Advanced Algorithms Assignment2

ID: MG1633052 **Name: Keyun Luo** **Email: keyun@smail.nju.edu.cn**

Problem 1

Consider the following optimization problem.

> *Instance*: **n** positive integers $x_1 < x_2 < \cdots < x_n$.
> Find two *disjoint* nonempty subsets $A, B \subset \{1, 2, \dots, n\}$ with $\sum_{i \in A} x_i \ge \sum_{i \in B} x_i$, such that the ratio $\frac{\sum_{i \in A} x_i}{\sum_{i \in B} x_i}$ is minimized.

Give a pseudo-polynomial time algorithm for the problem, and then give an FPTAS for the problem based on the pseudo-polynomial time algorithm.

My Solution

1. **PTAS Algorithm**

   Let $X = \sum_{i=1}^{n} x_i$, and table $T = [1, \dots, X][1, \dots, X]$, each value in $T_i[a][b]$ is a boolean value which indicates if there are disjoint subset **A** and **B** that the sum of element in X_A is **a** and the sum of element in X_B is **b** . For any x_i can be in X_A or X_B or neither of them.
   The table can be computed as follows:

$$T_i[a][b] = \begin{cases} 1 & \text{if } T_{i-1}[a][b] = 1 \\ 1 & \text{if } T_{i-1}[a - x_i][b] = 1 \\ 1 & \text{if } T_{i-1}[a][b - x_i] = 1 \\ 0 & \textit{otherwise} \end{cases}$$

   After filling the table, it takes overall $O(nX^2)$ time, the optimal result can be found by choosing the minimum ratio a/b with the constraint that $T[a][b] = 1 \text{ and } a \ge b$. This can be done in $O(X^2)$ time, after get the optimal value, we can re-construct the set A, B by the following algorithm:

```
Algorithm: Re-Construct-Set
Init: A,B = ∅
Input: a, b, i
Output: set A, B
Steps:
   if i == 1:
     if a == x_1 :
            return ({a}, ∅)
     if b == x_1 :
            return (∅,{b})
   if T_{i-1}[a][b] == 1:
     return Re-Construct-Set(a,b,i-1):
   if T_{i-1}[a - x_i][b] == 1:
     (A,B) ⇐ Re-Construct-Set(a,b,i-1):
     return ( A ⋃ {x_i}, B) :
   if T_{i-1}[a][b - x_i] == 1:
     (A,B) ⇐ Re-Construct-Set(a,b,i-1):
     return (A, B ⋃ {x_i}) :
```

This recursive algorithm takes $O(n)$ time, so all in all, the running time is $O(nX^2)$, it is a pseudo-polynomial time algorithm.

2. **FPTAS Algorithm** (unsolved)
   According to the algorithm above, we know that the input of $x_1, x_2, \ldots, x_n$ is ascending.

Problem 2

In the *maximum directed cut* (MAX-DICUT) problem, we are given as input a directed graph *G(V,E)*. The goal is to partition *V* into disjoint *S* and *T* so that the number of edges in $E(S,T) = \{(u,v) \in E \mid u \in S, v \in T\}$ is maximized. The following is the integer program for MAX-DICUT:

$$
\begin{array}{llr}
\text{maximize} & \sum_{(u,v)\in E} y_{u,v} & (1) \\
\text{subject to} & y_{u,v} \leq x_u, & \forall (u,v)\in E, (2) \\
& y_{u,v} \leq 1 - x_v, & \forall (u,v)\in E, (3) \\
& x_v \in \{0,1\}, & \forall v \in V, (4) \\
& y_{u,v} \in \{0,1\}, & \forall (u,v)\in E. (5)
\end{array}
$$

Let $x^*_v, y^*_{u,v}$ denote the optimal solution to the **LP-relaxation** of the above integer program.

- Apply the randomized rounding such that for every $v \in V, \hat{x}_v = 1$ independently with probability x^*_v. Analyze the approximation ratio (between the expected size of the random cut and OPT).
- Apply another randomized rounding such that for every $v \in V, \hat{x}_v = 1$ independently with probability $1/4 + x^*_v/2$. Analyze the approximation ratio for this algorithm.

My Solution

1. let **OPT** be the maximum weight of *MAX-DICUT*, and it equals the value of given *ILP* algorithm, and the result of the optimal LP-relaxation is OPT_{LP} , the probability of points(u,v) in cut is:

$$
\begin{aligned}
\Pr((u,v)\ in\ cut) &= \Pr(u \in S\ and\ v \in T) \\
&= \Pr(u \in S) \cdot \Pr(v \in T) \\
&= x_u(1 - x_v) \\
&= \frac{1}{2}x_u + x_u(\frac{1}{2} - x_v) \\
&\geq \frac{1}{2}y_{u,v} + x_u(\frac{1}{2} - x_v)
\end{aligned}
$$

since that $\sum_{(u,v\in E)} x_u(\frac{1}{2} - x_v) \geq 0$, so the total number of cuts W is as follows:

$$
E(W) = \sum_{(u,v\in E)} \Pr((u,v)\ in\ cut) \geq \sum_{(u,v\in E)} \frac{y_{u,v}}{2} \geq \frac{OPT_{LP}}{2} \geq \frac{OPT}{2}
$$

The approximation ratio for this algorithm is 0.5.

2. let **OPT** be the maximum weight of *MAX-DICUT*, and it equals the value of given *ILP* algorithm, and the result of the optimal LP-relaxation is OPT_{LP} , the probability of points(u,v) in cut is:

$$
\begin{aligned}
\Pr((u,v)\ in\ cut) &= \Pr(u \in S\ and\ v \in T) \\
&= \Pr(u \in S) \cdot \Pr(v \in T) \\
&= (\frac{1}{4} + \frac{x_u}{2})(1 - (\frac{1}{4} + \frac{x_v}{2})) \\
&= (\frac{1}{4} + \frac{x_u}{2})(\frac{1}{4} + \frac{1 - x_v}{2}) \\
&\geq (\frac{1}{4} + \frac{y_{u,v}}{2})(\frac{1}{4} + \frac{y_{u,v}}{2}) \\
&= (\frac{1}{4} - \frac{y_{u,v}}{2})^2 + \frac{y_{u,v}}{2} \\
&\geq \frac{y_{u,v}}{2}
\end{aligned}
$$

So, the total number of cuts W is as follows:

$$E(W) = \sum_{(u,v \in E)} \Pr((u,v)\ in\ cut) \geq \sum_{(u,v \in E)} \frac{y_{u,v}}{2} \geq \frac{OPT_{LP}}{2} \geq \frac{OPT}{2}$$

The approximation ratio for this algorithm is 0.5.

Problem 3

Recall the MAX-SAT problem and its integer program:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{j=1}^{m} y_j && && (6) \\
\text{subject to} \quad & \sum_{i \in S_j^+} x_i + \sum_{i \in S_j^-} (1 - x_i) \geq y_j, && 1 \leq j \leq m, && (7) \\
& x_i \in \{0,1\}, && 1 \leq i \leq n, && (8) \\
& y_j \in \{0,1\}, && 1 \leq j \leq m. && (9)
\end{aligned}
$$

Recall that $S_j^+, S_j^- \subseteq \{1, 2, \ldots, n\}$ are the respective sets of variables appearing positively and negatively in clause j.

Let x_i^*, y_j^* denote the optimal solution to the **LP-relaxation** of the above integer program. In our class we learnt that if $\hat{x}_i$ is round to 1 independently with probability x_i^*, we have approximation ratio $1 - 1/e$.

We consider a generalized rounding scheme such that every $\hat{x}_i$ is round to 1 independently with probability $f(x_i^*)$ for some function $f : [0,1] \to [0,1]$ to be specified.

- Suppose $f(x)$ is an arbitrary function satisfying that $1 - 4^{-x} \leq f(x) \leq 4^{x-1}$ for any $x \in [0,1]$. Show that with this rounding scheme, the approximation ratio (between the expected number of satisfied clauses and OPT is at least $3/4$.
- Is it possible that for some more clever f we can do better than this? Try to justify your argument.

My Solution

1. let $g(x)=1-4^{-x}, x\in[0,1], g(x)\le f(x)\le 1-g(1-x), g''(x)<0$, then g(x) is monotonically increasing and concavity, $0\le g(x)\le\frac{3}{4}, g(x)\ge\frac{3}{4}x$, let $X=\sum_{i=1}^{k}x_i', x_i'=x_i\ for\ i\in[0,l],\ x_i'=1-x_i\ for\ i\in(l,k]$, then $g(X)\ge\frac{3}{4}X, X\in[0,1], g(X)\ge\frac{3}{4}, X\in[1,\infty]$

$$
\begin{aligned}
\Pr(S_j\ is\ satisfied) &= 1-\prod_{i\in S_j^+}(1-f(x_i))\prod_{i\in S_j^-}f(x_i)\\
&\ge 1-\prod_{i=1}^{l}(1-g(x_i))\prod_{i=l+1}^{k}(1-g(1-x_i))\\
&= 1-\prod_{i=1}^{k}(1-g(x_i'))=1-\prod_{i=1}^{k}(4^{-x_i'})=1-\prod_{i=1}^{k}(4^{-X})\\
&= g(X)\\
&\ge \frac{3}{4}min(1,\sum_{i=1}^{k}x_i')=\frac{3}{4}min(1,\sum_{i=1}^{l}x_i+\sum_{i=l+1}^{k}(1-x_i))\\
&\ge \frac{3}{4}min(1,y_i)\ge\frac{3}{4}y_i
\end{aligned}
$$

So, $OPT\ge OPT_{LP}=\sum_{j=1}^{m}y_j^*$, the approximation ratio = $\frac{3}{4}$

2. It is impossible that for some more clever f we can do better than this.

   Considering the symmetric property of $1-f(x_i)$ and $f(x_i)$, the best result is that they are equal, and the expectation of S_j is not satisfied is $\frac{1}{4^{y_j^*}}$, so $\Pr(S_j\ is\ satisfied)\ge 1-\frac{1}{4^{y_j^*}}\ge\frac{3}{4}y_j^*$

Problem 4

Recall that the instance of **set cover** problem is a collection of m subsets $S_1,S_2,\ldots,S_m\subseteq U$, where U is a universe of size $n=|U|$. The goal is to find the smallest $C\subseteq\{1,2,\ldots,m\}$ such that $U=\bigcup_{i\in C}S_i$. The frequency f is defined to be $\max_{x\in U}|\{i\mid x\in S_i\}|$.

- Give the primal integer program for set cover, its LP-relaxation and the dual LP.
- Describe the complementary slackness conditions for the problem.
- Give a primal-dual algorithm for the problem. Present the algorithm in the language of primal-dual scheme (alternatively raising variables for the LPs). Analyze the approximation ratio in terms of the frequency f.

My Solution

1. (1)**Primal Integer Program**

Minimize: $\sum_{j=1}^{m} c_j x_j$
Subject to:

- $\sum_{j:e_i \in S_j} x_j \geq 1, i = 1, 2, \ldots, n$, *or* $: \sum_{j=1}^{m} a_{i,j} x_j \geq 1, i = 1, 2, \ldots, n$
- $x_j = \{0, 1\}, j = 1, 2, \ldots, m$

(2)**LP-relaxation**
Minimize: $\sum_{j=1}^{m} c_j x_j$
Subject to:

- $\sum_{j:e_i \in S_j} x_j \geq 1, i = 1, 2, \ldots, n$, *or* $: \sum_{j=1}^{m} a_{i,j} x_j \geq 1, i = 1, 2, \ldots, n$
- $x_j \geq 0, j = 1, 2, \ldots, m$

(3) **Dual LP**
Maximize: $\sum_{i=1}^{n} y_i$
Subject to:

- $\sum_{j:e_i \in S_j} y_j \leq c_j, j = 1, 2, \ldots, m$, *or* $: \sum_{i=1}^{n} a_{i,j} y_j \leq c_j, j = 1, 2, \ldots, m$
- $y_i \geq 0, i = 1, 2, \ldots, n$

2. Complementary Slackness Conditions

> ∀ feasible primal solution x and dual solution y, x and y are both optimal if:
>
> - For each $1 \leq i \leq n$, *either* $\sum_{j=1}^{m} a_{i,j} x_j = 1$ *or* $y_i = 0$
> - For each $1 \leq j \leq m$, *either* $\sum_{i=1}^{n} a_{i,j} y_j = c_j$ *or* $x_j = 0$
>
> ∀ feasible primal solution x and dual solution y, for $\alpha, \beta \geq 1$:
>
> - For each $1 \leq i \leq n$, *either* $1 \leq \sum_{j=1}^{m} a_{i,j} x_j \leq \beta$ *or* $y_i = 0$
> - For each $1 \leq j \leq m$, *either* $c_j \geq \sum_{i=1}^{n} a_{i,j} y_j \geq \frac{c_j}{\alpha}$ *or* $x_j = 0$

3.

```
1. Initially x = 0, y = 0;
2. While E ≠ ∅ :
3.    Pick an  e uncovered and raise y_e until some set goes tight
4.    Pick all tight sets in the cover and update  x
5.    Delete these sets from  E
6. Output the set cover x
```

here, $cx \leq \alpha\beta y \leq \alpha\beta OPT$, in primal conditions, the variables will be incremented integrally, $x_j \neq 0 \Rightarrow \sum_{j:e_i \in S_j} y_j = c_j, \alpha = 1$, in the dual conditions, each element having a nonzero dual value can be covered at most f times, $y_i \neq 0 \Rightarrow \sum_{j:e_i \in S_j} x_j \leq f, \beta = f$.
So, the approximation ratio is f.
