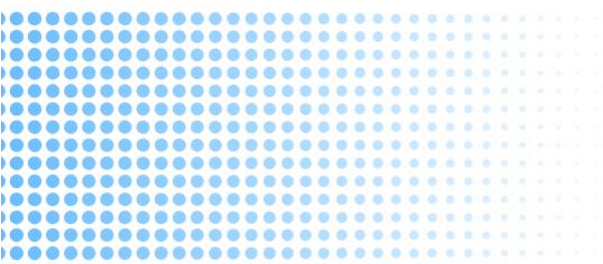


ANÁLISIS PRESCRIPTIVO **TAREA 8**



Andrei Alexandru Miu

Índice

Parte 1: Optimización Matemática (5 puntos)	3
Parte 2: Algoritmos Genéticos (5 puntos)	5

Práctica: Resolviendo Problemas con Modelos de Optimización

Objetivo

Aplicar técnicas de optimización matemática y algoritmos evolutivos para resolver problemas de decisión en diferentes contextos.

Parte 1: Optimización Matemática (5 puntos)

Situación: Una fábrica produce dos tipos de productos (A y B) y desea maximizar sus ganancias. Se tienen las siguientes condiciones:

Cada unidad de A genera 30€ de beneficio y cada unidad de B genera 50€.

La fábrica tiene disponibles 600 horas de trabajo y 500 kg de materia prima.

Producir una unidad de A requiere 5 horas y 4 kg de materia prima.

Producir una unidad de B requiere 8 horas y 6 kg de materia prima.

La producción total de A y B no puede superar 150 unidades por cuestiones de almacenamiento.

Formular el problema como un modelo de optimización lineal:

Definir las variables de decisión.

Las variables que he definido son:

- x: Número de unidades producidas del producto A.
- y: Número de unidades producidas del producto B.

Plantear la función objetivo.

Si queremos maximizar las ganancias, la función objetivo debería ser:

$$Z = 30x + 50y$$

Escribir las restricciones matemáticas.

Las restricciones que he podido encontrar son:

- Restricción de tiempo de trabajo: $5x + 8y \leq 600$
- Restricción de materia prima: $4x + 6y \leq 500$
- Restricción de almacenamiento: $x + y \leq 150$
- Restricciones de no negatividad: $x \geq 0$, $y \geq 0$

Representar gráficamente la región factible (opcional, si usas un software como Excel o GeoGebra).

A	B	C	D	E	F	G	H
Restricción		x (Producto A)	y (Producto B)	Descripción			
$5x + 8y \leq 600$ (Horas de trabajo)		0	75	Intersección (0,75)			
$4x + 6y \leq 500$ (Materia prima)		100	50	Intersección (100,50)			
$x + y \leq 150$ (Almacenamiento)		0	0	Origen (0,0)			
$x \geq 0, y \geq 0$ (No negatividad)							
Función Objetivo							
$Z = 30x + 50y$ (Maximizar ganancias)							

y (Producto B) frente a x (Producto A)

Determinar la solución óptima usando el método gráfico o, si prefieres, el algoritmo de Simplex.

La solución óptima es producir

- 0 unidades de A
- 75 unidades de B

Generando así una ganancia máxima de 3.750€

```
SistemasDeBigData > Prácticas > Tema8 > a.py > ...
1  import numpy as np
2  import scipy.optimize as opt
3
4  # Coeficientes de la función objetivo (Maximizar Z = 30x + 50y)
5  c = [-30, -50] # Se coloca en negativo porque scipy.optimize.minimize minimiza por defecto
6
7  # Coeficientes de las restricciones (Ax ≤ b)
8  A = [[5, 8], # Restricción de horas de trabajo
9       [4, 6], # Restricción de materia prima
10      [1, 1]] # Restricción de almacenamiento
11
12  b = [600, 500, 150] # Lado derecho de las restricciones
13
14  # Límites de las variables (x, y ≥ 0)
15  x_bounds = (0, None)
16  y_bounds = (0, None)
17
18  # Resolver el problema de optimización
19  res = opt.linprog(c, A_ub=A, b_ub=b, bounds=[x_bounds, y_bounds], method='highs')
20
21  # Obtener la solución óptima
22  res.x, -res.fun # Se invierte el signo de la función objetivo para obtener la ganancia máxima
23  print("Solución óptima:")
24  print("x =", res.x[0])
25  print("y =", res.x[1])
26  print("Ganancia máxima:", -res.fun)
```

PROBLEMAS SALIDA PUERTOS CODE REFERENCE LOG GIT COMENTARIOS GITLENS TERMINAL CONSOLA DE DEPURACIÓN

```
PS D:\WorkSpace_VSCode_IABD\SistemasDeBigData\Prácticas\Tema8> python .\a.py
Solución óptima:
x = 0.0
y = 75.0
Ganancia máxima: 3750.0
PS D:\WorkSpace_VSCode_IABD\SistemasDeBigData\Prácticas\Tema8> 
```

Aquí dejo el código hecho en Python, con el resultado.

Parte 2: Algoritmos Genéticos (5 puntos)

Situación: Un viajero desea visitar 6 ciudades minimizando la distancia recorrida. Las distancias entre ciudades están dadas en la siguiente matriz:

	C1	C2	C3	C4	C5	C6
C1	0	12	10	19	8	16
C2	12	0	3	7	2	4
C3	10	3	0	6	20	5
C4	19	7	6	0	9	12
C5	8	2	20	9	0	10
C6	16	4	5	12	10	0

Definir el problema como un modelo de optimización usando un algoritmo genético.

Un viajero debe visitar 6 ciudades exactamente una vez y regresar a la ciudad de origen, minimizando la distancia total recorrida.

Dado que el número de ciudades es pequeño, los algoritmos genéticos pueden ser una buena alternativa para encontrar una solución aproximada en un tiempo razonable.

Explicar cómo representar la solución (genotipo, cromosomas y genes).

- **Genotipo:** Es una secuencia que representa el orden en que se visitan las ciudades.
- **Cromosomas:** Cada cromosoma representa una ruta completa que el viajero tomará.
- **Genes:** Cada gen en el cromosoma representa una ciudad.

Describir los operadores genéticos a usar en la solución (selección, cruce, mutación).

Selección: Se usa una estrategia como torneo o ruleta:

- Se evalúa la distancia total recorrida de cada cromosoma.
- Se eligen los mejores individuos (rutas más cortas) para reproducirse.

Cruce: Dado que el orden importa, se usa el cruce por orden

- Se selecciona un segmento de la ruta del padre 1.
- Se rellenan los valores faltantes con el orden en que aparecen en el padre 2, sin repetir ciudades.

Mutación: Para evitar caer en óptimos locales, usamos mutación por intercambio.

- Se seleccionan dos ciudades al azar y se intercambian de posición

Indicar qué criterios usarías para determinar la mejor solución y cuándo detener el algoritmo.

Se mide la distancia total recorrida por cada ruta, cuanto menor sea la distancia, mejor es la solución.

El algoritmo se detendrá cuando:

- Se alcance un número máximo de iteraciones.
- La mejor solución encontrada no mejore en varias generaciones consecutivas.
- Se alcance una distancia mínima predeterminada.