Advanced Algorithms

Class Notes for Thursday, October 11, 2012

1 Randomized Algorithms (R.A.)

1.1 Why

We have seen so far analysis for deterministic algorithms, where the user can devise an input that would yield a worst case behaviour. This includes the competitive analysis of online algorithms. Now if you want to make your algorithm opaque to the user, s.t. predicting a behavior is not possible, say like in cryptography, you use R.A.. This is achieved by including randomness in the behaviour of the algo itself. Mind you, having randomness in the input set is not the same as randomness in the behaviour itself!

The output of a R.A. is a random variable, which allows you only to predict its performance with a certain probability, so we speak about *expected running time*. This means that in most of the cases, when doing the analysis for your algorithm you will have to compute sums like $\sum_i p_i f(i)$, where p_i is the probability of instance i to occur. This can be cumbersome and most of the time imply work with binomial coefficients, thus average case analysis requires some familiarity with probanility concepts and manipulating such coefficients.

Sometimes coming up with a deterministic solution is not straightforward, because of not enough knowledge about the input. In such cases, by designing randomized algorithms that solve the problem, you can actually get, by derandomization, support for comming up with the deterministic approach(e.g. expanditure graphs). One other feature is that you can divide them in two categories: the Las Vegas (which will run until the right answer is returned, but it could take longer to converge) and the Monte Carlo ones, that will return an answer with a probability of error, but within a limited time interval.

1.2 Examples

Lets take an example of each and do some expectation analysis.

1.2.1 Randomized quicksort

The non randomized version would yield a worst case behaviour of $O(n^2)$ when the input is a decreasingly sorted list (which occurs with very low probability). The alternative average case behaviour would be the logarithmic time, $O(nlogn)$, when the pivot elt. splits the list in two equally sized sub partitions. On the other hand, when using the randomized version, you have always an $O(nlogn)$ expected time. Lets proove this.

First lets assume the uniformity of the input (meaning for a given input size n, all n! distinct input sequences are equally likely). Build the tree from a rand sequence say $k_1, k_2, \ldots, k_n$, by picking at random a root, say k_i. This splits our input set into a left subset with $i-1$ nodes, all smaller than k_i and a right subset of $n-2-i$ nodes all bigger than k_i.

A good estimate of the running time is given by the internal path length, I(n). It accounts for all the *n-1* possible sorting paths. The recurrence is:

$$I(n) = n-1+I(X)+I(n-1-X) \tag{1}$$

where $n-1$ is the contribution of the root node to all the n-1 paths. The probability of choosing node i, is $1/n$. Thus $E[X] = Pr[X = i] = \frac{1}{n}$ for $0 \leq i \leq n-1$. Taking the expectation on both sides, we get the average internal path length ($I_{av}(n) = E[I(X)]$):

$$\begin{aligned} E[I(n)] &= n-1+\frac{1}{n}\sum_{i=0}^{n-1}(E[I(i)]+E[I(n-i-1)]) \\ E[I(n)] &= n-1+\frac{2}{n}\sum_{i=0}^{n-1}(E[I(i)]) \end{aligned} \tag{2}$$

We have $E[I(0)] = E[I(1)] = 0$ and reduce the *full-history recurrence* from above, to a simpler form, by telescoping the difference $nI_{av}(n) - (n-1)I_{av}(n-1)$. We get:

$$\begin{aligned} nI_{av}(n) &= n(n-1)+2(I_{av}(0)+I_{av}(1)+\cdots+I_{av}(n-1))|\text{"}-\text{"} \\ (n-1)I_{av}(n-1) &= (n-1)(n-2)+2(I_{av}(0)+I_{av}(1)+\cdots+I_{av}(n-2)) \end{aligned} \tag{3}$$

$$\begin{aligned} nI_{av}(n)-(n-1)I_{av}(n-1) &= 2(n-1)+2I_{av}(n-1)|\text{"}: n(n+1)\text{"} \\ \frac{1}{n+1}I_{av}(n)-\frac{1}{n}I_{av}(n-1) &= 2\frac{n-1}{n(n+1)} \end{aligned} \tag{4}$$

Let $g(n) = \frac{1}{n+1}I_{av}(n)$, and substitute it in the above equation. We get $g(n)-g(n-1) = 2\frac{n-1}{n(n+1)}$, with $g(0) = 0$. After repeated substitutions on values of n, we get:

$$g(n) = 2\sum_{i=1}^{n}\frac{i-1}{i(i+1)} \tag{5}$$

Note that the term we use to sum over, is roughly of $\Theta(\frac{1}{i})$. So for $i > 3$, we can bound the harmonic term by:

$$\frac{1}{i+3} \leq \frac{i-1}{i(i+1)} \leq \frac{1}{i+2} \tag{6}$$

which yields $g(n) = \Theta(\sum_{i=1}^{n}\frac{1}{i})$. This is the *n-th harmonic number, H_n*, which (recall from calculus) is bounded by $\int_1^{n+1}(1/x)$ and $1+\int_1^n(1/x)$, thus $ln(n+1) \leq H_n \leq lnn+1 \Rightarrow g(n) = \Theta(logn)$ and thus $I_{av}(n) = \Theta(nlogn)$.

1.2.2 Matrix multiplication checking

Now lets consider the problem of checking the equality of 2 elts., x and y, drawn at random from a large universe U. Any reasonable model of computation would solve this problem in $log\ |U|$ time. Once the space is huge, the computation is complicated. An alternative to this is to pick a random mapping from U to a significantly smaller universe V s.t. x and y are identical with high probability iif their images are identical. This is called *fingerprinting*, since the images of x and y are their fingerprints and their equality can be verified in $log\ |V|$ time.

A common example for fingerprinting is to reduce the computational amount in the case of matrix multiplication. Reasonable algorithms run in $O(n^{2.376})$ –much better than the obvious $O(n^3)$ – but they assume complicated computations. The alternative is given by Freivalds' algorithm . It runs in $O(n^2)$ and returns an answer with a bounded error probability.

Theorem 1 (Error analysis)*: Let A,B and C be nxn matrices, s.t. $A\cdot B \neq C$. Then for the vector $\bar{r}$, chosen uniformly at random from $\{0,1\}^n$, then the probability of error is $Pr[A\cdot B\cdot \bar{r} = C\bar{r}] \leq 1/2$.*

For $D = A\cdot B - C$, implies $D \neq 0$, since the assumption of the problem is that $A\cdot B \neq C$. We want to ahow that the probability of error is less than $\frac{1}{2}$.

Let $y = A\cdot(B\cdot r)$ and $z = C\cdot r$. Then $y = z$ iif $D\cdot r = 0$. Let d be the first row of D. Without loss of generality we assume that some of the elts that make $D \neq 0$ are in the first row, and there are k of them – $k > 0$ entries $d_1, \ldots, d_k$ that are non zero. The first entry of the product $D\cdot r$ is equal to $d\cdot r$. A lower bound for the probability that this is non zero is also a lower bound for the probability that $D\cdot r \neq 0$. And this follows from the assumption that d has the k elts that are non zero.

To bound the error on the result Freivalds' algo returns, which is the case when the assumption we made – $A\cdot B \neq C$, turns out to be not true –$A\cdot B\cdot \bar{r} = C\bar{r}$, we get the answer by estimating the an upper bound of the probability that $d\cdot r = 0$, since this would involve with high probability that $D = 0$, which is against our assumption from above. $D = 0$ if and only if $\sum_{i=1}^{k} d_i\cdot r_i = 0$. We can rewrite this sum as $\frac{\sum_{i=1}^{k-1} d_i\cdot r_i}{d_k} = r_k$, which is possible since our $d_k \neq 0$. This equality will hold, for all the values that r_k can take (meaning 0 and 1). The probability of r_k taking one of the possible values, is $\frac{1}{|\{0,1\}|} = \frac{1}{2}$. (from this fact follows the observation below, on reducing the probability to $\frac{1}{|\mathcal{F}|}$). Since the quantity on the right handside of the equality can also take other values that r_k (the right handside can take any other value different from 0 and 1), the $Pr[d * r = 0] \leq 1/2$. Hence, $Pr[d * r \neq 0] \geq 1/2$ and therefore $Pr[D * r \neq 0] \geq 1/2$.

Corollary*: If $A\cdot B = C$, then Freivalds algorithm always gives the right answer. Otherwise, it gives the right answer with probability at least 1/2.*

An error probability of 1/2 is still quite large, but it can be decreased by choosing r as a random bit vector with values from $\mathcal{F}^n$. Then it follows from the condition that $r_k = -\frac{\sum_{i=1}^{k-1} d_i\cdot r_i}{dk}$ in the proof above, that$Pr[d\cdot r = 0] \leq 1/|\mathcal{F}|$ and therefore $Pr[A\cdot(B\cdot r) = C\cdot r] \leq 1/|\mathcal{F}|$. Hence, the error probability decreases to $1/|\mathcal{F}|$. We can also repeat the

random experiment, to achieve a tighter error probability. ($\mathcal{F}$ is any space that you want to conduct your computation on)

1.3 Tail bounds

Since with R.A. it is not predictable what's happening behind the scene, it might be the case that even if the expectation of the running time is small, that it assumes values that are far higher(farther from the mean). Take the example of bimodal distributions. We want to be able to say that the behaviour of an algo. is good almost all the time (e.g. "small running time with high probability" instead of "it has a small expectation"). We want to study that a probability deviates from its mean, by a given amount, and this is done with tail bounds.

1.3.1 Markov's Inequality

This gives us the tightest possible bound when we know only that a r.v. X takes non negative values, and what its expectation E[X] is.

$\mu_X = E[X] = \sum_x xPr[X = x]$, $Var[X] = E[(X - \mu_X)^2] = E[X^2] - \mu_X^2 = \sigma_X^2$, standard deviation is $\sigma_X = \sqrt{Var[X]}$.

Theorem 2 (Markov Inequality)*: Let X be a r.v. that assumes only non negative values. Then, for all $t > 0$,*

$$Pr[X \geq t] \leq \frac{E[X]}{t} \Leftrightarrow Pr[X \geq kE[X]] \leq \frac{1}{k} \tag{7}$$

Proof:

$$\begin{aligned} E[X] &= \sum_{x \geq t} xPr[X = x] + \sum_{x < t} xPr[X = x] \\ &\geq \sum_{x \geq t} xPr[X = x] \\ &\geq t \sum_{x \geq t} Pr[X = x] = tPr[X \geq t] \end{aligned} \tag{8}$$

This result can be improved, once we have more information on the distribution of the variable. Additional information about a variable is often expressed in terms of its *moments*. The expectation is also called the first moment. In more general terms, we define the moments as: the kth moment of r.v. X is $E[X^k]$.

1.3.2 Chebyshev Bounds

A significantly stronger bound can be computed when the second moment is also known, that allows us to compute the variance and standard deviation. Intuitively the standard deviation and the variance give us information about how far the r.v. is likely to be from its expectation.

Using the expectation and variance, one can get significantly stronger bounds, known as the Chebyshev's inequality.

Theorem 3 (Chebyshev Bound)*: For a r.v. X, with expectation μ_X and standard deviation σ_X, and variance $\sigma_X^2 = E[(X-\mu_X)^2] = E[X^2] - \mu^2$. For any $t > 0$,*

$$Pr[|X-\mu_X| \geq t\sigma_X] \leq \frac{1}{t^2} \Leftrightarrow Pr[|X-\mu_X| \geq t\sigma_X] \leq \frac{{\sigma^2}_X}{t^2{\sigma^2}_X} \tag{9}$$

Proof:

$$Pr[|X-\mu_X| \geq t\sigma_X] = Pr[(X-\mu_X)^2 \geq t^2\sigma_X^2] \tag{10}$$

Since $(X-\mu_X)^2$ is a non negative r.v., we can apply Markov's inequality, then $E[(X-\mu_X)^2] = \sigma_X^2$ and hence, we get the desired result.

1.3.3 Coin flipping

A single coin flip is seen as a Bernoulli trial, having only 2 outcomes: success with probability p and failure with probability q=*1-p*. We want to bound the probability of obtaining more than 3n/4 heads from n fair coin flips. Let X_i be an *Indicator function*, where $X_i = 1$ if the *ith* flip is head, otherwise it is 0, meaning the $E[X_i] = Pr[X_i = 1] = 1/2$. Then X is a r.v. for the total number of heads in n coin flip, $E[X] = \sum_i E[X_i] = n/2$.

Now, applying Markov's inequality we get that:

$$Pr[X \geq k] \leq \frac{E[X]}{k} = \frac{n}{2k} = \frac{2}{3} \tag{11}$$

Now if we want to compute the bound using Chebyshev's inequality, we need to compute the variance:

$$\begin{aligned} Var[X_i] = E[X^2] - (E[X])^2 = \frac{1}{2} - \frac{1}{4} = \frac{1}{4} \\ Var[X] = \sum_{i=1}^{n} Var[X_i] = \frac{n}{4} \end{aligned} \tag{12}$$

Note that $E[X^2] = E[X]$ only because X is a 0-1 r.v. and that we can actually substitute the variance of the sum, with the sum of the variances only because X_i are independent. Applying Chebyshev's inequality we get:

$$\begin{aligned} Pr[X \geq \frac{3n}{4}] &= Pr[X - \frac{n}{2} \geq \frac{n}{4}] \\ &\leq Pr[|X-\mu_X| \geq \frac{n}{4}] \\ &\leq \frac{Var[X]}{(\frac{n}{4})^2} = \frac{4}{n} \end{aligned} \tag{13}$$

Which actually means that X is either larger than $\frac{3n}{4}$ or smaller than $\frac{n}{2}$, due to the symmetry given by the absolute value. This means that the probability of X being greater than 3n/4 is $\frac{2}{n}$, a tighter bound than the one given by Markov's inequality.

1.3.4 Chernoff Bounds

Now, assuming that the expectation can be interchanged with the derivatives of the moment generating function, we get to compute expectation and variance in terms of $M_X[t] = E[e^{(tX)}]$. This you get from Taylor series expansion of e^X. This substitution is possible whenever the m.g.f. exists in the neighbourhood of 0 – which is the case for us. Some of the useful properties of the mg.f. is that say if you have 2 r.v. that have the same mg.f, you can say they have the same distribution. Or, if you recognize the function $M_X(t)M_Y(t)$ as the m.g.f. of a certain known distribution, then you can say that the 2 r.v.s have the same distribution.

Chernoff bounds are going to answer questions like "what's the probability for r.v. X to deviate from its mean, by $\delta\mu$ or more". You get them, by applying Markov's ineq. to e^tX, for a well chosen t.

The most commonly used version is for the sum of independent 0-1 r.v. , known as Poisson trials. The distribution of the r.v. in a Poisson trial, are not necessarily identical, but in our Bernoulli trial case they are. The difference between Poisson r.v. and Bernoulli r.v. is the assumption that for Bernoulli, each time the probability that heads come up are equal, p, while for Poisson, they differ, e.g. in round i, heads come up with p_i.

As an assumption so far, we considered the probabilities p_i to be equal, it is not always the case. Thus, when $p_i \neq 1/2$, it is the case of unbalanced tails. Because of this property of the distribution, there are 2 Chernoff bounds defined. The left and the right tail. Lets see how to derive them:

Lemma 1*: Let $X_1, \ldots, X_n$ be independent Poisson trials (X_i is a r.v. with Pr[X_i = 1] = p_i for some $0 < p_i < 1$), s.t. $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then for any $t > 0$,*

$$E[e^{tX}] \leq e^{(e^t-1)\mu}. \tag{14}$$

Proof:

$$\begin{aligned} E[e^{tX_i}] &= p_i e^{tx1} + (1-p_i)e^{tx0} = p_i e^t + (1-p_i) \\ &= 1 + p_i(e^t - 1) \end{aligned} \tag{15}$$

But we have the inequality that holds for any y, $1 + y \leq e^y$ and that $\mu = E[X] = E[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} E[X_i] = \sum_{i=1} np_i$. Hence $E[e^{tX_i}] \leq e^{p_i(e^t-1)}$. On the other hand,

$$\begin{aligned} E[e^{tX}] = E[e^{t\sum_{i=1}^{n} X_i}] &= E[\prod_{i=1}^{n} e^{tX_i}] \\ &= \prod_{i=1}^{n} E[e^{tX_i}] \\ &\leq \prod_{i=1}^{n} e^{p_i(e^t-1)} = e^{(e^t-1)\sum_{i=1}^{n} p_i} \end{aligned} \tag{16}$$

and this gives us the desired result.

***Theorem 4 (Chernoff Bound for upper tail)**: Let $X_1, \ldots, X_n$ be **independent** Poisson trials (X_i is a r.v. with $Pr[X_i = 1] = p_i$ for some $0 < p_i < 1$), s.t. $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then for any $\delta > 0$,*

$$Pr[X \geq (1+\delta)\mu] \leq (\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}})^{\mu}. \tag{17}$$

Proof: we apply Markov's inequality, for any t ¿ 0 and we get:

$$\begin{aligned} Pr[X \geq (1+\delta)\mu] &= Pr[e^{tX} \geq e^{t(1+\delta)\mu}] \\ &\leq \frac{E[e^{tX}]}{e^{t(1+\delta)\mu}} \\ &\leq \frac{e^{(e^t-1)\mu}}{e^{t(1+\delta)\mu}} [Lemma 1] \end{aligned} \tag{18}$$

Setting $t = ln(1+\delta)$, for $\delta > 0$ and consequently we have $e^t = 1+\delta$, we get:

$$Pr[X \geq (1+\delta)\mu] \leq e^{(\delta-(1+\delta)ln(1+\delta))\mu} \tag{19}$$

Using the Taylor series expansion of $ln(1+\delta)$, we get the bound on the upper tail:

$$Pr[X \geq (1+\delta)\mu] \leq e^{\frac{-\delta^2\mu}{3}} \tag{20}$$

It will answer questions like "how large should δ be in order that $Pr[X > (1+\delta)\mu]$ exceeds with a small probability of say 0.01".

The deviation of X *below* its expectation μ, is given by the lower bound, as the second Chernoff bound. The derivation is very simple and similar to the upper bound. A similar calculation by using the McLaurin expansion of $ln(1-\delta)$, will give you the lower bound:

$$Pr[X < (1-\delta)\mu] \leq e^{(-\frac{\delta^2\mu}{2})} \tag{21}$$

1.3.5 Coin toss with Chernoff bounds

As a final example, lets revisit our example on coin toss, and lets show that Chernoff bounds are indeed tighter predictions when choosing the right value for .

Remember example (1.3.3). To apply Chernoff bounds, we just have to pick a favorable δ which in this case will be $\delta = \frac{1}{2}$ and we obtain:

$$Pr[X \geq \frac{3n}{4}] \leq e^{-\frac{n}{24}} \tag{22}$$

Which obviously is the tightest among the three bounds. Why? Note that if you repeat the coin toss $\Theta(n)$ times with the Chebyshev bound you can still get a constant value, but once you can assume for instance not only pairwise but n wise independence of the trial, then you can apply the Chernoff bound and it will tell you that the probability of actually achieving more than $\frac{3n}{4}$ successes in n trials, is essentially close to 0. Why?

Similarly to the relation $(1-\frac{1}{n})^n = \frac{1}{e}$, in our example, $(1-\frac{1}{e^{\frac{n}{24}}})^{\Theta(n)} \approx 1$, where $\Theta(n) << e^{\frac{n}{24}}$, and this is the probability of no success. Thus, estimating the the ratio of number of trial to get the desired result is not obvious.