第二章 和式基础

具体数学阅读笔记

2024 年 8 月 2 日

§1 和式及其基本操作

要使得求和便于书写和分析, 我们最好 (跟随傅里叶) 引入如下的记号:$\sum$.

1.1. 和式与求和记号.

定义 1.1 (求和记号 $\sum$)**.** 对于一个可数的集合 $S=\{a_1,a_2,\cdots,a_n\}$, 求和实际上是将这个集合的每一个元素在某一个函数 $f:S\to X$ 作用之后, 把对应的值相加. 记作 $\sum_{i\in S}f(i)$; 表示 $f(a_1)+f(a_2)+f(a_3)+\cdots+f(a_n)+\cdots$.

例子 1.1. 考虑

$$\sum_{1\le k\le n}a_k=a_1+a_2+\cdots+a_n.$$

$$\sum_{\substack{1\le k\le 100\\ k\text{ odd}}}k^2=\sum_{0\le k\le 49}(2k+1)^2.$$

a) 换元法 如果将 $\sum_{i\in S}f(i)$ 换为 $\sum_{k\in S}f(k)$, 求和的表示内容将不变. 这是因为仅仅是"当前 S 中的代表" 用来表示的字母不同, 从而肯定不会影响整个映射 f 所表达的意思.

例子 1.2. 考虑

$$\sum_{1\le k\le n}a_k\xlongequal{k\text{代换为}s+1}\sum_{1\le s+1\le n}a_{s+1}\xlongequal{s\text{代换为}k}\sum_{1\le k+1\le n}a_{k+1}$$

注: 有时候 $\sum_{a\le i\le b}f(i)$ 也写作 $\sum_{i=a}^{b}f(i)$. 但是看到这样一来, 变量代换之后就得到了

$$\sum_{i=1}^{n}a_i\xlongequal{i\text{代为}i+1}\sum_{i=0}^{n-1}a_{i+1}$$

更易于出错.

此外, 我们用 $k:=k+1$ 来表示把 k 代为 $k+1$. 最后, 数列可以看做特殊的函数. 实际上书写 a_k 实际上就相当于 $f(k)$.

b) 求和的性质 尽管我们可以对可数无限个元素进行求和操作, 但是, 那样的数列我们在高等数学的级数部分就已经学过了. 这里先讨论有限项求和的情形.

定理 1.1. 设 K 是某一有限个正整数的集合, 我们有如下的三条规则:

- 常数项进出求和记号:
$$\sum_{k\in k} cf(k) = c\sum_{k\in k} f(k);$$
- 求和记号的拆分:
$$\sum_{k\in K} f(k)+g(k) = \sum_{k\in K} f(k) + \sum_{k\in K} g(k);$$
- 若 $p(k)$ 应用于 K 中的每一个元素之后组成的集合依然是 K 的一个排列, 那么
$$\sum_{k\in K} f(k) = \sum_{p(k)\in K} f(p(k)).$$

上述定理的证明可以直接由定义得到.

例子 1.3. 如 $K=\{-1,0,1\}, p(k)=-k$, 由于 $p(-1)=1, p(0)=0$,$p(1)=-1, p$ 对 k 中每一个元素构成集合为 $\{-1,0,1\}=K$.

例子 1.4 (等差数列求和)**.** 求
$$S=\sum_{0\le k\le n}(a+bk)$$
的值.

考虑
$$\begin{aligned}S_2 &= \sum_{0\le k\le n}(a+bk)\\ &\overset{k:=n-k}{=\!=\!=} \sum_{0\le n-k\le n}(a+b(n-k)) = \sum_{0\le k\le n}(a+bn-bk)\end{aligned}$$

令 $S+S_2$ 得

$$\begin{aligned}S+S_2=2S &= \sum_{0\le k\le n}(a+bk)+\sum_{0\le k\le n}(a+bn-bk)\\ &= \sum_{0\le k\le n}(2a+bn) = (2a+bn)\sum_{0\le k\le n}1 = (2a+bn)(n+1)\end{aligned}$$

那
$$S=\frac{(n+1)(2a+bn)}{2}.$$

1.2. Iverson 括号.

定义 1.2 (Iverson 括号)**.** 假设 P 是一个命题, 定义

$$[P] = \begin{cases} 1, & \text{命题}P\text{为真} \\ 0, & \text{命题}P\text{为假} \end{cases}$$

这样的记号便于优化很复杂的求和操作, 并且可以把求和符号的下标转换为命题之间的操作.

例子 1.5. 求和式 $\sum_{0\le k\le n} k$ 可被改为 $\sum_k k[0 \le k \le n]$.

如果 k 未指定限定条件, 我们认为 $k \in \mathbb{Z}$. 也就是说上述式子

$$\begin{aligned}\sum_k k[0 \le k \le n] =& \cdots + (-3\cdot 0) + (-2\cdot 0) + (-1\cdot 0) + (0\cdot 0) + (1\cdot 1) + (2\cdot 1) + \cdots + (n\cdot 1) \\ =&(0\cdot 1) + (1\cdot 1) + \cdots + (n\cdot 1)\end{aligned}$$

例子 1.6. 如果 K 与 K' 是两个整数集合, 那么 $\forall k$,

$$[k \in K] + [k \in K'] = [k \in (K \cap K')] + [k \in (K \cup K')]$$

由此可以导出对应的和式

$$\sum_{k\in k} a_k + \sum_{k\in k'} a_k = \sum_{k\in k\cap k'} a_k + \sum_{k\in k\cup k'} a_k$$

这是一个很有用的命题. 下面我们会看到它的用途.

命题 1.2. $[k \in K] + [k \in K'] = [k \in (K \cap K')] + [k \in (K \cup K')]$ 对 k, k' 为可数集, $\forall k$.

1.3. 常见的求和方法.

a) 成套方法 这个方法类似于在解答微分方程的时候, 首先求解特解, 然后求解通解. 在这里我们不再赘述.

b) 扰动法 要计算 $S_n = \sum_{0\le k\le n} a_k$, 可以有两种方法改写

命题 1.3. 扰动法是指

$$\begin{aligned}\boxed{S_n + a_{n+1}} = \sum_{0\le k\le n+1} a_k = a_0 + \sum_{1\le k\le n+1} a_k \\ \xlongequal{k:=k+1} a_0 + \sum_{1\le k+1\le n+1} a_k \\ \boxed{= a_0 + \sum_{0\le k\le n} a_k}\end{aligned}$$

例子 1.7. 用上述的方法计算等比数列. $S_n = \sum_{0\le k\le n} ax^k$.

$$\begin{aligned} S_n + ax^{n+1} &= ax^0 + \sum_{0\le k\le n} ax^{k+1} \\ &= ax^0 + x\sum_{0\le k\le n} ax^k \\ &= ax^0 + xS_n \end{aligned}$$

对于 $x \neq 1$, 有

$$S_n = \sum_{k\neq 1}^{n} ax^k = \frac{a - ax^{n+1}}{1-x}$$

例子 1.8 (等差数列乘等比数列)**.** 计算

$$S_n = \sum_{0\le k\le n} k\cdot 2^k.$$

按照上面的方法,

$$\begin{aligned} S_n + (n+1)2^{n+1} &= 0 + \sum_{0\le k\le n} (k+1)2^{k+1} \\ &= 0 + \sum_{0\le k\le n} k\cdot 2^{k+1} + \sum_{0\le k\le n} 2^{k+1} \\ &= 2S_n + 2^{n+2} - 2 \end{aligned}$$

解出 S_n, 就是

$$S_n = \sum_{0\le k\le n} k2^k = (n-1)2^{n+1} + 2.$$

例子 1.9. 从上面的推导中, 我们知道 $\sum_{k=0}^{n} x^k = \frac{1-x^{n-1}}{1-x}$, 两边对 x 求导, 就有

$$\begin{aligned} \sum_{k=0}^{n} kx^{k-1} &= \frac{(1-x)\left(-(n+1)x^n\right) + 1 - x^{n+1}}{(1-x^2)} \\ &= \frac{1-(n+1)x^n + nx^{n+1}}{(1-x)^2}. \end{aligned}$$

同样也可以得到上一式子的结果.

c) 求和因子

例子 1.10. 由递推关系

$$\begin{aligned} T_0 &= 0 \\ T_n &= 2T_{n-1} + 1 \end{aligned}$$

倘若两端同时除以 2^n, 就得到

$$\begin{aligned} T_0/2^0 &= 0 \\ T_n/2^n &= T_{n-1}/2^{n-1} + 1/2^n \end{aligned}$$

令 $S=T_n/2^n$ ，得 $\begin{cases}S_0=0\\S_n=S_{n-1}+2^{-n}\end{cases}$，即 $S_n=\sum_{k=1}^{n}2^{-k}$ ，为一等比数列.

对于更为一般的式子, 如 $a_nT_n=b_nT_{n-1}+c_n$, 可变为 $S_n=S_{n-1}+S_nc_n$ 的形式.

1° 方法: 使两边同时乘以求和因子 S_n:

$$\underbrace{\boxed{s_na_nT_n}}_{:=S_n}=\underbrace{S_nb_nT_{n-1}}_{\text{必须得是}S_{n-1}=S_{n-1}a_{n-1}T_{n-1}}+S_nc_n$$

由此 $S_n=S_0a_0\cdot T_0+\sum_{k=1}^{n}S_kc_k=S_1b_1T_0+\sum_{k=1}^{n}S_kc_k$ ，那么

$$T_n=\frac{1}{S_na_n}\left(S_1b_1T_0+\sum_{k=1}^{n}S_kc_k\right)$$

.

2° 寻找 S_n 的方法: 由于上式要满足 $S_nb_nT_{n-1}=S_{n-1}a_{n-1}T_{n-1}$, 代入展开, 有

$$\begin{aligned}s_n&=\frac{s_{n-1}a_{n-1}}{b_n}\\&=\frac{s_{n-2}a_{n-1}a_{n-2}}{b_nb_{n-1}}=\cdots\\&=\frac{a_{n-1}a_{n-2}\cdots a_1}{b_nb_{n-1}\cdots b_2}.\end{aligned}$$

因此我们就这样找到了求和因子.

例子 1.11. 解由快速排序带来的递归式:

$$\begin{aligned}&C_0=0\\&C_n=n+1+\frac{2}{n}\sum_{k=0}^{n-1}C_k\quad,n>0.\end{aligned}$$

将 C_n 两侧同乘以 n, 得

$$nC_n=n^2+n+2\sum_{k=0}^{n-1}C_k\quad(n>0).$$

令 $n:=n-1$, 有

$$(n-1)C_{n-1}=(n-1)^2+(n-1)+2\sum_{k=0}^{n-2}c_k.$$

上面两式相减, 有 $nC_n-(n-1)C_{n-1}=2n+2C_{n-1}$, 即

$$\begin{aligned}C_0&=0\\nC_n&=(n+1)C_{n-1}+2n.\end{aligned}$$

将上述 $a_n=n,b_n=n+1,c_n=2n$, 得 $s=\frac{(n-1)\cdots1}{(n+1)\cdots3}=\frac{2}{n(n+1)}$.

得

$$C_n = 2(n+1)\sum_{k=1}^{n}\frac{1}{k+1} = 2(n+1)H_{n+1}.$$

§2 多重和式

2.1. 引例与启发性讨论. 考虑

$$\begin{aligned}\sum_{1\leqslant j,k\leqslant 3} a_j b_k =& a_1b_1 + a_1b_2 + a_1b_3 \\ &+ a_1b_1 + a_2b_2 + a_2b_3 \\ &+ a_3b_1 + a_3b_2 + a_3b_3.\end{aligned}$$

其中 j,k 是两个指标.

如果 $P(j,k)$ 是关于 j,k 的性质, 那么 $\sum_{P(j,k)} a_{j,k} = \sum a_{j,k}[P(j,k)]$, 有时候也写作 $\sum_j \left(\sum_k a_{j,k}[P(j,k)]\right)$, 可以简写做作 $\sum_j \sum_k a_{j,k}[P(j,k)]$.

命题 2.1. 如果 k 的表示不依赖于 j 的话, 那么

$$\sum_j \sum_k a_{j,k}[P(i,k)] = \sum_k \sum_j a_{j,k}[P(j,k)]$$

例子 2.1. 上述的考量最简单的情形如下:

$$\begin{aligned}\sum_{1\leqslant j\leqslant 2} \sum_{1\leqslant k\leqslant 2} a_{j,k} &= a_{1,1} + a_{1,2} + a_{2,1} + a_{2,2} \\ &= a_{2,1} + a_{2,1} + a_{1,2} + a_{2,2} = \sum_{1\leqslant k\leqslant 2} \sum_{1\leqslant j\leqslant 2} a_{j,k}.\end{aligned}$$

命题 2.2 (一般分配率)**.** 对于任意的整数集合 J,K 有

$$\sum_{\substack{j\in J\\ k\in K}} a_j b_k = \left(\sum_{j\in J} a_j\right)\left(\sum_{k\in K} b_k\right).$$

注: 这里 J 和 K 集合要不依赖对方就能确定. 例如,

$$\sum_{i=1}^{n} \sum_{j=i}^{m} f(i,j)$$

就无法使用这个方式.

证明. 我们可以考虑

$$
\begin{aligned}
&\left(\sum_{j\in J} a_j\right)\left(\sum_{k\in k} b_k\right) = \sum_j a_j[j\in J] \quad \sum_{k\in k} b_k[k\in K] \\
&= \sum_j a_j[j\in J]\left(\sum_k b_k[k\in k]\right) \\
&\xlongequal{j\text{于}k\text{ 独立}} \sum_j\sum_k a_j b_k[j\in J][k\in k] \\
&= \sum_j\sum_k a_j b_k[j\in J, k\in K].
\end{aligned}
$$

□

例子 2.2. 例如,

$$\sum_{1\leqslant j,k\leqslant 3} a_j b_k = \left(\sum_{j=1}^{3} a_j\right)\left(\sum_{k=1}^{3} b_k\right)$$

请再次注意这里的 k 的取值构成何种集合与 j 无关. 与这个例子相对, $\sum_{i=1}^{n}\sum_{j=i}^{m} f(i,j)$ 就不适用这个规则. 因为 $i\leq j\leq m$ 表示什么集合与 i 的值有关.

下面我们来看相互依赖的集合的情形.

命题 2.3. 对于依赖于 j 取值的集合 $K(j), j\in J$, 可以按照如下的方式进行交换求和记号,

$$\sum_{j\in J}\sum_{k\in K(j)} a_{j,k} = \sum_{k\in K'}\sum_{j\in J'(k)} a_{j,k}$$

满足 $[j\in J][k\in K(j)] = [k\in K'][j\in J'(k)]$.

注: 可以把这个与交换二重积分次序相联系起来. 例如对于区域 $D: 0\leq x\leq 1, 0\leq y\leq x$ 的区域积分, 可以写作

$$
\begin{aligned}
\iint_D f(x,y)dxdy &= \int_0^1 dx\int_0^x f(x,y)dy \qquad (\text{从下向上}) \\
&= \int_0^1 dy\int_x^1 f(x,y)dx. \qquad (\text{从左向右})
\end{aligned}
$$

命题中表述的事实是保证每个元素都计算了且仅仅被计算了一次.

推论 2.4. 类似上面举例的二重积分更换 x, y, 有

$$
\begin{aligned}
[1\leqslant j\leqslant n][1\leqslant k\leqslant n] &= [1\leqslant j\leqslant k\leqslant n] \\
&= [1\leqslant k\leqslant n][1\leqslant j\leqslant k]
\end{aligned}
$$

例子 2.3. 根据上述推论,

$$\sum_{j=1}^{n}\sum_{k=j}^{n} a_{j,k} = \sum_{1\leqslant j\leqslant k\leqslant n} a_{j,k} = \sum_{k=1}^{n}\sum_{j=1}^{k} a_{j,k}.$$

例子 2.4. 给出矩阵

$$\begin{bmatrix} a_1a_1 & a_1a_2 & \cdots & a_1a_n \\ a_2a_1 & & & \\ \vdots & & \ddots & \\ a_na_1 & & & a_na_n \end{bmatrix}$$

求

$$S_{\overline{\backslash|}} = \sum_{1\leqslant j\leqslant k\leqslant n} a_j a_k$$

由于该矩阵是对称的, 即 $a_ia_j = a_ja_i$, $S_{\overline{\backslash|}} \approx \frac{1}{2}S_{\square}$, 那么

$$S_{\overline{\backslash|}} = \sum_{1\leq j\leq k\leqslant n} a_j a_k = \sum_{1\leqslant j\leqslant k\leqslant n} a_k a_j \xlongequal{\text{对换}j,k} \sum_{l\leqslant k\leqslant j\leqslant n} a_j a_k = S_{\underline{/}|}$$

根据 $[k\in k'] + [k\in k] = [k\in (k\cap k')] + [k\in (k\cup K')]$, 就有

$$[1\leqslant j\leqslant k\leqslant n] + [1\leqslant k\leqslant j\leqslant n] = [1\leqslant j,k\leqslant n] + [1\leqslant j = k\leqslant n].$$

所以

$$\begin{aligned} 2S_{\overline{\backslash|}} = S_{\overline{\backslash|}} + S_{\underline{/}|} &= \sum_{1\leqslant i,k\leqslant n} a_j a_k + \sum_{1\leqslant j=k\leqslant n} a_j a_k \\ &= \left(\sum_{k=1}^{n} a_k\right)^2 + \sum_{k=1}^{n} a_k^2 \end{aligned}$$

那么

$$S_{\overline{\backslash|}} = \frac{1}{2}\left(\left(\sum_{k=1}^{n} a_k\right)^2 + \sum_{k=1}^{n} a_k^2\right).$$

例子 2.5. 求解

$$S = \sum_{1\leqslant j<k\leqslant n} (a_k - a_j)(b_k - b_j).$$

注意到 i, j 的对称性, 将 i, j 对换得:

$$S = \sum_{1\leqslant k<j\leqslant n} (a_j - a_k)(b_j - b_k) = \sum_{1\leqslant k<j\leqslant n} (a_k - a_j)(b_k - b_j)$$

由于

$$[1\leqslant j<k\leqslant n] + [1\leqslant k<j\leqslant n] = [1\leqslant j,k\leqslant n] - [1\leqslant j = k\leqslant n]$$

因此

$$\begin{aligned}2S &= \sum_{1\leqslant j,k\leqslant n}(a_j-a_k)(b_j-b_k)-\sum_{1\leqslant j=k\leqslant n}(a_j-a_k)(b_j-b_k)\,.\\&=\sum_{1\leqslant j\leqslant k\leqslant n}(a_j-a_k)(b_j-b_k)-0.\\&=\boxed{\sum_{1\leq j,k\leq n}a_jb_j}-\sum_{1\leq j,k\leq n}a_jb_k-\sum_{1\leq j,k\leqslant n}a_kb_j+\boxed{\sum_{1\leqslant j,k\leqslant n}a_kb_k}\\&=\boxed{2n\sum_{1\leqslant k\leqslant n}a_kb_k}-2\left(\sum_{k=1}^{n}a_k\right)\left(\sum_{k=1}^{n}b_k\right).\end{aligned}$$

那么

$$S=n\sum_{1\leqslant k\leqslant n}a_kb_k-\left(\sum_{k=1}^{n}a_k\right)\left(\sum_{k=1}^{n}b_k\right).$$

我们便得到

$$\left(\sum_{k=1}^{n}a_k\right)\left(\sum_{k=1}^{n}b_k\right)=n\sum_{k=1}^{n}a_kb_k-\sum_{1\leqslant j<k\leqslant n}(a_k-a_j)(b_k-b_j)$$

这是 Chebyshev 单调不等式的特例.

Chebyshev 单调不等式是说, 如果 $a_1\leqslant a_2\leqslant\cdots\leqslant a_n, b_n\leqslant b_2\leqslant\cdots\leqslant b_n$, 那么

$$\left(\sum_{k=1}^{n}a_k\right)\left(\sum_{k=1}^{n}b_k\right)\leqslant n\sum_{i=1}^{n}a_kb_k$$

反之亦然.

2.2. 与一重求和中第三条的联系. 回顾定理 1.1中的第三条 ($\sum_{k\in K}a_k=\sum_{p(k)\in K}a_{p(k)}$, $p(k)$ 是一个原集合的排列). 如果现在将 k 换做 $f(j)$, 其中 f 是一个 $J\to K$ 的映射, 那么

$$\sum_{j\in J}a_{f(j)}=\sum_{k\in K}a_k\#f^{-}(k).$$

其中 $f^{-}(k)=\{j\mid f(j)=k\}$, $\#S$ 是集合 S 的元素个数. 通俗地说, 就是对 $j\in J,\quad f(j)=k$ 的数量.

证明. 直接展开:

$$\begin{aligned}\sum_{j\in J}a_{f(j)}=\sum_{\substack{j\in J\\k\in K}}a_k[f(j)=k]&=\sum_{k\in K}a_k\sum_{j\in J}[f(j)=k]\\&=\sum_{k\in k}a_k\#f^{-}(k).\end{aligned}$$

□

例子 2.6. 若 f 是一个一对一的函数, 那么 $\#f^{-}(k) = 1$, 也就是

$$\sum_{j\in J} a_{f(j)} = \sum_{f(j)\in k} a_{f(j)} \cdot 1 = \sum_{k\in k} a_k.$$

例子 2.7. 求算

$$S_n = \sum_{1\leqslant j<k\leqslant n} \frac{1}{k-j}$$

$$\begin{aligned}
S_n &= \sum_{1\leqslant k\leqslant n} \sum_{1\leqslant j\leqslant k} \frac{1}{k-j} \\
&\overset{j:=k-j}{=\!=\!=} \sum_{1\leqslant k\leqslant n} \sum_{1\leqslant k-j<k} \frac{1}{j} \\
&= \sum_{1\leqslant j\leqslant n} \sum_{0<j\leqslant k-1} \frac{1}{j} \\
&= \sum_{1\leqslant k\leqslant n} H_{k-1} \\
&\overset{k:=k+1}{=\!=\!=} \sum_{0\leqslant k<n} H_k
\end{aligned}$$

如果在换元之前进行代换, 就有

$$\begin{aligned}
S_n &= \sum_{1\leqslant j<k\leqslant n} \frac{1}{k-j} \overset{k:=k+j}{=\!=\!=} \sum_{1\leqslant j\leqslant k+j\leqslant n} \frac{1}{k} = \sum_{1\leqslant k\leqslant n} \sum_{1\leqslant j\leqslant n-k} \frac{1}{k} \\
&= \sum_{1\leqslant k\leqslant n} \frac{n-k}{k} = \sum_{1\leqslant k\leqslant n} \frac{n}{k} - \sum_{1\leqslant k\leqslant n} 1 \\
&= n\left(\sum_{1\leqslant k\leqslant n} \frac{1}{k}\right) - n = nH_k - n.
\end{aligned}$$

§3 差分与微分、求和与积分

本节的目标是, 是不是有这样的一种记号, 使得我们可以使用类似于积分的方式记录求和? 是否能够像不定积分那样引申出 "不定求和"?

3.1. 回忆: 微分与差分. 数学分析中我们定义过微分算子

$$\mathscr{D}f(x) = \frac{df(x)}{dx} = \lim_{h\to 0}\frac{f(x+h)-f(x)}{h}.$$

同样的可以定义差分算子.

定义 3.1 (差分)**.** 对于一个数列, 定义其差分算子 $\Delta f(x)$ 为

$$\Delta f(x) := f(x+1) - f(x).$$

注:

1. 实际上这是极限定义中, $h = 1$ 的特例.
2. 算子作用在函数上, 给出新的函数. 其本质就是从函数到函数的映射. 例如在多项式上面求导算子给出的实际上是映射

$$\mathbb{P}[x]_n \xrightarrow{\mathscr{D}} \mathbb{P}[x]_{n-1}.$$

例子 3.1. 回忆 $\mathscr{D}x^m = mx^{m-1}$, 其中 m 是给定的数, x 是变元.

如果对它施以差分算子, 得到 $\Delta x^3 = (x+1)^3 - x^3 = 3x^2 + 3x + 1$. 这表明求导算子和差分算子他们之间的性质有所不同. 接下来考察特殊的多项式, 使得差分算子得到类似的结构.

例子 3.2 (下降幂)**.** 如果我们求

$$\Delta(\underbrace{x(x-1)\cdots(x-m+1)}_{m\text{ 项}})$$

就会得到

$$\begin{aligned}
&\Delta(\underbrace{(x(x-1)\cdots(x-m+1))}_{m\text{ 项}}\\
&= (x+1)x\cdots(x-m+2) - x(x-1)\cdots(x-m+1)\\
&= m\underbrace{(x-1)\cdots(x-m+2)}_{m-1\text{ 项}}
\end{aligned}$$

说明 Δ 算子在上述的多项式上面作用有与求导算子在 x^m 次多项式有类似的效果.

我们先定义

$$\underbrace{x(x-1)\cdots(x-m+1)}_{m\text{ 项}}$$

为 x 的 m 次下降幂. 记作 $x^{\underline{m}}$

命题 3.1. 由于上面的性质, 注意到

$$\Delta x^{\underline{m}} = m x^{\underline{m-1}}.$$

我们发现, $\mathscr{D}$ 有逆运算 $\int$, 意味着 $\int \cdot \mathscr{D} = \mathrm{id}$. 并且微积分基本定理告诉我们:

$$g(x) = \mathscr{D} f(x) \Leftrightarrow \int g(x)dx = f(x) + C.$$

定理 3.2 (差分基本定理)**.** 对于差分而言,

$$g(x) = \Delta f(x) \Leftrightarrow \sum g(x)\delta x = f(x) + C.$$

这里 $\sum g(x)\delta x$ 是 $g(x)$ 的不定和式, 是差分等于 $f(x)$ 的函数类. 其中 C 是满足 $p(x+1) = p(x)$ 的任意一个函数 $p(x)$.

正如微积分中有定积分这件事情, 有限和式也可以仿照写作 "定和式". 仿照 $\int_a^b g(x)dx = [f(x)]_a^b = f(b) - f(a)$. 我们给出下面的定义.

定义 3.2 (定和式)**.** 定义

$$\sum_a^b g(x)\delta x = [f(x)]_a^b = f(b) - f(a).$$

如果 $g(x) = \Delta f(x)$.

命题 3.3. 对于这样的一个和式, 有

$$\sum_a^b g(x)\delta x = \sum_{k=a}^{b-1} g(k) = \sum_{a \le k < b} g(k).$$

证明. 若 $b = a$, $\sum_a^a g(x)\delta x = f(a) - f(a) = 0$; 若 $b := a+1$, $\sum_a^b g(x)\delta x = f(a+1) - f(a) = g(a)$. 若 $b := b+1$,

$$\begin{aligned}\sum_a^{b+1} g(x)\delta x - \sum_a^b g(x)\delta x &= (f(b+1) - f(a)) - (f(b) - f(a)) \\ &= f(b+1) - f(b) = g(b).\end{aligned}$$

根据数学归纳法可以证明. □

证明. 另外的证明: 考虑裂项 (伸缩法, telescoping)

$$\begin{aligned}\sum_{a \le k < b} f(k+1) - f(k) =& (f(a-1) - f(a)) + (f(a+2) - f(a+1)) + \cdots \\ & + (f(b) - f(b-1)) \\ =& f(b) - f(a).\end{aligned}$$

□

命题 3.4. 像定积分那样, 不定求和有

$$\sum_a^b g(x)\delta x = -\sum_b^a g(x)\delta x.$$

证明. 实际上

$$f(b)-f(a) = -(f(a)-f(b)) = -\sum_b^a g(x)\delta x.$$

□

命题 3.5. 像定积分那样, 不定求和有

$$\sum_a^b g(x)\delta x + \sum_b^c g(x)\delta x = \sum_a^c g(x)\delta x.$$

3.2. 带来的好处: 有限微积分.

a) 下降幂 下面给 $m<0$ 的时候下降幂给出定义. 观察

$$\begin{array}{ll} x^3 = x(x-1)(x-2) & \text{除以}x-2 \\ x^2 = x(x-1) & \text{除以}x-1 \\ x^2 = x & \text{除以}x \\ x^{-0} = 1 & \text{除以}x+1 \\ x^{-1} = \frac{1}{x+1} & \text{除以}x+2. \\ x^{-2} = \frac{1}{(x+1)(x+2)} & \end{array}$$

定义 3.3 (下降幂)**.** 我们定义

$$\underbrace{x(x-1)\cdots(x-m+1)}_{m\text{ 项}}$$

为 x 的 m 次下降幂. 记作 $x^{\underline{m}}$.

如果 $m>0$,

$$m>0, x^{\underline{-m}} = \frac{1}{(x+1)\cdots(x+m)}$$

像这样的扩展定义仍然保留了对应的性质.

命题 3.6 (下降幂的性质)**.**

$$\forall m,n\in\mathbb{Z},\quad x^{m+n} = x^m x^n.$$

命题 3.7 (下降幂的满足差分的性质)**.** 若 $m>0, \Delta x^{\frac{-m}{n}} = -mx^{-m-1}$.

因此可得, 对于下降幂, 有

$$\sum_a^b x^m\delta x = \left[\frac{x^{m+1}}{m+1}\right]_a^b \quad m\neq -1.$$

b) 调和级数 根据上述的描述, 如果 $m=-1$, $\quad x^{-1}=\frac{1}{x+1}=\Delta f(x)=f(x+1)-f(x)$. 若 $x\in\mathbb{Z}, f(x)=\frac{1}{1}+\frac{1}{2}+\cdots+\frac{1}{x}=H_n$. 所以

命题 3.8. 类似与积分的情形,

$$\sum_a^b x^m \delta x = \begin{cases} \left[\frac{x^{m+1}}{m+1}\right]_a^b & ,m\neq -1 \\ [H_x]_a^b, & m=1 \end{cases}$$

c) 指数函数的类似物 由于 $\mathscr{D}e^x=e^x$, 希望找一个 $\Delta f(x)=f(x)$. 实际上,

$$\begin{aligned} & f(x+1)-f(x)=f(x)\Leftrightarrow f(x+1)=2f(x), \text{ 即} \\ & f(x)=2^x. \end{aligned}$$

对于 $\Delta c^x=c^{x+1}-c^x=(c-1)c^x$. 若 $c\neq 1$, 那

命题 3.9. 等比数列的求和

$$\sum_{a\leqslant k<b} = \sum_a^b c^x\delta x = \left[\frac{c^x}{c-1}\right]_a^b = \frac{c^b-c^a}{c-1}.$$

d) 差分表和加法, 乘法的差分 我们希望得到如下的差分表:

$f=\sum g$	$\boxed{\Delta f}=g.$
$x^{\underline{m}}$	$mx^{\underline{m-1}}$
2^x	2^x
c^x	$(c-1)c^x$
$c\cdot u$	$c\cdot\Delta u$
$u+v$	$\Delta u+\Delta v$
uv	$u\Delta v+Ev\Delta u$ (见下文)

具体地, 我们需要看加法, 乘法的情形下, 差分的变化. 注意, 复合函数差分在离散的情形没有很好的对应物.

3.3. 分部求和法. 注意到

$$\begin{aligned} & \Delta(u(x)v(x))=u(x+1)v(x+1)-u(x)v(x) \\ & =u(x+1)v(x+1)-u(x)v(x+1) \\ & +u(x)v(x+1)-u(x)v(x) \\ & =u(x)\Delta v(x)+v(x+1)\Delta u(x) \end{aligned}$$

若定义 $Ef(x):=f(x+1)$, 那么

$$\Delta(uv)=u\Delta v+(Ev)\Delta u.$$

此时, 对两边同时取 $\sum$, 有

$$\sum u\Delta v = uv - \sum (Ev)\Delta u.$$

注:　实际上, 这个还有另一种选择方式, 如

$$\begin{aligned}\Delta(uv) &= u\Delta v + Ev\Delta u\\ &= Eu\Delta v + v\Delta u\end{aligned}$$

两种形式都是正确的, 因此左右是对称的.

例子 3.3. 仿照 $\int xe^x dx$, 求 $\sum x2^x\delta x$

$$\begin{aligned}\sum x\boxed{2^x}\delta x = \sum x\delta\left(2^x\right) &= x2^x - \sum 2^{\boxed{x+1}}\delta x\\ &= x2^x - 2^{x+1} + c.\end{aligned}$$

那么

$$\begin{aligned}\sum_{k=0}^{n} k2^k = \sum_{0}^{n+1} x2^x\delta x &= \left[x2^x - 2^{x+1}\right]_0^{n+1}\\ &= (n-1)2^{n+1} + 2.\end{aligned}$$

例子 3.4. 仿照 $\int x\ln x dx$, 求 $\sum kH_k\delta k$,

$$\begin{aligned}\sum \boxed{x}H_x\delta x &= \sum H_x\delta\left(\frac{x^{\underline{2}}}{2}\right)\\ &= \frac{x^2}{2}H_x - \sum\frac{(x+1)^{\underline{2}}}{2}x^{\underline{-1}}\delta x\\ &= \frac{x^2}{2}H_x - \frac{1}{2}\sum x^{\underline{1}}\delta x\\ &= \frac{x^2}{2}H_x - \frac{x^2}{4} + C.\end{aligned}$$

那么

$$\sum_{0\leqslant k<n} kH_k = \left[\frac{x^2}{2}H_x - \frac{x^2}{4} + C\right]_0^n = \frac{n^{\underline{2}}}{2}\left(H_n - \frac{1}{2}\right).$$