ISE 5113 Advanced Analytics and Metaheuristics
Homework #3

Instructor: Charles Nicholson

See course website for due date

Requirement details

1. Homeworks should be submitted in a clean, clear, concise electronic format. You must show your logic, work, solutions, and/or code where appropriate.
2. Any code (e.g. AMPL) is part of your solution – make sure to provide comments on what your code is doing. Keep it clean and clear!
3. For any mathematical programming problem, in addition to solving the problem and responding to the questions, please ensure you clearly define the following elements in the your answer: (i) **any necessary assumptions**, (ii) **decision variables**, (iii) **objective and objective function**, and (iv) **constraints**.

Question 1: Mr. X (12 points)

Assume a continuous decision variable $0 \leq x \leq 1000$. Write constraint(s) to meet the following requirements. You can add any auxiliary variables as you like, but you must define them. Make sure your constraints are as "tight" as possible.

(a) $x = 0$ or $x \geq 10$

(b) $0 \leq x \leq 15$ or $30 \leq x \leq 100$

(c) $x \in \{12, 12.3, 87, 99.1\}$

Question 2: Valid inequalities (12 points)

Identify two valid inequalities for the following mathematical program. Explain how each is a valid inequality.

$$\begin{aligned} \min\ & 14x_1 + 2x_2 + 11x_3 + 9x_4 + x_5 \\ \text{s.t. } & 3x_1 - 4x_2 + 2x_3 - 3x_4 + x_5 \leq -2 \\ & x_i \in \{0, 1\} \text{ for } i = 1, \ldots, 5 \end{aligned}$$

Question 3: Gizmos and Gadgets (30 points)

You work for *Gizmos and Gadgets* and are helping the purchasing manager determine her monthly strategy for buying widgets (a necessary internal component of your company's gizmos). She has her choice of four suppliers: *Widgets International Incorporated* (WII), *Widgets 'R Us* (WRS), *Widgets Unlimited* (WU), and from *World of Widgets* (WOW). These suppliers have different price structures and available widgets as seen in the table below. She may source the widgets from any combination of these suppliers.

Table 1: Supplier data

Supplier	Price structure	Available
WII	\$4.25 per widget	10000
WRS	\$3.15 per widget, only if you buy at least 7500 (otherwise, no widgets for you!)	15000
WU	\$1.90 per widget + fixed fee: \$15000	9000
WOW	\$5.50 each for first 5000, \$3.50 each for next 7500, \$2.00 each for any more	25000

Create a table of results (similar to Table 2) for the following possible demands 17000, 18000, 19000, 28000, 32000.

Table 2: Example table of results (fake results data)

	Purchased from suppliers				
Demand	WII	WRS	WU	WOW	Total Cost
17,000	1500	9800	2000	3000	\$42,000
18,000	3200	1100	9700	5000	\$97,000
19,000			...		
28,000			...		
32,000			...		

Question 4: Facility Location (46 points)

Consider the Firehouse Facility Location Problem formulated in class: Assume that the population is concentrated in I districts within the city and that district i contains p_i people. Preliminary analysis (land surveys, politics, and so forth) has limited the potential location of firehouses to J sites. Let $d_{ij} \geq 0$ be the distance from the center of district i to site j. Determine the "best" site selection and assignment of districts to firehouses given a limited budget B. Every district should be assigned to exactly one firehouse. No district should be assigned to an "unused" site. It is important that either at least sites 1 and 2 or sites 3 and 4 are selected. There is a fixed cost to build a firehouse on a site; and a variable cost for servicing the population in that district. The "best" solution will minimize the "worst-case" distance from selected sites to assigned districts.

(a) (8 points) Create a valid AMPL model file for this problem (include the model in your submission document as well as a separate attachment.)

(b) (26 points) Use the NEOS server (`https://neos-server.org`) to submit and solve the instance of this problem provided in the `FacilityLocation.dat` file. Please see instructions on how to do this below.

Note: to demonstrate the basic branch-and-bound approach, please disable some of the more advanced solution techniques used in CPLEX. That is, turn off *presolve*, all cutting plane methods *mipcuts* and *splitcuts*, and the built-in *heuristics*; finally, set the *MIP search* to the basic method in the CPLEX options: `'presolve=0 mipcuts=-1 splitcuts=-1 heuristicfreq=-1 mipsearch=1'`

You will also need turn on some display options to evaluate the solution time and to access node-level results: `'timing=1 mipdisplay=5 mipinterval=1 mipsearch=1'`

You can read about these options and others here:
`http://ampl.com/products/solvers/ilogcp-options/`

i. (8 points) Solve the provided instance of the facility location problem
   - α) Determine optimal solution and objective value
   - β) Determine how much of the budget was used
   - γ) Determine total solution time; determine the total number of branch-and-bound nodes used in the algorithm
   - δ) What was the root relaxation value? What and when (node number) was the the first incumbent value found?

ii. (12 points) Create a graph of upper bound and lower bound objective values by node number (if there are too many nodes, you collect data and plot values for every k nodes, where $k > 1$ according to the size of the node tree.)

iii. (6 points) Turn the cutting plane options on (*mipcuts* and *splitcuts*) and re-solve. Describe how this affects the enumeration tree and solution time.

(c) (12 points) : Reformulate the problem to minimize the average distance (instead of minimizing the worst-case distance) and resolve. Note: the original problem is called the p-center problem, the new variation is called the p-median problem. How does your solution change?

Information regarding NEOS

The NEOS Server is a free internet-based service for solving numerical optimization problems. Hosted by the Wisconsin Institute for Discovery at the University of Wisconsin in Madison, the NEOS Server provides access to more than 60 state-of-the-art solvers in more than a dozen optimization categories.

By using NEOS you are able to have access to full commercial solvers (not limited versions) and solve large problems in with AMPL and CPLEX.

Submitting a job to NEOS `https://neos-server.org/neos/`)

The basic idea is that (1) you select "Submit a job to NEOS", (2) scroll down to select your problem type, solver, and input interface (e.g., Linear Programming, CPLEX, AMPL Input), (3) read the instructions on the page that comes up: you will submit a model and data file and, optionally, a commands file. Note: do not specify "CPLEX" as the solver in any options commands, (4) enter your email address and submit, (5) note the job number and password so you can access your results.