

Non-parametric Bayesian modeling of complex networks

Mikkel N. Schmidt and Morten Mørup

Abstract—Modeling structure in complex networks using Bayesian non-parametrics makes it possible to specify flexible model structures and infer the adequate model complexity from the observed data. This paper provides a gentle introduction to non-parametric Bayesian modeling of complex networks: Using an infinite mixture model as running example we go through the steps of deriving the model as an infinite limit of a finite parametric model, inferring the model parameters by Markov chain Monte Carlo, and checking the model’s fit and predictive performance. We explain how advanced non-parametric models for complex networks can be derived and point out relevant literature.

I. INTRODUCTION

We are surrounded by complex networks. From the networks of cell interaction in our immune system to the complex network of neurons communicating in our brain, our cells interplay to coordinate the functions of our body. We live in cities with complex energy supply networks and these cities are linked by advanced transportation systems. We interact within social circles and our computers are connected through the World Wide Web. To understand the structure of these large systems of biological, physical, social, and virtual networks, there is a great need to be able to model them mathematically [6].

Complex networks are studied in several different fields from computer science and engineering to physics, biology, sociology, and psychology. “Network science is an emerging, highly interdisciplinary research area that aims to develop theoretical and practical approaches and techniques to increase our understanding of natural and manmade networks” [6]. Network science can be considered “the study of network representations of physical, biological, and social phenomena leading to predictive models of these phenomena” [8].

In order to understand the many large-scale complex networks we sample and store nowadays there is a growing demand for advanced mathematical and statistical models that can account for the structure in these systems. The modeling aims are twofold; to provide a comprehensible description (i.e., descriptive modeling) and to infer unobserved properties (i.e., predictive modeling). In particular, a statistical analysis is useful when the focus lies beyond single node properties and local interactions but on the characteristics and behaviors of the entire system [6], [30], [54].

A complex network can be represented as a graph $G(V, E)$ with vertices (nodes) V and edges (links) E where an edge defines a connection between two of the vertices. In the following we denote the number of nodes in the graph by N and the number of links by L . Graphs are often represented

in terms of their corresponding adjacency matrix X defined such that $x_{i,j} = 1$ if there exists a link between node i and j and $x_{i,j} = 0$ otherwise. Common types of graphs include undirected, directed, and bipartite graphs, and these can in turn be weighted such that each link has an associated strength (see Figure 1). Complex networks are commonly stored in a sparse representation as an “edge list”; a set of L 3-tuples (i, j, w) where w is the weight of the link from node i to node j . Using this representation, the storage requirements for a network grows linearly in the number of edges of the graph.

A. Network characteristics

An important regimen in network science is to examine different characteristics or metrics computed from an observed network. The characteristics that have been examined include the distribution of the number of edges for each vertex (the degree distribution), the tendency of vertices to cluster together in tightly knit groups (the clustering coefficient), the average number of links required to move from one vertex to another (the characteristic path length), and many more (see Figure 2, and for a detailed list of studied network characteristics see [49].)

To assess the importance of these characteristics they can be contrasted with the properties of some class of random graphs: To discover significant properties which cannot be explained by pure chance. The most simple class of random graphs used for comparison is the so-called Erdős-Rényi graphs in which pairs of nodes connect independently at random with a given connection probability ϕ ,

$$x_{i,j} \sim \text{Bernoulli}(\phi), \quad \phi \in [0; 1]. \quad (1)$$

Amongst the findings is that many real networks exhibit “scale free” and “small-world” properties. A network is said to be scale free if its degree distribution follows a power law [4] in contrast to Erdős-Rényi random graphs which have a binomial degree distribution. The power law degree distribution indicates that many nodes have very few links whereas a few nodes (hubs) have a large number of links. A network is said to be small-world if it has local connectivity and global reach such that any node can be reached from any other node in a small number of steps along the edges. This associates with having a large clustering coefficient and small characteristic path length [57] and suggests that generic organizing principles and growth mechanisms may give rise to the structure of many existing networks [57], [4], [54], [10], [6], [30]. Using analytic tools from network science, studies

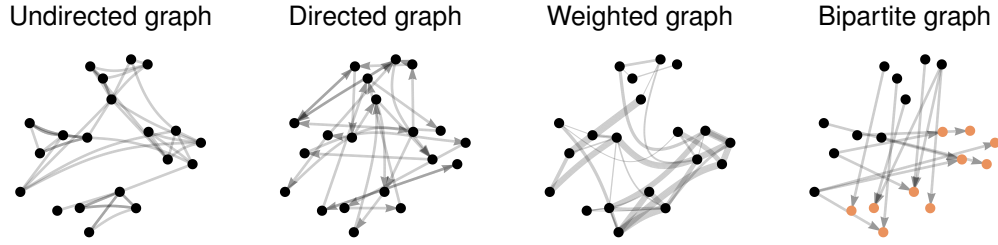


Fig. 1. Illustration of undirected, directed, weighted, and bipartite graphs. An undirected graph consists of a set of nodes and a set of edges. In directed graphs, edges point from one node to another. Edges in a weighed graph have an associated value e.g. representing the strength of the relation. A bipartite graph represents a set of relations between two disjoint sets of nodes. Non-parametric Bayesian models can be formulated for all of these types of network structures.

	Degree mean	Degree std.	Clustering coef.	Char. path length
Erdős-Rényi graph	1.8	1.1	0.0	2.9
Heavy tailed degree distribution	1.9	1.9	0.0	3.1
High clustering coefficient	3.1	1.2	0.5	3.4
Long characteristic path length	1.9	0.3	0.0	6.3

Fig. 2. Illustration of three important network characteristics: the degree distribution, clustering coefficient, and characteristic path length. The *degree* of a vertex is the number of edges that links it to the rest of the network. The *clustering coefficient*, defined as the average fraction of triangles relative to the total number of potential triangles given the vertex degree, quantifies the degree to which the vertices in a graph tend to cluster together. The *characteristic path length* is defined as the average shortest path between the vertices of the network.

have demonstrated that many complex networks behave far from random [57], [4], [54], [10].

B. Exponential random graphs

To understand the processes that govern the formation of links in complex networks, statistical models consider some class of probability distributions over networks. A prominent and very rich, general class of models for networks is the exponential random graph family [12], [46], [56], also denoted the p^* model. In the exponential random graph model the probability of an observed network takes the form of an exponential family distribution,

$$p(X|\theta) = \frac{1}{\kappa(\theta)} \exp \{ \theta^\top s(X) \}, \quad (2)$$

where θ is a vector of parameters, $s(X)$ is a vector of sufficient statistics, and $\kappa(\theta)$ is the normalizing constant that ensures that

the distribution sums to unity. In general, the sufficient statistic can depend on three different types of quantities:

Exogenous predictors: In addition to the network, side information is often available which can aid in modeling the network structure. Including such observed covariates on the node or dyad level allows the analysis of networks and side information in a single model.

Network statistics: Statistics computed on the network itself, such as counts of different network motifs can be included. This could be quantities such as the number of edges, triangles, two-stars, etc. Since these terms depend on the graph, they introduce a self-dependency in the model, significantly complicating the inference procedure. There is virtually no limit to which terms could potentially be included, and how to choose a suitable set terms for a specific network domain is an open problem.

Latent variables: The network can be endowed with a latent structure that characterizes the network generating process. The latent variables could for example be continuous or categorical variables on the node level or a latent hierarchical structure. The latent variables are most often jointly inferred with the model parameters. One reason for including latent variables is to aid in the understanding of the model: For example, if each network node is given a categorical latent variable, this corresponds to a clustering of the network nodes.

The parameters in exponential random graphs are usually estimated using maximum likelihood which can be non-trivial since the normalizing constant usually can not be explicitly evaluated. While exponential random graph models are very flexible and work well for predicting links, they have the following important shortcomings:

Model complexity: It can be difficult to determine the suitable model complexity: Which network statistics to include, how many latent dimensions or categories to include etc. To address this issue different approaches have been taken, including imposing sparsity on the parameters and using model order selection tools such as BIC and AIC.

Computational complexity: In general, the computational complexity of inference in exponential random graph models grows with the size of the network, $\mathcal{O}(N^2)$, rather than with the number of edges, $\mathcal{O}(L)$, making exact large scale analysis infeasible. There are, however, certain special cases for which the complexity of inference scales linearly in the number of edges, which we will discuss

further in the sequel.

Inferential complexity: When only exogenous predictors and latent variables are included in the model, inference is fairly straightforward; however, when *network statistics* are included inference can be challenging, involving either heuristics such as pseudo likelihood estimation or complicated Markov chain Monte Carlo methods [46], [45].

C. Non-parametric Bayesian network models

In the following we present a number of recent network modeling approaches [26], [58], [1], [25], [34], [37], [38], [36] based on Bayesian non-parametrics which can all be seen as extensions or special cases of the exponential random graph model. In non-parametric modeling, the structure of the model is not fixed, and thus the model complexity can adapt as needed according to the complexity of the data. This forms a principled framework for addressing the first issue (model complexity) mentioned above. With respect to the second issue (computational complexity), it turns out that many of these non-parametric Bayesian models can be constructed such that their computational complexity is linear in the number of links, allowing these methods to scale to large networks. While it certainly is possible to include network statistics in non-parametric Bayesian network models, Bayesian non-parametrics does not address the third issue (inferential complexity) which is an open area of research.

The focus of the remainder of this paper is twofold: i) To provide a comprehensible tutorial on the most simple non-parametric Bayesian network model: The infinite relational model [26], [58]. ii) To give a brief overview of current advances in non-parametric Bayesian network models.

II. TUTORIAL ON THE INFINITE RELATIONAL MODEL

In the following we give a tutorial introduction to the infinite relational model [26], [58] which is perhaps the most simple non-parametric Bayesian network model. We will derive the necessary Bayesian non-parametric machinery from first principles by taking limits of a parametric Bayesian model. Understanding the details of involved in deriving this simple model later serves as a foundation for understanding other more complicated non-parametric constructions. Further, we go through the details involved in inference by Markov chain Monte Carlo, and show how a Gibbs sampler can be implemented in a few lines of computer code. Finally, we demonstrate the model on three network datasets and compare with other models from the exponential random graph model family.

A. The infinite relational model

The infinite relational model is a latent variable model where each node is assigned to a category, corresponding to a clustering of the network nodes. The number of clusters is learned from data as part of the statistical inference. As a starting point, we introduce a Bayesian parametric version of the model, which we later extend to the non-parametric setting. For readers unaccustomed with Bayesian modeling, we provide a short introduction, see Figure 3.

1) *A parametric Bayesian stochastic blockmodel:* A simple and very powerful approach to modeling structure in a complex network is to use a *mixture model*, leading to a Bayesian version of the so-called *stochastic blockmodel* [42]. In a mixture model, the observations are assumed to be distributed according to a mixture of K components belonging to some parametric family. Conditioned on knowing which mixture components generated each datum, the observations are assumed independent. In a mixture model for network data, each node belongs to a single mixture component, and since each edge is associated with two nodes, its likelihood will depend on two components. Thus, the likelihood of the network will take the following form,

$$p(X|\theta) = \prod_{(i,j)} p(x_{i,j}|z_i, z_j, \phi) \quad (8)$$

where the product ranges over all node pairs, and the parameters are given by $\theta = \{\{z_i\}_{i=1}^N, \phi\}$ where z_i indicates which mixture component the i th node belongs to and ϕ denotes any further parameters. In the most simple setting, each term in the likelihood could be a Bernoulli distribution (a biased coin flip),

$$p(x_{i,j}|z_i, z_j, \phi) = \text{Bernoulli}(\phi_{z_i, z_j}) \quad (9)$$

$$= (\phi_{z_i, z_j})^{x_{i,j}} (1 - \phi_{z_i, z_j})^{1-x_{i,j}}, \quad (10)$$

such that $\phi_{k,\ell}$ denotes the probability of an edge between two nodes in group k and ℓ . To finish the specification of the model, we must define prior distributions for the mixture component indicators z as well as the link probabilities ϕ . Starting with ϕ , a natural choice would be independent Beta distributions for each pair of components,

$$p(\phi_{k,\ell}) = \text{Beta}(a, b) \quad (11)$$

$$= \frac{1}{B(a, b)} (\phi_{k,\ell})^{a-1} (1 - \phi_{k,\ell})^{b-1}, \quad (12)$$

where the parameters for example can be set to $a = b = 1$ to yield a uniform distribution. A natural choice for z would be a K -dimensional categorical distribution,

$$p(z_i = k|\pi) = \pi_k \quad (13)$$

parameterized by $\pi = \{\pi_k\}_{k=1}^K$ where $\sum_{k=1}^K \pi_k = 1$. How, then, should π be chosen? We could for example set each of these parameters to a fixed value, e.g. $\pi_k = \frac{1}{K}$, but this would be a strong prior assumption specifying that the mixture components have the same number of members on average. A more flexible option would be to define a hierarchical prior, where π is generated from a Dirichlet distribution,

$$p(\pi) = \text{Dirichlet}(\alpha) \quad (14)$$

$$= \frac{1}{B(\alpha)} \prod_{k=1}^K \pi_k^{\alpha_k - 1}. \quad (15)$$

where $B(\alpha)$ is the multinomial beta function, which can be expressed using the gamma function,

$$B(\alpha) = \frac{\prod_{k=1}^K \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^K \alpha_k)}. \quad (16)$$

Bayesian modeling

In traditional frequentist statistical modeling, probabilities describe relative frequencies of random variables in the limit of infinitely many trials. Model parameters are considered unknown but fixed quantities. A statistical model is characterized by a set of distributions,

$$p(X|\theta), \quad (\text{likelihood}) \quad (3)$$

where the unknown parameter θ takes values in parameter space Θ . When considered as a function of θ , the distribution $p(X|\theta)$ is known as the *likelihood*. A non-parametric model is, contrary to what one might expect from its name, not a model without parameters, but a model which can not be parameterized by a finite dimensional parameter space. In other words, we can think of a non-parametric model as one having an infinite number of parameters—a notion that will be made explicit later.

In Bayesian modeling, in addition to describing random variables, probabilities are used to describe inferences, i.e., to quantify degree of belief about the parameters. Although parameters are still thought of as unknown, fixed quantities, they are modeled as random variables where the randomness reflects our lack of knowledge about them. To this end, they are assigned a so-called *prior* probability distribution,

$$p(\theta), \quad (\text{prior}) \quad (4)$$

representing the degree of belief about the model parameters prior to observing

any data. Often, it is convenient to specify the prior using some parameterized family of distributions. The parameters of the prior distribution are often referred to as *hyper parameters* and can either be fixed or assigned *hyper priors* which themselves might have hyper-hyper parameters, etc. A model defined in this manner is referred to as a hierarchical Bayesian model.

Once the prior and the likelihood have been decided upon, the model is completely specified. Inference entails using the rules of probability to compute the conditional distribution of the parameters given the observations, also known as the *posterior*,

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{\int p(x|\theta)p(\theta)d\theta}, \quad (5)$$

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}. \quad (6)$$

Thus, we are not merely interested in a single parameter estimate, but aim at estimating a distribution over parameters quantifying our state of knowledge about the parameters after observing the data. Often, only a subset of the parameters is of interest—the others are simply used as a means to specifying a reasonable probabilistic model, but are not of interest themselves. Such parameters are often referred to as *nuisance* parameters. Assume for instance the parameters $\theta = \{\iota, \nu\}$ can be divided into interesting (ι) and nuisance (ν) parameters.

In that case, we compute the posterior

distribution of the parameters of interest,

$$p(\iota|x) = \int p(\theta|x)d\nu, \quad (7)$$

which can be found by marginalizing (integrating over) the nuisance parameters.

Although conceptually simple, inference might be computationally unwieldy because of high dimensional and analytically intractable integrals (or summations, in the case of discrete parameters). In practice one must therefore use some method of approximation, which we will discuss later.

Bayesian data modeling can be divided into three tasks [13]:

a) Joint distribution:: The first step involves formulating the probabilistic model, i.e. a joint distribution over data and parameters, by specifying the likelihood and priors.

b) Inference:: Next, the posterior distribution of the parameters is inferred, often using some method of numerical approximation such as Monte Carlo sampling.

c) Checking implications:: Finally we check how well the model describe the data and evaluate the implications of the posterior distribution by computing quantities of interest and making decisions.

In this paper we go through the details of these three steps in the context of the infinite relational model [26], [58].

Fig. 3. A brief introduction to Bayesian modeling introducing the concepts needed in this paper.

Since each component a priori is equally likely, we select the concentration parameters to be equal to each other, $\alpha_1 = \dots = \alpha_K = \frac{A}{K}$ such that the scale of the distribution is $\sum_{k=1}^K \alpha_k = A$. This results in a joint prior over z and π given by

$$p(z, \pi) = \left[\prod_{i=1}^N p(z_i|\pi) \right] \times p(\pi|\alpha) \quad (17)$$

$$= \frac{1}{B(\alpha)} \prod_{k=1}^K \pi_k^{n_k + \alpha_k - 1}, \quad (18)$$

where n_k denotes the number of z_i 's with the value k .

2) *Nuisance parameters:* As we are not particularly interested in the mixture component probabilities π (they can be considered nuisance parameters) we can compute the effective prior over z by marginalizing over π which has a closed form expression due to the conjugacy between the Dirichlet and

Categorical distributions (i.e., the posterior distribution of π has the same functional form as the prior),

$$p(z) = \int p(z, \pi)d\pi = \frac{B(\alpha + n)}{B(\alpha)} \quad (19)$$

$$= \frac{\Gamma(A)}{\Gamma(A + N)} \prod_{k=1}^K \frac{\Gamma(\alpha_k + n_k)}{\Gamma(\alpha_k)}. \quad (20)$$

This resulting effective prior distribution is known as a multivariate Pólya distribution.

Furthermore, the link probabilities ϕ can also be considered nuisance parameters, and can also be marginalized analytically due to the conjugacy between the Beta and Bernoulli distri-

butions,

$$p(X|z) = \int p(X|z, \phi) p(\phi) d\phi \quad (21)$$

$$= \prod_{(k,\ell)} \frac{B(m_{k,\ell} + a, \bar{m}_{k,\ell} + b)}{B(a, b)}, \quad (22)$$

where the product ranges over all pairs of components and $m_{k,\ell}$ and $\bar{m}_{k,\ell}$ denote the number of links and non-links between nodes in component k and ℓ respectively.

3) *An infinite number of components:* In the previous section we specified a *parametric* Bayesian mixture model for complex networks. In the following we move to the non-parametric setting in which the number of mixture components is allowed to be countably infinite. First, consider what happens when the number of components is much larger than the number of nodes in the graph. In that situation, many of the components will not have any nodes assigned to them; in fact, no more than N components can be non-empty, corresponding to the worst case situation where each node has a component of its own. To handle the situation with an infinite number of components, we can not explicitly represent the components but, as we will show in the following, we need only an explicit representation of the finite number of non-empty components.

As we defined the model so far we have introduced K *labelled* mixture components. This means that if we for example have $N = 5$ nodes and $K = 4$ components, we assign a separate probability to, say, the configurations $\{1, 2, 1, 4, 2\}$ and $\{3, 4, 3, 2, 4\}$ even though they correspond to the same clustering of the network nodes. A better choice is to specify the probability distribution directly over the *equivalence class* of partitions of the network nodes. Since we have K labels in total to choose from, there are K possible labellings for the first component, $K - 1$ for the second, etc. resulting in a total of

$$\frac{K!}{(K - \bar{K})!} \quad (23)$$

labellings corresponding to the same partitioning, where \bar{K} is the number of non-empty components. Thus, defining a parameter \bar{z} that holds the partitioning of the network nodes, we have

$$p(\bar{z}) = \frac{K!}{(K - \bar{K})!} \frac{\Gamma(A)}{\Gamma(A + N)} \prod_{k=1}^{\bar{K}} \frac{\Gamma(\alpha_k + n_k)}{\Gamma(\alpha_k)}. \quad (24)$$

Since \bar{z} represents partitions rather than labels it can be finitely represented. We can now simply let the number of components go to infinity by computing the limit of the prior distribution for \bar{z} ,

$$\lim_{K \rightarrow \infty} p(\bar{z}) = \frac{\Gamma(A) A^{\bar{K}}}{\Gamma(A + N)} \prod_{k=1}^{\bar{K}} \Gamma(n_k). \quad (25)$$

The details involved in computing this limit can be found in [15] and [40]. The limiting stochastic process is known as a *Chinese restaurant process* (CRP) [2] (for an introduction to the CRP, see [14]). Compactly, we may write

$$\bar{z} \sim \text{CRP}(A). \quad (26)$$

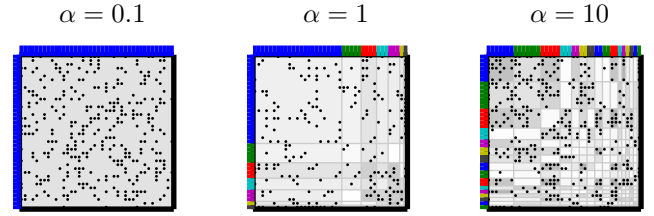


Fig. 4. Example of graphs generated according to the *infinite relational model* for different choices of the parameter α of the Chinese Restaurant Process.

4) *Summary of the generative model:* In summary, the generative process for the infinite relational model can be expressed as follows:

$$\bar{z} \sim \text{CRP}(A), \quad (27)$$

$$\phi_{k,\ell} \sim \text{Beta}(a, b), \quad (28)$$

$$x_{i,j} \sim \text{Bernoulli}(\phi_{z_i, z_j}). \quad (29)$$

The network nodes are partitioned according to a Chinese restaurant process; a probability of linking between each pair of node clusters is simulated from a Beta distribution; and each link in the network is generated according to a Bernoulli distribution depending on which clusters the pair of nodes belong to.

Identically, in the notation of exponential random graph models the likelihood in Eq. (29) can be expressed as

$$p(X|z, \phi) = \frac{1}{\kappa(z, \phi)} \exp[\theta(\phi)^\top s(X, z)] \quad (30)$$

where the sufficient statistics are the counts of links between each pair of clusters, $s(X, z) = \{m_{k,\ell}\}$, and the natural parameter is the log odds of links between each pair of clusters, $\theta(\phi) = \left\{ \log \frac{\phi_{k,\ell}}{1 - \phi_{k,\ell}} \right\}$.

B. Inference

Having specified the model in terms of the joint distribution, the next step is to examine the posterior distribution which is given as

$$p(\bar{z}|X) = \frac{p(X|\bar{z})p(\bar{z})}{\sum_{\bar{z}} p(X|\bar{z})p(\bar{z})}. \quad (31)$$

Here, the numerator is easy to compute as the product of Eq. (22) and (24); however, the denominator is difficult to handle as it involves an elaborate summation over all possible node partitionings. Consequently, some form of approximate inference is needed.

There are two major paradigms in approximate inference: Variational and Monte Carlo inference. The idea in variational inference is to approximate the posterior distribution with a simple, tractable distribution which is fitted to the posterior by minimizing some criterion such as the information divergence [5].

In Monte Carlo approximation the idea is to generate a number of random samples from the posterior distribution and approximate intractable integrals and summations by empirical averages based on the samples.

In the following, we focus on Monte Carlo inference. In particular we review the Gibbs sampler for the infinite relational model.

1) *Gibbs sampling*: In Gibbs sampling the variables are iteratively sampled from their conditional distribution, and repeating this process the samples will eventually approximate the posterior distribution. We iteratively sample the partition assignments, \bar{z}_n , from their conditional distribution,

$$p(\bar{z}_n = k | \bar{z}^{\setminus n}, X), \quad (32)$$

where $\bar{z}^{\setminus n}$ denotes all partition assignments except \bar{z}_n . An expression for this conditional distribution can be found by considering which terms in the likelihood and prior will change when node n is assigned to a different partition. For the prior in Eq. (24) we have

$$p(\bar{z}_n = k | \bar{z}^{\setminus n}) \propto \begin{cases} n_k^- & k \text{ is an existing partition,} \\ A & k \text{ is a new partition,} \end{cases} \quad (33)$$

where n_k^- is the number of nodes associated with component k not counting node n . Adding node n to an existing component increases the argument of the corresponding Gamma function by one, effectively multiplying the prior by n_k^- , whereas adding the node to a new cluster increases \bar{K} by one, effectively multiplying the prior by A . For the likelihood in Eq. (22), adding node n to partition k effectively multiplies the likelihood by

$$\prod_{\ell} \frac{B(m_{k,\ell}^{\setminus n} + r_{n,\ell} + a, \bar{m}_{k,\ell}^{\setminus n} + n_{\ell} - r_{n,\ell} + b)}{B(m_{k,\ell}^{\setminus n} + a, \bar{m}_{k,\ell}^{\setminus n} + b)}, \quad (34)$$

where $m_{k,\ell}^{\setminus n}$ and $\bar{m}_{k,\ell}^{\setminus n}$ denote the number of links and non-links between nodes in component k and ℓ , not counting any links from node n , and $r_{n,\ell}$ is the number of links from node n to any nodes in component ℓ .

In order to perform Gibbs sampling we can now simply consider each node in turn; for each partition (including a new, empty partition) compute the product of Eq. (33) and (34); normalize to yield a categorical distribution over partitions; and sample a new \bar{z}_n according to this distribution. The final algorithm is summarized in Fig. 5. The result after running the Gibbs sampler for $2T$ iterations is a set of samples of \bar{z} , where usually the first half is discarded for burn in. This yields a final ensemble $\{\bar{z}^{(t)} : t \in 1, \dots, T\}$ approximately sampled from the posterior.

2) *Computational complexity*: In the algorithm outlined in Figure 5 it can be observed that there are two loops: One over the T simulated samples and one over the N nodes in the network. In each run of the inner loop, a node is assigned to a cluster by the Gibbs sampler. In the following we consider the number of clusters K a constant (although of course it will vary depending on the network data), and examine how the computational complexity of the algorithm depends on the number of nodes and edges in the network.

In the code in Figure 5 the variables $M0$, $M1$ and m , which hold the counts of nonlinks, links, and nodes, are recomputed in each iteration. In a more sensible implementation, these quantities would be precomputed and efficiently updated during the Gibbs sampling.

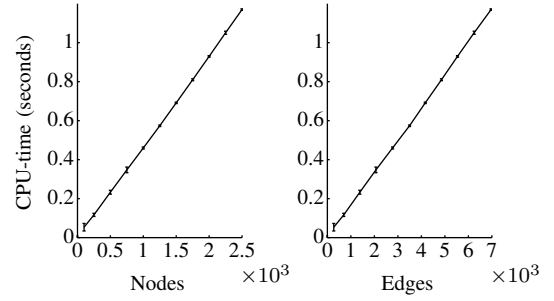


Fig. 6. Experiment demonstrating that the computational complexity grows linear in both the number of nodes N and edges L for the IRM model. The graphs used in the experiments are generated with $K = 5$ communities of equal size and $\phi = \phi_c/N$ where ϕ_c is kept constant in the experiments ensuring that the number of edges L grows linearly with the number of nodes N in the generated networks. The Gibbs sampler used in the experiment was implemented to pre-compute $M0$, $M1$ and m resulting in a computational complexity of $O(L)$ for each iteration of the sampler. Given are the mean CPU-times in seconds for the sampler and standard deviation across $T = 10$ iterations when varying the number of nodes (N) and edges (L) in the generated graphs.

Evaluating the probability of assigning a node to each cluster then requires the computation of the vector \mathbf{r} which holds the count of links from node n to each of the clusters. The time complexity of this computation is on the order of the node degree. Looping over the nodes gives a total time complexity of $O(L)$ where L is the number of edges in the graph. To calculate the probabilities of assigning the nodes to the clusters for all N Gibbs samples requires $2K^2N$ evaluations of the (logarithm of the) Beta function so the time complexity of this computation is $O(N)$. As a result, since in general $L > N$, the total computational complexity of the Gibbs sampler for the IRM model is $O(L)$. Figure 6 demonstrates that this linear scaling is observed in practice when analyzing networks of varying numbers of nodes and edges.

For comparison, Monte Carlo maximum likelihood inference in exponential random graph model based on endogenous network statistics requires the simulation of random networks from the ERGM distribution, which is inherently an $O(N^2)$ operation. We should note though, that in practice we would not expect the time complexity of the IRM to scale linearly in the number of edges, since the number of clusters most likely would increase with the size of the network and since the number of required iterations of the Gibbs sampler might also go up.

C. Checking model fit

Once an approximation of the posterior distribution has been obtained, we wish to check the implications of the model. This can include computing the posterior distribution of important quantities of interest, evaluating how well the model fits the data, and making predictions about unobserved data.

1) *Computing posterior quantities*: Say we are interested in some function $f(\bar{z})$ that depends on the model. We can now

```

function Z = irm(X,T,a,b,A)
N = size(X,1); z = true(N,1); Z = cell(T,1);
for t = 1:T
    for n = 1:N
        nn = [1:n-1 n+1:N];
        K = size(z,2);
        m = sum(z(nn,:))'; M = repmat(m,1,K);
        M1 = z(nn,:)'*X(nn,nn)*z(nn,:)- ...
            diag(sum(X(nn,nn)*z(nn,:).*z(nn,:)/2);
        M0 = m*m'-diag(m.*(m+1)/2) - M1;
        r = z(nn,:)'*X(nn,n); R = repmat(r,1,K);
        logP = sum([betaln(M1+R+a,M0+M-R+b)-betaln(M1+a,M0+b) ...
            betaln(r+a,m-r+b)-betaln(a,b)],1)' + log([m; A]);
        P = exp(logP-max(logP));
        i = find(rand<cumsum(P)/sum(P),1);
        z(n,:) = false; z(n,i) = true;
        z(:,sum(z)==0) = [];
    end
    Z{t} = z;
end

```

% Initialization
 % For each Gibbs sweep
 % For each node in the graph
 % All indices except n
 % No. of components
 % No. of nodes in each component
 % No. of links between components
 % No. of non-links between components
 % No. of links from node n
 % Log probability of n belonging
 % to existing or new component
 % Convert from log probability
 % Random component according to P
 % Update assignment
 % Remove any empty components
 % Save result

Fig. 5. MATLAB code implementing the infinite relational model. The code illustrates the computations involved in the Gibbs sampler, but is not efficient since it recomputes all the needed link counts in each iteration.

compute the posterior distribution of this quantity,

$$p(f(\bar{z})) = \sum_{\bar{z}'} \delta(f(\bar{z}) = f(\bar{z}')) p(\bar{z}'|X) \quad (35)$$

$$\approx \frac{1}{T} \sum_{t=1}^T \delta(f(\bar{z}) = f(\bar{z}^{(t)})), \quad (36)$$

approximated by an empirical average over the posterior samples. For example, the approach can be used to compute the posterior distribution over the number of components in the mixture model or other quantities of interest.

2) *Link prediction*: Missing data is easily handled in the Bayesian framework, simply by leaving out the terms in the likelihood corresponding to unobserved links. If we observe only a part of the network and are interested in predicting the presence or absence of an unobserved link between two nodes, we can simply compute the posterior predictive distribution of the missing link,

$$p(x_{i,j}|X) = \sum_{\bar{z}} p(x_{i,j}|\bar{z}, X) p(\bar{z}|X) \quad (37)$$

$$\approx \frac{1}{T} \sum_{t=1}^T p(x_{i,j}|\bar{z}^{(t)}, X). \quad (38)$$

Here X denotes the observed part of the network, and $\bar{z}^{(t)}$ is simulated from the posterior distribution where only the observed part of the network is conditioned on. Inserting $p(x_{i,j}|\bar{z}, X) = \int p(x_{i,j}|\theta, \bar{z}) p(\theta|\bar{z}, X) d\theta$ yields

$$p(x_{i,j}|X) \approx \text{Bernoulli}(\rho_{i,j}), \quad (39)$$

where

$$\rho_{i,j} = \frac{1}{T} \sum_t \frac{m_{z_i^{(t)}, z_j^{(t)}} + a}{m_{z_i^{(t)}, z_j^{(t)}} + \bar{m}_{z_i^{(t)}, z_j^{(t)}} + a + b} \quad (40)$$

Predicting missing links can be used to compare different models: A number of links can be excluded when fitting the models which can then be compared by assessing their ability

to predict the held-out links. Since the links in a network are highly correlated and because many networks exhibit a highly imbalanced distribution of links and nonlinks, care must be taken in choosing a hold out test set in an appropriate way. If the test set is chosen to balance the number of links and nonlinks, its distribution will not correspond to the full network which makes the absolute link prediction results difficult to interpret. Thus, although indicative of a model's predictive performance, this approach is perhaps best suited for the relative comparison of different models. If, on the other hand, several examples of full networks are available, a whole network can be used as test data making the absolute link prediction results directly interpretable.

3) *Posterior predictive checking*: Finally, we might be interested in examining how well our model describes the data to assess if the model is appropriate for the data at hand or if a more suitable model should be constructed. A principled approach to achieving this is posterior predictive checking. First, an ensemble of replicated networks from the posterior predictive distribution is generated from,

$$p(X^{\text{rep}}|X) = \sum_{\bar{z}} p(X^{\text{rep}}|\bar{z}, X) p(\bar{z}|X), \quad (41)$$

which as before can be approximated using samples of \bar{z} simulated from the posterior using Eq. (39). Now, the idea is to compare characteristics of the observed network, such as the degree distribution, clustering coefficient, and characteristic path length, with the posterior predictive distribution of these properties, approximated by the empirical distribution over the ensemble of replicated networks. If the model fits well, the observed characteristic of the network should be quite likely under the posterior predictive distribution, whereas a large discrepancy indicates model mismatch. Posterior predictive checking is useful for model *criticism*, i.e., for exploring lack of fit as opposed to testing whether the model is correct. Discovering network characteristics for which the model does not fit the data well can inspire to the development of more

sophisticated models; however, even a simple model which does not fit the data in all respects can be useful.

D. Directed, weighted, bipartite, and multiple networks

The infinite relational model readily extends to other types of graphs including directed, weighted and bipartite networks as well as multiple networks on the same set of nodes. These extensions can be arrived at by modifying the model parametrization and the observational model (the likelihood function) as well as making appropriate changes to the priors. The process of formulating the joint distribution and deriving a Markov chain Monte Carlo procedure for inference closely follows the steps we have taken for the basic infinite relational model described in the previous sections. The extensions described below can also be combined, for example to model a set of directed, bipartite networks with edge weights.

1) *Directed networks*: In a directed network, the links have an associated direction, so that they point from one node to another. A directed network can be represented by an asymmetric adjacency matrix, and the directionality of links between groups can be modelled through the parameter ϕ by the existence of asymmetric interactions between the groups such that $\phi_{k,\ell} \neq \phi_{\ell,k}$. This double the number of link probability parameters ϕ . The rest of the model is unaffected, except for the likelihood which must now be evaluated not for each *pair* of nodes but for each *ordered pair* of nodes. This extension of the infinite relational model assigns different probabilities to links in each direction between each pair of clusters, but has only a single parameter for the link probability within each cluster—thus, directionality is not modelled within clusters.

2) *Bipartite networks*: A bipartite network is defined as a set of links between two disjoint sets of nodes, possibly with different cardinality. The adjacency matrix for a bipartite network can thus be non-square. We can then use two independent Chinese restaurant processes to model the clustering of the two sets of nodes,

$$\bar{z} \sim \text{CRP}(A_z), \quad \bar{w} \sim \text{CRP}(A_w) \quad (42)$$

and change the likelihood to (cf. Eq. (29))

$$x_{i,j} \sim \text{Bernoulli}(\phi_{z_i,w_j}). \quad (43)$$

This latter parameterization is also useful for the modeling of directed networks when the groupings of the nodes may be different for the rows and columns of the adjacency matrix.

3) *Weighted networks*: In a weighted network, each edge has a (scalar) weight associated with it. Depending on the type of weights, the Bernoulli likelihood can be changed to some other suitable distribution: For example, if the weights are positive integers [36], a Poisson distribution could be employed,

$$x_{i,j} \sim \text{Poisson}(\lambda_{z_i,z_j}), \quad (44)$$

where λ is the rate parameter for the edge weights, playing the role of ϕ in the Bernoulli model, c.f. Eq. (29). As a prior over λ , the typical choice is a Gamma distribution, replacing the Beta priors for ϕ . If the weights are real numbers [18] an

observational model based on a Normal distribution might be appropriate,

$$x_{i,j} \sim \text{Normal}(\mu_{z_i,z_j}, \sigma_{z_i,z_j}^2). \quad (45)$$

Here, we have two sets of parameters, μ and σ^2 , denoting the means and variances of the edge weights between nodes in groups i and j . Again, appropriate priors for μ and σ^2 should be selected.

4) *Multiple networks*: Sometimes the data consists of multiple observations of networks on the same set of nodes (see [26], [34], [36], [3]). The only required change to the model is that the likelihood should be evaluated as the product of the likelihoods for each observed network. It can then either be assumed that the clustering structure as well as the link probabilities are equal across the multiple networks, that the clustering structure is shared but the link probabilities only shared according to an additional clustering of the multiple graphs, or that the clustering is shared but each network has an individual set of link probabilities, ϕ . When the link probabilities are analytically marginalized this leads to three different expressions for the marginal likelihood.

E. Experimental evaluation

In the following we conduct a series of experimental evaluations with the infinite relational model, highlighting some of its properties and comparing it with other models.

1) *Analysis of three example networks*: To demonstrate the non-parametric Bayesian modeling framework in practise, we analyzed three real networks:

Zachary's Karate Club: Zachary's Karate club is an undirected unweighted network of friendships between 34 members of a karate club at a US university in the 1970s [59]. A total of 74 undirected links between the members of the Karate club are observed. In the analysis the standard IRM model was used.

Connectome of Caenorhabditis Elegans: The only complete connectome currently recorded of an organism is the directed integer weighted network of the 8,799 connections between the 302 neurons of the Caenorhabditis Elegans. The network has been compiled in [57]. In the analysis the weighted IRM model with a Poisson likelihood and Gamma priors was used.

Drugs and side effects: The drugs and side effects network is a bipartite network on marketed medicines and their recorded adverse drug reactions extracted from public documents and package inserts. The network currently consists of 996 drugs and 4,199 side effects with 100,049 unweighted links between drugs and side effects [29]. In the analysis the bipartite IRM model was used.

These three networks in turn represent three important complex network application domains within social science, neuroscience and bio-informatics.

The parameters of the models were inferred by Markov chain Monte Carlo sampling such that 250 iterations were used as burn in for the sampler and 250 iterations for drawing samples from the posterior. To improve mixing the data was analyzed based on 5 randomly initialized runs. In addition

to a Gibbs sampler (as described in Fig. 5) the so-called split-merge sampler described in [26], [24] was also employed. The hyper-parameters for the Beta distribution we set to $a = b = 1$. The posterior distribution of the number of components was computed. For assessing model fit by prediction of missing links, excluded 10% of links and an equivalent number of non-links in the analysis of the Zachary’s Karate Club data and 5% of links and an equivalent number of non-links in the two larger Connectome and Drug-side effects networks. For posterior predictive checking, we stored every 25th posterior sample and generated 20 replicated networks for each sample for each of the five random initializations. From the ensemble of these networks the distribution over the network characteristics; *degree mean, degree standard deviation, characteristic path length and clustering coefficient* were calculated and compared to the true values of these quantities for the actual network.

The results of the modeling is given in Fig. 7. The figure illustrates the network as well as a permutation of the networks adjacency matrix. The nodes are color coded according to the partition given by the sample with highest posterior likelihood across the five random initializations. From the permuted adjacency matrices it can be seen that the nodes of the networks have been grouped into clusters that share similar patterns of interactions, defining regions of network homogeneities. These blocks are color coded according to the expected value of the corresponding group interactions using a logarithmic gray scale. On the right, the posterior distribution of the number of components is shown as well as the models performance in predicting held-out links. The link prediction performance is quantified by the area under curve (AUC) of the receiver operator characteristic (ROC) [34]. In addition, the results of the posterior predictive checking of the models ability to account for the mean and standard deviation of the degree distribution, characteristic path length and clustering coefficient are given. Since the degree is explicitly modelled in the IRM model, this posterior predictive check serves only as a sanity check: The IRM should by definition get this right except for a small bias due to the prior.

From these results it can be seen that the IRM model accounts well for all the considered characteristics in the Zachary Karate Club network but that it poorly accounts for the degree standard deviation, the clustering coefficient, and the characteristic path length of the connectome of *C. Elegans*. As expected, the average degree falls within the lower tails of the simulated distributions for all the estimated models, but they underestimate the standard deviation of the node degree of both the connectome and drugs and side effect networks. This highlights a deficiency of the IRM model, namely that it does not explicitly model the degree distribution.

While the IRM is adept in identifying blocks of homogeneous network regions with the ϕ (nuisance) parameter specifying the density of each of these blocks, it does not explicitly account for microscale properties such as triangles and node degree. Hence, the clustering coefficient, characteristic path length and the standard deviation of the degree distribution is not well accounted for by the model as is evident in the posterior predictive checks. Despite these limitations,

the infinite relational model does well account for mesoscale structure in the networks as quantified by its ability to predict links: For all the three networks the infinite relational model is able to predict links significantly better than random guessing.

Apart from being able to predict links the IRM model has made the structure of the networks substantially more comprehensible by reducing the complex network of pairwise interactions to a much smaller number of groups (defined by \bar{z}) and their interactions (defined by ϕ). The IRM can therefore be considered an efficient framework for compressing a large complex network to a smaller network constituting consistent patterns of interactions between groups of nodes which can substantially facilitate in the understanding of mesoscale network patterns. For example, the analysis of the Zachary’s Karate Club network, the infinite relational model reveals six groups of club members including two large groups and two singletons (actually the posterior has support for five to eight groups, so these other configurations should also be considered in the interpretation of the results). It is known from the literature [59] that the karate club later split into two fractions, corresponding to the two large groups, led by the president and the instructor which are the two singletons.

2) *Comparison with other models:* Next, we compare the IRM model to several other methods on a set of social networks derived from a study of intra-organizational relations: *Intra-organizational relations:* This set of undirected networks [9] consists of two types of relations defined on the same set of nodes corresponding to employees in a consultancy company. Links in the first network signifies employees who interact whereas links in the second network signifies that either of the employees thinks that the other has expertise in an area important to her. The networks were generated by thresholding and symmetrizing the original directed weighted networks [9]. The two networks are highly correlated since employees would be expected to interact frequently with colleagues with important expertise.

We used the first of the two networks for training and examined the model fit by assessing the posterior predictive distribution of the node degree distribution. We fit an IRM model as well as two other non-parametric Bayesian network models, the infinite multiple membership relational model (IMRM) and the Bayesian community detection model (BCD) which are discussed further in the sequel. These models were fit using MCMC with 10,000 rounds of Gibbs sampling where the first half of the samples were discarded for burn-in. Furthermore, we fit an exponential random graph model (ERGM) using the network statistics *sociality* and *gwdegree* [35] as well as a latent position and cluster model (ERGMM) [28] using a latent space of dimension four and six latent clusters (varying these parameters gave similar results). To compare how well the models fit the data we plotted the posterior predictive distribution of the degree distribution (see Figure 8). The results show that the two most flexible models, the ERGM and the IMRM fit the data very well in terms of reproducing the degree distribution. The fit of the IRM and BCD models which are both simple latent cluster models is less good: Both models appear to underestimate the number of nodes with a

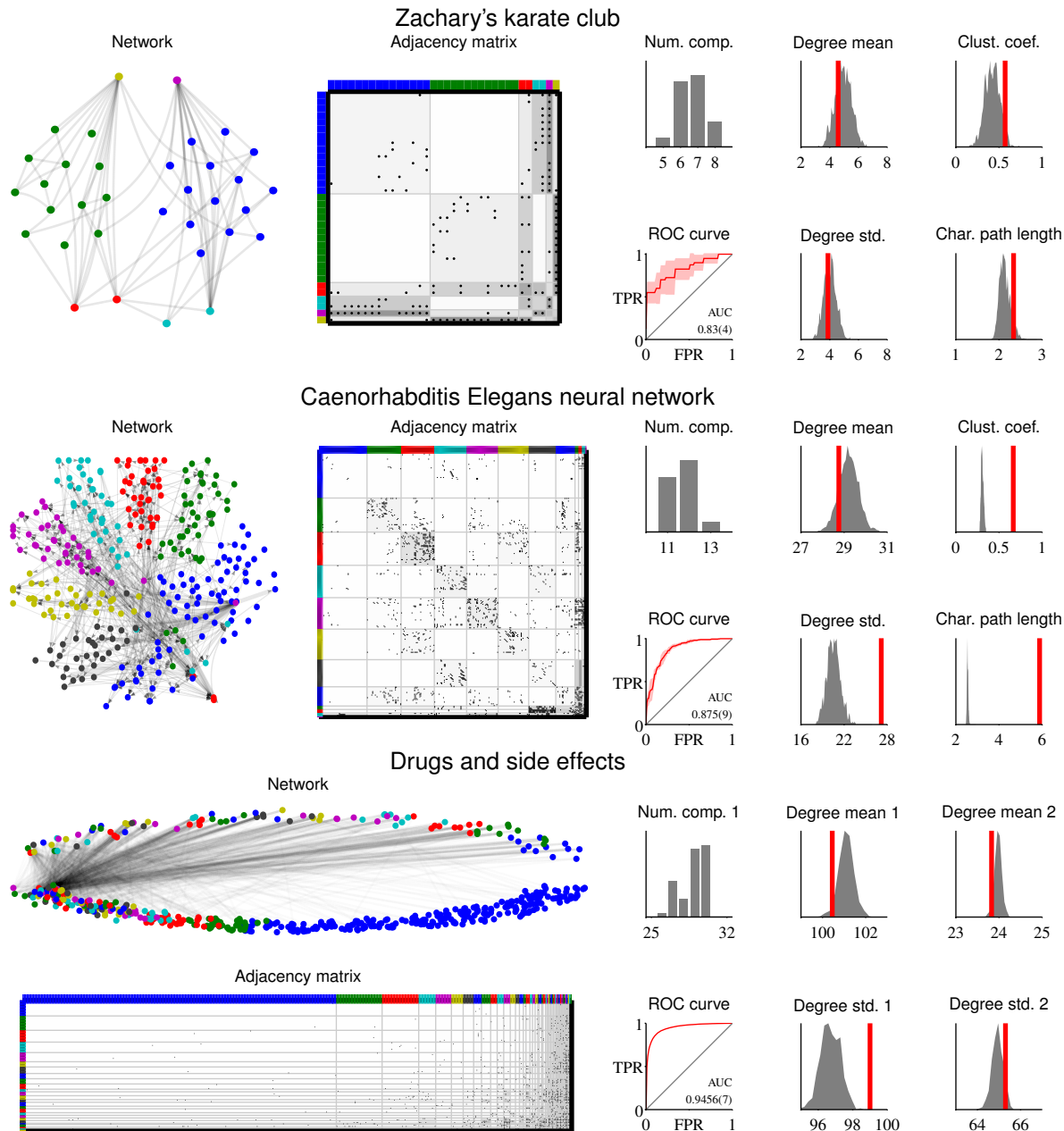


Fig. 7. Infinite relational model analysis of three networks: Social relations in Zachary’s karate club, neural network of *Caenorhabditis Elegans*, and relations between drugs and side effects. Networks are shown as graphs (20 pct. of links shown for *C. Elegans* and 10 pct. shown for Drugs and side effects) as well as adjacency matrix. Posterior distribution of the number of components as well as ROC curve indicating performance on predicting missing links is shown (shaded regions indicate two times the standard deviation on the mean across the separate runs). Posterior predictive distribution of node degree (mean and standard deviation), clustering coefficient, and characteristic path length is shown with vertical lines indicating values for the observed networks.

high degree, i.e., employees interacting with more than 15 colleagues. The ERGMM model on the other hand appears to overestimate the number of nodes with degrees around 15–20.

Next, we compared the models’ predictive performance by evaluating their ability to predict links in the second network (see Figure 8). Here, all models except the ERGM performed on par, suggesting that the inclusion of latent variables in the model is beneficial for this task.

III. REVIEW OF NON-PARAMETRIC BAYESIAN NETWORK MODELS

In the previous section we have discussed the infinite relational model, which is the most simple example of a non-parametric Bayesian latent variable model for complex network. In that model the latent variable is categorical, introducing a clustering of the network nodes; however, many other types of non-parametric Bayesian network models have been proposed in which the latent variables take other forms. Most of these can be classified as latent class, latent feature, or latent hierarchy models. In the following, we review a number of recent non-parametric Bayesian network models:

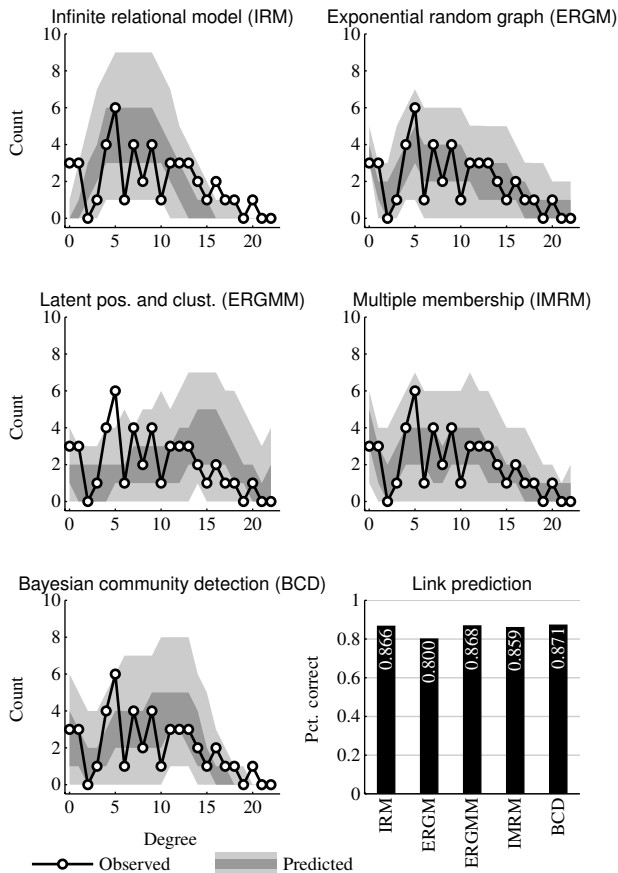


Fig. 8. Comparison of five network models: The plots show the network’s observed degree distribution as well as the posterior predictive 95% and 50% intervals (shaded areas) for each of the models. The plot on the lower right shows the fraction of correctly predicted links/nonlinks when the models are trained on one network and used to predict links in another related network.

We present their generative model and discuss the underlying modeling assumptions, but omit the specific details involved in inference and model checking.

A. Latent class models

In latent class models each node is assumed to belong to one class and the CRP is used a non-parametric distribution of these latent classes. The infinite relational model is the most prominent example of non-parametric latent class models for complex networks. This can be attributed to the fact that the model can capture multiple types of network structures. Contrary to other network modeling approaches such as spectral clustering [55] and modularity [41] groups are defined by how they interact not only internally but also externally. As such, groups are not only defined in terms of their internal properties but in particular by how they interact with the remaining parts of the network. Groups may therefore be defined as having no links between the nodes within the group as illustrated by the fourth (light blue) group of the Zachary Karate Club network in Fig. 7.

Communities in the IRM model can in turn be defined as clusters with high within-cluster density relative to their between-cluster density, interactions between groups can be accounted for by the off-diagonal elements of the ϕ matrix

while hierarchical structures form a structured system of interaction between the elements in the ϕ matrix, see also Fig. 11.

The IRM model can be considered a compression of a complex network into a subgraph formed by ϕ that accounts for the connectivity between the components. If the number of components is the same as the number of vertices of the graph the model will recover the actual graph (when we disregard potential influences of priors) and nothing is learned in terms of structure in networks. As such the IRM model can adjust its complexity, interpolating between the full graph and the Erdős-Rényi graph that corresponds to an IRM model with only one component. Bayesian non-parametrics, i.e. the Chinese Restaurant Process, here admits inference over the hypothesis space encompassing all models between these two extremes in order to find plausible accounts of block structure in networks.

1) *Restrictions on cluster interactions:* Although, the IRM model is very flexible in terms of the structure it is able to account for, specialized non-parametric latent class methods have been proposed that specifically aim at extracting specific types of network structures. These models can be characterized by the restrictions which they impose on the between-class interactions ϕ .

In [21] the ϕ matrix is constrained to only include two parameters, a within-group link probability ρ_w and a between-group link probability ρ_b such that

$$\phi_{k,\ell} = \begin{cases} \rho_w & \text{if } k = \ell, \\ \rho_b & \text{otherwise.} \end{cases} \quad (46)$$

In [36] the within-group link probabilities are individual for each group but between-group probabilities are shared for all combinations of groups,

$$\phi_{k,\ell} = \begin{cases} \rho_\ell & \text{if } k = \ell, \\ \rho_b & \text{otherwise.} \end{cases} \quad (47)$$

2) *Bayesian community detection:* Both of the models mentioned above are inspired by the notion of communities defined as

“the organization of vertices in clusters, with many edges joining vertices of the same cluster and comparatively few edges joining vertices of different clusters.” [11]

This definition is used explicitly in [36] forming the Bayesian Community Detection (BCD) method. The BCD is based on the following non-parametric generative model that strictly enforces community structure by constraining the diagonal elements of the ϕ matrix to be larger than the off-diagonal elements. The generative model for BCD is given by

$$\bar{z} \sim \text{CRP}(A), \quad (48)$$

$$\gamma_k \sim \text{Beta}(\varphi, \varphi), \quad (49)$$

$$\phi_{k,\ell} \sim \begin{cases} \text{Beta}(a, b) & \text{if } k = \ell, \\ \text{BetaInc}(a, b, w_{lm}) & \text{otherwise,} \end{cases} \quad (50)$$

$$\text{where } w_{k,\ell} = \min[\gamma_k \phi_{kk}, \gamma_\ell \phi_{\ell\ell}], \quad (51)$$

$$x_{ij} \sim \text{Bernoulli}(\phi_{z_i z_j}). \quad (52)$$

According to the model the probability of a link between communities k and ℓ is strictly smaller than $w_{k,\ell}$ defined as

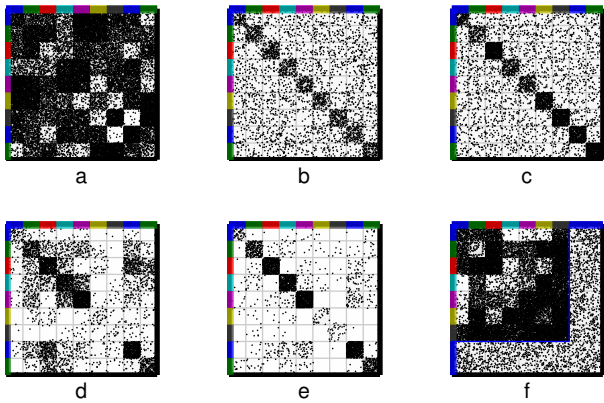


Fig. 9. Examples of existing latent class models. a) The IRM model assumes arbitrary interactions between clusters. b) The model proposed in [21] has two parameters specifying the within group link probability and between group link probability. c) One of the models described in [36] uses an individual within-cluster link probability and same between cluster link probability. d) The BCD model of [36] strictly imposes that within group link probability be larger than between group link probability here given for $\gamma_k = 0.5$ for all the ten clusters. e) BCD using $\gamma_k = 0.1$ for all the ten clusters. f) The SIRM model [23] where nodes are divided into relevant (first 8 clusters modelled by IRM) and irrelevant (last clusters modelled as noise).

the minimum over the two communities of some number γ_k times the within-community probability ϕ_k . This is enforced by generating the between-group probabilities according to an incomplete Beta distribution (BetaInc). The parameters γ_k define a relative gap between link probabilities within and between communities, such that $\gamma_k = 1$ says that there should be fewer links (on average) between than within, and $\gamma_k = 0$ says that no links can be generated from nodes in community k to other communities. The gap parameter γ_k can in turn be learned from data and used to define the extend to which networks are community structured.

3) *Subset infinite relational model*: In [23] the IRM model was extended to handle irrelevant data entries by letting these entries constitute a separate noise cluster forming the subset infinite relational model (SIRM). The generative model for SIRM can be written as

$$r_i \sim \text{Bernoulli}(\lambda), \quad (53)$$

$$\phi_{k,\ell} \sim \text{Beta}(a, b), \quad (54)$$

$$\rho \sim \text{Beta}(c, d), \quad (55)$$

$$\bar{z} \sim \text{CRP}(A), \quad (56)$$

$$x_{ij} \sim \text{Bernoulli}(\phi_{z_i z_j}^{r_i r_j} \rho^{1-r_i r_j}). \quad (57)$$

For each node, the binary variable $r_i = 0$ indicates that the node belongs to the noise cluster. For all pairs of nodes (i, j) not in the noise cluster the model is identical to the IRM model; however, links between pairs of nodes of which at least one is in the noise cluster are generated with a shared probability ρ .

All the above extensions can potentially improve on identification of structure in complex networks by substantially reducing the parameters space of the within and between group interaction matrix ϕ compared to the IRM model. The above extensions are illustrated in Figure 9.

B. Latent feature models

While latent class models restrict each node to belong to one and only one class, latent feature models endow each node with a vector of latent feature values. Exponential random graph models that embed each node in a latent feature space of fixed dimension belong to the class of latent feature models. In contrast, in non-parametric Bayesian latent feature models, the dimensionality of the latent space is learned from data to best fit the observed network. Existing non-parametric latent feature models for networks are based on the Indian Buffet Process (IBP) [16], [17]. Similarly to the CRP the IBP can be derived by starting with a finite model and considering the limit as the number of features goes to infinity. A finite set of $k = (1, \dots, K)$ binary features $z_{i,k}$ with entry 1 if node i possesses feature k and zero otherwise, can be generated according to

$$\pi_k \sim \text{Beta}(\alpha_k, 1), \quad (58)$$

$$z_{i,k} \sim \text{Bernoulli}(\pi_k). \quad (59)$$

Each $z_{i,k}$ is independent of all other assignments conditioned on π_k while the π_k are generated independently [17]. As in the derivation of the CRP, we define $\alpha_1 = \dots = \alpha_K = A/K$ and marginalize over the nuisance parameter π_k , yielding the expression [17]

$$p(\bar{Z}) = \prod_{k=1}^K \int \left(\prod_{i=1}^N p(z_{i,k}) \right) p(\pi_k) d\pi_k \quad (60)$$

$$= \prod_{k=1}^K \frac{\alpha_k \Gamma(n_k + \alpha_k) \Gamma(N - n_k + 1)}{\Gamma(N + 1 + \alpha_k)}. \quad (61)$$

Since again, as in the CRP, the labels of the features are arbitrary, we define an appropriate equivalence class for the binary matrix Z by ordering the columns of the matrix from left to right according to their ‘‘history’’ h in decreasing order. A history h denotes one of the potential 2^N specific combinations of nodes a feature can possess enumerated according to the order of the nodes such that a feature possessed by the n th node contributes by a factor of 2^{N-n} to its history. For example in a network with 3 nodes if only node 1 and 3 possess feature q the feature will have the history enumerated by $h = 2^{3-1} + 2^{3-3} = 5$ which is greater than a feature q' possessed by only node 2 and 3 which has the history $h' = 2^{3-2} + 2^{3-3} = 3$. As a result, feature q will be to the left of feature q' . Features which are not possessed by any nodes have $h = 0$ and are ordered last. Since a permutation of the ordering of the features in Z is inconsequential, we consider the equivalence class of features ordered by their history. The number of equivalent feature matrices can be computed as

$$\frac{K!}{\prod_{h=0}^{2^N-1} K_h!}, \quad (62)$$

where K_h is the number of features with history h and K_0 denotes the number of features that are empty. This equivalence class is used in a similar way as when we considered the distribution over partitions in the CRP. Taking the limit

yields

$$\lim_{K \rightarrow \infty} p(\bar{Z}) = \frac{A^{\bar{K}} \exp(-AH_N)}{\prod_{h=1}^{2^{\bar{N}}-1} K_h!} \times \prod_{k=1}^{\bar{K}} \frac{\Gamma(N - n_k + 1) \Gamma(n_k)}{\Gamma(N + 1)}, \quad (63)$$

where \bar{Z} denotes the left ordered equivalence class, \bar{K} the number of non-empty features and H_N denotes the N th harmonic number [17]. Since this defines a distribution over an infinite size feature matrix of which only a finite subset of the features are used, the construction makes it possible to infer the number of features best suited to model the data. Compactly, we write $\bar{Z} \sim \text{IBP}(A)$.

1) *Latent feature relational model*: In [34] the binary matrix factorization model [32] based on an IBP is considered for network data. The following generative model embodies the latent feature relational model (LFRM)

$$\bar{Z} \sim \text{IBP}(A), \quad (64)$$

$$\phi_{k,\ell} \sim \text{Normal}(0, \sigma_w^2), \quad (65)$$

$$x_{i,j} \sim \text{Bernoulli} \left(\sigma \left[\sum_{k,\ell} z_{i,k} z_{j,\ell} \phi_{k,\ell} \right] \right), \quad (66)$$

where $\sigma[x]$ is a sigmoid function such as the the logit or probit. This model is inspired by the IRM in its parameterization but the model admits the nodes to belong to multiple groups, i.e., for each node to possess multiple features.

2) *Infinite latent attribute model*: In [43] the infinite latent attribute model (ILAM) is proposed in which each of the nodes have a number of associated binary feature, and within each feature the nodes belong to an individual subcluster. The model can be summarized by the generative process,

$$\bar{Z} \sim \text{IBP}(A), \quad (67)$$

$$c^{(m)} \sim \text{CRP}(\gamma), \quad (68)$$

$$\phi_{k,\ell}^{(m)} \sim \text{Normal}(0, \sigma_w^2), \quad (69)$$

$$x_{i,j} \sim \text{Bernoulli} \left(\sigma \left[s + \sum_m z_{i,m} z_{j,m} \phi_{c_i^{(m)} c_j^{(m)}}^{(m)} \right] \right). \quad (70)$$

For each feature m , the nodes that possess that feature are clustered according to a CRP. Here, s is a bias term and $c_i^{(m)}$ is the cluster assignment of the i th node in the m th latent feature.

Both the LFRM and ILAM have been demonstrated to perform better than the IRM on a variety of link-prediction tasks [34], [43]. An important property of these models is that they allow for the membership of nodes in one group to inhibit the probability of linking to nodes in other groups as ϕ may include negative (i.e. antagonistic) elements. This property may indeed be an important reason for the models' superior link prediction performance over IRM.

3) *Infinite multiple-membership relational model*: In [37], [38] the infinite multiple-membership relational model (IMRM) was proposed. Here the probability of observing a link between vertex i and j is generated independently given

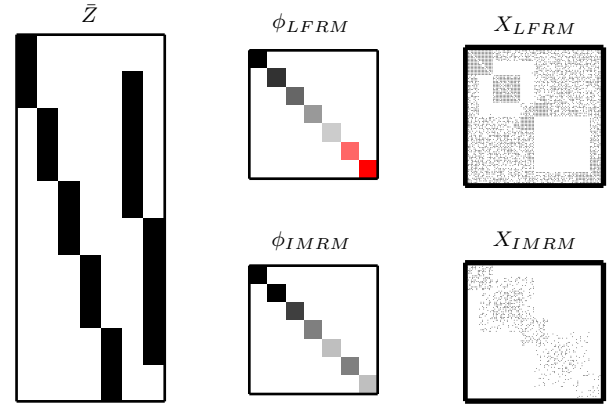


Fig. 10. Illustration of the LFRM and IMRM models. a) The LFRM model assumes arbitrary interactions between latent features, i.e. both positive (given by the interaction within the first five features) and negative (given for the interaction within the last two features). b) The IMRM model assumes features act as independent causes of links such that the link densities monotonically increase by the number of latent features the nodes possess.

the (multiple) groups that vertex i and j belongs to and their interactions ϕ . The generative model for the IMRM is given by

$$\bar{Z} \sim \text{IBP}(A), \quad (71)$$

$$\phi_{k,\ell} \sim \text{Beta}(a, b), \quad (72)$$

$$x_{i,j} \sim \text{Bernoulli} \left(1 - \prod_{k,\ell} (1 - \phi_{k,\ell})^{z_{i,k} z_{j,\ell}} \right). \quad (73)$$

If, for example, node i possesses feature k and node j possess feature ℓ the quantity $\phi_{k,\ell}$ denotes the probability of a link being generated between node i and j on account of that pair of features. The expression $1 - \prod_{k,\ell} (1 - \phi_{k,\ell})^{z_{i,k} z_{j,\ell}}$ defines the probability of observing a links between vertex i and j as the total probability of one or more of the pairs of features possessed by the two nodes to independently generate the link. This construction is referred to as a “noisy or process”. Notably, the IRM model is recovered when nodes belong to one and only one group.

Contrary to the LFRM and ILAM, the IMRM scales computationally in the number of observed links in the network rather than the number of potential links in the network which admits large scale analysis (see [38] for the details). However, scalability comes at the price of not being able to model antagonistic interactions between groups as for LFRM and ILAM. The LFRM and IMRM are illustrated in Figure 10.

4) *Latent factor models*: The IBP is useful for defining non-parametric representations of binary latent variable models and both the LFRM and ILAM can be considered non-parametric latent variable models within the exponential random graph formulation. One approach for model order selection within framework of exponential random graph models is to impose sparse priors. The IBP can here be considered a non-parametric sparse prior for latent variable modeling in general as also proposed for factor analysis in [27]. As such, the IBP works in a similar manner as a slab-and-spike type prior, where a feature is either present or not according to the IBP while its contribution if present can be drawn separately. This can

be used to extend existing sparse latent variable models within the exponential random graph model framework to form non-parametric models.

For instance, a non-parametric version of a latent factor model [20] can be defined by the following generative process using the IBP as a non-parametric sparsity promoting prior.

$$\bar{Z} \sim \text{IBP}(A), \quad (74)$$

$$u_{i,k} \sim \text{Normal}(0, \sigma_u^2), \quad (75)$$

$$x_{i,j} \sim \text{Bernoulli}\left(\sigma \left[\sum_k (z_{i,k} u_{i,k})(z_{j,k} u_{j,k}) \right]\right). \quad (76)$$

C. Latent hierarchical models

Many complex networks are believed to be hierarchically organized such that a latent hierarchy plays an important role in accounting for the structure of the network connectivity [53], [44], [47], [50], [7], [48], [33], [19]. Bayesian non-parametrics can be used to define flexible priors over all conceivable hierarchical structures and from data infer the particular hierarchical structure that is supported by the data in a similar manner as the CRP and IBP is used to infer latent clusters and features respectively.

1) *Hierarchical random graphs*: In [7] the perhaps most simple non-parametric model for hierarchical organization is proposed. This model imposes a uniform prior over all binary trees, which in the following we refer to as UBT. The probability of generating a link between two nodes is defined by a parameter located at the level of their nearest common ancestor in the binary tree. A model for network with N nodes thus has $N - 1$ such parameters associated with each of the internal nodes in the tree. The generative model for the hierarchical random graph is given by

$$T \sim \text{UBT}(N), \quad (77)$$

$$\phi_n \sim \text{Beta}(a, b), \quad (78)$$

$$x_{i,j} \sim \text{Bernoulli}(\phi_{t_{i,j}}), \quad (79)$$

where $t_{i,j}$ denotes the index the nearest common ancestral node of vertex i and j . In [47] a related generative model for binary hierarchies is proposed where each edge in the tree has an associated weight that defined the propensity in which the network complies with the given split.

2) *The Mondrian process*: One way to view the hierarchical random graph models is by first considering the top level of the hierarchy. Here the set of nodes is split into two partitions, and a single parameter is assigned to model the probability of observing a link between nodes in the two partitions. Next, the process continues recursively on the two partitions until each node is in a partition for itself. This framework was generalized and extended to the Mondrian process [48] which can be seen as a distribution over a k -dimensional tree. Used as a prior in a non-parametric Bayesian model of a bipartite network, at the top level the Mondrian process splits either of the two sets of nodes (chosen by random) into two partitions and continues this random bisectioning of the nodes until a stopping criterion is met. Parameters are then assigned to model the probability of links between each of the resulting pairs.

3) *Infinite tree-structured model*: In [19] the uniform prior over binary trees of [7] where replaced by a uniform prior over multifurcating trees and the leafs of the trees rather than terminating at each vertex of the graph terminate at the levels of clusters generated from a CRP based on the following generative model

$$\bar{z} \sim \text{CRP}(A), \quad (80)$$

$$T \sim \text{UT}(K_{\bar{z}}), \quad (81)$$

$$\phi_n \sim \text{Beta}(a, b), \quad (82)$$

$$x_{i,j} \sim \text{Bernoulli}(\phi_{t_{z_i, z_j}}). \quad (83)$$

Here $K_{\bar{z}}$ denotes the number of clusters in \bar{z} and UT defines a uniform prior over multifurcating trees. A benefit of this model is that it can be used to detect the presence of hierarchical structure as it includes the IRM model in its hypothesis space defined by a split at the root of the tree directly into all K clusters (i.e. forming a flat hierarchy). The model of [7] can on the other hand be considered the special case where the CRP only generates singleton clusters while the tree structure is strictly binary. As the leafs terminates in clusters rather than singletons the complexity of the model is in general substantially reduced compared to the models of [7], [47], [48] while the CRP defines the level at which to terminate the tree.

4) *Gibbs fragmentation trees*: In [52] the Gibbs fragmentation tree was used as prior over multifurcating trees terminating at the vertex level of the network according to the following generative model

$$T \sim \text{GFT}(\alpha, \beta), \quad (84)$$

$$\phi_n \sim \text{Beta}(a, b), \quad (85)$$

$$x_{i,j} \sim \text{Bernoulli}(\phi_{t_{z_i, z_j}}). \quad (86)$$

The Gibbs fragmentation tree is closely related to the two parameter nested Chinese restaurant process [2] differing in explicitly accounting for the occurrence in the nested CRP of trivial non-splits. The Gibbs fragmentation tree has several attractive properties. It is i) *exchangeable* in that the distribution does not depend on the labelling of the leaf nodes, ii) *Markovian* in that a subtree of the full tree is in turn a Gibbs fragmentation tree, and iii) *consistent* in that marginalizing over all leafs not considered in the subtree has the same distribution as only considering the Gibbs fragmentation tree of the subtree, see also [31], [52]. Apart from these attractive properties, the Gibbs fragmentation tree gives explicit control of the prior over multifurcating trees by its two parameters α and β , that makes it possible to bias the model toward deep vs. flat hierarchies. The probability of a given Gibbs fragmentation tree can be calculated using a simple recursive formula, see also [31], [52].

D. Modeling side-information

The Bayesian modeling framework readily extends to the modeling of side-information, i.e. exogenous predictors. The side-information can be used either for providing further data in support of the latent structure or directly for modeling the network links.

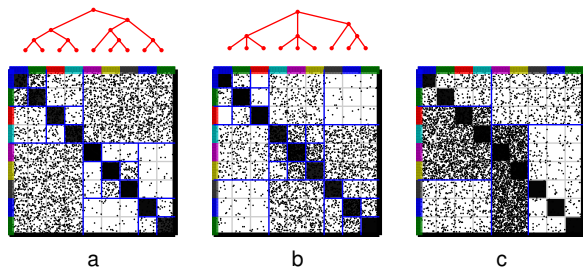


Fig. 11. Example of networks with hierarchical structure. a) In the binary hierarchical relational models [47], [7] link probability parameters are shared in a binary hierarchy. b) In the multifurcating hierarchical model [19], [52] the hierarchy at each level can make an arbitrary number of splits. The model is thereby able to infer whether or not hierarchical structure is present and includes the infinite relational model and the binary hierarchical model as special cases. c) In the Mondrian process [48] the link probabilities are shared in a binary k -dimensional tree, corresponding to a series of axis aligned cuts.

a) *Information about latent structure:* In [26], [58] multiple data sources were used in the IRM model to both model dyadic relationships as well as side information such that the partitioning of the nodes in the graph and the corresponding side-information available were identical.

b) *Information about the network:* Instead of having the side information inform about the latent variables, it can be used to directly model the links. This approach was used in the LFRM model [34] modifying the Bernoulli likelihood function in the LFRM model according to

$$x_{i,j} \sim \text{Bernoulli} \left(\sigma \left[\sum_{k,\ell} z_{i,k} z_{j,\ell} \phi_{k,\ell} + \mathbf{w}^\top \mathbf{r}_{ij} + (\boldsymbol{\gamma}^\top \mathbf{s}_i + a_i) + (\mathbf{v}^\top \mathbf{t}_j + b_j) + c \right] \right), \quad (87)$$

where \mathbf{r}_{ij} denotes a vector of various between node similarities, \mathbf{s}_i and \mathbf{t}_i denotes vectors of features (i.e. side-information) for node i and j respectively. \mathbf{w} , $\boldsymbol{\gamma}$, \mathbf{v} are parameters specifying the effect of the side-information in predicting links and \mathbf{a} , \mathbf{b} specify node specific biases whereas c is a global offset that can be used to define the overall link density. This formulation is closely related to the way in which exogenous predictors are included in the exponential random graph model.

These frameworks readily generalizes to the non-parametric latent class, feature, and hierarchical models described above and makes it possible to include all the available information when modeling complex networks. In particular including side information may improve the identification of latent structure [26], [58] as well as the prediction of links [34].

IV. OUTLOOK

The non-parametric models for complex networks use latent variables to represent structure in networks. As such they can be considered extensions of the traditional exponential random graph models. The non-parametric models here provide a principled framework for inferring the number of latent classes, features, or levels of hierarchy using non-parametric distributions such as the Chinese restaurant process (CRP),

Indian buffet process (IBP) and Gibbs fragmentation trees (GFT). A benefit of these non-parametric models over traditional parametric models of networks is that they can adapt to the complexity of the networks by defining an adaptive parametrization that can account for the needed level of model complexity. In addition, the Bayesian modeling approach admits a principled framework for the statistical modeling of networks and enables to take parameter uncertainty into account. In particular, the Bayesian modeling approach defines a generative process for networks which in turn can be used to simulate graphs, validate the models ability to account for network structure and predict links [34], [38], [43] while Bayesian non-parametrics bring an efficient framework for the inevitable issue of model order selection. The non-parametric Bayesian modeling of complex networks have many important challenges that are yet to be addressed. Below we outline some of these major challenges to point out some avenues of future research.

A. Scalability

Many networks are very large and efficient algorithms for inference in these large systems of millions to billions of nodes and billions to trillions of links will pose important challenges for inferring the parameters of the models. Here it is our firm belief it will be very important to focus on models that grow in complexity by the number of links rather than the sizes of the networks as well as inference procedures that can exploit distributed computing. As such, models will have to be carefully designed in order to be scalable and parallelizable. While the latent class models described all scale by the number of links the LFRM and ILAM models explicitly have to account for both links and non-links which makes them scale poorly compared to the more restricted IMRM model. Thus, flexibility here comes at the price of scalability. In particular, existing models that are scalable do not include the modeling of side-information for the direct modeling of links. Thus, future work should focus on building flexible scalable models for networks.

B. Structure emerging at multiple levels

Network structure is widely believed to emerge at multiple scales [53], [44], [47], [50], [7], [48], [33], [19]. A limitation of latent class models are that they define a given level of resolution in which structure is inferred. Whereas latent feature models can generate features defining clusters at multiple scales [43] this property can be explicitly taken into account by the latent hierarchical models. An important future challenge will be to define models that can operate at multiple scales while efficiently accounting for prominent network structure by combining ideas from the latent hierarchical models with existing latent class and feature models. This includes hierarchical models that explicitly account for community structure and models that allow for the nodes to be part of multiple groups on multiple hierarchical levels.

C. Temporal evolution

Many networks are not static but evolve over time [39], [22], [51]. Rather than modeling snapshots of graphs as

independent, taking into account the timing in which links are generated, when nodes emerge and vanish etc. potentially brings important information about the structure in these systems. To formulate non-parametric Bayesian models that can model networks exhibiting time-varying complexity, such as clusters that emerge and disappear and hierarchies that expand and contract, poses an important future challenge for the modeling of these time evolving networks.

D. Generic modeling tools

As of today non-parametric Bayesian models for complex networks often have to be implemented more or less from scratch in order to accommodate the specific structure of the networks at hand. It will be very useful in the future to develop generic modeling tools in which general non-parametric Bayesian models can be specified including how parameters are tied, various distributions invoked, and side-information incorporated. Publicly available non-parametric Bayesian software tools for complex networks that can well accommodate the needs of researchers modeling complex networks will be essential for these models to fully meet their potentials and be adopted by the many different research communities that today use models and analysis of complex network as an indispensable tool.

E. Testing efficiently multiple hypotheses

Despite the very different origin of complex networks it is widely believed generic properties exist across the domains of these systems. What are the generic properties of networks and how can they be best modelled is an important open problem that is in need of being addressed. Non-parametric Bayesian modeling forms a framework for inferring structure across multiple hypothesis. For example, the IRM model itself encompasses the hypotheses of the Erdős-Rényi random graph (an IRM with a single cluster) as well as the limit of the network itself (an IRM with a cluster for each node). Bayesian non-parametrics can here in general be used to infer structure across multiple hypotheses both including model order as in latent class models, feature representation as in the latent feature models, and types of hierarchies as in the latent hierarchical models.

Non-parametric Bayesian models for complex networks is emerging as a prominent modeling tool that both provides a principled framework for model order selection as well as model validation. As the non-parametric Bayesian models also can give an interpretable account of otherwise complex systems it is our firm belief these models will become essential in order to deepen our understanding of the structure and function of the many networks that surrounds us. There is no doubt the future will bring many new non-parametric Bayesian models for complex networks and that these models will find important new application domains. We hope this paper will facilitate researchers to tap into the power of Bayesian non-parametric modeling of complex networks as outlined in this paper to address the major challenges we face in our effort to understand and be able to predict the behaviors of the many complex systems we are an integral part of.

ACKNOWLEDGEMENTS

This work is funded in part by The Lundbeck Foundation.

REFERENCES

- [1] E. Airoldi, D. Blei, S. Fienberg, and E. Xing, "Mixed membership stochastic blockmodels," *The Journal of Machine Learning Research*, vol. 9, pp. 1981–2014, 2008.
- [2] D. Aldous, "Exchangeability and related topics," in *École d'Été de Probabilités de Saint-Flour XIII–1983*, ser. Lecture Notes in Mathematics. Springer, 1985, pp. 1–198.
- [3] K. Andersen, M. Mørup, H. Siebner, K. Madsen, and L. Hansen, "Identifying modular relations in complex brain networks," *IEEE International Workshop on Machine Learning for Signal Processing*, 2012.
- [4] A. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, Oct. 1999.
- [5] D. Blei and M. Jordan, "Variational inference for Dirichlet process mixtures," *Bayesian Analysis*, vol. 1, no. 1, pp. 121–144, 2006.
- [6] K. Börner, S. Sanyal, and A. Vespignani, "Network science," *Annual review of information science and technology*, vol. 41, no. 1, pp. 537–607, 2007.
- [7] A. Clauset, C. Moore, and M. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, no. 7191, pp. 98–101, 2008.
- [8] N. R. C. Committee on Network Science for Future Army Applications, *Network Science*. The National Academies Press, 2005.
- [9] R. Cross and A. Parker, *The Hidden Power of Social Networks*. Harvard Business School Press, 2004.
- [10] V. Eguiluz, D. Chialvo, G. Cecchi, M. Baliki, and A. Apkarian, "Scale-free brain functional networks," *Phys Rev Lett.*, vol. 94, no. 1, p. 018102, 2005. [Online]. Available: <http://arxiv.org/abs/cond-mat/0309092>
- [11] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75–174, 2010.
- [12] O. Frank and D. Strauss, "Markov graphs," *Journal of the American Statistical Association*, pp. 832–842, 1986.
- [13] A. Gelman, J. Carlin, H. Stern, and D. Rubin, *Bayesian Data Analysis*. Chapman & Hall/CRC, 2004.
- [14] S. Gershman and D. Blei, "A tutorial on Bayesian nonparametric models," *Journal of Mathematical Psychology*, vol. 56, no. 1, pp. 1–12, 2012. [Online]. Available: <http://arxiv.org/abs/1106.2697>
- [15] P. Green and S. Richardson, "Modelling heterogeneity with and without the Dirichlet process," *Scandinavian Journal of Statistics*, vol. 28, no. 2, pp. 355–375, 2001. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1111/1467-9469.00242/abstract>
- [16] T. Griffiths and Z. Ghahramani, "Infinite latent feature models and the indian buffet process," in *Advances in Neural Information Processing Systems*, 2006, pp. 475–482.
- [17] —, "The indian buffet process: An introduction and review," *Journal of Machine Learning Research*, vol. 12, no. April, pp. 1185–1224, 2011. [Online]. Available: <http://www.jmlr.org/papers/volume12/griffiths11a/griffiths11a.pdf>
- [18] T. Herlau, M. Mørup, M. Schmidt, and L. Hansen, "Dense relational modeling," *IEEE International Workshop on Machine Learning for Signal Processing*, 2012.
- [19] —, "Detecting hierarchical structure in networks," *Proceedings of Cognitive Information Processing*, 2012.
- [20] P. Hoff, "Multiplicative latent factor models for description and prediction of social networks," *Computational & Mathematical Organization Theory*, vol. 15, no. 4, pp. 261–272, 2009.
- [21] J. Hofman and C. Wiggins, "Bayesian approach to network modularity," *Physical Review Letters*, vol. 100, no. 25, Jun. 2008.
- [22] K. Ishiguro, T. Iwata, N. Ueda, and J. Tenenbaum, "Dynamic infinite relational model for time-varying relational data analysis," in *Advances in Neural Information Processing Systems*, 2010, pp. 919–927.
- [23] K. Ishiguro, N. Ueda, and H. Sawada, "Subset infinite relational models," *International Conference on Artificial Intelligence and Statistics*, 2012.
- [24] S. Jain and R. Neal, "A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model," *Journal of Computational and Graphical Statistics*, vol. 13, no. 1, pp. 158–182, 2004.
- [25] B. Karrer and M. Newman, "Stochastic blockmodels and community structure in networks," *Phys. Rev. E*, vol. 83, 2011.
- [26] C. Kemp, J. Tenenbaum, T. Griffiths, T. Yamada, and N. Ueda, "Learning systems of concepts with an infinite relational model," in *Proceedings of the national conference on artificial intelligence*, vol. 21. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006, p. 381.

- [27] D. Knowles and Z. Ghahramani, "Infinite sparse factor analysis and infinite independent components analysis," *Independent Component Analysis and Signal Separation*, pp. 381–388, 2007.
- [28] P. Krivitsky and M. Handcock, "Fitting latent cluster models for networks with latentnet," *Journal of Statistical Software*, vol. 24, no. 5, pp. 1–23, 5 2008. [Online]. Available: <http://www.jstatsoft.org/v24/i05>
- [29] M. Kuhn, M. Campillos, I. Letunic, L. Jensen, and P. Bork, "A side effect resource to capture phenotypic effects of drugs," *Molecular systems biology*, vol. 6, no. 1, 2010.
- [30] J. Leskovec, *Dynamics of large networks*. ProQuest, 2008.
- [31] P. McCullagh, J. Pitman, and M. Winkel, "Gibbs fragmentation trees," *Bernoulli*, vol. 14, no. 4, pp. 988–1002, 2008.
- [32] E. Meeds, Z. Ghahramani, and S. Neal, R.M. Roweis, "Modeling dyadic data with binary latent factors," in *Advances in Neural Information Processing Systems*, 2007, pp. 977–984.
- [33] D. Meunier, R. Lambiotte, and E. Bullmore, "Modular and hierarchically modular organization of brain networks," *Frontiers in neuroscience*, vol. 4, 2010. [Online]. Available: <http://dx.doi.org/10.3389/fnins.2010.00200>
- [34] K. Miller, T. Griffiths, and M. Jordan, "Nonparametric latent feature models for link prediction," in *Advances in Neural Information Processing Systems*, 2009, pp. 1276–1284.
- [35] M. Morris, M. Handcock, and D. Hunter, "Specification of exponential-family random graph models: Terms and computational aspects," *Journal of Statistical Software*, vol. 24, no. 4, pp. 1–24, 5 2008. [Online]. Available: <http://www.jstatsoft.org/v24/i04>
- [36] M. Mørup and M. Schmidt, "Bayesian community detection," *Neural Computation*, vol. 24, no. 9, pp. 2434–2456, 2012.
- [37] M. Mørup, M. Schmidt, and L. Hansen, "Infinite multiple membership relational modeling for complex networks," NIPS workshop on Networks across disciplines in theory and applications, 2010.
- [38] —, "Infinite multiple membership relational modeling for complex networks," *IEEE International Workshop on Machine Learning for Signal Processing*, 2011.
- [39] P. Mucha, T. Richardson, K. Macon, M. Porter, and J. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *Science*, vol. 328, no. 5980, pp. 876–878, 2010.
- [40] R. Neal, "Bayesian mixture modeling," *Maximum Entropy and Bayesian Methods: Proceedings of the 11th International Workshop on Maximum Entropy and Bayesian Methods of Statistical Analysis*, pp. 197–211, 1992.
- [41] M. Newman, "Modularity and community structure in networks," *Proc. Natl. Acad. Sci.*, vol. 103, no. 23, pp. 8577–8582, 2006.
- [42] K. Nowicki and T. Snijders, "Estimation and prediction for stochastic blockstructures," *Journal of the American Statistical Association*, vol. 96, no. 455, pp. 1077–1087, 2001.
- [43] K. Palla, D. Knowles, and Z. Ghahramani, "An infinite latent attribute model for network data," *International Conference on Machine Learning*, 2012.
- [44] E. Ravasz, A. Somera, D. Mongru, Z. Oltvai, and A. Barabási, "Hierarchical organization of modularity in metabolic networks," *Science*, vol. 297, no. 5586, pp. 1551–1555, 2002.
- [45] G. Robins, P. Pattison, Y. Kalish, and D. Lusher, "An introduction to exponential random graph (p^*) models for social networks," *Social networks*, vol. 29, no. 2, pp. 173–191, 2007.
- [46] G. Robins, T. Snijders, P. Wang, M. Handcock, and P. Pattison, "Recent developments in exponential random graph (p^*) models for social networks," *Social Networks*, vol. 29, no. 2, pp. 192–215, 2007.
- [47] D. Roy, C. Kemp, V. Mansinghka, and J. Tenenbaum, "Learning annotated hierarchies from relational data," in *Advances in Neural Information Processing Systems*, 2007, p. 1185.
- [48] D. Roy and Y. Teh, "The mondrian process," in *Advances in Neural Information Processing Systems*, 2009, pp. 1377–1384.
- [49] M. Rubinov and O. Sporns, "Complex network measures of brain connectivity: Uses and interpretations," *NeuroImage*, vol. 52, no. 3, pp. 1059–1069, 2010.
- [50] M. Sales-Pardo, R. Guimera, A. Moreira, and L. Amaral, "Extracting the hierarchical organization of complex systems," *Proceedings of the National Academy of Sciences*, vol. 104, no. 39, p. 15224, 2007.
- [51] P. Sarkar, D. Chakrabarti, and M. Jordan, "Nonparametric link prediction in dynamic networks," *ICML*, 2012.
- [52] M. Schmidt, T. Herlau, and M. Mørup, "Nonparametric bayesian models of hierarchical structure in complex networks," *ArXiv*, 2012.
- [53] H. Simon, "The architecture of complexity," *Proceedings of the American philosophical society*, vol. 106, no. 6, pp. 467–482, 1962.
- [54] O. Sporns, *Networks of the Brain*. The MIT Press, 2010.
- [55] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.
- [56] S. Wasserman and G. Robins, "An introduction to random graphs, dependence graphs, and p^* ," *Models and methods in social network analysis*, pp. 148–161, 2005.
- [57] D. Watts and S. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, Jun. 1998.
- [58] Z. Xu, V. Tresp, K. Yu, and H. Kriegel, "Learning infinite hidden relational models," *Uncertainty in Artificial Intelligence*, 2006.
- [59] W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of anthropological research*, pp. 452–473, 1977.