Gene clustering via integrated Markov models combining individual and pairwise features

Matthieu Vignes and Florence Forbes

***Abstract*— Clustering of genes into groups sharing common characteristics is a useful exploratory technique for a number of subsequent computational analysis. A wide range of clustering algorithms have been proposed in particular to analyze gene expression data, but most of them consider genes as independent entities or include relevant information on gene interactions in a sub-optimal way.**

We propose a probabilistic model that has the advantage to account for individual data (*eg.* expression) and pairwise data (*eg.* interaction information coming from biological networks) simultaneously. Our model is based on hidden Markov random field models in which parametric probability distributions account for the distribution of individual data. Data on pairs, possibly reflecting distance or similarity measures between genes, are then included through a graph where the nodes represent the genes and the edges are weighted according to the available interaction information. As a probabilistic model, this model has many interesting theoretical features. Also, preliminary experiments on simulated and real data show promising results and points out the gain in using such an approach.

Availability: The software used in this work is written in C++ and is available with other supplementary material at `http://mistis.inrialpes.fr/people/forbes/transparentia/supplementary.html`.

***Index Terms*— Markov random fields, model-based clustering, metabolic networks, gene expression.**

I. Introduction

As an increasing amount of post-genomic data is available, there is a great need to develop methodologies to analyze and to use theinformation contained in this data. A major challenge in bioinformatics is to reveal interactions between components of living organisms and discover the corresponding networks responsible for their biological complexity. In this framework, clustering of genes into groups sharing common characteristics is a useful exploratory technique. It is frequently used as the basis for further computational analysis. As an example, the function of a gene can be predicted according to known functions of other genes from the same cluster. With the introduction of DNA microarray technology, researchers are now able to measure the expression levels of thousands of genes simultaneously at various time points of the biological process or under various experimental conditions. As data accumulate, the tendency to investigate general regulatory mechanisms by clustering genes from their expression profiles increases. A wide range of clustering algorithms have been proposed to analyze gene expression data. Various methods have been applied such as hierarchical clustering [9], self-organizing maps [21], k-means algorithms [23], and more recently Support Vector Machines methods [5] or graph analysis by bi-clustering [22]. More generally, approaches fall mainly in two categories. Some focus on individual data and assume that they are independent. Typically, [26] use a statistically based model which does not incorporate possible relationships between genes. Others try to integrate several sources of data, setting for instance, expression data into a Bayesian graphical model framework [12], combining expression data with phylogenetic profiles [18], or defining distances between genes combining different data types [16]. Typically, the procedure in the work of [11] uses information on pairs of genes in the form of networks or graphs and combines it with distances computed from individual expression profiles. This requires transforming individual information into distances or similarity measures and does not directly use individual data associated to genes in the networks, loosing some potentially interesting information in the process. Kernel-based approaches to data fusion ([15], [25], [24]) also consist of representing various data sets via kernel functions which define generalized similarity relationships. Also, sequential procedures that cluster first individual data alone and incorporate additional information only after the clusters are determined are necessarily suboptimal.

It appears that models able to integrate simultaneously information on individuals (without reducing it to pairwise information) and pairwise relationships in the same procedure have not yet been proposed. The novelty of our work is to propose a model-based approach, as opposed to the distance-based approaches mentioned above, to take into account simultaneously data from individual genes, *ie.* data that make sense and exist for each genes, and data from pairs of genes reflecting for instance some distance or some similarity measure defined on the genes, possibly using some recent kernel-based approaches. To our knowledge, the only similar attempts have been proposed in [19]. However, the formulation of their probabilistic model does not fully exploit gene dependencies. It is written to account for gene interaction but one of the assumptions made is only valid under gene independence. In addition, no estimation procedure is proposed to estimate the model parameters and they then need to be fixed to arbitrary values. We propose an integrated Markov model, meaning by that a specific instance and usage of a Hidden Markov model. Parametric probability distributions account for the distribution of individual data while data on pairs are included through a graph where the nodes represent the genes and the edges are weighted according to the available interaction information (*eg.* distances or similarity measures between genes). As regards parameter estimation and classification step, we consider recent procedures based on the EM algorithm and *mean field*-like approximations [7]. Such procedures were shown to be more efficient in many ways than standard Gibbs samplers or Markov Chain Monte Carlo (MCMC) techniques traditionally used in computer vision.

Manuscript received July 24, 2006; revised February 22, 2007.

M. Vignes is with BioSS at the Scottish Crop Research Institute, Invergowrie, Dundee, DD2 5DA, Scotland, UK.
E-mail: matthieu@bioss.ac.uk

F. Forbes is head of team Mistis, INRIA Rhône-Alpes, ZIRST, 655, avenue de l'Europe, Montbonnot, 38334 Saint Ismier Cedex, France.
E-mail: florence.forbes@inrialpes.fr

This model and the EM classification framework (Section II) have many interesting features. As a probabilistic model, it leads to various possible statistical criteria to select automatically the number of clusters and it provides confidence measures such as posterior probabilities that an object (*eg.* a gene) is assigned to a class. It is flexible in that various pairwise relationship information and features on individual data can be easily incorporated possibly with different weights. Its generalization to include missing data, that often occur when dealing with expression data, is straightforward and its extension to overlapping clustering methods, to deal with more realistic situations where genes can belong to many groups at the same time, can also be considered. Although such a model is relevant in various other applications, we specify in Section III the type of data used in this work. Experiments on simulated data are reported and results on real data are then shown in Section IV. A discussion section ends the paper.

II. Integrated Markov models

The basic assumption is that measures (*e.g.* expression profiles) corresponding to each objects are random variables with a specific probability distribution in each class. A standard way to represent class-specific density functions is to approximate them as Gaussian distributions whose parameters depend on the class. In the work of [26], a Gaussian mixture model is assumed which corresponds to Gaussian class-specific distributions but also to genes independence. This is not fully satisfying since strong neighbourhood relationships between genes sharing common functions can exist. To overcome this limitation, we propose to improve on the Gaussian mixture model by assuming that the distribution of the observed features is that of a Hidden Markov Random Field (HMRF) with K components and appropriate parametrization. To define such a model, one needs to specify a neighbourhood structure indicating which genes are statistically linked but this structure is not necessarily related to the clusters. Dependent genes may be in different classes and genes from the same class may be independent.

A. Hidden Markov fields for biological networks

Let n be the number of genes to be clustered and $x_1, \ldots, x_n$ denote the individual data observed for the genes numbered by $\{1, \ldots, n\}$. The observed data are usually multi-dimensional vectors, *e.g.* expression profiles. For $i = 1, \ldots, n$, we model the probability of observing x_i as

$$P(x_i|\mathbf{\Psi}) = \sum_{k=1}^{K} P(Z_i = c_k|\beta)\ f(x_i|\theta_k),$$

where $f(x_i|\theta_k)$ denotes the multivariate Gaussian distribution with parameters θ_k namely the mean μ_k and covariance matrix Σ_k. Notation Z_i denotes the random variable representing the class of gene i. Z_i can take values in $\{c_k, k = 1 \ldots K\}$ denoting the K possible classes. More specifically, it is convenient to consider c_k as a K-dimensional indicator vector with all components being 0 except the kth which is 1. Note that we assume in this section that K is fixed but this can be generalized (see Section II-B). Notation β denotes additional parameters defining the distribution of the Z_i's and $\mathbf{\Psi}$ denotes the whole model parameters *i.e.* $\mathbf{\Psi} = (\theta_k, \beta, k = 1 \ldots K)$. As an example, the model used by [26] for $P(x_i|\mathbf{\Psi})$ is an *independent* Gaussian mixture model and corresponds, in our framework, to assume that the Z_i's are independent variables. Our approach differs in that our aim is to account for dependencies. This requires the definition of neighbourhood relationships between genes. We will think of a set of genes as a graph with edges emanating from each gene to other genes within its neighbourhood. We will illustrate in Section III how such a graph can be built from biological network data. The dependencies between neighbouring genes are then modelled by further assuming that the joint distribution of $Z_1, \ldots, Z_n$ is a discrete Markov Random Field on this specific graph. Denoting $\mathbf{z} = (z_1, \ldots, z_n)$ specified values of the Z_i's, we define

$$P(\mathbf{z}|\beta) \quad = \quad W(\beta)^{-1} \exp(-H(\mathbf{z}, \beta)) \qquad (1)$$

where $W(\beta)$ is a normalizing constant and H is a function assumed to be of the following form (we restrict to pair-wise interactions), $H(\mathbf{z}, \beta) = \sum_{i=1}^{n} V_i(z_i, \beta) + \sum_{\substack{i,j \\ i \sim j}} V_{ij}(z_i, z_j, \beta)$, where the V_i's and V_{ij}'s are respectively functions referred to as singleton and pair-wise potentials. We write $i \sim j$ when genes i and j are neighbours on the graph, so that the second sum above is over neighbouring genes. Parameters β consist of two sets $\beta = (\alpha, \mathbb{B})$ where α and $\mathbb{B}$ are defined as follows. We consider pair-wise potentials V_{ij} that depend on z_i and z_j but also possibly on i and j. Since the z_i's can only take a finite number of values, for each i and j, we can define a $K \times K$ matrix $\mathbb{B}_{ij} = (\mathbb{B}_{ij}(k,l))_{1 \leq k,l \leq K}$ and write without lost of generality $V_{ij}(z_i, z_j, \beta) = -\mathbb{B}_{ij}(k,l)$ if $z_i = c_k$ and $z_j = c_l$. Using the indicator vector notation and denoting z_i^t the transpose of vector z_i, it is equivalent to write $V_{ij}(z_i, z_j, \beta) = -z_i^t \mathbb{B}_{ij} z_j$. This latter notation has the advantage to make sense also when the vectors are arbitrary and not necessarily indicators. This will be useful when describing the algorithms of Section II-C. Similarly we consider singleton potentials V_i that may depend on z_i and on i, so that denoting by α_i a K-dimensional vector, we can write $V_i(z_i, \beta) = -\alpha_i(k)$ if $z_i = c_k$, where $\alpha_i(k)$ is the kth component of α_i, or equivalently $V_i(z_i, \beta) = -z_i^t \alpha_i$. This vector α_i acts as weights for the different values of z_i. When α_i is zero, no class is favored, *i.e.* for a given gene i, if no information on the neighbouring genes is available, then all classes have the same probability. If in addition, for all i and j, $\mathbb{B}_{ij} = b \times I_K$ where b is a real scalar and I_K is the $K \times K$ identity matrix, parameters β reduce to a single scalar interaction parameter b and we get the Potts model traditionally used for image segmentation. Note that this model is probably the more appropriate for classifying genes since it tends to favor neighbours that are in the same class. However, cases where the $\mathbb{B}_{ij}$'s are far from $b \times I_K$ could be useful in situations where neighbouring genes are likely to be in different classes. Also, when distance or similarity data, $(d_{ij})_{i,j=1,\ldots,n}$, between genes are available, $\mathbb{B}_{ij}(k,l)$ can be decomposed as $\mathbb{B}_{ij}(k,l) = F(d_{ij})\, c(k,l)$ where F is a non increasing function of $\mathbb{R}^+$ and $c(k,l)$ corresponds to a gain (or a loss depending on its sign) of assigning genes i and j respectively to class c_k and c_l. This is part of the flexibility and modelling capabilities of the model. However, without specific information, we can choose $c(k,l) = b$ if $k = l$ and $c(k,l) = 0$ otherwise. In this case, parameter b can be interpreted as a strength of interaction between neighbours. The higher b the more weight is given to the interaction graph. If b is set to 0, only the individual features are taken into account, reducing our model to traditional existing approaches. In practice, these parameters can be tuned

according to expert or *a priori* knowledge or they can be estimated from the data. In the first case, our software can deal with the most general parametrization, namely $\beta = (\alpha_i, \mathbb{B}_{ij})$. In the latter case, the part to be estimated is usually assumed independent of the genes indices i and j, so that in what follows we will reduce α and $\mathbb{B}$ respectively to a single vector and a single matrix. Note that in Section IV, the model is further reduced to α equal to 0 and $\mathbb{B}$ equal to $b \times I_K$ (see comments in this section).

Meanwhile, to keep a general presentation, the observed data is then represented by an HMRF defined by parameters $\mathbf{\Psi}$ being $\mathbf{\Psi} = (\{\mu_k, \Sigma_k\}_{k=1,\ldots,K}, \alpha, \mathbb{B})$.

B. Selecting the number of classes

Choosing the probabilistic model that best accounts for the observed data is an important first step for the quality of the subsequent estimation and classification stages. In statistical problems, a commonly used selection criterion is the Bayesian Information Criterion (BIC) of [20]. The BIC is computed given the data $\mathbf{x}$ and a model M with parameters $\mathbf{\Psi}$. It is defined by:

$$BIC(M) = 2 \log P(\mathbf{x} \mid \mathbf{\Psi}^{ml}) - d \log n \ ,$$

where $\mathbf{\Psi}^{ml}$ is the maximum likelihood estimate of $\mathbf{\Psi}$, $\mathbf{\Psi}^{ml} = \arg\max_{\mathbf{\Psi}} P(\mathbf{x} \mid \mathbf{\Psi}, M)$, d is the number of free parameters in model M and n is the number of observations. BIC allows comparison of models with differing parametrizations and/or differing number of classes. Many other approaches can be found in the literature on model selection but BIC has become quite popular due to its simplicity and its good results. In this study, we will consider the Markov model (α, $\mathbb{B}$) as fixed. More specifically, the experiments reported in Section IV correspond to the simplest model with $\alpha = 0$ and $\mathbb{B} = b \times I_K$. More important in practice is the choice of K and of the covariance model (Σ_k's). For multivariate Gaussian class-specific distributions, there exists a number of different choices for the Σ_k's. See [1] and [6] for a description of these forms and their meaning. The simplest models are those for which the Σ_k's are diagonal matrices. Our choice of K and Σ_k's then can be based on BIC. However, for HMRFs, its exact computation is not tractable due to the dependence structure induced by the Markov model. A possibility is then to compute BIC for independent mixture models, forgetting any spatial information. Not to loose such information, we rather choose to use the mean field like approximations of BIC proposed by [10] (see Section II-C for additional details). As regards covariance matrices, we restrict to diagonal models in most cases or consider an original reduction dimension techniques [4]. In the context of the present work however, we did not observe significant improvement over the simple diagonal models for the data (only 10 dimensional) we consider in Section IV.

C. Classifying genes

Our aim is to classify each gene in one of the K classes. To do so we consider a Maximum Posterior Marginal (MPM) principle consisting in assigning gene i to class c_k that maximizes $P(Z_i = c_k|\mathbf{x}, \mathbf{\Psi})$. Such maximizations depend on $\mathbf{\Psi}$ which is usually unknown, or partly unknown when prior knowledge can be incorporated, and has to be estimated. The parameters to be estimated are the parameters defining the Gaussian distributions namely the μ_k and Σ_k for $k = 1, \ldots K$ and the parameters defining the interaction model, namely the $\alpha(k)$ for $k = 1, \ldots K$ and the $K \times K$ dimensional matrix $\mathbb{B}$. The EM algorithm is a commonly usedalgorithm for parameter estimation in problems with missing data (here the class assignments). For independent mixture models, the independence assumption leads to an easy implementation of the algorithm. For HMRFs, due to the dependence structure, the exact EM is not tractable and approximations are required. In this paper, we use some of the approximations presented in [7]. These approximations are based on the mean field principle which consists in replacing the intractable Markov distributions by factorized ones for which the exact EM can be carried out. This allows to take the Markovian structure into account while preserving the good features of EM. [7] generalized the mean field principle and introduced different factorized models resulting in different procedures. Note that in practice, these algorithms have to be extended to incorporate the estimation of matrix $\mathbb{B}$ and to include irregular neighbourhood structure coming from biological networks and not from regular pixel grids like in [7].

Briefly, these algorithms can be presented as follow. They are based on the EM algorithm which is an iterative algorithm aiming at maximizing the log-likelihood (for the observed variables $\mathbf{x}$) of the model under consideration by maximizing at each iteration the expectation of the complete log-likelihood (for the observed and hidden variables $\mathbf{x}$ and $\mathbf{z}$) knowing the data and a current estimate of the model parameters. When the model is an Hidden Markov Model with parameters $\mathbf{\Psi}$, there are two difficulties in evaluating this expectation. Both the normalizing constant $W(\beta)$ in (1) and the conditional probabilities $P(z_i \mid \mathbf{x}, \mathbf{\Psi})$ and $P(z_i, z_j \mid \mathbf{x}, \mathbf{\Psi})$ for j in the neighbourhood $N(i)$ of i, cannot be computed exactly. Informally, the mean field approach consists in approximating the intractable probabilities by neglecting fluctuations from the mean in the neighbourhood of each gene i. More generally, we talk about mean field-like approximations when the value for gene i does not depend on the value for other genes which are all set to constants (not necessarily to the means) independently of the value for gene i. These constant values denoted by $\tilde{z}_1, \ldots, \tilde{z}_n$ are not arbitrary but satisfy some appropriate consistency conditions (see [7]). Let $z_{N(i)}$ denote the set of variables $\{z_j, j \in N(i)\}$ associated to the set $N(i)$ of neighbours of i. It follows that $P(z_i \mid \mathbf{x}, \mathbf{\Psi})$ is approximated by

$$\begin{aligned} P(z_i \mid \mathbf{x}, \tilde{z}_{N(i)}, \mathbf{\Psi}) \quad &\propto \quad f(x_i|z_i^t\theta).P(z_i|\tilde{z}_{N(i)}, \beta) \\ &\propto \quad f(x_i|z_i^t\theta). \\ &\qquad \exp[z_i^t(\alpha + \mathbb{B} \sum_{j \in N(i)} \tilde{z}_j)], \end{aligned}$$

where θ denotes the vector $(\theta_1, \ldots, \theta_K)$. The normalizing constant is not specified but its computation is not an issue. Then, for all $j \in N(i)$, $P(z_i, z_j \mid \mathbf{x}, \mathbf{\Psi})$ is approximated by $P(z_i \mid \mathbf{x}, \tilde{z}_{N(i)}, \mathbf{\Psi})\ P(z_j \mid \mathbf{x}, \tilde{z}_{N(j)}, \mathbf{\Psi})$. Both approximations are easy to compute. Using such approximations leads to algorithms which in their general form consist in repeating two steps. At iteration q,

(1) Create from the data $\mathbf{x}$ and some current parameter estimates $\mathbf{\Psi}^{(q-1)}$ a configuration $\tilde{z}_1^{(q)}, \ldots \tilde{z}_n^{(q)}$, *i.e.* values for the Z_i's. Replace the Markov distribution $P(\mathbf{z}|\beta)$ of (1) by the factorized distribution $\prod_{i=1}^{n} P(z_i|\tilde{z}_{N(i)}^{(q)}, \beta)$. It follows that the joint distribution $P(\mathbf{x}, \mathbf{z}|\mathbf{\Psi})$ can also be approximated by a factorized

distribution:

$$\prod_{i=1}^{n} f(x_i|z_i^t\theta)P(z_i|\tilde{z}_{N(i)}^{(q)},\beta)$$

and the two problems encountered when considering the EM algorithm with the exact joint distribution disappear. The second step is therefore,

(2) Apply the EM algorithm for this factorized model with starting values $\mathbf{\Psi}^{(q-1)}$, to get updated estimates $\mathbf{\Psi}^{(q)}$ of the parameters.

In particular the *mean field* and *simulated field* algorithms consist in two different ways of performing step **(1)**. The *mean field* algorithm consists in updating the $\tilde{z}_i^{(q)}$'s by setting, for all $i = 1, \ldots, n$, $\tilde{z}_i^{(q)}$ to the mean of distribution $P(z_i \mid \mathbf{x}, \tilde{z}_{N(i)}^{(q)}, \mathbf{\Psi}^{(q-1)})$. Note that as z_i is an indicator vector, the mean value $\tilde{z}_i^{(q)}$ is a vector made of the respective probabilities to be in each of the K classes. In the *simulated field* algorithm, $\tilde{z}_i^{(q)}$ is simulated from $P(zi \mid \mathbf{x}, \tilde{z}_{N(i)}^{(q)}, \mathbf{\Psi}^{(q-1)})$. Note also that to save additional notation, the updating described above is synchronous while we actually implemented a sequential updating of the $\tilde{z}_i^{(q)}$'s: each node i is updated in turn using the new values of the other nodes as soon as they become available rather than waiting until all nodes have been updated. Intuitively, the stochastic feature of the *simulated algorithm* enables to avoid convergence towards a saddle point or a local minimum, dependence of the converging state towards initialization or slow rate of convergence. These well-documented pitfalls of the EM algorithm are sorted out in the spirit of Stochastic EM (SEM [27]). However both algorithm cannot be really compared since SEM is not Markovian: it doesn't account for dependencies between observations and is well suited to deal with mixture models.

In practice, at step **(2)**, performing one EM iteration is usually enough. Then, the HMRF estimation provides us with estimations for the means and covariance matrices of the K Gaussian distributions, namely μ_k and Σ_k for $k = 1, \ldots K$, but also for the hidden field parameters, matrix $\mathbb{B}$ and vector α. It follows easily approximations of the $P(Z_i = c_k|\mathbf{x}, \mathbf{\Psi})$ required to classify each genes using the MPM principle.

In this work, we mainly consider the so-called *simulated field* (SF) algorithm for its better performance in practice.

III. From biological networks to interaction graphs

Many kinds of biological networks are freely available. They contain a lot of information that should not be ignored to provide optimal clustering but the quality and the access to that information is far from being uniform. As an example, biological networks are not all related to the same objects. They may contain links between genes, gene products, proteins complexes or families, *etc.* and the links may stand for experimentally based or assumed relationships. Our goal is to build a graph with objects which are individually subject to other measurements, as genes are to microarrays. There is no universal way to build such a graph but we give an illustration in this section. We choose to focus on gene expression data and metabolic networks like those given in the Reaction (part of Ligand) KEGG database (`http://www.genome.ad.jp/kegg/reaction/`). A mapping between genes and objects in the network must then be derived. Chemical reactions of interest are those which are assigned one or several EC numbers corresponding to enzymes that may catalyze them. Toeach EC number are associated one or more genes.

A first stage consists in building a graph whose nodes are enzymes. An edge exists between two enzymes if and only if they catalyze two reactions that share at least a common chemical compound either as substrate or product. The interpretation is that an edge stands for the possibility that two reactions follow each other in metabolic pathways. However, all the links between reactions cannot be considered. In particular those which involve compounds that are very common (*eg.* water, *etc.*) are usually not relevant to the biological interpretation and may hide or bias the biological information. Two possibilities are either to use the main compounds (according to KEGG database) or to remove compounds which would link too many reactions (above a given threshold). We choose the first solution for the restricted database has the advantage of being produced by experts who manually removed somewhat irrelevant compounds such as water, carbon dioxide, *etc.* In addition, weights $(d_{ij})_{i,j=1,\ldots,n}$ can be assigned to edges. They may reflect in a quantitative way enzymes proximity or thermodynamical properties. Such information is not available yet but would be easily dealt with in our model. As an illustration, we consider the *Saccharomyces cerevisiae* genome and derive a graph on genes with the network of chemical reactions given in the database. Figure 1 gives an example based on the compounds acting in the `Citrate cycle` with only part of the reactions represented for clarity (Figure 1 (A)). For example, reactions $R00479$, $R01900$ and $R00709$ all share compound `Isocitrate`. They are therefore neighbours and so are the enzymes catalyzing each of these reactions (Figure 1 (B)). Two enzymes catalyzing the same reaction are neighbours as well (*eg.* $EC\,2.3.3.8$ and $EC\,2.3.3.1$). Reversibility is allowed. Also a common enzyme may catalyze different reactions, *eg.* $EC1.1.1.42$ is active in reactions $R01899$, $R00268$ and $R00267$. It must then be linked to any enzyme catalyzing reactions sharing a compound with the later.

A second stage in building the final graph is to go from enzymes to genes. Two cases have to be considered. In the first one a gene maps to several enzymatic functions while in the second one several genes map to a single enzymatic EC number. A way to deal with both cases is to consider couples of objects $(gene, EC)$ and connect them in the graph as soon as their second components are connected. In the first case, enzymes already correspond to different nodes. These nodes only need to be fused keeping neighbourhood relationships. The measure about the gene is then assigned to the resulting node. The second case is illustrated in the transition from graph (B) to graph (C) of Figure 1. Links (see graph (B)) between enzymes correspond to solid lines while each set of associated genes corresponds to dotted lines. A node is added for each of the gene corresponding to the same enzymatic function. New nodes are then linked to keep the same relationships than that existing between enzymes. In our example (Figure 1 (C)), $EC\,4.2.1.3$ splits into genes $YJL200C$ and $YLR304C$. Note that information related to $EC\,2.3.3.8$ is lost because no known yeast gene is assigned to that enzyme. Besides an obvious limitation of our graph construction is that it ignores genes not related to EC numbers. Many of them (*eg.* regulators) can be responsible for relevant interactions. A more complete (and less automatic) graph construction would have required additional expert knowledge not available in this study. Note that beyond the biological relevance, the size of the graph is not a problem. The model can deal with large numbers of genes, edges and experiments. In different contexts, experiments

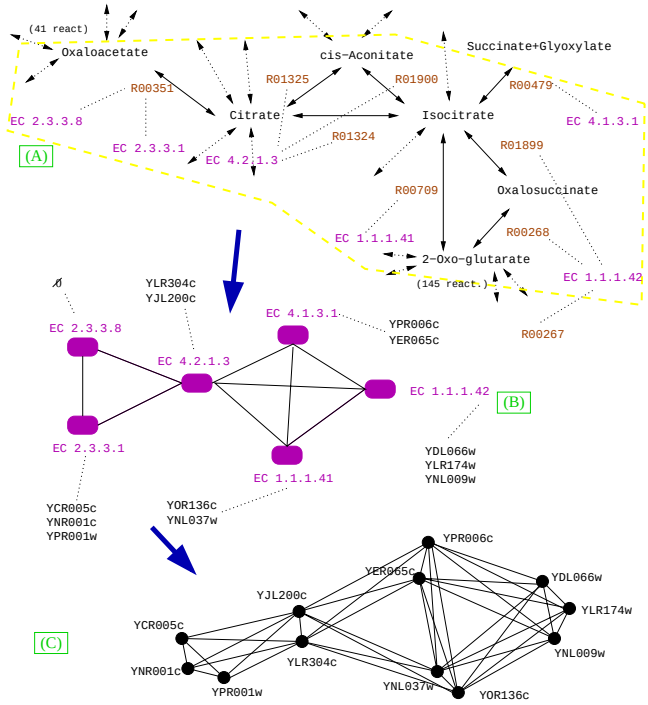


Fig. 1. From the graph of chemical metabolic reactions (A) to the gene interaction network (C) *via* the enzyme network (B). For clarity, only edges between reactions in the metabolic subgraph (A) are represented.

were carried out with the equivalent of thousands of genes and edges and up to 300 experimental conditions (dimension of the data) using diagonal covariance matrices or dimension reduction techniques of [4].

IV. Results

As mentioned earlier, the experiments reported in this section correspond to the simplest Markov model with $\alpha = 0$ and $\mathbb{B} = b \times I_K$. In particular for the yeast data, more complex models, when estimated, seem to be penalized by their larger number of parameters (see Figure 5 (b) for an illustration).

A. Synthetic data sets

We first assess our method performance on synthetic data for which the classes are known. Modelling gene expression data sets is an ongoing effort by many researchers and there is no well-established model to represent gene expression data yet. The simulation method we use is based on a proposal by [26]. It aims at simulating cyclic data, *ie.* cyclic behavior of genes over different time points. We create five data sets following the same model. Each set is made of 1536 genes for which we simulate 20 experiments. These genes come from 6 classes equal in size (256 genes per class) corresponding to different behavior over the time course. Let x_{ij} be the simulated expression level of gene i under experiment j in the data set. We first consider the following periodic behaviors (before adding noise). When the gene class is $z_i = c_k$ with $k = \{1, \ldots, 4\}$, we set

$$y_{ij} = sin(2\pi j/10 - \pi k/4) \quad \text{for } j = 1, \ldots 20.$$

When $k = 5$ and $k = 6$, we consider the linear behaviors $y_{ij} = j/20$ et $y_{ij} = -j/20$. Noise is then added,

$$x_{ij} = y_{ij} + \epsilon_{ij} \quad \text{for } i = 1, \ldots 1536 \text{ and } j = 1, \ldots, 20,$$

where the ϵ_{ij}'s are generated according to the normal distribution with mean 0 and standard deviation σ_{ij}. The σ_{ij}'s are drawn, randomly from standard deviations observed on the real data described by [13]. We further increase the noise by multiplying the ϵ_{ij}'s by 6 (the corresponding standard deviation is then $6 * \sigma_{ij}$). We refer to [17] and the web site `http://expression.washington.edu/publications/kayee/medvedovic_bioinf2003/` for a graphical illustration of such data (see also supplementary material).

As regards network data, we are not aware of any well established simulation methods. For a simple illustration, we consider the genes as the nodes of a 48×32 regular grid with neighbourhoods made of the 8 nearest neighbours. The 6 classes are then chosen as shown in Figure 2 (left-hand image) where each color is associated to a class. Although such a network has no biological interpretation, the classification quality is easy to assess by non expert users and it provides a clear visual illustration of the gain in taking into account network relationships. We compare the standard EM algorithm, which assumes genes independence and the EM-like procedures we propose. BIC is computed in both cases for $K = 3$ to $K = 9$. Typical curves are shown in Figure 2. EM-like procedures show higher BIC values than standard EM. The criterion selects the right number of classes except for data sets 1 and 4 for which 7 classes are preferred. However, this is consistent with the obtained classifications shown in Figure 3. For standard EM, 2 bands are wrongly merged except for data set 2. For the *simulated field* algorithm, the bands are correctly recovered except for sets 1 and 4 . In these latter cases, the 7-group classifications are visually better for data sets 1 and 4 (bottom row of Figure 3) as suggested by BIC values. The interpretation is that in these very noisy cases it may be worth considering an extra class with no specific meaning but that gathers outliers or too ambiguous measures. *Simulated field* and *mean field* algorithms perform similarly except for data set 5. In this case the *simulated field* algorithm selects 6 classes and gives a better classification. In the following developments, we will only refer to the *simulated field* algorithm.

Table I shows the global recognition rates (proportions of well-classified genes) obtained with the EM and *simulated field* algorithms for each data sets, while Table II shows the confusion matrix obtained for set 5. Rows correspond to the true classes while columns correspond to the obtained classes. The diagonal terms are the proportions of well classified genes in each class. The other terms are proportions of badly classified genes. All data sets show similar improvements when comparing EM to the *simulated field* algorithm.

On such synthetic data, the gain in taking into account network information or dependencies between genes through their labels appears clearly with improved recognition rates. BIC or its approximation in our Markov field setting, also appears to be satisfying criterion as regards the selection of the number of classes. It selects a number of classes which is consistent with the visual quality of the corresponding classification. These first conclusions will guide, in the next section, our analysis of the experiments on real data for which no ground-truth is available.

B. Saccharomyces cerevisiae (yeast) data

Although, our approach is valid for any organism provided individual data and network information is available, we focus

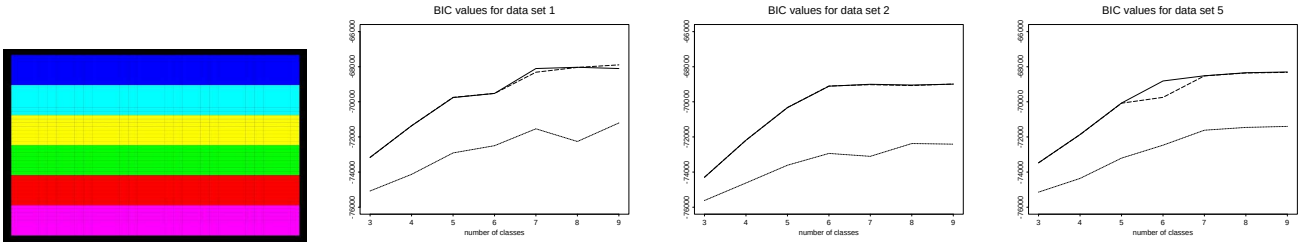


Fig. 2. Reference classification and BIC values for 3 data sets when K varies from 3 to 9. Solid line: Simulated field algorithm, Dotted line: EM algorithm, Dashed line: Mean field algorithm.

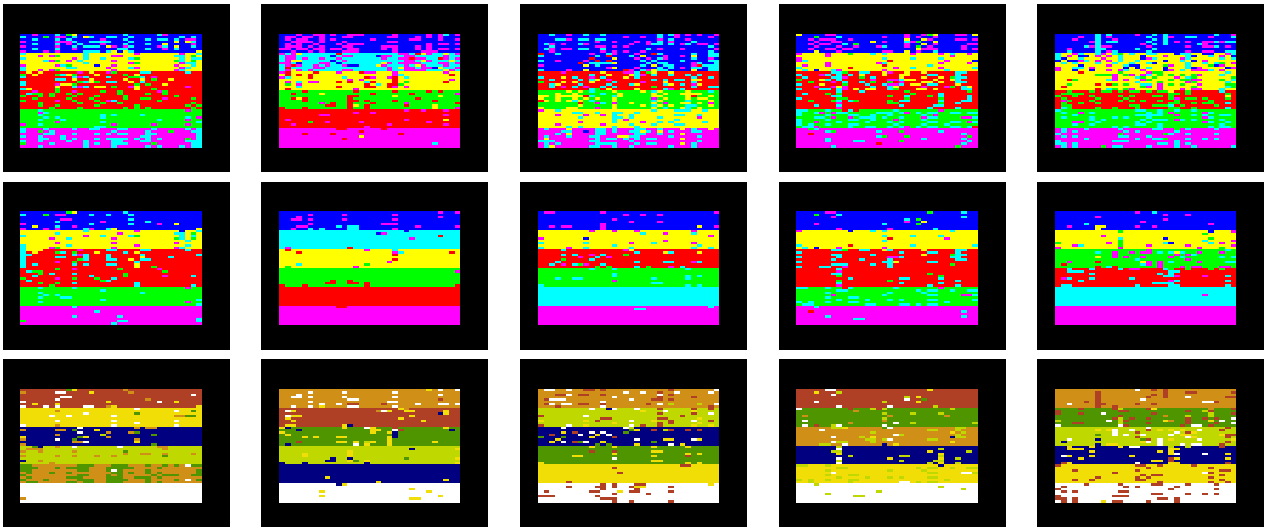


Fig. 3. Top and middle rows: 6 color classifications for 5 synthetic data sets using standard EM algorithm assuming independence (top row) and simulated field algorithm (middle row). Bottom row: 7 color classifications using the Simulated field algorithm. Note that the colors are arbitrarily assigned and may not match.

TABLE I
RECOGNITION RATES IN % FOR SYNTHETIC DATA ($K = 6$).

data sets	1	2	3	4	5
EM	64.8	79.2	63.3	68.1	64.3
Simulated field	77.5	95.8	93.6	78.6	91.3

TABLE II
CONFUSION MATRIX FOR FIGURE 3 MIDDLE RIGHT IMAGE.

global recognition rate= 91.3 %						
Class	1	2	3	4	5	6
1	94.1	1.2	0	0	0.8	3.9
2	1.2	89.1	3.1	0	2.0	4.7
3	0	1.2	80.9	1.6	5.9	10.5
4	0	0	1.6	84.0	12.1	2.3
5	0	0	0	0	99.6	0.4
6	0	0	0	0	0	100

on data related to *Saccharomyces cerevisiae* which is a widely studied organism with well established information and data on its mechanisms. The expression data we use are described by [8] and correspond to the developmental program of sporulation (gametogenesis in yeast). It consists of meiosis overlapped by spore formation. Sporulation can be characterized in terms of four distinct sets of genes which play different sequential roles according to their transcriptional activation during the process: early, middle, mid-late and late. The study proved this characterization to be suboptimal and a seven expression patterns description was preferred. Changes in the concentrations of the mRNA transcripts from each gene were measured at seven successive intervals after synchronisation; yeast cells were transferred to a nitrogen-limited medium that induces sporulation. The samples were taken at times (0h, 0.5h, 2h, 5h, 7h, 9h, 11.5h) based on the independently monitored expression pattern of known early, middle, mid-late, and late genes. Three additional points were measured when an essential transcription factor known to be activated at the end of the meotic prophase is missing; cells are then non-sporulating. The measures we use are related to these specific times. This leads to 10 dimensional profiles that should capture essential activity behavior of yeast genes during sporulation. As regards network data, we use the KEGG Reaction database as described in Section III. The resulting graph consists of 635 genes (amongst the 6118 ORFs expression measurements available, only 635 are present in the metabolic network). Since our aim is mainly to assess the benefit in adding network information, we then restrict to these 635 genes. The networks has 7111 edges and 92% of pairs are connected. Figure 4 gives a summary of characteristics of the network. In particular, the graph on the left reports that a significant number of edges are very connected : 245 nodes have more than 20 neighbours and 78 of them have 50 or more neighbours. The most connected nodes ($YER005W$) was found to have 145 interactors.

In this case, the appropriate number of classes is unknown. We compute BIC values for $K = 2$ to $K = 10$. The corresponding

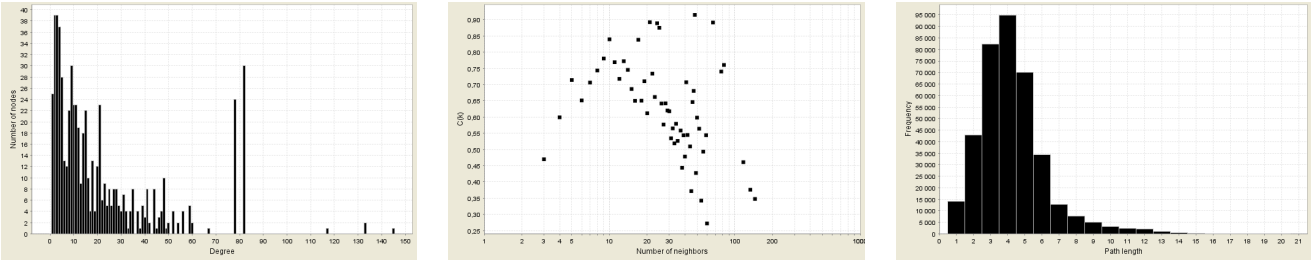


Fig. 4. Left: distribution of the node degrees of the network under study. Center: average clustering coefficients of nodes vs their number of neighbours. Right: distribution of the shortest path between nodes.

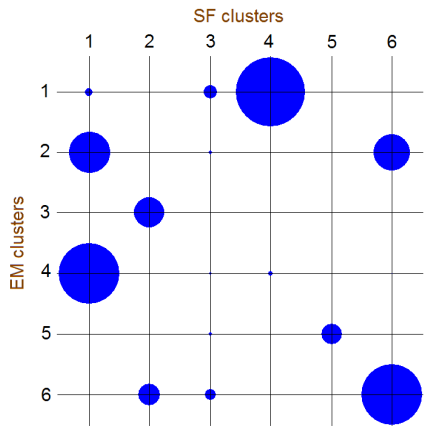


Fig. 6. Visualization of the confusion matrix of EM and SF classifications. Circle radius are proportional to cluster intersection cardinals. Note that SF and EM cluster labels have no special meaning and were arbitrarily and independently assigned as outputs of both clustering algorithm.

TABLE III
SOME CHARACTERISTICS OF SF CLUSTERS.

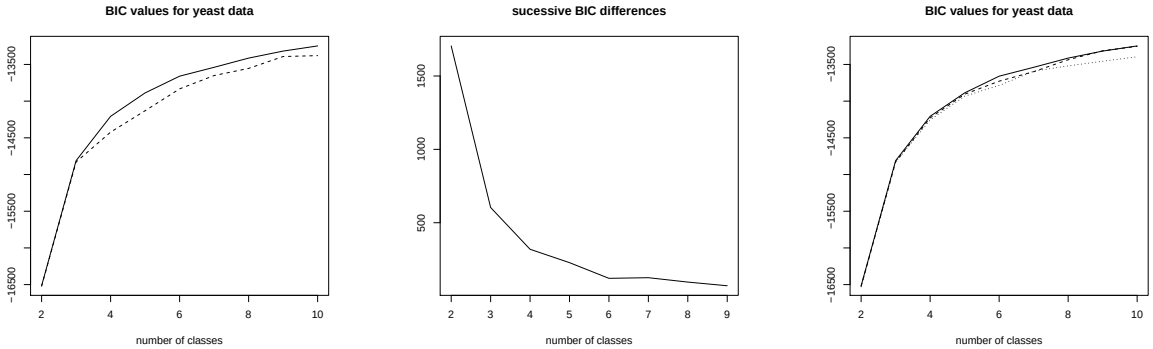
k	1	2	3	4	5	6
Nodes	180	86	52	123	34	160
Internal edges	537	195	75	254	8	903
% of connected pairs	14	66	17	14	1	16
Diameter	8	12	5	8	2	8

curve (Figure 5 (a)) does not show a clear maximum. We then consider as a reasonable choice of K, the value after which the difference in two successive BIC does not increase significantly anymore. This leads to selecting $K = 6$ as the number of classes (Figure 5 (b)). We then compare into more details classifications obtained with standard EM and with the *simulated field* algorithm for parameter $K = 6$. A visualization of the confusion matrix between SF and EM classifications is provided on Figure 6. While some clusters remain very similar (even identical in the cas of clusters SF and EM 5), it is sometimes difficult to identify correspondance; EM cluster 2 splits into SF clusters 1 and 6 and SF cluster 2 content comes from EM clusters 3 and 6. Moreover, characteristics for SF clusters are reported in Table III. All results are made available on the supplementary website.

To assess the quality of such classifications is not an easy task since there is no universal criteria to measure the relative performance of the algorithms. We therefore illustrate the gain in using our approach on the following specific features chosen for their relevance with regards to the data under consideration. Note that presenting the resulting clustering as a whole is not possible due to the size of the graph. An appropriate visualization tool is missing to provide a global biologically meaningful idea of the clusters. However, the clusters are available in separate files on our website.

Ideally, we would like to check whether our approach results in clusters better related to real biological networks. Since this experiment is based on a graph that accounts for dependencies that are expected to be strongly related to pathway information, we assess the quality and relevance of the various clusters by comparing them to groups of genes in the same metabolic pathway or in related pathways. [14] propose a method to detect significant pathways associated to the [8] expression data set we are using. They describe three scoring functions to characterize pathways at the transcriptional level based on gene expression, coregulation and cascade effect. Their pathway scores show relevance towards the biological background. This work provides an interesting tool to evaluate the performance of gene expression clustering techniques. High *Activity Scores* are awarded to pathways that exhibit many genes expressed above a given threshold or under another threshold in the case of repression effects. *Coregulation Scores* are higher for pathways in which genes show greater similarity in their expression patterns. *Cascade Scores* account for genes that do not show huge deviation from the reference time point and for the structure and ordering of the reactions in the pathway. In particular, they are useful to find out in which pathway a reaction chain is active or shut down for the particular experiment under study. For example, Transcription/Translation pathways are given a high *Activity Score* in results of [14]. This is well captured by our *simulated field* algorithm which gathers 16 out of the 28 genes involved in Transcription mechanisms in cluster 6. In comparison, standard EM succeeds in gathering 11 of these genes at best. When considering two clusters, these numbers raise respectively to 24 genes for our approach against 19 for the independent gene case (see Figure 7). Note that due to the restriction of our data set, we have no gene corresponding to B12 in our data although it corresponds to some yeast gene. Similar results hold for Translation involved genes.

We can also refer to the Vitamins metabolism that is given a high *Cascade Score* by [14]. The *simulated field* algorithm gathers 26 genes in the same cluster while EM recovers 19 at best. If two clusters are merged, these numbers respectively raise to 44 and 35 genes out of the 70 involved in the Vitamins Metabolism. Another pathway that is reported to be related to sporulation is the Oxydative Phosphorilation pathway that has a high *Coregulation Score* in [14]. Our method finds 24 genes in cluster 6 while EM groups at most 16 out of the 52 genes involved. The detected genes are up-regulated at the second time point and



(a) (b) (c)

Fig. 5. BIC values for yeast data when K varies from 2 to 10. **(a)** comparing Simulated field and EM algorithms. Solid line: Simulated field algorithm, Dashed line: EM algorithm; **(b)** Differences in two successive BIC for the Simulated field algorithm; **(c)** comparing various Markov models for the Simulated field algorithm. Solid line: $B = b \times I_K$ model, Dashed line: diagonal B model, Dotted line: full B model.

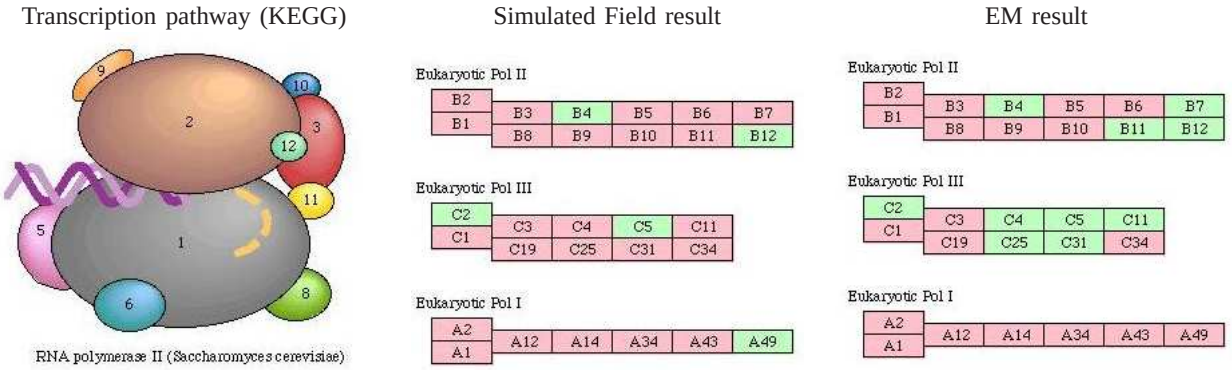


Fig. 7. RNA polymerase Transcription pathway as taken from KEGG. The middle and right columns show the results obtained by merging two clusters, respectively using the *simulated field* algorithm (middle) and the EM algorithm (right). Pink colored proteins correspond to genes that are included in the two merged clusters while green ones correspond either to genes that do not belong to the clusters or to genes that were not included in the analysis. The *simulated field* algorithm (missing proteins: B4, B12, C2, C5, A49) outperforms EM (missing proteins: B4, B7, B11, B12, C2, C4, C5, C11, C25, C31) in grouping together this class of genes.

are specific to ATP synthesis. The analysis shows that cluster 6 is related to Energy metabolism (*eg.* Oxydative Phosphorilation) as well as metabolisms that deal with Transcription (*eg.* RNA polymerase), Translation (*eg.* Aminoacyl-tRNA synthetase) and Vitamins. Other pathways can be more fully recovered using our approach and the additional graph information. As an illustration, for the glycolysis pathway, 24 genes belong to the same *simulated field* cluster while EM groups 19 out of the 44 in our data set. These results show the inclination of our method to a better sensitivity to group genes involved in meaningful identified metabolism subunits than standard EM. Figure 8 shows genes assigned to *simulated field* cluster 2 that are involved in Glycolysis. This cluster is mainly related to Carbohydrate metabolism (see Table IV).

Our method has the ability to group genes with a coordinated activity during glycolysis despite some expression dissimilarities. This is the case for $YLR153C$ ($EC6.2.1.1$) and $YPL061W$ ($EC1.2.1.3$) which have a slowly increasing expression while genes in the main way converting glucose 6-phosphate into pyruvate (or conversely) are immediately over-expressed. As a matter of fact the two former genes are not assigned to the same cluster as the others when using standard EM. The glycolysis example suggests that, as expected, our method outperforms traditional clustering methods in grouping functionally related genes into clusters even if their expression pattern is not a sufficient clue.

To further assess the gain in using network information, we also consider an ontological analysis approach to help with the biological interpretation of the results. We used the 1935 GO terms available -out of them is a subset of 1016 terms involved in *biological process*)- at the time of the study, from the Gene Ontology (`http://www.geneontology.org/`) database. The full list and additional information is made available on our website (`http://mistis.inrialpes.fr/people/forbes/transparentia/supplementary.html`). Two series of statistical tests are driven. The null hypothesis always being that a GO term is not over/under-represented and the alternative being that a GO term is over- (first series) or under-represented (second series). P-Values are computed with False Discovery Rate (expected proportion of erroneous rejections among all rejections) corrections, which addresses the multiple testing issue. Moreover, arbitrary dependencies between groups are taken into account (see [3]). The power of this correction is lower than in [2] (null hypothesis are more easily accepted even if wrong) that is valid under positive regression dependencies. It is not sure wheither the GO terms hierarchy fulfills this statement. Eventually, we will miss some over-represented GO categories but we can be confident about identified ones. This is the price to pay to account for multiple testing under arbitrary

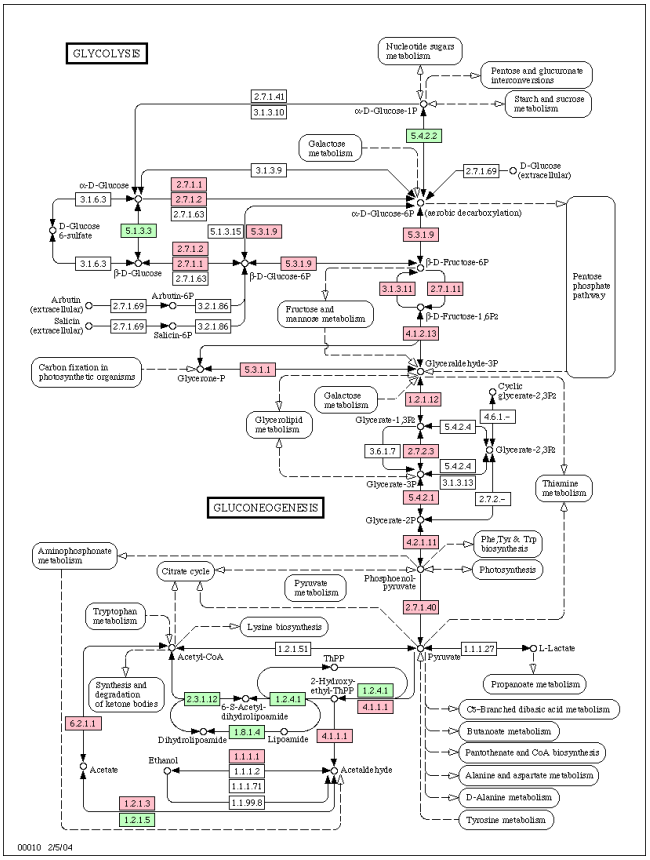


Fig. 8. Glycolysis pathway: colored EC numbers are in our data set. Pink ones belong to the same *simulated field* cluster while green ones (numbers 5.4.2.2, 5.1.3.3, 2.3.1.12, 1.2.4.1, 1.8.1.4, 1.2.1.5) do not.

dependencies. The analysis is summarized in Table IV which shows respective P-values for over-represented GO terms in the clusters found by the *simulated field* and standard EM algorithms. Note that since clustering results for *simulated field* and standard EM algorithms differ, the cluster numbering corresponds to the *simulated field* algorithm. The last column of the table shows the best corresponding P-Values computed among all the EM clusters. Although this does favor EM, the results show that the *simulated field* approach still performs better. Under-represented GO-terms are not listed for sake of brevity and because most of over-represented GO terms in one cluster show under-representation in the other clusters and this with significant P-values.

The ontological analysis is consistent with the previous observations on pathways. In addition, it suggests that our method tends to produce clusterings with more specificity than traditional EM in the sense that GO terms that are significantly over-expressed in one cluster are significantly under-expressed in the other clusters. This is usually true but to a much lesser extent for clusters found by EM. The *simulated field* algorithm provides highly specific clusters. To exhibit such a property we considered GO terms with a reasonnable number of components (ten or so). GO terms gathering are not specific to a distinctive class of genes. They do not give a satisfactory evidence that our method can distinguish between genes specific to some processes. GO terms with a too low number of genes cannot lead to a real validation of the method neither.

The genes classified under GO term $GO : 0008652$: amino-

TABLE IV

ONTOLOGICAL ANALYSIS: OVER-REPRESENTED GO CATEGORIES RELATED TO THE DIFFERENT CLUSTERS AND CORRESPONDING P-VALUES. SIMULATED FIELD (RESP. STANDARD EM) P-VALUES ARE COMPUTED FOR SIMULATED FIELD (RESP. STANDARD EM) CLUSTERS.

Simul. field Cluster	GO terms	Simul. field P-values	Standard EM P-values
1	GO:0008652: amino-acid biosynth.	**1.1E-2**	0.193
2	GO:0006006: glucose metabolism	**1.2E-7**	8.7E-7
	GO:0006090: pyruvate metabolism	5.9E-5	**8.7E-7**
	GO:0006144: purine base metabolism	**2.2E-2**	0.259
	GO:0015980 : energy dev. by oxid...	**1.8E-2**	3.3E-2
3	GO:0006259: DNA metabolism	**4.1E-2**	1
	GO:0006261: DNA-dep. DNA replic.	**4.1E-2**	0.193
	GO:0006271: DNA strand elong.	**4.1E-2**	0.208
4	no significant GO term	N.A.	N.A.
5	GO:0030437: sporulation	**4E-2**	0.26
6	GO:0006360: transcr. from RNA pol.	**1.6E-2**	2.5E-2
	GO:0006164: purine nucleo biosynt.	**2.0E-2**	6.1E-2

acid biosynthesis (see Table IV) are a first example. The P-Value (1.1%) shows that the *simulated field* cluster (1) is highly specific towards this function whereas the corresponding standard EM cluster isn't (P-value equal to 0.193). *Simulated field* cluster 5 is an even more relevant example. It contains most of the sporulation specific genes ($GO : 0030437$) listed in [8] (available on the paper website or with our data in supplementary material). The test conclusion is that this cluster is specific towards the invoked function with a P-Value of 4%. For comparison, the best results among standard EM clusters is 0.26 which does not lead to the conclusion that this term is over-represented. Note that this is somewhat surprising since these genes are apparently not linked by any of the association types provided in the STRING database (`http://string.embl.de`). We looked for links related to databases, co-expression, physical location on the chromosome, fusion, experiments, co-occurrence in different genomes. But only a text-mining link was detected, certainly due to the fact that many of the genes are referenced in the [8] paper. According to this paper, 32 among 34 genes in cluster 5 take a significant part in the temporal program of yeast sporulation. This cluster does not include however the class of *metabolic* genes (quickly induced) that are mainly recovered in another much bigger cluster. A possible interpretation is that these latter genes have a quite different regulator.

V. DISCUSSION AND CONCLUSION

Our aim was to show that Hidden Markov models could be introduced to incorporate various types of information about biological objects (*eg.* genes) and in particular to account for interactions between these objects (through biological networks for instance). We focused on the task of classifying genes from their expression profiles and from metabolic pathways data as an illustration. The introduction of Markov models in this context is new. They provide parametric models where the parameters have a natural interpretation. Some of them (the α_k's) can be related to class proportions while others (matrix $\mathbb{B}$) to pair-wise interactions (see Section II-A). In our method, parameters are estimated but tuning is also possible, for instance, to incorporate *a priori* knowledge regarding class proportions or strength of interactions to put more weight on network data. Other clustering methods are much less readable in that sense.

Preliminary results are promising. Experiments on simulated data show that our approach can improve significantly classifica-

tion rates. They also suggest that criteria based on BIC could be used to guide the choice of the number of classes. Additional experiments on real data (yeast) point out further interesting features of our approach. The *simulated field* algorithm leads to biologically more plausible and more fully identified clusters. When compared to clustering methods based on gene expression only (*eg.* EM clustering), it has the advantage to produce clusters associated to pathways with possible coordinated change in gene expression. When compared with methods incorporating network data, it has the advantage to consist in a statistically well founded approach which does not require to choose a distance or a kernel function and allows further statistical analysis regarding additional issues such as model selection. It is also part of the *soft* clustering methods that provide membership probabilities instead of *hard* (usually more biased) classifications.

Future work would be to investigate this general methodology in other contexts, with applications in proteomics, using genes or proteins as central concepts through a variety of information sources such as sequences, structures, expression patterns, position in networks, *etc*. Additional experiments could also be useful to challenge our model on incomplete interaction data. The difficult passage here is to determine the full pair interaction network. In the synthetic data case, the 8 nearest neighbourhood setting is chosen because it is widely used in image analysis. But it doesn't claim to be the network that accounts at best for interactions in an image. In the real biological dataset, we are certainly faced to a graph only summarizing a fraction of the real hypothetical gene interaction network of *Saccharomyces cerevisæ*. Biological interaction networks are known to be incomplete and the reliability of the interactions vary a lot. Hence it makes it difficult to assess the proportion of information initially present in the network. A possibility would be to take into account a wider range of pair dependencies and to assign weights according to prior knowledge on the reliability of the interaction. Our model was designed to deal with such a refinement. This kind of information is not available yet on the metabolic network we considered. Alternately, it might be worth building an interaction network using a database like STRING (`http://string.embl.de/`) which offers confidence level for gene interactions from various sources (physical chromosome neighbourhood, two-hybrid, literature link,...). Before that, more specific analysis would be useful as regards the generalization to missing data that often occur in biological studies. Our mean field-like framework allows such a generalization. Also, in a variety of applications, overlapping clustering, wherein some items are allowed to be members of two or more discovered clusters, is more appropriate. Methods have been proposed that would be worth investigating in the context of genetic data analysis.

ACKNOWLEDGMENT

The authors would like to thank Frédéric Boyer and Juliette Blanchet for their help with the data and the experiments. We are also grateful to Alain Viari and Éric Coissac for fruitful discussions. Lastly we would like to thank anonymous reviewers for their detailled comments and suggestions on our work.

REFERENCES

[1] J. D. Banfield and A. E. Raftery, "Model-based Gaussian and non Gaussian Clustering", *Biometrics*, vol. 49, no. 3, pp. 803–821, Sept. 1993.

[2] Y. Benjamini and Y. Hochberg, "Controlling the false dicovery rate - a practical and powerful approach to multiple testing", *Journal of the Royal Statistical Society B*, vol. 57, no.1, pp. 289–300, Feb. 1995.

[3] Y. Benjamini and D. Yekutieli, " The control of the false discovery rate in multiple testing under dependency," *Annals of Statistics*, vol. 29, no. 4, pp. 1165–1188, Aug. 2001.

[4] C. Bouveyron, S. Girard and C. Schmid, "Class specific subspace discriminant analysis for high dimensional data," In *Lect. Notes Comp. Sci., Springer*, no. 3940, pp139–150, 2006.

[5] M.P.S. Brown, W.N. Grundy, D. Lin, N. Cristianini, C.W. Sugnet, T.S. Furey, M. Ares Jr. and D. Haussler, "Knowledge-based analysis of microarray gene expression data by using support vector machines," *Proc. Nat. Acad. Sci.*, vol 97, no.1, pp.262–267, Jan 2000.

[6] G. Celeux and G. Govaert, "Gaussian Parsimonious clustering models", *J. of Pat. Rec. Soc.*, 28, pp. 781–793, 1995.

[7] G. Celeux, F. Forbes and N. Peyrard, "EM procedures using mean-field like approximations for Markov-model based image segmentatio," *Pat. rec.*, vol. 36, no. 1, pp. 131–144, Jan 2003.

[8] S. Chu, J.L. DeRisi, M.B. Eisen, J. Mulholland, D. Botstein, P.O. Brown and I. Herskowitz, " The transcriptional program of sporulation in budding yeast, " *Science*, vol. 282, pp. 699–705, Oct 1998.

[9] M.B. Eisen, P.T. Spellman, P.O. Brown and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proc. Nat. Acad. Sci.*, vol. 95, pp.14863–14868, Dec 1998.

[10] F. Forbes and N. Peyrard, "Hidden Markov random field model selection criteria based on mean field-like approximations," *IEEE Trans. PAMI*, vol. 25, no. 9, pp. 1089–1101, Sep 2003.

[11] D. Hanisch, A. Zien, R. Zimmer and T. Lengauer, "Co-clustering of biological networks and gene expression," *Bioinformatics*, vol. 18 no. Suppl.1, pp. S145–S154, Jul. 2002.

[12] A.J. Hartemink, D.K. Gifford, T.S. Jaakkola and R.A. Young, "Using graphical models and genomic espression data to statistically validate models of genetic regulatory networks," *Proc. Pacific Symp. Biocomputing 7*, pp. 422–433, Jan. 2002.

[13] T.R. Hugues, M.J. Marton, A.R. Jones, C.J. Roberts, R Stoughton, C.D. Armour, H.A. Bennett, E. Coffey, H. Dai, Y.D. He, M.J. Kidd, A.M. King, M.R. Meyer, D. Slade, P.Y. Lum, S.B. Stepaniants, D.D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard and S.H. Friend, "Functional discovery via a compendium of expression profiles," *Cell*, vol. 102, pp. 109–126, Jul. 2000.

[14] M.P. Kurhekar, S. Adak, S. Jhunjhunwala and K. Raghupathy, "Genome-wide pathway analysis and visualization using gene expression data," *Proc. Pacific Symp. Biocomputing 7*, pp. 462–473, Jan 2002.

[15] G. Lanckriet, T. De Bie, N. Christianini, M. I. Jordan and W. Noble, "A statistical framework for genomic data fusion," *Biofinformatics*, vol. 20, no. 16,pp. 2626–2635, Nov. 2004.

[16] E.M. Marcotte, M. Pellegrini, M.J. Thompson, T.O. Yeates and D. Eisemberg, "A combined algorithm for genome-wide prediction of protein function," *Nature*, vol. 402, pp. 83–86, Nov. 1999.

[17] M. Medvedovic, K.Y. Yeung and R.E. Bumgarner, "Bayesian mixture model based clustering of replicated microarray data," *Bioinformatics*, vol. 20, no. 8, pp. 763–774, Apr. 2004.

[18] P. Pavlidis, J. Weston, J. Cai and W. N. Grundy, "Gene functional classification from heterogeneous data," *Proc. Fifth Annual Int. Conf. Comp. Biol.*,, pp. 249–255, Apr. 2001.

[19] E. Segal, H. Wang and D. Koller, "Discovering molecular pathways from protein interaction and gene expression data," *Bioinformatics*, vol. 19, no. Suppl 1, pp. i264–i272, Jul. 2003.

[20] G. Schwarz, "Estimating the dimension of a model," *Ann. Stat.*, vol. 6, no.2, pp. 131–134, Apr. 1978.

[21] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, and T.R. Golub, "Interpreting patterns of gene expression with self-organizing maps : methods and application to hematopoietic differentiation," *Proc. Nat. Acad. Sci.*, vol. 96, no. 6, pp. 2907–2912, Mar. 1999.

[22] A. Tanay, R. Sharan, M. Kupiec and R. Shamir, "Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data," *Proc. Nat. Acad. Sci.*, vol. 101, no. 9, pp. 2981–2986, Mar. 2004.

[23] S. Tavazoie, J. D. Hughes, M.J. Campbell, R.J. Cho and G.M. Church, "Systematic determination of genetic network architecture," *Nature Genetics*, vol. 22, no.3, pp. 281–285., Jul. 1999.

[24] J.-P. Vert and M. Kanehisa, "Graph-driven features extraction from microarray data using diffusion kernels and kernal CCA," *Adv. Neural Inf. Proc. Sys. 15*, pp. 1425–1432, 2003.

[25] Y. Yamanishi, J.-P. Vert, A. Nakaya and M. Kanehisa, "Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical correlation analysis," *Bioinformatics*, vol. 19, no. Suppl 1, pp. i323–i330, Jul. 2003.

[26] K.Y. Yeung, C. Fraley, A. Murua, A. Raftery and L. Ruzzo, "Model-based clustering and data transformations for gene expression data," *Bioinformatics*, vol. 17, no. 10, pp. 977–987, Oct. 2001.

[27] G. Celeux and J. Diebolt, "The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem," *Computational Statistics Quaterly*, vol. 2, pp. 73–82, 1985.

Matthieu Vignes Matthieu Vignes graduated from University Claude Bernard, Lyon, France in 2002. He is now finishing a PhD in Statistics at University Joseph Fourier, Grenoble as part of team Mistis (INRIA Rhône-Alpes). He was recently appointed by BioSS (Biomathematics & Statistics Scotland). His research interests include Graphical models, Missing data, Overlapping clustering and Statistical applications to system biology and compositional analysis of crop plants.

Florence Forbes Florence Forbes received the PhD degree in applied probability in 1996, from University Joseph Fourier, Grenoble, France. She is a research scientist at the Institut National de Recherche en Informatique et Automatique (INRIA). She joined IS2 team at INRIA Rhone-Alpes in 1998 and is now leading research team Mistis. Her research activities include Bayesian image analysis, Mixture models, Markov processes, Markov random fields, and hidden structure models.