華東師範大學
EAST CHINA NORMAL UNIVERSITY

数据科学与工程算法基础

Algorithm Foundations of Data Science and Engineering

第二章 抽样算法

$$(1+x)^n = 1 + \frac{nx}{1!} + \frac{n(n-1)x^2}{2!} + \cdots$$

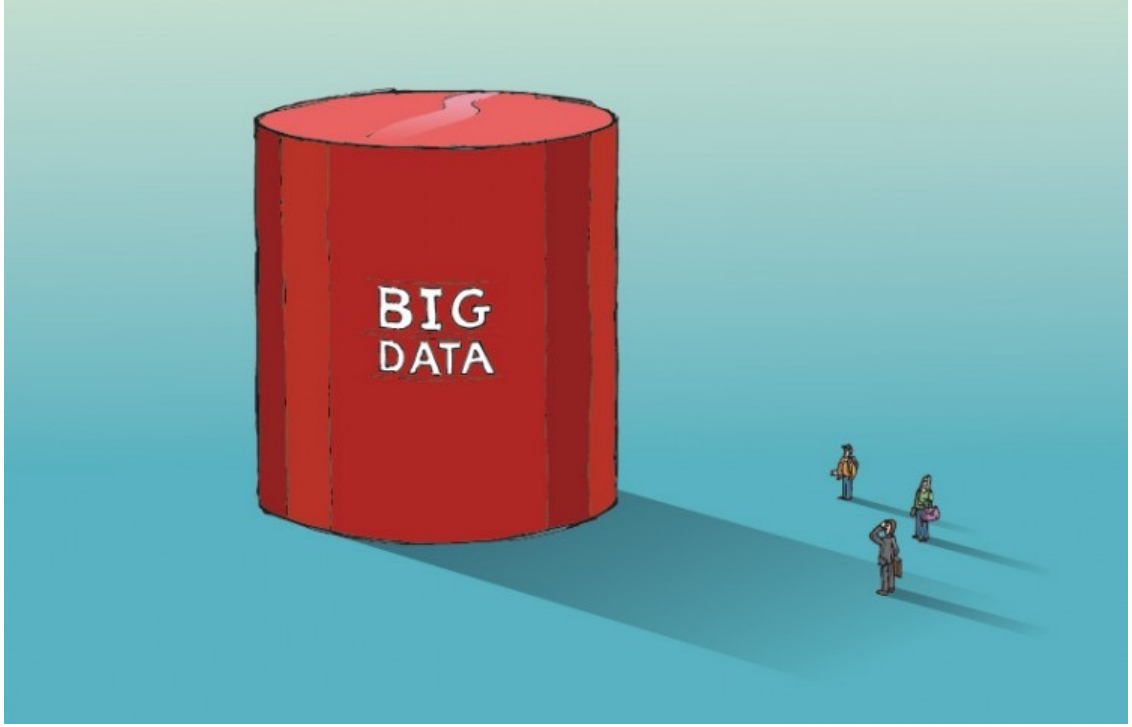
课程提纲

Content

课程提纲

Content

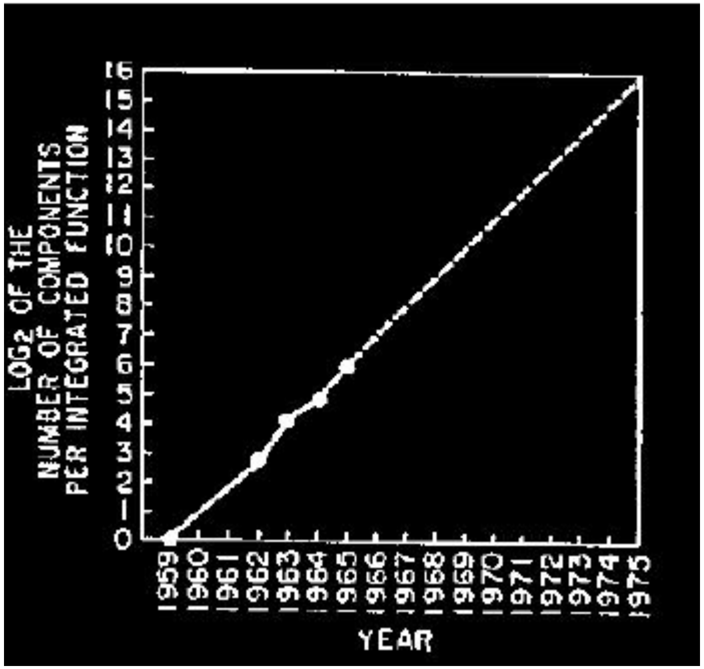
数据规模的变化趋势

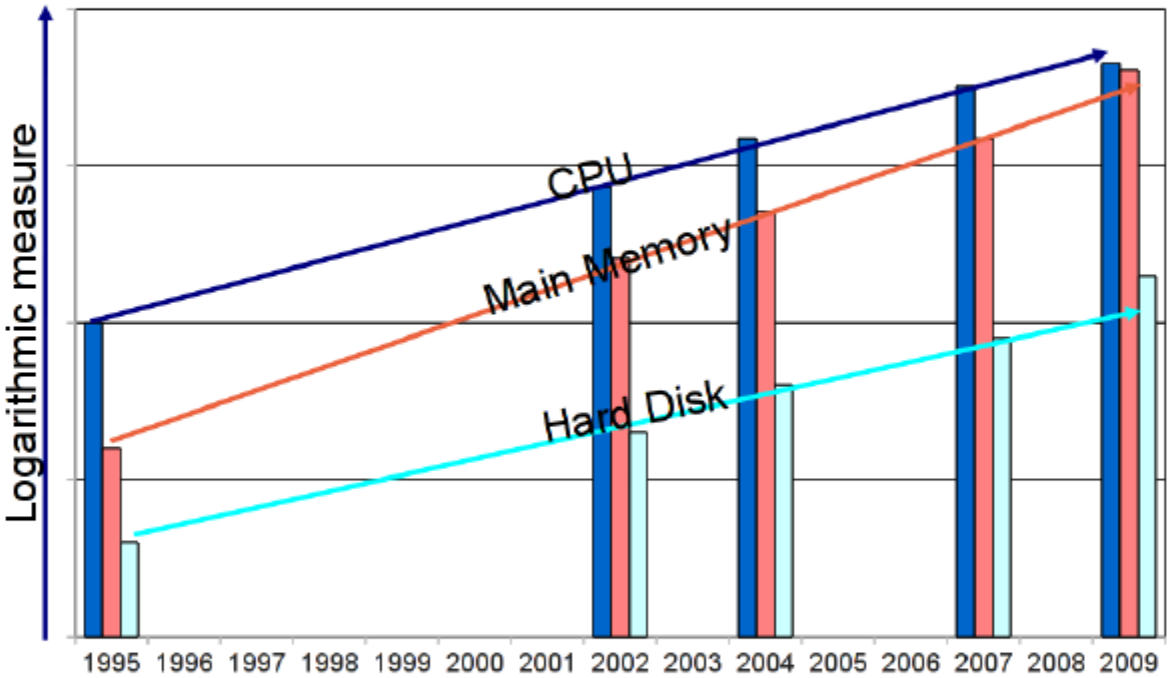


- 大数据年增加61%
- 90%的数据是过去10年内产生的
- 物联网进一步提升我们的数据获取能力
  - 5G时代来临
  - 人工智能
  - 云计算
  - 边缘计算

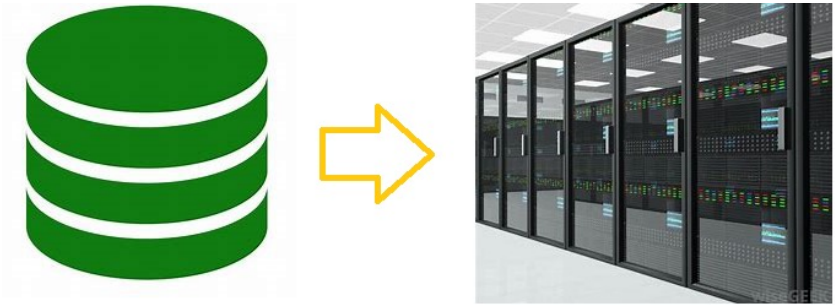
后摩尔定律时代

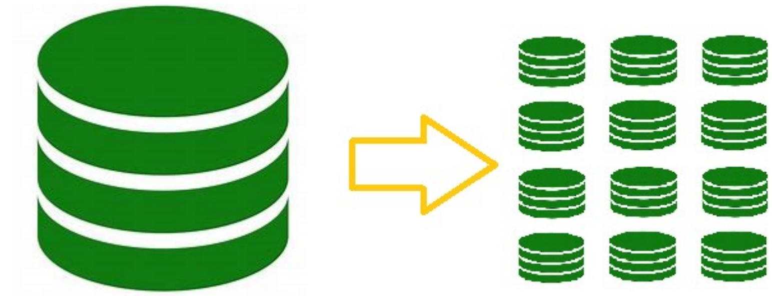
- 摩尔定律 (Moore’s Law)
- 以“九章”为代表的量子计算可能是应对后摩尔定律时代的有效方法





大数据的分析处理





- 由于数据的快速产生和数据结构的多样性，大规模数据处理能力至关重要
- 但这可能并不是唯一选择
- 在有些场景中，可能不需要处理每个数据对象
  - 比如在民调中，可能只需要随机抽取比如 1000 人进行调查
  - 有了这些数据，可以推断出其他公众的倾向
- 如何选择 1000 人成为一个问题

样本

- **样本**是总体的一个子集，对其进行观察以获得关于总体的信息
  - 研究样本的目的在于得到有关总体的有效结论
  - 因此，样本需要具有“代表性”和“广泛性”
- 常用的抽样技术有
  - 简单随机抽样
  - 系统抽样
  - 分层抽样
  - 聚类抽样
  - 多阶段抽样
- 所有这些方法都属于**概率抽样**

课程提纲

Content

简单随机抽样

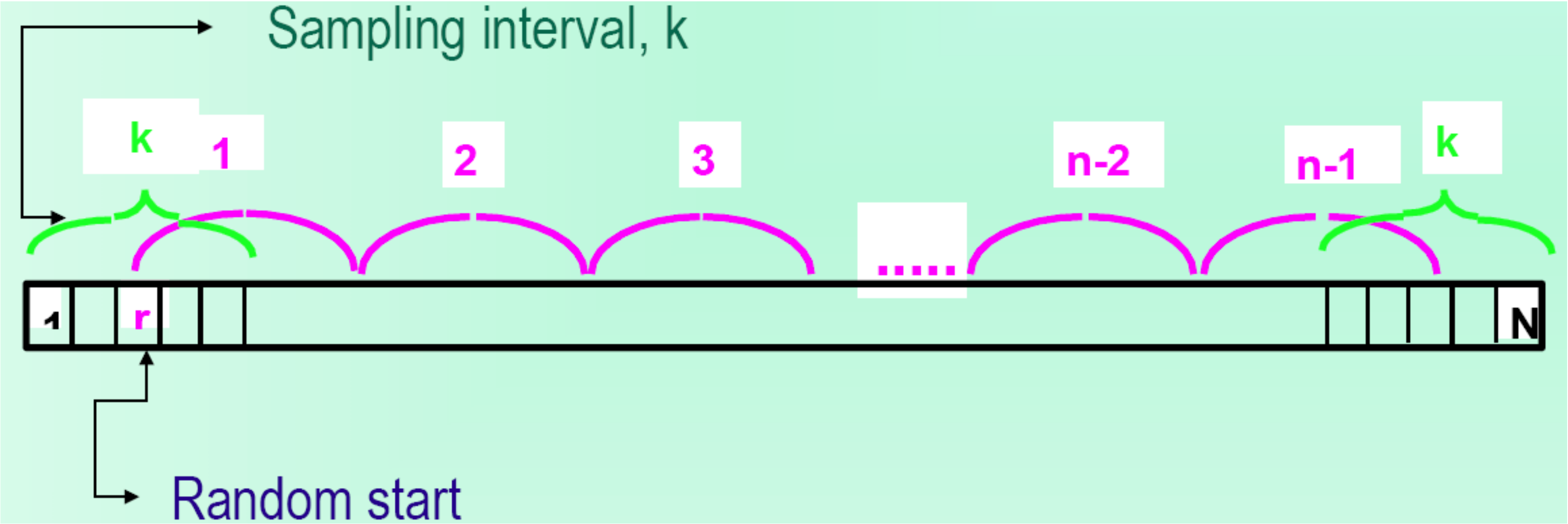
令 N 为总样本数，X_i 为一随机变量

$$X_i = \begin{cases} 1, & \text{第 } i \text{ 个对象被选择} \\ 0, & \text{否则} \end{cases}$$

其中 $\mathrm{P}(X_i = 1) = p$，为抽样率

- 在抽样过程中，选择每个对象的概率是相同的
  - 简单随机抽样是最简单的概率抽样方法
  - 简单随机抽样是一种特殊的等概率选择方法
    - ✓ 有放回的抽样统计
    - ✓ 无放回的抽样统计
  - 当总体容量很大时，无放回抽样可以近似看作等概率抽样

系统抽样



- 假设从容量为 N 的总体中抽取容量为 n 的样本
- 按下列步骤在总体中选择样本（N 能被 n 整除）
  - 确定样本大小 n 并计算跳跃间隔 $k=\frac{N}{n}$
  - 在 1 到 k 之间（包含 1 和 k）随机选择一个起始点 r
  - 反复地在选定第 $(r+k\cdot i)$ 个样本点，其中 $i=1,2,\ldots,n-1$

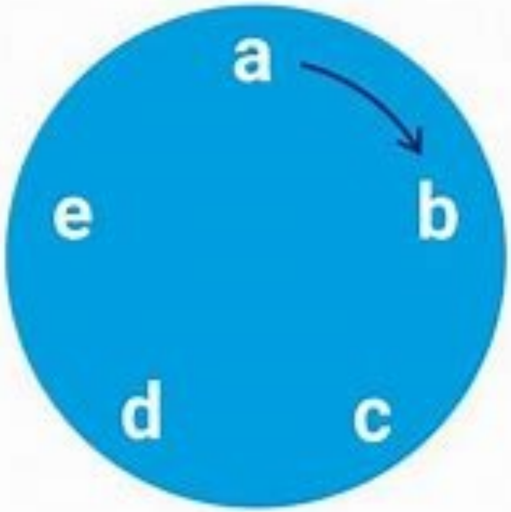
直线等距抽样的问题

- 如果 N 能被 n 整除
  - 系统抽样相当于总体被分成 n 组，每组有 k 个样本，每个样本点被选择的概率为 $\frac{1}{k}$
  - 在这种情况下，线性系统抽样方法是等概率的
- 但 N 如果不能被 n 整除
  - 最后一组样本点个数不及 k 个
  - 此时系统抽样就不是等概率的
- 可以利用循环系统抽样的方法来解决这一问题

例题：直线等距抽样

- 某车间生产100个产品，需要从中抽取10个产品作为样本。当第一个被抽取的样本为5号样本时，写出直线等距抽样所得到的样本集合。

圆形等距抽样



- 确定间隔 k，使得 $k=\left\lfloor\frac{N}{n}\right\rfloor$
  - $\mathrm{N}=15$，$n=4$
  - k 应取 3
- 从 1 到 N 之间随机开始
  - 每次在圆圈上跳过 k 个样本来选择下一个样本
  - 直到选择了 n 个样本为止
- 因此，可能选出N个不同的样本集，而不是 k 个
- 此时，这种抽样方法也是一种等概率抽样

例题：圆形等距抽样

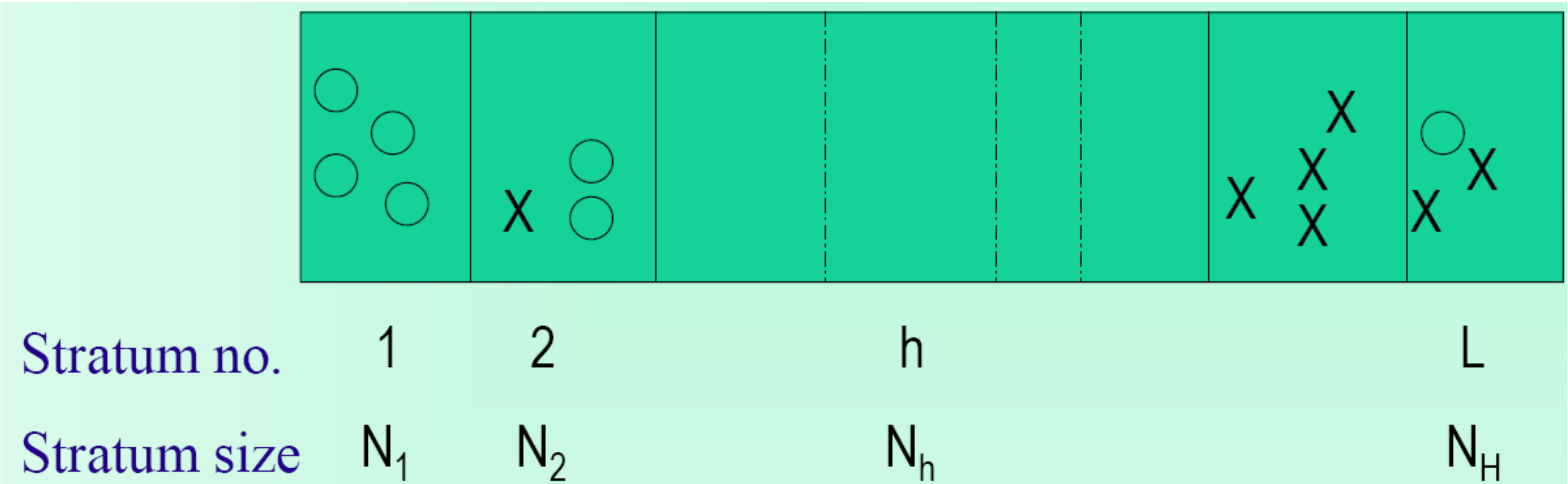
- 某车间生产100个产品，需要从中抽取9个产品作为样本。当第一个被抽取的样本为5号样本时，写出圆形等距抽样所得到的样本集合。

系统抽样的优缺点

- 优点
  - 操作简单，容易取样
  - 它使样本更均匀地分布在总体中
  - 比简单随机抽样更有效，尤其是当列表中的样本点顺序与关注变量的特征无关时

- 缺点
  - 一个不好的样本点排列可能会产生一个不具有代表性的样本集
  - 非严格等概率抽样，抽样误差计算复杂

分层抽样



- 根据某种属性（比如性别、职称、院系、地域等）将总体分为多个不同的组
- 每层由满足分层变量值设定条件的样本点组成
  - 减少估计的标准误差
  - 提供总体不同组别的单独估计（“域”估计）
  - 对不同组别，可以使用不同的抽样方法并行地提高抽样效率

样本量分配方法

- 等额样本法
  - 每层样本量都是 $n_h = \frac{n}{L}$

- 等比例分配法
  - 保证每组 $\frac{n_h}{N_h}$ 是相同的，其中 n_h 和 N_h 分别为第 h 层抽样数量和样本点总数
  - 抽样比例 $\frac{n_h}{N_h} = \frac{n}{N}$
  - 第 h 层被选择的样本数量为 $n_h = \frac{n}{N} N_h$

例题：分层抽样

- 某市有5个行政区，人口是分别为5万，3万，3万，2万，2万。如果我们抽取300人进行收入调查，使用等额样本法和等比例样本法应如何分配样本数量？

样本量分配方法

- 对每组使用不同的抽样率
  - **目的**：给定样本容量，使总体均值的方差最小
  - **奈曼分配法**：$n_h = \frac{N_h \cdot S_h}{\sum_{h=1}^{L} N_h \cdot S_h} \cdot n$，其中 S_h 为第 h 层的样本标准差（但是准确估计样本标准差有时是困难的）
  - **经济分配法**：$n_h = \frac{N_h \cdot S_h / C_h}{\sum_{h=1}^{L} N_h \cdot S_h / C_h} \cdot n$，其中 C_h 为第 h 层的抽样成本

课程提纲

Content

问题背景

- 想象流数据应用场景
  - 谷歌搜索引擎的关键词查询
  - 电信骨干网络中转发的数据包
- 如果想从流数据中抽取容量为 1000 的样本，怎么办？
  - 在流数据应用场景中，总体容量是无法事先知晓的
  - 另外，可能需要随时返回样本
- 一种简单方法
  - 为每个到达的元素产生一个随机数，每次返回 top-1000 个元素
  - 类似于包含`Order by Rand()` 的 SQL 查询语句
  - 但是这种方法的一个问题是，需要消耗 $O(N)$ 的空间
  - 在流数据场景下，可能是无法做到的
- 水库抽样是一种针对流数据的高效等概率抽样方法

水库抽样

- 水库抽样又叫做蓄水池抽样
  - 创建一个长度为 1000 的数组（水库）
  - 如果流数据不超过 1000 个元素，每个元素都存入该数组中
  - 处理第 i 个元素，比如 $i = 1001$，该如何做？
    - ✓ 第 1001 个元素出现在样本集合中的概率为 $\frac{1000}{1001}$
    - ✓ 为了等概率抽样，每个元素出现在样本集合中的概率都应该为 $\frac{1000}{1001}$
  - 当第 1001个元素到达时，是否能从数组中随机替换其中的一个元素呢？
    - ✓ 某个元素还在样本集合中的概率为

$$1\times\frac{1}{1001}+\frac{999}{1000}\times\frac{1000}{1001}=\frac{1000}{1001}$$

    - ✓ 因此，这是一种等概率抽样

水库抽样算法

```
algorithm reservoir(k, S)
/* take k random samples from the dataset S */
initialize an array samples of size k
for i = 1 to n = |S|
    o = the i-th item
    if i ≤ k then
        samples[i] = o
    else
        generate a random integer from 1 to i
        if i ≤ k then
            samples[i] = o
```

元素少于 k 个

元素不少于 k 个

- 水库抽样算法非常高效
  - 每个数据项处理的时间复杂度为 O(1)
  - 空间复杂度为 O(k)，其中 k 为水库容量大小
  - 而且是一种在总体容量未知情形下的等概率抽样方法

证明

- **定理**：对容量为 $n \ge k$ 的数据集合，水库抽样方法保证每个元素以 $\frac{k}{n}$ 的概率保留在水库中，其中 k 为水库容量大小
- **证明**（数学归纳法）：
  - 当 $n = k$ 时，显然每个元素被抽到的概率为 $\frac{k}{k} = 1$
  - 假设当 $n = m$ 时，每个元素被抽到的概率为 $\frac{k}{m}$
  - 当 $n = m + 1$ 时
    - ✓ 第 $m + 1$ 个元素被抽样，则随机数在 1 到 k 之间，因此概率为 $\frac{k}{m+1}$
    - ✓ 对于其他元素，当前 m 个元素到达后，该元素在样本中的概率为 $\frac{k}{m}$。如果第 $m + 1$ 个元素未被抽样，则该元素继续保存在水库中；如果第 $m + 1$ 个元素被抽样，则该元素被替换掉的概率为 $\frac{1}{k}$。
    - ✓ 因此，其他元素被抽样的概率是 $\frac{k}{m}\left(\frac{k}{m+1} \times \left(1 - \frac{1}{k}\right) + \left(1 - \frac{k}{m+1}\right)\right) = \frac{k}{m+1}$

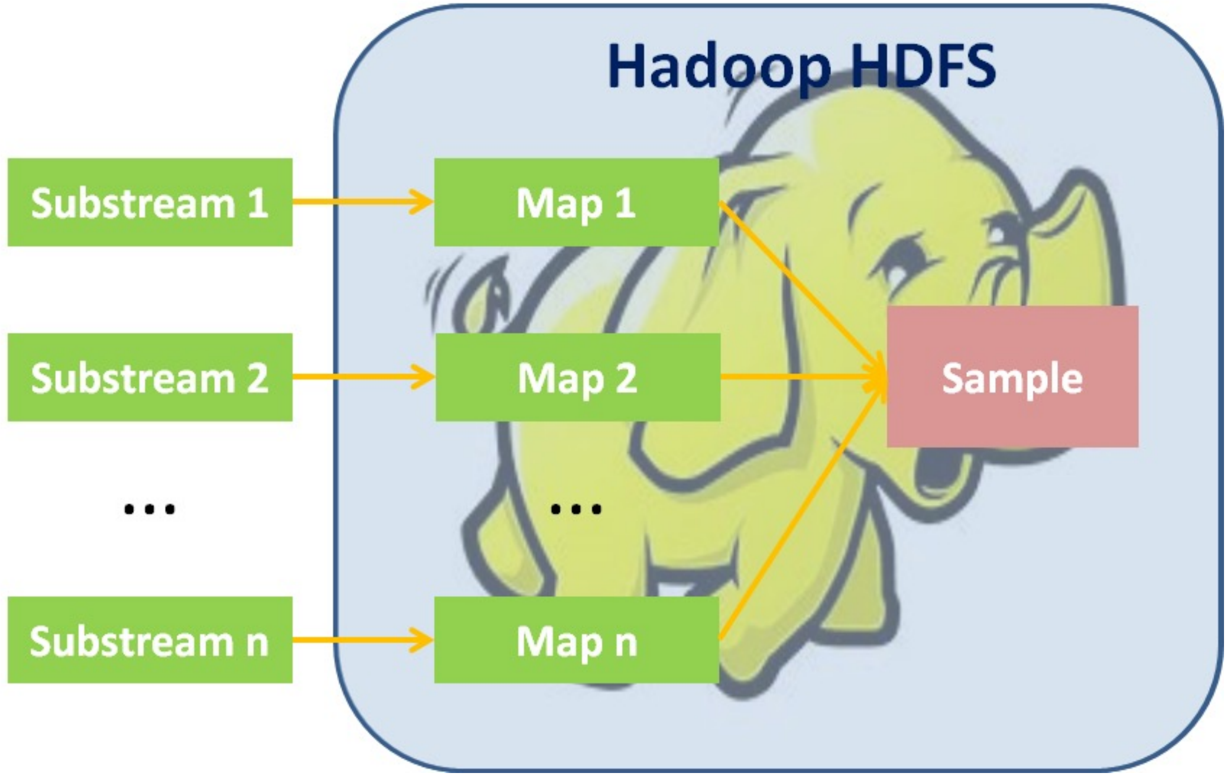
示例

- 设 $S = \{59, 100{,}2{,}30{,}63, \dots\}$ ，且 $k = 3$
- 前 k 个项直接添加到水库中，即样本集合为 $\{59, 100{,}2\}$
- 当第 4 个元素 30 到达时
  - 生成一个 1 到 4 之间的随机整数
  - 假定生成随机数 $x = 4$ ，因为 $x > k$ 该元素直接被忽略
- 当第 5 个元素 63 到达时
  - 生成一个 1 到 5 之间的随机整数
  - 假定生成随机数 $x = 2$ ，因为 $x < k$，元素 63 直接替换水库中的第 2 个元素 100
- 因此，最终的样本集合为 $\{59, 63{,}2\}$

例题：水库抽样

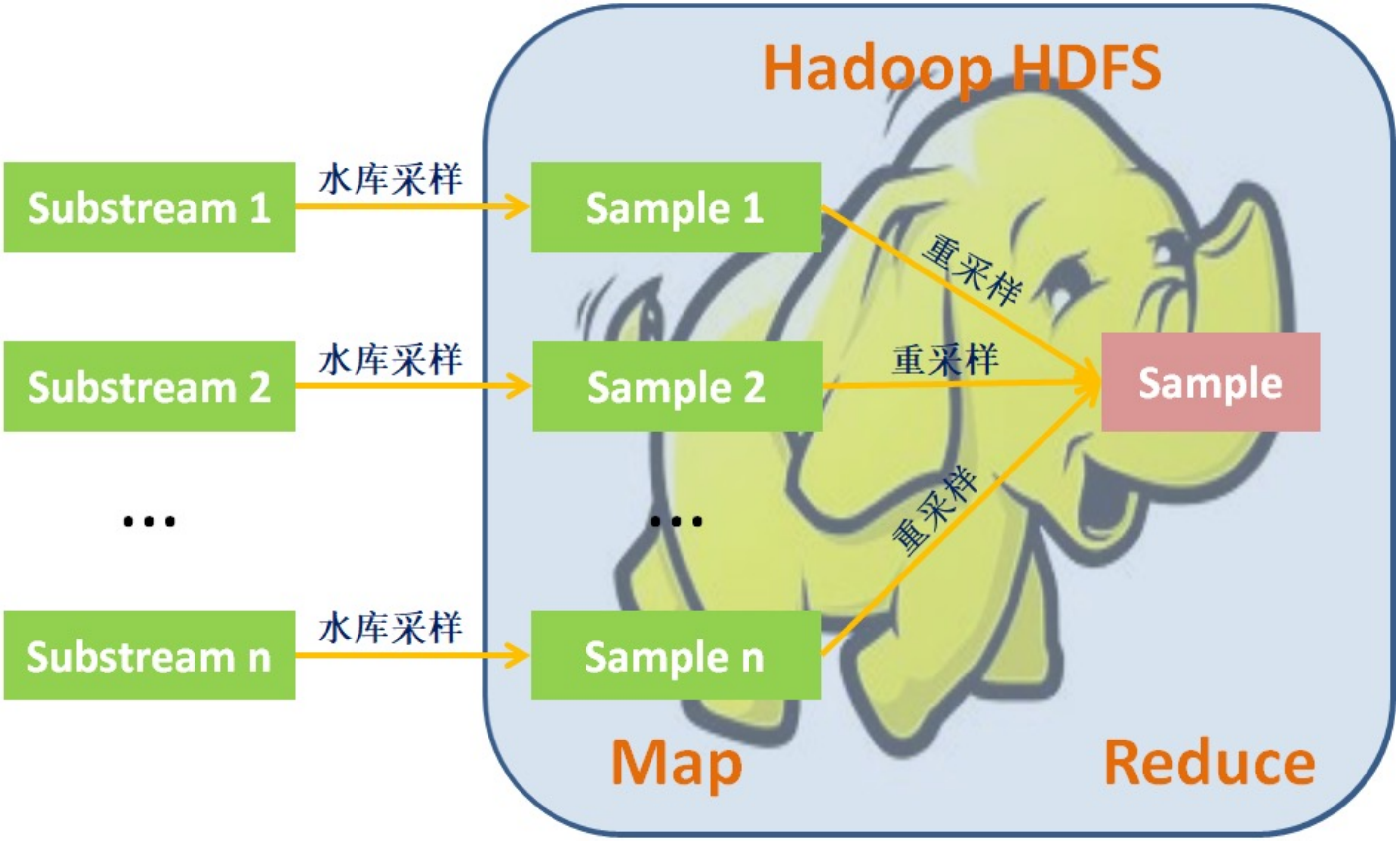
- 有如下抽样方案：当第i个元素到达时，以$1/i$的概率替换当前元素，否则保留当前元素。试证明按该方案当处理n个元素后每个元素被保留的概率都是$1/n$。

并行抽样



- 如果利用 10 台机器，能否加快水库抽样速度呢？
- 换句话说，水库抽样能否扩展成分布式算法呢？

并行抽样（续）



- 每台机器上维护同样大小的水库，分别执行水库抽样算法
- 对每个水库中的样本进行重抽样

分布式水库抽样方法

Algorithm: Distributed reservoir sampling algorithm

Input : # Maps is n
Output: Sample H of size k

for *ith Map for* $1 \leq i \leq n$ **do**
    $F_i \leftarrow$ sample of k size in ith Map;
    $N_i \leftarrow$ the number of items in ith Map;

每个节点分别进行水库采样

Initialize reservoir H;

for $1 \leq j \leq k$ **do**
    $p \leftarrow random(0, 1)$;
    Determine m s.t., $\sum_{i=1}^{m-1} N_i < p \sum_{i=1}^{n} N_i \leq \sum_{i=1}^{m} N_i$;
    Move an item from F_m into H;

重采样

return H;

- 分布式水库抽样算法也非常高效，时间复杂度为 O(1)
- 分布式水库抽样是正确的
  - 每个元素被抽样的概率是相同的
  - 任意两个元素被抽样是相互独立的

本章小结

- 抽样是处理大数据的常用方法
  - 总体容量已知
    - ✓ 简单随机抽样
    - ✓ 系统抽样
    - ✓ 分层抽样
  - 总体容量未知
    - ✓ 水库抽样
    - ✓ 分布式水库抽样
- 适用于不同数据相互独立的情形，但有些场景下数据并不一定独立
  - 图数据
  - 时间序列数据
  - ……

课后习题

课本第28-29页习题2

1，4，5，8，13，14