Problem Set on Multi-Stage Optimization
Approximation Algorithms for Stochastic Optimization
Anupam Gupta, Carnegie Mellon University

Some Notation

- For a universe U of elements with weights w_e, define $w(S) := \sum_{e \in S} w_e$ for every $S \subseteq U$.
- Some problems marked ($\star$) are more challenging. Problems marked (?) are potentially open-ended.

LP-Based Approximations

1. **(Stochastic Set Cover.)** In the deterministic set cover problem, we are given a universe U of "elements", and a subset $R \subseteq U$ of elements that need to be covered.[1] We are given a collection of sets $\mathscr{F} = \{S_1, S_2, \ldots, S_m\}$, with S_i having cost $c(S_i)$. The goal is to pick a sub-collection $\mathscr{G} \subseteq \mathscr{F}$ of sets such that the union of sets in $\mathscr{G}$ cover all of R (i.e., $R \subseteq \cup_{S \in \mathscr{G}} S$) and so that the total cost $\sum_{S \in \mathscr{G}} c(S)$ is minimized.

   The following integer linear program (ILP) is a formulation of this problem:

$$\min \sum_{S \in \mathscr{F}} c(S) x_S$$
$$\sum_{S \in \mathscr{F} : e \in S} x_S \geq 1 \qquad \forall e \in R$$
$$x_S \in \{0,1\} \qquad \forall S \in \mathscr{F}$$

   The natural LP relaxation replaces the integrality constraints by the weaker constraint $x_S \geq 0$. The following are standard ideas from the theory of approximation algorithms.

   (a) Let $\{x_S^*\}$ be any solution to the LP relaxation. Let $OPT_{LP} := \sum_{S \in \mathscr{F}} c(S) x^*(S)$. Let OPT be the cost of the optimal solution to the original problem. Show that $OPT_{LP} \leq OPT$.

   (b) Consider the following "randomized rounding" algorithm.

   Repeat the following process independently $O(\log |R|)$ times: for each set $S \in \mathscr{F}$, add it to $\mathscr{G}$ with probability $\min\{x_S, 1\}$.

   Show that the collection $\mathscr{G}$ obtained thus is a feasible solution with probability at least $1 - 1/\operatorname{poly}(|R|)$. What is the expected cost of this solution (in terms of OPT_{LP}).

   (c) Suggest a way to output a set $\mathscr{G}'$ that is guaranteed to be a feasible solution for the set cover instance (with probability 1), and whose expected cost is only $O(\log |R|) \cdot OPT_{LP}$.

   Now consider the two-stage stochastic set cover (SSC) problem in the explicit scenario model. You are given a cost function $c_0 : \mathscr{F} \to \mathcal{R}_{\geq 0}$ which is the cost of picking sets in the first stage. Moreover, there are K scenarios, with scenario i specified by

   - a probability p_i,

[1] Usually $R = U$, i.e., we have to cover all the elements of U, but this will be convenient for us to consider.

- a different cost function $c_i : \mathscr{F} \to \mathcal{R}_{\geq 0}$, and
- a different set R_i of required elements to pick.

(a) Write down an ILP formulation and an LP relaxation for this problem.

(b) How would you round an optimal solution to this LP relaxation.

(c) Show an $O(\sum_i \log |R_i|)$ approximation algorithm for this problem.

2. **(Vertex Cover, Improved.)** Consider the following randomized rounding algorithm for Stochastic Vertex Cover (SVC), which is a 2-approximation.

   Take the LP we wrote in Lecture #1. Now, pick a random real value α uniformly from the interval $[0, 1/2]$. For every vertex $v \in V$, do the following:

   - If $x(v) \geq \alpha$, then add v to V_0.
   - For each scenario k, if $x(v) < \alpha$ but $x(v) + y_k(v) \geq \alpha$, then add v to V_k.

   Show that for each k, $V_0 \cup V_k$ is a feasible vertex cover for the edges in E_k (with probability 1). Show that the expected cost of the solution is 2 times the cost of the LP solution.

3. **(Side Constraints.)** Take the SVC problem we solved in Lecture #1. Suppose you have a constraint that the total expenditure in the first stage ("Monday") is at most the "first-stage budget" B_M, and the expenditure for every scenario in the second stage ("Tuesday")is at most B_T. Show how to find a constant-factor approximation for the problem with these added constraints.

4. **(Multi-Stage Optimization.)** We can do multiple stages of decisions, the probability distribution gets refined over time. One formalization is the following — on the second day ("Tuesday") we are given some signal I which is a random variable taking values in some signal space $\{1, \ldots, M\}$ according to distribution τ. On the third day ("Wednesday") the actual scenario is revealed, drawn from the probability distribution π^I. The costs are now $c_0()$ for the first stage, $c_i()$ for the second stage when $I = i$, and $c_{i,k}$ for the third stage. We have to ensure the decisions made in the first, second, and third stage lead to a feasible solution at the end of the third stage; again we want to minimize the expected cost. As the optimizer, we can assume knowledge (either explicitly, or in some black-box way) of the distributions τ and $\{\pi^i\}_{i=1}^M$.

   For instance, in the explicit scenario model for StocVC, we are explicitly given these distributions, along with the scenario sets $E_{i,k}$, and the costs, and we want to minimize

   $$c_0(V_0) + \mathbb{E}_{I \leftarrow \tau}\Big[c_I(V_I) + \mathbb{E}_{J \leftarrow \pi^I}\left[c_{I,J}(V_{I,J})\right]\Big]$$

   subject to $V_0 \cup V_I \cup V_{I,J}$ being a feasible vertex cover for the edges in the scenario $S_{I,J}$.

   (a) Give a 6-approximation for this problem by solving an LP. (Hint: extend the idea from the 4-approximation in lecture.)

   (b) Give a 2-approximation for this problem using the ideas from Prob. 2.

Combinatorial Approximations

5. **(Stochastic Steiner Tree on a Tree.)** The stochastic Steiner tree (SST) is as defined in Lecture #1: we are given

- a graph $G = (V, E)$ with a root vertex $r \in V$,
- non-negative edge lengths functions $c_0(e)$ and $\{c_S(e)\}_{S \subseteq V}$, and
- a probability distribution π over subsets of V.

The goal is to specify some subset of edges E_0, and subsets $\{E_S\}_{S \subseteq E}$ to minimize

$$c_0(E_0) + \mathbb{E}_{S \leftarrow \pi}[c_S(E_S)]$$

subject to $E_0 \cup E_S$ forming a Steiner tree connecting S to the root r, for each $S \subseteq V$.

How is π specified? For this problem, suppose we are in the SQ (statistical query) *model, where we are given an oracle that given a random variable X can compute $\mathbb{E}[X]$ in unit time.*

In this problem we first explore what happens when G is a tree.

(a) For fixed E_0, show how to find optimal choice of E_S in linear time. Infer that the only decisions to be made are whether to add edges to E_0 or not.

(b) Deleting any edge e in G splits the tree into two connected components with vertices $(L_e, R_e = V \setminus L_e)$, where L_e is the part not containing r. Let

$$a_e = \mathbb{E}_{S \leftarrow \pi}[c_S(e) \cdot \mathbf{1}_{(S \cap L_e \neq \emptyset)}].$$

What is the cost of adding e to E_0? What is the expected cost of not adding e to E_0?

(c) Give an exact algorithm for solving SST when G is a tree, in the SQ model.

(d) Instead of an exact statistical query oracle, suppose we were given an *ε-approximate oracle* that given r.v. X outputs some value in $(1 \pm \varepsilon)\,\mathbb{E}[X]$. Give an algorithm that outputs a $(1 + O(\varepsilon))$-approximate SST solution in this case.

6. **(SST on General Graphs.)** The following theorem from metric embeddings is a powerful tool to use. For any graph G with (non-negative) edge-lengths ℓ_e, let $d_G(x, y)$ be the shortest-path distance between x and y according to these edge lengths.

Theorem 1 (Abraham Neiman) *Given a graph $G = (V, E)$ with edge lengths ℓ_e, there exists a sampler $\mathscr{A}$ that outputs a random spanning tree T of G such that for all $x, y \in V$:*

(a) $d_G(x, y) \leq d_T(x, y)$

(b) $\mathbb{E}_{T \leftarrow \mathscr{A}}[d_T(x, y)] \leq \alpha\, d_G(x, y)$ *for* $\alpha = O(\log |V| \log\log |V|)$.

(Distances in T are according to the same edge-lengths ℓ_e.)

(The former inequality is immediate from the fact that T is a subgraph of G; the hard work is to prove the second inequality.)

(a) Using this theorem, consider the following algorithm for Stochastic Steiner Tree on a general graph G. (Assume uniform inflations: $c_{T,k}(e) = \lambda\, c_M(e)$ for all e and all scenarios k.)

- Draw a random tree T from the sampler $\mathscr{A}$ on G.
- Run an optimal SST algorithm on this tree.

Show this is an α-approximation algorithm for the SST problem on G. Hint: (a) show that if there is an SST solution of expected cost C in the graph G, there exists an SST solution of expected cost $\alpha\, C$ on T, and (b) observe that any SST solution on T is also a solution on G with the same cost.

(b) Suppose we had a different sampler that, instead of the property (a) of Theorem 1 above, only gave an "expected" guarantee $d_G(x, y) \leq \mathbb{E}[d_T(x, y)]$. Where does your argument break down? Also, where did you use the uniform inflations?

7. **Two-Stage Maximization Problems are Easier.** Consider the Two-Stage Maximum-Matching problem: given a graph $G = (V, E)$, first-stage weights w_0 and second-stage weights $\{w_k\}_{k=1}^K$, and a distribution π over the scenarios, the goal is to pick edges E_0, and then for each scenario k, edges E_k such that (a) for every k, the set $E_0 \cup E_k$ is a feasible matching, to (b) maximize the expected weight of the matching

$$w_0(E_0) + \mathbb{E}_{k \leftarrow \pi}[w_k(E_k)].$$

(a) One algorithm is the following:

- with probability 1/2 pick max-weight matching according to weights w_0 in the first-stage (and then do nothing in the second stage),
- with the remaining probability 1/2, pick nothing in the first stage and then pick a max-weight matching in the second stage.

Show that this algorithm (which does not even look at the distribution π) outputs a solution whose expected weight is at least 1/2 of the optimal value.

(b) (?) What properties of the max-weight matching problem did you use above?

(c) (?) Can you show a better-than 1/2 approximation algorithm for this problem? Or a matching hardness result?

Sample Average Approximation

8. **SST(tree) again: From Statistical Query to Black-Box models.** Suppose you are guaranteed that π is the independent choice model — there exist $p_i \in [0, 1]$ for all $i \in V$ such that $\pi_S = \prod_{i \in S} p_i \prod_{j \notin S}(1 - p_j)$. Now, instead of the SQ model, we want to work in the *black-box (BB)* model where you are just given a sampling oracle: you get an independent sample $S \leftarrow \pi$ at unit cost.

Consider the special case of SST on a tree (Problem 5) with *uniform inflation*: i.e., $c_S(e) = \lambda_e \cdot c_0(e)$ for all edges e and sets $S \subseteq V$.

(a) Suppose the p_i values are given explicitly, give an exact algorithm for this problem in polynomial time.

(b) For any i, show how to get an estimate $\hat{p}_i$ such that $\Pr[\hat{p}_i \in p_i \pm \varepsilon] \geq 1 - \delta$ using $\frac{O(1)}{\varepsilon^2 \delta}$ samples from the black box. (Hint: Chebyshev.)

(c) Improve to using only $\frac{O(\log \delta^{-1})}{\varepsilon^2}$ samples from the black box. (Hint: Chernoff bounds)

(d) Is this an ε-approximate oracle for r.v. $\mathbf{1}_{(i \in S)}$, in the sense of Problem 5(d)? If not, why not? (Hint: additive vs multiplicative guarantees.)

(e) ($\star$) Let $\lambda_{\max} = \max_e \lambda_e$. Show an algorithm in the BB model that outputs a $(1+\varepsilon)$ approximation to SST on a tree with probability at least $1 - \delta$ in time $\text{poly}(n, \lambda_{\max}, \varepsilon^{-1}, \log \delta^{-1})$.

The ideas you develop here will be ones used in the Sample Average Approximation techniques in Lecture #2.

Extensions

9. **(Robust Optimization.)** Recall the StocVC problem (Prob. 2) where we sought to find sets $V_0, \{V_k\}$ such that for each scenario k, $V_0 \cup V_k$ was a vertex cover for the edge set E_k, and the goal was to minimize the expected cost:

$$\min c_0(V_0) + \mathbb{E}_{k \leftarrow \pi}\big[c_k(V_k)\big] = c_0(V_0) + \sum_{k=1}^{K} \pi_k \, c_k(V_k).$$

   I.e., we just care about the expectation, so there could be some scenarios with small probability π_k for which the cost $c_k(V_k)$ is very large. The model of *robust optimization* is a more risk-averse model, the objective is now

$$\min c_0(V_0) + \max_k \big\{c_k(V_k)\big\}$$

   Show how the LP-based approach for StocVC (and also StocSC, and some other problems) naturally extends to this model.

10. **(Chance-Constrained Optimization.)** A more nuanced model is where we want to output feasible solutions to only a p^{th} percentile of the scenarios. For vertex cover, we are given an instance of Stochastic Vertex Cover, and also a parameter $p \in [0, 1]$. We want to optimize

$$\min c_0(V_0) + \max \big\{c_k(V_k)\big\}$$

   (like in robust optimization), but we only want that the probability mass of ks for which $V_0 \cup V_k$ is a feasible solution is at least p.[2]

   A useful (though seemingly unrelated) problem is the *dense-k-subgraph* (DkS) problem where given a graph G we want to pick a set S of vertices, such that the number of edges in the induced subgraph $G[S]$ is at least k. We want to minimize the size of S. This problem is believed to be very hard, both in theory and in practice.

   Give a reduction from DkS to chance-constrained vertex cover problem, showing the latter problem is as hard as the former. This should even hold in the explicit scenario model with polynomially many scenarios.

[2] Observe that when $p = 1$, we get the robust model from the Problem 9. When $p = 0$ then outputting empty sets V_0 and V_k is a feasible solution of cost 0.

Useful Bounds

We all know Markov's inequality (which only depends on the mean of the random variable):

Theorem 2 (Markov's Inequality) *For a non-negative random variable X with expectation $\mathbb{E}[X] = \mu$,*

$$\Pr[X \geq \lambda] \leq \frac{\mu}{\lambda}$$

and Chebyshev's inequality (which depends on the first two moments, the mean and the variance) which controls how concentrated the random variable is around its mean:

Theorem 3 (Chebyshev's Inequality) *For a non-negative random variable X with expectation $\mathbb{E}[X] = \mu$ and variance σ^2,*

$$\Pr[|X - \mu| \geq \lambda] \leq \frac{\sigma^2}{\lambda^2}.$$

A more powerful class of bounds are the so-called "Chernoff bounds" (which includes inequalities due to Chernoff, Hoeffding, Bernstein, and many others). Here is one useful bound:

Theorem 4 (A Chernoff Bound) *For independent $[0, m]$-bounded random variables $X_1, X_2, \ldots$, with $X := \sum_i X_i$ having mean $\mathbb{E}[X] \leq \mu$, given any $\lambda \geq 0$,*

$$\Pr\left[X \geq \mathbb{E}[X] + \lambda\right] \leq \exp\left\{-\frac{\lambda^2}{m(2\mu + \lambda)}\right\}$$

$$\Pr\left[X \leq \mathbb{E}[X] - \lambda\right] \leq \exp\left\{-\frac{\lambda^2}{3m\mu}\right\}.$$

Corollary 5 *For independent $\{0, 1\}$-valued r.v.s $X_1, X_2, \ldots$, with $X := \sum_i X_i$ having mean $\mathbb{E}[X] \leq \mu$, given any $\lambda \geq 0$,*

$$\Pr\left[|X - \mathbb{E}[X]| \geq O(\sqrt{\mu \log \delta^{-1}} + \log \delta^{-1})\right] \leq \delta.$$

These bounds consider the r.v. $X = f(X_1, X_2, \ldots)$ where f is the sum function, and the $\{X_i\}$ are independent and bounded random variables. Bounds can be given for more general cases, such as

- the function f is not just the sum, but some low-degree polynomial of the X_i's; or some other "well-behaved" function, or
- the X_is are not bounded but themselves have good tail behavior (like sub-Gaussian tails), or
- the X_is are mildly dependent on each other.

Bibliographic Notes

Stochastic Set Cover was first studied by Ravi and Sinha [7], and also by Immorlica, Karger, Minkoff, and Mirrokni [6]. The improved 2-approximation in Problem 2 for StocVC is due to Aravind Srinivasan. Problem 5 is based on [6].

Tree-embedding results start with the work of Alon, Karp, Peleg, and West [1], and were extended by Bartal [2], Fakcharoenphol, Rao, and Talwar [5], Elkin, Emek, Spielman and Teng [4], and Abraham and Neiman [**?**]. The $O(\log n)$ approximation for StocST appears in [6].

Robust optimization has been widely studied: Problem 9 is based on a paper of Dhamdhere, Goyal, Ravi, and Singh [3]. The chance-constrained optimization model is from the thesis of Vineet Goyal.

References

[1] N. Alon, R. M. Karp, D. Peleg, and D. West. A graph-theoretic game and its application to the k-server problem. *SIAM J. Comput.*, 24(1):78–100, 1995.

[2] Y. Bartal. On approximating arbitrary metrics by tree metrics. In *Proceedings of the 30th ACM Symposium on the Theory of Computing (STOC)*, pages 161–168, 1998.

[3] K. Dhamdhere, V. Goyal, R. Ravi, and M. Singh. How to pay, come what may: Approximation algorithms for demand-robust covering problems. In *Proceedings of the 46th Symposium on the Foundations of Computer Science (FOCS)*, pages 367–378, 2005.

[4] M. Elkin, Y. Emek, D. A. Spielman, and S.-H. Teng. Lower-stretch spanning trees. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 494–503, New York, NY, USA, 2005. ACM Press.

[5] J. Fakcharoenphol, S. Rao, and K. Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. System Sci.*, 69(3):485–497, 2004.

[6] N. Immorlica, D. Karger, M. Minkoff, and V. Mirrokni. On the costs and benefits of procrastination: Approximation algorithms for stochastic combinatorial optimization problems. In *Proceedings of the 15th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 684–693, 2004.

[7] R. Ravi and A. Sinha. Hedging uncertainty: Approximation algorithms for stochastic optimization problems. In *Proceedings of the 10th Integer Programming and Combinatorial Optimization Conference (IPCO)*, pages 101–115, 2004.