Lecture 4: “Randomized selection”

Sotiris Nikoletseas
Professor

CEID - ETY Course
2017 - 2018
1. Preliminaries

(i) Boole’s inequality (or union bound)

Let random events $\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_n$. Then

$$Pr \left\{ \bigcup_{i=1}^{n} \mathcal{E}_i \right\} = Pr\{\mathcal{E}_1 \cup \mathcal{E}_2 \cup \cdots \cup \mathcal{E}_n\} \leq \sum_{i=1}^{n} Pr\{\mathcal{E}_i\}$$

Note: If the events are disjoint, then we get equality.
1. Preliminaries

(ii) Expectation (or Mean)

Let $X$ a random variable with probability density function (pdf) $f(x)$. Its expectation is:

$$\mu_x = E[X] = \sum_x x \cdot Pr\{X = x\}$$

If $X$ is continuous, $\mu_x = \int_{-\infty}^{\infty} x f(x) \, dx$
1. Preliminaries

(ii) Expectation (or Mean)

Properties:

- $\forall X_i \ (i = 1, 2, \ldots, n) : E \left[ \sum_{i=1}^{n} X_i \right] = \sum_{i=1}^{n} E[X_i]$
  
  This important property is called “linearity of expectation”.

- $E[cX] = cE[X]$, where $c$ constant

- if $X, Y$ stochastically independent, then
  $E[X \cdot Y] = E[X] \cdot E[Y]$

- Let $f(X)$ a real-valued function of $X$. Then
  $E[f(x)] = \sum_{x} f(x) Pr\{X = x\}$
1. Preliminaries

(iii) Markov’s inequality

**Theorem:** Let $X$ a non-negative random variable. Then, $\forall t > 0$

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

**Proof:** $E[X] = \sum_x x Pr\{X = x\} \geq \sum_{x \geq t} x Pr\{X = x\}$

$$\geq \sum_{x \geq t} t Pr\{X = x\} = t \sum_{x \geq t} Pr\{X = x\} = t \Pr\{X \geq t\}$

**Note:** Markov is a (rather weak) concentration inequality, e.g.

$$Pr\{X \geq 2E[X]\} \leq \frac{1}{2}$$

$$Pr\{X \geq 3E[X]\} \leq \frac{1}{3}$$

etc
1. Preliminaries

(iv) Variance (or second moment)

- Definition: \( \text{Var}(X) = E[(X - \mu)^2] \), where \( \mu = E[X] \)
  i.e. it measures (statistically) deviations from mean.

- Properties:
  - \( \text{Var}(X) = E[X^2] - E^2[X] \)
  - \( \text{Var}(cX) = c^2 \text{Var}(X) \), where \( c \) constant.
  - if \( X, Y \) independent, it is \( \text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) \)

Note: We call \( \sigma = \sqrt{\text{Var}(X)} \) the standard deviation of \( X \).
1. Preliminaries

(v) Chebyshev’s inequality

Theorem: Let $X$ a r.v. with mean $\mu = E[X]$. It is:

$$ Pr\{|X - \mu| \geq t\} \leq \frac{Var(X)}{t^2} \quad \forall t > 0 $$

Proof: $Pr\{|X - \mu| \geq t\} = Pr\{(X - \mu)^2 \geq t^2\}$

From Markov’s inequality:

$$ Pr\{(X - \mu)^2 \geq t^2\} \leq \frac{E[(X-\mu)^2]}{t^2} = \frac{Var(X)}{t^2} $$

Note: Chebyshev’s inequality provides stronger (than Markov’s) concentration bounds, e.g.

$$ Pr\{|X - \mu| \geq 2\sigma\} \leq \frac{1}{4} $$
$$ Pr\{|X - \mu| \geq 3\sigma\} \leq \frac{1}{9} $$

etc
2. The Randomized Selection Algorithm

- **The problem:** We are given a set $S$ of $n$ distinct elements (e.g. numbers) and we are asked to find the $k$th smallest.

- **Notation:**
  - $r_S(t)$: the rank of element $t$ (e.g. the smallest element has rank 1, the largest $n$ and the $k$th smallest has rank $k$).
  - $S_{(i)}$ denotes the $i$th smallest element of $S$ (clearly, we seek $S_{(k)}$ and $r_S(S_{(k)}) = k$).

- **Remark:** the fastest known deterministic algorithm needs $3n$ time and is quite complex. Also, any deterministic algorithm requires $2n$ time (a tight lower bound).
2. The basic idea: random sampling

we will randomly sample a subnet of elements from $S$, trying to optimize the following trade-off:

- the sample should be \underline{small enough} to be processed (e.g. ordered) in small time

- the sample should be \underline{large enough} to contain the $k$\textsuperscript{th} smallest element, with high probability
2. The Lazy Select Algorithm

1. Pick randomly uniformly, with replacement, a subset $R$ of $n^{3/4}$ elements from $S$.

2. Sort $R$ using an optimal deterministic sorting algorithm.

3. Let $x = k \cdot n^{-1/4}$.
   
   \[ l = \max \{ \lfloor x - \sqrt{n} \rfloor, 1 \} \]  
   \[ h = \min \{ \lceil x + \sqrt{n} \rceil, n^{3/4} \} \]  
   
   $a = R(l)$ and $b = R(h)$
   
   By comparing $a$ and $b$ to every element of $S$, determine $r_S(a), r_S(b)$.

4. If $k \in [n^{1/4}, n - n^{1/4}]$, let $P = \{ y \in S : a \leq y \leq b \}$.
   
   Check whether $S(k) \in P$ and $|P| \leq 4n^{3/4} + 2$. If not, repeat steps 1-3 until such a $P$ is found.

5. By sorting $P$, identify $P(k - r_S(a) + 1) = S(k)$. 
2. Remarks on the Lazy Select Algorithm

- In Step 1, sampling is done with replacement to simplify the analysis. Sampling without replacement is marginally faster but more complex to implement.
- Step 2 takes $O(n^{\frac{3}{4}} \log n)$ time (which is $o(n)$).
- Step 3 clearly takes $2n$ time ($2n$ comparisons). Graphically,

An example: assume $r_S(a) = 3$ and we want $S(7)$. In the sorted list of $P$ elements, $S(7) = P_{(k-r_S(a)+1)} = P_{7-3+1} = P_5$, i.e. the 5th element indeed.
2. Remarks on the Lazy Select Algorithm

- In Step 4, it is easy to check (in constant time) whether $S(k) \in P$ by comparing $k$ to (the now known) $r_S(a), r_S(b)$.

- In Step 5, sorting $P$ takes $O(n^{\frac{3}{4}} \log n) = o(n)$ time.

Note: we skip in Step 4 the (less interesting) cases where $k < n^{\frac{1}{4}}$ and $k > n - n^{\frac{1}{4}}$. Their analysis is similar.
2. When Lazy Select fails?

The algorithm may fail in Step 4, either because \( S(k) \notin P \) because \( |P| \) is large. We will show that the probability of failure is very small.

**Lemma 1.** The probability that \( S(k) \notin P \) is \( O(n^{-\frac{1}{4}}) \).

**Proof:** This happens if i) \( S(k) < a \) or ii) \( S(k) > b \).

i) \( S(k) < a \iff \) fewer than \( l \) (\( l = k \cdot n^{-\frac{1}{4}} - \sqrt{n} \)) of the samples in \( R \) are less than or equal to \( S(k) \). Let:

\[
X_i = \begin{cases} 
1, & \text{the } i\text{th random sample is at most } S(k) \\
0, & \text{otherwise}
\end{cases}
\]

Clearly, \( E(X_i) = Pr\{X_i\} = \frac{k}{n} \) and \( Var(X_i) = \frac{k}{n} \left(1 - \frac{k}{n}\right)\).

Let \( X = \sum_{i=1}^{\text{|R|}} X_i = \# \text{ samples in } R \text{ that are at most } S(k) \). Then
2. When Lazy Select fails?

\[
\mu_X = E[X] = |R| \cdot E[X_i] = n^{\frac{3}{4}} \frac{k}{n} = kn^{-\frac{1}{4}} \quad \text{and}
\]
\[
\sigma^2_X = Var[X] = \sum_{i=1}^{\frac{|R|}{n}} Var(X_i) = n^{\frac{3}{4}} \frac{k}{n} (1 - \frac{k}{n}) \leq n^{\frac{3}{4}} \frac{n}{4} \quad \text{(since the samples are independent)}
\]

Thus, \( Pr\{|X - \mu_X| \geq \sqrt{n}\} \leq \frac{\sigma^2_X}{n} \leq \frac{n^{\frac{3}{4}}}{4n} = O(n^{-\frac{1}{4}}) \)

\( \Rightarrow \) \( Pr\{X - \mu_X < -\sqrt{n}\} \leq O(n^{-\frac{1}{4}}) \)
\( \Rightarrow \) \( Pr\{X < \mu_X - \sqrt{n}\} = Pr\{X < \underbrace{kn^{-\frac{1}{4}} - \sqrt{n}}_{l} \} \leq O(n^{-\frac{1}{4}}) \)
ii) The case $S(k) > b$ is essentially symmetric (at least $h$ of the random samples should be smaller than $S(k)$), so

$$Pr\{S(k) > b\} = O(n^{-\frac{1}{4}})$$

Overall $Pr\{S(k) \notin P\} = Pr\{S(k) < a \cup S(k) > b\} = O(n^{-\frac{1}{4}}) + O(n^{-\frac{1}{4}}) = O(n^{-\frac{1}{4}})$
Lemma 2 The probability that \( P \) contains more than \( 4n^{\frac{3}{4}} + 2 \) elements is \( O(n^{-\frac{1}{4}}) \)

Proof: Very similar to the proof of Lemma 1: Let
\[
k_e = \max\{1, k - 2n^{\frac{3}{4}}\} \quad \text{and} \quad k_n = \min\{k + 2n^{\frac{3}{4}}, n\}
\]
If \( S(k_l) < a \) or \( S(k_h) > b \) then \( P \) contains more than \( 4n^{\frac{3}{4}} + 2 \) elements. For simplicity, let \( k_l = k - 2n^{\frac{3}{4}}, k_h = k + 2n^{\frac{3}{4}} \)
Then, it suffices to “simulate” the proof of Lemma 1 for \( k = k_l \) and then for \( k = k_h \).
2. The Lazy Select Algorithm

Theorem The Algorithm Lazy Select finds the correct solution with probability \(1 - O(n^{-\frac{1}{4}})\) performing \(2n + o(n)\) comparisons.

Proof: Due to Lemmata 1, 2 the Algorithm finds \(S_{(k)}\) on the first pass through steps 1-5 with probability \(1 - O(n^{-\frac{1}{4}})\) (i.e., it does not fail in Step 4 avoiding a loop to Step 1). Step 1 obviously takes \(o(n)\) time. Step 2 requires \(O(n^{\frac{3}{4}} \log n) = o(n)\) time, and Step 3 clearly needs \(2n\) comparisons (comparing each of the \(n\) elements of \(S\) to \(a\) and \(b\)). Overall the time needed is thus \(2n + o(n)\).