Problem Set 0

Due date: Jan 12, 2017 at 9pm

1. Consider a gamble with n boxes, each box i being labeled with a probability p_i and a cost c_i; to open box i, a player needs to pay c_i, and the content of the box is revealed: a voucher is in it with probability p_i, otherwise it's empty. The event that box i contains a voucher is independent of all other boxes. The player will get a reward W if and only if every box is opened and each yields a voucher. The player can choose the order in which she opens the boxes, and can quit at any point. For example, whenever she sees an empty box, she knows she will never collect any reward, and therefore she should quit. This question asks for a polynomial time algorithm that, given the n labels and the reward W, outputs the optimal order in which the player should open the boxes, and determines if she should enter the game at all. (The player should enter the game if and only if the expected reward is no less than the expected cost.)

   For example, if the two boxes are labeled $p_1 = 0.3$, $c_1 = 10$, and $p_2 = 0.5$, $c_2 = 5$. If the player opens the first box first, she will incur a cost of 10 on box 1, and she will continue opening box 2 with probability 0.3 (because if box 1 is empty she should quit the game), which will cost her 5. The expected cost is $10 + 0.3 \times 5 = 11.5$. Similarly, if the player opens box 2 first, the expected cost is $5 + 0.5 * 10 = 10$. Therefore the optimal ordering is first box 2 then box 1. The expected reward is $0.3 \times 0.5W = 0.15W$. Therefore the player should enter the game if and only if W is at least $10/0.15$.

2. We consider the allocation of m indivisible items to n agents, in order to maximize the sum of valuations.

   (a) Suppose each agent i is interested in precisely one subset S_i of the m items; her value for S_i or any superset of S_i is $v_i \geq 0$, and her value for any other set of items is 0. Given all the S_i's and v_i's as input, show that an allocation $T_1, \cdots, T_n$ that maximizes the total values is NP-hard to compute. (By total value we mean the sum of the agents' values for their allocated subsets.)

   (b) Suppose each agent i has value $v_i(j)$ for each item j, and then for any subset S_i her value is $\min\{\sum_{j \in S} v_i(j), B_i\}$, where B_i can be seen as some budget. Given all $v_i(j)$'s and B_i's as input, show that an allocation $T_1, \cdots, T_n$ that maximizes the total values is NP-hard to compute.

3. We use (x, y) to denote an interval that starts at $x \in \mathbb{R}$ and ends at $y \in \mathbb{R}$. Let $I = \{(x_1, y_1), , (x_n, y_n)\}$ be a set of n intervals. Assume that $0 \leq x_i < y_i$ for each $i = 1, \ldots, n$. We say (x, y) is contained in (x', y') if $x' < x$ and $y < y'$. You may assume that $x_1, \ldots, x_n$ are all distinct and $y_1, \ldots, y_n$ are all distinct as well.

   (a) Give an $O(n \log n)$ time algorithm to determine how many intervals in I are contained in some other interval in I.

(b) We say (x_i, y_i) and (x_j, y_j) form an inclusion relation if one of them contains the other. Give an $O(n \log n)$ time algorithm to determine how many inclusion relations there are among the intervals in I.