Pontificia Universidad Católica de Chile
Departamento Ingeniería Industrial y Sistemas
ICS113H -Optimización Honors
Profesor: Juan Pablo Contreras

Tarea 2

Fecha de Entrega: Viernes 19 de Abril 10:00

Reglas de la Tarea

- La tarea se puede realizar de a pares o de forma individual.
- La tarea parte el día miércoles 10 de Abril a las 10:00 am y termina el día viernes 19 de Abril 10:00 am.
- Esta tarea es grupal y el desarrollo y discusión debe ocurrir dentro de cada grupo. No se recomienda distribuir la resolución de las preguntas por separado, ya que ello va contra la idea de aprendizaje colaborativo y preparación individual de la interrogación a la que está asociada la tarea. Pueden discutir los problemas con el profesor y los ayudantes del curso, pero al final cada grupo debe entregar sus propias soluciones, desarrolladas y escritas por el grupo. La copia o intento de copia a otros grupos será sancionada dependiendo la gravedad (a ser determinada por el equipo docente del curso) con consecuencias que podrían ir desde un 1.0 en la nota de la Tarea hasta escalar a la Dirección de Docencia de la Escuela, con posible REPROBACIÓN AUTOMÁTICA del curso.
- Por entregar en LaTeX se darán 3 décimas en la nota. Estas **no** son transferibles a otras actividades. Para subirlo tendrá que enviar un archivo .pdf y el archivo .tex (o un .zip si desea). En caso de no entregar en LaTeX basta con subir un archivo .pdf que contenga las respuestas (con letra legible). En ambos casos los .pdf tienen que contener los nombres de los integrantes. El no cumplir con el formato conllevará descuentos en la nota. La nota de la tarea será a lo más 7,0 en caso de exceder la nota máxima.
- Ponga especial atención a el formato de entrega del código en python que debe ser un archivo llamado MAIN.PY. Puede entregar este archivo junto con el .pdf comprimidos en un único archivo .zip. El orden, legibilidad y documentación del código serán parte de la evaluación.
- El problema 1 vale 3 puntos. Los problemas 2 y 3 vales 1.5 puntos cada uno.

Problema 1: Implementación de un PL

Una empresa manufacturera está buscando ampliar su operación por lo que desea abrir M nuevas plantas de producción. Un estudio de mercado ha determinado un total $N = 2M$ de posibles ubicaciones para abrir nuevas plantas. Cada potencial nueva planta $i = 1, ..., N$ tiene asociado un costo de adquisición F_i por adquisición del terreno (que se paga solo una vez si es que se decide instalar una sucursal en la ubicación), un costo unitario de producción c_i y una capacidad máxima de producción O_i. Las nuevas plantas se construirán para satisfacer la demanda D_{lt} de L clientes $l = 1, ..., L$ en los próximos $t = 1, ..., T$ periodos. Por otro lado, dado que las plantas no tienen capacidad de almacenamiento, la compañía cuenta con K bodegas distribuidas por la ciudad y cada bodega $k = 1, ..., K$ tiene una capacidad de almacenamiento Q_k. De esta forma, en cada periodo t la demanda de un cliente se puede satisfacer usando la producción de las plantas o el inventario de las bodegas (o una mezcla de ambas). El costo unitario de almacenamiento en cada bodega es h_k. Considere que las distancias de las potenciales ubicaciones a las bodegas y los clientes son conocidas. Además, asuma que el costo de transporte por unidad se calcula como una constante C multiplicada por la distancia (Euclidiana) entre el origen y el destino.

Definamos d^1_{ik} a las distancias entre potenciales plantas y bodegas, d^2_{il} a las distancias entre potenciales plantas y clientes, y d^3_{kl} a las distancias entre bodegas y clientes. Sea además s_{k0} la cantidad de producto almacenada en la bodega k al comienzo del horizonte temporal. El problema anterior se puede formular de la siguiente forma

$$
\begin{aligned}
\text{mín} & \sum_{i=1}^{N}\sum_{t=1}^{T} c_i x_{it} + \sum_{i=1}^{N} F_i z_i + \sum_{t=1}^{T}\sum_{k=1}^{K} h_k s_{kt} \\
& \qquad + C \sum_{t=1}^{T} \left(\sum_{i=1}^{N}\sum_{k=1}^{K} d^1_{ik} y_{ikt} + \sum_{i=1}^{N}\sum_{l=1}^{L} d^2_{il} w_{ilt} + \sum_{k=1}^{K}\sum_{l=1}^{L} v_{klt} d^3_{kl} \right) \\
\text{s.a.} & \sum_{i=1}^{N} z_i = M \\
& x_{it} \leq O_i z_i, \qquad \forall i = 1, ..., N;\ t = 1, ..., T \\
& s_{kt} \leq Q_k, \qquad \forall k = 1, ..., K \\
& \sum_{k=1}^{K} y_{ikt} + \sum_{l=1}^{L} w_{ilt} = x_{it}, \qquad \forall t = 1, ..., T;\ i = 1, ..., N \\
& \sum_{i=1}^{N} w_{ilt} + \sum_{k=1}^{K} v_{klt} = D_{lt}, \qquad \forall t = 1, ..., T; l = 1, ..., L \\
& s_{kt} = s_{k,t-1} + \sum_{i=1}^{N} y_{ikt} - \sum_{l=1}^{L} v_{klt}, \qquad \forall t = 1, ..., T; k = 1, ..., K \\
& x_{it}, s_{kt}, y_{ikt}, w_{ilt}, v_{klt} \geq 0, \qquad \forall t = 1, ..., T; k = 1, ..., K; i = 1, ..., N, l = 1, ..., L \\
& z_i \in \{0, 1\}, \qquad \forall i = 1, ..., N
\end{aligned}
$$

Base de datos:

La implementación computacional de este problema debe ser un código que pueda recibir cualquier instancia de valores de los parámetros. No puede asumir valores ni cardinalidades de ningún parámetro o conjunto. Toda la información de las instancias de datos vendrán en archivos con extensión .csv. Ejemplos de estos archivos se pueden descargar desde canvas, junto al enunciado de la tarea. Es fuertemente recomendado testear la robustez del código con estas instancias, ya que para la evaluación se utilizarán unos de idéntico formato pero diferentes valores y dimensiones. A continuación se detalla el formato de cada uno de estos archivos.

- El archivo COSTOS.CSV tiene 4 filas. La primera fila contiene los costos c_i para $i = 1, ..., N$. La segunda fila contiene los costos F_i para $i = 1, ..., N$ y la tercera fila contiene los costos h_k para $k = 1, ..., K$. La última fila contiene el valor de C.
- El archivo OFERTAS.CSV tiene 1 fila con las ofertas O_i para $i = 1, ..., N$.
- El archivo DEMANDAS.CSV contiene L filas y T columnas con las demandas D_{lt}.
- El archivo BODEGAS.CSV contiene 2 filas y K columnas con las capacidades (primera fila) y stock inicial (segunda fila) para cada bodega $k = 1, ..., K$.
- El archivo DISTANCIAS1.CSV contiene N filas y K columnas con las distancias d^1_{ik}.
- El archivo DISTANCIAS2.CSV contiene N filas y L columnas con las distancias d^2_{il}.
- El archivo DISTANCIAS3.CSV contiene K filas y L columnas con las distancias d^3_{kl}.

Código main.py

Se debe elaborar y entregar, junto con el desarrollo de la tarea, un código llamado main.py. Ese código debe cumplir una serie de requisitos:
1. Que al ser ejecutado en una carpeta que contenga los archivos de la base de datos, logre abrir los archivos y extraer su información. Puede asumir que los archivos seguirán el formato de archivos de manera exacta.
2. Se debe resolver el modelo utilizando la librería Gurobi. Cada parte del código debe estar comentada de manera brevísima y clara.
3. El código debe imprimir en consola de manera clara la solución óptima. Para ello, primero se debe imprimir cuáles son las ubicaciones que se seleccionaron. Luego, se debe imprimir por cada periodo las cantidades a producir (x_{it}), las cantidades a enviar desde las plantas a los clientes (w_{iklt}), las cantidades a enviar desde las plantas a las bodegas (y_{ikt}) y las cantidades a enviar desde las bodegas a los clientes (v_{klt}). Note que solo nos interesan aquellos transportes que son no nulos, los demás se deben omitir al presentar la solución.

Problema 2: Modelo no lineal

La empresa AguasCity tiene el monopolio de distribución de agua en el país. AguasCity quiere minimizar sus costos y satisfacer la demanda en un día típico de 24 horas. A lo largo y ancho del país, la empresa tiene instaladas plantas de tratamiento de agua siendo N el conjunto de plantas. Además, existe un conjunto

M de puntos de distribución de agua en el país. Para cada hora $h \in H = \{1, \ldots, 24\}$, existe una demanda de agua $d_{m,h}$ en litros que debe llegar al punto de distribución $m \in M$, después de considerar las pérdidas en las tuberías. Cada planta $n \in N$ decide cuánta agua enviar de manera constante a cada punto de distribución durante cada hora $h \in H$, sin superar su capacidad máxima de producción por hora $CN_{n,h}$. Todas las plantas están apagadas inicialmente, sin poder producir. Las plantas se pueden encender para producir a un costo F_n para la planta $n \in N$. Además, una vez encendidas las plantas solo pueden ser apagadas al final del día y por cada hora de que la planta esté encendida se debe pagar un costo de k_n pesos.

Para la red de agua, AguasCity tiene un grafo dirigido $G(V, E)$ donde $V = N \cup M$ es le conjunto de vértices compuesto por las plantas y los puntos de distribución, y $E \subseteq N \times M$ es el conjunto de aristas o tuberías conectando las plantas con los centros de distribución. Cada tubería $e \in E$ tiene una capacidad máxima CL_e de litros de agua cada hora y un valor ρ_e con $0 \leq \rho_e < CL_e$ que índica la capacidad regulada de la tubería. De acuerdo a los expertos de AguasCity, si una tubería opera por debajo de la capacidad regulada entonces no hay pérdidas de agua por esa tubería. Sin embargo, si la cantidad de agua por hora que pasa por la tubería $e \in E$ sobrepasa el límite ρ_e entonces existirá una perdida de agua que es proporcional al cuadrado del exceso de agua (con respecto al límite ρ_e).

El costo variable medio de producir L litros de agua está determinado en cada planta $n \in N$ y en cada hora $h \in H$ por una función afín decreciente: $CMeV(L) = a_{n,h} - b_{n,h} \cdot L$. Se debe considerar que $a_{n,h}$ está en pesos, y que la de $b_{n,h}$ está en pesos por litro. El costo de producción en una hora se obtiene multiplicando la cantidad de agua producida por el costo variable medio de producir esa cantidad.

Formule un modelo de optimización no lineal que ayude a la empresa AguasCity a minimizar sus costos en total satisfaciendo la demanda.

Problema 3: Convexidad

Resuelva dos de los tres problemas siguientes:

$i)$ Demuestre que el conjunto de soluciones óptimas de un problema convexo es un conjunto convexo.

$ii)$ Sea C un convexo y $\alpha, \beta > 0$ dos escalares. Demuestre que $(\alpha + \beta)C = \alpha C + \beta C$. Muestre con un ejemplo que esta igualdad no es cierta si C no es convexo.

Aclaración: En este problema el conjunto $\alpha C := \{\alpha x \ : \ x \in C\}$, mientras que la suma se refiere a la suma de Minkowski $A + B := \{a + b \ : \ a \in A, b \in B\}$.

$iii)$ Sea $f : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}$ una función convexa. Sea $C \subseteq \mathbb{R}^m$ un convexo y asumamos que el problema $\min_{y \in C} f(x, y)$ tiene solución óptima para todo $x \in \mathbb{R}^n$. Definamos $g(x) = \min_{y \in C} f(x, y)$. Demuestre que g es una función convexa.