Exploration des mémoires à court et long terme pour la classification multi-labels en flux

Xihui WANG, Pascale KUNTZ, Frank MEYER

Orange Labs | LS2N - Univ. Nantes

EGC 2021

Motivation applicative – Classification Multi-labels

Instance (texte, image, audio, gène, …) décrite par un vecteur d'attributs

$$x_t = [x_{t,j}]_{j \in [1,M]} \in R^M$$

Ensemble de labels décrits par un vecteur binaire

$$y_t = [y_{t,j}]_{j \in [1,L]} \in \{1,0\}^L$$

Attributs	att 1	att 2	…	att M
Inst 1	0,678	0.549	…	0.127
…	…	…	…	
Inst N	0.008	0.575	…	0.731

Labels	lab 1	lab 2	…	lab L
labelset 1	0	1	…	1
…	…	…	…	
labelset N	1	1	…	0

Motivation applicative – Classification en flux

- De nombreuses organisations génèrent des données massives à grande vitesse chaque instant et la distribution de ces données peut changer au cours du temps.
- Pour un système interactif, il est nécessaire qu'il puisse s'adapter à un changement et répondre au besoin des utilisateurs rapidement.

On souhaite construire un modèle qui puisse prédire et apprendre à chaque instant sur des flux de données.

Classification Multi-labels en flux - Definition

Le problème générique de classification multi-labels en flux consiste à **générer une séquence de modèles** $\{h_0, h_1, \ldots, h_t, \ldots\}$**sur un flux de données** $D = \{(x_0, y_0), (x_1, y_1), \ldots, (x_t, y_t), \ldots\}$:

- avec (x_t, y_t) un exemple arrivant au temps t
- et h_t le modèle calculé en fonction du modèle précédent h_{t-1} et de l'exemple (x_t, y_t)
-

Quatre propriétés [Bifet et Gavaldà, 2009] :

(I) chaque exemple arrive séquentiellement

(II) il existe potentiellement une infinité d'exemples

(III) la distribution statistique des exemples n'est pas nécessairement stationnaire (dérive de concept)

(IV) après traitement, chaque exemple est supprimé ou archivé dans une mémoire de taille très limitée

Etat de l'art

Les approches de la littérature peuvent se structurer en trois grandes familles :

- Les approches par transformation qui transforment le problème multi-labels en plusieurs problèmes mono-label
  - *Binary Relevance*: construire un classifieur binaire pour chaque label [Woloszynski et Kurzynski, 2011]
- Les approches par adaptation qui adaptent les algorithmes en-ligne mono-label au traitement des données multi-labels
  - *ISOUPTree:* est basée sur un arbre de Hoeffding [Osojnik et al., 2017]
- Les approches ensemble comme des techniques de "bagging" ou de "boosting" qui agrègent les résultats d'un ensemble de modèles plus simples
  - EaHTps: surveille la performance des membres de l'ensemble et réinitialise le plus mauvais classifieur [Read et al., 2012].

Les limites des méthodes de l'état de l'art et notre objectif

Les limites des méthodes de la littérature :

- Absence de prise en compte des corrélations entre les labels
- Mauvaise adaptation aux dérives de concept
- Problème d'oubli catastrophique
- Grande complexité temporelle

OMk se focalise sur trois impératifs :

- la prise en compte des corrélations entre labels,
- l'équilibre entre l'oubli (dérive de concept) et la mémoire (oubli catastrophique),
- la rapidité de l'apprentissage sans détriment de sa qualité.

Craft-ML [Siblini et al., 2018]

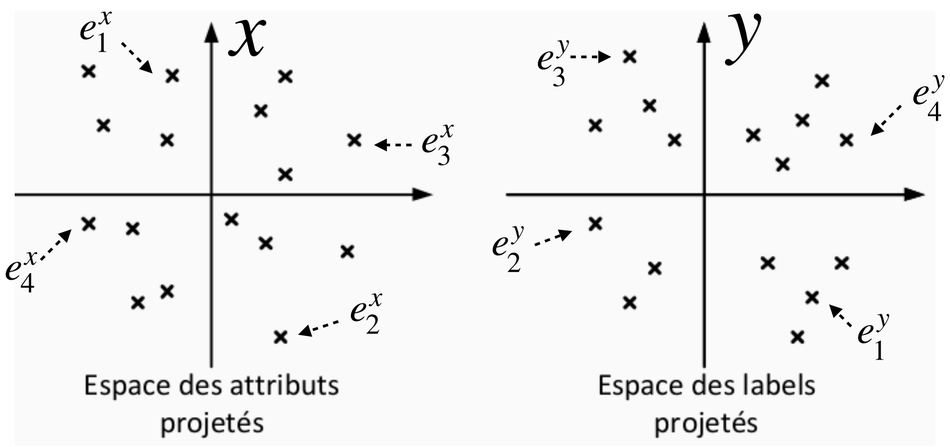


Image 1

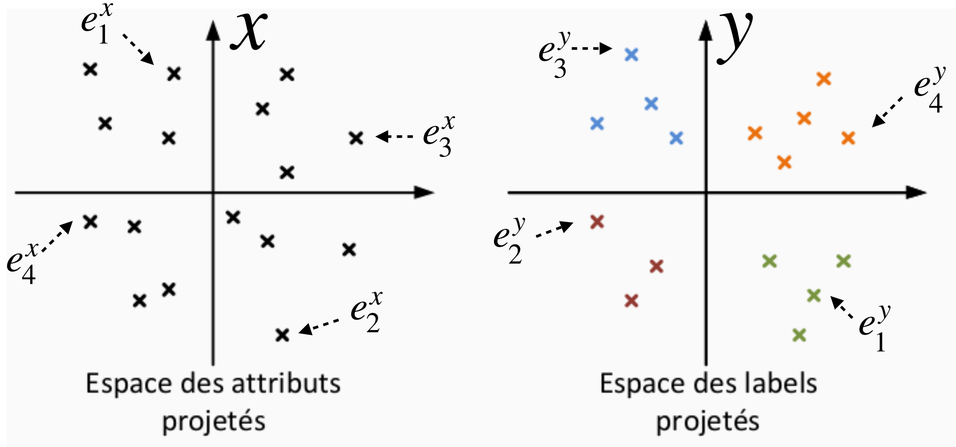


Image 2

1. K-means sur Y → $Clus_{Y1}, \ldots, Clus_{Yn}$
2. $Clus_{Y1}, \ldots, Clus_{Yn} \rightarrow Clus_{X1}, \ldots, Clus_{Xn}$
3. Pour chaque cluster X, calcul de c_i^x= moyenne des attributs de la classe

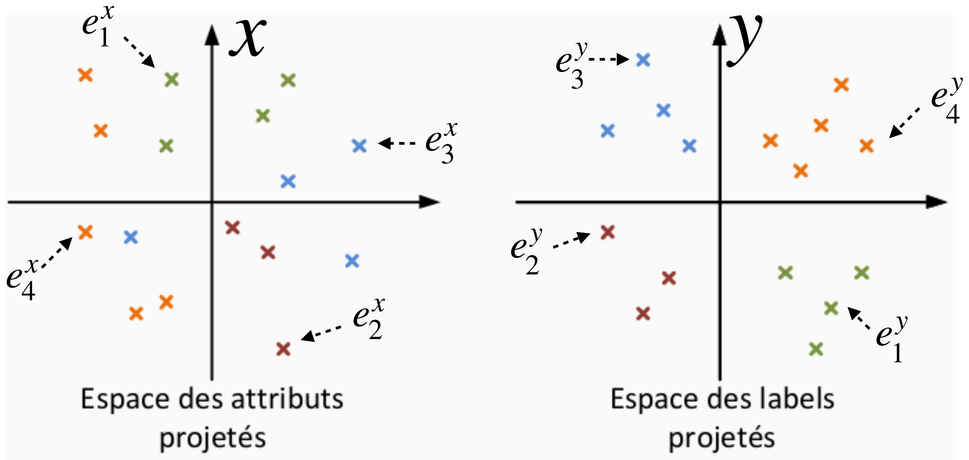
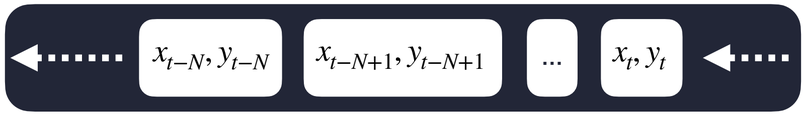


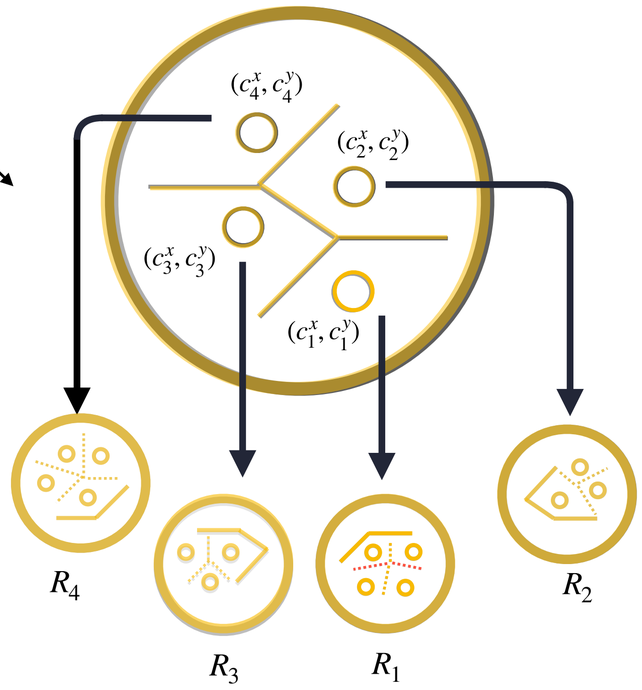
Image 3

Module algorithmique "Online Memory k-means" (OMk)

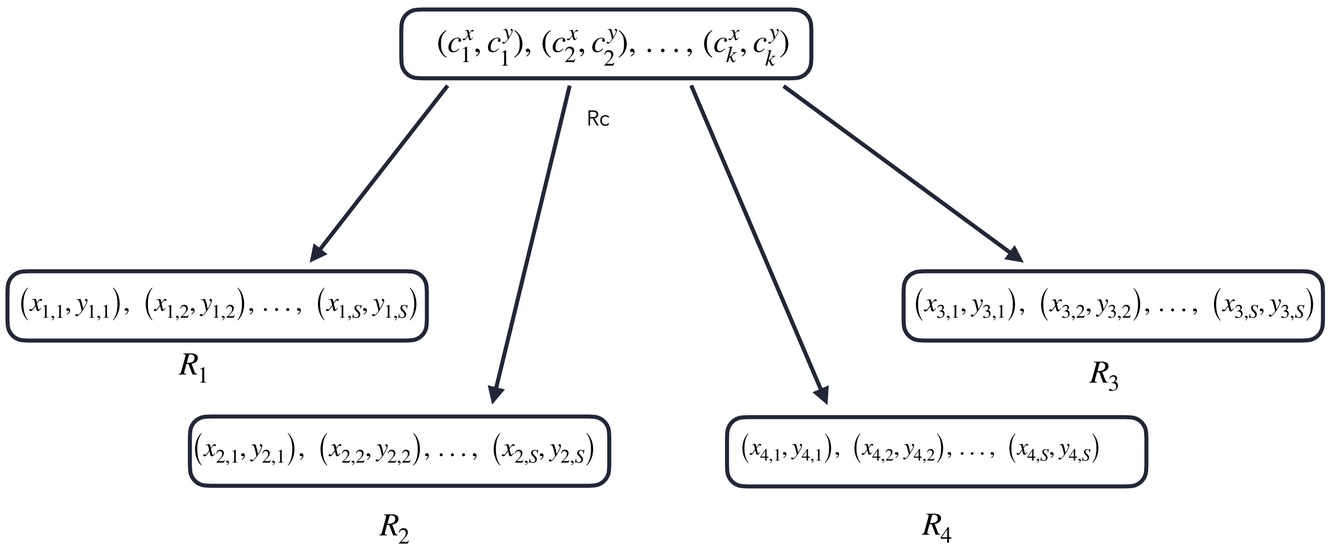


Fenêtre glissante sur le flux de données pour une mémoire à court terme

OMk

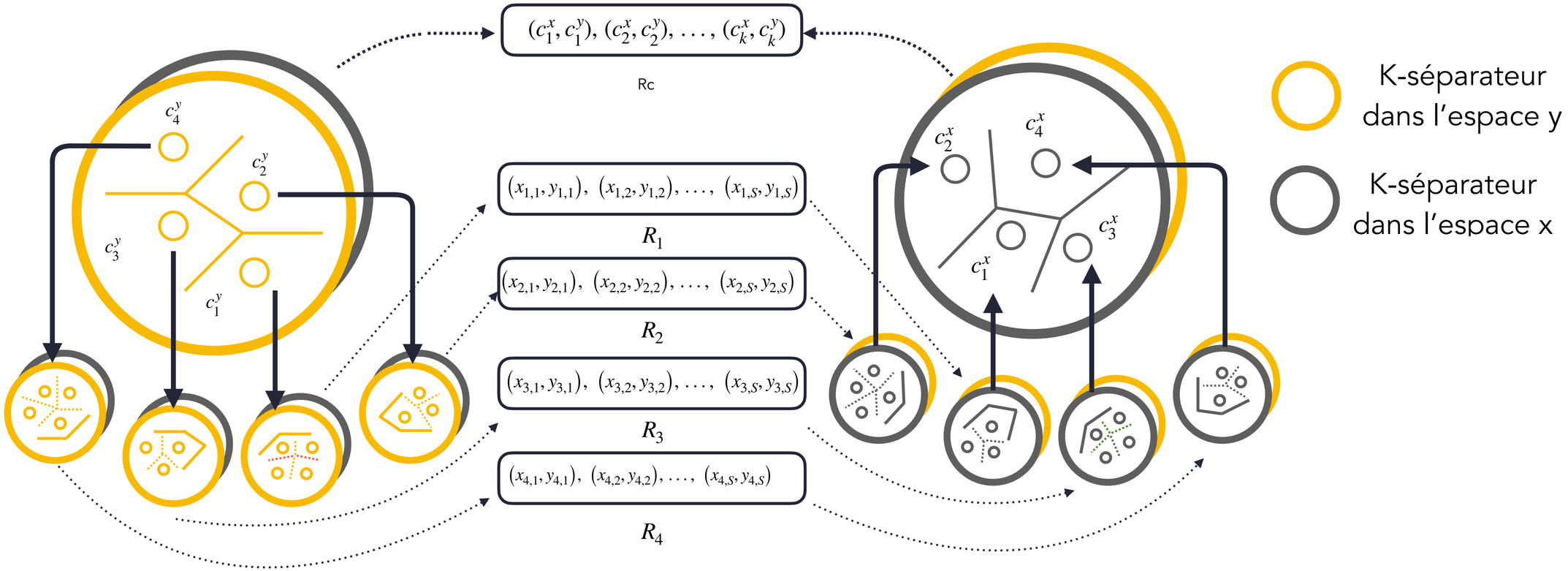


=



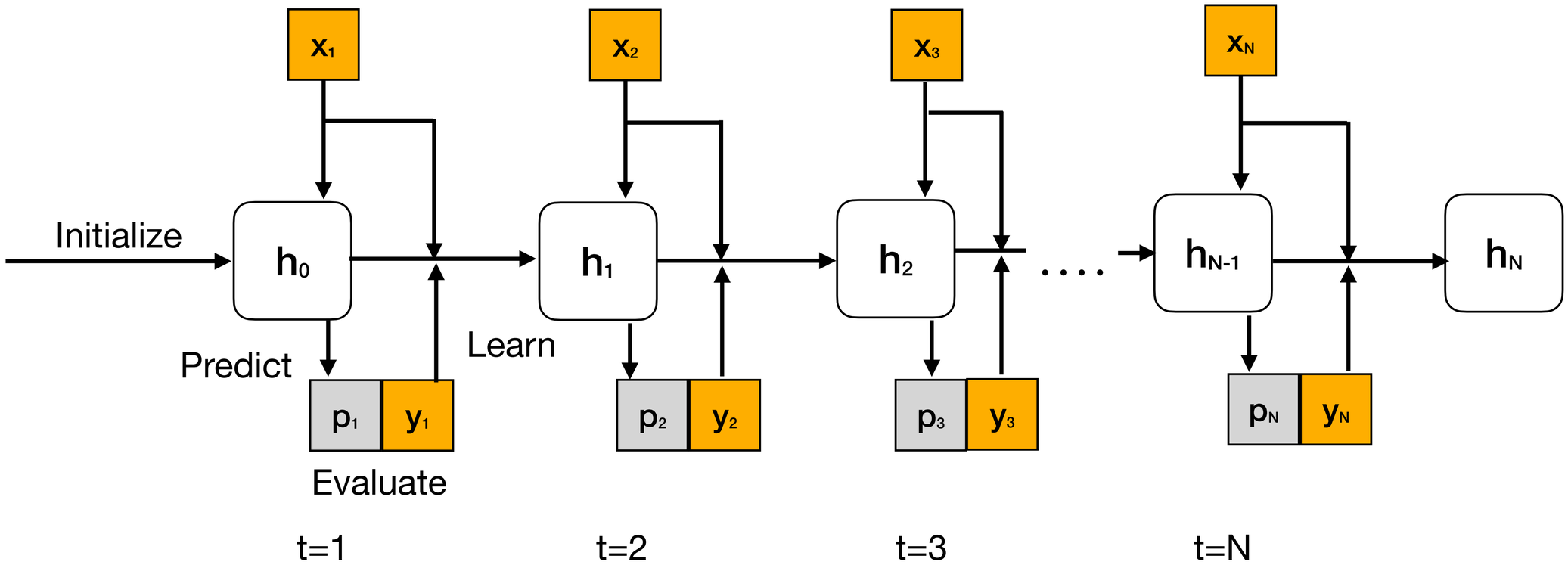
une structure en arbre associée à un échantillonnage en réservoir pour une mémoire à long terme

Module algorithmique "Online Memory k-means" (OMk)



K-séparateur et réservoir sampling pour une mémoire à long terme

Protocole expérimental - *Interleaved Test-Then-Train*



A chaque instant *t*, l'exemple (x_t , y_t) arrivant est traité selon trois étapes dans l'ordre : la prédiction, l'évaluation et l'apprentissage.

Jeux de données

Jeu	Domaine	Nombre d'exemples	Nombre d'attributs	Nombre de labels	Cardinalité de labels
20NG	Text	19 300	1006	20	1.02
Bookmarks	Text	87 856	2150	208	2.03
Corel16k	Image	13 770	500	153	2.859
Enron	Text	1 702	1001	53	3.38
Mediamill	Video	43 907	120	101	4.38
Ohsumed	Text	13 930	1002	23	1.663
TMC2007	Text	28 596	500	22	2.22
RBF-drift1	Synthétique	100 000	10	10	$1.5 \rightarrow 3.5$
RTG-drift1	Synthétique	100 000	10	10	$1.8 \rightarrow 3.0$
RBF-drift2	Synthétique	100 000	10	10	2.8 (LD :$0.25 \rightarrow 1$)
RTG-drift2	Synthétique	100 000	10	10	1.6 (LD :$0.25 \rightarrow 1$)

* La cardinalité des labels (LC) : le nombre moyen de labels positives par exemple
* La dépendance conditionnelle (LD) : $p(y \mid x)$

Comparaison avec trois algorithmes de référence

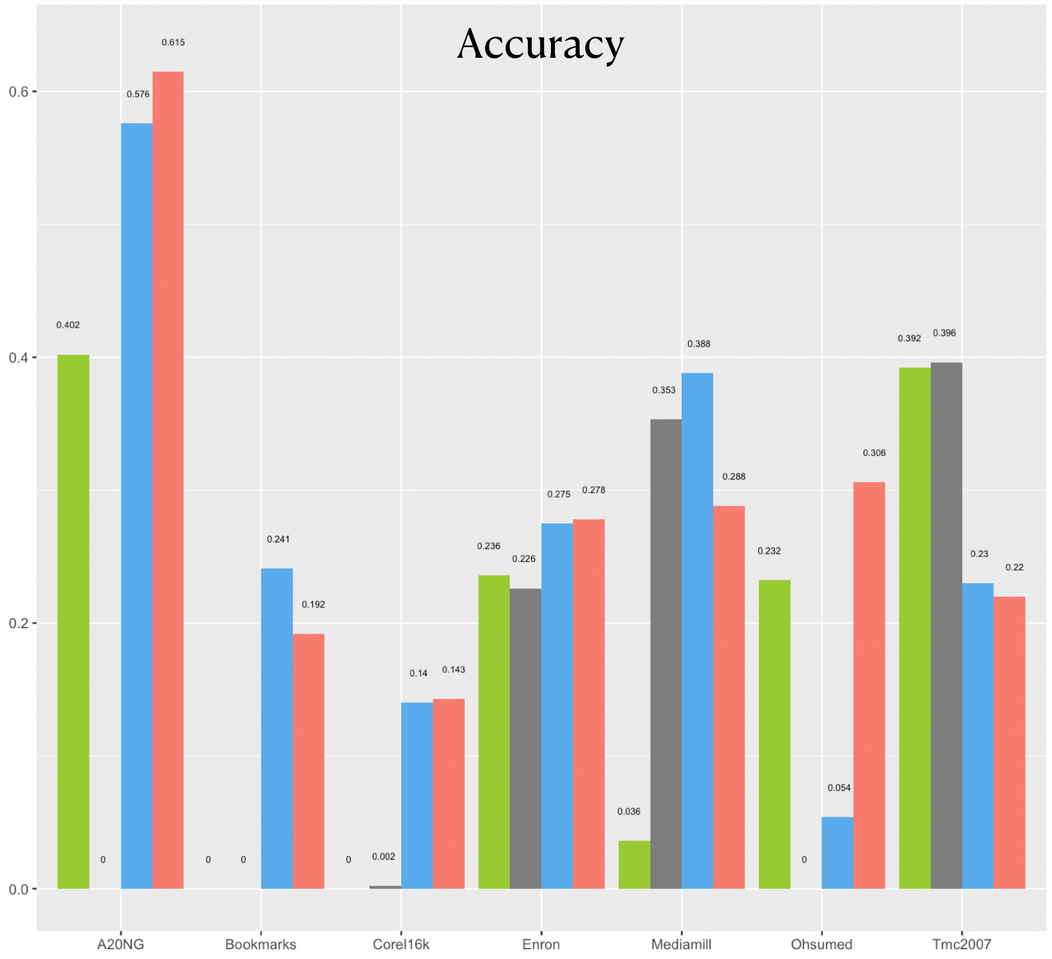
- **EaHTps**: est une méthode ensemble composée de plusieurs modèles d'HTps, eux-mêmes fondés sur un arbre de Hoeffding avec une stratégie de PS [Read et al., 2012]

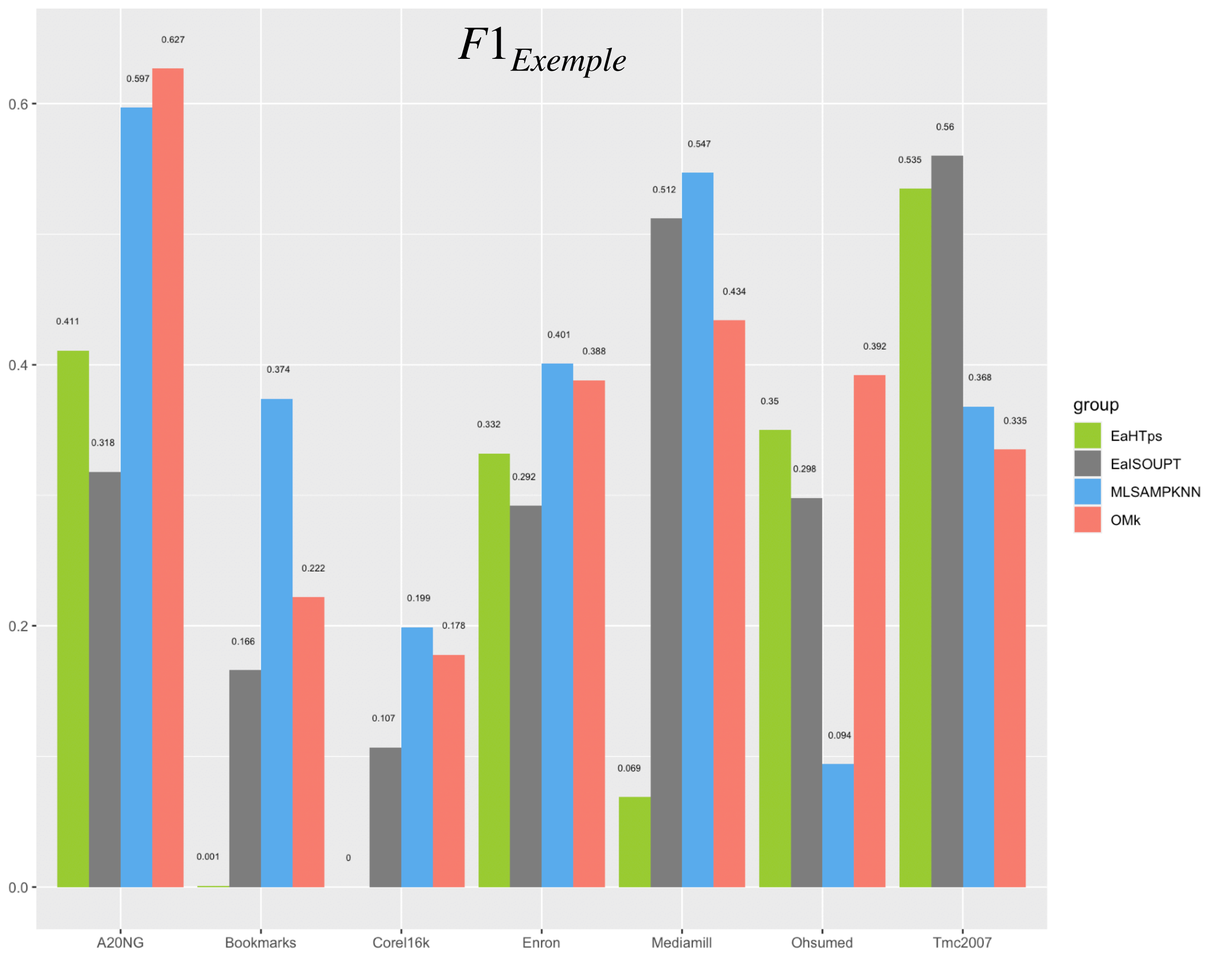
- **EaISOUPTree**: Idem que EaHTps mais avec une stratégie basée sur des percetrons adaptatifs [Osojnik et al., 2017]

- **MLSAMPKNN**: utilise la stratégie KNN avec une fenêtre auto-ajustable qui garde en mémoire certaines données du passée [Roseberry et al., 2019]

Résultats expérimentaux

Performances basé sur exemple - Stationnaire

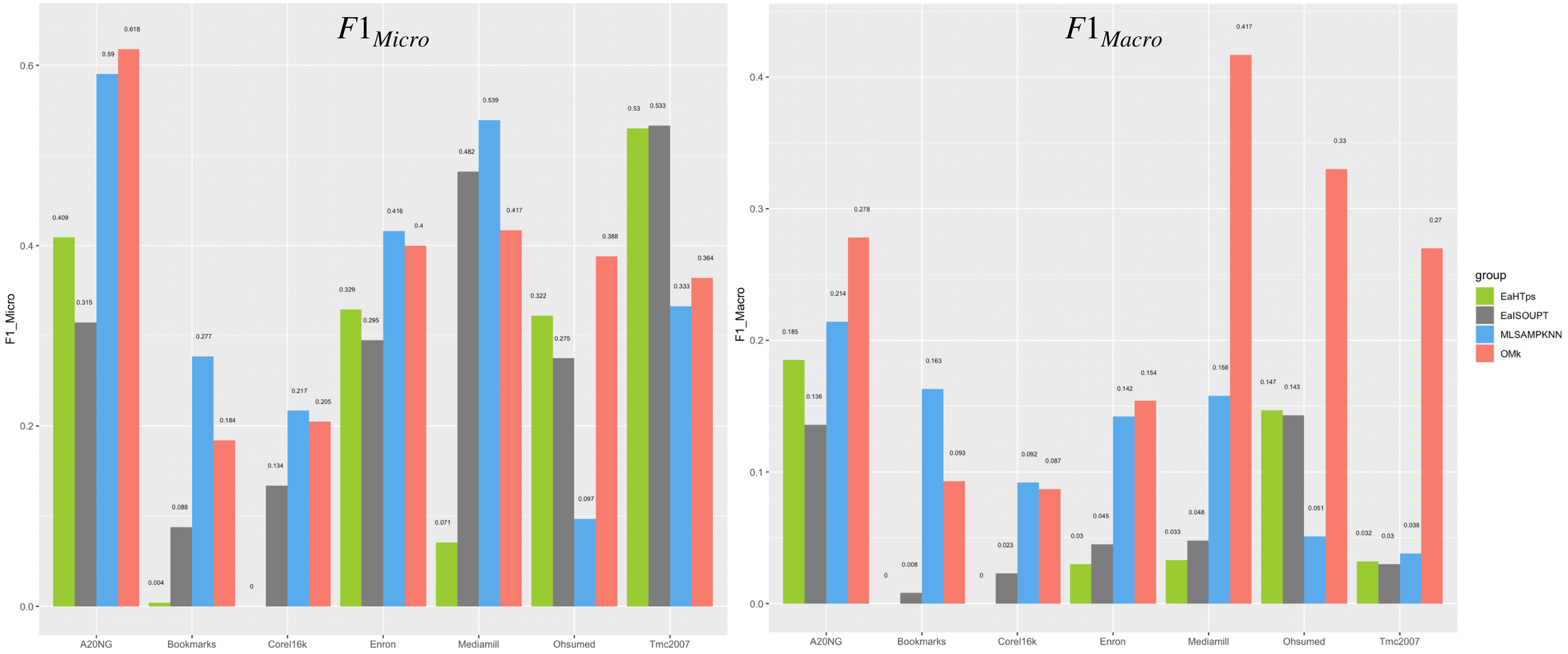




L'algorithme MLSAMPKNN est performant, et OMK obtient des performances très compétitives

Résultats expérimentaux

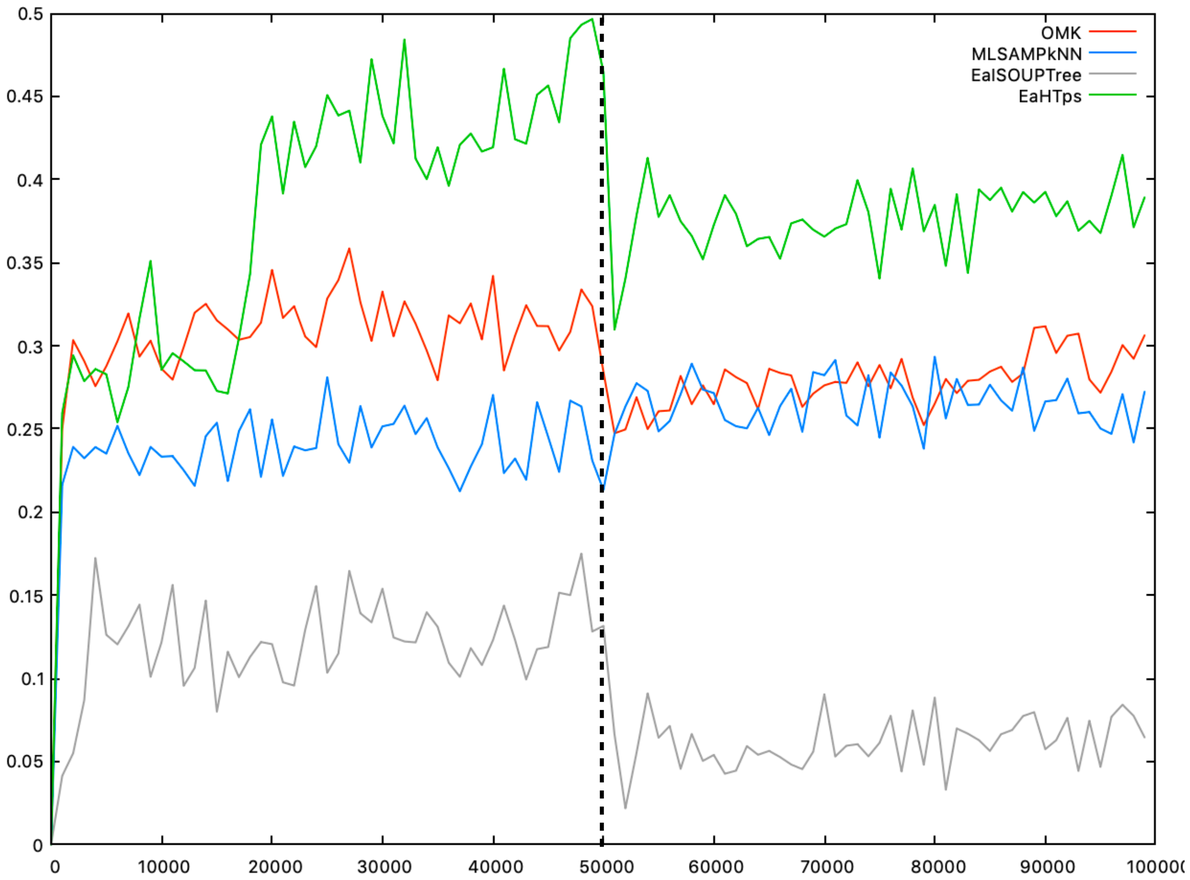
Performances basé sur label - Stationnaire

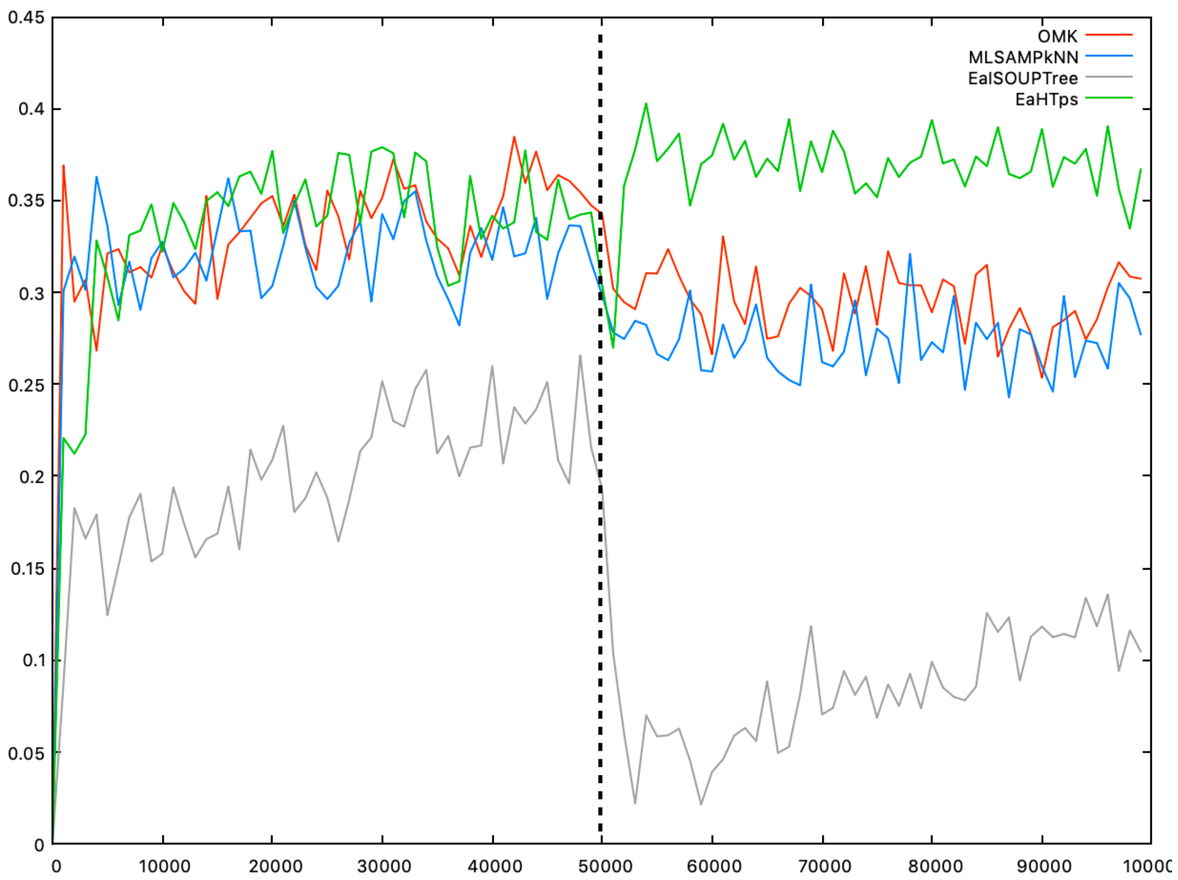


OMK obtient de bonnes performances pour la mesure F1-macro

Résultats expérimentaux

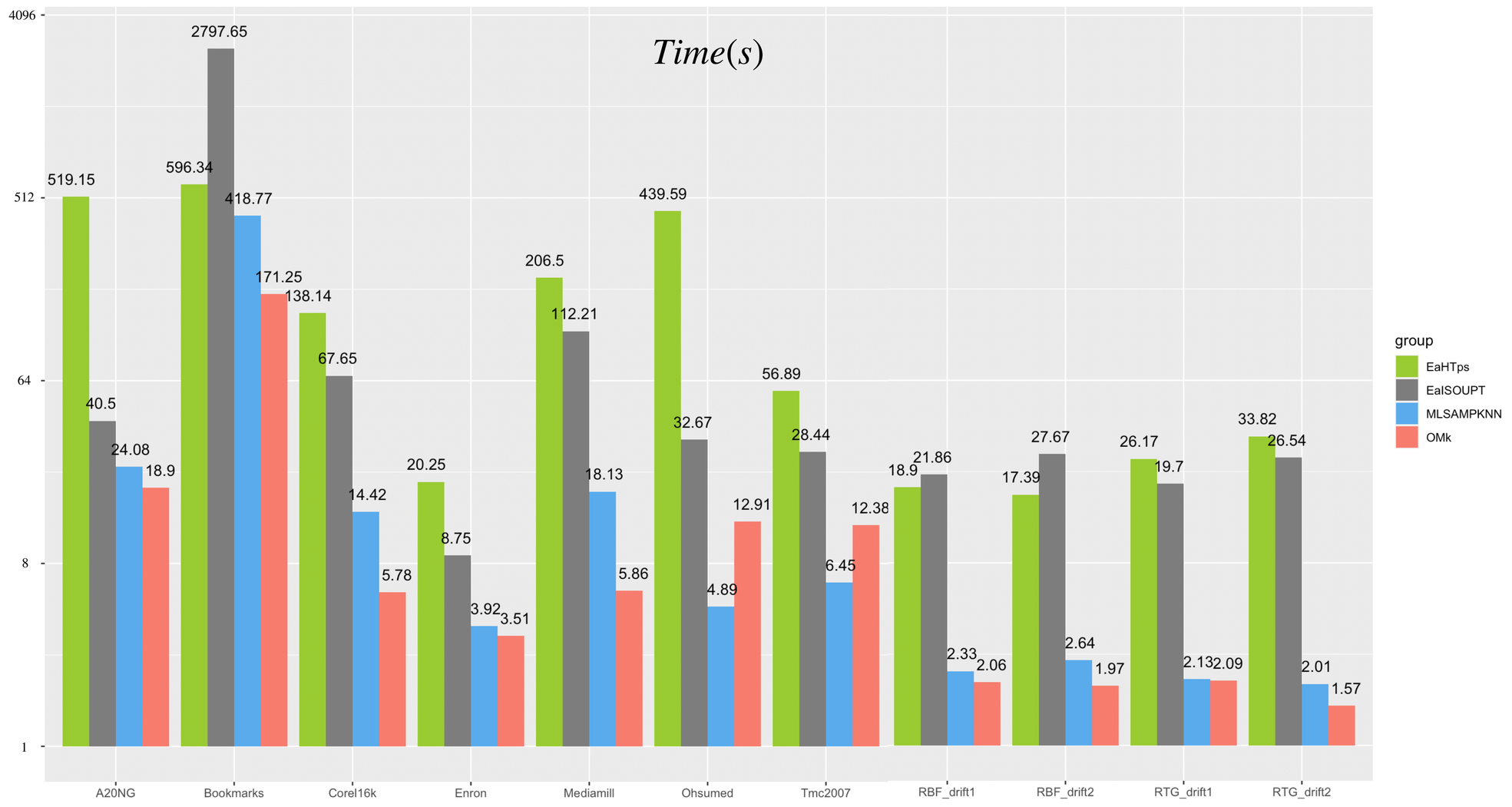
Performances(F1_Exemple) au cours de temps - Non-stationnaire (LD Drift)





L'algorithme EaHTps est performant, et OMK obtient des performances prometteuses

Temps de calculs moyens des algorithmes sur les jeux de données



OMK obtient dans 9 cas sur 11 les meilleurs temps d'exécution.

Conclusions et travaux futurs

Faits marquants :

- la prise en compte des corrélations entre labels : 👍 grâce au co-clustering
- l'oubli catastrophique (cas stationnaire) : 👍
- la rapidité de l'apprentissage est un atout majeur de la méthode : 👍
- la dérive de concept (cas non stationnaire) : 🙂

Proposition :

Proposer le **modèle OMk** pour résoudre le problème de **classification multi-labels en flux** :

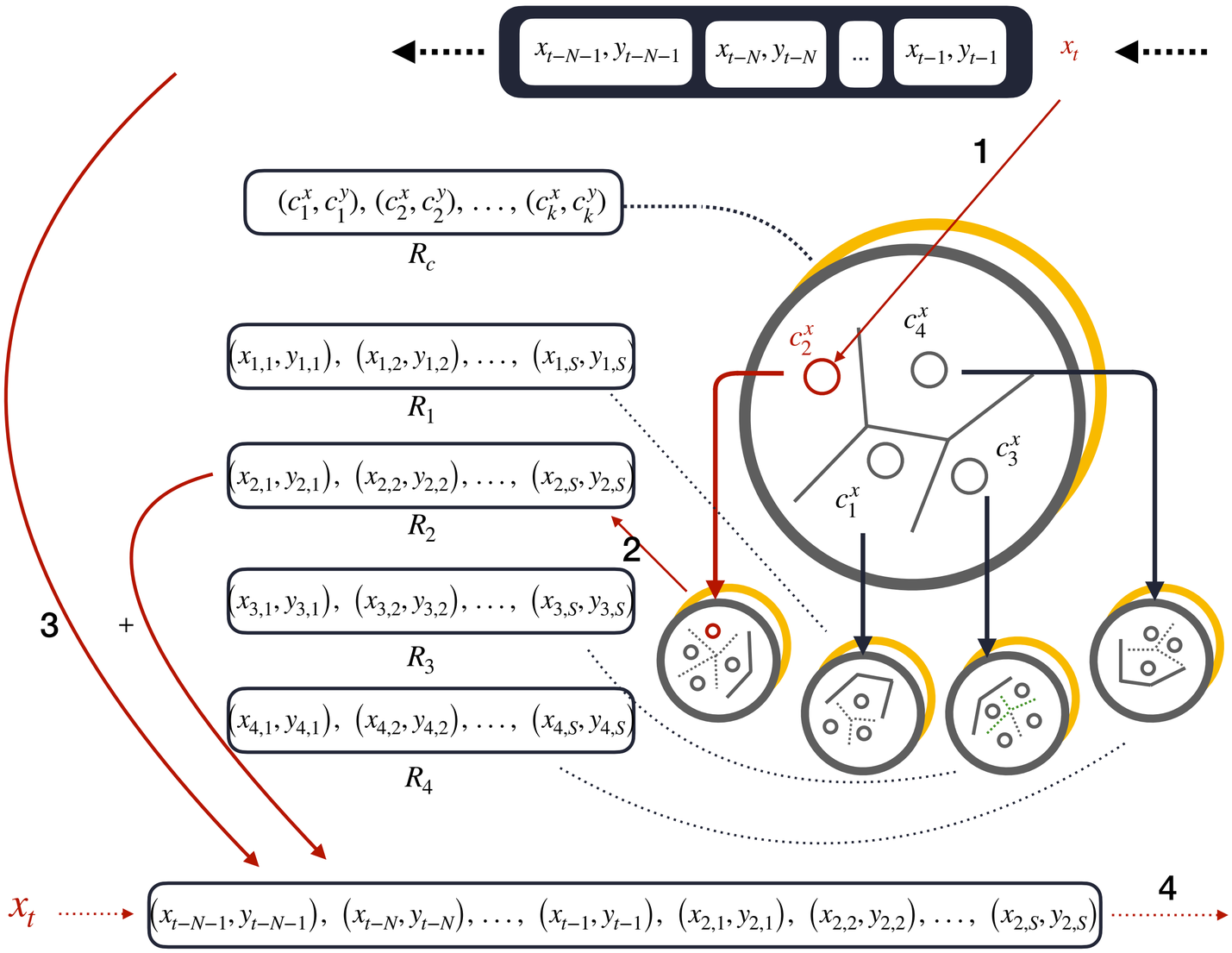
- Basé sur une structure d'arbre et d'une mémoire à long terme sous forme d'un réservoir d'exemples construit par co-clustering sur l'espace des labels vers l'espace des attributs
- Et d'une fenêtre glissante en guise de mémoire à court terme

Améliorer notre **modèle OMk** pour résoudre le problème de **classification extreme multi-labels en flux**

Merci de votre attention

ANNEXE

OMk - Prédiction



1, chercher le plus proche centroides de x_t

2, identifier le réservoir associé au centroïde trouvé

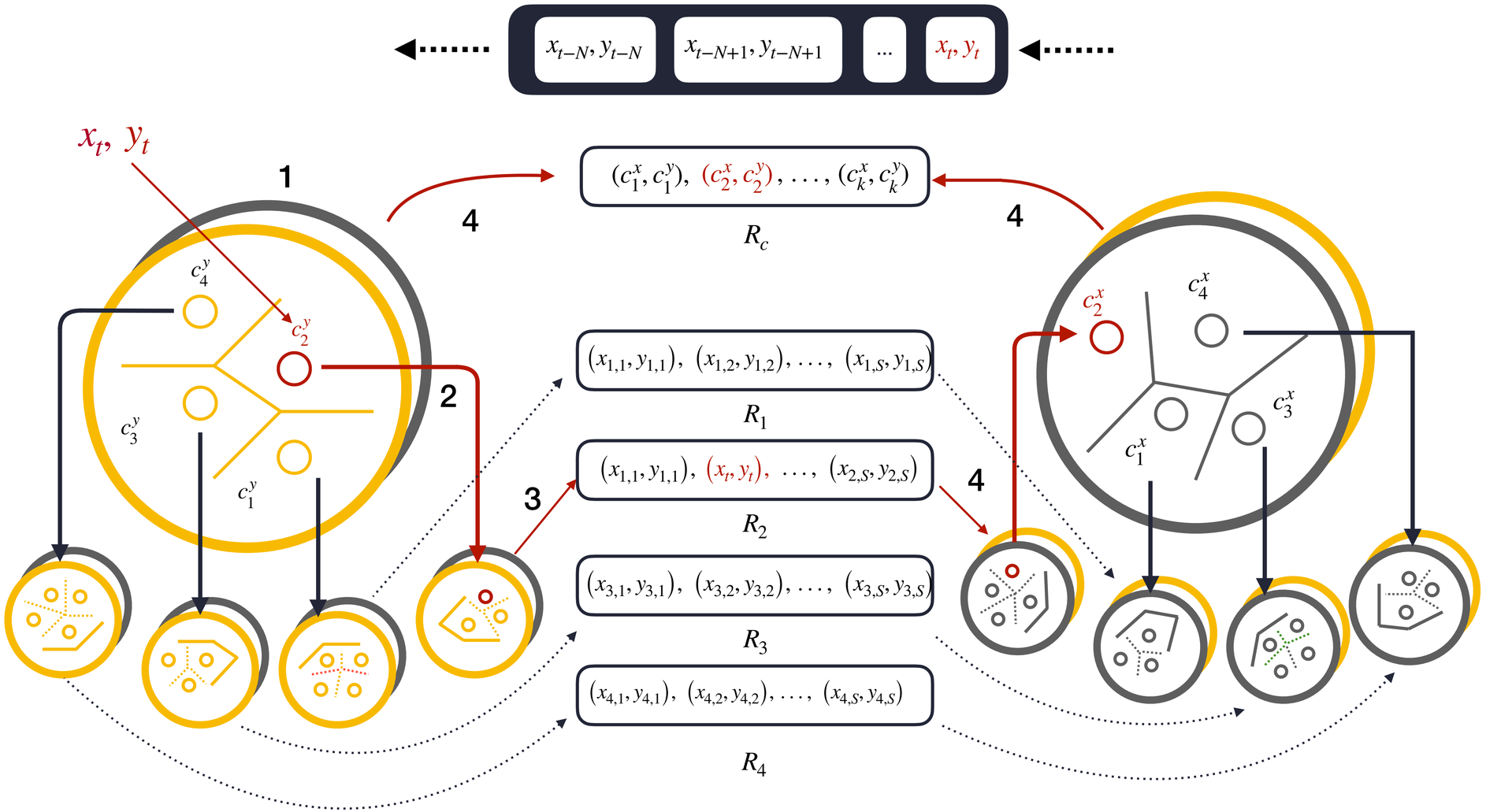
3, fusionner les deux mémoires (long et court termes)

4, chercher les n plus proches voisins de x_t

5, voter pour les labels y_t à associer à x_t

x_t → $(x_{t-N-1}, y_{t-N-1}), (x_{t-N}, y_{t-N}), \ldots, (x_{t-1}, y_{t-1}), (x_{2,1}, y_{2,1}), (x_{2,2}, y_{2,2}), \ldots, (x_{2,S}, y_{2,S})$ **4** → $y_{p,1}, y_{p,2}, \ldots, y_{p,n}$ **5** Vote → Prédiction

OMk - Apprentissage



1, chercher le plus proche centroide de y_t

2, identifier le réservoir associé au centroïde

3, ajouter la nouvelle donnée au réservoir

4, mettre à jour les centroïdes sur X et Y

Mesures d'évaluation

La mesure F1: la moyenne harmonique de la *precision* et du *recall* - basée sur les exemples ($F1_{ex}$) et les labels ($F1_{Micro}$, $F1_{Macro}$) [Pereira et al., 2018].

$$F1 = \frac{2 * precision * recall}{precision + recall}$$

Basée sur les exemples:

$$precision_{ex} = \frac{1}{t}\sum_{i=1}^{t}\frac{|p_i \cap y_i|}{y_i} \qquad recall_{ex} = \frac{1}{t}\sum_{i=1}^{t}\frac{|p_i \cap y_i|}{p_i}$$

Basée sur les labels: Les mesures basées sur les labels dépendent de *TP*, *FP*, *TN* et *FN* pour chaque label.

$$TP_j = \sum_{i=1}^{t}\{p_{i,j} == 1 \wedge y_{i,j} == 1\} \qquad FP_j = \sum_{i=1}^{t}\{p_{i,j} == 1 \wedge y_{i,j} == 0\}$$

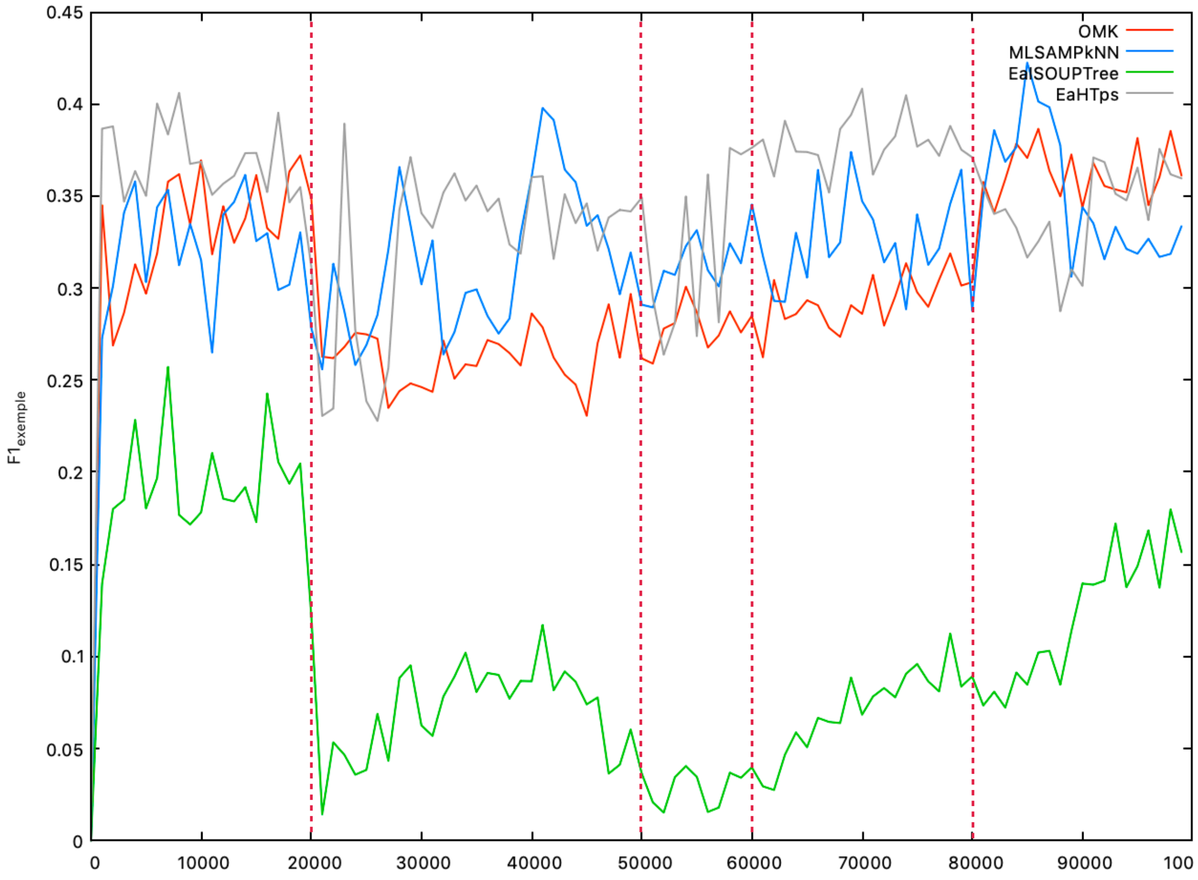
$$TN_j = \sum_{i=1}^{t}\{p_{i,j} == 0 \wedge y_{i,j} == 0\} \qquad FN_j = \sum_{i=1}^{t}\{p_{i,j} == 0 \wedge y_{i,j} == 1\}$$

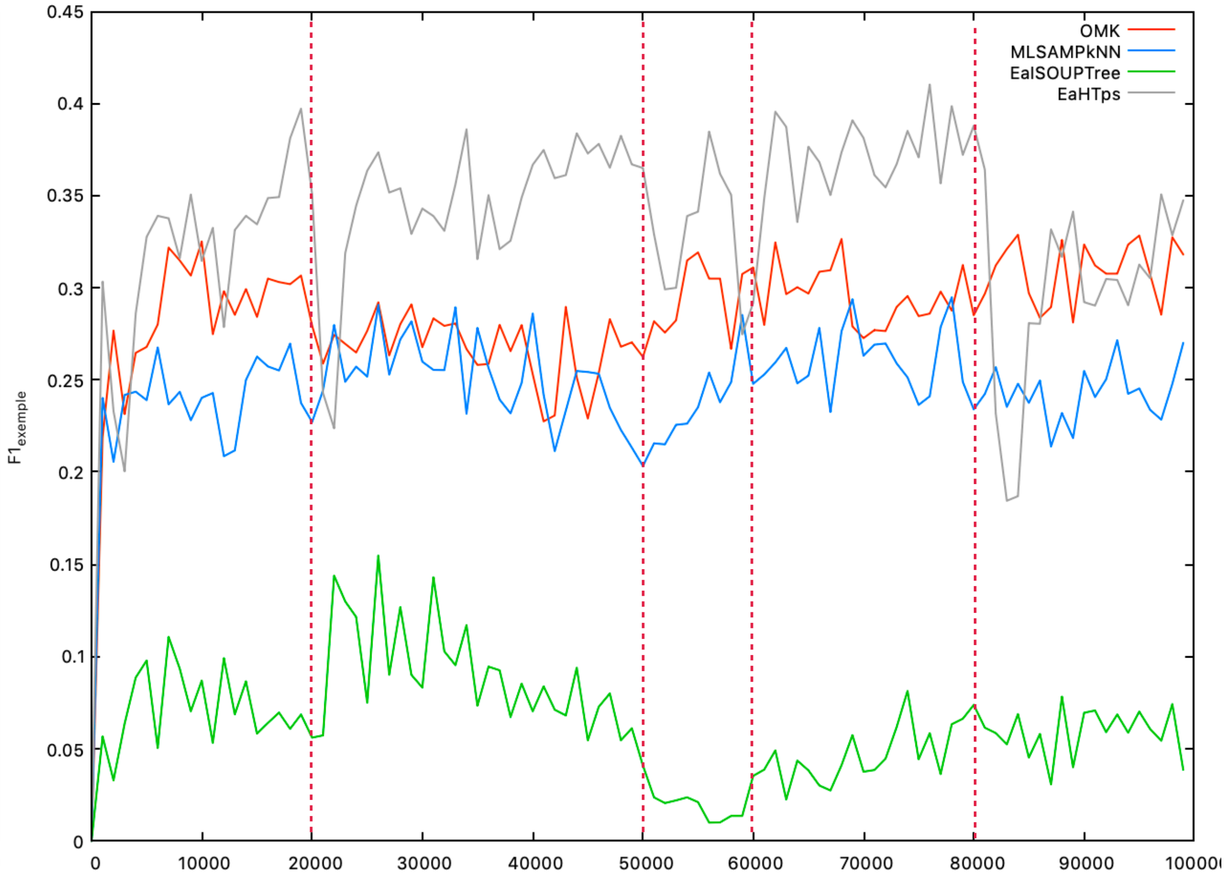
$$precision_{macro} = \frac{1}{L}\sum_{j=1}^{L}\frac{TP_j}{TP_j + FP_j} \qquad recall_{macro} = \frac{1}{L}\sum_{j=1}^{L}\frac{TP_j}{TP_j + FN_j}$$

$$precision_{micro} = \frac{\sum_{j=1}^{L} TP_j}{\sum_{j=1}^{L} TP_j + \sum_{j=1}^{L} FP_j} \qquad recall_{micro} = \frac{\sum_{j=1}^{L} TP_j}{\sum_{j=1}^{L} TP_j + \sum_{j=1}^{L} FN_j}$$

Résultats expérimentaux

les performances au cours de temps sur Data Stream Non-stationnaire (LD Drift)





L'algorithme EaHTps est performant, et OMK obtient des performances prometteuses