Fall 2014 Randomized Algorithms Oct 1, 2012

Lecture 2

Prof. Friedrich Eisenbrand *Scribes: Christos Kalaitzis*

During the last lecture, we had a small taste of the kind of problems we will be focusing on during this class, as well as some of the basic techniques and ideas that will be coming up a lot. During this lecture, we will try to set up a small toolbox, albeit a very powerful one. We will talk about the Chernoff bounds, one of the most widely used (set of) probabilistic inequalities. The point behind these inequalities(which also describe the settings in which we wish to use them) is very simple: guaranteeing concentration around the mean. In other words, we want to have some strong guarantees on the probability of the sample of a random variable being close to the variable's mean, and the Chernoff bounds offer such guarantees, when we are dealing with sums of independent random Bernoulli variables. Finally, we will also see some of its applications, namely how to apply these bounds in parameter estimation, and an algorithm to efficiently approximate the number of different solutions to a Knapsack problem.

1 Chernoff bounds

To start things off, we will first state what could be the most fundamental probabilistic inequality in Theoretical Computer Science, the Markov inequality. This inequality relates the probability of the sample of a nonnegative random variable deviating from the mean to its expected value:

Markov inequality. *Given a nonnegative random variable X, we have*

$$\Pr[X \geq \alpha] \leq \frac{\mathrm{E}[X]}{\alpha}$$

It is worth noting that this inequality is tight, in the sense that there are random variables for which there are values of α for which the statement holds with equality. Furthermore, as we will see later on, many other "stronger" probabilistic inequalities are applications of the Markov inequality.

Now, while the Markov inequality might prove useful in many situations, there are many others in which it is underwhelming: for example, consider the random variable $X = \sum_{i=1}^{n} X_i$, where the X_i-s are i.i.d. random variables, uniformly distributed in $\{0, 1\}$. Then we have that

$$\mathrm{E}[X] = n/2$$

$$\Pr[X \geq n] = 2^{-n}$$

On the other hand, if we tried getting a bound on the probability of X being large using the Markov inequality, we would get

$$\Pr[X \geq n] \leq 1/2$$

which is hardly a good bound(observe that the gap between the Markov bound and the actual probability is exponential in n). While this is an annoying fact, there is a way to come around it: Chernoff bounds.

The idea between Chernoff bounds is to transform the original random variable into a new one, such that the distance between the mean and the bound we will get is significantly stretched. Towards this end, consider the random variable e^X; then we have:

$$\Pr[X \geq 2\mathrm{E}[X]] = \Pr[e^X \geq e^{2\mathrm{E}[X]}]$$

Let us first calculate $\mathrm{E}[e^X]$:

$$\mathrm{E}[e^X] = \mathrm{E}\left[\prod_{i=1}^{n} e^{X_i}\right] = \prod_{i=1}^{n} \mathrm{E}[e^{X_i}] = \left(\frac{1+e}{2}\right)^n$$

where in the second to last equality we used independence of our random variables. Now, select α to be

$$\alpha = \left(\frac{2e}{1+e}\right)^n$$

The reason behind the choice of this α will become apparent in the next calculation; applying Markov inequality to e^X we will have:

$$\Pr[e^X \geq e^{2\mathrm{E}[X]}] = \Pr[e^X \geq e^n] = \Pr[e^X \geq \alpha \mathrm{E}[e^X]] \leq \frac{1}{\alpha} = \left(\frac{1+e}{2}\right)^n$$

which is a far better bound compared to the one given by using Markov inequality on the original random variable X.

After this discussion, we are ready to state the Chernoff bounds for deviation above the mean formally:

Chernoff bounds. *Let* $X = \sum_{i=1}^{n} X_i$ *be the sum of* n *i.i.d. Bernoulli random variables such that*

$$\Pr[X_i = 1] = p_i$$

$$\Pr[x_i = 0] = 1 - p_i$$

for all $i = 1$ *to* n*, and let* $\mathrm{E}[X] = \sum_{i=1}^{n} p_i = \mu$*. Then*

- $\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^\delta}{(1+\delta)^{1+\delta}}\right)^\mu$*, for* $\delta > 0$*.*

- $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\mu\delta^2}{3}}$, *for* $0 < \delta \leq 1$.

Proof In similar fashion as before, we first express the actual probability we want to bound differently:

$$\Pr[X \geq (1+\delta)\mu] = \Pr[e^{tX} \leq e^{t(1+\delta)\mu}]$$

Again similarly to before, we take a look at the expectation of e^{tX}:

$$\mathrm{E}[e^{tX}] = \mathrm{E}[\prod_{i=1}^{n} e^{tX_i}] = \prod_{i=1}^{n} \mathrm{E}[X_i] = \prod_{i=1}^{n} \left(p_i e^t + (1-p_i)\right) = \prod_{i=1}^{n} 1 + p_i(e^t - 1) \leq \prod_{i=1}^{n} e^{p_i(e^t-1)}$$

Here, the second equality is due to independence of random variables and the last one due to elementary calculus.

Now, let $t = \ln(\delta+1)$; $t > 0$ since $\delta > 0$. Since $\prod_{i=1}^{n} e^{p_i\delta} = e^{\mu\delta}$, and applying Markov inequality, we have:

$$\Pr[X \geq (1+\delta)\mu] = \Pr[e^{tX} \leq e^{t(1+\delta)\mu}] \leq \frac{e^{\mu\delta}}{e^{t(1+\delta)\mu}} = \frac{e^{\mu\delta}}{(1+\delta)^{(1+\delta)\mu}} = \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu}$$

which proves the first case of $\delta > 0$; if in addition $\delta \leq 1$, we have that

$$(1+\delta)\ln(1+\delta) + \frac{\delta^2}{3} \leq 0$$

which implies

$$\left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu} \leq e^{-\frac{\mu\delta^2}{3}}$$

■

The proof of the Chernoff bounds for deviation under the mean is similar:

Chernoff bounds. *Let* $X = \sum_{i=1}^{n} X_i$ *be the sum of* n *i.i.d. Bernoulli random variables such that*

$$\Pr[X_i = 1] = p_i$$
$$\Pr[x_i = 0] = 1 - p_i$$

for all $i = 1$ *to* n*, and let* $\mathrm{E}[X] = \sum_{i=1}^{n} p_i = \mu$*. Then*

- $\Pr[X \leq (1-\delta)\mu] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right)^{\mu}$, *for* $\delta > 0$.
- $\Pr[X \leq (1-\delta)\mu] \leq e^{-\frac{\mu\delta^2}{2}}$, *for* $0 < \delta \leq 1$.

To get an idea of how these bounds can be used in practice, consider the following vanilla problem: we are given an n-dimensional cube C, and we are told that inside this cube someone has placed a finite number of flags at a set of points $X \subseteq C$. We would like to count how many these flags are, but we cannot search the whole cube; instead, the access we are given is that we can sample a random subcube $S \subseteq C$ such that for any two points $x, y \in C$ $\Pr[x \in S] = \Pr[y \in S]$, and find out how many flags are within S. In that case, if we are guaranteed that the volume of S is not insignificant relative to that of C(i.e. at least a $1/p(n)$-fraction, where p is a polynomial), then taking polynomially many such samples would allow us to estimate the number of flags with very good precision. The key fact however is that this is a pattern that will be coming back in our next applications.

2 Parameter estimation

Next, we will turn our attention to parameter estimation, i.e. the problem of estimating the parameters of an underlying random variable, from which we can only sample. Consider the following instance of parameter estimation: we are given n i.i.d. samples $\{X_1 ... X_n\}$ of a Bernoulli random variable, which is set to 1 with probability p and 0 with probability $1-p$, and we have to output an estimate $\tilde{p}$ of p within an ϵ-error, i.e. $(1-\epsilon)p \leq \tilde{p} \leq (1+\epsilon)p$. Of course, the most natural answer would be to output $\tilde{p} = \frac{\sum_{i=1}^{n} X_i}{n}$. Now the question is: how large does n have to be in order to get a good confidence level in our estimate, i.e. in order to guarantee that $(1-\epsilon)p \leq \tilde{p} \leq (1+\epsilon)p$ with probability at least $1-\alpha$, for some $\alpha > 0$? As we will see, answering this question involves using the Chernoff bounds crucially.

Let $X = \sum_{i=1}^{n} X_i$, and let $\delta > 0$ be a parameter. Then, if $p < \tilde{p} - \delta$, $X = n\tilde{p} > m(p+\delta) = \mathrm{E}[X](1+\delta/p)$, while if $p > \tilde{p} + \delta$, $X + n\tilde{p} < n(p-\delta) = \mathrm{E}[X](1-\delta/p)$. Using the Chernoff bounds, we get that the probability of failing is

$$\Pr[p \notin [\tilde{p} - \delta, \tilde{p} + \delta]] \leq e^{-np\left(\frac{\delta}{p}\right)^2/2} + e^{-np\left(\frac{\delta}{p}\right)^2/3}$$

Remember that we would like

$$\Pr[\tilde{p} \notin [(1-\epsilon)p, (1+\epsilon)p]] \leq \alpha$$

Hence, as long as $\delta/p \leq \epsilon$, all we have to do is demand that

$$e^{-np\left(\frac{\delta}{p}\right)^2/2} + e^{-np\left(\frac{\delta}{p}\right)^2/3} \leq \alpha$$

which implies that in order to get an accurate estimate, it suffices to have $n \in \Omega\left(\frac{-\ln \alpha}{\epsilon^2 p}\right)$.

3 Approximate counting of 0/1 Knapsack solutions

The last problem we will look at is how to count how many feasible solutions a Knapsack problem has. First, let us define what a Knapsack problem is: we are given n items with nonnegative integer weights $0 < a_1 \leq a_2 ... \leq a_n$, and a nonnegative integer capacity b. A feasible solution to the Knapsack problem is any selection of items whose sum of weights does not exceed the capacity, i.e. any vector $x \in \{0,1\}^n$ such that $a^\top x \leq b$.

Now, let K be the set of all feasible solutions; both optimizing linear functions over K and computing its cardinality are NP-hard problems. What we are going to present is, an algorithm by Dyer[1] that approximates the cardinality of K up to any desired degree of accuracy(of course, the running time will depend badly on the accuracy level). The main idea is the following: we will set up a sampling scheme that will return a feasible solution chosen uniformly at random with a probability that is not insignificant(i.e. at least inverse polynomial in n); after this is achieved, simply sampling enough times(i.e. to achieve the desired precision) and checking whether the returned solution is feasible will be enough for us to output a very good estimate.

So, let K' be a set of subsets of items(equivalently, a set of vectors $x \in \{0,1\}^n$) such that $\tilde{a}^\top x \leq n^2$, where $\tilde{a}_i = \lfloor \frac{a_1 n^2}{b} \rfloor$. Achieving the following requirements will be sufficient for our purposes:

a. K' is a superset of K.

b. We can compute $|K'|$.

c. We can sample some $x \in K'$ uniformly at random.

d. $\frac{|K|}{|K'|} \geq n + 1$.

It is straightforward to observe that if we can make these requirements hold, we will be able to estimate $|K|$ up to any desired precision by taking polynomially(in n) many samples from K' and outputting $p|K'|$, where p was the fraction of samples that actually belonged to K.

Now, let us focus on each of these requirements individually:

a. For any $x \in \mathbb{R}^n_{\geq 0}$(and in particular for any $x \in \{0,1\}^n$, we have that if $a^\top x \leq b$ then:
$$\sum_{i=1}^{n} \frac{a_i n^2}{b} x_i \leq n^2$$
and hence the first requirement holds.

b. The way we will compute $|K'|$ is through dynamic programming. Let $D(i,w)$ be the number of solutions in $|K'|$ using the first i items of total weight at most w. If we have computed all the values of D for $i' \leq i-1$

and $w' \leq w$, then $D(i,w) = D(i-1,w) + D(i-1,w-a_i)$, where the first term corresponds to not including item i in our solutions and the second corresponds to including it(we have made the implicit assumption that $D(i,w) = 0$ if $w < 0$). Then $|K'| = D(n,n^2)$.

c. The sampling will be done by performing backtracking on the computed table D; essentially, we start off at $D(n,n^2)$, and when we are at point $D(i,w)$ we choose to backtrack towards $D(i-1,w)(D(i-1,w-a_i))$ with probability $\frac{D(i-1,w)}{D(i-1,w)+D(i-1,w-a_i)}\left(\frac{D(i-1,w-a_i)}{D(i-1,w)+D(i-1,w-a_i)}\right)$.

d. To prove this, we will essentially show that there is a mapping from K' to K which corresponds at most $n+1$ solutions of K' to each solution in K; then the claim follows directly.

Consider $x \in K' \setminus K$; it must be then the case that there is some i such that $x_i = 1$ and $a_i > \frac{b}{n}$. Let $y \in \{0,1\}^n$ be such that $y_j = x_j$ if $j \neq i$, and 0 otherwise. We will show $y \in K$. Let $\delta \in [0,1]^n$ such that $\delta_j = \frac{a_j n^2}{b} - \lfloor \frac{a_j n^2}{b} \rfloor$(remember $\tilde{a}_j = \lfloor \frac{a_j n^2}{b} \rfloor$):

$$a^\top y = \frac{b}{n^2}\sum_{j=1}^{n} \tilde{a}_j y_j + \delta_j y_j = \frac{b}{n^2}(\tilde{a}^\top x - \tilde{a}_i + \delta^\top y) \leq b$$

where the last inequality follows from the following facts:

- $\tilde{a}^\top x \leq n^2$ since $x \in K'$.
- $\tilde{a}_i \geq n$ since $a_i > b/n$.
- $\delta^\top y \leq n$ since for all j $0 \leq \delta_j \leq 1$.

Now, consider the mapping $f : K' \to K$ that is derived from the above process plus the rule $f(x) = x$ for $x \in K$; this is a mapping for which it holds that for all $x \in K$ $|f^{-1}(x)| \leq n+1$. Hence the requirement follows.

This concludes the analysis of the four requirements for Dyer's algorithm; now, one can observe that choosing a sufficiently large(but at most polynomial) number of samples, one can estimate the true number of feasible Knapsack solutions up to any desired degree of precision.

References

[1] Martin Dyer, *Approximate counting by dynamic programming.* In Proceedings of the 35th ACM Symposium on Theory of Computing, 2003, pp. 693-699.