Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería Industrial y de Sistemas
Profesores: Gustavo Angulo, Raimundo Cuadrado,
Camila Balbontín, Gonzalo Pérez
Ayudante Coordinador de Tareas: Juan Enrique Hurtado

ICS1113 - Optimización
1er semestre del 2024

Tarea 1 ICS1113

Reglas de la tarea:

- La tarea se puede realizar de a pares o de forma individual (las parejas pueden ser de diferentes secciones). En caso de ser de a pares se debe inscribir con su pareja en los grupos de Canvas.
- La tarea debe ser subida **por solo uno de los integrantes** a Canvas.
- La tarea se entrega el día 15/04/2024 a las 23:59 horas.
- Existirá la opción de usar un cupón de atraso solamente una vez entre las tres tareas, dando la posibilidad de entregar hasta 24 horas después de finalizada la entrega. Para poder acceder a este cupón se deben inscribir con el nombre de los integrantes de la tarea en un Google Forms que será habilitado en su debido tiempo y que cierra junto a la fecha oficial de la entrega. En caso de ser de a pares ambos deben tener un cupón disponible y se usará uno por cada uno y tendrán 24 horas extra (no 48).
- Las tareas atrasadas no serán corregidas, así que recuerden entregarlas a tiempo.
- Esta tarea es grupal y el desarrollo y discusión debe ocurrir dentro de cada grupo. No se distribuyan la resolución de las preguntas por separado, hagan realmente un trabajo grupal de desarrollo ya que el no hacerlo 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 los profesores 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 la escalación a la Dirección de Docencia de la Escuela, con posible **REPROBACIÓN AUTOMÁTICA** del curso.
- Por entregar en latex se darán 3 décimas para la tarea entregada, las décimas no son transferibles a otras actividades. Para subirlo tendrá que enviar un archivo .pdf y el archivo .tex correspondientes (también se puede incluir el .zip de Overleaf). En caso de no entregar en latex basta con subir un archivo .pdf que contenga las respuestas. Si se hace a mano, debe ser una tarea legible o habrán descuentos a criterio del corrector. En ambos casos los .pdf tienen que contener los nombres de los integrantes.
- El formato de entrega oficial es subir 2 archivos como mínimo y un tercer archivo opcional. Los dos archivos obligatorios son el pdf y main.py. El archivo opcional es el de extensión .tex o el .zip del proyectod e latex de Overleaf.
- Se abrirá un foro en Canvas donde se responderán las dudas que puedan surgir del enunciado.

Pregunta 1 **[20 Puntos]**

Aguas es una empresa de camiones aljibe de una gran región. Busca minimizar los costos asociados a cumplir una demanda de despachos de agua a lo largo de un conjunto T de días. Por una parte, existe un conjunto N de puntos de demanda, y cada punto $n \in N$ solicita exactamente $d_{n,t}$ m^3 de agua cada día $t \in T$. Por otra parte, la empresa tiene un conjunto Γ de camiones aljibe idénticos que tienen una capacidad máxima de W m^3 de agua para transportar en cada viaje. La empresa cuenta con la posibilidad de estacionar sus camiones en un conjunto Θ de parques de estacionamientos, cada uno con capacidad mayor o igual a $|\Gamma|$ estacionamientos. Todas las noches, y cuando no están viajando, los camiones deben permanecer en un parque de estacionamiento. En cada parque, cada camión debe pagar una tarifa que se cobra sólo en la noche a partir del día 1. Ahí recargan el agua necesaria sin costo. Todos los camiones, comienzan el día 1 en la mañana desde el parque de estacionamiento número 1. Cada día, cada camión debe decidir si sale o no a despachar agua a un único punto de demanda y regresar a cualquier parque de estacionamiento para pasar la noche. Si no sale, debe permanecer en el parque hasta el próximo día, pagando denuevo la tarifa del parque en el que permanece. La tarifa para cada camión en un parque $b \in \Theta$ es CL_b pesos por noche. Para cualquier camión, el costo de viajar entre el parque $b \in \Theta$ y el punto de demanda $n \in N$ es $c_{b,n}$ pesos, ya sea de ida o de vuelta. Las demandas de agua suelen ser muy grandes y las distancias a recorrer muy largas, pero realizables en un día. Es por esa razón que, para simplificar la logística, sólo se le permite, como máximo, a cada camión viajar a un punto de demanda cada día. Los camiones no puede devolverse con agua en sus estanques, por lo que se debe conocer la cantidad exacta de agua que llevará cada uno cada día.

Formule un modelo de optimización lineal mixto, que logre minimizar los costos de la empresa Aguas durante el marco de tiempo indicado, y satisfaciendo la demanda. Escriba claramente las variables de decisión, restricciones y función objetivo del modelo.

Solution:

Variables [7.5 pts]

[1.5 pts] por cada variable bien definida incluyendo su N.V. Las variables a usar son:

- $x_{c,n,t} := \begin{cases} 1 & \text{si el camión } c \in \Gamma \text{ va a dejar agua a } n \in N \text{ el día } t \in T. \\ 0 & \text{e.o.c.} \end{cases}$
- $k_{c,n,t}$:= cantidad de m^3 de agua que el camión $c \in \Gamma$ va a dejar a $n \in N$ el día $t \in T$.
- $y_{c,b,t} := \begin{cases} 1 & \text{si el camión } c \in \Gamma \text{ termina en el parque } b \in \Theta \text{ el día } t \in T. \\ 0 & \text{e.o.c.} \end{cases}$
- $ZI_{c,b,n,t} := \begin{cases} 1 & \text{si el camión } c \in \Gamma \text{ viaja desde el parque } b \in \Theta \text{ al punto } n \in N \text{ el día } t \in T. \\ 0 & \text{e.o.c.} \end{cases}$
- $ZV_{c,n,b,t} := \begin{cases} 1 & \text{si el camión } c \in \Gamma \text{ viaja desde el punto } n \in N \text{ al parque } b \in \Theta \text{ el día } t \in T. \\ 0 & \text{e.o.c.} \end{cases}$

Restricciones [10 pts]

Las restricciones son:

1. Construcción de la variable X.

- No se puede viajar a más de un punto cada día.[1 pto]

$$\sum_{n\in N} x_{c,n,t} \leq 1 \ \forall c \in \Gamma \ \forall t \in T$$

2. Construcción de la variable K.

- No se puede llevar agua si no se viaja ese día. [1 pto]

$$k_{c,n,t} \leq x_{c,n,t} \cdot W \ \forall c \in \Gamma \ \forall n \in N \ \forall t \in T$$

- Se debe cumplir la demanda de manera exacta.[1 pto]

$$\sum_{c\in \Gamma} k_{c,n,t} = d_{n,t} \ \forall n \in N \ \forall t \in T$$

3. Constucción de la variable Y.

- Si desde cualquier punto se viaja a b, entonces se aloja en b. [1 pts]

$$\sum_{n\in N} ZV_{c,n,b,t} \leq y_{c,b,t} \ \forall c \in \Gamma \ \forall b \in \Theta \ \forall t \in T$$

- Todos los días se aloja en algún parque estacionamento y sólo en uno. [1 pts]

$$\sum_{b\in \Theta} y_{c,b,t} = 1 \ \forall c \in \Gamma \ \ \forall t \in T$$

- Condición inicial de Y. Si no se viaja el primer día, entonces se aloja en el parque estacionamiento 1 el día 1. [0.5 pts]

$$1 - \sum_{n\in N} x_{c,n,1} \leq y_{c,1,1} \ \forall c \in \Gamma$$

- Si no se viaja un día, entonces se debe alojar en el mismo parque estacionamiento que el día anterior.[0.5 pts]

$$-\sum_{n\in N} [x_{c,n,t}] + y_{c,b,t-1} \leq y_{c,b,t} \ \forall c \in \Gamma \ \forall b \in \Theta \ \forall t \in T : t \geq 2$$

4. Construcción de ZI.

- Si y solo si se viaja a n, entonces se tendrá que viajar a n desde algún parque. [1 pto]

$$\sum_{b\in \Theta} ZI_{c,b,n,t} = x_{c,n,t} \ \forall n \in N \ \forall c \in \Gamma \ \forall t \in T$$

- El primer día sólo se puede viajar desde b=1. [1 pto]

$$ZI_{c,1,n,1} \leq 1 \ \forall n \in N \ \forall c \in \Gamma$$

$$ZI_{c,b,n,1} = 0 \ \forall n \in N \ \forall b \in \{2, \ldots, |B|\} \ \forall c \in \Gamma$$

- Si se viaja a cualquier n desde b, es porque la noche anterior se alojó en b. Además, sólo se puede viajar a máximo 1 destino. [1 pto]

$$\sum_{n\in N} ZI_{c,b,n,t} \leq y_{c,b,t-1} \ \forall b \in \Theta \ \forall c \in \Gamma \ \forall t \in T : t \geq 2$$

5. Construcción de ZV.

   - Si y solo si se viaja a n, entonces se retorna hacia un y solo un parque estacionamiento.[1 pto]

$$\sum_{b\in\Theta} ZV_{c,n,b,t} = x_{c,n,t} \; \forall c \in \Gamma \; \forall n \in N \;\; \forall t \in T$$

6. Naturaleza de las variables.

$$\begin{aligned}
x_{c,n,t} &\in \{0,1\} \; \forall c \in \Gamma \; \forall n \in N \; \forall t \in T \\
k_{c,n,t} &\geq 0 \quad \forall c \in \Gamma \; \forall n \in N \; \forall t \in T \\
y_{c,b,t} &\in \{0,1\} \; \forall c \in \Gamma \; \forall b \in \Theta \; \forall t \in T \\
ZI_{c,b,n,t} &\in \{0,1\} \; \forall c \in \Gamma \; \forall b \in \Theta \; \forall n \in N \; \forall t \in T \\
ZV_{c,n,b,t} &\in \{0,1\} \; \forall c \in \Gamma \; \forall b \in \Theta \; \forall n \in N \; \forall t \in T
\end{aligned}$$

Función Objetivo [2.5 pts]

La función objetivo es:

- Minimizar costos totales

$$\text{mín} \sum_{t\in T} \sum_{c\in\Gamma} \sum_{b\in\Theta} [\sum_{n\in N} [ZI_{c,b,n,t} \cdot c_{b,n}] + \sum_{n\in N} [ZV_{c,n,b,t} \cdot c_{b,n}] + y_{c,b,t} \cdot CL_b]$$

Pregunta 2 **[20 Puntos]**

La empresa Rayos tiene el monopolio de generación y transmisión de energía eléctrica en el país Demo. Rayos quiere minimizar sus costos, satisfaciendo la demanda en un día típico de 24 horas. A lo largo y ancho del país, la empresa tiene instalado un conjunto N de plantas generadoras. Además, existe un conjunto M de nodos de consumo eléctrico en el país. Para cada hora $h \in H = \{1, ..., 24\}$, existe una demanda de $D_{m,h}$ MWh que debe llegar al nodo de consumo de índice $m \in M$, después de restar las pérdidas en los cables. Cada generadora $n \in N$ decide cuánta energía en MWh envía de manera constante a cada nodo durante cada hora $h \in H$, sin superar su capacidad máxima de producción por hora que es $CN_{n,h}$. Todas las generadoras están apagadas inicialmente, sin poder producir. Sin embargo, si se decide encender la generadora $n \in N$ en la hora $h \in H$, entonces desde esa hora se podrá producir hasta el final del día costando $c_{n,h}$ pesos dicha acción. Existe un conjunto Φ que contiene todos los pares (n, m) tales que existe una conexión única por cable entre la generadora $n \in N$ y el nodo $m \in M$. Para cada conexión, existe una pérdida de la energía enviada por concepto de resistencia normal y resitencia no óhmica en los cables. Esta pérdida de energía es proporcional a la energía enviada, según un coeficiente de pérdida $R_{n,m}$, entre la generadora $n \in N$ y el nodo de consumo $m \in M$. Además, el parámetro $G_{n,m}$ (medido en $\frac{1}{MWh}$) representa un factor que debe multiplicarse por el cuadrado de la energía enviada cada hora para obtener la energía perdida en el cable por a su comportamiento no óhmico. Además, esa conexión solo puede transmitir un máximo de $CL_{n,m}$ MWh cada hora. Si $(n, m) \notin \Phi$, entonces $R_{n,m}$, $G_{n,m}$ y $CL_{n,m}$ deben ser ignorados. El costo medio variable de generar L unidades de energía (MWh) 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 la unidad de medida de $a_{n,h}$ es el $\frac{\$CLP}{MWh}$, y que la de $b_{n,h}$ es $\frac{\$CLP}{(MWh)^2}$. El costo de producir, en una hora de generación, se obtiene multiplicando lo producido por el costo medio variable de producir esa cantidad.

Formule un modelo de optimización no lineal que ayude a la empresa Rayos a minimizar sus costos en total satisfaciendo la demanda de todos los nodos de consumo del país. Escriba claramente las variables de decisión, restricciones y función objetivo del modelo.

Solution:

Variables [5 pts]

Las variables a usar son:

- $x_{n,m,h}$:= Cantidad de KWh que $n \in N$ envía a el nodo de consumo $m \in M$ en la hora $h \in H$.
- $y_{n,h} := \begin{cases} 1 & \text{si la generadora } n \in N \text{ es encendida en la hora } h \in H. \\ 0 & \text{e.o.c.} \end{cases}$

Definición de $x_{n,m,h}$ incluyendo correcta N.V. [2 pts].
Definición de $y_{n,h}$ incluyendo correcta N.V. [3 pts].
Pueden haber otras variables auxiliares definidas, pero no tienen puntaje.

Restricciones [10 pts]

Las restricciones son:

1. Sobre encender las generadoras.[2.5 pts]

- Cada generadora se puede encender máximo 1 vez en el período.

$$\sum_{h \in H} y_{n,h} \leq 1 \; \forall n \in N$$

2. Sobre enviar energía por los cables. [2.5 pts]

- Si una conexión no existe, entonces no se puede enviar energía a través de ella.

$$x_{n,m,h} = 0 \; \forall (n,m) \notin \Phi \; \forall h \in H$$

- Si no se ha prendido una generadora, no puede enviar energía. Además no se puede enviar más energía que la máxima que se puede enviar en cada cable.

$$x_{n,m,h} \leq CL_{n,m} \cdot \sum_{j=1}^{j=h} y_{n,j} \; \forall (n,m) \in \Phi \; \forall h \in H$$

3. Sobre capacidad de las plantas generadoras.[2.5 pts]

- Capacidad máxima de producción de energía por hora en la planta n

$$\sum_{m \in M} x_{n,m,h} \leq CN_{n,h} \; \forall n \in N \; \forall h \in H$$

4. Sobre la satisfacción de la demanda. [2.5 pts]

- A cada punto de consumo le debe llegar la energía que demanda. La suma de las energías enviadas por cada cable menos las pérdidas, deben satisfacer la demanda.

$$\sum_{n \in N} [x_{n,m,h} - R_{n,m} \cdot x_{n,m,h} - G_{n,m} \cdot x^2_{n,m,h}] = D_{m,h} \; \forall m \in M \; \forall h \in H$$

5. Naturaleza de las variables. [0 pts]

$$\begin{aligned} x_{n,m,h} &\geq 0 \; \forall n \in N \; \forall m \in M \; \forall h \in H \\ y_{n,h} &\in \{0,1\} \; \forall n \in N \; \forall h \in H \end{aligned}$$

Función Objetivo [5 pts]

La función objetivo es:

- Minimizar lo producido en cada central por su costo total.

$$\text{mín} \sum_{h \in H} \sum_{n \in N} [\sum_{m \in M} [x_{n,m,h}] \cdot (a_{n,h} - b_{n,h} \cdot \sum_{m \in M} [x_{n,m,h}])] + \sum_{h \in H} \sum_{n \in N} y_{n,h} \cdot c_{n,h}$$

[3 pts] Por el primer término. Si hay un error, máximo [1 pto] en este término.

[2 pts] Por el segundo término. No hay puntaje si hay error.

Pregunta 3 **[20 Puntos]**

1. Implementación computacional

1.1. Descripción del problema

Considere el siguiente problema de optimización:
En la empresa Gallos, se enfrentan al desafío de alimentar a los pollos de la manera más adecuada y económica posible. La idea es combinar diferentes tipos de cereales para satisfacer las necesidades nutricionales de las aves, al mismo tiempo que se minimizan los costos. El veterinario de la empresa ha identificado un conjunto I de nutrientes esenciales, y es crucial que cada nutriente $i \in I$ esté presente en la mezcla final dentro de un rango de proporciones específico. Este rango se define por dos valores: el límite inferior b_i y el límite superior l_i. Estos valores representan porcentajes de masa-masa que oscilan entre 0 y 1. Por ejemplo, si b_i es igual a 0.15, esto significa que al menos el 15 % de la masa total de la mezcla final debe consistir en masa del nutriente $i \in I$. Para cumplir con esos requerimientos se dispone de un conjunto J de diferentes tipos de cereales. Cada cereal $j \in J$ tiene un costo por kilogramo c_j y contiene un porcentaje masa-masa $a_{i,j}$ del nutriente $i \in I$.

1.2. Modelo

Un modelo de optimización lineal que modela esta situación es el siguiente:

Variables:

- $x_j :=$ Proporción (masa-masa) en la que está presente el cereal $j \in J$ en la mezcla final.

Restricciones:

- La mezcla está compuesta únicamente por cereales.
$$\sum_{j \in J} x_j = 1$$
- Se debe cumplir una proporción mínima de nutrientes.
$$\sum_{j \in J} a_{i,j} \cdot x_j \geq b_i \ \forall i \in I$$
- Se debe cumplir una proporción máxima de nutrientes.
$$\sum_{j \in J} a_{i,j} \cdot x_j \leq l_i \ \forall i \in I$$
- Naturaleza de las variables.
$$x_j \geq 0 \ \forall j \in J$$

Función Objetivo:

-
$$\text{mín} \sum_{j \in J} c_j \cdot x_j$$

Lo que se pide a grandes rasgos en esta pregunta es la implementación computacional del modelo de este problema. Se solicita la elaboración de un código de Python en que se utilice la librería Gurobi para resolver el problema. Ese código debe poder recibir cualquier instancia de parámetros, como se explica en el apartado. A continuación se indicarán los elementos técnicos a considerar en esta pregunta.

1.3. 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. Tanto los cereales como los nutrientes serán trabajados de manera abstracta, sólo considerándolos por sus índices. Toda la información de las instancias de datos vendrán en 3 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 con diferentes valores y dimensiones. A continuación se detalla el formato de cada uno de estos archivos.

1.3.1. costos.csv

Este archivo CSV tiene solo 1 columna y $(|J|+1)$ filas. Contiene los costos c_j de cada cereal $j \in J$. En la primera fila, está el encabezado del archivo: costo_por_kg. A partir de la segunda fila en adelante vienen los valores $\frac{\$CLP}{kg}$ de cada uno de los cereales. Por lo tanto, en la segunda fila del archivo está el valor del precio del cereal 1, en la tercera está el precio del cereal 2, y así sucesivamente.

1.3.2. limites.csv

Este archivo CSV tiene 2 columnas e $(|I|+1)$ filas. Contiene los límites inferior b_i y superior l_i para las proporciones en que debe estar el nutriente $i \in I$ en la mezcla. En la primera fila, está el encabezado del archivo: limite_inferior, limite_superior. A partir de la segunda fila en adelante, vienen los límites de cada uno de los nutrientes. Primero viene el límite inferior, y despues de una coma viene el límite superior. Por lo tanto, en la segunda fila del archivo están los límites del nutriente 1, en la tercera están los límites del nutriente 2, y así sucesivamente.

1.3.3. contenidos_nutricionales.csv

Este archivo CSV tiene $|J|$ columnas e $(|I|+1)$ filas. Contiene las fracciones masa-masa $a_{i,j}$ de nutriente $i \in I$ que tiene cada cereal $j \in J$. En la primera fila, está el encabezado de las columnas del archivo: cereal1,cereal2,ceral3,..,cerealJ. En las siguientes filas, se muestran separados por comas, las proporciones masa-masa en que el nutriente está contenido en cada cereal. Por lo tanto, en la segunda fila del archivo están las proporciones del nutriente 1 en cada cereal, en la tercera están las proporciones del nutriente 2 en cada cereal, y así sucesivamente.

1.4. 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 3 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. Además debe asumir que la solución óptima existe.
3. El código debe imprimir en consola de manera clara el valor óptimo y su unidad de medida. Además debe imprimir en consola de manera clara las proporciones óptimas en que se debe incluir cada cereal en la mezcla final de la empresa Gallos.

Solution:
Para esta pregunta se debe evaluar la construcción del código (10 pts) y el correcto funcionamiento del código (10 pts).

A continuación se presenta el puntaje máximo otorgado por la construcción de un código. Si se logra parcialemente se debe asignar la mitad del puntaje en cada ítem.

[3pts] Presentación de código para abrir los 3 archivos .csv. Si se logra abrir los archivos y guardarlos en listas o diccionarios otrorgar máximo puntaje.

[1 pto] Creación del modelo con la línea modelo = gp.Model().

[1 pto] Creación de las variables.

[2 pts] Creación de todas las restricciones.

[1 pto] Creación de la función objetivo y resolver el modelo.

[2 pts] Código para imprimir en consola el valor óptimo y la solución óptima.

[-1 pto] Si es que no se comenta el código.

Aparte de los puntajes, se debe correr el main.py en una carpeta junto con los archivos correspondientes. Si el resultado es el solicitado otorgar [10 puntos] por el correcto funcionamiento del código. En otro caso otorgar [0 puntos]. Si el código corre pero no se imprimen los valores correctos (o no se imprime nada en consola) también otorgar [0 puntos].
La solución óptima para la instancia de datos con que se evalúa es:
$x_1 = 0.8$
$x_2 = 0$
$x_3 = 0$
$x_4 = 0.2$
El valor óptimo es: 0.6 pesos por cada kilogramo.