15-852 Randomized Algorithms
Notes for 1/20/97

```
* useful probabilistic inequalities
* Randomized complexity classes
```

Useful probabilistic inequalities

Say we have a random variable X. Often want to bound the probability that X is too far away from its expectation. [In first class, we went in other direction, saying that with reasonable probability, a random walk on n steps reached at least $\sqrt{n}$ distance away from its expectation]
Here are some useful inequalities for showing this:

Markov's inequality: Let X be a non-negative r.v. Then for any positive k:

$$\mathbf{Pr}[X \geq k\mathbf{E}[X]] \;\leq\; 1/k.$$

(No need for k to be integer.) Equivalently, we can write this as:

$$\mathbf{Pr}[X \geq t] \;\leq\; \mathbf{E}[X]/t.$$

Proof. $\mathbf{E}[X] = \mathbf{Pr}[X \geq t] \cdot t + \mathbf{Pr}[X < t] \cdot 0 \geq t \cdot \mathbf{Pr}[X \geq t]$.

Defn of Variance: $\mathbf{var}[X] = \mathbf{E}[(X - \mathbf{E}[X])^2]$. Standard deviation is square root of variance. Can multiply out variance definition to get:

$$\mathbf{var}[X] = \mathbf{E}[X^2 - 2X\mathbf{E}[X] + \mathbf{E}[X]^2] = \mathbf{E}[X^2] - (\mathbf{E}[X])^2.$$

Chebyshev's inequality: Let X be a r.v. with mean μ and standard deviation σ. Then for any positive t, have:

$$\mathbf{Pr}[|X - \mu| > t\sigma] \;\leq\; 1/t^2.$$

Proof. Equivalently asking what is the probability that $(X-\mu)^2 > t^2\mathbf{var}[X]$. Now, just think of l.h.s. as a new non-negative random variable Y. What is its expectation? So, just apply Markov's inequality.

Let's suppose that our random variable $X = X_1 + \ldots + X_n$ where the X_i are simpler things that we can understand. Suppose there is not necessarily any independence. Then we can still compute the expectation

$$\mathbf{E}[X] = \mathbf{E}[X_1] + \ldots + \mathbf{E}[X_n]$$

and use Markov. (i.e., expectation is same as if they were independent)

Suppose we have pairwise independence. Then, $\mathbf{var}[X]$ is same as if the X_i were fully independent. In fact, $\mathbf{var}[X] = \sum_i \mathbf{var}[X_i]$.
Proof.

$$\begin{aligned}\mathbf{E}[X^2] - (\mathbf{E}[X])^2 &= \sum_{i,j} \mathbf{E}[X_i X_j] - \sum_{i,j} \mathbf{E}[X_i]\mathbf{E}[X_j] \\ &= \sum_i E[X_i^2] - \sum_i E[X_i]^2\end{aligned}$$

where the last inequality holds because $\mathbf{E}[XY] = \mathbf{E}[X]\mathbf{E}[Y]$ for independent random variables, and all pairs here are independent except when $i = j$. So, can apply Chebyshev easily.

Chernoff and Hoeffding bounds

What if the X_i's are fully independent? Let's say X is the result of a fair, n-step $\{-1, +1\}$ random walk (i.e., $\mathbf{Pr}[X_i = -1] = \mathbf{Pr}[X_i = +1] = 1/2$ and the X_i are mutually independent.) In this case, $\mathbf{var}[X_i] = 1$ so $\mathbf{var}[X] = n$ and $\sigma(X) = \sqrt{n}$. So, Chebyshev says:

$$\mathbf{Pr}[|X| \geq t\sqrt{n}] \leq 1/t^2.$$

But, in fact, because we have full independence, we can use the stronger *Chernoff* and *Hoeffding* bounds that in this case tell us:

$$\mathbf{Pr}[X \geq t\sqrt{n}] \leq e^{-t^2/2}.$$

The book contains some forms of these bounds. Here are some forms of them that I have found to be especially convenient.
Let $X_1, \ldots, X_n$ be a sequence of m independent $\{0, 1\}$ random variables with $\mathbf{Pr}[X_i = 1] = p_i$ not necessarily the same. Let S be the sum of the r.v., and $\mu = \mathbf{E}[S]$. Then, for $0 \leq \delta \leq 1$, the following inequalities hold:

- $\mathbf{Pr}[S > (1 + \delta)\mu] \leq e^{-\delta^2\mu/3}$,
- $\mathbf{Pr}[S < (1 - \delta)\mu] \leq e^{-\delta^2\mu/2}$.

Additive bounds:

- $\mathbf{Pr}[S - \mu > \delta n] \leq e^{-2n\delta^2}$.
- $\mathbf{Pr}[S - \mu < -\delta n] \leq e^{-2n\delta^2}$.

Here is a somewhat intuitive proof, for the case of a fair random walk. The book has some less intuitive but shorter proofs too.

Theorem 1 *Let $X = X_1 + \ldots + X_n$ with $\mathbf{Pr}[X_i = 1] = \mathbf{Pr}[X_i = -1] = 1/2$, and X_i mutually independent. Then*

$$\mathbf{Pr}[X > \lambda\sqrt{n}] < e^{-\lambda^2/2}$$

for $\lambda > 0$.

Proof. Let's look at a multiplicative version of the random walk. Let's say that we start at 1, and on a heads we multiply our current position by $(1+\epsilon)$ and on a tails we divide our current position by $(1+\epsilon)$. So, we can write the random variable Y for this walk as:

$$Y = Y_1 \cdot Y_2 \cdots Y_n$$

where $\mathbf{Pr}[Y_i = (1+\epsilon)] = \mathbf{Pr}[Y_i = 1/(1+\epsilon)] = 1/2$ and the Y_i are independent. What does the distribution on Y look like? Just like in the standard additive random walk, the median of the distribution is our starting point (i.e., there is a 50/50 chance we will end up below 1 and a 50/50 chance we will end up above 1). But, the *expectation* is much larger, since only a few additional steps to the right can move us large distances. Formally, doing a simple calculation gives us:

$$\mathbf{E}[Y_i] = 1 + e^2/(2+2\epsilon) \leq 1 + \epsilon^2/2$$

and therefore (using the fact that the Y_i are independent):

$$\mathbf{E}[Y] \leq (1+\epsilon^2/2)^n.$$

Let's now think about what Markov's inequality applied to Y, i.e.,

$$\mathbf{Pr}[Y > k \cdot \mathbf{E}[Y]] \leq 1/k$$

tells us about our original (additive) version of the random walk. What happens is we lose something (compared to applying Markov to X directly) in that $\mathbf{E}[Y]$ is pretty far to the right — we think it is "expected" for X to be as large as $\log_{1+\epsilon}(\mathbf{E}[Y])$ — but we *gain* something critical: if X is just, say, $20/\epsilon$ steps larger than this value, then that corresponds to Y being a huge $(1+\epsilon)^{20/\epsilon} \approx e^{20}$ times larger than its expectation, which by Markov has probability only $1/e^{20}$. Formally,

$$\begin{aligned}
\mathbf{Pr}[X > \log_{1+\epsilon}(k \cdot \mathbf{E}[Y])] &\leq 1/k \\
\mathbf{Pr}[X > \log_{1+\epsilon}(k) + \log_{1+\epsilon}((1+\epsilon^2/2)^n)] &\leq 1/k \\
\mathbf{Pr}[X > \log_{1+\epsilon}(k) + n\epsilon/2] &\leq 1/k
\end{aligned}$$

(where a bit of calculation gets you from the second-to-last to the last line). If we now set $k = (1+\epsilon)^{n\epsilon/2} \approx e^{n\epsilon^2/2}$, we get:[1]

$$\mathbf{Pr}[X > n\epsilon] \leq e^{-n\epsilon^2/2}$$

and setting $\epsilon = \lambda/\sqrt{n}$ gives us:

$$\mathbf{Pr}[X > \lambda\sqrt{n}] \leq e^{-\lambda^2/2}$$

as desired. ∎

[1] Actually, I believe this approximation is slightly off in the wrong direction. So, to do this formally we need to have been more careful with our approximations above...

Randomized complexity classes

Let A denote a poly time algorithm that takes two inputs: a (regular) input x and an "auxiliary" input y where y has length $l(|x|)$ where l is a polynomial and is poly-time computable. Think of y as the random bits.

- **RP**: One-sided error. Language L (decision problem) is in **RP** if there exists a poly time A:

  For all $x \in L$, $\mathbf{Pr}_y[A(x,y) = 1] \geq 1/2$.

  For all $x \notin L$, $\mathbf{Pr}_y[A(x,y) = 1] = 0$.

  ($x \in L$ means x is something the algorithm is supposed to output 1 on.)

  For instance, there are algorithms for primality that have the following property: If the number is prime, then they output "PRIME". If it is composite, then they output "PRIME" with prob. at most 1/2. So, this is RP for compositeness.

- **BPP**: Like RP, but:

  For all $x \in L$, $\mathbf{Pr}_y[A(x,y) = 1] \geq 3/4$.

  For all $x \notin L$, $\mathbf{Pr}_y[A(x,y) = 1] \leq 1/4$.

- It is believed that $BPP \subseteq P$. I.e., Randomness is useful for making things simpler and faster (or for protocol problems) but not for polynomial versus non-polynomial.

- **P/poly**: L is in P/Poly if there exists a poly time A such that for every $n = |x|$, there exists a fixed y such that $A(x,y)$ is always correct. I.e., y is an "advice" string. (Remember, $|y|$ has to be polynomial in n, etc.) Also, can view as class of polynomial-size circuits.

  RP in P/poly: Say A is an **RP** algorithm for L that uses ℓ random bits. Consider an algorithm $\tilde{A}$ that uses an auxiliary input y of length $\ell(n+1)$ to run $n+1$ copies of A, and then outputs 1 if any of them produced a 1 and outputs 0 otherwise. Then, the probability (over y) that $\tilde{A}$ fails on a given input x of length n is at most $1/2^{n+1}$. Therefore, with probability at least 1/2, a single random string y will cause $\tilde{A}$ to succeed on *all* inputs of length n. Therefore, such a y must exist. ■

Another kind of distinction: Algs like quickselect where always give right answer, but running time varies are called *Las-Vegas algs*. Another type are *Monte-Carlo algs* where always terminate in given time bound, but say have only 3/4 prob. of producing the desired solution (like RP or BPP or primality testing).