

Mathematics Notes
for
Computer Science
Information Technology

Hazer-BJTU

2024 / 2 / 16

目录

0.1	深度学习中的线性代数/概率论	4
0.1.1	多元函数微分	4
0.1.2	线性回归模型的解析解	5
0.1.3	SVD奇异值分解	6
0.1.4	拉格朗日乘子法与Karush-Kuhn-Tucker条件	7
0.2	算法/基础数学	8
0.2.1	离散傅里叶变换DFT与快速傅里叶变换FFT	8
0.2.2	傅里叶变换	12
0.3	算法竞赛中的数论与组合数学	15
0.3.1	容斥原理与二项式反演	15
0.3.2	容斥原理与二项式反演的实际应用举例	17

0.1 深度学习中的线性代数/概率论

0.1.1 多元函数微分

考虑定义在 \mathbb{R}^n 上的函数 f ，其输出为一个向量 $\mathbf{y} \in \mathbb{R}^m$ ，如果存在线性函数 L ，使得：

$$f(\mathbf{x} + \mathbf{h}) = f(\mathbf{x}) + L(\mathbf{h}) + O(\|\mathbf{h}\|_2)$$

其中线性函数 L 满足：

$$L(\mathbf{x} + \mathbf{y}) = L(\mathbf{x}) + L(\mathbf{y})$$

$$L(\lambda \cdot \mathbf{x}) = \lambda \cdot L(\mathbf{x}), \lambda \in \mathbb{R}$$

那么我们就认为该函数 f 是可微的，一般来说，我们可以将线性函数 L 简单理解为线性变换，如果我们限制函数 f 的输出为一个实数 $y \in \mathbb{R}$ ，则微分也可以被表示为如下形式：

$$f(\mathbf{x} + \mathbf{h}) = f(\mathbf{x}) + \mathbf{w}^\top \mathbf{h} + O(\|\mathbf{h}\|_2), \mathbf{w} \in \mathbb{R}^n$$

一个基本的事实是可微 \Rightarrow 偏导数存在，因为：

$$\begin{aligned} \frac{f(\mathbf{x} + \mathbf{h}_i) - f(\mathbf{x})}{\Delta \mathbf{x}_i} &= \frac{\mathbf{w}_i \cdot \Delta \mathbf{x}_i}{\Delta \mathbf{x}_i} + \frac{O(\Delta \mathbf{x}_i)}{\Delta \mathbf{x}_i} = \mathbf{w}_i + \frac{O(\Delta \mathbf{x}_i)}{\Delta \mathbf{x}_i} \\ \Rightarrow \lim_{\Delta \mathbf{x}_i \rightarrow 0} \frac{f(\mathbf{x} + \mathbf{h}_i) - f(\mathbf{x})}{\Delta \mathbf{x}_i} &= \mathbf{w}_i + \lim_{\Delta \mathbf{x}_i \rightarrow 0} \frac{O(\Delta \mathbf{x}_i)}{\Delta \mathbf{x}_i} = \mathbf{w}_i \\ \Rightarrow \frac{\partial f}{\partial \mathbf{x}_i} &= \mathbf{w}_i \end{aligned}$$

由此可见，实际上向量 \mathbf{w} 就是由函数 f 关于各分量的偏导数构成的：

$$\mathbf{w} = \left(\frac{\partial f}{\partial \mathbf{x}_1}, \frac{\partial f}{\partial \mathbf{x}_2}, \frac{\partial f}{\partial \mathbf{x}_3}, \dots, \frac{\partial f}{\partial \mathbf{x}_n} \right)^\top$$

定义对于向量 $\mathbf{x} \in \mathbb{R}^n$ ： $d\mathbf{x} = (dx_1, dx_2, dx_3, \dots, dx_n)$ ，则根据全微分公式可以得出如下关系：

$$d\mathbf{x}^\top \mathbf{x} = 2\mathbf{x}^\top d\mathbf{x}$$

$$d(\mathbf{x} + \mathbf{y}) = d\mathbf{x} + d\mathbf{y}$$

$$d\mathbf{A}\mathbf{x} = \mathbf{A}d\mathbf{x}$$

$$d\mathbf{x}^\top \mathbf{A}\mathbf{x} = 2\mathbf{x}^\top \mathbf{A}d\mathbf{x}$$

在此只证明最后一条，注意到：

$$\begin{aligned} \mathbf{x}^\top \mathbf{A}\mathbf{x} &= \sum_{i=1}^n \sum_{j=1}^n \mathbf{A}_{i,j} \mathbf{x}_i \mathbf{x}_j \\ \frac{\partial}{\partial \mathbf{x}_i} \mathbf{x}^\top \mathbf{A}\mathbf{x} &= 2\mathbf{A}_{i,i} \mathbf{x}_i + 2 \sum_{1 \leq j \leq n, j \neq i} \mathbf{A}_{i,j} \mathbf{x}_j = 2 \sum_{j=1}^n \mathbf{A}_{i,j} \mathbf{x}_j \\ \Rightarrow d\mathbf{x}^\top \mathbf{A}\mathbf{x} &= 2 \sum_{i=1}^n \sum_{j=1}^n \mathbf{A}_{i,j} \mathbf{x}_j d\mathbf{x}_i = 2\mathbf{x}^\top \mathbf{A}d\mathbf{x} \end{aligned}$$

与一元函数同理，如果上述函数 f 满足二阶偏导数连续的条件，则我们也可以利用Hessian矩阵做出更高阶的估计：

$$f(\mathbf{x} + \mathbf{h}) = f(\mathbf{x}) + \mathbf{w}^\top \mathbf{h} + \frac{1}{2} \mathbf{h}^\top \mathbf{H} \mathbf{h} + O(\|\mathbf{h}\|_2^2)$$

其中Hessian矩阵的形式为：

$$\mathbf{H}_{i,j} = \frac{\partial^2 f}{\partial \mathbf{x}_i \partial \mathbf{x}_j}$$

一般我们会将向量 \mathbf{w} 称为函数 f 的梯度，记为 $\mathbf{grad}f$ ，其还可以使用哈密顿算符表示：

$$\mathbf{grad}f = \nabla f$$

$$\mathbf{H} = \nabla \nabla^\top f$$

如果我们在上述表示中舍弃高阶项得到函数 f 的近似表达：

$$f(\mathbf{x} + \mathbf{h}) \approx f(x) + (\nabla f)^\top \mathbf{h} + \frac{1}{2} \mathbf{h}^\top (\nabla \nabla^\top f) \mathbf{h}$$

根据上述导数公式，我们可以令函数 f 关于 \mathbf{h} 的导数为零来求得函数的极值点(驻点)，当函数存在二阶连续偏导数时：

$$\begin{aligned} \frac{df}{d\mathbf{h}} &= (\nabla f)^\top + \mathbf{h}^\top (\nabla \nabla^\top f) = \mathbf{0} \\ \Rightarrow \mathbf{h} &= -(\nabla \nabla^\top f)^{-1} (\nabla f) \end{aligned}$$

于是我们可以得到牛顿迭代法求函数极值的表达式：

$$\mathbf{x}_{n+1} \leftarrow \mathbf{x}_n - (\nabla_{\mathbf{x}} \nabla_{\mathbf{x}}^\top f)^{-1} (\nabla_{\mathbf{x}} f)$$

也可以简写为如下形式：

$$\mathbf{x}_{n+1} \leftarrow \mathbf{x}_n - \left(\frac{\partial^2 f}{\partial \mathbf{x} \partial \mathbf{x}^\top} \right)^{-1} \left(\frac{\partial f}{\partial \mathbf{x}} \right)$$

0.1.2 线性回归模型的解析解

一般的线性模型可以被描述为以下形式，其中 $\hat{y} \in \mathbb{R}, \mathbf{x} \in \mathbb{R}^d, \mathbf{w} \in \mathbb{R}^d$ ：

$$\hat{y} = \mathbf{w}^\top \mathbf{x} + \mathbf{b}$$

而对于批量的样本数据，使用 $\mathbf{X} \in \mathbb{R}^{n \times d}$ 表示 n 组样本， $\hat{\mathbf{Y}} \in \mathbb{R}^n$ 表示对于数据集上所有样本的预测结果向量，则可以进行如下矩阵表示：

$$\hat{\mathbf{Y}} = \mathbf{X}\mathbf{w} + \mathbf{B}$$

对于真实的数据 \mathbf{Y} ，线性回归要求我们最小化均方误差MSE，这是一个十分简单的优化问题，存在解析解，证明如下：

$$\begin{aligned} \frac{1}{n} \|\hat{\mathbf{Y}} - \mathbf{Y}\|_2^2 &= \frac{1}{n} (\hat{\mathbf{Y}} - \mathbf{Y})^\top (\hat{\mathbf{Y}} - \mathbf{Y}) \\ &= \frac{1}{n} (\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y})^\top (\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y}) \end{aligned}$$

故问题转化为最小化 $(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y})^\top(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y})$ ，这是一个二次型，我们对于 \mathbf{w} 求导：

$$\begin{aligned} d(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y})^\top(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y}) &= 2(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y})^\top d(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y}) \\ &= 2(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y})^\top \mathbf{X} d\mathbf{w} \\ &= 0 \end{aligned}$$

故可以得到：

$$(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y})^\top \mathbf{X} = \mathbf{O}$$

等式两边同时取转置可知：

$$\begin{aligned} \mathbf{X}^\top(\mathbf{X}\mathbf{w} + \mathbf{B} - \mathbf{Y}) &= \mathbf{O} \\ \mathbf{X}^\top \mathbf{X} \mathbf{w} &= \mathbf{X}^\top(\mathbf{Y} - \mathbf{B}) \\ \mathbf{w} &= (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top(\mathbf{Y} - \mathbf{B}) \end{aligned}$$

即可得到参数的最优解，前提是矩阵 $\mathbf{X}^\top \mathbf{X}$ 可逆。

0.1.3 SVD奇异值分解

一般来说，任何实矩阵 $\mathbf{A} \in \mathbb{R}^{n \times m}$ 都可以被无条件地分解为如下三个矩阵的乘积：

$$\mathbf{A}_{n \times m} = \mathbf{U}_{n \times n} \mathbf{\Sigma}_{n \times m} \mathbf{V}_{m \times m}^\top$$

其中 \mathbf{U}, \mathbf{V} 均为正交矩阵，并且 $\mathbf{\Sigma}$ 满足：

$$\Sigma_{i,j} = \begin{cases} \sqrt{\lambda_i} & i = j \\ 0 & i \neq j \end{cases}$$

考虑 $\mathbf{A}^\top \mathbf{A}$ ，这是一个实对称矩阵，故其一定可以被正交对角化，也即存在正交矩阵 \mathbf{V} ，使得：

$$\mathbf{A}^\top \mathbf{A} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^\top$$

其中：

$$\mathbf{\Lambda}_{m \times m} = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_m \end{bmatrix}$$

考虑如下一组向量，我们断言它们之间是互相正交的：

$$\frac{\mathbf{A}\mathbf{v}_1}{\sqrt{\lambda_1}}, \frac{\mathbf{A}\mathbf{v}_2}{\sqrt{\lambda_2}}, \frac{\mathbf{A}\mathbf{v}_3}{\sqrt{\lambda_3}}, \dots, \frac{\mathbf{A}\mathbf{v}_m}{\sqrt{\lambda_m}}$$

证明如下：

$$\frac{\mathbf{A}\mathbf{v}_i}{\sqrt{\lambda_i}} \cdot \frac{\mathbf{A}\mathbf{v}_j}{\sqrt{\lambda_j}} = \frac{\mathbf{v}_i^\top \mathbf{A}^\top \mathbf{A} \mathbf{v}_j}{\sqrt{\lambda_i \lambda_j}} = \frac{\lambda_j \mathbf{v}_i^\top \mathbf{v}_j}{\sqrt{\lambda_i \lambda_j}} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

若 $m \geq n$ ，考虑如下矩阵：

$$\mathbf{U}_{n \times n} = \left(\frac{\mathbf{A}\mathbf{v}_1}{\sqrt{\lambda_1}}, \frac{\mathbf{A}\mathbf{v}_2}{\sqrt{\lambda_2}}, \frac{\mathbf{A}\mathbf{v}_3}{\sqrt{\lambda_3}}, \dots, \frac{\mathbf{A}\mathbf{v}_n}{\sqrt{\lambda_n}} \right)$$

根据上述证明， \mathbf{U} 是正交矩阵，并且满足：

$$\mathbf{U}\Sigma = \mathbf{A}\mathbf{V}$$

$$\mathbf{A} = \mathbf{A}\mathbf{V}\mathbf{V}^\top = \mathbf{U}\Sigma\mathbf{V}^\top$$

若 $m < n$ ，我们可以反过来对 \mathbf{A}^\top 做奇异值分解，也可以得到相同的结果，奇异值分解告诉我们：任何线性变换都可以被分解为一次旋转(旋转、反射或其复合)，一次维度变换及拉伸，一次旋转的复合。除此之外，其还可以被用于求一般矩阵的“逆”：

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^\top$$

$$\mathbf{A}^+ = \mathbf{V}\Sigma^+\mathbf{U}^\top$$

其中 Σ^+ 由将 Σ 中非零元素取倒数后再转置得到。

0.1.4 拉格朗日乘子法与Karush-Kuhn-Tucker条件

对于具有连续偏导数的函数 $f: \mathbb{R}^m \rightarrow \mathbb{R}$ ， $g: \mathbb{R}^m \rightarrow \mathbb{R}$ ，考虑如下约束极值问题：

$$\max(\min) f(\mathbf{x})$$

$$s.t. g(\mathbf{x}) = 0$$

我们知道拉格朗日乘子法可以帮助我们普适地解决多元函数约束极值问题，而不用依赖于技巧。其描述如下，当 $\nabla g_{\mathbf{x}=\mathbf{x}_0} \neq 0$ 时，点 \mathbf{x}_0 是函数 $f(\mathbf{x})$ 在约束条件 $g(\mathbf{x}) = 0$ 下的极值点的必要条件是：

$$\nabla f_{\mathbf{x}=\mathbf{x}_0} = \lambda \nabla g_{\mathbf{x}=\mathbf{x}_0} \quad \lambda \in \mathbb{R}$$

简单推导如下：考虑超曲面 $g(\mathbf{x}) = 0$ 上的一条曲线 Γ ，满足：

$$g(\Gamma(t)) = 0 \quad t \in I$$

$$\Gamma(0) = \mathbf{x}_0$$

等式两边关于变量 t 求导可得：

$$\frac{\partial g}{\partial x_1} \cdot \frac{\partial x_1}{\partial t} + \frac{\partial g}{\partial x_2} \cdot \frac{\partial x_2}{\partial t} + \dots + \frac{\partial g}{\partial x_m} \cdot \frac{\partial x_m}{\partial t} = 0$$

也即：

$$(\nabla g)^\top \cdot \frac{\partial \Gamma}{\partial t} = 0 \quad t \in I$$

令 $t = 0$ 有：

$$(\nabla g_{\mathbf{x}=\mathbf{x}_0})^\top \cdot \left[\frac{\partial \Gamma}{\partial t} \right]_{t=0} = 0$$

由于 $f(\Gamma(t))$ 在 $t = 0$ 处取得极值，故必有：

$$\left[\frac{\partial f}{\partial t} \right]_{t=0} = 0$$

也即：

$$(\nabla f_{\mathbf{x}=\mathbf{x}_0})^\top \cdot \left[\frac{\partial \Gamma}{\partial t} \right]_{t=0} = 0$$

由于此处曲线 Γ 在点 \mathbf{x}_0 处切向量的方向是任意的，故只能：

$$\nabla g_{\mathbf{x}=\mathbf{x}_0} \parallel \nabla f_{\mathbf{x}=\mathbf{x}_0}$$

即可得到结论。同理，考虑如下多约束极值问题：

$$\begin{aligned} & \max(\min) f(\mathbf{x}) \\ & s.t. g^i(\mathbf{x}) = 0 \quad i = 1, 2, \dots, k \end{aligned}$$

根据上述推导我们可以知道点 \mathbf{x}_0 是该约束问题下的一个极值点的必要条件是：

$$\nabla f_{\mathbf{x}=\mathbf{x}_0} \in \text{span} \left\{ \nabla g_{\mathbf{x}=\mathbf{x}_0}^1, \nabla g_{\mathbf{x}=\mathbf{x}_0}^2, \dots, \nabla g_{\mathbf{x}=\mathbf{x}_0}^k \right\}$$

注意此时我们要求这里所有的 $\nabla g_{\mathbf{x}=\mathbf{x}_0}^i$ 必须是线性无关的。接下来，我们进一步考虑如下不等式多约束极值问题：

$$\begin{aligned} & \max(\min) f(\mathbf{x}) \\ & s.t. g^i(\mathbf{x}) \leq 0 \quad i = 1, 2, \dots, k \end{aligned}$$

首先，如果

0.2 算法/基础数学

0.2.1 离散傅里叶变换DFT与快速傅里叶变换FFT

对于数列 $\{a_n\}, \{b_n\}, 0 \leq n < N$ ，我们可以如下定义其离散卷积：

$$(a * b)_k = \sum_{i=0}^k a_i b_{k-i} \quad 0 \leq k < N$$

我们记单位根 $e^{\frac{2k\pi i}{n}} = \omega_n^k$ ，则可以如下定义其离散傅里叶变换及其逆变换：

$$\begin{aligned} DFT(a)_k &= \sum_{t=0}^{N-1} a_t \cdot \omega_N^{-kt} \\ a_k &= \frac{1}{N} \sum_{t=0}^{N-1} DFT(a)_t \cdot \omega_N^{kt} \end{aligned}$$

其中逆变换的证明如下：

$$\begin{aligned} \frac{1}{N} \sum_{t=0}^{N-1} DFT(a)_t \cdot \omega_N^{kt} &= \frac{1}{N} \sum_{t=0}^{N-1} \left(\sum_{u=0}^{N-1} a_u \cdot \omega_N^{-tu} \right) \cdot \omega_N^{kt} \\ &= \frac{1}{N} \sum_{t=0}^{N-1} \sum_{u=0}^{N-1} a_u \cdot \omega_N^{t(k-u)} \\ &= \frac{1}{N} \sum_{u=0}^{N-1} \sum_{t=0}^{N-1} a_u \cdot \omega_N^{t(k-u)} \end{aligned}$$

首先考虑如果 $u = k$ ，则有以下式成立：

$$\begin{aligned} \omega_N^{t(k-u)} &= \omega_N^0 = 1 \\ \sum_{t=0}^{N-1} a_u \cdot \omega_N^{t(k-u)} &= Na_u = Na_k \end{aligned}$$

然后考虑如果 $u \neq k$ ，注意到 $\omega_N^N = 1$ ，则有以下式成立：

$$\begin{aligned} \sum_{t=0}^{N-1} a_u \cdot \omega_N^{t(k-u)} &= a_u \sum_{t=0}^{N-1} \omega_N^{t(k-u)} \\ &= a_u \cdot \frac{1 - \omega_N^{N(k-u)}}{1 - \omega_N^{k-u}} \\ &= 0 \end{aligned}$$

故综上所述，逆变换得证：

$$\frac{1}{N} \sum_{t=0}^{N-1} DFT(a)_t \cdot \omega_N^{kt} = \frac{1}{N} \cdot Na_k = a_k$$

接着我们引入卷积定理的离散形式：

$$a * b = DFT^{-1}(DFT(a) \odot DFT(b))$$

为了证明上式，我们只需要证明：

$$(a * b)_k = DFT^{-1}(DFT(a) \odot DFT(b))_k \quad 0 \leq k < N$$

利用定义展开右式，同理可证：

$$\begin{aligned}
& DFT^{-1}(DFT(a) \odot DFT(b))_k \\
&= \frac{1}{N} \sum_{t=0}^{N-1} (DFT(a) \odot DFT(b))_t \cdot \omega_N^{kt} \\
&= \frac{1}{N} \sum_{t=0}^{N-1} DFT(a)_t \cdot DFT(b)_t \cdot \omega_N^{kt} \\
&= \frac{1}{N} \sum_{t=0}^{N-1} \left(\sum_{n=0}^{N-1} a_n \cdot \omega_N^{-tn} \right) \cdot \left(\sum_{m=0}^{N-1} b_m \cdot \omega_N^{-tm} \right) \cdot \omega_N^{kt} \\
&= \frac{1}{N} \sum_{t=0}^{N-1} \left(\sum_{n=0}^{N-1} \sum_{m=0}^{N-1} a_n b_m \cdot \omega_N^{-t(n+m)} \right) \cdot \omega_N^{kt} \\
&= \frac{1}{N} \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \sum_{t=0}^{N-1} a_n b_m \cdot \omega_N^{t(k-n-m)} \\
&= \frac{1}{N} \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} [n+m=k] N a_n b_m \\
&= \sum_{n=0}^k a_n b_{k-n} \\
&= (a * b)_k
\end{aligned}$$

快速傅里叶变换算法可以帮助我们高效地计算离散傅里叶变换：

$$\{a_n\} \xrightarrow{FFT} \{DFT(a)_n\}$$

结合卷积定理，我们得以加速多项式乘法至 $O(n \log n)$ 时间复杂度：

$$\begin{array}{ccc}
\{a\}, \{b\} & \xrightarrow{O(n^2)} & \{a * b\} \\
\downarrow O(n \log n) & & \uparrow O(n \log n) \\
\{DFT(a)\}, \{DFT(b)\} & \xrightarrow{O(n)} & \{DFT(a) \odot DFT(b)\}
\end{array}$$

为了简化问题，此处我们只讨论 $N = 2^K, K \in \mathbb{N}$ 的简单情形，为了计算离散傅里叶变换，我们的目标是计算下列数值：

$$\begin{aligned}
\mathbf{F} &= [F(w_N^0), F(w_N^{-1}), F(w_N^{-2}), \dots, F(w_N^{-(N-1)})] \\
F(x) &= \sum_{t=0}^{N-1} a_t x^t \Rightarrow DFT(a)_k = F(\omega_N^{-k})
\end{aligned}$$

我们将函数 $F(x)$ 拆分为如下两个部分，注意到当 $N \neq 1$ 时其为偶数：

$$\begin{aligned}
 F(x) &= a_0 + a_1x + a_2x^2 + \cdots + a_{N-1}x^{N-1} \\
 &= (a_0 + a_2x^2 + \cdots + a_{N-2}x^{N-2}) + (a_1x + a_3x^3 + \cdots + a_{N-1}x^{N-1}) \\
 &= (a_0 + a_2x^2 + \cdots + a_{N-2}x^{N-2}) + x(a_1 + a_3x^2 + \cdots + a_{N-1}x^{N-2}) \\
 &= A_e(x^2) + xA_o(x^2)
 \end{aligned}$$

故问题转化为计算如下数值：

$$A_e(\omega_N^{-2k}), A_o(\omega_N^{-2k}) \quad 0 \leq k < N$$

注意到：

$$\begin{aligned}
 \omega_N^{-2k} &= \omega_{\frac{N}{2}}^{-k} \\
 \omega_{\frac{N}{2}}^{-(k+\frac{N}{2})} &= \omega_{\frac{N}{2}}^{-k} \\
 \omega_N^{-(k+\frac{N}{2})} &= -\omega_N^{-k}
 \end{aligned}$$

所以我们实际上只需要计算如下数值：

$$\begin{aligned}
 \mathbf{A}_e &= [A_e(\omega_{\frac{N}{2}}^0), A_e(\omega_{\frac{N}{2}}^{-1}), A_e(\omega_{\frac{N}{2}}^{-2}), \dots, A_e(\omega_{\frac{N}{2}}^{-(\frac{N}{2}-1)})] \\
 \mathbf{A}_o &= [A_o(\omega_{\frac{N}{2}}^0), A_o(\omega_{\frac{N}{2}}^{-1}), A_o(\omega_{\frac{N}{2}}^{-2}), \dots, A_o(\omega_{\frac{N}{2}}^{-(\frac{N}{2}-1)})]
 \end{aligned}$$

便可以计算出所需要的数值：

$$\begin{aligned}
 \mathbf{F}[k] &= \mathbf{A}_e[k] + \omega_N^{-k} \mathbf{A}_o[k] \\
 \mathbf{F}\left[k + \frac{N}{2}\right] &= \mathbf{A}_e[k] - \omega_N^{-k} \mathbf{A}_o[k] \\
 0 \leq k &< \frac{N}{2}
 \end{aligned}$$

在此我们只证明第二个算式：

$$\begin{aligned}
 \mathbf{F}\left[k + \frac{N}{2}\right] &= F(\omega_N^{-(k+\frac{N}{2})}) \\
 &= A_e(\omega_N^{-2(k+\frac{N}{2})}) + \omega_N^{-(k+\frac{N}{2})} A_o(\omega_N^{-2(k+\frac{N}{2})}) \\
 &= A_e(\omega_{\frac{N}{2}}^{-(k+\frac{N}{2})}) - \omega_N^{-k} A_o(\omega_{\frac{N}{2}}^{-(k+\frac{N}{2})}) \\
 &= A_e(\omega_{\frac{N}{2}}^{-k}) - \omega_N^{-k} A_o(\omega_{\frac{N}{2}}^{-k}) \\
 &= \mathbf{A}_e[k] - \omega_N^{-k} \mathbf{A}_o[k]
 \end{aligned}$$

而 $\mathbf{A}_e, \mathbf{A}_o$ 的计算又可以递归地使用上述方法，并且问题的规模在指数级地缩减，故我们可以利用FFT算法高效地实现离散傅里叶变换地计算。以下为其算法时间复杂度的分析，假设问题规

模为 N 时所对应的时间复杂度为 $T(N)$ ，则根据上述讨论可知：

$$T(N) = 2 \cdot T(N/2) + N$$

我们不难归纳证明出：

$$T(N) = 2^k \cdot T(N/2^k) + kN \quad k \in \mathbb{N}$$

因为：

$$\begin{aligned} T(N) &= 2^k \cdot T(N/2^k) + kN \\ &= 2^k \cdot [2 \cdot T(N/2^{k+1}) + N/2^k] + kN \\ &= 2^{k+1} \cdot T(N/2^{k+1}) + (k+1)N \end{aligned}$$

令 $k = \log_2 N$ ，可知：

$$T(N) = N \cdot T(1) + N \log_2 N = O(N \log N)$$

0.2.2 傅里叶变换

如果一个具有周期 T 的函数 $f(t)$ 在区间 $[0, T]$ 上满足Dirichlet条件，那么我们可以将其展开为如下形式的傅里叶级数：

$$f(t) \sim \frac{a_0}{2} + \sum_{k=0}^{\infty} \left(a_k \cos\left(\frac{2\pi}{T}kt\right) + b_k \sin\left(\frac{2\pi}{T}kt\right) \right)$$

我们可以通过如下变换将其转化为更为简洁的形式，令 $\frac{2\pi}{T} = \omega$ ，于是有：

$$\begin{aligned} \cos\left(\frac{2\pi}{T}kt\right) &= \cos(\omega kt) = \frac{e^{i\omega kt} + e^{-i\omega kt}}{2} \\ \sin\left(\frac{2\pi}{T}kt\right) &= \sin(\omega kt) = \frac{e^{i\omega kt} - e^{-i\omega kt}}{2i} \end{aligned}$$

故而：

$$\begin{aligned} &\frac{a_0}{2} + \sum_{k=0}^{\infty} \left(a_k \cos\left(\frac{2\pi}{T}kt\right) + b_k \sin\left(\frac{2\pi}{T}kt\right) \right) \\ &= \frac{a_0}{2} + \sum_{k=0}^{\infty} \left(a_k \cdot \frac{e^{i\omega kt} + e^{-i\omega kt}}{2} + b_k \cdot \frac{e^{i\omega kt} - e^{-i\omega kt}}{2i} \right) \\ &= \frac{a_0}{2} + \sum_{k=0}^{\infty} \left(\frac{a_k + ib_k}{2} e^{-i\omega kt} + \frac{a_k - ib_k}{2} e^{i\omega kt} \right) \\ &= \sum_{k=-\infty}^{+\infty} c_k \cdot e^{i\omega kt} \end{aligned}$$

显然，其中：

$$c_k = \begin{cases} \frac{a_k - ib_k}{2} & k > 0 \\ \frac{a_0}{2} & k = 0 \\ \frac{a_k + ib_k}{2} & k < 0 \end{cases}$$

如果上述级数在区间 $[0, T]$ 内一致收敛于 $f(t)$ ，这通常要求其极限函数的导数是平方可积的，那么我们可以通过如下方法求得系数：

$$c_k = \frac{1}{T} \int_0^T f(t) e^{-i\omega kt} dt$$

证明如下，由一致收敛性我们可以交换积分以及级数求和的次序：

$$\begin{aligned} \int_0^T f(t) e^{-i\omega kt} dt &= \int_0^T \sum_{n=-\infty}^{+\infty} c_n \cdot e^{i\omega nt} \cdot e^{-i\omega kt} dt \\ &= \sum_{n=-\infty}^{+\infty} \int_0^T c_n \cdot e^{i\omega nt} \cdot e^{-i\omega kt} dt \\ &= \sum_{n=-\infty}^{+\infty} c_n \int_0^T e^{i\omega(n-k)t} dt \\ &= \sum_{n=-\infty}^{+\infty} c_n [n = k] T \\ &= T c_k \end{aligned}$$

我们可以通过平移使得函数 $f(t)$ 成为区间 $[-\frac{T}{2}, \frac{T}{2}]$ 上的周期函数，故而有下式成立：

$$\begin{aligned} f(t) &= \sum_{k=-\infty}^{+\infty} c_k \cdot e^{i\omega kt} \\ &= \sum_{k=-\infty}^{+\infty} \left(\frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} f(\xi) e^{-i\omega k\xi} d\xi \right) \cdot e^{i\omega kt} \\ &= \sum_{k=-\infty}^{+\infty} \left(\int_{-\frac{T}{2}}^{\frac{T}{2}} f(\xi) e^{-i2\pi v\xi} d\xi \right) \cdot e^{i2\pi vt} \cdot \frac{1}{T} \end{aligned}$$

其中 $v = \frac{k}{T}$ ，考虑令 $T \rightarrow \infty$ ，此时我们可以将一般函数 f 视为具有无穷大周期的周期函数，于是在满足Dirichlet条件的情形下我们有：

$$\begin{aligned} f(t) &= \lim_{T \rightarrow \infty} \sum_{k=-\infty}^{+\infty} \left(\int_{-\frac{T}{2}}^{\frac{T}{2}} f(\xi) e^{-i2\pi v\xi} d\xi \right) \cdot e^{i2\pi vt} \cdot \frac{1}{T} \\ &= \int_{-\infty}^{+\infty} \left(\int_{-\infty}^{+\infty} f(\xi) e^{-i2\pi v\xi} d\xi \right) e^{i2\pi vt} dv \end{aligned}$$

令 $2\pi v = \omega$ 就可以得到如下熟悉的傅里叶变换形式:

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \left(\int_{-\infty}^{+\infty} f(\xi) e^{-i\omega\xi} d\xi \right) e^{i\omega t} d\omega$$

于是我们可以由此定义傅里叶变换以及傅里叶逆变换:

$$\begin{aligned} F(\omega) &= \int_{-\infty}^{+\infty} f(t) e^{-i\omega t} dt \\ f(t) &= \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega) e^{i\omega t} d\omega \end{aligned}$$

对于 \mathbb{R} 上的两个可积函数 $f(x)$, $g(x)$, 我们定义其卷积如下:

$$(f * g)(x) = \int_{-\infty}^{+\infty} f(t) g(x - t) dt$$

通过换元 $u = x - t$ 我们可以得到卷积运算的交换律:

$$\begin{aligned} (f * g)(x) &= \int_{-\infty}^{+\infty} f(t) g(x - t) dt \\ &= - \int_{+\infty}^{-\infty} f(x - u) g(u) du \\ &= \int_{-\infty}^{+\infty} g(u) f(x - u) du \\ &= (g * f)(x) \end{aligned}$$

卷积定理是傅立叶变换满足的一个重要性质, 其指出, 函数卷积的傅立叶变换是函数傅立叶变换的乘积。

$$\mathcal{F}[f * g](\omega) = \mathcal{F}[f](\omega) \cdot \mathcal{F}[g](\omega)$$

证明如下:

$$\begin{aligned} \mathcal{F}[f * g](\omega) &= \int_{-\infty}^{+\infty} (f * g)(t) e^{-i\omega t} dt \\ &= \int_{-\infty}^{+\infty} \left(\int_{-\infty}^{+\infty} f(\tau) g(t - \tau) d\tau \right) e^{-i\omega t} dt \\ &= \int_{-\infty}^{+\infty} \left(\int_{-\infty}^{+\infty} g(t - \tau) e^{-i\omega t} dt \right) f(\tau) d\tau \\ &= \int_{-\infty}^{+\infty} \left(\int_{-\infty}^{+\infty} g(t - \tau) e^{-i\omega(t - \tau)} dt \right) f(\tau) e^{-i\omega\tau} d\tau \\ &= \int_{-\infty}^{+\infty} \left(\int_{-\infty}^{+\infty} g(u) e^{-i\omega u} du \right) f(\tau) e^{-i\omega\tau} d\tau \\ &= \mathcal{F}[g](\omega) \cdot \int_{-\infty}^{+\infty} f(\tau) e^{-i\omega\tau} d\tau \\ &= \mathcal{F}[f](\omega) \cdot \mathcal{F}[g](\omega) \end{aligned}$$

0.3 算法竞赛中的数论与组合数学

0.3.1 容斥原理与二项式反演

容斥原理的描述如下，对于一系列集合 A_1, A_2, \dots, A_n ，我们有：

$$\left| \bigcup_{i=1}^n A_i \right| = \sum_{i=1}^n (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_i \leq n} \left| \bigcap_{j=1}^i A_{k_j} \right|$$

证明如下：我们使用数学归纳法，当 $n = 2$ 时，显然有：

$$|A_1 \cup A_2| = |A_1| + |A_2| - |A_1 \cap A_2|$$

假设当 $n = m$ 时有结论成立，则当 $n = m + 1$ 时，我们有：

$$\left| \bigcup_{i=1}^{m+1} A_i \right| = \left| \bigcup_{i=1}^m A_i \right| + |A_{m+1}| - \left| \bigcup_{i=1}^m (A_i \cap A_{m+1}) \right|$$

其中：

$$\begin{aligned} - \left| \bigcup_{i=1}^m (A_i \cap A_{m+1}) \right| &= - \sum_{i=1}^m (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_i \leq m} \left| \bigcap_{j=1}^i (A_{k_j} \cap A_{m+1}) \right| \\ &= \sum_{i=1}^m (-1)^{i+2} \sum_{1 \leq k_1 < k_2 < \dots < k_i \leq m} \left| \left(\bigcap_{j=1}^i A_{k_j} \right) \cap A_{m+1} \right| \\ &= \sum_{i=2}^{m+1} (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_{i-1} \leq m} \left| \left(\bigcap_{j=1}^{i-1} A_{k_j} \right) \cap A_{m+1} \right| \\ &= \sum_{i=2}^{m+1} (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_{i-1} < k_i = m+1} \left| \bigcap_{j=1}^i A_{k_j} \right| \end{aligned}$$

故：

$$|A_{m+1}| - \left| \bigcup_{i=1}^m (A_i \cap A_{m+1}) \right| = \sum_{i=1}^{m+1} (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_{i-1} < k_i = m+1} \left| \bigcap_{j=1}^i A_{k_j} \right|$$

而：

$$\begin{aligned} \left| \bigcup_{i=1}^m A_i \right| &= \sum_{i=1}^m (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_i \leq m} \left| \bigcap_{j=1}^i A_{k_j} \right| \\ &= \sum_{i=1}^m (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_i < m+1} \left| \bigcap_{j=1}^i A_{k_j} \right| \end{aligned}$$

综上所述:

$$\begin{aligned}
\left| \bigcup_{i=1}^{m+1} A_i \right| &= \left| \bigcup_{i=1}^m A_i \right| + |A_{m+1}| - \left| \bigcup_{i=1}^m (A_i \cap A_{m+1}) \right| \\
&= \sum_{i=1}^m (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_i < m+1} \left| \bigcap_{j=1}^i A_{k_j} \right| \\
&\quad + \sum_{i=1}^{m+1} (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_{i-1} < k_i = m+1} \left| \bigcap_{j=1}^i A_{k_j} \right| \\
&= \sum_{i=1}^{m+1} (-1)^{i+1} \sum_{1 \leq k_1 < k_2 < \dots < k_i \leq m+1} \left| \bigcap_{j=1}^i A_{k_j} \right|
\end{aligned}$$

归纳成立, 结论得证。接下来我们考虑如下特殊情形, 假如对于:

$$\forall k_1, k_2, \dots, k_i \quad 1 \leq k_1 < k_2 < \dots < k_i \leq n$$

均有:

$$\left| \bigcap_{j=1}^i A_{k_j} \right| = S_i$$

则我们显然有:

$$\left| \bigcup_{i=1}^n A_i \right| = \sum_{i=1}^n (-1)^{i+1} C_n^i S_i$$

在之后的讨论中, 我们一般将上述条件简称为**完备对称条件**, 二项式反演指出, 如果对于:

$$\forall k_1, k_2, \dots, k_i \quad 1 \leq k_1 < k_2 < \dots < k_i \leq n$$

均有:

$$\left| \bigcup_{j=1}^i A_{k_j} \right| = U_i$$

则:

$$S_n = \sum_{i=1}^n (-1)^{i+1} C_n^i U_i$$

证明如下, 根据之前的结论, 我们有:

$$\begin{aligned}
\sum_{i=1}^n (-1)^{i+1} C_n^i U_i &= \sum_{i=1}^n (-1)^{i+1} C_n^i \sum_{j=1}^i (-1)^{j+1} C_i^j S_j \\
&= \sum_{i=1}^n \sum_{j=1}^i (-1)^{i+j} C_n^i C_i^j S_j \\
&= \sum_{j=1}^n S_j \sum_{i=j}^n (-1)^{i+j} C_n^i C_i^j
\end{aligned}$$

事实上，根据组合数公式，我们有：

$$C_n^i C_i^j = \frac{n!}{(n-i)!(i-j)!j!} = C_{n-j}^{i-j} C_n^j$$

故而：

$$\begin{aligned} \sum_{i=1}^n (-1)^{i+1} C_n^i U_i &= \sum_{j=1}^n S_j \sum_{i=j}^n (-1)^{i+j} C_{n-j}^{i-j} C_n^j \\ &= \sum_{j=1}^n S_j C_n^j \sum_{t=0}^{n-j} (-1)^t C_{n-j}^t \\ &= \sum_{j=1}^n S_j C_n^j [n=j] \\ &= S_n \end{aligned}$$

证毕。于是我们得到重要公式，在满足完备对称条件下二项式反演的第一种表达形式形式：

$$\begin{aligned} S_n &= \sum_{i=1}^n (-1)^{i+1} C_n^i U_i \\ U_n &= \sum_{i=1}^n (-1)^{i+1} C_n^i S_i \end{aligned}$$

如果我们设 $U'_i = n - U_i$ ，并且 $U'_0 = S_0 = n$ ，那么上式也可以被转化为如下形式：

$$\begin{aligned} S_n &= \sum_{i=0}^n (-1)^i C_n^i U'_i \\ U'_n &= \sum_{i=0}^n (-1)^i C_n^i S_i \end{aligned}$$

0.3.2 容斥原理与二项式反演的实际应用举例

容斥原理与二项式反演最重要的作用在于实现组合问题中交集的计数与并集的计数之间的转化，而通常来说二者在计算上的难度并不均等。这意味着我们可以实现从较为困难的问题向较为简单的问题的转化，这是十分重要的技巧。作为第一个例子，我们考虑如下问题，已知集合：

$$N = \{1, 2, \dots, n\}$$

考虑选取若干 N 的子集(至少选取一个)，求所选取子集的交集恰好包含 k 个元素的选法数量。通常情况下，“恰好”型问题不会比“至少”型问题更容易解决，所以我们容易想到做出如下转化，设 G_k 表示所选取子集的交集至少包含 k 个元素的选法数量，则显而易见答案为：

$$answer = G_k - G_{k+1}$$

接下来我们做出如下思考，如果我们想使用二项式反演解决问题，那么我们必须找到一组满足完备对称条件的集合。所以对于每一个 $i \in N$ ，考虑：

$$A_i = \{s | s \subseteq N, i \in s\}$$