Submodular minimization in combinatorial problems

Zoya Svitkina

joint work with *Lisa Fleischer*

Classical problems in new light

What happens to the classical, well-understood computer science problems:

- Knapsack, bin packing, scheduling, graph cuts

when a submodular function is involved?

Classical problems in new light

What happens to the classical, well-understood computer science problems:

- Knapsack, bin packing, scheduling, graph cuts

when a submodular function is involved?

They become very hard to approximate

Submodular minimization with cardinality lower bound (SML)

- Given: ground set V, function f, integer W
- $f(S)$ submodular, not necessarily monotone
- Find $S \subseteq V$ with $|S| \geq W$ minimizing $f(S)$

Submodular minimization with cardinality lower bound (SML)

- Given: ground set V, function f, integer W
- $f(S)$ submodular, not necessarily monotone
- Find $S \subseteq V$ with $|S| \geq W$ minimizing $f(S)$

Results:

- Algorithm: $f(S) \leq O\left(\sqrt{\frac{n}{\log n}}\right) OPT,\ |S| \geq \frac{1}{2} \cdot W$
- Lower bound: $\alpha/\beta = \Omega\left(\sqrt{\frac{n}{\log n}}\right)$

Lower bound technique

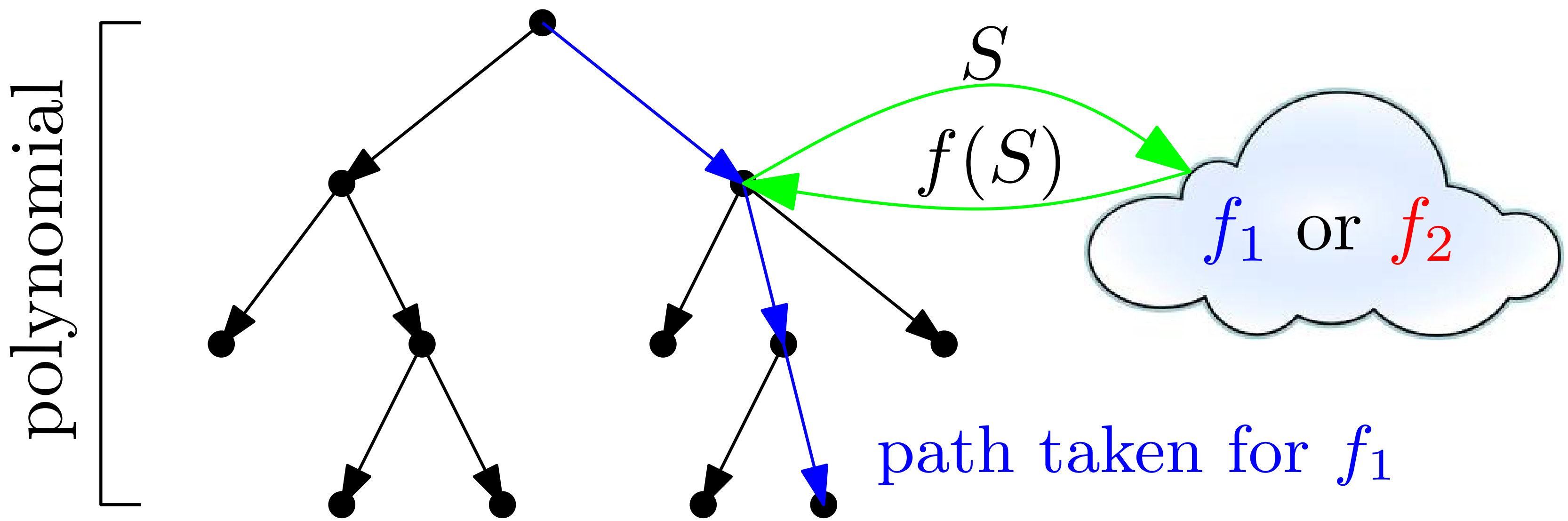
- Take advantage of the oracle model to fool the algorithm

- Define a function f_1 and a distribution of functions f_2

- For any set S, $\Pr[f_1(S) \neq f_2(S)] = n^{-\omega(1)}$

Lower bound technique

Computation tree for deterministic algorithm $\mathcal{A}$:



$\mathcal{A}$ cannot distinguish f_1 and f_2 with high probability

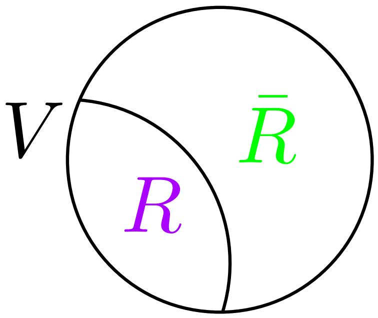
- use $\Pr[f_1(S) \neq f_2(S)] = n^{-\omega(1)}$
- union bound over blue path

Lower bound technique

- Find f_1 and f_2 s.t. $OPT(f_1) \geq \gamma \cdot OPT(f_2)$ for a given problem
- Algorithm $\mathcal{A}$ cannot distinguish f_1 and f_2, so outputs solution S with $Cost(S) \geq OPT(f_1)$
- But then $Cost(S) \geq \gamma \cdot OPT(f_2)$
- So approximation ratio of $\mathcal{A}$ is at least γ
- (Also applies to randomized algorithms)

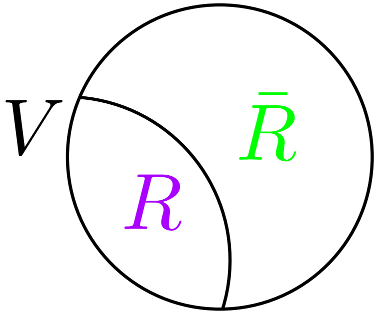
Lower bound for SML

- $f_1(S) = \min(|S|, \alpha)$
- $f_2(S) = \min(\beta + |S \cap \bar{R}|, |S|, \alpha)$
- Random R with $|R| = \alpha$,
  $\alpha = \frac{x\sqrt{n}}{5}$, $\beta = \frac{x^2}{5}$, $x^2 = \omega(\ln n)$



Lower bound for SML

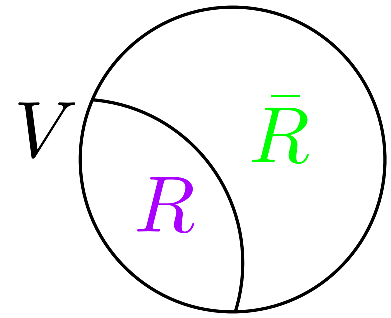
- $f_1(S) = \min(|S|, \alpha)$
- $f_2(S) = \min(\beta + |S \cap \bar{R}|, |S|, \alpha)$
- Random R with $|R| = \alpha$,
  $\alpha = \frac{x\sqrt{n}}{5}$, $\beta = \frac{x^2}{5}$, $x^2 = \omega(\ln n)$



- $\Pr[f_1(S) > f_2(S)]$ maximized for $|S| = \alpha$
- W.h.p., for any S with $|S| = \alpha$, $|S \cap R| < \beta$, and $f_1(S) = f_2(S)$

Lower bound for SML

- $f_1(S) = \min(|S|, \alpha)$
- $f_2(S) = \min(\beta + |S \cap \bar{R}|, |S|, \alpha)$
- Random R with $|R| = \alpha$,
  $\alpha = \frac{x\sqrt{n}}{5}$, $\beta = \frac{x^2}{5}$, $x^2 = \omega(\ln n)$
- Hardness of SML with $W = \alpha$ is
  $\frac{OPT(f_1)}{OPT(f_2)} = \frac{\alpha}{\beta} = \Theta(\sqrt{\frac{n}{\ln n}})$
- Also applies to bicriteria guarantees



Algorithm for SML

Bicriteria decision procedure:

- Given: function f, bound W, guess B, probability p
- If there is S with $|S| \geq W$ and $f(S) < B$, outputs, with probability at least p, a set U with $|U| \geq \frac{W}{2}$ and $f(U) \leq 5\sqrt{\frac{n}{\ln n}} \cdot B$

Algorithm building blocks

Find a set S of density $\frac{f(S)}{|S|} < \lambda$:

- Use submodular function minimization to minimize $f(S) - \lambda \cdot |S|$
- If the result is negative, the low-density set is found
- Else such set does not exist

The easy case: $W \geq n/2$

- Let $U_0 = \emptyset$ be the current solution.
- While $|U_i| < W/2$:
  - Minimize $f(T_i) - \frac{2B}{W} \cdot |T_i \setminus U_i|$
  - If negative, let $U_{i+1} = U_i \cup T_i$, else fail

The easy case: $W \geq n/2$

- Let $U_0 = \emptyset$ be the current solution.
- While $|U_i| < W/2$:
  - Minimize $f(T_i) - \frac{2B}{W} \cdot |T_i \setminus U_i|$
  - If negative, let $U_{i+1} = U_i \cup T_i$, else fail

If feasible, there is U^* such that:

- $f(U^*) < B$, $|U^*| \geq W$, $|U^* \setminus U| > W/2$
- minimized expression is negative

The easy case: $W \geq n/2$

- Let $U_0 = \emptyset$ be the current solution.
- While $|U_i| < W/2$:
  - Minimize $f(T_i) - \frac{2B}{W} \cdot |T_i \setminus U_i|$
  - If negative, let $U_{i+1} = U_i \cup T_i$, else fail

Algorithm terminates with a set U of low density:

- $|U| \geq W/2$
- $f(U) \ \leq \ \sum_i f(T_i) \ < \ \frac{2B}{W} \sum_i |T_i \setminus U_i|$
  $\leq \frac{2B}{W} \cdot n \ \leq \ 4B$

The hard case: $W < n/2$

- Just a low-density set can be too expensive
- "Guess" a set S with high overlap with OPT (pick each element with prob. W/n)
- Minimize $f(T) - \alpha \cdot |T \cap S|$
- $\alpha = \frac{2B}{W}\sqrt{\frac{n}{\ln n}}$

Algorithm for $W < n/2$

- While $|U_i| < W/2$:
  - random $S_i \subseteq V \setminus U_i$:
    include each element w/prob $\frac{W}{w(V)}$
  - minimize $f(T_i) - \alpha \cdot w(T_i \cap S_i)$
  - **if** $f(T_i) \leq \alpha \cdot w(T_i \cap S_i)$ and
    $f(T_i) \leq 4B\sqrt{\frac{n}{\ln n}}$:
    $U_{i+1} = U_i \cup T_i$
  - **if** too many iterations, fail

$Algorithm\ for\ W < n/2$

Lucky case:

- $|U^* \cap S| > \frac{B}{\alpha} = \frac{W}{2}\sqrt{\frac{\ln n}{n}}$
- $|\bar{U}^* \cap S| \leq 1.5W$
- Both happen with probability $\approx n^{7/2}$

Algorithm for $W < n/2$

Then:

- Negative minimization result:
- $f(T_i) - \alpha \cdot |T_i \cap S_i| \leq f(U^*) - \alpha \cdot |U^* \cap S_i| < f(U^*) - B < 0$
- $f(T_i)$ is not too large:
- $f(T_i) \leq f(U^*) + \alpha \cdot (|T_i \cap S_i| - |U^* \cap S_i|) \leq B + \alpha \cdot |\bar{U}^* \cap S_i| \leq B + 1.5\alpha W \leq 4B\sqrt{\frac{n}{\ln n}}$
- New set added to U by the algorithm

Bounding solution cost

- Separate the cost of the last set and other sets:
- $f(U) = \sum_{j=0}^{i-1} f(U_j) + f(U_i) \leq \alpha \cdot \frac{W}{2} + 4B\sqrt{\frac{n}{\ln n}} = 5B\sqrt{\frac{n}{\ln n}}$

Other problems

Submodular sparsest cut

- find set S minimizing $\frac{f(S)}{\min(|S|,|\bar{S}|)}$

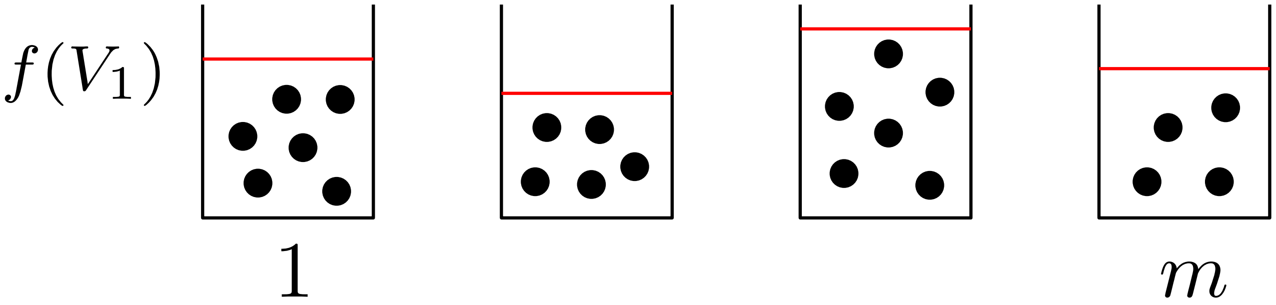
Other problems

Submodular sparsest cut

- find set S minimizing $\frac{f(S)}{\min(|S|,|\bar{S}|)}$

Submodular load balancing (monotone f)

- find partition $\{V_1, \ldots, V_m\}$ minimizing $\max_i f(V_i)$



Other problems

Submodular sparsest cut

- find set S minimizing $\frac{f(S)}{\min(|S|,|\bar{S}|)}$

Submodular load balancing (monotone f)

- find partition $\{V_1, ..., V_m\}$ minimizing $\max_i f(V_i)$

Results:

- Algorithms: $O\left(\sqrt{\frac{n}{\log n}}\right)$
- Lower bounds: $\Omega\left(\sqrt{\frac{n}{\log n}}\right)$

Summary

- New problems involving submodular functions
  - Sparsest cut, load balancing, submodular minimization with cardinality lower bound
- Tight approximability bounds
  - Lower bounds for oracle query complexity
  - Approximation algorithms based on random sampling and submodular function minimization