



Méthode du simplexe

Le but de ce TP est de coder en Python l'algorithme du simplexe pour résoudre un problème d'optimisation linéaire.

Exercice 1. (Problème d'optimisation linéaire)

On considère le problème d'optimisation linéaire suivant :

$$\begin{aligned} \max \quad & f = 2 - x_1 + 2x_2 + 3x_3 \\ \text{s.c.} \quad & 4x_1 + 4x_2 + 8x_3 \leq 2 \\ (P) \quad & x_1 - 3x_2 + 5x_3 \leq 3 \\ & 3x_1 + 7x_2 - 3x_3 \leq 0 \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

- (1) Traduire cet énoncé sous la forme d'un problème d'optimisation linéaire sous forme standard.
- (2) Coder la matrice augmentée M correspondante.
- (3) Créer le vecteur de base b défini tel que $b_j = i$ si x_j est une variable de base, présente dans la ligne i , et $b_j = -1$ sinon.
- (4) Afficher M et b .

Exercice 2. (Opérations sur la base)

- (1) Créer la fonction *solBase* qui retourne la solution de base d'une matrice augmentée accompagnée de sa base.
- (2) Créer la fonction *solBaseObj* qui retourne la valeur de la fonction objectif évaluée en la solution de base.
- (3) Créer la fonction *echanger* qui retourne le nouveau vecteur de base en fonction d'un pivot donné en (i, j) .
- (4) Tester ces fonctions.

Exercice 3. (Opérations sur la matrice augmentée)

- (1) Créer la fonction *echelRed* ramenant à 1 un pivot donné en (i, j) et à 0 les autres coefficients de la colonne du pivot à l'aide d'opérations usuelles sur les matrices augmentées.
- (2) Tester cette fonction.

Exercice 4. (Étapes de recherche du pivot)

- (1) Créer la fonction *jPivot* retournant l'indice de colonne donné par l'algorithme du simplexe sur une matrice M si il existe, -1 sinon.
- (2) Créer la fonction *iPivot* retournant l'indice de ligne donné par l'algorithme du simplexe sur la colonne d'une matrice M si il existe, -1 sinon.
- (3) Tester ces fonctions.

Exercice 5. (Algorithme du simplexe)

- (1) Créer la fonction *simplexe* résolvant une matrice augmentée en affichant la solution optimale avec la valeur de la fonction objectif, ou si le problème est non borné. On donnera aussi le nombre d'itérations de l'algorithme.
- (2) Utiliser la fonction *simplexe* pour résoudre le problème de l'exercice 1.
- (3) Testez l'algorithme sur un problème non borné.