Sheet 3

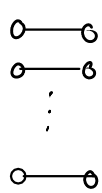
1\. a) Primal-Dual for unweighted VC:
What does it do?

$$\max \sum_{e\in E} y_e$$
$$\text{s.t.} \sum_{(u,v)\in E} y_{(u,v)} \le 1, \quad u \in V$$
$$y_e \ge 0, \quad e \in E$$

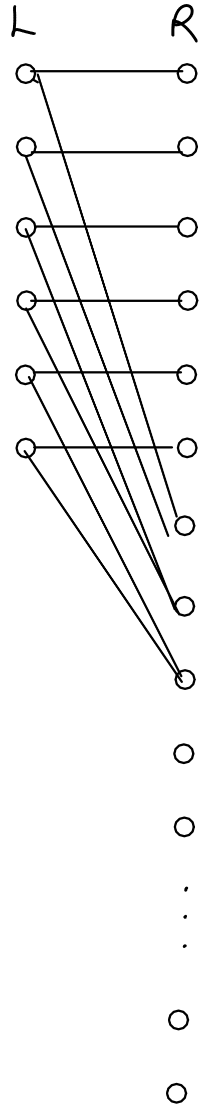
Pick uncovered edge (u,v).
$y_{(u,v)} \leftarrow 1$
Constraints corr. to u and v become tight
Include u and v in VC

b) Write comb. alg. corr. to primal-dual.
Consider edges one by one.
If uncovered, include both endpoints in VC

c) Approx. factor ≥ 2:



2. Greedy for VC – approx. factor > 2:



m vertices, each adj. to 1 vertex

$\lfloor \frac{m}{2} \rfloor$ vertices, each adj. to 2 vertices

$\lfloor \frac{m}{3} \rfloor$ vertices, each adj. to 3 vertices

$\vdots$

$\lfloor \frac{m}{m-1} \rfloor$ vertex adj. to $m-1$ vertices

1 vertex adj. to m vertices

$|L| = m$

$$|R| = \sum_{i=1}^{m} \left\lfloor \frac{m}{i} \right\rfloor > m\ln(m) - m$$

m groups of vertices in R.
Each vertex in L is connected to at most one vertex in each group in R.
Hence, Greedy may start with the group of size $\lfloor \frac{m}{m} \rfloor$, then the group of size $\lfloor \frac{m}{m-1} \rfloor$, then ...

Just before the group of size $\lfloor \frac{m}{m-c} \rfloor$ is picked, each group in L has at most m-c uncovered edges.
Hence, it may continue like this, until all vertices in R have been picked.

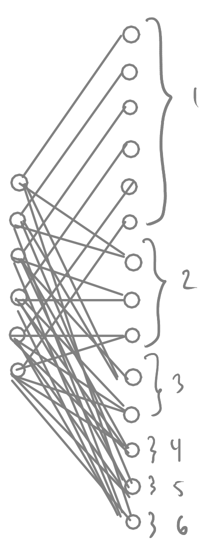
OPT chooses the vertices in L.

$$\frac{\text{Greedy}}{\text{OPT}} = \frac{|R|}{|L|} > \frac{m\ln(m) - m}{m} = \ln(m) - 1$$

$$\in \Theta(\ln n) :$$

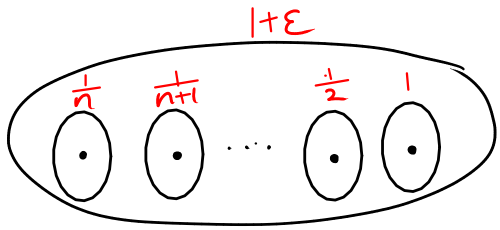
$$n = |L| + |R|$$
$$\Downarrow$$
$$m \ln m < n < m \ln m + m$$
$$\Downarrow$$
$$\ln n \approx \ln m + \ln \ln m$$
$$\Downarrow$$
$$\ln n \in \Theta(\ln m)$$

Ex: $m = 6$



$$\frac{|R|}{|L|} = \frac{6+3+2+1+1+1}{6} = \frac{14}{6} > 2$$

3. Greedy approx. factor $\geq H_n$



$$\frac{\text{Greedy}}{\text{OPT}} = \frac{H_n}{1+\varepsilon} \longrightarrow H_n \text{ for } n \to \infty$$

Section 1.7: Randomized Rounding

Alg RR_1

```
Solve LP
I ← ∅
For j ← 1 to m
    With probability x_j
        I ← I ∪ {j}
```

Expected cost $= Z^*_{LP} \leq OPT$, but the result is most likely <u>not</u> a set cover.

Alg RR_2

```
Solve LP
  I ← ∅
  For i ← 1 to 2·ln(n)
    For j ← 1 to m
        With probability x_j
            I ← I ∪ {j}
```

Expected cost $\leq 2 \cdot \ln(n) \cdot OPT$, and high probability that all elements are covered.
(Calculations below)

Alg RR_3

Solve LP

Repeat

$\quad I \leftarrow \emptyset$

$\quad$ For $i \leftarrow 1$ to $2 \cdot \ln(n)$

$\quad\quad$ For $j \leftarrow 1$ to m

$\quad\quad\quad$ With probability x_j

$\quad\quad\quad\quad I \leftarrow I \cup \{j\}$

Until $\{S_j \mid j \in I\}$ is a set cover

and $w(I) \leq 4 \ln(n) Z^*_{LP}$

Cost $\leq 4 \cdot \ln(n) \cdot OPT$

Result is a set cover.

Expected running time is polynomial.

(Calculations below)

Alg RR_1:

$$\Pr[e_i \text{ not covered}] = \prod_{j: e_i \in S_j} (1 - x_j)$$

$$\leq \prod_{j: e_i \in S_j} e^{-x_j}, \quad \text{since } 1 - s \leq e^{-s}, \text{ for any } s \in \mathbb{R}$$

$$= e^{-\sum_{j: e_i \in S} x_j}$$

$$\leq e^{-1}, \quad \text{by the LP-constraint corresponding to } e_i$$

Alg RR_2:

$$\Pr[e_i \text{ not covered}] \leq (e^{-1})^{2\ln(n)}$$

$$= (e^{\ln(n)})^{-2}$$

$$= n^{-2}$$

$$\Pr[\text{not set cover}] \leq n \cdot \Pr[e_i \text{ not covered}]$$

$$= n \cdot n^{-2}$$

$$= n^{-1}$$

Alg RR_3:

$$\text{Prob}\left[w(I) \le 4 \cdot \ln(n) \cdot Z^*_{LP}\right] \le \frac{1}{2},$$

by Markov's Inequality
since $E[w(I)] \le 2\ln(n)$.

Hence,

$$\text{Prob}[\text{not vertex cover or too expensive}] \le \frac{1}{2} + \frac{1}{n}$$

Thus,

$$\text{Exp. \# iterations} \le \frac{1}{\frac{1}{2} + \frac{1}{n}} \approx 2$$

Sometimes randomized algorithms are simpler / easier to describe/come up with.
Sometimes randomized algorithms can be derandomized.
More about this in Chapter 5.