

# Classification sous contraintes probabilistes par les cartes topologiques

Jihène SNOUSSI et Khalid BENABDESLEM

Université de Lyon, F69622, Lyon, France ;

Université Lyon1, Villeurbanne ;

LIESP, EA4125

[jsnoussi, kbenabde@bat710.univ-lyon1.fr](mailto:jsnoussi, kbenabde@bat710.univ-lyon1.fr)

**Résumé.** La classification automatique est un processus non supervisé qui vise à regrouper des données en un ensemble de classes hétérogènes. En outre, Différents travaux ont montré que l'intégration de contraintes peut augmenter le taux de ce processus de classification tout en diminuant le temps d'exécution. Cette nouvelle démarche a connu, ces dernières années, un travail bien étendu. La forme la plus répandue de ces dites contraintes est de type « Must-Link » dont le nom indique l'obligation d'avoir les données dans une même classe, et les contraintes « Cannot-Link » dont le nom indique l'interdiction d'avoir les données dans une même classe. Le travail présenté dans cet article décrit une nouvelle version des cartes topologiques que nous appelons « PrTM » (Probabilistic constrained Topological Map) pour intégrer des contraintes probabilistes. PrTM représente une variante d'un algorithme populaire des cartes topologiques probabilistes GTM (Generative Topographic Mapping). Pour valider notre approche, des comparaisons entre notre proposition « PrTM » et d'autres algorithmes de classification sous contraintes, sont présentées sur différentes bases de données issues de la littérature.

## 1 Introduction

L'extraction de connaissances par exploration des données fait souvent appel à des processus de classification automatique en mode non supervisé. Effectuer une classification, c'est mettre en évidence d'une part, les relations entre les différentes observations et d'autre part, les relations entre ces observations et leurs caractéristiques (leurs variables). A partir d'une certaine mesure de proximité ou de dissemblance, il s'agit de regrouper un ensemble de données en un ensemble de classes qui soient les plus hétérogènes possible (Saporta, 2006). Cependant, les algorithmes de classification automatique ont accès seulement à l'ensemble des variables; et il n'est fourni aucune information concernant l'affectation d'une observation à une classe. La prise en compte de ces connaissances additionnelles constitue un problème essentiel et un vrai défi pour la recherche actuelle puisqu'il s'agit à la fois de

## Classification sous contraintes probabilistes par les cartes topologiques

l'expression, de la structuration et de la formalisation de ces connaissances (appelées aussi connaissances « *a priori* ») pour les intégrer dans le processus de la classification automatique (Benhassena, et al., 2008). La classification automatique a comme but de découvrir les relations intrinsèques entre les données non étiquetées. Or, des relations extrinsèques peuvent être imposées sous forme de « contraintes ». Ces contraintes peuvent se présenter sous la forme de regroupements ou d'associations entre les données, des estimations de similarités additionnelles ou de la pertinence de certaines variables par rapport aux autres, etc.

L'intégration des contraintes dans la classification non supervisée représente un axe de recherche assez récent et émergent. Le premier type de contraintes étudié est celui des contraintes déterministes " positives " et " négatives " (Wagstaff et Cardie, 2000). Les contraintes déterministes positives correspondent à ce que deux individus doivent être dans la même classe, par contre, les contraintes négatives correspondent à ce que deux individus doivent être affectés à deux classes différentes. Par ailleurs, une contrainte peut être formulée sous forme d'une probabilité et définit donc un autre type de contraintes dites probabilistes (à intégrer dans un processus de classification). Cependant, il existe peu de travaux ayant traité ce genre de problème (Law et al., 2004), (Law et al, 2005). Dans cet article, nous introduisons une nouvelle approche pour l'intégration de ce type de contraintes dans un algorithme de la classification non supervisée basée sur les cartes auto-organisatrices.

La section suivante est consacrée à la présentation du modèle génératif des cartes topologiques GTM (Bishop et al, 1998). Nous décrivons par la suite notre approche proposée : PrTM pour intégrer les contraintes probabilistes en se basant sur le principe de GTM. Finalement, nous présentons quelques résultats expérimentaux montrant une comparaison de l'approche proposée avec GTM et d'autres méthodes de classification sous contraintes : COP-COBWEB (Wagstaff et Cardie, 2000), COP-Kmeans (Wagstaff et al., 2001), COP-b-Coloring (Elghazel et al., 2007) et CrTM (Bellal et al, 2008) sur quelques bases issues de la littérature.

## 2 Carte topologique générative : GTM

La carte topographique générative (GTM) est proposée par Bishop, Svensén et Williams en 1998. Elle est présentée comme une nouvelle formulation probabiliste de la carte auto-organisatrice (SOM) (Kohonen, 1994). Elle offre un certain nombre d'avantages, comparée à SOM et elle a déjà été utilisée dans une variété d'applications. Elle représente une démarche de transformation des observations entre deux couches, une première couche qui représente l'espace d'entrée qui représente l'ensemble des données, et une deuxième couche qui représente l'ensemble de classes (l'espace de sortie) Figure 1. Il s'agit d'un modèle non-linéaire de variables cachées, qui représente la densité de probabilité des données dans un espace de haute dimension (l'espace de données) en terme d'un nombre petit de variables latentes (l'espace caché) et dont lequel ses paramètres sont optimisés en utilisant l'algorithme Espérance-Maximisation (EM) (Dempster, 1977). L'espace caché joue le rôle d'intermédiaire entre l'espace d'entrée et l'espace de sortie qui permet la transformation non-linéaire entre les deux espaces.

En considérant une grille de points dans l'espace latent, correspondant aux neurones dans SOM, on utilise des fonctions non-linéaires entre les deux espaces pour la représentation de l'un dans l'autre, comme illustré dans Figure 1. Certains modèles qui sont utilisés pour la visualisation des données, l'ont considéré comme la projection de l'espace de données dans un espace bidimensionnel.

Néanmoins, le modèle GTM définit la visualisation des données par la configuration de l'espace latent dans l'espace de données, donc on effectue la projection inverse en utilisant le théorème de Bayes, provoquant une distribution postérieure dans l'espace latent.

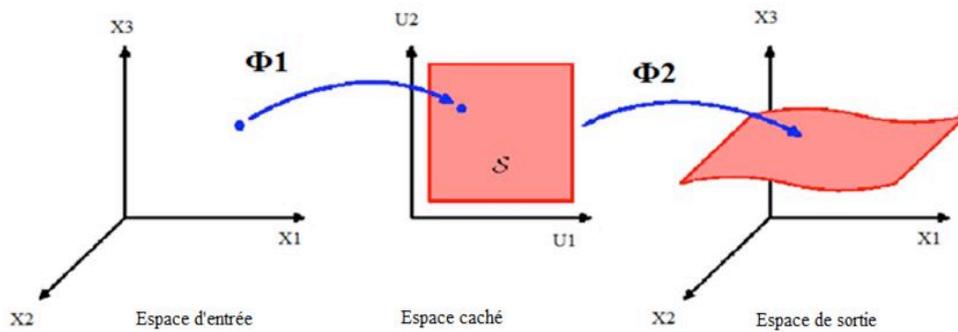


Fig. 1. Représentation de l'espace de données dans l'espace latent et la visualisation des données.

La carte topologique générative est définie par :

- Un espace de données  $X = (X_1, \dots, X_D)$  de dimension  $D$ ,
- Un espace latent  $U = (U_1, \dots, U_L)$  de dimension  $L$ ,
- Dans l'espace latent, une matrice des points étiquetés par l'index  $k = 1 \dots K$ , (correspondent aux neurones dans SOM) est présentée.
- Un ensemble de  $M$  fonctions de bases, non-linéaires noté  $\Phi = \{\Phi_j\}$ , qui permet de définir une transformation non-linéaire de l'espace latent à l'espace de données de telle façon à ce que pour chaque unité  $u_k$  (correspondant à un neurone dans SOM) dans l'espace latent, on lui associe un point  $y_k$  (correspond à un vecteur référent dans SOM) dans l'espace de données obtenu par une fonction de cartographie  $y(u; W)$  (Figure 2) qui dresse la carte de l'espace latent dans un collecteur non-euclidien qui a la même dimension  $L$  que l'espace latent. Cette fonction de transformation s'écrit sous la forme suivante:

$$y_k = W\phi(u_k) \tag{1}$$

avec  $W$  est la matrice des poids et  $\Phi$  l'ensemble de fonctions de bases.

## Classification sous contraintes probabilistes par les cartes topologiques

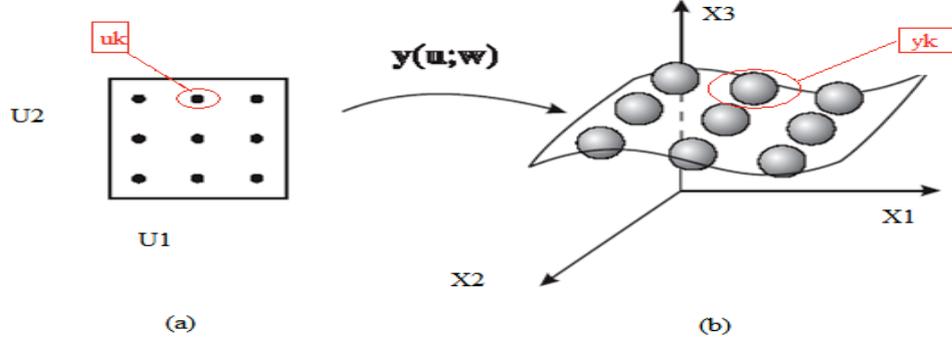


Fig. 2. Fonction de Cartographie  $y(u; W)$ .

Pour tout neurone  $u$  présenté sur la grille dans l'espace latent, on peut définir sa probabilité par:

$$p(u) = \frac{1}{K} \sum_{k=1}^K \delta(u - u_k) \quad (2)$$

Chaque vecteur référent  $y_k$  de la carte forme le centre d'une distribution Gaussienne dans l'espace de données, où  $\beta$  est considérée comme la variance inverse commune, donc pour toute donnée  $x$ , la probabilité conditionnelle d'appartenance à un neurone  $u_k$ , sachant les paramètres du modèle, est donnée dans la formule suivante :

$$p(x|u_k, W, \beta) = \left(\frac{\beta}{2\pi}\right)^{D/2} \exp\left(-\frac{\beta}{2} \|y_k - x\|^2\right) \quad (3)$$

Finalement, la fonction de densité de probabilité pour le modèle GTM est donnée en résumant toutes les composantes Gaussiennes, elle est obtenue par l'intégrale de l'équation ci-dessus par rapport à la distribution  $u$  :

$$p(x | W, \beta) = \int p(x | u, W, \beta) p(u) du = \frac{1}{K} \sum_{k=1}^K \left(\frac{\beta}{2\pi}\right)^{D/2} \exp\left\{-\frac{\beta}{2} \|y_k - x\|^2\right\} \quad (4)$$

Les paramètres que doit optimiser le modèle sont la matrice de poids  $W$  et la variance inverse  $\beta$ . Pour effectuer cette tâche, le modèle utilise l'algorithme EM, proposé par Dempster et al. (1977), en alternant les deux étapes E-étape et la M-étape. L'algorithme EM est une classe d'algorithmes qui permettent de trouver le maximum de vraisemblance (Log-likelihood) des paramètres de modèles probabilistes lorsqu'ils dépendent de variables cachées non observables.

Soit  $X = \{x_1, \dots, x_N\}$ , l'ensemble des données, avec  $N$  le nombre d'observations. La vraisemblance (Log-likelihood) est donnée par la formule suivante :

$$L(W, \beta) = \sum_{n=1}^N \ln[p(x_n | W, \beta)] \quad (5)$$

Dans l'E-étape, l'algorithme GTM calcule les valeurs initiales de probabilités postérieures des données, pour pouvoir évaluer les paramètres du modèle. Ces probabilités sont obtenues en utilisant le théorème de Bayes, elles sont données par:

$$Z_{kn} = P(u_k | x_n, W, \beta) = \frac{\exp\left\{-\frac{\beta}{2}\|y_k - x_n\|^2\right\}}{\sum_{k'=1}^K \exp\left\{-\frac{\beta}{2}\|y_{k'} - x_n\|^2\right\}} \quad (6)$$

Dans la M-étape, ces probabilités sont utilisées pour ré-estimer les nouvelles valeurs des paramètres  $W$  et  $\beta$ . La nouvelle matrice de poids  $W^{new}$  est recalculée en résolvant l'équation suivante :

$$\Phi^T G \Phi W^{new} - \Phi^T R X = 0 \quad (7)$$

Avec :

$\Phi$  est une matrice de dimensions  $K$  (nombre de neurones)  $\times$   $M$  (nombre de fonctions de bases utilisées), avec les éléments  $\Phi_{kj} = \Phi_j(u_k)$ .  $X$  est une matrice de dimensions  $N$  (nombre de données)  $\times$   $D$  (dimension de l'espace de données),  $Z$  est une matrice de probabilités de dimensions  $K \times N$ , et  $G$  une matrice carrée diagonale de dimension  $K$  dont les éléments sont données par  $G_{ii} = \sum_n Z_{nk}$ .

La nouvelle valeur  $\beta^{new}$  est donnée par la formule suivante

$$(\beta^{new})^{-1} = \frac{1}{ND} \sum_{n=1}^N \sum_{k=1}^K R_{kn} \|y_k - x_n\|^2 \quad (8)$$

### 3. Carte topologique et contraintes probabilistes : PrTM

Dans cette section, nous présentons une variante de GTM, appelée PrTM. Tout d'abord, nous définissons les différents types de contraintes qui peuvent être appliquées sur les données. Nous présentons par la suite un processus que nous avons développé pour géné-

rer les contraintes probabilistes à partir de la base d'apprentissage. Enfin, nous montrons comment modifier le modèle GTM pour l'adapter à ce type de contraintes.

### 3.1 Type des Contraintes

Les contraintes sont des règles obligatoires qui réduisent la liberté d'action. Elles se présentent sous forme de regroupement ou d'associations entre les données, des estimations de similarités additionnelles ou de la pertinence de certaines variables par rapport aux autres.

Il existe plusieurs types de contraintes :

*Contrainte déterministe positive* : appelée contrainte « Must-Link », Figure.3. (a), qui force deux observations d'être dans la même classe. Elle définit une relation transitive entre les observations

*Contrainte déterministe négative* : appelée contrainte « Cannot-Link », Figure.3. (b), qui interdit à deux observations d'être dans la même classe.

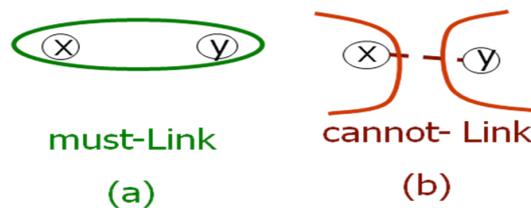


Fig.3. (a) Contrainte Must-Link, (b) Contrainte Cannot-Link.

*Contrainte probabiliste* : une contrainte probabiliste représente la probabilité avec laquelle deux observations peuvent être ou non dans la même classe.

D'autres types de contraintes peuvent être étudiées, à savoir, les contraintes conditionnelles, les contraintes de séparation minimale (les  $\delta$ -contraintes), les «  $\epsilon$ -contraintes », les «  $\gamma$ -contraintes » (Davidson et al., 2005).

### 3.2 Génération des contraintes probabilistes

Les contraintes probabilistes peuvent être données directement par l'utilisateur, ou bien être générées artificiellement à partir d'une base d'apprentissage étiquetée. Dans le deuxième cas, nous montrons comment produire des contraintes probabilistes de types « Must-Link » (ML) et « Cannot-Link » (CL) en se basant sur le calcul de distances entre les observations. Une contrainte est générée en comparant les étiquettes de ces observations. C'est-à-dire, si

deux observations ont la même étiquette, nous générons une contrainte ML, sinon une contrainte CL est mise en place. Pour chaque type des contraintes, une formule mathématique est donnée pour calculer la probabilité que deux observations soient ou non dans la même classe.

De même, nous avons imposé nos contraintes sur les probabilités générées. En d'autres termes, pour une contrainte probabiliste ML notée  $p_{ML}$  entre deux observations, il faut vérifier l'inéquation suivante :

$$0,5 \leq p_{ML} \leq 1 \quad (9)$$

et pour une contrainte probabiliste CL notée  $p_{CL}$ , il faut vérifier :

$$0 < p_{CL} \leq 0,5 \quad (10)$$

Puisque le calcul des probabilités se base essentiellement sur la distance entre les observations, une autre condition doit être vérifiée, c'est que tant que la distance entre deux observations augmente, la probabilité d'être dans la même classe diminue, autrement dit, si nous avons deux observations proches (distance petite) leur probabilité d'occuper la même classe doit être plus grande que celle générée entre deux autres observations plus éloignées (distance grande).

Les contraintes peuvent être propagées, on prenant en compte la fermeture transitive:

- $ML(x_i, x_j) \wedge ML(x_j, x_k) \Rightarrow ML(x_i, x_k)$ .
- $ML(x_i, x_j) \wedge CL(x_j, x_k) \Rightarrow CL(x_i, x_k)$ .

Notons par :

- $d(x_i, x_j)$  = distance entre l'individu  $x_i$  et l'individu  $x_j$ , et
- $p(x_i, x_j)$  = la probabilité que  $x_i$  et  $x_j$  soient dans la même classe,

Pour vérifier l'inéquation (9) pour la génération des contraintes de type ML (respectivement l'inéquation (10) pour les contraintes CL), on doit suivre une démarche mathématique pour trouver les bonnes formules de calcul des probabilités.

En effet, si on veut écrire la probabilité sous forme d'un quotient  $\frac{a}{b}$ , avec  $a, b \in \mathbb{R}^+$  et  $a < b$ ,

on doit trouver une relation entre  $a$  et  $b$  qui satisfait l'inéquation (9). Puisqu'une probabilité est toujours inférieure à 1, on a donc :

$$1 - \frac{a}{b} < 1 \quad \text{toujours vrai,}$$

On a choisi d'avoir, pour une contrainte positive, une probabilité supérieure à 0,5 pour rapprocher les deux observations contraintes même si elles sont éloignées donc :

Classification sous contraintes probabilistes par les cartes topologiques

$$1 - \frac{a}{b} > 0,5 \quad (11)$$

$$\frac{a}{b} < 0,5 \quad (12)$$

$$2a < b \quad (13)$$

Avec  $\Delta > 0$ , si on pose  $b = 2a + \Delta$ , on vérifié toujours (13).

Donc, remplaçant maintenant  $a$  par la distance entre les deux observations, on aura :

$$\underbrace{d(x_i, x_j)}_a < 2 \underbrace{d(x_i, x_j)}_b < 2 \underbrace{d(x_i, x_j)}_b + \Delta$$

La même démarche à suivre pour la génération des contraintes CL, mais en remplaçant (11)

par :

$$\frac{1}{2} - \frac{a}{b} > 0 \quad (14)$$

Finalement, les formules proposées pour calculer les probabilités sont données par:

- pour les contraintes ML:

$$p(x_i, x_j) = 1 - \frac{d(x_i, x_j)}{2 * d(x_i, x_j) + \Delta} \quad (15)$$

- pour les contraintes CL :

$$p(x_i, x_j) = \frac{1}{2} - \frac{d(x_i, x_j)}{2 * d(x_i, x_j) + \Delta} \quad (16)$$

### 3.3 Intégration des contraintes probabilistes dans GTM

Cette étape consiste à modifier l'algorithme GTM pour l'adapter aux contraintes probabilistes. Dans ce modèle, on calcule la probabilité d'affectation d'une observation  $x_i$  à un neurone sachant les paramètres du modèle, à partir de l'équation 3, qu'on résume dans :

$$R(x_i) = p(x_i | W, \beta) \quad (17)$$

avec  $\beta$  est la variance inverse et  $W$  est la matrice des poids.

L'idée de base consiste à tenir compte des contraintes probabilistes au cours de l'apprentissage. Cela se fait, dans la phase du calcul des probabilités d'affectation d'une observation sur les neurones de GTM. Nous testons si cette observation impose une contrainte ou non, si c'est le cas, sa probabilité d'affectation aux neurones est influencée par la pro-

babilité de la contrainte, sinon on garde la même façon que GTM pour sa probabilité d'appartenir aux différents neurones.

En effet, remarquons que d'après la formule (17), la probabilité d'appartenance d'une observation à un neurone est conditionnée par les paramètres du modèle. Ces conditions peuvent être vues comme des contraintes d'appartenance. Nous avons donc ajouté une contrainte additionnelle dans cette formule en cas de présence d'une contrainte probabiliste. Si on a une contrainte probabiliste entre  $x_i$  et  $x_j$  Alors on calcule

$$p(x_i | W, \beta, x_j) \quad (18)$$

Et donc si ML ( $x_i, x_j$ ), alors la probabilité d'affectation est donnée :

$$R(x_i) * p(x_i, x_j) * R(x_j) \quad (19)$$

Et si CL ( $x_i, x_j$ ), alors la probabilité d'affectation est donnée :

$$R(x_i) * (1 - p(x_i, x_j)) * R(x_j) \quad (20)$$

Par conséquent, l'approche PrTM opère comme suit :

- On initialise les valeurs de  $W$  et  $\beta$  et calcule les probabilités initiales d'affectations des individus sur les différents neurones (Etape E de EM).
- Dans l'étape M, on ré-estime les valeurs de  $W$  et  $\beta$  pour calculer les nouvelles probabilités d'affectation. Ici se fait notre modification majeure. En effet, à chaque fois qu'on calcule les nouvelles probabilités, on teste, si l'observation impose une contrainte ou non, si oui, sa probabilité est modifiée en tenant compte de la probabilité de la contrainte (équations 19 et 20) sinon, on calcule sa probabilité comme proposée par l'algorithme GTM.

## 4 Résultats expérimentaux

Dans cette section, nous présentons quelques résultats expérimentaux afin de valider notre algorithme. On a choisi de faire quelques tests sur différentes bases issues de la banque UCI (Blake, 1998), dont les données sont étiquetées. Les bases choisies sont: « *Tic-Tac-Toe* », « *Glass* », « *Cleve* » et « *Anneaux* ». Ces bases sont choisies pour pouvoir faire une comparaison, dans un premier temps entre PrTM et GTM, et dans un second temps avec les résultats d'autres méthodes de classifications sous contraintes utilisant les mêmes bases.

## Classification sous contraintes probabilistes par les cartes topologiques

Base	#d'exemples	dimension	# de classes
Glass	214	9	6
Anneaux	1000	3	2
Tic Tac Toe	958	9	2
Cleve	303	13	2

Table.1. bases de données.

### 4.1 Evaluation de PrTM

Pour l'évaluation de notre approche, on considère l'indice de Rand (Rand, 1971) pour calculer le taux de bonne classification. Nous avons choisi cet indice pour s'aligner par rapport à toutes les autres méthodes de classification sous contraintes qui l'ont utilisé. Nous signalons, que notre approche étant probabiliste, nous avons sélectionné la probabilité maximale pour chaque observation sur la carte.

Pour la base « *Glass* », sans aucune contrainte, GTM fournit 80% de précision, et avec PrTM et après intégration de 150 contraintes on arrive à 97% de précision. Avec la base « *Anneaux* », GTM donne 75% de bonne classification et PrTM donne, pour la même base, une précision de 94% (Figure 4) après l'intégration de 450 contraintes.

On peut remarquer que l'intégration des contraintes dans GTM permet une amélioration nette de cet algorithme de classification.

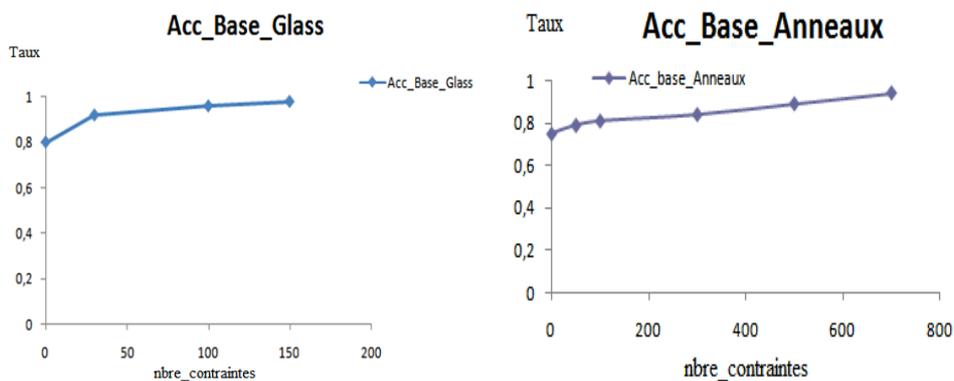


Fig. 4. Evaluation de PrTM sur les bases : « *Glass* » et « *Anneaux* »

Comparant notre nouvel algorithme PrTM avec d'autres algorithmes de classification sous contraintes, on remarque aussi une bonne amélioration de la précision de la classification. Sur la base « *Tic-Tac-Toe* », COP-COBWEB réalise un taux faible même après l'intégration de 500 contraintes. COP\_Kmeans, avec la même base et après l'intégration de 500 contraintes, la précision complète augmente jusqu'à 92 %. Avec PrTM on arrive à une précision de 94% avec 700 contraintes probabilistes aléatoires (Figure 5). Les meilleurs résultats pour cette base de données sont obtenus par CrTM avec un taux atteint 96,70% avec 500 contraintes intégrées et aussi avec COP-b-coloring pour un taux de 95 % avec le même nombre de contraintes. Le taux de bonne classification du CrTM avec la base « *Glass* » réalise 88,5 % de précision, avec l'incorporation de 500 contraintes, notre approche atteint 97% après avoir intégré 150 contraintes (Figure 4). Avec la base « *Cleve* », et avec 500 contraintes aléatoires, l'algorithme COB-b-coloring arrive jusqu'à 89% de précision, notre algorithme arrive jusqu'à 91% avec 300 contraintes probabilistes (Figure 6). Les mêmes résultats (91%) sont obtenus avec CrTM après l'incorporation de 240 contraintes.

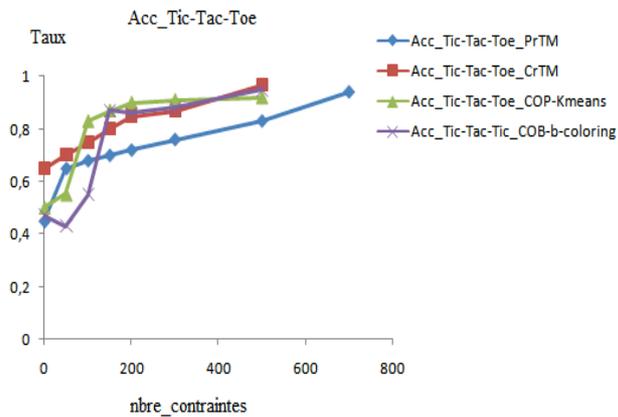


Fig. 5. Comparaison entre PrTM et CrTM et COP-b-coloring sur la base : *Tic-Tac-Toe*

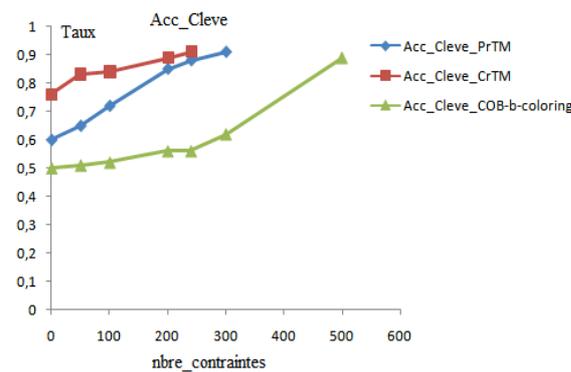


Fig. 6. Comparaison entre PrTM et CrTM, COP-b-coloring et COP\_Kmeans sur la base : *Cleve*

## Classification sous contraintes probabilistes par les cartes topologiques

Par conséquent, PrTM a montré que l'intégration des contraintes probabilistes dans GTM a bien amélioré ce dernier. De plus, les performances du PrTM avoisinent et sont parfois meilleures que celles des méthodes de classification sous contraintes déterministes mais avec en avantage d'un lissage important qui permet d'avoir la représentation d'une observation donnée sur tous les neurones de la carte en présence des contraintes probabilistes.

### 4.2 Visualisation des données

Nous allons projeter maintenant, un sous ensemble de données de la base « *Anneaux* » sur l'espace latent de GTM et celui de PrTM (Figure 7). Les '\*' représentent les observations projetées à partir de l'espace de sortie sur l'espace latent. Les 'o' représentent la projection des moyennes des distributions de ces observations à partir de l'espace de sortie sur l'espace latent (ces moyennes sont représentées par les neurones de cet espace).

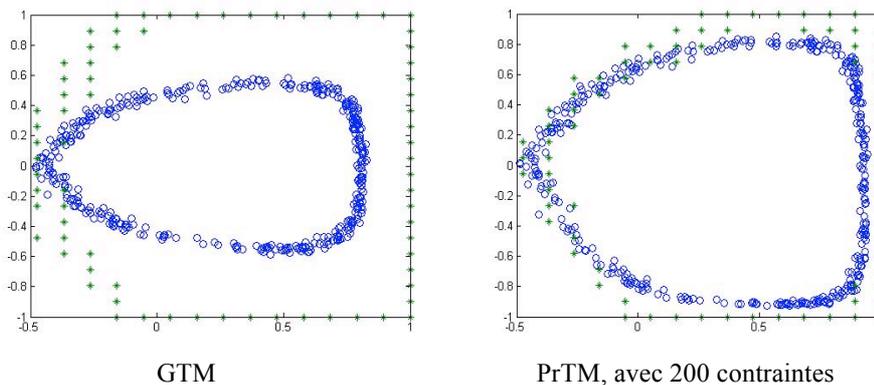


Fig.7. Projection d'un sous-ensemble de la base « *Anneaux* » sur l'espace latent des deux modèles : GTM et PrTM

Nous remarquons que l'intégration des contraintes dans GTM améliore bien la visualisation des données. Pour chaque observation on sélectionne le neurone ayant la probabilité maximale. On peut remarquer clairement dans Figure 7, que l'anneau formé par les neurones de PrTM retrouve sa forme sur celui formé par les observations. Ce qui est moins important avec GTM. On constate que la projection couvre bien la distribution des données dans PrTM que dans GTM.

### 4.3 Cartes topologiques du PrTM

Dans cette section, nous présentons une comparaison des cartes topologiques induites par GTM et celles induites par PrTM. Chaque carte représente l'affectation (distribution a posteriori sur l'espace latent) d'une observation sur tous les neurones. Deux observations

sont donc choisies aléatoirement (observation N°1 et observation N°50) de la base « *Glass* », ces deux dernières ont la même étiquette. Une contrainte probabiliste ML est donc générée entre elles.

GTM produit les cartes illustrées dans Figure 8. Ces deux observations sont un peu distantes dans la base, ce qui explique, sur la figure, la différence des deux cartes. En imposant 200 contraintes probabilistes sur l'ensemble des données, deux nouvelles cartes de PrTM sont obtenues (Figure 9), ces deux cartes sont plus semblables que celles obtenus par GTM puisqu'une contrainte ML est imposée entre les deux observations. On remarque aussi que les probabilités d'affectation des deux individus à certains neurones sont augmentées par rapport à celles de GTM.

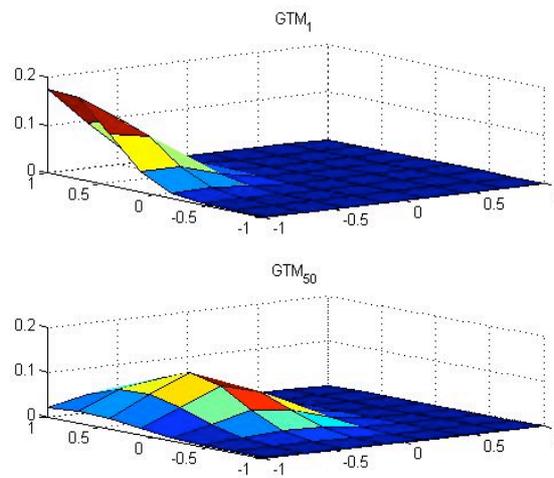


Fig.8. Représentation des observations N°1 et N°50 de la base *Glass* sur une carte GTM

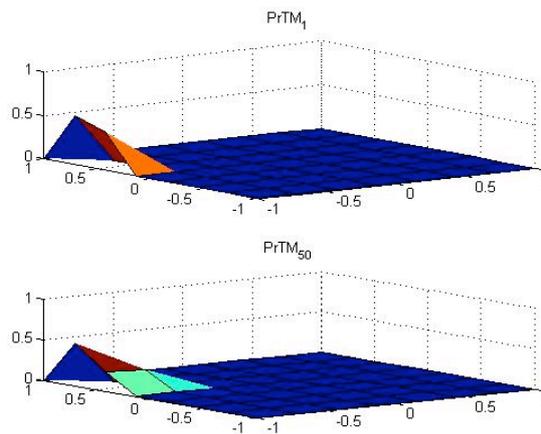


Fig.9. Représentation des observations N°1 et N°50 de la base *Glass* sur une carte PrTM.

## 5. Conclusion

Dans cet article nous avons présenté une nouvelle approche pour l'intégration de contraintes probabilistes dans un processus de classification automatique non supervisée. Nous nous sommes intéressés à l'intégration des contraintes « Soft », positives et négatives. Dans ce contexte, nous avons étudié les propriétés de l'algorithme GTM, un modèle probabiliste de classification automatique non supervisée des cartes topologiques.

En proposant quelques modifications sur cet algorithme, nous avons pu l'adapter aux contraintes probabilistes. Différentes expérimentations nous ont permis d'avoir des résultats très encourageants en comparant notre approche PrTM avec GTM et avec d'autres méthodes de classification sous contraintes.

De nombreuses perspectives peuvent être envisagées comme suite à ce travail. En effet, le même travail peut se faire avec des groupes des contraintes, des contraintes conditionnelles, etc. Aussi, la dynamique peut être intégrée dans la construction du modèle probabiliste pour la classification sous ce type de contraintes.

## Références

- Bellal, F., K. Benabdeslem, A. Aussem. (2008). SOM based clustering with instance level constrains. European Symposium on Artificial Neural Networks, pp 313-318, Bruges (Belgium).
- Benhassena, A., K. Benabdeslem, F. Bellal, A. Aussem et B. Canitia . (2008). Intégration des contraintes dans les cartes auto-organisatrices. 8ème journées francophones :Extraction et gestion des Connaissances, pp 643-648, Nice
- Bishop, CM., M. Svensén, and C. K. I. Williams. (1998). "GTM: the Generative Topographic Mapping." *Neural Computation*, 10(1):215-234.
- Bishop, CM., M. Svensén, and C. K. I. Williams: "Developments of the Generative Topographic Mapping". *Neurocomputing*, 21 (1998) 203, 224.
- Blake, C. et C. Merz. (1998). Uci repository of machine learning databases. Technical report, University of California.
- Elghazel, H., K. Benabdeslem et A. Dussauchoy. (2007). Constrained Graph b-coloring based Clustering approach. 9th International Conference on Data Warehousing and Knowledge Discovery. pp 262-271, Regensburg, Germany.
- Davidson, I and S. S. Ravi. (2005). Clustering with constraints. Feasibility issues and the k-means algorithm." In *SDM*.
- Davilson, I, K. Wagstaff and S. Basu. (2006). Measuring Constraint-Set Utility for Partitional Clustering Algorithms. J. F. urnkranz, T. Scheffer, and M. Spiliopoulou (Eds.): *PKDD , LNAI 4213*, pp. 115–126.
- Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, B 39 (1), 1-38.
- Kamvar, SD., D. Klein and C.D. Manning. (2002). From instance-level constraints to space-level constraints : Making the most of prior knowledge in data clustering. pages 307–314, Sydney, Australia.
- Kohonen, T. (1994). *Self-Organizing Map*. Berlin: Springer.
- Law, M A. Topchy and A. K. Jain. (2004). Clustering with Soft and Group Constraints. *Structural, Syntactic, and Statistical Pattern Recognition*, Springer, LNCS, vol. 3138, Lisbon, pp. 662-670
- Law, M., A. Topchy and A. K. Jain. (2005). Model-based Clustering With Probabilistic Constraints. In *Proceedings of SIAM Data Mining*, pp. 641-645, Newport Beach, CA, USA.

- Lange, T., M. Law, A. Jain and J. Buhmann. (2005). Learning With Constrained and Unlabelled Data. Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Volume 1 , pp 731 - 738
- Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* 66, 846–850.
- Saporta, G. (2006). Probabilités, Analyse des données et Statistiques. Eds Technip, Paris.
- Wagstaff, K, and Claire Cardie. (2000). Clustering with instance-level Constraints. In International conference on machine learning, pp 1103–1110.
- Wagstaff, K, Claire Cardie, Seth Rogers, and Stefan Schrödl. (2001). Constrained kmeans clustering with background knowledge.” In International conference on machine learning, pp 577–584.

**Summary.** This paper describes a new topological map dedicated to clustering under probabilistic constraints. In general, traditional clustering is used in an unsupervised manner. However, in some cases, background information about the problem domain is available or imposed in the form of constraints, in addition to data instances. In this context, we modify the popular GTM algorithm to take these “soft” constraints into account during the construction of the topology. We present experiments on synthetic known databases with artificial constraints for comparison with both GTM and another constrained clustering methods.