Approximation Algorithms

Lecture 5:
LP-based Approximation Algorithms for SetCover

Part I:
SetCover as an ILP
SetCover as an ILP

\[
\begin{align*}
\text{minimize} & \quad \sum_{S \in \mathcal{S}} c_S x_S \\
\text{subject to} & \quad \sum_{S \ni u} x_S \geq 1 \quad u \in \mathcal{U} \\
& \quad x_S \in \{0, 1\} \quad S \in \mathcal{S}
\end{align*}
\]

Ground set \( \mathcal{U} \)
Family \( \mathcal{S} \subseteq 2^{\mathcal{U}} \) with \( \bigcup \mathcal{S} = \mathcal{U} \)
Costs \( c : \mathcal{S} \rightarrow \mathbb{Q}^+ \)

Find cover \( \mathcal{S}' \subseteq \mathcal{S} \) of \( \mathcal{U} \) with minimum cost.
Approximation Algorithms

Lecture 5:
LP-based Approximation Algorithms for SetCover

Part II:
LP-Rounding

Alexander Wolff
Winter 2022/23
Consider a minimization problem $\Pi$ in ILP form.

Compute a solution for the LP-relaxation.

Round to obtain an integer solution for $\Pi$.

Difficulty: Ensure the feasibility of the solution.

Approximation factor: $\frac{\text{ALG}}{\text{OPT}_\Pi} \leq \frac{\text{ALG}}{\text{OPT}_{\text{relax}}}$. 

Technique I) LP-Rounding
SetCover – LP-Relaxation

\[
\begin{align*}
\text{minimize} & \quad \sum_{S \in S} c_S x_S \\
\text{subject to} & \quad \sum_{S \ni u} x_S \geq 1 \quad u \in U \\
& \quad x_S \geq 0 \quad S \in S
\end{align*}
\]

Optimal?

integer: 2

fractional: \( \frac{3}{2} \)
LP-Rounding: Approach I

minimize $\sum_{S \in S} c_S x_S$

subject to $\sum_{S \ni u} x_S \geq 1 \quad u \in U$

$x_S \geq 0 \quad S \in S$

LP-Rounding-One($U, S, c$)

Compute optimal solution $x$ for LP-relaxation.
Round each $x_S$ with $x_S > 0$ to 1.

– Generates a valid solution.
– Scaling factor arbitrarily large.

Use frequency $f$
LP-Rounding: Approach II

\[
\begin{align*}
\text{minimize} & \quad \sum_{S \in S} c_S x_S \\
\text{subject to} & \quad \sum_{S \ni u} x_S \geq 1 & u \in U \\
& \quad x_S \geq 0 & S \in S
\end{align*}
\]

LP-Rounding-Two\((U, S, c)\)
Compute optimal solution \(x\) for LP-Relaxation.
Round each \(x_S\) with \(x_S \geq 1/f\) to 1; remaining to 0.

Let \(f\) be the frequency of (i.e., the number of sets containing) the most frequent element.

**Theorem.** LP-Rounding-Two is a factor-\(f\) approximation algorithm for SetCover.
Approximation Algorithms

Lecture 5:
LP-based Approximation Algorithms for SetCover

Part III:
The Primal-Dual Schema
Consider a minimization problem $\Pi$ in ILP form.

Start with (trivial) feasible dual solution and infeasible primal solution (e.g., all variables $= 0$).

Compute dual solution $s_d$ and integral primal solution $s_\Pi$ for $\Pi$ iteratively:

increase $s_d$ according to CS and make $s_\Pi$ “more feasible”.

Approximation factor \( \leq \frac{\text{obj}(s_\Pi)}{\text{obj}(s_d)} \)

Advantage: don’t need LP-“machinery”; possibly faster, more flexible.
SetCover – Dual LP

minimize \[ \sum_{S \in S} c_S x_S \]
subject to \[ \sum_{S \ni u} x_S \geq 1 \quad u \in U \]
\[ x_S \geq 0 \quad S \in S \]

maximize \[ \sum_{u \in U} y_u \]
subject to \[ \sum_{u \in S} y_u \leq c_S \quad S \in S \]
\[ y_u \geq 0 \quad u \in U \]
**Theorem.** Let \( x = (x_1, \ldots, x_n) \) and \( y = (y_1, \ldots, y_m) \) be valid solutions for the primal and dual program (resp.). Then \( x \) and \( y \) are optimal if and only if the following conditions are met:

**Primal CS:**
For each \( j = 1, \ldots, n: \ x_j = 0 \) or \( \sum_{i=1}^{m} a_{ij} y_i = c_j \)

**Dual CS:**
For each \( i = 1, \ldots, m: \ y_i = 0 \) or \( \sum_{j=1}^{n} a_{ij} x_j = b_i \)
Relaxing Complementary Slackness

**Primal CS:**
For each \( j = 1, \ldots, n \):
\[
x_j = 0 \quad \text{or} \quad \sum_{i=1}^{m} a_{ij} y_i = c_j
\]
\[
c_j / \alpha \leq \sum_{i=1}^{m} a_{ij} y_i \leq c_j
\]

**Dual CS:**
For each \( i = 1, \ldots, m \):
\[
y_i = 0 \quad \text{or} \quad \sum_{j=1}^{n} a_{ij} x_j = b_i
\]
\[
b_i \leq \sum_{j=1}^{n} a_{ij} x_j \leq \beta \cdot b_i
\]

\[
\Leftrightarrow \sum_{j=1}^{n} c_j x_j = \sum_{i=1}^{m} b_i y_i
\]
\[
\Rightarrow \sum_{j=1}^{n} c_j x_j \leq \alpha \beta \sum_{i=1}^{m} b_i y_i \leq \alpha \beta \cdot \text{OPT}_\text{LP}
\]
Primal–Dual Schema

Start with a feasible **dual** and infeasible **primal** solution (often trivial).

“Improve” the feasibility of the **primal** solution...

...and simultaneously the obj. value of the **dual** solution.

Do so until the relaxed CS conditions are met.

Maintain that the **primal** solution is integer valued.

The feasibility of the **primal** solution and relaxed CS condition provide an approximation ratio.
## Relaxed CS for SetCover

<table>
<thead>
<tr>
<th>Minimize</th>
<th>Maximize</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \sum_{S \in S} c_S x_S )</td>
<td>( \sum_{u \in U} y_u )</td>
</tr>
<tr>
<td>subject to</td>
<td>subject to</td>
</tr>
<tr>
<td>( \sum_{S \ni u} x_S \geq 1 )</td>
<td>( \sum_{u \in S} y_u \leq c_S )</td>
</tr>
<tr>
<td>( x_S \geq 0 )</td>
<td>( y_u \geq 0 )</td>
</tr>
<tr>
<td>( S \in S )</td>
<td>( u \in U )</td>
</tr>
</tbody>
</table>

### Critical Set

**Unrelaxed** primal CS: \( x_S \neq 0 \) \( \Rightarrow \sum_{u \in S} y_u = c_S \)

**Relaxed** dual CS: \( y_u \neq 0 \) \( \Rightarrow 1 \leq \sum_{S \ni u} x_S \leq f \cdot 1 \)

**Trivial for Binary** \( x \)
Primal–Dual Schema for SetCover

PrimalDualSetCover(\(U, S, c\))

\[ x \leftarrow 0,\ y \leftarrow 0 \]

repeat

Select an uncovered element \(u\).
Increase \(y_u\) until a set \(S\) is critical (\(\sum_{u' \in S} y_{u'} = c_S\)).
Select all critical sets and update \(x\).
Mark all elements in these sets as covered.

until all elements are covered.

return \(x\)
Primal–Dual Schema for SetCover

PrimalDualSetCover(U, S, c)

\[ x \leftarrow 0, \ y \leftarrow 0 \]

repeat

Select an uncovered element \( u \).
Increase \( y_u \) until a set \( S \) is critical (\( \sum_{u' \in S} y_{u'} = c_S \)).
Select all critical sets and update \( x \).
Mark all elements in these sets as covered.

until all elements are covered.

return \( x \)

Theorem. PrimalDualSetCover is a factor-\( f \) approximation algorithm for SetCover. This bound is tight.
Tight Example

\[ 1 + \varepsilon \]
Consider a minimization problem $\Pi$ in ILP form.

Dual methods (without outside help) are limited by the *integrality gap* of the LP-relaxation

$$\alpha \geq \gamma = \sup_{I} \frac{\text{OPT}_{\Pi}(I)}{\text{OPT}_{\text{primal}}(I)}$$
Approximation Algorithms

Lecture 5:
LP-based Approximation Algorithms for SetCover

Part IV:
Dual Fitting

Alexander Wolff
Winter 2022/23
Technique III) Dual Fitting

Consider a minimization problem $\Pi$ in ILP form.

Combinatorial algorithm (e.g., greedy) computes feasible primal solution $s_\Pi$ and infeasible dual solution $s_d$ that completely “pays” for $s_\Pi$, i.e., $\text{obj}(s_\Pi) \leq \text{obj}(s_d)$.

Scale the dual variables $\mapsto$ feasible dual solution $\tilde{s}_d$.

$$\Rightarrow \frac{\text{obj}(s_\Pi)}{\alpha} \leq \frac{\text{obj}(s_d)}{\alpha} = \frac{\text{obj}(\tilde{s}_d)}{\alpha} \leq \text{OPT}_{\text{dual}} \leq \text{OPT}_\Pi$$

$\Rightarrow$ Scaling factor $\alpha$ is approximation factor.
Dual Fitting for SetCover

Combinatorial (greedy) algorithm (see Lecture #2):

GreedySetCover(\(U, S, c\))

\[
\begin{align*}
C &\leftarrow \emptyset \\
S' &\leftarrow \emptyset \\
\text{while } C \neq U &\text{ do} \\
S &\leftarrow \text{Set from } S \text{ that minimizes } \frac{c(S)}{|S\setminus C|} \\
\text{foreach } u \in S \setminus C &\text{ do} \\
\text{price}(u) &\leftarrow \frac{c(S)}{|S\setminus C|} \\
C &\leftarrow C \cup S \\
S' &\leftarrow S' \cup \{S\}
\end{align*}
\]

return \(S'\) // Cover of \(U\)

Reminder: \(\sum_{u \in U} \text{price}(u)\) completely pays for \(S'\).
New: LP-based Analysis

**Observation.** For each \( u \in U \), \( \text{price}(u) \) is a dual variable. But this dual solution is in general not feasible.

Homework exercise: Construct instance where some \( S \) are “overpacked” by factor \( \approx H |S| \).

**Dual-fitting trick:**
Scale dual variables such that no set is overpacked.
Take \( \bar{y}_u = \text{price}(u) / H_k \). \((k = \text{cardinality of largest set in } S.)\)
The greedy algorithm uses these dual variables as lower bound for OPT.

\[
\begin{align*}
\text{maximize} & \quad \sum_{u \in U} y_u \\
\text{subject to} & \quad \sum_{u \in S} y_u \leq c_S \quad S \in S \\
& \quad y_u \geq 0 \quad u \in U
\end{align*}
\]
Proof. To prove: No set is overpacked by $\bar{y}$.
Let $S \in S$ and $\ell = |S| \leq k$.
Let $u_1, \ldots, u_\ell$ be the elements of $S$ – in the order in which they are covered by greedy.
Consider the iteration in which $u_i$ is covered.
Before that, $\geq \ell - i + 1$ elem. of $S$ are uncovered.
So $\text{price}(u_i) \leq \frac{c(S)}{(\ell - i + 1)}$.

\[ \Rightarrow \bar{y}_{u_i} \leq \frac{c(S)}{\mathcal{H}_k} \cdot \frac{1}{\ell - i + 1} \Rightarrow \sum_{i=1}^{\ell} \bar{y}_{u_i} \leq \frac{c(S)}{\mathcal{H}_k} \cdot \left( \frac{1}{\ell} + \cdots + \frac{1}{1} \right) \leq c(S) \]

Lemma.
The vector $\bar{y} = (\bar{y}_u)_{u \in U}$ is a feasible solution for the dual LP.

\[
\begin{align*}
\text{maximize} & \quad \sum_{u \in U} y_u \\
\text{subject to} & \quad \sum_{u \in S} y_u \leq c_S \quad S \in S \\
& \quad y_u \geq 0 \quad u \in U
\end{align*}
\]
Result for Dual Fitting

Theorem. GreedySetCover is a factor-$\mathcal{H}_k$ approximation algorithm for SetCover, where $k = \max_{S \in S} |S|$.

Proof. $\text{ALG} = c(S') \leq \sum_{u \in U} \text{price}(u) = \mathcal{H}_k \cdot \sum_{u \in U} \bar{y}_u \leq \mathcal{H}_k \cdot \text{OPT}_{\text{relax}} \leq \mathcal{H}_k \cdot \text{OPT} \quad \square$

Strengthened bound with respect to $\text{OPT}_{\text{relax}} \leq \text{OPT}$.

Dual solution allows a per-instance estimation

…which may be stronger than worst-case bound $\mathcal{H}_k$. 