جامعة شعيب الدكالي
Université Chouaib Doukkali

Royaume du Maroc
Ministère de l'Education Nationale, de la Formation Professionnelle
de l'Enseignement Supérieur et de la Recherche Scientifique

ENSA
المدرسة الوطنية للعلوم التطبيقية بالجديدة

Université Chouaib Doukkali,
Ecole Nationale des Sciences Appliquées d'El Jadida (ENSA)
2022-2023

PROGRAMMATION LINÉAIRE : SIMPLEXE

Realized by : Mohamed OUHAMI, Sanae OULQAID
Directed by : Pr.Abdelilah JRAIFI

But:

-Réalisation d'un script pour résoudre des problèmes de programmation linéaire en utilisant la méthode de simplexe

Sommaire :

Exercice 1 : Gestion de production d'une entreprise

Une entreprise peut fabriquer un même bien selon trois techniques différentes de production utilisant les services d'une même machine et de la main d'œuvre. Produire une unité de bien nécessite :

- 0,5 heure de machine et 2 heures de main d'œuvre avec la première technique ;
- 1,5 heure de machine et 1,5 heure de main d'œuvre pour la deuxième technique ;
- 2 heures de machine et 0,5 heure de main d'œuvre pour la troisième technique.

On suppose que la capacité d'usinage de la machine est de 12 heures et que le nombre d'heures de travail disponibles est de 15 heures. Les profits unitaires sont de 3 dirhams, 4 dirhams et 5 dirhams selon que le bien est fabriqué à l'aide de la première, deuxième ou troisième technique.

- **La modelisation du probleme peut etre resume dans le système suivant :**

$$Max\ z = 24x_1 + 16x_2$$
$$\begin{cases} 1x_1 + 2x_2 \leq 150 \\ 4x_1 + 2x_2 \leq 15 \\ x_1 \geq 0,\ x_2 \geq 0 \end{cases}$$

On utilisera le module de scipy et la méthode de "simplexe" pour résoudre ce système

```python
import numpy as np
from scipy.optimize import linprog

# Vecteur des coeffcients
C = np.array([-3,-4,-5])

# Matrice des contraitnes
A = np.array([
    [0.5,1.5,2],
    [2,1.5,0.5],
    [0,0,0]
]
)

# Second membre
B = np.array([12,15,0])

# Solve the actual problem

X =linprog(C,A_ub = A,b_ub = B, method='highs')

print(f"""
La combinaison la plus optimale est de vendre {X.x[0]} vestes avec technique 1,
{X.x[1]} vestes avec technique 2
 et {X.x[2]} vestes avec technique 3""")
print(f"Ceci va genere {-X.fun} Dhs")
```

Ca nous donne comme résultat :

```
La combinaison la plus optimale est de vendre 6.4 vestes avec technique 1,
0.0 vestes avec technique 2
 et 4.4 vestes avec technique 3
Ceci va genere 41.2 Dhs
```

Exercice 2 : Gestion de fabrication des chemises

Un atelier de confection fabrique en série deux modèles de chemises. Une chemise du premier modèle nécessite 1 mètre de tissu, 4 heures de travail et rapporte. 24.-dirhams Une chemise du deuxième modèle exige 2 mètres de tissu, 2 heures de travail et rappo- -rte. 16-dirhams. Sachant que l'atelier dispose quotidiennement de 150 mètres de tissu et de 400 heures de travail, et qu'il peut vendre toute sa fabrication, combien de lots de 10 chemises de chaque modèle faut-il fabriquer pour obtenir un bénéfice maximal?

- **La modelisation du probleme peut etre resume dans le système suivant :**

$$Max\ z = 24x_1 + 16x_2$$

$$\begin{cases} 10x_1 + 20x_2 \leq 1500 \\ 40x_1 + 20x_2 \leq 4000 \\ x_1 \geq 0\ ,\ x_2 \geq 0 \end{cases}$$

- On peut resoudre ce probleme en utilisant le module de Scipy et méthode de linearprog

```python
import numpy as np
from scipy.optimize import linprog
import math

# Vecteur des coeffcients
C = np.array([-24,-16])
C_max = C*-1

# Matrice des contraitnes
A = np.array([
    [1,2],
    [4,2]
    ]
)

# Second membre
B = np.array([150,400])

# Solve the actual problem

X = linprog(C,A_ub = A,b_ub = B, method='highs')
result = X.x

# Nombre de lots pour chaque modele :
model_1 = math.floor(result[0]/10)
model_2 = math.floor(result[1]/10)
result = np.array([model_1,model_2])
profit = np.dot(result,C_max)
print(f"La combinaison la plus optimale est de vendre {model_1} 'lots de 10' du modele 1 et {model_2} 'lots de 10' du modele 2")
print(f"Ceci va genere {profit * 10} Dhs")
```

En donnant une configuration plus réaliste, en travaillant avec des lots de 10, on aura :

```
 /home/mohamed/Documents/Projects/simplex/Linear-Programming-Simplexe/script2.py
La combinaison la plus optimale est de vendre 8 'lots de 10' du modele 1 et 3 'lots de 10' du modele 2
Ceci va genere 2400 Dhs
```

Exercice 3 : Gestion de coût du transport pour les usines

> Une entreprise dispose de trois usines localisées à différents endroits au pays. La production annuelle de chaque usine est la suivante :

Usine	Production annuelle
Usine 1	15 000 unités
Usine 2	12 000 unités
Usine 3	23 000 unités

Ces usines alimentent quatre points de vente dont la demande annuelle est la suivante :

Points de vente	Demande annuelle
A	10 000 unités
B	5 000 unités
C	20 000 unités
D	15 000 unités

Les coûts unitaires de transport de chaque usine à chaque point de vente sont indiqués dans le tableau suivant :

		Points de ventes			
		A	B	C	D
Usines	Usine 1	5	6	6	8
	Usine 2	11	9	4	7
	Usine 3	12	7	8	5

Formuler le modèle de programmation linéaire qui permettrait d'obtenir un plan de transport à un coût minimum.

- **On peut donner une modèle du problème :**

$$Min\ z = 5x_{1A} + 6x_{1B} + 6x_{1C} + 8x_{1D} + 11x_{2A} + 9x_{2B} + 4x_{2C} + 7x_{2D} + 12x_{3A} + 7x_{3B} + 8x_{3C} + 5x_{3D}$$

$$\begin{cases} x_{1A} + x_{1B} + x_{1C} + x_{1D} \leq 15000 \\ x_{2A} + x_{2B} + x_{2C} + x_{2D} \leq 12000 \\ x_{3A} + x_{3B} + x_{3C} + x_{3D} \leq 23000 \\ x_{1A} + x_{2A} + x_{3A} \geq 10000 \\ x_{1B} + x_{2B} + x_{3B} \geq 5000 \\ x_{1C} + x_{2C} + x_{3C} \geq 20000 \\ x_{1D} + x_{2D} + x_{3D} \geq 15000 \\ x_{1A} \geq 0\ ,\ x_{1B} \geq 0\ ,\ x_{1C} \geq 0\ ,\ x_{1D} \geq 0\ ,\ x_{2C} \geq 0\ ,\ x_{2D} \geq 0 \\ x_{2A} \geq 0\ ,\ x_{2B} \geq 0\ ,\ x_{3A} \geq 0\ ,\ x_{3B} \geq 0\ ,\ x_{3C} \geq 0\ ,\ x_{3D} \geq 0 \end{cases}$$

Avec l'utilisation du module Scipy et la méthode de Simplexe

```python
from scipy.optimize import linprog

obj_fun = [5,6,6,8,11,9,4,7,12,7,8,5]

ineq_coef = [
    [1,1,1,1,0,0,0,0,0,0,0,0],
    [0,0,0,0,1,1,1,1,0,0,0,0],
    [0,0,0,0,0,0,0,0,1,1,1,1]
]
eq_coef = [
    15000,12000,23000
]

ineq_equa = [
    [1,0,0,0,1,0,0,0,1,0,0,0],
    [0,1,0,0,0,1,0,0,0,1,0,0],
    [0,0,1,0,0,0,1,0,0,0,1,0],
    [0,0,0,1,0,0,0,1,0,0,0,1]
]
eq_equa = [
    10000,5000,20000,15000
]

X = linprog(c = obj_fun, A_ub = ineq_coef, b_ub = eq_coef, A_eq = ineq_equa, b_eq = eq_equa, method = 'highs')

print("La configuration la plus optimale est la suivant : ")

print("\nConcernant l'usine A")
print(f"Transport de l'usine A vers Point de vente 1 : {int(X.x[0])}")
print(f"Transport de l'usine A vers Point de vente 2 : {int(X.x[1])}")
print(f"Transport de l'usine A vers Point de vente 3 : {int(X.x[2])}")
print(f"Transport de l'usine A vers Point de vente 4 : {int(X.x[3])}")

print("\nConcernant l'usine B")
print(f"Transport de l'usine B vers Point de vente 1 : {int(X.x[4])}")
print(f"Transport de l'usine B vers Point de vente 2 : {int(X.x[5])}")
print(f"Transport de l'usine B vers Point de vente 3 : {int(X.x[6])}")
print(f"Transport de l'usine B vers Point de vente 4 : {int(X.x[7])}")

print("\nConcernant l'usine C\n")
print(f"Transport de l'usine C vers Point de vente 1 : {int(X.x[8])}")
print(f"Transport de l'usine C vers Point de vente 2 : {int(X.x[9])}")
print(f"Transport de l'usine C vers Point de vente 3 : {int(X.x[10])}")
print(f"Transport de l'usine C vers Point de vente 4 : {int(X.x[11])}")

print(f"\nEnfin, le cout minimal du transport pour les 3 usines et 4 points de vente est {X.fun} Dhs")
```

Ce qui nous donne la solution suivant :

```
La configuration la plus optimale est la suivant :

Concernant l'usine A
Transport de l'usine A vers Point de vente 1 : 10000
Transport de l'usine A vers Point de vente 2 : 0
Transport de l'usine A vers Point de vente 3 : 5000
Transport de l'usine A vers Point de vente 4 : 0

Concernant l'usine B
Transport de l'usine B vers Point de vente 1 : 0
Transport de l'usine B vers Point de vente 2 : 0
Transport de l'usine B vers Point de vente 3 : 12000
Transport de l'usine B vers Point de vente 4 : 0

Concernant l'usine C

Transport de l'usine C vers Point de vente 1 : 0
Transport de l'usine C vers Point de vente 2 : 5000
Transport de l'usine C vers Point de vente 3 : 3000
Transport de l'usine C vers Point de vente 4 : 15000

Enfin, le cout minimal du transport pour les 3 usines et 4 points de vente est 262000.0 Dhs
```