investigacion operativa

Edison Achalma

1/8/23

Tabla de contenidos

Preface

This is a Quarto book.

To learn more about Quarto books visit https://quarto.org/docs/books.

```
1 + 1
```

```
[1] 2
```

Introduction

This is a book created from markdown and executable code.

See Knuth [1] for additional discussion of literate programming.

La investigación operativa es una disciplina que se enfoca en el análisis y resolución de problemas de optimización en diferentes ámbitos, tales como la empresa, la industria y el gobierno. Esta disciplina se basa en el uso de métodos matemáticos y estadísticos para encontrar la solución óptima a un problema específico, teniendo en cuenta diferentes factores y restricciones.

Para aprender investigación operativa, es importante tener conocimientos básicos de matemáticas, especialmente en áreas como álgebra y cálculo. También es útil conocer algoritmos y estructuras de datos, así como tener habilidades de programación para poder implementar y probar soluciones.

El proceso de aprendizaje de investigación operativa suele incluir la exposición a diferentes tipos de problemas y el análisis de cómo se pueden resolver de manera óptima. También se pueden estudiar métodos y técnicas específicas, como programación lineal, programación no lineal, teoría de colas y teoría de juegos.

Además de los conocimientos teóricos, es importante desarrollar habilidades prácticas en la resolución de problemas, lo que incluye la identificación de las variables y restricciones relevantes, la formulación de un modelo matemático y la implementación de una solución. La investigación operativa también implica la toma de decisiones y la evaluación de diferentes escenarios y alternativas, por lo que es importante desarrollar habilidades de análisis y pensamiento crítico.

Summary

In summary, this book has no content whatsoever.

```
1 + 1
```

```
[1] 2
```

Parte I

Capítulos

1 Introducción a la investigación operativa

Definición 1.1 (Investigacion de operaciones). La Investigación de Operaciones es una disciplina que se ocupa del estudio y análisis de sistemas de producción y servicios. Su objetivo es encontrar las mejores soluciones posibles a los problemas que surgen en la toma de decisiones en dichos sistemas. La disciplina se originó durante la Segunda Guerra Mundial, cuando se necesitaba optimizar la producción de material bélico y mejorar la eficiencia logística. Desde entonces, ha tenido un papel importante en la mejora de la productividad y la eficiencia en diversos campos, como la manufactura, la ingeniería, la logística, la finanzas, la salud y la educación, entre otros:

La Investigación de Operaciones (IO) es un campo de la matemática y la informática que se dedica a la resolución de problemas prácticos mediante el uso de modelos matemáticos y algoritmos. Estos problemas pueden ser de diversa índole, tales como la optimización de recursos en una empresa, el diseño de un plan de producción eficiente, la gestión de inventarios, entre otros. La IO utiliza técnicas formales para analizar y resolver estos problemas de manera óptima y eficiente.

1.1 Técnicas y aplicaciones de la IO

La Investigación de operaciones (IO) es una disciplina que se ocupa del estudio y análisis de sistemas complejos y procesos de decisión en diversos contextos, con el objetivo de mejorar su eficiencia y efectividad. Para ello, se utilizan diversas técnicas y herramientas matemáticas y estadísticas. Algunas de las técnicas más comunes en la IO son:

- Programación lineal: permite optimizar una función objetivo, sujeta a un conjunto de restricciones, mediante el uso de técnicas de optimización matemática.
- Análisis de transporte: se utiliza para determinar la forma más eficiente de mover un producto o material desde un origen hasta un destino, minimizando costos de transporte.
- Programación dinámica: permite modelar y resolver problemas que involucran decisiones a lo largo del tiempo, considerando factores como el costo y la incertidumbre.
- Simulación: es una técnica que permite evaluar el comportamiento de un sistema o proceso a través de la repetición de escenarios hipotéticos.

- Teoría de colas: se utiliza para modelar y analizar sistemas de atención y espera, como por ejemplo, una fila de personas esperando ser atendidas en una institución bancaria.

Algunas de las aplicaciones de la IO son:

- Diseño y gestión de sistemas de producción y distribución.
- Diseño y optimización de sistemas de transporte y logística.
- Gestión de inventarios y suministros.
- Diseño y gestión de sistemas de atención y servicio al cliente.
- Gestión de proyectos y programas.
- Diseño y optimización de sistemas de información y tecnología.

1.2 Fundamentos de álgebra matricial

Los fundamentos de álgebra matricial son un conjunto de conceptos y herramientas matemáticas que se utilizan para trabajar con matrices, es decir, arreglos de números dispuestos en filas y columnas. Algunos de los conceptos fundamentales de álgebra matricial son:

- Suma y resta de matrices: para sumar o restar dos matrices, deben tener la misma dimensión (es decir, el mismo número de filas y columnas). Se realiza elemento por elemento, es decir, se suman o restan los elementos correspondientes de cada matriz.

- Multiplicación de una matriz por un escalar: para multiplicar una matriz por un escalar (un número), se multiplican todos los elementos de la matriz por ese escalar.

- Multiplicación de matrices: para multiplicar dos matrices, deben cumplirse ciertas condiciones. La matriz resultante tendrá tantas filas como la primera matriz y tantas columnas como la segunda. El elemento ij-ésimo de la matriz resultante se obtiene multiplicando cada elemento de la i-ésima fila de la primera matriz por el elemento correspondiente de la j-ésima columna de la segunda matriz, y sumando todos esos productos.

- Transpuesta de una matriz: la transpuesta de una matriz A es una matriz A^T que se obtiene intercambiando filas por columnas en A.

- Matriz inversa: una matriz inversa es una matriz A^(-1) que, al multiplicarla por la matriz A original, da como resultado la matriz identidad.

- Determinante de una matriz: el determinante de una matriz es un número que se puede calcular a partir de los elementos de la matriz y que tiene ciertas propiedades. Por ejemplo, si el determinante de una matriz es cero, la matriz no tiene inversa.

1.3 Matriz inversa según el método de Gauss

Para obtener la matriz inversa de una matriz A mediante el método de Gauss, se puede seguir el siguiente proceso:

1. Escribir la matriz A junto con la matriz identidad (I) de tamaño igual a la matriz A. Esto se conoce como la matriz aumentada (A|I).
2. Aplicar las operaciones elementales necesarias para convertir la matriz A en la matriz diagonal (en la que sólo hay elementos distintos de cero en la diagonal principal). Estas operaciones deben aplicarse también a la matriz identidad para mantener la relación entre ambas.
3. Una vez que se ha obtenido la matriz diagonal, se puede obtener la matriz inversa aplicando las mismas operaciones elementales a la matriz identidad, pero en orden inverso y con los coeficientes cambiados de signo.

```r
# Para calcular la matriz inversa de una matriz A en R, puedes utilizar la
# función solve() de la siguiente manera:
# Primero, debes cargar la matriz A

A <- matrix(c(1, 2, 3, 4), nrow = 2, ncol = 2)
# A
# Luego, puedes calcular la inversa de A utilizando solve()

A_inv <- solve(A)
# A_inv

# También puedes utilizar la función ginv() del paquete MASS, que utiliza el
# método de Moore-Penrose para calcular la inversa generalizada de una matriz:

# Primero, debes cargar la matriz A y el paquete MASS

A <- matrix(c(1, 2, 3, 4), nrow = 2, ncol = 2)
library(MASS)

# Luego, puedes calcular la inversa de A utilizando ginv()
A_inv <- ginv(A)
```

Ejemplo 1.1. Para obtener la matriz inversa de A.

$$A_{2\times 2} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}_{2\times 2}$$

Se puede seguir el siguiente proceso:

La matriz aumentada es A|I =

$$A_{2\times 2} = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 3 & 4 & 0 & 1 \end{pmatrix}_{2\times 2}$$

Aplicar una operación elemental de intercambio de filas para convertir el primer elemento de la matriz A en 1:

$$A_{2\times 2} = \begin{pmatrix} 3 & 4 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{pmatrix}_{2\times 2}$$

Aplicar una operación elemental de multiplicación de fila por un escalar para que el primer elemento de la segunda fila sea -1:

$$A_{2\times 2} = \begin{pmatrix} 3 & 4 & 0 & 1 \\ -1 & -2 & -1 & 0 \end{pmatrix}_{2\times 2}$$

Aplicar una operación elemental de suma de filas para eliminar el primer elemento de la segunda fila:

$$A_{2\times 2} = \begin{pmatrix} 3 & 4 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{2\times 2}$$

Aplicar las operaciones elementales necesarias a la matriz identidad para obtener la matriz inversa:

$$A_{2\times 2} = \begin{pmatrix} 1/3 & -2/3 & 0 & -1/3 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{2\times 2}$$

Por tanto, la matriz inversa de A es:

$$A_{2\times 2} = \begin{pmatrix} 1/3 & -2/3 \\ 0 & 1 \end{pmatrix}_{2\times 2}$$

2 Optmización lineal contínua

La optimización lineal contínua es una técnica matemática que se utiliza para encontrar el valor óptimo de una función lineal sujeta a ciertas restricciones o condiciones. Esta técnica se aplica a problemas que involucran variables que pueden tomar cualquier valor en un rango específico, en lugar de solo valores discretos. Los problemas de optimización lineal contínua se pueden resolver utilizando diferentes métodos, como el método simplex o el algoritmo de punto interior. La optimización lineal contínua se utiliza en diversas áreas, como la ingeniería, la economía y la ciencia de la computación.

2.1 Modelo general de Programación Lineal (PL)

La Programación Lineal (PL) es una técnica de optimización que se utiliza para encontrar el valor óptimo de una función de varias variables (llamada función objetivo) sujeta a un conjunto de restricciones. En el modelo general de PL, la función objetivo y las restricciones están expresadas mediante ecuaciones o inecuaciones que involucran una combinación lineal de las variables. La solución del problema de PL consiste en encontrar los valores de las variables que optimizan la función objetivo, sujeto a cumplir con las restricciones.

El modelo general de Programación Lineal (PL) puede ser representado de la siguiente manera en LaTeX:

2.2 Variables de decisión

2.3 Función Objetivo

2.4 Restricciones

2.5 Formulación de modelos de PL

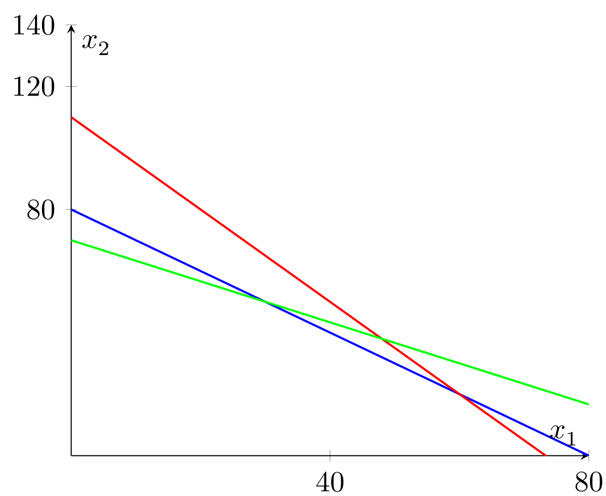
2.6 Algunos casos de estudios clásicos

2.7 Método Gráfico

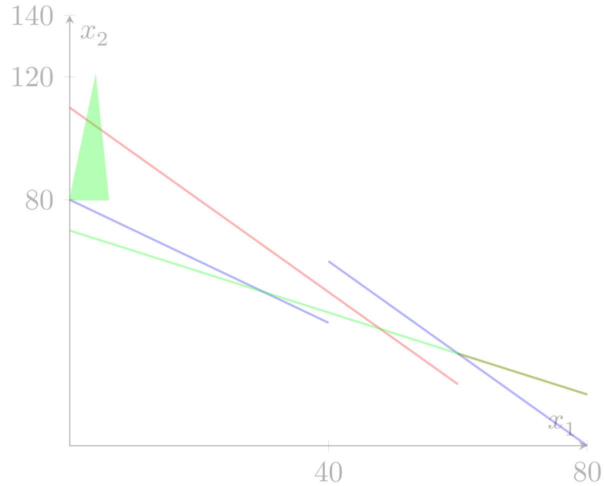
problema en forma estándar

$$
\begin{aligned}
&\text{Maximizar Z} && 50x_1 + 56x_2 && (2.1)\\
&\text{Sujeto a} && x_1 + x_2 \leq 80 && (2.2)\\
& && 3x_1 + 2x_2 \leq 220 && (2.3)\\
& && 2x_1 + 3x_2 \leq 210 && (2.4)\\
& && x_1, x_2 \geq 0 && (2.5)
\end{aligned}
$$

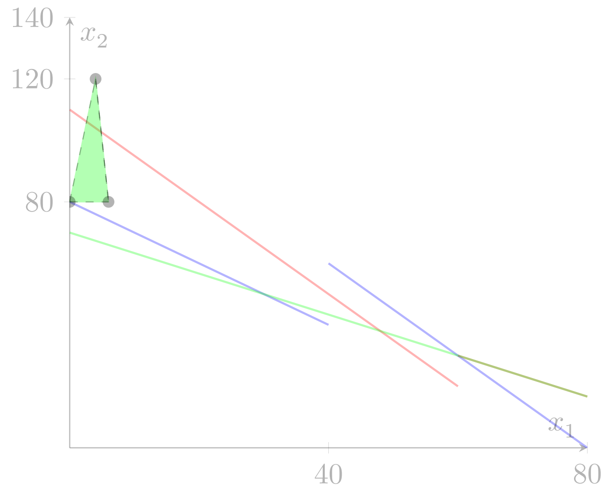
Representar gráficamente las restricciones



Identificar el conjunto solución factible.



Identificar los puntos extremos del conjunto solución factible.



2.7.1 Casos Especiales del Método Gráfico

2.8 Ejercicios de aplicación

3 Problemas de transporte y de asignación

3.1 Método Simplex

3.1.1 Procedimiento del Método simplex

3.1.2 pasos para aplicar método simplex

3.1.3 tablero Simplex

3.1.4 formas de presentación de un programa lineal

3.1.5 forma canónica y estándar

3.1.6 variables de exceso (Surplus) y Holgura (Slack)

3.1.7 aplicaciones con Excel y lindo

3.2 Soluciones iniciales Método de dos fases y Coeficiente de castigo

3.2.1 Rompimiento de empates en el método simplex

3.2.2 Otros métodos de solución

3.3 Análisis de la solución óptima

3.3.1 Definición análisis de sensibilidad

3.3.2 Definición del modelo dual

3.3.3 Procedimiento de Conversión Primal- Dual

3.3.4 relaciones del Primal-Dual

3.3.5 Interpretación económica del dual

3.3.6 Cambios en los coeficientes de la función objetivo.

3.3.7 Cambios en los lados derechos de las restricciones

3.3.8 Ejercicios de aplicación

3.3.9 Aplicaciones con Excel y lindo

4 Optimizacion lineal entera

5 Optimizacion no lineal continua

6 Problemas de inventario

7 Teoria de colas o fenomenos de espera

7.1 Acknowledgments

I am grateful for the insightful comments offered by the anonymous peer reviewers at Books & Texts. The generosity and expertise of one and all have improved this study in innumerable ways and saved me from many errors; those that inevitably remain are entirely my own responsibility.

References

apendice

[1] Donald E. Knuth. “Literate Programming”. En: Comput. J. 27.2 (mayo de 1984), págs. 97-111. ISSN: 0010-4620. DOI: 10 . 1093 / comjnl / 27 . 2 . 97[1]. URL: https://doi.org/10.1093/comjnl/27.2.97.

[1] https://doi.org/10.1093/comjnl/27.2.97