Pattern Recognition Letters 31 (2010) 830–836

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Clustering based on random graph model embedding vertex features

Hugo Zanghi [a,*], Stevenn Volant [b], Christophe Ambroise [c]

[a] Exalead, 10 place de la Madeleine, 75008 Paris, France
[b] Agroparistech (UMR 518), 16 rue Claude Bernard, 75231 Paris, France
[c] Statistique et Génome (UMR CNRS 8071, INRA 1152), La genopole Tour Evry 2, 523 place des Terrasses, 91000 Evry, France

ARTICLE INFO

Article history:
Received 23 July 2009
Received in revised form 21 December 2009
Available online 1 February 2010

Communicated by R.P.W. Duin

Keywords:
Variational EM algoritm
Graph clustering
Vertex features

ABSTRACT

Large datasets with interactions between objects are common to numerous scientific fields including the social sciences and biology, as well as being a feature of specific phenomena such as the internet. The interactions naturally define a graph, and a common way of exploring and summarizing such datasets is graph clustering. Most techniques for clustering graph vertices use only the topology of connections, while ignoring information about the vertices' features. In this paper we provide a clustering algorithm that harnesses both types of data, based on a statistical model with a latent structure characterizing each vertex both by a vector of features and by its connectivity. We perform simulations to compare our algorithm with existing approaches, and also evaluate our method using real datasets based on hypertext documents. We find that our algorithm successfully exploits whatever information is found both in the connectivity pattern and in the features.

1. Introduction

Classical data analysis has been developed for sets of objects with features, but when explicit relationships exist between objects, classical data analysis cannot take these relationships into account. Much recent research has, however, been concerned with analyzing graphs, for example when seeking relationships in the social sciences, studying gene interactions in biology, or analyzing hyperlinks in computer science, which has led to a deeper awareness of the nature of the interactions in these different networks (Schenker et al., 2005; Cook and Holder, 2007). Many approaches to graph analysis have been proposed. Model-based approaches, i.e., methods which rely on a statistical model of network edges and vertices, such as those first proposed by Erdös–Rényi, can often provide insights into the structure of networks, enabling deductions to be made regarding their internal properties.

An interesting alternative to using the basic Erdös–Rényi model (often ill-suited to real networks) is to consider a mixture of distributions (Frank and Harary, 1982; Snijders and Nowicki, 1997; Newman and Leicht, 2007; Daudin et al., 2008) where it is assumed that nodes are spread over an unknown number of latent connectivity classes. Conditional on the hidden class label, edges are still independent and Bernoulli distributed, but their marginal distribution is a mixture of Bernoulli distributions with strong dependencies between the edges. Several names have been suggested for this model, and here we have chosen to use the term MixNet, which is equivalent to the Block Clustering of Snijders and Nowicki (1997). Block Clustering for classical binary data can be dated back to early work in the seventies (Lorrain and White, 1971; Govaert, 1977).

In addition to the network information used in the methods mentioned above, vertex content will sometimes be available. A typical example is the World Wide Web, which can be described either in terms of the hyperlinks between web pages or by the words occurring in the web pages: each vertex represents a web page containing occurrences of certain words, and each directed edge represents a hyperlink. The additional information corresponding to the vertex features is rarely used in network clustering, but can provide crucial information. Here we combine information from vertex content, traditionally used in classical data analysis, with information inherent in the graph structure, with the aim of clustering objects into coherent groups. Our paper proposes a statistical model, `CohsMix` (for Covariates on hidden structure using Mixture models), which considers the dependent nature of the data and the relation with vertex features (or covariates) in order to capture a hidden structure.

Considering spatial or relational data neighborhoods is not an original approach in clustering. For instance, Hidden Markov Random Fields (HMRF) are well adapted to handling spatial data and are widely used in image analysis. When the spatial network is not given, it is generally obtained using Delaunay triangulation (Ambroise et al., 1997).

Hoff (2003) proposed a new way of dealing with covariates. He suggested modeling the expected value of the relational ties with a

* Corresponding author. Tel.: +33 (0) 1 55 35 27 36; fax: +33 (0) 1 55 35 26 27.
E-mail addresses: hugo.zanghi@exalead.com (H. Zanghi), stevenn.volant@agroparistech.fr (S. Volant), cambroise@genopole.cnrs.fr (C. Ambroise).

0167-8655/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2010.01.026

logistic regression. The problem with this method is the dependency between the observations conditional on the regression parameters and the covariates. He therefore proposed incorporating random effect structures in a generalized linear model setting. The distribution of dependencies among the random effects determines the dependencies among the edges.

There are also approaches based on non-statistical frameworks. In particular, there is clearly a strong similarity between multiple view and graph models with covariates. Multiple view learning algorithms (Ruping and Scheffer, 2005) consider instances which have multiple representations and make use of these views simultaneously so as to obtain a consensus partition.

The second section introduces our proposed model, which is an extension of the MixNet model. Since the model considers a great number of dependencies, the proposed estimation scheme includes a variational approach of the EM algorithm, which can deal with larger networks than the Bayesian framework is able to handle. We then introduce practical strategies for initializing and choosing the number of groups. In the third section extensive simulations illustrate the efficiency of the this algorithm, and real datasets dealing with hypertext documents are examined.

A R package named `CohsMix` is available upon request.

2. Mixing network and covariates

This section introduces the proposed model. We have chosen a model which assumes, first, that the covariates and the edges conditional on the node classes are mutually independent and, secondly, that both the connectivity pattern and the vertex features can be explained by the class. In the web context this model considers that a given class contains documents that are similar not only in the words they contain, but also in their connectivity patterns with documents inside and outside the class. Although this assumption does not explicitly model the idea that authors tend to link similar topics (words that occur) thus creating a thematic locality (Davison, 2000), it nevertheless allows clusters with local themes to be detected. Its simplicity makes it a robust model well suited to the real web.

2.1. Models and notation

Let us define a random graph G, where $\mathcal{V}$ denotes the set of vertices. Based on the MixNet model, our model assumes that $\mathcal{V}$ is partitioned into Q hidden classes. Let us denote by Z_{iq} the indicator variable such that $\{Z_{iq}=1\}$ if node i belongs to class q. $\boldsymbol{Z}=(\boldsymbol{Z}_1,\ldots,\boldsymbol{Z}_n)$ is the vector of random independent indicator variables such that

$$\boldsymbol{Z}_i \sim \mathcal{M}(1,\boldsymbol{\alpha}=\{\alpha_1,\ldots,\alpha_Q\}), \tag{1}$$

with $\boldsymbol{\alpha}$ the vector of class proportions. Edges are Bernoulli random variables

$$X_{ij}|Z_{iq}Z_{jl}=1 \sim \mathcal{B}(\pi_{ql}), \tag{2}$$

conditionally independent, given the node classes

$$P(\boldsymbol{X}|\boldsymbol{Z})=\prod_{ij}\prod_{q,l}P(X_{ij}|Z_{iq}Z_{jl}=1)^{Z_{iq}Z_{jl}}.$$

In this paper we consider an undirected graph. We assume that there are no self-loops, i.e. a node cannot be connected to itself ($X_{ii}=0$). Nevertheless, the method can easily be generalized to encompass directed graphs with self-loops.

2.1.1. Vertex features

We shall consider n objects described both by their connections and p features. The data can consequently be represented in different forms. One might, for example, wish to characterize each object using a two-part vector, where the first part contains the feature of the object $\boldsymbol{Y}_i$ and the second part contains a binary vector representing the connection to all $n-1$ other objects $\boldsymbol{X}_i$. Continuing our example of the World Wide Web, web pages can be viewed either as a vector of word-occurrences with hyperlinks, or as two matrices, one based on the adjacency matrix describing the topology of the graph generated by the hyperlinks and the other by the features matrix generated by the word-occurrences in each web page.

We consider that the p-dimensional feature vector corresponding to object i is defined by:

$$\boldsymbol{Y}_i=\begin{pmatrix} Y_i^{(1)} \\ Y_i^{(2)} \\ \vdots \\ Y_i^{(p)} \end{pmatrix}.$$

We assume that the feature vectors $\boldsymbol{Y}_i$ are multivariate normally-distributed conditionally on the latent structure

$$\boldsymbol{Y}_i|Z_{iq}=1 \sim \mathcal{N}(\boldsymbol{\mu}_q,\Sigma_q), \tag{3}$$

where

$\boldsymbol{\mu}_q=\begin{pmatrix} \mu_q^{(1)} \\ \mu_q^{(2)} \\ \vdots \\ \mu_q^{(p)} \end{pmatrix}$ and $\boldsymbol{\Sigma_q}=\sigma I$ the covariance matrix is proportional to the identity.

Notice that this assumption is not systemically supported by the data. As the class structure is not available beforehand, assuming that the data is normally distributed within each class, is difficult to check *a priori*. It thus would be a reasonable practice to check *a posteriori*.

The random feature vectors $\boldsymbol{Y}_i$ are conditionally independent, given the node classes

$$P(\boldsymbol{Y}|\boldsymbol{Z})=\prod_i\prod_q P(\boldsymbol{Y}_i|Z_{iq})^{Z_{iq}}.$$

The conditional distribution corresponding to covariates can be written as follows:

$$\begin{aligned}&\log P(\boldsymbol{Y}|\boldsymbol{Z})\\ &=\sum_i\sum_q Z_{iq}\log P(\boldsymbol{Y}_i|Z_{iq})\\ &=\sum_i\sum_q Z_{iq}\left[\left(\log\frac{1}{2\pi^{\frac{n}{2}}det(\Sigma)^{\frac{1}{2}}}\right)-\frac{1}{2}(\boldsymbol{Y}_i-\boldsymbol{\mu}_q)^T\sigma^{-1}(\boldsymbol{Y}_i-\boldsymbol{\mu}_q)\right].\end{aligned}$$

The proposed mixture model assumes an independence of $\boldsymbol{X}$ and $\boldsymbol{Y}$ conditional on $\boldsymbol{Z}$. Given this independence between edges and covariates, the complete log-likelihood can be written as (Fig. 1):

$$P(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z})=P(Z)P(\boldsymbol{X},\boldsymbol{Y}|Z)=P(\boldsymbol{Z})P(\boldsymbol{Y}|\boldsymbol{Z})P(\boldsymbol{X}|\boldsymbol{Z}).$$

The following section proposes an estimation scheme for the CohsMix model.

2.2. Variational EM algorithm for CohsMix

In the classical EM framework developed by Dempster et al. (1977), where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the available data, inferring the unknown parameters $\boldsymbol{\Theta}$ spread over a latent structure $\boldsymbol{Z}$ involves the following conditional expectation:

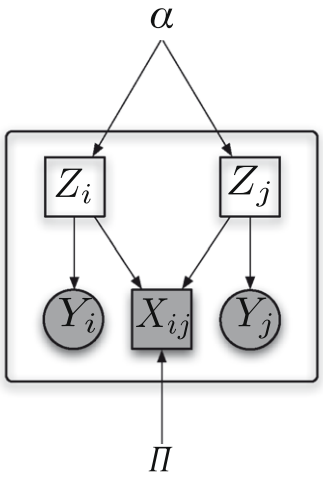

Fig. 1. Graphical representation of the CohsMix model. The squares represent discrete random variables and the circles continuous random variables.

$$Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(m)}) = \mathbb{E}\{\log \mathcal{L}_c(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z};\boldsymbol{\Theta})|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta}^{(m)}\} = \sum_{\boldsymbol{Z}\in\mathcal{Z}} \mathbb{P}(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta}^{(m)}) \log \mathcal{L}_c(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z};\boldsymbol{\Theta}) \quad (4)$$

where

$$\boldsymbol{\Theta}^{(m+1)} = \underset{\boldsymbol{\Theta}}{\text{Argmax}}\, Q(\boldsymbol{\Theta},\boldsymbol{\Theta}^{(m)}).$$

The usual EM strategy would be to alternate an E-step computing the conditional expectation (4) with an M-step maximizing this quantity over the parameter of interest $\boldsymbol{\Theta}$. Unfortunately, no closed form of $Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(m)})$ can be formulated in the present case. The technical difficulty lies in the complex dependency structure of the model. Indeed, $\mathbb{P}(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta})$ cannot be factorized, as argued in (Daudin et al., 2008). This makes the direct calculation of $Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(m)})$ impossible. To tackle this problem we use a variational approach (see, e.g. (Jordan et al., 1999), for elementary results on variational methods). In this framework, the conditional distribution of the latent variables $\mathbb{P}(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta}^{(m)})$ is approximated by a more convenient distribution denoted by $R(\boldsymbol{Z})$, which is chosen carefully in order to be tractable. Hence, our EM-like algorithm includes the following approximation of the conditional expectation (4)

$$\mathbb{E}_R\{\log \mathcal{L}(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z};\boldsymbol{\Theta})\} = \sum_{\boldsymbol{Z}\in\mathcal{Z}} R(\boldsymbol{Z}) \log \mathcal{L}(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z};\boldsymbol{\Theta}). \quad (5)$$

In the following section we develop a variational argument in order to choose an approximation $R(\boldsymbol{Z})$ of $\mathbb{P}(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta}^{(m)})$. This enables us to compute the conditional expectation (5) and proceed to the maximization step.

2.3. Variational estimation of the latent structure (E-step)

In this part $\boldsymbol{\Theta}$ is assumed to be known, and we are looking for an approximate distribution $R(\cdot)$ of the latent variables. The variational approach consists in maximizing a lower bound $\mathcal{J}$ of the log-likelihood $\log \mathbb{P}(\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta})$, defined as follows:

$$\mathcal{J}(\boldsymbol{\Theta}) = \log \mathbb{P}(\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta}) - \mathrm{D}_{KL}\{R(\boldsymbol{Z})\|\mathbb{P}(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta}^{(m)})\} \quad (6)$$

where D_{KL} is the Küllback–Leibler divergence. This measures the difference between the probability distribution $\mathbb{P}(\cdot|\boldsymbol{\Theta})$ in the underlying model and its approximation $R(\cdot)$. An intuitively straightforward choice for $R(\cdot)$ is a completely factorized distribution (see (Mariadassou and Robin, 2007; Zanghi et al., 2008))

$$R(\boldsymbol{Z}) = \prod_{i\in\mathcal{P}} h_{\tau_i}(\boldsymbol{Z}_i), \quad (7)$$

where h_{τ_i} is the density of the multinomial probability distribution $\mathcal{M}(1;\boldsymbol{\tau}_i)$, and $\boldsymbol{\tau}_i = (\tau_{i1},\ldots,\tau_{iQ})$ is a random vector containing the variational parameters to optimize. The complete set of parameters $\boldsymbol{\tau} = \{\tau_{iq}\}_{i\in\mathcal{P},q\in\mathcal{Q}}$ is what we are seeking to obtain via the variational inference. In the case in hand the variational approach intuitively operates as follows: each τ_{iq} can be seen as an approximation of the probability that vertex i belongs to cluster q, conditional on the data, that is, τ_{iq} estimates $\mathbb{P}(Z_{iq}=1|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta})$, under the constraint $\sum_q \tau_{iq} = 1$. In the ideal case where $\mathbb{P}(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta})$ can be factorized as $\prod_i \mathbb{P}(\boldsymbol{Z}_i|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta})$ and the parameters τ_{iq} are chosen as $\tau_{iq} = \mathbb{P}(Z_{iq}=1|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta})$, the Küllback–Leibler divergence is null and the bound $\mathcal{J}$ reaches the log-likelihood.

The lower bound $\mathcal{J}$ to be maximized in order to estimate $\boldsymbol{\tau}$ can be expressed as

$$\mathcal{J}_\tau = \mathbb{E}_{R(Z)}\{\mathcal{J}(\boldsymbol{\Theta})\} = \mathbb{E}_{R(Z)}\{\log(P(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z}))|\boldsymbol{X},\boldsymbol{Y};\boldsymbol{\Theta}\} - \sum_{\boldsymbol{Z}} R(\boldsymbol{Z})\log(R(\boldsymbol{Z})).$$

The optimal approximate distribution R is then derived by direct maximization of $\mathcal{J}_\tau$. Let all the parameters $\hat{\pi}_{ql}, \hat{\alpha}_q, \hat{\boldsymbol{\mu}}_q$ and $\hat{\sigma}$ be known. The following fixed-point relationship holds for the optimal variational parameters $\hat{\boldsymbol{\tau}} = \arg\max_\tau \mathcal{J}_\tau$.

$$\hat{\tau}_{iq}^{(m+1)} \propto \hat{\alpha}_q \prod_{j\neq i}\prod_l \left[\hat{\pi}_{ql}^{x_{ij}}(1-\hat{\pi}_{ql})^{1-x_{ij}}\right]^{\tau_{jl}^{(m)}} \times \prod_{k=1}^{p}\left[\exp\left(\frac{1}{2\hat{\sigma}^2}\left(-(Y_i^{(k)}-\hat{\boldsymbol{\mu}}_q^{(k)})\right)^2\right)\right]. \quad (8)$$

Once again, the maximization of $\mathcal{J}_\tau$ provides the optimal values of the parameters. The optimal parameters $\alpha_q, \pi_{ql}, \boldsymbol{\mu}_q$ and σ, i.e. the parameters maximizing $\mathcal{J}_\tau$ satisfy the following relations:

$$\hat{\alpha}_q = \frac{1}{n}\sum_{i=1}^{n}\tau_{iq}, \quad \hat{\pi}_{ql} = \frac{\sum_{i\neq j}\tau_{iq}\tau_{jl}x_{ij}}{\sum_{i\neq j}\tau_{iq}\tau_{jl}}, \quad \hat{\mu}_q = \frac{\sum_i \tau_{iq}\boldsymbol{Y}_i}{\sum_i \tau_{iq}} \text{ and } \hat{\sigma} = \frac{\sum_i\sum_q \tau_{iq}(\boldsymbol{Y}_i - \hat{\zeta}_q)^T(\boldsymbol{Y}_i - \hat{\zeta}_q)}{\sum_i\sum_q \tau_{iq}}. \quad (9)$$

For completeness, we summarize the variational EM algorithm for CohsMix in the Algorithm 1.

Algorithm 1. Variational EM CohsMix algorithm

Data: Matrices of connectivities $\boldsymbol{X}$ and similarities $\boldsymbol{Y}$
/* Initialization of the parameters */
$\boldsymbol{\Theta}^{(0)} = (\alpha_1^{(0)},\ldots,\alpha_Q^{(0)},\pi_{11}^{(0)},\ldots,\pi_{QQ}^{(0)},\boldsymbol{\mu}_1^{(0)},\ldots,\mu_p^{(0)},\sigma^{(0)}), m = 0$
while not convergence **do**
  /* Estimation step */
  /* Compute $\boldsymbol{\tau} = \{\tau_{iq}\}_{i\in\mathcal{P},q\in\mathcal{Q}}$ the probabilities that vertex i belong to cluster q finding fix point of $g()$ */
  foreach $i \in \{1,\ldots,N\}$ **do**
    foreach $q \in \{1,\ldots,Q\}$ **do**
      $\tau_{iq}^{(m+1)} = g(\boldsymbol{\tau}^{(m)})$ (see Eq. (8))
    /* normalize posterior probabilities */
    $scale = \sum_{q=1}^{Q}\tau_{iq}$
    $\tau_{iq} = \tau_{iq}\frac{1}{scale}, \forall q \in \{1,\ldots,Q\}$
  /* Maximization step */
  /* re-estimate the distribution parameters to maximize the likelihood of the data */
  Update parameters according to Eq. (9):
  foreach $q \in \{1,\ldots,Q\}$ **do**
    $\alpha_q^{(m+1)} = \text{Argmax}_{\alpha_q}\mathcal{J}_\tau(\boldsymbol{\Theta})$
    foreach $l \in \{1,\ldots,Q\}$ **do**
      $\pi_{ql}^{(m+1)} = \text{Argmax}_{\pi_{ql}}\mathcal{J}_\tau(\boldsymbol{\Theta})$
      $\mu_q^{(m+1)} = \text{Argmax}_{\mu_q}\mathcal{J}_\tau(\boldsymbol{\Theta})$
  m = m + 1
Result: Estimated parameters $\boldsymbol{\Theta}$ and posterior probabilities τ_{iq}

2.4. Model selection: ICL algorithm

As the number of clusters is an unknown parameter of our statistical model, it is possible to use the Integrated Classification Likelihood (ICL) to choose the optimal number of classes (Biernacki et al., 2000). The ICL criterion is essentially derived from the ordinary BIC considering the complete log-likelihood instead of the log-likelihood. This optimal number is obtained by running our algorithm concurrently for models from 2 to Q classes and selecting the solution which maximizes the ICL criterion. In our situation where additional covariates are considered, the ICL criterion can be written as:

$$\begin{aligned} ICL(Q) = \max_{\boldsymbol{\Theta}} \log \mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}; \boldsymbol{\Theta}, Q) \\ - \underbrace{\frac{1}{2} \times Q(Q-1) \log\left(\frac{n(n-1)}{2}\right)}_{\text{related to } \pi_{ql}} - \underbrace{\frac{Q-1}{2} \log(n)}_{\text{related to } \alpha_q} \\ - \underbrace{p(p-1) \log\left(\frac{n(n-1)}{2}\right) + p \times Q \log\left(\frac{n(n-1)}{2}\right)}_{\text{related to } \mu_q \text{ and } \sigma} \end{aligned}$$

This expression of the ICL criterion is based on the method described in (Daudin et al., 2008).

3. Experiments

In this section we report experiments to assess the performances and limitations of the proposed model in a clustering context. We consider both synthetic data generated with respect to the assumed random graph model and real data from the web. Synthetic graphs are useful for evaluating the quality of parameter estimation. In parallel, we also compare classification results with alternative clustering methods using a ground truth. The real datasets consist of hypertext documents retrieved from a web search. An R package we have called `CohsMix` is available upon request.

3.1. Comparison of algorithms

3.1.1. Simulation setup

In these experiments we consider simple affiliation models with two parameters defining the probabilities of connection between nodes of the same class and between nodes of different classes, respectively, $\pi_{qq} = \lambda$ and $\pi_{ql} = \epsilon$, and equal mixture proportions $\alpha_1 = \ldots = \alpha_Q = \frac{1}{Q}$. Models have $n = 150$ nodes.

Graph models were generated in order to evaluate the performances of the algorithm as the difficulty of the problem varies. The clustering problem increases in difficulty with the number of classes Q, the number of features *nbCov*, the Euclidean distance $d(\lambda, \epsilon)$ between intra and extra connectivity parameters, and the distance $d(\mu_q, \mu_l)$ between the feature mean vectors of classes. We decided to focus on these parameters to produce data with different levels of structure and used 43 different graph models whose description are summarized in Table 1. Each model is simulated 20 times.

We use the adjusted Rand Index (Hubert and Arabie, 1985) to evaluate the agreement between the estimated and the actual partition. The Rand Index is based on a ratio between the number of node pairs belonging to the same and to different classes when considering both partitions. It lies between 0 and 1, two identical partitions having an adjusted Rand Index equal to 1.

To avoid initialization issues, the algorithm is started with multiple initialization points and the best result is selected based on its likelihood. Thus, for each simulated graph, the algorithm is run 10 times and the number of clusters is chosen using the Integrated Classification Likelihood criterion, as proposed in the previous section.

Table 1
Parameters of the four different settings used to generate the 43 affiliation models considered in the experiments.

Experiments	Q	*nbCov*	$d(\lambda, \epsilon)$	$d(\mu_q^{(j)}, \mu_l^{(j)})$
a	$\{2, \ldots, 12\}$	3	0.4	4
b	5	$\{2, \ldots, 15\}$	0.2	4
c	3	3	$\{0, \ldots, 0.5\}$	4
d	3	3	0	$\{4, \ldots, 8.5\}$

3.1.2. Alternative clustering methods

Additionally to the `CohsMix` algorithm study, we compared it with two "rivals": a multiple view learning algorithm (Ruping and Scheffer, 2005; Zhang et al., 2006), and a Hidden Markov Random Fields (Ambroise et al., 1997):

- *Spectral Multiple View Learning (SMVL)*: there exits a strong similarity between multiple view and graph models with covariates. Multiple view learning algorithms consider instances which have multiple representations and use these views simultaneously to obtain a consensus partition. This is achieved via spectral clustering on a linear combination of a standard kernel corresponding to the graph structure and a kernel corresponding to vertex proximity.
- *Hidden Markov Random Fields (HMRF)*: hidden Markov Random Fields are commonly used to handle spatial data and are widely used in image analysis. We use a classical Potts model on the latent structure, which encourages spatial smoothing of the cluster. This kind of approach uses the graph structure to smooth the partition of the vertex over the graph, whereas the approach proposed in this paper uses the graph structure directly to estimate the vertex partition.

3.1.3. Simulation results

We focus our attention on the Rand Index for each algorithm, inasmuch as a well-estimated partition yields good estimates.

As expected, the performance of the three algorithms deteriorates as the number of groups increases (Fig. 2a).

A first interesting result is that where there is a modular structure (Fig. 2a–c) in the network and weakly-informative features, `CohsMix` algorithms always perform better than the SMLV and HMRF algorithms.

It is noticeable that the performance of `CohsMix` improves as the number of features increases, and/or as the distance between mean vectors increases (Fig. 2b and d). The HMRF algorithm with a Potts model will generally use the neighborhood structure for smoothing the partition. A vertex whose neighbors are all in the same given class has a high probability of also being assigned to this class, but HRMF does not take advantage of the graph structure as fully as `CohsMix`. Our model is thus particularly suited to datasets with an existing graph structure.

When there is no graph structure at all and few informative features (Fig. 2d) the `CohsMix` is no match for HMRF or SMLV. The `CohsMix` algorithm is more sensitive to the total absence of graph structure than its competitors.

In all other setups, however, the quality of partition estimation remains good with different kind of models, the `CohsMix` algorithm appears very attractive and suitable for structured graphs with vertex features. We shall see in the next section that this algorithm also performs well on real web datasets.

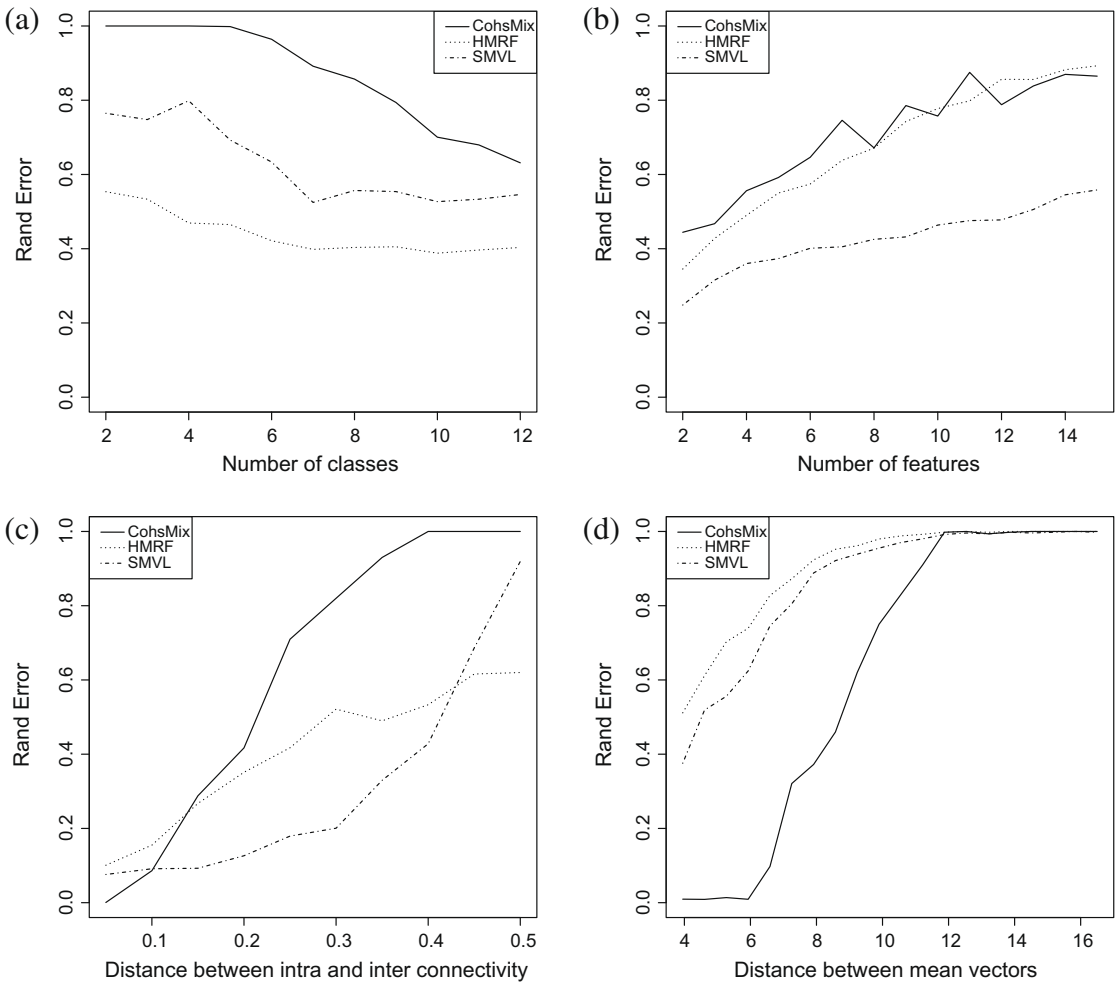

Fig. 2. Comparison of HMRF, spectral MLV and `CohsMix`. (a) varying Q the number of classes. (b) Varying the number of features. (c) Varying the distance between intra and inter connectivity parameters. (d) Varying the distance between the mean vector of the classes.

3.2. Real data

Exhaustivity is an essential feature for information retrieval systems like Web search engines. However, ambiguous queries tend to produce a huge diversity of responses that can be a real impediment to understanding. A common way of circumventing this problem is to organize search results into groups (clusters), one for each meaning of the query. This has been a focal point within the information retrieval community (Hearst and Pedersen, 1996; Zamir and Etzioni, 1998) since the early days of the web. More recently, academic (Zeng et al., 2004) and industrial (Bertin and Bourdoncle, 2002) (`exalead.com` or `clusty.com`) offerings have made the clustering of search results a common feature for a WWW user.

The main drawback of many web page clustering methods is that they only take account of the topical similarity between documents in the ranked list, without considering the topology formed by hyperlinks. In competitive or controversial queries (such as "abortion", or "Scientology") such methods fail to reveal community information visible in the link topology: by affinity, authors tend to link to pages with similar topics or points of view, which creates a thematic locality (Davison, 2000). In addition, ambiguous queries like "orange" or "jaguar" might also harness link topology so as to produce a more accurate separation of results. Combining topological and topical clustering methods is a proven strategy in building an effective system. One of the most relevant contributions to the literature is He et al. (2002), which describes a web page clustering system taking into account the hyperlink structure of the Web, considering two web pages to be similar if they are in parent/child or sibling relations in the web graph. A more general multi-agent framework based on the path between each pair of results was proposed by Bekkerman et al. (2006), but these methods, not model-based, use various heuristics and fine tunings.

3.2.1. Datasets setup

We use the `exalead.com` search engine in our real data experiments. For each query, we retrieve the first 150 search results in order to build our graph and feature structures. The web is a very sparse graph and thematic subgraphs may amplify this property, creating unconnected components which reduce the feasibility of using classical graph clustering algorithms directly on the observed adjacency matrix. In order to increase the graph density, that is to say the probability of there being a link between two nodes, we propose using the site graph of `exalead.com`, based on the concepts of Raghavan and Garcia-Molina (2003). In this graph, nodes represent websites (a website contains a set of pages) and edges represent hyperlinks between websites. Multiple links between two different websites are collapsed into a single link. Intra-domain links are taken into account if hostnames/websites are not similar. The site graph is previously computed. It will be remarked that this methodology is similar to the Exalead application *Constellations*: `constellations.labs.exalead.com`.

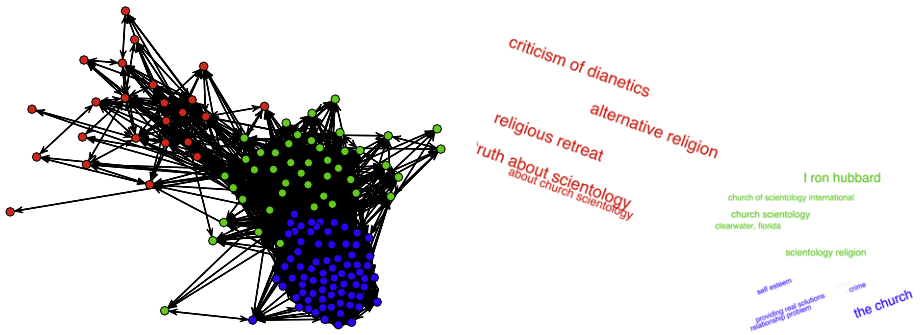

Fig. 3. Representation of the results of a clustering of the web pages returned by the controversial query "Scientology" using `CohsMix`. The graph structure is shown on the left, and on the right are the main features. Colors indicate the `CohsMix` classification.

Fig. 4. Representation of the results of a clustering of the web pages returned by the ambiguous query "jaguar".

Text features are extracted from the content of the web page returned by the search engine. The features are built using various text-processing techniques including normalization, tokenization, entity-detection, noun-phrase detection and related term detection. Rare features which do not appear more than twice are removed. The resulting feature vectors are approximately of dimension $p = 100$ and summarize the entire text returned pages.

3.2.2. Algorithm results

We have selected one ambiguous query ("jaguar") and one controversial query ("Scientology") to illustrate the behavior of our algorithm with real datasets. In Fig. 3, corresponding to the query "Scientology", we observe a well structured graph which fits our estimated latent partition with an optimal number of classes $Q = 3$. Basically, this partition yields the pro- and anti-Scientology clusters, and identifies a gateway cluster (composed for example by http://en.wikipedia.org/wiki/Scientology) bridging the pro- and anti-clusters. We then focus on the most representative text features of each class q. To this end we select the best occurrence-of-term features in the different μ_q. Once again (see Fig. 3), we notice pro-terms ("self esteem", or "providing real solutions") and anti-terms ("criticism of dianetics", or "truth about Scientology"). The interface class is composed of common terms describing the Church of Scientology. Thus, in a web context, the `CohsMix` algorithm is enable to name the different partitions obtained, which is very useful for communicating rapidly a global overview of the hidden structure.

The results of the processing of the ambiguous query "jaguar" is represented in Fig. 4. `CohsMix` clearly identifies three contexts: computer, animal and car model related web pages.

The above results illustrate that our algorithm `CohsMix` seems well adapted to detecting ambiguous or controversial queries by WWW search engine users.

4. Conclusion

This paper has proposed an algorithm for clustering datasets that can be modeled with a graph structure embedding vertex features. Characterizing each vertex both by a vector of features and by its connectivity, the `CohsMix` algorithm, based on a variational approach of EM, uses both these elements to cluster the data and estimate the model parameters. Simulation and comparison results show our algorithm to be attractive and competitive for various kind of models. When `CohsMix` is used to cluster web search results based on hypertextuality and content, the relevance of this approach is amply demonstrated. We find that our algorithm successfully harnesses whatever information is found both in the connectivity pattern and in the features. In the short term we plan to investigate how to focus one type of information, graph or features, when it becomes predominant.

References

Ambroise, C. et al., 1997. Clustering of spatial data by the EM algorithm. geoENV I-Geostatist. Environ. Appl. 9, 493–504.
Bekkerman, R. et al., 2006. Web Page Clustering using Heuristic Search in the Web Graph.
Bertin, P. and Bourdoncle, F. 2002. Searching tool and process for unified search using categories and keywords. EP Patent 1,182,581.
Biernacki, C. et al., 2000. Assessing a mixture model for clustering with the integrated completed likelihood. IEEE PAMI 22 (7), 719–725.
Cook, D., Holder, L. (Eds.), 2007. Mining Graph Data. Wiley-Interscience.
Daudin, J. et al., 2008. A mixture model for random graph. Statist. Comput. 18 (2), 1–36.
Davison, B.D., 2000. Topical locality in the web. In: SIGIR '00: Proc. 23rd Annual Internat. ACM SIGIR Conf. Research and Development in Information Retrieval. ACM, New York, NY, USA, pp. 272–279.
Dempster, A. et al., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. Royal Statist. Soc. 39 (1), 1–38.
Frank, O., Harary, F., 1982. Cluster inference by using transitivity indices in empirical graphs. J. Am. Statist. Assoc. 77 (380), 835–840.
Govaert, G., 1977. Algorithme de classification d'un tableau de contingence. In: First Internat. Symposium on Data Analysis and Informatics. INRIA, Versailles, pp. 487–500.
He, X. et al., 2002. Web document clustering using hyperlink structures. Computat. Statist. Data Anal. 41 (1), 19–45.
Hearst, M., Pedersen, J., 1996. Reexamining the cluster hypothesis: scatter/gather on retrieval results. In: Proc. 19th Annual Internat. ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 76–84.
Hoff, P., 2003. Random effects models for network data. Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, pp. 303–312.
Hubert, L., Arabie, P., 1985. Comparing partitions. J. Classif. 2, 193–218.
Jordan, M. et al., 1999. An introduction to variational methods for graphical models. Machine Learn. 37 (2), 183–233.
Lorrain, F., White, H., 1971. Structural equivalence of individuals in social networks. J. Math. Sociol. 1, 49–80.
Mariadassou, M., Robin, S. 2007. Uncovering latent structure in valued graphs: a variational approach. Tech. Rep. 10, SSB.
Newman, M., Leicht, E., 2007. Mixture models and exploratory analysis in networks. PNAS 104 (23), 9564–9569.
Raghavan, S., Garcia-Molina, H. 2003. Representing web graphs. In: Proc. 19th Internat. Conf. on Data Engineering, pp. 405–416.

Ruping, S. and Scheffer, T. 2005. Learning with multiple views. In: Proc. ICML Workshop on Learning with Multiple Views.
Schenker, A. et al., 2005. Graph-theoretic Techniques for Web Content Mining. World Scientific.
Snijders, T.A.B., Nowicki, K., 1997. Estimation and prediction for stochastic block-structures for graphs with latent block structure. J. Classif. 14, 75–100.
Zamir, O., Etzioni, O., 1998. Web document clustering: a feasibility demonstration. In: Proc. 21st Annual Internat. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 46–54.
Zanghi, H. et al., 2008. Strategies for Online Inference of Network Mixture. Tech. Rep., Statistique et Genome, INRA, SSB.
Zeng, H. et al. 2004. Learning to cluster web search results. In: Proc. 27th Annual Internat. Conf. on Research and Development in Information Retrieval, pp. 210–217.
Zhang, T. et al., 2006. Linear prediction models with graph regularization for web-page categorization. In: Proc. 12th ACM SIGKDD Internat. Conf. Knowledge Discovery and Data Mining. ACM, New York, NY, USA, pp. 821–826.