

Contrôle Terminal BUT3 - Semestre 5

R5.A.11 Méthodes d'optimisation Parcours A



Nom Responsable	Godin Thibault
Date contrôle	17/01
Durée contrôle	1h30
Nombre total de pages	8
Impression	recto-verso
Documents autorisés	1 feuille A4 notes personnelles
Calculatrice autorisée	NON
Réponses	directement sur le sujet



NOM Prénom :

Groupe :

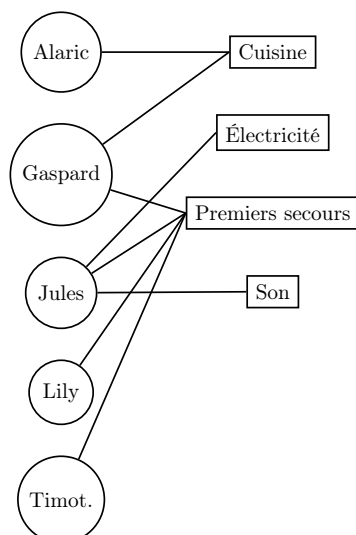
Exercice 1 : (Recuit)

1. Questions de cours

(a) En quelques mots, donner des avantages de la méthode du recuit simulé par rapport à la méthode du gradient vue en BUT2

(b) En quelques mots, donner des inconvénients de la méthode du recuit simulé par rapport à la méthode du gradient vue en BUT2

2. Mise en situation. on cherche à résoudre le problème d'affectation suivant à l'aide du recuit simulé :



(a) Décrire le fonctionnement global de l'algorithme sur cet exemple

(b) Proposer, en justifiant, une fonction de coût (énergie) adaptée à ce problème

(c) Donner une solution optimale au problème.

NOM Prénom :

Groupe :

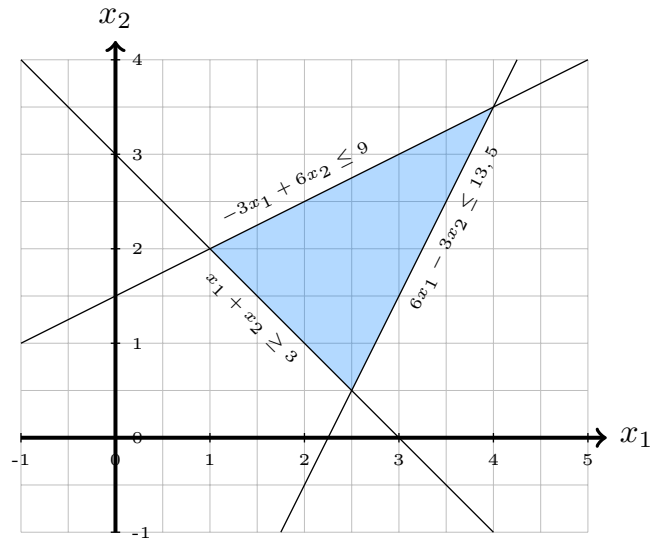
Exercice 2 : (programmation linéaire)

On cherche à maximiser $c : \mathbb{R}^2 \rightarrow \mathbb{R}, (x_1, x_2) \rightarrow 2x_1 + x_2$

On commence par s'intéresser au problème d'optimisation linéaire suivant :

Maximiser la fonction
 $c : \mathbb{R}^2 \rightarrow \mathbb{R}, (x_1, x_2) \rightarrow 2x_1 + x_2$
sous les contraintes

$$\begin{cases} x_1 + x_2 & \geq 3 \\ 6x_1 - 3x_2 & \leq 13,5 \\ -3x_1 + 6x_2 & \leq 9 \end{cases}$$



1. Donner la valeur optimale de c sur le domaine admissible. En quel point est-elle réalisée ?
2. Écrire le problème en écriture matricielle, un problème linéaire (de dimension q) sous forme canonique (c'est-à-dire sous la forme :

$$\text{Maximiser } c^T x \text{ sous les contraintes } \begin{cases} Ax & \leq b \\ x & \geq 0 \end{cases}$$

où

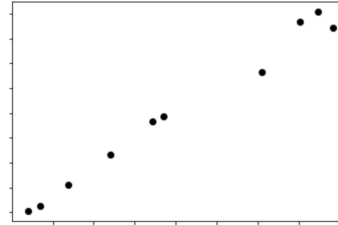
- $c = (c_1, \dots, c_q)^T$ est le vecteur colonne (à q lignes) de calcul du coût ;
 - $x = (x_1, \dots, x_q)^T$ est le vecteurs colonnes (à q lignes) des variables ;
 - $A = (a_{i,j})_{1 \leq i \leq p, 1 \leq j \leq q}$ est une matrice (à p lignes et q colonnes) des contraintes ;
 - et $b = (b_1, \dots, b_p)^T$ est le vecteur colonne (à p lignes) des bornes
-)

NOM Prénom :

Groupe :

Exercice 3 : (QCM) Entourer la ou les bonnes réponses. QCM à points négatifs.

1. (TP2b) On souhaite modéliser prix d'un appartement en fonction de surface. En étudiant le marché, on obtient le graphe suivant, où chaque point (s, p) représente un appartement vendu l'an dernier, avec la surface en abscisse et le prix en ordonnée.



- a. L'erreur commise en utilisant la méthode des moindres carrées (norme L_2) est plus faible que celle commise par la méthode de la moindre valeur absolue (norme L_1)
 - b. L'erreur commise en utilisant la méthode des moindres carrées (norme L_2) est plus grande que celle commise par la méthode de la moindre valeur absolue (norme L_1)
 - c. Les deux méthodes sont incomparables car elles utilisent des mesures de qualité différente
 - d. La méthode méthode de la moindre valeur absolue (norme L_1) est moins couteuse à calculer que la méthode des moindres carrées (norme L_2)
2. On s'intéresse à la méthode du recuit simulé avec des paramètres (température, énergies ...) et données fixées (TP1).
- a. La méthode est déterministe et renvoie toujours le même résultat sur un même ensemble de données
 - b. La méthode est aléatoire mais renvoie toujours le même résultat sur un même ensemble de données
 - c. La méthode est aléatoire et peut renvoyer des résultats différents sur même un ensemble de données
 - d. La méthode est déterministe mais peut renvoyer des résultats différents sur même un ensemble de données
3. On s'intéresse aux problèmes d'optimisation linéaire (TP2)
- a. La méthode du simplexe donne une solution en temps polynomial (selon le nombre de variables du problème).
 - b. La méthode du simplexe *en utilisant le critère de Dantzig* donne une solution en temps polynomial (selon le nombre de variables du problème).
 - c. La méthode du simplexe peut-être de complexité exponentielle, mais elle est efficace en pratique.
 - d. Il n'existe pas d'algorithme en temps polynomial pour cette classe de problèmes mais la méthode du simplexe est efficace en pratique.
4. Dans l'algorithme du K-mean
- a. Les centroïdes sont mis à jour en calculant la moyenne de tous les centroïdes précédents
 - b. Le nombre de clusters vaut toujours la dimension de l'espace des données
 - c. L'erreur diminue toujours si on augmente le nombre de clusters
 - d. Utilise le deuxième vecteur propre pour calculer les clusters

NOM Prénom :

Groupe :

Exercice 4 : (recommandation)

L'objectif de cet exercice est de réécrire un algorithme de recommandation en langage naturel. Pour cela on considère des données **data** (sous forme de tableau), contenant **number_user** lignes correspondant aux différents utilisateurs ayant noté des films, ainsi que **number_movie** colonnes contenant les notes de 1 à 5 attribuées aux films par les utilisateurs. La valeur -1 est indiquée lorsqu'un utilisateur n'a pas vu le film.

On suppose que l'on a déjà implémenté les trois fonctions suivantes :

- La fonction `historique`, prenant en entrée un utilisateur **user**, et donnant la liste des indices des films vu par **user**.
 - La fonction `user_distance`, prenant en entrée deux utilisateurs **user_1** et **user_2**, et donnant une distance positive. (Cette distance augmente si les utilisateurs **user_1** et **user_2** ont des profils similaires).
 - La fonction `movie_score`, prenant en entrée un film **movie** et une liste d'utilisateur et donnant la somme des notes données par ces utilisateurs à **movie**.
1. (a) Écrire en pseudocode une méthode `neighborhood` qui prend en entrée un utilisateur **user** et un nombre de voisins **k**, et qui donne la liste des **k** utilisateurs ayant la plus grande distance avec **user**.

- (b) Écrire en pseudocode une méthode `recommandation` qui prend en entrée un utilisateur **user** et un nombre de voisin **k**, et qui donne le film non-vu par **user** ayant un `movie_score` le plus élevé sur son voisinage `neighborhood`.

2. Nous allons maintenant utiliser le tableau de données suivant pour effectuer à la main une recommandation de film.

Utilisateur	Film 0	Film 1	Film 2	Film 3	Film 4	Film 5
A	-1	4	3	3	5	-1
B	2	3	-1	3	-1	4
C	1	2	1	4	1	-1
D	5	2	3	-1	4	-1

Voici les différents paramètres de notre recommandation :

- La distance en utilisateur est le nombre de films vu en commun.
- La taille d'un voisinage est 2
- Le score d'un film dans un voisinage est la somme des notes qui lui ont été attribué par les utilisateurs du voisinage.

- (a) Déterminer le voisinage de l'utilisateur *A* en détaillant et expliquant vos étapes de calcul.

- (b) Quel film notre algorithme recommandera à l'utilisateur *A* ? Expliquer les étapes de votre raisonnement.