15-451/651 Algorithms, Fall 2017
Recitation #9 Worksheet

Rock-Paper-Scissors with a Twist Suppose we have a non-standard game of rock-paper-scissors, which is still zero-sum, but with the following payoffs for the row player (Alice):

		Bob plays		
		r	p	s
Alice plays	r	0	-1	2
	p	1	0.5	-1
	s	-1	2	-1

1. If Alice decides to play $\mathbf{p} = (p_1, p_2, 1 - p_1 - p_2)$ as her strategy, what should Bob play to minimize the payoff to Alice? What is Alice's payoff if he does this.

   Solution: Bob can either play rock, or paper, or scissors. He'll play whichever gives the lowest payoff to Alice. All other mixed strategies are averages of these three pure strategies. In this case Alice's payoff will be

$$\min \Big\{ p_1 \cdot 0 + \quad p_2 \cdot 1 + \quad (1 - p_1 - p_2) \cdot (-1),$$
$$p_1 \cdot (-1) + \quad p_2 \cdot (0.5) + \quad (1 - p_1 - p_2) \cdot (2),$$
$$p_1 \cdot (2) + \quad p_2 \cdot (-1) + \quad (1 - p_1 - p_2) \cdot (-1) \Big\}$$

2. Hence, write down the linear program that Alice must solve in order to find her best strategy $\mathbf{p}^\star$. You don't have to solve for the value of this game.

   Solution: Alice will try to maximize her payoff, given Bob plays the strategy in part B. So she will try to solve

$$\begin{aligned} & \max v \\ \text{subject to} \quad & p_1 \cdot 0 + p_2 \cdot 1 + (1 - p_1 - p_2) \cdot (-1) \geq v \\ & p_1 \cdot (-1) + p_2 \cdot (0.5) + (1 - p_1 - p_2) \cdot (2) \geq v \\ & p_1 \cdot (2) + p_2 \cdot (-1) + (1 - p_1 - p_2) \cdot (-1) \geq v \\ & p_1 + p_2 \leq 1 \\ & p_1 \geq 0, p_2 \geq 0 \end{aligned}$$

   Note that taking the new variable v and making all the three expressions at least as large as v, and then maximizing v makes sure that v is the minimum of the three expressions as desired in the previous part.

Coding up shortest paths as an LP: You can code up the s-t shortest-path problem as an LP. The input is a directed graph G with edge weights $w(e) \geq 0$, start node s, and a target t. We want to find a path from s to t of least weight.

1. *What are the variables?* Suppose we have a variable d_v for every vertex v, representing its distance from s. What are the constraints you should write?

   Solution: For any edge $e = (u, v)$ we put the constraint that $d_v \leq d_u + w(e)$. In other words, since one way to reach v is to reach u first and then go on edge e to v, the shortest-path distances must satisfy this property. We also add the constraint $d_s = 0$.

2. The objective? We claim it is *maximize* d_t. Why does this make sense?

   Solution: Any solution will have values d_v that are *no larger* than the shortest path from s to v, since by induction (on nodes in order of their true distance from s) the node u that comes right before v in the true shortest path from s will never have too large a value, and we have constrained $d_v \leq d_u + w(u, v)$. So d_t cannot be more than the shortest s-t distance.

   Also, if you set d_v to be the actual shortest-path distance from s to v, then this satisfies all the constraints in the LP. So the optimal value of d_t cannot be less than the shortest s-t distance either. Hence it must be equal.

   Another way to think about this: take a set of stones (one for each node) and strings (one for each edge), and tie a string of length $w(u, v)$ between stones u and v. Now pull s and t as far as possible from each other: the strings that become taut are those on the shortest path from s to t, and the furthest s and t can be pulled from each other equals their shortest-path distance.

Coding up shortest paths as an LP (Approach II): Have a variable x_e for each edge e, with constraints that $0 \leq x_e \leq 1$. (Think of $x_e = 1$ meaning we use that edge, and $x_e = 0$ meaning we don't, but of course the LP might assign fractional values.) Our goal is to minimize $\sum_e w(e)x_e$, subject to:

- One unit of "flow" leaves s: $\sum_{e=(s,v)} x_{sv} = 1$
- One unit of "flow" enters t: $\sum_{e=(v,t)} x_{vt} = 1$.
- For all $v \notin \{s, t\}$, we have flow-in = flow-out: $\sum_{e=(u,v)} x_{uv} = \sum_{e=(v,u)} x_{vu}$.

This is a min-cost flow (send 1 unit of s-t-flow with least cost). You can put edge-capacity 1, but since we send only 1 unit of flow, the capacity constraints don't matter.

1. Solve the LP to get an optimal solution. If we get back an integer solution (i.e., if $x_e \in \{0, 1\}$ for each edge e) argue that the edges with $x_e = 1$ give a shortest s-t path.

   Solution: The flow gives a path. And if there is a shorter path, that would be a solution with smaller cost to the LP as well, contradicting the optimality of x.

2. Suppose the optimal LP solution returns a fractional flow. Argue that any flow-carrying path from s to t is a shortest path.

Solution: Take the flow f and decompose it into flows on s-t paths, say α_i flow on path P_i, with $\sum_i \alpha_i = 1$. The LP objective function is $\sum_i \alpha w(P_i)$, where $w(P_i) = \sum_{e \in P_i} w_e$. Now if some P_i with $\alpha_i > 0$ was not a shortest path, we could reduce flow on that and put more flow on other paths to get a smaller cost LP solution.

Taking Duals Consider this maximization linear program:

$$\begin{aligned} \max(x_1 + 3x_2 - 3x_3) \\ \text{s.t.} \quad x_1 + x_2 + 2x_3 &\leq 2 \\ 7x_1 + 2x_2 + 5x_3 &\leq 6 \\ 2x_1 + 2x_2 - x_3 &\leq 1 \\ x_1, x_2, x_3 &\geq 0 \end{aligned}$$

1. Write down its dual LP. (Is it a maximization or minimization problem? What are the variables? Constraints?)

Solution: Remember that we are trying to find the lowest upper bound on the value on the primal LP by taking combinations of constraints. So the dual should be a minimization problem with one variable y_i for each constraint in the primal, where each y_i represent "how much" constraint i should be used in the optimal solution. We sum constraints of the form $thing \geq thing$, so we want each $y_i \geq 0$ to not flip the inequalities.

$$\begin{aligned} \min(2y_1 + 6y_2 + 1y_3) \\ \text{s.t.} \quad y_1 + 7y_2 + 2y_3 &\geq 1 \\ y_1 + 2y_2 + y_3 &\geq 3 \\ 2y_1 + 5y_2 - y_3 &\geq -3 \\ y_1, y_2, y_3 &\geq 0 \end{aligned}$$

2. What is the dual of the dual?

Solution: When we take the dual of the dual, we will get back the primal.

Taking Duals II Write down the dual of this minimization LP. Some of the inequalities are greater-than and some are less-than, and not all contraints have all variables

$$\begin{aligned} \max(x_1 - 3x_2 + 2x_3) \\ \text{s.t.} \quad 3x_1 + 2x_2 &\geq 2 \\ 2x_2 - x_3 &\leq 5 \\ x_1, x_2, x_3 &\geq 0 \end{aligned}$$

Solution: To make life easy, we first want to convert the LP to a nicer form. Since we have a maximization problem, we want all of the constraints to have a less than. We do this by multiplying the first constraint by negative one.

$$\begin{aligned} &\max(x_1 - 3x_2 + 2x_3) \\ \text{s.t.} \quad & -3x_1 - 2x_2 \le -2 \\ & 2x_2 - x_3 \le 5 \\ & x_1, x_2, x_3 \ge 0 \end{aligned}$$

Now we fill in all of the missing constraints.

$$\begin{aligned} &\max(x_1 - 3x_2 + 2x_3) \\ \text{s.t.} \quad & -3x_1 + 0x_2 - 2x_2 \le -2 \\ & 0x_1 + 2x_2 - x_3 \le 5 \\ & x_1, x_2, x_3 \ge 0 \end{aligned}$$

Now taking the dual is easy, just like before we want a minimization problem with one y_i for each constraint.

$$\begin{aligned} &\min(-2y_1 + 5y_2) \\ \text{s.t.} \quad & -3y_1 + 0y_2 \ge 1 \\ & 0y_1 - 2y_2 \ge -3 \\ & -2y_1 - y_2 \ge 2 \\ & y_1, y_2 \ge 0 \end{aligned}$$

Minimax from Duality Let the row-player's payoff be given by this (non-negative) matrix

	L	R
L	1	5
R	3	2

1. If the probabilities on the two rows are p_1 and p_2, write down an LP for the row player's optimal strategy.

   Solution: If the row player puts $p_1 \ge 0$ and $p_2 \ge 0$ on L and R respectively, then he/she wants to solve $\max_{p_1,p_2} \min(p_1 + 3p_2, 5p_1 + 2p_2)$ The LP is:

$$\begin{aligned} &\max v \\ \text{s.t.} \quad & p_1 + 3p_2 \ge v \\ & 5p_1 + 2p_2 \ge v \\ & p_1 + p_2 \le 1 \\ & p_1, p_2 \ge 0 \end{aligned}$$

   Note: We could do $p_1 + p_2 = 1$, but to maximize v it will be the case anyways.

2. Now take the dual of this LP. Show thhis dual is an LP computing the column player's optimal strategy (And hence string duality implies the minimax theorem)

   Solution: We are going to follows the same steps as Taking Duals II. Since we've got a maximization problem, we will make all of the constraints less than, and make the right side constants

$$\begin{aligned} &\max v \\ \text{s.t.} \quad & v - p_1 - 3p_2 \leq 0 \\ & v - 5p_1 - 2p_2 \leq 0 \\ & p_1 + p_2 \leq 1 \\ & v, p_1, p_2 \geq 0 \end{aligned}$$

   Now we do find the dual, calling the variables q_1, q_2 and w, for constraints 1,2 and 3.

$$\begin{aligned} &\max w \\ \text{s.t.} \quad & q_1 + q_2 \geq 1 \\ & -q_1 - 5q_1 + w \geq 0 \\ & -3q_1 - 2q_1 + w \geq 0 \\ & w, q_1, q_2 \geq 0 \end{aligned}$$

   Finally we put this in a similar form to the original LP:

$$\begin{aligned} &\max w \\ \text{s.t.} \quad & q_1 + q_2 \geq 1 \\ & q_1 + 5q_2 \leq w \\ & 3q_1 + 2q_1 \leq w \\ & w, q_1, q_2 \geq 0 \end{aligned}$$

   We again notice that in the optimal solution $q_1 + q_2 = 1$, so the optimal solution will give us the $\min_{q_1,q_2} \max(q_1 + 5q_2, 3q_1 + 2q_2)$, which is exactly what we would get if we assume the row player plays L and R with probabilities q_1 and q_2 and tried to optimize the column players strategy. These ideas hold in general, and for lots of other problems like maxflow-mincut.

Some Fun Stuff We can also model various NP-complete problems at (Integer) Linear programs.

1. K-Coloring as an LP. Given a graph G = (E, [n]), we want to give the graph a K-coloring.

   Solution: Let y_k be an indicator of whether or not color k is used, and x_{ik} be an indicator of whether vertex i is assigned to color k.

$$\begin{aligned} &\min \sum_k y_k \\ \text{s.t.} \quad & \sum_k x_{ik} = 1 \quad \forall i \leq n \\ & x_{ik} + x_{jk} \leq 1 \quad \forall k \leq K, (i,j) \in E \\ & x_{ik} \leq y_k \quad \forall k \leq K, i \leq n \\ & x_{ik}, y_k \geq 0 \quad \forall k \leq K, i \leq n \end{aligned}$$

program@epst

2. Vertex Cover as an LP. Given a graph G = (V, E) and a cost function $w : V \to \mathbb{R}^+$, we want to find the minimum cost vertex set such that every edge has at least one vertex in the set.

Solution: This one is really simple, make a variable x_v for every vertex, that represents whether or not it is in the vertex cover.

$$\begin{aligned} \min \sum_{v \in V} w_v x_v \\ \text{s.t.} \quad x_u + x_v \geq 1 \quad \forall (u, v) \in E \\ x_u \geq 0 \quad \forall u \in V \end{aligned}$$

If we make a vertex cover out of all of the vertices with $x_v \geq 1/2$, then we will get a 2 approximation for vertex cover.