ILP-Based Approximations
CSCI 532

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$x_i \in \{0,1\}$, for each vertex i

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$\boldsymbol{x_i \in \{0, 1\}}$, **for each vertex** $\boldsymbol{i}$

$\in$ NP-Complete

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$\boldsymbol{x_i \in \{0, 1\}}$**, for each vertex** $\boldsymbol{i}$

∈ NP-Complete

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$\boldsymbol{0 \leq x_i \leq 1}$**, for each vertex** $\boldsymbol{i}$

∈ P

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$\boldsymbol{x_i \in \{0, 1\}}$**, for each vertex** $\boldsymbol{i}$

$\in$ NP-Complete

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$\boldsymbol{0 \leq x_i \leq 1}$**, for each vertex** $\boldsymbol{i}$

$\in$ P

LP Relaxation: Remove all integrality constraints to turn ILP into LP.

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

?

Vertex Selection

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

?

Vertex Selection

If $x_i = 1$, what should we do with vertex i?

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective: $\min \sum_i x_i$
Subject to: $x_i + x_j \geq 1$, for each edge $e = (i, j)$
$0 \leq x_i \leq 1$, for each vertex i

?

Vertex Selection

If $x_i = 1$, what should we do with vertex i? Add to subset S
If $x_i = 0$, what should we do with vertex i?

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective: $\min \sum_i x_i$
Subject to: $x_i + x_j \geq 1$, for each edge $e = (i, j)$
$0 \leq x_i \leq 1$, for each vertex i

$\Rightarrow$? $\Rightarrow$ Vertex Selection

If $x_i = 1$, what should we do with vertex i? Add to subset S
If $x_i = 0$, what should we do with vertex i? Don't add to subset S

Vertex Cover ILP

Vertex Cover: Given a graph, find the smallest subset of vertices such that every edge in the graph has at least one vertex in the subset.

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

? $\Rightarrow$ Vertex Selection

If $x_i = 1$, what should we do with vertex i? Add to subset S

If $x_i = 0$, what should we do with vertex i? Don't add to subset S

If $x_i = \frac{126}{337}$, what should we do with vertex i?

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Vertex Cover ILP

Objective: $\min \sum_i x_i$
Subject to: $x_i + x_j \geq 1$, for each edge $e = (i, j)$
$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Is S a vertex cover?

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

$+$ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Is S a vertex cover?

Yes. For every edge, $x_i + x_j \geq 1$.

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

$+$ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Is S a vertex cover?

Yes. For every edge, $x_i + x_j \geq 1$. Thus, at least one of x_i or $x_j \geq \frac{1}{2}$.

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Is S a vertex cover?

Yes. For every edge, $x_i + x_j \geq 1$. Thus, at least one of x_i or $x_j \geq \frac{1}{2}$. So for every edge, at least one of its vertices will be in S.

Vertex Cover ILP

Objective: $\min \sum_i x_i$
Subject to: $x_i + x_j \geq 1$, for each edge $e = (i, j)$
$0 \leq x_i \leq 1$, for each vertex i

$+$

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

What is the relationship between $\text{ALG} = |S|$ and OPT?

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

$+$ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Can we bound OPT from below?

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

$+$ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Can we bound OPT from below?

Let x_{ILP} and x_{LP} be the set of x values found by the ILP and LP

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Can we bound OPT from below?

Let x_{ILP} and x_{LP} be the set of x values found by the ILP and LP

Claim: $\sum x_{LP} \leq$ OPT.

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

Can we bound OPT from below?

Let x_{ILP} and x_{LP} be the set of x values found by the ILP and LP

Claim: $\sum x_{LP} \leq$ OPT.

Proof: OPT $= \sum x_{ILP}$, where $x_i \in \{0,1\}$. When x_i is relaxed so that $0 \leq x_i \leq 1$, this gives more possibilities to further decrease $\sum_i x_i$. Thus, $\sum x_{LP} \leq$ OPT.

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

$+$ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x}_{\mathrm{LP}}$ relate to ALG?

$$\sum \mathrm{x}_{\mathrm{LP}} = \sum_{x_i \in \mathrm{x}_{\mathrm{LP}}} x_i \geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2}} x_i, \text{ because...?}$$

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x_{LP}}$ relate to ALG?

$\sum \mathrm{x_{LP}} = \sum_{x_i \in \mathrm{x_{LP}}} x_i \geq \sum_{x_i \in \mathrm{x_{LP}} : x_i \geq \frac{1}{2}} x_i$, because it's a subset of $\mathrm{x_{LP}}$

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x}_{\mathrm{LP}}$ relate to ALG?

$$\sum \mathrm{x}_{\mathrm{LP}} = \sum_{x_i \in \mathrm{x}_{\mathrm{LP}}} x_i \geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2}} x_i \text{, because it's a subset of } \mathrm{x}_{\mathrm{LP}}$$

$$\geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2}} \frac{1}{2} \text{, because...?}$$

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x_{LP}}$ relate to ALG?

$$\sum \mathrm{x_{LP}} = \sum_{x_i \in \mathrm{x_{LP}}} x_i \geq \sum_{x_i \in \mathrm{x_{LP}} : x_i \geq \frac{1}{2}} x_i\text{, because it's a subset of } \mathrm{x_{LP}}$$

$$\geq \sum_{x_i \in \mathrm{x_{LP}} : x_i \geq \frac{1}{2}} \frac{1}{2}\text{, because each } x_i \text{ is at least } \frac{1}{2}$$

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x}_{\mathrm{LP}}$ relate to ALG?

$$\sum \mathrm{x}_{\mathrm{LP}} = \sum_{x_i \in \mathrm{x}_{\mathrm{LP}}} x_i \geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}}: x_i \geq \frac{1}{2}} x_i, \text{ because it's a subset of } \mathrm{x}_{\mathrm{LP}}$$

$$\geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}}: x_i \geq \frac{1}{2}} \frac{1}{2}, \text{ because each } x_i \text{ is at least } \frac{1}{2}$$

$$= \frac{1}{2} \left| \left\{ x_i \in \mathrm{x}_{\mathrm{LP}}: x_i \geq \frac{1}{2} \right\} \right|$$

Vertex Cover ILP

Objective: $\min \sum_i x_i$
Subject to: $x_i + x_j \geq 1$, for each edge $e = (i, j)$
$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x}_{\mathrm{LP}}$ relate to ALG?

$$\sum \mathrm{x}_{\mathrm{LP}} = \sum_{x_i \in \mathrm{x}_{\mathrm{LP}}} x_i \geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2}} x_i, \text{ because it's a subset of } \mathrm{x}_{\mathrm{LP}}$$

$$\geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2}} \frac{1}{2}, \text{ because each } x_i \text{ is at least } \frac{1}{2}$$

$$= \frac{1}{2} \left| \left\{ x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2} \right\} \right| = ?$$

Vertex Cover ILP

Objective: $\min \sum_i x_i$
Subject to: $x_i + x_j \geq 1$, for each edge $e = (i, j)$
$0 \leq x_i \leq 1$, for each vertex i

+

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x}_{\mathrm{LP}}$ relate to ALG?

$$\sum \mathrm{x}_{\mathrm{LP}} = \sum_{x_i \in \mathrm{x}_{\mathrm{LP}}} x_i \geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2}} x_i, \text{ because it's a subset of } \mathrm{x}_{\mathrm{LP}}$$

$$\geq \sum_{x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2}} \frac{1}{2}, \text{ because each } x_i \text{ is at least } \frac{1}{2}$$

$$= \frac{1}{2} \left| \left\{ x_i \in \mathrm{x}_{\mathrm{LP}} : x_i \geq \frac{1}{2} \right\} \right| = \frac{1}{2} \text{ ALG}$$

Vertex Cover ILP

Objective:	$\min \sum_i x_i$
Subject to:	$x_i + x_j \geq 1$, for each edge $e = (i, j)$
	$0 \leq x_i \leq 1$, for each vertex i

$+$

If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

What is the relationship between ALG and OPT?

$\sum x_{LP} \geq \frac{1}{2}$ ALG and $\sum x_{LP} \leq$ OPT

ALG ≤ 2 OPT

Set Cover ILP

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

$$U = \{1, 4, 7, 8, 10\}$$

$$S = \begin{Bmatrix} \{1, 7, 8\}, \{1, 4, 7\}, \\ \{7, 8\}, \{4, 8, 10\} \end{Bmatrix}$$

Set Cover ILP

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective: $\min \sum_s x_s$

Subject to: $\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$

$x_s \in \{0,1\}$, for each set s

$$U = \{1, 4, 7, 8, 10\}$$

$$S = \left\{ \begin{matrix} \{1, 7, 8\}, \{1, 4, 7\}, \\ \{7, 8\}, \{4, 8, 10\} \end{matrix} \right\}$$

Set Cover ILP

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

<u>Example:</u>

Objective:	$\min x_1 + x_2 + x_3 + x_4$
Subject to:	$x_1 + x_2 \geq 1$
	$x_2 + x_4 \geq 1$
	$x_1 + x_2 + x_3 \geq 1$
	$x_1 + x_3 + x_4 \geq 1$
	$x_4 \geq 1$
	$x_1, x_2, x_3, x_4 \in \{0,1\}$

$U = \{1, 4, 7, 8, 10\}$

$S = \left\{ \begin{matrix} \{1,7,8\}, \{1,4,7\}, \\ \{7,8\}, \{4,8,10\} \end{matrix} \right\}$

Set Cover ILP

Objective: $\min \sum_s x_s$

Subject to: $\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$

$0 \leq x_s \leq 1$, for each set s

+

If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.

Set Cover ILP

Objective: $\min \sum_s x_s$
Subject to: $\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
$0 \leq x_s \leq 1$, for each set s

+

If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.

Could this lead to an invalid solution?

Set Cover ILP

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$0 \leq x_s \leq 1$, for each set s

+

If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.

Could this lead to an invalid solution?

Objective:	$\min x_1 + x_2 + x_3 + x_4$
Subject to:	$x_1 + x_2 + x_3 \geq 1$
	$x_1 + x_2 + x_4 \geq 1$
	$x_1 + x_3 + x_4 \geq 1$
	$x_2 + x_3 + x_4 \geq 1$
	$x_1, x_2, x_3, x_4 \in [0,1]$

$U = \{1, 2, 3, 4\}$

$S = \begin{Bmatrix} \{1,2,3\}, \{1,2,4\}, \\ \{1,3,4\}, \{2,3,4\} \end{Bmatrix}$

Set Cover ILP

Objective: $\min \sum_s x_s$
Subject to: $\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
$0 \leq x_s \leq 1$, for each set s

$+$

If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.

Could this lead to an invalid solution?

Objective: $\min x_1 + x_2 + x_3 + x_4$
Subject to: $x_1 + x_2 + x_3 \geq 1$
$x_1 + x_2 + x_4 \geq 1$
$x_1 + x_3 + x_4 \geq 1$
$x_2 + x_3 + x_4 \geq 1$
$x_1, x_2, x_3, x_4 \in [0,1]$

$U = \{1, 2, 3, 4\}$

$S = \begin{Bmatrix} \{1,2,3\}, \{1,2,4\}, \\ \{1,3,4\}, \{2,3,4\} \end{Bmatrix}$

Yes, in this case $x_s = \frac{1}{3}, \forall s \Rightarrow$ No sets are selected (invalid solution).

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

+

If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

+ ~~If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.~~

+ ?

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

\+ ~~If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.~~

\+ If $x_s \geq \frac{1}{d}$, add set s to our subset S_{ALG}.

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

Valid?

+ ~~If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.~~

+ If $x_s \geq \frac{1}{d}$, add set s to our subset S_{ALG}.

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

Valid? ✓

$+$ ~~If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.~~

$+$ If $x_s \geq \frac{1}{d}$, add set s to our subset S_{ALG}.

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

$+$ ~~If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.~~

$+$ If $x_s \geq \frac{1}{d}$, add set s to our subset S_{ALG}.

Valid? ✓

Approximation Ratio?

Vertex Cover ILP

Objective: $\min \sum_i x_i$
Subject to: $x_i + x_j \geq 1$, for each edge $e = (i, j)$
$0 \leq x_i \leq 1$, for each vertex i

$+$ If $x_i \geq \frac{1}{2}$, add vertex i to our subset S.

How does $\sum \mathrm{x_{LP}}$ relate to ALG?

$$\sum \mathrm{x_{LP}} = \sum_{x_i \in \mathrm{x_{LP}}} x_i \geq \sum_{x_i \in \mathrm{x_{LP}} : x_i \geq \frac{1}{2}} x_i, \text{ because it's a subset of } \mathrm{x_{LP}}$$

$$\geq \sum_{x_i \in \mathrm{x_{LP}} : x_i \geq \frac{1}{2}} \frac{1}{2}, \text{ because each } x_i \text{ is at least } \frac{1}{2}$$

$$= \frac{1}{2} \left| \left\{ x_i \in \mathrm{x_{LP}} : x_i \geq \frac{1}{2} \right\} \right| = \frac{1}{2} \text{ALG}$$

Set Cover ILP

Where each element appears in at most d sets.

Set Cover: Given a universe of elements U and sets S, find the smallest subset of S such that every element in U is in some selected subset.

Objective:	$\min \sum_s x_s$
Subject to:	$\sum_{s:\, u \in s} x_s \geq 1$, for each $u \in U$
	$x_s \in \{0,1\}$, for each set s

+ ~~If $x_s \geq \frac{1}{2}$, add set s to our subset S_{ALG}.~~

+ If $x_s \geq \frac{1}{d}$, add set s to our subset S_{ALG}.

Valid? ✓

Approximation Ratio?

$$\text{ALG} \leq d \text{ OPT}$$

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

$W = 11$

Item	Value	Weight
1	1	1
2	6	2
3	18	5
4	22	6
5	28	7

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

$W = 11$

Item	Value	Weight	Ratio
1	1	1	1
2	6	2	3
3	18	5	3.6
4	22	6	3.67
5	28	7	4

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

$W = 11$

Item	Value	Weight	Ratio
1	1	1	1
2	6	2	3
3	18	5	3.6
4	22	6	3.67
5	28	7	4

Algorithm: Select highest value-weight items until no space is left.

Approximation Ratio?

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

$W = 11$

Item	Value	Weight	Ratio
1	1	1	1
2	6	2	3
3	18	5	3.6
4	22	6	3.67
5	28	7	4

Algorithm: Select highest value-weight items until no space is left.

Approximation Ratio?

Are there circumstances where this will do very poorly?

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2

Algorithm: Select highest value-weight items until no space is left.

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2

Algorithm: Select highest value-weight items until no space is left.

$$OPT = ?$$

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2

Algorithm: Select highest value-weight items until no space is left.

$OPT = W$
$ALG = ?$

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2

Algorithm: Select highest value-weight items until no space is left.

$OPT = W$

$ALG = 2$

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2

Algorithm: Select highest value-weight items until no space is left.

$$OPT = W$$
$$ALG = 2$$

$$ALG \geq \frac{1}{\alpha} OPT$$

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2

Algorithm: Select highest value-weight items until no space is left.

$$OPT = W$$
$$ALG = 2$$

$$ALG \geq \frac{1}{\alpha} OPT \Rightarrow \alpha \geq \frac{W}{2}$$

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2

Algorithm: Select highest value-weight items until no space is left.

$$OPT = W$$
$$ALG = 2$$

$$ALG \geq \frac{1}{\alpha} OPT \Rightarrow \alpha \geq \frac{W}{2} \xrightarrow[W \to \infty]{} \infty$$

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2
3	ε	1	ε
…	…	…	…
W+1	ε	1	ε

Algorithm: Select highest value-weight items until no space is left.

$OPT = W$

$ALG = 2$

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Example:

Item	Value	Weight	Ratio
1	W	W	1
2	2	1	2
3	ε	1	ε
…	…	…	…
W+1	ε	1	ε

Algorithm: Select highest value-weight items until no space is left.

$$OPT = W$$
$$ALG = 2 + (W - 1)\varepsilon$$

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Suppose we could select fractional items.

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Suppose we could select fractional items.

OPT_{frac} = ?

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Suppose we could select fractional items.

OPT_{frac} = Items selected by greedy algorithm up to some item that doesn't fit + fractional amount of that item to fill sack.

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Suppose we could select fractional items.

OPT_{frac} = Items selected by greedy algorithm up to some item that doesn't fit + fractional amount of that item to fill sack.

$OPT \leq OPT_{frac}$

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Suppose we could select fractional items.

OPT_{frac} = Items selected by greedy algorithm up to some item that doesn't fit + fractional amount of that item to fill sack.

$$OPT \leq OPT_{frac} \leq v(ALG_1) + v_j$$

Need to show.

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Knapsack

Knapsack: Given a set of n items with values $v_1, \dots, v_n$ and weights $w_1, \dots, w_n$, select the most valuable combination with total weight $\leq W$.

Suppose we could select fractional items.

OPT_{frac} = Items selected by greedy algorithm up to some item that doesn't fit + fractional amount of that item to fill sack.

$$OPT \leq OPT_{frac} \leq v(ALG_1) + v_j$$

$$\Rightarrow \max\{v(ALG_1), v_j\} \geq \frac{OPT}{2}$$

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Knapsack

Knapsack: Given a set of n items with values $v_1, \ldots, v_n$ and weights $w_1, \ldots, w_n$, select the most valuable combination with total weight $\leq W$.

Suppose we could select fractional items.

OPT_{frac} = Items selected by greedy algorithm up to some item that doesn't fit + fractional amount of that item to fill sack.

$$OPT \leq OPT_{frac} \leq v(ALG_1) + v_j$$

$$\Rightarrow \max\{v(ALG_1), v_j\} \geq \frac{OPT}{2}$$

Algorithm: Select highest value-weight items until no space is left OR select the single highest valued item.

Work Scheduling

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Algorithm: Assign each task to the worker with the smallest current work time.

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\, OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}).

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}). Why was W_{ALG}'s last task i assigned to it?

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}). Why was W_{ALG}'s last task i assigned to it? $ALG - t_i \leq T_w$, for all workers.

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}). Why was W_{ALG}'s last task i assigned to it? $ALG - t_i \leq T_w$, for all workers.

This holds when task i is assigned and at the end of the schedule, since $ALG - t_i$ does not change as more tasks are scheduled, but T_w's may get larger.

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}). Why was W_{ALG}'s last task i assigned to it? $ALG - t_i \leq T_w$, for all workers.

Sum up all work done by all of the workers:

$$\sum_w (ALG - t_i) \leq \sum_w T_w$$

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}). Why was W_{ALG}'s last task i assigned to it? $ALG - t_i \leq T_w$, for all workers.

Sum up all work done by all of the workers:

$$\sum_w (ALG - t_i) \leq \sum_w T_w$$

Does this relate to this?

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}). Why was W_{ALG}'s last task i assigned to it? $ALG - t_i \leq T_w$, for all workers.

Sum up all work done by all of the workers:

$$\sum_w (ALG - t_i) \leq \sum_w T_w$$

Can we get a $\frac{1}{w}$ over here?

Work Scheduling

A_w - Tasks assigned to worker w.
$T_w = \sum_{i \in A_w} t_i$ - Worker w's work time.

Given a set of n tasks and w identical workers, assign all tasks such that the longest work time is minimized.

Why can't $OPT < \frac{1}{w} \sum_i t_i$?

Total work time $\leq w\ OPT < w \frac{1}{w} \sum_i t_i = \sum_i t_i =$ Total work time needed.

Consider the worker that dictates ALG (W_{ALG}). Why was W_{ALG}'s last task i assigned to it? $ALG - t_i \leq T_w$, for all workers.

Sum up all work done by all of the workers:

$$\sum_w (ALG - t_i) \leq \sum_w T_w$$

What happens if we end up with something like this:

$$\boldsymbol{ALG \leq OPT + t_i}$$

Max Cut

Given an edge weighted undirected graph, find a cut such that the sum of edge weights that cross the cut is maximized.

Algorithm: Assigned each vertex to set A with probability ½. Let $B = V \backslash A$.

Max Cut

Given an edge weighted undirected graph, find a cut such that the sum of edge weights that cross the cut is maximized.

Algorithm: Assigned each vertex to set A with probability ½. Let $B = V \backslash A$.

Consider edge $e = (u, v)$. What is the probability that e crosses the cut?

Max Cut

Given an edge weighted undirected graph, find a cut such that the sum of edge weights that cross the cut is maximized.

Algorithm: Assigned each vertex to set A with probability ½. Let $B = V \backslash \mathrm{A}$.

Consider edge $e = (u, v)$. What is the probability that e crosses the cut?

1. $u \in A, v \in A$
2. $u \in A, v \in B$
3. $u \in B, v \in A$
4. $u \in B, v \in B$

$\frac{1}{2}$ **probability that e crosses the cut.**

Max Cut

Given an edge weighted undirected graph, find a cut such that the sum of edge weights that cross the cut is maximized.

Algorithm: Assigned each vertex to set A with probability ½. Let $B = V \backslash \mathrm{A}$.

Consider edge $e = (u, v)$. What is the probability that e crosses the cut?

1. $u \in A, v \in A$
2. $u \in A, v \in B$
3. $u \in B, v \in A$
4. $u \in B, v \in B$

$\frac{1}{2}$ **probability that e crosses the cut.**

Let random variable $X_e = \begin{cases} 1 \text{ if } e \text{ crosses cut} \\ 0 \text{ if not} \end{cases}$

Max Cut

Given an edge weighted undirected graph, find a cut such that the sum of edge weights that cross the cut is maximized.

Algorithm: Assigned each vertex to set A with probability ½. Let $B = V \backslash A$.

Consider edge $e = (u, v)$. What is the probability that e crosses the cut?

1. $u \in A, v \in A$
2. $u \in A, v \in B$
3. $u \in B, v \in A$
4. $u \in B, v \in B$

$\frac{1}{2}$ **probability that e crosses the cut.**

Let random variable $X_e = \begin{cases} 1 \text{ if } e \text{ crosses cut} \\ 0 \text{ if not} \end{cases}$

Value of Cut = $\sum_e ?$

Max Cut

Given an edge weighted undirected graph, find a cut such that the sum of edge weights that cross the cut is maximized.

Algorithm: Assigned each vertex to set A with probability ½. Let $B = V\backslash A$.

Consider edge $e = (u, v)$. What is the probability that e crosses the cut?

1. $u \in A, v \in A$
2. $u \in A, v \in B$
3. $u \in B, v \in A$
4. $u \in B, v \in B$

$\frac{1}{2}$ probability that e crosses the cut.

Let random variable $X_e = \begin{cases} 1 \text{ if } e \text{ crosses cut} \\ 0 \text{ if not} \end{cases}$

Value of Cut = $\sum_e ?$

$E[\text{Value of Cut}] = \ldots$

Use fact: $E[X_e] = \frac{1}{2}$ and $\sum_e w_e \geq OPT$