1 Randomized Rounding

For more on LP rounding techniques, let’s look at a randomized algorithm.

- Recall the MAX-SAT problem.
- Given a Boolean formula $\phi$ in CNF, find a truth assignment that satisfies as many clauses of $\phi$ as possible.
- Previously, we saw a randomized (but derandomizable) 8/7 approximation when $\phi$ was 3-CNF.
- Will now derive a (slightly weaker) randomized result for arbitrary $\phi$!

We’ll formulate MAX-SAT on formula $\phi$ as a 0-1 integer program $P(\phi)$.

- For each variable $v_i$ of $\phi$, define indicator $y_i$ to be 1 if $v_i$ is true, 0 otherwise.
- For each clause $C_j \in \phi$, define indicator $z_j$ to be 1 if clause $C_j$ is satisfied, 0 otherwise.
- For a literal $\ell$, define
  \[ f(\ell) = \begin{cases} y_i & \text{if } \ell = v_i \\ 1 - y_i & \text{if } \ell = \neg v_i \end{cases} \]
- For clause $C_j \ell_1^j \lor \ldots \lor \ell_k^j$, add constraint
  \[ \sum_{p=1}^{k} f(\ell_p^j) \geq z_j. \]
  In other words, $z_j$ can be 1 iff some literal of $C_j$ is true.
- **Example**: $C_4 = v_3 \lor \neg v_5 \lor v_9$ gives
  \[ y_3 + 1 - y_5 + y_9 \geq z_4. \]
- Objective is to maximize
  \[ \sum_j z_j, \]
  that is, to satisfy as many clauses as possible.
Now let’s describe our rounding strategy.

- Algorithm ROUNDING-MS($\phi$) works as follows.
- Construct integer program $P(\phi)$.
- Solve LP relaxation of $P$, in which $0 \leq y_i \leq 1$ and $0 \leq z_j \leq 1$.
- Let $\overline{X}$ be LP optimum; construct feasible ILP solution $\hat{X}$ by rounding $\overline{y}_i$’s as follows.
  - For $1 \leq i \leq n$, set $\hat{y}_i = 1$ with probability $\overline{y}_i$, 0 otherwise.
  - Now, set $\hat{z}_j = 1$ if $\hat{y}$’s satisfy clause $C_j$, 0 otherwise.
- Return truth assignment corresponding to $\hat{X}$.

**Claim**: on average, truth assignment found by ROUNDING-MS satisfies at least $1 - 1/e$ times as many clauses of $\phi$ as an optimal assignment.

- $(1 - 1/e$ is about 0.632, so this is about a 1.582-approximation.)
- **Pf**: will start by analyzing algorithm’s behavior for a single clause.
- Suppose WLOG that $C_j = v_1 \lor \ldots \lor v_k$.
- (Argument extends easily to negated literals.)
- Because LP solution satisfies all constraints, we know that
  $$\sum_{i=1}^{k} \overline{y}_i \geq \overline{z}_j.$$  
  and that $0 \leq \overline{z}_j \leq 1$.
- Rounding procedure satisfies $C_j$ with probability at least
  $$\Pr(\hat{z}_j = 1) = 1 - \prod_{i=1}^{k} (1 - \overline{y}_i).$$
- Using a bit of calculus, this probability is minimized subject to the LP constraint when $\overline{y}_i = \overline{z}_j/k$.
- Conclude that
  $$\Pr(\hat{z}_j = 1) \geq 1 - \left(1 - \frac{\overline{z}_j}{k}\right)^k.$$  
- (Try it for $k = 2$ or $k = 3$ – set all derivatives w/r to $y$’s to zero, and solve for their values.)
- Moreover, can show by elementary analysis that, for any $0 \leq \overline{z}_j \leq 1$,
  $$1 - \left(1 - \frac{\overline{z}_j}{k}\right)^k \geq \overline{z}_j \left[1 - \left(1 - \frac{1}{k}\right)^k\right] \geq \overline{z}_j \left(1 - \frac{1}{e}\right).$$
• (For first step, $1 - (1 - z/k)^k$ is a convex function of $z$, and it equals $[1 - (1 - 1/k)^k] \cdot z$ for $z = 0$ and $z = 1$. Hence, it is $\geq$ the linear function for every $z$ between 0 and 1.)

• (For second step, $(1 - 1/k)^k$ increases with increasing $k$, so it achieves its maximum of $1/e$ as $k \to \infty$. Hence, the linear function is at least $(1 - 1/e) \cdot z$ for any $k$.)

• Conclude that
  \[
  \Pr(\hat{z}_j = 1) \geq \bar{z}_j \left( 1 - \frac{1}{e} \right).
  \]

• Now let $S(X) = \sum_j z_j$ be value of LP solution $X$.

• Let $X^*$ be ILP optimum; let $\bar{X}$ be LP optimum, and let $\hat{X}$ be solution constructed by algorithm.

• As usual, we have $S(\bar{X}) \geq S(X^*)$.

• Conclude that
  \[
  E[S(\hat{X})] = \sum_j E[\hat{z}_j]
  \geq \sum_j \Pr(\hat{z}_j = 1)
  \geq \sum_j \bar{z}_j \left( 1 - \frac{1}{e} \right)
  = \left( 1 - \frac{1}{e} \right) S(\bar{X})
  \geq \left( 1 - \frac{1}{e} \right) S(X^*)
  \]
  which proves the desired approximation ratio. QED

• This algorithm can be derandomized by the same conditional-expectations approach that we considered before.

2 **Additional Fun Facts about MAX-SAT**

• The above algorithm is due to Goemans and Williamson (1994).

• If we upper-bound the number of variables in each clause, the $1 - 1/e$ bound can be improved. It gives a better result for formulas with lots of relatively small clauses.

• In contrast, Johnson’s algorithm (i.e. set each variable true with probability $1/2$) gives a better result for formulas with lots of large clauses.

• The G&W paper shows that if you take the better of the LP algorithm’s solution and that given by Johnson’s algorithm, the result is a $4/3$-approximation.

• (There are many other ways to get a $4/3$-approximation.)
• Best known approximation ratio for this problem is 1.2522 (Avidor, Berkovitch, and Zwick, 2006). Same work claims an algorithm with ratio 1.1857, subject to the truth of an unproven but numerically plausible conjecture of Zwick (1999).

• MAX-SAT cannot be approximated within better than 8/7 unless P=NP (Hastad, 2001).

• (Above results also hold for clause-weighted version of problem.)