CSC373 Fall'19

Assignment 3 Solutions

Due Date: Nov. 17, 2019, by 11:59pm

Q1 [20 Points] LP and IP

Consider the following primal LP and IP in standard form:

$$\begin{array}{ll} \text{Maximize} & x_2 \\ \text{Subject to} & -3x_1 + 5x_2 \leq 8 \\ & 7x_1 + 3x_2 \leq 12 \\ & x_1, x_2 \geq 0 \end{array}$$

For the IP, add the constraints that x_1 and x_2 are integers.

Plot the feasible region of this program. *Note: You do not need to submit this with the assignment, but it will be helpful to plot the feasible region.* You can use any online graphing programs such as desmos, fooplot, etc.

(a) [5 Points] What are the vertices of the feasible region of the primal LP? (No explanation is needed.)

Solution: $(0, 0)$, $(0, 8/5)$, $(12/7, 0)$, and $(9/11, 23/11)$ (the last vertex is the point where both constraints are met with equality).

(b) [5 Points] What are the optimal solutions of the primal LP and IP? What are the corresponding optimal objective values? (No explanation is needed.)

Solution: The optimal solution of the primal LP is the $(9/11, 23/11)$ vertex, and the optimal objective value is 23/11.
For the IP, we note that the only feasible points are $(0, 0)$, $(1, 0)$, $(0, 1)$, and $(1, 1)$. So the optimal solutions are $(0, 1)$ and $(1, 1)$, where the optimal objective value is 1.

(c) [5 Points] Provide the dual LP of the primal LP above. Clearly indicate which dual variable in your formulation corresponds to which primal constraint.

Solution: Let y_1 and y_2 be the dual variables corresponding to the first and second constraint, respectively. The dual is then:

$$\begin{array}{ll} \text{Minimize} & 8y_1 + 12y_2 \\ \text{Subject to} & -3y_1 + 7y_2 \geq 0 \\ & 5y_1 + 3y_2 \geq 1 \\ & y_1, y_2 \geq 0 \end{array}$$

(d) [5 Points] What are the optimal solutions of the dual LP and its IP version? What are the corresponding optimal objective values? Does strong duality hold for this particular pair of primal and dual IP?

Solution: The optimal dual LP solution is at $(y_1, y_2) = (7/44, 3/44)$, where the optimal objective value is $23/11$. This matches the optimal primal LP objective value as expected by strong duality. For the dual IP, we can check that $(0, 0)$ is not a feasible solution, but both $(1, 0)$ and $(0, 1)$ are. The optimal dual IP solution is $(y_1, y_2) = (1, 0)$, where the optimal objective value is 5, different from the optimal primal IP objective. This is an example showing that strong duality does not hold for IPs.

Q2 [10 Points] Team Building

You are putting together a team of m players. The m positions in your team are ranked (so there is a position k for every $k \in \{1, 2, \ldots, m\}$. You can select your players from a pool of n people, denoted $N = \{1, \ldots, n\}$. Assume $n \geqslant m$.

Each person $i \in N$ has a celebrity rating c_i and suitability $s_{ik} \in [0, 1]$ that measures how well person i can play in position k on the team, where $k \in \{1, 2, \ldots, m\}$. You are also given a relation $I \subseteq N \times N$, where $(i, j) \in I$ indicates that person i and person j are *incompatible*, and should never be put on the team together. You can assume that this relation is symmetric, so $(i, j) \in I$ if and only if $(j, i) \in I$.

Your goal is to pick a team that maximizes the total celebrity rating of all selected players, subject to making sure that the total suitability of players for their assigned positions is at least 1 and no pair of players on the team is incompatible.

Give an linear or integer programming formulation for choosing the desired optimal team. Please include a high-level verbal description of your program and justify the correctness of your solution.

Solution: Create an Integer Program as follows. For each $1 \leqslant i \leqslant n$ and $1 \leqslant k \leqslant m$, there is an integer variable $t_{i,k} \in \{0, 1\}$ denoting whether person i is selected on the team and put into position k. The IP is then as follows:

Maximize $\displaystyle\sum_{i=1}^{n} \left(\sum_{k=1}^{m} t_{i,k} \right) \cdot c_i$

Subject to

$\displaystyle\sum_{k=1}^{m} t_{i,k} \leqslant 1, \forall 1 \leqslant i \leqslant n$ #A person can be in at most one position

$\displaystyle\sum_{i=1}^{n} t_{i,k} = 1, \forall 1 \leqslant k \leqslant m$ #Exactly one person should be in each position

$\displaystyle\sum_{k=1}^{m} t_{i,k} + \sum_{k=1}^{m} t_{j,k} \leqslant 1, \forall (i, j) \in I$ #From any incompatible pair, at most one person is selected

$\displaystyle\sum_{i=1}^{n} \sum_{k=1}^{m} t_{i,k} \cdot s_{i,k} \geqslant 1$ #Total suitability is at least 1

$t_{i,k} \in \{0, 1\}, \forall 1 \leqslant i \leqslant n, 1 \leqslant k \leqslant m.$

Correctness: As indicated by the comments next to each constraint, the program captures all the required constraints. The first two ensure that the variables represent a feasible selection of players into team and their placement into the various positions. The third constraint ensures that no two incompatible people are selected. And the fourth constraint makes sure that the total suitability of people for their assigned position (if selected) is at least 1. Subject to this, the objective function maximizes the total celebrity rating of the selected people. Note that given the first constraint, $\sum_{k=1}^{m} t_{i,k}$ is a binary expression indicating whether person i is selected at all.

Q3 [20 Points] P, NP, and $coNP$

For each decision problem below, state whether it belongs to P, NP, or $coNP$. Make the strongest claim that you can. E.g. if you can show that a problem is in P, then you should claim so, as this implies membership in NP and $coNP$. Similarly, e.g., if you think it is not in P but is in both NP and $coNP$, then you should claim so, instead of claiming membership in just one of them. Note that if you claim a problem is in NP or $coNP$, you *do not* have to show NP- or $coNP$-completeness.

In all the problems below, you may assume that a cycle in a graph means a *simple* cycle with no repeated vertices. Assume all graphs are undirected. $\mathbb{Z}^+$ is the set of *positive* integers.

Justify your answers. If you claim a problem is in P, give a polynomial-time algorithm and argue its correctness and running time. If you claim a problem is in NP and/or $coNP$, then prove this membership.

(a) [5 Points] **AllSmallCycles** ("ASC" for short)
Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.
Question: Does EVERY cycle in G that includes vertex s have total weight at most B?

(b) [5 Points] **AllLargeCycles** ("ALC" for short)
Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.
Question: Does EVERY cycle in G that includes vertex s have total weight at least B?

(c) [5 Points] **SomeLargeCycles** ("SLC" for short)
Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.
Question: Does SOME cycle in G include vertex s and have total weight at least B?

(d) [5 Points] **SomeSmallCycles** ("SSC" for short)
Input: Graph $G = (V, E)$, edge weights $w : E \to \mathbb{Z}^+$, vertex $s \in V$, bound $B \in \mathbb{Z}^+$.
Question: Does SOME cycle in G include vertex s and have total weight at most B?

Solution:

(a) AllSmallCycles $\in$ coNP

If the answer to the problem is NO, i.e., if there exists a cycle in G including vertex s with total weight greater than B, then given such a cycle, a TM can verify that (a) the given list of vertices indeed forms a cycle, (b) it indeed includes s, and (c) it indeed has total weight greater than B. All these operations require polynomial time.

(b) AllLargeCycles $\in P$

We first show that we can find the smallest weight W of any cycle including vertex s in polynomial time. Let C be such a cycle of weight W including s, and let v be a vertex adjacent to s in the cycle. Then, by the optimal substructure property, W is equal to the smallest weight of any path from s to v plus the weight of the edge from v to s.

In our algorithm, for every vertex $v \neq s$, we run Dijkstra's algorithm to find the smallest weight path from s to v and add to that the weight of the edge from v to s. The smallest of the $n-1$ numbers obtained is precisely the W we wanted.

Once we compute W, we can answer our question easily by checking whether $W \geqslant B$.

(c) SomeLargeCycles $\in$ NP

This problem is almost the complement of AllSmallCycles (except when the weight is exactly equal to B). So the same verifier as in Question **??** works, with minor modifications. For completeness, the exact verifier is as follows.

If the answer to the problem is YES, i.e., if there exists a cycle in G including vertex s with total weight at least B, then given such a cycle, a TM can verify that (a) the given list of vertices indeed forms a cycle, (b) it indeed includes s, and (c) it indeed has total weight at least B. All these operations require polynomial time.

(d) SomeSmallCycles $\in P$

This problem is almost the complement of AllLargeCycles. In part (b), we showed how to compute W in polynomial time. This question can then be answered by simply checking whether $W \leqslant B$.

Q4 [20 Points] Friendly Representatives

There is a set of n people, denoted $N = \{1, \ldots, n\}$. Some of them are friends with some other people. This is captured by a friendship relation $F \subseteq N \times N$, where $(i,j) \in F$ indicates that person i and person j are friends. You can assume that friendship is symmetric, so $(i,j) \in F$ if and only if $(j,i) \in F$.

Here is a decision problem, termed **FriendlyRepresentatives**.

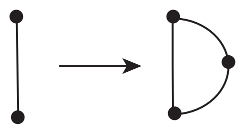
Input: Set of people N, friendship relation F, integer m.
Question: Does there exist $S \subseteq N$ with $|S| = m$ such that every person who is not in S is friends with someone who is in S?

(a) [5 Points] Show that this problem is in NP.

Solution: NOTE: This is precisely the problem of finding a dominating set of size m in a graph. To show the problem is in NP, take an instance (N, F, m) of the problem whose answer is YES. Let a solution S be given as advice. It is easy to verify in polynomial time that (a) S is indeed a subset of N, (b) $|S| = m$, and (c) for each $i \in N \setminus S$, $(i,j) \in F$ for some $j \in S$. The last step can be done in $O(n^2)$ time by checking every pair of people.

(b) [15 Points] Show that this problem is NP-complete. For this part, you can use the fact that **ConnectedVertexCover** problem, which takes a connected graph $G = (V, E)$ as input and decides whether it admits a vertex cover of size exactly k, is NP-complete.

[Hint: The following gadget might be useful!]



Solution: We show that ConnectedVertexCover $\leqslant_p$ FriendlyRepresentatives.

Given an instance (G, k) of ConnectedVertexCover, where $G = (V, E)$, $V = \{v_1, \dots, v_n\}$ and $E = \{e_1, \dots, e_m\}$, let us create an instance (N, M, m) of FriendlyRepresentatives as follows. We let $N = VV \cup \{w_1, \dots, w_m\}$ (add one person for each vertex and each edge), $M = E \cup \{(u, w_i), (w_i, v) : e_i = (u, v)\}$ (every pair of adjacent vertices are friends, and every vertex is also friends with each of its incident edges). Note that this recreates the structure in the gadget given in the hint. We let $m = k$.

Clearly, this reduction can be executed in polynomial time. We now show that the answer to the ConnectedVertexCover instance (G, k) is YES if and only if the answer to FriendlyRepresentatives (N, M, m) is YES.

ConnectedVertexCover is YES, then FriendlyRepresentatives is YES: If G contains a vertex cover C of size k, we show that the corresponding set of people is a set of friendly representatives. Note that every person corresponding to a vertex of G in $N \setminus C$ must be friends with someone in C by definition of vertex cover and the fact that G is connected. Also, by definition of vertex cover, every edge e_i is incident on some vertex in C, so each corresponding person w_i is also friends with someone in C. This shows that C is the desired solution of the FriendlyRepresentatives problem.

FriendlyRepresentatives is YES, then ConnectedVertexCover is YES: If (N, M, m) contains a set F of friendly representatives of size $m = k$, then we construct a vertex cover of G of size k as follows. First, we note that if some w_i corresponding to edge $e_i = (u, v)$ is selected as a friendly representative, then replacing it with either of u or v still keeps the set a valid solution of FriendlyRepresentatives (because w_i only helped with u and v anyway). Thus, given the set F, we can find another set $F' \subseteq V$ (i.e. containing no edge-people) of friendly representatives of size k. By definition, this is precisely a vertex cover of G as every edge is incident to at least one vertex in F'.