Notes from Lec 1's PDF
continued and refined
Algorithmic Gap: Does greedy do better than \( \ln n \)?

**Fact:** Greedy no better than \( \ln n (1 - \varepsilon) \). (even for unweighted)

**Pf:**

\[
\begin{align*}
\text{OPT} &= 2, \\
A(S) &= \log(2\varepsilon/2) \\
\Rightarrow \text{gap} &= \frac{\log_2 n - 1}{2} \\
&\leq \frac{\ln n}{2}.
\end{align*}
\]

To set \( k \ln n \), use sets \( S_i \) where \( \text{OPT} = k \) vertically.

But each other set covers \( \frac{1}{k} \) at remainder.

\[
\Rightarrow \# \text{sets} = \left( \frac{1}{k} \right)^n. \Rightarrow \text{gap} = \frac{\log(P(k))n}{k} = \frac{\ln n}{k \ln(1 - \varepsilon)}
\]

but \( \ln(1 + \varepsilon) = \Theta(\varepsilon^2) \)

for \( \varepsilon \) small.

So another algorithm?

Before that: am greedy also for weighted case

At each step, pick set that \( \max \left( \frac{\text{coverage}}{\text{cost}} \right) \).

Thus: Greedy is \( \Theta(\ln n) \)-approximation for weighted set cover.

**Pf:** (Sketch) Same idea as before. Show that if costs \( c_1, c_2 \ldots c_k \)

\[
\begin{align*}
\text{OPT} &\leq n \left( 1 - \frac{c_1}{\text{OPT}} \right) \left( 1 - \frac{c_2}{\text{OPT}} \right) \ldots \left( 1 - \frac{c_k}{\text{OPT}} \right) \\
&\leq n \exp \left( \frac{\sum c_i}{\text{OPT}} \right),
\end{align*}
\]

\( \text{etc.} \)
A charging proof for weighted set cover.

Greedy: Pick set that has the max \( \text{new coverage} \) \( \frac{n_t - n_{t-1}}{C(S_t)} \), until all elements covered.

Claim: \( \text{cost} \leq H_n \cdot \text{cost}(\text{opt}) \).

Proof: wts. \( S_t \) = \( t \)th set picked, \# elements uncovered before pick \( S_t \). \( n_t \) after pick \( S_t \).

\[ \Rightarrow \text{by greedy choice } \frac{n_t - n_{t-1}}{C(S_t)} \geq \frac{n_t}{\text{opt}} \]

\[ \Rightarrow C(S_t) \leq \text{opt} \cdot \frac{n_t - n_{t-1}}{n_t} \]

\[ \Rightarrow \sum_{t} C(S_t) \leq \text{opt} \left[ \frac{n_0 - n_1}{n_0} + \frac{n_1 - n_2}{n_1} + \ldots \right] \]

\[ \text{ALG} = \text{opt} \left[ \frac{1}{n} + \frac{1}{n} + \ldots + \frac{1}{n} \right] \]

\[ \leq \text{opt} \left[ \frac{1}{n} + \frac{1}{n} + \ldots + \frac{1}{n} \right] \]

\[ = \text{opt} \cdot H_n. \]

A different view: charge \( \text{cost} \) of \( C(S_t) \) to all elements newly covered by it, equally.

Now: Consider set \( S^* \) in \( \text{opt} \), covers some elements \( e_1, e_2, \ldots , e_t \)
change to \( e_i \leq \frac{C(S^*)}{t} \) (wlog \( t \) set picked, \( S^* \) at least as good as \( S^* \))

\[ b_t \leq \frac{C(S^*)}{t-1} \]

\[ \Rightarrow \text{total charge} \leq \sum_{S^*} \frac{C(S^*) + H_n}{t} \Rightarrow \text{total charge to elements covered by } S^* \leq \text{opt} \cdot H_n \]
Linear Programming-based Algos:

Idea: Relax and Round

1. Write an IP for Set Cover. \( (IP = \text{Integer (Linear) Program}) \)
2. "Relax" it to an LP \( (LP = \text{Linear Program}) \)
3. Solve this LP.
4. "Round" the fractional solution to integers.

Usually:
1. \( IP(I) = \text{Opt}(I) \).
2. \( LP(I) \leq IP(I) \).
3. \( \text{Alg}(I) \leq \alpha \cdot LP(I) \implies \text{Alg}(I) \leq \alpha \cdot \text{Opt}(I) \).

Set Cover: variable \( x_s \in \{0, 1\} \) for each set \( s \in \{ S_1, S_2, \ldots, S_m \} \).

\[
\begin{align*}
\min & \sum S_s x_s \\
\text{subject to} & \sum x_s = 1 \forall e \in U \\
& x_s \in \{0, 1\} \\
& x_s \geq 0
\end{align*}
\]

Round: Imagine each \( x_s \) as a prob. value. \( (\text{Fact: } x_s \in [0,1], \text{no reason for } x_s \text{ to be integer}) \)

Also: \( \left[ \begin{array}{c}
\text{For } T \text{ times} \\
\forall s \in F \\
\text{select } S \text{ independently up } x_s \\
\end{array} \right] \) T rounds a sampling
What if this is not a feasible solution?

**Clean-up:** For element \( e \), pick cheapest set covering \( e \) if \( e \) not covered by sampling.

---

**Lemma:** \( E[\text{cost of solution}] \leq T \cdot LP(I) + \left[ \sum_{e \in S} (\text{cheapest set covering}) \right] e^{-T} \)

**Pf:** \( E[\text{cost of each round}] = \sum_{c} c_i \cdot \Pr[S \text{ picked}] = \sum_{c} c_i x_c = LP(I) \)

\[ \Pr[\text{en not covered in T rounds}] \leq e^{-T} \]

Now use linearity of expectation again.

---

Hence set \( T = \frac{1}{\ln n} \).

\[ E[\text{cost}] \leq (\ln n) \cdot LP + \frac{1}{\ln n} \cdot LP \]

\[ = (\ln n + 1) \cdot LP \]

(b/c LP value \( \geq \) cheapest set covering for any \( e \))

HW: Show that if set size \( B \), then LP randomly gives \( O(\ln B) \) apx.

Greedy too (but see more later).

---

**Picture**

```
\[ \text{OPT}(I) \]
\[ \leq \log n \]
```

\[ \text{LP}(I) \quad \Rightarrow \quad \text{IP}(I) \quad \Rightarrow \quad \text{Alg}(I) \]

\[ \Rightarrow \text{increase} \]
Ask 2 questions:

1. **Algorithmic gap**: does instance where \( \frac{A_f(L)}{\text{OPT}(L)} = \Omega(\log n) \).

2. **Infeasibility gap**: does instance \( \frac{\text{OPT}(L)}{\text{LP}(L)} = \Omega(\log n) \).

\( \text{LP} \) shows that using this approach cannot beat the log-approximation.

\[ \text{no matter what we do.} \]

[as long as we relate ourselves to the LP value, of course!]

Also gap: \( \text{see in HW} \).

Infeasibility gaps:

- Take \( L = \frac{3}{2} \times \{ 0, 1, \ldots, d \} \times \{ \frac{d}{2} \} \)

\( n = |U| = (\frac{d}{2})^d = \Theta\left(\frac{2^d}{\sqrt{d}}\right) \).

- \( \text{Set 6: all "dictator" sets } S_i = \frac{d}{2} \times \{ 0, 1 \} \times \{ \frac{d}{2} \} \).

\( \text{cost} = 1. \)

- \( \text{OPT} \geq \frac{d}{2} + 1 \) else \( \exists \) element not covered.

- \( \text{LP value: set } \frac{2}{d} \text{ on each set } S_i \) (ie. \( x_i = 1 + \epsilon S \)).

\( \Rightarrow \) total LP value = \( d \times \frac{2}{d} = 2 \).

\( \Rightarrow \) infeasibility gap \( \geq \frac{d}{2} \times \left( \frac{d}{2} + 1 \right) \), \( \Rightarrow \Theta(d) = \Theta(\log n) \).

Fact: can do better, get HW for infeasibility gap as well.
Derandomizing the Randomized Aho - (Pippage Rounding)

\[ \text{Sp s.t. } x^* = \text{OPT fractional LP soln.} \]
\[ z^* = \sum \zeta \]
\[ \text{LP } = \sum \zeta x^* \]

**Fact 1:** LP \( \geq \) OPT (relaxation).

In "idealized" rounding:

Pick each set \( S \) up \( x_s \), indep.

**Fact 2:** 
\[
E[\text{coverage}] = \sum_{s \in \epsilon} (1 - e^{-\epsilon}) \geq \sum_{s \in \epsilon} (1 - e^{-2\epsilon}) \geq \sum_{s \in \epsilon} (1 - e^{-\epsilon})
\]
\[
\text{LP} \geq \sum_{s \in \epsilon} f(x) = \sum_{s \in \epsilon} f(x)
\]
\[
f(x) = \frac{1}{|S|} f(x)
\]

**Fact 3:** But only guarantees that we pick \( k \) elements w/ respect to optimized.

**Fact 4:** If solution is integral, then coverage = \( f(x) \).

and set = \( k \) sets

So idea:

\[ x^* \rightarrow x_0 \rightarrow x_{\epsilon} \rightarrow \ldots \rightarrow x_{\text{int}} \]

Integral.

\[ f(x^*) \leq f(x_{\epsilon}) \leq f(x_{\text{int}}) \leq f(x_{\text{int}}) \]

\[ \geq (1 - \epsilon) \text{LP} \]

\[ \geq (1 - \epsilon) \text{OPT} \]

\[ \geq \text{fact 1} \]

\[ \text{but this is integral so actually tells you soln.} \]
How to find (increase integer coordinates)?

Assume \( \sum x_s = k \) else raise some \( x \) until sat.

If \( x \) fractional, has at least 2 fractional sets

say \( x_{s_1}, x_{s_2} \)

"Move along line \( e_{s_1} - e_{s_2} \)"

Consider solution \( x^* \leftarrow x + \varepsilon (e_{s_1} - e_{s_2}) \)

Let \( g(\varepsilon) = f(x^*) \) be univariate function in \( \varepsilon \).

- \( g(0) = f(x^0) = f(x) \)

- \( g(\varepsilon) = \sum f_\varepsilon (x^*) = \sum f_\varepsilon (x^0) + \sum f_\varepsilon (x^0) + \sum f_\varepsilon (x^0) + \sum f_\varepsilon (x^0) \)

\[
1 - \left[ \prod_{s \neq s_1} (1-x_{s}) \right] (1-x_{s_1}-\varepsilon)
\]

= linear fn in \( \varepsilon \).

Also linear fn in \( \varepsilon \)

Claim: \( g(\varepsilon) \) is convex in \( \varepsilon \).

\[
\Rightarrow \text{consider } x_{s_0} \text{ st } x_{s_1} \text{ reaches 1 or } x_{s_2} \text{ reaches 0}
\]

\[
-x_{s_0} \text{ st } x_{s_1} \text{ reaches 0 or } x_{s_2} \text{ reaches 1}
\]

One must be at least as high as \( x_0 \)!