Atelier CluCo :

Clustering et Co-clustering

Organisateurs :

Vincent Lemaire (Orange Labs),
Pascal Cuxac (CNRS-inist),
Jean-Charles Lamirel (Loria)

Organisé conjointement à la conférence EGC
(Extraction et Gestion des Connaissances)
le 19 janvier 2016 à Reims

Editeurs :

Vincent Lemaire - Orange Labs
2 avenue Pierre Marzin, 2300 Lannion
Email : vincent.lemaire@orange.com

Pascal Cuxac - INIST - CNRS
2 allée du Parc de Brabois, CS 10310, 54519 Vandoeuvre les Nancy Cedex
Email : pascal.cuxac@inist.fr

Jean-Charles Lamirel - LORIA - SYNALP Research Team
Campus Scientifique, BP. 239, 54506 Vandoeuvre les Nancy Cedex
Email : jean-charles.lamirel@loria.fr

———

Publisher:

Vincent Lemaire, Pascal Cuxac, Jean-Charles Lamirel
2 avenue Pierre Marzin
22300 Lannion

Lannion, France, 2016

PRÉFACE

La classification non supervisée ou clustering est de nos jours largement utilisée dans un nombre croissant d'applications dans des domaines aussi divers que l'analyse d'image, la reconnaissance de formes , la fouille de textes, la gestion de la relation client, la veille scientifique ou technologique, la bio-informatique, la recherche d'information, l'analyse de réseaux sociaux...

Bien que le clustering forme un domaine de recherche en soi, avec une longue histoire, et d'innombrables méthodes, de nombreuses questions se posent toujours, telles que par exemple:

- quels sont les bons paramètres : nombre de classes versus finesse d'analyse ?
- comment estimer la qualité d'un clustering, l'importance des variables explicatives ?
- les classes doivent-elles être strictes, floues, ou recouvrantes ?
- comment rendre un clustering robuste et résistant au bruit ?
- comment évaluer l'évolution temporelle du déploiement d'un clustering ?
- ...

L'objectif de cet atelier est de favoriser des présentations et des discussions plutôt que de se focaliser sur des articles écrits complets. La soumission de prises de position bien articulées, d'expériences industrielles et de travaux en cours sont les bienvenus et privilégiés. Des contributions portant sur l'intérêt pratique des travaux, qu'elles viennent de l'industrie ou du monde académique, ou présentant des collaborations entre les deux seraient appréciées.

Le but est le partage d'expérience et de savoir sur les problématiques liées au clustering (coclustering). Le but est aussi de vous (industriels et/ou universitaires) permettre de présenter des problèmes non résolus avec les méthodes de l'état de l'art et/ou les logiciels sur étagères.

V. Lemaire — Orange Labs
P. Cuxac — cnrs-inist
J.-Ch. Lamirel — Loria

Membres du comité de lecture

Le Comité de Lecture est constitué de:

Violaine Antoine (ISIMA-LIMOS)
Gilles Bisson (LIG)
Alexis Bondu (EDF RD)
Marc Boullé (Orange Labs)
Laurent Candillier (Expertise lcandillier.free.fr)
Fabrice Clérot (Orange Labs)
Guillaume Cleuziou (LIFO)
Antoine Cornuéjols (AgroParisTech)
Pascal Cuxac (INIST-CNRS)
Nicolas Dugué (LORIA-SYNALP)
Patrick Gallinari (LIP6)
Nistor Grozavu (LIPN)
Romain Guigoures (Data Scientist, Zalando)
Pascale Kuntz-Cosperec (Polytech'Nantes)
Nicolas Labroche (Université de Tours)
Jean-Charles Lamirel (LORIA-SYNALP)
Mustapha Lebbah (LIPN)
Vincent Lemaire (Orange Labs)
Jacques-Henri Sublemontier (CEA-LIST)
Fabien Torre (Lille 3)
Christel Vrain (LIFO)

TABLE DES MATIÈRES

Exposé Invité

Un système d'apprentissage semi-supervisé pour la caractérisation d'articles de mode
Romain Guigoures 1

Usages multiples des modèles de grille
Dominique Gay 3

Session Exposés

Décomposition et classification de données fonctionnelles pour l'analyse de la consommation d'eau
Allou Samé, Zineb Noumir, Nicolas Cheifetz, Anne-Claire Sandraz et Cédric Féliers . 5

Construction incrémentale de graphes de voisinages pour le traitement de flux de données
Ibrahim Louhi, Lydia Boudjeloud-Assala et Thomas Tamisier 17

Démonstration logiciel

Un outil pour la classification à base de clustering pour décrire et prédire simultanément
Vincent Lemaire et Oumaima Alaoui Ismaili 25

Table Ronde

Index des auteurs **27**

Romain Guigoures (Zalando)
"Un système d'apprentissage semi-supervisé pour la caractérisation d'articles de mode"

Zalando est le leader de la vente en ligne de vêtements, chaussures et accessoires en Europe. Chaque année, ce sont plus de 400 mille nouveaux articles qui sont mis en vente. Si un tel nombre de produits offre un large choix aux consommateurs, il complique également la navigation sur le site web. C'est la raison pour laquelle il est primordial d'avoir une description très précise de chacun des articles, afin que l'utilisateur puisse, de manière simple et rapide, trouver l'article qu'il recherche.

Les descripteurs caractérisant un article sont variés. Ils peuvent être objectifs, comme la couleur d'un article, le motif sur une chemise ou encore le type de talon d'une paire de chaussures. Mais ils peuvent également être subjectifs et définir un style, comme par exemple les chemises de bûcherons ou le style hippster.

Lors de la création d'un nouveau descripteur, il n'est pas envisageable de parcourir manuellement plusieurs milliers d'articles. C'est pourquoi un système d'apprentissage semi-supervisé a été mis en place. Il consiste dans un premier temps à demander à un utilisateur d'étiqueter manuellement quelques articles pour apprendre au fur et à mesure à reconnaître ce que les articles sélectionnés ont en commun. À chaque itération, un apprentissage actif sélectionne un ensemble d'articles pour lesquels l'algorithme manque d'informations et demande à l'utilisateur de les étiqueter manuellement. Lorsque l'algorithme dispose de suffisamment d'informations, l'ensemble des articles est automatiquement étiqueté.

Pour comprendre ce qui lie les articles sélectionnés par l'utilisateur, l'algorithme analyse plusieurs attributs comme la distribution des couleurs ou encore des mesures de similarités construites à partir d'approches de coclustering entre articles et consommateurs, de traitement de l'image et de deep learning. Ces attributs peuvent être utilisés seuls ou combinés. Les attributs basés sur les images elles-mêmes vont permettre de découvrir ce qui lie des descripteurs objectifs, tandis que les attributs basés sur les données de navigation vont permettre de comprendre des caractéristiques plus subjectives.

Un tel outil permet à l'utilisateur de diviser par près de 200 le nombre d'articles à parcourir lors du processus de création d'un nouveau descripteur.

Note : Zalando est une entreprise de commerce électronique allemande, spécialisée dans la vente de chaussures et de vêtements, basée à Berlin. Créée en 2008 par Rocket Internet, elle est présente dans quatorze pays européens.

"Usages multiples des modèles en grille"
Dominique Gay (Université de la Réunion)

Les modèles en grille permettent de manière efficace, rapide et sûre d'évaluer la probabilité jointe d'un ensemble de variables en apprentissage non-supervisé. Les modèles en grille s'appuient sur un partitionnement de chaque variable en intervalles dans le cas numérique ou en groupe de valeurs dans le cas catégoriel. Le résultat de ces partitions univariées forme une partition multivariée de l'espace de description, alors constituée d'un ensemble de cellules. Cette partition multivariée, qu'on appellera grille (de données), est un estimateur non-paramétrique constant par morceaux de la probabilité jointe. La meilleure grille peut être obtenue en utilisant une approche Bayesienne de sélection de modèles dépendant des données via des algorithmes combinatoires efficaces. Les modèles en grille sont déjà exploités pour de nombreuses tâches de fouille de données, e.g., dans le cas non-supervisé pour le coclustering appliqué à des données texte, de graphes, fonctionnelles, séquentielles ou encore temporelles...).

Décomposition et classification de données fonctionnelles pour l'analyse de la consommation d'eau

Allou Samé*, Zineb Noumir*,**
Nicolas Cheifetz**, Anne-Claire Sandraz**, Cédric Féliers**

*IFSTTAR, COSYS, GRETTIA, F-77447 Marne-la-Vallée, France
prenom.nom@ifsttar.fr
**Veolia Eau d'Ile-de-France, Le Vermont, 28 Boulevard de Pesaro
92739 Nanterre Cedex
prenom.nom@veolia.com

Résumé. Cet article décrit une nouvelle méthodologie permettant d'extraire des profils-types de consommation à partir de séries temporelles collectées via des compteurs intelligents sur un réseau de distribution d'eau potable. L'approche proposée opère en deux étapes. Dans un premier temps, un modèle de décomposition additif de type tendance-saison-bruit est appliqué à chacune des séries afin d'en extraire des patterns saisonniers reflétant les habitudes routinières de consommation. Dans un second temps, un algorithme de classification fonctionnelle basé sur un modèle de régression de Fourier est mis en place afin de regrouper les patterns saisonniers en classes homogènes. Des expérimentations effectuées sur une base de données réelles ont montré tout le potentiel de l'approche proposée.

1 Introduction

Face à la préoccupation croissante suscitée par les problèmes environnementaux et ceux liés à la gestion des ressources comme l'eau et l'électricité, le concept des réseaux intelligents (Smart Grids) est de nos jours en pleine émergence. Il repose sur le déploiement de compteurs intelligents (Smart Meters) dans les villes, qui permettent désormais la collecte massive de données sur la consommation réelle des ménages, à une fréquence journalière voire horaire. L'analyse exploratoire de ces données doit permettre d'identifier les principaux profils d'usage, qui pourront eux-mêmes servir de base à des méthodes de détection de fuites et, à termes, contribuer à une meilleure gestion des réseaux.

Si la bibliographie fait état de plusieurs travaux de recherche sur la prévision et l'analyse exploratoire de données de consommation électrique, peu de travaux sont consacrés à la classification non supervisées de données issues de panels de compteurs d'eau. On peut à ce sujet faire référence aux travaux de McKenna et al. (2014) qui se basent sur le modèle de mélange gaussien, non pas pour regrouper les données en classes, mais pour en extraire des caractéristiques qui sont ensuite classifiées par l'algorithme des k-means (MacQueen, 1967). En suivant la même démarche, García et al. (2015) ont proposé une méthode consistant à extraire un vec-

teur moyen résumant les caractéristiques hebdomadaires de chaque compteur, puis à appliquer l'algorithme des k-means sur l'ensemble des vecteurs moyens.

Cet article propose une nouvelle méthodologie pour la classification non supervisée de séries temporelles saisonnières télé-relevées sur des compteurs d'eau. Compte tenu de la nature potentiellement continue des index de consommation d'eau, nous nous inspirons des méthodes de classification de données fonctionnelles (courbes) (Jacques et Preda, 2014; Samé et al., 2011), notamment celles sur les mélanges de régressions (Gaffney et Smyth, 1999). Notre approche repose sur deux étapes dans lesquelles les séries de Fourier sont exploitées pour prendre en compte la double saisonnalité (journalière et hebdomadaire) des données. Un modèle de décomposition additif combinant la tendance générale, la saisonnalité et le bruit est d'abord défini pour extraire de chaque série un profil saisonnier résumant les habitudes de consommation. Dans un second temps, un modèle de mélange de régressions spécifique à base de séries de Fourier (FReMix) est proposé pour regrouper les profils saisonniers en classes. Le choix des bases de Fourier pour représenter les séries temporelles de consommation d'eau est motivé ici par la connaissance a priori de leurs composantes fréquencielles. Pour la classification de séries sujettes à des changements de fréquence, les représentations à base d'ondelettes (Mallat, 1989) constituent un cadre plus adapté (Ray et Mallick, 2006; Vlachos et al., 2003; Antoniadis et al., 2013).

L'organisation de l'article est la suivante : la section 2 développe la méthode d'extraction de patterns saisonniers et la section 3 est dédiée à l'algorithme de clustering fonctionnel. La section 4 est quant-à-elle consacrée à l'application de la méthodologie proposée sur des données réelles de consommation d'eau, collectées en Ile de France.

2 Extraction de patterns saisonniers

L'application directe des méthodes de clustering fonctionnelles aux données de consommation d'eau conduit généralement à des classes reflétant principalement les niveaux de consommation. Souhaitant principalement dans cette étude caractériser les habitudes plutôt que les niveaux de consommation, nous proposons d'extraire de chaque série un profil saisonnier à partir d'un modèle de décomposition.

Les n séries temporelles à classifier sont désignées par $(\mathbf{y}_1, \ldots, \mathbf{y}_n)$, où chaque série $\mathbf{y}_i = (y_{i1}, \ldots, y_{iT})$ correspond aux consommations horaires relevées sur le compteur indexé par i, avec $y_{it} \in \mathbb{R}$. Nous supposons implicitement que toutes les séries sont alignées sur la même grille temporelle indexée par les entiers $\{1, \ldots, T\}$.

2.1 Décomposition à base de séries de Fourier

Le modèle de décomposition utilisé pour extraire de chaque série $\mathbf{y}_i$ son profil saisonnier est le suivant :

$$y_{it} = f_{it} + x_{it} + d_{it} + \varepsilon_{it}, \tag{1}$$

où

- f_{it} est la tendance globale de la série que nous modélisons de manière non paramétrique en utilisant une moyenne mobile (Gourieroux et Monfort, 1997).

– x_{it} est la composante saisonnière. Comme les séries de consommation d'eau sont sujettes à des périodicités journalière et hebdomadaire, nous utilisons la modélisation suivante exploitant une base de Fourier (De Livera et al., 2011) :

$$x_{it} = \sum_{j=1}^{q_1} \left(\alpha_{2j-1} \cos(\frac{2\pi jt}{24}) + \alpha_{2j} \sin(\frac{2\pi jt}{24}) \right) + \sum_{j=1}^{q_2} \left(\alpha_{2q_1+2j-1} \cos(\frac{2\pi jt}{168}) + \alpha_{2q_1+2j} \sin(\frac{2\pi jt}{168}) \right), \quad (2)$$

où q_1 et q_2 sont respectivement les nombre de termes trigonométriques dédiés à la modélisation des saisonnalités journalière et hebdomadaire, et les α_ℓ $(\ell = 1, \ldots, 2(q_1 + q_2))$ sont les coefficients à estimer. Notons que cette modélisation trigonométrique a l'avantage de nécessiter peu de paramètres par rapport aux approches utilisant des variables indicatrices comme régresseurs (De Livera et al., 2011).

– d_{it} est la composante permettant de gérer l'effet des jours fériés en France, comme le 1er janvier ou le 1er mai. Elle est définie par

$$d_{it} = \sum_{j=1}^{24} \gamma_j \delta_{tj}, \quad (3)$$

où $\delta_{tj} = 1$ si t correspond à l'heure j d'un jour férié et $\delta_{tj} = 0$ dans le cas contraire ; les γ_j $(j = 1, \ldots, 24)$ sont les coefficients à estimer ;

– ε_{it} est un bruit supposé gaussien centré.

Pour rester conforme aux hypothèses d'additivité et de gaussianité du modèle de décomposition défini par l'équation (1), chaque série temporelle a été remplacée par son logarithme. Cette transformation est similaire à celle de Box et Cox (1964).

2.2 Estimation des paramètres et usage pratique du modèle

Pour chaque série $\mathbf{y}_i$, l'estimation de la tendance $(f_{it})_{t=1,\ldots,T}$ est réalisée en utilisant une moyenne mobile (Gourieroux et Monfort, 1997). L'ordre de périodicité le plus élevé étant pair, nous estimons donc la tendance à l'aide d'une moyenne mobile centrée d'ordre 168 (Gourieroux et Monfort, 1997; Shumway et Stoffer, 2011).

Pour un couple (q_1, q_2) fixé, une fois la tendance estimée, les coefficients α_j et γ_j sont estimés par la méthode des moindres carrés associée à une régression linéaire multiple de $(y_{it} - f_{it})$ par rapport aux régresseurs $\cos(\frac{2\pi jt}{24})$, $\sin(\frac{2\pi jt}{24})$, $\cos(\frac{2\pi jt}{168})$, $\sin(\frac{2\pi jt}{168})$ et δ_{tj}.

La sélection du couple (q_1, q_2) est réalisée à partir du critère BIC (Schwarz, 1978) ; ce qui nous a conduit à utiliser le couple $(4, 16)$ (voir Section 4). On peut facilement montrer que ce critère peut s'écrire, à une constante additive près,

$$\mathrm{BIC}(q_1, q_2) \quad = \quad T \log(\mathrm{SSE}/T) + 2(q_1 + q_2) \log(T), \quad (4)$$

avec $\mathrm{SSE} = \sum_{t=1}^{T} (y_{it} - \widehat{f}_{it} - \widehat{s}_{it} - \widehat{d}_{it})^2$, où $\widehat{f}_{it}$, $\widehat{s}_{it}$ et $\widehat{d}_{it}$ sont les composantes estimées. La figure 1 illustre la décomposition obtenue sur une série réelle de consommation d'eau.

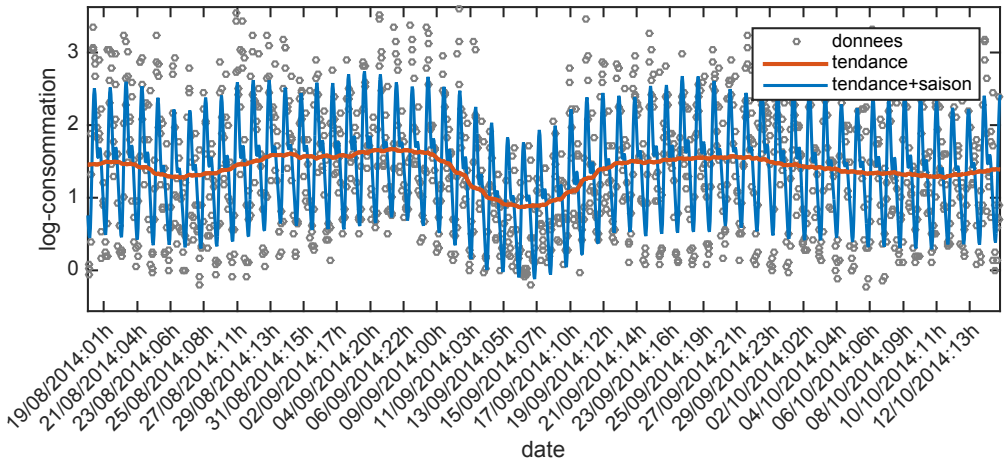

FIG. 1 – *Composantes tendancielle et saisonnière extraites d'une série de consommation*

Les points représentent la série initiale et les traits continus les composantes tendancielle et saisonnière.

Pour chaque série $\mathbf{y}_i$, les composantes du modèle défini par l'équation (1) sont ainsi estimées et le profil saisonnier périodique défini par $\mathbf{x}_i = (x_{i1}, \ldots, x_{im})$, avec $m = 168$, est extrait. Le choix a également été fait d'uniformiser l'amplitude des profils saisonniers en appliquant la normalisation suivante suggérée dans (Gaffney, 2004) :

$$x_{it} \leftarrow \frac{x_{it} - (1/m)\sum_{j=1}^{m} x_{ij}}{\sigma(\mathbf{x}_i)}, \tag{5}$$

où $\sigma(\mathbf{x}_i)$ est l'écart type de $\mathbf{x}_i$.

Les profils saisonniers $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ extraits des séries $(\mathbf{y}_1, \ldots, \mathbf{y}_n)$ sont utilisés comme données d'entrée de l'algorithme de clustering qui sera présenté dans la section suivante. Notons également que la décomposition proposée peut également servir à combler d'éventuelles valeurs manquantes dans les séries, en utilisant la reconstruction $\widehat{y}_{it} = \widehat{f}_{it} + \widehat{x}_{it} + \widehat{d}_{it}$.

3 Clustering de profils saisonniers

Le modèle présenté ici est une extension du mélange de régressions polynomiales (Gaffney et Smyth, 1999), où les courbes polynomiales désignant les centres des classes sont remplacées par des séries de Fourier. Ce modèle, nommé FReMix (Fourier regression mixture model), a été choisi conformément à la modélisation adoptée dans l'étape d'extraction des profils saisonniers (voir Section 2). De manière plus générale, les polynômes de Fourier, en tant qu'approximateurs universels de fonctions, restent adaptés à la modélisation de classes à profils non linéaires, voire périodiques.

3.1 Le modèle FReMix

La densité de chaque composante du modèle FReMix est une densité gaussienne dont la moyenne est une fonction trigonométrique paramétrée par des coefficients de régression et une variance. Les fonctions trigonométriques représentent donc l'espérance de $\mathbf{x}_i$ conditionnellement aux classes. La densité de probabilité associée au modèle FReMix est ainsi définie par

$$f(\mathbf{x}_i;\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mathbf{x}_i; \mathbf{U}\boldsymbol{\alpha}_k, \sigma_k^2\mathbf{I}), \tag{6}$$

où $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_K, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K, \sigma_1^2, \ldots, \sigma_K^2)$ est vecteur des paramètres du modèle. Les π_k sont les proportions du mélange, qui vérifient $\sum_{k=1}^{K} \pi_k = 1$. Les paramètres $\boldsymbol{\alpha}_k = (\alpha_{k,1}, \ldots, \alpha_{k,2(q_1+q_2)})' \in \mathbb{R}^{2(q_1+q_2)}$ et $\sigma_k^2 > 0$ sont respectivement le vecteur des coefficients et la variance du bruit associés à la k^{e} composante du mélange. La matrice $\mathbf{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]'$ est la matrice de régression de dimension $m \times 2(q_1 + q_2)$, où, $\forall t = 1, \ldots, m$, le vecteur $\boldsymbol{u}_t \in \mathbb{R}^{2(q_1+q_2)}$ est défini par

$$\begin{aligned} \boldsymbol{u}_t &= \left(\cos(\frac{2\pi t}{24}) \ \sin(\frac{2\pi t}{24}) \cdots \cos(\frac{2\pi q_1 t}{24}) \ \sin(\frac{2\pi q_1 t}{24}) \right. && (7) \\ &\quad \left. \cos(\frac{2\pi t}{168}) \ \sin(\frac{2\pi t}{168}) \cdots \cos(\frac{2\pi q_2 t}{168}) \ \sin(\frac{2\pi q_2 t}{168}) \right)', && (8) \end{aligned}$$

et $\mathcal{N}(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ est la densité gaussienne de moyenne $\boldsymbol{\mu}$ et de matrice de covariance $\boldsymbol{\Sigma}$. Les centres de classes associés à ce modèle de mélange sont les fonctions $g_k(t) = \boldsymbol{\alpha}_k' \boldsymbol{u}_t$ qui peuvent également s'écrire

$$\begin{aligned} g_k(t) = \sum_{j=1}^{q_1} \left(\alpha_{k,2j-1} \cos(\frac{2\pi jt}{24}) + \alpha_{k,2j} \sin(\frac{2\pi jt}{24}) \right) + \\ \sum_{j=1}^{q_2} \left(\alpha_{k,2q_1+2j-1} \cos(\frac{2\pi jt}{168}) + \alpha_{k,2q_1+2j} \sin(\frac{2\pi jt}{168}) \right). \end{aligned} \tag{9}$$

3.2 Algorithme EM et aspects pratiques

Le vecteur des paramètres du modèle est estimé de manière similaire au cas du modèle de mélange de régressions polynomiales (Gaffney et Smyth, 1999), en maximisant la log-vraisemblance

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mathbf{x}_i; \mathbf{U}\boldsymbol{\alpha}_k, \sigma_k^2\mathbf{I}) \tag{10}$$

via l'algorithme Expectation-Maximization (EM) (Dempster et al., 1977; McLachlan et Krishnan, 2008; Gaffney et Smyth, 1999). L'algorithme proposé se distingue de algorithme EM gaussien classique par l'estimation, dans l'étape M, des coefficients de régression $\boldsymbol{\alpha}_k$ associés aux séries de Fourier. L'algorithme 1, appelé également EM-FReMix, détaille la procédure itérative d'estimation.

Algorithme 1 : EM-FReMix

Entrées : n séries $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$, nombre de classes K, paramètre initial $\boldsymbol{\theta}^{(0)}$
$c \leftarrow 0$
répéter

Étape E : calcul des probabilités a posteriori :

$$\tau_{ik}^{(c)} \leftarrow \frac{\pi_k^{(c)} \mathcal{N}(\mathbf{x}_i; \mathbf{U}\boldsymbol{\alpha}_k^{(c)}, \sigma_k^{2^{(c)}}\mathbf{I})}{\sum_{\ell=1}^{K} \pi_\ell^{(c)} \mathcal{N}(\mathbf{x}_i; \mathbf{U}\boldsymbol{\alpha}_\ell^{(c)}, \sigma_\ell^{2^{(c)}}\mathbf{I})}$$

Étape M : mise à jour des paramètres :

$$\begin{aligned}
\pi_k^{(c+1)} &\leftarrow (1/n)\textstyle\sum_{i=1}^{n} \tau_{ik}^{(c)} \\
\boldsymbol{\alpha}_k^{(c+1)} &\leftarrow \left[\left(\textstyle\sum_{i=1}^{n} \tau_{ik}^{(c)}\right) \sum_{t=1}^{m} \boldsymbol{u}_t \boldsymbol{u}_t'\right]^{-1} \left[\sum_{t=1}^{m} \left(\sum_{i=1}^{n} \tau_{ik}^{(c)} x_{it}\right) \boldsymbol{u}_t\right] \\
\sigma_k^{2^{(c+1)}} &\leftarrow \textstyle\sum_{i=1}^{n} \tau_{ik}^{(c)} \sum_{t=1}^{m} \left(x_{it} - \boldsymbol{u}_t' \boldsymbol{\alpha}_k^{(c+1)}\right)^2 / \left(m \sum_{i=1}^{n} \tau_{ik}^{(c)}\right)
\end{aligned}$$

$c \leftarrow c + 1$
jusqu'à *ce que la vraisemblance converge*;
Sorties : paramètre $\widehat{\boldsymbol{\theta}}$

Cet algorithme est appliqué aux données $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ avec les paramètres $(q_1, q_2) = (4, 16)$ obtenus dans l'étape d'extraction de profils saisonniers (voir Sections 2 et 4). Il est initialisé comme suit : les coefficients de régression initiaux et les variances sont calculés en effectuant une régression de Fourier séparément sur K profils tirés au hasard parmi $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$, en adoptant la méthode d'estimation décrite dans la section 2. Les proportions initiales des classes sont fixées à $\pi_k = \frac{1}{K}$. Ce processus de choix de paramètres initiaux et d'exécution de l'algorithme EM-FReMix est répété 20 fois, et les paramètres ayant fourni la log-vraisemblance la plus élevée sont finalement retenus.

Une fois que les paramètres du modèle ont été estimés, une partition des données est obtenue en affectant à chaque série $\mathbf{x}_i$ la classe k dont la probabilité a posteriori

$$\tau_{ik} = \frac{\widehat{\pi}_k \mathcal{N}(\mathbf{x}_i; \mathbf{U}\widehat{\boldsymbol{\alpha}}_k, \widehat{\sigma}_k^2\mathbf{I})}{\sum_{\ell=1}^{K} \widehat{\pi}_\ell \mathcal{N}(\mathbf{x}_i; \mathbf{U}\widehat{\boldsymbol{\alpha}}_\ell, \widehat{\sigma}_\ell^2\mathbf{I})} \tag{11}$$

est la plus élevée, où $(\widehat{\pi}_k, \widehat{\boldsymbol{\alpha}}_k, \widehat{\sigma}_k^2)$ sont les paramètres estimés par l'algorithme EM-FReMix.

Le nombre de classes est quant-à-lui sélectionné via le critère BIC (Schwarz, 1978) défini par

$$BIC(K) = -2\mathcal{L}(\widehat{\boldsymbol{\theta}}_K) + \nu_K \log(n), \tag{12}$$

où $\widehat{\boldsymbol{\theta}}_K$ et ν_K sont respectivement le vecteur des paramètres estimés et le nombre de paramètres libres pour un modèle à K classes. Pour le modèle FReMix, on a $\nu_K = 2K(q_1 + q_2 + 1) - 1$.

4 Expérimentation sur des données réelles

Le jeu de données étudié représente le volume horaire d'eau potable collecté sur $n = 10233$ compteurs répartis autour de Paris, pour une période de 490 jours (de novembre 2013 à avril 2015). Ces compteurs sont associés à des habitations individuelles et à des immeubles collectifs. La longueur des séries $(\mathbf{y}_1, \ldots, \mathbf{y}_n)$ est donc de $T = 11760$ heures. L'extraction de la composante saisonnière par l'approche décrite dans la section 2 conduit à un ensemble de profils saisonniers $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ de longueur $m = 168$ qui est classifié par algorithme EM-FReMix.

4.1 Sélection des nombres d'harmoniques q_1 et q_2 des séries de Fourier

Les nombres d'harmoniques q_1 et q_2 permettant de gérer les saisonnalités journalière et hebdomadaire (voir Section 2) a été estimé par le critère BIC donné par la formule (4). Compte tenu du nombre élevé de séries à traiter, ce critère a été évalué sur la série moyenne $\frac{1}{n}\sum_{i=1}^{n}\mathbf{y}_i$, en faisant varier q_1 de 2 à 12 et q_2 de 2 à 50. La figure 2 montre les valeurs obtenues, la couleur bleu foncé indiquant les plus petites valeurs. Le critère minimum est atteint pour le couple $(q_1, q_2) = (4, 16)$ que nous avons donc retenu dans la suite des expérimentations.

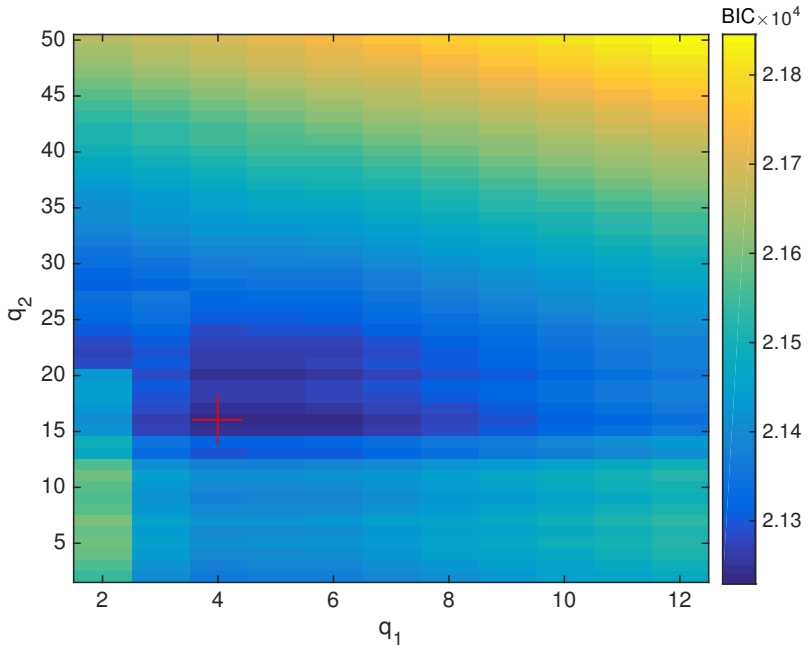

FIG. 2 – *Critère BIC en fonction des nombres d'harmoniques q_1 et q_2 ; le couple optimal $(4, 16)$ est indiqué en rouge avec le symbole $+$.*

4.2 Sélection du nombre de classes

La procédure de sélection du nombre de classes consiste à lancer l'algorithme EM-FReMix pour K variant de 1 à 20, puis à choisir le nombre de classes minimisant le critère BIC. La figure 3 met cependant en évidence une décroissance stricte du critère BIC. Le taux de décroissance n'étant pas très significatif à partir 8 classes, nous avons donc choisi ce dernier nombre de classes qui, en outre, nous a permis d'obtenir une interprétation cohérente des classes.

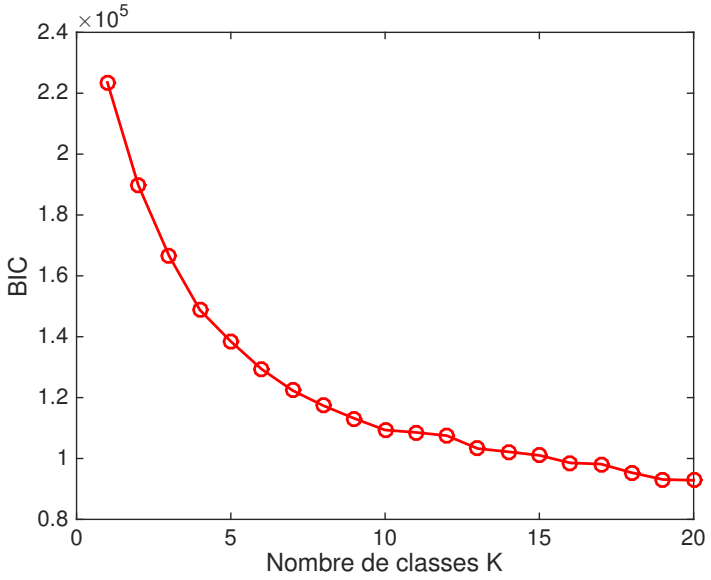

FIG. 3 – *Critère BIC en fonction du nombre de classes.*

4.3 Résultats et interprétation

La figure 4 représente les 8 classes obtenues et leurs profils hebdomadaires associés. Pour faciliter l'analyse des profils, nous les avons également représenté (voir graphiques de droite) sous la forme journalière, où les couleurs (du bleu vers le rouge) indiquent le jour de la semaine (du lundi au dimanche). Le pourcentage d'observations appartenant à chaque classe a également été évalué.

Une évaluation qualitative des résultats a permis d'attribuer les profils présentés dans la figure 4 aux catégories suivantes :

- Bureaux ou usage industriel (cluster 1) : profil moyen caractérisé par une consommation active du lundi au vendredi pendant les heures de travail, et une consommation faible le weekend.
- Usage résidentiel (clusters 3, 4, 5) : profils moyens caractérisés par des pics de consommation le matin (vers 10h) et le soir (vers 20h). Le pallier de consommation entre les deux pics peut être attribué à la présence de quelques personnes dans certaines habitations pendant les heures de travail.

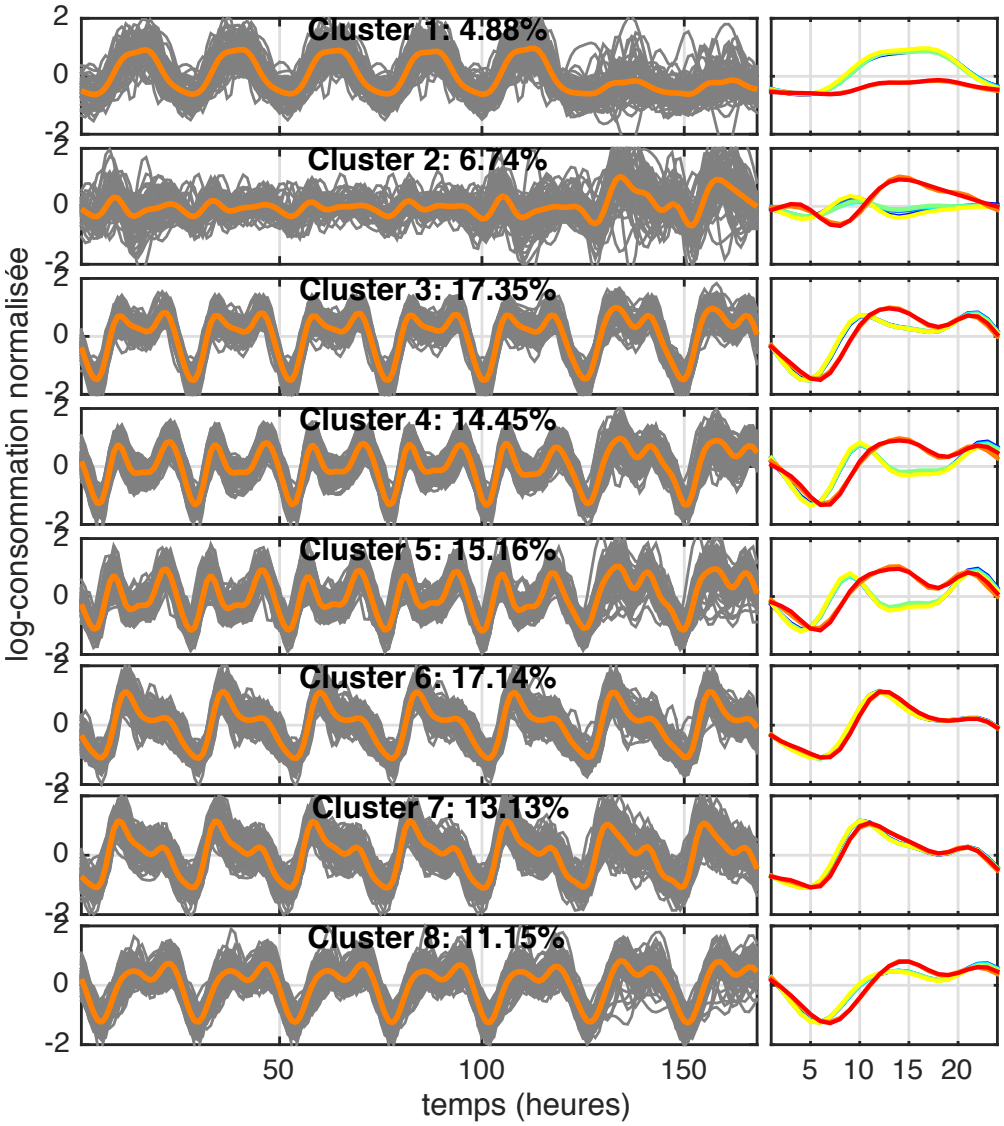

FIG. 4 – *Partitionnement des courbes saisonnières de consommation en huit classes et prototypes hebdomadaires associés (à gauche) ; prototypes sous forme journalière, où les couleurs (du bleu vers le rouge) indiquent le jour de la semaine (du lundi au dimanche) (à droite) ; par souci de lisibilité, seules 1000 courbes saisonnières sont représentées sur les figures de gauche.*

- Usage commercial (clusters 6, 7, 8) : profils moyens caractérisés par des habitudes de consommation identiques les jours ouvrés et les weekends. Ces profils pourraient également correspondre à des ménages dont les habitants sont souvent à la maison (retraités par exemple).
- Usage atypique (cluster 2) : profil moyen d'amplitude très faible mais dont les courbes associées sont très dispersées. Cette classe peut être considérée comme une classe de « bruit » constituées de courbes atypiques ne rentrant pas dans les trois catégories décrites ci-dessus. Une analyse détaillée de cette classe a révélé qu'elle était essentiellement constituée de compteurs ayant des consommations relativement faibles, ce qui explique notamment leur forte dispersion.

Il convient de préciser que l'application de l'algorithme EM-FReMix avec 4 clusters n'a pas conduit aux quatre catégories identifiées ci-dessus, ce qui peut se justifier par le type de représentation utilisé. Il a été également été observé qu'au delà de 8 clusters, les profils fournis par l'algorithme étaient quasi identiques à ceux obtenus avec 8 clusters.

5 Conclusions et perspectives

Une méthodologie générique d'extraction de profils-types à partir de séries temporelles de consommation d'eau a été proposée dans cet article. L'objectif principal de l'étude étant de caractériser les habitudes et non les niveaux de consommation, nous avons d'abord été amenés à extraire des séries initiales des patterns saisonniers à l'aide d'un modèle approprié. Ce dernier, qui s'appuie sur une décomposition combinant de manière additive la tendance, la saisonnalité et le bruit, a notamment permis de prendre en compte, par le biais des séries de Fourier, la double périodicité propre aux données étudiées. L'extraction proprement dite des profils-types (clustering) a été réalisée grâce à l'apprentissage d'un mélange de régressions de Fourier mené via un algorithme EM dédié. Huit classes ont ainsi été identifiées à partir d'une base de données réelles et une catégorie réaliste a pu être attribuée à chaque cluster.

Des expérimentations sont en cours dans l'optique de raffiner l'interprétation des classes obtenues. Elles montrent la nécessité de croiser ces résultats à des variables contextuelles de type socio-démographique. Il parait également pertinent de comparer la qualité de représentation et de partitionnement de la méthode proposée à celle issue des approches à base d'ondelettes.

Références

Antoniadis, A., X. Brossat, J. Cuglari, et J.-M. Poggi (2013). Clustering functional data using wavelets. *International Journal of Wavelets, Multiresolution and Information Processing 11*(01).

Box, G. E. P. et D. R. Cox (1964). An analysis of transformations. *Journal of the Royal Statistical Society, Series B* (26), 211–252.

De Livera, A. M., R. J. Hyndman, et R. D. Snyder (2011). Forecasting time series with complex seasonal patterns using exponential smoothing. *Journal of the American Statistical Association 106*(496), 1513–1527.

Dempster, A. P., N. M. Laird, et D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B 39*, 1–38.

Gaffney, S. et P. Smyth (1999). Trajectory clustering with mixtures of regression models. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '99, New York, NY, USA, pp. 63–72. ACM.

Gaffney, S. J. (2004). *Probabilistic Curve-Aligned Clustering and Prediction with Regression Mixture Models*. Ph. D. thesis, University of California.

García, D., D. Gonzalez, J. Quevedo, V. Puig, et J. Saludes (2015). Water demand estimation and outlier detection from smart meter data using classification and big data methods. In *New Developments in IT and Water Conference*.

Gourieroux, C. et A. Monfort (1997). *Time series and dynamic models*. Cambridge University Press.

Jacques, J. et C. Preda (2014). Functional data clustering : a survey. *Advances in Data Analysis and Classification 8*(3), 231–255.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1 : Statistics*, Berkeley, California, pp. 281–297. University of California Press.

Mallat, S. G. (1989). A theory for multiresolution signal decomposition : the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 11*(7), 674–693.

McKenna, S. A., F. Fusco, et B. J. Eck (2014). Water demand pattern classification from smart meter data. *Procedia Engineering 70*, 1121–1130.

McLachlan, G. J. et T. Krishnan (2008). *The EM Algorithm and Extensions*. Wiley, New York.

Ray, S. et B. Mallick (2006). Functional clustering by bayesian wavelet methods. *Journal of the Royal Statistical Society : Series B (Statistical Methodology) 68*(2), 305–332.

Samé, A., F. Chamroukhi, G. Govaert, et P. Aknin (2011). Model-based clustering and segmentation of time series with changes in regime. *Advances in Data Analysis and Classification 5*(4), 301–321.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics 6*, 461–464.

Shumway, R. H. et D. S. Stoffer (2011). *Time series analysis and its applications*. Springer.

Vlachos, M., J. Lin, E. Keogh, et D. Gunopulos (2003). A wavelet-based anytime algorithm for k-means clustering of time series. In *In Proc. Workshop on Clustering High Dimensionality Data and Its Applications*, pp. 23–30.

Summary

This paper describes a novel methodology for identifying relevant usage profiles from hourly water consumption series collected by smart meters located on a water distribution network. The proposed approach operates in two stages. First, an additive time series decomposition model is used in order to extract seasonal patterns from the time series, which are intended to represent the customers habits in terms of water consumption. Then, a Fourier regression

mixture-model-based algorithm is applied in order to group the extracted seasonal patterns into homogeneous clusters. Experimentations performed on a real database have shown the potential of the proposed approach.

Construction Incrémentale de Graphes de Voisinage pour le Traitement de Flux de Données

Ibrahim Louhi[*,**] Lydia Boudjeloud-Assala[*]
Thomas Tamisier[**]

[*]Université de Lorraine, Laboratoire d'Informatique Théorique et Appliquée.
{ibrahim.louhi, lydia.boudjeloud-assala}@univ-lorraine.fr
[**]Luxembourg Institute of Science and Technology.
{ibrahim.louhi, thomas.tamisier}@list.lu

Résumé. Nous présentons NG-Stream une nouvelle approche utilisant les graphes de voisinage pour le clustering des flux de données. Nous définissons une méthode de construction incrémentale d'un graphe de voisinage en fonction de l'évolution du flux. Un clustering basé sur le voisinage est appliqué sur chaque nouveau groupe et les résultats sont représentés à l'aide de visualisations spécifiques. Au lieu de traiter les nouveaux éléments un par un, nous choisissons de traiter chaque groupe de nouveaux éléments simultanément. En vue de valider l'approche, nous l'appliquons sur plusieurs jeux de données et la comparons avec divers algorithmes de clustering de flux. Les premiers résultats que nous publions sont encourageants et montrent l'efficacité de NG-Stream.

1 Introduction

Afin d'extraire des connaissances utiles à partir de données brutes, l'utilisation d'une méthode de fouille de données est incontournable. La fouille de données est considérée comme l'étape la plus importante dans le processus d'extraction de connaissances. En l'absence d'informations complémentaires sur les données, aucune technique d'apprentissage ne peut être utilisée et le traitement de données doit se faire d'une façon non supervisée. Parmi les méthodes non supervisées, le clustering essaye de trouver des groupes d'éléments homogènes appelés *clusters*, de telle sorte que les éléments de chaque cluster sont différents des autres éléments selon certains critères (Berkhin (2006)).

Dans plusieurs cas, les données sont caractérisées par un aspect temporel. Effectivement, dans plusieurs domaines les données sont générées d'une façon continue et souvent à une très grande vitesse. Le suivi de la consommation énergétique par exemple, des opérations financières ou même la géolocalisation des smartphones génèrent une série de données à une fréquence qui peut être stable ou variable. Ce type de données est connu sous le nom de flux de données. Face à ce type de données, les méthodes classiques de clustering doivent s'adapter afin de prendre en considération l'aspect temporel.

Dans la premiere section de ce papier nous présentons un bref état de l'art sur les techniques de clustering des flux de données. Ensuite dans la section suivante, nous présentons notre propre approche NG-Stream pour un clustering de flux basé sur les graphes de voisinage.

Enfin, Nous testons notre approche et nous évaluons les résultats obtenus tout en se comparant avec quelques algorithmes de la littérature.

2 Etat de l'art

Plusieurs approches ont essayé d'adapter les méthodes classiques du clustering aux flux de données. L'algorithme STREAM (Guha et al. (2000)) découpe le flux en fenêtres, *k* médianes sont choisies dans chaque fenêtre, ensuite chaque élément est affecté à sa médiane la plus proche. Quand le nombre des médianes atteint un seuil *m*, *k* médianes sont choisies parmi l'ensemble des *m* médianes. Ainsi de suite, à chaque fois que le seuil *m* est atteint sur un niveau *n*, des médianes sont désignés au niveau *n+1*. La limite de l'algorithme STREAM réside dans son insensibilité face à l'évolution du flux dans le temps. Les clusters ne prennent pas en considération l'aspect temporel.

L'algorithme CLUSTREAM (Aggarwal et al. (2003)) apporte une solution à cette limite. Afin de permettre de garder une trace des clusters obtenus antérieurement dans le flux, il utilise deux types de clustering. Un micro-clustering online (en temps réel) et un macro-clustering offline (en temps différé). Le micro-clustering stocke un résumé statistique sur le flux de données (les clusters et leur position temporelle dans le flux), le macro-clustering utilise ce résumé de données pour fournir le résultat obtenu par le clustering à n'importe quel point temporel du flux. CLUSTREE (Kranen et al. (2009)) est un algorithme, basé aussi sur le micro-clustering, qui s'adapte à la vitesse du flux. Le processus d'affectation des nouveaux éléments aux micro-clusters change suivant la vitesse du flux. Ce type d'algorithmes dépend fortement de la fréquence avec laquelle ils stockent le résumé statistique sur le flux. Sachant que les anciennes données sont stockées avec un minimum de détails, ce qui peut impliquer une imprécision sur l'état des clusters au niveau des anciens points temporels.

Les approches basées sur la densité essayent de diviser l'ensemble de données en régions denses. Ces dernières sont déterminées en se basant sur un seuil de densité. Vu que la construction d'un profil de densité nécessite de multiples itérations sur les données, le challenge dans le cas des flux de données est d'estimer la densité avec un seul parcours (ou quelques parcours au plus). L'algorithme DENSTREAM (Cao et al. (2006)) combine le micro-clustering avec l'estimation de densité. La première étape consiste à définir un élément de départ. Ce dernier est un élément dont le poids de son voisinage (le nombre de ses voisins) est supérieur à un seuil. Une région dense est constituée de l'ensemble des voisins de cet élément. Contrairement au micro-clustering standard, DENSTREAM ne fixe pas le nombre des micro-clusters. Dans le cas où les éléments de l'ensemble de données sont très éloignés entre eux, les méthodes basées sur la densité rencontrent des difficultés à définir les régions denses.

Dans les méthodes de clustering basées sur les grilles, et comme leur nom l'indique, une structure de grille est utilisée pour estimer la densité autour de chaque élément. L'espace de données est représenté par ses dimensions, chaque dimension est divisée en plusieurs partitions. L'espace de données est donc divisé en des cellules $dimension$ x $partition$. La densité d'une cellule est définit par le nombre de ses points. Une région dense est constituée de l'union des cellules denses connectées. L'algorithme DSTREAM (Chen et Tu (2007)) essaye de déterminer les régions denses dans un ensemble de données en utilisant une grille. La différence entre la densité basée sur le micro-clustering et la densité basée sur les grilles est que dans cette dernière, la valeur de la densité est mise à jour chaque fois qu'un nouveau élément arrive, ce

qui fait qu'une cellule peut changer de statut (dense ou à faible densité). DSTREAM suppose qu'il n'est pas nécessaire de garder les cellules vides. Il utilise des méthodes pour identifier et supprimer de telles cellules. La faiblesse de cet algorithme réside dans le faite que supprimer les cellules vides adjacentes aux régions denses peut détériorer la qualité du clustering

Dans ce papier, nous essayons principalement d'adapter une méthode classique de clustering au flux de données. Nous considérons que le traitement de chaque nouvel élément séparément peut induire en erreur (il peut être affecté au mauvais cluster). Dans la perspective de trouver le meilleur résultat possible, nous proposons une nouvelle approche NG-Stream pour le traitement des flux de données, tout en prenant en considération l'aspect temporel des données. NG-Stream utilise un clustering incrémental basé sur les graphes de voisinage, et il permet également une visualisation continue des résultats du clustering.

3 Approche proposée

Pour traiter les flux de données nous proposons une nouvelle approche NG-Stream. Plutôt que de traiter chaque nouvel élément individuellement dès son arrivée, NG-Stream traite un ensemble de nouveaux éléments simultanément. Cela permet de prendre en compte les caractéristiques d'un groupe de données arrivant dans la même période.

Soit $E = \{e_1, e_2, ...\}$ l'ensemble des éléments du flux F où la cardinalité $|E|$ (La taille du flux) est inconnue. Soit $G = (V, E, p)$ un graphe pondéré non orienté qui représente les clusters du flux en temps réel, chaque noeud représente un élément, chaque arête représente une relation de voisinage entre deux éléments, et le poids d'une arête la valeur de la distance entre les deux éléments.

En premier lieu, nous attendons l'arrivée du premier groupe d'éléments. Ces éléments constituent la première fenêtre $f_1 = \{e_{1.1}, e_{1.2}, ..., e_{1.n}\}$. La taille des fenêtres $|f_n| = n$ est fixée par l'utilisateur suivant son expertise et ses préférences. Nous appliquons un clustering basé sur le voisinage sur les éléments de la première fenêtre f_1 : Nous calculons la distance entre chaque couple d'éléments de f_1 : $(e_{1.i}, e_{1.j})$ où $i \in \{1, ..., n\}$, $j \in \{1, ..., n\}$ et $i \neq j$. Nous considérons deux éléments $e_{1.i}$, $e_{1.j}$ comme voisins si la distance entre ces deux éléments est inférieure à un seuil *s*, ce dernier est également choisi par l'utilisateur. Nous considérons que chaque ensemble de voisins constitue un cluster. $C = \{c_1, c_2, ..., c_p\}$ est l'ensemble des clusters obtenus, p est le nombre des clusters de f_1. Nous déterminons ensuite le médoïde de chaque cluster : au sein d'un même cluster, nous calculons la moyenne des distances de chaque élément, l'élément ayant la moyenne la plus petite (l'élément le plus proche du reste des éléments du cluster) est considéré comme étant le médoïde du cluster. Les clusters obtenus sont représentés par le graphe G comme expliqué précédemment. Un cluster est représenté par ces éléments et par la distance entre ses éléments et le médoïde du cluster.

Les éléments du groupe suivant $f_2 = \{e_{2.1}, e_{2.2}, ..., e_{2.n}\}$ sont, dans un premier temps, traités indépendamment et avec le même processus que les éléments de f_1. De la même manière nous obtenons les clusters $C_{new} = \{c_1, c_2, ..., c_q\}$; où q est le nombre de clusters du deuxième groupe ; et nous identifions également les médoïdes des nouveaux clusters. L'ensemble des clusters C_{new} est utilisé pour mettre à jour le graphe G. Nous calculons la distance entre chaque médoïde des nouveaux clusters et les médoïdes des anciens clusters, si la distance entre deux médoïdes de clusters est inférieure au seuil s, les deux clusters sont reliés. Cela se traduit par la création d'une arête entre les noeuds représentant les deux médoïdes. Dans le cas

ou un nouveau cluster n'est similaire à aucun des anciens clusters, il est rajouté au graphe sans qu'il ne soit relié avec un autre cluster (Ce qui représente l'apparition d'un nouveau cluster dans le flux).

Ainsi de suite, chaque groupe d'éléments qui arrive participe, d'une façon continue et incrémentale, à la construction du graphe G. Affecter un groupe de nouveaux éléments aux clusters, en se basant seulement sur des éléments représentatifs des clusters (les médoïdes), évite de comparer les éléments un à un et d'augmenter la complexité. Dans le but d'éviter que le nombre d'éléments dans le graphe n'augmente au point de saturer la visualisation, nous fixons un nombre maximal d'éléments à visualiser. Au delà de ce nombre, nous supprimons les clusters qui ont disparu du flux. Un cluster disparu du flux est un cluster qui n'a pas été mis-à-jour depuis une longue période. Le graphe et ses changements sont visualisés par l'utilisateur en temps réel.

4 Expérimentations

Nous avons effectué des tests sur différents jeux de données labellisés :

— Parkinson (Little et al. (2007)) : des enregistrements vocaux décrits par 26 attributs de quarante personnes dont vingt sont atteints de la maladie de Parkinson.

— QSAR (Mansouri et al. (2013)) : les valeurs de 41 attributs (déscripteurs moléculaires) utilisés pour définir si un produit est biodégradable ou non .

— Banknote (Lichman(a) (2013)) : des données d'une procédure d'identification de billets de banque. L'ensemble de données est décrit par 5 attributs.

Chaque jeu de données est transformé en un flux de cent mille éléments. Nous comparons ensuite les résultats obtenus sur chaque flux par NG-Stream avec ceux des algorithmes CluStream, ClusTree et DStream. Deux types d'évaluations sont utilisées (Desgraupes (2013)) :

— Une évaluation externe pour comparer les clusters obtenus avec les classes réelles, les critères utilisés sont Rogers Tanimoto, Recall, rand Index, Precision, Kulczynsky, Jaccard, Fowlkes-Mallows et Czekanowski Dice. Les valeurs de ces critères sont comprises entre 0 et 1, où 1 signifie que les résultats sont identiques.

— Une évaluation interne pour évaluer les clusters obtenus par chaque algorithme en termes de compacité et séparabilité. Les critères utilisés sont Davis Bouldin, Dunn, Silhouette Calinski Harabasz. La valeur du premier critère est à minimiser, contrairement aux trois autres où plus la valeur est élevée mieux est le résultat.

Les valeurs des paramètres de chaque algorithme sont mentionnées dans la table ci-dessous (table 1). Les paramètres optimaux sont utilisés pour chaque algorithme.

	NG-Stream		Clustream	ClusTree	DStream	
	Seuil	Taille	m	maxHeigt	Gridsize	cm
Parkinson	40	100	2	1	25	25
QSAR	30	100	2	1	35	35
Banknote	3.9	100	1	0.5	7	8

TAB. 1: Paramétrage optimal de chaque algorithme pour chaque jeu de données

4.1 Evaluation externe

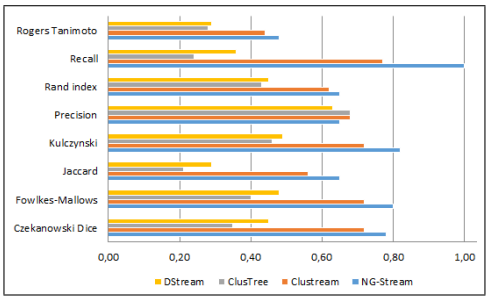

FIG. 1: l'évaluation externe sur le jeu de données Parkinson

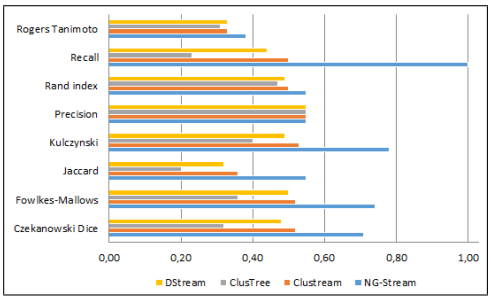

FIG. 2: l'évaluation externe sur le jeu de données QSAR

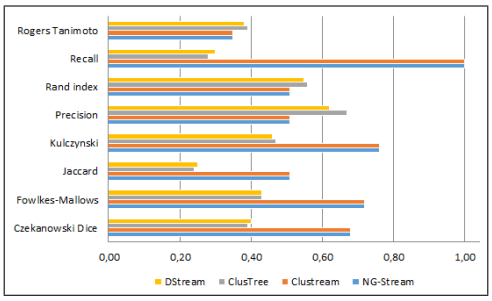

FIG. 3: l'évaluation externe sur le jeu de données Banknote

Pour la plupart des jeux de données, nos résultats sont meilleurs ou se rapprochent des résultats obtenus par les autres algorithmes.

Sur le jeu de données Parkinson, le graphique (figure 1) montre que les résultats de NG-Stream sont meilleurs que ceux des autres algorithmes, sauf selon le critère Precision, où nos résultats se rapprochent de ceux de Clustree et Clustream.

Sur la figure 2 représentant les résultats de l'évaluation externe sur le jeu de données QSAR, notre approche a également eu de meilleurs résultats que les autres algorithmes. Selon le critère Precision, nous avons obtenu les mêmes résultats que ceux des autres algorithmes.

Sur la figure 3 qui représente les valeurs des critères externes sur le jeu de donnée Banknote, nous avons eu les mêmes résultats que ceux obtenus par Clustream.

Nous remarquons que NG-Stream obtient dans la plus part des cas la valeur maximale du critère Recall. Sachant que le critère Recall calcule, (pour chaque cluster) le rapport entre le nombre d'éléments affectés au bon cluster et le nombre d'éléments attendu. Nous concluons que NG-Stream réussi à bien séparer les éléments différents.

4.2 Evaluation interne

Les résultats obtenus en utilisant les critères de l'évaluation internes sont compris dans les tables ci-dessous (table 2). La meilleure valeur pour chaque critère est marquée en caractère gras.

	NG-Stream	Clustream	ClusTree	DStream
		Davies Bouldin		
Parkinson	**0**	1.12	0.88	1.85
QSAR	**0**	41.60	21.30	23.20
Banknote	**0**	**0**	1.24	1.24
		Dunn		
Parkinson	**3.66**	0.02	0.04	0,01
QSAR	**2.08**	0	0	0
Banknote	**5.21**	**5.21**	0	0
		Silhouette		
Parkinson	**1**	0.31	0.50	-0,07
QSAR	**1**	-0.01	-0.06	-0.01
Banknote	**1**	**1**	0.27	0.27
		Calinski Harabasz		
Parkinson	0	104.16	**328.15**	12.80
QSAR	0	0.48	**1.73**	1.17
Banknote	0	0	**392.67**	**392.67**

TAB. 2: Résultats de l'évaluation selon le critère Davies Bouldin

Les résultats de l'évaluation interne montrent que NG-Stream a de bons résultats. NG-Dtream a obtenu des clusters de la même qualité que ceux de Clustream selon les critères Davies Bouldin, Dunn et Silhouette, et meilleurs que ceux obtenus par Clustree et DStream.

4.3 Visualisation du flux

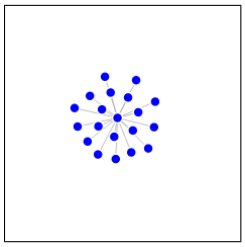

FIG. 4: Capture T1

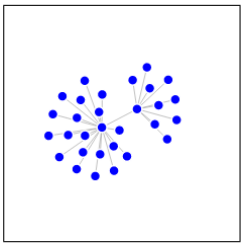

FIG. 5: Capture T2

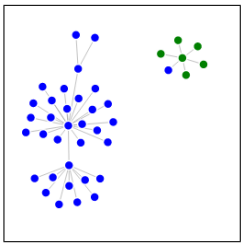

FIG. 6: Capture T3

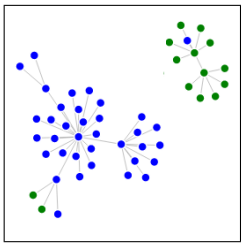

FIG. 7: Capture T4

NG-Stream propose également une visualisation en temps réel de l'évolution du flux. Afin de refléter le processus du traitement, les clusters obtenus sont représentés sous la forme de

graphes de voisinage (figures de 4 à 7). Ces figures représentent des captures à quatre instants du traitement du jeu de données Parkinson. Chaque capture est prise juste après la fin du traitement d'un nouveau groupe d'éléments à partir de T1. La taille de chaque groupe est fixée à 10 éléments. Les sous-graphes représentent les clusters détectés par NG-Stream. Les couleurs représentent les vraies classes des éléments (deux noeuds de la même couleur représentent deux éléments ayant la même classe dans l'ensemble de données d'origine).

Dans la première capture d'écran (figure 4) nous avons un seul cluster composé des éléments déjà traités du flux. Dans la deuxième capture (figure 5), un nouveau groupe est arrivé et il est constitué d'un seul cluster. le médoïde du nouveau cluster est similaire au médoïde de l'ancien cluster, donc les clusters ont été reliés.

Dans la troisième capture (figure 6) le nouveau groupe contient également un seul cluster. Par contre, cette fois-ci le médoïde du nouveau cluster ne ressemble pas au médoïde de l'ancien cluster. Donc le cluster est rajouté dans le graphe de voisinage sans qu'il ne soit relié à aucun cluster. Nous remarquons aussi qu'un des nouveaux éléments (en bleu) est affecté au cluster vert selon le critère de la distance.

Dans la quatrième capture (figure 7), le nouveau groupe contient deux clusters. Les deux nouveaux clusters sont reliés avec deux différents anciens clusters.

5 Conclusion

NG-Stream est une nouvelle approche pour le traitement des flux de données. Plutôt que de traiter chaque nouveau élément séparément, NG-Stream prend en compte les caractéristiques de tout un groupe d'éléments arrivant presque simultanément .

Dès l'arrivé du premier groupe, les clusters de ce groupe et les médoïdes des clusters sont identifiés. Les clusters sont représentés par un graphe de voisinage, ce dernier est continuellement mis-à-jour tout au long de l'évolution du flux de données. Chaque nouveau groupe d'éléments contribue à la construction incrémentale du graphe de voisinage. NG-Stream permet également de visualiser la construction du graphe en temps réel. Pour représenter le traitement effectué sur le flux de données, les clusters sont également représentés dans la visualisation par des graphes.

Lors de nos expérimentations, nous avons comparé NG-Stream à trois algorithmes de clustering de flux de données. Les évaluations internes et externes ont montré que NG-Stream a de meilleurs résultats, dans la plus part des cas, par rapport aux autres algorithmes.

En perspective nous envisageons dans un premier lieu d'étudier l'impact des valeurs des paramètres sur les résultats obtenus par NG-Stream et sur le temps d'execution. Une comparaison du temps d'exécution par rapport aux autres algorithmes de la littérature est également envisagée. Ensuite, tester NG-Stream avec des jeux de données mixtes (contenant des variables catégorielles en plus des variables numériques). Notre objectif à long terme est d'améliorer et d'intégrer cette approche dans un environnement interactif de fouille visuelle de flux de données, où l'utilisateur pourra interagir avec le processus du traitement à partir de la visualisation.

Références

Aggarwal, C. C., J. Han, J. Wang, et P. S. Yu (2003). A framework for clustering evolving data streams. In *Proceedings of the 29th international conference on Very large data bases-Volume 29*, pp. 81–92. VLDB Endowment.

Berkhin, P. (2006). A survey of clustering data mining techniques. In *Grouping multidimensional data*, pp. 25–71. Springer.

Cao, F., M. Ester, W. Qian, et A. Zhou (2006). Density-based clustering over an evolving data stream with noise. In *Proceedings of 2006 SIAM Conference on Data Mining*, Volume 6, pp. 328–339. SIAM.

Chen, Y. et L. Tu (2007). Density-based clustering for real-time stream data. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 133–142. ACM.

Desgraupes, B. (2013). Clustering indices. *University of Paris Ouest-Lab Modal'X*, 1–34.

Guha, S., N. Mishra, R. Motwani, et L. O'Callaghan (2000). Clustering data streams. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, FOCS '00, Washington, DC, USA, pp. 359–366. IEEE Computer Society.

Kranen, P., I. Assent, C. Baldauf, et T. Seidl (2009). Self-adaptive anytime stream clustering. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*, pp. 249–258. IEEE.

Lichman(a), M. (2013). Uci machine learning repository. `http://archive.ics.uci.edu/ml/datasets/banknote+authentication`.

Little, M. A., P. E. McSharry, S. J. Roberts, D. A. Costello, I. M. Moroz, et al. (2007). Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *BioMedical Engineering OnLine 6*(1), 23.

Mansouri, K., T. Ringsted, D. Ballabio, R. Todeschini, et V. Consonni (2013). Quantitative structure–activity relationship models for ready biodegradability of chemicals. *Journal of chemical information and modeling 53*(4), 867–878.

Summary

We present NG-Stream which is a new neighborhood-based approach for data streams clustering. We define an incremental construction method of a neighborhood graph based on the stream evolution. A neighborhood-based clustering is applied on each new group. The results are represented using specific visualization. Instead of processing each new element one by one, we propose to process each group of new elements simultaneously. To validate the approach, we apply it to multiple data sets and we compare it with various stream clustering approaches. The first results are encouraging and they show the effectiveness of NG-Stream.

"Un outil pour la classification à base de clustering pour décrire et prédire simultanément "

Vincent Lemaire, Oumaima Alaoui Ismaili

Cette démonstration présente un logiciel (accessible à tous) intégrant un algorithme de "k-moyennes supervisées" produisant un modèle (« déployable ») et des rapports décrivant les clusters obtenus.

Depuis quelques années, les chercheurs ont concentré leur attention sur l'étude d'un nouveau aspect d'apprentissage connu sous le nom de la classification à base de clustering (ou Supervised clustering en anglais) (e.g., (Eick et al., 2004) et (Cevikalp et al., 2007)). Les approches appartenant à ce type d'apprentissage visent à décrire et à prédire d'une manière simultanée (Alaoui Ismaili et al., 2015a). Dans ce cadre d'étude, on suppose que la classification à base de clustering est étroitement liée à l'estimation de la distribution des données conditionnellement à une variable cible. A partir d'une base de données étiquetée, ces approches cherchent à découvrir la structure interne de la variable cible afin de pouvoir prédire ultérieurement la classe des nouvelles instances.

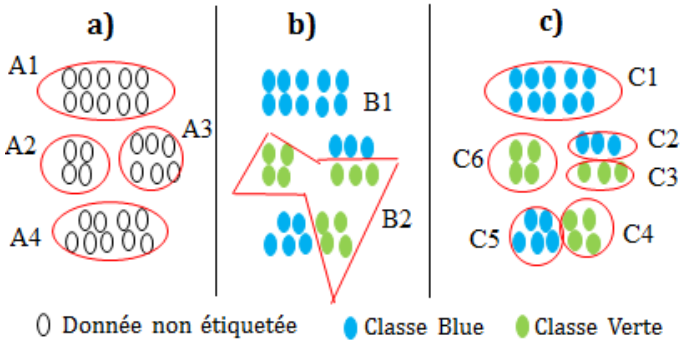

FIG. 1: Types d'apprentissage

La figure 1 illustre la différence entre les trois types d'apprentissage : le clustering standard a), la classification supervisée b) et la classification à base de clustering c). Dans la classification supervisée, la compacité des classes apprises (dans la phase d'apprentissage) n'est pas une condition importante (e.g, le groupe B2 de b)). Le clustering regroupe les instances homogènes sans tenir en compte leur étiquetage (e.g, le groupe A4 de la a)). La classification à base de clustering vise à former des groupes compacts et purs en termes de classes (e.g, les 6 groupes de c)).

La classification à base de clustering est très utile dans les domaines critiques où l'interprétation des résultats fournis par un système d'apprentissage est une condition primordiale. Elle permet à l'utilisateur de découvrir les différentes voies qui peuvent mener à une même prédiction : par exemple de découvrir que deux instances de même classe peuvent être très hétérogènes (e.g., les instances appartenant au groupe C1 et au groupe C5 de c)). La classification à base des K-moyennes est une version modifiée de l'algorithme des K-moyennes standard. Elle cherche à générer des partitions ayant un bon compromis entre la compacité des groupes formés et leurs puretés en termes de classes (voir la partie c) de la figure ci-dessus). La prédiction de la classe des nouvelles instances se réalise par la suite en se basant sur la structure interne découverte lors de la phase d'apprentissage.

Eick, C. F., N. Zeidat, et Z. Zhao (2004). Supervised clustering-algorithms and benefits. In Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on, pp. 774–776. IEEE.

Cevikalp, H., D. Larlus, et F. Jurie (2007). A supervised clustering algorithm for the initialization of rbf neural network classifiers. In Signal Processing and Communications Applications, 2007. SIU 2007. IEEE 15th, pp. 1–4. IEEE.

Alaoui Ismaili, O., V. Lemaire, et A. Cornuéjols (2015a). Classification à base de clustering ou comment décrire et prédire simultanément. In Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA).

Note : Ce logiciel se nomme

 Khiops Ennéade

Il est protégé par l'enregistrement FR.001.520021.000.S.P.2012.000.00000 to the Agency of Software Protection.

Il peut être testé gratuitement pour une période d'évaluation.

Il est diffusé commercialement (en dehors d'Orange) par la société **Predicsis**

Index

Alaoui Ismaili, Oumaima, 25

Boudjeloud-Assala, Lydia, 17

Cheifetz, Nicolas, 5

Féliers, Cédric, 5

Gay, Dominique, 2
Guigoures, Romain, 1

Lemaire, Vincent, 25
Louhi, Ibrahim, 17

Noumir, Zineb, 5

Samé, Allou, 5
Sandraz, Anne-Claire, 5

Tamisier, Thomas, 17