CSE 521: Design and Analysis of Algorithms I **Fall 2025**

Lecture 1: Contraction Algorithm

Lecturer: Shayan Oveis Gharan *09-24-2025* *Scribe:*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications.*

1.1 Introduction to Randomized Algorithms

Deterministic algorithms take input and produce output. In Randomized Algorithms, in addition to input algorithms take a source of random bits and makes random choices during execution - which leads behavior to vary even on a fixed input. For many problems a randomized algorithm is the simplest or fastest or both.

Let $p > 0$ be the probability that a (randomized) algorithm generates the correct (optimal) solution. It turns out that, even if p is much smaller than 1 we can obtain an algorithm with that succeeds with high probability just by executing the original algorithm independently many times. In particular, if we run the original algorithm k times, the probability that at least one copy succeeds is at least $1-(1-p)^k$. For small values of p one can always approximate $1-p$ with e^{-p}, and during this course we always use such an approximation. It follows that for $k = 100/p$, at least one copy finds the correct (optimal) solution with probability at least

$$1 - e^{-pk} = 1 - e^{-100}.$$

In other words, even if we have an algorithm with a small probability of success we can boost the success probability to a number very close to 1.

In this lecture, we describe a randomized algorithm for the minimum cut problem. Let us start with the definition of minimum cut problem. Let $G = (V, E)$ be a graph with $n = |V|$ vertices. Let

$$E(S, \overline{S}) := \{(u, v) : u \in S, v \notin S\},$$

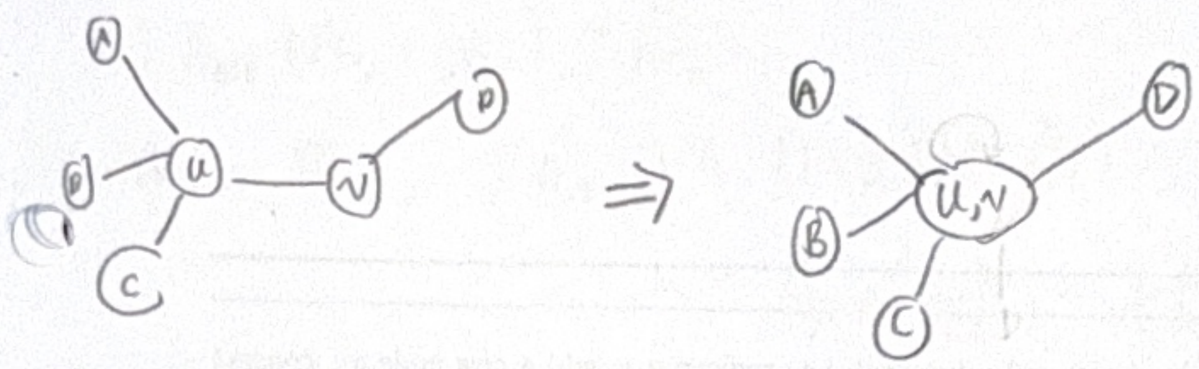
be the set of edges that have exactly one endpoint in S. In the minimum cut problem, we want to partition the into two subsets which are joined together with *minimum* number of edges, i.e.,

$$\min_{\emptyset \subset S \subset V} |E(S, \overline{S})|.$$

1.2 Karger's Algorithm

In this lecture we will discuss Karger's [**Kar93**] and Karger-Stein's [**KS93**] algorithm for the minimum cut problem. We will show that the former finds the minimum cut in time $O(n^4)$ and the latter finds it in time $O(n^2)$ with high probability.

Before describing these algorithms, let us define a contraction procedure. Contraction of an edge (u, v) in G, merges the endpoints u and v to create a new (super) node uv. This reduces the total number of nodes in the graph by 1. All other edges which were previously attached to u or v are attached to the new (super) node uv. Note that this might lead to multiple parallel edges; in particular, if a node z has a edges to u and b edges to v, after contraction it will have $a + b$ edges to uv.



let $E(S^*, \overline{S^*})$ be the min-cut

then $\mathbb{P}[e \in E(S^*, \overline{S^*})] \leq \frac{2}{n}$

$\hookrightarrow = \frac{|E(S^*, \overline{S^*})|}{|E|}$ Let k be $|E(S^*, \overline{S^*})|$ notice $|E| \geq \frac{kn}{2}$ or else min-cut be cut 1node off

$\leq \frac{2}{n}$

Karger's Algorithm

- contract until there are only 2 nodes left, then $|E(S^*, \overline{S^*})|$ will be the # edges between them
- boost probability

Give when contracting edges, size of min-cut does not decrease because new min-cut is still a subset of edges that form a cut in the original graph

- For $G=(V,E)$ with $|V|=n$ then $\uparrow$ returns $(S^*, \overline{S^*})$ with $P \leq \frac{2}{n(n-1)}$

Algorithm 1 Karger's Algorithm

for $i = 1 \to n-1$ **do**

Choose a uniformly random edge (u, v) and contract it, i.e., remove u, v, add a new node uv, connect all edges that go to u or v to the new node uv, also remove all loops.

end for

Return the number edges between the final two super-nodes as the size of the min-cut.

Proof. Let, A_i be the event that the edge picked in step i of the loop is not in $E(S^*, \overline{S^*})$. Observe that the algorithm succeeds in finding $(S^*, \overline{S^*})$ if $A_1, A_2, \ldots, A_{n-2}$ occur, i.e., if we never contract an edge of $E(S^*, \overline{S^*})$. So, we just need to lower bound $\mathbb{P}[A_1, A_2, \ldots, A_{n-2}]$. By Bayes rule we have,

$$\mathbb{P}[A_1, \ldots, A_{n-2}] = \mathbb{P}[A_1]\,\mathbb{P}[A_2|A_1]\,\mathbb{P}[A_3|A_1, A_2] \ldots \mathbb{P}[A_{n-2}|A_1, A_2, \ldots, A_{n-3}].$$

Now, by Claim 1.1 and Fact 1.2, for all i,

$$\mathbb{P}\left[\overline{A_{i+1}}|A_1, \ldots, A_i\right] \leq \frac{2}{n-i}.$$

This is because, at the $i+1$ step, there are only $n-i$ vertices in the graph. By Fact 1.2, the size of the min cut is at least k. So, the degree of each of these $n-i$ vertices is at least k, and the graph has at least $(n-i)k/2$ edges. Now, the above inequality follows by a similar reasoning as in Claim 1.1.

Therefore,

$$\begin{aligned}\mathbb{P}[A_1, \ldots, A_{n-2}] &= \mathbb{P}[A_1] \geq \left(1 - \frac{2}{n}\right)\left(1 - \frac{2}{n-1}\right) \cdots \left(1 - \frac{2}{3}\right) \\ &= \frac{n-2}{n} \cdot \frac{n-3}{n-1} \cdot \frac{n-4}{n-2} \cdots \frac{1}{3} \\ &= \frac{2}{n(n-1)} = \frac{1}{\binom{n}{2}}.\end{aligned}$$

□

The following corollary is immediate from the above theorem.

Corollary 1.4. *Any graph has at most $\binom{n}{2}$ min-cuts.*

Proof. Suppose there is a graph G that has more than $\binom{n}{2}$ min-cuts. Then, for one of those cuts, say $(S, \overline{S})$ the probability that Algorithm 1, finds $(S, \overline{S})$ is less than $1/\binom{n}{2}$. But, this is a contradiction. □

In fact, the above bound is tight. A cycle of length n has exactly $\binom{n}{2}$ min-cuts, one corresponding to each possible way to delete two edges of the cycle.

Runtime and Boosting Probability of Success. It is not hard to see that one can run Algorithm 1 in time $O(n^2)$, but as we proved above, the success probability of each execution is just $O(1/n^2)$. Using the boosting idea, we can boost the probability of success to $1 - 1/n$ by running $O(n^2 \log n)$ independent copies of the above algorithm and returning the best cut that any of the copies find. In the next section we describe a much faster algorithm due to Karger and Stein.

$P[\bar{A}_{i+1} | A_1, \cdots, A_i] \leq \frac{2}{n-i}$

Then $P[A_1, \cdots, A_{n-2}] \in (1-\frac{2}{n})(1-\frac{2}{n-1}) \cdots (1-\frac{2}{3})$

$= \frac{n-2}{n}\left(\frac{n-3}{n-1}\right) \cdots \left(\frac{1}{3}\right)$

$= \frac{2}{n(n-1)} \leq \frac{2}{n^2}$

$= \frac{1}{\binom{n}{2}}$

Corollary: at most $\binom{n}{2}$ min-cuts

run time: $O(n^2)$

$1 - (1 - 1/\binom{n}{2})^K \geq 1 - 1/n$

$e^{-K/\binom{n}{2}} \leq 1/n$ $\quad K = \frac{n-1}{2} \in O(n)$

$-K/\binom{n}{2} \geq \log(1/n)$

$K \leq \binom{n}{2} \log(1/n)$

$K \geq \binom{n}{2} \log(n) \in O(n^2 \log n)$ times

$\Rightarrow$ Total runtime $\in O(n^4 \log n)$

1.3 Karger-Stein Algorithm

Recall that Karger's algorithm only fails if it contracts an edge of the min-cut. Also, note that the probability that we contract an edge of the min-cut at the beginning is only $2/n$ while towards the end of the algorithm this probability goes up to a constant. In particular, in the very last step there is a probability of $1/3$ that we contract an edge of the min-cut.

The idea of Karger-Stein algorithm is to run multiple independent copies of the Karger's algorithm when the size of the Graph gets smaller. This idea kind of resembles the idea fault tolerant systems where one stores multiple copies of the data to decrease the probability of failure.

Let us describe Karger-Stein's algorithm.

Algorithm 2 Min-cut($G = (V, E)$)

Let $n = |V|$. If $n = 2$ return the unique cut separating the two nodes of G.
for $i = 1 \to n - n/\sqrt{2}$ **do**
  Choose a uniformly random edge and contract it.
end for
Let G' be the contracted graph. Call Min-cut(G') twice and return the best cut that any of these two copies find.

Let us divide the work of the algorithm into $O(\log n)$ phases; in the first phase the algorithm goes from n to $n/\sqrt{2}$, in the second phase it goes from $n/\sqrt{2}$ to $n/\sqrt{2}^2$ and so on. The number of copies are chosen such that the algorithm spends exactly the same amount of work $O(n^2)$ in each phase.

Let $T(n)$ be the time it takes to compute the min-cut of a graph of size n. Then,

$$T(n) = O(n^2) + 2T(n/\sqrt{2})$$

We can use the master theorem to solve the above recurrence. But, usually it is easier to open it up a couple of times and see the pattern. We can write

$$\begin{aligned} T(n) &= O(n^2) + 2O((n/2^{1/2})^2) + 4O((n/2^1)^2) + \cdots \\ &= O(n^2) + O(n^2) + O(n^2) + \cdots = O(n^2 \log n) \end{aligned}$$

It remains the calculate the probability of success. In the next theorem we show that the algorithm succeeds with probability $\Omega(1/\log n)$. This shows that to boost the probability of success to $1 - 1/n$ it is enough to run $\log^2 n$ independent copies of the above algorithm. Therefore, the algorithm finds the min-cut with probability $1 - 1/n$ in time $O(n^2 \log^3 n)$.

Theorem 1.5. *For any min-cut $(S^*, \overline{S^*})$, Algorithm 2 finds this cut with probability at least $1/2 \log n$.*

Proof. Suppose we call Min-cut function on G with n vertices. By an analysis similar to Theorem 1.3, the probability that we do not contract any edge of $(S^*, \overline{S^*})$ in the $n - n/\sqrt{2}$ steps of the loop is at least

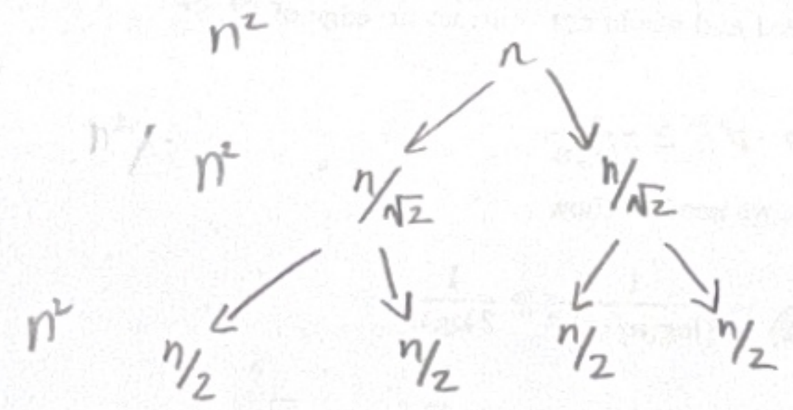
$$\frac{n-2}{n} \cdot \frac{n-3}{n-1} \cdots \frac{n/\sqrt{2}-2}{n/\sqrt{2}} \approx \frac{(n/\sqrt{2})^2}{n^2} = 1/2.$$

The algorithm succeeds in finding the min-cut $(S^*, \overline{S^*})$ if Min-cut(G) does not contract an edge of $(S^*, \overline{S^*})$ in its for loop and at least one of the two copies succeeds in finding the cut. We prove inductively that for

Notice probability of failure goes up as $i \to n-2$

$G=(V,E)$
for $i = 1 \to n(1-1/\sqrt{2})$ do
  contract
return better(G', G')

Divide the work into $O(\log n)$ phase $n \to n/\sqrt{2}$ each stage



$T(n) = O(n^2) + 2T(n/\sqrt{2})$

run time $= n^2 \log n$

At each stage, $P(\text{success}) = \left(1-\frac{2}{n}\right)\left(1-\frac{2}{n-1}\right)\cdots\left(1-\frac{2\sqrt{2}}{n}\right)$

$= \frac{n-2}{n}\left(\frac{n-3}{n-1}\right)\cdots\left(\frac{n/\sqrt{2}-1}{n/\sqrt{2}+1}\right)\left(\frac{n/\sqrt{2}-2}{n/\sqrt{2}}\right)$

$= \frac{(n/\sqrt{2}-1)(n/\sqrt{2}-2)}{n(n-1)} \approx 1/2\left(\frac{n^2/2 - \frac{3}{\sqrt{2}}n + 2}{n^2-n}\right)$

Then at each stage, $P(1 \text{ of child succeed}) \,\&\, I \text{ succeed}$
$\Downarrow \qquad \Downarrow$
$1/2$

any graph with n vertices the probability of success is at least $1/2\log n$. Assuming by induction hypothesis, that the algorithm succeeds on G' with probability at least $p \geq \frac{1}{2\log(n/\sqrt{2})}$. Here the logs are all in base 2.

The probability that the algorithm succeeds in G is at least

$$\frac{1}{2}(1-(1-p)^2)) = \frac{1}{2}(2p-p^2) = p - p^2/2$$

Note that $(1-p)^2$ is the probability that both of the two independent copies fail in finding the cut. So, $1-(1-p)^2$ is the probability that at least one of them succeed. The $1/2$ ratio is the probability that in the first $n - n/\sqrt{2}$ iterations of the loop we succeed and we do not contract an edge of $S^*, \overline{S^*}$.

So, it is enough to show that

$$p - p^2/2 \geq \frac{1}{2\log n}.$$

In the worst case we have $p = \frac{1}{2\log(n/\sqrt{2})}$. So, we need to show

$$\frac{1}{2\log(n/\sqrt{2})} - \frac{1}{8\log(n/\sqrt{2})^2} \geq \frac{1}{2\log n}.$$

Equivalently, it is enough to show

$$\frac{1}{\log(n/\sqrt{2})} - \frac{1}{\log n} \geq \frac{1}{4\log(n/\sqrt{2})^2}$$

The latter holds because,

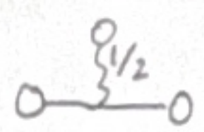
$$\frac{1}{\log(n/\sqrt{2})} - \frac{1}{\log n} = \frac{\log n - \log(n/\sqrt{2})}{\log n \log(n/\sqrt{2})} = \frac{1/2}{\log n \log(n/\sqrt{2})} \geq \frac{1}{4\log(n/\sqrt{2})^2}.$$

□

References

[Kar93] D. R. KARGER., "Global min-cuts in RNC, and other ramifications of a simple min-out algorithm," in *SODA*. , 1993, pp. 21–30.

[KS93] D. R. KARGER AND C. STEIN, "An $O(n^2)$ algorithm for minimum cuts", in *STOC*., (1993)



$\mathbb{P}(1/2\log n)$

$\frac{1}{2}\left[1-(1-\frac{1}{2\log n})^2\right] = p - \frac{p^2}{2} \geq \frac{1}{2\log n}$

CSE 521: Design and Analysis of Algorithms I **Fall 2025**

Lecture 2: Concentration Bounds

Lecturer: Shayan Oveis Gharan *09/29/2025* *Scribe:*

Suppose there is an unknown distribution, D, and we want to estimate the mean. A possible suggestion is to draw independent samples

$$x_1, x_2, \ldots, x_n$$

from D and return the empirical average,

$$\frac{1}{n}\sum_{i=1}^{n} x_i.$$

Laws of large number say that as n goes to infinity the empirical average converges to the mean. The question we want to address in this lecture is "how large should n be" in order to get a an ϵ-additive approximation of the true expectation? As a real world application, we can use this idea to estimate the people opinion in polling by asking only a few of the voters randomly.

We start this lecture by a simple example: Suppose that the average GPA in CSE 521 is 3.0 / 4.0. At most, what fraction of the students have received at least a 3.5? It turns out in the worst case 1/7-th fraction have received 0.0 and the rest, i.e., 6/7-th fraction have received 3.5. In other words, the worst case is when everybody who has received below 3.5 indeed got 0 and all of those who got more than 3.5 indeed receive nothing more than 3.5. We can justify this claim using Markov's inequality.

2.1 Markov's Inequality

Theorem 2.1 (Markov's Inequality). *Let $X \geq 0$ be a random variable. Then for all k,*

$$\mathbb{P}[X \geq k \cdot \mathbb{E}[X]] \leq \frac{1}{k}$$

equivalently:

$$\mathbb{P}[X \geq k] \leq \frac{\mathbb{E}[x]}{k}.$$

So, in our class average GPA example, X denotes the GPA of a random student, $\mathbb{E}[X] = 3$ and $k = 7/6$. The inequality says at most 6/7 fraction of the students received at least 3.5 or at least 1/7 receive less than 3.5.

Proof. The proof is a simple one line argument,

$$\mathbb{E}[X] = \sum_i \mathbb{P}[X = i] \geq \sum_{i \geq k} i \cdot \mathbb{P}[X = i] \geq \sum_{i \geq k} k \cdot \mathbb{P}[X = i] = k \cdot \mathbb{P}[X \geq k]$$

So, $\mathbb{P}[X \geq k] \leq \mathbb{E}[X]/k$ as desired. □

Markov

$X \geq 0$ be r.v. $\forall k$

$\mathbb{P}[X \geq k \cdot \mathbb{E}X] \leq 1/k \quad \Rightarrow \quad \mathbb{P}[X \geq k] \leq \frac{\mathbb{E}[X]}{k}$

proof

$$\mathbb{E}X = \sum_i \mathbb{P}[X=i]\, i \geq \sum_{i \geq k} \mathbb{P}[X=i]\, k = \mathbb{P}[X \geq k] \cdot k$$

Then $\mathbb{P}[X \geq k] \leq \frac{\mathbb{E}X}{k}$

bound if tight if we only know $\mathbb{E}X$

Observe that in the above proof is tight, i.e., all inequalities are equalities, if the distribution of X has only two points mass,

$$X = \begin{cases} 0 & \text{w.p. } 1 - 1/k \\ k + \epsilon & \text{w.p. } 1/k \end{cases}$$

In other words, this example shows that if $\mathbb{E}[X]$ is the only information that we have about X, then Markov's inequality is the best bound we can prove on deviations from the expectation of X.

2.1.1 Applications of Markov's Inequality: Fixed points of permutations

Let $[n] := \{1, \ldots, n\}$. A permutation, $\sigma : [n] \to^{1-1}_{\text{onto}} [n]$, is a bijection between $[n]$ and $[n]$. Suppose we a choose a uniformly random permutation σ. What is the probability that for two i, j, $\sigma_i = i$ and $\sigma_j = j$, i.e., that the permutation has two fixed points?

Let $X_i = \mathbb{I}[\sigma_i = i]$. Let $X = \sum X_i$. Note that X is exactly equal to the number of fixed points of σ. So we want to upper bound $\mathbb{P}[X \geq 2]$. We are going to use Markov's inequality, but first we need to calculate $\mathbb{E}[X]$.

$$\begin{aligned} \mathbb{E}[X] &= \mathbb{E}\left[\sum X_i\right] \\ &= \sum \mathbb{E}[X_i] \qquad \text{(by linearity of expection, not proven here)} \\ &= \sum \mathbb{P}[X_i = 1] \qquad \text{(expectation of an indicator)} \\ &= \sum_i \frac{1}{n} \\ &= 1 \end{aligned}$$

So by Markov Inequality,

$$\mathbb{P}[X \geq 2] \leq \frac{1}{2}.$$

2.2 Chebyshev's Inequality

Markov's Inequality is the best bound you can have if all you know is the expectation. In its worst case, the probability is very spread out. The Chebyshev Inequality lets you say more if you know the distribution's variance.

Definition 2.2 (Variance). *The variance of a random variable X is defined as*

$$\text{Var}(X) = \mathbb{E}\left[(X - \mathbb{E}X)^2\right]$$

Let us prove an identity on $\text{Var}(X)$.

$$\begin{aligned} \text{Var}(X) &= \mathbb{E}\left[(X - \mathbb{E}X)^2\right] \\ &= \mathbb{E}\left[X^2 - 2X\mathbb{E}[x] + (\mathbb{E}[X])^2\right] \\ &= \mathbb{E}\left[X^2\right] + (\mathbb{E}[X])^2 - 2(\mathbb{E}[X])^2 \\ &= \mathbb{E}\left[X^2\right] - \mathbb{E}[X]^2 \end{aligned}$$

Chebyshev's Inequality

$$\mathbb{P}[|X - \mathbb{E}X| > K] \leq \frac{\text{Var}X}{K^2}$$

where we used linearity of expectation. Note that for any number X, $(X - \mathbb{E}X)^2 \geq 0$. Therefore, for any random variable X, $\mathrm{Var}(X) \geq 0$. So, by above identity we always have

$$\mathbb{E}[X^2] \geq \mathbb{E}[X]^2,$$

i.e., the 2nd moment is at least the 1st moment squared.

Theorem 2.3 (Chebyshev's Inequality). *For any random variable X,*

$$\mathbb{P}[|X - \mathbb{E}X| > \epsilon] < \frac{\mathrm{Var}(X)}{\epsilon^2}$$

or equivalently

$$\mathbb{P}[|X - \mathbb{E}[X]| > k\sigma] \leq \frac{1}{k^2}$$

where $\sigma = \sqrt{\mathrm{Var}(X)}$ is the standard deviation of X.

The second inequality in theorem can be read that any random variable is within 3 standard deviation of the expectation with probability 90%. It turns out that Chebyshev's inequality is just Markov's inequality applied to the variance R.V., $Y = (X - \mathbb{E}[X])^2$.

Proof. Let $Y := (X - \mathbb{E}X)^2$ be a nonnegative random variable. So, by Markov's inequality,

$$\mathbb{P}[Y \geq \epsilon^2] \leq \frac{\mathbb{E}[Y]}{\epsilon^2}$$

In other words,

$$\mathbb{P}[|X - \mathbb{E}[X]|^2 \geq \epsilon^2] \leq \frac{\mathrm{Var}(X)}{\epsilon^2}.$$

Taking square root of the both sides of the inequality gives,

$$\mathbb{P}[|X - \mathbb{E}[X]| \geq \epsilon] \leq \frac{\mathrm{Var}(X)}{\epsilon^2}$$

as desired □

2.2.1 Polling

In this section we use Chebyshev's inequality to answer the question that we raised at the beginning of this lecture. Suppose there is an unknown distribution D with mean μ and we want to estimate μ using independent samples of D,

$$X_1, X_2, \ldots, X_n$$

First, observe that by linearity of expectation,

$$\mathbb{E}\left[\frac{1}{n}\sum_i X_i\right] = \mu.$$

So, we want to use Chebyshev's inequality to upper bound,

$$\mathbb{P}\left[\left|\frac{X_1 + X_2 + \cdots + X_n}{n} - \mu\right| \geq \epsilon\right]$$

Chebyshev $\mathbb{P}[|X-\mathbb{E}X|>\varepsilon] < \frac{\mathrm{Var}X}{\varepsilon^2}$

$\equiv \mathbb{P}[(X-\mathbb{E}X)^2 > \varepsilon]$

$\mathbb{E}[(X-\mathbb{E}X)^2] = \mathrm{Var}X$

$\Rightarrow \mathbb{P}[(X-\mathbb{E}X)^2 > \varepsilon] \leq \frac{\mathrm{Var}X}{\varepsilon^2}$ by Markov

To use Chebyshev's inequality, first we need to calculate the variance. Let $X = \frac{X_1+\cdots+X_n}{n}$ be the empirical average. We use the following lemma to bound the variance of X.

We say a set of random variables $X_1, X_2, \ldots, X_n$ are *pairwise independent* if for all $1 \le i, j \le n$

$$\mathbb{E}[X_i X_j] = \mathbb{E}[X_i]\,\mathbb{E}[X_j].$$

Lemma 2.4. *For any set of pairwise independent random variables $X_1, \ldots, X_n$*

$$\mathrm{Var}(X_1 + \cdots + X_n) = \mathrm{Var}\,X_1 + \cdots + \mathrm{Var}\,X_n$$

Proof. We can write,

$$\begin{aligned}
\mathrm{Var}(X_1 + \cdots + X_n) &= \mathbb{E}\left[(X_1 + \cdots + X_n)^2\right] - (\mathbb{E}X_1 + \mathbb{E}X_2 + \cdots + \mathbb{E}X_n)^2 \\
&= \mathbb{E}\left[\sum_{i,j} X_i X_j\right] - \sum_{i,j} \mathbb{E}[X_i]\,\mathbb{E}[X_j] \\
&= \sum \mathbb{E}\left[X_i^2\right] - (\mathbb{E}[X_i])^2 \\
&= \sum_{i=1}^{n} \mathrm{Var}(X_i).
\end{aligned}$$

In the second to last equality we used pairwise independence. □

Let's go back to the polling example; recall $X_1, \ldots, X_n$ are independent samples of D, so they are pairwise independent, and by the above lemma,

$$\mathrm{Var}(X) = \mathrm{Var}\left(\frac{X_1 + \cdots + X_n}{n}\right) = \frac{1}{n^2}\mathrm{Var}(X_1 + \cdots + X_n) = \frac{1}{n^2}(\mathrm{Var}(X_1) + \cdots + \mathrm{Var}(X_n)) = \frac{\mathrm{Var}(D)}{n}$$

Therefore, by Chebyshev's inequality,

$$\mathbb{P}[|X - \mu| \ge \epsilon] \le \frac{\mathrm{Var}(D)}{n\epsilon^2} \tag{2.1}$$

Now, let's continue on the polling example, suppose for all i,

$$X_i = \begin{cases} 1 & \text{w.p. } p \\ 0 & \text{otherwise} \end{cases},$$

i.e., p fraction of the population would vote yes on the election, and we want to estimate p within ϵ additive error. So, it all we need to do is to upper bound the variance of X_i, First, we calculate the second moment, for all i,

$$\mathbb{E}\left[X_i^2\right] = 1^2 \cdot p + 0^2 \cdot (1-p) = p.$$

Therefore,

$$\mathrm{Var}(X_i) = \mathbb{E}\left[X_i^2\right] - \mathbb{E}[X_i]^2 = p - p^2 = p(1-p) \le \frac{1}{4}.$$

Therefore, by (2.1)

$$\mathbb{P}\left[\left|\frac{\sum_i X_i}{n} - p\right| \ge \epsilon\right] \le \frac{1}{4n\epsilon^2}$$

$\mathrm{Var}(X_1 + \cdots + X_n) = \sum \mathrm{Var}\,X_i$ if X is pairwise independent

Suppose we choose 10,000 individuals from the population randomly and we calculate the empirical mean; by above inequality with probability 15/16 our estimate is within 2% of the true mean. Note that the importance of this inequality is the the size of the sample is independent of the size of the population. In general if we want to obtain an ϵ-additive error with probability $1-\delta$ we need $O(1/\delta\epsilon^2)$ many samples.

Note that the above analysis can easily be extended to the case where X_i's are not necessarily Bernoulli. In particular, suppose D is distributed on an interval $[a,b]$ where D can take any real number in this interval. It fallows that the variance of D is at most $(b-a)^2$. This is because the different of any two numbers in the support of D is at most $b-a$. Therefore, following the same analysis if we have n samples $X_1,\ldots,X_n$ of such a D then

$$\mathbb{P}\left[\left|\frac{\sum_i X_i}{n}-\mu\right|\geq\epsilon\right]\leq\frac{(b-a)^2}{n\epsilon^2}.$$

where μ is the mean of D. So, to get an ϵ-additive error with probability at least $1-\delta$ it is enough to have $n\geq\frac{(b-a)^2}{\epsilon^2\delta}$ many samples.

Next lecture we will see a stronger concentration bounds, a.k.a., Chernoff bounds. We see that for the same polling example it is enough to use $O(\frac{1}{\epsilon^2}\log\frac{1}{\delta})$ samples to obtain an ϵ-additive approximation of the mean with probability $1-\delta$.

2.3 Birthday Paradox

The Birthday paradox is a well-known problem in probability theory which finds the probability that some pairs of individuals in a set of n randomly chosen group of people will have the same birthday. It assumes that each day of the 365 days of a year is equally probable for a birthday. It can be easily noted that the probability reaches 100% when the number of people reaches 366, since there are 365 days in a year.

Let $X_1, X_2, \ldots X_n$ be n independent and identically distributed (i.i.d.) random variables, in the range $\{1,2,\ldots N\}$ where X_i denotes the birthday of the person i. We say there is a *collision* if for some $1\leq i,j\leq n$, we have $X_i=X_j$. Otherwise, (if for all i,j, $X_i\neq X_j$) we say there is no collision. We prove the following two claims:

Lemma 2.5. *If $n\leq\sqrt{N}$, then,*

$$\mathbb{P}\left[\textit{no collision}\right]\geq\frac{1}{2}.$$

Lemma 2.6. *If $n\geq c\sqrt{N}$, then,*

$$\mathbb{P}\left[\textit{collision}\right]\geq 1-\frac{2}{c^2}.$$

Let $Y_{i,j}=\mathbb{I}[X_i=X_j]$ be the random variable indicating that $X_i=X_j$. Let $Y=\sum_{i,j}Y_{i,j}$. Note that by definition Y is always a nonnegative integer.

We start by proving Lemma 2.5. By definition of Y, it is enough to show $\mathbb{P}[Y=0]\geq 1/2$; equivalently, it is enough to show $\mathbb{P}[Y\geq 1]\leq 1/2$. The latter inequality is very suitable for an application of Markov's inequality. To show the latter it is enough to show $\mathbb{E}[Y]\leq 1/2$. By linearity of expectation,

$$\mathbb{E}[Y]=\mathbb{E}\left[\sum_{i,j}Y_{i,j}\right]=\sum_{i,j}\mathbb{E}[Y_{i,j}]=\sum_{i,j}\mathbb{P}[Y_{i,j}=1]=\frac{\binom{n}{2}}{N} \tag{2.2}$$

The last equality uses the fact that for all i,j, $\mathbb{P}[Y_{i,j}=1]=\frac{1}{N}$.

Birthday Paradox

$X_{ij}=\mathbb{1}[i \text{ and } j \text{ share birthday}]$

$X=\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}X_{ij}$ $\quad \mathbb{E}X=\binom{n}{2}1/N=\binom{n}{2}/N$

Then $\mathbb{P}[X\geq 1]\leq \mathbb{E}X$ $\quad$ if $n<\sqrt{N}$ then $\mathbb{E}X<1/2$

$\mathbb{P}[X<1]\geq 1-\mathbb{E}X=1/2$

probability of no collision is at least $1/2$

So, by Markov's inequality,

$$\mathbb{P}[Y \geq 1] \leq \frac{\binom{n}{2}}{N} = \frac{n(n-1)}{2N} \leq \frac{1}{2} \Rightarrow \mathbb{P}[Y = 0] \geq \frac{1}{2} \tag{2.3}$$

which proves Lemma 2.5

Next, we prove Lemma 2.6. In this case, we want to lower bound $\mathbb{P}[Y \geq 1]$; or equivalently, upper bound $\mathbb{P}[Y = 0]$. Note that Markov inequality does not give any interesting bound in this case. In fact if the only information we have about Y is its expectation then Y could be 0 with probability $1 - \epsilon$ and $\mathbb{E}[Y]/\epsilon$ with probability ϵ. So, to prove the claim we upper bound the variance of Y and use Chebyshev's inequality.

First, observe that the random variables $Y_{i,j}$'s are pairwise independent, since $X_i = X_j$ does not convey any information about whether or not $X_i = X_k$ for some $k \neq j$. Also, note that $Y_{i,j}$'s are not three-way independent; in particular, if $Y_{i,j} = 1, Y_{j,k} = 1$ then $Y_{i,k} = 1$.

Therefore, by pairwise independence property of $Y_{i,j}$'s, we get

$$\text{Var}[Y] = \sum_{i,j} \text{Var}(Y_{i,j}) = \sum_{i,j} \mathbb{E}[Y_{i,j}^2] - (\mathbb{E}[Y_{i,j}])^2 = \sum_{i,j} \frac{1}{N} - \frac{1}{N^2} \leq \sum_{i,j} \frac{1}{N} = \frac{\binom{n}{2}}{N} \tag{2.4}$$

Observe that variance of Y is less than its expectation. So, $\sigma Y \leq \sqrt{\mathbb{E}Y}$. As we mentioned in the previous lecture, we expect that with high probability Y is within 3 standard deviation of its expectation. So, if $\mathbb{E}[Y] \gg 0$, we have $Y \geq 1$ with high probability.

Now, let's make this formal. Using Chebyshev's inequality with $\epsilon = \mathbb{E}[Y]$, we get

$$\mathbb{P}[|Y - \mathbb{E}[Y]| \geq \mathbb{E}[Y]] \leq \frac{\binom{n}{2}/N}{(\binom{n}{2}/N)^2} = \frac{N}{\binom{n}{2}} \approx \frac{2}{c^2} \tag{2.5}$$

Therefore,

$$\mathbb{P}[Y = 0] \leq \mathbb{P}[|Y - \mathbb{E}[Y]| \geq \mathbb{E}[Y]] \leq \frac{2}{c^2}. \tag{2.6}$$

This shows that $\mathbb{P}[Y \geq 1] \geq 1 - \frac{2}{c^2}$ as desired. This proves Lemma 2.6.

Lemma 2.6 if $n \geq c\sqrt{N}$ then

$\mathbb{P}[\text{collision}] \geq 1 - \frac{2}{c^2}$

Chebyshev $\mathbb{P}[|X - \mathbb{E}X| \geq k] \leq \frac{\text{Var}X}{k^2}$

Pf.

$\mathbb{E}X = \frac{\binom{n}{2}}{N}$

CSE 521: Design and Analysis of Algorithms I **Fall 2025**

Lecture 3: Concentration Bounds and Hashing

Lecturer: Shayan Oveis Gharan *10/01/2025* *Scribe:*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications.*

In the previous lecture, we learnt about some of the concentration bounds commonly used in probability theory. We are going to learn some more in this lecture.

3.1 Law of Large Numbers

The Law of Large Numbers (LLN) is a theorem which states that the average of the results obtained from a large number of independent trials of an experiment tends towards the expected value. Central limit theorems state that for an infintie sequence of random independent variables $X_1, X_2, \ldots$ with mean μ and a bounded variance σ^2,

$$\sqrt{n}\left(\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right) \to \mathcal{N}(0, \sigma^2). \tag{3.1}$$

as n goes to infinity. In this course, we are interested in quantitative forms of this convergence. We will study this in the form of *strong concentration bounds*, a.k.a., Chernoff bounds.

Recall that Chebyshev's inequality implies that for any random variable X,

$$\mathbb{P}\left[|X - \mathbb{E}[X]| \geq k\sigma\right] \leq \frac{1}{k^2} \tag{3.2}$$

Strong concentration bounds imply that if X is an average of independent random variables with standard deviation σ, and satisfy certain other properties, then

$$\mathbb{P}\left[|X - \mathbb{E}X| \geq k\sigma\right] \leq e^{-\Omega(k^2)}$$

In other words, they give exponentially improved bounds compared to Chebyshev's inequality. Note that to get this strong bound we want X to be an average of mutually independent random variables; so unlike Chebyshev's inequality pairwise independent is not enough.

3.2 Hoeffding's Inequality

In this part we discuss Hoeffding's inequality. We will see more concentration bounds in the assignments.

Let $X_1, X_2, \ldots X_n$ be n independent random variables such that for all i, $X_i \in [a_i, b_i]$, and let $X = \frac{X_1 + \cdots + X_n}{n}$. Then,

$$\mathbb{P}\left[|X - \mathbb{E}[X]| \geq \epsilon\right] \leq 2\exp\left(\frac{-2n^2\epsilon^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right) \tag{3.3}$$

Let us give another interpretation of this inequality. First, observe that

$$\mathrm{Var}[X] = \frac{1}{n^2}\sum_i \mathrm{Var}[X_i] \leq \frac{\sum_i (b_i - a_i)^2}{n^2}.$$

Why $\mathrm{Var}[X_i] \leq (b_i - a_i)^2$

~~$\mathrm{Var}(X_i) = \mathbb{E}[X^2] - \mathbb{E}[X]^2$~~

~~$= \sum_{i \neq j} \mathbb{P}[X=i]\,\mathbb{P}[X=j]\, ij$~~

Let $Y = \frac{X-a}{b-a}$

$\mathrm{Var}(Y) = \frac{1}{(b-a)^2}\mathrm{Var}(X)$

$Y \in [0,1]$. Then max variance is $1/4$ (or 1 to be simple)

thus $\mathrm{Var}(Y)(b-a)^2 = \mathrm{Var}(X)$

$\mathrm{Var}(X) \leq (b-a)^2$

The last inequality uses that $X_i \in [a_i, b_i]$ for all i. So, we can rewrite the above as

$$\mathbb{P}[|X - \mathbb{E}X| \geq \epsilon] \leq 2\exp(-2\epsilon^2/\operatorname{Var} X)$$

Next, we discuss several applications of Hoeffding's inequality.

3.2.1 Application 1: Polling

Let us continue the polling example that we discussed in the last lecture. Consider a set of n Bernoulli random variables $X_1, X_2, \ldots X_n$ where for all i, $X_i = 1$ w.p. p and $X_i = 0$ w.p. $1-p$. By Hoeffding's inequality,

$$\mathbb{P}\left[\left|\frac{\sum X_i}{n} - p\right| \geq \epsilon\right] \leq 2\exp(\frac{-2n^2\epsilon^2}{n}) = 2\exp(-2n\epsilon^2) \tag{3.4}$$

where we used that $a_i = 0, b_i = 1$.

So, if we want to estimate the probability p within an additive error ϵ with probability $1-\delta$ it is enough to let

$$n = \frac{\ln(2/\delta)}{2\epsilon^2}.$$

To give you a point of comparison, recall that in the last lecture we showed that using Chebyshev inequality, to estimate p with additive error of ϵ with probability at $1-\delta$ we need about $\frac{1}{\delta\epsilon^2}$. So, for example, if we want $1 - 2^{-100}$ probability of success Hoeffding inequality implies we only need $100/\epsilon^2$ many samples, whereas Chebyshev's inequality says we want $2^{100}/\epsilon^2$ many samples. You can see that Hoeffding's inequality implies a significantly smaller number of samples.

Let us give a second example: suppose we have $n = 100$ samples; we want to see for what value of ϵ we can have 99% probability of success? It follows that we get $\epsilon = \frac{1}{6}$. Now, suppose we increase the number of samples to $n = 10000$. How much can we decrease ϵ to get the same 99% probability of success? Observe that we can only let $\epsilon \approx \frac{1}{60}$. Thus, we can decrease ϵ only proportional to square-root of the number of samples.

Upshot: The failure probability decreases exponentially with respect to the number of samples whereas the confidence interval ϵ only decreases proportional to the square-root of the number of samples.

3.2.2 Application 2: Random walk

A random walk is a random process that describes a path consisting of a succession of random steps on some mathematical space. Let us consider the one-dimensional random walk model, in which a person starts at the position 0 (origin) of the integer number line and moves to his right (+1) or to his left (-1) at each step with equal probability. Fig. 3.1 shows the model.

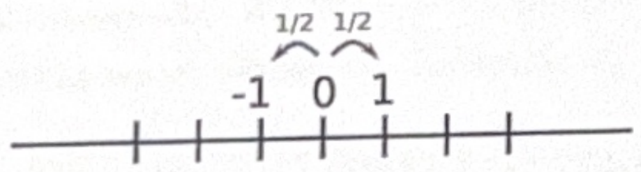


Figure 3.1: One-dimensional random walk model

To define the walk formally, let X_i be a random variable denoting whether the person moves to his right or left in the i^{th} step, so that $X_i = +1$ w.p. $\frac{1}{2}$ and $X_i = -1$ w.p. $\frac{1}{2}$. Thus $X = \sum_{i=1}^{n} X_i$ denotes the final position of the person after n steps. We want to estimate X.

$\mathbb{P}[|X| \geq \varepsilon] \leq 2\exp(-2\varepsilon^2/1)$ $\operatorname{Var} X_i = 1 - 0 = 1$

$= 2\exp(-2\varepsilon^2 n^2)$ $\operatorname{Var} X = 1/n^2$

First, observe that $\mathbb{E}[X] = \sum_{i=1}^{n} \mathbb{E}[X_i] = 0$. Hence according to Hoeffding's inequality,

$$\mathbb{P}\left[\left|\frac{X}{n} - 0\right| \geq \epsilon\right] \leq 2e^{\frac{-2n^2\epsilon^2}{4n}} \approx 2e^{\frac{-n\epsilon^2}{2}} \tag{3.5}$$

So,

$$\mathbb{P}[X \geq n\epsilon] \leq 2e^{\frac{-n\epsilon^2}{2}} \tag{3.6}$$

So with high probability, we can say that the person is at most at a distance of $\sqrt{n}$ away from the origin after n steps.

In the next lecture, we prove that with high probability the person is at distance at least $\Omega(\sqrt{n})$ of the origin.

3.2.3 Application 3: Discrepancy theory

In this part we discuss an application of Hoeffding's inequality in discrepancy theory. This is an important area of mathematics and it has many application is several areas of computer science including computational complexity and approximation algorithms.

Given a matrix $A \in \{0,1\}^{n\times n}$, we want to find a vector $x \in \{-1,+1\}^n$ such that $\|Ax\|_\infty \leq$ a small value. Note that for a vector $x \in \mathbb{R}^n$,

$$\|x\|_\infty = \max_i |x_i|.$$

In other words, we would like to color the columns of A with $+1$ or -1 such that the sum is as close to $\mathbf{0}$ as possible in the ℓ_∞ norm.

We show that if we choose x uniformly at random, then with high probability $\|Ax\|_\infty \leq O(\sqrt{n\log n})$.

Theorem 3.1. *Suppose we choose each coordinate of x uniformly at random in $\{+1,-1\}$. Then,*

$$\mathbb{P}\left[\|Ax\|_\infty \leq \sqrt{4n\ln n}\right] \geq 1 - 2/n.$$

Let $a_1, a_2, \ldots, a_n$ be the rows of A, i.e.,

$$A = \begin{pmatrix} \cdots a_1 \cdots \\ \cdots a_2 \cdots \\ \cdot \\ \cdot \\ \cdots a_n \cdots \end{pmatrix} \tag{3.7}$$

Then, we can write,

$$Ax = \begin{pmatrix} < a_1, x > \\ < a_2, x > \\ \cdot \\ \cdot \\ < a_n, x > \end{pmatrix} \tag{3.8}$$

So, to upper bound $\|Ax\|_\infty$, it is enough to show that with high probability for i, $\langle a_i, x\rangle \leq \sqrt{4n\log n}$.

Fix some $1 \leq i \leq n$. First, we show that with high probability $\langle a_i, x\rangle \leq \sqrt{4n\log n}$. First, observe that

$$\mathbb{E}[\langle a_i, x\rangle] = \sum_j a_{i,j}\mathbb{E}[x_j] = 0.$$

$2\exp\left(\frac{-2n^2\varepsilon^2}{4n}\right)$

$\hookrightarrow b(b_i - a_i)^2 n$

$$A = \begin{pmatrix} A_{11} & A_{12} \cdots & A_{1n} \\ \vdots & & A_{nn} \\ A_{n1} & & \end{pmatrix} \quad x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

$$Ax = \begin{pmatrix} A_{11}x_1 + A_{12}x_2 + \cdots + A_{1n}x_n \\ \cdots \\ A_{n1}x_1 + \cdots + A_{nn}x_n \end{pmatrix}$$

For each value we range from n

$[-\sum_j \|A_{ij}\|, \sum_j \|A_{ij}\|]$

$\mathbb{P}[|\frac{X}{n}| \geq \log n] \leq 2e^{\frac{-2n^2\varepsilon^2}{4n^2}}$

$\mathbb{P}(\|Ax\|_\infty \leq O\sqrt{n\log n})$

then $\mathbb{P}(\|Ax\|_\infty \leq c n\log n)$

$\geq (1 - 2e^{-(c\log n)^2})^n$

$\mathbb{P}[X \geq n\log n] \leq 2e^{-(c\log n)^2/2}$

$\leq 2\sqrt{\left(\frac{1}{nF}\right)}$

Note that in the expression $\sum a_{i,j}x_j$ any j for which $a_{i,j} = 0$, the value of x_j is irrelevant. Let $\|a_i\|_1 = \sum_j |a_{i,j}|$ be the number of nonzero entries of row i. Then, $\sum_j a_{i,j}x_j$ is distributed exactly the same as a random walk process on a line (that we discussed in the last section) of length $\|a_i\|_1$.

So, by (3.6), we have

$$\mathbb{P}\left[|\langle a_i, x\rangle| \geq \epsilon\|a_i\|_1\right] \leq 2e^{-\|a_i\|_1\epsilon^2/2} \tag{3.9}$$

We just replaced n with $\|a_i\|_1$ in the bound in (3.6).

So, for $\epsilon = \sqrt{4\ln n/\|a_i\|_1}$, we have

$$\mathbb{P}\left[|\langle a_i, x\rangle| \geq \sqrt{4\ln n\|a_i\|_1}\right] \leq 2e^{-2\ln n} = \frac{2}{n^2}. \tag{3.10}$$

Now, we use the union bound.

Union Bound Suppose we have a m (possibly intersecting) probability events $E_1, E_2, \ldots, E_m$. Then,

$$\mathbb{P}\left[\cup E_i\right] \leq \sum \mathbb{P}\left[E_i\right].$$

The proof of this simply follows from the following set-theoretic statement: For any family of sets $S_1, \ldots, S_m$,

$$|S_1 \cup S_2 \cup \cdots \cup S_m| \leq |S_1| + \ldots |S_m|.$$

Union bound is used a lot in conjunction with strong concentration bound. The reason is that strong concentration bounds prove a very sharp and small probability of failure so that even if we have many possibilities for failure still we can say none of them occur with high probability.

In the above, we showed that for any row i, with probability at most $2/n^2$,

$$|\langle a_i, x\rangle| \geq \sqrt{4\|a_i\|_1 \ln n}.$$

So, by union bound, with probability at least $1 - 2/n$, for all i,

$$|\langle a_i, x\rangle| \leq \sqrt{4\|a_i\|_1 \ln n} \leq \sqrt{4n \ln n}.$$

In other words, $\|Ax\|_\infty \leq \sqrt{4n \ln n}$ with probability at least $1 - 2/n$ as desired.

3.3 Introduction to Hashing

Now, we start talking about applications of randomization and probability in real-world problems. In the next couple of lectures we talk about Hashing.

Hashing is a technique of mapping the input data (images, vectors etc.) of arbitrary size to a finite set of hash values using a suitable hash function. Usually a special data structure called hash table is created to store the hash values, which makes data search, insertion and deletion faster. Suppose we have a set of large images (say each of size 1 MB) and we want to store them.

Since each image has $1000,000$ bits, we assume that we have a universe of numbers $U = \{1, 2, 3, \ldots, 2^{1000000}\}$ of all possible images. Say we want to store our images in a table of size N. Ideally we want $N \ll |U|$. So, we need a function $h : U \to \{1, 2, \ldots N\}$. Usually h is called a hash function. The question that we want to study is how to choose h.

$\mathbb{P}[|\langle a_i, x\rangle| \geq \sqrt{4\ln n(n)}] \leq 2e^{-2\ln n}$

$\hookrightarrow \leq 2e^{-\|a_i\|\varepsilon^2/2}$

$\mathbb{P}\left[\left|\frac{\langle a_i, x\rangle}{n}\right| \geq \varepsilon\right] \leq 2e^{\frac{-2n^2(\varepsilon)^2}{4n}} = 2e^{-\varepsilon n/2}$

let $\varepsilon = \sqrt{\frac{4\ln n}{n}}$

$= 2e^{-2\ln n} = \frac{2}{n^2}$

$\mathbb{P}[\cup E_i] \leq \sum \mathbb{P}[E_i]$

thus, $\mathbb{P}[\text{any row} \geq \sqrt{4n\ln n}] \leq \sum \mathbb{P}[\text{any row} \geq \sqrt{4n\ln n}]$

$= \frac{2}{n}$

$\mathbb{P}[\text{all row} < \sqrt{4n\ln n}] \leq 1 - \frac{2}{n}$

The choice of the hash function may depend on the nature of the input data and their distribution. An ideal hash function should have the property that the probability that two or more input samples getting mapped to the same hash value is low, that is it should be almost injective. If two or more samples map to the same hash value, we store them in a linked list whose address is stored at the location of the hash value in the hash table. So in order to avoid collision, we create a hash table such that each entry in the table is a linked list. So, ideally we want the length of the largest list to be as small as possible to minimize the time to query a given image.

At first, one might suggest a function that maps each image $h(X_i) = i \bmod N$. But, for such a function if all of our images have the same reminder modulo N, then they all map to the same location of the hash-table and the hashing is useless. In general, for a fixed hash function, we cannot expect to prove any worst case guarantee. So, instead we choose our hash function h from a family of functions $\mathcal{H}$ and we show that a random function chosen from $\mathcal{H}$ has a small number of collisions.

So, the question is how should we choose $\mathcal{H}$. Ideally, we want to choose $\mathcal{H}$ such that a random function maps each image to a uniformly and independently chosen location of the hash-table. Let us formalize this. We say a family $\mathcal{H}$ is 1-wise independent if for any image $X_1 \in U$ and any number $a_1 \in [N]$,

$$\mathbb{P}_{h\sim\mathcal{H}}[h(X_1) = a_1] = \frac{1}{N}$$

Here the probability is over a uniformly random function h chosen from $\mathcal{H}$.

We say $\mathcal{H}$ is 2-wise independent if for any pair of images X_1, X_2 and any pair of numbers $a_1, a_2 \in [N]$,

$$\mathbb{P}_{h\sim\mathcal{H}}[h(X_1) = a_1, h(X_2) = a_2] = \frac{1}{N^2}$$

The ideal case is if for any sequence of numbers $a_1, a_2, \ldots, a_{|U|}$ (with repetition) chosen from $[N]$,

$$\mathbb{P}\left[h(1) = a_1, \ldots, h(|U|) = a_{|U|}\right] = \frac{1}{N^{|U|}}$$

We shall study hashing and universal hash functions in more detail in the next lecture.

CSE 521: Design and Analysis of Algorithms I **Fall 2025**

Lecture 4: Hashing

Lecturer: Shayan Oveis Gharan *10/06/2025* *Scribe:*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications.*

At the end of the previous lecture, we introduced the basic notions of hashing and saw some of its applications. In this lecture, we are going to study hashing in more detail.

4.1 The Problem of Hashing

Let $U = \{0, 2, ..., 2^{100000} - 1\}$ be a large universe of numbers, let $X_1, ..., X_n \in U$ be n input numbers in such a universe where $n \ll |U|$. We think of them as images. Recall from the last lecture, our goal is to construct a family of hash functions $\mathcal{H}$, where every function in this family maps from U to $[N] := \{0, 2, ..., N-1\}$. We want to store these images in a data structure in order to be able to answer the following query in constant time, $O(1)$: Given an image Y, is there an image $X_i = Y$?

Throughout this document we always write $[p]$, for an integer $p > 0$ to denote the set $\{0, 1, \ldots, p-1\}$.

Recall that in the example of birthday paradox, where we have n people and N days in a year, the probability that two people were born in the same day is small when $N \gg n^2$. Therefore, if we map the n input numbers uniformly to $\{1, 2, ..., cn^2\}$ for some large enough constant c, then by an analysis similar to what we did in birthday paradox, we can show that the probability of collision is small. However, the problem here is that to record the uniform hash function, we need $|U| \log N$ bits, which is too big. Therefore, instead of choosing uniformly from all the possible mappings, we choose uniformly from a smaller set of functions. This motivates us to use hash functions with limited independence.

4.2 Limited Independence

Definition 4.1 (One-way Independence). *Let $\mathcal{H}$ be a family of hash functions, we say $\mathcal{H}$ is one-way independent if for all $X_1 \in U$ and for all $a_1 \in [N]$, we have*

$$\mathbb{P}_{h \sim H}[h(x_1) = a_1] = \frac{1}{N}.$$

Note that the above definition of one-way independence is not enough for a good family of hash functions. The family of constant functions $\{h_1, h_2, ..., h_N\}$ where $h_i(x) = i$ for every $x \in U$ is one-way independent. Constant functions give us the largest amount of collisions we can imagine. However, when we take one step further and use a pairwise independent family of functions, we are able to achieve small collision probability.

Definition 4.2 (Pairwise Independence). *Let $\mathcal{H}$ be a family of hash functions, we say $\mathcal{H}$ is pairwise independent if for all distinct $x_1, x_2 \in U$ and for all $a_1, a_2 \in [N]$, we have*

$$\mathbb{P}_{h \sim H}[h(x_1) = a_1, h(x_2) = a_2] = \frac{1}{N^2}.$$

In fact, in the case of our problem, hash function families with the below definition of approximate pairwise independence property is sufficient.

Definition 4.3 (Approximate Pairwise Independence). *Let $\mathcal{H}$ be a family of hash functions, we say $\mathcal{H}$ is α-approximate pairwise independent if for all distinct $x_1, x_2 \in U$ and for all $a_1, a_2 \in [N]$, we have*

$$\mathbb{P}_{h\sim H}[h(x_1) = a_1, h(x_2) = a_2] \leq \frac{\alpha}{N^2}.$$

Before proving that pairwise independence is sufficient for a good family of hash functions, we remark that we can extend the definitions 4.1 and 4.2 to k-wise independence. Although pairwise independence is already sufficient for our application today, k-wise independent hash functions are very important objects in computer science, and thus have found a lot of applications elsewhere.

Definition 4.4 (k-wise Independence). *Let $\mathcal{H}$ be a family of hash functions, we say $\mathcal{H}$ is k-wise independent if for all distinct $x_1, x_2, ...x_k \in U$ and for all $a_1, a_2, ..., a_k \in [N]$, we have*

$$\mathbb{P}_{h\sim H}[h(x_1) = a_1, h(x_2) = a_2, ..., h(x_k) = a_k] = \frac{1}{N^k}.$$

4.3 Birthday Paradox Revisit

Suppose we $\mathcal{H} = \{h : U \to [N]\}$ is a pairwise independent family of hash functions. Given n images $X_1, \ldots, X_n$, we choose a function $h \sim \mathcal{H}$ uniformly at random and we map each X_i to $h[X_i]$. We prove the following lemma:

Lemma 4.5. *If $N \geq \alpha n^2$ then with probability $1/2$ there is no collision.*

Note that assuming this lemma we can simply choose multiple $h \sim \mathcal{H}$ until we get no collision.

Now, let us revisit the analysis of the birthday paradox. We will see that the actual property that we need is approximate pairwise independence instead of mutual independence.

Similar to the analysis of the birthday paradox, we define Y_{ij} to be the indicator random variables that $h(X_i) = h(X_j)$, and let $Y = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} Y_{ij}$. Suppose that $\mathcal{H}$ is an α-approximate pairwise independent hash function family, then for every distinct $i, j \in [n]$

$$\mathbb{E}[Y_{ij}] = \mathbb{P}_{h\sim\mathcal{H}}[h(X_i) = h(X_j)] = \sum_{a\in[N]} \mathbb{P}_{h\sim\mathcal{H}}[h(X_i) = a, h(X_j) = a] \leq \frac{\alpha}{N^2} \cdot N = \frac{\alpha}{N}.$$

Therefore

$$\mathbb{E}[Y] = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \mathbb{E}[Y_{ij}] \leq \binom{n}{2}\alpha/N.$$

So if $\alpha n^2 < N$, then by Markov's inequality, we have

$$\mathbb{P}_{h\sim\mathcal{H}}[\forall \text{ distinct } i, j \in \{1, ..., n\}, h(X_i) \neq h(X_j)] = \mathbb{P}[Y = 0] \geq \frac{1}{2}$$

The above analysis shows that a family of hash functions with the property of α-approximate pair independence for some constant α would be suffice for our purpose.

As we will see in the following sections, to store the (approximate) pairwise independence hash functions, we will need $O(\log |U|)$ space. Therefore, the main downside of the above method is a quadratic loss in memory, i.e., to store n images we need to use $O(n^2)$ memory.

$\mathbb{E}X = \alpha\frac{\binom{n}{2}}{N} \leq 1/2$

$\mathrm{Var}X = \binom{n}{2}\left(\frac{1}{N}\right)\left(1-\frac{1}{N}\right) \leq \frac{(n)(n-1)(N-1)}{2N^2} \leq \frac{\alpha}{2}(N-1)$

$\mathbb{P}[|X - \mathbb{E}X| \geq \mathbb{E}X] \leq$

$\mathbb{P}[Y=0] = \mathbb{P}[$

$\mathbb{P}[Y=0] = 1 - \mathbb{P}[Y \geq 1] \geq 1 - \mathbb{E}X \geq 1/2$

4.4 Double Hashing

The material of this section follows from the work of Fredman et al. [**FKS84**]. In this section we see how to use a two layers hashing scheme to reduce the memory size to $O(n)$.

Instead of choosing $N = \Theta(n^2)$, we choose $N = n$ and we first choose $h \sim \mathcal{H}$ to map all images $X_1, \ldots, X_n$ to to N buckets. Note that for this choice of N we will have many collisions. Say Z_i be the random variable is the number of images that map to the i-th location. We choose another family pairwise independent hash functions $\mathcal{H}_i$ which map $U \to [N_i]$ for $N_i = \alpha Z_i^2$. Then, we choose $h \sim \mathcal{H}_i$ for a second layer of hashing; we choose $h_i \sim \mathcal{H}_i$. Then, we map each of the Z_i images which map to location i by h to one of $[N_i]$ locations using h_i. By the analysis of the previous section the probability of collision in the second layer is at most 1/2. So, we can test multiple samples for h_i until we get one with no collisions.

This double hashing method has obviously an $O(1)$ search time. Given a query Y, first we find $h(Y)$; say $h(Y) = i$. Then, we compute $h_i(Y)$ for the second layer. If no image is stored in $h_i(Y)$ we output "no". Otherwise, we check Y agains the unique image stored in $h_i(Y)$ and we output "yes" if they are the same and "no" otherwise.

Now, let us compute the expected size of the memory of this double hashing scheme. We use $O(n \log |U|)$ to store $n+1$ hash functions. The expected number of memory locations is at most

$$\mathbb{E}\sum_{i=1}^{N} \alpha Z_i^2 = \alpha\left(2\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \mathbb{E}Y_{ij} + n\right) = \frac{2\alpha^2\binom{n}{2}}{N} + \alpha n = O(\alpha^2 n).$$

The see the first identity note that

$$Z_i = \sum_{k=1}^{n} \mathbb{I}\left[h(X_k) = i\right].$$

Therefore,

$$Z_i^2 = Z_i + 2\sum_{1\le k<\ell\le n} \mathbb{I}\left[h(X_k) = h(X_\ell) = i\right]$$

So,

$$\sum_{i=1}^{n} Z_i^2 = \sum_{i=1}^{n} Z_i + 2\sum_{1\le k<\ell\le n} \mathbb{I}\left[h(X_k) = h(X_\ell)\right] = \sum_{i=1}^{n} Z_i + 2\sum_{1\le k<\ell\le n} Y_{k,\ell}.$$

Taking expectation from both sides proves the identity.

4.5 Construction of Pairwise Independent Hash Function

Let p be a prime so that $|U| \le p \le 2|U|$. Let variables a and b both be uniformly chosen from $\{0, \ldots, p-1\}$. We show that the family of functions $f_{a,b}(x) = ax + b \mod p$ is pairwise independent. Note that these functions map $[p] \to [p]$. Later we see that using a mod operation we can construct pairwise independent hash functions that map $[p] \to [N]$.

Claim 4.6. *For all $x, y \in \{0, \ldots, p-1\}$ such that $x \neq y$, and for all $s, t \in \{0, \ldots, p-1\}$, we have*

$$\mathbb{P}_{a,b}[f_{a,b}(x) = s, f_{a,b}(y) = t] = \frac{1}{p^2}$$

choose $h \to \mathcal{H}$ to map $X_1, \cdots, X_n$ to n buckets

in each bucket map $Y_1, \cdots, Y_{B_i}$ to square in bucket

expected space $= \mathbb{E}\sum a Z_i^2 \to$ probability $1/2$ with no collision in each bucket

$= \left(\sum\sum \mathbb{E}Y_{ij} + n\right)\alpha$ since $Z_i = \sum\sum \mathbb{E}Y_{ij} + n$

$Y_{ij} = \mathbb{1}[h(Y_i) = h(Y_j)]$

$\left(\binom{n}{2}\frac{\alpha}{N} + n\right)\alpha = O(\alpha^2 n)$

$Z_i = \sum_{k=1}^{n} \mathbb{1}\, h(X_k) = i$

$Z_i^2 = Z_i + 2\sum_{1\le k<\ell\le n} \mathbb{1}\left[h(X_k) = h(X_\ell) = i\right]$

$\sum Z_i^2 = \sum Z_i + 2\sum\sum \mathbb{1}\left(h(X_k) = h(X_\ell)\right)$

$= \sum Z_i + 2Y_{k,\ell}$

$\uparrow$

n

Proof. When $f_{a,b}(x) = s$ and $f_{a,b}(y) = t$, we have

$$\begin{aligned} ax + b &\equiv s \mod p \\ ay + b &\equiv t \mod p. \end{aligned}$$

Subtracting one from the other, we get

$$a(x - y) \equiv s - t \mod p.$$

Since p is a prime number, for every number $k \in \{1, ..., p-1\}$, there is a unique (modular) inverse k^{-1} of k so that $kk^{-1} \equiv 1 \mod p$. We do not discuss the algorithm for finding modular inverse, we refer students to https://en.wikipedia.org/wiki/Modular_multiplicative_inverse for details.

Since $x \neq y$, $x - y$ has a modular inverse. So, we can write solve the above system of modular equations for a and b; in particular, we have

$$a \equiv (s - t)(x - y)^{-1} \mod p.$$

Furthermore,

$$b \equiv s - ax \mod p$$

The above analysis shows that for a fixed x, y the following holds: Given any $s, t \in [p]$ there exists a pair (a, b) so that $f_{a,b}(x) = s$ and $f_{a,b}(y) = t$. Since there are p^2 possible options for (a, b) to take and p^2 many options for (s, t) this mapping is one-to-one. Therefore,

$$\mathop{\mathbb{P}}_{a,b}[f_{a,b}(x) = s, f_{a,b}(y) = t] = \frac{1}{p^2}$$

as desired. □

Now, we choose the family of hash functions to be $\mathcal{H} = \{h_{a,b}\}$ where

$$h_{a,b}(x) = f_{a,b}(x) \mod N.$$

Note that to store this function in memory, we only have to store a and b, which takes only $O(\log p) = O(\log |U|)$ many bits. For the particular application to Hashing universe U, we can use another idea to reduce the memory size to $O(\log n)$. Please refere to [**FKS84**] for details.

Now, we show that $\mathcal{H}$ is the family of hash functions with 2-approximate pairwise independence property.

Claim 4.7. *For all $x, y \in U$ so that $x \neq y$, we have*

$$\mathop{\mathbb{P}}_{a,b}[h_{a,b}(x) = h_{a,b}(y)] \leq \frac{1}{N} + \frac{1}{p}.$$

Proof. For all $x, y \in U$ so that $x \neq y$, $h_{a,b}(x) = h_{a,b}(y)$ if and only if $f_{a,b}(x) \equiv f_{a,b}(y) \mod N$. Thus, by Claim 4.6

$$\begin{aligned} \mathop{\mathbb{P}}_{a,b}[h_{a,b}(x) = h_{a,b}(y)] &= \mathop{\mathbb{P}}_{a,b}[f_{a,b}(x) \equiv f_{a,b}(y) \mod N] \\ &= \sum_{0 \leq s,t < p : s \neq t} \mathbb{P}[f_{a,b}(x) = s, f_{a,b}(y) = t] \mathbb{I}[s \equiv t \bmod N] \\ &= \sum_{0 \leq s,t < p : s \neq t} \frac{\mathbb{I}[s \equiv t \bmod N]}{p^2} \\ &\leq \frac{p\left\lceil \frac{p}{N} \right\rceil}{p^2} \leq \frac{1}{N} + \frac{1}{p} \end{aligned}$$

construction of pairwise indep fn.

$$\mathop{\mathbb{P}}_{a,b}(h_{a,b}(x) = h_{a,b}(y)) \leq \frac{1}{N} + \frac{1}{p}$$

2 case $x = y$ & $x \neq y$

$$\sum_{x \neq y} h_{a,b}(x) = h_{a,b}(y) + \sum_{x = y} h_{a,b}(x) = h_{a,b}(y)\ \mathbb{P}(X = x)^2 \mathbb{P}(Y = y)$$

$$= \frac{\binom{n}{2}}{p^2} + \frac{1}{N} < \frac{1}{N} + \frac{1}{p}$$

The first inequality follows by the fact that for any $s \in [p]$ there are at most $\lceil p/n \rceil$ numbers t such that $s \equiv t \bmod N$. □

Note that the family of functions that we construct above is approximately pairwise independent but that is enough for all interesting applications.

We remark that we can extend the above construction and obtain a family of hash functions that is k-wise independent. For some prime number p, consider the family of hash functions

$$f_{a_0,...,a_{k-1}}(x) = a_{k-1}x^{k-1} + ... + a_1 x + a_0,$$

where $a_0, ..., a_{k-1}$ are uniformly chosen in $\{0, 1, ..., p-1\}$. Similar to the invertible argument we used above, the proof that this construction is k-wise independent follows from the fact that the Vandermonde matrix

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_k \\ x_1^2 & x_2^2 & \cdots & x_k^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{k-1} & x_2^{k-1} & \cdots & x_k^{k-1} \end{bmatrix}$$

is invertible for distinct $x_1, ..., x_k$. So, in general we can store a k-wise independent hash function with only $O(k \log |U|)$ amount of memory.

CSE 521: Design and Analysis of Algorithms I **Fall 2025**

Lecture 5: Unbiased Estimators, Streaming

Lecturer: Shayan Oveis Gharan

10/08/2025

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications.*

In previous lectures, we showed that by Chebyshev's inequality, any random variable has chance at least $1 - \frac{1}{k^2}$ of taking a value in interval $[\mu - k\sigma, \mu + k\sigma]$, where μ, σ are the mean and standard deviation, respectively. If we take t independent samples (sometimes pairwise independent is also enough), then the variance of the sample average is σ^2/t. Hence by increasing t, we can a better estimate of μ. How many samples do we need to get a good estimate of μ? In particular, to get ϵ-additive approximation of μ with probability $1 - \delta$ it is enough to use $O(1/\delta\epsilon^2)$ many independent samples.

An ϵ-additive approximation is not desirable in many applications because the range of the ϵ may be independent of the magnitude of μ. For example, if μ is in the interval $[0.001, 0.002]$, a 0.1-additive approximation to μ has no information. Instead, a multiplicative approximation scales proportional to the magnitude of μ. In the next section we will see how many samples we need to obtain a $1 \pm \epsilon$ approximation to μ.

5.1 Unbiased Estimators

We say a random variable X is an unbiased estimator of μ if

$$\mathbb{E}[X] = \mu.$$

It turns out the the number of samples is proportional to the relative variance of X.

Definition 5.1 (Relative Variance). *Say X is an unbiased estimator of μ, then, the relative variance of X is defined as*

$$\frac{\sigma^2(X)}{\mu^2}, \tag{5.1}$$

where by $\sigma^2(X) = \mathbb{E}[X]^2 - (\mathbb{E}[X])^2$ is the variance of X. We typically use γ^2 to denote the relative variance.

The following theorem is the main result of this section.

Theorem 5.2. *Given $\epsilon, \delta > 0$, and an unbiased estimator of μ, X. We can approximate μ within $1 \pm \epsilon$ multiplicative factor using only $O(\frac{t}{\epsilon^2} \log \frac{1}{\delta})$ independent samples of X with probability $1 - \delta$.*

Before going into the details of the proof let us discuss a motivating example.

Dart throwing method of estimating areas. Suppose we want to estimate the area of a closed curve on the plane (see curve A of section 5.1). We can use the well-known Monte Carlo method. The idea is to draw a rectangle B that includes A. Then, we randomly sample a point in B. Let

$$X = \begin{cases} 1, & \text{if the point is belong to } B \\ 0, & \text{otherwise} \end{cases}$$

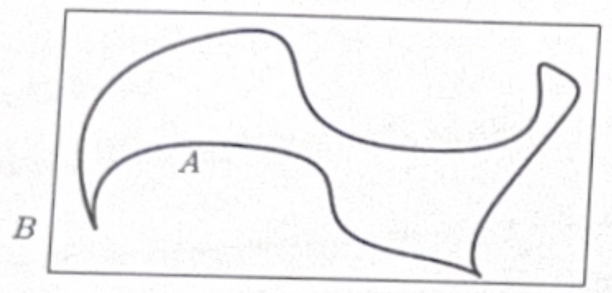


Figure 5.1: Estimating Area by Monte Carlo Method

Observe that $\mathbb{E}X = \mathbb{P}[X = 1] = s(A)/s(B)$, where $s(.)$ denotes the surface area function. Since we can exactly calculate $s(B)$, we can use $Y = s(B)X$ as an unbiased estimator of $s(A)$. Now, we can use Theorem 5.2 to find the number of independent samples of X that we need to estimate Y within a $1 \pm \epsilon$ factor.

All we need to know is that relative variance of X; we have

$$t = \frac{\sigma^2(Y)}{\mu^2} = \frac{s(B)^2\sigma^2(X)}{s(A)^2} \leq \frac{s(B)^2\mathbb{E}[X^2]}{s(A)^2} = \frac{s(A)\cdot s(B)}{s(A)^2} = \frac{s(B)}{s(A)} \tag{5.2}$$

where we used that X is a Bernoulli random variable with prior $s(A)/s(B)$. So, we need $O(\frac{s(B)}{s(A)} \cdot \frac{\log\frac{1}{\delta}}{\epsilon^2})$ independent samples of X to find an $1 \pm \epsilon$ approximation of $s(A)$ with probability $1 - \delta$. Note that we need to sample X t many times (in expectation) to just get 1 point that belongs to A.

Remark. The above Monte Carlo method is a fairly general method to estimate a quantity of interest. Suppose we have an object X that we want to measure. The idea is to find a bigger object Y that contains X such that (i) We can measure Y and (ii) We can generate samples from Y. Then, we can use the above method to approximately measure X. As we discussed the number of samples that we need is proportional to the measure of Y with respect to X, up to a $\log(1/\delta)/\epsilon^2$ factor.

Proof of Theorem 5.2. First, we give an algorithm that estimates μ within $1 \pm \epsilon$ factor with probability 9/10 using only $O(t/\epsilon^2)$ samples. Then, we show how we can boost the success probability to $1 - \delta$ using the "median trick".

Let $X_1, \ldots, X_k$ be k independently chosen samples of X. Since X is an unbiased estimator, for all i, $\mathbb{E}[X_i] = \mu$. Let $Y = \frac{1}{k}(X_1 + \cdots + X_k)$ be the average of X_i's; by linear property of expectation $\mathbb{E}[Y] = \mu$. So, by Chebyshev's inequality, we have

$$\begin{aligned} \mathbb{P}[(1-\epsilon)\mu \leq Y \leq (1+\epsilon)\mu] &= \mathbb{P}[|Y - \mu| \leq \epsilon\mu] \\ &\geq 1 - \frac{\sigma^2(Y)}{\epsilon^2\mu^2} \qquad (5.3) \\ &= 1 - \frac{\sigma^2(X)}{\epsilon^2 k\mu^2} = 1 - \frac{t}{k\epsilon^2}. \qquad (5.4) \end{aligned}$$

Hence, by taking $k = O(\frac{t}{\epsilon^2})$, samples, we can get a $1 \pm \epsilon$ approximation of μ with probability 9/10.

To obtain $\log\frac{1}{\delta}$ probability of success we need to use Chernoff type of bounds. However, these bounds usually need some specific assumption on the distribution of the random variables that we average out, e.g., that the third or fourth moments are bounded. In our particular case, we have no prior assumption on the distribution of X. We only have a handle on the expectation and variance of X because we know the relative variance.

$\mathbb{P}(|\mu_A - \mu| \leq \varepsilon\mu) \geq 1-\delta$ with $O(\frac{t}{\varepsilon^2}\log\frac{1}{\delta})$ samples

$X = \sum \mathbb{1}[X_i] \times s(B)$ $\quad t = \frac{\mathrm{Var}\,X}{\mu^2}$

$Y = s(B)X$, $\quad \frac{\mathrm{Var}(Y)}{(\mathbb{E}Y)^2} = \frac{s(B)^2\mathrm{Var}\,X}{s(B)^2\mathbb{E}X^2} = \frac{s(B)^2\left(\frac{s(A)}{s(B)}\right)\left(1-\frac{s(A)}{s(B)}\right)}{s(B)^2\frac{s(A)}{s(B)}}$

$= \frac{s(B)-s(A)}{s(A)}$

$\leq \frac{s(B)}{s(A)}$

with $O\left(\frac{s(B)}{s(A)}\,\frac{\log\frac{1}{\delta}}{\varepsilon^2}\right)$

Pf of needing $O(t/\varepsilon^2 \log\frac{1}{\delta})$ samples

1. Give alg that succeeds w.p. 9/10 $\quad X$ = r.v. $\quad Y = \frac{\sum X_i}{k}$

$\mathbb{P}(|Y-\mu| \leq \varepsilon\mu] \geq 1 - \frac{\mathrm{Var}\,Y}{\varepsilon^2\mu^2}$ Chebyshev

$= 1 - \frac{\mathrm{Var}\,X}{k\varepsilon^2\mu^2} = 1 - \frac{t}{k\varepsilon^2}$ $\Rightarrow$ take $k \in O(t/\varepsilon^2)$ for approximate

2. boost using median trick

let Z_i be ith Y fail w.p. $\leq t/k\varepsilon^2$

$\mathbb{P}[Z \geq \ell/2] \leq \frac{2t/k\varepsilon^2}{\ell}$ $\leftarrow$ not useful

The idea is to use a trick called "median trick". Fix, $k = O(t/\epsilon^2)$, such that

$$\mathbb{P}\left[(1-\epsilon)\mu \le Y \le (1+\epsilon)\mu\right] \ge 9/10. \tag{5.5}$$

This follows simply from (5.4). We output the median value from the ℓ independent samples of Y. Call these samples, $Y_1, \ldots, Y_\ell$. Observe that the median of Y_i's will be in the interval $[(1-\epsilon)\mu, (1+\epsilon)\mu]$ if at least half of Y_i's are in this interval $[(1-\epsilon)\mu, (1+\epsilon)\mu]$.

We show that the probability that half of the Y_i's are outside this interval is very small. Define

$$Z_i := \mathbb{I}\left[Y_i \in [(1-\epsilon)\mu, (1+\epsilon)\mu]\right]$$

be the random variable indicating that Y_i is in $[(1-\epsilon)\mu, (1+\epsilon)\mu]$. Note that by (5.5) for each i, $\mathbb{P}[Z_i] \ge 9/10$. By linearity property of expectation, we have $\mathbb{E}\left[\sum_i Z_i\right] = \sum_i \mathbb{E} Z_i \ge \frac{9\ell}{10}$. By Hoeffding's inequality,

$$\mathbb{P}\left[\sum_{i=1}^{\ell} Z_i \le \frac{\ell}{2}\right] \le \mathbb{P}\left[\left|\sum_{i=1}^{\ell} Z_i - \mathbb{E}\left[\sum_{i=1}^{\ell} Z_i\right]\right| > \left(\frac{\ell}{4}\right)\right] \quad \text{why} \tag{5.6}$$

$$\le e^{-\ell/8} \tag{5.7}$$

where in the first inequality we used that $\mathbb{E}\left[\sum_i Z_i\right] \ge 9\ell/10$. Choosing ℓ such that $e^{-\ell/8} \le \delta$, i.e., $\ell = O(\log 1/\delta)$, we only need $O(t \log \frac{1}{\delta}/\epsilon^2)$ samples of X to obtain a $1 \pm \epsilon$ approximation of μ with probability at least $1-\delta$. □

5.2 Introduction to Streaming Algorithms

As an application of hashing and the unbiased estimator, we are going to discuss streaming algorithms. Streaming algorithms has become a hot topic in computer science nowadays because of the massive amount of data that we have to process. Typically, we do not have enough space to store the entire data. Instead, we process the data in a streaming fashion, and sketch the information we want from the data by a few passes.

We will talk about algorithms for F_0 and F_2 estimation. Those are classic results appeared in the first paper of streaming algorithms [AMS96]. The problem is as follows.

Let $\mathcal{U} = \{1, ..., |\mathcal{U}|\}$ be a large universe of numbers, and let $X_1, ..., X_n$ be a sequence of numbers in $\mathcal{U}$. Let $f_i = \sum_{j=1}^{n} \mathbb{I}[X_j = i]$ be the number of times i appears in the sequence. For $0 \le k \le \infty$, we let $F_k = \sum_{i=1}^{|\mathcal{U}|} f_i^k$, where we define $0^0 = 0$. The interesting values of k for us are

- When $k = 0$, F_0 counts the number of distinct elements in the sequence.
- When $k = 2$, F_2 is the second moment of the vector $(f_1, ..., f_{|\mathcal{U}|})$.
- When $k = \infty$, F_∞ corresponds to the number of times the most frequent number shows up in the sequence.

The following theorem is proven in [AMS96]

Theorem 5.3. *There is a streaming algorithm that for any sequence $x_1, \ldots, x_n$ of the universe $\{1, 2, \ldots, |U|\}$ gives a $(1-\epsilon)$ approximation of F_0 and F_2 using $O(\frac{\log |U| + \log n}{\epsilon^2} \cdot \log \frac{1}{\delta})$ space with probability $1-\delta$.*

Here, we only give an algorithm for F_2 and we leave the algorithm for F_0 as a homework exercise.

We remark that allowing randomness and approximated solution is crucial. There is no hope to use a deterministic or exact algorithm to achieve logarithmic amount of space. Please see Tim Roughgarden's Lecture notes for more details.

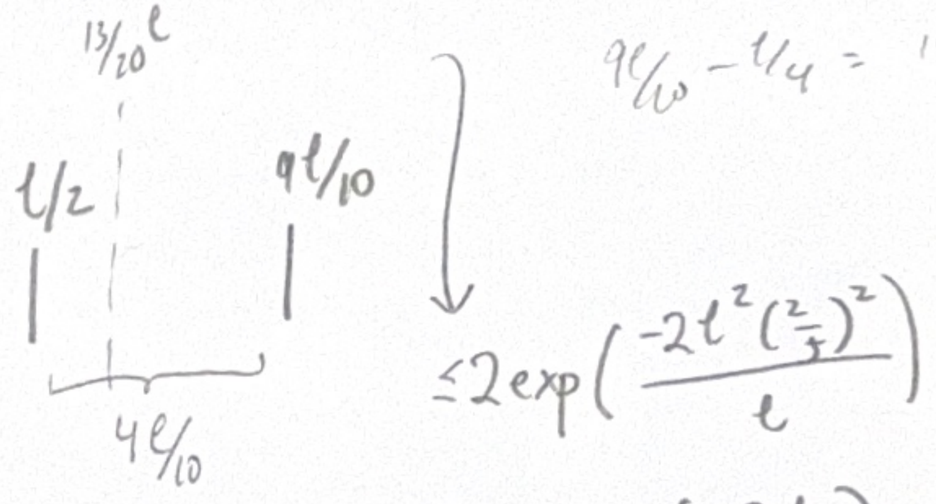
$\mathbb{E}[\sum Z_i] \ge 9\ell/10$

Hoeffding

$\mathbb{P}[\sum Z_i \le \ell/2] \le \mathbb{P}[|\sum Z_i - \mathbb{E}\sum Z_i| > 4\ell/5]$

$\le e^{-\ell/8}$ Hoeffding

succeed



$9\ell/10 - \ell/4 = 18\ell/20 - 5\ell/20 = 13\ell/20$

$\le 2\exp\left(\frac{-2\ell^2(\frac{2}{5})^2}{\ell}\right)$

$= 2\exp\left(\frac{-2\ell}{25}\right)$