Assignment 3

Q1 [20 Points] Auction Optimization

Suppose there is a marketplace with S suppliers, C customers and G goods. Supplier i supplies at most S_{ik} units of good k and asks for $\$A_{ik}$ price per unit of good k. customer j demands at most D_{jk} units of good k and bids $\$B_{jk}$ per unit of good k. Imagine that you are the auctioneer: you collect all the pricing and bidding information, then allocate supplies to customers. You would be making profit of an amount equal to the absolute difference between the bidding price $\$B_{jk}$ and the asking price $\$A_{ik}$ for every unit of good k sold by supplier i to customer j.

(a) [10 Points] Provide an LP formulation to maximize your profit.

(b) [5 Points] Suppose supplier i and customer j are not at the same location and the available fleet capacity allows for transportation of at most U_{ij} goods from supplier i to customer j. In addition you incur a cost of C_{ij} per unit of any good transported from supplier i to customer j. Provide an LP formulation to maximize market clearing profit under these additional conditions.

(c) [5 Points] Suppose that the fleet capacities U_{ij} of (b) are given in terms of kilograms and each good k weighs W_k kilograms. Modify your formulation to (b).

Q2 [20 Points] Minimizing Maximum Deviation

Linear programming is often used to solve statistical problems. We consider two examples here. We are given n data points (x_i, y_i) and wish to find a line of best fit $y = ax + b$ for some coefficients a, b that best approximates the behaviour of the data points. The l_∞ distance between a point (x_i, y_i) and a line $y = ax + b$ is defined as the quantity $|y_i - ax_i - b|$.

(a) [10 Points] Suppose we want to minimize the l_∞ error for the line of best fit, defined as the maximum of l_∞ distance over the set of data points. Give a linear program that will produce a line of best fit with minimum l_∞ error.

(b) [10 Points] In the *classification* problem, you are given m data points of type 1 and n data points of type 2. We say that a line $y = ax + b$ separates the points all data points of type 1 satisfy $y_i < ax_i + b$ and all data points of type 2 satisfy $y_i > ax_i + b$. Give a linear program that will determines if there is a line that separates the points, and if one exists, maximizes the gap δ defined as $\max(e_1, e_2)$, where e_i is the minimum l_∞ distance between the line and points of type i.
QUESTIONS BELOW HAVE BEEN MOVED TO A4, PLEASE DO NOT SUBMIT WITH A3

Q3 [20 Points] Nearly-SAT

Nearly-SAT is the problem described as follows:

- Input: A CNF Formula $\mathcal{F}$ over n variables $x_1, \ldots, x_n$ and having m clauses
- Output: "Yes" if there is an assignment of the variables $x_1, \ldots, x_n$ that satisfies exactly $m-1$ clauses, "No" otherwise

(a) [5 Points] Show that Nearly-SAT is in NP.

(b) [7 Points] Given a CNF formula $\mathcal{F}$ over the variables $x_1, \ldots, x_n$ and which has m clauses, construct a CNF formula $\mathcal{F}'$ over the same set of variables and which has two additional clauses (so $m+2$ in all) such that an assignment to the variables $x_1, \ldots, x_n$ that satisfies all m clauses of $\mathcal{F}$ satisfies precisely $m+1$ clauses of $\mathcal{F}'$.

(c) [8 Points] Use part (b) to give a polynomial time reduction from SAT to Nearly-SAT and show that Nearly-SAT is NP-hard.

Q4 [20 Points] Tile Covering

An architect is given a set of rectangular tiles t_i, each with some length l_i and width w_i ($l_i \geq w_i$). There is a wall with length l and width w, and the architect would like to see if there is a way to cover the wall with tiles so that each tile is placed horizontally or vertically, there are no overlaps between the tiles or gaps on the wall, and furthermore each tile is placed on the wall. Assume that all dimensions are given as rational numbers.

(a) [5 Points]The **Architect** problem is defined as the following decision problem: given the tiles and the dimensions of the wall, is there a way to cover the wall according to constraints? Show that the Architect problem is in NP.

(b) [15 Points] Show that the Architect problem is NP-Complete. You may assume that the following problem is NP-Complete:
Partition: Given a set of positive integers S that sum to n, is there a subset $S_1 \subseteq S$ that sums to $\frac{n}{2}$?