数学基础

本附录介绍一些深度学习涉及的数学基础知识，包括线性代数、微积分、数值优化、概率论和信息论等。

A　线性代数

线性代数主要包含向量、向量空间（或称线性空间）以及向量的线性变换和有限维的线性方程组。

A.1　向量和向量空间

A.1.1　向量

标量（scalar）是一个实数，只有大小，没有方向。而向量（vector）是由一组实数组成的有序数组，同时具有大小和方向。一个n维向量$\mathbf{a}$是由n个有序实数组成，表示为

$$\mathbf{a} = [a_1, a_2, \cdots, a_n], \tag{A.1}$$

其中a_i称为向量$\mathbf{a}$的第i个分量，或第i维。向量符号一般用黑体小写字母$\mathbf{a}, \mathbf{b}, \mathbf{c}$，或小写希腊字母$\alpha, \beta, \gamma$等来表示。

A.1.2　向量空间

向量空间（vector space），也称线性空间（linear space），是指由向量组成的集合，并满足以下两个条件：

1. 向量加法$+$：向量空间$\mathcal{V}$中的两个向量$\mathbf{a}$和$\mathbf{b}$，它们的和$\mathbf{a}+\mathbf{b}$也属于空间$\mathcal{V}$；
2. 标量乘法$\cdot$：向量空间$\mathcal{V}$中的任一向量$\mathbf{a}$和任一标量c，它们的乘积$c \cdot \mathbf{a}$也属于空间$\mathcal{V}$。

欧氏空间 一个常用的线性空间是欧氏空间（Euclidean space）。一个欧氏空间表示通常为$\mathbb{R}^n$，其中n为空间维度（dimension）。欧氏空间中向量的加法和标量乘法定义为：

$$[a_1, a_2, \cdots, a_n] + [b_1, b_2, \cdots, b_n] = [a_1 + b_1, a_2 + b_2, \cdots, a_n + b_n], \tag{A.2}$$

$$c[a_1, a_2, \cdots, a_n] = [ca_1, ca_2, \cdots, ca_n], \tag{A.3}$$

其中$a, b, c \in \mathbb{R}$为一个标量。

线性无关 线性空间$\mathcal{V}$中的一组向量$\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}$，如果对任意的一组标量$\lambda_1, \lambda_2, \cdots, \lambda_n$，满足$\lambda_1\mathbf{v}_1 + \lambda_2\mathbf{v}_2 + \cdots + \lambda_n\mathbf{v}_n = 0$，则必然$\lambda_1 = \lambda_2 = \cdots = \lambda_n = 0$，那么$\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}$是线性无关的，也称为线性独立的。

基向量 向量空间$\mathcal{V}$的基（bases）$\mathcal{B} = \{\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_n\}$是$\mathcal{V}$的有限子集，其元素之间线性无关。向量空间$\mathcal{V}$所有的向量都可以按唯一的方式表达为$\mathcal{B}$中向量的线性组合。对任意$v \in \mathcal{V}$，存在一组标量$(\lambda_1, \lambda_2, \cdots, \lambda_n)$使得

$$\mathbf{v} = \lambda_1\mathbf{e}_1 + \lambda_2\mathbf{e}_2 + \cdots + \lambda_n\mathbf{e}_n, \tag{A.4}$$

其中基$\mathcal{B}$中的向量称为基向量（base vector）。如果基向量是有序的，则标量$(\lambda_1, \lambda_2, \cdots, \lambda_n)$称为向量$\mathbf{v}$关于基$\mathcal{B}$的坐标（coordinates）。

n维空间$\mathcal{V}$的一组标准基（standard basis）为

$$\mathbf{e}_1 = [1, 0, 0, \cdots, 0], \tag{A.5}$$

$$\mathbf{e}_2 = [0, 1, 0, \cdots, 0], \tag{A.6}$$

$$\cdots \tag{A.7}$$

$$\mathbf{e}_n = [0, 0, 0, \cdots, 1], \tag{A.8}$$

$\mathcal{V}$中的任一向量$\mathbf{v} = [v_1, v_2, \cdots, v_n]$可以唯一的表示为

$$[v_1, v_2, \cdots, v_n] = v_1\mathbf{e}_1 + v_2\mathbf{e}_2 + \cdots + v_n\mathbf{e}_n, \tag{A.9}$$

$v_1, v_2, \cdots, v_n$也称为向量$\mathbf{v}$的笛卡尔坐标（Cartesian coordinates）。

向量空间中的每个向量可以看作是一个线性空间中的笛卡儿坐标。

内积 一个n维线性空间中的两个向量$\mathbf{a}$和$\mathbf{b}$，其内积为

$$\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{i=1}^{n} a_i b_i, \tag{A.10}$$

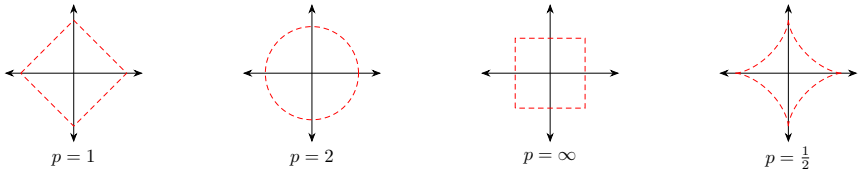


图 A.1 常见的范数。红线表示不同范数的 $\ell_p=1$ 的点。

A.1.3 范数

范数（norm）是一个表示向量“长度”的函数，为向量空间内的所有向量赋予非零的正长度或大小。对于一个 n 维向量 $\mathbf{v}$，一个常见的范数函数为 ℓ_p 范数，

$$\ell_p(\mathbf{v}) \equiv \|\mathbf{v}\|_p = \left(\sum_{i=1}^{n} |v_i|^p\right)^{1/p}, \tag{A.11}$$

其中 $p \geq 0$ 为一个标量的参数。常用的 p 的取值有 1，2，∞ 等。

ℓ_1 范数 ℓ_1 范数为向量的各个元素的绝对值之和。

$$\|\mathbf{v}\|_1 = \sum_{i=1}^{n} |v_i|. \tag{A.12}$$

ℓ_2 范数 ℓ_2 范数为向量的各个元素的

$$\|\mathbf{v}\|_2 = \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{\mathbf{v}^{\mathrm{T}}\mathbf{v}}. \tag{A.13}$$

ℓ_2 范数又称为 *Euclidean* 范数或者 *Frobenius* 范数。从几何角度，向量也可以表示为从原点出发的一个带箭头的有向线段，其 ℓ_2 范数为线段的长度，也常称为向量的模。

ℓ_∞ 范数 ℓ_∞ 范数为向量的各个元素的最大绝对值，

$$\|\mathbf{v}\|_\infty = \max\{v_1, v_2, \cdots, v_n\}. \tag{A.14}$$

图A.1给出了常见范数的示例。

A.1.4 常见的向量

全 *0* 向量指所有元素都为 0 的向量，用 $\mathbf{0}$ 表示。全 0 向量为笛卡尔坐标系中的原点。

全 *1* 向量指所有值为1的向量，用 $\mathbf{1}$ 表示。

one-hot 向量为有且只有一个元素为1，其余元素都为0的向量。one-hot 向量是在数字电路中的一种状态编码，指对任意给定的状态，状态寄存器中只有1位为1，其余位都为0。

A.2 矩阵

A.2.1 线性映射

线性映射（linear map）是指从线性空间 $\mathcal{V}$ 到线性空间 $\mathcal{W}$ 的一个映射函数 $f:\mathcal{V}\to\mathcal{W}$，并满足：对于 $\mathcal{V}$ 中任何两个向量 $\mathbf{u}$ 和 $\mathbf{v}$ 以及任何标量 c，有

$$f(\mathbf{u}+\mathbf{v})=f(\mathbf{u})+f(\mathbf{v}), \tag{A.15}$$

$$f(c\mathbf{v})=cf(\mathbf{v}). \tag{A.16}$$

两个有限维欧氏空间的映射函数 $f:\mathbb{R}^n\to\mathbb{R}^m$ 可以表示为

$$\mathbf{y}=A\mathbf{x}\triangleq\begin{bmatrix}a_{11}x_1+a_{12}x_2+\cdots+a_{1n}x_n\\ a_{21}x_1+a_{22}x_2+\cdots+a_{2n}x_n\\ \vdots\\ a_{m1}x_1+a_{m2}x_2+\cdots+a_{mn}x_n\end{bmatrix}, \tag{A.17}$$

其中 A 定义为 $m\times n$ 的矩阵（matrix），是一个由 m 行 n 列元素排列成的矩形阵列。一个矩阵 A 从左上角数起的第 i 行第 j 列上的元素称为第 i,j 项，通常记为 $[A]_{ij}$ 或 a_{ij}。矩阵 A 定义了一个从 $\mathbb{R}^n$ 到 $\mathbb{R}^m$ 的线性映射；向量 $\mathbf{x}\in\mathbb{R}^n$ 和 $\mathbf{y}\in\mathbb{R}^m$ 分别为两个空间中的列向量，即大小为 $n\times 1$ 的矩阵。

$$\mathbf{x}=\begin{bmatrix}x_1\\ x_2\\ \vdots\\ x_n\end{bmatrix},\qquad \mathbf{y}=\begin{bmatrix}y_1\\ y_2\\ \vdots\\ y_m\end{bmatrix}. \tag{A.18}$$

如果没有特别说明，本书默认向量为列向量。

为简化书写、方便排版起见，本书约定逗号隔离的向量表示 $[x_1,x_2,\cdots,x_n]$ 为行向量，列向量通常用分号隔开的表示 $\mathbf{x}=[x_1;x_2;\cdots;x_n]$，或行向量的转置 $[x_1,x_2,\cdots,x_n]^{\mathrm{T}}$。

A.2.2 矩阵操作

加 如果 A 和 B 都为 $m \times n$ 的矩阵，则 A 和 B 的加也是 $m \times n$ 的矩阵，其每个元素是 A 和 B 相应元素相加。

$$[A + B]_{ij} = a_{ij} + b_{ij}. \tag{A.19}$$

乘积 假设有两个 A 和 B 分别表示两个线性映射 $g : \mathbb{R}^m \to \mathbb{R}^k$ 和 $f : \mathbb{R}^n \to \mathbb{R}^m$，则其复合线性映射

$$(g \circ f)(\mathbf{x}) = g(f(\mathbf{x})) = g(B\mathbf{x}) = A(B\mathbf{x}) = (AB)\mathbf{x}, \tag{A.20}$$

其中 AB 表示矩阵 A 和 B 的乘积，定义为

$$[AB]_{ij} = \sum_{k=1}^{m} a_{ik} b_{kj}. \tag{A.21}$$

两个矩阵的乘积仅当第一个矩阵的列数和第二个矩阵的行数相等时才能定义。如 A 是 $k \times m$ 矩阵和 B 是 $m \times n$ 矩阵，则乘积 AB 是一个 $k \times n$ 的矩阵。

矩阵的乘法满足结合律和分配律：

- 结合律：$(AB)C = A(BC)$,
- 分配律：$(A + B)C = AC + BC$，$C(A + B) = CA + CB$.

Hadamard 积 A 和 B 的 *Hadamard* 积，也称为逐点乘积，为 A 和 B 中对应的元素相乘。

$$[A \odot B]_{ij} = a_{ij} b_{ij}. \tag{A.22}$$

一个标量 c 与矩阵 A 乘积为 A 的每个元素是 A 的相应元素与 c 的乘积

$$[cA]_{ij} = ca_{ij}. \tag{A.23}$$

转置 $m \times n$ 矩阵 A 的转置（transposition）是一个 $n \times m$ 的矩阵，记为 A^{T}，A^{T} 的第 i 行第 j 列的元素是原矩阵 A 的第 j 行第 i 列的元素，

$$[A^{\mathrm{T}}]_{ij} = [A]_{ji}. \tag{A.24}$$

向量化 矩阵的向量化是将矩阵表示为一个列向量。这里，**vec** 是向量化算子。设 $A = [a_{ij}]_{m \times n}$，则

$$\mathbf{vec}(A) = [a_{11}, a_{21}, \cdots, a_{m1}, a_{12}, a_{22}, \cdots, a_{m2}, \cdots, a_{1n}, a_{2n}, \cdots, a_{mn}]^{\mathrm{T}}.$$

迹 方块矩阵 A 的对角线元素之和称为它的迹（trace），记为 $tr(A)$。尽管矩阵的乘法不满足交换律，但它们的迹相同，即 $tr(AB) = tr(BA)$。

行列式 方块矩阵 A 的行列式是一个将其映射到标量的函数，记作 $\det(A)$ 或 $|A|$。行列式可以看做是有向面积或体积的概念在欧氏空间中的推广。在 n 维欧氏空间中，行列式描述的是一个线性变换对“体积”所造成的影响。

一个 $n \times n$ 的方块矩阵 A 的行列式定义为：

$$\det(A) = \sum_{\sigma \in S_n} \text{sign}(\sigma) \prod_{i=1}^{n} a_{i,\sigma(i)} \tag{A.25}$$

其中 S_n 是 $\{1, 2, ..., n\}$ 的所有排列的集合，σ 是其中一个排列，$\sigma(i)$ 是元素 i 在排列 σ 中的位置，$\text{sign}(\sigma)$ 表示排列 σ 的符号差，定义为

$$(\sigma) = \begin{cases} 1 & \sigma\text{中的逆序对有偶数个} \\ -1 & \sigma\text{中的逆序对有奇数个} \end{cases} \tag{A.26}$$

其中逆序对的定义为：在排列 σ 中，如果有序数对 (i, j) 满足 $1 \leq i < j \leq n$ 但 $\sigma(i) > \sigma(j)$，则其为 σ 的一个逆序对。

秩 一个矩阵 A 的列秩是 A 的线性无关的列向量数量，行秩是 A 的线性无关的行向量数量。一个矩阵的列秩和行秩总是相等的，简称为秩（rank）。

一个 $m \times n$ 的矩阵的秩最大为 $\min(m, n)$。两个句子的乘积 AB 的秩 $\text{rank}(AB) \leq \min\Big(\text{rank}(A), \text{rank}(B)\Big)$。

范数 矩阵的范数有很多种形式，其中常用的 ℓ_p 范数定义为

$$\|A\|_p = \Big(\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^p\Big)^{1/p}. \tag{A.27}$$

A.2.3 矩阵类型

对称矩阵 对称矩阵（symmetric）指其转置等于自己的矩阵，即满足 $A = A^{\mathrm{T}}$。

对角矩阵 对角矩阵（diagonal matrix）是一个主对角线之外的元素皆为 0 的矩阵。对角线上的元素可以为 0 或其他值。一个 $n \times n$ 的对角矩阵 A 满足：

$$[A]_{ij} = 0 \text{ if } i \neq j \qquad \forall i, j \in \{1, \cdots, n\} \tag{A.28}$$

对角矩阵 A 也可以记为 $\text{diag}(\mathbf{a})$，$\mathbf{a}$ 为一个 n 维向量，并满足

$$[A]_{ii} = a_i. \tag{A.29}$$

$n \times n$的对角矩阵$A = \text{diag}(\mathbf{a})$和$n$维向量$\mathbf{b}$的乘积为一个$n$维向量

$$A\mathbf{b} = \text{diag}(\mathbf{a})\mathbf{b} = \mathbf{a} \odot \mathbf{b}, \tag{A.30}$$

其中$\odot$表示点乘，即$(\mathbf{a} \odot \mathbf{b})_i = a_i b_i$。

对角矩阵 单位矩阵（identity matrix）是一种特殊的的对角矩阵，其主对角线元素为1，其余元素为0。n阶单位矩阵$\mathbf{I}_n$，是一个$n \times n$的方块矩阵。可以记为$\mathbf{I}_n = \text{diag}(1, 1, ..., 1)$。

一个$m \times n$的矩阵A和单位矩阵的乘积等于其本身。

$$A\mathbf{I}_n = \mathbf{I}_m A = A. \tag{A.31}$$

逆矩阵 对于一个$n \times n$的方块矩阵A，如果存在另一个方块矩阵B使得

$$AB = BA = \mathbf{I}_n \tag{A.32}$$

为单位阵，则称A是可逆的。矩阵B称为矩阵A的逆矩阵（inverse matrix），记为A^{-1}。

一个方阵的行列式等于0当且仅当该方阵不可逆。

正定矩阵 对于一个$n \times n$的对称矩阵A，如果对于所有的非零向量$\mathbf{x} \in \mathbb{R}^n$都满足

$$\mathbf{x}^{\mathrm{T}} Ax > 0, \tag{A.33}$$

则A为正定矩阵（positive-definite matrix）。如果$\mathbf{x}^{\mathrm{T}} Ax \geq 0$，则$A$是半正定矩阵（positive-semidefinite matrix）。

正交矩阵 正交矩阵（orthogonal matrix ）A为一个方块矩阵，其逆矩阵等于其转置矩阵。

$$A^{\mathrm{T}} = A^{-1}, \tag{A.34}$$

等价于$A^{\mathrm{T}} A = AA^{\mathrm{T}} = \mathbf{I}_n$。

Gram矩阵 向量空间中一组向量$\mathbf{v}_1, \mathbf{v}_2 \cdots, \mathbf{v}_n$的*Gram*矩阵（Gram matrix）$G$是内积的对称矩阵，其元素$G_{ij}$为$\mathbf{v}_i^{\mathrm{T}} \mathbf{v}_j$。

A.2.4 特征值与特征矢量

如果一个标量λ和一个非零向量$\mathbf{v}$满足

$$A\mathbf{v} = \lambda \mathbf{v}, \tag{A.35}$$

则λ和$\mathbf{v}$分别称为矩阵A的特征值（eigenvalue）和特征向量（eigenvector）。

A.2.5 矩阵分解

一个矩阵通常可以用一些比较“简单”的矩阵来表示，称为矩阵分解（matrix decomposition, matrix factorization）。

奇异值分解 一个$m \times n$的矩阵A的奇异值分解（Singular Value Decomposition，SVD）定义为

$$A = U\Sigma V^{\mathrm{T}}, \tag{A.36}$$

其中U和V分别为$m \times m$和$n\times$的正交矩阵，Σ为$m \times n$的对角矩阵，其对角线上的元素称为奇异值（singular value）。

特征分解 一个$n \times n$的方块矩阵A的特征分解（Eigendecomposition）定义为

$$A = Q\Lambda Q^{-1}, \tag{A.37}$$

其中Q为$n \times n$的方块矩阵，其每一列都为A的特征向量，为对角阵，其每一个对角元素为A的特征值。

如果A为对称矩阵，则A可以被分解为

$$A = Q\Lambda Q^{\mathrm{T}}, \tag{A.38}$$

其中Q为正交阵。

B 微积分

B.1 导数

导数（derivative）是微积分学中重要的基础概念。

对于定义域和值域都是实数域的函数$f : \mathbb{R} \to \mathbb{R}$，若$f(x)$在点$x_0$的某个邻域$\Delta x$内，极限

$$f'(x_0) = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} \tag{B.1}$$

存在，则称函数$f(x)$在点x_0处可导，$f'(x_0)$称为其导数，或导函数。

若函数 $f(x)$ 在其定义域包含的某区间内每一个点都可导，那么也可以说函数 $f(x)$ 在这个区间内可导。连续函数不一定可导，可导函数一定连续。例如函数 $|x|$ 为连续函数，但在点 $x=0$ 处不可导。

对于一个多变量函数 $f:\mathbb{R}^d \to \mathbb{R}$，它的偏导数（partial derivative ）是关于其中一个变量 x_i 的导数，而保持其他变量固定，可以记为 $f'_{x_i}(\mathbf{x})$，$\nabla_{x_i} f(\mathbf{x})$，$\frac{\partial f(\mathbf{x})}{\partial x_i}$ 或 $\frac{\partial}{\partial x_i} f(\mathbf{x})$。

对于一个 d 维向量 $\mathbf{x} \in \mathbb{R}^d$，函数 $f(\mathbf{x}) = f(x_1, \cdots, x_d) \in \mathbb{R}$，则 $f(\mathbf{x})$ 关于 $\mathbf{x}$ 的偏导数为

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} \\ \vdots \\ \frac{\partial f(\mathbf{x})}{\partial x_d} \end{bmatrix} \in \mathbb{R}^d. \tag{B.2}$$

若函数 $f(\mathbf{x}) \in \mathbb{R}^k$ 的值也为一个向量，则 $f(\mathbf{x})$ 关于 $\mathbf{x}$ 的偏导数为

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \cdots & \frac{\partial f_k(\mathbf{x})}{\partial x_1} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_1(\mathbf{x})}{\partial x_d} & \cdots & \frac{\partial f_k(\mathbf{x})}{\partial x_d} \end{bmatrix} \in \mathbb{R}^{d \times k}. \tag{B.3}$$

称为 *Jacobian* 矩阵。

B.1.1 导数法则

一个复杂函数的导数的计算可以通过以下法则来简化。

B.1.1.1 加（减）法则

$\mathbf{y} = f(\mathbf{x})$,$\mathbf{z} = g(\mathbf{x})$ 则

$$\frac{\partial(\mathbf{y}+\mathbf{z})}{\partial \mathbf{x}} = \frac{\partial \mathbf{y}}{\partial \mathbf{x}} + \frac{\partial \mathbf{z}}{\partial \mathbf{x}}. \tag{B.4}$$

B.1.1.2 乘法法则

（1）若 $\mathbf{x} \in \mathbb{R}^p$，$\mathbf{y} = f(\mathbf{x}) \in \mathbb{R}^q$，$\mathbf{z} = g(\mathbf{x}) \in \mathbb{R}^q$，则

$$\frac{\partial \mathbf{y}^{\mathrm{T}}\mathbf{z}}{\partial \mathbf{x}} = \frac{\partial \mathbf{y}}{\partial \mathbf{x}}\mathbf{z} + \frac{\partial \mathbf{z}}{\partial \mathbf{x}}\mathbf{y}. \tag{B.5}$$

（2）若 $\mathbf{x} \in \mathbb{R}^p$，$\mathbf{y} = f(\mathbf{x}) \in \mathbb{R}^s$，$\mathbf{z} = g(\mathbf{x}) \in \mathbb{R}^t$，$A \in \mathbb{R}^{s \times t}$ 和 $\mathbf{x}$ 无关，则

$$\frac{\partial \mathbf{y}^{\mathrm{T}} A\mathbf{z}}{\partial \mathbf{x}} = \frac{\partial \mathbf{y}}{\partial \mathbf{x}} A\mathbf{z} + \frac{\partial \mathbf{z}}{\partial \mathbf{x}} A^{\mathrm{T}}\mathbf{y}. \tag{B.6}$$

（3）若 $\mathbf{x} \in \mathbb{R}^p$，$y = f(\mathbf{x}) \in \mathbb{R}$，$\mathbf{z} = g(\mathbf{x}) \in \mathbb{R}^p$，则

$$\frac{\partial y\mathbf{z}}{\partial \mathbf{x}} = y\frac{\partial \mathbf{z}}{\partial \mathbf{x}} + \frac{\partial y}{\partial \mathbf{x}}\mathbf{z}^{\mathrm{T}}. \tag{B.7}$$

B.1.1.3 链式法则

链式法则（chain rule），是求复合函数导数的一个法则，是在微积分中计算导数的一种常用方法[1]。

（1）若 $\mathbf{x} \in \mathbb{R}^p$，$\mathbf{y} = g(\mathbf{x}) \in \mathbb{R}^s$，$\mathbf{z} = f(\mathbf{y}) \in \mathbb{R}^t$，则

$$\frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \frac{\partial \mathbf{y}}{\partial \mathbf{x}}\frac{\partial \mathbf{z}}{\partial \mathbf{y}}. \tag{B.8}$$

（2）若 $X \in \mathbb{R}^{p\times q}$ 为矩阵，$Y = g(X) \in \mathbb{R}^{s\times t}$，$z = f(Y) \in \mathbb{R}$，则

$$\frac{\partial z}{\partial X_{ij}} = \mathrm{tr}\left((\frac{\partial z}{\partial Y})^{\mathrm{T}}\frac{\partial Y}{\partial X_{ij}}\right). \tag{B.9}$$

（3）若 $X \in \mathbb{R}^{p\times q}$ 为矩阵，$\mathbf{y} = g(X) \in \mathbb{R}^s$，$z = f(\mathbf{y}) \in \mathbb{R}$，则

$$\frac{\partial z}{\partial X_{ij}} = (\frac{\partial z}{\partial \mathbf{y}})^{\mathrm{T}}\frac{\partial \mathbf{y}}{\partial X_{ij}}. \tag{B.10}$$

（4）若 $x \in \mathbb{R}$，$\mathbf{u} = u(x) \in \mathbb{R}^p$，$\mathbf{g} = g(\mathbf{u}) \in \mathbb{R}^q$，则

$$\frac{\partial \mathbf{g}}{\partial x} = (\frac{\partial \mathbf{g}}{\partial \mathbf{u}})\frac{\partial \mathbf{u}}{\partial x}. \tag{B.11}$$

B.2 常见函数的导数

这里我们介绍本书中常用的几个函数。

B.2.1 标量函数及其导数

指示函数 指示函数 $I(x = c)$ 为

$$I(x = c) = \begin{cases} 1 & \text{if} \quad x = c, \\ 0 & \text{else} \quad 0. \end{cases} \tag{B.12}$$

指示函数 $I(x = c)$ 除了在 c 外，其导数为 0。

[1] 详细的矩阵偏导数参考https://en.wikipedia.org/wiki/Matrix_calculus。

多项式函数 如果 $f(x) = x^r$，其中 r 是非零实数，那么导数

$$\frac{\partial x^r}{\partial x} = rx^{r-1}. \tag{B.13}$$

当 $r = 0$ 时，常函数的导数是 0。

指数函数 底数为 e 的指数函数 $f(x) = \exp(x) = e^x$ 的导数是它本身。

$$\frac{\partial \exp(x)}{\partial x} = \exp(x). \tag{B.14}$$

对数函数 底数为 e 对数函数 $\log(x)$ 的导数则是 x^{-1}。

$$\frac{\partial \log(x)}{\partial x} = \frac{1}{x}. \tag{B.15}$$

B.2.2 向量函数及其导数

$$\frac{\partial \mathbf{x}}{\partial \mathbf{x}} = I, \tag{B.16}$$

$$\frac{\partial A\mathbf{x}}{\partial \mathbf{x}} = A^{\mathrm{T}}, \tag{B.17}$$

$$\frac{\partial \mathbf{x}^{\mathrm{T}} A}{\partial \mathbf{x}} = A \tag{B.18}$$

B.2.3 按位计算的向量函数及其导数

假设一个函数 $f(x)$ 的输入是标量 x。对于一组 K 个标量 $x_1, \cdots, x_K$，我们可以通过 $f(x)$ 得到另外一组 K 个标量 $z_1, \cdots, z_K$，

$$z_k = f(x_k), \forall k = 1, \cdots, K \tag{B.19}$$

为了简便起见，我们定义 $\mathbf{x} = [x_1, \cdots, x_K]^{\mathrm{T}}$，$\mathbf{z} = [z_1, \cdots, z_K]^{\mathrm{T}}$，

$$\mathbf{z} = f(\mathbf{x}), \tag{B.20}$$

其中，$f(\mathbf{x})$ 是按位运算的，即 $[f(\mathbf{x})]_i = f(x_i)$。

当 x 为标量时，$f(x)$ 的导数记为 $f'(x)$。当输入为 K 维向量 $\mathbf{x} = [x_1, \cdots, x_K]^{\mathrm{T}}$ 时，其导数为一个对角矩阵。

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \Big[\frac{\partial f(x_j)}{\partial x_i}\Big]_{K\times K} \tag{B.21}$$

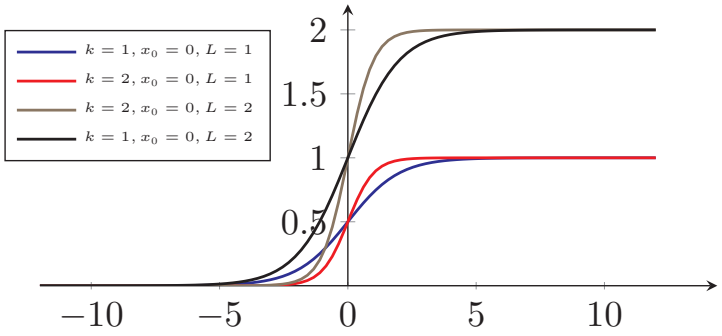


图 B.1 Logistic 函数

$$= \begin{bmatrix} f'(x_1) & 0 & \cdots & 0 \\ 0 & f'(x_2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & f'(x_K) \end{bmatrix} \tag{B.22}$$

$$= \text{diag}(f'(\mathbf{x})). \tag{B.23}$$

B.2.4 Logistic 函数

Logistic 函数是一种常用的S形函数，是比利时数学家 Pierre François Verhulst 在 1844-1845 年研究种群数量的增长模型时提出命名的，最初作为一种生态学模型。

Logistic 函数定义为：

$$\text{logistic}(x) = \frac{L}{1+\exp(-k(x-x_0))}, \tag{B.24}$$

这里 $\exp(\cdot)$ 函数表示自然对数，x_0 是中心点，L 是最大值，k 是曲线的倾斜度。图B.1给出了几种不同参数的 logistic 函数曲线。当 x 趋向于 $-\infty$ 时，$\text{logistic}(x)$ 接近于 0；当 x 趋向于 $+\infty$ 时，$\text{logistic}(x)$ 接近于 L.

当参数为 $(k=1, x_0=0, L=1)$ 时，logistic 函数称为标准 logistic 函数，记为 $\sigma(x)$。

$$\sigma(x) = \frac{1}{1+\exp(-x)}. \tag{B.25}$$

标准 logistic 函数在机器学习中使用得非常广泛，经常用来将一个实数空间的数映射到 $(0,1)$ 区间。

标准 logistic 函数的导数为

$$\sigma'(x) = \sigma(x)(1 - \sigma(x)) \tag{B.26}$$

当输入为 K 维向量 $\mathbf{x} = [x_1, \cdots, x_K]^{\mathrm{T}}$ 时，其导数为

$$\sigma'(\mathbf{x}) = \mathrm{diag}\big(\sigma(\mathbf{x}) \odot (1 - \sigma(\mathbf{x}))\big). \tag{B.27}$$

B.2.5 softmax 函数

softmax 函数是将多个标量映射为一个概率分布。

对于 K 个标量 $x_1, \cdots, x_K$，softmax 函数定义为

$$z_k = \mathrm{softmax}(x_k) = \frac{\exp(x_k)}{\sum_{i=1}^{K} \exp(x_i)}, \tag{B.28}$$

这样，我们可以将 K 个变量 $x_1, \cdots, x_K$ 转换为一个分布：$z_1, \cdots, z_K$，满足

$$z_k \in [0, 1], \forall k, \qquad \sum_{i=1}^{K} z_k = 1. \tag{B.29}$$

当 softmax 函数的输入为 K 维向量 $\mathbf{x}$ 时，

$$\hat{\mathbf{z}} = \mathrm{softmax}(\mathbf{x}) \tag{B.30}$$

$$= \frac{1}{\sum_{k=1}^{K} \exp(x_k)} \begin{bmatrix} \exp(x_1) \\ \vdots \\ \exp(x_K) \end{bmatrix} \tag{B.31}$$

$$= \frac{\exp(\mathbf{x})}{\sum_{k=1}^{K} \exp(x_k)} \tag{B.32}$$

$$= \frac{\exp(\mathbf{x})}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})}, \tag{B.33}$$

其中，$\mathbf{1}_K = [1, \cdots, 1]_{K \times 1}$ 是 K 维的全 1 向量。

其导数为

$$\frac{\partial \operatorname{softmax}(\mathbf{x})}{\partial \mathbf{x}} = \frac{\partial \left(\frac{\exp(\mathbf{x})}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})}\right)}{\partial \mathbf{x}} \tag{B.34}$$

$$= \frac{1}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})} \frac{\partial \exp(\mathbf{x})}{\partial \mathbf{x}} + \frac{\partial \left(\frac{1}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})}\right)}{\partial \mathbf{x}} (\exp(\mathbf{x}))^{\mathrm{T}} \tag{B.35}$$

$$= \frac{\operatorname{diag}(\exp(\mathbf{x}))}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})} - \left(\frac{1}{(\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x}))^2}\right) \frac{\partial \left(\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})\right)}{\partial \mathbf{x}} (\exp(\mathbf{x}))^{\mathrm{T}} \tag{B.36}$$

$$= \frac{\operatorname{diag}(\exp(\mathbf{x}))}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})} - \left(\frac{1}{(\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x}))^2}\right) \operatorname{diag}(\exp(\mathbf{x})) \mathbf{1}_K (\exp(\mathbf{x}))^{\mathrm{T}} \tag{B.37}$$

$$= \frac{\operatorname{diag}(\exp(\mathbf{x}))}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})} - \left(\frac{1}{(\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x}))^2}\right) \exp(\mathbf{x}) (\exp(\mathbf{x}))^{\mathrm{T}} \tag{B.38}$$

$$= \operatorname{diag}\left(\frac{\exp(\mathbf{x})}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})}\right) - \frac{\exp(\mathbf{x})}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})} \cdot \frac{(\exp(\mathbf{x}))^{\mathrm{T}}}{\mathbf{1}_K^{\mathrm{T}} \exp(\mathbf{x})} \tag{B.39}$$

$$= \operatorname{diag}(\operatorname{softmax}(\mathbf{x})) - \operatorname{softmax}(\mathbf{x}) \operatorname{softmax}(\mathbf{x})^{\mathrm{T}}. \tag{B.40}$$

C 数学优化

数学优化（Mathematical Optimization）问题，也叫最优化问题，是指在一定约束条件下，求解一个目标函数的最大值（或最小值）问题。

数学优化问题的定义为：给定一个目标函数（也叫代价函数）$f : \mathcal{A} \to \mathbb{R}$，寻找一个变量（也叫参数）$\mathbf{x}^* \in \mathcal{D}$，使得对于所有 $\mathcal{D}$ 中的 $\mathbf{x}$，$f(\mathbf{x}^*) \leq f(\mathbf{x})$（最小化）；或者 $f(\mathbf{x}^*) \geq f(\mathbf{x})$（最大化），其中 $\mathcal{D}$ 为变量 $\mathbf{x}$ 的约束集，也叫可行域；$\mathcal{D}$ 中的变量被称为是可行解。

C.1 数学优化的类型

C.1.1 离散优化和连续优化

根据输入变量 $\mathbf{x}$ 的值域是否为实数域，数学优化问题可以分为离散优化问题和连续优化问题。

C.1.1.1 离散优化问题

离散优化（Discrete Optimization）问题是目标函数的输入变量为离散变量，比如为整数或有限集合中的元素。离散优化问题主要有两个分支：

1. 组合优化（Combinatorial Optimization）：其目标是从一个有限集合中找出使得目标函数最优的元素。在一般的组合优化问题中，集合中的元素之间存在一定的关联，可以表示为图结构。典型的组合优化问题有旅行商问题、最小生成树问题、图着色问题等。很多机器学习问题都是组合优化问题，比如特征选择、聚类问题、超参数优化问题以及结构化学习（Structured Learning）中标签预测问题等。
2. 整数规划（Integer Programming）：输入变量$\mathbf{x} \in \mathbb{Z}^d$为整数。一般常见的整数规划问题为整数线性规划（Integer Linear Programming，ILP）。整数线性规划的一种最直接的求解方法是：1）去掉输入为整数的限制，得到一个就成为一个一般的线性规划问题，这个线性规划问题为原整数线性规划问题的松弛问题；2）求得相应松弛问题的解；3）把松弛问题的解四舍五入到最接近的整数。但是这种方法得到的解一般都不是最优的，因此原问题的最优解不一定在松弛问题最优解的附近。另外，这种方法得到的解也不一定满足约束条件。

离散优化问题的求解一般都比较困难，优化算法的复杂度都比较高。

C.1.1.2 连续优化问题

连续优化（Continuous Optimization）问题是目标函数的输入变量为连续变量$\mathbf{x} \in \mathbb{R}^d$，即目标函数为实函数。本节后面的内容主要以连续优化为主。

C.1.2 无约束优化和约束优化

在连续优化问题中，根据是否有变量的约束条件，可以将优化问题分为无约束优化问题和约束优化问题。

无约束优化问题（Unconstrained Optimization）的可行域为整个实数域$\mathcal{D} = \mathbb{R}^d$，可以写为

$$\min_{\mathbf{x}} \quad f(\mathbf{x}) \tag{C.1}$$

其中$\mathbf{x} \in \mathbb{R}^d$为输入变量，$f : \mathbb{R}^d \to \mathbb{R}$为目标函数。

最优化问题一般可以表示为求最小值问题。求$f(\mathbf{x})$最大值等价于求$-f(\mathbf{x})$的最小值。

约束优化问题（Constrained Optimization）中变量$\mathbf{x}$需要满足一些等式或不等式的约束。约束优化问题通常使用拉格朗日乘数法来进行求解。

拉格朗日乘数法参见第C.3节。

C.1.3 线性优化和非线性优化

如果在公式(C.1)中，目标函数和所有的约束函数都为线性函数，则该问题为*线性规划问题*（Linear Programming）。相反，如果目标函数或任何一个约束函数为非线性函数，则该问题为*非线性规划问题*（Nonlinear Programming）。

在非线性优化问题中，有一类比较特殊的问题是*凸优化问题*（Convex Programming）。在凸优化问题中，变量$\mathbf{x}$的可行域为凸集，即对于集合中任意两点，它们的连线全部位于在集合内部。目标函数f也必须为凸函数，即满足

$$f(\alpha\mathbf{x}+(1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x})+(1-\alpha)f(\mathbf{y}), \forall \alpha \in [0,1]. \tag{C.2}$$

凸优化问题是一种特殊的约束优化问题，需满足目标函数为凸函数，并且等式约束函数为线性函数，不等式约束函数为凹函数。

C.2 优化算法

优化问题一般都是通过迭代的方式来求解：通过猜测一个初始的估计$\mathbf{x}_0$，然后不断迭代产生新的估计$\mathbf{x}_1, \mathbf{x}_2, \cdots \mathbf{x}_t$，希望$\mathbf{x}_t$最终收敛到期望的最优解$\mathbf{x}^*$。一个好的优化算法应该是在一定的时间或空间复杂度下能够快速准确地找到最优解。同时，好的优化算法受初始猜测点的影响较小，通过迭代能稳定地找到最优解$\mathbf{x}^*$的邻域，然后迅速收敛于$\mathbf{x}^*$。

优化算法中常用的迭代方法有线性搜索和置信域方法等。线性搜索的策略是寻找方向和步长，具体算法有梯度下降法、牛顿法、共轭梯度法等。

本书中只介绍梯度下降法。

C.2.0.1 全局最优和局部最优

对于很多非线性优化问题，会存在若干个局部的极小值。*局部最小值*，或*局部最优解*$\mathbf{x}^*$定义为：存在一个$\delta > 0$，对于所有的满足$\|\mathbf{x}-\mathbf{x}^*\| \leq \delta$的$\mathbf{x}$，公式$f(\mathbf{x}^*) \leq f(\mathbf{x})$成立。也就是说，在$\mathbf{x}^*$的附近区域内，所有的函数值都大于或者等于$f(\mathbf{x}^*)$。

对于所有的$\mathbf{x} \in A$，都有$f(\mathbf{x}^*) \leq f(\mathbf{x})$成立，则$\mathbf{x}^*$为*全局最小值*，或*全局最优解*。

一般的，求局部最优解是容易的，但很难保证其为全局最优解。对于线性规划或凸优化问题，局部最优解就是全局最优解。

要确认一个点$\mathbf{x}^*$是否为局部最优解，通过比较它的邻域内有没有更小的函数值是不现实的。如果函数$f(\mathbf{x})$是二次连续可微的，我们可以通过检查目标函数在点$\mathbf{x}^*$的梯度$\nabla f(\mathbf{x}^*)$和Hessian矩阵$\nabla^2 f(\mathbf{x}^*)$来判断。

定理 C.1–局部最小值的一阶必要条件： 如果 $\mathbf{x}^*$ 为局部最优解并且函数 f 在 $\mathbf{x}^*$ 的邻域内一阶可微，则在 $\nabla f(\mathbf{x}^*) = 0$。

证明. 如果函数 $f(\mathbf{x})$ 是连续可微的，根据泰勒展开公式（Taylor's Formula），函数 $f(\mathbf{x})$ 的一阶展开可以近似为

$$f(\mathbf{x}^* + \triangle\mathbf{x}) = f(\mathbf{x}^*) + \triangle\mathbf{x}^{\mathrm{T}}\nabla f(\mathbf{x}^*), \tag{C.3}$$

假设 $\nabla f(\mathbf{x}^*) \neq 0$，则可以找到一个 $\triangle\mathbf{x}$（比如 $\triangle\mathbf{x} = -\alpha\nabla f(\mathbf{x}^*)$，$\alpha$ 为很小的正数），使得

$$f(\mathbf{x}^* + \triangle\mathbf{x}) - f(\mathbf{x}^*) = \triangle\mathbf{x}^{\mathrm{T}}\nabla f(\mathbf{x}^*) \leq 0. \tag{C.4}$$

这和局部最优的定义矛盾。 □

定理 C.2–局部最优解的二阶必要条件： 如果 $\mathbf{x}^*$ 为局部最优解并且函数 f 在 $\mathbf{x}^*$ 的邻域内二阶可微，则在 $\nabla f(\mathbf{x}^*) = 0$，$\nabla^2 f(\mathbf{x}^*)$ 为半正定矩阵。

证明. 如果函数 $f(\mathbf{x})$ 是二次连续可微的，函数 $f(\mathbf{x})$ 的二阶展开可以近似为

$$f(\mathbf{x}^* + \triangle\mathbf{x}) = f(\mathbf{x}^*) + \triangle\mathbf{x}^{\mathrm{T}}\nabla f(\mathbf{x}^*) + \frac{1}{2}\triangle\mathbf{x}^{\mathrm{T}}(\nabla^2 f(\mathbf{x}^*))\triangle\mathbf{x}. \tag{C.5}$$

由一阶必要性定理可知 $\nabla f(\mathbf{x}^*) = 0$，则

$$f(\mathbf{x}^* + \triangle\mathbf{x}) - f(\mathbf{x}^*) = \frac{1}{2}\triangle\mathbf{x}^{\mathrm{T}}(\nabla^2 f(\mathbf{x}^*))\triangle\mathbf{x} \geq 0. \tag{C.6}$$

即 $\nabla^2 f(\mathbf{x}^*)$ 为半正定矩阵。 □

C.2.0.2 梯度下降法

梯度下降法（Gradient Descent Method），也叫最速下降法（Steepest Descend Method），经常用来求解无约束优化的极小值问题。

对于函数 $f(\mathbf{x})$，如果 $f(\mathbf{x})$ 在点 $\mathbf{x}_t$ 附近是连续可微的，那么 $f(\mathbf{x})$ 下降最快的方向是 $f(\mathbf{x})$ 在 $\mathbf{x}_t$ 点的梯度方法的反方向。

根据泰勒一阶展开公式，

$$f(\mathbf{x}_{t+1}) = f(\mathbf{x}_t + \triangle\mathbf{x}) \approx f(\mathbf{x}_t) + \triangle\mathbf{x}^{\mathrm{T}}\nabla f(\mathbf{x}_t). \tag{C.7}$$

要使得 $f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t)$，就得使 $\triangle\mathbf{x}^{\mathrm{T}}\nabla f(\mathbf{x}_t) < 0$。我们取 $\triangle\mathbf{x} = -\alpha\nabla f(\mathbf{x}_t)$。如果 $\alpha > 0$ 为一个够小数值时，那么 $f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t)$ 成立。

这样我们就可以从一个初始值 $\mathbf{x}_0$ 出发，通过迭代公式

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha_t\nabla f(x_t),\ t \geq 0. \tag{C.8}$$

生成序列 $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$ 使得

$$f(\mathbf{x}_0) \geq f(\mathbf{x}_1) \geq f(\mathbf{x}_2) \geq \cdots \tag{C.9}$$

如果顺利的话，序列 $(\mathbf{x}_n)$ 收敛到局部最优解 $\mathbf{x}^*$。注意每次迭代步长 α 可以改变，但其取值必须合适，如果过大就不会收敛，如果过小则收敛速度太慢。

梯度下降法的过程如图C.1所示。曲线是等高线（水平集），即函数 f 为不同常数的集合构成的曲线。红色的箭头指向该点梯度的反方向（梯度方向与通过该点的等高线垂直）。沿着梯度下降方向，将最终到达函数 f 值的局部最优解。

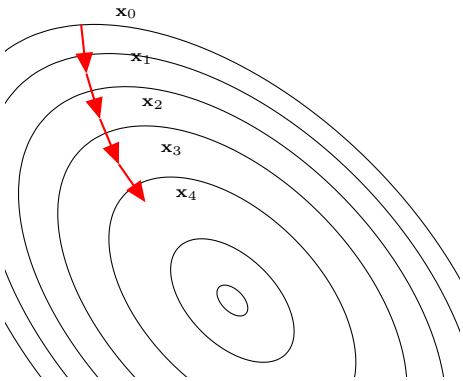


图 C.1 梯度下降法

梯度下降法为一阶收敛算法，当靠近极小值时梯度变小，收敛速度会变慢，并且可能以“之字形”的方式下降。如果目标函数为二阶连续可微，我们可以采用牛顿法。*牛顿法*为二阶收敛算法，收敛速度更快，但是每次迭代需要计算Hessian矩阵的逆矩阵，复杂较高。

相反，如果我们要求解一个最大值问题，就需要向梯度正方向迭代进行搜索，逐渐接近函数的局部极大值点，这个过程则被称为*梯度上升法*（gradient ascent）。

C.3 拉格朗日乘数法与 KKT 条件

拉格朗日乘数法（Lagrange Multiplier）是约束优化问题的一种有效求解方法。约束优化问题可以表示为

以数学家约瑟夫·拉格朗日命名。

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & h_i(\mathbf{x}) = 0, \quad i = 1, \ldots, m \\ & g_j(\mathbf{x}) \leq 0, \quad j = 1, \ldots, n \end{aligned} \tag{C.10}$$

其中 $h_i(\mathbf{x})$ 为等式约束函数，$g_j(\mathbf{x})$ 为不等式约束函数。$\mathbf{x}$ 的可行域为

$$\mathcal{D} = \mathbf{dom} f \cap \bigcap_{i=1}^{m} \mathbf{dom} h_i \cap \bigcap_{j=1}^{n} \mathbf{dom} g_j \subseteq \mathbb{R}^d, \tag{C.11}$$

其中 $\mathbf{dom} f$ 是函数 f 的定义域。

C.3.1 等式约束优化问题

如果公式 (C.10) 中只有等式约束，我们可以构造一个拉格朗日函数 $\Lambda(\mathbf{x}, \lambda)$

$$\Lambda(\mathbf{x}, \lambda) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i h_i(\mathbf{x}), \tag{C.12}$$

其中 λ 为拉格朗日乘数，可以是正数或负数。如果 $f(\mathbf{x}^*)$ 是原始约束优化问题的局部最优值，那么存在一个 λ^* 使得 $(\mathbf{x}^*, \lambda^*)$ 为拉格朗日函数 $\Lambda(\mathbf{x}, \lambda)$ 的平稳点（stationary point）。因此，只需要令 $\frac{\partial \Lambda(\mathbf{x}, \lambda)}{\partial \mathbf{x}} = 0$ 和 $\frac{\partial \Lambda(\mathbf{x}, \lambda)}{\partial \lambda} = 0$，得到

平稳点是指一阶偏导数为0的点。平稳点不一定为极值点。

$$\nabla f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \nabla h_i(\mathbf{x}) = 0, \tag{C.13}$$

$$h_i(\mathbf{x}) = 0, \qquad i = 0, \cdots, m \tag{C.14}$$

上面方程组的解即为原始问题的可能解。在实际应用中，需根据问题来验证是否为极值点。

拉格朗日乘数法是将一个有 d 个变量和 m 个等式约束条件的最优化问题转换为一个有 $d + m$ 个变量的函数求平稳点的问题。拉格朗日乘数法所得的平稳点会包含原问题的所有极值点，但并不保证每个平稳点都是原问题的极值点。

C.3.2 不等式约束优化问题

对于公式 (C.10) 中定义的一般约束优化问题，其拉格朗日函数为

$$\Lambda(\mathbf{x}, \mathbf{a}, \mathbf{b}) = f(\mathbf{x}) + \sum_{i=1}^{m} a_i h_i(\mathbf{x}) + \sum_{j=1}^{n} b_j g_j(\mathbf{x}), \tag{C.15}$$

其中 $\mathbf{a} = [a_1, \cdots, a_m]^{\mathrm{T}}$ 为等式约束的拉格朗日乘数，$\mathbf{b} = [b_1, \cdots, b_n]^{\mathrm{T}}$ 为不等式约束的拉格朗日乘数。

不等式约束优化问题中的拉格朗日乘数也称为KKT乘数。

当约束条件不满足时，有 $\max_{\mathbf{a},\mathbf{b}} \Lambda(\mathbf{x},\mathbf{a},\mathbf{b}) = \infty$；当约束条件满足时并且 $\mathbf{b} \geq 0$ 时，$\max_{\mathbf{a},\mathbf{b}} \Lambda(\mathbf{x},\mathbf{a},\mathbf{b}) = f(\mathbf{x})$。因此原始约束优化问题等价于

$$\min_{\mathbf{x}} \max_{\mathbf{a},\mathbf{b}} \quad \Lambda(\mathbf{x},\mathbf{a},\mathbf{b}), \tag{C.16}$$

$$\text{subject to} \quad \mathbf{b} \geq 0, \tag{C.17}$$

这个 min-max 优化问题称为主问题（primal problem）。

对偶问题 主问题的优化一般比较困难，我们可以通过交换 min-max 的顺序来简化。定义拉格朗日对偶函数为

$$\Gamma(\mathbf{a},\mathbf{b}) = \inf_{\mathbf{x}\in\mathcal{D}} \Lambda(\mathbf{x},\mathbf{a},\mathbf{b}). \tag{C.18}$$

$\Gamma(\mathbf{a},\mathbf{b})$ 是一个凹函数，即使 $f(\mathbf{x})$ 是非凸的。

当 $\mathbf{b} \geq 0$ 时，对于任意的 $\tilde{\mathbf{x}} \in \mathcal{D}$，有

$$\Gamma(\mathbf{a},\mathbf{b}) = \inf_{\mathbf{x}\in\mathcal{D}} \Lambda(\mathbf{x},\mathbf{a},\mathbf{b}) \leq \Lambda(\tilde{\mathbf{x}},\mathbf{a},\mathbf{b}) \leq f(\tilde{\mathbf{x}}), \tag{C.19}$$

令 p^* 是原问题的最优值，则有

$$\Gamma(\mathbf{a},\mathbf{b}) \leq p^*, \tag{C.20}$$

即拉格朗日对偶函数 $\Gamma(\mathbf{a},\mathbf{b})$ 为原问题最优值的下界。

优化拉格朗日对偶函数 $\Gamma(\mathbf{a},\mathbf{b})$ 并得到原问题的最优下界，称为拉格朗日对偶问题（Lagrange dual problem）。

$$\max_{\mathbf{a},\mathbf{b}} \quad \Gamma(\mathbf{a},\mathbf{b}), \tag{C.21}$$

$$\text{subject to} \quad \mathbf{b} \geq 0. \tag{C.22}$$

拉格朗日对偶函数为凹函数，因此拉格朗日对偶问题为凸优化问题。

令 d^* 是拉格朗日对偶问题的最优值，则有 $d^* \leq p^*$，这个性质称为弱对偶性（weak duality）。如果 $d^* = p^*$，这个性质称为强对偶性（strong duality）。

当强对偶性成立时，令 $\mathbf{x}^*$ 和 $\mathbf{a}^*, \mathbf{b}^*$ 分别是原问题问题和对偶问题的最优解，那么它们满足以下条件：

$$\nabla f(\mathbf{x}^*) + \sum_{i=1}^{m} a_i^* \nabla h_i(\mathbf{x}^*) + \sum_{j=1}^{n} b_j^* \nabla g_j(\mathbf{x}^*) = 0, \tag{C.23}$$

$$h_i(\mathbf{x}^*) = 0, \quad i = 0, \cdots, m \tag{C.24}$$

$$g_j(\mathbf{x}^*) \leq 0, \quad j = 0, \cdots, n \tag{C.25}$$

$$b_j^* g_j(\mathbf{x}^*) = 0, \quad j = 0, \cdots, n \tag{C.26}$$

$$b_j^* \geq 0, \qquad j = 0, \cdots, n \tag{C.27}$$

称为不等式约束优化问题的*KKT*条件（Karush-Kuhn-Tucker conditions）。KKT条件是拉格朗日乘数法在不等式约束优化问题上的泛化。当原问题是凸优化问题时，满足KKT条件的解也是原问题和对偶问题的最优解。

KKT条件中需要关注的是公式(C.26)，称为*互补松弛条件*（complementary slackness）。如果最优解$\mathbf{x}^*$出现在不等式约束的边界上$g_j(\mathbf{x}) = 0$，则$b_j^* > 0$；如果$\mathbf{x}^*$出现在不等式约束的内部$g_j(\mathbf{x}) < 0$，则$b_j^* = 0$。互补松弛条件说明当最优解出现在不等式约束的内部，则约束失效。

关于数学优化的内容，可以阅读《Numerical Optimization》[Nocedal and Wright, 2006]和《Convex Optimization》[Boyd and Vandenberghe, 2004]。

D 概率论

概率论主要研究大量随机现象中的数量规律，其应用十分广泛，几乎遍及各个领域。

D.1 样本空间

*样本空间*是一个随机试验所有可能结果的集合。例如，如果抛掷一枚硬币，那么样本空间就是集合{正面，反面}。如果投掷一个骰子，那么样本空间就是$\{1, 2, 3, 4, 5, 6\}$。随机试验中的每个可能结果称为*样本点*。

有些试验有两个或多个可能的样本空间。例如，从52张扑克牌中随机抽出一张，样本空间可以是*数字*（A到K），也可以是*花色*（黑桃，红桃，梅花，方块）。如果要完整地描述一张牌，就需要同时给出数字和花色，这时样本空间可以通过构建上述两个样本空间的*笛卡儿乘积*来得到。

D.2 事件和概率

随机事件（或简称*事件*）指的是一个被赋予概率的事物集合，也就是样本空间中的一个子集。*概率*表示一个随机事件发生的可能性大小，为0到1之间的一个非负实数。比如，一个0.5的概率表示一个事件有50%的可能性发生。

数学小知识 | 笛卡儿乘积

在数学中，两个集合 $\mathcal{X}$ 和 $\mathcal{Y}$ 的笛卡儿乘积（Cartesian product），又称直积，在集合论中表示为 $\mathcal{X} \times \mathcal{Y}$，是所有可能的有序对组成的集合，其中有序对的第一个对象是 X 的成员，第二个对象是 Y 的成员。

$$\mathcal{X} \times \mathcal{Y} = \{\langle x, y\rangle \mid x \in \mathcal{X} \wedge y \in \mathcal{Y}\}.$$

比如在扑克牌的例子中，如果集合 $\mathcal{X}$ 是 13 个元素的点数集合 {A, K, Q, J, 10, 9, 8, 7, 6, 5, 4, 3, 2}，而集合 $\mathcal{Y}$ 是 4 个元素的花色集合 {♠,♥,♦,♣}，则这两个集合的笛卡儿积是有 52 个元素的标准扑克牌的集合 {(A, ♠), (K, ♠), ..., (2, ♠), (A, ♥), ..., (3, ♣), (2, ♣)}。

对于一个机会均等的抛硬币动作来说，其样本空间为“正面”或“反面”。我们可以定义各个随机事件，并计算其概率。比如，

- {正面}，其概率为 0.5；
- {反面}，其概率为 0.5；
- 空集 ∅，不是正面也不是反面，其概率为 0；
- {正面 | 反面}，不是正面就是反面，其概率为 1。

D.2.1 随机变量

在随机试验中，试验的结果可以用一个数 X 来表示，这个数 X 是随着试验结果的不同而变化的，是样本点的一个函数。我们把这种数称为随机变量。例如，随机掷一个骰子，得到的点数就可以看成一个随机变量 X，X 的取值为 $\{1, 2, 3, 4, 5, 6\}$。

如果随机掷两个骰子，整个事件空间 Ω 可以由 36 个元素组成：

$$\Omega = \{(i, j) | i = 1, \ldots, 6; j = 1, \ldots, 6\} \tag{D.1}$$

一个随机事件也可以定义多个随机变量。比如在掷两个骰子的随机事件中，可以定义随机变量 X 为获得的两个骰子的点数和，也可以定义随机变量 Y 为获得的两个骰子的点数差。随机变量 X 可以有 11 个整数值，而随机变量 Y 只有 6 个。

$$X(i, j) := i + j, \quad x = 2, 3, \ldots, 12 \tag{D.2}$$

$$Y(i, j) := |i - j|, \quad y = 0, 1, 2, 3, 4, 5. \tag{D.3}$$

其中 i, j 分别为两个骰子的点数。

D.2.1.1 离散随机变量

如果随机变量 X 所可能取的值为有限可列举的，有 n 个有限取值

$$\{x_1, \cdots, x_n\},$$

则称 X 为离散随机变量。

一般用大写的字母表示一个随机变量，用小字字母表示该变量的某一个具体的取值。

要了解 X 的统计规律，就必须知道它取每种可能值 x_i 的概率，即

$$P(X = x_i) = p(x_i), \qquad \forall i \in [1, n]. \tag{D.4}$$

$p(x_1), \cdots, p(x_n)$ 称为离散型随机变量 X 的概率分布（probability distribution）或分布，并且满足

$$\sum_{i=1}^{n} p(x_i) = 1 \tag{D.5}$$

$$p(x_i) \geq 0, \qquad \forall i \in [1, n], \tag{D.6}$$

常见的离散随机变量的概率分布有：

伯努利分布 在一次试验中，事件 A 出现的概率为 μ，不出现的概率为 $1-\mu$。若用变量 X 表示事件 A 出现的次数，则 X 的取值为 0 和 1，其相应的分布为

$$p(x) = \mu^x (1-\mu)^{(1-x)}, \tag{D.7}$$

这个分布称为伯努利分布（Bernoulli Distribution），又名两点分布或者 0-1 分布。

二项分布 在 n 次伯努利分布中，若以变量 X 表示事件 A 出现的次数，则 X 的取值为 $\{0, \cdots, n\}$，其相应的分布为二项分布（Binomial Distribution）。

$$P(X = k) = \binom{n}{k} \mu^k (1-\mu)^{n-k}, \qquad k = 1 \cdots, n \tag{D.8}$$

其中 $\binom{n}{k}$ 为二项式系数（这就是二项分布的名称的由来），表示从 n 个元素中取出 k 个元素而不考虑其顺序的组合的总数。

D.2.1.2 连续随机变量

与离散随机变量不同，一些随机变量 X 的取值是不可列举的，由全部实数或者由一部分区间组成，比如

$$X = \{x | a \leq x \leq b\}, \quad -\infty < a < b < \infty$$

则称 X 为连续随机变量。连续随机变量的值是不可数及无穷尽的。

数学小知识 | 排列组合

排列组合是组合学最基本的概念。排列是指从给定个数的元素中取出指定个数的元素进行排序。组合则是指从给定个数的元素中仅仅取出指定个数的元素，不考虑排序。排列组合的中心问题是研究给定要求的排列和组合可能出现的情况总数。

排列的任务是确定 n 个不同的元素的排序的可能性。n 个不同的元素可以有 $n!$ 种不同的排列方式，即 n 的阶乘。

$$n! \triangleq n \times (n-1) \times \cdots \times 3 \times 2 \times 1.$$

如果从 n 个元素中取出 k 个元素，这 k 个元素的排列总数为

$$P_n^k \triangleq n \times (n-1) \times \cdots \times (n-k+1) = \frac{n!}{(n-k)!}.$$

从 n 个元素中取出 k 个元素，这 k 个元素可能出现的组合数为

$$C_n^k \triangleq \binom{n}{k} = \frac{P_n^k}{k!} = \frac{n!}{k!(n-k)!}.$$

区分排列与组合的关键是“有序”与“无序”。

对于连续随机变量 X，它取一个具体值 x_i 的概率为 0，这个离散随机变量截然不同。因此用列举连续随机变量取某个值的概率来描述这种随机变量不但做不到，也毫无意义。

连续随机变量 X 的概率分布一般用*概率密度函数*（probability density function，PDF）$p(x)$ 来描述。$p(x)$ 为可积函数，并满足

$$\int_{-\infty}^{+\infty} p(x)dx = 1 \tag{D.9}$$

$$p(x) \geq 0. \tag{D.10}$$

给定概率密度函数 $p(x)$，便可以计算出随机变量落入某一个区间的概率，而 $p(x)$ 本身反映了随机变量取落入 x 的非常小的邻近区间中的概率大小。

常见的连续随机变量的概率分布有：

均匀分布 若a, b为有限数，$[a, b]$上的均匀分布（uniform distribution）的概率密度函数定义为

$$p(x) = \begin{cases} \frac{1}{b-a} & , \quad a \leq x \leq b \\ 0 & , \quad x < a或x > b \end{cases} \tag{D.11}$$

正态分布 正态分布（Normal Distribution），又名高斯分布（Gaussian Distribution），是自然界最常见的一种分布，并且具有很多良好的性质，在很多领域都有非常重要的影响力，其概率密度函数为

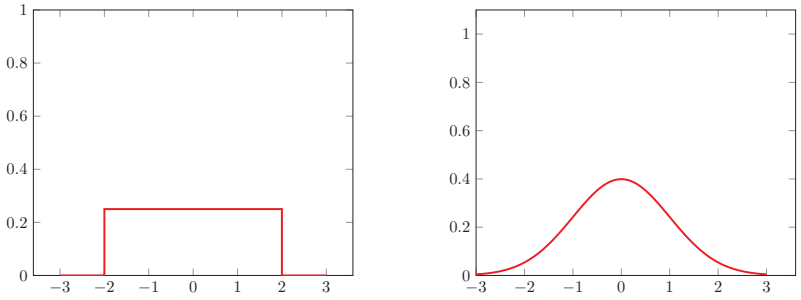
$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\Big(-\frac{(x-\mu)^2}{2\sigma^2}\Big), \tag{D.12}$$

其中，$\sigma > 0$，μ和σ均为常数。若随机变量X服从一个参数为μ和σ的概率分布，简记为

$$X \sim \mathcal{N}(\mu, \sigma^2). \tag{D.13}$$

当$\mu = 0$，$\sigma = 1$时，称为标准正态分布（Standard Normal Distribution）。

图D.1a和D.1b分别显示了均匀分布和正态分布的概率密度函数。



(a) 均匀分布 (b) 正态分布

图 D.1 连续随机变量的密度函数

D.2.1.3 累积分布函数

对于一个随机变量X，其累积分布函数（cumulative distribution function，CDF）是随机变量X的取值小于等于x的概率。

$$\text{cdf}(x) = P(X \leq x). \tag{D.14}$$

以连续随机变量X为例，累积分布函数定义为

$$\text{cdf}(x) = \int_{-\infty}^{x} p(t)\, dt, \tag{D.15}$$

其中 $p(x)$ 为概率密度函数。图D.2给出了标准正态分布的累计分布函数。

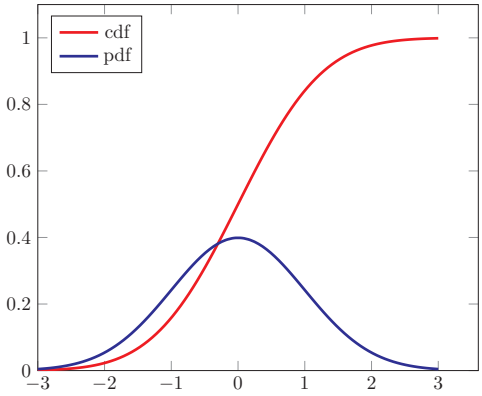


图 D.2 标准正态分布的概率密度函数和累计概率分布

D.2.2 随机向量

随机向量是指一组随机变量构成的向量。如果 $X_1, X_2, \cdots, X_n$ 为 n 个随机变量，那么称 $[X_1, X_2, \cdots, X_n]$ 为一个 n 维随机向量。一维随机向量称为随机变量。

随机向量也分为离散随机向量和连续随机向量。

D.2.2.1 离散随机向量

离散随机向量的联合概率分布（Joint Probability Distribution）为

$$P(X_1 = x_1, X_2 = x_2, \cdots, X_n = x_n) = p(x_1, x_2, \cdots, x_n),$$

其中 $x_i \in \omega_i$ 为变量 X_i 的取值，ω_i 为变量 X_i 的样本空间。

和离散随机变量类似，离散随机向量的概率分布满足

$$p(x_1, x_2, \cdots, x_n) \geq 0, \qquad \forall x_1 \in \omega_1, x_2 \in \omega_2, \cdots, x_n \in \omega_n \tag{D.16}$$

$$\sum_{x_1 \in \omega_1} \sum_{x_2 \in \omega_2} \cdots \sum_{x_n \in \omega_n} p(x_1, x_2, \cdots, x_n) = 1. \tag{D.17}$$

多项分布 一个常见的离散向量概率分布为多项分布（Multinomial Distribution）。多项分布是二项分布在随机向量的推广。假设一个袋子中装了很多球，总共有 K 个不同的颜色。我们从袋子中取出 n 个球。每次取出一个球时，就在袋子中放入一个同样颜色的球。这样保证同一颜色的球在不同试验中被取出的概率是相等的。令 $\mathbf{X}$ 为一个 K 维随机向量，每个元素 $X_k (k = 1, \cdots, K)$ 为取出

的 n 个球中颜色为 k 的球的数量，则 X 服从多项分布，其概率分布为

$$p(x_1,\dots,x_K|\boldsymbol{\mu}) = \frac{n!}{x_1!\cdots x_K!}\mu_1^{x_1}\cdots\mu_K^{x_K}, \tag{D.18}$$

其中 $\boldsymbol{\mu} = [\mu_1,\cdots,\mu_K]^{\mathrm{T}}$ 分别为每次抽取的球的颜色为 $1,\cdots,K$ 的概率；$x_1,\cdots,x_K$ 为非负整数，并且满足 $\sum_{k=1}^{K} x_k = n$。

多项分布的概率分布也可以用 gamma 函数表示：

$$p(x_1,\cdots,x_K|\boldsymbol{\mu}) = \frac{\Gamma(\sum_k x_k+1)}{\prod_k \Gamma(x_k+1)}\prod_{k=1}^{K}\mu_k^{x_k}, \tag{D.19}$$

其中 $\Gamma(z) = \int_0^{\infty}\frac{t^{z-1}}{\exp(t)}dt$ 为 gamma 函数。这种表示形式和 Dirichlet 分布类似，而 Dirichlet 分布可以作为多项分布的共轭先验。

D.2.2.2 连续随机向量

连续随机向量的其联合概率密度函数（Joint Probability Density Function）满足

$$p(\mathbf{x}) = p(x_1,\cdots,x_n) \geq 0, \tag{D.20}$$

$$\int_{-\infty}^{+\infty}\cdots\int_{-\infty}^{+\infty} p(x_1,\cdots,x_n)dx_1\cdots dx_n = 1. \tag{D.21}$$

多元正态分布 一个常见的连续随机向量分布为多元正态分布（Multivariate Normal Distribution），也称为多元高斯分布（Multivariate Gaussian Distribution）。若 n 维随机向量 $\mathbf{X} = [X_1,\dots,X_n]^{\mathrm{T}}$ 服从 n 元正态分布，其密度函数为

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}}\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right), \tag{D.22}$$

其中 $\boldsymbol{\mu}$ 为多元正态分布的均值向量，Σ 为多元正态分布的协方差矩阵，$|\Sigma|$ 表示 Σ 的行列式。

各项同性高斯分布 如果一个多元高斯分布的协方差矩阵简化为 $\Sigma = \sigma^2 I$，即每一个维随机变量都独立并且方差相同，那么这个多元高斯分布称为各项同性高斯分布（Isotropic Gaussian Distribution）。

Dirichlet 分布 一个 n 维随机向量 $\mathbf{X}$ 的 Dirichlet 分布为

$$p(\mathbf{x}|\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_1)\cdots\Gamma(\alpha_n)}\prod_{i=1}^{n} x_i^{\alpha_i-1}, \tag{D.23}$$

其中 $\boldsymbol{\alpha} = [\alpha_1,\dots,\alpha_K]^{\mathrm{T}}$ 为 Dirichlet 分布的参数。

D.2.3 边际分布

不失一般性，下面以二维随机向量进行讨论，这些结论在多维时依然成立。

对于二维离散随机向量(X,Y)，假设X取值空间为Ω_x，Y取值空间为Ω_y。其联合概率分布满足

$$p(x,y) \geq 0, \qquad \sum_{x\in\Omega_x}\sum_{y\in\Omega_y} p(x_i,y_j) = 1. \tag{D.24}$$

对于联合概率分布$p(x,y)$，我们可以分别对x和y进行求和。

（1）对于固定的x，

$$\sum_{y\in\Omega_y} p(x,y) = P(X=x) = p(x). \tag{D.25}$$

（2）对于固定的y，

$$\sum_{x\in\Omega_x} p(x,y) = P(Y=y) = p(y). \tag{D.26}$$

由离散随机向量(X,Y)的联合概率分布，对Y的所有取值进行求和得到X的概率分布；而对X的所有取值进行求和得到Y的概率分布。这里$p(x)$和$p(y)$就称为$p(x,y)$的边际分布（Marginal Distribution）。

对于二维连续随机向量(X,Y)，其边际分布为：

$$p(x) = \int_{-\infty}^{+\infty} p(x,y)dy \tag{D.27}$$

$$p(y) = \int_{-\infty}^{+\infty} p(x,y)dx \tag{D.28}$$

一个二元正态分布的边际分布仍为正态分布。

D.2.4 条件概率分布

对于离散随机向量(X,Y)，已知$X=x$的条件下，随机变量$Y=y$的条件概率（Conditional Probability）为：

$$p(y|x) = P(Y=y|X=x) = \frac{p(x,y)}{p(x)}. \tag{D.29}$$

这个公式定义了随机变量Y关于随机变量X的条件概率分布（Conditional Probability Distribution），简称条件分布。

对于二维连续随机向量(X,Y)，已知$X=x$的条件下，随机变量$Y=y$的条件概率密度函数（Conditional Probability Density Function）为

$$p(y|x)=\frac{p(x,y)}{p(x)}. \tag{D.30}$$

同理，已知$Y=y$的条件下，随机变量$X=x$的条件概率密度函数为

$$p(x|y)=\frac{p(x,y)}{p(y)}. \tag{D.31}$$

通过公式(D.30)和(D.31)，我们可以得到两个条件概率$p(y|x)$和$p(x|y)$之间的关系。

$$p(y|x)=\frac{p(x|y)p(y)}{p(x)}. \tag{D.32}$$

这个公式称为贝叶斯定理（Bayes' theorem），或贝叶斯公式。

D.2.5 独立与条件独立

对于两个离散（或连续）随机变量X和Y，如果其联合概率（或联合概率密度函数）$p(x,y)$满足

$$p(x,y)=p(x)p(y), \tag{D.33}$$

则称X和Y相互独立（independence），记为$X \perp\!\!\!\perp Y$。

对于三个离散（或连续）随机变量X、Y和Z，如果条件概率（或联合概率密度函数）$p(x,y|z)$满足

$$p(x,y|z)=P(X=x,Y=y|Z=z)=p(x|z)p(y|z), \tag{D.34}$$

则称在给定变量Z时，X和Y条件独立（conditional independence），记为$X \perp\!\!\!\perp Y|Z$。

D.2.6 期望和方差

期望 对于离散变量X，其概率分布为$p(x_1),\cdots,p(x_n)$，X的期望（expectation）或均值定义为

$$\mathbb{E}[X]=\sum_{i=1}^{n}x_i p(x_i). \tag{D.35}$$

对于连续随机变量X，概率密度函数为$p(x)$，其期望定义为

$$\mathbb{E}[X]=\int_{\mathbb{R}}xp(x)\,dx. \tag{D.36}$$

方差 随机变量 X 的*方差*（variance）用来定义它的概率分布的离散程度，定义为

$$\mathrm{var}(X) = \mathbb{E}\Big[\Big(X - \mathbb{E}[X]\Big)^2\Big]. \tag{D.37}$$

随机变量 X 的方差也称为它的*二阶矩*。$\sqrt{\mathrm{var}(X)}$ 则称为 X 的*根方差*或*标准差*。

协方差 两个连续随机变量 X 和 Y 的*协方差*（covariance）用来衡量两个随机变量的分布之间的总体变化性，定义为

$$\mathrm{cov}(X, Y) = \mathbb{E}\Big[\Big(X - \mathbb{E}[X]\Big)\Big(Y - \mathbb{E}[Y]\Big)\Big], \tag{D.38}$$

协方差经常也用来衡量两个随机变量之间的线性相关性。如果两个随机变量的协方差为 0，那么称这两个随机变量是*线性不相关*。两个随机变量之间没有线性相关性，并非表示它们之间独立的，可能存在某种非线性的函数关系。反之，如果 X 与 Y 是统计独立的，那么它们之间的协方差一定为 0。

这里的线性相关和线性代数中的线性相关含义不同。

协方差矩阵 两个 m 和 n 维的连续随机向量 $\mathbf{X}$ 和 $\mathbf{Y}$，它们的*协方差*（covariance）为 $m \times n$ 的矩阵，定义为

$$\mathrm{cov}(\mathbf{X}, \mathbf{Y}) = \mathbb{E}\Big[\Big(\mathbf{X} - \mathbb{E}[\mathbf{X}]\Big)\Big(\mathbf{Y} - \mathbb{E}[\mathbf{Y}]\Big)^{\mathrm{T}}\Big]. \tag{D.39}$$

协方差矩阵 $\mathrm{cov}(\mathbf{X}, \mathbf{Y})$ 的第 (i, j) 个元素等于随机变量 X_i 和 Y_j 的协方差。两个向量变量的协方差 $\mathrm{cov}(\mathbf{X}, \mathbf{Y})$ 与 $\mathrm{cov}(\mathbf{Y}, \mathbf{X})$ 互为转置关系。

如果两个随机向量的协方差矩阵为对角阵，那么称这两个随机向量是无关的。

单个随机向量 $\mathbf{X}$ 的协方差矩阵定义为

$$\mathrm{cov}(\mathbf{X}) = \mathrm{cov}(\mathbf{X}, \mathbf{X}). \tag{D.40}$$

D.2.6.1 Jensen 不等式

如果 X 是随机变量，g 是凸函数，则

$$g\left(\mathbb{E}[X]\right) \leq \mathbb{E}\left[g(X)\right]. \tag{D.41}$$

等式当且仅当 X 是一个常数或 g 是线性时成立。

D.2.6.2 大数定律

大数定律（law of large numbers）是指n个样本$X_1, \cdots, X_n$是独立同分布的，即$\mathbb{E}[X_1] = \cdots = \mathbb{E}[X_n] = \mu$，那么其均值

$$\bar{X}_n = \frac{1}{n}(X_1 + \cdots + X_n), \tag{D.42}$$

收敛于期望值μ。

$$\bar{X}_n \to \mu \qquad \text{for} \qquad n \to \infty \tag{D.43}$$

D.2.6.3 中心极限定理

D.2.7 指数族分布

D.3 随机过程

随机过程（stochastic process）是一组随机变量X_t的集合，其中t属于一个索引（index）集合$\mathcal{T}$。索引集合$\mathcal{T}$可以定义在时间域或者空间域，但一般为时间域，以实数或正数表示。当t为实数时，随机过程为连续随机过程；当t为整数时，为离散随机过程。日常生活中的很多例子包括股票的波动、语音信号、身高的变化等都可以看作是随机过程。常见的和时间相关的随机过程模型包括贝努力过程、随机游走、马尔可夫过程等。和空间相关的随机过程通常称为随机场（random field）。比如一张二维的图片，每个像素点（变量）通过空间的位置进行索引，这些像素就组成了一个随机过程。

D.3.1 马尔可夫过程

马尔可夫性质 在随机过程中，马尔可夫性质（Markov property）是指一个随机过程在给定现在状态及所有过去状态情况下，其未来状态的条件概率分布仅依赖于当前状态。以离散随机过程为例，假设随机变量$X_0, X_1, \cdots, X_T$构成一个随机过程。这些随机变量的所有可能取值的集合被称为状态空间（state space）。如果X_{t+1}对于过去状态的条件概率分布仅是X_t的一个函数，则

$$P(X_{t+1} = x_{t+1} | X_{0:t} = x_{0:t}) = P(X_{t+1} = x_{t+1} | X_t = x_t), \tag{D.44}$$

其中$X_{0:t}$表示变量集合$X_0, X_1, \cdots, X_t$，$x_{0:t}$为在状态空间中的状态序列。

马尔可夫性质也可以描述为给定当前状态时，将来的状态与过去状态是条件独立的。

D.3.1.1 马尔可夫链

离散时间的马尔可夫过程也称为马尔可夫链（Markov chain）。如果一个马尔可夫链的条件概率

$$P(X_{t+1} = s_i | X_t = s_j) = \mathbf{T}(s_i, s_j), \tag{D.45}$$

在不同时间都是不变的，即和时间t无关，则称为时间同质的马尔可夫链（time-homogeneous Markov chains）。如果状态空间是有限的，$T(s_i, s_j)$也可以用一个矩阵T表示，称为状态转移矩阵（transition matrix），其中元素t_{ij}表示状态s_i转移到状态s_j的概率。

平稳分布 假设状态空间大小为M，向量$\pi = [\pi_1, \cdots, \pi_M]^{\mathrm{T}}$为状态空间中的一个分布，满足$0 \leq \pi_i \leq 1$和$\sum_{i=1}^{M} \pi_i = 1$。

对于状态转移矩阵为$\mathbf{T}$的时间同质的马尔可夫链，如果存在一个分布π满足

$$\pi = \mathbf{T}\pi, \tag{D.46}$$

即分布π就称为该马尔可夫链的平稳分布（stationary distribution）。根据特征向量的定义可知，π为矩阵$\mathbf{T}$的（归一化）的对应特征值为1的特征向量。

如果一个马尔可夫链的状态转移矩阵$\mathbf{T}$满足所有状态可遍历性以及非周期性，那么对于任意一个初始状态分布$\pi^{(0)}$，将经过一定时间的状态转移之后，都会收敛到平稳分布，即

$$\pi = \lim_{N \to \infty} \mathbf{T}^N \pi^{(0)}. \tag{D.47}$$

定理 D.1 – 细致平稳条件（Detailed Balance Condition）： 如果一个马尔科夫链满足

$$\pi_i t_{ij} = \pi_j t_{ji}, \tag{D.48}$$

则一定会收敛到平稳分布π。

细致平稳条件保证了从状态i转移到状态j的数量和从状态j转移到状态i的数量相一致，相互抵消，所以数量不发生改变。

细致平稳条件只是马尔科夫链收敛的充分条件，不是必要条件。

D.3.2 高斯过程

高斯过程（Gaussian Process）也是一种应用广泛的随机过程模型。假设有一组连续随机变量 $X_0, X_1, \cdots, X_T$，如果由这组随机变量构成的任一有限集合

$$X_{t_1,\cdots,t_k} = [X_{t_1}, \cdots, X_{t_n}]^{\mathrm{T}}$$

都服从一个多元正态分布，那么这组随机变量为一个随机过程。高斯过程也可以定义为：如果 $X_{t_1,\cdots,t_n}$ 的任一线性组合都服从一元正态分布，那么这组随机变量为一个随机过程。

高斯过程回归 高斯过程回归（Gaussian process regression）是利用高斯过程来对函数分布进行直接建模。和机器学习中参数化建模（比如贝叶斯线性回归）相比，高斯过程是一种非参数模型，可以拟合一个黑盒函数，也可以给出拟合结果的置信度 [Rasmussen, 2004]。

假设函数 $f(\mathbf{x})$ 服从高斯过程，且为平滑函数，即如果两个样本点 $\mathbf{x}_1, \mathbf{x}_2$ 比较接近，那么对应的 $f(\mathbf{x}_1), f(\mathbf{x}_2)$ 也比较接近。两个样本点的距离可以用核函数来定义。函数 $f(x)$ 的的有限采样点服从一个多元正态分布，即

$$[f(\mathbf{x}_1), f(\mathbf{x}_2), \cdots, f(\mathbf{x}_n)]^{\mathrm{T}} \sim \mathcal{N}\Big(\mu(X), K(X,X)\Big), \tag{D.49}$$

其中 $X = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]$，$\mu(X) = [\mu(\mathbf{x}_1), \mu(\mathbf{x}_2), \cdots, \mu(\mathbf{x}_n)]^{\mathrm{T}}$ 是均值向量，$K(X,X) = [k(\mathbf{x}_i, \mathbf{x}_j)]_{n\times n}$ 是协方差矩阵，$k(\mathbf{x}_i, \mathbf{x}_j)$ 为衡量两个输入距离的核函数。一个常用的核函数是平方指数（squared exponential）函数

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2l^2}\right), \tag{D.50}$$

其中 l 为超参数。当 $\mathbf{x}_i$ 和 $\mathbf{x}_j$ 越接近，其核函数的值越大，表明 $f(\mathbf{x}_i)$ 和 $f(\mathbf{x}_j)$ 越相关。

在支持向量机中，平方指数核函数也叫高斯核函数或高斯我们将GP的均值设置为0——它们实际上已经足够强大，可以在不改变均值的情况下拟合各种函数。径向基函数。这里为了避免混淆，我们称为平方指数核函数。

假设 $f(\mathbf{x})$ 的一组带噪声的观测值为 $(\mathbf{x}_i, y_i)\quad 1 \le i \le n$，其中

$$y_i \sim \mathcal{N}(f(\mathbf{x}_i), \sigma^2),$$

为正态分布，σ 为噪声方差。

对于一个新的样本点 $\mathbf{x}^*$，$f(\mathbf{x}^*)$ 满足

$$\begin{bmatrix} \mathbf{y} \\ f(\mathbf{x}^*) \end{bmatrix} \sim N\left(\begin{bmatrix} \mu(X) \\ \mu(\mathbf{x}^*) \end{bmatrix}, \begin{bmatrix} K(X,X) + \sigma^2\mathbf{I} & K(\mathbf{x}^*, X)^{\mathrm{T}} \\ K(\mathbf{x}^*, X) & k(\mathbf{x}^*, \mathbf{x}^*) \end{bmatrix}\right), \tag{D.51}$$

其中 $\mathbf{y} = [y_1, y_2, \cdots, y_n]$，$K(\mathbf{x}^*, X) = [k(\mathbf{x}^*, \mathbf{x}_1), \cdots, k(\mathbf{x}^*, \mathbf{x}_n)]$。

根据上面的联合分布，$f(\mathbf{x}^*)$ 的后验分布满足

$$f(\mathbf{x}^*)|X,\mathbf{y} \sim \mathcal{N}(\hat{\mu},\hat{\sigma}^2), \tag{D.52}$$

其中均值 $\hat{\mu}$ 和方差 $\hat{\sigma}$ 为

$$\hat{\mu} = K(\mathbf{x}^*,X)(K(X,X)+\sigma^2\mathbf{I})^{-1}(\mathbf{y}-\mu(X))+\mu(\mathbf{x}^*), \tag{D.53}$$

$$\hat{\sigma}^2 = k(\mathbf{x}^*,\mathbf{x}^*)-K(\mathbf{x}^*,X)(K(X,X)+\sigma^2\mathbf{I})^{-1}K(\mathbf{x}^*,X)^{\mathrm{T}}. \tag{D.54}$$

从公式 (D.53) 可以看出，均值函数 $\mu(x)$ 可以近似地互相抵消。在实际应用中，一般假设均值函数为 0，均值 $\hat{\mu}$ 可以将简化为

$$\hat{\mu} = K(\mathbf{x}^*,X)(K(X,X)+\sigma^2\mathbf{I})^{-1}\mathbf{y}. \tag{D.55}$$

高斯过程回归可以认为是一种有效的贝叶斯优化方法，广泛地应用于机器学习中。

E 信息论

信息论（information theory）是数学、物理、统计、计算机科学等多个学科的交叉领域。信息论是由 Claude Shannon 最早提出的，主要研究信息的量化、存储和通信等方法。这里，“信息”是指一组消息的集合。假设在一个噪声通道上发送消息，我们需要考虑如何对每一个信息进行编码、传输以及解码，使得接收者可以尽可能准确地重构出消息。

Claude Shannon，1916 年 4 月 30 日－2001 年 2 月 26 日），美国数学家、电子工程师和密码学家，被誉为信息论的创始人。

在机器学习相关领域，信息论也有着大量的应用。比如特征抽取、统计推断、自然语言处理等。

E.1 熵

E.1.1 自信息和熵

熵（Entropy）最早是物理学的概念，用于表示一个热力学系统的无序程度。在信息论中，熵用来衡量一个随机事件的不确定性。假设对一个随机变量 X（取值集合为 $\mathcal{X}$，概率分布为 $p(x),x\in\mathcal{X}$）进行编码，自信息 $I(x)$ 是变量 $X=x$ 时

的信息量或编码长度，定义为

$$I(x) = -\log(p(x)), \tag{E.1}$$

那么随机变量X的平均编码长度，即熵定义为

$$\begin{aligned} H(X) &= \mathbb{E}_X[\mathrm{I}(x)] && \text{(E.2)} \\ &= \mathbb{E}_X[-\log(p(x))] && \text{(E.3)} \\ &= -\sum_{x\in\mathcal{X}} p(x)\log p(x), && \text{(E.4)} \end{aligned}$$

其中当$p(x_i)=0$时，我们定义$0\log 0=0$，这与极限一致，$\lim_{p\to 0+} p\log p = 0$。

在熵的定义中，对数的底可以使用2、自然常数e，或是10。

熵是一个随机变量的平均编码长度，即自信息的数学期望。熵越高，则随机变量的信息越多；熵越低，则信息越少。如果变量X当且仅当在x时$p(x)=1$，则熵为0。也就是说，对于一个确定的信息，其熵为0，信息量也为0。如果其概率分布为一个均匀分布，则熵最大。假设一个随机变量X有三种可能值x_1, x_2, x_3，不同概率分布对应的熵如下：

$p(x_1)$	$p(x_2)$	$p(x_3)$	熵
1	0	0	0
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{3}{2}$
$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	$\log(3)$

E.1.2 联合熵和条件熵

对于两个离散随机变量X和Y，假设X取值集合为$\mathcal{X}$；Y取值集合为$\mathcal{Y}$，其联合概率分布满足为$p(x,y)$，则

X和Y的联合熵（Joint Entropy）为

$$H(X,Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x,y). \tag{E.5}$$

X和Y的条件熵（Conditional Entropy）为

$$\begin{aligned} H(X|Y) &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log p(x|y) && \text{(E.6)} \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log \frac{p(x,y)}{p(y)}. && \text{(E.7)} \end{aligned}$$

根据其定义，条件熵也可以写为

$$H(X|Y) = H(X,Y) - H(Y). \tag{E.8}$$

E.2 互信息

互信息（mutual information）是衡量已知一个变量时，另一个变量不确定性的减少程度。两个离散随机变量 X 和 Y 的互信息定义为

$$I(X;Y) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log\frac{p(x,y)}{p(x)\,p(y)}. \tag{E.9}$$

互信息的一个性质为

$$I(X;Y) = H(X) - H(X|Y) \tag{E.10}$$

$$= H(Y) - H(Y|X). \tag{E.11}$$

如果 X 和 Y 相互独立，即 X 不对 Y 提供任何信息，反之亦然，因此它们的互信息为零。

E.3 交叉熵和散度

E.3.1 交叉熵

对应分布为 $p(x)$ 的随机变量，熵 $H(p)$ 表示其最优编码长度。*交叉熵*（cross entropy）是按照概率分布 q 的最优编码对真实分布为 p 的信息进行编码的长度，定义为

$$H(p,q) = \mathbb{E}_p[-\log q(x)] \tag{E.12}$$

$$= -\sum_x p(x)\,\log q(x). \tag{E.13}$$

在给定 p 的情况下，如果 q 和 p 越接近，交叉熵越小；如果 q 和 p 越远，交叉熵就越大。

E.3.2 KL 散度

KL 散度（Kullback-Leibler divergence），也叫 *KL 距离*或*相对熵* (relative entropy)，是用概率分布 q 来近似 p 时所造成的信息损失量。KL 散度是按照概率分布 q 的最优编码对真实分布为 p 的信息进行编码，其平均编码长度 $H(p,q)$ 和 p 的最优平均编码长度 $H(p)$ 之间的差异。对于离散概率分布 p 和 q，从 q 到 p 的 KL 散度定义为

$$D_{\mathrm{KL}}(p\|q) = H(p,q) - H(p) \tag{E.14}$$

$$= \sum_x p(x) \log \frac{p(x)}{q(x)}, \tag{E.15}$$

其中为了保证连续性，定义 $0 \log \frac{0}{0} = 0, 0 \log \frac{0}{q} = 0$。

KL 散度可以是衡量两个概率分布之间的距离。KL 散度总是非负的，$D_{\mathrm{KL}}(p\|q) \geq 0$。只有当 $p = q$ 时，$D_{\mathrm{KL}}(p\|q) = 0$。如果两个分布越接近，KL 散度越小；如果两个分布越远，KL 散度就越大。但 KL 散度并不是一个真正的度量或距离，一是 KL 散度不满足距离的对称性，二是 KL 散度不满足距离的三角不等式性质。

E.3.3 JS 散度

JS 散度（Jensen-Shannon divergence）是一种对称的衡量两个分布相似度的度量方式，定义为

$$D_{\mathrm{JS}}(p\|q) = \frac{1}{2} D_{\mathrm{KL}}(p\|m) + \frac{1}{2} D_{KL}(q\|m), \tag{E.16}$$

其中 $m = \frac{1}{2}(p + q)$。

JS 散度是 KL 散度一种改进。但两种散度有存在一个问题，即如果两个分布 p, q 个分布没有重叠或者重叠非常少时，KL 散度和 JS 散度都很难衡量两个分布的距离。

E.3.4 Wasserstein 距离

Wasserstein 距离（Wasserstein distance）也是用于衡量两个分布之间的距离。对于两个分布 q_1, q_2，p^{th}-Wasserstein 距离定义为

$$W_p(q_1, q_2) = \left(\inf_{\gamma(x,y) \in \Gamma(q_1, q_2)} \mathbb{E}_{(x,y)\sim\gamma(x,y)}[d(x,y)^p] \right)^{\frac{1}{p}}, \tag{E.17}$$

其中 $\Gamma(q_1, q_2)$ 是边际分布为 q_1 和 q_2 的所有可能的联合分布集合，$d(x, y)$ 为 x 和 y 的距离，比如 ℓ_p 距离等。

如果将两个分布看作是两个土堆，联合分布 $\gamma(x, y)$ 看作是从土堆 q_1 的位置 x 到土堆 q_2 的位置 y 的搬运土的数量，并有

$$\sum_x \gamma(x, y) = q_2(y), \tag{E.18}$$

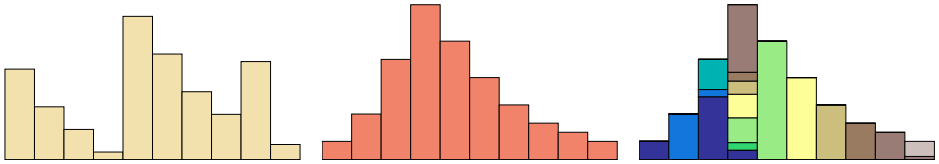
$$\sum_y \gamma(x, y) = q_1(x). \tag{E.19}$$

q_1 和 q_2 为 $\gamma(x, y)$ 的两个边际分布。

$\mathbb{E}_{(x,y)\sim\gamma(x,y)}[d(x,y)^p]$ 可以理解为在联合分布 $\gamma(x, y)$ 下把形状为 q_1 的土堆搬运到形状为 q_2 的土堆所需的工作量，

$$\mathbb{E}_{(x,y)\sim\gamma(x,y)}[d(x,y)^p] = \sum_{(x,y)} \gamma(x, y) d(x, y)^p, \tag{E.20}$$

其中从土堆 q_1 中的点 x 到土堆 q_2 中的点 y 的移动土的数量和距离分别为 $\gamma(x,y)$ 和 $d(x,y)^p$。因此，Wasserstein 距离可以理解为搬运土堆的最小工作量，也称为推土机距离（Earth-Mover's Distance，EMD）。图E.1给出了两个离散变量分布的 Wasserstein 距离示例。图E.1c中同颜色方块表示在分布 q_1 中为相同位置。



(a) $q_1(x)$ (b) $q_2(x)$ (c) q_1 到 q_2 最优的运输方案

图 E.1 Wasserstein 距离示例

Wasserstein 距离相比 KL 散度和 JS 散度的优势在于：即使两个分布没有重叠或者重叠非常少，Wasserstein 距离仍然能反映两个分布的远近。

对于 $\mathbb{R}^n$ 空间中的两个高斯分布 $p=\mathcal{N}(\mu_1,\Sigma_1)$ 和 $q=\mathcal{N}(\mu_2,\Sigma_2)$，它们的 2^{nd}-Wasserstein 距离为

$$D_{\text{W}}(p\|q)=\|\mu_1-\mu_2\|_2^2+\text{tr}\left(\Sigma_1+\Sigma_2-2\left(\Sigma_2^{\frac{1}{2}}\Sigma_1\Sigma_2^{\frac{1}{2}}\right)^{1/2}\right). \tag{E.21}$$

当两个分布的的方差为 0 时，2^{nd}-Wasserstein 距离等价与欧氏距离。

参考文献

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.