So far we have seen some applications of LP techniques but we did not need to solve an LP for those applications.

Next we will see how approximation algorithms can be designed using an LP-solver.

We will see some general purpose LP-solving methods towards the end of this course.

General Strategy:

1) Express your optimization problem as an integer program

2) Relax the integer constraints to make it a linear program

3) Find an optimal solution for the LP using an LP-solver

4) If the solution is non-integral then convert it into an integral solution by a rounding procedure.

5) Compare the obtained solution with the optimal solution. Ideally we should compare it with the integral optimal solution. But we have no clue about the integral optimal solution. So, instead compare it with LP optimal solution and get an approximation guarantee.
What is rounding?

\[ x^* \rightarrow x \]

\[ \text{LP opt} \rightarrow \text{Integral} \]

Example 1: \[ x_i = \begin{cases} 0 & \text{if } 0 \leq x_i^* < \frac{1}{2} \\ 1 & \text{if } \frac{1}{2} \leq x_i^* < 1 \end{cases} \]

2. Threshold \[ 0 \leq \lambda \leq 1 \]

\[ x_i = \begin{cases} 0 & \text{if } x_i^* < \lambda \\ 1 & \text{if } x_i^* \geq \lambda \end{cases} \]

3. Randomized rounding

\[ x_i = \begin{cases} 0 \text{ with prob } 1 - x_i^* / f(x_i^*) & \text{if } x_i^* < \lambda \\ 1 \text{ with prob } x_i^* / f(x_i^*) & \text{if } x_i^* \geq \lambda \end{cases} \]

Reference: Williamson Shmoys Section 5.1-5.5

Maximum Satisfiability Problem (NP-hard)

Given a set of clauses we want to satisfy maximum number of clauses

Def: Clause is an OR of literals

\[ x_1 \lor x_2 \]

Ex: \( (x_1 \lor \overline{x}_2), (x_2 \lor \overline{x}_3), (\overline{x}_3 \lor \overline{x}_1), (x_3 \lor x_4), (\overline{x}_4 \lor \overline{x}_3) \)

\[ x_1 = 1, x_2 = 1, x_3 = 1, x_4 = 0 \]

\[ \text{HW}_\text{max} \]
Maximum Weight Satisfiability
Each clause comes with a weight.
Maximize total weight of satisfied clauses

Greedy Algorithm?

\[
x_i \rightarrow \text{no. of clauses with } x_i
\]

\[
\overline{x}_i \rightarrow \text{no. of clauses with } \overline{x}_i
\]

\[
\frac{1}{2} - \text{approx}
\]

A Randomized algorithm:
set each \( x_i \) to 1 with prob. \( \frac{1}{2} \)
and to 0 with prob \( \frac{1}{2} \)
Independently.

What is the expected weight of the satisfied clauses?

How does it compare with the optimal?

\( W \leftarrow \text{total weight of satisfied clauses} \)

\[
E[W] = \sum_{A \in \{0,1\}^n} \frac{1}{2^n} (\text{total weight of satisfied clauses by } A)
\]

\[
\Pr (x = A) = \frac{1}{2^n}
\]
Linearity of expectation:

Let \( w_1, w_2, \ldots, w_m \) be the weights of the given clauses.

Let us define a random variable \( W(C_j) \) for each clause \( C_j \).

\[
W(C_j) = \begin{cases} 
  w_j & \text{if clause } C_j \text{ is satisfied} \\
  0 & \text{otherwise}
\end{cases}
\]

Claim \( W = \sum_{j=1}^{m} W(C_j) \)

By linearity of expectation

\[
E[W] = \sum_{j=1}^{m} E[W(C_j)] \overset{HW}{=} \sum_{j=1}^{m} w_j \times \Pr[C_j \text{ is satisfied}] + 0 \times \Pr[C_j \text{ is not satisfied}]
\]

What is the probability that \( C_j \) is satisfied when each variable is assigned 0 or 1 with prob 1/2.

\( C_j = x_1 \lor \overline{x_2}, \Pr = 3/4 \quad C_j = x_1 \lor x_3 \lor \overline{x_2} \lor x_4, \Pr[C_j \text{ is satisfied}] = 7/8 \)
\[ \Pr \left[ \text{C}_j \text{ is satisfied} \right] = 1 - \frac{1}{2^{l_j}} \]

where \( l_j \) is the number of literals in clause \( \text{C}_j \).

\[ E[W] = \sum_{j=1}^{m} w_j \left( 1 - \frac{1}{2^{l_j}} \right) \]

\[ l_j \geq 1 \implies 1 - \frac{1}{2^{l_j}} \geq \frac{1}{2} \]

\[ E[W] \geq \sum_{j=1}^{m} w_j \times \frac{1}{2} = \frac{1}{2} \left( \sum_{j=1}^{m} w_j \right) \geq \frac{1}{2} \text{ OPT} \]

If all clauses are large then better approximation factor:

\( \forall j \quad l_j \geq 3 \implies \text{Approx factor} = \frac{7}{8} \)

Deterministic Implementation of above:

there exists at least one assignment \( A \) for which

\( W_A \geq E[W] \)

Find \( A \) via self reducibility and conditional expectation

\[ E[W \mid x_1 = 1] = \sum_{A \in \{0,1\}^{n-1}} \frac{1}{2^{n-1}} \cdot \begin{cases} \text{Weight of the} \\ \text{satisfied clauses} \\ \text{under } A \text{ and} \\ x_1 = 1 \end{cases} \]
Claim
\[ E[W] = \frac{E[W | x_1 = 0] + E[W | x_1 = 1]}{2} \]

More generally,
\[ E[W | x_2 = 0, x_3 = 1] = \frac{E[W | x_2 = 0, x_3 = 1, x_1 = 0] + E[W | x_2 = 0, x_3 = 1, x_1 = 1]}{2} \]

Fact: computing \( E[W] \) is easy by linearity of expectation.
Similarly, computing \( E[W | x_2 = 0, x_3 = 1] \) etc.
HW is easy.

Deterministic strategy
One of \( E[W | x_1 = 0] \) and \( E[W | x_1 = 1] \) must be larger than \( E[W] \).
Set \( x_1 \) based on that.
If \( E[W | x_1 = 0] > E[W | x_1 = 1] \)
then set \( x_1 = 0 \). Otherwise \( x_1 = 1 \).
Say we have set \( x_1 = 0 \).
If \( E[W | x_1 = 0, x_2 = 1] > E[W | x_1 = 0, x_2 = 0] \)
then set \( x_2 = 1 \).
Continue for each variable.
Prove that the above deterministic algorithm gives objective value as good as the expected objective value of the randomized algorithm.

**Example**

\[(\overline{x}_1 \lor x_2), (\overline{x}_2 \lor x_3), (\overline{x}_3 \lor \overline{x}_1), (x_1)\]

Weights 16  20   12  18

\[E[W] = \frac{3}{4} \times 16 + \frac{3}{4} \times 20 + \frac{3}{4} \times 12 + \frac{1}{2} \times 18\]

\[= 12 + 15 + 9 + 9 = 45\]

When each of \(x_1, x_2, x_3\) is set with prob \(\frac{1}{2}\)

The above numbers in red show the probability of a clause being satisfied under a random assignment.

Now we want to find \(E[W \mid x_1 = 0]\) & \(E[W \mid x_1 = 1]\)

We will fix \(x_1\) and let \(x_2\) and \(x_3\) be randomly chosen.

\[E[W \mid x_1 = 0] = 1 \times 16 + \frac{3}{4} \times 20 + 1 \times 12 + 0 \times 18\]

\[= 43\]

\[E[W \mid x_1 = 1] = \frac{1}{2} \times 16 + \frac{3}{4} \times 20 + \frac{1}{2} \times 12 + 1 \times 18\]

\[= 47\]
Since $E[W | x_1 = 1] > E[W | x_1 = 0]$

let us set $x_1 = 1$

Now we want to decide about $x_2$.

Let $x_1$ be fixed to 1.

Compute $E[W | x_1 = 1, x_2 = 0]$ and $E[W | x_1 = 1, x_2 = 1]$

Here $x_2$ will be fixed.

and only $x_3$ will be randomly chosen.

$E[W | x_1 = 1, x_2 = 0] = 0 \times 16 + 1 \times 20 + \frac{1}{2} \times 12 + 1 \times 18$

$= 44$

$E[W | x_1 = 1, x_2 = 1] = 1 \times 16 + \frac{1}{2} \times 20 + \frac{1}{2} \times 12 + 1 \times 18$

$= 50$

Since $E[W | x_1 = 1, x_2 = 1] > E[W | x_1 = 1, x_2 = 0]$

we set $x_2 = 1$

Now we want to decide about $x_3$

We will compute

$E[W | x_1 = 1, x_2 = 1, x_3 = 0]$ and $E[W | x_1 = 1, x_2 = 1, x_3 = 1]$

Everything is fixed in these computations.

No variable is random.
\[ E[W | x_1 = 1, x_2 = 1, x_3 = 0] \]
\[ = 1 \times 16 + 0 \times 20 + 1 \times 12 + 1 \times 18 \]
\[ = 46 \]

\[ E[W | x_1 = 1, x_2 = 1, x_3 = 1] \]
\[ = 1 \times 16 + 1 \times 20 + 0 \times 12 + 1 \times 18 \]
\[ = 54 \]

Since \( E[W | x_1 = 1, x_2 = 1, x_3 = 1] > E[W | x_1 = 1, x_2 = 1, x_3 = 0] \)
we set \( x_3 = 1 \)

Final assignment \( x_1 = 1 \), \( x_2 = 1 \), \( x_3 = 1 \)

weight of satisfied clauses = 54

Clearly greater than the expectation \( E[W] = 45 \)

Coincidentally, 54 is the optimal value.

There was a question in the class:

Can we set \( x_1, x_2, x_3 \) simultaneously?
The idea was to compare 
\[ E[W \mid x_1=0] \text{ with } E[W \mid x_1=1], \Rightarrow \text{ decide } x_1 \]
\[ E[W \mid x_2=0] \text{ with } E[W \mid x_2=1], \Rightarrow \text{ decide } x_2 \]
and \[ E[W \mid x_3=0] \text{ with } E[W \mid x_3=1], \Rightarrow \text{ decide } x_3 \]

Let's try this with above example.

\[ E[W \mid x_1=0] = 43 \]
\[ E[W \mid x_1=1] = 47 \]

\[ E[W \mid x_2=0] = \frac{1}{2} \times 16 + 1 \times 20 + \frac{3}{4} \times 12 + \frac{1}{2} \times 18 \]
\[ = 46 \]
\[ E[W \mid x_2=1] = 1 \times 16 + \frac{1}{2} \times 20 + \frac{3}{4} \times 12 + \frac{1}{2} \times 18 \]
\[ = 44 \]

\[ \Rightarrow x_2 = 0 \]

\[ E[W \mid x_3=0] = \frac{3}{4} \times 16 + \frac{1}{2} \times 20 + 1 \times 12 + \frac{1}{2} \times 18 \]
\[ = 43 \]
\[ E[W \mid x_3=1] = \frac{3}{4} \times 16 + 1 \times 20 + \frac{1}{2} \times 12 + \frac{1}{2} \times 18 \]
\[ = 47 \]

\[ \Rightarrow x_3 = 1 \]

We get \( x_1=1, x_2=0, x_3=1 \)

Total weight of satisfied clauses = \( 20 + 18 = 38 \)

Less than \( E[W] = 45 \). **Bad strategy.**
Can we improve $1/2$ factor?

Yes, by LPs. $1 - 1/e \approx 0.63$

**LP**

Variable $Z_j$ for clause $C_j$

$$\max \sum_{j=1}^{m} w_j z_j$$

$s \geq Z_j \leq 1$

$0 \leq y_i \leq 1$

$Z_j \leq \sum_{i \in p_j} y_i + \sum_{i \in N_j} (1-y_i)$

Define

$p_j \leftarrow \{ i : \alpha_i \text{ appears in clause } C_j \}$

$N_j \leftarrow \{ i : \overline{\alpha_i} \text{ appears in clause } C_j \}$

Rounding

Solve this LP.

Let $(y^*, Z^*)$ be the optimal solution for the LP.

Set $\alpha_i$ to TRUE with prob. $y_i^*$

and FALSE with prob. $1 - y_i^*$

**Expected weight of satisfied clauses $E[W]$?**

$$E[W] = \sum_{j=1}^{m} w_j \times (\text{Prob of } C_j \text{ being satisfied})$$
\[ E[W] = \sum_{j=1}^{m} w_j \left( 1 - \prod_{i \in P_j} (1 - y_i^*) \right) \prod_{i \in N_j} y_i^* \]

\[ \text{AM} \geq \text{GM} \geq \beta \cdot Z_j^* \quad \text{(to show)} \]

\[ \left( \frac{\alpha_1 + \alpha_2 + \cdots + \alpha_k}{k} \right)^k \geq (\alpha_1 \cdot \alpha_2 \cdots \alpha_k) \]

\[ E[W] \geq \sum_{j=1}^{m} w_j \left( 1 - \left( \frac{\sum_{i \in P_j} (1 - y_i^*) + \sum_{i \in N_j} y_i^*}{l_j^*} \right)^{l_j^*} \right) \]

where \( l_j = |P_j| + |N_j| \)

Recall \( Z_j^* \leq \sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*) \)

\[ \Rightarrow l_j - Z_j^* \geq l_j - \left( \sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*) \right) = \sum_{i \in P_j} (1 - y_i^*) + \sum_{i \in N_j} y_i^* \]

\[ E[W] \geq \sum_{j=1}^{m} w_j \left( 1 - \left( \frac{l_j - Z_j^*}{l_j} \right)^{l_j} \right) \]

\[ = \sum_{j=1}^{m} w_j \left( 1 - \left( 1 - \frac{Z_j^*}{l_j} \right)^{l_j} \right) \geq \beta \sum_{j=1}^{m} w_j Z_j^* \]
would like to show

\[
\left(1 - \left(1 - \frac{Z_j^*}{k_j}ight)^l\right) \geq \beta \cdot Z_j^*
\]

for some \(\beta > 0\)

Define \(g(z) = 1 - \left(1 - \frac{z}{l}\right)^l\)

\(l=1 \Rightarrow g(z) = z\)

\(l=2 \Rightarrow z - \frac{z^2}{4}\)

\(l=3 \Rightarrow g(z) = \frac{19}{27}\)

Claim: \(g(z)\) is a concave function in \([0,1]\)

you can verify that \(g''(z) \leq 0\). \(\text{HW}\)

Observation \(g^*(z) \geq g(1)z\)

\[
1 - \left(1 - \frac{z}{l}\right)^l \geq z \left(1 - \left(1 - \frac{1}{l}\right)^l\right)
\]

for \(z \in [0,1]\)

\(l=1 \Rightarrow \beta_l = 1\)

\(l=2 \Rightarrow \beta_l = 3/4\)

\(l=3 \Rightarrow \beta_l = 19/27\)

\(l \to \infty \Rightarrow \beta_l \to 1 - \frac{1}{e}\)
\[ E[W] \geq (1 - \frac{1}{e}) \cdot \sum_{j=1}^{\infty} w_j Z_j^* = (1 - \frac{1}{e}) \cdot \text{LP-OPT} \]

\[ E[W] \geq (1 - \frac{1}{e}) \cdot \text{OPT} \]

Deterministic Implementation?

by computing conditional expectations.

Will get something as good as \( E[W] \).

Can we further improve from \((1 - \frac{1}{e})\)?

First randomized scheme \( \text{PR} = V_2 \)

\[ E[w] = \sum_j \left( 1 - \frac{1}{2^j} \right) w_j \quad \text{worst case} \]

LP based Scheme \( \text{PR} = y_i^* \)

\[ E[w] \geq \sum_j \left( 1 - (1 - \frac{1}{l_j})^j \right) w_j Z_j^* \quad \text{Best case} \]

\[ \text{Bad for large change} \]
Combination of the two schemes.
\[ x_i \triangleq \text{TRUE with prob } \frac{1}{2}, \frac{1}{2} + \frac{1}{2} y_i^* = \frac{1}{4} + \frac{y_i^*}{2} \]

Show that
\[ E[W] \geq \sum_j w_j \left(1 - \left[\frac{1}{4} + \frac{1}{2}(1 - \frac{y_j^*}{l_j})\right]^{l_j}\right) \]

\[ \Rightarrow E[W] \geq \frac{3}{4} \sum_j w_j y_j^* \]

Deterministic

Is there a rounding scheme with better than \(3/4\) approx guarantee?

Integrality Gap of an LP formulation

\[ \min \begin{array}{c} \text{OPT(I)} \\ \text{I} \end{array} \leq \frac{\text{OPT(I)}}{\text{LP-OPT(I)}} \]

Rounded \( \text{OPT} \) \( \Rightarrow \) \( \text{LP-OPT} \)
Example where integrality gap is $3/4$

$$(x_1 \lor x_2), (x_1 \lor \overline{x}_2), (\overline{x}_1 \lor x_2), (\overline{x}_1 \lor \overline{x}_2)$$

\[
\begin{array}{cccc}
1 & 1 & 1 & 1 \\
\end{array}
\]

OPT = 3 \hspace{1cm} LP - OPT = 4.