哈希与数据摘要

用少量空间组织和概括数据

用少量随机性构造紧凑的数据结构

哈希函数为每个键分配一个随机标签，同一个键始终对应同一个标签。只要控制好 这些标签之间的独立性，就可以用它们构造精确字典、近似成员查询结构，以及 不同元素计数和元素频率的数据摘要。本讲始终围绕两个问题展开：分析究竟 用到了随机函数的哪些性质？怎样用一个很短的随机种子保留这些性质？

高级算法课程教学团队

南京大学

学习目标与阅读建议

本讲接续上一讲的随机指纹，进一步讨论如何把随机化用于数据的存储、查询和统计。读完本讲，应当能够：

1. 区分碰撞意义下的通用性与两两独立性，并构造只需短随机种子的哈希族；
2. 从碰撞对数的期望上界出发，推导 FKS 完美哈希的空间和时间保证；
3. 推导 Bloom 过滤器的启发式公式，指出独立性假设的漏洞，并区分启发式分析与严格保证；
4. 证明基于最小值、末尾连续零和最小 k 个哈希值的不同元素计数摘要的误差界；
5. 推导 Count-Min 摘要的加性误差保证，并同时核算计数器和随机种子的空间。

第一次阅读可先掌握第 1节的碰撞分析和第 2节的通用哈希、两两独立哈希构造，再学习第 3–4节的字典与过滤器。进入第 5节后，先用理想哈希模型理解最小值估计，再研究有限位实现的尾零摘要和 bottom-k 摘要，最后学习第 6节的 Count-Min 摘要。

掌握主线后，可以回过头来细读更高阶独立性的构造、第 4.5节中 Bloom 过滤器的初等补充分析，以及各节的研究拓展。Bloom 过滤器的正文先做启发式计算，再讨论独立性问题和精确公式；后续课程还将用测度集中重新分析这个问题。附录只整理有限独立性与偏差不等式两类数学工具；有限域基础沿用第二讲“随机指纹”，构造哈希函数时会简要回顾所需事实。每个问题的记号和计算模型均在相应部分引入。

目录

1 随机映射、碰撞与生日悖论

1.1 模型与记号

设键的全集为 $\mathcal{U}$，大小为 $N \geq 2$，并约定 $[r] = \{1, \dots, r\}$。*哈希函数*（hash function）$h : \mathcal{U} \to [r]$ 为每个键指定一个哈希值；值域通常比全集小，同一个键每次得到的哈希值必须相同。需要表示模 r 的剩余类时，我们显式写作 $\{0, \dots, r-1\}$。

若 h 从全部映射中均匀随机选取，那么任意多个*互异*键的哈希值相互独立，且各自在 $[r]$ 上均匀分布。这就是*简单均匀哈希假设*（simple uniform hashing assumption，SUHA）。这里先将输入键固定，再随机选择映射；概率描述的是随机映射的行为。

在这一模型中，计算哈希值被视为一种理想的基本操作。若真的把整张函数表存下来，则需要 $N\lceil \log_2 r \rceil$ 位。后面构造短种子哈希族，正是为了保留算法所需的随机性，同时避免存储整张函数表。

1.2 生日碰撞概率

固定 $n \geq 1$ 个互异键，将它们用均匀随机映射哈希到 $m \geq 1$ 个值。这等价于把 n 个球独立均匀地投入 m 个桶：每个球代表一个互异的键，每个桶代表一个哈希值。同一个键重复出现时仍落入原来的桶，并不相当于再独立投一次球。当 $n \leq m$ 时，设 $\mathcal{E}$ 表示每个桶至多含一个球，则

$$\mathbb{P}(\mathcal{E}) = \frac{m(m-1)\cdots(m-n+1)}{m^n} = \prod_{i=0}^{n-1}\left(1 - \frac{i}{m}\right). \tag{1}$$

分子是在计算单射的个数。也可以逐步条件化：若前 i 个球尚未碰撞，则第 $i+1$ 个球避开已占用桶的概率为 $1 - i/m$，再用概率的乘法公式便得到上述乘积。当 $n > m$ 时，不碰撞的概率为零。

先看这一乘积揭示的数量级。用 $1 - u \approx e^{-u}$ 逐项近似，得到 $\mathbb{P}(\mathcal{E}) \approx e^{-n(n-1)/(2m)}$。指数的绝对值在 n 达到 $\sqrt{m}$ 量级时变为常数，因此约 $\sqrt{m}$ 个键就可能产生碰撞，而不必等到接近 m 个键。这便是*生日悖论*的直觉。

这一近似的适用范围可以精确说明。对 $n \leq m$，令 $\rho = (n-1)/m < 1$、$A = n(n-1)/(2m)$。利用初等不等式

$$-x - \frac{x^2}{2(1-\rho)} \leq \ln(1-x) \leq -x \qquad (0 \leq x \leq \rho),$$

对各项求和，得到

$$-A - \frac{n(n-1)(2n-1)}{12m^2(1-\rho)} \;\leq\; \ln \mathbb{P}(\mathcal{E}) \;\leq\; -A. \tag{2}$$

例如，当 $n^3/m^2 \to 0$ 时，上述指数近似的相对误差趋于零。特别地，若 $n/\sqrt{m} \to c > 0$，则不碰撞的概率趋于 $e^{-c^2/2}$，从而严格验证了前面的数量级判断。以一年 365 天、生日独立均匀分布为例，58 名学生中至少两人生日相同的概率约为 0.9917。

同一个随机映射还引出其他问题：所有桶是否都被占用，对应优惠券收集问题；最大的桶含有多少球，对应最大负载问题。本讲关注的是碰撞，以及怎样利用碰撞分析构造字典与摘要。

1.3 只估计碰撞对数，需要多少随机性？

对互异的键 $x_1, \dots, x_n$，定义碰撞对数

$$C = \sum_{1 \le i < j \le n} \mathbf{1}\{h(x_i) = h(x_j)\}.$$

只要每一对键的碰撞概率至多为 $1/m$，由期望的线性性和 Markov 不等式（见附录 B）便有

$$\mathbb{E}C \le \frac{\binom{n}{2}}{m}, \qquad \mathbb{P}(C \ge 1) \le \frac{\binom{n}{2}}{m}. \tag{3}$$

这里完全不需要各个碰撞指示变量相互独立。特别地，取 $m = n^2$ 后，一个固定集合无碰撞的概率严格大于 $1/2$。

完美哈希的构造只需用到这一碰撞概率上界。它并没有刻画完整的生日碰撞分布，也不能说明键数超过 $\sqrt{m}$ 后碰撞很可能发生；这些更强的结论需要更强的随机性假设。

2 用短种子描述哈希函数

实现哈希函数时，可以先随机抽取一段较短的*种子*（seed），再由种子确定一个函数。种子只抽取一次，此后所有求值都使用这个函数。因此，所谓随机哈希族，既包括可供选择的函数，也包括选择它们的概率分布。我们不再要求保留完全随机映射的全部性质，而是先问算法的分析究竟需要什么。

2.1 通用性与两两独立性

定义 2.1。 设 $h : \mathcal{U} \to [m]$ 按某个给定分布选取。若对任意 $x \neq y$ 都有

$$\mathbb{P}[h(x) = h(y)] \le 1/m,$$

则称该分布给出一个*通用哈希族*（universal hash family），这里的通用性专指碰撞概率。若对任意 $x \neq y$ 和 $a, b \in [m]$ 都有

$$\mathbb{P}[h(x) = a,\ h(y) = b] = 1/m^2,$$

则称该哈希族*两两独立且均匀*（pairwise independent and uniform）。

第二种性质也称*强 2-通用性*（strong 2-universality）。将联合概率对 $a = b$ 求和可知，两两独立且均匀蕴含通用性，反过来则不成立。两者的差别在于：通用性只控制两个键撞到一起的概率，两两独立性还规定它们各自的分布及整个联合分布。通用哈希由 Carter 和 Wegman 提出 [1]。一般独立性及其在确定性变换下的性质见附录 A。

2.2 有限域上的仿射哈希

设 $\mathbb{F}_q$ 是含 q 个元素的域，全集已单射嵌入 $\mathbb{F}_q$。独立均匀地选取 $a, b \in \mathbb{F}_q$，*允许* $a = 0$，并定义

$$h_{a,b}(x) = ax + b.$$

给定互异的 x, y 以及任意目标值 u, v，方程组 $ax + b = u$、$ay + b = v$ 有唯一解

$$a = (u - v)(x - y)^{-1}, \qquad b = u - ax.$$

因此，在 q^2 个等可能种子中，恰有一个产生指定的输出对。这就证明了该哈希族两两独立且均匀。若 q 是 2 的幂，种子长度为 $2\log_2 q$ 位。

第二讲“随机指纹”的有限域工具中，已经介绍了这里用到的基本事实：非零域元素都有乘法逆元，$\mathbb{F}_{2^w}$ 的元素可用 w 位表示。计算时仍须区分域运算与整数运算：前者的加法是逐位异或，乘法是多项式相乘后对固定的不可约多项式取模，并非模 2^w 的整数乘法。有限域的构造与表示详见第二讲。

2.3 映射到任意大小的哈希表

设 $\mathcal{U} = \{0, \dots, N-1\}$，选定素数 $p \geq N$。从 $\{1, \dots, p-1\}$ 中均匀选取 a，从 $\{0, \dots, p-1\}$ 中独立均匀选取 b，定义

$$g_{a,b}(x) = ((ax + b) \bmod p) \bmod m.$$

种子占 $O(\log p)$ 位。对任意 $x \neq y$，第一次取模后得到的有序对均匀分布在全部 $p(p-1)$ 个互异剩余类对上。

写成 $p = qm + r$，其中 $0 \leq r < m$。按模 m 的余数分组，每组含 q 或 $q+1$ 个数，因此落入同一组的有序互异对共有

$$A = mq(q-1) + 2rq.$$

又因为

$$p(p-1) - mA = mq(m-1) + r(r-1) \geq 0,$$

故碰撞概率 $A/[p(p-1)] \leq 1/m$，该构造确实是通用哈希族。这一计数也说明，碰撞概率界并不要求输出严格均匀。

再次取模不保持均匀性与独立性

上述构造通常不满足两两独立。即便 $m = p$，两个互异键的输出也永不相等，而两个独立均匀的值相等的概率应当为 $1/p$。若 m 不整除 p，单个输出的边缘分布通常也不均匀。

当 $N > 1$ 时，可以预先固定一个满足 $N \leq p < 2N$ 的素数。选定域及其参数属于初始化步骤；后面字典的构造时间均从已有合适哈希族开始计算。

2.4 小值域上的精确独立性

在需要均匀位串或均匀随机阈值的算法中，仅有碰撞概率上界还不够。我们希望在缩小值域的同时保留均匀性和两两独立性；任意取模做不到这一点，但均衡投影可以做到。

若 $q = 2^w$、$m = 2^\ell$ 且 $\ell \leq w$，可以把域元素写成 w 位二进制串，再保留 $ax + b$ 的某 ℓ 个固定坐标。每个输出串恰有 $2^{w-\ell}$ 个原像，因此这一均衡投影把均匀域元素变成均匀位串；分别作用于相互独立的域元素，也会保持独立性。这样便得到值域大小为 2 的幂的两两独立哈希族。取 $w \geq \lceil \log_2 N \rceil$ 且 $w \geq \ell$，种子只需 $O(\log N + \log m)$ 位。

2.5 更高阶独立性与多项式哈希

两两独立性足以分析后面采用有限哈希族的估计器，不过短种子的思想还可以推广：如果一个分析同时涉及更多个键，应当怎样控制它们的联合分布？

若对任意 $t \le k$ 个互异输入，它们的哈希值都相互独立，且各自在值域上均匀分布，则称该哈希族*k 阶独立且均匀*（k-wise independent and uniform）。要构造 k 阶独立的哈希族，先将全集单射嵌入大小为 $q \ge N$ 的有限域 $\mathbb{F}_q$，再独立均匀地选取系数 $a_0, \dots, a_{k-1} \in \mathbb{F}_q$，使用

$$h(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}.$$

这里多项式的次数至多为 $k-1$。下面直接证明所得哈希族具有 k 阶独立性。固定 $t \le k$ 个互异输入 $x_1, \dots, x_t$，构造次数为 $t-1$ 的 Lagrange 基多项式

$$\ell_i(X) = \prod_{\substack{j \in [t] \\ j \ne i}} \frac{X - x_j}{x_i - x_j}.$$

输入互异，保证了每个分母都非零。代入这些输入可见：$\ell_i(x_i) = 1$，而在其他 x_j 处取值均为零。因此，对任意指定的输出 $y_1, \dots, y_t$，多项式 $\sum_i y_i \ell_i(X)$ 都能在各点取得相应的值，且次数不超过 $t-1 \le k-1$。

这说明从 k 个系数到 t 个输出的求值映射是线性满射，其核的维数为 $k-t$。每组输出因而恰有 q^{k-t} 组系数作为原像，出现概率为 q^{-t}，独立性得证。整个函数用 $k \log_2 q$ 位描述，按 Horner 法则求值需要 $O(k)$ 次域运算。前面的均衡投影同样保持 k 阶独立性。

文献对 “k-universal” 有不同约定；本讲用“通用”表示两两碰撞概率界，需要联合均匀分布时则明确说“k 阶独立”。另一个较弱的性质是 k 个互异键全部碰撞的概率满足

$$\mathbb{P}[h(x_1) = \cdots = h(x_k)] \le m^{-(k-1)}.$$

它可由 k 阶独立推出：对每个共同输出值，联合概率为 m^{-k}，再对 m 个值求和。但这一碰撞界本身弱于 k 阶独立。

3 精确成员查询：FKS 完美哈希

给定一个固定集合 $S \subseteq \mathcal{U}$，记 $n = |S|$。*静态字典*（static dictionary）的任务是存储 S，并精确回答给定的键是否属于 S。如果一个哈希函数在 S 上是单射，即集合内不同的键不会碰撞，就称它是 S 上的*完美哈希函数*（perfect hash function）。下面先用无碰撞映射得到一个平方空间的字典，再用两层结构将空间降为线性。

本节的时间复杂度在 word-RAM 模型下计算：每个机器字含 $\Theta(\log N)$ 位，常数个机器字上的算术和内存访问各计为一步，并假设所选哈希族能够常数时间求值。合适的哈希族及其域参数已经选定，构造过程中再随机抽取种子；下面核算字典本身的构造和查询代价，空间则包含存储的键、辅助信息和种子。空集合只需一个空标记，下面先讨论 $n \ge 1$ 的情形。

3.1 从平方空间的简单构造开始

先选取一个通用哈希函数，把集合映射到 n^2 个槽位。由式 (3)，随机选择的函数以严格大于 $1/2$ 的概率在该集合上无碰撞。可以逐个插入键来检测碰撞；一旦发现碰撞，就重新选择函数。找到无碰撞的函数后，把每个键 x 存入槽位 $h(x)$。

查询时必须将槽位中存储的键与 x 比较，不能只检查该位置是否非空：不属于集合的键也可能与集合中的键映射到同一位置。未使用的槽位要带有单独的空标记。这样便得到一个查询时间为 $O(1)$ 的精确字典，代价是需要 $O(n^2)$ 个槽位。

3.2 两层结构如何把空间降为线性？

Fredman、Komlós 和 Szemerédi 提出的 FKS 方案 [2]先用通用哈希 $h : \mathcal{U} \to [n]$ 把 S 分到 n 个桶中。记

$$B_i = \{x \in S : h(x) = i\}, \qquad b_i = |B_i|, \qquad Q = \sum_{i=1}^{n} b_i^2.$$

令 C 为第一层哈希产生的碰撞对数，即 $C = \sum_i \binom{b_i}{2}$。对每个桶内的碰撞对计数，可得一个对任何分桶结果都成立的恒等式：

$$Q = \sum_i \left(b_i + 2\binom{b_i}{2} \right) = n + 2C.$$

由此立即得到

$$\mathbb{E}Q \le n + 2\frac{\binom{n}{2}}{n} = 2n - 1 < 2n. \tag{4}$$

Markov 不等式进一步给出 $\mathbb{P}(Q > 4n) < 1/2$。因此，只需期望常数次尝试，就能找到满足 $Q \le 4n$ 的第一层哈希函数。随后为每个桶 B_i 单独建立一个含 b_i^2 个槽位、在该桶内无碰撞的第二层哈希表。

FKS 字典的构造与查询

构造：输入为含 n 个互异键的集合。

1. 若 $n = 0$，存储一个空字典并结束构造。
2. 反复独立选取映射到 $[n]$ 的通用哈希函数，直到 $\sum_i b_i^2 \le 4n$。
3. 对每个非空桶 B_i，反复独立选取通用哈希函数 $h_i : \mathcal{U} \to [b_i^2]$，直到它在 B_i 上无碰撞。记录其种子、指向第二层表的指针，并在表中存储各个键。若桶中只有一个键，直接存储该键即可。

查询 x：若字典为空，回答“不属于”。否则计算 $i = h(x)$。若 B_i 为空，回答“不属于”；若桶中只有一个键，直接将它与 x 比较；否则检查 $h_i(x)$ 对应的第二层槽位，恰在其中存储的键等于 x 时回答“属于”。

定理 3.1 (FKS 静态字典)。 设已有合适的通用哈希族，且哈希函数能够常数时间求值。上述构造占用 $O(n)$ 个机器字，期望构造时间为 $O(n)$，构造完成后每次成员查询都正确，且最坏情况下只需 $O(1)$ 时间。每个机器字含 $\Theta(\log N)$ 位。

证明. 查询时比较完整的键，保证了答案精确。一次查询至多计算两个哈希值，再进行常数次内存访问，故最坏查询时间为 $O(1)$。

被接受的第一层哈希函数保证第二层总共至多使用 $4n$ 个槽位。各桶的大小、指针和第二层种子还需 $O(n)$ 个机器字。由于最终布局中每个 $b_i^2 \le 4n$，全部地址均可用 $O(\log N)$ 位表示。

每次尝试第一层哈希时，初始化桶计数并分配全部键需 $O(n)$ 时间，成功概率严格大于 $1/2$。第一层分桶确定后，对任意固定非空桶，一次全新的第二层尝试失败的概率至多为

$$\frac{\binom{b_i}{2}}{b_i^2} < \frac{1}{2}.$$

即使每次尝试都清空整个第二层表，时间也只是 $O(b_i^2)$。因此，以已接受的第一层分桶为条件，全部第二层表的期望构造时间为 $O(\sum_i b_i^2) = O(n)$。加上第一层的工作，即得总期望时间界。 □

这是一种 *Las Vegas 算法*：随机性影响的是构造所需时间，而构造完成后的字典对所有查询均给出精确答案。“期望时间”仅针对预处理，不影响查询的最坏情况保证。

3.3 动态字典与简洁表示

FKS 面向固定集合。若还要支持集合的更新，就需要动态完美哈希。Dietzfelbinger 等人 [3] 给出的结构使用线性机器字空间，查询的最坏时间为常数，每次更新的均摊期望时间为常数；其构造不同于上面的静态 FKS 方案。

空间还可以从位数而非机器字数的角度衡量。若一种精确表示方法能够表示任意 n 元集合，就必须区分 $\binom{N}{n}$ 种可能，因而至少需要 $\lceil \log_2 \binom{N}{n} \rceil$ 位。通常逐一存储键的字典使用 $O(n \log N)$ 位；当 n 与 N 的相对大小不同时，这未必达到上述信息论下界。

进一步的研究关注*简洁动态字典*（succinct dynamic dictionary）：当空间接近信息论下界时，操作时间需要付出多少代价？对于键和值均占 $\Theta(\log n)$ 位的情形，Bender 等人 [4]给出了更新时间 $O(k)$ 与每个键 $O(\log^{(k)} n)$ 位冗余之间的权衡，同时保持常数查询时间，并给出高概率保证。这里 $\log^{(k)}$ 表示迭代 k 次对数，k 可取到 $\log^* n$。在相应冗余范围内，Li 等人 [5] 证明插入、删除、查询三种操作中，至少一种的期望时间具有匹配的 $\Omega(k)$ 下界。其单元探测模型（cell-probe model）设全集大小为 $N = n^{1+\Theta(1)}$，字长为 $\Theta(\log N)$，且两次内存访问之间的计算免费。对于存储键值对的版本，还有不依赖查询时间的更新下界。这些时间与空间的权衡依赖具体模型，其证明超出本讲的入门范围。

4 近似成员查询：Bloom 过滤器

给定固定集合 $S \subseteq \mathcal{U}$，记 $n = |S| \geq 1$。*过滤器*（filter）通过允许*假阳性*（false positive）来节省空间：对不在集合中的键，它可以误答“属于”；对已存储的键，则必须回答“属于”。以下固定 $0 < \delta \leq 1/2$，目标是对每个事先固定的非成员 $x \notin S$，将假阳性概率控制在 δ 以内。集合 S 和查询键 x 均与哈希种子无关。这是单次查询的概率保证；它并不表示所有非成员都能同时得到正确回答。

4.1 一个位数组与一个哈希函数

将一个 m 位数组初始化为全零，对每个 $x \in S$，将第 $h(x)$ 位设为 1。查询 x 时，只检查这一位。若使用理想的均匀哈希函数，固定非成员的哈希值避开所有 n 个已存储键的概率为 $(1 - 1/m)^n$，因此

$$\mathbb{P}(\text{假阳性}) = 1 - (1 - 1/m)^n.$$

若只假设哈希族通用，使用并合界也可得到上界 n/m。因此，取 $m \geq n/\delta$ 即足以保证所需错误率；在理想哈希模型下，当 δ 较小时，精确概率表达式也给出这一空间量级。通过重复检验，可以改善空间对 δ 的依赖。

4.2 共享数组的 Bloom 过滤器：先做启发式分析

Bloom 的构造 [6]使用 $k \geq 1$ 个哈希函数 $h_1, \ldots, h_k : \mathcal{U} \to [m]$，共用一个位数组。以下对共享数组的分析均采用理想模型：每个函数完全随机，各函数相互独立。因此，插入 n 个互异键所产生的 kn 个哈希值，可以看成向 m 个位置进行的 kn 次独立均匀投球。

Bloom 过滤器

将 $A[1],\dots,A[m]$ 初始化为 0，并选取哈希函数 $h_1,\dots,h_k$。插入键 x 时，对每个 $j\in[k]$ 都设置 $A[h_j(x)]=1$。查询 x 时，当且仅当所检查的 k 位均为 1，才回答“属于”。

这个结构没有假阴性（false negative）。为了估计假阳性概率，我们先看数组中的一个固定位置。只有全部 kn 次插入哈希都避开它，这一位才会保持为零。因此，它变为 1 的概率恰为

$$p_1 = 1-\left(1-\frac{1}{m}\right)^{kn}.$$

对一个固定的非成员，它的每次查询哈希也都以概率 p_1 命中值为 1 的位置。*如果暂且把这 k 次检查返回阳性的事件视为相互独立*，就得到如下启发式计算：

$$\mathbb{P}(\text{假阳性}) \overset{\text{启发式}}{\approx} p_1^k = \left(1-\left(1-\frac{1}{m}\right)^{kn}\right)^k \approx \left(1-e^{-kn/m}\right)^k. \tag{5}$$

这里有两步，性质并不相同：将 k 个边缘概率相乘，依赖的是尚未证明的独立性；将 $(1-1/m)^{kn}$ 换成 $e^{-kn/m}$，则是一个解析近似。稍后我们会仔细检查第一步。

启发式公式给出的参数选择。 记 $c=m/n$，即平均为每个已存储的键分配的数组位数。上述公式变为

$$\mathbb{P}(\text{假阳性}) \approx (1-e^{-k/c})^k. \tag{6}$$

固定 c，暂时允许 k 取正实数，右侧在 $k=c\ln 2$ 时达到最小。此时预计约有一半的位为 1，假阳性概率约为

$$\exp[-c(\ln 2)^2] \approx (0.6185)^c, \qquad m \approx \frac{n\ln(1/\delta)}{(\ln 2)^2}.$$

实际实现中 k 必须是正整数，可比较实数最优值两侧相邻的合法整数；若实数最优值小于 1，则取 $k=1$。这就解释了常用的参数设计规则，但我们还没有得到有限规模下的严格错误率保证。

4.3 有条件的独立，为什么还不够？

查询所用的哈希值确实独立，为什么不能直接把概率相乘？关键在于，k 次检查面对的是*同一个随机生成的数组*。

记插入完成后数组中值为 1 的位数为 B。先固定建好的整个数组。对于固定非成员，查询哈希值仍然独立且均匀；在这个条件下，k 次检查返回阳性的事件确实相互独立，每个事件的概率都是 B/m。先在给定数组的条件下相乘，再对数组的随机性取平均，得到准确的恒等式

$$\mathbb{P}(\text{假阳性}) = \mathbb{E}\left[\left(\frac{B}{m}\right)^k\right]. \tag{7}$$

而启发式计算使用的是 $(\mathbb{E}B/m)^k = p_1^k$。条件独立性并不能让我们把期望移到幂运算里面。如果生成的数组恰好有较多的位为 1，那么每一次检查都更容易返回阳性；把不同数组的情况混合起来之后，这些事件便产生了依赖。

事实上，由 $u\mapsto u^k$ 在 $[0,1]$ 上的凸性，

$$p_1^k = \left(\frac{\mathbb{E}B}{m}\right)^k \le \mathbb{E}\left[\left(\frac{B}{m}\right)^k\right].$$

因此，这个乘积其实是*下界*，不能作为一般的有限规模上界。考虑一个很小的例子：$n=1, k=2, m=2$。两次插入哈希以概率 $1/2$ 落到同一位置，此时 $B=1$；否则 $B=2$。于是

$$\mathbb{P}(\text{假阳性}) = \frac{1}{2}\left(\frac{1}{2}\right)^2 + \frac{1}{2}\cdot 1 = \frac{5}{8}, \qquad p_1^k = \left(\frac{3}{4}\right)^2 = \frac{9}{16}.$$

这个差异完全来自独立性假设，与指数函数的近似无关。

4.4 用占用位数的分布精确计算

既然式 (7)是正确的，我们只需求出 B 的分布，就能写出有限规模下的精确公式。记 $\left\{ {t \atop b} \right\}$ 为第二类 *Stirling 数*：把 t 个有标号的对象划分为 b 个非空、无标号的组的方法数。

现在数一数，哪些插入结果恰好占用 b 个位置。先以 $\binom{m}{b}$ 种方式选出这些位置，再把 kn 次有标号的投球划分为 b 个非空组，最后以 $b!$ 种方式将各组分配给选定的位置。总共有 m^{kn} 种等可能的插入结果，因此

$$\mathbb{P}(B=b) = \frac{\binom{m}{b}\, b!}{m^{kn}} \left\{ {kn \atop b} \right\}, \qquad 1 \le b \le \min(m, kn).$$

给定 $B=b$，假阳性概率为 $(b/m)^k$。将两者相乘后对 b 求和，得到

$$\mathbb{P}(\text{假阳性}) = \frac{1}{m^{k(n+1)}} \sum_{b=1}^{\min(m,kn)} b^k \binom{m}{b}\, b! \left\{ {kn \atop b} \right\}. \tag{8}$$

这个公式是严格的，但若想直观理解空间与错误率之间的关系，它又不如启发式公式简洁。Wikipedia 的 Bloom 过滤器假阳性分析同时列出了这两种计算；Bose 等人 [7]对独立性问题作了纠正，并给出了详细分析。

留待后续几讲：用测度集中补全证明。 学过集中不等式之后，我们将回到这个问题。思路是证明：B/m 以很高的概率接近其均值 p_1，然后使用式 (7)中*正确的条件概率计算*。例如，对任意 $\eta > 0$，按照 $B/m \le p_1 + \eta$ 是否成立分成两种情况，即可得到严格上界

$$\mathbb{P}(\text{假阳性}) \le \min\{1, p_1+\eta\}^k + \mathbb{P}(B/m > p_1 + \eta).$$

集中不等式将用来控制最后的偏差概率。特别地，当 $m/n \to c > 0$ 且 k 固定时，我们将证明

$$\mathbb{P}(\text{假阳性}) = (1 - e^{-k/c})^k + o(1) \qquad (n, m \to \infty).$$

这样既能严格说明启发式公式为何给出了正确的渐近预测，也能得到带有明确修正项的概率界。需要区分的是：后续证明并不会让原来的独立性假设成立，也不会把 p_1^k 本身变成有限规模下的上界。

4.5 补充阅读：两个初等的严格估计

上面的分析为后续学习集中不等式留下了一个具体问题。本小节可选读：先用两个初等论证，看看在现阶段已经能证明什么。

放宽常数，立即得到有限规模保证。 无论如何投球，总有 $B \le kn$。因此，只要取

$$k = \lceil \log_2(1/\delta) \rceil, \qquad m = 2kn, \tag{9}$$

式 (7)就至多为 $2^{-k} \le \delta$。这需要 $O(n \log(1/\delta))$ 个数组位，每次操作计算 $O(\log(1/\delta))$ 个哈希值。虽然没有得到启发式分析中的较优常数，但空间与错误率之间的量级关系已经获得严格保证。这里仍将理想哈希函数视为可直接调用的操作，计算的只是数组空间，并未包括完整随机函数的描述空间。

用二阶矩验证渐近公式。 当 k 固定时，估计方差就已足够。设 $m \ge 2$，令 Z_i 表示位置 i 为空的指示变量。对于不同的 i, j，

$$\mathbb{E}[Z_i Z_j] = (1 - 2/m)^{kn} \le (1 - 1/m)^{2kn} = \mathbb{E}Z_i \, \mathbb{E}Z_j.$$

所以，空位指示变量的两两协方差非正。又因为 $B = m - \sum_i Z_i$，且每个指示变量的方差至多为 $1/4$，可得 $\operatorname{Var}(B) \le m/4$。

对 $u, v \in [0,1]$，分解 $u^k - v^k$ 可知 $|u^k - v^k| \le k|u - v|$。先用这一估计，再用 Cauchy–Schwarz 不等式，得到

$$\left| \mathbb{E}\left[(B/m)^k \right] - p_1^k \right| \le k\mathbb{E}|B/m - p_1| \le \frac{k\sqrt{\operatorname{Var}(B)}}{m} \le \frac{k}{2\sqrt{m}}.$$

结合前面证明的下界，便有

$$p_1^k \le \mathbb{P}(\text{假阳性}) \le p_1^k + \frac{k}{2\sqrt{m}}.$$

当 k 固定且 $m/n \to c > 0$ 时，修正项趋于零，而 $p_1 \to 1 - e^{-k/c}$，因而式 (6)在这一范围内成立。这个初等证明不需要指数型尾界。不过，如果目标错误率 δ 很小，上述加性误差未必小到可以忽略，尤其是在 k 也随规模增长时；后续的集中分析将提供更精细的控制。对于 $m = 1$ 的特殊情形，假阳性概率直接等于 1。

4.6 用有限独立性给出显式实现

另一种布局是让每个哈希函数各自使用一行位数组，称为*分区 Bloom 过滤器*（partitioned Bloom filter）。改为分行存储后，仅用通用哈希族就能得到一个非常直接的证明。

定理 4.1 (分区过滤器)。 设 $n \ge 1$、$0 < \delta \le 1/2$。独立选择 $k = \lceil \log_2(1/\delta) \rceil$ 个映射到 $b = 2n$ 个槽位的通用哈希函数，每个函数对应一个 b 位数组。每次插入都更新所有行；查询时恰在每一行均返回阳性时回答“属于”。该结构没有假阴性，对任意固定非成员的假阳性概率至多为 δ。包括随机种子在内的总空间为

$$O\big(n \log(1/\delta) + \log N \log(1/\delta)\big) \quad \text{位}.$$

证明. 对固定的非成员，第 j 行返回阳性，意味着它在这一行与某个已存储键碰撞。并合界给出其概率至多为 $n/b = 1/2$。各行采用独立的种子，因此所有行同时返回阳性的概率至多为 2^{-k}。位数组共占 $bk = 2nk$ 位；第 2节的通用哈希构造每行只需 $O(\log N)$ 位种子。 □

插入和查询都需要 $O(k)$ 次哈希求值及位访问。空集合可以仅用一个标记表示。上述保证要求事先选定的容量足以覆盖被插入的不同键的个数。还要注意，删除键时不能简单地将相应位

清零，因为其他键可能共用这些位；普通 Bloom 过滤器不能用“逆向执行插入”的方式支持删除。

这一构造清楚地区分了两层随机性：同一行内只需碰撞概率界，不同行的独立种子则负责将错误概率指数级降低。

合并过滤器。 若两个过滤器采用相同的布局、参数和哈希种子，将对应位逐位取或，就得到集合并集的过滤器；共享数组和分区布局都满足这一性质。原因是插入一个键只会将相应位设为 1，重复插入没有额外影响。为了保持指定的假阳性上界，容量参数须按并集的规模配置。

5 频率矩与不同元素计数

5.1 数据流模型与频率矩

前两节把一个集合组织起来，以便回答成员查询。现在转向统计问题：键逐个到达，我们希望只保留少量信息，便能概括整个输入。

设数据流为 $x_1, \dots, x_L \in \mathcal{U}$，其中 L 是数据流长度。对每个键 $x \in \mathcal{U}$，定义其*频率*（frequency）

$$f_x = |\{i : x_i = x\}|.$$

已经出现过的键组成集合 $D = \{x \in \mathcal{U} : f_x > 0\}$，不同元素个数记为 $z = |D|$。因此 $\sum_{x \in \mathcal{U}} f_x = L$；$L$ 计算每一次出现，z 则对每个不同的键只计一次。

对 $p > 0$，定义第 p 阶*频率矩*（frequency moment）为

$$F_p = \sum_{x \in \mathcal{U}} f_x^p.$$

另外单独定义 $F_0 = z$，从而避免使用 0^0。只有插入时，$F_1 = L$ 可以直接精确计数，而 F_2 反映重复程度：所有元素互异时 $F_2 = L$，所有元素相同时 $F_2 = L^2$。例如，数据流 (a, b, a, c, b, a) 满足

$$L = 6, \quad D = \{a, b, c\}, \quad z = 3, \qquad f_a = 3,\ f_b = 2,\ f_c = 1, \quad F_2 = 3^2 + 2^2 + 1^2 = 14.$$

*单遍数据流算法*按顺序读取每个元素一次，之后不能重新访问已经读过的输入，只能保留自己的内部状态。用于查询或估计的紧凑状态称为*数据摘要*（sketch）。空间复杂度计算保留状态所需的位数，包括描述随机哈希函数的种子，而不包括输入本身。本节的数据流预先固定，与算法抽取的随机种子无关。同一个键无论出现多少次，都必须使用相同的哈希值，不能把每次出现看成一次新的独立抽样。

下面集中研究 F_0，即不同元素计数；本节末尾再介绍其他频率矩的相关结果。

5.2 不同元素计数的误差与空间目标

我们希望得到一个 (ε, δ)-估计：

$$\mathbb{P}\big((1-\varepsilon)z \le \widehat{z} \le (1+\varepsilon)z\big) \ge 1 - \delta.$$

这里 $0 < \varepsilon < 1$ 控制*相对误差*，$0 < \delta \le 1/2$ 控制*失败概率*。两者描述不同的要求：减小 ε 是要求估计得更准，减小 δ 是要求达到这一精度的概率更高。空流的答案为零，必要时额外用一位记录数据流是否为空。

用精确字典存储所有出现过的键，需要 $O(z \log N)$ 位；也可以使用一个 N 位的位图。如果允许近似，随机摘要可以显著降低空间用量。然而，即使只要求一个固定的、较小的相对误差，确定性数据流算法也无法取得同样的空间改进。

命题 5.1 (确定性算法的空间下界)。 任何确定性单遍数据流算法，若对所有输入都能将不同元素个数估计到相对误差 $1/10$ 以内，则其最坏情形空间为 $\Omega(N)$ 位。

证明. 先假设 N 是 16 的倍数，并令 $s = N/8$。我们先构造一个集合族 $\mathcal{F}$：它包含 $\lfloor 2^{N/64} \rfloor$ 个大小均为 s 的集合，且任意两个集合的交集大小都小于 $s/2$。

固定一个大小为 s 的集合 S，再从全集的所有 s 元子集中均匀选取 T。对 S 中任何指定的 $s/2$ 个元素，它们全部属于 T 的概率至多为 $(s/N)^{s/2}$。对这些元素的选法使用并集界，得到

$$\mathbb{P}(|S \cap T| \geq s/2) \leq \binom{s}{s/2} (s/N)^{s/2} \leq 2^s 8^{-s/2} = 2^{-N/16}.$$

独立选取 $\lfloor 2^{N/64} \rfloor$ 个 s 元子集，再对所有集合对使用并集界，存在不满足交集要求的集合对的概率小于一。因此，这样的 $\mathcal{F}$ 确实存在。

若算法使用的状态少于 $\log_2 |\mathcal{F}|$ 位，则存在不同的 $S, T \in \mathcal{F}$，使得分别按固定顺序读入 S 和 T 的全部元素后，算法处于同一状态。接下来，给两个数据流都追加 S 的全部元素。由于算法是确定性的，最后输出必然相同；但两个数据流的不同元素个数分别为 s 和 $|S \cup T| > 3s/2$，而允许的估计区间互不相交，因为 $1.1s < 0.9(3s/2)$，矛盾。对一般的 N，将输入限制在大小为 $16\lfloor N/16 \rfloor$ 的子全集中即可。 □

这是 Alon、Matias 和 Szegedy 的工作 [9]中确定性空间下界的一种简单形式。虽然输出只是一个数，随机化仍然能够从根本上改变所需的存储空间。

5.3 最小值摘要：从理想哈希模型理解估计思想

先考虑一个理想模型：哈希函数 $h : \mathcal{U} \to (0, 1]$ 为不同的键赋予相互独立的连续均匀随机值。排除零这一端点不改变均匀分布，因为单点的概率为零，同时保证后面取倒数总有定义。对非空数据流，维护

$$Y = \min_{1 \leq i \leq L} h(x_i) = \min_{x \in D} h(x).$$

这个最小值涉及 z 个独立随机数。因此，对 $0 \leq t \leq 1$，

$$\mathbb{P}(Y > t) = (1 - t)^z.$$

利用附录 B 中由尾概率计算矩的方法，可得

$$\begin{aligned} \mu := \mathbb{E}Y &= \int_0^1 (1-t)^z \, dt = \frac{1}{z+1}, \\ \mathbb{E}Y^2 &= \int_0^1 2t(1-t)^z \, dt = \frac{2}{(z+1)(z+2)}, \\ \operatorname{Var}(Y) &= \frac{z}{(z+1)^2(z+2)} \leq \mu^2. \end{aligned} \tag{10}$$

由此可见，最小值通常处于 $1/z$ 的量级，自然会想到取倒数来估计 z。不过，$\mathbb{E}(1/Y) = \infty$：Y 的密度 $z(1-t)^{z-1}$ 在零点附近有正的常数下界，而 $\int_0^\eta dt/t$ 发散。因此，不能对期望公式直接取倒数，就声称得到了无偏估计量。

可行的做法是先计算若干个独立最小值的*平均数，再取倒数*。

最小值摘要与平均值技巧（理想哈希模型）

独立选取 $k = \lceil 9/(\varepsilon^2\delta)\rceil$ 个理想哈希函数。对每个 $j \in [k]$，维护 $Y_j = \min_i h_j(x_i)$。数据流非空时，返回

$$\widehat{z} = \frac{1}{\overline{Y}} - 1, \qquad \overline{Y} = \frac{1}{k}\sum_{j=1}^{k} Y_j.$$

定理 5.2。 上述理想最小值摘要给出一个 (ε,δ)-估计。

证明. 不同哈希函数相互独立，故 $\mathbb{E}\overline{Y} = \mu$，且 $\mathrm{Var}(\overline{Y}) \le \mu^2/k$。由 Chebyshev 不等式（见附录 B），

$$\mathbb{P}\big(|\overline{Y} - \mu| > (\varepsilon/3)\mu\big) \le \frac{9}{k\varepsilon^2} \le \delta.$$

在其余情形下，

$$|\widehat{z} - z| = \left|\frac{1}{\overline{Y}} - \frac{1}{\mu}\right| \le \frac{(\varepsilon/3)(z+1)}{1-\varepsilon/3} \le \varepsilon z.$$

最后一步用到了 $z+1 \le 2z$ 和 $1-\varepsilon/3 \ge 2/3$。 □

这里将 $\overline{Y}$ 的误差转化为 $\widehat{z}$ 的误差时，必须保留 $z+1$ 这一因子，特别是 z 很小时更不能忽略。证明针对的是 $1/\overline{Y} - 1$，并不适用于对各个 $1/Y_j - 1$ 取平均的做法。

还可以先取 $k = \lceil 36/\varepsilon^2\rceil$，将单次失败概率降到 $1/4$，再独立运行 $O(\log(1/\delta))$ 次，返回各次估计的中位数。由引理 B.3，这样只需 $O(\varepsilon^{-2}\log(1/\delta))$ 个实数寄存器。不过，这仍是*理想模型*下的空间核算：任意精度实数和完全独立的随机函数都不能直接用有限位实现。

合并最小值摘要。 合并摘要时忽略空的数据片段；若所有片段都为空，则返回零。对使用相同哈希函数的各段数据流，将对应寄存器逐坐标取最小值，就得到整个数据流的最小值摘要。合并的是原始寄存器，随后才计算平均值和倒数；不能直接合并各段给出的估计值。

下面两种算法会给出实际的位空间保证。

5.4 只用两两独立性的尾零摘要

如果只记录迄今出现过的最罕见哈希模式，就可能用一个很小的寄存器估计不同元素个数。下面介绍一种简化的 Flajolet–Martin 型估计方法，它是 Alon、Matias 和 Szegedy 所分析的两两独立版本 [8, 9]，并非 Flajolet–Martin 原论文中完整的位图估计器。

令 $w = \lceil \log_2 N\rceil$，使用有限域上的仿射构造，选取一个输出为 w 位串的两两独立且均匀的哈希函数。对非零位串 y，用 $\mathrm{tz}(y)$ 表示其末尾连续的零的个数，简称*尾零数*；并约定

$$\mathrm{tz}(0) = w.$$

这样所有尾零数都有有限取值。对每个整数 $0 \le r \le w$，恰有 2^{w-r} 个位串的尾零数至少为 r。

记录最大的尾零数

对非空数据流，维护

$$R = \max_i \mathrm{tz}(h(x_i)), \qquad \text{返回 } \widehat{z} = 2^R.$$

一个键的所有出现都使用同一个哈希函数。空流返回零。

定理 5.3 (常数因子不同元素计数)。 对任意 $C \geq 1$ 以及预先固定的非空数据流，

$$\mathbb{P}\big(\widehat{z} < z/C \text{ 或 } \widehat{z} > Cz\big) \leq 3/C.$$

包括随机种子在内，整个摘要占用 $O(\log N)$ 位。

证明. 对 $0 \leq r \leq w$，定义达到阈值 r 的不同元素个数

$$T_r = \sum_{x \in D} \mathbf{1}\{\mathrm{tz}(h(x)) \geq r\}.$$

由均匀性，$\mathbb{E}T_r = z2^{-r}$。由两两独立性，

$$\mathrm{Var}(T_r) = z2^{-r}(1 - 2^{-r}) \leq z2^{-r}, \qquad R \geq r \iff T_r \geq 1. \tag{11}$$

先分析高估的概率。若 $Cz \geq 2^w$，则 $\widehat{z} > Cz$ 不可能发生；否则取 $r = \lceil \log_2(Cz) \rceil \leq w$。由 Markov 不等式，

$$\mathbb{P}(\widehat{z} > Cz) \leq \mathbb{P}(T_r \geq 1) \leq z2^{-r} \leq 1/C.$$

再分析低估的概率。若 $z/C \leq 1$，则 $\widehat{z} < z/C$ 不可能发生；否则 $r = \lceil \log_2(z/C) \rceil \in \{1, \ldots, w\}$。由 Chebyshev 不等式，

$$\mathbb{P}(\widehat{z} < z/C) \leq \mathbb{P}(T_r = 0) \leq \frac{\mathrm{Var}(T_r)}{(\mathbb{E}T_r)^2} \leq \frac{2^r}{z} \leq \frac{2}{C}.$$

合并两个坏事件，便得到所需概率界。

寄存器 R 的取值属于 $\{0, \ldots, w\}$，本身只需 $\lceil \log_2(w+1) \rceil$ 位。但仿射哈希函数的种子还需要 $2w$ 位；若存储域的描述，也需要另加 $O(w)$ 位。因此，完整摘要的空间为 $O(\log N)$ 位。 □

取 $C = 12$，单次失败概率至多为 $1/4$。对独立副本的输出取中位数，便可将成功概率提高到 $1 - \delta$。此时总空间为 $O(\log N \log(1/\delta))$ 位。但近似因子仍是 12：提高置信度不会自动改善近似精度。

合并尾零摘要。 用空流标记排除空的数据片段后，各非空片段使用相同的哈希种子时，将尾零寄存器取最大值，就得到完整数据流的寄存器。若维护多个副本，则对相应副本逐坐标取最大值，再计算输出及其中位数。

要得到任意指定的相对误差，还需要保留更多信息。

5.5 Bottom-k：用顺序统计量控制相对误差

与只保存一个最小值相比，保存最小的若干个哈希值可以更准确地判断不同元素个数。这就是 bottom-k 方法，也称 k-minimum-values 方法。下面分析 BJKST 方法的有限值域版本，相关工作来自 Bar-Yossef、Jayram、Kumar、Sivakumar 和 Trevisan [10]。

一次试验使用以下参数：

$$k = \left\lceil \frac{64}{\varepsilon^2} \right\rceil, \qquad M = \text{不小于 } 4N^2 \text{ 的最小的 2 的幂}. \tag{12}$$

选取两两独立且均匀的哈希函数 $h : \mathcal{U} \to \{1, \ldots, M\}$。具体地，可先在 $\mathbb{F}_M$ 上使用仿射哈希，再将输出位串解释为 0 至 $M-1$ 的整数，最后加一。这样所得哈希值均为正数，后面取倒数时不会遇到零分母；随机种子只需 $O(\log N)$ 位。

有限值域上的 bottom-k 摘要

维护一个至多含 k 个*互异哈希值*的集合 K，初始为空。

1. 读入键 x 时，若 $h(x) \notin K$，则将其插入 K。
2. 若插入后 $|K| = k+1$，则删除 K 中的最大值。
3. 数据流结束时，若 $|K| < k$，则返回 $|K|$；否则令 $T = \max K$，返回 kM/T。

为使失败概率至多为 δ，独立运行奇数次 $t \geq 8\ln(1/\delta)$ 试验，返回各次输出的中位数。

在任意时刻，K 恰好包含已经出现过的最小的 k 个互异哈希值；若互异哈希值不足 k 个，则全部保留。被删除的值此后不会重新成为需要保留的值：一旦 K 存满，所保留的最大值就只能减小。因此，同一个键重复到达不会造成重复计数。

不过，不同的键仍可能得到相同的哈希值。令 $\mathcal{E}$ 表示全部 z 个不同键的哈希值互异。由碰撞概率界，

$$\mathbb{P}(\mathcal{E}^c) \leq \frac{\binom{z}{2}}{M} \leq \frac{1}{8}. \tag{13}$$

在 $\mathcal{E}$ 上，若 $z < k$，算法会精确返回 z。当 $z \geq k$ 时，需要下面的顺序统计量引理。应用引理时始终在原来的概率空间中计算，不对 $\mathcal{E}$ 作条件化。

引理 5.4 (有限值域上的顺序统计量)。 设 $H_1, \dots, H_z$ 是在 $\{1, \dots, M\}$ 上均匀分布的两两独立随机变量，其中 $M \geq z \geq k$ 且 $k \geq 2/\varepsilon$。令 Y_k 为它们的第 k 小值，排序时*相等的数值按出现次数重复计入*。则

$$\mathbb{P}\left(\left| \frac{kM}{Y_k} - z \right| > \varepsilon z \right) \leq \frac{6}{k\varepsilon^2}.$$

证明. 先考虑高估。此时 Y_k 偏小。令 $a = kM/((1+\varepsilon)z)$，并定义

$$V = \sum_{i=1}^{z} \mathbf{1}\{H_i < a\}.$$

若 $kM/Y_k > (1+\varepsilon)z$，则必有 $V \geq k$。其期望满足 $\mu_V \leq za/M = k/(1+\varepsilon)$，方差不超过期望。因此，

$$\mathbb{P}(V \geq k) \leq \frac{\mu_V}{(k - \mu_V)^2} \leq \frac{1+\varepsilon}{k\varepsilon^2} \leq \frac{2}{k\varepsilon^2}.$$

再考虑低估。令 $b = kM/((1-\varepsilon)z)$。若 $b \geq M$，则 $Y_k > b$ 不可能发生；否则定义 $W = \sum_i \mathbf{1}\{H_i \leq b\}$，于是 $Y_k > b$ 蕴含 $W < k$。这里必须计入离散值域带来的取整误差：

$$\frac{k}{1-\varepsilon} - 1 \leq \frac{k}{1-\varepsilon} - \frac{z}{M} \leq \mu_W = \frac{z\lfloor b \rfloor}{M} \leq \frac{k}{1-\varepsilon}.$$

由 $k \geq 2/\varepsilon$，

$$\mu_W - k \geq \frac{k\varepsilon}{1-\varepsilon} - 1 \geq \frac{k\varepsilon}{2(1-\varepsilon)}.$$

再次利用 $\operatorname{Var}(W) \leq \mu_W$ 及 Chebyshev 不等式，得到

$$\mathbb{P}(W < k) \leq \frac{\mu_W}{(\mu_W - k)^2} \leq \frac{4(1-\varepsilon)}{k\varepsilon^2} \leq \frac{4}{k\varepsilon^2}.$$

将高估与低估的概率相加即得结论。 □

定理 5.5 (有限位实现的 bottom-k 摘要)。 采用式 (12) 中的参数，一次试验不能达到相对误差 ε 的概率至多为 $1/4$。进一步对独立试验的结果取中位数，可得到一个 (ε,δ)-估计，空间为

$$O\big(\varepsilon^{-2}\log N\log(1/\delta)\big)\quad \text{位},$$

其中已经包含随机种子的空间。

证明. 当 $z<k$ 时，只有 $\mathcal{E}^c$ 发生才可能出错。当 $z\ge k$ 时，在 $\mathcal{E}$ 上，实际维护的阈值 T 就是引理 5.4 中的顺序统计量。因此，算法失败必然属于以下两个事件的并集：出现哈希碰撞，或顺序统计量偏离所需范围。其概率至多为

$$\frac{1}{8}+\frac{6}{k\varepsilon^2}\le\frac{1}{8}+\frac{6}{64}<\frac{1}{4}.$$

这个并集界不要求两个事件独立。尤其不能声称，在给定“没有碰撞”后，哈希值仍然保持两两独立。

由引理 B.3，独立运行 t 次后取中位数，失败概率至多为 $e^{-t/8}\le\delta$。每次试验保存至多 k 个各占 $\log_2 M=O(\log N)$ 位的值，外加 $O(\log N)$ 位的种子及辅助状态。若 $k>N$，也可以直接用精确字典存储所有不同键，至多占用 $N<k$ 个存放键的机器字，同样满足上述空间界。 □

用平衡搜索树维护 K 时，每个试验中一次更新需要 $O(\log k)$ 次比较，并且可以同时排除重复的哈希值。因此，在前述哈希求值模型下，直接实现的每次更新需要 $O(t\log k)$ 个机器字操作。若只要求常数成功概率，一次试验即可得到$O(\varepsilon^{-2}\log N)$ 位空间界。

扩大有限哈希值域与重复独立试验解决的是两个不同问题：前者控制不同键之间的碰撞，后者将总失败概率降到任意指定的 δ。若固定使用例如 N^3 大小的值域，却完全忽略碰撞概率，就不能据此声称支持任意小的失败概率。

合并 bottom-k 摘要。 对使用相同参数和哈希种子的摘要，将各自保留的集合求并，再留下最小的 k 个互异值，即可得到完整数据流的摘要。若某个值没有被一个部分保留，那么该部分中已经存在 k 个更小的互异值，因此它也不可能属于全局最小的 k 个值。多个独立试验分别按此规则合并，之后再取估计值的中位数。

5.6 频率矩的进一步结果

Alon、Matias 和 Szegedy 的工作 [9]确立了用小空间随机算法估计 F_0 和 F_2 的研究方向。在精度和成功概率均为常数时，F_0,F_1,F_2 都可以用关于 $\log N+\log(L+1)$ 的多项式空间处理。对固定的 $p>2$，当数据流长度为 N 的多项式时，经典的空间量级为 $\widetilde{\Theta}(N^{1-2/p})$，其中波浪号隐藏对数因子；Indyk 和 Woodruff [11] 给出的上界达到了这一指数。这些结果将不同元素计数放在了频率矩估计的整体背景中；前面算法的分析不依赖于 F_2 或更高阶频率矩的估计器。

对于不同元素计数本身，Kane、Nelson 和 Woodruff [12] 在常数成功概率下实现了 $O(\varepsilon^{-2}+\log N)$ 位空间，改进了这里初等 bottom-k 构造中的乘积项 $\varepsilon^{-2}\log N$。理解这一界的最优性时，应限于通常讨论的非平凡精度范围；无论如何，精确的 N 位位图始终可用。这些更精细的算法可作为进一步阅读。

6 频率估计：Count-Min 摘要

沿用第 5节的数据流模型和频率向量 $(f_x)_{x\in\mathcal{U}}$。不同元素计数只关心一个键是否出现过；频率查询则给定一个键 x，要求估计它出现的次数 f_x。本节先处理只有插入的情形，数据流及查询键都预先固定，与哈希种子无关。取 $0<\varepsilon<1$、$0<\delta\le 1/2$，追求以下*加性误差*保证：

$$\mathbb{P}\big(|\widehat{f}_x - f_x| > \varepsilon L\big) \le \delta.$$

这样的误差足以帮助识别高频元素，但弱于对每个 f_x 都给出相对误差保证。

一个自然的尝试是把键哈希到若干桶，并为每个桶记录所有落入该桶的元素出现次数之和。查询时，键所在桶的计数包含真实频率，也混入了其他键碰撞产生的贡献。用多行独立的哈希重复这一做法，便有机会找到碰撞噪声较小的一行。Cormode 和 Muthukrishnan 提出的 Count-Min 摘要 [13] 正是采用这一思路，其参数为

$$b = \lceil e/\varepsilon \rceil, \qquad d = \lceil \ln(1/\delta) \rceil,$$

并独立选取 d 个通用哈希函数 $h_j : \mathcal{U} \to [b]$。不同的行使用独立种子；对每一行，只需其哈希函数满足碰撞意义下的通用性。

插入数据流的 Count-Min 摘要

建立一个 d 行、b 列的计数器数组 A，初值均为零。

- **读入元素 y：** 对每一行 j，将 $A[j, h_j(y)]$ 加一。
- **查询键 x：** 返回 $\widehat{f}_x = \min_{j\in[d]} A[j, h_j(x)]$。

查询键所在桶的计数包含它的全部出现次数，还可能包含其他键碰撞到该桶的贡献。因此，每一行给出的都是上界，取最小值后仍不会低估。

定理 6.1 (Count-Min 的单点查询保证)。 对与随机种子无关、预先固定的插入数据流和查询键 x，有

$$\mathbb{P}\big(f_x \le \widehat{f}_x \le f_x + \varepsilon L\big) \ge 1 - \delta.$$

对非空数据流，摘要的总空间为

$$O\left(\varepsilon^{-1}\log(1/\delta)\log(L+1) + \log(1/\delta)\log N\right) \quad \text{位}, \tag{14}$$

其中已经包含随机种子。每次更新和查询均需 $O(\log(1/\delta))$ 次哈希求值及计数器访问。

证明. 若 $L = 0$，所有计数器均为零，结论显然成立。以下假定 $L > 0$。固定一行 j，将该行的估计写成真实频率与碰撞噪声之和：

$$A[j, h_j(x)] = f_x + Z_j, \qquad Z_j = \sum_{y\neq x} f_y\, \mathbf{1}\{h_j(y) = h_j(x)\} \ge 0.$$

由通用性和期望的线性性，

$$\mathbb{E}Z_j \le \frac{1}{b}\sum_{y\neq x} f_y = \frac{L - f_x}{b} \le \frac{L}{b}.$$

这一步不要求各个碰撞指示变量独立。再由 Markov 不等式，

$$\mathbb{P}(Z_j > \varepsilon L) \le \frac{1}{\varepsilon b} \le e^{-1}.$$

由于整个数据流预先固定，Z_j 只依赖于第 j 行的种子，因此不同行的 Z_j 相互独立。最终结果偏大，意味着每一行的碰撞噪声都偏大，故

$$\mathbb{P}(\widehat{f}_x > f_x + \varepsilon L) = \mathbb{P}\left(\bigcap_{j=1}^{d} \{Z_j > \varepsilon L\}\right) \le e^{-d} \le \delta.$$

每个计数器的取值为零至 L，需要 $\lceil \log_2(L+1) \rceil$ 位。数组共有 bd 个计数器，另有 d 个各占 $O(\log N)$ 位的种子，由此得到式 (14)。一次操作在每行只访问一个计数器。 □

位空间分析假设计数器能够容纳已处理数据流的长度，也可以按预先给定的长度上界配置；若最终长度未知，可通过通常的扩容方法处理。操作次数的分析则假设机器字足以容纳一个键、一个计数器的值以及一个数组地址。计数器的*个数*与数据流长度无关，但存储这些计数器的*位数*依赖于数据流长度。

6.1 从单点查询到同时保证与高频元素

上述定理针对一个预先固定的查询键。如果事先给定 Q 个候选键，可以将构造中的失败概率设为 δ/Q，再使用并集界；于是，以至少 $1-\delta$ 的概率，这 Q 个键的估计全部满足误差要求。特别地，取深度 $d = \lceil \ln(N/\delta) \rceil$，即可在数据流结束时对全集中的所有键同时作出保证。这样也允许在观察最终摘要后再选择查询键，前提是数据流本身仍然预先固定、与种子无关。

给定阈值 ϕ，满足 $f_x > \phi L$ 的元素称为*高频元素*（heavy hitter）。在所有候选键均估计成功的事件上，若报告估计值超过 ϕL 的候选键，就会包含每个高频元素，并排除所有满足 $f_x \le (\phi - \varepsilon)L$ 的候选键；处于两者之间的元素则可能被报告。但单点查询摘要本身不会列出候选键：还需要另外提供候选集合、枚举整个全集，或使用额外的恢复算法。当全集非常大时，必须区分“查询某个键的频率”和“找出所有高频键”这两项任务。

6.2 合并摘要与更新模型

如果各段数据流使用相同的数组尺寸和哈希种子，将 Count-Min 数组逐项相加，所得计数器就恰好对应各段数据流串接后的结果。这是因为每个桶累计的是频率之和，而完整数据流的频率向量也是各段频率向量之和。因此，无需重放各段输入就能合并摘要；合并后的误差界以总数据流长度为准，计数器也须能容纳合并后的总计数。

这里要区分两种种子使用方式：为了提高置信度，*不同试验*或*不同行*使用独立种子；为了合并同一试验在*不同数据片段*上的状态，对应位置必须使用相同种子。前面各种摘要的合并规则也遵循这一原则。

Count-Min 的证明实际用到的是最终频率非负。因此，它也适用于非负的带权更新：将 L 换成总权重，并为计数器选择合适的表示即可。若允许删除，但任何时刻各键的频率均非负，即*严格转门模型*（strict turnstile model），则对每个预先固定的合法状态，同样的代数分析仍然成立。如果允许任意正负频率，碰撞噪声就可能为负，取最小值后也不再保证给出上界。

7 习题

以下习题用于巩固本讲算法及其依赖的数学假设。

1. **生日概率的近似。** 利用式 (2) 证明：若 $n/\sqrt{m} \to c > 0$，则至少发生一次碰撞的概率趋于 $1 - e^{-c^2/2}$。说明为什么仅有 $n = o(m)$，还不能保证 $e^{-n(n-1)/(2m)}$ 是不碰撞概率的相对近似。

2. **两两独立还不够。** 在 $\mathbb{F}_p$ 上令 $h(x) = ax + b$，其中 a, b 独立均匀选取。固定任意 $2 \le n \le p$ 个互异键，证明它们发生碰撞当且仅当 $a = 0$，故碰撞概率为 $1/p$。将这一结论与独立生日模型比较。

3. **种子如何选取很重要。** 设 a, b 在域上均匀选取。解释为什么排除 $a = 0$ 会破坏两两独立性，以及为什么再对任意整数 m 取模可能破坏输出的边缘均匀性。

4. **FKS 的常数。** 将第一层的接受阈值由 $4n$ 改为 αn，其中 $\alpha > 2$，求第一层期望试验次数的上界。分别说明哪些保证描述随机构造过程，哪些保证描述构造完成后的查询。

5. **Bloom 中的相关性。** 对 $n = 1, k = 2, m = 2$，枚举被置一的位数的所有可能取值，推导出精确的假阳性概率 $5/8$。另一个数值 $9/16$ 是怎样算出来的？

6. **Bloom 的参数选择。** 对 $\ln[(1 - e^{-k/c})^k]$ 求导，找出最优点。解释为什么实现时还必须将最优实数 k 转换为整数。

7. **取倒数与取期望。** 直接证明理想最小值满足 $\mathbb{E}(1/Y) = \infty$。进一步证明：若使用式 (10) 中的精确方差，并直接计算取倒数后对应的准确阈值，而不经过较简单的 $\varepsilon/3$ 偏差事件，则平均值摘要可以只取 $k = \lceil 4/(\varepsilon^2\delta) \rceil$。

8. **寄存器与种子的空间。** 解释为什么尾零寄存器本身只占 $O(\log\log N)$ 位，而本讲分析的完整摘要占 $O(\log N)$ 位。为什么将 $\mathrm{tz}(0)$ 定义为无穷大会使上述输出为有限数值的估计器及其误差保证失效？

9. **Bottom-k 中的相等哈希值。** 将摘要改为存储二元记录 $(h(x), x)$，按字典序排序，同一个键只保留一条记录。证明即使不同键的哈希值相等，也可以应用引理 5.4。由此说明，值域大小只需选取满足 $M \ge N$ 的 2 的幂，无需再单独分析碰撞事件。

10. **提高置信度。** 为什么取中位数能够保留相对误差区间，而取最小值通常不行？为什么 Count-Min 恰好适合取最小值？

11. **多个查询。** 推导对 Q 个固定候选键作出同时保证所需的 Count-Min 深度。解释为什么只证明一个固定查询的误差界，不足以保证正确报告所有高频元素。

12. **合并摘要。** 证明 bottom-k 的并集合并规则。举例说明，若两个摘要使用无关的哈希种子，一般不能直接按上述逐坐标规则合并。

A 独立性与有限独立性

随机变量 $X_1, \dots, X_s$ 相互独立，是指对任意集合 $A_1, \dots, A_s$ 都有

$$\mathbb{P}(X_1 \in A_1, \dots, X_s \in A_s) = \prod_{i=1}^{s} \mathbb{P}(X_i \in A_i).$$

对于有限随机变量，只需对单点集合 $A_i = \{a_i\}$ 验证此式；对于连续随机变量，单点概率通常都是零，不能据此判断独立性。如果任取不超过 k 个变量，它们都相互独立，就称这些变量具

有 *k 阶独立性*（k-wise independence）。独立性本身不要求各变量同分布，也不要求其分布均匀。

分别对各变量作确定性变换，独立性仍然成立。原因是事件 $g_i(X_i) \in B_i$ 等价于 $X_i \in g_i^{-1}(B_i)$，可以直接代入定义。同理，k 阶独立性也得到保留。如果 X_i 在有限集合 A 上均匀分布，而映射 $g_i : A \to B$ 的每个原像集合大小相等，那么 $g_i(X_i)$ 还会在 B 上均匀分布。正文使用的二进制坐标投影正是这种情形。

少量随机位可以生成许多两两独立的位。 取均匀随机向量 $R \in \mathbb{F}_2^r$。对每个非零向量 $a \in \mathbb{F}_2^r$，定义

$$X_a = a \cdot R = \sum_{i=1}^{r} a_i R_i \pmod 2.$$

先取一个满足 $a_i = 1$ 的坐标。固定其余随机位后，翻转 R_i 恰好翻转 X_a，所以 X_a 是均匀随机位。再取两个不同的非零向量 a, b。它们在 $\mathbb{F}_2$ 上线性无关，因为这个域中唯一的非零标量是 1。于是线性映射 $R \mapsto (a \cdot R, b \cdot R)$ 的秩为二，每个输出对恰有 2^{r-2} 个原像，出现概率都是 $1/4$。

这样得到的 $2^r - 1$ 个随机位两两独立。但在 $r \geq 2$ 时，它们并不相互独立，因为

$$X_a + X_b = X_{a+b} \pmod 2 \qquad (a \neq b).$$

最小的例子是 $R_1, R_2, R_1 \oplus R_2$：任意两个独立，知道其中两个却能确定第三个。节省随机种子的代价正是放弃完全独立性；输出的总熵没有超过种子的熵。

区分两种独立性要求。 本讲常常只要求同一哈希函数在不同输入上的输出两两独立。重复运行一个随机过程时，则要另外说明不同次运行如何选择种子。输出具有两两独立性，并不意味着所有依赖整个种子的事件都相互独立；只有两两独立的重复，也不能直接得到 p^t 形式的全部失败概率。

B 期望、方差与偏差不等式

B.1 示性变量、条件化与一阶矩

事件 A 的示性变量 $\mathbf{1}\{A\}$ 在 A 发生时取 1，否则取 0，因而 $\mathbb{E}\mathbf{1}\{A\} = \mathbb{P}(A)$。期望的线性性

$$\mathbb{E}\left[\sum_i c_i X_i\right] = \sum_i c_i \mathbb{E} X_i$$

不需要任何独立性。由逐点不等式 $\mathbf{1}\{\bigcup_i A_i\} \leq \sum_i \mathbf{1}\{A_i\}$，还得到*联合界*（union bound）

$$\mathbb{P}\left(\bigcup_i A_i\right) \leq \sum_i \mathbb{P}(A_i),$$

它同样不需要独立性。下文作期望的代数运算时，均假设相关期望有限。

对有限随机变量 Y 分类讨论，就得到全概率公式和全期望公式：

$$\mathbb{P}(A) = \sum_y \mathbb{P}(Y = y)\mathbb{P}(A \mid Y = y), \qquad \mathbb{E}X = \sum_y \mathbb{P}(Y = y)\mathbb{E}[X \mid Y = y].$$

概率为零的项可以略去。反复使用条件概率的定义，则有链式法则

$$\mathbb{P}\left(\bigcap_{i=1}^{t} A_i\right) = \mathbb{P}(A_1)\prod_{i=2}^{t}\mathbb{P}\left(A_i \mid \bigcap_{j<i} A_j\right).$$

这里要求各条件事件概率为正；若某个前缀事件概率已为零，整个交事件的概率也为零。

引理 B.1 (Markov 不等式)。 若 $X \geq 0$ 且 $a > 0$，则 $\mathbb{P}(X \geq a) \leq \mathbb{E}X/a$。

证明. 对逐点不等式 $a\mathbf{1}\{X \geq a\} \leq X$ 取期望。 □

B.2 方差与取平均

对二阶矩有限的随机变量，定义

$$\mathrm{Var}(X) = \mathbb{E}[(X - \mathbb{E}X)^2] = \mathbb{E}[X^2] - (\mathbb{E}X)^2, \qquad \mathrm{Cov}(X,Y) = \mathbb{E}[XY] - \mathbb{E}X\mathbb{E}Y.$$

将平方展开，得到

$$\mathrm{Var}\left(\sum_i X_i\right) = \sum_i \mathrm{Var}(X_i) + 2\sum_{i<j}\mathrm{Cov}(X_i, X_j).$$

独立性蕴含 $\mathbb{E}[XY] = \mathbb{E}X\mathbb{E}Y$，所以独立变量的协方差为零。但协方差为零不蕴含独立：例如 X 在 $\{-1,0,1\}$ 上均匀分布时，X 与 X^2 的协方差为零，却显然有关联。对于两两独立的变量，求和时方差可以直接相加。

引理 B.2 (Chebyshev 不等式)。 对任意 $a > 0$，有 $\mathbb{P}(|X - \mathbb{E}X| \geq a) \leq \mathrm{Var}(X)/a^2$。

证明. 对非负变量 $(X - \mathbb{E}X)^2$ 应用 Markov 不等式。 □

若 B_i 是两两独立的示性变量，$p_i = \mathbb{E}B_i$，令 $S = \sum_i B_i$、$\mu = \mathbb{E}S$，则

$$\mathrm{Var}(S) = \sum_i p_i(1-p_i) \leq \mu, \qquad \mathbb{P}(S = 0) \leq \frac{1}{\mu} \quad (\mu > 0).$$

后一式来自 $S = 0 \Rightarrow |S - \mu| \geq \mu$ 和 Chebyshev 不等式。注意：只有两两独立时，S 未必服从二项分布。

若 $Y_1, \ldots, Y_r$ 两两独立，具有相同期望 μ，且各自方差至多为 v，那么平均值满足

$$\mathbb{E}\overline{Y} = \mu, \qquad \mathrm{Var}(\overline{Y}) \leq \frac{v}{r}, \qquad \mathbb{P}(|\overline{Y} - \mu| \geq \eta\mu) \leq \frac{v}{r\eta^2\mu^2} \quad (\eta, \mu > 0).$$

这就是*取平均降低方差*（mean trick）：副本数增加 r 倍，平均值的方差降低 r 倍。若最终输出是平均值的倒数等非线性函数，还需另外证明误差如何传递。

B.3 两个常用的期望不等式

由 $\mathrm{Var}(|X|) \geq 0$ 可得 $(\mathbb{E}|X|)^2 \leq \mathbb{E}[X^2]$，因而 $\mathbb{E}|X| \leq \sqrt{\mathbb{E}[X^2]}$。这是 Cauchy–Schwarz 不等式的一个特例：用它处理中心化变量，可以把方差界转化为绝对偏差的期望界。

若 $U \geq 0$，k 为正整数，则在相关期望有限时有 $\mathbb{E}[U^k] \geq (\mathbb{E}U)^k$。当 $k = 1$ 时两侧相等；当 $k \geq 2$ 时，函数 u^k 的导数递增，因此曲线位于每条切线上方：

$$u^k \geq a^k + ka^{k-1}(u - a) \qquad (u, a \geq 0).$$

令 $a = \mathbb{E}U$ 后取期望，即得结论。这就是正文分析 Bloom 过滤器时使用的 Jensen 不等式的特例。

B.4 由尾概率计算矩

对非负随机变量 W，有

$$\mathbb{E}W = \int_0^\infty \mathbb{P}(W > u)\,du, \qquad \mathbb{E}[W^2] = \int_0^\infty 2u\mathbb{P}(W > u)\,du.$$

证明只需从逐点恒等式出发：

$$W = \int_0^\infty \mathbf{1}\{W > u\}\,du, \qquad W^2 = \int_0^\infty 2u\mathbf{1}\{W > u\}\,du.$$

对非负函数交换期望与积分，就得到结论；期望为无穷时，这些恒等式仍成立。如果 $W \in [0,1]$ 且 $\mathbb{P}(W > u) = (1-u)^z$，代入即得

$$\mathbb{E}W = \frac{1}{z+1}, \qquad \mathbb{E}[W^2] = \frac{2}{(z+1)(z+2)}.$$

第二个积分可将 u 写成 $1-(1-u)$，化为两个幂函数的积分。

B.5 独立重复与中位数

若 t 个事件相互独立，每个事件的概率至多为 p，则它们全部发生的概率至多为 p^t。另一种常见用法是不断独立尝试，直到成功为止。设每次成功概率为 $q > 0$，T 表示首次成功所需的尝试次数，则对每个非负整数 r，有

$$\mathbb{P}(T > r) = (1-q)^r, \qquad \mathbb{E}T = \sum_{r\ge 0}\mathbb{P}(T > r) = \sum_{r\ge 0}(1-q)^r = \frac{1}{q}.$$

这里用了整数值非负变量的尾概率求和公式，其证明与上一节的积分公式相同。对于可能向两侧偏离的估计值，常用的是中位数的保证。

引理 B.3 (中位数放大成功概率)。 设 t 为奇数，$Z_1,\dots,Z_t$ 相互独立，且每个 Z_i 落在区间 I 内的概率至少为 $3/4$。则

$$\mathbb{P}\big(\operatorname{median}(Z_1,\dots,Z_t) \notin I\big) \le e^{-t/8}.$$

因此，取任意满足 $t \ge 8\ln(1/\delta)$ 的奇数 t，失败概率就至多为 δ。

证明. 令 $B_i = \mathbf{1}\{Z_i \notin I\}$。如果中位数不在 I 内，至少有 $(t+1)/2$ 个副本失败。由于 B_i 相互独立且 $\mathbb{E}B_i \le 1/4$，应用 Markov 不等式可得

$$\mathbb{P}\left(\sum_i B_i \ge t/2\right) \le 3^{-t/2}\mathbb{E}\left[3^{\sum_i B_i}\right] = 3^{-t/2}\prod_i(1+2\mathbb{E}B_i) \le \left(\frac{\sqrt{3}}{2}\right)^t.$$

再由 $1-x \le e^{-x}$，得到 $(\sqrt{3}/2)^t = (1-1/4)^{t/2} \le e^{-t/8}$。 □

B.6 一个对数不等式

当 $0 \le x \le \rho < 1$ 时，

$$-x - \frac{x^2}{2(1-\rho)} \le \ln(1-x) \le -x.$$

下面的积分同时证明两侧：

$$0 \le -\ln(1-x) - x = \int_0^x \frac{u}{1-u}\,du \le \frac{1}{1-\rho}\int_0^x u\,du = \frac{x^2}{2(1-\rho)}.$$

上界还给出 $1-x \le e^{-x}$。对乘积的各个因子取对数后逐项使用该式，就能为指数近似给出明确的误差界。

文献说明

通用哈希由 Carter 和 Wegman 提出 [1]，两层静态字典来自 Fredman、Komlós 和 Szemerédi 的工作 [2]。动态完美哈希可参阅 Dietzfelbinger 等人 [3]；简洁动态字典的时间与空间权衡见 [4, 5]。

允许假阳性的成员查询结构由 Bloom 提出 [6]。Bose 等人 [7]澄清了共享位数组分析中的独立性问题：将各查询位置被置为 1 的边缘概率相乘，并不能得到有限参数下精确的假阳性概率。

利用罕见哈希模式计数的思想来自 Flajolet 和 Martin [8]。本讲的最大尾零估计量是 Alon、Matias 和 Szegedy 在命题 2.3 中分析的两两独立版本 [9]，与原始位图估计量有所区别。bottom-k 方法见 Bar-Yossef 等人 [10]，Count-Min 摘要见 Cormode 和 Muthukrishnan [13]。若希望了解更精细的空间界，可继续阅读 Indyk 和 Woodruff 关于高阶频率矩的工作 [11]，以及 Kane、Nelson 和 Woodruff 关于不同元素计数的工作 [12]。

参考文献

[1] J. Lawrence Carter and Mark N. Wegman. Universal Classes of Hash Functions. *Journal of Computer and System Sciences*, 18(2):143–154, 1979. Primary publication record.

[2] Michael L. Fredman, János Komlós, and Endre Szemerédi. Storing a Sparse Table with $O(1)$ Worst Case Access Time. *Journal of the ACM*, 31(3):538–544, 1984. Original paper.

[3] Martin Dietzfelbinger, Anna R. Karlin, Kurt Mehlhorn, Friedhelm Meyer auf der Heide, Hans Rohnert, and Robert E. Tarjan. Dynamic Perfect Hashing: Upper and Lower Bounds. *SIAM Journal on Computing*, 23(4):738–761, 1994; conference version, *FOCS 1988*. doi:10.1137/S0097539791194094.

[4] Michael A. Bender, Martín Farach-Colton, John Kuszmaul, William Kuszmaul, and Mingmou Liu. On the Optimal Time/Space Tradeoff for Hash Tables. In *Proceedings of STOC*, pages 1284–1297, 2022. Author manuscript; doi:10.1145/3519935.3519969.

[5] Tianxiao Li, Jingxun Liang, Huacheng Yu, and Renfei Zhou. Tight Cell-Probe Lower Bounds for Dynamic Succinct Dictionaries. In *Proceedings of FOCS*, pages 1842–1862, 2023. Author manuscript; doi:10.1109/FOCS57990.2023.00112.

[6] Burton H. Bloom. Space/Time Trade-offs in Hash Coding with Allowable Errors. *Communications of the ACM*, 13(7):422–426, 1970. doi:10.1145/362686.362692.

[7] Prosenjit Bose, Hua Guo, Evangelos Kranakis, Anil Maheshwari, Pat Morin, Jason Morrison, Michiel Smid, and Yihui Tang. On the False-Positive Rate of Bloom Filters. *Information Processing Letters*, 108(4):210–213, 2008. Author manuscript; doi:10.1016/j.ipl.2008.05.018.

[8] Philippe Flajolet and G. Nigel Martin. Probabilistic Counting Algorithms for Data Base Applications. *Journal of Computer and System Sciences*, 31(2):182–209, 1985. Original paper; doi:10.1016/0022-0000(85)90041-8.

[9] Noga Alon, Yossi Matias, and Mario Szegedy. The Space Complexity of Approximating the Frequency Moments. *Journal of Computer and System Sciences*, 58(1):137–147, 1999; conference version, *STOC 1996*. Author manuscript.

[10] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, D. Sivakumar, and Luca Trevisan. Counting Distinct Elements in a Data Stream. In *RANDOM 2002*, Lecture Notes in Computer Science 2483, pages 1–10, 2002. doi:10.1007/3-540-45726-7_1.

[11] Piotr Indyk and David Woodruff. Optimal Approximations of the Frequency Moments of Data Streams. In *Proceedings of STOC*, pages 202–208, 2005. doi:10.1145/1060590.1060621.

[12] Daniel M. Kane, Jelani Nelson, and David P. Woodruff. An Optimal Algorithm for the Distinct Elements Problem. In *Proceedings of PODS*, pages 41–52, 2010. doi:10.1145/1807085.1807094.

[13] Graham Cormode and S. Muthukrishnan. An Improved Data Stream Summary: The Count-Min Sketch and Its Applications. *Journal of Algorithms*, 55(1):58–75, 2005. doi:10.1016/j.jalgor.2003.12.001.