Random Select

497 - Randomized Algorithms

Sariel Har-Peled

September 12, 2002

1 Randomized Selection

We are given a set S of n distict elements, with an associated ordering. For $t \in S$, let $r_S(t)$ denote the rank of t (the smallest elelenmt in S has rank 1). Let $S_{(i)}$ denote the i-th element in the sorted list of S.

Given k, we would like to compute S_k (i.e., select the k-th element).

```
FUNC LazySelect( S, k )
  Input: S - set of n elements, k - index of element to be output.
begin
  repeat
      R ← {Sample with replacement of n^{3/4} elements from S} ∪ {−∞, +∞}.
      Sort R.
      l ← max(1, ⌊kn^{−1/4} − √n⌋), h ← min(n^{3/4}, ⌊kn^{−1/4} + √n⌋)
      a ← R_(l), b ← R_(h).
      Compute the rank r_S(a) of a and the rank r_S(b) of b in S (2n comparisons).
      P ← {y ∈ S | a ≤ y ≤ b} /* can be done while computing the
                                  rank of a and b */
  Until (r_S(a) ≤ k ≤ r_S(b)) and (|P| ≤ 8n^{3/4} + 2)
  Sort P in O(n^{3/4} log n) time.
  return P_{k−r_S(a)+1}
end LazySelect
```

Exercise 1.1 *Show how to compute the ranks of $r_S(a)$ and $r_S(b)$, such that the expected number of comparisions performed is $1.5n$.*

Lemma 1.2 `LazySelect` *succeeds with probability $\geq 1 - O(n^{-1/4})$ in the first iteration. And it performs only $2n + o(n)$ comparisons.*

Proof: One possible bad event is that $a > S_{(k)}$. Let X_i be an indicator variable which is 1 if the i-th sample is smaller equal to $S_{(k)}$, otherwise 0. WE have $p = \Pr[X_i] = k/n$, $q = 1 - k/n$, and let $X = \sum_{i=1}^{n^{3/4}} X_i$. Clearly, $X \sim B(n^{3/4}, k/n)$ (i.e., X has a binomial distribution with $p = k/n$, and $n^{3/4}$ trials).

By Chebyshev inequality

$$\Pr\left[|X - pn^{3/4}| \geq t\sqrt{n^{3/4}pq}\right] \leq \frac{1}{t^2}.$$

Since $pn^{3/4} = kn^{-1/4}$ and $\sqrt{n^{3/4}(k/n)(1-k/n)} \leq n^{3/8}/2$, we have that the probability of $a > S_{(k)}$ is

$$\Pr\left[X < (kn^{-1/4} - \sqrt{n})\right] \leq \Pr\left[|X - kn^{-1/4}| \geq 2n^{1/8} \cdot \frac{n^{3/8}}{2}\right] \leq \frac{1}{\left(2n^{1/8}\right)^2} = \frac{1}{4n^{1/4}}.$$

Thus, the probablity that $a > S_{(k)}$ is smaller than $1/(4n^{1/4})$. And similarly, the probablity that $b < S_{(k)}$ is smalelr than $1/(4n^{1/4})$.

So the only other source for a failure of the algorithm, is that the set P has more than $4n^{3/4} + 2$ elements. Let $I = \left\{S_{(k)}, S_{(k+1)}, \ldots, S_{(k+4n^{3/4})}\right\}$. Clearly, a is not in I, only if we pick less than $2\sqrt{n}$ elements from this interval into P. This, however, is $O(1/n^{1/4})$ using he same argumentation as above. Using a symetrical argument, we conclude that $P \subseteq \left\{S_{(k-4n^{3/4})}, S_{(k+1)}, \ldots, S_{(k+4n^{3/4})}\right\}$, with probability $\geq 1 - c/n^{1/4}$, where c is an appropriate constant. ∎

Any deterministic selection algorithm requires $2n$ comparisons, and `LazySelect` can be changes to require only $1.5n + o(n)$ comparisons (expected).

2 Two-Point Sampling

2.1 About Modulo Rings and Pairwise Independence

Let p be a prime number, and let $\mathbb{Z}_p = \{0, 1, \ldots, p-1\}$ denote the ring of integers modules p. Two integers a, b are equivalent modulo p, if $a \equiv p (mod p)$; namely, the reminder of dividing a and b by p is the same.

Lemma 2.1 *Given $y, i \in \mathbb{Z}_p$, and choosing a, b randomly and uniformly from $\mathbb{Z}_p$, the probablity of $y \equiv ai + b \pmod p$ is $1/p$.*

Proof: Imagine that we first choose a, then the required probablity, is that we choose b such that $y - ai \equiv b \pmod p$. And the probablity for that is $1/p$, as we choose b uniformly. ∎

Lemma 2.2 *Given $y, z, x, w \in \mathbb{Z}_p$, such that $x \neq w$, and choosing a, b randomly and uniformly from $\mathbb{Z}_p$, the probablity that $y \equiv ax + b \pmod p$ and $z = aw + b$ is $1/p^2$.*

Proof: This equivalent to claiming that the system of equalities $y \equiv ax + b \pmod p$ and $z = aw + b$ have a unique solution in a and b.

To see why this is true, substract one equation from the other. We get $y - z \equiv a(x - w) \pmod p$. Since $x - w \not\equiv 0 \pmod p$, it must be that there is a unique value of a such that the equation holds. This in turns, imply a specific value for b. ∎

Lemma 2.3 *Let i, j be two distinct elements of $\mathbb{Z}_p$. And choose a, b randomly and independetly from $\mathbb{Z}_p$. Then, the two random variables $Y_i = ai + b \pmod p$ and $Y_j = aj + b \pmod p$ are uniformly distributed on $\mathbb{Z}_p$, and are pairwise independent.*

Proof: The claim about the uniform distribution follows from Lemma 2.1, as $\Pr\left[Y_i = \alpha\right] = 1/p$, for any $\alpha \in \mathbb{Z}_p$. As for being pairwise indepedent, observe that

$$\Pr\left[Y_i = \alpha \,\middle|\, Y_j = \beta\right] = \frac{\Pr\left[Y_i = \alpha \cap Y_j = \beta\right]}{\Pr\left[Y_j = \beta\right]} = \frac{1/n^2}{1/n} = \frac{1}{n} = \Pr\left[Y_i = \alpha\right],$$

by Lemma 2.1 and Lemma 2.2. Thus, Y_i and Y_j are pairwise independent. ■

Remark 2.4 It is important to understand what independence between random variables mean: It means that having information about the value of X, gives you no infomration about Y. But this is only pairwise independence. Indeed, consider the variables Y_1, Y_2, Y_3, Y_4 defined above. Every pair of them are pairwise independent. But, if you give the value of Y_1 and Y_2, I know the value of Y_3 and Y_4 immediately. Indeed, giving me the value of Y_1 and Y_2 is enough to figure out the value of a and b. Once we know a and b, we immidately can compute all the Y_is.

Thus, the notion of independence can be extended k-pairwise independence of n random variables, where only if you know the value of k variables, you can compute the value of all the other variables. More on that later in the course.

Lemma 2.5 *Let $X_1, X_2, \ldots, X_n$ be pairwise independent random variables, and $X = \sum_{i=1}^{n} X_i$. Then* $\mathrm{var}\left[X\right] = \sum_{i=1}^{n} \mathrm{var}\left[X_i\right]$.

Proof: Observe, that

$$\mathrm{var}\left[X\right] = E\left[\left(X - E\left[X\right]\right)^2\right] = E\left[X^2\right] - \left(E\left[X\right]\right)^2.$$

Let X and Y be pairwise independent variables. Observe that $E\left[XY\right] = E\left[X\right]E\left[Y\right]$, as can be easily verfied. Thus,

$$\begin{aligned}
\mathrm{var}\left[X+Y\right] &= E\left[(X+Y-E[X]-E[Y])^2\right] \\
&= E\left[(X+Y)^2 - 2(X+Y)(E[X]+E[Y]) + (E[X]+E[Y])^2\right] \\
&= E\left[(X+Y)^2\right] - (E[X]+E[Y])^2 \\
&= E\left[X^2+2XY+Y^2\right] - (E[X])^2 - 2E[X]E[Y] - (E[Y])^2 \\
&= \left(E\left[X^2\right] - (E[X])^2\right) + \left(E\left[Y^2\right] - (E[Y])^2\right) + 2E\left[XY\right] - 2E[X]E[Y] \\
&= \mathrm{var}\left[X\right] + \mathrm{var}\left[Y\right] + +2E\left[X\right]E\left[Y\right] - 2E[X]E[Y] \\
&= \mathrm{var}\left[X\right] + \mathrm{var}\left[Y\right].
\end{aligned}$$

Using the above argumentation for several varaibles, isntead of just two, implies the lemma. ■

2.2 What is a randomzied algorithm? And how to save random bits?

We can consider a randomized algorithm, to be a deterministic algorithm $A(x,r)$ that receives together with the input x, a random string r of bits, that it uses to read random bits from. Let us redefine **RP**:

Definition 2.6 The class **RP** (for Randomized Polynomial time) consists of all languages L that have a deterministic algorithm $A(x,r)$ with worst case polynomial running time such that for any input $x \in \Sigma^*$,

- $x \in L \Rightarrow A(x,r) = 1$ for half the possible values of r.
- $x \notin L \Rightarrow A(x,r) = 0$ for all values of r.

LEt assume that we now want to minimize the number of random bits we use in the execution of the algorithm (Why?). If we run the algorithm t times, we have confidence 2^{-t} in our result, while using $t \log n$ random bits (assuming our random algorithm needs only $\log n$ bits in each execution). Simialrly, let us choose two random numbers from $\mathbb{Z}_n$, and run $A(x,a)$ and $A(x,b)$, gaining us only confidence $1/4$ i nthe correctness of our results, while requiring $2\log n$ bits.

Can we do better? Let us define $r_i = ai + b \mod n$, where a,b are random values as above (note, that we assume that p is prime), for $i = 1,\ldots,t$. Thus $Y = \sum_{i=1}^{t} A(x,r_i)$ is a sum of random variables whcih are pairwise independent, as the r_i are pairwise independent. Assume, that $x \in L$, then $E[Y] = t/2$, and $\sigma_Y^2 = \mathrm{var}\left[Y\right] = \sum_{i=1}^{t} \mathrm{var}\left[A(x,r_i)\right] \leq t/4$, and $\sigma_Y \leq \sqrt{t}/2$. The probablity that all those executions failed, corresponds to the event that $Y = 0$, and

$$\Pr\left[Y = 0\right] \leq \Pr\left[\left|Y - E\left[Y\right]\right| \geq \frac{t}{2}\right] = \Pr\left[\left|Y - E\left[Y\right]\right| \geq \frac{\sqrt{t}}{2} \cdot \sqrt{t}\right] \leq \frac{1}{t},$$

by the Chebyshev inequality. Thus we were able to “extract” from our random bits, much more than one would naturally suspect is possible.