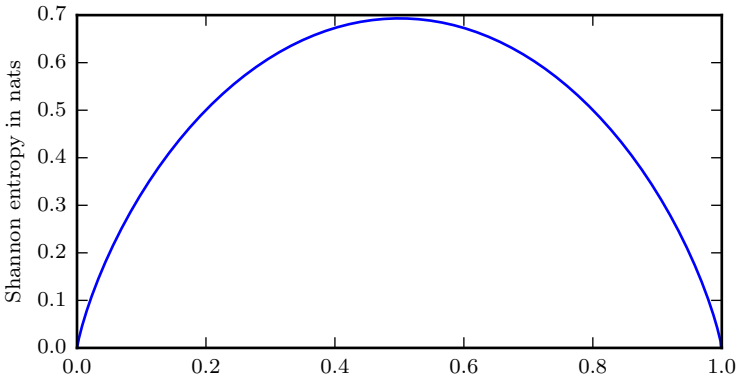

图 3.5: 二值随机变量的香农熵。该图说明了更接近确定性的分布是如何具有较低的香农熵，而更接近均匀分布的分布是如何具有较高的香农熵。水平轴是 p，表示二值随机变量等于 1 的概率。熵由 $(p-1)\log(1-p) - p\log p$ 给出。当 p 接近 0 时，分布几乎是确定的，因为随机变量几乎总是 0。当 p 接近 1 时，分布也几乎是确定的，因为随机变量几乎总是 1。当 $p = 0.5$ 时，熵是最大的，因为分布在两个结果（0 和 1）上是均匀的。

如果我们对于同一个随机变量 x 有两个单独的概率分布 $P(\mathrm{x})$ 和 $Q(\mathrm{x})$，我们可以使用 **KL 散度**（Kullback-Leibler (KL) divergence）来衡量这两个分布的差异：

$$D_{\mathrm{KL}}(P||Q) = \mathbb{E}_{\mathrm{x}\sim P}\left[\log \frac{P(x)}{Q(x)}\right] = \mathbb{E}_{\mathrm{x}\sim P}[\log P(x) - \log Q(x)]. \tag{3.50}$$

在离散型变量的情况下，KL 散度衡量的是，当我们使用一种被设计成能够使得概率分布 Q 产生的消息的长度最小的编码，发送包含由概率分布 P 产生的符号的消息时，所需要的额外信息量 (如果我们使用底数为 2 的对数时，信息量用比特衡量，但在机器学习中，我们通常用奈特和自然对数。)

KL 散度有很多有用的性质，最重要的是它是非负的。KL 散度为 0 当且仅当 P 和 Q 在离散型变量的情况下是相同的分布，或者在连续型变量的情况下是 “几乎处处” 相同的。因为 KL 散度是非负的并且衡量的是两个分布之间的差异，它经常被用作分布之间的某种距离。然而，它并不是真的距离因为它不是对称的：对于某些 P 和 Q，$D_{\mathrm{KL}}(P||Q) \neq D_{\mathrm{KL}}(Q||P)$。这种非对称性意味着选择 $D_{\mathrm{KL}}(P||Q)$ 还是

$D_{\mathrm{KL}}(Q||P)$ 影响很大。更多细节可以看图 3.6 。

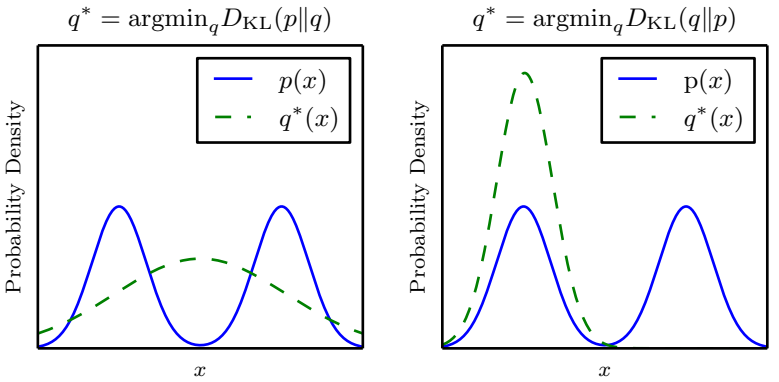

图 3.6: KL 散度是不对称的。假设我们有一个分布 $p(x)$，并且希望用另一个分布 $q(x)$ 来近似它。我们可以选择最小化 $D_{\mathrm{KL}}(p||q)$ 或最小化 $D_{\mathrm{KL}}(q||p)$。为了说明每种选择的效果，我们令 p 是两个高斯分布的混合，令 q 为单个高斯分布。选择使用 KL 散度的哪个方向是取决于问题的。一些应用需要这个近似分布 q 在真实分布 p 放置高概率的所有地方都放置高概率，而其他应用需要这个近似分布 q 在真实分布 p 放置低概率的所有地方都很少放置高概率。KL 散度方向的选择反映了对于每种应用，优先考虑哪一种选择。*(左)* 最小化 $D_{\mathrm{KL}}(p||q)$ 的效果。在这种情况下，我们选择一个 q 使得它在 p 具有高概率的地方具有高概率。当 p 具有多个峰时，q 选择将这些峰模糊到一起，以便将高概率质量放到所有峰上。*(右)* 最小化 $D_{\mathrm{KL}}(q||p)$ 的效果。在这种情况下，我们选择一个 q 使得它在 p 具有低概率的地方具有低概率。当 p 具有多个峰并且这些峰间隔很宽时，如该图所示，最小化 KL 散度会选择单个峰，以避免将概率质量放置在 p 的多个峰之间的低概率区域中。这里，我们说明当 q 被选择成强调左边峰时的结果。我们也可以通过选择右边峰来得到 KL 散度相同的值。如果这些峰没有被足够强的低概率区域分离，那么 KL 散度的这个方向仍然可能选择模糊这些峰。

一个和 KL 散度密切联系的量是 **交叉熵**（cross-entropy）$H(P,Q) = H(P) + D_{\mathrm{KL}}(P||Q)$，它和 KL 散度很像但是缺少左边一项：

$$H(P,Q) = -\mathbb{E}_{\mathrm{x}\sim P} \log Q(x). \tag{3.51}$$

针对 Q 最小化交叉熵等价于最小化 KL 散度，因为 Q 并不参与被省略的那一项。

当我们计算这些量时，经常会遇到 $0 \log 0$ 这个表达式。按照惯例，在信息论中，我们将这个表达式处理为 $\lim_{x\to 0} x \log x = 0$。

3.14 结构化概率模型

机器学习的算法经常会涉及到在非常多的随机变量上的概率分布。通常，这些概率分布涉及到的直接相互作用都是介于非常少的变量之间的。使用单个函数来描述整个联合概率分布是非常低效的 (无论是计算上还是统计上)。

我们可以把概率分布分解成许多因子的乘积形式，而不是使用单一的函数来表示概率分布。例如，假设我们有三个随机变量 a, b 和 c，并且 a 影响 b 的取值，b 影响 c 的取值，但是 a 和 c 在给定 b 时是条件独立的。我们可以把全部三个变量的概率分布重新表示为两个变量的概率分布的连乘形式：

$$p(\mathrm{a}, \mathrm{b}, \mathrm{c}) = p(\mathrm{a})p(\mathrm{b} \mid \mathrm{a})p(\mathrm{c} \mid \mathrm{b}). \tag{3.52}$$

这种分解可以极大地减少用来描述一个分布的参数数量。每个因子使用的参数数目是它的变量数目的指数倍。这意味着，如果我们能够找到一种使每个因子分布具有更少变量的分解方法，我们就能极大地降低表示联合分布的成本。

我们可以用图来描述这种分解。这里我们使用的是图论中的 ‘‘图’’ 的概念：由一些可以通过边互相连接的顶点的集合构成。当我们用图来表示这种概率分布的分解，我们把它称为 **结构化概率模型**（structured probabilistic model）或者 **图模型**（graphical model）。

有两种主要的结构化概率模型：有向的和无向的。两种图模型都使用图 $\mathcal{G}$，其中图的每个节点对应着一个随机变量，连接两个随机变量的边意味着概率分布可以表示成这两个随机变量之间的直接作用。

有向（directed）模型使用带有有向边的图，它们用条件概率分布来表示分解，就像上面的例子。特别地，有向模型对于分布中的每一个随机变量 x_i 都包含着一个影响因子，这个组成 x_i条件概率的影响因子被称为 x_i 的父节点，记为 $Pa_{\mathcal{G}}(\mathrm{x}_i)$：

$$p(\mathbf{x}) = \prod_i p(\mathrm{x}_i \mid Pa_{\mathcal{G}}(\mathrm{x}_i)). \tag{3.53}$$

图 3.7 给出了一个有向图的例子以及它表示的概率分布的分解。

无向（undirected）模型使用带有无向边的图，它们将分解表示成一组函数；不像有向模型那样，这些函数通常不是任何类型的概率分布。$\mathcal{G}$ 中任何满足两两之间有边连接的顶点的集合被称为团。无向模型中的每个团 $\mathcal{C}^{(i)}$ 都伴随着一个因子 $\phi^{(i)}(\mathcal{C}^{(i)})$。这些因子仅仅是函数，并不是概率分布。每个因子的输出都必须是非负

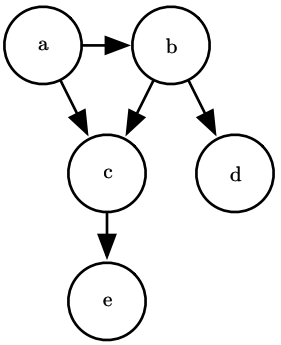

图 3.7: 关于随机变量 a, b, c, d 和 e 的有向图模型。这幅图对应的概率分布可以分解为

$$p(\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, \mathrm{e}) = p(\mathrm{a})p(\mathrm{b} \mid \mathrm{a})p(\mathrm{c} \mid \mathrm{a}, \mathrm{b})p(\mathrm{d} \mid \mathrm{b})p(\mathrm{e} \mid \mathrm{c}). \tag{3.54}$$

该图模型使我们能够快速看出此分布的一些性质。例如，a 和 c 直接相互影响，但 a 和 e 只有通过 c 间接相互影响。

的，但是并没有像概率分布中那样要求因子的和或者积分为 1。

随机变量的联合概率与所有这些因子的乘积 **成比例**（proportional）——意味着因子的值越大则可能性越大。当然，不能保证这种乘积的求和为 1。所以我们需要除以一个归一化常数 Z 来得到归一化的概率分布，归一化常数 Z 被定义为 ϕ 函数乘积的所有状态的求和或积分。概率分布为：

$$p(\mathbf{x}) = \frac{1}{Z} \prod_i \phi^{(i)}\left(\mathcal{C}^{(i)}\right). \tag{3.55}$$

图 3.8 给出了一个无向图的例子以及它表示的概率分布的分解。

请记住，这些图模型表示的分解仅仅是描述概率分布的一种语言。它们不是互相排斥的概率分布族。有向或者无向不是概率分布的特性；它是概率分布的一种特殊 **描述**（description）所具有的特性，而任何概率分布都可以用这两种方式进行描述。

在本书第一部分和第二部分中，我们仅仅将结构化概率模型视作一门语言，来描述不同的机器学习算法选择表示的直接的概率关系。在讨论研究课题之前，读者不需要更深入地理解结构化概率模型。在第三部分的研究课题中，我们将更为详尽地探讨结构化概率模型。

本章复习了概率论中与深度学习最为相关的一些基本概念。我们还剩下一些基