Carte auto-organisatrice probabiliste sur données binaires

Rodolphe Priam, Mohamed Nadif

LITA, Université de Metz
Ile du Saulcy, 57045 Metz

Résumé. Les méthodes factorielles d'analyse exploratoire statistique définissent des directions orthogonales informatives à partir d'un ensemble de données. Elles conduisent par exemple à expliquer les proximités entre individus à l'aide d'un groupe de variables caractéristiques. Dans le contexte du datamining lorsque les tableaux de données sont de grande taille, une méthode de cartographie synthétique s'avère intéressante. Ainsi une carte auto-organisatrice (SOM) est une méthode de partitionnement munie d'une structure de graphe de voisinage -sur les classes- le plus souvent planaire. Des travaux récents sont développés pour étendre le SOM probabiliste *Generative Topographic Mapping* (GTM) aux modèles de mélanges classiques pour données discrètes. Dans ce papier nous présentons et étudions un modèle génératif symétrique de carte auto-organisatrice pour données binaires que nous appelons *Bernoulli Aspect Topological Model* (BATM). Nous introduisons un nouveau lissage et accélérons la convergence de l'estimation par une initialisation originale des probabilités en jeu.

1 Introduction

La visualisation des corrélations et similarités principales dans un échantillon de données est l'objectif des méthodes factorielles (Lebart et al., 1984). Ces méthodes cherchent souvent des directions informatives orthogonales dans un nuage de données. Ces directions concentrent l'essentiel de la variance projetée car l'inertie est porteuse de sens. Une décomposition pertinente de l'inertie sur des plans de projection révèle quels individus sont similaires et quelles variables sont dépendantes. Bien que ces méthodes soient très pertinentes, les grands échantillons de données demandent de nouvelles méthodes efficaces pour leur analyse. Dans ce contexte, les cartes de Kohonen (1997) sont connues dans le domaine de l'analyse exploratoire des données pour généraliser les méthodes factorielles telles que la méthode d'Analyse en Composantes Principales ou ACP (Lebart et al., 1984) pour les données continues. Plus généralement, les cartes auto-organisatrices ou SOM (Kohonen, 1997) sont des méthodes de classification avec une contrainte de voisinage sur les classes conférant un sens topologique à la partition finale. Le GTM ou Generative Topographic Mapping (Bishop et al., 1998) est une carte auto-organisatrice probabiliste avec des contraintes sur les moyennes d'un mélange gaussien pour données continues, mais ce modèle est inopérant pour des données catégorielles ou binaires. Des modèles récents (Girolami, 2001; Kabán et Girolami, 2001; Tipping, 1999) ont été proposés pour étendre le GTM aux modèles de mélanges classiques pour données discrètes. Hofmann et Puzicha (1998) ont par contre proposé l'approche du modèle symétrique à *aspects*

qui traite de la classification simultanée des lignes et colonnes d'un tableau de contingence. Cette approche est bénéfique dans plusieurs domaines tels que le textmining et la segmentation d'image. Dans ce papier, nous nous intéressons aux données binaires et nous étudions un modèle original en présentant un nouveau lissage de carte auto-organisatrice et une initialisation adaptée pour accélérer la convergence des algorithmes d'estimation des paramètres du modèle. Les probabilités sont paramétrées de façon adéquate comme le GTM afin d'induire une auto-organisation des facteurs latents, ce qui nous amène à une nouvelle méthode pour visualiser les données discrètes ou vecteurs multidimensionnels de composantes 1/0.

Ce papier est organisé comme suit. Dans la section 2 nous décrivons notre modèle et abordons le problème d'estimation des paramètres par la maximisation de la vraisemblance. Dans la section 3 nous réalisons des expériences numériques pour valider notre modèle. Enfin la section 4 résume les points principaux et présente les travaux en cours.

2 Le modèle BATM

Le modèle proposé repose sur l'hypothèse d'indépendance des $I \times J$ cellules $x_{ij} \in \{0, 1\}$ d'un tableau binaire en modélisant chaque probabilité unidimensionnelle d'observer x_{ij} comme un mélange de K lois de Bernoulli : $Pr(x_{ij} = 1) = \mathbb{E}(x_{ij}) = \sum_k \pi_{ki} a_{jk}$ avec π_{ki} les proportions des K composants telles que $\sum_k \pi_{ki} = 1$. Ce modèle génératif correspond à sélectionner pour chaque ligne $x_i = (x_{i1}, x_{i2}, \cdots, x_{iJ})$ fixée, une distribution discrète π_i de composantes π_{ki}, puis pour chaque j-ième composante x_{ij}, sélectionner un état k avec la probabilité π_{ki} afin de lui attribuer une valeur binaire selon la loi de Bernoulli de paramètre a_{jk}. Un modèle comparable pour la classification sans contrainte a été récemment proposé. La log-vraisemblance de $\mathcal{D} = \{x_i\}_{i=1}^{i=I}$ s'écrit alors :

$$\mathcal{L}(\theta|\mathcal{D}) \quad = \quad \sum_{i,j} \log \Big[\sum_k \pi_{ki} a_{jk}^{x_{ij}} (1 - a_{jk})^{1-x_{ij}} \Big].$$

Afin d'induire une auto-organisation topologique des probabilités, nous considérons les K coordonnées $\{s_k\}_{k=1}^{k=K}$ d'une grille bidimensionnelle régulière qui modélise un plan discrétisé sur lequel l'ensemble des données est disposé par le BATM. La grille est projetée non linéairement dans un espace de plus grande dimension L, par une transformation non linéaire constituée de L bases fonctionnelles ϕ_ℓ, et telle que $\xi_k = (\phi_1(s_k), \phi_2(s_k), \cdots, \phi_L(s_k))^T$; on note la matrice $\Phi = [\xi_1|\xi_2|\cdots|\xi_K]^T$. Les a_{jk} forment alors les noeuds d'une surface non linéaire discrète : les probabilités de Bernoulli sont paramétrées par des fonctions $a_{jk} = \sigma(w_j^T \xi_k)$ où $\sigma(u) = e^u/(1 + e^u)$ est une fonction sigmoïde et w_j un paramètre inconnu appartenant à $\mathbb{R}^L$. La log-vraisemblance devient :

$$\mathcal{L}(\theta|\mathcal{D}) = \sum_{i,j} \log \Big[\sum_k \pi_{ki} \sigma(w_j^T \xi_k)^{x_{ij}} (1 - \sigma(w_j^T \xi_k))^{1-x_{ij}} \Big].$$

Ce modèle s'interprète comme une version binaire cartographique du LSA probabiliste ou pLSA de Hofmann et Puzicha (1998). Les paramètres inconnus sont estimés dans la section suivante.

2.1 Estimation par GEM

L'inférence de notre modèle se réalise en maximisant la log-vraisemblance par une méthode itérative ; une solution analytique exacte n'existe pas en raison des non linéarités du mélange et des fonctions sigmoïdes. Donc nous étudions l'approche de l'algorithme de montée de gradient par EM (Dempster et al., 1977) généralisé (GEM) de McLachlan et Peel (2000). Cette approche suppose la vraisemblance complétée par la connaissance de la partition $\mathcal{Z} = \{\mathcal{Z}_1, \mathcal{Z}_2, \cdots, \mathcal{Z}_K\}$:

$$\mathcal{L}(\theta, Z|\mathcal{D}) \quad = \quad \sum_{i,j} \log\Big[\pi_{z_i i}\, a_{jz_i}^{x_{ij}} (1 - a_{jz_i})^{1-x_{ij}}\Big],$$

avec z_i les variables latentes ayant pour support $\{1, 2 \cdots, K\}$. L'algorithme EM ou *Expectation-Maximisation* repose sur la maximisation de l'espérance conditionnelle sachant les données et les paramètres de l'itération précédente. Ayant $P^{(t)}(Z|\mathcal{D})$ du pas t précédent, nous maximisons au pas $t+1$:

$$\begin{aligned}\mathcal{Q}(\theta|\theta^{(t)}) &= \mathbb{E}_{P^{(t)}(Z|\mathcal{D})}[\mathcal{L}(\theta, Z|\mathcal{D})] \\ &= \sum_{i,j,k} P^{(t)}_{k|i,j,x_{ij}} \Big\{ \log \pi_{ki} + x_{ij} \xi_k^T w_j - \log(1 + exp(\xi_k^T w_j)) \Big\},\end{aligned}$$

avec $P_{k|i,j,x_{ij}} \propto \pi_{ki} a_{jk}^{x_{ij}} (1 - a_{jk})^{1-x_{ij}}$ la probabilité a posteriori que x_{ij} soit générée pas le composant k. Un calcul direct donne $\pi_{ki}^{(t+1)} = argmax_{\pi_{ki}} Q(\theta|\theta^{(t)}) = \sum_j P^{(t)}_{k|i,j,x_{ij}}/J$. Pour résoudre $w^{(t+1)} = argmax_w Q(\theta|\theta^{(t)})$, nous effectuons des dérivations élémentaires du critère qui aboutissent au gradient $\mathbf{Q}_j^{(t)}$ et au bloc de la hessienne $\mathbf{H}_j^{(t)}$. Le pas de Newton-Raphson suivant augmente alors localement la log-vraisemblance :

$$w_j^{(t+1)} = w_j^{(t)} - {\mathbf{H}_j^{(t)}}^{-1} \mathbf{Q}_j^{(t)}.$$

A la convergence du GEM, nous obtenons un estimateur au maximum de vraisemblance noté $\hat{\theta}$. Pour éviter un surapprentissage et une instabilité numérique, nous ajoutons à la fonction $\mathcal{Q}$ un paramètre de régularisation bayésien (MacKay, 1992) : $-\alpha \sum_j w_j^T w_j$. Cette correction ajoute $-\alpha w_j$ au gradient $\mathbf{Q}_j$ et $-\alpha$ à la diagonale de la hessienne $\mathbf{H}_j$. La valeur de l'hyperparamètre α est choisie manuellement comme la plupart du temps dans la littérature, ici nous avons pris $\alpha = 0.01$.

2.2 Formulation IRLS

Nous écrivons l'algorithme de Newton sous une forme matricielle qui est proche d'un pas d'*Iteratively Reweighted Least Squares* ou IRLS (McCullagh et Nelder, 1983) pour la régression logistique. Pour j de 1 à J :

$$\begin{aligned}\mathbf{Q}_j^{(t)} &= \Phi^T \Big[R_j^{(t)} A_j - G_j^{(t)} a_j^{(t)} \Big] - 0.01 w_j^{(t)}, \\ \mathbf{H}_j^{(t)} &= -\Phi^T G_j^{(t)} F_j^{(t)} \Phi - 0.01 \mathbb{I}_L.\end{aligned}$$

Nous avons $R_j^{(t)}$ la matrice de taille $K \times I$ qui compte pour cellules les probabilités a posteriori $P_{k|i,j,x_{ij}}^{(t)}$, la matrice diagonale $G_j^{(t)}$ a pour éléments non nuls $\sum_i P_{k|i,j,x_{ij}}^{(t)}$, A_j est le vecteur de i-ième composante a_{ij}, $a_j^{(t)}$ est un vecteur colonne avec $a_{jk}^{(t)}$ pour k-ième composante, $F_j^{(t)}$ est la matrice diagonale avec $a_{jk}^{(t)}(1 - a_{jk}^{(t)})$ sur sa diagonale et enfin, $\mathbb{I}_L$ est la matrice identité de taille L.

Pour accélérer numériquement l'algorithme, l'approche de Bohning (1993) remplace la matrice exacte, relativement lourde à calculer en pratique, par une matrice alternative fixe. Par exemple, la matrice $\mathbf{B} = -\frac{I}{4}\Phi^T\Phi - 0.01\mathbb{I}_L$, qui est telle que $\mathbf{H}_j^{(t)} \succeq \mathbf{B}$, i.e. $\mathbf{H}_j^{(t)} - \mathbf{B}$ est non négative, symétrique, ce qui permet de maximiser la vraisemblance. Comme la convergence est lente, nous proposons un algorithme de type variationnel alternatif.

2.3 Estimation variationnelle

En suivant la borne[1] (Saul et al., 1996) sur $\log(1 + exp(\xi_k^T w_j))$, nous obtenons le nouveau critère à optimiser :

$$\tilde{\mathcal{Q}}(\theta|\theta^{(t)}) = \sum_{i,j,k} P_{k|i,j,x_{ij}}^{(t)} \quad \Big\{ \quad \log \pi_{ki} + (x_{ij} - 0.5)\xi_k^T w_j$$
$$+ \quad \lambda(\epsilon_j)[(\xi_k^T w_j)^2 - \epsilon_j^2] + 0.5\epsilon_j - \log(1 + exp(\epsilon_j)) \Big\}.$$

Avec $\lambda(\epsilon_j) = -tanh(0.5\epsilon_j)/(4\epsilon_j)$ tel que $\mathcal{Q}(\theta|\theta^{(t)}) \geq \tilde{\mathcal{Q}}(\theta|\theta^{(t)})$ où ϵ_j est un paramètre variationnel à estimer en maximisant $\tilde{\mathcal{Q}}$. En dérivant ce nouveau critère, nous obtenons le pas de maximisation :

$$\epsilon_j^{(t)} = \sqrt{\frac{{w_j^{(t)}}^T \Phi^T G_j^{(t)} \Phi w_j^{(t)}}{I}},$$
$$w_j^{(t+1)} = \Big[-2\lambda(\epsilon_j^{(t)})\Phi^T G_j^{(t)} \Phi - 0.01\mathbb{I}_L \Big]^{-1} \Phi^T R_j^{(t)} A'_j,$$

où A'_j est le vecteur colonne ayant $x_{ij} - 0.5$ pour i-ème composante. Finalement, trois algorithmes, et un quatrième décrit ci-après, sont présentés pour estimer les paramètres du modèle. Ayant éliminé la solution du gradient simple mais inefficace, on constate que l'algorithme IRLS donne la meilleure vraisemblance dans notre cas comme le montre les expériences dans la section suivante.

3 Simulations

Dans cette section, nous abordons tout d'abord deux éléments complémentaires à la méthode proposée, l'initialisation des paramètres du modèle et l'auto-organisation des probabilités sur les lignes afin d'obtenir la meilleure carte projective finale possible. Nous décrivons alors les résultats numériques de nos simulations sur des données binaires réelles.

[1] $\log \sigma(u) \geq u/2 + \lambda(\epsilon)(u^2 - \epsilon^2) + \log \sigma(\epsilon) - \epsilon/2$ pour des raisons de concavité.

3.1 Initialisation du modèle

Des tirages aléatoires répétés des paramètres initiaux sont une solution aux minima locaux que rencontrent les algorithmes basés sur le gradient. Pour obtenir la meilleure convergence possible on procède à une "bonne initialisation". Puisque les cartes de Kohonen sont des généralisations de l'ACP, le premier plan de cette méthode fournit une intéressante première position (Elemento, 1999) des centres de classes de la carte. Notons (X_i^c, Y_i^c) les coordonnées sur le premier plan factoriel de l'ACP (Jolliffe, 2002), AFC (Benzécri, 1992), LSA (Deerwester et al., 1990) ou même celles obtenues suite à une projection non linéaire telle qu'un MDS (Sammon, 1969). Alors une grille régulière est dessinée sur cette première projection et chaque cellule de la grille correspond à un facteur du modèle BATM : x_i est affectée à la $z_i^{(0)}$-ième classe correspondant à la cellule dans laquelle ses coordonnées (X_i^c, Y_i^c) de projection tombent -sur le plan d'initialisation-. Nous initialisons les probabilités de mélange par $\pi_{ki}^{(0)} \propto h(k, z_i^{(0)})$ pour une fonction de lissage telle que celle de voisinage des cartes de Kohonen, i.e. $h(k, z_i^{(0)}) \propto exp(-||s_k - s_{z_i^{(0)}}||^2/\sigma)$ pour σ bien choisi. Alors on pose :

$$a_{jk}^{(0)} = \frac{\sum_i \pi_{ki}^{(0)} x_{ij} + \alpha}{\sum_i \pi_{ki}^{(0)} + I\alpha},$$

où $\alpha > 0$, bien choisi, a pour rôle de régulariser les paramètres a_{jk} qui correspondent à des cellules vides. Finalement, $LP_{\mathcal{J}}^{(0)}$ est la matrice de taille $K \times J$ dont les cellules ont pour valeurs $\log[a_{jk}^{(0)}/(1 - a_{jk}^{(0)})]$. Cette matrice nous permet d'évaluer une matrice $W_{\mathcal{J}}^{(0)}$ qui a pour colonnes les paramètres initiaux $w_j^{(0)}$ des fonctions logistiques. La solution au problème de régression associé s'écrit alors :

$$W_{\mathcal{J}}^{(0)} = [w_1^{(0)}|w_2^{(0)}|\cdots|w_J^{(0)}] = (\Phi^T\Phi)^{-1}\Phi^T LP_{\mathcal{J}}^{(0)}.$$

Nous construisons les centres en effectuant un pas de l'algorithme non séquentiel de Kohonen pour les affectations évaluées sur le plan de projection initiale : l'affectation s'effectue en attribuant le label $z_i^{(0)}$ du noeud le plus proche de (X_i^c, Y_i^c) pour un treillis régulier dessiné sur le nuage bidimensionnel des projetés afin de discrétiser celui-ci. Cette approche doit également induire par ailleurs un niveau d'entropie plus élevé de la classification initiale des données -comparativement à la simple classification dure sur le plan initial- et facilite ainsi une convergence des paramètres vers une solution encore meilleure.

3.2 Lissage des paramètres sur les lignes

Il peut être intéressant d'ajouter une contrainte topologique sur les paramètres partitionnant les lignes afin d'accélérer la convergence de l'algorithme et améliorer la carte finale. Comme une solution par un soft-max (Bishop, 1995) nous apparaît relativement lourde, nous proposons une solution alternative en ajoutant un simple lissage par un terme de pénalisation issu de l'approche du TNEM (Priam, 2003). Brièvement, il s'agit de classer les vecteurs de données avec un lissage spatial sur les composantes π_{ki} du modèle BATM, à la manière d'un champ de Markov caché (Zhang, 1992; Celeux et al., 2003; Ambroise et Govaert, 1998). On pose :

$$\mathcal{Q}_\beta(\theta|\theta^{(t)}) \quad = \quad \mathcal{Q}(\theta|\theta^{(t)}) + \frac{\beta}{2}\sum_i \pi_i^T \mathbf{V}\pi_i,$$

où π_i est le vecteur de composantes π_{ki}, et $\mathbf{V}$ est -soit la matrice des fonctions de voisinage de la carte auto-organisatrice, i.e. $V_{k\ell} = h(k, \ell)$, -soit la matrice binaire d'adjacence du treillis correspondant à notre carte probabiliste, i.e. $V_{k\ell} = 1$ ssi le k-ième noeud est voisin du ℓ-ième. Le pas complémentaire associé s'écrit :

$$\pi_{ki}^{(t+1)} = \frac{\sum_j P_{k|i,j,x_{ij}}^{(t)} + \beta\, \pi_{ki}^{(t+1)} \sum_\ell V_{k\ell}\, \pi_{\ell i}^{(t+1)}}{J + \beta\, \pi_i^{(t+1)^T} \mathbf{V} \pi_i^{(t+1)}},$$

et se résout en itérant l'égalité et en réinjectant dans le membre de droite les anciennes valeurs courantes des $\pi_{ki}^{(t+1)}$ jusqu'à ce que la stabilisation de leurs valeurs soit atteinte. Nous retrouvons évidemment le pas d'estimation non contrainte en annulant β. Nous avons éliminé le terme additif du TNEM qui porte sur les entropies des π_i, donc nous obtenons un nouvel algorithme appelé TNEM2, plus général que le TNEM original puisqu'il s'applique à des probabilités non forcément a posteriori. Une alternative au TNEM est une estimation des π_{ki} paramétrés comme le GTM. La fonction à optimiser s'écrit alors $\mathcal{Q}_I(\theta|\theta^{(t)}) = \sum_{i,j,k} P_{k|i,j,x_{ij}}^{(t)} \Big\{ \xi_k^T w_i - \log \sum_\ell exp(\xi_\ell^T w_i) \Big\}$ où les w_i sont les inconnues à déterminer. Leur estimation s'effectue comme précédemment, par une montée de gradient en réalisant une boucle sur l'indice i des lignes, de 1 à I, et en calculant les gradients $\mathbf{Q}_i^{(t)}$ et les matrices hessiennes $\mathbf{H}_i^{(t)}$. Enfin, les paramètres $w_i^{(0)}$ s'initialisent à l'aide d'une régression sur la nouvelle matrice $LP_{\mathcal{I}}^{(0)}$ qui a pour cellules les logarithmes des $\pi_{ki}^{(0)}$. Nous proposons finalement le quatrième algorithme d'estimation noté IRLS+TNEM2 qui associe une maximisation sur les paramètres des colonnes par l'IRLS à un lissage des probabilités des lignes par le TNEM2. Nous expliquons dans la suite comment ce lissage se comporte en pratique.

3.3 Post-processing de la carte finale

La carte finale montre une grille de centres de classes bien organisées ; à chacune on affecte les données dont elle est le plus proche. Pour les cartes auto-organisatrices classiques on utilise la distance euclidienne entre le vecteur centre et le vecteur donnée. Ici le modèle permet une alternative probabiliste puisque nous avons la probabilité de génération d'une donnée par un composant k du mélange. Donc chacun des vecteurs x_i est affecté à un centre par un maximum a posteriori (MAP), i.e. $\hat{z}_i = argmax_k\, \hat{\pi}_{ki}$. De la même manière, la j-ième variable est affectée au centre de label $\hat{z}_j = argmax_k\, \hat{a}_{jk}$. Le MAP aboutit aux positions bidimensionnelles $p_i = s_{\hat{z}_i}$ et $p_j = s_{\hat{z}_j}$ pour les lignes et colonnes de la matrice projetée. Une seconde manière de projeter chaque donnée est par sa position moyenne (Bishop et al., 1998) au lieu du MAP précédent, i.e. les positions $\tilde{p}_i = \sum_k \hat{\pi}_{ki} s_k$ et $\tilde{p}_j = \sum_k (\hat{a}_{jk} / \sum_\ell \hat{a}_{j\ell}) s_k$.

3.4 Expériences

Nous expérimentons notre modèle sur plusieurs échantillons de données pour valider notre approche, par exemple, sur l'échantillon Zoo[2], qui compte 101 animaux avec sept classes et 21 caractéristiques binaires. Notre méthode BATM converge vers une carte bien organisée où

[2] ftp ://ftp.ics.uci.edu/pub/machine-learning-databases/zoo/zoo.names

l'on reconnaît les sept classes que l'algorithme a projetées. La segmentation de la grille sur la figure 1 s'effectue à l'aide d'une procédure automatique consistant (Vesanto et Alhoniemi, 2000) en une classification ascendante hiérarchique avec agrégation par le diamètre (*complete-linkage*) associée à une distance du χ^2 sur la matrice des $\hat{\pi}_{ki}$, qui donne les meilleurs résultats en pratique. En remarque, la classe contenant les reptiles est peu homogène d'après nos expériences, car ces animaux se regroupent mal. L'évolution de la log-vraisemblance par les quatre algorithmes est présentée à la figure 2 pour le tableau du Zoo, démontrant la supériorité de l'algorithme IRLS comparativement à des algorithmes plus récents alternatifs. Cependant, un surapprentissage peut amener à une solution non suffisamment lissée, et nous préférons l'approche IRLS+TNEM2 à cause de son efficacité malgré sa vraisemblance moins élevée. Cette valeur plus faible s'explique par le terme de pénalisation qui aide à une plus rapide et meilleure auto-organisation des lignes comme vérifiée ici puisque cet algorithme s'arrête bien plus tôt que les trois autres, pour un critère d'arrêt identique (log-vraisemblance relative inférieure au seuil 10e-5). Notre initialisation originale par une régression adéquate se base sur le premier plan principal d'une Analyse des Correspondances (AFC). Celle-ci s'illustre sur la figure 3, démontrant l'intérêt d'une carte auto-organisatrice : alors que la méthode factorielle linéaire n'est pas capable de montrer les sept classes sur ce premier plan, notre carte par BATM extrait ces sept classes et trouve leur lien statistique grâce à la propriété de voisinage. Un ensemble de données textuelles est également projeté. Cette base est un échantillon du fichier Classic3 (Dhillon et al., 2003) qui compte trois classes (MEDLINE, CISI, CRANFIELD). Nous avons tiré aléatoirement (tirage équiprobable sans remise) 450 documents de ce fichier en prenant 150 documents dans chaque classe. Nous avons sélectionné les termes dont la fréquence totale est supérieure à 30 sur le corpus entier et pour l'ensemble du vocabulaire de 4303 termes. Nous aboutissons, en éliminant les textes vides, à une matrice de taille aléatoire : 450 par 170 environ. Nous montrons les positions moyennes des labels des documents correspondants et projetés sur la figure 4 pour l'une de ces matrices. Nous sommes en mesure de visualiser assez distinctement les trois classes séparées par notre projection non linéaire.

4 Conclusion et discussion

Nous avons présenté une nouvelle méthode de carte auto-organisatrice -récapitulée sur la figure 5- pour données binaires, comme on peut en trouver dans le domaine du traitement de l'image et du texte. De nouveaux résultats pour l'initialisation d'une méthode de projection probabiliste de données qualitatives a également été introduit.

Une perspective de nos travaux est la construction de biplots non linéaires par carte topologique. Nous travaillons actuellement à la projection de matrices de taille plus importante ainsi que sur des ensembles d'images binaires qui donnent des résultats encourageants. Ensuite des variantes au modèle BATM se posent en remplaçant $\mathbb{E}(x_{ij})$ par une hypothèse alternative, i.e. un mélange de lois différent tel que par exemple $\mathbb{E}(x_{ij}) = \sum_k \pi_k \pi_{i|k} b_{jk}^{x_{ij}} (1 - b_{jk})^{(1-x_{ij})}$ ou bien $\mathbb{E}(x_{ij}) = \sum_k \pi_k a_{ik}^{x_{ij}} (1 - a_{ik})^{(1-x_{ij})} b_{jk}^{x_{ij}} (1 - b_{jk})^{(1-x_{ij})}$. Le modèle BATM s'étend également à d'autres types de données comme il est proposé dans le paragraphe suivant. L'estimation peut encore être améliorée en déterminant notamment le meilleur hyperparamètre β. En conclusion, le récent modèle du *Block Clustering* (Govaert et Nadif, 2003, 2005) effectue une classification simultanée des lignes et colonnes d'un tableau numérique en étendant le modèle

		stingray	bass catfish chub dogfish herring pike piranha tuna				
	seawasp		carp haddock seahorse sole		dolphin porpoise	seal	
octopus	clam starfish		pitviper seasnake slowworm			sealion	
crayfish lobster	crab scorpion slug worm		frog frog newt toad		platypus	mink	
flea gnat honeybee housefly ladybird moth termite wasp			tortoise	tuatara		aardvark bear mole opossum hare vole	boar cheetah leopard lion lynx mongoose polecat puma raccoon wolf antelope buffalo deer elephant giraffe oryx
	crow duck gull hawk skimmer skua	kiwi penguin swan	ostrich rhea		gorilla squirrel wallaby	cavy	calf goat hamster pony pussycat reindeer
		chicken dove flamingo lark parakeet pheasant sparrow vulture wren			fruitbat vampire	girl	

FIG. 1 – *Une carte de taille* 8×8*, pour l'échantillon Zoo par BATM, segmentée en sept macro-classes représentées visuellement par sept niveaux de gris au niveau des cellules agrégées.*

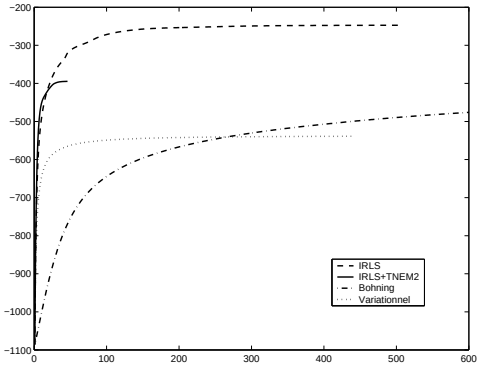

FIG. 2 – *Les courbes de la log-vraisemblance du BATM obtenues par les quatre algorithmes présentés (IRLS, IRLS+TNEM2, Bohning (incomplète) et variationnel) pour les 101 animaux.*

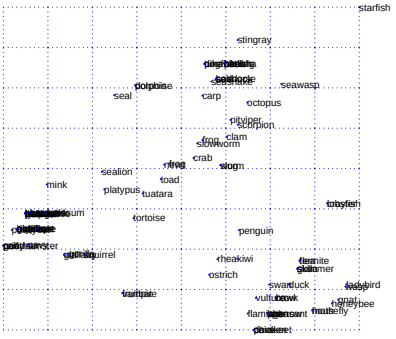

FIG. 3 – *Initialisation de la carte BATM pour les 101 animaux.*

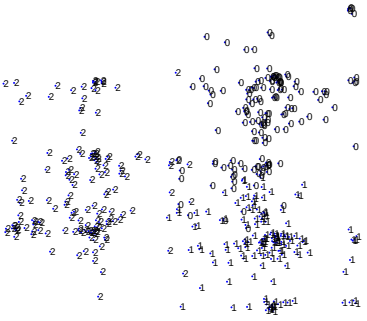

FIG. 4 – *Projections moyennes de l'échantillon des 450 documents dans Classic3.*

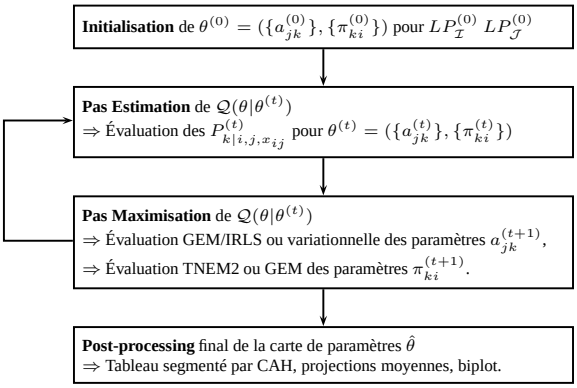

FIG. 5 – *Schéma récapitulatif de la méthode BATM.*

de mélange classique à un modèle de mélange croisé. Celui-ci est un modèle génératif flexible qui s'avère une alternative efficace et prometteuse au modèle à aspects. Il serait intéressant de l'étendre en lui ajoutant une propriété d'auto-organisation.

Annexe : paramétrisation probabiliste alternative au *soft-max*

Lorsque la matrice de données est un tableau de contingence, la loi de Bernoulli n'est plus valable, et une hypothèse de loi multinomiale est généralement supposée. Un paramétrage soft-max est alors introduit pour le cas de probabilités contraintes en régression et classification notamment. On écrit dans notre cas $p_{j|k} = e^{w_j^T \xi_k} / \sum_{j'} e^{w_{j'}^T \xi_k}$ avec $\sum_j p_{j|k} = 1$. Donc cette paramétrisation implique l'inversion d'une matrice hessienne pleine pour procéder à l'optimisation. Nous proposons un moyen alternatif plus efficace. L'idée principale est d'aboutir à de nouveaux paramètres -sans la contrainte de somme à l'unité classique pour une multinomiale- en écrivant $p_{j|k}$ comme une loi jointe de variables de loi de Bernoulli de paramètres inconnus. Il s'agit d'écrire la loi jointe de la j-ième colonne associée en mettant à l'unité la composante qui nous intéresse, et à zéro les autres, puis en supposant des lois de Bernoulli sur l'ensemble des composantes prises indépendantes pour le vecteur binaire résultant. L'expression $p_{j|k} = p_{jk} \prod_{j' \neq j} (1 - p_{j'k}) \simeq p_{jk}$ avec $p_{jk} \in [0, 1]$ donne une solution valide au maximum de vraisemblance d'une loi multinomiale, pour des probabilités assez petites (éventuellement par l'ajout de composantes artificielles supplémentaires pour diminuer les valeurs), i.e. $p_{jk}^{(t+1)}$ (optimum global) annule la dérivée -sans contrainte donc sans lagrangien- de :

$$\sum_i \sum_j \sum_k p_{kij}^{(t)} x_{ij} \log[p_{j|k}] = \sum_i \sum_j \sum_k p_{kij}^{(t)} x_{ij} \log \left[p_{jk} \prod_{j' \neq j} (1 - p_{j'k}) \right].$$

Nous retrouvons l'expression classique de l'estimation des paramètres de la loi multinomiale, $p_{jk}^{(t+1)} = \frac{\sum_i p_{kij}^{(t)} x_{ij}}{\sum_i \sum_{j'} p_{kij'}^{(t)} x_{ij'}}$, pour des probabilités a posteriori $p_{kij}^{(t)}$ et induisant leur normalisation

automatique. Comme aucune contrainte ne devient nécessaire sur les p_{jk}, la paramétrisation par des sigmoïdes $p_{jk} = \sigma(\xi_k^T w_j)$ est licite pour une auto-organisation des valeurs recherchées, sans paramètre soft-max. Nous aboutissons à une formulation IRLS (ou variationnelle) très générale pour la loi multinomiale, d'où une nouvelle expression du vecteur gradient et de la matrice hessienne pour l'estimation du modèle BATM sur données catégorielles.

Références

Ambroise, C. et G. Govaert (1998). Convergence of an em-type algorithm for spatial clustering. *Pattern Recogn. Lett. 19*(10), 919–927.

Benzécri, J. P. (1992). *Correspondence Analysis Handbook*. New-York : Dekker.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Clarendon Press.

Bishop, C. M., M. Svensén, et C. K. I. Williams (1998). Developpements of generative topographic mapping. *Neurocomputing 21*, 203–224.

Bohning, D. (1993). Construction of reliable maximum likelihood algorithms with application to logistic and cox regression. *Handbook of Statistics 9*, 409–422.

Celeux, G., F. Forbes, et N. Peyrard (2003). Em procedures using mean field-like approximations for markov model-based image segmentation. *Pattern Recognition 36*, 131–144.

Deerwester, S., S. T. Dumais, G. W. Furnas, T. Landauer, et R. Harshman (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41*(6), 391–407.

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

Dhillon, I. S., S. Mallela, et D. S. Modha (2003). Information-theoretic co-clustering. In *Proceedings of The Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD-2003)*, pp. 89–98.

Elemento, O. (1999). Initialisation, convergence, et validation de cartes topologiques de kohonen (in french). Master's thesis, Rapport de DEA (INRIA, Yves Lechevallier).

Girolami, M. (2001). Document representation based on generative multivariate bernoulli latent topics models. In U. of Cambridge (Ed.), *BCS-IRSG 22nd Annual Colloquium on Information Retrieval Research*, pp. 194–201.

Govaert, G. et M. Nadif (2003). Clustering with block mixture models. *Pattern Recognition 36*(2), 463–473.

Govaert, G. et M. Nadif (2005). An EM algorithm for the block mixture model. *IEEE Transactions on Pattern Analysis and Machine Intelligence 27*(4), 643–647.

Hofmann, T. et J. Puzicha (1998). Statistical models for co-occurrence data. Technical Report AIM-1625, MIT.

Jolliffe, I. (2002). *Principal Component Analysis*. Springer Verlag.

Kabán, A. et M. Girolami (2001). A combined latent class and trait model for analysis and visualisation of discrete data. *IEEE Transactions on Pattern Analysis and Machine Intelli-*

gence 23(8), 859–872.

Kohonen, T. (1997). *Self-organizing maps*. Springer.

Lebart, L., A. Morineau, et K. Warwick (1984). *Multivariate Descriptive Statistical Analysis*. J. Wiley.

MacKay, D. J. C. (1992). Bayesian interpolation. *Neural Computation 4*(3), 415–447.

McCullagh, P. et J. Nelder (1983). *Generalized linear models*. London : Chapman and Hall.

McLachlan, G. J. et D. Peel (2000). *Finite Mixture Models*. New York : John Wiley and Sons.

Priam, R. (2003). *Méthodes de carte auto organisatrice par mélange de lois contraintes. Application à l'exploration dans les tableaux de contingence textuels (in french)*. Ph. D. thesis, Université de Rennes 1.

Sammon, J. (1969). A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers 5*(18C), 401–409.

Saul, L. K., T. Jaakkola, et M. I. Jordan (1996). Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research 4*, 61–76.

Tipping, M. E. (1999). Probabilistic visualisation of high-dimensional binary data. *Advances in Neural Information Processing Systems*, 592–598.

Vesanto, J. et E. Alhoniemi (2000). Clustering of the self-organizing map. *IEEE Trans. on Neural Networks 11*(3), 586–600.

Zhang, J. (1992). The mean field theory in EM procedures for markov random fields. *IEEE Transactions on Signal Processing 10*(40), 2570–2583.

Summary

The mixture models behave very well to cluster large samples of continuous or categorical data. Adding a vicinity constraint permits them to project data like factorial methods but in a nonlinear way. In this paper we present a new model called *Bernoulli Aspect Topological Mapping* (BATM) : a generative self-organizing map to deal with binary data by an automatic map smoothing and an original initialization.