Agenda

What have we done?

- Probabilistic thinking!
- Mild introduction to
  - Probability theory
  - The probabilistic method
  - Randomized algorithms
  with quite a few examples.

Next

- The Balls into Bins Model
Balls into Bins

Throw $m$ balls into $n$ bins, compute

1. the distribution of \# balls thrown until bin 1 is not empty
2. the distribution of \# balls thrown until no bin is empty
3. the distribution of the numbers of balls in bins?
4. $\Pr[\text{some bin has } \geq 2 \text{ balls}]$ (birthday paradox, hash collision)
5. $\Pr[\text{bin } i \text{ has } c \text{ balls}], E[\# \text{ balls in bin } i]$
   - when $c = 0$, think of the number of empty hash buckets
6. the distribution of the maximum load
3. The Exact Distribution

- Let $X_i = \#$ balls in bin $i$, $i \in [n]$
- For any $k_1, \ldots, k_n$ with $\sum k_i = m$,

$$\text{Prob}[(X_1, \ldots, X_n) = (k_1, \ldots, k_n)] = \binom{m}{k_1, \ldots, k_n} \left(\frac{1}{n}\right)^m$$

(Just a multinomial distribution with $p_i = 1/n, \forall i$.)

- It’s often hard/messy/impossible to compute things with this formula
- Try: probability that some bin has $\geq 2$ balls

$$= 1 - \sum_{\substack{k_1 + \cdots + k_n = m \\ k_i \leq 1, \forall i}} \binom{m}{k_1, \ldots, k_n} \left(\frac{1}{n}\right)^m$$

- Depending on the question, two typical strategies:
  - A more “local” look (see next two examples)
  - A good approximation (examples after that)
4. Probability that some bin has $\geq 2$ balls

- $m$: number of passwords, $n$: hash domain size
- $=\text{hash collision probability (huge assumption on uniformity)}$
- Want to know
  - How small should $m$ be s.t. $\text{Prob[collision]} \leq \epsilon$ (hash collision)
  - How large should $m$ be s.t. $\text{Prob[collision]} \geq 1/2$ (birthday paradox)

Looking at non-empty bins one by one,

$$\text{Prob[no collision]} = \left(1 - \frac{1}{n}\right) \left(1 - \frac{2}{n}\right) \ldots \left(1 - \frac{m - 1}{n}\right)$$

Applying $e^{-x-x^2} \leq 1 - x \leq e^{-x}$,

$$\exp\left\{-\sum_{i=1}^{m-1} \left(i/n+i^2/n^2\right)\right\} \leq \text{Prob[no collision]} \leq \exp\left\{-\sum_{i=1}^{m-1} i/n\right\}$$
There are constants $c_1, c_2, c_3$ such that
\[
\exp \{- (c_1 m^2 / n + c_2 m^3 / n)\} \leq \text{Prob}[\text{no collision}] \leq \exp \{- c_3 m^2 / n\}
\]

For $\text{Prob}[\text{collision}] \leq \epsilon$, only need
\[
\exp \{- (c_1 m^2 / n + c_2 m^3 / n)\} \geq 1 - \epsilon
\]

$m = O(\sqrt{n})$ is sufficient

For $\text{Prob}[\text{collision}] \geq 1/2$, only need
\[
\exp \{- c_3 m^2 / n\} \leq 1/2
\]

and $m = \Omega(\sqrt{n})$ is sufficient
5. Distribution of the number of balls in a given bin

- For any \( k \), the probability that bin \( i \) has \( k \) balls is
  \[
  \text{Prob}[X_i = k] = \binom{m}{k} \left( \frac{1}{n} \right)^k \left( 1 - \frac{1}{n} \right)^{m-k}
  \]
  - i.e. \( X_i \in \text{Binomial}(m, 1/n) \)

- Question: what’s the expected number of bins with \( k \) balls?

  - Note:
    \[
    \frac{1}{k!} \cdot \frac{m(m-1) \cdots (m-k+1)}{n^k} \cdot \left( 1 - \frac{1}{n} \right)^{m-k} 
    \approx e^{-m/n} \left( \frac{m}{n} \right)^k \frac{1}{k!}
    \]
X has a Poisson distribution with mean \( \lambda \) iff

\[
\Pr[X = k] = \frac{e^{-\lambda} \lambda^k}{k!}, \quad k = 0, 1, 2, \ldots
\]

\[
E[X] = \lambda
\]

\[
\text{Var}[X] = \lambda
\]

\( X \in \text{Poisson}(\lambda), Y \in \text{Poisson}(\mu), \) then \( X + Y \in \text{Poisson}(\lambda + \mu) \)

**Theorem (Poisson Approximation to the Binomial)**

Let \( Y_n \in \text{Binomial}(n, p) \), where \( \lim_{n \to \infty} np = \lambda \). Then,

\[
\lim_{n \to \infty} \Pr[Y_n = k] = \frac{e^{-\lambda} \lambda^k}{k!}
\]
PTCF: A Chernoff Bound for Poisson Variable

Let $X \in \text{Poisson}(\lambda)$,

- If $k > \lambda$, then
  \[
  \text{Prob}[X > k] \leq e^{-\lambda} \left( \frac{e\lambda}{k} \right)^k
  \]

- If $k < \lambda$, then
  \[
  \text{Prob}[X < k] \leq e^{-\lambda} \left( \frac{e\lambda}{k} \right)^k
  \]
The Poisson Approximation for Balls into Bins

- Recall $X_i = \# \text{ balls in bin } i$, and $X_i \in \text{Binomial}(m, 1/n)$
- Each $X_i$ is approximately Poisson$(m/n)$
- For $i = 1, \ldots, n$, let $Y_i$ be independent Poisson$(m/n)$ variables

**Theorem**

For any $k_1 + \cdots + k_n = m$,

$$\text{Prob} [(X_1, \ldots, X_n) = (k_1, \ldots, k_n)] =$$

$$\text{Prob} \left[ (Y_1, \ldots, Y_n) = (k_1, \ldots, k_n) \mid \sum_{i=1}^{n} Y_i = m \right]$$
The Poisson Approximation for Balls into Bins

**Theorem**

Let $f(x_1, \ldots, x_n)$ be any non-negative function,

$$E[f(X_1, \ldots, X_n)] \leq e\sqrt{m}E[f(Y_1, \ldots, Y_m)]$$

$$E[f(Y_1, \ldots, Y_m)] \geq E[f(Y_1, \ldots, Y_m) \mid \sum Y_i = m] \text{Prob}[\sum Y_i = m]$$

$$= E[f(X_1, \ldots, X_m)] \frac{e^{-m}m^m}{m!}$$

$$> E[f(X_1, \ldots, X_m)]/(e\sqrt{m})$$

**Corollary**

An event taking place with probability $p$ in the Poisson takes place with probability $\leq e\sqrt{mp}$ in the exact case.
6. The Maximum Load

Throw $m = n$ balls into $n$ boxes

- What’s the typical order of the maximum load? Intuitively,
  - $\text{Prob}[\text{max load is too large}]$ is small
  - $\text{Prob}[\text{max load is too small}]$ is small

- Ideally, there’s some $f(n)$ s.t.
  - $\text{Prob}[\text{max load} = \Omega(f(n))] = o(1)$
  - $\text{Prob}[\text{max load} = O(f(n))] = o(1)$

- It’s quite amazing that such “threshold function” $f(n)$ exists

$$f(n) = \frac{\ln n}{\ln \ln n}$$
Upper Threshold for Maximum Load

- First trial

\[
\text{Prob}[X_i \geq c] = \sum_{k=c}^{m} \binom{m}{k} \left( \frac{1}{n} \right)^k \left( 1 - \frac{1}{n} \right)^{m-k} = \ldots \text{messy}
\]

- Second trial: break it down and use union bound!

- For any set \( S \) of \( c \) balls, let \( A_S \) be the event bin \( i \) contains \( S \)

- One union bound application

\[
\text{Prob}[X_i \geq c] = \text{Prob}[A_S \text{ occurs for some } S] \leq \binom{n}{c} \left( \frac{1}{n} \right)^c
\]

- Another union bound application

\[
\text{Prob}[\text{Some bin has } \geq c \text{ balls}] \leq n \binom{n}{c} \left( \frac{1}{n} \right)^c
\]

- \( \text{Prob}[\text{Some bin has } \geq c \ln n / \ln \ln n \text{ balls}] \leq 1/n \), when \( n \) large
Lower Threshold for Maximum Load

\[ \text{Prob}[X_i < c, \forall i] \leq e^{\sqrt{n}} (\text{Prob}[Y_i \leq c - 1])^n \]
\[ = e^{\sqrt{n}} \left( \sum_{k=0}^{c-1} \frac{e^{-1}(1)^k}{k!} \right)^n \]
\[ < e^{\sqrt{n}} \left( 1 - \frac{1}{e \cdot c!} \right)^n \]
\[ < e^{\sqrt{n}e^{-\frac{n}{e \cdot c!}}} \]
\[ \leq 1/n \]

when \( c = \ln n / \ln \ln n \) and \( n \) sufficiently large.
A Real Problem: Distributed Web Caching

- Web proxies cache web-pages for fast delivery, network load reduction, etc.
- When a new URL is requested, a proxy needs to know if it or another proxy has a cached copy
- Periodically, proxies exchange list of (thousands of) URLs they have cached
- Reducing periodic traffic requires reducing sizes of these exchanged lists

Question
How would you solve this problem?
First Solution: Hash Function

- Say, a proxy has \( m \) URLs \( x_1, \ldots, x_m \) in its cache
- Brute-force solution requires hundreds of \( KB \)
- To reduce space, use a hash function \( h : \{\text{URL}\} \rightarrow [n] \)
- Assume each URL mapped to \( i \in [n] \) with probability \( 1/n \) (very strong assumption)

Two ways to transmit

- \( n \)-bit string, bits \( h(x_i) \) are set to 1
- \( m \log_2 n \)-bit string, \( \log_2 n \) bits for each \( h(x_i) \)

Main Question

Choose \( n \) as small as possible so that, if \( x \) is a URL not on the list,

\[
\text{Prob}[h(x) = h(x_i) \text{ for some } i] \leq \epsilon
\]
\[
\text{Prob}[h(x) = h(x_i), \text{ for some } i] \leq m \text{ Prob}[h(x) = h(x_1)] \\
= mn \text{ Prob}[h(x) = h(x_1) = 1] \\
= mn \left(\frac{1}{n}\right)^2 \\
\leq \epsilon
\]

as long as \( n \geq m/\epsilon \).

Number of bits used is either \( n = m/\epsilon \) or \( m \log n = m(\log m + \log(1/\epsilon)) \).
Bloom Filter (Bloom, 1970) has been “blooming” in databases, networking, etc.

Idea:
- choose $k$ random hash functions $h_1, \ldots, h_k : \{\text{URL}\} \to [n]$.
- transmit $n$-bit string: all bits $h_j(x_i)$ are set to 1 ($j \in [k], i \in [m]$).
- querying for $x$: return YES if bits $h_j(x)$ are 1 for all $j \in [k]$.

Want:

\[
\text{Prob}[x \text{ is a false positive}] \leq \epsilon
\]

Or,

\[
\text{Prob[all } k \text{ balls thrown into non-empty bins]} \leq \epsilon
\]
Bloom Filter: Preliminary Analysis

Let $Y = \# \text{ empty bins}$

$$E[Y] = \sum_{i=1}^{n} \text{Prob}[X_i = 0] = n \left(1 - \frac{1}{n}\right)^{mk} = np \approx ne^{-\frac{mk}{n}} = np_a$$

Probability that all $k$ balls thrown into non-empty bins is

$$\left(1 - \frac{Y}{n}\right)^k \approx (1 - p)^k \approx (1 - p_a)^k$$

- First $\approx$ good if $Y$ is highly concentrated
- Second $\approx$ good for large $n$

Minimizing $(1 - p_a)^k$ leads to $k = n \ln 2 / m$. With this $k$,

$$\text{Prob[false positive]} = (1 - p_a)^k = \left(\frac{1}{2}\right)^{n \ln 2 / m} \leq \epsilon$$

as long as $n \geq m \log(1/\epsilon) / \ln 2$
Is $Y$ Highly Concentrated?

$$\text{Prob}\left[\frac{Y}{n} \text{ is } \delta\text{-close to } p\right] = 1 - \text{Prob}[|Y - np| > \delta n]$$

- Let $Z_i$ indicates if bin $i$ is empty, then $Y = \sum Z_i$
- The event $|\sum Z_i - np| > \delta n$ is in the exact case, the $Z_i$ are not independent
- In the Poisson, $\text{Prob}[Y_i = 0] = \frac{e^{-mk/n}(mk/n)^0}{0!} = p_a$
- With Chernoff’s help, we get

$$\text{Prob}[|Y - np| > \delta n] \leq e^{\sqrt{m}} \cdot 2e^{-(np_a)(\delta/p_a)^2/3} = \frac{\sqrt{m}}{e^{2\delta n/3 - 1}}$$

Exponentially small! Thus, $Y$ is highly concentrated.
What is the least number of bits needed if
- No false negative is allowed
- False positive probability is at most $\epsilon$

Say, the universe (of all URLs) has $U$ elements
Each subset of size $m$ is represented by a string of length $n$
Each string of length $n$ can only represent at most \( \binom{m+\epsilon(U-m)}{m} \) subsets

\[
\binom{U}{m} \leq 2^n \binom{m+\epsilon(U-m)}{m}
\]

Hence,
\[
n \geq m \log_2 \left( \frac{1}{\epsilon} \right)
\]