Demand-Robust Optimization

Instructor: Thomas Kesselheim

We have now seen a number of stochastic multi-stage optimization problems. In each of them, we could make a first-stage decision, when knowing only the probability distribution of the demand, and a second-stage decision, where we see the actual demand. Today, we will consider similar questions but with a different model of uncertainty.

We take a more worst-case perspective and assume to even not know the probability distribution over scenarios. Instead, we only know a list of scenarios. Rather than minimizing the expected cost, we would like to minimize the highest cost that we can possibly incur in a scenario.

1 Robust Vertex Cover

We will consider again the Vertex Cover problem. In the first stage, we do not know which edges E have to be covered and we can pick vertices as we like; vertex v has a cost of c_v^{I}. In the second stage, when knowing E, picking vertex v costs $c_v^{\mathrm{II}} \geq c_v^{\mathrm{I}}$.

Instead of knowing probability distributions, we only know that the edge set E will be one of the sets in a set $\mathcal{E}$. Our goal is to minimize the highest cost over all scenarios, that is,

$$\sum_{v \text{ selected in first stage}} c_v^{\mathrm{I}} + \max_{E \in \mathcal{E}} \sum_{v \text{ selected in second stage}} c_v^{\mathrm{II}} .$$

Let us call F_0^* the optimal first-stage choice, F_E^* the optimal second-stage choice if the scenario is E. Our goal is to devise an α-approximation to the optimal solution, that is, to come up with choices of F_0 and $(F_E)_{E \in \mathcal{E}}$ such that

$$\sum_{v \in F_0} c_v^{\mathrm{I}} + \max_{E \in \mathcal{E}} \sum_{v \in F_E} c_v^{\mathrm{II}} \leq \alpha \left(\sum_{v \in F_0^*} c_v^{\mathrm{I}} + \max_{E \in \mathcal{E}} \sum_{v \in F_E^*} c_v^{\mathrm{II}} \right) .$$

2 Explicit Scenarios

If the scenario set $\mathcal{E}$ is small, we can indeed follow the same approach as in the stochastic setting. We can first write the following LP relaxation.

$$\begin{aligned}
\text{minimize } & \sum_{v \in V} c_v^{\mathrm{I}} x_v + z & \\
\text{subject to } & \sum_{v \in V} c_v^{\mathrm{II}} y_{E,v} \leq z & \text{for all } E \in \mathcal{E} \\
& x_u + y_{E,u} + x_v + y_{E,v} \geq 1 & \text{for all } E \in \mathcal{E},\ \{u,v\} \in E \\
& x_u, y_{E,u} \geq 0 & \text{for all } E \in \mathcal{E},\ u \in V
\end{aligned}$$

The LP relaxation uses an auxiliary variable z to express the maximum operator.

Given a solution (x^*, y^*, z^*), we can choose again in the first stage all vertices for which $x_v^* \geq \frac{1}{4}$ and in the second stage the ones for which $y_{E,v}^* \geq \frac{1}{4}$.

The first-stage cost is clearly bounded by $4\sum_{v\in V} c^{\mathrm{I}}_v x^*_v$. The second-stage cost is at most $4\sum_{v\in V} c^{\mathrm{II}}_v y^*_{E,v} \leq 4z^*$ for every scenario. Therefore, this is a 4-approximation to the optimal choices.

This approach readily generalizes to Set Cover and, in fact, any approximation algorithm that uses this pattern. It is crucial that we are able to solve the LP relaxation. This is possible in polynomial time in the input length if the scenarios are listed explicitly in the input because then then input length is at least $|\mathcal{E}|$.

3 Implicit Scenarios: Cardinality Robust Version

There are also alternatives to an explicit list of scenarios. In particular, $\mathcal{E}$ could simply contain all sets of k edges. In this case, we would have $|\mathcal{E}| = \binom{m}{k}$, where m is the number of possible edges. This makes the LP too large to be solved efficiently.

Let's see what we can do instead. Our first algorithm will assume to know the optimal second-stage cost $T^* = \max_{E\in\mathcal{E}} \sum_{v\in F^*_E} c^{\mathrm{II}}_v$. Later on, we will get rid of this assumption.

What does it help us to know T^*? We know that E can be any set of size k. In any of these scenarios, the optimal solution cannot spend more than $\frac{T^*}{k}$ on average per edge. So, if an edge is significantly more expensive to cover in the second stage, we should do this in the first stage. This gives rise to the following algorithm.

- Let E_0 be set of edges such both endpoints have second-stage cost at least $4\frac{T^*}{k}$.
- Let F_0 be the Vertex Cover computed by an arbitrary (approximation) algorithm on E_0 with costs $(c^{\mathrm{I}}_v)_{v\in V}$.
- Let F_E be the cheaper endpoints of all edges in E that are not covered by F_0.

Theorem 12.1. *The algorithm is an $8\beta+4$-approximation to Cardinality Robust Vertex Cover if a β-approximation algorithm is used to compute F_0.*

Proof. It is very easy to upper-bound the cost of F_E. To this end, we use that every edge that is not contained in E_0 has at least one endpoint v with $c^{\mathrm{II}}_v \leq 4\frac{T^*}{k}$. So, even if we cover each $E\setminus E_0$ separately with the cheaper of its endpoints, we cannot incur cost more than $|E|4\frac{T^*}{k} \leq 4T^*$ in the second phase. So,

$$\max_{E\in\mathcal{E}} \sum_{v\in F_E} c^{\mathrm{II}}_v \leq 4T^* \quad .$$

Bounding the cost to cover the set E_0 is much more tricky. The optimal solution may cover only part of E_0 by its first-stage choice F^*_0. So, let $Q\subseteq E_0$ denote the edges are not covered by F^*_0. The important question is why it is not too expensive to cover *all of* Q. We will show the following lemma, which states that even covering all of Q in the second stage would have bounded cost.

Lemma 12.2. *There is a cover F_Q of Q with $\sum_{v\in F_Q} c^{II}_v \leq 8T^*$.*

Having shown the lemma, we can upper-bound the first-stage cost of our algorithm. Note that F_0 covers $E_0\setminus Q$, so $F^*_0\cup F_Q$ is a cover of E_0. Therefore

$$\sum_{v\in F_0} c^{\mathrm{I}}_v \leq \beta\left(\sum_{v\in F^*_0\cup F_Q} c^{\mathrm{I}}_v\right) \leq \beta\sum_{v\in F^*_0} c^{\mathrm{I}}_v + \beta\sum_{v\in F_Q} c^{\mathrm{II}}_v \leq \beta\sum_{v\in F^*_0} c^{\mathrm{I}}_v + 8\beta T^* \quad .$$

In combination

$$\sum_{v \in F_0} c_v^{\mathrm{I}} + \max_{E \in \mathcal{E}} \sum_{v \in F_E} c_v^{\mathrm{I}} \leq \left(\beta \sum_{v \in F_0^*} c_v^{\mathrm{I}} + 8\beta T^* \right) + 4T^* \leq (8\beta + 4) \left(\sum_{v \in F_0^*} c_v^{\mathrm{I}} + \max_{E \in \mathcal{E}} \sum_{v \in F_E^*} c_v^{\mathrm{II}} \right) \quad . \square$$

3.1 Proof of Lemma 12.2

The technical heart of the analysis is to prove Lemma 12.2. To this end, let V' be the endpoints of Q. We now have an induced graph $G' = (V', Q)$. In this entire proof, we will only speak about second-stage costs. So, each vertex $v \in V'$ has a cost c_v^{II}. By our choice, $c_v^{\mathrm{II}} \geq \gamma$ for all $v \in V'$, where $\gamma = 4\frac{T^*}{k}$. We would like to show that there is a vertex cover in this graph of cost at most $8T^*$.

Let $d_v = \left\lfloor \frac{c_v^{\mathrm{II}}}{\gamma} \right\rfloor$ denote the (rounded) factor by how much c_v^{II} exceeds γ. Clearly, $d_v \geq 1$ for all $v \in V'$.

We now construct the vertex cover F_Q in an almost greedy fashion as follows. Initially, set $x_v = 0$ for all $v \in V'$.

For all edges $e = \{u, v\} \in Q$

- Set $y_e = \min\{d_u - x_u, d_v - x_v\}$.
- Increase both x_u and x_v by y_e each.

Set $F_Q = \{v \in V' \mid x_v = d_v\}$. Note that this set covers all $e \in Q$ because after the respective iteration, $x_u = d_u$ or $x_v = d_v$ and $x_u \leq d_u$ as well as $x_v \leq d_v$.

The intuition behind this construction of the set F_Q as follows: For the first edge, we take the cheaper endpoint but we also "invest" in the more expensive endpoint. For later edges, we can make use of earlier investments.

The vector $(y_e)_{e \in Q}$ splits up the cost of covering Q by[1]

$$\sum_{v \in V'} x_v = 2 \sum_{e \in Q} y_e \quad ,$$

because each y_e is used to increase the x_v values of its two endpoints. Therefore, we have

$$\sum_{v \in F_Q} c_v^{\mathrm{II}} \leq \sum_{v \in F_Q} 2 d_v \gamma \leq 2\gamma \sum_{v \in V'} x_v = 4\gamma \sum_{e \in Q} y_e \quad .$$

So, it only remains to upper-bound $\sum_{e \in Q} y_e$. For subsets of Q of size at most k, we can use the following lemma.

Lemma 12.3. *For every $S \subseteq Q$ with $|S| \leq k$, we have $\sum_{e \in S} y_e \leq 2\frac{T^*}{\gamma}$.*

Proof. By adding arbitrary edges to S, we get a set E of size exactly k. This set E is a potential scenario that also the optimum has to be able to deal with. Note that none of the edges in Q and therefore in E are covered by F_0^*, so all of them are covered by F_E^*. We have

$$\sum_{v \in F_E^*} c_v^{\mathrm{II}} \leq T^*$$

[1] This vector can be viewed as a dual solution to the LP relaxation.

but also $c_v^{\mathrm{II}} \geq \gamma d_v$ for all endpoints of edges in Q. So

$$\sum_{v \in F_E^*} \gamma d_v \leq T^* \ .$$

Each edge is covered by at most two vertices in F_E^*. Therefore

$$\sum_{e \in S} y_e \leq 2 \sum_{v \in F_E^*} d_v \leq 2 \frac{T^*}{\gamma} \ .$$

□

Let $R = \{e \in Q \mid y_e > 0\}$. We will show that $|R| < k$. For the sake of a contradiction, suppose there is a subset $S \subseteq R$ of size exactly k. Note that all y_e are integral. Therefore, by Lemma 12.3,

$$|S| \leq \sum_{e \in S} y_e \leq 2 \frac{T^*}{\gamma} = \frac{k}{2} \ .$$

This is a contradiction because S was assumed to be of size exactly k.

So, because $|R| < k$, we can apply Lemma 12.3 on $S = R$. This now gives us

$$\sum_{e \in R} y_e \leq 2 \sum_{v \in F_E^*} d_v \leq 2 \frac{T^*}{\gamma} \ .$$

For the cost of F_Q, this implies

$$\sum_{v \in F_Q} c_v^{\mathrm{II}} \leq 4\gamma \sum_{e \in Q} y_e = 4\gamma \sum_{e \in R} y_e \leq 8T^* \ .$$

This proves Lemma 12.2.

3.2 Finding out T^*

Our algorithm so far relies on knowing T^*. We will now get rid of this assumption by essentially trying out different values. There is a big difficulty: We cannot actually tell how expensive our solution is. Computing $\sum_{v \in F_0} c_v^{\mathrm{I}} + \max_{E \in \mathcal{E}} \sum_{v \in F_E} c_v^{\mathrm{II}}$ requires taking the maximum over all possible E.

So, let us revisit where we need to know T^* and how it makes the analysis work. More precisely: What happens if we use a wrong value $T \ll T^*$ or $T \gg T^*$? Note that the second-stage cost will still be upper-bounded by $4T$. Only Lemma 12.2 and in particular Lemma 12.3 would not work. Fortunately, these are only used to bound the first-stage cost of our algorithm $\sum_{v \in F_0} c_v^{\mathrm{I}}$, which we can compute easily.

So, we can try out values for T. For each guess, we compute F_0 by our algorithm and then $\sum_{v \in F_0} c_v^{\mathrm{I}} + 4T$. Choose the T that minimizes this term.

It is important to observe that T being off T^* by a factor of $1 + \epsilon$ does not hurt. Therefore, we it suffices to try out all powers of $1 + \epsilon$.

4 Outlook

We have seen one example of demand-robust optimization. The approach can be generalized to many other problems such as Set Cover or Steiner Tree. Again, one covers in the first stage all demands that make the second stage non-trivial. The difficulty, as in our proof, is to show that this does not make the first stage too expensive. One can also go beyond the setting in which the scenarios are all size-k sets.

There are also other problems in the context of robust optimization. Rather than uncertainty in the demand, one can have uncertainty in the objective function. For example, you might want to find a vertex cover without knowing the actual vertex costs. These you only get to know after you have committed to the vertex cover. Such problems require different techniques.

References

- Thresholded Covering Algorithms for Robust and Max-Min Optimization, A. Gupta, V. Nagarajan, R. Ravi, ICALP 2010 (Result on Set Cover and others; our algorithm here is a simplification for Vertex Cover)