SCP（集合覆盖问题）介绍

SCP简介、近似算法、问题难度

梁宇方 171860695

计算机科学与技术系

2019年5月27日

摘要

- 本次关于SCP的Open Topic的思路如下：

  - 简要介绍SCP问题。SCP是什么？SCP有什么用？
  - 介绍SCP的贪心近似算法。算法的时间复杂度、近似度是什么？
  - 证明SCP属于NPO(IV)，说明贪心近似算法的有效性。

01 SCP简介

概述、应用

1.1 概述

- 全称：set cover problem（集合覆盖问题）

- 描述：

Input:

Ground elements, or Universe $U = \{u_1, u_2, ..., u_n\}$

Subsets $S_1, S_2, ..., S_k \subseteq U$

Costs $c_1, c_2, ..., c_k$

Goal:

Find a set $I \subseteq \{1,2,..., k\}$ that minimizes $\sum_{i \in I} c_i$, such that $\bigcup_{i \in I} S_i = U$.

(note: in the un-weighted Set Cover Problem, $c_j = 1$ for all j)

1.2 应用

- 应用：查找计算机病毒

Interesting example: IBM finds computer viruses (wikipedia)

elements- 5000 known viruses

sets- 9000 substrings of 20 or more consecutive bytes from viruses, not found in 'good' code

A set cover of 180 was found. It suffices to search for these 180 substrings to verify the existence of known computer viruses.

- 翻译：
  - 元素：5000种已知的电脑病毒
  - 子集：9000个特征子字符串
  - SCP：180个特征子字符串，所有的5000种已知的病毒都含有其中的某个特征子字符串

02

贪心近似算法

概述、举例、近似度证明

2.1 概述

- 算法：
  - 见右上图
- 时间复杂度：
  - $O(n^{3/2})$
  - 见右下图
- 近似比：
  - $H_n = 1 + \frac{1}{2} + \ldots + \frac{1}{|U|} \approx \log|U|$

Let C represent the set of elements covered so far

Let cost effectiveness, or α, be the average cost per newly covered node

Algorithm

1. C ← 0
2. While C ≠ U do

   Find the set whose cost effectiveness is smallest, say S

   Let $\alpha = \frac{c(S)}{|S - C|}$

   For each e ∈ S-C, set price(e) = α

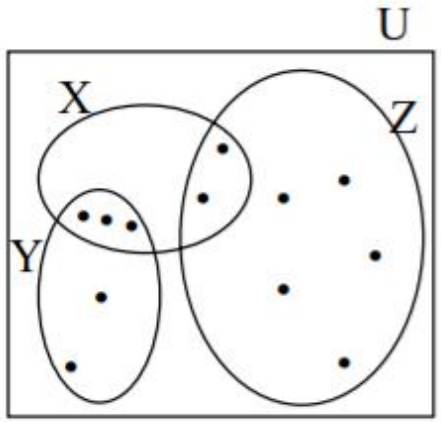
   C ← C ∪ S
3. Output picked sets

Proof. Let us encode the inputs in such a wav that one can consider |U|, |S| to be the input size of an input instance (U, S). One run of Step 2 costs $O($ |U|*|S| $)$ time. The number of runs of Step 2 is bounded by min{ |U|, |S| } ≤ (|U|*|S|)$^{1/2}$. Thus, the time complexity of Algorithm 4.3.2.11 is in $O(n^{3/2})$. □

2.2 举例

- 第0步　空集
  - $a_X = 6/5 = 1.2$
  - $a_Y = 15/5 = 3$
  - $a_Z = 7/7 = 1$
- 第1步　选Z
  - $a_X = 6/3 = 2$
  - $a_Y = 15/5 = 3$
- 第2步　选X
  - $a_Y = 15/2 = 7.5$
- 第3步　选Y
- 输出：X,Y,Z

Example

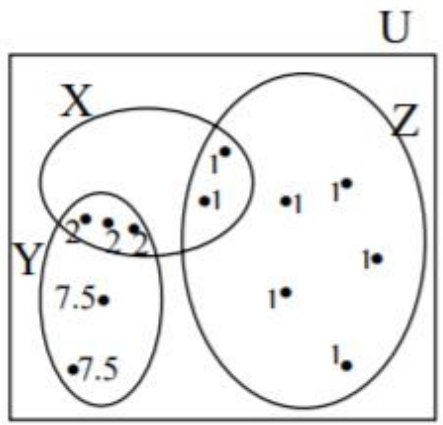


c(X) = 6
c(Y) = 15
c(Z) = 7

Choose Z: $\alpha_Z = \frac{c(Z)}{|S-C|} = \frac{7}{7} = 1$

Choose X: $\alpha_X = \frac{c(X)}{|S-C|} = \frac{6}{3} = 2$

Choose Y: $\alpha_Y = \frac{c(Y)}{|S-C|} = \frac{15}{2} = 7.5$



Total cost = 6 + 15 + 7 = 28

2.3 近似度证明（1）

- 证明目标：

$$H_n = 1 + \frac{1}{2} + \ldots + \frac{1}{|U|} \approx \log|U|$$

- 证明思路：
  - 放缩
  - 级数

Proof:

(i) We know $\sum_{e \in U} price(e) =$ cost of the greedy algorithm $= c(S_1) + c(S_2) + \ldots + c(S_m)$

because of the nature in which we distribute costs of elements.

(ii) We will show $price(e_k) \le \frac{OPT}{n-k+1}$, where e_k is the kth element covered.

Say the optimal sets are $O_1, O_2, \ldots, O_p$.

So, OPT $\overset{a}{=} c(O_1) + c(O_2) + \ldots + c(O_p)$.

Now, assume the greedy algorithm has covered the elements in C so far. Then we know the uncovered elements, or $|U - C|$, are at most the intersection of all of the optimal sets intersected with the uncovered elements:

$$|U - C| \overset{b}{\le} |O_1 \cap (U - C)| + |O_2 \cap (U - C)| + \ldots + |O_p \cap (U - C)|$$

In the greedy algorithm, we select a set with cost effectiveness α, where

$\alpha \overset{c}{\le} \frac{c(O_i)}{|O_i \cap (U - C)|}$, $i = 1 \ldots p$. We know this because the greedy algorithm will always choose the set with the smallest cost effectiveness, which will either be smaller than or equal to a set that the optimal algorithm chooses.

2.3 近似度证明（2）

Algebra: $c(O_i) \overset{c}{\geq} \alpha \cdot |O_i \cap (U - C)|$

$$\text{OPT} \overset{a}{=} \sum_i c(O_i) \overset{c}{\geq} \alpha \cdot \sum_i |O_i \cap (U - C)| \overset{b}{\geq} \alpha \cdot |U - C|$$

$$\alpha \leq \frac{OPT}{|U - C|},$$

Therefore, the price of the k-th element is:

$$\alpha \leq \frac{OPT}{n - (k-1)} = \frac{OPT}{n - k + 1}$$

Putting (i) and (ii) together, we get the total cost of the set cover:

$$\sum_{k=1}^{n} price(e_k) \leq \sum_{k=1}^{n} \frac{OPT}{n - k + 1} = OPT \cdot \left(1 + \frac{1}{2} + \ldots + \frac{1}{n}\right) = OPT \cdot H_n$$

03

SCP属于NPO(IV)

引言、证明、拓展

3.1 引言

NPO(IV): Contains every $U \in NPO$ such that
(i) there is a polynomial-time $f(n)$-approximation algorithm for U for some $f : \mathbb{N} \to \mathbb{R}^+$, where f is bounded by a polylogarithmic function, and
(ii) under some reasonable assumption like P $\neq$ NP, there does not exist any polynomial-time δ-approximation algorithm for U for any $\delta \in \mathbb{R}^+$.
{The set cover problem belongs to this class.}

- NPO(IV):
  - 存在多项式时间、多重对数近似度的近似算法
  - 不存在多项式时间、常数近似度的近似算法，除非P=NP

3.2 证明

- （i）存在多项式时间、多项式近似度的近似算法
  - 前文提及，贪心近似算法，其时间复杂度为O($n^{3/2}$)，近似度为O(lgn)
- （ii）不存在多项式时间、常数近似度的近似算法，除非P=NP
  - 已知MIN-VCP是NPO(III)，满足(ii)，而 $MIN-VCP \leq_P SCP$ ，所以SCP也满足(ii)

> One can easily observe that MIN-VCP is a special case of SCP. For a graph $G = (V, E)$, $V = \{v_1, \ldots, v_n\}$, the corresponding input instance of SCP is $(E, \{E_1, \ldots, E_n\})$, where $E_i \subseteq E$ contains all edges adjacent to v_i for $i = 1, \ldots, n$. Algorithm 4.3.2.11, then, is a $(\ln n)$-approximation algorithm for MIN-VCP, too.

3.3 拓展（1）

对于$1 \le i \le k$，定义变量x_i如下：

$$x_i = \begin{cases} 1, & i \in I \\ 0, & i \notin I \end{cases}$$

对于$1 \le i \le k$和$1 \le j \le n$，定义常量a_{ij}如下：

$$a_{ij} = \begin{cases} 1, & u_j \in S_i \\ 0, & u_j \notin S_i \end{cases}$$

IP的目标最小化函数：

$$f = \sum_{i=1}^{k} c_i \cdot x_i$$

限制条件：对于$1 \le j \le n$，

$$\sum_{i=1}^{k} a_{ij} \cdot x_i \ge 1$$

Input:

Ground elements, or Universe $U = \{u_1, u_2, ..., u_n\}$

Subsets $S_1, S_2, ..., S_k \subseteq U$

Costs $c_1, c_2, ..., c_k$

Goal:

Find a set $I \subseteq \{1,2,..., k\}$ that minimizes $\sum_{i \in I} c_i$, such that $\bigcup_{i \in I} S_i = U$.

(note: in the un-weighted Set Cover Problem, $c_j = 1$ for all j)

- SCP的形式化描述：
  - 见右上角的图片
- SCP归约为ILP：
  - 见左下角的图片
- 对ILP的限制条件放松使之变为LP
  - 可以用线性规划算法来求SCP的近似解

3.3 拓展（2）

- 近似度下界：

> In what follows we consider the set cover problem (SCP). This problem is in the class NPO(IV). Surprisingly, the naive greedy approach provides the best possible approximation ratio.

- 文献：

[11] Carsten Lund and Mihalis Yannakakis. On the hardness of approximating minimization problems. J. ACM, 41(5):960–981, 1994.

04 结语

参考、提问

4.1 参考

- MIT　不详　课程讲义

  https://math.mit.edu/~goemans/18434S06/setcover-tamara.pdf

- 山东大学　冯富宝　硕士学位论文

  http://xuewen.cnki.net/readarticleview.aspx?filename=2006164591.nh&dbtype=cmfd

- Algorithmics for Hard Problems [JH] 教科书

4.2 提问

- SCP的贪心算法的近似度为?

- 前文通过将什么问题规约到SCP来证明SCP属于NPO(IV)?

THANKS