CSIS0351/8601: Randomized Algorithms
Lecture 5: More Measure Concentration: Counting DNF Satisfying Assignments, Hoeffding's Inequality
Lecturer: Hubert Chan
Date: 4 Oct 2010

These lecture notes are supplementary materials for the lectures. They are by no means substitutes for attending lectures or replacement for your own notes!

1 Approximate Counting by Random Sampling

Suppose there is a bag containing red balls and blue balls. You would like to estimate the fraction of red balls in the bag. However, you are only allowed to sample randomly from the bag with replacement. The straightforward method is to make T random samples and use the fraction of red balls in the random samples as an estimate of the true fraction. The question is: how many samples are enough so that with probability at least $1-\delta$, our estimate is within a multiplicative error of $0 < \epsilon < 1$ from the true value?

Let $Y := \sum_{\tau \in [T]} Z_\tau$, where Z_τ is the $\{0,1\}$-random variable that takes value 1 if the τth sample is red. The estimator is $\frac{Y}{T}$. Suppose the true fraction of red balls in the bag is p. Then, by Chernoff Bound,

$$Pr[|\frac{1}{T}Y - p| \geq \epsilon p] = Pr[|Y - E[Y]| \geq \epsilon E[Y]] \leq 2\exp(-\frac{1}{3}\epsilon^2 Tp).$$

In order for this probability to be less than δ, it is enough to have $T = \Theta(\frac{1}{p\epsilon^2}\log\frac{1}{\delta})$. If p is very small, then we would need a lot of samples.

1.1 Counting DNF Satisfying Assignments

Definition 1.1 *A formula ϕ in disjunctive normal form consists of disjunction of clauses in n Boolean variables $x_1, x_2, \ldots, x_n$: $\phi = \vee_{j=0}^{m-1} C_j$, where each clause C_j is a conjunction of literals. Given an assignment of variables, a DNF formula is satisfied if at least one clause C_j is satisfied. A clause C_j is satisfied if all literals in C_j evaluate to TRUE.*

Observe that while it is hard to find a satisfying assignment for a CNF formula, it is easy to satisfy a DNF formula. However, the problem of finding the total number of satisfying assignments for a DNF formula is #P complete, which means it is believed that there is no polynomial time algorithm that can give the exact number of satisfying assignments.

Is it possible to get an estimate with multiplicative error ϵ by random sampling?

Note that there are totally 2^n assignments and an assignment can be sampled uniformly at random with n independent bits. Note that if there are L satisfying assignments, then the fraction of satisfying assignments is $p := \frac{L}{2^n}$. From the discussion above on estimating the fraction of red balls, we know that to get an estimate with multiplicative error ϵ and failure probability at most δ, the required number of samples can be as large as $\Theta(\frac{2^n}{L\epsilon^2} \log \frac{1}{\delta})$.

Hence, if L is small, this is no better than trying out all 2^n possible assignments. Moreover, since we do not know the value of L, we cannot even tell how many samples are enough in the first place!

1.2 Importance Counting

Our problem is that the number L of satisfying assignments could be small compared to 2^n, the total number of assignments. Hence, sampling uniformly from all possible assignments is wasteful. We can do better if we only sample "useful" assignments.

For each clause C_j, let S_j be the set of assignments that satisfy C_j. Observe that if C_j contains d literals, then in order to satisfy C_j, those d corresponding variables have no freedom, while the rest of the variables can take any values. Hence, in this case $|S_j| = 2^{n-d}$, and we can see it is easy to sample from S_j: we just need $n - d$ independent random bits.

For each j, we define $\widehat{S_j} := \{(j, a) : a \in S_j\}$. For each assignment satisfying C_j, we attach an extra tag specifying that it satisfies clause C_j. Let $S := \cup_{j \in [m]} S_j$ and $\widehat{S} := \cup_{j \in [m]} \widehat{S_j}$. Also, let $L := |S|$ and $N := |\widehat{S}|$ So, an assignment appears in $\widehat{S}$ as many times as the number of clauses it satisfies. Observe the following.

1. For each j, we can compute $|\widehat{S_j}| = |S_j|$, and hence we can compute $N := |\widehat{S}|$. However, it is $L := |S|$ that we want to estimate.

2. We can generate a uniform random sample from $\widehat{S_j}$ by first generating a sample from S_j.

3. Observe that $p := \frac{L}{N} \geq \max_j \frac{|S_j|}{\sum_k |S_k|} \geq \frac{1}{m}$.

Remark 1.2 Suppose we can sample a $\{0, 1\}$-random variable X that takes value 1 with probability exactly $p := \frac{L}{N}$. Then, by Chernoff Bound, with failure probability at most δ, the value of p and hence that of $|S|$ can be estimated with multiplicative error ϵ by drawing $\Theta(\frac{1}{p\epsilon^2} \log \frac{1}{\delta})$ samples of X. Since we know $\frac{1}{p} \leq m$, we conclude that $O(\frac{m}{\epsilon^2} \log \frac{1}{\delta})$ samples are enough, even though we do not know the value of p.

Sampling Uniformly at Random from $\widehat{S}$. Note that a pair $(j, a) \in \widehat{S}$ can be sampled uniformly at random by the following procedure.

1. Pick $j \in [m]$ with probability $\frac{|\widehat{S_j}|}{|\widehat{S}|}$.

   The expected number of random bits required is $O(\log_2 |\widehat{S}|) = O(n + \log m)$

2. Pick a satisfying assignment $a \in S_j$ uniformly at random. The number of random bits required is at most n.

Sampling $X \in \{0, 1\}$ with mean $\frac{L}{N}$. We are going to describe how to sample such a random variable X. First, define for each assignment $a \in S$, $\overline{j}(a) := \min\{j : a \in S_j\}$. If $a \in S_j$, this means a satisfies at least one clause, and $\overline{j}(a)$ corresponds to the clause with the smallest index that it satisfies. Note that given a, we just need to test, starting from the smallest j, whether it satisfies C_j. This takes $O(nm)$ time. Here is the description of the sampling procedure.

1. Pick (J, A) uniformly at random from $\widehat{S}$.
2. If $J = \overline{j}(A)$, then set $X := 1$; otherwise, set $X := 0$.

Claim 1.3 $Pr[X = 1] = \frac{L}{N}$.

Proof: Let $N := \widehat{S}$. Suppose we index the assignments in S by $\{a_l : l \in [L]\}$. For each l, let N_l be the number of clauses that a_l satisfies. Note that $N = \sum_{l \in [L]} N_l = \sum_{j \in [m]} |S_j|$.

$$
\begin{aligned}
Pr[X = 1] &= Pr[J = j(A)] \\
&= \sum_{l \in [L]} Pr[A = a_l \wedge J = j(A)] \\
&= \sum_{l \in [L]} Pr[A = a_l] \cdot Pr[J = j(A) | A = a_l] \\
&= \sum_{l \in [L]} \frac{N_l}{N} \cdot \frac{1}{N_l} \\
&= \frac{L}{N}
\end{aligned}
$$

■

2 Hoeffding's Inequality

We study another measure concentration inequality.

Theorem 2.1 *Suppose $X_1, X_2, \ldots, X_n$ are independent real-valued random variables, such that for each i, X_i takes values from the interval $[a_i, b_i]$. Let $Y := \sum_i X_i$. Then, for all $\alpha > 0$,*

$Pr[|Y - E[Y]| \geq n\alpha] \leq 2\exp(-\frac{2n^2\alpha^2}{\sum_i R_i^2})$,

where $R_i := b_i - a_i$.

Note that the main differences from the Chernoff Bound are:

1. We are giving an upperbound for the event $|\frac{Y}{n} - E[\frac{Y}{n}]| \geq \alpha$. Hence, when all X_i are identically distributed, $\frac{Y}{n}$ is an estimator for $E[X_i]$, and we are dealing with additive error (as opposed to multiplicative error).
2. Each random variable X_i takes continuous values (as opposed to $\{0, 1\}$).
3. The mean of X_i does not appear in the upperbound. This would turn out to be useful for sampling, when we do not know the mean of X_i in advance.

Again, we use the technique of moment generating function to prove the Hoeffding's Inequality. For simplicity, we prove a slightly weaker result:

$Pr[|Y - E[Y]| \geq n\alpha] \leq 2\exp(-\frac{n^2\alpha^2}{2\sum_i R_i^2})$.

Once again, here are the three steps.

2.1 Transform the Inequality into a Convenient Form

We use the inequality $Pr[|Y - E[Y]| \geq n\alpha] \leq Pr[Y - E[Y] \geq n\alpha] + Pr[Y - E[Y] \leq -n\alpha]$, and give bounds for the two probabilities separately. Here, we consider $Pr[Y - E[Y] \geq n\alpha]$. The other case is similar.

Observe that the expectation of Y does not appear in the upperbound. It would be more convenient to first translate the variable X_i. Define $Z_i := X_i - E[X_i]$. Observe that $E[Z_i] = 0$. Moreover, since both X_i and $E[X_i]$ are in the interval $[a_i, b_i]$, it follows that Z_i takes values in the interval $[-R_i, R_i]$.

For $t > 0$, we have

$Pr[Y - E[Y] \geq n\alpha] = Pr[t\sum_i Z_i \geq tn\alpha]$.

2.2 Using Moment Generating Function and Independence

Applying the exponentiation function to both sides of the inequality, we follow the standard calculation, using independence of the Z_i's.

$$\begin{aligned}
Pr[t\sum_i Z_i \geq tn\alpha] &= Pr[\exp(t\sum_i Z_i) \geq \exp(tn\alpha)] \\
&\leq \exp(-tn\alpha) \cdot E[\exp(t\sum_i Z_i)] \\
&\leq \exp(-tn\alpha) \cdot \prod_i E[\exp(tZ_i)]
\end{aligned}$$

The next step is the most technical part of the proof. Recall that we want to find some appropriate function $g_i(t)$ such that $E[\exp(tZ_i)] \leq \exp(g_i(t))$. All we know about Z_i is that $E[Z_i] = 0$ and Z_i takes value in $[-R_i, R_i]$.

Think of a simple (but non-trivial) random variable that has mean 0 and takes values in $[-R_i, R_i]$. Consider $\widehat{Z}_i$ that takes value R_i with probability $\frac{1}{2}$ and $-R_i$ with probability $\frac{1}{2}$. Then, it follows that $E[\exp(t\widehat{Z}_i)] = \frac{1}{2}(e^{tR_i} + e^{-tR_i}) \leq \exp(\frac{1}{2}t^2R_i^2)$, the last inequality follows from the fact that for all real x, $\frac{1}{2}(e^x + e^{-x}) \leq e^{\frac{1}{2}x^2}$.

Therefore, for such a simple $\widehat{Z}_i$, we have a nice bound $E[\exp(t\widehat{Z}_i)] \leq \exp(g_i(t))$, where $g_i(t) := \frac{1}{2}t^2R_i^2$.

Using Convexity to Show that the Extreme Points are the Worst Case Scenario. Intuitively, $\widehat{Z}_i$ is the worst case scenario. Since we wish to show measure concentration, it is a bad case if there is a lot of variation for Z_i. However, we have the requirement that $E[Z_i] = 0$ and $Z_i \in [-R_i, R_i]$. Hence, intuitively, Z_i has the most variation if it takes values only at the extreme points, each with probability $\frac{1}{2}$. We formalize this intuition using the convexity of the exponentiation function.

Definition 2.2 *A function $f : \mathbb{R} \to \mathbb{R}$ is convex if for all $x, y \in R$ and $0 \leq \lambda \leq 1$,*

$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$.

Fact 2.3 *If a function is doubly differentiable and $f''(x) \geq 0$, then f is convex.*

We use the fact that the exponentiation function $x \mapsto e^{tx}$ is convex. First, we need to express Z_i as a convex combination of the end points, i.e., we want to find $0 \leq \lambda \leq 1$ such that $Z_i = \lambda R_i + (1-\lambda)(-R_i)$. We have $\lambda := \frac{Z_i + R_i}{2R_i}$.

Using the convexity of the function $x \mapsto e^{tx}$, we have

$$\begin{aligned}
\exp(tZ_i) &= \exp(t(\lambda R_i + (1-\lambda)(-R_i))) \\
&\leq \lambda \exp(tR_i) + (1-\lambda)\exp(-tR_i) \\
&= (\frac{1}{2} + \frac{Z_i}{2R_i}) \exp(tR_i) + (\frac{1}{2} - \frac{Z_i}{2R_i}) \exp(-tR_i).
\end{aligned}$$

Take expectation on both sides, and observing that $E[Z_i] = 0$, we have

$E[\exp(tZ_i)] \leq \frac{1}{2}(\exp(tR_i) + \exp(-tR_i)) = E[\exp(t\widehat{Z}_i)]$, as required.

2.3 Picking the Best Value for t

We have shown that for all $t > 0$,

$Pr[Y - E[Y] \geq n\alpha] \leq \exp(\frac{1}{2} t^2 \sum_i R_i^2 - n\alpha t)$, and we want to find a value of t that minimizes the right hand side.

Note that the exponent is a quadratic function in t and hence is minimized when $t := \frac{n\alpha}{\sum_i R_i^2} > 0$. Hence, in this case, we have

$Pr[Y - E[Y] \geq n\alpha] \leq \exp(-\frac{n^2\alpha^2}{2\sum_i R_i^2})$, as required.

2.4 Estimating the Fraction of Red Balls

Using the Hoeffding's Inequality, with $X_i \in [0,1]$ and $E[X_i] = p$, we have

$Pr[|\frac{1}{T}\sum_i X_i - p| \geq \alpha] \leq 2\exp(-2T\alpha^2)$.

Hence, in order to estimate the fraction of red balls with additive error at most α and failure probability at most δ, it suffices to use $T = \Theta(\frac{1}{\alpha^2} \log \frac{1}{\delta})$.

3 Homework Preview

1. **Integration by Sampling.** Suppose we are given an integrable function $f : [0,1] \to [0, M]$,

and we wish to estimate the integral $I(f) := \int_0^1 f(x)dx$. We only have *black box access* to the function f: this means that we are given a box such that when we provide it with a real number x, the box returns the value $f(x)$. Moreover, we assume the real computation model. In particular, we assume that storing a real number takes constant space, and basic arithmetic and comparison operator ($\leq$) take constant time. Suppose we are also given a random number generator **Rand[0,1]** that returns a number uniformly at random from [0,1], and subsequent runs of **Rand[0,1]** are independent. The goal is to design an algorithm that given black box access to a function $f : [0,1] \to [0,M]$ and parameters $0 < \epsilon, \delta < 1$, return an estimate of $I(f)$ with additive error at most ϵ and failure probability at most δ.

(a) Show that this is not achievable by a deterministic algorithm. In particular, show that for any deterministic algorithm A, there is some function f such that the algorithm A returns an answer with additive error $\frac{M}{2}$.

(b) Using the random generator **Rand[0,1]**, design a randomized algorithm to achieve the desired goal. Give the number of black box accesses to the function f and the number of accesses to **Rand[0,1]** used by your algorithm.

2. **Estimating the (Unknown) Fraction of Red Balls.** Suppose a bag contains an unknown number of red balls (assume there is at least one red ball) and you are only allowed to sample (with replacement) uniformly at random from the bag. Design an algorithm that, given $0 < \epsilon, \delta < 1$, with failure probability at most δ, returns an estimate of the fraction of red balls with **multiplicative** error at most ϵ, i.e., if the real fraction is p, the algorithm returns a number $\widehat{p}$ such that $|\widehat{p} - p| \leq \epsilon p$. Give the number of random samples used by your algorithm.