

Identifying Graph Clusters using Variational Inference and links to Covariance Parameterisation

BY DAVID BARBER

Department of Computer Science, University College London

Finding clusters of well-connected nodes in a graph is useful in many domains, including Social Network, Web and molecular interaction analyses. From a computational viewpoint, finding these clusters or graph communities is a difficult problem. We consider the framework of Clique Matrices to decompose a graph into a set of possibly overlapping clusters, defined as well-connected subsets of vertices. The decomposition is based on a statistical description which encourages clusters to be well connected and few in number. The formal intractability of inferring the clusters is addressed using a variational approximation which has links to mean-field theories in statistical mechanics. Clique matrices also play a natural role in parameterising positive definite matrices under zero constraints on elements of the matrix. We show that clique matrices can parameterise all positive definite matrices restricted according to a decomposable graph and form a structured Factor Analysis approximation in the non-decomposable case.

Keywords: Mean Field Theory, Community Identification, Network, Variational Inference

1. Introduction

An area of large-scale data analysis concerns the discovery of ‘similar’ objects. The notation of ‘similar’ we are interested in here is that two objects are similar if they are close neighbours on a graph representing the data objects; here the links on the graph represent the network of interactions between the objects. The structure of the connections on these graphs or networks, has been of intense interest recently, particularly concerning the degree structure of the graph and related small world phenomena, see for example[26].

In the field of social-networks each individual is represented as a node(vertex) in a graph, with a link (edge) between two nodes if the individuals are friends. Given a potentially very large such graph our interest is in identifying communities of closely linked friends. A characteristic of such social-networks is that they are sparse since each individual will typically have only a small number of friends relative to the total number of people in the network[22].

The field of Collaborative Filtering also contains related data-analysis challenges. Here nodes may represent products, with a link between two nodes meaning that the two products are frequently both bought by customers. The identification of ‘product groups’ is often of interest[16].

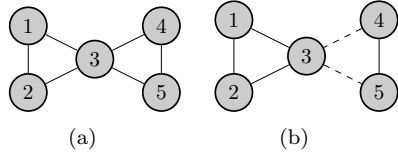


Figure 1. (a) The social network of a set of 5 individuals, represented as an undirected graph. Here individual 3 belongs to the group (1, 2, 3) and also (3, 4, 5). (b) By contrast, in Graph Partitioning, one breaks the graph into roughly equally sized disjoint partitions such that each node is a member of only a single partition, with a minimal number of edges between partitions.

A growing area of application is in bioinformatics in which nodes represent genes, and a link between them representing that the two genes have similar activity profiles. The task is then to identify groups of similarly behaving genes[2].

More specifically, here we use undirected graphs to represent connectivity or adjacency structures in data. For example, in Collaborative Filtering, the nodes(vertices) in the graph may represent products, and a link(edge) between nodes i and j could be used to indicate that customers who buy product i frequently also buy product j . Our interest then is to decompose the graph into well-connected clusters; for example product groups that are commonly co-bought by customers; groups of friends, similarly functioning genes *etc.* Importantly, the same object (product, person, gene) can appear in multiple groups. For example, interpreted as a social network, in figure 1a, individual 3 is a member of his work group (1, 2, 3) and also the poker group (3, 4, 5). These two groups of individuals are otherwise disjoint.

The task of Graph Clustering contrasts with the perhaps more common task of Graph Partitioning in which each node is assigned to only one of a set of subgraphs, figure 1b. Typically the criterion is that each subgraph should be roughly of the same size and there are few connections between the subgraphs[21].

(a) *Mixed Membership models*

A fundamental difference between standard statistical clustering and our interest here is that an object may be a member of more than one group. Such so-called mixed membership models have been developed extensively in recent years, [1, 13]. Such models are particularly useful when the object of interest cannot be easily expressed as a member of a single group. For example, a newspaper article may discuss several topics – characterising the article as belonging to a single topic would then be inaccurate and potentially misleading. Mixed membership models are used in a variety of contexts and are distinguished also by the form of data available. Here our interest is the analysis of a representation of the interaction of a collection of objects; in particular, the data has been processed such that all the information of interest is characterised by a single interaction matrix. Typically these matrices represent two forms of interaction: dyadic and monadic, examples of which are discussed below.

(i) *Dyadic Data*

Consider a collection of documents, which is summarised by an interaction matrix $M_{w,d}$ in which the w, d element of M represents the number of times the w^{th} word in a fixed set of words occurs in document d . For simplicity, let's consider the

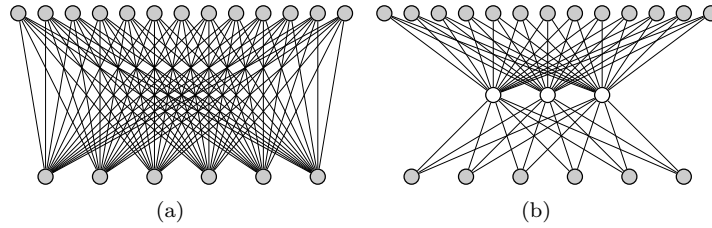


Figure 2. Graphical representation of dyadic data. (a) Here we have say 6 documents and 13 words. A link represents that that particular word-document pair occurs in the dataset. Here all links are shown; in practice, such graphs are typically very sparse. (b) A latent decomposition of (a) using 3 ‘topics’. A topic corresponds to a collection of words, and each document a collection of topics. The open nodes indicate latent variables.

case in which the $M_{w,d}$ is 1 if word w appears in document d and zero otherwise. A graphical depiction of this matrix is a bi-partite graph, as is schematically depicted in figure 2a. The upper nodes represent documents, and the lower nodes words, with a link between them if that word occurs in that document. One might then seek the for assignments of documents to groups or latent ‘topics’ so that one can accurately explain the link structure of the bipartite graph via a small number of latent nodes, as schematically depicted in figure 2b. One may view this as a form of binary matrix factorisation[20, 25]

(ii) *Monadic Data*

In monadic data there is only one type of object and the interaction between the objects representable by a square interaction matrix. Here we also make the assumption also that this similarity is binary. For example one might have a matrix A with elements $A_{i,j} \in \{0,1\}$, with $A_{i,j} = 1$ if proteins i and j can bind to each other. Another example from document analysis might that the matrix has elements $A_{d,d'}$ with $A_{d,d'} = 1$ if documents d and d' are ‘similar’, and 0 otherwise. A graphical depiction of the interaction matrix is given by a graph in which an edge represents an interaction, for example figure 3. Our interest is then to find a decomposition that explains this link structure in a parsimonious way using latent variables. Graphically this means that we seek a bipartite representation of the original graph figure 3 as say figure 4a.

The perhaps more standard concept of statistical clustering is to assign each object to only one of a number of clusters – it cannot be a member of more than one cluster itself. For example, a gene might be assigned to a single gene-cluster such that all genes in a cluster have a similar microarray expression profile[19]. Graphically this would restrict the degree of each shaded node (observed variable) in figure 4a to 1. However, here we do not make any restriction on the degree of the nodes. In this case, for example, a gene may be a member of several regulatory gene networks; a product might belong to many different kinds of product groupings, *etc.*

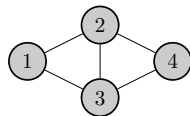


Figure 3. Canonical example used throughout the text. The minimal clique cover is $(1, 2, 3), (2, 3, 4)$.

Our aim here is not to provide an extensive survey of the literature available on mixed membership models, but rather to explain one method in some detail and discuss the computational issues that result. In particular we wish to explain the recent application of techniques originally derived in statistical mechanics to finding approximate solutions to these problems of a more statistical nature.

We shall focus on the monadic case, although the extension to the dyadic case is essentially straightforward. The monadic case is of additional interest since it has natural links to the field of parameterising positive definite matrices, which we shall discuss in Section 5.

(b) *Cliques and Adjacency matrices for Monadic Data*

Our interest is to identify well-connected clusters of nodes in a potentially large graph. A set of nodes which are all connected to each other is called a clique. For example nodes $(1, 2, 3)$ form a clique in figure 1a. A clique is usually defined in the maximal sense such that a clique cannot be contained within a larger clique. For example, $(1, 2)$ in figure 1a would typically not be termed a clique since it is part of a larger completely connected set of nodes, namely $(1, 2, 3)$. In this work, however, we will use the term clique also to refer to non-maximal cliques.

We can equivalently describe an undirected graph using an adjacency matrix. The symmetric adjacency matrix has elements $A_{ij} \in \{0, 1\}$, with a 1 indicating a link between nodes i and j . For the graph in figure 3, the adjacency matrix is

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} \quad (1.1)$$

where we include self connections on the diagonal. Given a graph G with adjacency matrix A , our aim is to find a ‘simpler’ description of A that reveals underlying cluster structure.

Computational Difficulty

A formal specification of the problem of finding a minimum number of maximal fully-connected subsets is the computational problem `MIN CLIQUE COVER`[15, 32]. This is a computationally hard problem and approximations are therefore unavoidable, in general. One may view this formal requirement that all nodes be connected to be somewhat severe. In some applications, provided that only a small number of links in an ‘almost clique’ are missing, this may be considered a sufficiently well-connected group of nodes to form a cluster. We therefore relax the hard constraints of `MIN CLIQUE COVER` and develop a statistical technique to reveal clusters of ‘well-connected’ nodes and to identify the smallest number of such clusters. The resulting problem is still formally computationally intractable, and requires the development of techniques to yield an efficient numerical approximation.

To phrase the our clustering requirement more precisely we will use the *clique matrix* formalism, a generalisation of the incidence matrix. We apply this to the clustering problem, in addition to demonstrating an application in constrained covariance parameterisation in Section 5.

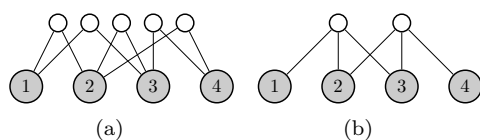


Figure 4. Bipartite representations of the decompositions of figure 3. Shaded nodes represent observed variables, and open nodes latent variables. (a) Incidence, (b) Minimal Clique decomposition.

2. Clique Decompositions

Given the undirected graph in figure 3, the incidence matrix F_{inc} is an alternative description of the adjacency structure[10]. Given the V nodes in the graph, we construct F_{inc} as follows: For each link $i \sim j$ in the graph, form a column of the matrix F_{inc} with zero entries except for a 1 in the i^{th} and j^{th} row. The column ordering is arbitrary. For example, for the graph in figure 3 an incidence matrix is

$$F_{inc} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

The incidence matrix has an interesting property, namely that the adjacency structure of the original graph is related by the product of the incidence matrix with itself. The diagonal entries contain the degree (number of links) of each node. For our example, this gives†

$$F_{inc}F_{inc}^T = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 1 & 3 & 1 & 1 \\ 1 & 1 & 3 & 1 \\ 0 & 1 & 1 & 2 \end{pmatrix}$$

so that the outer-product with itself satisfies

$$A = H(F_{inc}F_{inc}^T) \quad (2.1)$$

Here $H(\cdot)$ is the element-wise Heaviside step function, so that $[H(M)]_{ij} = 1$ if $M_{ij} > 0$ and is 0 otherwise.

A useful viewpoint of the incidence matrix is that it identifies two-cliques in the graph (here we are using the term ‘clique’ in the non-maximal sense). There are five 2-cliques in figure 3, and each column of F_{inc} specifies which elements are in each 2-clique. Graphically we can depict this incidence decomposition as a bipartite graph, as in figure 4a where the open nodes presents the five 2-cliques.

The incidence matrix can be generalised to describe larger cliques. Consider the following matrix as a decomposition for figure 3, and its outer-product:

$$F = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad FF^T = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} \quad (2.2)$$

† $(\cdot)^T$ represents matrix transpose.

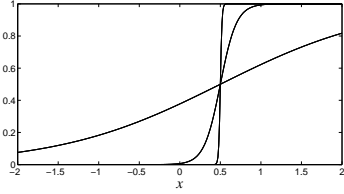


Figure 5. The function $\sigma(x) \equiv \left(1 + e^{\beta(0.5-x)}\right)^{-1}$ for $\beta = 1, 10, 100$. As β increases, this sigmoid function tends to a step function.

The interpretation is that F represents a decomposition into two 3-cliques. As in the incidence matrix, each column represents a clique, and the rows containing a ‘1’ express which elements are in the clique defined by that column. This decomposition can be represented as the bipartite graph of figure 4b. For the graph of figure 3, both F_{inc} and F satisfy

$$A = H(FF^T) = H(F_{inc}F_{inc}^T) \quad (2.3)$$

One can view equation (2.3) as a form of binary matrix factorisation of the binary square (symmetric) matrix A into non-square binary matrices. For our clustering purposes, the decomposition using factor F is to be preferred to the incidence decomposition since F decomposes the graph into a smaller number of larger cliques. Indeed, F solves MIN CLIQUE COVER for figure 1b.

(a) Clique matrices

More generally, given an adjacency matrix $[A]_{ij}, i, j = 1, \dots, V$ ($A_{ii} = 1$), we define a clique matrix F to have elements $F_{i,c} \in \{0, 1\}, i = 1, \dots, V, c = 1, \dots, C$ such that $A = H(FF^T)$. (The formal definition of a clique matrix is that the cliques must be maximal[17]. Here we used a relaxed definition in which cliques are not required to be maximal). The interpretation of the elements of FF^T is that diagonal elements $[FF^T]_{ii}$ express the number of cliques/columns that vertex i occurs in. Off-diagonal elements $[FF^T]_{ij}$ contain the number of cliques/columns that vertices i and j jointly inhabit, [5].

Whilst finding a clique decomposition F is easy (use the incidence matrix for example), finding a clique decomposition with the minimal number of columns, *i.e.* solving MIN CLIQUE COVER, is NP-Hard[15, 4]. One approach would be to use a recursive procedure that searches for maximal cliques in the graph or related techniques based on finding large densely connected subgraphs[32]. The route that we take here is different and motivated by the idea that perfect clique decomposition is not necessarily desirable if the aim is only to find well-connected clusters in G .

(b) Statistical Clique Decompositions

To find ‘well-connected’ clusters, we relax the constraint that the decomposition is in the form of cliques in the original graph. Our approach is to view the absence of links as statistical fluctuations away from a perfect clique. Given a $V \times C$ matrix F , we desire that the higher the overlap between rows[†] f_i and f_j is, the greater the probability of a link between i and j . This may be achieved using, for example,

$$p(i \sim j|F) = \sigma(f_i f_j^T) \quad (2.4)$$

[†] We use lower indices f_i to denote the the i^{th} row of F .

where \sim denotes that i and j are linked, and

$$\sigma(x) \equiv \left(1 + e^{\beta(0.5-x)}\right)^{-1} \quad (2.5)$$

and β controls the steepness of the function, see figure 5. The 0.5 shift in equation (2.5) ensures that σ approximates the step-function since the argument of σ is an integer. Under equation (2.4), if f_i and f_j have at least one '1' in the same position, $f_i f_j^T - 0.5 > 0$ and $p(i \sim j)$ is high. Absent links contribute $p(i \not\sim j|F) = 1 - p(i \sim j|F)$. β controls how strictly $\sigma(F F^T)$ matches A ; for large β , very little flexibility is allowed and only cliques will be identified. For small β , subsets that would be cliques if it were not for a small number of missing links, are clustered together. The setting of β is user and problem dependent.

Given F , and assuming each element of the adjacency matrix is sampled independently from the generating process, the joint probability of observing A is (neglecting its diagonal elements),

$$p(A|F) = \prod_{i \sim j} \sigma(f_i f_j^T) \prod_{i \not\sim j} (1 - \sigma(f_i f_j^T)) \quad (2.6)$$

The ultimate quantity of interest is the posterior distribution of clique structure, given the known adjacency structure which, according to Bayes' rule is given by,

$$p(F|A) \propto p(A|F)p(F) \quad (2.7)$$

where $p(F)$ is a prior over clique matrices. Later we place a prior on F to encourage the smallest number of clusters to be identified (and hence for the size of the clusters to be large). Even in the case of a fixed desired number of clusters, determining the most likely clique matrix F is hard, we develop an algorithm to approximately discover clique matrices, before discussing non-uniform priors $p(F)$.

3. Approximate Inference in Graphical Models

Graphs have a long history in the sciences, often used to represent physical interactions between objects. They also have a long history in statistics and form the backbone of Graphical Models, a marriage between Graph and Probability theory[24]. A challenge to which considerable research effort has been directed is the development of approximate inference techniques on the probability distributions represented by these Graphical Models. More recently, the field of Machine Learning has been active in interfacing techniques developed initially within the statistical physics community with Statistics and large scale data analysis.

Our aim here is to outline some of these developments which we shall make use of in deriving an approximate inference algorithm for the Clique Decomposition problem. To keep the exposition relatively straightforward, we'll discuss a simpler distribution than that of our Clique Decomposition problem, before applying a similar methodology in Section 4.

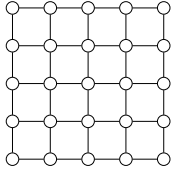


Figure 6. A square lattice pairwise Markov Random field on a set of variables x_1, \dots, x_{25} , representing a distribution of the form $\prod_{i \sim j} \phi_{ij}(x_i, x_j)$. In statistical physics such lattice models include the Ising model on binary ‘spin’ variables $x_i \in \{+1, -1\}$ with $\phi_{ij}(x_i, x_j) = e^{w_{ij}x_i x_j}$. Additional singleton terms x_i in the exponent are ‘external field’ terms.

(a) *Ising Model*

A canonical example from statistical physics is the Ising model, which corresponds to a distribution on a set of N binary variables $x_i \in \{+1, -1\}$,

$$p(x) = \frac{1}{Z} e^{\sum_{i,j} w_{ij} x_i x_j + \sum_i b_i x_i} \quad (3.1)$$

Here Z is a constant that ensure normalisation of the distribution,

$$Z = \sum_x e^{\sum_{i,j} w_{ij} x_i x_j + \sum_i b_i x_i}$$

In statistical physics this is called the partition function. If we consider

$$\frac{\partial}{\partial b_i} \log Z = \frac{1}{Z} \sum_x x_i e^{\sum_{i,j} w_{ij} x_i x_j + \sum_i b_i x_i} = \langle x_i \rangle_p$$

where angled brackets $\langle f(x) \rangle_p$ denote expectation of the function $f(x)$ with respect to the distribution $p(x)$. The log partition function is therefore closely related to a cumulant generating function. In Graphical Models, the distribution (3.1) is called a pairwise Markov Random Field (MRF) and its graphical depiction depends on the interaction matrix w , figure 6.

Two celebrated results are that for planar Ising model with pure interactions ($b = 0$), the partition function can be computed in polynomial time[14] and that the most likely joint configuration for the pure interaction case can also be computed in polynomial time[18]. However, for the case of non-planar interactions w and/or the inclusion of ‘external fields’, $b \neq 0$, no polynomial time algorithm is known for computing either the partition function or the most likely joint configuration. For this reason, many different techniques have been proposed to approximate partition functions and moments in such intractable cases. Here we will discuss one that is particularly beneficial to our purposes since it closely relates to techniques in approximate Bayesian inference.

(b) *Variational Inference for the Ising Model*

A useful definition is the Kullback-Leibler divergence

$$KL(q|p) = \langle \log q \rangle_q - \langle \log p \rangle_q$$

which is a measure of the dissimilarity of the distributions $p(x)$ and $q(x)$. It is straightforward to show that $KL(q|p) \geq 0$ and is zero if and only if the distribution p and q are identical. Note that the KL divergence is not symmetric in its arguments. For the MRF $p(x)$, the KL divergence $KL(q|p)$ gives the bound

$$\langle \log q \rangle_q - \sum_{ij} w_{ij} \langle x_i x_j \rangle_q - \sum_i b_i \langle x_i \rangle_q + \log Z \geq 0$$

Rewriting, this gives the celebrated ‘free energy’ bound

$$\log Z \geq \underbrace{-\langle \log q \rangle_q}_{\text{entropy}} + \underbrace{\sum_{ij} w_{ij} \langle x_i x_j \rangle_q + \sum_i b_i \langle x_i \rangle_q}_{\text{energy}}$$

The bound saturates when $q \equiv p$. However, this is clearly of little help, since we cannot compute the averages of variables $\langle x_i x_j \rangle_p, \langle x_i \rangle_p$. The idea of a variational method is to assume a simpler ‘tractable’ distribution q .

The simplest assumption is the fully factorised distribution

$$q(x) = \prod_i q_i(x_i)$$

which corresponds to Naive Mean Field theory, the Graphical Model for which is the set of disconnected nodes. In this case

$$\log Z \geq -\sum_i \langle \log q_i \rangle_{q_i} + \sum_{ij} w_{ij} \langle x_i x_j \rangle_{q(x_i, x_j)} + \sum_i b_i \langle x_i \rangle_{q(x_i)}$$

For a factorised distribution and bearing in mind that $x_i \in \{+1, -1\}$,

$$\langle x_i x_j \rangle = \begin{cases} 1 & i = j \\ \langle x_i \rangle \langle x_j \rangle & i \neq j \end{cases}$$

For a binary variable, one may use the convenient parametrization

$$q_i(x_i = 1) = \frac{e^{\theta_i}}{e^{\theta_i} + e^{-\theta_i}} \quad (3.2)$$

so that

$$\langle x_i \rangle_{q_i} = +1 \times q(x_i = 1) - 1 \times q(x_i = -1) = \tanh(\theta_i)$$

Expressed in terms of the mean-field parameters θ , the bound on the log partition function is

$$\log Z \geq \mathcal{B}(\theta) \equiv \sum_i H(\theta_i) + \sum_{i \neq j} w_{ij} \tanh(\theta_i) \tanh(\theta_j) + \sum_i b_i \tanh(\theta_i) + \sum_i w_{ii}$$

where $H(\theta_i)$ is the binary entropy of a distribution parameterised according to equation (3.2). Finding the best factorised approximation in the minimal Kullback-Liebler divergence sense then corresponds to maximising the bound on $\log Z$ with respect to the variational parameters θ . However, the variational bound $\mathcal{B}(\theta)$ is non-convex with respect to θ so that finding the optimal θ is typically formally a computationally hard problem. (Indeed, there are many situations where it is known that a factorised approximation of a distribution is accurate – the difficulty is in finding it). At first sight it seems that we have simply replaced a computationally hard summation (computing $\log Z$) by an equally hard maximisation (optimising $\mathcal{B}(\theta)$). Indeed, the ‘graphical structure’ of this optimisation problem matches exactly that of the original MRF. Whilst this is undeniably correct, the hope is that

by transforming a difficult discrete summation into a continuous optimisation problem, we will be able to bring to the table the techniques of numerical optimisation to find a good approximation, exploiting any smoothness properties of the bound function to rapidly find local optima.

A particularly simple optimisation technique is to differentiate the bound and equate to zero. Straightforward algebra leads to the so-called Mean Field equations

$$\theta_i = b_i + \sum_{j \neq i} w_{ij} \tanh(\theta_j)$$

Iterating these equations for each i in sequence (asynchronous updating) is guaranteed to increase the bound on the partition function and lead to a local minima of the KL divergence. Once a converged solution has been identified, we have a bound on the log partition function and an approximation of the intractable p in terms of a factorised distribution q . In particular the average of a single variable can be computed using the approximation

$$\langle x_i \rangle_p \approx \langle x_i \rangle_q = \tanh(\theta_i)$$

Approximation Validity

When might one expect such a naive fully factorised approximation to work well? Clearly, if w_{ij} is very small for $i \neq j$, the distribution p will be effectively a factorised distribution. However, a more interesting case is when there are many neighbours in a graph. In this case, it is useful to write

$$p(x) = \frac{1}{Z} e^{\sum_{ij} w_{ij} x_i x_j} = \frac{1}{Z} e^{N \sum_i x_i \frac{1}{N} \sum_j w_{ij} x_j}$$

If we assume that $p(x)$ is approximately factorised then each of the terms x_j in $w_{ij} x_j$ is independent. Provided that the w_{ij} are not strongly correlated and $O(1)$ the conditions of validity of the Central Limit theorem hold, and $z \equiv \frac{1}{N} \sum_j w_{ij} x_j$ will tend to be Gaussian distributed with mean $\langle z \rangle = \sum_j w_{ij} \langle x_j \rangle$ and variance $O(1/N)$. As N increases the fluctuations around the mean therefore rapidly diminish, which means we may write

$$p(x) \approx \frac{1}{Z} e^{N \sum_i x_i \langle z_i \rangle} \approx \prod_i p(x_i)$$

We have shown therefore that the assumption that p is approximately factorised is self-consistent in the limit of a large systems with a large number of neighbours (densely connected system). Indeed, this is the origin of the term ‘mean-field’ approximation since the effect of the neighbouring interactions can be replaced by a single scalar mean interaction; since this occurs for each site i , one therefore has a field of such mean contributions.

More generally one can use structured Mean Field approximations. Those for which averages of the variables can be computed in linear time include spanning trees, and decomposable graphs, [7] – see Appendix Appendix C. More generally one can exploit any structural approximation with arbitrary hypertree width q by use of the Junction Tree algorithm in combination with the KL divergence, although the computational expense increases exponentially with the hypertree width [35].

There are many more recent advanced in approximate inference in Graphical Models. However, the variational technique based on the Kullback-Leibler divergence is particularly convenient for two reasons (i) in the clique decomposition case the distribution is densely connected, and (ii) variational inference also leads to a natural way to deal with learning parameters, as we shall discuss in Section 4.

4. Clique Decomposition using Variational Inference

Formally, our task is to find the Most likely A Posteriori (MAP) solution

$$\arg \max_F p(F|A)$$

corresponding to equation (2.7), where F is a $V \times C$ binary matrix. Initially, we shall assume a ‘flat prior’ $p(F) = \text{const.}$, so that the most likely solution corresponds to finding that F that maximises

$$p(A|F) = \prod_{i \sim j} \sigma(f_i f_j^T) \prod_{i \not\sim j} (1 - \sigma(f_i f_j^T))$$

A variety of deterministic and randomised methods could be brought to bear on this problem. The approach we take here is to approximate the marginal posterior $p(f_{ij}|A)$ and then to assign each f_{ij} to that state which maximises this posterior marginal (MPM). This has the advantage of being closely related to marginal likelihood computations, which will prove useful later for addressing the issue of finding the number of clusters. The technique is analogous to the Naive Mean Field theory for the Ising Model, and the details are given in Appendix Appendix A, together with the corresponding Mean Field equations which describe the iterative procedure for finding the approximate MPM solution.

Finding the number of clusters

To bias the contributions to the adjacency matrix A to occur from a small number of columns of F , we first reparameterize F as

$$F = (\alpha_1 f^1, \dots, \alpha_{C_{max}} f^{C_{max}}) \quad (4.1)$$

where $\alpha_c \in \{0, 1\}$ play the role of indicators and f^c is the vector of column c of F . C_{max} is an assumed maximal number of clusters we desire to find. Ideally, we would like to find a likely solution F with a low number of indicators $\alpha_1, \dots, \alpha_{C_{max}}$ in state 1. To achieve this we define a prior distribution on the binary hypercube $\alpha = (\alpha_1, \dots, \alpha_{C_{max}})$,

$$p(\alpha|\nu) = \prod_c \nu^{I[\alpha_c=1]} (1 - \nu)^{I[\alpha_c=0]} \quad (4.2)$$

where we use the indicator function

$$I[x = y] = \begin{cases} 1 & x = y \\ 0 & x \neq y \end{cases}$$

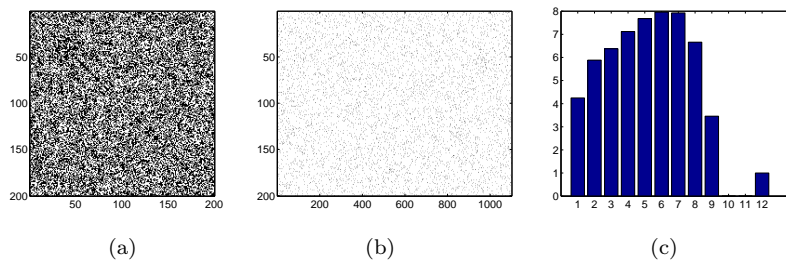


Figure 7. (a) Adjacency matrix for the DIMACS **brock200-2 MAX-CLIQUE** challenge. Black denotes the presence of a link. (b) Clique Matrix. (c) Log₂-histogram of clique occurrence (+1); correctly solves **MAX-CLIQUE** (12) as well as identifying all remaining clusters.

To encourage a small number of α 's to be used, we use a Beta prior $p(\nu)$. This gives rise to a Beta-Bernoulli distribution

$$p(\alpha) = \int_{\nu} p(\alpha|\nu)p(\nu) = \frac{B(a + N, b + C_{max} - N)}{B(a, b)} \quad (4.3)$$

where $B(a, b)$ is the normalisation constant of the beta-distribution and

$$N \equiv \sum_{c=1}^{C_{max}} I[\alpha_c = 1]$$

namely the number of indicators in state 1. To encourage strongly that a small number of components should be active, we set $a = 1, b = 3$. The geometric picture is of a distribution on the vertices of the binary hypercube $\{0, 1\}_{max}^C$ with a bias towards vertices close to the origin $(0, \dots, 0)$. Through equation (4.1), the prior on α induces a prior on F . The resulting distribution $p(F, \alpha|A) \propto p(F|\alpha)p(\alpha)$ is formally intractable and needs to be dealt with in an approximate manner.

Variational Bayes

To deal with the intractable joint posterior we adopt a similar strategy to the fixed C case and employ a variational procedure to seek a factorised approximation $p(\alpha, F|A) \approx q(\alpha)q(F)$ based on minimising

$$KL(q(\alpha)q(F)|p(\alpha, F|A)) \quad (4.4)$$

This is analogous to a form of Mean-Field theory, and results in a set of alternating mean-field equation updates for $q(\alpha)$ and $q(F)$. The details are given in Appendix (Appendix B).

(a) Demonstrations

(i) DIMACS MAX-CLIQUE

In figure 7a we show the adjacency matrix for a 200 vertex graph, taken from the DIMACS 1996 **MAX CLIQUE** challenge [8]. This graph was constructed by the challenge coordinators to hide the largest clique in the graph and evade discovery

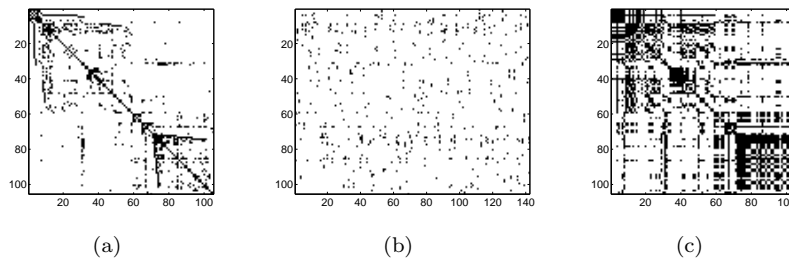


Figure 8. (a) Adjacency matrix of 105 Political Books (black=1). (b) Clique matrix: 521 non-zero entries. (c) Adjacency reconstruction using an Approximate Clique Matrix with 10 cliques – see also figure 9.

based on the recursive algorithms available at that time. Whilst more recent algorithms have been constructed that readily find the largest clique in this graph [28], this problem serves as an interesting baseline to see if our algorithm, in searching for a complete decomposition, also solves MAX-CLIQUE for this graph.

Our method aims to find a complete characterisation of an undirected graph into constituent clusters. By setting β suitably high ($\beta = 10$ in the experiments), we impose that perfect cliques constitute clusters. Running our Mean-Field algorithm with $C_{max} = 2000$ results in a clique-decomposition, figure 7b, containing 1102 cliques[†]. In figure 7c we plot a log histogram of the cluster sizes, indicating that there is only a single largest clique of size 12. The largest clique in the graph is indeed 12[8].

(ii) Political Books Clustering

The data consists of 105 books on US politics sold by the online bookseller Amazon. Edges in graph G , figure 8a, represent frequent co-purchasing of books by the same buyers, as indicated by the ‘customers who bought this book also bought these other books’ feature on Amazon[23]. Additionally, books are labelled ‘liberal’, ‘neutral’, or ‘conservative’ according to the judgement of a politically astute reader[‡]. The interest is to assign books to clusters, depending on the graph G alone, and then see if these clusters correspond in some way to the ascribed political leanings of each book. Note that the information here is minimal – all that is known to the clustering algorithm is which books were co-bought (matrix A); no other information on the content or title of the books are exploited by the algorithm.

Running our algorithm with an initial $C_{max} = 200$ cliques, the posterior contains 142 cliques[¶], figure 8b, giving a perfect reconstruction of the adjacency A . For comparison, the incidence matrix has 441 2-cliques.

To cluster the data more aggressively, we fix $C = 10$ and run our fixed- C algorithm. As expected, this results only in an approximate clique decomposition, $A \approx H(FF^T)$, as plotted in figure 8c. The resulting 105×10 approximate clique matrix is plotted in figure 9 and demonstrates how individual books are present in more than one cluster. For visualisation purposes, we plot the clique matrix in 3

[†] This takes roughly 30s using a 1Ghz machine.

[‡] See www-personal.umich.edu/~mejn/netdata/.

[¶] This take roughly 10s on a 1GHz machine.

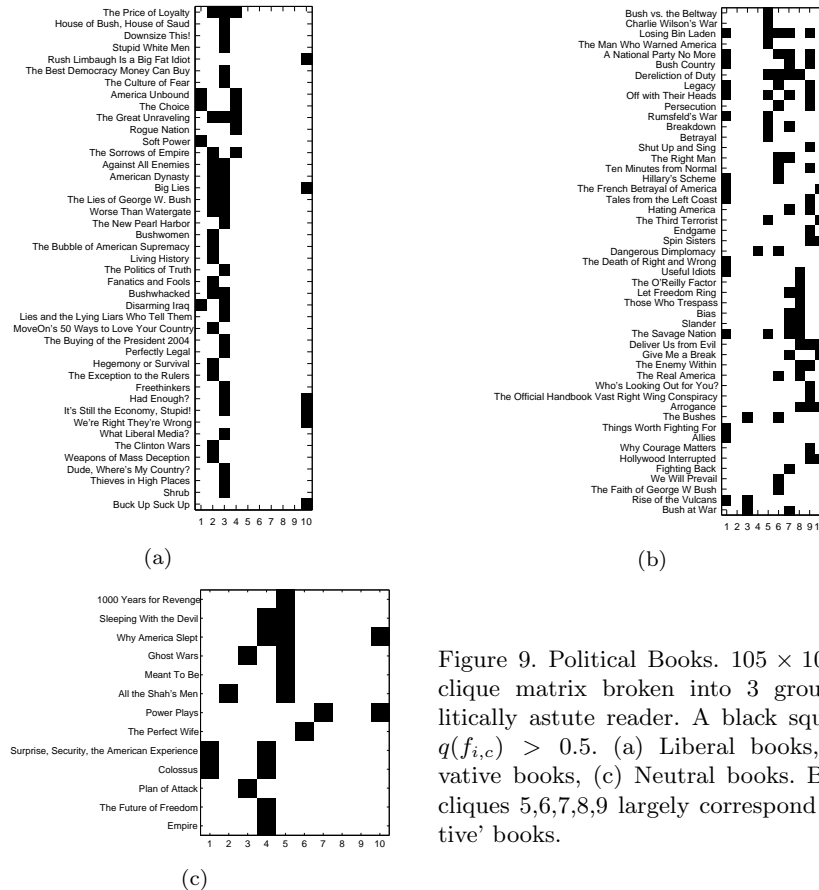


Figure 9. Political Books. 105×10 dimensional clique matrix broken into 3 groups by a politically astute reader. A black square indicates $q(f_{i,c}) > 0.5$. (a) Liberal books, (b) Conservative books, (c) Neutral books. By inspection, cliques 5,6,7,8,9 largely correspond to ‘conservative’ books.

parts, where each part corresponds to the political leaning of the book, according to an independent reader. Interestingly, the clusters found only on the basis of the adjacency matrix have some correspondence with the ascribed political leanings of each book since one can see that cliques 5, 6, 7, 8, 9 correspond to largely ‘conservative’ books. Most books belong to more than a single clique/cluster, suggesting that they are not single topic books.

5. Latent Parameterisations for Zero-Constrained Positive Matrices

We turn now to what may at first seem an unrelated issue : parameterising positive definite matrices. As is well-known, any positive definite matrix K can be parameterised using a Cholesky factor,

$$K = CC^T$$

where the Cholesky factor C is a lower triangular real matrix.

Recently, interest is growing in relational machine learning in which constraints are imposed on the dependence between objects. In the simplest case of modelling

the interaction between objects using a Gaussian distribution this corresponds to imposing that certain elements of a covariance matrix (or possibly its inverse) must be zero. Our interest here then is the parameterisation of covariance matrices Σ with a set of given zeros for the elements of Σ .

We may use an undirected graph G to represent zero constraints on a positive definite matrix K . In particular, missing edges in G with corresponding adjacency matrix elements $A_{ij} = 0$, correspond to zero entries $K_{ij} = 0$. We denote the space of positive definite matrices constrained through G by $M^+(G)$.

Parameterisation using Clique Matrices

One approach to parameterising K based on given zero restrictions represented by A is to begin with a clique decomposition of A ,

$$A = H(FF^\top)$$

By construction the matrix FF^\top is positive semi-definite and has zeros where A has zeros and integer values where A has 1's. Hence by replacing non-zero entries of a clique matrix F with arbitrary real values, $F \rightarrow F^*$, the matrix $F^*(F^*)^\top$ is also positive semi-definite and has the same zero-structure as A . This therefore immediately gives a naive parameterisation of the class of covariances matrices which have zeros specified according to G . An immediate question is the richness of such a parameterisation – can all of $M^+(G)$ be reached in this way?

(a) Decomposable Case

For G decomposable, parameterising $M^+(G)$ is straightforward[29, 27, 34]. For decomposable G , provided the vertices are perfect elimination ordered (when eliminated in the sequence, no additional links in the subgraphs are introduced – see Appendix (Appendix C)), the Cholesky factor has the same structure as G [34]. In other words, provided the vertices are ordered correctly, the lower triangular part of the adjacency matrix is a clique matrix and furthermore parameterises all of $M^+(G)$. All positive definite matrices under decomposable zero-constraints can therefore be parameterised by some clique matrix. Below we describe how a clique matrix can be derived that guarantees all of $M^+(G)$ can be reached for decomposable G .

Expanded Clique Matrix

Given a Clique Matrix $F \in \{0, 1\}^{V \times C}$, the Expanded Clique matrix consists of F appended with columns corresponding to all unique sub-columns of F . A subcolumn of f^c is defined by replacing one or more entries containing $f_i^c = 1$ by $f_i^c = 0$.

Furthermore, a Clique Matrix $F \in \{0, 1\}^{V \times C}$ is *minimal* for A if there exists no other clique matrix for $F \in \{0, 1\}^{V \times C'}$ with a smaller number of columns $C' < C$. The expanded Clique Matrix corresponding to the minimal clique matrix derived

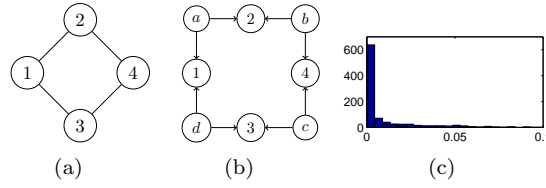


Figure 10. (a) Non-decomposable graph. (b) Correlations can be induced via latent variables. (c) Histogram of the rms errors in approximating covariances according to graph (a) with an expanded incidence matrix.

from figure 1b is

$$\begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.1)$$

In the above, the expansion is ordered such that all 3-cliques are enumerated, then all 2-cliques and finally all 1-cliques.

Starting from a minimal clique matrix for a decomposable graph, the expansion of this minimal clique matrix must contain all the columns of the Cholesky factor T^T . For the example for G in figure 1b, the lower triangular Cholesky factor is[†]

$$\begin{pmatrix} * & 0 & 0 & 0 \\ * & * & 0 & 0 \\ * & * & * & 0 \\ 0 & * & * & * \end{pmatrix}$$

which corresponds to columns 1, 2, 7, 11 of the expanded clique matrix, equation (5.1). In a similar way, any expanded minimal clique matrix will always contain the columns patterns of the Cholesky factor of a decomposable G . However, clearly, in general, the expanded clique matrix is an over-parameterisation of $M^+(G)$.

(b) Non-decomposable Case

For G non-decomposable, no explicit parameterisation is generally possible and techniques based on Positive Definite matrix completion are required[27, 33, 9, 29]. For the specific example in figure 10a, the lower Cholesky factor has the form

$$\begin{pmatrix} c_{11} & 0 & 0 & 0 \\ c_{21} & c_{22} & 0 & 0 \\ c_{31} & c_{32} & c_{33} & 0 \\ 0 & c_{42} & c_{43} & c_{44} \end{pmatrix}, \text{ with } c_{21}c_{31} + c_{22}c_{32} = 0 \quad (5.2)$$

which can be found explicitly in this case. However, more generally, for non-decomposable graphs, one cannot explicitly identify those elements of the Cholesky factor which may be set to zero[29, 34].

[†] In general, for a matrix with elements $d_{ij} \in \{0, 1\}$, we use D^* to denote a matrix with $d_{ij}^* = 0$ if $d_{ij} = 0$, and arbitrary values elsewhere.

An alternative is to use latent variables to explicitly parameterise $M^+(G)$. One may use Factor Analysis

$$x = F\epsilon, \quad \epsilon \sim \mathcal{N}(0, I) \quad \Rightarrow \quad \Sigma = FF^T$$

where the factor matrix F is suitably structured in order to force zeros in specific elements of Σ^\dagger .

A special case of the above is to use a latent variable to induce correlation between x_1 and x_2 via a local Directed Graph element $x_1 \leftarrow \epsilon_{12} \rightarrow x_2$. For each edge in G , a corresponding latent ϵ can thus be introduced to form correlations between all pairs of variables, without introducing correlations on missing edges in G [12]. By taking $F = [F_{inc}^* | I^*]$, it is clear that this ‘ancillary variable’ approach (see, for example, [31]) is reproduced and is a special case of restricting Cliques to Incidence matrices.

To show that not all of $M^+(G)$ can be reached by clique matrices, consider figure 10a. In this particularly simple case, the minimal clique matrix is the same as the incidence matrix, and the expanded clique matrix is simply the incidence matrix with the identity matrix appended. In this case, therefore, the expanded clique matrix contains columns with only two non-zero entries. However, the Cholesky factor equation (5.2) contains columns with 3 non-zero entries, so that there is no immediate assignment of $[F_{inc}^* | I^*]$ which will match the Cholesky factor.



For the non-decomposable graph the minimal clique matrix contains 3-cliques so that its expansion contains columns that an expansion based on an incidence matrix would not. In this case our approximate parameterisation is therefore richer than would be obtained from simply introducing a latent auxiliary variable for each edge of the graph[12, 30].

(c) *Maximum Likelihood Solution*

In fitting a Gaussian $\mathcal{N}(0, \Sigma)$ to zero mean data, with sample covariance S , the ML solution minimises

$$\kappa(\Sigma) \equiv \text{Tr}(\Sigma^{-1}S) + \log \det \Sigma \tag{5.3}$$

Our interest is to minimize κ subject to zero constraints on Σ specified through G , with $\sigma_{ij} = 0$ if $A_{ij} = 0$. For G decomposable, the problem is essentially trivial, since $M^+(G)$ is easily characterized via a structured Cholesky factor, $\Sigma \equiv C^T(\theta)C(\theta)$ see for example [29], for which one can parameterise equation (5.3) using $\kappa(\theta)$ and perform unconstrained minimisation over the free parameters θ of the Cholesky factor.

In the non-decomposable G case, no explicit parameterisation of $M^+(G)$ is feasible. A common approach in this case is to recognise that solutions to this satisfy $[\Sigma^{-1}]_{ij} = [\Sigma^{-1}S\Sigma^{-1}]_{ij}$ for $A_{ij} = 1$ and $\sigma_{ij} = 0$ otherwise[3] and define iterative procedures to solve this equation[11]. Alternatively, Positive Definite Completion

† By writing $F = [\tilde{F}|D]$ where D is diagonal, this is explicitly Factor Analysis. Unlike standard FA, the matrix \tilde{F} will typically be non-square and sparse.

methods may be used to parameterise $M^+(G)$. Our approach uses the parameterisation $\Sigma = F^*(F^*)^\top$ where F should be chosen as large as can be computationally afforded. F can be determined by running the algorithm of Appendix Appendix A. Although for non-decomposable G , not all of $M^+(G)$ is guaranteed reachable through this parameterisation, one may expect that a large fraction of $M^+(G)$ is within reach. A benefit of this approach is that one may then minimize equation (5.3) with respect to the free parameters of F^* using any standard optimisation technique, and convergence is guaranteed. Since our parameterisation has a natural latent variable representation (it is a form of structured Factor Analysis), EM and Bayesian techniques can also be used in this case. A numerical example is plotted in figure 10c. We take the 4×8 expanded clique matrix corresponding to figure 10a and minimise equation (5.3) with respect to the non-zero entries of the clique matrix[‡]. Each sample matrix S is generated randomly by drawing values of the Cholesky factor equation (5.2) independently from a zero mean unit variance Gaussian. In figure 10c we plot the root mean square error between the learned Σ and sample covariance S , averaged over all non-zero components of Σ . The histogram of the error, computed from 1000 simulations, shows that whilst a few have appreciable error, the vast majority of cases are numerically well approximated by the expanded clique matrix technique, even though the graph G is non-decomposable.

6. Summary

We introduced a graph matrix decomposition based on an extension of the incidence matrix concept. Finding the clique decomposition corresponding to the smallest number of cliques is a hard problem, and we considered a relaxed version of the problem to find an approximate clique decomposition based on a variational algorithm. The approach can be seen as a form of binary factorisation of the adjacency matrix (a parallel development to our own work is [25], which considers binary factorisation of more general matrices). An application of clique matrices is to parameterising positive definite matrices under specified zero constraints. We showed that constraints corresponding to decomposable graphs trivially admit a clique matrix representation, and how our structured Factor Analysis technique can be used to approximate the non-decomposable case. This is a richer parameterisation than those latent models which consider only pairwise correlations in forming the latent model. Indeed, the so-called ancillary variable technique is a special case of using incidence, as opposed to clique matrices. The latent variable formulation additionally offers an alternative to recent works on conjugate priors for constrained covariances in Bayesian learning.

C-code for clique matrices is available from the author.

[‡] We chose this simple case since the exact parameterisation of all $M^+(G)$ is easy to write down. Whilst here the expanded clique and incidence matrices are equivalent, the reader should bear in mind that in more complex situations, the expansion based on a clique matrix provides a richer parameterisation than that of the incidence matrix.

Appendix A. Mean Field Approximation

Given the intractable $p(F|A) \propto p(A|F)$, a fully factorised mean-field approximation (see, *e.g.* [35])

$$q(F) = \prod_{i=1}^V \prod_{c=1}^C q(f_{i,c}) \quad (\text{A } 1)$$

can be found by minimising the KL divergence

$$KL(q, p) = \langle \log q \rangle_q - \langle \log p \rangle_q \quad (\text{A } 2)$$

where $\langle \cdot \rangle_q$ represents expectation with respect to q . The first ‘entropic’ term simply decomposes into $\sum_{i,c} \langle \log q(z_{i,c}) \rangle$. The second, ‘energy’ term, up to a constant is

$$\sum_{i \sim j} \left\langle \log \sigma \left(\sum_c f_{ic} f_{j,c} \right) \right\rangle_q + \sum_{i \not\sim j} \left\langle \log \left(1 - \sigma \left(\sum_c f_{ic} f_{j,c} \right) \right) \right\rangle_q \quad (\text{A } 3)$$

The first term of equation (A 3) encourages graph links to be preserved under the decomposition, and is given by

$$\sum_{i \sim j} \left\langle f \left(\sum_{d=1}^C f_{id} f_{jd} \right) \right\rangle_{\prod_{e=1}^C q(f_{ie}) q(f_{je})} \quad (\text{A } 4)$$

where $f(x) \equiv \log \sigma(x)$. Minimising equation (A 2) can be achieved by differentiation. Differentiating the energy contribution from the present links, equation (A 4) with respect to $q(f_{kc})$ we identify two cases: when $i = k$ and when $j = k$. Due to symmetry, the derivative is

$$2 \sum_{k \sim j} \left\langle f \left(\sum_d f_{kd} f_{jd} \right) \right\rangle_{\prod_e q(f_{je}) \prod_{g \neq c} q(f_{kg})} \equiv \Psi(Q) \quad (\text{A } 5)$$

Similarly, the derivative of the absent-links energy is

$$2 \sum_{k \not\sim j} \left\langle f' \left(\sum_d f_{kd} f_{jd} \right) \right\rangle_{\prod_e q(f_{je}) \prod_{g \neq c} q(f_{kg})} \equiv \Psi'(Q) \quad (\text{A } 6)$$

where $f'(x) \equiv \log(1 - \sigma(x))$. Equating the derivative of equation (A 2) to zero, a fixed point condition for each $q_{k,c}$ $k = 1, \dots, V, c = 1, \dots, C$ is

$$q(f_{kc}) \propto e^{\Psi(Q) + \Psi'(Q)} \quad (\text{A } 7)$$

A difficulty here is that neither $\Psi(Q)$ nor $\Psi'(Q)$ are easy to compute, due to the non-linearities. A simple Gaussian Field approximation[6] assumes $\sum_d f_{kd} f_{jd}$ is Gaussian distributed for a fixed state of $f_{i,c}$. In this case, we need to find the mean and variance of $\sum_d f_{kd} f_{jd}$. Writing $\theta_{ab} \equiv q(f_{ab} = 1)$, and using the independence of q , the mean is given by

$$\mu_{kj} = f_{kc} \theta_{jc} + \sum_{d \neq c} \theta_{kd} \theta_{jd}$$

A similar expression is easily obtained for the variance σ_{kj}^2 . The Gaussian Field approximation then becomes,

$$q(f_{kc}) \propto e^{2\langle \sum_{j \sim k} f(x) + \sum_{j \not\sim k} f'(x) \rangle_{\mathcal{N}(x|\mu_{kj}, \sigma_{kj}^2)}} \quad (\text{A } 8)$$

where the one dimensional averages are performed numerically. By evaluating equation (A 8) for the two states of f_{kc} (and noting that the mean and variance of the field depends on these states), the approximate update for θ_{kc} is obtained. A simpler alternative is to assume that the variance of the field is zero, and approximate the averages by evaluating the functions at the mean of the field. We found that this latter procedure often gives satisfactory performance and therefore used this simpler and faster approach in the experiments.

One epoch corresponds to updating all the θ_{kc} , $k = 1, \dots, V$, $c = 1 \dots, C$. During each epoch the order in which the parameters are updated is chosen randomly.

Appendix B. Variational Bayes

$q(F)$ updates

A fixed point condition for the optimum of equation (4.4) is

$$q(F) \propto e^{\langle \log p(A|F, \alpha) \rangle_{q(\alpha)}} \approx e^{\log p(A|F, \langle \alpha \rangle)} \quad (\text{B } 1)$$

The average over $q(\alpha)$ in equation (B 1) in the first expression is complex to carry out and we simply approximate at the average value of the distribution. This reduces the problem to one similar to that of inferring F for a fixed C , as in Section Appendix A. We therefore make the same assumption that $q(F)$ factorizes according to equation (A 1). This gives updates of the form equation (A 8) where α has been set to its mean value.

$q(\alpha)$ updates

A fixed point condition for the optimum of equation (4.4) is

$$q(\alpha) \propto p(\alpha) e^{\langle \log p(A|F, \alpha) \rangle_{q(F)}},$$

Additionally we assume that $q(\alpha) = \prod_c q(\alpha_c)$. The resulting update

$$q(\alpha_c) \propto e^{\langle \log p(A|F, \alpha) \rangle_{q(F)} + \langle \log p(\alpha) \rangle_{\prod_{d \neq c} q(\alpha_d)}}$$

is difficult to compute and we take the naive approach of replacing averages by evaluation at the mean

$$q(\alpha_c) \propto p(\alpha_c, \langle \alpha_{\setminus c} \rangle) p(A | \langle f \rangle, \alpha_c, \langle \alpha_{\setminus c} \rangle) \quad (\text{B } 2)$$

Since α_c is binary, we can easily find equation (B 2) by evaluating at its two states.

Formally, the prior $p(\alpha)$ requires $\alpha \in \{0, 1\}^C$. However, in the above approximation, the mean α is non-binary. To deal with this, we replace $\sum_c I[\alpha_c = 1]$ by $\sum_c \langle \alpha_c \rangle$ and $\sum_c I[\alpha_c = 0]$ by $C_{max} - \sum_c \langle \alpha_c \rangle$. Since the expressions are valid for non-integer sums, this approximate procedure remains well defined.

The algorithm then updates $q(\alpha)$ and $q(F)$ until convergence. The effect is that, beginning with C_{max} clusters, under the updating, the posterior assigns α 's not required to state zero.

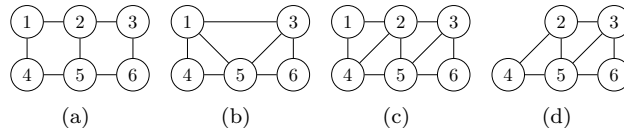


Figure 11. (a) Non decomposable Graph. (b) Elimination of node 1 for (a). (c) Decomposable graph. (d) Elimination of node 1 for (c).

Appendix C. Decomposable Graph

The concept of a decomposable graph lies at the heart of many issues related to the computational complexity of graph algorithms. For this reason, the same concept appears in a variety of communities with different terms : triangulated (Computer Science), chordal (mathematics), decimable (statistical physics).

We first define the graph operation of eliminating a variable (node). When a node i is eliminated, links are added between all the neighbours of node i . For example, if we eliminate node 2 in figure 11a, we remove node 2 from the graph and add links between the neighbours of node 2 (nodes 1,3,5), giving figure 11b. In this case, eliminating a node has introduced links between variables of the remaining subgraph. A decomposable graph is one for which there exists a variable elimination sequence such that no additional links appear. For example, in figure 11c, if we first eliminate node 1, then we arrive at figure 11d. Subsequently, one can eliminate nodes 4, 2, 5, 3 without inducing additional links in the remaining subgraphs (note that in general decomposable graphs can be non-planar). This means that figure 11c is a decomposable graph. Note that the elimination sequence is not unique. For decomposable graphs computational complexity is often dominated by the largest clique on the graph. For example, the partition function of an Ising model with interaction specified by a decomposable graph can be computed in less than order $N2^c$ steps, where c is the size of the largest clique in the graph.

References

- [1] E. Airoldi, D. Blei, E. Xing, and S. Fienberg. A latent mixed membership model for relational data. In *LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery*, pages 82–89, New York, NY, USA, 2005. ACM.
- [2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research.*, 9:1981–2014, 2008.
- [3] T. W. Anderson and I. Olkin. Maximum-likelihood estimation of the parameters of a multivariate normal distribution. *Linear Algebra and its Applications*, 70:147–171, 1985.
- [4] S. Arora and C. Lund. Hardness of approximations. In *Approximation algorithms for NP-hard problems*, pages 399–446. PWS Publishing Co., Boston, MA, USA, 1997.

- [5] D. Barber. Clique Matrices for Statistical Graph Decomposition and Parameterising Restricted Positive Definite Matrices. In D. A. McAllester and P. Myllymäki, editors, *UAI*, pages 26–33. AUAI Press, 2008.
- [6] D. Barber and P. Sollich. Gaussian Fields for Approximate Inference in Layered Sigmoid Belief Networks. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2000.
- [7] D. Barber and W Wiegnerinck. Tractable Undirected Approximations for Graphical Models. In *ICANN'98: International Conference on Artificial Neural Networks, Skövde*, 1998. ISBN 3 540 76263 9.
- [8] M. Brockington and J. Culberson. Camouflaging Independent Sets in Quasi-Random Graphs. In D. S. Johnson, editor, *Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge*, volume 26 of *DIMACS*. American Mathematical Society, 1996.
- [9] S. Chaudhuri, M. Drton, and T S. Richardson. Estimation of a covariance matrix with zeros. *Biometrika*, 94:199–216, 2007.
- [10] R. Diestel. *Graph Theory*. Springer, 2005.
- [11] M. Drton and T. Richardson. A New Algorithm for Maximum Likelihood Estimation in Gaussian Graphical Models for Marginal Independence. *Uncertainty in Artificial Intelligence*, 2003.
- [12] D. B. Dunson, J. Palomo, and K. Bollen. Bayesian Structural Equation Modeling. SAMSI 2005-5, Statistical and Applied Mathematical Sciences Institute, 2005.
- [13] E. Erosheva, S. Fienberg, and J. Lafferty. Mixed membership models of scientific publications. In *Proceedings of the National Academy of Sciences*, volume 101, pages 5220–5227, 2004.
- [14] M. E. Fisher. On the dimer solution of planar ising models. *J. Math. Phys.*, 7:17761781, 1966.
- [15] M. R. Garey and D. S. Johnson. *Computers and Intractability, A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, New York, 1979.
- [16] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications ACM*, 35:61–70, 1992.
- [17] M. C. Golumbic and I. Ben-Arroyo Hartman. *Graph Theory, Combinatorics, and Algorithms*. Birkhäuser, 2005.
- [18] F. Hadlock. Finding a Maximum Cut of a Planar Graph in Polynomial Time. *SIAM Journal on Computing*, 4(3):962–1030, 1975.
- [19] L. J. Heyer, S. Kruglyak, and S. Yooseph. Exploring Expression Data: Identification and Analysis of Coexpressed Genes. *Genome Research*, 9:1106–1115, 1999.

- [20] T. Hofmann, J. Puzicha, and M. I. Jordan. Learning from dyadic data. *NIPS*, pages 466–472, 1999.
- [21] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *Siam Journal on Scientific Computing*, 20(1):359–392, 1998.
- [22] H. Kautz, B. Selman, and M. Shah. ReferralWeb: Combining social networks and collaborative filtering. *Communications ACM*, 40:63–65, 1997.
- [23] V. Krebs. Political books data. www.orgnet.com.
- [24] S. L. Lauritzen. *Graphical Models*. Oxford University Press, 1996.
- [25] E. Meeds, Z. Ghahramani, R. Neal, and S.T. Roweis. Modelling dyadic data with binary latent factor. *NIPS*, 19:466–472, 1999.
- [26] M. E. J. Newman. The Structure and Functino of Complex Networks. *SIAM Review*, 45(2):167–256, 2003.
- [27] V. I. Paulsen, S. C. Power, and R. R. Smith. Schur products and matrix completions. *Journal of Functional Analysis*, 85:151–178, 1989.
- [28] W. J. Pullan and H. H. Hoos. Dynamic local search for the maximum clique problem. *Journal of Artificial Intelligence Research (JAIR)*, 25:159–185, 2006.
- [29] A. Roverato. Hyper Inverse Wishart Distribution for Non-decomposable Graphs and its Application to Bayesian Inference for Gaussian Graphical Models. *Scandanavian Journal of Statistics*, 29:391–411, 2002.
- [30] R. Silva, W. Chu, and Z. Ghahramani. Hidden Common Cause Relations in Relational Learning. *Neural Information Processing Systems*, 2007.
- [31] R. Silva and Z. Ghahramani. Bayesian Inference for Gaussian Mixed Graph Models. *Uncertainty in Artificial Intelligence*, 2006.
- [32] S. S. Skiena. *The algorithm design manual*. Springer-Verlag, New York, USA, 1998.
- [33] T. P. Speed and H. Kiiveri. Gaussian markov distributions over finite graphs. *Annals of Statistics*, 14:138–150, 1986.
- [34] N. Wermuth. Linear Recursive Equations, Covariance Selection, and Path Analysis. *Journal of the American Statistical Association*, 75(372):963–972, 1980.
- [35] W. Wiegnerinck. Variational approximations between mean field theory and the junction tree algorithm. *Uncertainty in Artificial Intelligence*, 16:626–633, 2000.