COMS 6998: Randomness in Computing — Oct 8, 2012

Set Cover and Chernoff Bound

Instructor: *Xi Chen* — Scribes: *Chun Ye*

1 Randomized Algorithm for Set Cover

Given a set $U = \{e_1, \ldots, e_n\}$, a collection of subsets $S_1, \ldots, S_m \subseteq U$ such that $S_1 \cup S_2 \cup \ldots \cup S_m = U$, and $\{c_i\}$ where each c_i denotes the cost associated with the subset S_i, the set cover problem asks us to choose a subcollection $\{S_{i'}\}$ of the subsets with the minimum cost such that $S_{1'} \cup S_{2'} \cup \ldots \cup S_{k'} = S$.

The set cover problem is known to be NP-hard. In the first part of the lecture, we consider a randomized approximation algorithm for the set cover problem. The algorithm first solves a linear programming relaxation of an integer programming formulation of the problem and rounds the LP solution in some way via randomization. We will show that the algorithm terminates (with high probability) in polynomial times and gives a $O(logn)$ approximation (with high probability) of the optimal solution. By a $O(logn)$ approximation we mean that there exists some constant C such that for n sufficiently large, $C(logn)(OPT) \geq Alg$, where Alg denotes the cost of the solution found by the approximation algorithm and OPT denotes the cost of an optimal solution.

Let x_i equal 1 if S_i is included in the subcollection and 0 otherwise, then the following integer program solves the set cover problem:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} c_i x_i \\ \text{s.t.} \quad & \sum_{i: e \in S_i} x_i \geq 1 \quad \forall e \in U \\ & x \in \{0,1\}^m \end{aligned}$$

Notice that the constraints $\sum_{i:e\in S_I} x_i \geq 1 \quad \forall e \in U$ requires each element in U to be covered by some subset S_j in the subcollection chosen. Since an integer program in general is hard to solve, we wil consider the following LP relaxation of the above IP:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} c_i x_i \\ \text{s.t.} \quad & \sum_{i: e \in S_i} x_i \geq 1 \quad \forall e \in U \\ & x \in [0,1]^m \end{aligned}$$

An optimal solution of the above LP can be found in polynomial time (via an interior point algorithm for example). Notice that the optimal cost of the LP lower bounds that of the IP because we minimize the same objective function over a bigger feasible set in the LP. Having obtained an optimal LP solution x^*, we need to figure out how to round this solution to a feasible IP solution. First we present a deterministic rounding scheme. Before we do so, we need to introduce the notion of the frequency of an element.

Definition 1. *Given $u_j \in U$ for some $1 \leq j \leq n$, the frequency of u_j, denoted by F_j, is the number of subsets S_i such that $u_j \in S_i$.*

Let $F = max\{F_1, \ldots, F_n\}$, and we are ready to describe the deterministic algorithm.

Solve the LP relaxation of the set cover IP to obtain an optimal solution x^*.
For each x_i^*, $i = 1, \ldots, m$.
Set $y_i = 1$ if $x_i^* \geq \frac{1}{F}$.
Set $y_i = 0$ otherwise.

Proposition 1. *$y = (y_1, \ldots, y_m)$ obtained from the above rounding scheme is feasible for the integer program and that the rounding solution we found is an F-approximation of OPT.*

Proof. For the sake of contradiction, suppose y is not a feasible solution of the IP, then for some $e \in U$, we must have that $\sum_{i:e \in S_i} y_i < 1$. Since $y_i = \{0, 1\}$, we must have that $y_i = 0 \ \forall i : e \in S_i$, which implies that $x_i^* < \frac{1}{F} \ \forall i : e \in S_i$. Consequently, we must have that $\sum_{i:e \in S_i} x_i^* < \sum_{i:e \in S_i} \frac{1}{F} \leq \frac{F_e}{F} \leq 1$, which contradicts the feasiblity of x^*. Hence, y is a feasbile solution of the IP. To obtain the desired approximation ratio, we observe that $y_i \leq F x_i$ for all $i = 1, \ldots, m$. Summing everything up, we have that $\sum_{i=1}^m c_i y_i \leq F \sum_{i=1}^m c_i x_i = F(OPT(LP)) \leq F(OPT)$ □

Remark: notice that this deterministic algorithm gives a good approximation when F is small. In the case of the vertex cover problem, U is the set of edges and we can view each vertex as a subset of U in terms of the edges incident to it. Consequently, since each edge is incident to exactly two vertices, $F_i = 2$ for all $e \in U$, which means that the above algorithm is a 2-approximation algorithm for the vertex cover problem.

Now we will present the randomized approximation algorithm. Consider the following subroutine of the algorithm.

For each x_i^*, $i = 1, \ldots, m$.
Set $y_i = 1$ w.p. x_i^*.
Set $y_i = 0$ otherwise.

Notice that $E[\sum_{i=1}^m c_i y_i] = \sum_{i=1}^m c_i E[y_i] = \sum_{i=1}^m c_i x_i^* = OPT(LP)$. Hence, fixing some large M, we have that $P(\sum_{i=1}^m c_i y_i \geq M(OPT(LP))) \leq \frac{1}{M}$ from Markov's inequality. This implies that with high probability, $Alg \leq O(1)OPT(LP) \leq O(1)OPT$, which implies that we obtain a $O(1)$-approximation ratio with high probability. However, the issue with running this subroutine just once is that the rounded vector y we get might be infeasible for the IP. We overcome this by running the subroutine for more than $log(n)$ iterations.

Solve the LP relaxation of the set cover IP to obtain an optimal solution x^*.
Initialize y to the 0 vector and repeat the following for $log(n) + M$ times for some fix constant M large.
For each x_i^*, $i = 1, \ldots, m$.

4: Set $y_i = 1$ w.p. x_i^*.

Let y_i^j be the value of y_i at the j$^{\text{th}}$ iteration of the algorithm. To analyze the approximation ratio, notice that the value of y_i at the end of the algorithm, denote by y_i^*, is set to $max(y_i^1, y_i^2, \ldots, y_i^{log(n)+M})$. Consequently, we have that $E[\sum_{i=1}^m c_i y_i^*] \leq E[\sum_{i=1}^m c_i \sum_{j=1}^{logn+M} y_i^j] = \sum_{i=1}^m \sum_{j=1}^{logn+M} c_i E[y_i^j] = (log(n) + M) \sum_{i=1}^m c_i x_i^* = (log(n)+M) OPT(LP)$. Consequently, from Markov's inequality, we can conclude that the randomized algorithm gives a $O(log(n))$-approximation of OPT with high probability.

To see that y^* is feasible with high probability. We have that for each $u \in U$, $P(\sum_{i:e\in S_i} y_i^* < 1) = (\prod_{i:e\in S_i}(1-x_i))^{log(n)+M}$. Notice that $\max \prod_{i:e\in S_i}(1-x_i)$ s.t. $\sum_{i:e\in S_i} x_i = 1$ occurs when $x_i = \frac{1}{F_e}$ $\forall i$. Thus, we have that

$$(\prod_{i:e\in S_i}(1-x_i))^{log(n)+M} \leq (1-\frac{1}{F_e})^{F_e(log(n)+M)} \simeq e^{-F_e(log(n)+M)} \leq e^{-(log(n)+M)} = \frac{e^{-M}}{n}$$

Hence, we get that

$$P(y^* \text{is feasible}) = 1 - P(\cup_{e\in U}(\sum_{i:e\in S_i} y_i^* < 1)) \geq 1 - \sum_{e\in U} P(\sum_{i:e\in S_i} y_i^* < 1) \geq 1 - \frac{ne^{-M}}{n} = 1 - e^{-M}$$

which is close to 1 when M is large.

2 Chernoff Bound

In addition to Markov and Chebyshev inequalities, Chernoff bound also gives us a way to bound large deviation of a random variable from its mean provided that the moment generating function of the random variable exists. The bound we get from Chernoff is usually tighter than that from Markov or Chebyshev inequality as the decay we get is an exponential function of the deviation from the mean.

Consider i.i.d bernoulli random variables $X_1, \ldots, X_n$, if we bound the large deviation probability of $\sum_{i=1}^n X_i$ using Chebyshev inequality, we get $P(\sum_{i=1}^n X_i > (\frac{1}{2}+\delta)n) \leq \frac{var(\sum_{i=1}^n X_i)}{(\delta n)^2} = O(\frac{1}{n})$. We can do much better using Chernoff bound.

Proposition 2. *Let $X_1, \ldots, X_n$ be independent random variable such that for all $t > 0$, $E[e^{tX_i}]$ is finite for all i. Let $X = \sum_{i=1}^n X_i$ and $\mu = E[X]$, then we have that $P(X \geq (1+\delta)\mu) \leq \frac{\prod_{i=1}^n E[e^{tX_i}]}{e^{t(1+\delta)\mu}}$*

Proof. For $t > 0$, we have that $P(X \geq (1+\delta)\mu) = P(e^{tX} \geq e^{t(1+\delta)\mu}) \leq \frac{E[e^{tX}]}{e^{t(1+\delta)\mu}} = \frac{E[\prod_{i=1}^n e^{tX_i}]}{e^{t(1+\delta)\mu}} = \frac{\prod_{i=1}^n E[e^{tX_i}]}{e^{t(1+\delta)\mu}}$ where the inequality follows from Markov inequality and the last equality follows from the independence of X_i's. □

When $X_1, \ldots, X_n$ are independent random variables where $X_i = 1$ with probability p_i and $X_i = 0$ with probability $1-p_i$, then we have that $E[e^{tX_i}] = p_i e^t + (1-p_i) = 1 + p_i(e^t - 1) \leq e^{p_i(e^t-1)}$, which implies that $E[e^{tX}] \leq \prod_{i=1}^n e^{p_i(e^t-1)} = e^{u(e^t-1)}$. Therefore, we have that $P(X \geq (1+\delta)\mu) \leq \frac{\prod_{i=1}^n E[e^{tX_i}]}{e^{t(1+\delta)\mu}} \leq \frac{e^{u(e^t-1)}}{e^{t(1+\delta)\mu}}$.

To obtain the tightest bound possible, we minimize the expression $\frac{e^{u(e^t-1)}}{e^{t(1+\delta)\mu}}$. The minimizer takes place at $t = ln(1+\delta)$. Thus, we have that $P(X \geq (1+\delta)\mu) \leq (\frac{e^\delta}{(1+\delta)^{1+\delta}})^\mu$. It can be shown via doing some work that $(\frac{e^\delta}{(1+\delta)^{1+\delta}}) \leq e^{-\frac{\delta^2}{3}}$ (see p65 of Probability and Computing for a detailed derivation). Hence, we have that $P(X \geq (1+\delta)\mu) \leq e^{-\frac{\mu\delta^2}{3}}$.

Example 2. *When $X_1, \ldots, X_n$ are i.i.d. bernoulli random variables with $p = 0.5$, we can use Chernoff bound to upper bound $P(X \geq \frac{2}{3}n)$. Notice that in this case $\mu = \frac{n}{2}$ and $\delta = \frac{1}{3}$. Consequently, we get that $P(X \geq \frac{2}{3}n) \leq e^{-\frac{n}{54}}$*

Analogous to the bound we obtained for $P(X \geq (1+\delta)\mu)$, we have that $P(X \geq (1-\delta)\mu) \leq (\frac{e^\delta}{(1-\delta)^{1-\delta}}) \leq e^{-\frac{\mu\delta^2}{2}}$. Moreover, we have that $P(|X-\mu| \geq 1+\delta) \leq 2e^{-\frac{\mu\delta^2}{3}}$.

Suppose we have $X_1, \ldots, X_n$ independent random variables where $X_i = 1$ with probability p_i and $X_i = -1$ with probability $1 - p_i$ and we are interested in bounding $P(X \geq (1+\delta)\mu)$. Notice that if we define $Y_i = \frac{X_i+1}{2}$, then X_i becomes a bernoulli random variable with parameter p_i. Thus, we have that $P(X \geq (1+\delta)\mu_X) = P(Y \geq (1+\delta)\mu_Y + \delta n)$ which can be bounded using the expression we derived above.

2.1 Application: Polling Error

Suppose there are two people running for an election and we have a population in which p fraction of the people favors candidate A over candidate B. p is an unknown quantity and we would like to conduct a poll to estimate p. In order to simplify the model, we will assume that the preference of each member of the population can be captured by a bernoulli random variable X_i with parameter p, where $X_i = 1$ if he favors candidate A over candidate B and $X_i = 0$ otherwise.

Suppose we want to poll a sample size of n people and compute the sample mean of their preference, which is denoted by $q = \frac{1}{n}\sum_{i=1}^n X_i$. We would like to know how large of an sample size n we need so that $P(|p-q| \geq \epsilon) \leq 0.01$, meaning with 99% certainty that p and q will be within ϵ distance apart.

From Chernoff bound, we have that $P(|p-q| \geq \epsilon) = P(|\sum_{i=1}^n X_i - \mu| \geq \epsilon n) = P(|\sum_{i=1}^n X_i - \mu| \geq \frac{\epsilon}{p}\mu) \leq 2e^{-\mu\frac{\epsilon^2}{3p^2}} \leq 2e^{-\frac{\epsilon^2 n}{3}}$. To ensure that $P(|p-q| \geq \epsilon) \leq 0.01$, we would like to find n large enough such that $2e^{-\frac{\epsilon^2 n}{3}} \leq 0.01$, which implies that $n \geq \frac{-3ln(0.005)}{\epsilon^2}$

2.2 Application: Hamming Distance

Definition 3. *Given two binary strings A and B of length n. The Hamming distance between A and B, denoted by $dist_H(A,B)$, is $\sum_{i=1}^n |A_i - B_i|$.*

We would like to solve the following problem: given m binary strings $A^1, \ldots, A^m$ of length n each, we would like to find a binary string X of length n such that X minimizes $max_i\ dist_H(X, A^i)$. Similar to the technique we used to solve the set cover problem, we will formulate an IP that solves the Hamming distance problem, solve a LP relaxation of the IP, and then do some randomized rounding using the LP solution. We will use Chernoff bound to obtain some performance measure on our algorithm.

Below is the IP formulation of the problem:

$$
\begin{aligned}
&\min \quad d \\
\text{s.t.} \quad & \sum_{j:A^i_j=1} (1-X_j) + \sum_{j:A^i_j=0} X_j \le d \quad \forall\, i = 1,\dots,m \\
& X \in \{0,1\}^n
\end{aligned}
$$

Again, we relax the constraint that $X \in \{0,1\}^n$ to the constraint that $X \in [0,1]^n$ to obtain the following LP:

$$
\begin{aligned}
&\min \quad d \\
\text{s.t.} \quad & \sum_{j:A^i_j=1} (1-X_j) + \sum_{j:A^i_j=0} X_j \le d \quad \forall\, i = 1,\dots,m \\
& X \in [0,1]^n
\end{aligned}
$$

Then we round the LP solution in the following manner:

Solve the LP relaxation of the Hamming distance IP to obtain an optimal solution X^*.
For each X^*_j, $j = 1,\dots,n$.
Set $y_j = 1$ w.p. x^*_j.
Set $y_j = 0$ otherwise.
Set $d = max_{i=1}^m \sum_{j:A^i_j=1}(1-y_j) + \sum_{j:A^i_j=0} y_j$

In this case, y is clear a feasible solution of the IP. We claim that the following inequality holds with high probability: $max_i\ dist_H(A^i,Y) \le OPT_d + O(\sqrt{nlnn})$, where OPT_d denotes the cost of an optimal IP solution.

Proof. Notice that we can write $dist_H(A^i,Y) = Z^i_1 + \ldots + Z^i_n$, where $Z^i_j = (1-Y_j)$ if $A^i_j = 1$, and $Z^i_j = Y_j$ if $A^i_j = 0$. Notice that $E[\sum_{j=1}^n Z^i_j] = \sum_{j:A^i_j=1}(1-X_j) + \sum_{j:A^i_j=0} X_j \le OPT_d$. Consequently, we have that $P(\sum_{j=1}^n Z^i_j \ge OPT_d + M\sqrt{nlnn}) = P(\sum_{j=1}^n Z^i_j \ge E[\sum_{j=1}^n Z^i_j] + (OPT_d - E[\sum_{j=1}^n Z^i_j]) + M\sqrt{nlnn}) \le P(\sum_{j=1}^n Z^i_j \ge E[\sum_{j=1}^n Z^i_j] + M\sqrt{nlnn})$, where M is some large fixed constant. Let $\mu = E[\sum_{j=1}^n Z^i_j]$, we apply the Chernoff bound with $\delta = \frac{M\sqrt{nlnn}}{\mu}$, we get that $P(\sum_{j=1}^n Z^i_j \ge OPT_d + M\sqrt{nlnn}) \le P(\sum_{j=1}^n Z^i_j \ge E[\sum_{j=1}^n Z^i_j] + M\sqrt{nlnn}) \le e^{-\frac{\mu M^2 nlnn}{3\mu^2}} \le e^{-\frac{M^2 lnn}{3}} = (\frac{1}{n})^{\frac{M^2}{3}}$, where the last inequality follows because $\mu \le n$. Thus, we have that $P(max_i\ dist_H(A^i,Y) \le OPT_d + M\sqrt{nlnn}) = 1 - P(\cup_i\ (dist_H(A^i,Y) \ge OPT_d + M\sqrt{nlnn})) \le 1 - \sum_i P(dist_H(A^i,Y) \ge OPT_d + M\sqrt{nlnn}) \le 1 - (\frac{1}{n})^{\frac{M^2}{3}-1}$, which is very close to 1 when we pick M to be a large constant. □

2.3 Application: $\{0,1\}$ Matrix and Scheduling

Let A be a $n \times n$ $\{0,1\}$ matrix and we would to like to find a $n \times 1$ vector b where $b_i \in \{-1,1\}$ such that $\|c\|_\infty$ is minimized, where c = Ab. This problem has an application in scheduling. Suppose we would like to schedule n jobs given two machines and each machine can run as many jobs as possible at any point in time. $b_i = 1$ if we decide to schedule job i on machine 1 and $b_i = -1$ if we decide to schedule job i on machine 2. Once we scheduled a job on a machine, we must run the job until termination on the machine that it is assigned to. The matrix entry $A_{ij} = 1$ if job j requires processing at time i and 0 if it doesn't. $c_i = \sum_j A_{ij} b_j$ represents the difference between the utility of machine 1 and machine 2 at time i. Thus, finding a vector b that minimizes $\|c\|_\infty$ is equivalent to finding a schedule that makes the maximum absolute difference in utility of the two machines over a time horizon of n periods as small as possible.

Proposition 3. *For $i = 1, \ldots, n$, if we randomly choose $b_i = 1$ with probability 0.5 and $b_i = -1$ with probability 0.5 independently of the other entries of b, then for all matrices $A \in \{0,1\}^{n \times n}$, we have that $\|c\|_\infty \leq O(\sqrt{nlnn})$ with high probability.*

Proof. Notice that $c_i = \sum_{j=1}^n A_{ij} b_j = \sum_{j \in S_i} b_j$, where $S_i = \{j : A_{ij} = 1\}$. Thus, for some large constant M fixed, with the approxiate transformation to bernoulli random variable and apply Chernoff bound, we have that $P(|c_i| \geq M\sqrt{nlnn}) = P(|\sum_{j \in S_i} b_j| \geq M\sqrt{nlnn}) \leq 2e^{-\frac{2M^2 nlnn|S|}{3|S|^2}} \leq 2e^{-\frac{2M^2 lnn}{3}} = 2n^{-\frac{2M^2}{3}}$, where the last inequality follows because $|S| \leq n$. Thus, we may conclude that $P(max_i \ |c_i| \geq M\sqrt{nlnn}) = P(\cup_i \ |c_i| \geq M\sqrt{nlnn}) \leq \sum_i P(|c_i| \geq M\sqrt{nlnn}) \leq 2n^{-\frac{2M^2}{3}+1}$, which is close to 0 when M is a large constant. □