

# 幂律与富者愈富

姜少峰



北京大学前沿计算研究中心

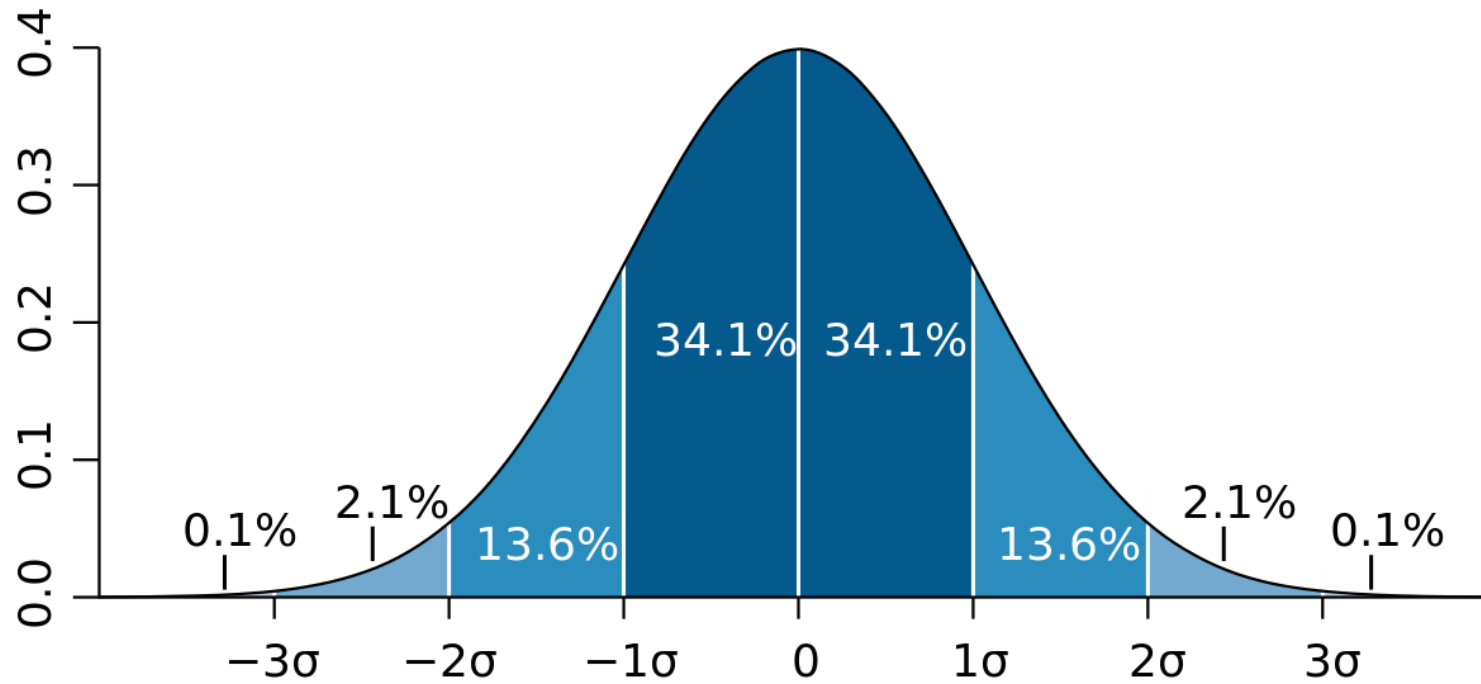
Center on Frontiers of Computing Studies, Peking University

- 流行度一般是极端不平衡的
  - 多数人只是“普通人”，少数人有更广泛的知名度，极少数世界闻名
  - 艺术与科学亦如此（书籍、绘画、电影、研究成果等）
  - 任何有“观众”、“评价”的事物皆类似

# 定量研究知名度

- 通过完整互联网快照，算出Google Amazon等知名网站的链入链接数
  - 定义指向某个网页的所有链接为该网页的链入集合
- “整个互联网中有多少个网页有 $k$ 个链入链接数？”
- “该链接数关于 $k$ 的函数是怎样的？”

# 正态分布?



$$\text{PDF: } f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$

如果是正态分布，那么每个网页应该会独立（同分布）随机决定要链接到哪个网页

# 正态分布符合直观吗？

- 如果真是正态，那应当链入数增大时，概率**指数下降**
- 但显然没看到这种情况
  - 例如，高知名度的网站有很多
  - 知名度高的往往知名度不相上下
  - 同时也存在大量中等知名度的网站，尤其是各个细分领域的网站

# 幂律分布

- 在互联网数据上的分析发现，**k个链入数的网页的比例与 $\frac{1}{k^2}$ 成正比** (Broder et al., 2000)
- 和正态分布明显的不同：随着k的增大，概率密度降低明显更慢
- **k非常大的时候，还是有相对显著的比例的网页有k个链入**



# 幂律的性质

- 一般的幂律函数  $f(k) = ak^{-c}$  ( $c > 1$ ) -- 如果  $c \leq 1$  呢?
- 标度不变性 (scale invariant) -- 正态分布/函数有吗?
  - $f(\beta k) = a(\beta k)^{-c} = \beta^{-c} f(k) \propto f(k)$
- 当  $c \leq 2$  时不存在均值/数学期望
  - $\int x \cdot x^{-c} dx = \frac{1}{2-c} x^{2-c} + C$
- 当  $c \leq 3$  时不存在方差
- 无均值/方差说明普遍存在“无限可能” / “黑天鹅”事件
  - 中心极限定理不再适用，大独立样本的平均依然不稳定
  - 人群收入符合幂律：抽样估计人群收入均值，突然抽到了大老板，均值发散
- 然而，中值median总是存在的 -- 用中值当统计量更加robust

# 幂律广泛存在

- 8-2法则：80%的财富被20%的人所有；80%的销售额来自20%的客户
- Wiki的90-9-1定律：90%用户只看不写，9%改/更新，1%新增内容
- CPU的cache miss rate是size的幂律函数
- 收入分布
- 城市人口数



# 长尾现象

- 促销应选极为流行的项目，还是种类极多但是不那么流行的项目？
  - 前一种“畅销品”数量少收益高；后一种“利基” (niche) 吸引大量细分用户
- Chris Anderson (2004)认为，**互联网的特点**将媒体和娱乐业推向了“利基”战略：用大量非流行品的“长尾”来吸引大量用户从而盈利
  - 例子：Netflix, Amazon；推荐系统的重要性；仓储压力小

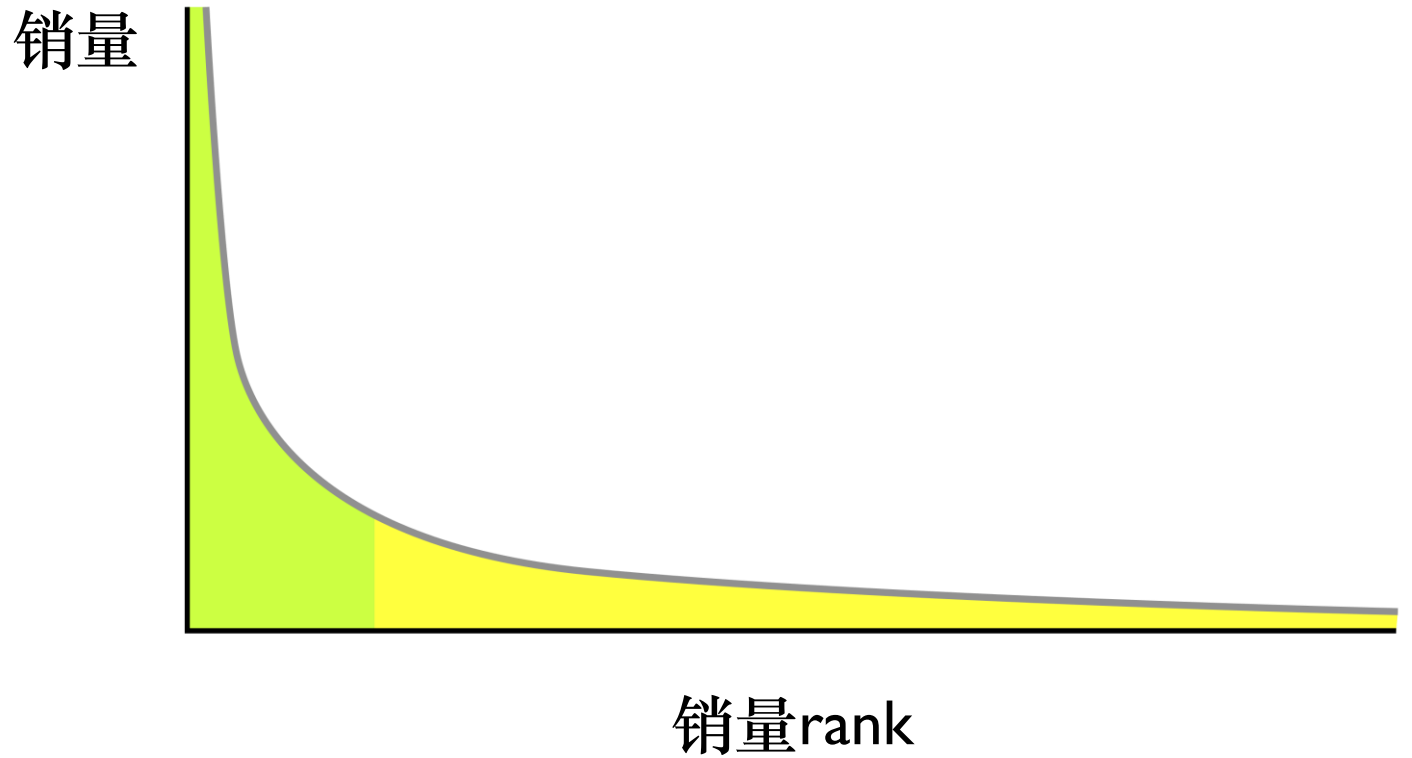


# Rank-frequency plot

- 如何平衡“畅销” vs “利基”？
- 将商品类型按照销量（降序）排序，画出对应类型的销量

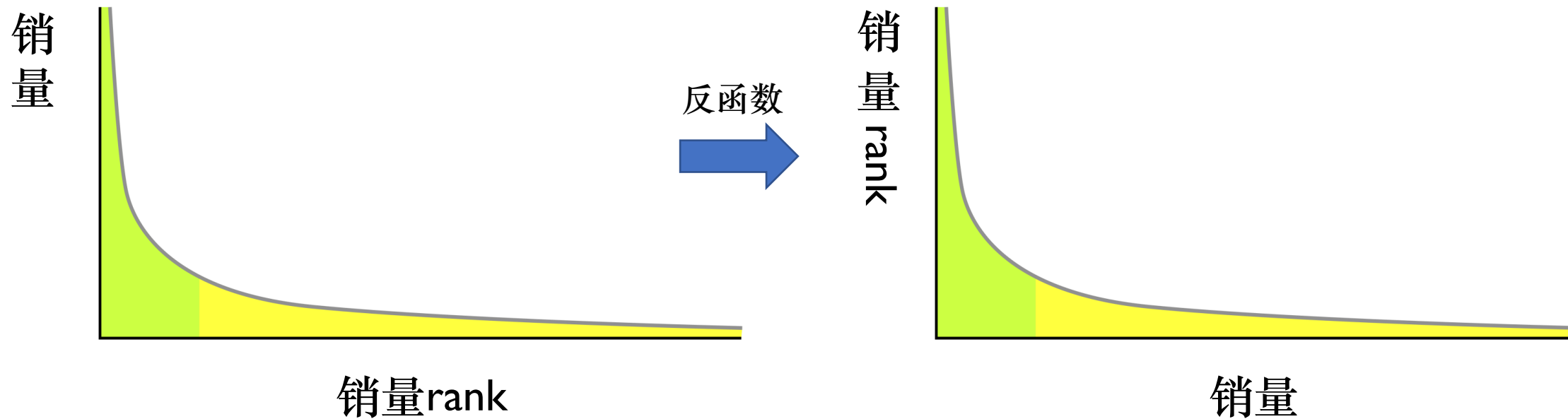
曲线下面积就是对应  
rank商品的销量

找出自己想要的分位点



# 商品排序-销量图的分布是什么

- 商品排序-销量图的分布是什么？考虑反函数：



- 反函数：作为 $k$ 的函数，销量至少是 $k$ 的商品有多少？

# 反函数服从什么分布？

- 已知：“作为k的函数，销量恰好为k的项目有数”符合幂律
- 我们需要：“作为k的函数，销量至少是k的商品有多少？”
- $f(k) = k^{-c}$  ( $c > 1$ )
- $g(k) = \int_{x=k}^{\infty} f(x) dx = \frac{x^{1-c}}{1-c} \Big|_{x=k}^{x=\infty} \propto k^{1-c}$
- 结论：g也符合幂律分布

- Rank-frequency plot称为Zipf plot
- 源于语言学家George Zipf的研究：
  - /zɪf/, not /tsɪpf/ as in German
  - Latex发音?
  - 欧洲人的名字? Jiri Matousek, Artur Czumaj, Michal Koucky, Christian Sohler?
  - 相信中译
- Zipf's law: 在英文中，第j常用的单词出现频率与  $1/j$  成正比
- 另一个相关的Zipf的发现：词长与词频成负相关

# 实证研究：如何检测是否服从幂律

- 找到 $f(k)$ 代表某个总体中，值为 $k$ 的那部分所占比例
- 希望验证 $f(k) = ak^{-c}$
- 取对数： $\log f(k) = \log a - c \log k$
  
- 如果符合幂律，那么 $\log k$ 为变量， $\log f(k)$ 为值的函数是线性的
  - 斜率是 $-c$ ，截距是 $\log a$
- 思考：如果是正态分布呢？会看到什么？

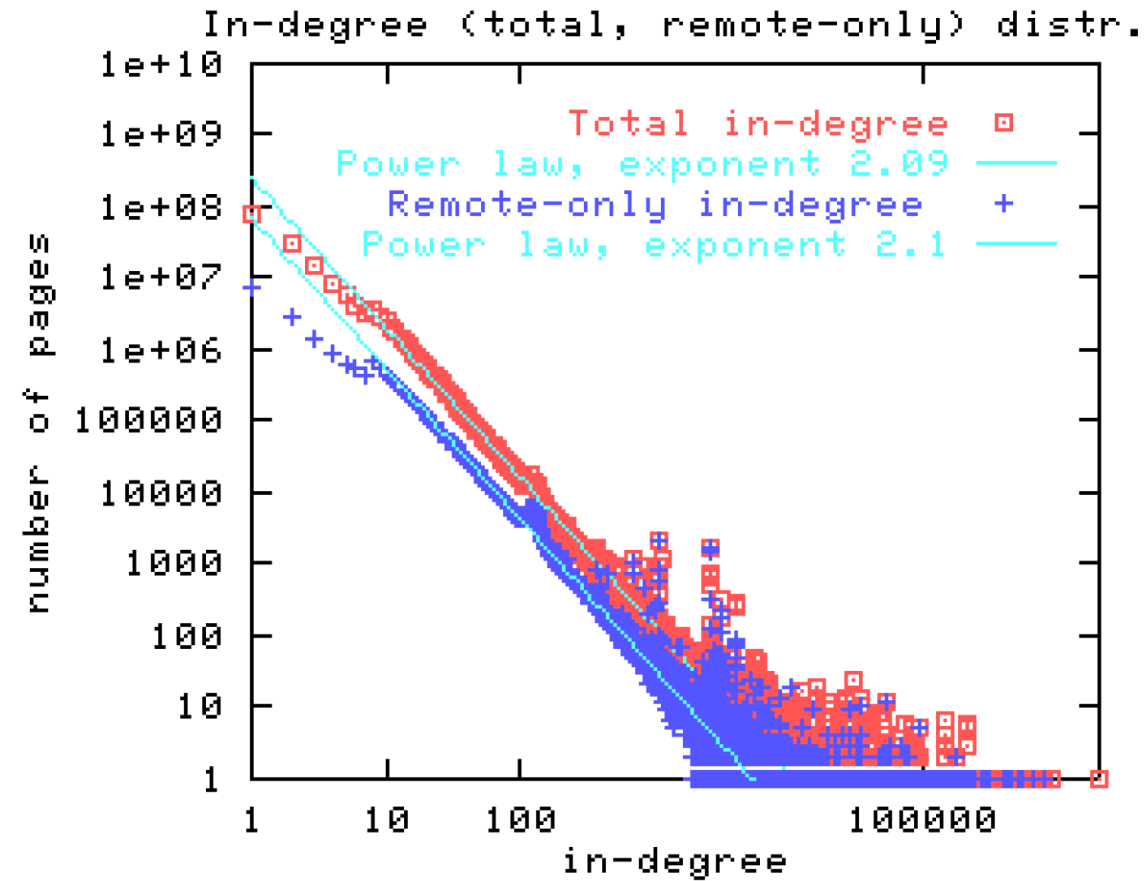


Figure 18.2: A power law distribution (such as this one for the number of Web page in-links, from Broder et al. [80]) shows up as a straight line on a log-log plot.

# 幂律存在的内部机制？

- 我们知道中心极限定理是独立同分布多次平均的结果
- 幂律：体现一个群体相互关联的决定形成的反馈
- 我们一种“总体”模型：忽略个体决策细节，而看宏观的概率行为



# Barabási-Albert富者愈富模型

(Barabási and Albert 1999; Kumar et al., 2000)

1. 网页按照顺序被创建，标为  $1, 2, \dots, N$
2. 网页  $j$  创建时，按以下方法随机产生链接 ( $p$  为  $[0, 1]$  之间的一个数)
  - a) 以概率  $p$ ， $j$  均匀随机地从已创建网页中选一个  $i$ ，创建指向  $i$  的链接
  - b) 以概率  $1 - p$ ， $j$  均匀随机地从已创建网页中选一个  $i$ ，创建指向  $i$  指向的网页的链接

(潜在地，每个网页  $j$  可以重复第2步独立建立多次链接)

- 可以证明符合幂律，幂律指数与  $p$  的选取有关 (Bollobás 2003)
  - $p$  越小，复制越频繁， $c$  相应减小，从而流行网页更可见

# 直观解释

b) 以概率  $1 - p$ ,  $j$  均匀随机地从已创建网页中选一个  $i$ , 创建**指向  $i$  指向的网页的连接**

- 这一步履行“富者愈富”原则：
  - 最终链接到某个网页的概率与该网页当前入度成正比

b') 以概率  $1 - p$ ,  $j$  根据当前链入数成正比的概率选择一个网页并创建链接

- 这种连到流行度高的网页的方式又叫“择优连接”
- 择优连接的后果是概率较大的网页的链入数会加速增长
  - 而较小的还是会维持较小：他们的微小变化趋势会成为noise被抵消

**问题：** 如果只有2.a而没有2.b呢 ( $p = 1$ ) ?

# “富者”的不可预测性

- 早期的随机连接可能会对后来谁成为“富者”产生巨大影响
  - 初始大家“富有”程度是一样的，谁更“富有”是完全随机的
  - 通常“富者”运气成分很多；有很多类似/更好的人/产品却完全销声匿迹
- 一个实证研究 (Salganik, Dodds, Watts, 2006):
  - 创造一个音乐下载网站，提供48首冷门歌曲
  - 歌曲创作质量有明显不同，全都由真实乐队演唱
  - 用户可以下载歌曲，同时每首歌会显示累计下载次数
  - 幕后：8个网站的copy同时运行，初态完全一致
- 结果：
  - 允许看下载次数时，不同网站相同歌曲流行度差别大
  - 不允许看下载次数时，不同网站歌曲流行度差别明显变小

# 与信息级联的对比

- 相似点：早期一些随机扰动对结果会产生显著影响
- 不同：
  - 流行度模型包含的可能性大大多于级联（通常只有YES NO）
  - 复制模型更加“local”，只能看到看到随机选择的其他一个网页的决定而已
  - 复制模型更加“非理性”：后人只是模仿先人的处理做出决定

# 幂律模型的分析证明

考虑等价版本（替换2.b）

1. 网页按照顺序被创建，标为  $1, 2, \dots, N$
  2. 网页  $j$  创建时，按以下方法随机产生链接（ $p$  为  $[0, 1]$  之间的一个数）
    - a) 以概率  $p$ ， $j$  均匀随机地从已创建网页中选一个  $i$ ，创建指向  $i$  的链接
    - b) 以概率  $1 - p$ ， $j$  根据当前链入数成正比的概率选择一个网页并创建链接
- 终极目标：对于每个  $k$ ，具有  $k$  个链入边的网页的个数的期望

# 模型的确定性近似

- 考虑简化版本：近似成确定性过程，但仍能证明幂律分布特征
- 设随机变量 $X_j(t)$ 代表 $t$ 时刻 $j$ 的入边数；显然需要 $t \geq j$
- 考察某个 $t + 1$ 时刻，连到 $j$ 的概率是多少？

- 以 $p$ 的概率随机连，所以这部分概率是 $\frac{p}{t}$

- 以 $q = 1 - p$ 的概率以正比于已经连的边的条数连接

- 此时，总连接边是 $t$ ，与 $j$ 连接的是 $X_j(t)$ ，所以这部分概率是 $\frac{qX_j(t)}{t}$

- 所以 $t + 1$ 时刻连到 $j$ 的概率总共是：

$$\frac{p}{t} + \frac{qX_j(t)}{t}$$

# 离散随机过程的连续确定近似

- 将这个离散事件随机过程看作连续时间上的
  - 好处：一些细微概率变化被忽略，只关注主要变化量
  - 用微分方程来刻画随机过程，随机性被建模在了微分方程里 -> 确定性过程

• 设函数  $x_j(t)$  代表  $t$  时刻  $j$  网页的连入数  $X_j(t)$  的连续近似

• 根据刚刚的分析， $t + 1$  轮连到  $j$  的概率增量是：

$$\frac{p}{t} + \frac{qX_j(t)}{t}$$

• 那么将这个考虑成连续时间上的变化，就是

$$\frac{dx_j}{dt} = \frac{p}{t} + \frac{qx_j}{t}$$

# 解微分方程

$$\frac{dx_j}{dt} = \frac{p + qx_j}{t}$$

$$\frac{dx_j}{p + qx_j} = \frac{dt}{t}$$

• 两边积分:

$$\ln(p + qx_j) = q \ln t + C$$

$$p + qx_j = A \cdot t^q$$

$$x_j(t) = \frac{1}{q} (At^q - p)$$



$$x_j(t) = \frac{1}{q} (At^q - p)$$

- 由初始条件  $x_j(j) = 0$ , 可得  $A = \frac{p}{j^q}$ , 代入可得

$$x_j(t) = \frac{1}{q} \left( \frac{p}{j^q} \cdot t^q - p \right) = \frac{p}{q} \left( \left( \frac{t}{j} \right)^q - 1 \right)$$

- 此时我们有了对每个时刻  $t$ ,  $j$  的链入数

$$x_j(t) = \frac{1}{q} \left( \frac{p}{j^q} \cdot t^q - p \right) = \frac{p}{q} \left( \left( \frac{t}{j} \right)^q - 1 \right)$$

- 我们要 $f(k)$ : 对每个链入数 $k$ , 时间 $t$ , 有多少比例的 $j$ 的链入数是 $k$
- 中间步骤 $F(k)$ : 对每个链入数 $k$ 时间 $t$ , 有多少比例的 $j$ 的链入数至少是 $k$
- 也就是问有多少 $j$ , 使得 $x_j(t) \geq k$ ?

$$x_j(t) \geq k$$

$$x_j(t) = \frac{p}{q} \left( \left( \frac{t}{j} \right)^q - 1 \right) \geq k$$
$$j \leq t \cdot \left( \frac{qk}{p} + 1 \right)^{-\frac{1}{q}}$$

- 这样的j在t时刻所占比例:

$$F(k) = \left( \frac{q}{p} \cdot k + 1 \right)^{-\frac{1}{q}}$$

- $F(k)$ : 对每个链入数 $k$ 时间 $t$ , 有多少比例的 $j$ 的链入数至少是 $k$

- $F(k) = \left(\frac{q}{p} \cdot k + 1\right)^{-\frac{1}{q}}$

- $f(k)$ 应该对应于 $F(k) - F(k+1)$

- 但是我们进一步用连续近似, 对 $F(k)$ 求导数后取负:

- $f(k) = -F'(k) = -\frac{q}{p} \left(-\frac{1}{q}\right) \left(\frac{p}{q} \cdot k + 1\right)^{-1-\frac{1}{q}} = \frac{1}{p} \left(\frac{p}{q} \cdot k + 1\right)^{-1-\frac{1}{q}}$

- 所以对应于power law的指数 $1 + \frac{1}{q} = 1 + \frac{1}{1-p}$

计算问题： heavy hitter

# heavy hitter

- $n$ 个整数 $a_1, \dots, a_n$ ，输出至少出现了 $\alpha n$ 次的整数 ( $0 < \alpha < 1$ )
  - 利用哈希表可以做到 $O(n)$ 时间和空间
- 假设 $n$ 个整数以巨大的数据流给出，如何能以 $o(n)$ 的空间求heavy hitter?
  - 只单向扫数据流一遍，不能回退
- 场景：
  - 大规模数据挖掘，数据在外部存储而不在主存上（而且也存不下）
  - 只关心heavy hitter，其他的不想存
  - “小”设备求heavy hitter，计算资源极端有限（路由器统计top访问ip）
- 观察：出现大于 $\alpha n$ 次的整数至多只有 $1/\alpha$ 个
  - 能不能用这个空间来做？

# 计算采样的heavy hitter算法

- 设 $m = \alpha n$ ,  $L \approx \log \frac{n}{m}$ 是一个待定参数
  1. 每来一个元素 $a_i$ , 以 $\frac{L}{m}$ 的概率留下 $a_i$ , 否则丢弃
  2. 如果 $a_i$ 留下了, 把 $a_i$ 插入哈希表
  3. 如果 $a_i$ 没留下, 但是 $a_i$ 在哈希表里, 将 $a_i$ 的counter + 1
  4. 最后, 将counter至少是 $\frac{m}{2}$ 的留下, 输出
- $\mathbb{E}[\text{size of hash table}] = \frac{Ln}{m}$ , Chernoff bound可以推出high prob bound
- Correctness: 返回的都出现了至少 $\frac{m}{2}$ 次, 出现了 $m$ 次的一定都返回了

### Chernoff bound (Chernoff, 1952)

设 $X_1, \dots, X_n$ 是独立的 $[0, 1]$ 上的随机变量。令 $X = \sum_{i=1}^n X_i$ ,  $\mu = E[X]$ 。那么

$$\forall t \in (0, 1) \quad \Pr[|X - \mu| \geq t\mu] \leq 2\exp(-t^2\mu/3)$$

$$\forall t > 0 \quad \Pr[|X - \mu| \geq t\mu] \leq 2\exp(-t^2\mu/(2+t))$$

- $E[\text{size of hash table}] = \frac{Ln}{m}$ , Chernoff bound可以推出high prob bound
- Correctness: 返回的都出现了至少 $\frac{m}{2}$ 次, 出现了 $m$ 次的一定都返回了
- L最后该怎么选?



- 设  $m = \alpha n$ ,  $L \approx \log \frac{n}{m}$  是一个待定参数

1. 每来一个元素  $a_i$ , 以  $\frac{L}{m}$  的概率留下  $a_i$ , 否则丢弃
2. 如果  $a_i$  留下了, 把  $a_i$  插入哈希表
3. 如果  $a_i$  没留下, 但是  $a_i$  在哈希表里, 将  $a_i$  的 counter + 1
4. 最后, 将 counter 至少是  $\frac{m}{2}$  的留下, 输出

- 如果不知道  $n$ , 只知道  $\alpha$  呢? 问题: 不知道  $m$ 。 (但是假定知道  $L$ )

- 算法改成:  $a_i$  留下的概率改成  $\frac{L}{\alpha i}$  其他分析?

# 基于hash的方法：CountMin Sketch

For integer  $m, k$ , function  $h: [m] \rightarrow [k]$  is called **universal hash** if

$$\forall x \neq y \in [m], \quad \Pr[h(x) = h(y)] \leq \frac{1}{k}$$

这里  $[n] = \{1, 2, \dots, n\}$

换句话说，就是collision的概率是值域大小分之一

**CountMin Sketch** (Cormode, Muthukrishnan, 2004)

1. 初始化一个长度为  $w$  的计数数组  $A$ ,  $w = \lceil 2/\alpha \rceil$
2. 构造一个 universal hash function  $h: [m] \rightarrow [w]$
3. 对于  $a_i$ , 更新  $A[h(a_i)] = A[h(a_i)] + 1$

- 设每个元素 $e$ 的frequency是 $f_e$
- 对于每个元素 $e$ ，以至少1/2概率，我们有

$$f_e \leq A[h(e)] \leq f_e + \alpha n$$

Markov's inequality (Markov or Chebyshev, ~18XX)

设 $X$ 是**非负**随机变量。那么

$$\forall t > 0 \quad \Pr[X \geq t] \leq \frac{E[X]}{t}$$

- 首先， $f_e \leq A[h(e)]$ 是显然的，因为 $h$ 是将大域映射到小域的哈希
- $\mathbb{E}[A[h(e)]] - f_e = \sum_{d \neq e} f_d \cdot \Pr[h(d) = h(e)] \leq \frac{n}{w} \leq \frac{\alpha}{2} \cdot n$
- 由Markov's inequality:  $\Pr[A[h(e)] \geq f_e + \alpha \cdot n] \leq \frac{1}{2}$

$$w = \lceil 2/\alpha \rceil$$

# Probability amplification

- 如何把0.5概率变成高概率保证

1. 独立运行 $t$ 个独立的算法实例,

2. 对于每个查询 $e$ , 返回  $\min_{i=1, \dots, t} A_i[h_i(e)]$

- 因为  $\Pr[A[h(e)] \geq f_e + \alpha \cdot n] \leq \frac{1}{2}$

- $\Pr \left[ \min_{i=1, \dots, t} A_i[h_i(e)] > f_e + \alpha \cdot n \right] = \prod_{i=1}^t \Pr[A_i[h_i(e)] > f_e + \alpha n] \leq \frac{1}{2^t}$

# 一般的amplification方法

- 刚刚可以取min是因为从来不会低估
- 那么如果可能低估呢?
- 例如算法A以2/3概率可以输出一个在 $[f_e, f_e + \alpha n]$ 的解
- 如何通过类似的amplification来把概率变成任意高?
- **median trick: run A independently t time, and report the median value**
- 分析:
  - 用Chernoff bound证明小于 $f_e$ 以及大于 $f_e + \alpha n$ 的输出的个数高概率在 $\frac{t}{3}$ 附近
  - 所以排 $\frac{t}{2}$ 的结果高概率在 $[f_e, f_e + \alpha n]$

# 总结

- 以至少  $1/2$  概率，对于所有元素  $e$ ，我们有
$$f_e \leq A[h(e)] \leq f_e + \alpha n$$
- 推论：我们对于每个频数有  $\alpha n$  相加误差的估计 – point count
- 扩展：inner product
  - 输入两列数  $A$  和  $B$ ，用同一个 hash  $h$ ，分别维护  $A$  和  $B$  两个 count 数组
  - 最后直接返回  $A$  数组和  $B$  数组的 inner product
  - 误差分析？ Additive error  $\alpha \cdot |A|_1 |B|_1$
- 扩展：range count
  - $Q(i, j): \sum_{k=i}^j a_k$
  - 误差：  $\alpha \log n \cdot |A|_1$

- CountMin sketch是可合并的、支持删除的
  - 用同一套universal hash, 分别计算两个输入序列X、Y上的CountMin sketch
  - 最后可以把维护的A数组逐项求和叠加得到整个 $X \cup Y$ 数据集的CountMin
  - 同样的, 逐项相减可以得到从X删除Y后的数据集

# 如何构造universal hash?

- 想要构造  $h : [m] \rightarrow [k]$

$$\forall x \neq y \in [m], \quad \Pr[h(x) = h(y)] \leq \frac{1}{k}$$

1. Pick a prime number  $p > m$
2. 从  $\{1, \dots, p-1\}$  随机选取一个  $a$
3. 从  $\{0, \dots, p-1\}$  随机选取一个  $b$
4. 返回  $h(x) = ((ax + b) \bmod p) \bmod k$

分析:  $x \neq y$  且  $h(x) = h(y)$  仅当对  $i \in \left[0, \frac{p-1}{k}\right]$

$$ax + b \equiv ay + b + i \cdot k \pmod{p}$$

等价于  $a \equiv i \cdot k \cdot (x - y)^{-1} \pmod{p}$

$$\Pr[\ ] \leq \frac{p-1}{k} \cdot (p-1) \leq \frac{1}{k}$$

Universal hash 必须是随机的:

对于 fixed hash, 存在约  $\frac{m^2}{k}$  对  $x, y$  s.t.  $h(x) = h(y)$

随机性: 将  $m^2$  对冲突 “均分”

相关例子: integration by sampling

Oracle access to  $f : [0, 1] \rightarrow [0, 1]$

估计  $\int_0^1 f(x) dx$

确定性算法: 必然存在一个  $f$  使得 additive error 0.5

随机算法: Chernoff bound