Détection de nouveautés en utilisant un nouveau score de détection de ”groupes-outliers“

Amine Chaibi*, Mustapha Lebbah*, Hanane Azzag*,

* {prenom.nom}@lipn.univ-paris13.fr
*Université Paris 13, Sorbonne Paris Cité - CNRS
LIPN-UMR 7030
99, av. J-B Clément - F-93430 Villetaneuse

Résumé. Dans cet article, nous introduisons une nouvelle mesure pour qualifier “l'outlier-ness“ de chaque groupe/cluster. Cette mesure, nommée GOF, est intégrée et estimée dans un processus d'apprentissage non supervisé en utilisant les cartes topologiques. Ceci permet d'apprendre la structure des données tout en fournissant un nouveau score (GOF). Ce paramètre est basé sur la densité et quantifie ainsi la particularité de chaque groupe (cluster) : plus la valeur est grande, plus le groupe est susceptible d'être un ”groupe-outlier“. GOF est utilisé par la suite comme classifieur pour le problème de détection de nouveautés.

1 Introduction

Avec la quantité croissante des données recueillies, il devient plus important et difficile de repérer les observations inhabituelles ou inattendues. Un tel comportement inattendu peut être soit non désirée (par exemple, la détection d'intrusion réseau, la surveillance des maladies), nécessitant une intervention de l'utilisateur, ou intéressant (par exemple en astronomie), ce qui conduit à une meilleure compréhension du système. La tâche de détection d'outliers joue un rôle important, puisque dans la plupart des cas, elle permet de prévenir ou d'atténuer les effets d'une situation indésirable.

Les données sont généralement un ensemble d'enregistrements décrite par un ensemble d'attributs (ou caractéristiques). D'une manière générale, les outliers peuvent être soit des outliers individuels (un seul enregistrement) ou des “groupes-outliers“, aussi appelés outliers collectifs (correspondant à des groupes d'enregistrements). Dans le cas de la détection d'outlier individuel, une approche standard consiste à créer un modèle de données normales, et de comparer les enregistrements de la base de test. Cependant, dans le cas étudié dans ce papier qui concerne les ”groupes-outliers”, plutôt que de trouver des comportements individuels anormaux (bruit ou des erreurs dans les données), nous nous sommes intéressés plus par la détection de l'émergence de nouvelles observations, qui ne peuvent être expliquées par un précédent modèle. En général, ces comportements donnent lieu à des enregistrements multiples dans un même ensemble de données formant un groupe dense et significativement isolé. Notre objectif dans ces travaux est d'utiliser la présence de ces multiples cas afin de mieux détecter

les groupes de données anormaux que nous appelons donc ”groupes-outliers“.

Plusieurs chercheurs estiment qu'il existe une forte synergie entre la détection d'outliers et de nouveautés Markou et Singh (2003a). Cependant, un défi important dans la détection d'outlier/ nouveauté est la difficulté d'obtenir des données suffisamment marquées pour caractériser les outliers/ nouveautés. Par conséquent, dans la plupart des cas, nous devons opérer dans un environnement non supervisé, où seul le comportement normal est caractérisé, et est utilisé pour détecter les écarts par rapport à celui-ci. Dans le cadre de l'exploration des données, il est généralement admis que nous avons un ensemble de données d'apprentissage suffisamment grand qui ne contient pas ou très peu de cas anormaux. Cet ensemble de données est supposé définir le comportement normal du système. Parallèlement à cela, nous avons également besoin d'une mesure (score) “d'outlier-ness” [1], qui permet de comparer les nouvelles observations aux données initiales. Ainsi, toute observation qui s'écarte de manière significative de l'habituel est signalée comme une donnée outlier ou nouvelle. Nous allons dans toute la suite de cet article utiliser le terme ”outlier“ pour les données aberrantes et “nouveauté“ pour les données inattendues (voir définitions ci-dessous).

Définition 1 : *Un **“outlier“** est une observation ou un motif d'observations qui n'est pas conforme ou “normal” [2] par rapport au comportement global de l'ensemble des données.*
Définition 2 : *Un **”groupe-outlier”** est un ensemble de données formant un groupe dense et significativement isolé.*
Définition 3 : *Une **“nouveauté“** est une donnée qui n'était pas connue dans la base d'apprentissage et qui apparaît dans la base de test. Le meilleur synonyme du terme nouveauté est ”inattendu“.*

2 Etat de l'art

2.1 Détection d'outliers

L'analyse d'outliers dans la littérature scientifique remonte à longtemps. En effet, en 1852, Peirce, le premier auteur à s'intéresser au problème des valeurs anormales disait : *”dans presque toutes les séries de données, il y a des observations qui diffèrent tellement des autres, qu'elles servent uniquement à rendre l'expérimentateur perplexe et à l'induire en erreur“.*

La détection d'outliers peut être classée en 4 catégories (Liu et al. (2010)) : les approches basées sur la distribution des données, les approches basées sur le clustering, les approches basées sur la modélisation des données et les approches basées sur la densité des données. Les approches basées sur la distribution des données sont considérées comme les plus anciennes méthodes dans ce domaine. Elles se basent essentiellement sur les modèles statistiques (boite à moustache, loi normale,...). Les approches basées sur le clustering proposent de caractériser le comportement de chaque cluster. Elles utilisent essentiellement des techniques de visualisation. Les clusters isolés qui contiennent beaucoup moins de données que les autres clusters sont considérés comme clusters outliers. Les approches basées sur la modélisation des données

1. “outlier-ness” est un terme qui signifie “aberrance“
2. Nous utilisons le mot “normal“ ici en terme simple, et non comme une référence à la distribution normale dans les statistiques

sont des méthodes très sensibles aux bruits des données en entrée. L'idée est de caractériser les données normales via des modèles prédictifs. Ainsi les outliers seront détectés suivant la déviation des modèles appris. Concernant les approches basées sur la densité des données, ce sont des méthodes qui utilisent la notion de densité des observations pour la détection d'outliers. L'approche Local Outlier Factor (LOF) (Breunig et al. (2000)) apparue dans les années 2000 reste la plus utilisée dans ce type de modèles. L'avantage de cette méthode est qu'elle ne fait aucune hypothèse sur la distribution des données. Hasan et al. (2009) ont donné une définition simplifiée de l'approche LOF. En effet, cette méthode consiste à comparer la densité locale d'une observation avec la densité moyenne de ses k-plus proches voisins (k-ppv). La valeur de LOF est calculée après avoir défini les k-ppv, la densité locale et la densité relative de chaque donnée. Plus la valeur de LOF est grande, plus la donnée est considérée comme outlier. Zengyou et al. (2003) ont utilisé LOF au niveau des clusters pour donner de l'importance aux données au niveau local. Le modèle utilisé permet d'affecter une mesure pour chaque cluster afin d'identifier les outliers. D'autres variantes plus récentes de l'algorithme LOF continuent toujours d'apparaître dans la littérature scientifique (Mennatallah Amer (2012); E. Schubert (2012)).

Dans les travaux de Gao et al. (2010), une approche appelée noyau local multi-échelle de régression est introduite. Cette méthode permet de transformer le problème classique de détection d'outliers en un problème d'apprentissage de régression non paramétrique. Pour mettre en place cette méthode, ils introduisent la notion des k-ppv. Cette méthode permet de classer les observations non étiquetées sur la base de leur similarité avec les exemples de la base d'apprentissage. L'algorithme k-ppv nécessite seulement : un entier k, une base d'apprentissage et une métrique pour la proximité. En se basant sur cette logique, Fabrizio et Pizzuti (2002) ont proposé une méthode qui a comme principe d'attribuer à chaque point, un poids. Ce poids représentent la somme des distances de ses k-ppv. Les outliers sont les points ayant les plus grandes valeurs du poids. Pour calculer ces poids, ils cherchent les k-ppv de chaque point d'une manière rapide et efficace en linéarisant l'espace de recherche à travers la courbe de remplissage de l'espace de Hilbert. L'algorithme se compose de deux phases, la première fournit une solution approchée et la deuxième retourne la solution exacte.

Les cartes auto-organisatrices aussi appelées carte de Kohonen ou carte SOM sont souvent utilisées pour la classification et la visualisation dans le but d'analyser des données structurées (Kohonen (1995)). Cai et al. (2009) ont utlisé le modèle SOM afin d'étudier le comportement des outliers dans les cartes auto-organisatrices. Les outliers sont des points anormaux qui ont des valeurs d'attributs significativement distinctes par rapport à leurs voisins. La particularité de la méthode c'est l'utilisation de la distance Mahalanolis. Cette méthode a été améliorée par les même auteurs une année après (Cai et al. (2010)) dans l'objectif de réduire la dimension des données, de conserver les informations topologiques de la carte et éventuellement de réduire l'influence des outliers potentiels.

La principale contribution que nous proposons dans ce papier est la proposition d'un score (paramètre), qui permet d'identifier les "groupes-outliers". Contrairement aux méthodes classiques, qui travaillent uniquement à l'échelle des données pour la détection d'outliers, notre approche est basée sur la densité relative de chaque groupe de données, et fournit simultanément un partitionnement des données et un indicateur quantitatif (GOF) sur "la particularité" de chaque cluster ou groupe. Ensuite, GOF a été utilisé pour la détection de nouveautés.

Plusieurs auteurs introduisent la détection de nouveautés comme un cas particulier de la

détection d'outliers (Lauer (2001); Hodge et Austin (2004); Chandola et al. (2009)).

2.2 Détection de nouveautés

La détection de nouveautés a eu une attention particulière dans le domaine de l'exploration des données en raison de ces nombreuses applications. Nous citons à titre d'exemple la détection des fraudes ainsi que la découverte d'activités criminelles dans le commerce électronique. Les modèles de détection de nouveautés ne dépendent pas seulement du type de la méthode utilisée, mais également des propriétés statistiques des données traitées.

Il existe plusieurs approches dans la littérature scientifique qui traitent le problème de la détection de nouveautés. Par ailleurs, la détection de nouveautés reste un problème difficile à résoudre car souvent les méthodes proposées sont fortement sensibles aux distibutions statistiques des données à traiter (Markou et Singh (2003a)). La détection de nouveautés est basée principalement sur deux approches : statistique et réseau de neurones (Markou et Singh (2003a,b)).

L'objectif principal des approches statistiques est la modélisation des distributions des données et l'estimation de la probabilité de ces dernières d'appartenir à une distribution. Cela induit une large sensibilité des modèles par rapport aux données (Markou et Singh (2003a)). Dans les approches statistiques pour la détection de nouveautés, on se base principalement sur la modélisation des données en fonction de leurs propriétés statistiques et on utilise ces informations pour tester si un échantillon provient de la population ou non (densité de probabilité). Les techniques utilisées varient en fonction de leur complexité. Il existe deux méthodes pour l'estimation de la densité de probabilité, les méthodes paramétriques et non paramétriques. Les méthodes paramétriques supposent que les données proviennent d'une famille de distributions connues, telle que la distribution normale. Certains paramètres sont calculés pour s'adapter à cette distribution. Dans la plupart des situations du monde réel, la distribution des données n'est pas connue. La technique intuitive la plus utilisée dans les modèles non paramétrique est l'analyse de l'histogramme. La difficulté apparaît lorsqu'il s'agit d'estimer la densité des données multidimensionnelles. Dans ce cas de figure, une façon pour l'estimation de la fonction de densité c'est l'algorithme de k-plus proche voisin (Odin et D. (2000)).

La détection de nouveautés est l'une des exigences fondamentales d'une bonne classification car parfois les données de test contiennent des informations qui n'étaient pas connues au moment de l'apprentissage du modèle. Les réseaux de neurones sont largement utilisés dans la détection de nouveautés (Markou et Singh (2003b)). Plusieurs type de réseaux de neurones peuvent traiter le problème, notamment, le perceptron multi-couches (Moya et al. (1993); Rusiecki (2012)), les cartes auto-organisatrices (Ypma et al. (1997); Xing et al. (2009)), les réseaux de Hopfield (Chandola et al. (2009)), les réseaux d'oscillation (Borisyuk et Kazanovich (2004)), etc.

Hoffmann (2007) a proposé une approche basée sur l'analyse en composantes principales (ACP). Les données d'apprentissage sont affectées à un espace de dimension infinie. L'ACP extrait les composantes principales de la distribution des données. La nouveauté est mesurée par le carré de la distance au sous-espace principal correspondant.

Les machines à vecteurs de support (SVM) sont également utilisées pour la détection de nouveautés. SVM est basé sur le concept de la détermination des hyperplans optimaux pour séparer les données de différentes classes (Vapnik (1995)). Tax et al. (2001), utilisent le principe des SVMs pour traiter le problème de détection de nouveautés. En effet, ils proposent une ap-

proche qui permet de distinguer (séparer) la classe des observations représentée par l'ensemble d'apprentissage et le reste des observations. Une méthode de détection de nouveautés est généralement évaluée en utilisant le taux des vrais positifs et l'aire sous la courbe ROC (Receiver Operating Characteristics) (Markou et Singh (2003b)). Ces mesures sont utilisées pour l'évaluation des performances d'un classifieur.

3 Cartes auto-organisatrices et référents outliers

Soit $\mathcal{A}$ l'ensemble des données $\mathbf{x}_i$ d'apprentissage, de taille N, où chaque observation $\mathbf{x}_i = (x_i^1, x_i^2, ..., x_i^j, ..., x_i^d) \in \Re^d$. Notre approche propose un apprentissage à l'aide des cartes topologiques tout en détectant les "groupes-outliers". Nous rappelons qu'un "groupe-outlier" n'est pas nécessairement un groupe aberrant ; il peut être un groupe d'intérêt, de nouveauté, etc. En fait, c'est un groupe qui a un comportement largement différent du reste des données. Ce type de groupe peut biaiser les résultats comme il peut constituer un échantillon exhaustif.

Le modèle classique des cartes auto-organisatrices se présente sous forme d'une grille possédant un ordre topologique de K cellules. Les cellules sont réparties sur les nœuds d'un maillage. La prise en compte dans la carte de la notion de proximité impose de définir une relation de voisinage topologique. L'influence mutuelle entre deux cellules c et r est donc définie par la fonction $\mathcal{K}^{\mathcal{T}}(\delta(c,r))$ où $\delta(c,r)$ est la distance de graphe entre les deux cellules c et r. Dans notre approche, chaque cellule c de la grille $\mathcal{C}$ est associée à la fois à deux paramètres : un vecteur référent $\mathbf{w}_c = (w_c^1, w_c^2..., w_c^j, ..., w_c^d)$ de dimension d et une nouvelle valeur que nous proposons d'appeler "GOF" (Group Outlier Factor). On note par la suite $\mathcal{W} = \{\mathbf{w}_c, \mathbf{w}_c \in \Re^d\}_{c=1}^K$ l'ensemble des référents et par $GOF_c \in \Re$ l'indicateur "d'outlierness" associé à chaque cellule c.

Chaque référent est associé à un sous-ensemble de données affecté à la cellule c qui est noté P_c. L'ensemble des sous ensembles forme la partition de l'ensemble des données $\mathcal{A}$, $\mathcal{P} = \{P_1, ...P_c, ..., P_K\}$ où $P_c = \{\mathbf{x}_i, \phi(\mathbf{x}_i) = c\}$.

Nous rappelons ici que la fonction d'affectation ϕ est définie de la manière suivante :

$$\phi(\mathbf{x}_i) = \arg \min_{1 \leq c \leq K} \parallel \mathbf{x}_i - \mathbf{w}_c \parallel^2$$

Chaque cellule c est associée à une valeur réelle GOF_c qui indique "l'outlier-ness" de la cellule, et qui résume en d'autre terme "l'outlier-ness" de toutes les observations $\mathbf{x}_i$ affectées à la cellule c. L'estimation de "l'outlier-ness" de chaque observation est liée à la densité. Chaque cellule de la carte est associée à un référent. La fonction $f_c(x)$ permet d'estimer la densité des données au niveau de chaque cellule c définie comme suit :

$$f_c(\mathbf{x}_i) = \exp^{-} \frac{\parallel \mathbf{x}_i - \mathbf{w}_c \parallel^2}{2\sigma^2}$$

où le paramètre σ est l'écart type standard entre les données. Le choix du paramètre σ est important et sa valeur optimale est difficile à calculer. En effet, si σ est trop grand, alors la répartition des données va influencer les valeurs de densité de tous les prototypes, les prototypes proches sont alors associés à des densités similaires, ce qui induit une diminution de la *Précision* de l'estimation. Cependant, si σ est trop petit, une grande proportion des données

(les plus éloignées des prototypes) n'influenceront pas les valeurs de la densité des prototypes, ce qui induit une perte d'information.

Dans notre cas, la densité est définie par une fonction de type gaussienne. Ainsi, nous proposons d'estimer “l'outlier-ness“ de chaque observation $\mathbf{x}$ associé à un référent $\mathbf{w}_c$ en utilisant l'expression suivante :

$$OF_c(\mathbf{x}_i) = \frac{\frac{\sum_{\mathbf{x}_j \in P_c} \frac{1}{f_c(\mathbf{x}_j)}}{|P_c|}}{\frac{1}{f_c(\mathbf{x}_i)}}$$

Ainsi, nous proposons de minimiser la fonction de coût suivante :

$$\mathcal{R}(\mathcal{W}, GOF, \phi) = \mathcal{R}(\mathcal{W}, \phi) + \mathcal{R}(GOF, \phi)$$

où

$$\mathcal{R}(\mathcal{W}, \phi) = \sum_{i=1}^{N} \sum_{c=1}^{K} \mathcal{K}^{\mathcal{T}}(\delta(\phi(\mathbf{x}_i), c)) \|\mathbf{w}_c - \mathbf{x}_i\|^2$$

et

$$\mathcal{R}(GOF, \phi) = \sum_{i=1}^{N} \sum_{c=1}^{K} \mathcal{K}^{\mathcal{T}}(\delta(\phi(\mathbf{x}_i), c)) \left(GOF_c - OF_c(\mathbf{x}_i)\right)^2$$

$$\mathcal{R}(GOF, \phi) = \sum_{i=1}^{N} \sum_{c=1}^{K} \mathcal{K}^{\mathcal{T}}(\delta(\phi(\mathbf{x}_i), c)) \left(GOF_c - \frac{\frac{\sum_{\mathbf{x}_j \in P_c} \frac{1}{f_c(\mathbf{x}_j)}}{|P_c|}}{\frac{1}{f_c(\mathbf{x}_i)}} \right)^2$$

La notion de voisinage est introduite par la fonction noyau :

$$\mathcal{K}^T(\delta(c, r)) = \exp\left(\frac{-\delta(c, r)}{T}\right)$$

Où T est la température qui varie de T_{max} à T_{min}.

Le premier terme $\mathcal{R}(\mathcal{W}, \phi)$ dépend des paramètres $\mathcal{W}$ et permet d'estimer les référents. Le deuxième terme $\mathcal{R}(GOF, \phi)$ est lié à l'estimation des valeurs GOF associées à chaque cellule. L'algorithme d'apprentissage suivant (algorithme 1) propose une solution pour la minimisation de la fonction coût en utilisant la méthode de la descente du gradient.

Chaque paramètre est “récompensé“ en augmentant de sa valeur. Cette valeur est d'autant plus importante que l'apprentissage est avancé, et que les référents représentent bien les données. Cependant, les autres seront “punies“ en diminuant de leurs valeurs. Ainsi, à la fin de l'apprentissage, un ensemble de prototypes $\mathbf{w}_c$ et de score GOF_c sera représentatif d'un sous-groupe P_c ou cluster de l'ensemble des données.

Algorithm 1 : Algorithme GOF-SOM

1: **ENTRÉES :**
   - Les données $\mathcal{A} = \{\mathbf{x}_i\}_{i=1..N}$
   - La carte SOM avec K référents initialisés $\{\mathbf{w}_c, c = 1 \dots K\}$
   - t_{max} : le nombre maximum d'itérations.
   - Initialisation des valeurs GOF.

2: **SORTIES :**
   - Une partition $P = \{P_c\}_{c=1...K}$.
   - Les valeurs de GOF= $\{GOF_c, c = 1 \dots K\}$

3: **Phase de compétition** : affecter une donnée $\mathbf{x}_i$ en utilisant la fonction

$$\phi(\mathbf{x}_i) = \arg \min_{1 \leq c \leq K} \parallel \mathbf{x}_i - \mathbf{w}_c \parallel^2$$

4: **Phase d'adaptation**
   - Mettre à jour les référents $\mathbf{w}_c$ de chaque cellule c

$$\mathbf{w}_c(t) = \mathbf{w}_c(t-1) - \varepsilon(t)\mathcal{K}^{\mathcal{T}}(\delta(\phi(\mathbf{x}_i), c))\,(\mathbf{w}_c(t-1) - \mathbf{x}_i)$$

   - Mettre à jour les valeurs de GOF_c associées à chaque cellule c

$$GOF_c(t) = GOF_c(t-1) - \varepsilon(t)\mathcal{K}^{\mathcal{T}}(\delta(\phi(\mathbf{x}_i), c)) \left(GOF_c(t-1) - \frac{\frac{\sum_{\mathbf{x}_j \in P_c} \frac{1}{f_c(\mathbf{x}_j)}}{|P_c|}}{\frac{1}{f_c(\mathbf{x}_i)}} \right)$$

     où $\varepsilon(t)$ est le pas d'apprentissage, T est la température qui varie au cours de l'apprentissage.

5: Répéter les phases de compétition et d'adaptation jusqu'à un nombre d'itérations fixé $t = t_{max}$.

Cet algorithme est proche de l'algorithme classique SOM, ce qui permet de conserver la topologie en deux dimensions de la carte, et de fournir une visualisation simple de la structure des données. Par ailleurs, l'utilisation d'une valeur GOF_c donne une information locale sur "l'outliner-ness" du sous-ensemble associé à la cellule c, et garde une information sur la structure générale des données et des clusters entre eux. La complexité de calcul de l'algorithme GOF-SOM est $O(N\ K)$ pour une itération t.

Il est à noter que le résultat final dépend en partie de l'ordre de présentation des données (cet ordre est souvent aléatoire), et peut donc varier légèrement d'une exécution à l'autre. Les résultats dépendent aussi de l'initialisation des référents de la carte (qui peut être aléatoire). Concernant l'initialisation des valeurs de GOF_c, nous avons opté pour une initialisation équiprobable. Nous avons évalué les performances de cet algorithme sur un ensemble de jeux de données présentant des difficultés pour la classification.

4 Détection de nouveautés : classifieur ”GOF-Novelty”

Dans cette section, nous allons montrer comment utiliser la mesure GOF pour la détection des nouveautés. Nous utilisons la même structure et architecture des cartes déjà définies dans la section 3. La méthode que nous proposons consiste à affecter les données $\mathbf{x}_i$ de la nouvelle base $\mathcal{A}^{'}$ en utilisant les résultats de l'algorithme GOF-SOM obtenus à partir de la base $\mathcal{A}$. Ainsi, nous proposons de calculer pour chaque cluster c et pour chaque donnée $\mathbf{x}_i \in \mathcal{A}^{'}$, un score “Outlier Factor” ($OF_c(\mathbf{x}_i)$) comme suit :

$$OF_c(\mathbf{x}_i) = \frac{\frac{\sum_{\mathbf{x}_j \in P_c} \frac{1}{f_c(\mathbf{x}_j)}}{|P_c|}}{\frac{1}{f_c(\mathbf{x}_i)}}$$

Si la valeur de $OF_c(\mathbf{x}_i)$ est plus grande que la valeur de Group Outlier Factor du cluster (GOF_c), alors, nécessairement, la donnée $\mathbf{x}_i$ est nouvelle. C'est ainsi que nous proposons l'algorithme 2, qui permet de traiter le problème de la détection des nouveautés en utilisant le paramètre GOF.

Algorithm 2 Algorithm GOF-Novelty

1: Entrées :
   – La partition $P = \{P_c\}_{c=1..K}$,
   – Valeurs de GOF $= \{GOF_c, c = 1..K\}$,
   – La nouvelle base de données $\mathcal{A}^{'} = \{\mathbf{x}_i\}_{i=1..M}$.
2: Sorties :
   – $(Novelty\_label)$: vecteur binaire des nouveautés.
3: **for** i=1 : M **do**
4: $OF_{\phi(\mathbf{x}_i)}(\mathbf{x}_i) = \frac{\frac{\sum_{\mathbf{x}_j \in P_{\phi(\mathbf{x}_i)}} \frac{1}{f_c(\mathbf{x}_j)}}{|P_{\phi(\mathbf{x}_i)}|}}{\frac{1}{f_c(\mathbf{x}_i)}}$
5: $Dif = |OF_{\phi(\mathbf{x}_i)}(\mathbf{x}_i) - GOF_{\phi(\mathbf{x}_i)}|$
6: **if** $(Dif < threshold)$ **then**
7: $Novelty\_label(\mathbf{x}_i) = 0$;
8: **else**
9: $Novelty\_label(\mathbf{x}_i) = 1$;
10: **end if**
11: **end for**

$threshold$ peut varier selon les bases (σ par défaut).

5 Expérimentations et évaluations de la détection de "groupes-outliers" et de nouveautés

5.1 Description des bases de données utilisées

Nous avons utilisé dans ces expérimentations différentes bases de données. Pour la détection d'outliers, nous avons évalué notre approche sur des bases provenant du répertoire UCI (Bache et Lichman (2013)) ainsi que sur des bases simulées générées aléatoirement suivant une loi normale de et de manière à créer des groupes largement isolés du reste des données. Concernant l'évaluation de la détection de nouveautés, nous avons utilisée dans nos expérimentations des bases de données de type "One-class classifier" introduite par Pekalska et al. (2003) téléchargeable à partir de cette adresse : *http ://homepage.tudelft.nl/n9d04/occ/index.html*, et aussi des bases de données publiques Bache et Lichman (2013). Pour ces bases, l'ensemble des données d'apprentissage est formé uniquement des données étiquetées 0 (normal). Les bases de test contiennent des données étiquetées 1 (nouveautés) et 20% de données étiquetés 0. Pour les bases publiques, les étiquettes 1 sont attribuées à la classe minoritaire (nouveautés). Le tableau 1 représente les descriptions de ces différentes bases de données.

Bases publiques et simulées						Base de type "One-class classifier"				
Base	# Obs	# Var	Taille carte	# Nor	# Out	Base	# Obs	# Var	# Nor	# Out
anneauxM	1072	2	14×12	943	129	IrisSetosa	150	4	50	100
demicercleM	638	3	13×10	586	52	SonarMines	108	60	11	97
HeptaM	212	2	9×8	136	76	BiomedHealthy	194	5	127	67
LsunModif	400	2	11×9	300	100	HepatitisNormal	155	19	123	32
TargetM	951	2	13×12	787	164	DiabetesPresent	768	8	500	268
GolfBallM	4343	3	19×17	3941	402	EcoliPeriplasm	336	7	52	284
B S 1	160	4	5×13	143	17	Spectf 1	349	44	254	95
B S 2	234	4	3×26	208	26	Balance-Scale	625	4	288	337
B S 3	569	4	8×15	357	212	GlassBuilding	214	9	70	144
B S 4	402	4	8×13	292	110	Waveform 2	900	21	300	600

TAB. 1 – *Description des bases publiques, simulées et One-class classification. B S : bases simulées. M : modifiées. Obs : observation. Var : variable. Out : outlier. Nor : normal.*

Remarques :
- La taille de la carte est choisie selon l'heuristique de Kohonen [3] $Taille = 5 \times K^{0.54321}$. Dans certain cas, nous choisissons une taille proportionnelle à la taille de la base de données ;
- L'initialisation des prototypes est réalisée d'une façon aléatoire ;
- L'initialisation des valeurs de GOF est réalisée d'une manière équiprobable $= \frac{1}{K}$;
- Les bases simulées sont générées aléatoirement suivant une loi normale (une gaussienne) pour créer des groupes largement isolés du reste des données.

3. http ://www.cis.hut.fi/somtoolbox/package/docs2/som_make.html

5.2 Mesures de performances

Afin de calculer les critères de performances de notre algorithme, nous avons d'abord défini une matrice de confusion (Kohavi et Provost (1998)) représentée dans le tableau 2. Cette matrice contient les informations sur les classes réelles et les classes prédites par notre classifieur. Ensuite, nous calculons les indices : *Rappel*, *Précision*, *F-mesure* et *AUC* afin d'évaluer les performances de notre algorithme.

		Classes réelles +	**Classes réelles** −
Classes prédites	+	Vrais Positives (VP)	Faux Positives (FP)
	−	Faux Negatives (FN)	Vrais Negatives (VN)

TAB. 2 – *Matrice de confusion.*

La défintion des formules de calcul des indices du *Rappel*, *Précision*, *F-mesure* et *AUC* sont comme suit (Powers (2007)) :

– Le *Rappel* est le pourcentage des données positives bien classées. Cet indice et aussi appelé : sensibilité, taux de vrais positifs (TVP) ou recall.

$$Rappel = \frac{VP}{VP + FN}$$

– L'indice de *Précision* est la proportion des données prédictives positives correctement classées :

$$Precision = \frac{VP}{VP + FP}$$

– La *F-mesure* est une combinaison pondérée de *Rappel* et *Précision* :

$$F - mesure = \frac{2 \times Rappel \times Precision}{Precision + Rappel}$$

– La courbe Receiver Operating Characteristic (ROC) est un graphique exprimant la capacité d'un classifieur de faire la distinction entre les classes positives et les classes négatives. L'aire sous la courbe ROC (*AUC*) (Bradley (1997)) indique la *Précision* d'un classifieur.

$$AUC = \frac{TVP + TFN}{2}$$

5.3 Critère de sélection des “groupes-outliers“ : "*Scree Acceleration Test*"

Afin de détecter les ”groupes-outliers”, nous avons utilisé un test statistique proposé par Cattell (1966) appelé "Scree Test". Ce test nous permet de faire une sélection des valeurs GOF d'une manière automatique.

L'utilisation initiale du test “Scree Test“ (Cattell (1966)) consiste en la détermination visuelle du nombre de valeurs propres à prendre en compte lors d'une analyse en composantes principales. L'idée de base est de représenter graphiquement les valeurs propres et de trouver la valeur pour laquelle le changement brutal est apparu (scree). Le nombre de composantes à garder correspond au nombre de valeurs propres précédant ce 'Scree'. Fréquemment, ce 'Scree' apparait là où la pente du graphe change radicalement. Ainsi, il s'agit de trouver la décélération maximale dans ce graphique.

L'utilisation de ce test sur notre vecteur de paramètre, consiste à détecter, par exemple, le changement brutal dans le vecteur $GOF = (GOF_1, GOF_2..., GOF_j, ..., GOF_K)$. Ainsi, il faudrait détecter la plus forte décélération. La procédure de sélection est composée des étapes suivantes :

Algorithm 3 : Algorithme scree test adapté à GOF

1: Ordonner le vecteur $GOF = (GOF_1, GOF_2..., GOF_j, ..., GOF_K)$ suivant un ordre décroissant. Le nouveau vecteur ordonné est noté $GOF = (GOF^1_{..}, GOF^2_{..}..., GOF^j_{..}, ..., GOF^K_{..})$ où l'exposant i de GOF^i indique l'ordre.
2: Calculer les premières différences $df_i = GOF^i_{..} - GOF^{i+1}_{..}$
3: Calculer les deuxièmes différences (l'accélération) $acc_i = df_i - df_{i+1}$
4: Chercher le changement brutal 'scree' à l'aide de la fonction suivante : $\max_i (abs(acc_i) + abs(acc_{i+1}))$
Ce processus permet de sélectionner toutes les composantes se trouvant avant le changement brutal.

Base de données	# G.O réels	# G.O “Scree Test“	# G.O sans répétition
anneauxModif	1	1	1
demicerModif	1	1	1
HeptaModif	1	1	1
LsunModif	1	1	1
TargetModif	4	4	4
GolfBallModif	1	1	1
base simulée 1	1	1	1
base simulée 2	2	2	2
base simulée 3	3	5	3
base simulée 4	4	6	4

TAB. 3 – *Détection automatique des ”groupes-outliers“.*

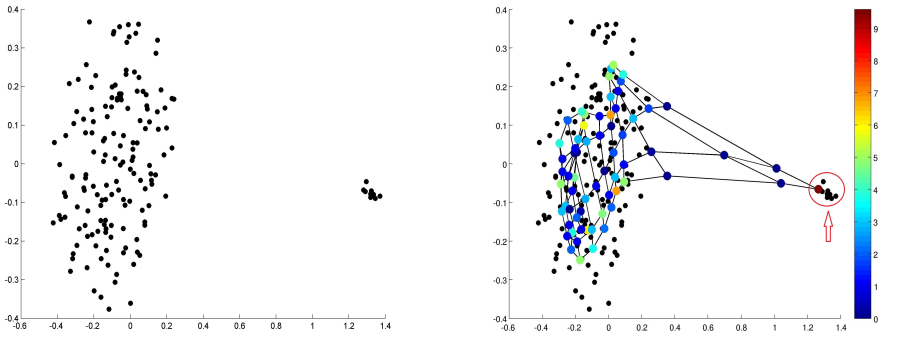
Le tableau 3 présente les résultats obtenus après l'application du “Scree Test”. Chaque valeur GOF sélectionnée représente un “groupe-outlier”. Il existe des cas où plusieurs référents

sélectionnés décrivent ensemble le même cluster. Par exemple dans le cas de la base simulée 3, “Scree Test“ a sélectionné 5 “groupes-outliers“ dont 2 groupes sont des sous ensembles du cluster outlier simulée, les deux autres appartiennent à un autre cluster et le dernier ”groupe-outlier” représente le 3e cluster. Finalement, les ”groupes-outliers” sélectionnés ne détectent que 3 clusters.

Afin de montrer l’intérêt de détecter les “groupes-outliers“, nous avons calculé le paramètre LOF pour chacune des bases. Particulièrement pour les bases simulées, nous avons constaté que l’utilisation de LOF ne permet de détecter aucune donnée outlier. Cet inconvénient est connu, car LOF repose sur deux principes : la distance entre les données et la densité de chacune d’elle. Généralement, dans un cluster, les distances entre les données sont petites et les densités locales des données en les comparant avec les densités moyennes de leurs k-ppv restent relativement égales. Ainsi dans le cas de nos bases simulées, les valeurs du LOF seront presque les mêmes que celles des données normales.

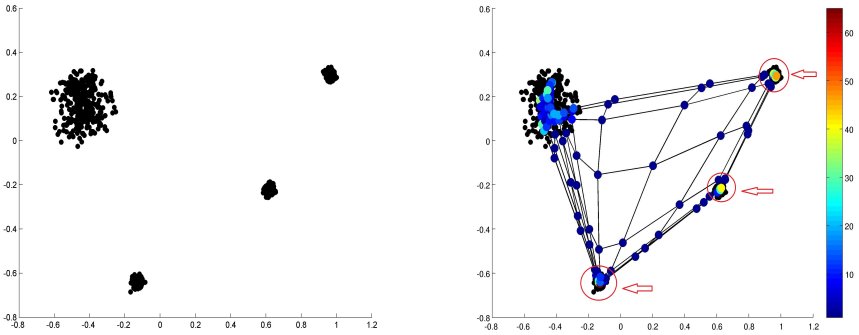
5.4 Résultats visuels de la détection de ”groupes-outliers“

Nous avons visualisé les données avec les référents de la carte. Nous pourrions obtenir le même résultat avec les cartes auto-organisatrices classiques, sauf que dans notre modèle, nous estimons au cours de l’apprentissage le paramètre GOF qui est visualisé à l’aide d’une couleur associée à chaque cellule de la carte. Plus la couleur est rouge, plus le groupe a une forte valeur de GOF.



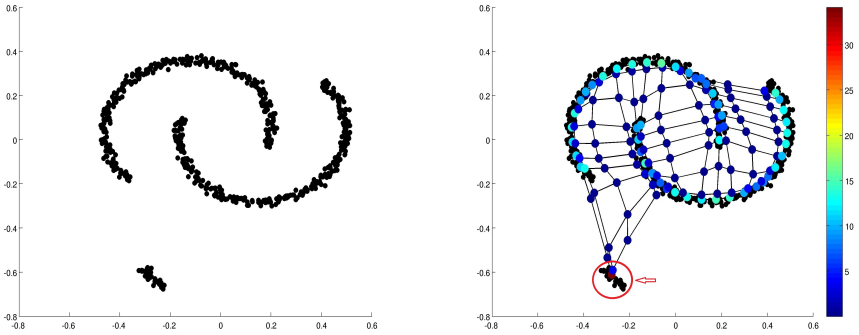
(a) Données de la base simulée 1 (b) Carte associée aux données de la base simulée 1

FIG. 1 – *GOF-SOM appliqué sur les données de la base simulée 1*



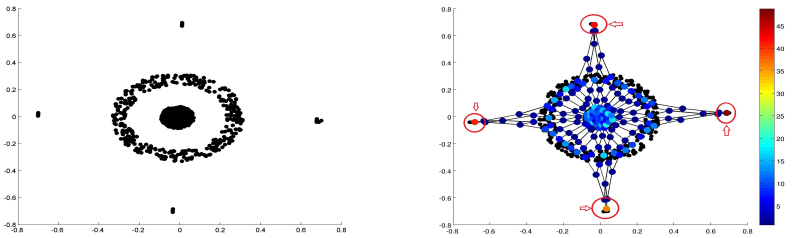
(a) Données de la base simulée 3 (b) Carte associée aux données de la base simulée 3

FIG. 2 – *GOF-SOM appliqué sur les données de la base simulée 3*



(a) Données de la base demicercleModif (b) Carte associée aux données de la base demicercle-Modif

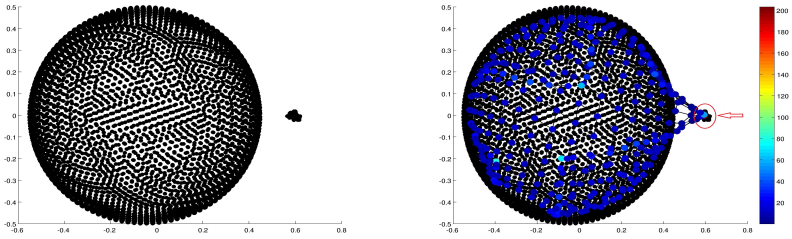
FIG. 3 – *GOF-SOM appliqué sur les données de la base demicercleModif*



(a) Données de la base TargetModif

(b) Carte associée aux données de la base Target-Modif

FIG. 4 – *GOF-SOM appliqué sur les données de la base TargetModif*



(a) Données de la base GolfBallModif

(b) Carte associée aux données de la base Golf-BallModif

FIG. 5 – *GOF-SOM appliqué sur les données de la base GolfBallModif*

Nous constatons clairement à partir des figures 1, 2, 3, 4 et 5, que les ”groupes-outliers” qui sont clairement visibles (voir les flèches rouges des figures 1.b, 2.b, 3.b, 4.b et 5.b) sont associés à des valeurs GOF très fortes et sont représentées par la couleur rouge (colonne à droite). Les référents et le paramètre GOF s’adaptent parfaitement et simultanément avec les groupes isolés.

5.5 Evaluations de la détection de nouveautés : validation croisée

La validation croisée est une technique de ré-échantillonnage permettant d’estimer le taux d’erreur d’un classifieur. La procédure suivie est la suivante : nous divisons la base en 5 sous-ensembles. Pour chaque expérimentation, nous sélectionnons 4 sous-ensembles pour la base d’apprentissage et un sous-ensemble pour la base de teste. Ce processus est répété 10 fois. Nous avons comparé GOF-Novelty avec l’approche ACP (Hoffmann (2007)) en prenant en compte 2 composantes principales et le One-SVM (Scholkopf et al. (2001)). Les tableaux 4, 5, 6 et 7 représentent les résultats expérimentaux des indices du *Rappel*, *Précision*, *F-mesure* et l’aire sous la courbe (*AUC*).

Bases	GOF-Novelty		ACP		One-SVM	
	Moyenne	Ecart	Moyenne	Ecart	Moyenne	Ecart
b s 1	0.750	± **0.033**	0.723	± 0.092	**0.852**	± 0.081
b s 2	0.508	± 0.067	0.522	± 0.121	**0.843**	± **0.059**
b s 3	0.490	± **0.030**	0.382	± 0.043	**0.529**	± 0.062
b s 4	0.453	± **0.046**	0.412	± 0.051	**0.521**	± 0.051
demicercleM	**0.625**	± 0.100	0.213	± 0.038	0.272	± **0.025**
anneauxM	**0.800**	± **0.001**	0.713	± 0.018	0.672	± 0.021
LsunM	0.722	± 0.056	**0.798**	± **0.039**	0.213	± 0.043
TargetM	0.709	± **0.016**	**0.715**	± 0.028	0.590	± 0.002
HeptaM	**0.586**	± 0.066	0.512	± **0.042**	0.240	± 0.051
GolfBallM	**0.836**	± 0.061	0.703	± 0.041	0.119	± **0.032**
Iris S	0.698	± 0.073	0.431	± 0.081	**0.762**	± **0.061**
Sonar M	**0.673**	± 0.092	0.381	± 0.058	0.352	± **0.031**
Biomed H	0.520	± 0.092	**0.721**	± 0.081	0.393	± **0.063**
Hepatitis N	**0.689**	± 0.056	0.619	± **0.045**	0.538	± 0.131
Diabetes P	0.666	± 0.072	0.532	± 0.062	**0.712**	± **0.059**
Ecoli P	**0.980**	± **0.001**	0.818	± 0.013	0.883	± 0.009
Spectf 1	0.620	± **0.060**	0.723	± 0.082	**0.731**	± 0.091
Balance S L	0.783	± **0.013**	**0.853**	± 0.042	0.520	± 0.031
Glass B F	0.584	± 0.022	0.601	± 0.058	**0.661**	± **0.018**
Waveform 2	**0.823**	± 0.091	0.539	± 0.083	0.453	± **0.031**

TAB. 4 – *Moyenne et écart de l'indice du Rappel obtenu sur GOF-Novelty, ACP et One-SVM en utilisant une validation croisée.*

Bases	GOF-Novelty		ACP		One-SVM	
	Moyenne	Ecart	Moyenne	Ecart	Moyenne	Ecart
b s 1	0.343	± 0.067	**0.521**	± 0.053	0.431	± **0.012**
b s 2	0.483	± 0.082	**0.631**	± 0.092	0.420	± **0.061**
b s 3	**0.842**	± **0.028**	0.591	± 0.043	0.723	± 0.031
b s 4	0.700	± 0.047	**0.721**	± 0.040	0.713	± **0.038**
demicercleM	0.498	± **0.024**	0.513	± 0.032	**0.621**	± 0.059
anneauxM	**0.527**	± **0.005**	0.502	± 0.012	0.343	± 0.009
LsunM	0.771	± 0.036	0.629	± **0.029**	**0.812**	± 0.048
TargetM	**0.961**	± 0.012	0.912	± **0.009**	0.953	± 0.014
HeptaM	0.830	± 0.030	**0.892**	± 0.041	0.431	± **0.021**
GolfBallM	**0.753**	± **0.009**	0.432	± 0.093	0.620	± 0.021
Iris S	0.637	± 0.101	**0.652**	± 0.098	0.628	± **0.063**
Sonar M	0.457	± **0.016**	**0.562**	± 0.142	0.493	± 0.041
Biomed H	0.348	± **0.016**	**0.432**	± 0.023	0.160	± 0.043
Hepatitis N	0.753	± 0.041	**0.812**	± 0.058	0.661	± **0.023**
Diabetes P	**0.698**	± **0.018**	0.654	± 0.028	0.682	± 0.034
Ecoli P	**0.812**	± 0.082	0.445	± **0.030**	0.809	± 0.068
Spectf 1	0.286	± 0.054	**0.352**	± 0.110	0.221	± **0.031**
Balance S L	**0.832**	± **0.008**	0.453	± 0.049	0.738	± 0.020
Glass B F	0.659	± 0.035	**0.693**	± 0.032	0.662	± **0.030**
Waveform 2	**0.738**	± 0.020	0.712	± **0.018**	0.703	± 0.024

TAB. 5 – *Moyenne et écart de l'indice de Précision obtenu sur GOF-Novelty, ACP et One-SVM en utilisant une validation croisée.*

Bases	GOF-Novelty		ACP		One-SVM	
	Moyenne	Ecart	Moyenne	Ecart	Moyenne	Ecart
b s 1	0.471	± 0.044	**0.606**	± 0.067	0.572	± **0.021**
b s 2	0.495	± 0.074	**0.571**	± 0.105	0.561	± **0.060**
b s 3	**0.619**	± **0.029**	0.464	± 0.043	0.611	± 0.041
b s 4	0.550	± 0.046	0.524	± **0.045**	**0.602**	± 0.044
demicercleM	0.554	± 0.038	0.301	± **0.035**	0.378	± **0.035**
anneauxM	**0.635**	± **0.002**	0.589	± 0.014	0.454	± 0.013
LsunM	**0.746**	± 0.044	0.703	± **0.033**	0.337	± 0.045
TargetM	0.816	± 0.014	**0.802**	± 0.014	0.729	± **0.004**
HeptaM	**0.687**	± 0.041	0.651	± 0.041	0.308	± **0.030**
GolfBallM	**0.792**	± **0.016**	0.535	± 0.057	0.200	± 0.025
Iris S	0.666	± 0.085	0.519	± 0.089	**0.689**	± **0.062**
Sonar M	**0.544**	± **0.027**	0.454	± 0.082	0.411	± 0.035
Biomed H	0.417	± 0.027	**0.540**	± **0.015**	0.227	± 0.020
Hepatitis N	**0.720**	± 0.047	0.702	± 0.051	0.593	± **0.039**
Diabetes P	0.682	± **0.029**	0.587	± 0.039	**0.697**	± 0.043
Ecoli P	**0.888**	± **0.002**	0.576	± 0.018	0.844	± 0.016
Spectf 1	0.391	± 0.057	**0.473**	± 0.094	0.339	± **0.046**
Balance S L	**0.807**	± **0.010**	0.592	± 0.045	0.610	± 0.024
Glass B F	0.619	± 0.027	**0.644**	± 0.041	0.661	± **0.022**
Waveform 2	**0.778**	± 0.033	0.614	± 0.030	0.551	± **0.027**

TAB. 6 – *Moyenne et écart de l'indice de la F-mesure obtenu sur GOF-Novelty, ACP et One-SVM en utilisant une validation croisée.*

Bases	GOF-Novelty		ACP		One-SVM	
	Moyenne	Ecart	Moyenne	Ecart	Moyenne	Ecart
b s 2	0.505	± **0.062**	**0.662**	± 0.072	0.482	± 0.068
b s 3	**0.528**	± 0.055	0.432	± 0.040	0.503	± **0.012**
b s 4	**0.528**	± 0.039	0.500	± 0.035	0.430	± **0.020**
demicercleM	**0.499**	± 0.030	0.431	± **0.026**	0.381	± 0.031
anneauxM	0.488	± **0.018**	**0.610**	± 0.052	0.512	± 0.021
LsunM	**0.537**	± 0.071	0.380	± 0.063	0.483	± **0.059**
TargetM	0.556	± 0.086	**0.681**	± 0.162	0.502	± **0.061**
HeptaM	0.439	± 0.093	**0.503**	± **0.082**	0.501	± 0.129
GolfBallM	**0.693**	± 0.039	0.439	± **0.021**	0.538	± 0.123
Iris S	0.484	± **0.064**	0.531	± 0.092	**0.582**	± 0.072
Sonar M	0.488	± **0.014**	0.491	± 0.018	**0.494**	± 0.021
Biomed H	0.494	± **0.019**	0.397	± 0.020	**0.502**	± 0.027
Hepatitis N	0.638	± 0.019	0.603	± **0.016**	**0.651**	± 0.031
Diabetes P	**0.830**	± 0.018	0.431	± 0.031	0.520	± **0.002**
Ecoli P	0.793	± 0.093	**0.821**	± 0.089	0.753	± **0.037**
Spectf 1	**0.496**	± **0.027**	0.382	± 0.031	0.431	± 0.030
Balance S l	0.703	± **0.006**	**0.802**	± 0.020	0.721	± 0.012
Glass B L	**0.513**	± 0.028	0.472	± **0.011**	0.500	± 0.021
Waveform 2	**0.712**	± 0.025	0.698	± **0.019**	0.621	± 0.038

TAB. 7 – *Moyenne et écart de l'indice du l'AUC obtenu sur GOF-Novelty, ACP et One-SVM en utilisant une validation croisée.*

Indice du *Rappel* : le tableau 4 résume les résultats expérimentaux obtenus sur l'indice du

Rappel. Notre méthode GOF-Novelty fournit les valeurs les plus élevées de l'indice du *Rappel* pour les bases de données suivantes : demicercleM, anneauxM, HeptaM, GolfBallM, Sonar Mines, Hepatitis Normal, Ecoli Periplasm et Waveform 2.

One-SVM donne de meilleurs résultats dans les bases : simulées 1, 2, 3 et 4, Iris Setosa, Spectf 1 et Glass Building Float. L'ACP est plus performante dans les bases LsunM, TargetM, Biomed Healthy et Balance-Scale Left.

En dépit d'une diminution des performances dans les bases précédentes, GOF-Novelty reste la méthode la plus stable par rapport à l'ACP et One-SVM. Par exemple dans le jeu de données demicercleM, GOF-Novelty fournit 0.625. Nous observons clairement la baisse de l'indice du *Rappel* dans les approches ACP et One-SVM (0.213 et 0.272 respectivement).

Indice de *Précision* : l'analyse de l'indice de *Précision* présenté dans le tableau 5 montre que GOF-Novelty, ACP et One-SVM fournissent des valeurs équivalentes dans les bases de données suivantes : Iris Setosa, Sonar Mines, Hepatitis Normal, Diabets Present, Spectf 1, Glass-Building Float, Waveform 2. Une diminution de performance pour notre approche GOF-Novelty est observée dans les bases simulées 1 et 2 et Biomed Healthy. GOF-Novelty fournit les valeurs les plus élevées de l'indice de *Précision* dans les bases : simulée 3, GolfBall, Ecoli Periplasm et Balance-Scale left.

On observe une faible diminution en termes d'indice de *Précision* dans les approche ACP et One-SVM dans certaines bases de données. Par exemple, dans les bases Hepta et Biomed Healthy, One-SVM fournit respectivement 0.431 et 0.16 alors que la meilleure valeur de l'indice de *Précision* est fournie par l'approche ACP (0.892 et 0.432 respectivement). Notre méthode GOF-Novelty reste compétitive à l'approche ACP où elle fournit 0.83 et 0.348 respectivement.

Indice de F-mesure : observant l'indice *F-mesure* résumé dans le tableau 6, notre méthode GOF-Novelty fournit les valeurs les plus élevées de la *F-mesure* dans plusieurs bases de données excepté dans les bases simulées 1, 2 et 4, Iris setosa, Biomed Healthy, Diabetes Present, spectf 1 et Glass Building Fload où l'on observe une diminution de cet indice.

La *F-mesure* diminue sensiblement dans les bases Ecoli Periplasm et Balance-Scale Lef (respectivement 0.576 et 0.592) pour l'approche ACP. On observe également une très faible diminution des performances de l'approches One-SVM dans les bases HeptaM, GolfballM et Balance-Scale Left.

Indice de l'*AUC* : pour l'indice *AUC* représenté dans le tableau 7, GOF-Novelty, ACP et One-SVM fournissent des résultats équivalent dans la majorité des bases de données excepté pour la base Diabets Present où l'on remarque une diminution importante de la valeur de l'*AUC* pour les approches ACP et One-SVM (0.431 et 0.52 respectivement), GOF-Novelty fournit pour cette même base une performance de 0.83.

Ecart de la validation croisée : concernant l'analyse des écarts obtenus, les trois méthodes donnent des valeurs similaires. Dans la plupart des bases de données et des indices de performance, les valeurs obtenues restent largement équivalentes.

Cependant, certaines exceptions sont observées. En effet, la base de données anneauxM fournit un écart de l'indice du *Rappel* de 0.1% pour notre approche GOF-Novelty. APC et One-SVM donnent respectivement 1.8% et 2.1%.

La même observation peut être faite pour l'écart de l'indice de *Précision* de la base de données GolfballM où GOF-Novelty fournit 0.9%, les approches ACP et One-SVM fournissent 9.3% et 2.1% respectivement.

Dans la majorité des cas, notre méthodes obtient de meilleurs résultats surtout pour la *F-mesure*. Nous remarquons que notre algorithme est performant lorsque les bases de données contiennent des clusters denses avec beaucoup de données. C'est le cas de la base Golfball où notre approche obtient les meilleurs performances sur les 4 indices. Par contre dans la base simulée 1, on a des nuages de point relativement éparpillés. C'est ce qui explique la baisse des performances obtenues.

Enfin, nous concluons au regard de ces analyses réalisées sur différents indices de performances, que GOF-Novelty est une approche qui permet de détecter les nouveautés d'une manière pertinente. Notre approche donne dans la plus part des bases de données des résultats stables par rapport aux approches ACP et One-SVM.

6 Conclusion et perspectives

Nous nous sommes intéressés dans la première partie de ce travail au problème de détection de “groupes-outliers“. Nous avons présenté un nouveau score (paramètre) GOF qui se base sur les densités locales des clusters. Ce score a été intégré aux cartes auto-organisatrices. Une série d'expériences a été réalisée pour valider la méthode proposée. Les résultats obtenus ont été analysés visuellement et analytiquement. Ceci nous a permis de mieux évaluer notre approche qui s'est avérée prometteuse comme solution au problème de détection de ”groupes-outliers“.

Dans la seconde partie, nous avons utilisé GOF comme classifieur pour le problème de la détection de nouveautés. Une série d'expérimentations a été réalisée en comparant les performances de notre approche avec deux méthodes classiques de détection de nouveautés (ACP et One-SVM). Les résultats expérimentaux montrent que notre approche est compétitive et prometteuse.

Nombreuses sont les perspectives qu'offre notre approche telle que l'adaptation de notre algorithme au cas des flux de données. Cette question est incontournable car les groupes que nous classons comme “outliers”, peuvent être classées comme “normaux“ si un grand nombre de données sont affectées aux “groupes-outliers“. L'enjeu est la détermination, ou du moins l'estimation du seuil de passage d'un ”groupe-outlier” à un “groupe-normal”.

Références

Bache, K. et M. Lichman (2013). UCI machine learning repository.

Borisyuk, R. M. et Y. B. Kazanovich (2004). Oscillatory model of attention-guided object selection and novelty detection. *Neural Netw. 17*(7), 899–915.

Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition 30*, 1145–1159.

Breunig, M., H. Kriege, R. Ng, et J. Sander (2000). Lof: Identifying density-based local outliers. *ACM SIGMOD 2000 International congerence on Management of Data*.

Cai, Q., H.He, et H.Man (2009). Somso: A self-organizing map appoach for spacial outlier detection with multiple attributes. *occedings of International Joint Conference on N.N*.

Cai, Q., H.He, H. Man, et J. Qiu (2010). Iterativesomso: An iterative self-organizing map for spatial outlier detection. *occedings of International Joint Conference on Neural Networks 1*, 325–330.

Cattell, R. (1966). The scree test for the number of factors. *M.B.R 1*, 245–276.

Chandola, V., A. Banerjee, et V. Kumar (2009). Anomaly detection: A survey. *ACM Comput. Surv. 41*(3), 15:1–15:58.

E. Schubert, A. Zimek, H.-P. K. (2012). Local outlier detection reconsidered: a generalized view on locality with applications to spatial, video, and network outlier detection. Data Mining and Knowledge Discovery.

Fabrizio, A. et C. Pizzuti (2002). Fast outlier detection in high dimensional spaces. *PKDD '02 Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery*.

Gao, J., W. Hu, W. Li, et Z. Zhang (2010). Local outlier detection based on kernel regression. *International Conference on Pattern Recognition*.

Hasan, M. A., V. Chaoji, S. Salem, et M. J. Zaki (2009). Robust partitional clustering by outlier and density insensitive seeding. *Pattern Recogn. Lett. 30*, 994–1002.

Hodge, V. J. et J. Austin (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review 22*, 2004.

Hoffmann, H. (2007). Kernel pca for novelty detection. *Pattern Recognition 40*.

Kohavi et Provost (1998). Glossary of terms. *Editorial for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process*.

Kohonen (1995). *Self-Organizing Maps*. Berlin: Springer Verlag.

Lauer, M. (2001). A mixture approach to novelty detection using training data with outliers. In *Lecture Notes in Computer Science*, pp. 300–311. Springer.

Liu, B., J. Yin, Y. Xiao, L. Cao, et P. Yu (2010). Exploiting local data uncertainly to boost global outlier detection. *IEEE International Conference On Data Mining*.

Markou, M. et S. Singh (2003a). Novelty detection: a review part 1: statistical approaches. *Signal Process. 83*, 2481–2497.

Markou, M. et S. Singh (2003b). Novelty detection: a review part 2: neural network based approaches. *Signal Process. 83*, 2499–2521.

Mennatallah Amer, M. G. (2012). Nearest-neighbor and clustering based anomaly detection algorithms for rapidminer. pp. 1–12. Shaker Verlag GmbH, Aachen.

Moya, M. R., M. W. Koch, et L. D. Hostetler (1993). One-class classifier networks for target recognition applications. *World Congress on Neural Networks, International Neural Network Society (INNS)*.

Odin, T. et A. D. (2000). Novelty detection using neural network technology.

Pekalska, E., D. M. J. Tax, et R. P. W. Duin (2003). One-class lp classifiers for dissimilarity representations. In *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*, pp. 761–768. MIT Press.

Powers, D. M. W. (2007). Technical Report SIE-07-001, School of Informatics and Engineering, Flinders University, Adelaide, Australia.

Rusiecki, A. (2012). Robust neural network for novelty detection on data streams. In *ICAISC (1)*, pp. 178–186.

Scholkopf, B., J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, et R. C. Williamson (2001). Estimating the support of a high-dimensional distribution. *Neural Comput. 13*(7), 1443–1471.

Tax, D. M., R. P. Duin, N. Cristianini, J. Shawe-taylor, et B. Williamson (2001). Uniform object generation for optimizing one-class classifiers. *Journal of Machine Learning Research 2*, 155–173.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York: Springer.

Xing, H.-J., M.-H. Ha, et X.-Z. Wang (2009). Combining som and local minimum enclosing spheres for novelty detection. In *Proceedings of the 21st annual international conference on Chinese control and decision conference*, CCDC'09, Piscataway, NJ, USA, pp. 3814–3819. IEEE Press.

Ypma, A., E. Ypma, et R. P. Duin (1997). Novelty detection using self-organizing maps. In *In Proc. of ICONIP'97*, pp. 1322–1325. Springer.

Zengyou, H., X. Xu, et S. Deng (2003). Discovering cluster-based local outliers. *Journal Pattern Recognition Letter 24*, 9–10.

Summary

We present in this paper a new measure called GOF (Group Outlier Factor) for “groups-outliers“ detection and novelty detection. To validate this measure we integrated it in a clustering process using Self-organizing Map. GOF is based on relative density of each group of data and simultaneously provides a partitioning of data and a quantitative indicator of outlier-ness. After learning GOF, we use it as a classifier for the problem of novelty detection. The obtained results are very encouraging to continue in this direction.