Fast online Graph Clustering via Erdös-Rényi Mixture

by

Hugo Zanghi, Christophe Ambroise and Vincent Miele

Research Report No. 8
April 2007

Statistics for Systems Biology Group
Jouy-en-Josas/Paris/Evry, France
`http://genome.jouy.inra.fr/ssb/`

Fast online Graph Clustering via Erdös-Rényi Mixture

Hugo Zanghi, Christophe Ambroise and Vincent Miele

April 25, 2007

Abstract

In the context of graph clustering, we consider the problem of estimating simultaneously both the partition of the graph nodes and the parameters of an underlying mixture of affiliation networks. In numerous applications the rapid increase of data size with time makes classical clustering algorithms too slow because of the high computational cost. In such situations online clustering algorithms are an efficient alternative to classical batch algorithms. We present an original online algorithm for graph clustering based on a Erdös Régnyi Graph Mixture. The relevance of the algorithm is illustrated, using both simulated and real data sets. The real data is a network extracted from the french political blogosphere and presents an interesting community organization.

1 Introduction

In many scientific fields, systems can be modeled using networks to represent data relationships. World-wide-web, gene interactions, social networks, authors citations, are examples of fields where graph representation makes sense interpreting relations between nodes. Considering the web, nodes represent web sites or web pages, and each edge represents an hyperlink relating two nodes.

These so-called real networks share some properties such as small-world phenomenon in the sense that most nodes are close to each other (6 degree of separation in social networks), scale-free distribution of the degrees (distribution of degrees does not depend on the graph size), degree distribution which obeys a power law (presence of hubs), giant components (connected subgraph that contains a majority of the entire graph's nodes), high clustering coefficient (important aggregative trend of a graph) and finally preferential attachment (the nodes connect with higher probability compared to those nodes that already have a large number of edges).Random graphs are a possible model for networks where nodes are given and vertex considered as random variables. The simplest and most studied random graph model is the Erdös-Rényi graph, where each pair of nodes is connected with probability p. But Erdös-Rényi graphs do not exhibit most of the properties of real networks, particulary the degree distribution and the high clustering coefficient. Thus alternative models have been developed. For example, mechanistic models [Albert and Barabasi, 2002] have been proposed to model network growth. They are mainly based on informative summary statistics, such as the degree distribution, but do not allow for any statistical inference.

When the number of nodes and egdes is important, it is a difficult task to have a global synthetic view of the network structure. Finding groups or communities of nodes which have a higher within-group density of edges than between-groups can have a significant importance for interpreting the network. For instance, groups within the worldwide web might correspond to sets of web pages on related topics [Flake et al., 2002]; groups within social networks might correspond to social units or communities [Wasserman et al., 1994a] The mere fact of finding a network that contains tightly-knit groups (so-called community structure) at all can convey useful information: if for instance a metabolic network were to be divided into such groups, it could provide evidence for a modular view of the network's dynamics, with different groups of nodes performing different functions with some degree of independence [Guimera and Amaral, 2005].

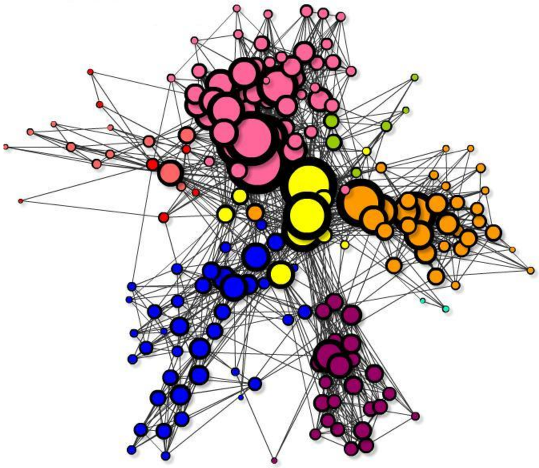

Figure 1: Network of the blogopole `www.blogopole.fr`

This detection of community structure is a well studied problem, which is closely related to classical clustering. Thus proposed approaches for communities detection take their inspiration from classical clustering algorithms, proposing criterion adapted to this particular problem.

The Erdös-Renyi Mixture Model for Graph (ERMG) has been proposed by [Daudin et al., 2006] with an associated EM [Dempster et al., 1977] estimation algorithm and is not to be confused with Exponential Random Graph Models for Network Data (ERGM), which consider distributions ensuing from the exponential familiy to model the edge distribtion [Snijders et al., ress].

There exists a strong connection between ERMG and block clustering [Snijders and Nowicki, 1997]. Block clustering searches for homogeneous blocks in a data matrix by simultaneous clustering of rows and columns. When partitions of rows and columns are assumed to be identical Bernoulli Block Mixture model and associated algorithm [Govaert and Nadif, 2005] is equivalent to ERMG.

In this context of Block-Clustering for classical binary data, an almost equivalent ERMG algorithm without any underlying statistical model can be dated back to the early work of Govaert [Govaert, 1977, Govaert, 1983].

There is also a long tradition of developing statistical graph models for social networks [Wasserman et al., 1994b, Handcock et al., 2006], but unfortunately these models often raise computational issues when dealing with large networks.

In recent years several algorithms have been proposed. One of the most effective and competitive, is a hierarchical agglomeration algorithm proposed by Newman [Newman, 2004], based on the greedy optimization of the quantity known as *modularity*.It runs in time $O((m+n)n)$, or $O(n^2)$ on a sparse graph, where m and n are respectively the number of vertex and nodes. In [Clauset et al., 2004], exploiting more sophisticated data structures and some shortcuts in the optimization problem makes run the communities detection problem in time $O(md \;\; log(n))$ where d is the depth of the "‘dendrogram"’ describing the network's community structure.

In numerous applications the rapid increase of data size implies a regular run of the classical batch algorithm in order to reduce the time latency between the appearance of a new data and its treatment (its classification). In such situations online clustering algorithms are an efficient alternative. We present an original online algorithm for graph clustering based on a Erdös-Rényi Graph Mixture.

This first section of the paper introduces the model, its online estimation and the practical strategies for the initalization and the choice of the number of groups. In the second section extensive simulation illustrates the efficiency of this algorithm and a real data set dealing with the french political blogosphere is studied.

2 Online ERMG algorithm

2.1 Affilation model

Following Frank and Harary [Frank and Harary, 1982], the ERMG model [Daudin et al., 2006] assumes that given a set of n nodes partionned into Q classes, edges are random variables conditionaly independent given the class of the nodes

$$X_{ij}|\{i \in q, \; j \in l\} \sim \mathcal{B}(\pi_{ql}).$$

The class vector Z_i is a random vector following a multinomial distribution:,

$$Z_i \sim \mathcal{M}(1, \alpha_1, ..., \alpha_q).$$

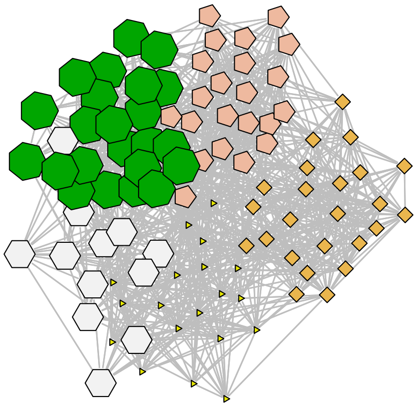

Figure 2: Simulation of a 100 nodes graph with 5 classes according to an affiliation model

Thus the model parameters are the proportion of the classes and the probability of connectivity between classes. In this paper, we consider a simple affiliation model where two types of egdes exist: egdes between nodes of the same class and egdes between nodes of different classes. Each type of edge has a given probability: $\pi_{qq} = \lambda$ and $\pi_{ql} = \epsilon$. Self-loops are not taken into account (see for example Figure 2).

This model allows to explicitly compute the distribution of the degree K_i of node i, which is approximately a mixture of Poisson distribution $\mathcal{P}((n-1)[(1-\alpha_q)\epsilon + \alpha_q\lambda])$.

The clustering coefficient is defined as the probability of having an edge between two nodes, which are both related to a third node. The ERMG model allows again an explicit computation of this coefficient.

Daudin et. al [Daudin et al., 2006] tackle the problem of ERMG parameter estimation using an Expectation-Maximisation algorithm [Dempster et al., 1977], with an approximate E-step. The problem induced by the dependence between the nodes of the graph, makes it difficult to compute the expectation of the missing data conditionally to the available knowledge of the network structure, $P(Z_i|\mathbf{x})$. Let us also note that the likelihood of the model is impossible to compute since it implies a sum over all possible partitions of the graph's nodes. In this paper, we propose a fast alternative algorithm, well adapted for clustering, based on stochastic approximation and the maximization of the complete data log-likelihood.

	neighbours	not neighbours
same class	$n_\lambda = \sum_q n_{qq}$	$n_{1-\lambda} = \sum_q \frac{n_q(n_q-1)}{2} - n_\lambda$
different class	$n_\epsilon = \sum_{q \neq l} n_{ql}$	$n_{1-\epsilon} = \frac{n(n-1)}{2} - n_\lambda - n_{1-\lambda} - n_\epsilon$

Table 1: Statistics of the affiliation model computed from $n_{ql} = \sum_{i>j} x_{ij} z_{iq} z_{jl}$, the number of egdes having nodes in class q and l, and $n_q = \sum_i z_{iq}$, the number of nodes of class q.

2.2 Classification log-likelihood

The Classification EM (CEM) algorithm is an iterative clustering algorithm which yields simultaneously the parameters and the classification. In this paper the considered classification log-likelihood is defined by

$$\begin{aligned} L_C(\mathbf{x}, \mathbf{z}, \Phi) &= \log P_\Phi(\mathbf{z}) + \log P_\Phi(\mathbf{x}|\mathbf{z}) \\ &= \sum_{i,q} z_{iq} \log \alpha_q + \sum_{q,l,i>j} z_{iq} z_{jl} \log\left(\pi_{ql}^{x_{ij}} (1-\pi_{ql})^{1-x_{ij}}\right), \end{aligned}$$

where $\Phi = \{\alpha_1, ..., \alpha_Q, (\pi_{ql})\}$ represents the parameter vector. Maximizing $L_C(\mathbf{x}, \mathbf{z}, \Phi)$ according to $(\mathbf{z}, \Phi)$ is equivalent to maximizing the criterion

$$L_C(\mathbf{x}, \mathbf{\Phi}) = \max_{\mathbf{z}} [L_C(\mathbf{x}, \mathbf{z}; \mathbf{\Phi})].$$

When considering the simple affiliation model, the term $z_{iq} z_{jl} \log\left(\pi_{ql}^{x_{ij}} (1-\pi_{ql})^{1-x_{ij}}\right)$ takes four different values according to the class of i and j, and their neighbourhood relationship. The classication log-likelihood can then be expressed as:

$$\begin{aligned} L_C(\mathbf{x}, \Phi) &= \sum_q n_q \log \alpha_q + n_\lambda \log \lambda + n_{1-\lambda} \log(1-\lambda) \\ &\quad + n_\epsilon \log \epsilon + n_{1-\epsilon} \log(1-\epsilon), \end{aligned}$$

where n_λ, $n_{1-\lambda}$, n_ϵ and $n_{1-\epsilon}$ are the number of node pairs defined by class and neighbourhood relationship (see Table 1). Note that these four statistics can be computed from two different type of basic statistics:

- $n_{ql} = \sum_{i>j} x_{ij} z_{iq} z_{jl}$, the number of egdes having nodes in class q and l,
- $n_q = \sum_i z_{iq}$, the number of nodes of class q.

The parameters $\hat{\Phi}$ maximizing this criterion for a given partition are as follows:

- $\hat{\alpha} = \frac{n_q}{n}$,
- $\hat{\lambda} = \frac{n_\lambda}{n_\lambda + n_{1-\lambda}}$,
- $\hat{\epsilon} = \frac{n_\epsilon}{n_\epsilon + n_{1-\epsilon}}$.

2.3 Online Estimation

In numerous applications the rapid increase of data size with time makes classical clustering algorithms too slow. In such situations online clustering algorithms are an efficient alternative to classical batch algorithms. Online parameter estimation using mixture models has already been studied by many authors (Titterington 1984 [Titterington, 1984], Wang and Zhao 2002 [Y., 2002]). More recently Liu et al. [Liu et al., 2006] have considered, for the modeling of internet traffic, a recursive EM algorithm for the estimation of Poisson mixture models. Typical clustering algorithms include the online k-means (MacQueen 1967 [MacQueen, 1967]) algorithm.

This section describes an original incremental Classification version of the EM algorithm. Incremental algorithms recursively update parameters, using current parameters and new observations.

Let us note $\mathbf{x}^{m-1}$ the adjacency matrix of a graph with $m-1$ nodes and $\mathbf{z}^{m-1}(\Phi)$, the classification matrix verifying:

$$\mathbf{z}^{m-1}(\Phi) = \underset{\mathbf{z}}{\text{Argmax}}\, L_C(\mathbf{x}^{m-1}, \mathbf{z}^{m-1}, \Phi).$$

Let Φ^{m-1} be the parameter vector maximizing $L_C^{m-1}(\mathbf{x}^{m-1}, \mathbf{z}^{m-1}(\Phi^{m-2}), \Phi)$ the complete log-likelihood expressed in function of $m-1$ nodes. When a new node $\boldsymbol{x}_m$ becomes available, the new complete log-likelihood of Φ is expressed as the sum of the previous complete log-likelihood and a new term function of the egdes between the new node and the existing network:

$$L_C^m(\mathbf{x}^m, \mathbf{z}^m(\Phi^{m-1}), \Phi) = L_C^{m-1}(\mathbf{x}^{m-1}, \mathbf{z}^{m-1}, \Phi) + \max_q L_C(\boldsymbol{x}_m, q, \Phi). \tag{1}$$

The principle of the recursive algorithm consists in computing the parameter $\Phi^{(m)}$ maximizing $L_C^m(\mathbf{x}^m, \mathbf{z}^m(\Phi^{m-1}), \Phi)$ and exploiting the fact that the new estimates are function of the old ones.

The recursive algorithm is described by the two following steps each time a new (m)th node (and corresponding vertices) is considered:

- **Step 1** assign each new node $\boldsymbol{x}_m$ to the class q^* which maximizes

$$L_C(\boldsymbol{x}_m, q; \boldsymbol{\Phi}) = \log \alpha_q + \sum_l \sum_{j \neq m} z_{jl} \log\left(\pi_{ql}^{x_{mj}} (1-\pi_{ql})^{1-x_{mj}}\right).$$

  Thus set z_{mq} equal to 1 if $q = q^*$, 0 otherwise.

- **Step 2** update the parameters for all classes:

$$\begin{aligned} n_q^{(m)} &= n_q^{(m-1)} + z_{mq}, & (2)\\ n_{ql}^{(m)} &= n_{ql}^{(m-1)} + \sum_{j \neq m} z_{mq} z_{jl} x_{mj}, & (3)\\ \alpha_q^{(m)} &= \frac{n_q^{(m)}}{m}, & (4) \end{aligned}$$

$$\pi_{ql}^{(m)} = \frac{n_{ql}^{(m)}}{n_q^{(m)} n_l^{(m)}}, \quad (5)$$

$$\pi_{qq}^{(m)} = \frac{2n_{qq}^{(m)}}{n_q^{(m)}(n_q^{(m)} - 1)}. \quad (6)$$

This algorithm increases the complete-log likelihood at each step, and requires at most as many iteration as the number of nodes. Still the parameters can be further improved and the complete log-likelihood further increased by revisiting each node a few times. When m the number of iteration is greater than n the size of the network, it is possible to apply the above described recursive principle, and the algorithm can be continued as follows

- **Step 1** find a node $\boldsymbol{x}_i$, whose class change improves the classification log-likelihood.
- **Step 2** update the parameters for all classes:

$$n_q^{(m)} = n_q^{(m-1)} - z_{iq}^{m-1} + z_{iq}^{m}, \quad (7)$$

$$n_{ql}^{(m)} = n_{ql}^{(m-1)} + (z_{iq}^{m} - z_{iq}^{m-1}) \sum_{j \neq i} z_{jl} x_{ij}, \quad (8)$$

$$\alpha_q^{(m)} = \frac{n_q^{(m)}}{n}, \quad (9)$$

$$\pi_{ql}^{(m)} = \frac{n_{ql}^{(m)}}{n_q^{(m)} n_l^{(m)}}, \quad (10)$$

$$\pi_{qq}^{(m)} = \frac{2n_{qq}^{(m)}}{n_q^{(m)}(n_q^{(m)} - 1)}. \quad (11)$$

When step 1 is not possible, the algorithm can be stopped.

The adaptation to the affiliation model is straightforward and requires only the computation of the four statistics of the affiliation model (see Table 1) from the $n_{ql}^{(n)}$.

2.4 Initialization and online supervised classification

If one is mainly interested in finding clusters of nodes which have strong interconnection and weak between-connection, the above described algorithm can be simplified and accelerated by working with fixed parameter values. A possible interesting choice for clustering consists in choosing a high value for λ probabilities on within-connection and a small one for ϵ the probability of between-group connection (i.e. $\lambda = 0.8$ and $\epsilon = 0.05$). This choice implicitly assumes the existence of clique-like cluster. Concerning the proportions, without any additonnal knowledge, it seems reasonable to consider cluster of the same size $\alpha_1 = ... = \alpha_Q = \frac{1}{Q}$.

Repeating step 2, with a given value of the parameter vector, produces a partition in a finite number of iterations. Starting from a random partition $\mathbf{z}^0$, each iteration considers a randomly chosen node and assigns this node to the cluster which results in a greater increase of the classification log-likelihood. It is obvious this relaxation procedure increases the classification log-likelihood at each iteration and converges toward a local maximum since the criterion is upper-bounded.

This minimal clustering algorithm can be used as an effective initialization strategy for the previously described online ERMG algorithm.

2.5 Choosing the number of clusters

As the algorithm relies on a statistical model, it is possible to use the Integrated Classification Likelihood (ICL) to choose the optimal number of classes [Biernacki et al., 2000]. This choice is done by running our online algorithm concurrently for models from 2 to Q classes and selecting the solution which maximizes the ICL criterion. In our situation, the ICL criterion can be written as :

$$ICL(Q) \quad = \underbrace{-2L_C(\mathbf{x}, \Phi)}_{\text{A}} + \underbrace{(Q-1)log(n) + 2log(\frac{n(n-1)}{2})}_{\text{B}}$$

where A is related to the classification log-likelihood, B to the free number of parameters and n the number of nodes treated. The ICL criterion is essentially the ordinary BIC considering the complete log-likelihood instead of the log-likelihood.

An ANSI `C++` implementation of the CEM algorithm is available in the `ermg` package. Compilation and installation are compliant with the GNU standard procedure.

The package is free and available at http://stat.genopole.cnrs.fr/software/ermg. On-line documentation is also available. `ermg` is licensed under the GNU General Public License (http://www.gnu.org/licences.html).

3 Applications

We carried out experiment to assess how well the proposed online clustering algorithm discovers node clusters. We consider simulation experiment using synthetic data generated according to the assumed random graph model, as well as real data coming from the internet sphere.

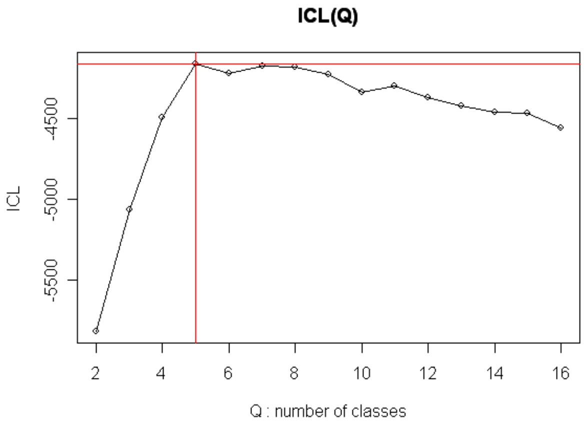

Figure 3: Integrated Classification Likelihood Criterion in function of the number of clusters computed for the simulated graph of Figure 2.

3.1 Synthetic Data

Four affiliation models have been considered (see Table 2). The difference among the four models is related to their modular structure, which varies from no structure (almost the Edrös-Renyi model) to strong modular structure (low inter-module connectivity and strong intra-module connectivity).

Model	λ	ϵ	Q
1	0.8	0.02	3
2	0.5	0.05	8
3	0.6	0.25	5
4	0.55	0.35	5

Table 2: Parameters of the four affililation models of the experiment. The Q modules are mixed in the same proportion. Each model consider $n = 1000$ nodes.

Given the number of nodes n and the class proportions (α_q), the color of each node is simulated via a multinomial distribution $\mathcal{M}(1, \alpha_1, ..., \alpha_Q)$. Conditionally to the node colors, edges between two nodes of the same class are drawn according to a probability λ and edges between nodes of different colors are drawn according a probability ϵ.

Comparing the estimated partition with the true partition is not as straitghtforward as comparing the parameter estimates. In order to evaluate agreement between these two partitions, we use the adjusted Rand index [Hubert and Arabie, 1985] which lies bewteen 0 and 1. The computation of this index is based on a ratio between the number of node pairs belonging to the same and to different classes when considering the true partition and the estimated partition. Two identical partitions have an adjusted Rand index equal to 1.

We have simulated 30 networks for each model and run the online ERMG algorithm for estimating the model parameters. Figure 3.1 shows two boxplots for each experiment: one boxplot for ϵ and one for λ. Notice that for the first model, the highly structured one, the estimation is very close to the true parameters and exhibits no variance. The estimation of the second model shows a small upward bias and small variance. But the third and fourth models are more difficult and the algorithm overestimates λ the probability of within-cluster connection. In summary the less obvious the structure of the network is, the highest bias we observe in the resulting estimation. Let us also note that even for the easiest model (model 1), the algorithm has a slight tendency to produce biased estimates. This is a phenomenon generally observed on Classification version of the EM algorithm. When considering Table 4, we can observe that the poor estimation of λ, the probability of within-cluster connection reveals also a small Rand index. This means that the poor estimation of λ makes it impossible to retrieve the modular structure of the network.

Model	$\bar{\epsilon}$	σ_ϵ	$\bar{\lambda}$	σ_λ
1	0.022	0.000	0.802	0.002
2	0.053	0.001	0.506	0.008
3	0.238	0.004	0.524	0.015
4	0.348	0.002	0.394	0.006

Table 3: Table means and standart deviation of the parameter estimate of the four model computer over 30 different runs

We also compared the results of the online ERMG algorithm with an alternative clustering method. We consider a computationally heavy method which builds a dissimilarity matrix based on the shortest paths beetween all pairs of nodes. Floyd's algorithm is specifically designed to resolve this problem. Note that these shortest paths are computed in $O(n^3)$ runtime, thus it does not allow us to use it with huge and dynamic networks. After building the dissimilarity matrix, we partitionate the data set into Q classes with the PAM (Partitioning Around Medoids) algorithm which operates on the dissimilarity matrix. PAM is a robust version of the Kmeans, which minimizes a sum of dissimilarities instead of a sum of squared Euclidean distance and considers medoid instead of barycenter to represent cluster. We have also tested this algorithm on the previous networks and have computed the Rand index on each of them. When considering Table 4, we observe that the ERMG online algorithm is very close to the PAM algorithm with a very reduced computational cost $O(n^2)$.

3.2 French Political Blogosphere network

We also studied our algorithm on a real data set. The data consits of a single day snapshot of over 1,1000 political blogs automatically extracted the 14 october

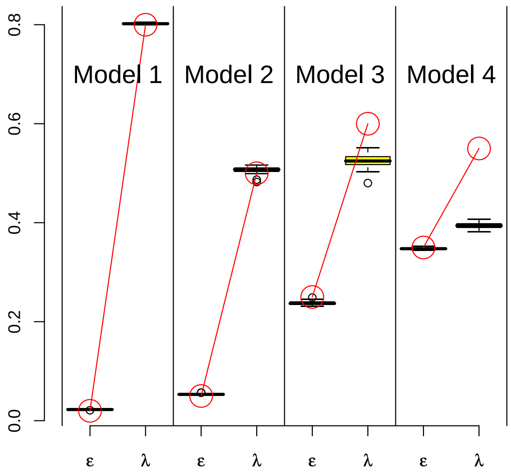

Figure 4: Boxplot of the parameter estimates for 30 estimations of the 4 models. Each model is described by two boxplots, one for the estimations of ϵ and the other for the estimation of λ. The circles show the true value of the parameters.

	PAM		online ERMG	
Model	$\overline{rand_{PAM}}$	$\sigma_{rand_{PAM}}$	$\overline{rand_{ERMG}}$	$\sigma_{rand_{ERMG}}$
1	1.000	0.000	1.000	0.000
2	0.997	0.003	0.978	0.020
3	0.901	0.035	0.883	0.057
4	0.129	0.040	0.071	0.037

Table 4: Table means and standart deviation of the Rand index of the four model computer over 30 different runs for PAM and online Classification ERMG algorithms

2006 and manually classified by the "Observatoire Présidentielle" project. This project is the fruit of a collaboration bewteen RTGI SAS and Exalead and aims at analyzing the presidential campaign on the web.

In this data set, nodes represent hostnames (a hostname contains a set of pages) and edges represent hyperlinks beetwen different hostnames. If several links exist between two different hostnames, we collapse them into a single one. Note that intra domain links can be considered if hostnames are not identical. Finally, in this experimentation we consider that edges are not oriented witch is not realistic but witch does not affect the interpretation of the groups. This network presents an interesting communities organization due to the existence of severals political parties and commentators. We assume that authors of these blogs tend to link, by political affinities, blogs with similar political positions. A sample of 250 nodes of this political blogoshere can be seen on Figure 1. Six known communities compose this newtork : **G**auche ("french democrat"), **D**ivers **C**entre (Moderate party), **D**roite (french republican), **E**cologiste (green), **L**iberal (supporters of economic-liberalism) and finally **A**nalysts. Proportions of blogs in these communities are respectively 0.36, 0.23,0.21, 0.08, 0.08 and 0.04.

In this experimentation we are interested in finding six groups of blogs using the online clustering algorithm for a given couple of ϵ and λ. The number of groups is fixed to six in order to compare the true partition of blogs with the estimated partition running our algorithm. Finding similar partitions validates the assumption that political affinities can be detected using the structure of the political blogosphere. The couple ($\lambda = 0.55$, $\epsilon = 0.04$) gives the maximal agreement bewteen the real and estimated partitions with an acceptable Rand index value (0.34).

Table 3.2 shows a contingency table of the counts of given and estimated blogs classes. Except for the **A** class, we can observe a relative coherence between these two partitions. In fact, the **A** class is a hub class constitued of blogs which links the other classes in order to analyze and comment the french political news. Our algorithm overestimates the number of blogs contained in this class and generates important classification differences. We can also observe that there are mistakes produced by the political proximity of parties. For ex-

		Estimated					
		DC	A	D	E	G	L
True	DC	172	32	4	18	3	25
	A	2	22	2	5	5	5
	D	3	27	165	11	1	26
	E	1	2	0	80	1	0
	G	5	97	12	66	181	45
	L	1	1	3	1	0	82

Table 5: Contingency table comparing true and estimated partitions

ample, the estimated **E** class has many blogs belonging to the real **G** class. Finally, we illustrate on figure 5 the density of links using the adjacency matrix projection of the network after reordering by class estimation. We can observe that there is a higher density of intra-group links than inter-groups which validates our ERMG model. Note that mistakes of the **A** can be observed creating horizontal and vertical bands by linking blogs of other classes.

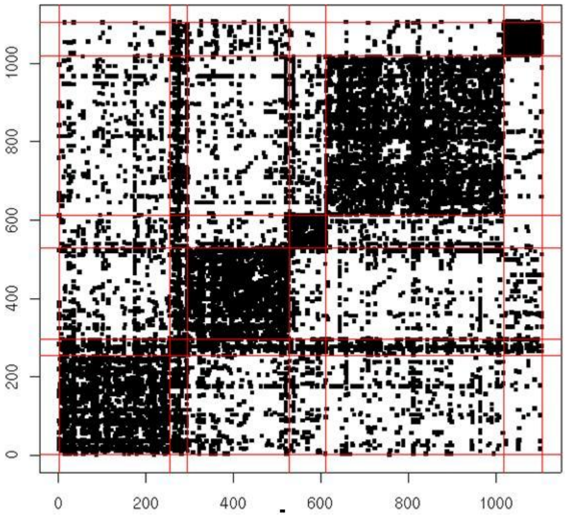

Figure 5: Adjacency matrix of the blogopole network after reordering according the estimated partition

4 Conclusion

The proposed online classification EM algorithm classifies the nodes of a network as they are discovered. The algorithm is based on Erdös Régnyi Graph Mixture, which is a well known model [Frank and Harary, 1982], for which we provide a fast estimation procedure. The algorithm runs in $O(n^2)$ and is thus able to tackle networks with thousand nodes. When starting with a reasonable initialization, this strategy allows to both find communities in a network and a reliable estimation of the model parameters. When the cluster structure is weak the estimates are biased. In the near future, we plan to investigate pure EM online strategies for finding a better estimation for difficult situations.

References

[Albert and Barabasi, 2002] Albert, R. and Barabasi, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47.

[Anderson and Wasserman, 1987] Anderson, C. and Wasserman, S. (1987). Stochastic a posteriori blockmodels: construction and assessment. *Social Networks*, 9:1–36.

[Biernacki et al., 2000] Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE PAMI*, 22(7):719–725.

[Bzioui et al., 1998] Bzioui, M., Nadif, M., and Govaert, G. (1998). Classification croisée et modèle. In *ASU'98, XXXieme journées de Statistiques*, pages 86–88, Rennes.

[Clauset et al., 2004] Clauset, A., Newman, M. E. J., and Moore, C. (2004). Finding community structure in very large networks.

[Daudin et al., 2006] Daudin, J., Picard, F., and Robin, S. (2006). A mixture model for random graph. Technical report, INRIA.

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

[Flake et al., 2002] Flake, G. W., Lawrence, S., Giles, C. L., and Coetzee, F. (2002). Self-organization of the web and identification of communities. *IEEE Computer*, 35(3):66–71.

[Frank and Harary, 1982] Frank, O. and Harary, F. (1982). Cluster inference by using transitivity indices in empirical graphs. *Journal of the American Statistical Association*, 77(380):835–840.

[Govaert, 1983] Govaert (1983). *Classification croisée. Thèse d'état, Université.* PhD thesis, Paris 6, Paris 6, France.

[Govaert, 1977] Govaert, G. (1977). Algorithme de classification d'un tableau de contingence. In *First international symposium on data analysis and informatics*, pages 487–500, Versailles. INRIA.

[Govaert, 1995] Govaert, G. (1995). Simultaneous clustering of rows and columns. *Control Cybern*, 4(24):437–458.

[Govaert and Nadif, 2005] Govaert, G. and Nadif, M. (2005). An em algorithm for the block mixture model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(4):643–647.

[Guimera and Amaral, 2005] Guimera, R. and Amaral, L. A. N. (2005). Functional cartography of complex metabolic networks. *Nature*, 433:895.

[Handcock et al., 2006] Handcock, M., Raftery, A., and Tantrum, J. (2006). Model based clustering for social networks. *JRSS A*.

[Hubert and Arabie, 1985] Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2:193–218.

[Liu et al., 2006] Liu, Z., Almhana, J., Choulakian, V., and McGorman, R. (2006). Online em algorithm for mixture with application to internet traffic modeling. *Computational Statistics & Data Analysis*, 50(4):1052–1071. available at http://ideas.repec.org/a/eee/csdana/v50y2006i4p1052-1071.html.

[MacQueen, 1967] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematics, Statistics and Probability*, 1:281–296.

[Newman, 2004] Newman, M. E. J. (2004). Fast algorithm for detecting community structure in networks. *Physical Review E*, 69:066133.

[Nowicki and Snijders, 2001] Nowicki, K. and Snijders, T. A. B. (2001). Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–??

[Same et al., pear] Same, A., Ambroise, C., and Govaert, G. (to appear). An online classification em algorithm based on the mixture model. *Statistics and Computing*.

[Sato and Ishii, 1999] Sato, M. and Ishii, S. (1999). Fast learning of on-line EM algorithm. Technical report, ATR Human Information Processing Research Laboratories.

[Sato and Ishii, 2000] Sato, M. and Ishii, S. (2000). On-line em algorithm for the normalized gaussian network. *Neural Computation*, 12(2):407–432.

[Snijders and Nowicki, 1997] Snijders, T. A. B. and Nowicki, K. (1997). Estimation and prediction for stochastic block-structures for graphs with latent block structure. *Journal of Classification*, (14):75–100.

[Snijders et al., ress] Snijders, T. A. B., Pattison, P. E., Robins, G. L., and Handcock, M. S. (in Press). New specifications for exponential random graph models. *Sociological Methodology*.

[Titterington, 1984] Titterington, D. M. (1984). Recursive parameter estimation using incomplete data. 46:257–267.

[Wasserman et al., 1994a] Wasserman, S., Faust, K., and Iacobucci, D. (1994a). *Social Network Analysis*. Cambridge University Press, Cambridge (UK).

[Wasserman et al., 1994b] Wasserman, S., Faust, K., and Iacobucci, D. (1994b). *Social Network Analysis : Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press.

[Y., 2002] Y., W. S. Z. (2002). Almost sure convergence of titterington's recursive estimator for finite mixture models. *IEEE International Symposium on Information Theory IST*.