4 May 2018. Approximation Algorithms for Vertex Cover

Vertex Cover: Integer Programming formulation

$$\min \sum_{v \in V} x_v$$

s.t. $x_u + x_v \geq 1 \quad \forall$ edges (u,v)

$x_v \in \{0,1\} \quad \forall$ vertex v

Models vertex cover exactly.
Solution vectors $\vec{x}$ are in 1:1 correspondence with vertex covers of G.
Objective function $\sum x_v$ maps to the vertex cover objective function under this correspondence.

Linear Programming relaxation

$$\min \sum x_v$$

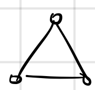
s.t. $x_u + x_v \geq 1 \quad \forall$ edges (u,v)

$0 \leq x_v \leq 1 \quad \forall$ vertex v

Solvable in poly time.
Doesn't model vertex cover exactly.
(Has fractional solutions that don't correspond to vertex covers.)
But its integer solutions correspond to vertex covers.

A 2-approximation algorithm for vertex cover:

1. Solve LP relaxation to obtain vector $\vec{x}$.
2. Output $\{v \mid x_v \geq \frac{1}{2}\}$.

Example. $G =$  $\quad x_u = x_v = x_w = \frac{1}{2} \implies$ alg outputs $\{u,v,w\}$.

Min vertex cover is any two of these

Analysis of the algorithm.

1. The output is always a vertex cover.
   For every edge (u,v)
   $$x_u + x_v \geq 1$$
   $$\max\{x_u, x_v\} \geq \frac{1}{2} \implies$$ Either u or v or both are chosen to be in the output set.

2. The output is at most twice as big as the min vertex cover.
   Prove this using two inequalities

$$(\text{size of output set}) \leq 2 \sum_{v \in V} x_v \leq 2 \cdot (\text{size of minimum vertex cover})$$

$$\text{(size of output set)} \overset{(*)}{\leq} 2 \sum_{v \in V} x_v \overset{(**)}{\leq} 2 \cdot \text{(size of minimum vertex cover)}$$

Let S = output set, S^* = minimum vertex cover.

$$\text{size of output set} = \sum_{v \in S} 1 \leq \sum_{v \in S} (2 \cdot x_v) \leq 2 \sum_{v \in V} x_v. \qquad (*)$$

Let $\hat{x}_v = \begin{cases} 1 & \text{if } v \in S^* \\ 0 & \text{if } v \notin S^* \end{cases}$. Then $\hat{x}$ satisfies the constraint set $\left\{ \begin{array}{ll} \hat{x}_u + \hat{x}_v \geq 1 & \forall (u,v) \in E \\ 0 \leq \hat{x}_v \leq 1 & \forall v \in V \end{array} \right\}$

So $\hat{x}$ belongs to the feasible set for the LP.
Since x is an opt LP solution,

$$\sum_{v \in V} x_v \leq \sum_{v \in V} \hat{x}_v = \text{size of minimum vertex cover} \qquad (**)$$

Remark. The reasoning in step (**) is stereotypical for reasoning about approx algs based on linear programs. The value of the LP solution is always a lower bound (for minimization problems) or an upper bound (for maximization) on the true optimal solution.

A really simple greedy algorithm for vertex cover

Initialize $S = \emptyset$
while $\exists$ an edge (u,v) with no endpoint in S
  add u and v to S
endwhile
output S.

1. This outputs a vertex cover.
   The termination condition for the while-loop is that S is a vertex cover. The algorithm terminates because # uncovered edges strictly decreases in each iteration.

2. The output set is at most twice the size of the min vtx cover.
   In every loop iteration we insert 2 vertices into S and (at least) one of them also belongs to the minimum vertex cover, S^*. This defines a 2-to-1 mapping from S to subset of S^*.

Weighted vertex cover

Input: graph G, positive weights $w(v)$, $\forall v \in V$.
Goal: find a vertex cover with minimum combined weight.

LP relaxation: $\min \sum_{v \in V} w_v x_v$

$$\text{s.t.} \quad x_u + x_v \geq 1 \quad \forall (u,v) \in E$$
$$0 \leq x_v \leq 1 \quad \forall v \in V$$

Algorithm: Solve LP, output $S = \{v \mid x_v \geq \frac{1}{2}\}$,
Analysis: If S^* is the vtx cover with minimum total weight...

$$\sum_{v \in S} w_v \overset{(*)}{\leq} 2 \sum_{v \in V} w_v x_v \overset{(**)}{\leq} 2 \sum_{v \in S^*} w_v$$

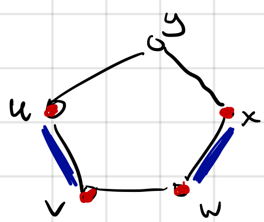
Justifications for the two inequalities are same as before.

Greedy algorithm can fail badly on weighted instances, e.g.

1 o————o 1000 Take both endpoints $\Rightarrow$ combined weight 1001
Just left endpoint: weight 1.

Primal-Dual Algorithm ("The Pricing Method", §11.4)

One way of presenting the analysis of the greedy algorithm....



One possible execution of GREEDY picks edge (u,v), then (w,x).

One way to see this is a 2-approx. to opt.

$$\min \quad x_u + x_v + x_w + x_x + x_y$$
$$\text{s.t.} \quad \boxed{x_u + x_v \geq 1}$$
$$x_v + x_w \geq 1$$
$$\boxed{x_w + x_x \geq 1}$$
$$x_x + x_y \geq 1$$
$$x_y + x_u \geq 1$$
$$0 \leq x_i \leq 1 \quad \forall i$$

Add these together

$$x_u + x_v + x_w + x_x \geq 2.$$

$\therefore$ LP-OPT ≥ 2.

Our solution has value 4.

In general we will design an approx alg. that does two things simultaneously.

A. Picks a vertex cover.

B. Chooses coefficients y_e to multiply the LP constraints so the weighted sum of these constraints, weighted by the chosen coefficients, constitutes a lower bound on LP-OPT.

Ensure that lower bound in part B is at least $\frac{1}{2}$ weight of vertex cover in part A.

<u>Algorithm</u>: (Primal-dual vertex cover)

$z_v = \sum_{e \text{ has } v \text{ as an endpt.}} y_e$

Initialize $y_e = 0 \ \forall e$, $z_v = 0 \ \forall v$, $S = \emptyset$.

While ($\exists$ edge $e = (u,v)$ with no endpoint in S)