

# *supports attendus locaux*

## 1 Introduction

Le programme *supportsAttendusLocaux* calcule les supports attendus de patterns clos abstraits locaux identifiés par *minerLC* dans un graphe donné.

## 2 Algorithme de calcul d'un support attendu local

L'algorithme 1 décrit le mode de calcul du support attendu local d'un pattern clos abstrait  $c$  possédant un support (non abstrait) observé  $K = |ext(c)|$  dans un graphe  $G$ .

## 3 Dossiers disponibles

- *include* contient les fichiers d'en-têtes,
- *src* contient les programmes sources.

## 4 Compiler *supportsAttendusLocaux*

*supportsAttendus* peut être exécuté sur tout système qui compile des programmes c++11. Ouvrir un terminal, aller dans le dossier *src* et taper *make* pour créer un fichier exécutable *supportsAttendus*.

## 5 Executer *supportsAttendusLocaux*

### 5.1 Paramètres obligatoires

*supportsAttendusLocaux* attend obligatoirement et dans l'ordre 1) le nom du fichier (format nri) contenant les transactions et le graphe et 2) le nom du fichier contenant les patterns clos et clos locaux abstraits identifiés par *minerLC*. Par exemple :

---

**Algorithm 1** *supportAttendu*

---

**Input**

-  $G$  /\* graphe initial \*/  
-  $c$  /\* pattern clos abstrait identifié par *minerLC* \*/  
-  $a$  /\* une abstraction (éventuellement composée) \*/  
-  $K$  /\* support (non abstrait) de  $c$  observé dans  $(G)$  ( $K = |ext(c)|$ ) \*/  
-  $N$  /\* nombre de tirages \*/

**Output**

support attendu de  $c$  dans  $a(G)$

**Begin**

$somme \leftarrow 0$

**for**  $i \leftarrow 1$  à  $N$  **do**

$T \leftarrow K$  noeuds tirés au hasard dans  $G$

$G_A \leftarrow a(G(T))$  /\*  $G(T)$  est le sous-graphe de  $G$  induit par  $T$  \*/

/\*  $a(G(T))$  est le graphe issu de l'application de l'abstraction  $a$  à  $G(T)$  \*/

calculer les composantes connexes de  $G_A$

$somme \leftarrow somme +$  taille moyenne d'une composante connexe  
dans  $G_A$

**end for**

**return**  $somme/N$

**End.**

---

```
supportsAttendusLocaux mougel.nri mougel.res
```

## 5.2 Paramètres facultatifs

On peut utiliser les paramètres concernant les abstractions globales ( $a_1$ ) et locales ( $a_2$ ) définies dans la documentation de *minerLC*.

Par ailleurs, l'argument  $-n N$  signifie qu'on demande  $N$  tirages aléatoires (cf algorithme 1) (par défaut  $N = 1$ ). Par exemple

```
supportsAttendusLocaux mougel.nri mougel.res -c1 4 -t2 -n 5
```

exécute *supportsAttendusLocaux* à partir du graphe contenu dans le fichier *mougel.nri* et des patterns clos abstraits issus du programme *minerLC* contenus dans le fichier *mougel.res*. Le programme exécutera  $k = 5$  fois le processus décrit dans l'algorithme 1 en appliquant une abstraction de composante connexes de taille au moins 4 et une abstraction de triangles.

## 6 Format des données d'entrée

Le fichier *mougel.nri* et *mougel.res* du dossier *data* contiennent des exemples des données d'entrées. Chaque ligne d'un fichier *.res* (résultats issus de *minerLC*) est de la forme

$$(c, ext(c), ext_{a_1}(c), ext(l_{cc}), ext_{a_1}(l_{cc}), cc, l_{cc})$$

où  $c$  est un motif clos abstrait,  $ext(c)$  son extension globale et  $ext_{a_1}(c)$  son extension abstraite par  $a_1$ .  $cc$  est une composante connexe du graphe induit par  $c$  lorsque les 2 abstractions  $a_1$  et  $a_2$  ont été appliquées.  $l_{cc}$  est le motif clos local dont l'extension est  $cc$ ,  $ext(l_{cc})$  est son extension globale et  $ext_{a_1}(l_{cc})$  est son extension abstraite par  $a_1$ .

Par exemple (par souci de lisibilité, chaque élément du sextuplet est mis sur une ligne séparée) :

```
{"jazz"},
{"A", "B", "C", "D", "G", "J", "L", "P"},
{"A", "B", "C", "D", "G", "J", "L"},
{"A", "B", "C", "D", "G", "P"},
{"A", "B", "C", "D", "G"},
{"A", "B", "C", "D"},
{"jazz", "rock", "folk"}
```

## 7 Format des sorties

Chaque ligne du fichier de sortie de *supportsAttendusLocaux* est de la forme :

$$(c, K = |ext(c)|, |ext_{a_1}(c)|, |ext(l_{cc})|, |ext_{a_1}(l_{cc})|, |cc|, s_{a_1+2}(K), \sigma, l_{cc})$$

où les supports remplacent les extensions,  $s_{a_1+2}(K)$  étant la taille moyenne attendue d'une composante connexe et  $\sigma$  son écart-type.

Par exemple :

`{"jazz"}, 8, 7, 6, 5, 4, 1.611, 2.42357, {"jazz", "rock", "folk"}`

correspondant au sextuple donné à la section précédente.