22C:199 Lecture 6

Scribe: Ganesh Venkataraman
10th September 2003

Chebyshev’s Inequality

\[ Pr[|X - E[X]| \geq t] \leq \frac{\text{var}[X]}{t^2} \]  

Chebyshev’s inequality is an example of a concentration result. The Chernoff-Hoeffding bounds that we will come up later are much stronger. We shall look at two applications of the Chebyshev’s inequality:

1. Second moment method in number theory

2. Randomized selection algorithm

Application 1
Consider the set \{2, 6, 9, 10\} and consider the 16 possible subsets. We claim that all the subsets have distinct sums. The above example can be generalized and stated as a problem below:

**Problem:** What is the size of the largest subset \(S \subseteq \{1, 2, \ldots, n\}\) that has all distinct sums?

For any subset \(A\) of integers, let

\[ s(A) = \sum_{x \in A} x \]  

\[ S(A) = \{s(X) | X \subseteq A\} \]  

\(A\) is said to have all distinct sums if \(|S(A)| = 2^{|A|}\). More precisely, we are looking for a natural number \(n\) such that there is a \(S \subseteq \{1, 2, \ldots, n\}\) of size \(f(n)\) that has all distinct sums, but there is no larger subset with this property. It is easy to see that \(\log_2 n\) is an easy lower bound since the set \(S = \{2^0, 2^1, \ldots, 2^{\log_2 n}\}\) has all distinct sums.

**Upper Bound**

Suppose the largest subset size is \(k\). Clearly \(2^k < kn\). Using this and the fact that \(k < n\), we get the following bound:

\[ f(n) < \log_2 n + \log_2 (\log_2 n) + 1 \]  

An open problem (with a fair amount of money involved, courtesy Erdos) is whether \(f(n) < \log_2 n + O(1)\).

By using Chebyshev’s inequality, we now prove the following theorem. (All logarithms are to the base 2 unless otherwise specified)

**Theorem:**

\[ f(n) < \log(n) + \frac{1}{2} \cdot \log(\log(n)) + O(1) \]  

1
**Proof:** Fix a subset \( \{a_1, a_2, \ldots, a_k\} \) of \( \{1, 2, \ldots, n\} \) that has all distinct sums. Let \( X_1, X_2, \ldots, X_k \) be independent random variables with \( Pr[X_1 = 1] = Pr[X_i = 0] = \frac{1}{2} \). Let \( X = \sum_{i=1}^{k} a_i X_i \). Note that all distinct sums of \( \{a_1, a_2, \ldots, a_k\} \) can be generated using this. The probability space contains all distinct sums of \( \{a_1, a_2, \ldots, a_k\} \) of size \( 2^k \). Each point is generated with probability \( \frac{1}{2^k} \).

\[
E[X] = \sum_{i=1}^{k} a_i E[X_i] = \frac{1}{2} \sum_{i=1}^{k} a_i
\]  

(6)

Our objective now is to compute the variance.

\[
(E[X])^2 = \frac{1}{4} (\sum_{i=1}^{k} a_i)^2
\]

(7)

\[
E[X^2] = E[2 \sum_{1\leq i < j \leq k} a_i X_i a_j X_j + \sum_{i=1}^{k} a_i^2 X_i^2]
\]

(8)

\[
\Rightarrow E[X^2] = \frac{1}{2} \sum_{1\leq i < j \leq k} a_i a_j + \frac{1}{2} \sum_{i=1}^{k} a_i^2
\]

\[
\Rightarrow var[X] = \frac{1}{4} \sum_{i=1}^{k} a_i^2 \leq \frac{n^2 k}{4}
\]

(9)

Denoting \( var[X] \) by \( \sigma \), we get \( \sigma \leq \frac{n \sqrt{(k)}}{2} \). Hence by Chebyshev’s inequality,

\[
Pr[|X - E[X]| \geq n \sqrt{(k)}] \leq \frac{n^2 k/4}{n^2 k} = \frac{1}{4}
\]

(11)

From the above inequality, we conclude that at least \( 3/4 * 2^k \) sums are contained in the range

\( (E[X] - n \sqrt{(k)}, E[X] + n \sqrt{(k}) \).

Since at most \( 2n \sqrt{(k)} \) integer sums can lie in this range we have the inequality

\[
\frac{3}{4} * 2^k < 2n \sqrt{(k)}.
\]

Solving this for \( k \) in terms of \( n \), we get the bound claimed in the theorem. \( \square \)

**Application 2: SELECTION**

**Input** Sequence \( S \) of \( n \) integers and an integer \( 1 \leq k \leq n \)

**Output** \( k \)th largest element in \( S \)

There exists a deterministic linear time algorithm that does this. However, the algorithm is seldom used in practice since the constants hidden inside the “big Oh” expression are high. We describe a randomized algorithm which has the same expected run-time, but is simpler to implement and is makes fewer pairwise comparisons. **Lazy Sort**

1. Pick \( n^{3/4} \) elements from \( S \) independently and uniformly at random with replacement into \( R \).
2 Sort $R$. Let $R_l$ denote the $l^{th}$ smallest element in $R$. Let $r_s(q)$ denote the rank of an element $q$ in set $S$.

3 Let $x = k n^{-1/4}$, $l = \max\{\lfloor x - \sqrt{n} \rfloor, 1\}$, $h = \min\{\lceil x + \sqrt{n} \rceil, n^{3/4}\}$, $a = R_l$ and $b = R_h$.
By comparing every element in $S$ with $a$ determine $r_s(a)$. Similarly determine $r_s(b)$.

4 If $k < n^{1/4}$, then $P = \{y \in S \mid y \leq b\}$.
If $k > n - n^{1/4}$, then $P = \{y \in S \mid y \geq a\}$.
If $k \in [n^{1/4}, n - n^{1/4}]$, then $P = \{y \in S \mid a \leq y \leq b\}$

5 Check if $S_k \in P$ and $|p| \leq 4n^{3/4}$ + 2 otherwise repeat [1] to [3].

6 Sort $P$ and return $P_{(k-r_s(a)+1)}$.

We shall analyze the expected run-time of the above algorithm using Chebyshev’s inequality.