Set Covering Problems

Formulations

SET COVERING PROBLEM

Given a finite base set U and a family of its subsets $\mathcal{S} \subseteq 2^U$ together with nonnegative real weights $c : \mathcal{S} \longrightarrow \mathbb{R}_+$ associated to each subset, **find** a subfamily $\mathcal{C} \subseteq \mathcal{S}$ such that

$$\bigcup_{S \in \mathcal{C}} S = U, \tag{1}$$

and $c(\mathcal{C}) = \sum_{S \in \mathcal{C}} c(S)$ is as small as possible.

A family $\mathcal{C} \subseteq \mathcal{S}$ for which (1) holds is called a *covering subfamily* of $\mathcal{S}$.

HITTING SET PROBLEM

Given a finite base set V, nonnegative real weights $c_v \in \mathbb{R}_+$ associated to the vertices $v \in V$, and a family $\mathcal{H} \subseteq 2^V$ of subsets of V, **find** a subset $C \subseteq V$ such that

$$C \cap H \neq \emptyset \quad \text{for all} \quad H \in \mathcal{H}, \tag{2}$$

and $c(C) = \sum_{v \in C} c_v$ is as small as possible.

A subset $C \subseteq V$ for which (2) holds is called a *vertex cover* of $\mathcal{H}$.

The complement of a vertex cover is called an *independent* set, i.e. a set of vertices which does not contain any of the hyperedges of $\mathcal{H}$. Clearly, independent sets of $\mathcal{H}$ form an independence system (WHY??). Thus, finding a minimum weight vertex cover is equivalent with finding a maximum weight independent set.

Remark: The set covering problem and the hitting set problem are equivalent, and in fact one is the *transpose* of the other as it can be seen from the identifications

$$U \equiv \mathcal{H} \quad \text{and} \quad \mathcal{S} \equiv V.$$

Let us associate to a hitting set problem a binary matrix $A_{\mathcal{H}} \in \{0,1\}^{\mathcal{H}\times V}$ defined by

$$a_{H,v} = \begin{cases} 1 & \text{if} \quad v \in H, \\ 0 & \text{otherwise.} \end{cases}$$

In other words, the rows of $A_{\mathcal{H}}$ are the characteristic vectors of the sets $H \in \mathcal{H}$.

Introducing binary decision variables $x = (x_v \mid v \in V) \in \{0,1\}^V$ defined as

$$x_v = \begin{cases} 1 & \text{if } v \in C, \\ 0 & \text{otherwise,} \end{cases}$$

$c = (c_v \mid v \in V) \in \mathbb{R}^V_+$, and $b = \mathbf{1} = (1,1,...,1) \in \mathbb{R}^{\mathcal{H}}_+$, the hitting set problem can equivalently be formulated as the following binary integer programming problem:

$$\begin{array}{rcl} c^T x & \rightarrow & \min \\ s.t. \quad A_{\mathcal{H}} x & \geq & b \\ x & \in & \{0,1\}^V. \end{array} \tag{3}$$

The optimum value of this problem is denoted by $\tau_c(\mathcal{H})$ (or simply by $\tau(\mathcal{H})$ when $c = (1,1,...,1)$) and is called the (weighted) *covering number* of the family $\mathcal{H}$.

The set covering and hitting set problems are NP-hard optimization problems, since they include, as special case, the vertex cover problem for graphs, which on its turn is equivalent with the maximum stable set problem, which is known to be NP-hard.

A strongly related problem is the hypergraph analogue of matching problems in graphs:

SET PACKING PROBLEM

Given a finite base set U and a family of its subsets $\mathcal{S} \subseteq 2^U$ together with nonnegative real weights $b : \mathcal{S} \longrightarrow \mathbb{R}_+$ associated to each subset, **find** a subfamily $\mathcal{M} \subseteq \mathcal{S}$ such that

$$H \cap H' = \emptyset \quad \text{for all} \quad H, H \in \mathcal{M},\ H \neq H' \tag{4}$$

for which $b(\mathcal{M}) = \sum_{S\in\mathcal{M}} b(S)$ is as large as possible.

A subfamily $\mathcal{M} \subseteq \mathcal{S}$ for which (4) holds is called a *matching* of $\mathcal{S}$.

Let us assume for a little while that all weights are 1, i.e. that $c = (1, ..., 1)$ and $b = (1, ..., 1)$ (of appropriate dimensions). Then we can write the following inequalities

$$\max_{\substack{y^T A_{\mathcal{H}} \le c^T \\ y \in \{0,1\}^{\mathcal{H}}}} y^T b \quad \le \quad \max_{\substack{y^T A_{\mathcal{H}} \le c^T \\ y \in [0,1]^{\mathcal{H}}}} y^T b \quad = \quad \min_{\substack{A_{\mathcal{H}} x \ge b \\ x \in [0,1]^V}} c^T x \quad \le \quad \min_{\substack{A_{\mathcal{H}} x \ge b \\ x \in \{0,1\}^V}} c^T x \qquad (5)$$

The equality in the middle follows by linear programming duality (assuming that the sets in $\mathcal{H}$ cover all points of the base set), while the inequalities follow from the facts that the linear programming problems in the middle are respectively the relaxations of the integer programming problems at the two ends [WHY? HOW?].

Thus, introducing the standard notation

$$\nu(\mathcal{H}) = \max\{\, y^T b \mid y^T A_{\mathcal{H}} \le c^T \ \text{ and } \ y \in \{0,1\}^{\mathcal{H}}\},$$

$$\nu^*(\mathcal{H}) = \max\{\, y^T b \mid y^T A_{\mathcal{H}} \le c^T \ \text{ and } \ y \in [0,1]^{\mathcal{H}}\},$$

$$\tau^*(\mathcal{H}) = \min\{\, c^T x \mid A_{\mathcal{H}} x \ge b \ \text{ and } \ x \in [0,1]^V\},$$

we have

$$\nu(\mathcal{H}) \quad \le \quad \nu^*(\mathcal{H}) \quad = \quad \tau^*(\mathcal{H}) \quad \le \quad \tau(\mathcal{H}).$$

The quantity $\nu(\mathcal{H})$ is called the *matching number* of the hypergraph $\mathcal{H}$, while $\nu^*(\mathcal{H}) = \tau^*(\mathcal{H})$ are called respectively the *fractional matching* and *fractional covering* numbers of $\mathcal{H}$.

The greedy algorithm

Let us denote by $d_{\mathcal{H}}(v)$ the degree of vertex $v \in V$ in $\mathcal{H}$, i.e.

$$d_{\mathcal{H}}(v) = |\{\, H \in \mathcal{H} \mid H \ni v \,\}|,$$

and let $D_{\mathcal{H}} = \max_{v \in V} d_{\mathcal{H}}(v)$.

GREEDY ALGORITHM (FOR THE HITTING SET PROBLEM)

Input: A hypergraph $\mathcal{H} \subseteq 2^V$.

Initialize: Set $k = 0$, $C = \emptyset$, and $\mathcal{H}_k = \mathcal{H}$.

Main Loop: While $\mathcal{H}_k \neq \emptyset$ **do**:

(a) Choose $v_k \in V \setminus C$ for which $d_{\mathcal{H}_k}(v_k)$ is the largest.

(b) Set $\mathcal{H}_{k+1} = \{H \in \mathcal{H}_k \mid v_k \notin H\}$.

(c) Set $C = C \cup \{v_k\}$ and $k = k + 1$.

Output: Output the set $C^G = C$.

Lemma 1 *For every hypergraph $\mathcal{H} \subseteq 2^V$ we have*

$$\max_{\mathcal{H}' \subseteq \mathcal{H}} \frac{|\mathcal{H}'|}{D_{\mathcal{H}'}} \leq \tau^*(\mathcal{H}).$$

Proof. Let us observe that for every subfamily $\mathcal{H}' \subseteq \mathcal{H}$ the vector

$$y^*_H = \begin{cases} \frac{1}{D_{\mathcal{H}'}} & \text{if } H \in \mathcal{H}', \\ 0 & \text{otherwise} \end{cases}$$

is a feasible solution of the dual of the linear programming relaxation of the hitting set problem. This is because for every vertex $v \in V$ we have

$$\sum_{\substack{H \ni v \\ H \in \mathcal{H}}} y^*_H \;=\; \sum_{\substack{H \ni v \\ H \in \mathcal{H}'}} \frac{1}{D_{\mathcal{H}'}} \;=\; \frac{d_{\mathcal{H}'}(v)}{D_{\mathcal{H}'}} \;\leq\; 1.$$

Thus the corresponding objective function value is a lower bound on $\tau^*(\mathcal{H})$, yielding the inequality

$$\sum_{H \in \mathcal{H}} y^*_H \;=\; \sum_{H \in \mathcal{H}'} \frac{1}{D_{\mathcal{H}'}} \;=\; \frac{|\mathcal{H}'|}{D_{\mathcal{H}'}} \;\leq\; \tau^*(\mathcal{H}).$$

□

Theorem 1 (Lovász (1975), Stein (1974)) *For every hypergraph $\mathcal{H} \subseteq 2^V$ we have*

$$\tau(\mathcal{H}) \leq |C^G| \leq \tau^*(\mathcal{H})(1 + \ln D_{\mathcal{H}}).$$

Proof. Let us observe that

$$|\mathcal{H}_{k+1}| = |\mathcal{H}_k| - D_{\mathcal{H}_k} = |\mathcal{H}_k| \left(1 - \frac{D_{\mathcal{H}_k}}{|\mathcal{H}_k|}\right) \leq |\mathcal{H}_k| \left(1 - \frac{1}{\tau^*(\mathcal{H})}\right),$$

where the last inequality is implied by Lemma 1. Thus

$$|\mathcal{H}_k| \leq |\mathcal{H}| \left(1 - \frac{1}{\tau^*(\mathcal{H})}\right)^k,$$

since $\mathcal{H} = \mathcal{H}_0$. For $s = \lceil \tau^*(\mathcal{H}) \ln D_{\mathcal{H}} \rceil$ the above inequality implies

$$\begin{aligned} |\mathcal{H}_s| &\leq |\mathcal{H}| \left(1 - \tfrac{1}{\tau^*(\mathcal{H})}\right)^s \\ &\leq |\mathcal{H}| \left(1 - \tfrac{1}{\tau^*(\mathcal{H})}\right)^{\tau^*(\mathcal{H}) \ln D_{\mathcal{H}}} \\ &\leq |\mathcal{H}| e^{-\ln(D_{\mathcal{H}})} \\ &= \tfrac{|\mathcal{H}|}{D_{\mathcal{H}}} \\ &\leq \tau^*(\mathcal{H}), \end{aligned}$$

where the last inequality follows again by Lemma 1. Thus, the greedy algorithm will certainly stop after at most $s + \lfloor \tau^*(\mathcal{H}) \rfloor \approx \tau^*(\mathcal{H})(1 + \ln D_{\mathcal{H}})$ steps, completing the proof. □

Corollary 1 *For every hypergraph $\mathcal{H}$ the greedy algorithm runs in $O(|\mathcal{H}||V|)$ time, and produces a solution C^G for which we have*

$$\tau(\mathcal{H}) \leq |C^G| \leq \tau^*(\mathcal{H})(1 + \ln D_{\mathcal{H}}) \leq \tau(\mathcal{H})(1 + \ln D_{\mathcal{H}}).$$

In other words, the greedy algorithm is a polynomial time $(1 + \log D_{\mathcal{H}})$-factor approximation algorithm for the hitting set (and thus for the set covering) problem. □

When $\mathcal{H} = E(G)$ for a graph $G = (V, E)$ the hitting set problem is known as the *vertex cover* problem for graphs. It is easy to see that the above greedy procedure does not have better guarantees, even in this special case.

Theorem 2 (Johnson (1974), Papadimitriou and Steigliz (1982)) *For all $k \geq 3$ there exists a graph $G = (V, E)$ for which*

$$|V| \;=\; k + \sum_{j=2}^{k} \left\lfloor \frac{k}{j} \right\rfloor,$$

the maximum degree in G is k, $\tau(G) = k$, and for which the greedy algorithm finds only a vertex cover $C^G \subseteq V$ of size

$$|C^G| = |V| - k.$$

Proof. Let A_j be a set of size $\lfloor \frac{k}{j} \rfloor$, such that these sets are pairwise disjoint for $j = 2, ..., k$. Let further S be a set of size k, disjoint from all sets A_j, $j = 2, ..., k$, and set

$$V = S \cup \bigcup_{j=2}^{k} A_j.$$

Let us then add edges between S and $V \setminus S$ such that each vertex in S has at most one neighbor in each of the groups A_j, $j = 2, ..., k$, and that each vertex of A_j has exactly j neighbors in S, for all $j = 2, ..., k$. Since $j|A_j| \leq |S|$, this is always possible.

Then S is an optimal vertex cover of size k, while the greedy procedure will choose all the vertices in $V \setminus S$, first those in A_k, then those in A_{k-1}, etc. □

It can be shown however that for the vertex cover problem in graphs there exists a polynomial time 2-factor approximation (HOW?? THINK OF MAXIMUM MATCHINGS ...), and that this performance guarantee is just about tight. We do not know any better, and it was shown by Hastad (1997) that there cannot be a polynomial time approximation algorithm for the vertex cover problem with a performance guarantee better than 1.166 unless $P = NP$.

Another special case of the set covering problem, is when $\mathcal{S} = E(G)$ for a graph G. This problem is known as the (weighted) *edge covering* problem in graphs. For this problem the greedy algorithm is a $\frac{3}{2}$-approximation [WHY?]. Moreover, unlike the vertex covering, this problem can be solved in polynomial time! [HOW?? THINK OF EXTENDING A MATCHING ...]

Weighted set covering

For an integer $r \in \mathbb{Z}_+$ let $H(r) = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{r}$. Note that

$$\ln(r+1) \leq H(r) \leq 1 + \ln r.$$

Let us now return to the weighted case. Given a hypergraph $\mathcal{H} \subseteq 2^V$, and a weight function $c \in \mathbb{R}_+^V$, let us denote by $\tau_c(\mathcal{H})$ the optimum of the weighted hitting set problem, i.e.,

$$\tau_c(\mathcal{H}) = \min\{c(S) \mid S \text{ is a hitting set for } \mathcal{H}\}$$

where $c(S) = \sum_{v \in S} c_v$.

Chvátal (1979) generalized the above algorithm and proposed the following variant:

GREEDY ALGORITHM (FOR WEIGHTED HITTING SET)

Input: A hypergraph $\mathcal{H} \subseteq 2^V$ and a weight function $c \in \mathbb{R}_+^V$.

Initialize: Set $k = 0$, $C = \emptyset$, and $\mathcal{H}_k = \mathcal{H}$.

Main Loop: While $\mathcal{H}_k \neq \emptyset$ **do**:

(a) Choose $v_k \in V \setminus C$ for which $\frac{c(v_k)}{d_{\mathcal{H}_k}(v_k)}$ is the smallest.

(b) Set $\mathcal{H}_{k+1} = \{H \in \mathcal{H}_k \mid v_k \notin H\}$.

(c) Set $C = C \cup \{v_k\}$ and $k = k + 1$.

Output: Output the set $C^G = C$.

Theorem 3 (Chvátal (1979)) *For any hypergraph $\mathcal{H} \subseteq 2^V$ and weight function $c : V \to \mathbb{R}_+$ the above greedy algorithm outputs in $O(|\mathcal{H}||V|)$ time a vertex cover $C^G \subseteq V$ of $\mathcal{H}$ such that*

$$|C^G| \leq H(D_{\mathcal{H}})\, \tau_c(\mathcal{H}).$$

Proof. Let $j(H) = \min\{k \mid v_k \in H\}$ for $H \in \mathcal{H}$, and set

$$y(H) = \frac{c(v_{j(H)})}{d_{\mathcal{H}_{j(H)}}(v_{j(H)})}.$$

Let us also note that

$$|\mathcal{H}_k \setminus \mathcal{H}_{k+1}| = d_{\mathcal{H}_k}(v_k)$$

for all $k = 0, ..., |C^G| - 1$.

For an arbitrary element $v \in V$ let $n_v = \max\{j(H) \mid H \ni v\}$. Then, we can write

$$\begin{aligned}\sum_{H \ni v} y(H) &= \sum_{k=0}^{n_v} \sum_{\substack{H \ni v \\ j(H)=k}} y(H) \\ &= \sum_{k=0}^{n_v} \frac{c(v_k)}{d_{\mathcal{H}_k}(v_k)}\; d_{\mathcal{H}_k \setminus \mathcal{H}_{k+1}}(v)\end{aligned}$$

Let us denote by $s_k = d_{\mathcal{H}_k}(v)$, and observe that $s_k > 0$ for all $k = 0, ..., n_v$.

$$= \sum_{k=1}^{n_v} \frac{c(v_k)}{d_{\mathcal{H}_k}(v_k)}(s_k - s_{k+1})$$

By the choice of v_k we have $c(v_k)/d_{\mathcal{H}_k}(v_k) \leq c(v)/d_{\mathcal{H}_k}(v)$, implying

$$\begin{aligned}
&\leq \sum_{k=0}^{n_v} \frac{c(v)}{s_k}(s_k - s_{k+1}) \\
&\leq c(v) \sum_{k=0}^{n_v} \frac{s_k - s_{k+1}}{s_k} \\
&\leq c(v) \sum_{k=0}^{n_v} \left(\frac{1}{s_k} + \frac{1}{s_k - 1} + \cdots + \frac{1}{s_{k+1} + 1} \right) \\
&= c(v) \sum_{k=0}^{n_v} (H(s_k) - H(s_{k+1})) \\
&= c(v) \left(H(s_0) - H(s_{n_v}) \right) \\
&\leq c(v) H(s_0).
\end{aligned}$$

Since $s_0 = d_{\mathcal{H}}(v) \leq D_{\mathcal{H}}$ we can conclude that

$$\sum_{H \ni v} y(H) \leq c(v) H(D_{\mathcal{H}}).$$

Summing up these for all points of an optimal hitting set C^{OPT} we get

$$\begin{aligned}
c(C^{OPT}) H(D_{\mathcal{H}}) &\geq \sum_{v \in C^{OPT}} \sum_{H \ni v} y(H) \geq \sum_{H \in \mathcal{H}} y(H) \\
&= \sum_{k=0}^{|C^G|-1} \sum_{H : j(H) = k} y(H) \\
&= \sum_{k=0}^{|C^G|-1} c(v_k) = c(C^G).
\end{aligned}$$

□

Let us remark that an approximation ratio of $\gamma(\ln |\mathcal{H}|)$ cannot be achieved for any $0 < \gamma < 1$ unless all NP-hard problems can be solved in quasi-polynomial time, as was shown by Feige (1998) (for a similar statement, see also Safra and Raz (1997)).