Análisis Prescriptivo y Optimización
Maestría en Ciencia de Datos
Universidad de San Andrés

Nataly Kraselnik
Florencia Vermeulen
Leonardo Dufour

Trabajo Práctico Final: *"Field Service Management"*

Objetivo

El objetivo del presente trabajo es modelizar la planificación semanal de cuadrillas de trabajo dadas las órdenes a realizar. La idea será maximizar la ganancia total, teniendo en cuenta el beneficio que nos otorga resolver un trabajo y considerando los pagos a los trabajadores.
En los siguientes apartados desarrollaremos en detalle cada uno de los elementos del proceso de modelización y la búsqueda de una solución óptima mediante la herramienta de *CPLEX.*

Modelo

1. Problema

El modelo lo que busca es maximizar la ganancia de la asignación para lo cual se van a tener en cuenta cuatro elementos principales: Trabajador (j) , Orden (n), Día (d) y Turno (t) .
Tanto la cantidad de trabajadores como la cantidad de órdenes, va a depender de las características de la base de datos que queramos probar. En cuanto a los días y turnos disponibles para la asignación de las tareas, ya se encuentran definidos de antemano. En este caso los días disponibles serán 6 en total y contamos con 5 turnos dentro de cada día.
La relación entre estas cuatro entidades es la que va a ser modelizada en el problema. Como output del modelo deberíamos obtener qué órdenes se asignan a qué trabajadores para ser realizados en un día y turno específico, como ya mencionamos, con el fin de maximizar las ganancias en esta asignación.

2. Definición de variables

Las variables que caracterizan al modelo que desarrollamos son las siguientes:

- v_{njdt} *vale 1 si la orden n es hecha en el día d en el turno t por el trabajador j, 0 en caso contrario.*
- γ_n *vale 1 si la orden n se realiza sin importar en cualquier turno y horario.*
- ε_{ndt} *vale 1 si la orden n es hecha en el día d en el turno t, 0 en caso contrario.*
- Z_{jd} *vale 1 si el trabajador j realiza alguna orden en el día d.*
- T_n *cantidad de trabajadores necesarios para hacer la orden n.*
- x_j *cantidad de órdenes que realiza el trabajador j.*
- x_{0j} *vale entre 0 y 5 representando las órdenes realizadas por el trabajador j que se pagan $1.000 cada una.*
- x_{1j} *vale entre 0 y 5 representando las órdenes realizadas por el trabajador j que se pagan $1.200 cada una.*
- x_{2j} *vale entre 0 y 5 representando las órdenes realizadas por el trabajador j que se pagan $1.400 cada una.*
- x_{3j} *vale entre 0 y 30 representando las órdenes realizadas por el trabajador j que se pagan $1.500 cada una. (se elige 30 como máximo ya que es el total de turnos posibles contando el total de días que puede hacerse).*

- w_{0j} *vale 1 cuando el trabajador j alcanza el máximo de órdenes que se pueden realizar en el tramo en el cual se pagan $1.000 cada una.*
- w_{1j} *vale 1 cuando el trabajador j alcanza el máximo de órdenes que se pueden realizar en el tramo en el cual se pagan $1.200 cada una.*
- w_{2j} *vale 1 cuando el trabajador j alcanza el máximo de órdenes que se pueden realizar en el tramo en el cual se pagan $1.400 cada una.*
- w_{3j} *vale 1 cuando el trabajador j alcanza el máximo de órdenes que se pueden realizar en el tramo en el cual se pagan $1500 cada una.*

$\gamma_n,\ v_{njdt},\ \varepsilon_{ndt},\ Z_{jd},\ w_{0j},\ w_{1j},\ w_{2j},\ w_{3j} \quad \in \{0,1\}$

$x_j,\ x_{0j},\ x_{1j},\ x_{2j},\ x_{3j} \quad \in \aleph$

Cada uno de los subíndices está definido como:

j **:** cantidad de trabajadores. $j \in \{0, \ldots, J\}$

n **:** cantidad de órdenes. $n \in \{0, \ldots, N\}$

d **:** cantidad de días laborales. Definido en 6 días. $d \in \{0, \ldots, 5\}$

t **:** cantidad de turnos dentro del día laboral. Definido en 5 turnos. $t \in \{0, \ldots, 4\}$

k **:** cantidad de tramos del salario de los trabajadores. $k \in \{0, 1, 2, 3\}$

b **:** beneficio monetario por la realización de la orden n. $n \in \{0, 1, \ldots, N\}$.

3. Función objetivo

El problema consiste en encontrar la asignación óptima de trabajadores a horarios, días y órdenes, que cumplan las restricciones, y maximice el beneficio.

El beneficio está dado por el ingreso que genera realizar cada orden, menos los costos asociados que implica realizar esa orden, es decir, el sueldo de los trabajadores que se necesitan para cumplirla.

A su vez el sueldo de los trabajadores es escalado de acuerdo a la cantidad total de órdenes que llevan realizadas y se define con los siguientes tramos:

- Si realizan entre 0 y 5 órdenes: Obtienen una remuneración de $1.000 por cada orden.
- Si realizan entre 6 y 10 órdenes: Obtienen una remuneración de $1.200 por cada orden.
- Si realizan entre 11 y 15 órdenes: Obtienen una remuneración de $1.400 por cada orden.
- Si realizan más de 15 órdenes: Obtienen una remuneración de $1.500 por cada orden.

Por lo tanto, la función objetivo queda definida de la siguiente manera:

$$max \sum_{\forall n} \gamma_n * b_n - \left(\sum_{\forall j} x_{kj} * \$1.000 + \sum_{\forall j} x_{kj} * \$1.200 + \sum_{\forall j} x_{kj} * \$1.400 + \sum_{\forall j} x_{kj} * \$1.500\right)$$

$con\ k \in \{0, 1, 2, 3\},\ n \in \{0, \ldots, N\},\ j \in \{0, \ldots, J\}$

.

$siendo\ N\ la\ cantidad\ total\ de\ órdenes\ y\ J\ la\ cantidad\ total\ de\ trabajadores\ disponibles.$

4. Restricciones

- Una orden se puede hacer en un único turno y día.

$$\gamma_n = \sum_{\forall d}\sum_{\forall t} \varepsilon_{ndt} \quad \forall\, t\, \forall\, d$$

- No se pueden realizar varias órdenes en un mismo turno si comparten trabajadores.

$$\sum_{\forall n} v_{njdt} \leq 1 \quad \forall\, t\, \forall\, d\, \forall\, j$$

$$\sum_{\forall j} v_{njdt} = \sum_{\forall n} \varepsilon_{ndt} * T_n \quad \forall\, t\, \forall\, d\, \forall\, n$$

- Una orden de trabajo debe tener asignada sus T_n trabajadores en un mismo turno para poder ser resuelta.

$$\sum_{\forall j}\sum_{\forall t}\sum_{\forall d} v_{njdt} = \gamma_n * T_n \quad \forall\, n$$

- Ningún trabajador puede trabajar los 6 días de la semana.

$$\sum_{\forall t}\sum_{\forall n} v_{tnjd} \leq 5 * z_{jd} \quad \forall j$$

$$Z_{jd} \in \{0,1\}\, \forall j \forall d$$

$$\sum_{\forall d} z_{jd} \leq 5 \quad \forall j$$

- Ningún trabajador puede trabajar los 5 turnos de un día.

$$\sum_{\forall n}\sum_{\forall t} v_{tnjd} \leq 4 \quad \forall d\, \forall j$$

- Existen algunos pares de órdenes de trabajo correlativas.

$\varepsilon_{n\,dt} = \varepsilon_{(n+1)d(t+1)}$ $\forall d\, \forall\, t$ *donde n y n+1 son ordenes correlativas que deben ser satisfechas en turnos consecutivos.*

Adicionalmente, para que se cumpla la restricción de correlatividad incorporamos dos restricciones:

- La orden correlativa número 1 no se puede realizar el último turno de un día.

$\varepsilon_{n\,d4} = 0 \;\forall d$ *donde n es la primera orden correlativa que debe ser satisfecha en turnos consecutivos.*

- La orden correlativa número 2 no se puede realizar en el primer turno del día.

$\varepsilon_{n+1\,d0} = 0 \;\forall d$ *donde n+1 es la segunda orden correlativa que debe ser satisfechas en turnos consecutivos a n.*

- Hay pares de órdenes de trabajo que no pueden ser satisfechas en turnos consecutivos de un mismo trabajador.

$$v_{njdt} - v_{n+1jdt+1} \leq 1 \quad \forall d \,\forall t \;\forall j$$

$$v_{njdt} - v_{n+1jdt-1} \leq 1 \;\forall d \,\forall t \;\forall j$$

- La diferencia entre el trabajador con más órdenes asignadas y el trabajador con menos órdenes no puede ser mayor a 10.

$$-10 \leq x_j - x_g \leq 10 \quad \forall\, j, g \in \{1,\ldots,J\},\; j \neq g$$

- Los trabajadores son remunerados según la cantidad de órdenes asignadas.

$$\sum_{\forall n} v_{tnjd} = x_j \qquad \forall j \,\forall d \,\forall t$$

$$x_j = x_{1j} + x_{2j} + x_{3j} + x_{4j} \quad \forall j$$

$$5 * w_{1j} \leq x_{1j} \leq 5 \quad \forall j$$

$$(10 - 5) * w_{2j} \leq x_{2j} \leq (10 - 5) * w_{1j} \quad \forall j$$

$$(15 - 10) * w_{3j} \leq x_{3j} \leq (15 - 10) * w_{2j} \quad \forall j$$

$$0 \leq x_{4j} \leq (30 - 15) * w_{3j} \quad \forall j$$

$$w_{2j} \leq w_{1j} \ \forall j$$
$$w_{3j} \leq w_{2j} \quad \forall j$$
$$w_{4j} \leq w_{3j} \ \forall j$$

$$w_{1j}, w_{2j}, w_{3j}, w_{4j} \quad \in \{0, 1\} \ \forall j$$

Restricciones deseables

A continuación se presentan las restricciones deseables del problema, que se suman a las restricciones anteriores:

- Hay conflictos entre algunos trabajadores que hacen que prefieran no ser asignados a una misma orden de trabajo.

$$\sum_{\forall d}\sum_{\forall t} v_{tnjd} + \sum_{\forall d}\sum_{\forall t} v_{tnj-1d} \leq 1 \quad \forall n$$

$siendo\ j\ y\ j - 1\ trabajadores\ que\ prefieren\ no\ trabajar\ juntos.$

- Hay pares de órdenes de trabajo que son repetitivas por lo que sería bueno que un mismo trabajador no sea asignado a ambas.

$$\sum_{\forall d}\sum_{\forall t} v_{tnjd} + \sum_{\forall d}\sum_{\forall t} v_{tn-1jd} \leq 1 \ \forall j \ \ siendo\ n\ y\ n - 1\ las\ órdenes\ que\ son\ repetitivas.$$

Implementación

La implementación del modelo se realizó mediante la API de Python de la herramienta *CPLEX.*

El código donde se realiza la creación de las variables y restricciones está disponible en el siguiente repositorio GitHub.

Experimentación

Para la experimentación probamos nuestro modelo en dos bases de datos, que plantean problemas diferentes:

- Menos cantidad de órdenes que la capacidad que tienen los trabajadores.
- Mayor cantidad de órdenes que la capacidad que tienen los trabajadores.

La idea es ver en ambos casos como el modelo asigna de la manera más eficiente los recursos priorizando obtener la mayor ganancia posible, dado los trabajadores disponibles que tiene y sus costos asociados.

- **Experimentación 1**: *Menos cantidad de órdenes que la capacidad que tienen los trabajadores*

La base de datos que construimos para testear la primera experimentación tiene las siguientes características:

- Cantidad de trabajadores: 10
- Cantidad de órdenes: 100
- Cantidad de conflicto entre trabajadores: 2
- Cantidad de órdenes correlativas: 5
- Cantidad de órdenes conflictivas: 4
- Cantidad de órdenes repetitivas: 2

Dadas las restricciones del modelo, la máxima capacidad que tendrían 10 trabajadores teniendo en cuenta los 5 días y 4 turnos disponibles de cada uno, sería en total 200 órdenes. Sin embargo, cada orden puede requerir de más de un trabajador por lo que es posible que no todas logren realizarse. Por ejemplo, hay 8 órdenes en la base de datos que dado la cantidad de trabajadores que necesitan y teniendo en cuenta el salario base (primer tramo de $1.000), al hacerlas incurrimos en pérdida por lo que el modelo no debería elegir realizarlas.
Para el armado de la base de datos, el beneficio fue definido aleatoriamente entre $2.000 y $5.000 pesos. La cantidad de trabajadores necesarios también fue definida aleatoriamente entre 1 y 3 trabajadores.
Con el mismo proceso se definieron la cantidad de conflictos entre trabajadores, órdenes correlativas, órdenes conflictivas y órdenes repetitivas, definidas por un número entre 1 y 10, y cada una de las órdenes implicadas también sampleadas aleatoriamente.

Resultados

Ganancia final: $134.095

Input órdenes	100
Órdenes realizadas	70
% órdenes realizadas	70,00%

Trabajadores Disponibles	10
Turnos Disponibles	200
Turnos Utilizados	109
% de turnos utilizados	54,50%

En este caso, tal como mencionamos anteriormente, no todos los turnos posibles fueron utilizados. Eso se debe a que el modelo no tiene necesidad de realizar una orden si la ganancia por completarla no es positiva. Como mencionamos, de partida ya existen 8 órdenes que tomando el salario básico, no deberían realizarse. A este conjunto se le pueden sumar más casos dado que el salario de los trabajadores es escalonado y depende de la cantidad de órdenes que llevan realizadas.
Por lo tanto, aquellas órdenes que requieran un costo mayor a su beneficio, no fueron realizadas. Esto se podría solucionar si se incorporan restricciones que penalicen la ociosidad de los empleados.

A su vez, comprobamos que las restricciones tanto originales como las deseables se cumplen en la solución final. Para más detalle, dejamos los resultados de la optimización de *CPLEX* en el siguiente repositorio.

- **Experimentación 2**: *Mayor cantidad de órdenes que la capacidad que tienen los trabajadores*

La base de datos que construimos para testear la segunda experimentación tiene las siguientes características:

- Cantidad de trabajadores: 10
- Cantidad de órdenes: 300
- Cantidad de conflicto entre trabajadores: 2
- Cantidad de órdenes correlativas: 5
- Cantidad de órdenes conflictivas: 4
- Cantidad de órdenes repetitivas: 2

En este caso, la base de datos cuenta con muchas más órdenes que la capacidad que tendrían los trabajadores para realizarlas (200 órdenes como máximo). Por lo tanto, deberíamos ver que se elijan las órdenes que tengan un beneficio positivo y se realice la mayor cantidad posible.
En cuanto a las características de la cantidad de conflictos entre trabajadores, órdenes correlativas, órdenes conflictivas y órdenes repetitivas, se mantienen las mismas características que en la base de datos del apartado anterior.

Resultados

Ganancia final: $298.480

Input órdenes	300
Órdenes realizadas	150
% realizado	50,00%

Trabajadores Disponibles	10
Turnos Disponibles	200
Turnos Utilizados	200
% de turnos utilizados	100,00%

Lo que podemos ver en este caso es que los trabajadores utilizan todos los turnos disponibles para realizar las órdenes. Las 150 órdenes que son realizadas demandan de un total de 200 trabajadores y es por este motivo que no se realiza mayor cantidad de las mismas.

En este caso también, comprobamos que las restricciones tanto originales como las deseables se cumplen en la solución final. Para más detalle, dejamos los resultados de la optimización de *CPLEX* en el siguiente repositorio.