Devoir Informatique de base

A remettre le 16 Novembre 2018 par mail à henry.soldano@icloud.com sous forme d'une archive contenant les sources et un fichier pdf décrivant le travail et donnant des exemples d'exécution. Le devoir est prévu pour être fait en binôme.

Sujet

Le travail consiste à développer et expérimenter un programme qui sélectionne un sous-ensemble S d'un ensemble E tel que S maximise un fonction positive et additive f en respectant une contrainte de distance entre éléments de S. Pour cela on implémentera un algorithme glouton décrit dans la suite de l'énoncé.

Le devoir s'appuiera sur les sources `ensemblesI.h` et `ensemblesI.c` développées par les élèves pendant ce cours. Ces sources permettent de manipuler des ensembles d'entiers et représentent un ensemble par une liste simplement chaînée triée par ordre croissant. Nous aurons également besoin de transposer une partie de ces fonctionnalités pour travailler sur des ensembles d'ensembles d'entiers. Pour cela nous associerons à un ensemble d'entiers son indice dans un tableau d'ensembles d'entiers.

Définitions

Soit un ensemble E muni d"une distance d entre éléments. Soit S un sous-ensemble de E, S est un *candidat* si :

1. Pour toute paire x, y d'éléments de S on a $d(x,y) > \beta$, et
2. S est maximal au sens de l'inclusion : il n'existe pas de sous-ensemble $S' \supset S$ satisfaisant la condition 1

Exemple 1 $E = \{1,2,3,4\}$ *avec* $d(1,2) = d(2,3) = d(1,3) = d(1,4) = 1$ *et* $d(2,4) = d(3,4) = 0.1$ *et* $\beta = 0.9$

— $\{1,2,3\}$ *est candidat : il satisfait la condition 1 et est maximal car* $\{1,2,3,4\}$ *ne satisfait pas la condition 1.*

— $\{1,2\}, \{1,3\}, \{2,3\}$ *satisfont la condition 1 mais ne sont pas candidats car ils sont inclus dans* $\{1,2,3\}$ *qui satisfait la condition 1 .*

— $\{1,4\}$ *est candidat car* 2 *et* 3 *sont tels que* $d(2,4) \leq 0.9$ *et* $d(3,4) \leq 0.9$*, on ne peut donc rien ajouter à* $\{1,4\}$ *sans contredire la condition 1.*

Soit g une fonction positive définie sur E et f une fonction qui appliquée à un sous-ensemble S de E renvoie la somme des valeurs de g pour ses éléments :

$$f(S) = \Sigma_{x \in S} g(x)$$

On cherche alors à résoudre le problème suivant :

Problème 1 Trouver $S^* \subseteq E$ tel que S^* est un candidat et $f(S*)$ est maximum

Exemple 2 *Reprenons l'exemple 1 et $g(1) = g(2) = g(3) = g(4) = 1$. Pour tout sous-ensemble S, $f(S)$ est son cardinal $|S|$. Le maximum de f pour les candidats est 3 qui est atteint pour un seul candidat $S^* = \{1, 2, 3\}$*

Ce problème a une complexité importante, mais on a un algorithm glouton, peu couteux et qui donne de bons résultats :

Glouton
/ * *On suppose E ordonné par valeurs de g décroissantes.* */
$F = E$
$S \leftarrow \emptyset$
Tantque $F \neq \emptyset$
— On cherche le premier x dans F tel que pour tout y de S, $d(x, y) > \beta$.
— Si on a trouvé x alors
— — On élimine de F x et ses prédécesseurs dans F.
— S=$S \cup \{x\}$
— Sinon $F = \emptyset$
— finSi
FinTq

Exemple 3 *Nous reprenons l'exemple précédent avec toujours $\beta = 0.9$. Comme g est uniforme, il n'y a pas d'ordre privilégié. On considère ici l'ordre $E = [1, 2, 4, 3]$. A l'initialisation de* **Glouton** *on a donc $F = [1, 2, 4, 3]$ et $S = \emptyset$ et on effectue les itérations suivantes :*

- *$S = \{1\}, F = [2, 4, 3]$ car S étant vide la condition 1 est satisfaite quand on ajoute 1.*
- *$S = \{1, 2\}$ et $F = [4, 3]$ car $d(1, 2) > 0.9$.*
- *$S = \{1, 2, 3\}$ et $F = \emptyset$ car $d(2, 4) \leq 0.9$ et donc on ne peut ajouter 4 à S, mais on peut ajouter 3 car $d(3, 1) > 0.9$ et $d(3, 2) > 0.9$.*

On obtient donc $S = \{1, 2, 3\}$ de valeur $f(S) = 3$. Il est clair qu'on ne peut faire mieux : le seul sous-ensemble tel que $f(S) > 3$ est E mais E contredit la condition 1 et n'est donc pas candidat. Remarquons cependant qu'avec l'ensemble E ordonné par $E = [4, 1, 2, 3]$ on aurait obtenu $S = \{4, 1\}$ de valeur $f(S) = 2$ car on n'aurait pas pu ajouter 2 et 3. **Glouton** *ne donne donc pas toujours une solution exacte au Problème 1.*

Travail

On veut implémenter et expérimenter **Glouton** pour travailler sur un ensemble E d'ensemble d'entiers. La notion de distance entre ensembles d'entiers utilisée est la distance de Jaccard et s'écrit

$$d(e_1, e_2) = 1 - \frac{|e_1 \cap e_2|}{|e_1 \cup e_2|}$$

Pour fonction g sur les ensembles d'entiers on prendra d'abord la cardinalité :

$$g(e) = |e|$$

On expérimentera d'abord **Glouton** sur l'exemple donné dans l'énoncé ainsi que sur l'exemple fourni dans le fichier `ensembleE.txt`. Dans celui-ci chaque ligne représente un ensemble d'entiers e_i. Une ligne est constituée de l'identifiant i de l'ensemble (ici entre 0 et 70) suivi des éléments de e_i.

On fera une deuxième expérimentation en utilisant une fonction g différente. Nous considérons maintenant la centralité de e relativement à la taille moyenne $\hat{t}$ des ensembles dans E :

$$g(e) = \hat{t} - abs(\ t(e) - \hat{t}\ ) \text{ si } \hat{t} \geq abs(\ t(e) - \hat{t}\ ) \text{ et 0 sinon}$$

La centralité de e est donc la taille moyenne $\hat{t}$ si la taille de e est la taille moyenne, est nulle si e est de taille nulle ou si e a plus de deux fois plus d'éléments que la moyenne.

Ces expérimentations seront exécutées par un programme qui pour chaque valeur de β entre 0 et 1 par pas de 0..05 envoie sur la sortie standard l'ensemble S trouvé, sa taille ainsi que sa valeur $f(S)$. Le programme donnera également un tableau représentant les tailles de S en fonction de β. On utilisera un outil graphique pour représenter la courbe correspondante dans le rapport à remettre avec les sources du programme.¿

Exemple 4 *Voici un exemple simple*

E=e1, e2, e3, e1=1,2 e2=2,3 e3=3,4

d(e1,e2)=2/3 d(e2,e3) =2/3 d(e1,e3) =1

tMoy=2

g(e1)=g(e2)=g(e3)= 2 -(2-2) =2 car 2≥2-2=0

C'est le cas uniforme, on prend arbitrairement l'ordre e1,e2,e3 (mais on peut en choisir un autre). Et on prend beta=3/4

Glouton :

- S= e1

- S= e1,e3 car d(e1,e2) =2/3 ≤3/4 et d(e1,e3)=1

Fin

———-

Autre exemple

E=e1, e2, e3 e1=1,2 e2=1 e3=1,3
d(e1,e2)=1-1/2=1/2 d(e2,e3) =1- 1/2 =1/2 d(e1,e3) =1 -1/3= 2/3
tMoy=5/3
g ?
5/3 $\geq$ *Abs(t(e1)-5/3)= 2-5/3 = 1/3* $\Rightarrow$ *g(e1)=5/3 -(1/3) = 4/3*
5/3 $\geq$ *Abs(t(e2)-5/3)= 5/3 -1 = 2/3* $\Rightarrow$ *g(e2)=5/3 -(2/3) = 1*
5/3 $\geq$ *Abs(t(e3)-5/3)= 2-5/3 = 1/3* $\Rightarrow$ *g(e3)=5/3 -(2-5/3) = 4/3*
On obtient l'ordre e1, e3 ,e2 On prend beta=1/2
Glouton :
- S= e1
- S= e1, e3 car d(e1,e3)=2/3>1/2
- Fin car d(e2,e1) =1/2 $\leq$ *1/2*