Journal de la Société Française de Statistique
Vol. 160 No. 1 (2019)

Exact integrated completed likelihood maximisation in a stochastic block transition model for dynamic networks

Titre: Maximisation d'un critère exact de classification pour un modèle des blocs latents pour les réseaux dynamiques

Riccardo Rastelli[1]

Abstract: The latent stochastic block model is a flexible and widely used statistical model for the analysis of network data. Extensions of this model to a dynamic context often fail to capture the persistence of edges in contiguous network snapshots. The recently introduced stochastic block transition model addresses precisely this issue, by modelling the probabilities of creating a new edge and of maintaining an edge over time. Using a model-based clustering approach, this paper illustrates a methodology to fit stochastic block transition models under a Bayesian framework. The method relies on a greedy optimisation procedure to maximise the exact integrated completed likelihood. The computational efficiency of the algorithm used makes the methodology scalable and appropriate for the analysis of large network datasets. Crucially, the optimal number of latent groups is automatically selected at no additional computing cost. The efficacy of the method is demonstrated through applications to both artificial and real datasets.

Résumé : Le modèle des blocs latents est un modèle statistique largement utilisé et très flexible. Les extensions de ce modèle à l'analyse des réseaux dynamiques ne peut pas capturer la persistance des liens dans les temps contigus. Le modèle des blocs latents avec des transitions aborde cette question et modélise la propension à créer et à maintenir les liens dans les temps. On présente ici une extension bayésienne de ce modèle et une nouvelle méthodologie pour la classification des nœuds. La méthode repose sur une procédure d'optimisation afin de maximiser un critère exact de classification. L'algorithme est très efficace et rend la méthodologie appropriée pour l'analyse de grands ensembles de données de réseaux. De plus, l'algorithme sélectionne le nombre optimal de groupes latents sans aucun coût supplémentaire. L'efficacité de la méthode est démontrée par des applications à des ensembles de données artificielles et réelles.

Keywords: stochastic block transition models, dynamic networks, integrated completed likelihood, greedy optimisation, clustering
Mots-clés : modèle des blocs latents avec des transitions, réseaux dynamiques, vraisemblance complétée intégrée, algorithme glouton, partitionnement de données
AMS 2000 subject classifications: 90B15, 90C27, 62F15, 62H30, 91D30

1. Introduction

Research on networks has gained significant momentum in the last few decades. In fact, networks can be used to represent observed phenomena in a variety of research areas, including the social sciences, epidemiology, biology, technology and finance. Social networks, which include collaboration networks or proximity networks, are largely available. The analyses of these datasets

[1] School of Mathematics and Statistics, University College Dublin, Dublin, Ireland.
E-mail: riccardo.rastelli@ucd.ie

 ISSN: 2102-6238

pose a number of research challenges, since they typically require scalable and well-thought statistical methodologies.

Most frequently, network data is provided in the form of an adjacency matrix, where each entry x_{ij} characterises the interaction between the nodes i and j. The Stochastic Block Model (SBM), as characterised by Wang and Wong (1987), is a flexible statistical model that can be used to analyse large social networks. In the SBM, the nodes of the network are assigned to latent groups based on their connection preferences: two nodes belonging to the same group are said stochastically equivalent, meaning that they have the same probability of connecting to any other node in the network. This concept generalises the idea of community structure (see Fortunato 2010 and references therein), since disassortative behaviours and other topological structures may also be represented.

The SBM framework effectively defines a clustering problem, where one has to estimate from the data both the nodes' cluster membership variables (*allocations*) and the underlying number of clusters. This may be tackled in a number of ways. One strand of research relies on sampling: this includes the work of Nowicki and Snijders (2001), and the more recent allocation sampler introduced by McDaid et al. (2013). Although the allocation sampler is able to efficiently sample from the posterior distribution of a SBM, a research question that still remains open is how one may summarise in a sensible way the collection of partitions obtained. A different estimation approach for the SBM relies instead on adaptations of the Expectation-Maximisation (EM) algorithm. Daudin et al. (2008) introduce a variational EM in a frequentist setting, and they propose, as model-choice criterion, an adaptation of the Integrated Complete Likelihood (ICL) of Biernacki et al. (2000) to the SBM context. A Bayesian version of the variational EM has been introduced by Latouche et al. (2012), along with an alternative expression for the ICL obtained through the variational approximation. A third approach to fit SBMs has been introduced by Côme and Latouche (2015), and it relies on the optimisation of an exact version of the ICL by means of greedy heuristics. An interesting aspect of this approach is that the clustering and the selection of the number of groups are performed at the same time, hence making the procedure very computationally efficient. A more detailed survey of the methodologies introduced for the SBM can be found in Matias and Robin (2014).

In recent years, a number of works have extended the *static* SBM to the *dynamic* framework, whereby the observed interactions are, in some way, dynamically evolving over time. One type of extension considers the interactions as instantaneous events which may be observed in any given instant. For example, Matias et al. (2018) and Corneli et al. (2018) model these interactions as realised events of non-homogeneous Poisson point processes, where the intensity parameters are determined by the cluster memberships of the corresponding nodes.

This paper belongs to a different strand of literature, where the time dimension is discretised and the observed data can be represented as a collection of adjacency matrices indexed according to their ordering in time. Most of the works following this approach typically introduce a Markov property that creates a temporal dependency between any two contiguous network snapshots. For example, Yang et al. (2011) assume a hidden Markov model where the hidden states are the cluster membership variables of the nodes. In their model, the temporal dependency is captured only through the evolution of the latent allocation variables over time. By contrast, Xu and Hero (2014) characterise the time dependency through a state-space model on the connection probabilities between the SBM blocks. Matias and Miele (2017) consider a more general framework that

 ISSN: 2102-6238

includes the previous two as special cases, proving that the identifiability of these Markovian models may be lost if both cluster membership variables and connectivity parameters are allowed to change over time. They also propose an estimation method that can handle networks with non-binary interactions. Rastelli et al. (2018) also focus on the same type of model, studying the computational efficiency and scalability of the inferential process. Other relevant dynamic extensions of the SBM are introduced in Ishiguro et al. (2010) and Bartolucci et al. (2018).

In many cases, however, the observed dynamic networks tend to be particularly stable over time, or, equivalently, they exhibit a strong temporal dependency. This may have important repercussions, since it ultimately questions whether the temporal dynamics of the models are necessary, or if the static frameworks may be just as effective. The dynamic SBM models mentioned so far are not able to capture these additional temporal dependencies. Xu (2015) addresses exactly this issue, proposing an original model that builds upon a dynamic SBM to include a Markov property on the observed edge values. Differently from the SBM structure, this model clusters the nodes in each time frame according to their propensity to create new edges and maintaining existing ones. Since it directly models the transitions of the edge values, it is called the Stochastic Block Transition Model (SBTM). In the SBTM, the probability of observing an edge depends on whether the same edge was present or absent in the previous time frame, creating a direct dependency between any two contiguous network snapshots. Xu (2015) gives evidence that the SBTM can successfully model the creation and the duration of the interactions, hence being much more flexible than the dynamic SBM of Xu and Hero (2014).

The idea of modelling the persistence of the edges over time has been also proposed in other frameworks: building upon the Latent Position Model of Hoff et al. (2002), Friel et al. (2016) consider a new bipartite dynamic structure that explicitly captures the time persistence using a two-regimes representation. Zhang et al. (2017), instead, study a model similar to the SBTM, obtained through a discretisation of an underlying continuous-time process. They consider a framework that facilitates both the theoretical characterisation of such model and a likelihood-based inferential procedure. Finally, Heaukulani and Ghahramani (2013) propose a type of block model where the evolution over time of the latent allocation of each node is affected by the cluster memberships of its neighbours.

This paper focuses on the SBTM, and it extends the work of Xu (2015) in a number of ways. First, a new Bayesian hierarchical structure for this model is introduced, following ideas similar to those in Rastelli et al. (2018). The generative process proposed allows for non-informative priors and, crucially, it directly captures the fact that nodes may become inactive in certain time intervals. This feature makes the model proposed particularly suitable for the analysis of longitudinal network data, whereby some nodes are added or removed at any time frame. Then, the modelling assumptions are exploited to analytically integrate out (*collapse*) most of the model parameters, as also advocated by Nobile and Fearnside (2007), McDaid et al. (2013) and Côme and Latouche (2015). This collapsing leads to an exact formula for the well known Integrated Completed Likelihood (ICL), which is widely used as an optimality criterion in the statistical analysis of finite mixtures (Biernacki et al., 2000). The exact ICL value obtained is maximised with respect to the allocation variables using a scalable heuristic greedy procedure, which resembles the algorithms described by Côme and Latouche (2015), Wyse et al. (2017), and Rastelli et al. (2018).

An important advantage of the methodology proposed is that the number of latent groups can

 ISSN: 2102-6238

be automatically deduced from the allocation variables at any stage of the optimisation. In fact, to the best of my knowledge, this is currently the only paper addressing the problem of model choice for the SBTM. Another facet of this method is that, due to the optimisation context, it is unaffected by label-switching issues. In addition, the algorithm can exploit the presence of inactive nodes, which further reduces the computational burden.

Taking advantage of the non-informative setting, the methodology is tested as a black-box tool on artificially generated networks, showing that it generally converges to good clustering solutions. The procedure is also compared with other available methods, showing that the introduction of the edge-persistence feature is essential to recover the true partitioning of the nodes and the correct generative mechanism. In addition, a large longitudinal human contact dataset is used to give a demonstration of the procedure, showing that the results obtained are easy to interpret, and that the behaviours of the nodes can be analysed in detail.

Finally, the `R` package `GreedySBTM` accompanies this paper and it provides an implementation of the algorithm described. The package is publicly available on `CRAN` (R Core Team, 2017).

The paper is organised as follows: Sections 2 and 3 illustrate the Bayesian SBTM, Sections 5 and 6 describe the exact ICL approach and the optimisation algorithm, and finally the methodology is applied to simulated and a real dataset in Sections 7 and 8, respectively.

2. The Stochastic Block Transition Model

The statistical model used in this paper is a variation of that introduced by Xu (2015). The differences between the two models are minor and do not affect the principle ideas that motivate the use of the SBTM; nonetheless they are necessary to give integrity to the inferential procedure used in this paper. A more detailed account of the modifications and a comparison with other statistical models for dynamic network data is provided in Section 4.

The observed data consist of a collection of T graphs, where the edges of each of these represent interactions between the corresponding nodes at different times. In each time frame $t = \{1,\dots,T\}$, some of the nodes may be *inactive*, in which case none of their edge values are observed, or they simply do not have any interaction. Since this information may be derived from the collected data, the activity status of the nodes is assumed to be known and observed. Hence, the observed data may be described through two binary cubes $\mathcal{X}$ and $\mathcal{Y}$ of size $N \times N \times T$, which are characterised by:

$$y_{ij}^{(t)} = \begin{cases} 1 & \text{if both nodes } i \text{ and } j \text{ are active at time frame } t, \\ 0 & \text{otherwise;} \end{cases} \tag{1}$$

$$x_{ij}^{(t)} = \begin{cases} 1 & \text{if } y_{ij}^{(t)} = 1 \textbf{ and} \text{ an edge between } i \text{ and } j \text{ exists at time } t, \\ 0 & \text{if } y_{ij}^{(t)} = 0 \textbf{ or} \text{ no edge exists between } i \text{ and } j \text{ at time } t, \end{cases} \tag{2}$$

for all i and j in $\{1,\dots,N\}$ and t in $\{1,\dots,T\}$. Evidently, $\mathcal{Y}$ simply serves as an activity indicator, whereas $\mathcal{X} = \left\{\mathbf{X}^{(1)},\dots,\mathbf{X}^{(T)}\right\}$ corresponds to a collection of canonical adjacency matrices for the observed edge values. These T graphs are assumed to be undirected and without self-edges, hence each of the adjacency matrices is symmetric and has zeros on the diagonal.

In the SBTM, a clustering structure is hypothesised on the nodes of the T observed graphs. Each of the nodes, at each time frame, is characterised by a cluster membership variable taking

 ISSN: 2102-6238

values in the discrete set $\{0,1,\dots,K\}$. The notation $\mathcal{Z} = \left\{z_i^{(t)} : i = 1,\dots,N,\ t = 1,\dots,T\right\}$ is used to denote all such allocations. Also, the equivalent notation $z_{ig}^{(t)} = \mathbb{1}(\{z_i^{(t)} = g\})$ may be used in some equations ($\mathbb{1}$ denotes the indicator function). Note that the vector $\mathbf{z}^{(t)} = \left\{z_1^{(t)},\dots,z_N^{(t)}\right\}$ denotes a partition of $\{1,\dots,N\}$, for every t. Finally, the label zero is reserved for inactive nodes, i.e. $z_i^{(t)} = 0$ iff i is inactive at time t: since the inactive nodes are known, there is no interest in inferring these allocations and hence they are kept fixed throughout. The activity information is encoded in both the allocation variables and in the matrix $\mathcal{Y}$, so that it can be easily denoted at the nodes level and at the edge level, respectively. The relation between the two representations is given by $y_{ij}^{(t)} = z_{i0}^{(t)} z_{j0}^{(t)}$.

The probability that the observed edge indexed by (i,j,t) takes value 1 is defined as:

$$\begin{aligned}\rho_{ij}^{(t)} &= \mathbb{P}\left(x_{ij}^{(t)} = 1 \middle| y_{ij}^{(t)} = 1, y_{ij}^{(t-1)}, x_{ij}^{(t-1)}, z_i^{(t)} = g, z_j^{(t)} = h, \theta_{gh}, P_{gh}, Q_{gh}\right) \\ &= \begin{cases} \theta_{gh} & \text{if } y_{ij}^{(t-1)} = 0 \\ P_{gh} & \text{if } y_{ij}^{(t-1)} = 1 \text{ and } x_{ij}^{(t-1)} = 0 \\ 1 - Q_{gh} & \text{if } y_{ij}^{(t-1)} = 1 \text{ and } x_{ij}^{(t-1)} = 1. \end{cases}\end{aligned} \tag{3}$$

Note that if $t = 1$ then $y_{ij}^{(t-1)} = 0$ for all i and j. The probability of an edge $\rho_{ij}^{(t)}$ is defined by (3) only if $y_{ij}^{(t)} = 1$. In fact, only active nodes may contribute to the likelihood value, and the probability of an edge is simply not defined if at least one of the nodes at its extremities is inactive. Equation (3) essentially characterises the alternation of three regimes:

— A SBM-type of connection probability θ_{gh} is selected whenever there is no information regarding the previous value of the edge considered.
— A SBTM probability P_{gh} is used when it is known that the edge considered had value zero in the previous time frame. The value P_{gh} corresponds to the probability of creating a new edge.
— A SBTM probability Q_{gh} is used when it is known that the edge considered had value one in the previous time frame. The probability of confirming the edge is $1 - Q_{gh}$, hence Q_{gh} may be interpreted as the probability of deleting an existing edge.

These parameters $\left\{\theta_{gh}\right\}_{g,h}$, $\left\{P_{gh}\right\}_{g,h}$ and $\left\{Q_{gh}\right\}_{g,h}$ form the matrices Θ, $\mathbf{P}$ and $\mathbf{Q}$ respectively, which contain the edge probabilities between nodes belonging to any two given groups.

The conditional likelihood of the model reads as follows:

$$\mathcal{L}_{\mathcal{X},\mathcal{Y}}(\mathcal{Z},\Theta,\mathbf{P},\mathbf{Q}) = p(\mathcal{X},\mathcal{Y}|\mathcal{Z},\Theta,\mathbf{P},\mathbf{Q}) = \prod_{t=1}^{T}\prod_{i<j}\left\{\left[\rho_{ij}^{(t)}\right]^{x_{ij}^{(t)}}\left[1-\rho_{ij}^{(t)}\right]^{1-x_{ij}^{(t)}}\right\}^{y_{ij}^{(t)}} \tag{4}$$

which is simply a product of contributions given by Bernoulli variables. Hereafter, the product $\prod_{i<j}$ stands for $\prod_{i=1}^{N-1}\prod_{j=i+1}^{N}$, for brevity. The likelihood function may be reformulated in a more convenient way, taking advantage of the block structure and hence grouping up the terms in (4). In order to do this, the following quantities are needed, for all g and h in $\{1,\dots,K\}$:

$$\eta_{gh} = \sum_{i<j} y_{ij}^{(1)} x_{ij}^{(1)} \lambda_{ij1gh} + \sum_{t>1}\sum_{i<j} y_{ij}^{(t)}\left(1 - y_{ij}^{(t-1)}\right) x_{ij}^{(t)} \lambda_{ijtgh}; \tag{5}$$

 ISSN: 2102-6238

$$\zeta_{gh} = \sum_{i<j} y_{ij}^{(1)} \left(1 - x_{ij}^{(1)}\right) \lambda_{ijtgh} + \sum_{t>1} \sum_{i<j} y_{ij}^{(t)} \left(1 - y_{ij}^{(t-1)}\right) \left(1 - x_{ij}^{(1)}\right) \lambda_{ijtgh}; \tag{6}$$

$$U_{gh}^{uv} = \sum_{t>1} \sum_{i<j} y_{ij}^{(t)} y_{ij}^{(t-1)} \left[1 - \left(u - x_{ij}^{(t-1)}\right)^2\right] \left[1 - \left(v - x_{ij}^{(t)}\right)^2\right] \lambda_{ijtgh}; \tag{7}$$

$$\begin{aligned} \lambda_{ijtgh} &= z_{ig}^{(t)} z_{jh}^{(t)} + z_{ih}^{(t)} z_{jg}^{(t)} - z_{ig}^{(t)} z_{jg}^{(t)} z_{ih}^{(t)} z_{jh}^{(t)} \\ &= \begin{cases} 1 & \text{if } (i,j,t) \text{ refers to an edge between groups } g \text{ and } h; \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{8}$$

The values u and v are in $\{0,\ 1\}$. Also note that for binary values c_1 and c_2:

$$1 - (c_1 - c_2)^2 = \begin{cases} 1 & \text{if } c_1 = c_2; \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

The quantities introduced in (5), (6) and (7) are crucial summaries of the data. They can be interpreted as the number of successes in creating a SBM-edge (η_{gh}), in creating a new edge (U_{gh}^{01}), and deleting an existing edge (U_{gh}^{10}), between a node in group g and one in group h. Similarly, ζ_{gh}, U_{gh}^{00} and U_{gh}^{11} correspond to the number of failures for the same events, respectively. Using these new quantities, the likelihood function factorises as follows:

$$\mathcal{L}_{\mathcal{X},\mathcal{Y}}(\mathcal{Z}, \Theta, \mathbf{P}, \mathbf{Q}) = \prod_{g=1}^{K} \prod_{h=g}^{K} \theta_{gh}^{\eta_{gh}} \left(1 - \theta_{gh}\right)^{\zeta_{gh}} P_{gh}^{U_{gh}^{01}} \left(1 - P_{gh}\right)^{U_{gh}^{00}} Q_{gh}^{U_{gh}^{11}} \left(1 - Q_{gh}\right)^{U_{gh}^{10}}. \tag{10}$$

This likelihood formulation mirrors the one proposed by Zhang et al. (2017): exactly as in SBMs, the presence of blocks simplifies the model structure, and can be exploited to design efficient inferential procedures.

3. Bayesian hierarchical structure

This section introduces a Bayesian hierarchical structure for the SBTM, hence proposing a generative mechanism for the observed data. The prior distributions described here are all conjugate, and, as a special case, they permit a non-informative framework which may be used to nullify the subjective contribution induced by the user.

As concerns the allocations, these are assumed to evolve according to N independent Markov chains on the states $\{0,\ldots,K\}$. The processes share the same transition probability matrix Π, which is hence characterised by:

$$\pi_{gh} = \mathbb{P}\left(z_i^{(t)} = h \middle| z_i^{(t-1)} = g\right), \tag{11}$$

for all $i = 1,\ldots,N$ and $t = 2,\ldots,T$. The initial states are assumed to be drawn from a categorical distribution with probabilities $\alpha_0,\ldots,\alpha_K$ proportional to the aggregated group sizes:

$$\alpha_g \propto \sum_{t=2}^{T} \sum_{i=1}^{N} z_{ig}^{(t)}. \tag{12}$$

 ISSN: 2102-6238

These group proportions approximate the probabilities of the stationary distribution of the Markov chain. Note that the initial allocations are not considered for the calculation of the αs.

The prior probability of a set of allocations $\mathcal{Z}$ may be written as follows:

$$\begin{aligned}\mathbb{P}(\mathcal{Z}|\Pi) &= \mathbb{P}\left(\mathbf{z}^{(1)}\middle|\alpha\right)\prod_{t=2}^{T}\mathbb{P}\left(\mathbf{z}^{(t)}\middle|\mathbf{z}^{(t-1)},\Pi\right)\\ &= \prod_{g=0}^{K}[\alpha_g]^{N_g^{(1)}}\prod_{g=0}^{K}\prod_{h=0}^{K}[\pi_{gh}]^{R_{gh}},\end{aligned} \tag{13}$$

where $N_g^{(t)} = \sum_{i=1}^{N} z_{ig}^{(t)}$ and $R_{gh} = \sum_{t=2}^{T}\sum_{i=1}^{N} z_{ig}^{(t-1)} z_{ih}^{(t)}$, for all t, i and g. Note that, while inactive nodes do not give any contribution to the likelihood in (4), their group membership affects the prior distribution at all times. In fact, the transition probability matrix Π has size $(K+1)\times(K+1)$, and it includes the probabilities for a node to be activated or inactivated. In other words, the nodes are allowed to migrate among $K+1$ groups, however, the first of these groups (labelled by 0) is characterised by fixed connection probabilities which ensure that no edges are created for the nodes inside this group.

The rows of the transition probability matrix Π are assumed to be independent realisations of Dirichlet random vectors:

$$(\pi_{g0},\ldots,\pi_{gK}) \sim Dir(\delta_{g0},\ldots,\delta_{gK}), \tag{14}$$

with δ_{gh} being a user-defined hyperparameter, for all gs and hs.

As concerns the likelihood parameters, the entries of the matrices Θ, $\mathbf{P}$ and $\mathbf{Q}$ all correspond to the success probabilities of Bernoulli random variables: for this reason, independent Beta priors are adopted:

$$\begin{aligned}\theta_{gh} &\sim Beta(\eta_{gh}^0, \zeta_{gh}^0);\\ P_{gh} &\sim Beta(a_{gh}^{\mathbf{P}}, b_{gh}^{\mathbf{P}});\\ Q_{gh} &\sim Beta(a_{gh}^{\mathbf{Q}}, b_{gh}^{\mathbf{Q}}).\end{aligned} \tag{15}$$

The complete set of hyperparameters is $\phi = \left\{\delta_{gh}, \eta_{gh}^0, \zeta_{gh}^0, a_{gh}^{\mathbf{P}}, b_{gh}^{\mathbf{P}}, a_{gh}^{\mathbf{Q}}, b_{gh}^{\mathbf{Q}}\right\}_{g,h}$. These values should be set so that the corresponding prior distributions describe the prior knowledge available on the model parameters. In this paper, non-informative Jeffreys' priors are assumed throughout on all model parameters: this is achieved by setting all the components of ϕ to 0.5.

A graphical representation of the dependencies in the model is shown in Figure 1.

4. Comparisons with other approaches

The dynamic network data analysed in this paper may also be studied using the SBM-based methods introduced by Matias and Miele (2017) and Rastelli et al. (2018). However, the modelling approaches in their works are fundamentally different from the one introduced in this paper. While Matias and Miele (2017) and Rastelli et al. (2018) study an extension of the canonical SBM to a dynamic setting, the SBTM considered here is conceived directly as a framework for networks

 ISSN: 2102-6238

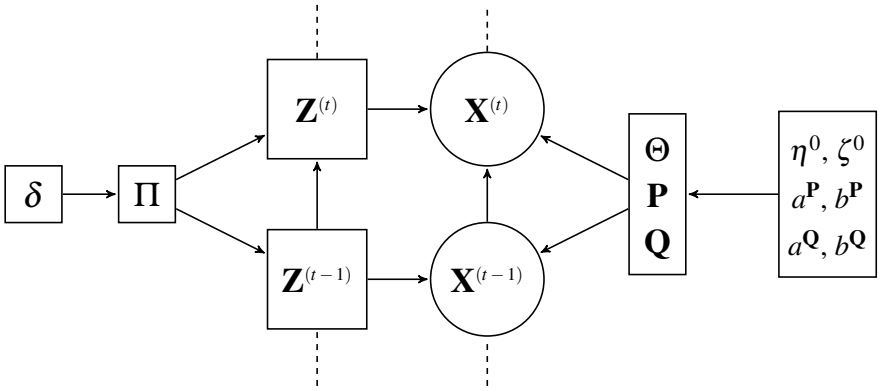

FIGURE 1. *Graphical model for the SBTM described.*

evolving over time, as it models the transitions of the edge values. Relating the SBTM mechanism to the original definition of SBM from Holland et al. (1983): at any given time, two nodes allocated to the same group are said stochastically equivalent, in the sense that they have the same probabilities of creating (or deleting) edges towards any other given node. Furthermore, the method introduced in this paper can also deal with inactive nodes: as it will be shown in the applications, this feature becomes crucial as a means to save computing time, but also to obtain a more reasonable generative process for longitudinal data.

The Bayesian hierarchical structure introduced in the previous section modifies and extends the SBTM model proposed by Xu (2015). In the Bayesian SBTM, the evolution of the allocation variables over time is modelled with a Markov process, imitating the approaches of Yang et al. (2011), Matias and Miele (2017) and Rastelli et al. (2018). This type of specification creates an additional temporal dependency, and it ultimately permits an assessment of the stability of the network. The same feature also distinguishes the Bayesian SBTM from the model analysed by Zhang et al. (2017), where, by contrast, the authors do not let the allocations change over time.

In Xu (2015), the author introduces the scaling factors: these are extra model parameters that can be used to tune the transition probabilities $\mathbf{P}$ and $\mathbf{Q}$. The author imposes constraints on these scaling factors to guarantee that each network snapshot marginally follows a canonical SBM structure with connection probabilities Θ. This is an interesting property for the SBTM to satisfy, since it forces the transition probabilities $\mathbf{P}$ and $\mathbf{Q}$ to yield a SBM structure coinciding with Θ. In other words, the SBM characterised by Θ may be seen as the asymptotic structure the generative model of the SBTM converges to.

On the other hand, it should be noted that the scaling factors must be estimated from the data, hence they make the inferential task more problematic and less tractable. The present paper does not take advantage of the scaled representation: this leads to a simpler structure which allows the proposed estimation method to be efficient and properly defined. In fact, it may not be possible to define the same methodology using the scaling factors introduced by Xu (2015). Therefore, in the Bayesian SBTM, the marginal SBM structure on the network snapshots is inevitably lost.

 ISSN: 2102-6238

5. Exact Integrated Completed Likelihood

The Integrated Completed Likelihood (ICL), first introduced in Biernacki et al. (2000), is a model-based clustering criterion usually used to estimate the number of clusters in finite mixture models. In the dynamic network context addressed in this paper, the exact ICL corresponds to the following value:

$$\mathcal{ICL}_{ex} = \mathbb{P}(\mathcal{X}, \mathcal{Y}, \mathcal{Z} | \phi, K). \tag{16}$$

Since the data $(\mathcal{X}, \mathcal{Y})$ is fixed, the $\mathcal{ICL}_{ex}$ index is also equivalent to the marginal posterior for the allocations:

$$\mathcal{ICL}_{ex} \propto \mathbb{P}(\mathcal{Z} | \mathcal{X}, \mathcal{Y}, \phi, K). \tag{17}$$

In other words, the $\mathcal{ICL}_{ex}$ value can be obtained by analytically integrating out all of the model parameters from the full posterior distribution $\pi(\mathcal{Z}, \Theta, \mathbf{P}, \mathbf{Q}, \Pi | \mathcal{X}, \mathcal{Y}, \phi, K)$. In fact, thanks to the conjugacy of the prior distributions, such integration is analytically possible, and the exact ICL results as follows:

$$\begin{aligned}
\mathcal{ICL}_{ex} \propto & \prod_{g=0}^{K} \left\{ [\alpha_g]^{N_g^1} \cdot \frac{\Gamma\left(\sum_{h=0}^{K} \delta_{gh}\right)}{\Gamma\left(\sum_{h=0}^{K} \delta_{gh} + \sum_{h=0}^{K} R_{gh}\right)} \prod_{h=0}^{K} \frac{\Gamma\left(\delta_{gh} + R_{gh}\right)}{\Gamma\left(\delta_{gh}\right)} \right\} \\
& \cdot \prod_{g=1}^{K} \prod_{h=g}^{K} \left\{ \frac{\Gamma\left(\eta_{gh}^0 + \zeta_{gh}^0\right)}{\Gamma\left(\eta_{gh}^0\right)\Gamma\left(\zeta_{gh}^0\right)} \cdot \frac{\Gamma\left(\eta_{gh}^0 + \eta_{gh}\right)\Gamma\left(\zeta_{gh}^0 + \zeta_{gh}\right)}{\Gamma\left(\eta_{gh}^0 + \eta_{gh} + \zeta_{gh}^0 + \zeta_{gh}\right)} \right\} \\
& \cdot \prod_{g=1}^{K} \prod_{h=g}^{K} \left\{ \frac{\Gamma\left(a_{gh}^{\mathbf{P}} + b_{gh}^{\mathbf{P}}\right)}{\Gamma\left(a_{gh}^{\mathbf{P}}\right)\Gamma\left(b_{gh}^{\mathbf{P}}\right)} \cdot \frac{\Gamma\left(a_{gh}^{\mathbf{P}} + U_{gh}^{01}\right)\Gamma\left(b_{gh}^{\mathbf{P}} + U_{gh}^{00}\right)}{\Gamma\left(a_{gh}^{\mathbf{P}} + U_{gh}^{01} + b_{gh}^{\mathbf{P}} + U_{gh}^{00}\right)} \right\} \\
& \cdot \prod_{g=1}^{K} \prod_{h=g}^{K} \left\{ \frac{\Gamma\left(a_{gh}^{\mathbf{Q}} + b_{gh}^{\mathbf{Q}}\right)}{\Gamma\left(a_{gh}^{\mathbf{Q}}\right)\Gamma\left(b_{gh}^{\mathbf{Q}}\right)} \cdot \frac{\Gamma\left(a_{gh}^{\mathbf{Q}} + U_{gh}^{10}\right)\Gamma\left(b_{gh}^{\mathbf{Q}} + U_{gh}^{11}\right)}{\Gamma\left(a_{gh}^{\mathbf{Q}} + U_{gh}^{10} + b_{gh}^{\mathbf{Q}} + U_{gh}^{11}\right)} \right\}.
\end{aligned} \tag{18}$$

6. Greedy optimisation

The only unknown quantities in (17) and (18) are the allocations $\mathcal{Z}$. In fact, for a given clustering configuration $\mathcal{Z}$, the corresponding value of K may be automatically deduced by counting the number of non empty groups. Hence, an optimisation problem can be set up to find the allocations $\hat{\mathcal{Z}}$ maximising $\log(\mathcal{ICL}_{ex})$, by searching in the space of all possible clustering configurations.

This discrete optimisation problem is known to be NP-hard, and it can be solved exactly only through enumeration, which is impractical even for very small datasets. However, heuristic greedy algorithms have been shown to perform well in similar types of clustering problems: the procedure proposed here follows ideas similar to those of Karrer and Newman (2011), Côme and Latouche (2015), Bertoletti et al. (2015), and Rastelli et al. (2018).

First, a maximum number of groups allowed, denoted K_{up}, is fixed. For small datasets this may be set to NT, however, for larger networks, a smaller value may be chosen to reduce the computing time. Then, an initial clustering configuration with K_{up} groups is generated. This may be created at random, or following initialisation methods based on the k-means algorithm, such as

 ISSN: 2102-6238

those described in Matias and Miele (2017) or Rastelli et al. (2018). At this point the main routine of the algorithm starts, where an active node (t,i) is selected, and its allocation is updated. For the update, all possible moves to groups $1,\dots,K_{up}$ are tested, and, finally, the change yielding the best increase in the objective function is performed. Note that the label zero remains exclusive of inactive nodes, therefore, since activity information is observed, the set of nodes allocated to group 0 remains the same throughout. This process continues in a loop until no further increase is possible. After convergence, hierarchical clustering updates are attempted on the final solution obtained, following exactly the same procedure described in Côme and Latouche (2015) and Rastelli et al. (2018). The computing time demanded by this last step is usually negligible, yet it may improve the final solution by merging together some of the groups. The pseudocode for the algorithm (called `GreedyICL`) is provided in Algorithm 1. Note that, in the pseudocode, $\ell_{(t,i)\to\hat{g}}$ denotes the value corresponding to the current allocations with node (t,i) moved to group g.

Algorithm 1 `GreedyICL`

Set K_{up} and initialise the allocations $\mathcal{Z}$.
Evaluate the objective function and set $\ell = \ell_{stop} = \log(\mathcal{ICL}_{ex})$.
Set $stop = false$.
while $!stop$ **do**
  Set $\mathcal{U} = \left\{(t,i) : z_i^{(t)} \neq 0,\ t = 1,\dots,T,\ i = 1,\dots,N\right\}$.
  Shuffle the elements of $\mathcal{U}$.
  while $\mathcal{U}$ is not empty **do**
    $(t,i) = pop(\mathcal{U})$.
    $\hat{g} = \arg\max_{g=1,2,\dots,K_{up}} \ell_{(t,i)\to g}$.
    $\ell = \ell_{(t,i)\to\hat{g}}$.
    $z_i^{(t)} = \hat{g}$.
  end while
  if $\ell \leq \ell_{stop}$ **then** $stop = true$ **else** $\ell_{stop} = \ell$.
  end if
end while
Return $\mathcal{Z}$ and ℓ.

Both `GreedyICL` and the final merge procedure only involve greedy updates, so they can only increase the objective function value. However, there is no guarantee that the final solution will correspond to a global optimum of $\log(\mathcal{ICL}_{ex})$: for this reason, several restarts of the whole procedure may be beneficial to avoid local optima.

From the algorithmic point of view, one main advantage of these greedy updates is their scalability: the increase in the objective function for each move can be evaluated very efficiently. Furthermore, convergence is usually reached after very few updates of each of the allocations. More detailed explanations regarding the computational savings are provided for example in Côme and Latouche (2015) and Rastelli et al. (2018) and references therein.

As already pointed out, the number of groups can be deduced from the allocation variables at any stage. This makes `GreedyICL` particularly appealing, because, in one single algorithmic framework, one can obtain an estimate of the best K, according to the exact ICL criterion. In fact, an advantage of the exact ICL approaches of Bertoletti et al. (2015), Côme and Latouche (2015), Wyse et al. (2017), and Rastelli et al. (2018) is that they do not rely on a grid search over all possible K values, which becomes impractical if the number of groups is large.

 ISSN: 2102-6238

7. Simulations

In this section, artificial data is used to validate the methodology described in this paper. All the experiments have been run on a Debian machine with 8 cores at 2.2 GHz.

7.1. Simulation study 1

In the first simulated setting considered, the number of time frames is $T = 20$, whereas two scenarios are possible for the number of nodes: $N = 50$ or $N = 250$. The artificial networks are generated using the hierarchical structure described in Section 3, with $K = 3$ and the hyperparameters all set to 0.5. 100 networks are independently generated, and the methodology described in Section 6 is run on each of them, once for each K_{up} in $\{10, 20, 30\}$. Figure 2 shows the objective function values for the true allocations and for the estimated clustering after each of the steps of the optimisation. For most datasets, and for both small and large networks, the final

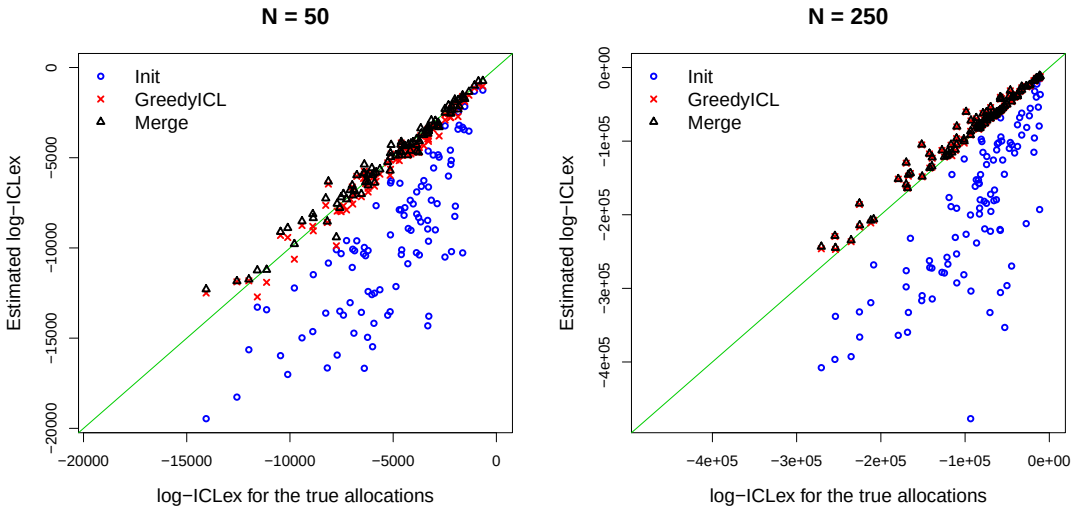

FIGURE 2. ***Simulation study 1***. $\log(\mathcal{ICL}_{ex})$ *values for the true allocations (on the horizontal axes) and for the best estimated clustering across all* K_{up} *values (on the vertical axes). The blue circles correspond to the values obtained after the initialisation using k-means, the red x-marks to those obtained after the* `GreedyICL` *described in 1 and the black triangles correspond to the values obtained at the end of the merging procedure.*

solution achieves better $\log(\mathcal{ICL}_{ex})$ values than the true clustering, suggesting that the method converges to excellent clustering solutions, in the exact ICL sense. Also, the increase granted by the `GreedyICL` step is generally much larger than that given by the final merge step.

Figure 3 focuses on the assessment of the clustering solutions obtained. The Normalised Mutual Information (NMI) criterion (Strehl and Ghosh, 2002) is used to compare each estimated partition to its corresponding true counterpart. The plot on the left panel of Figure 3 shows a very high level of agreement, particularly for the larger datasets. Note that this criterion is normalised, hence it should not be affected by the value of N.

 ISSN: 2102-6238

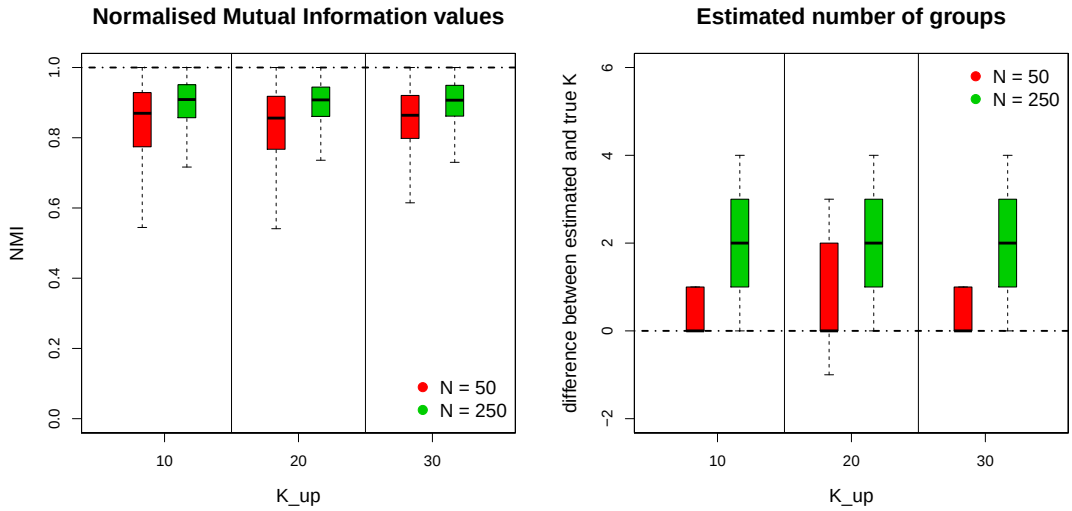

FIGURE 3. ***Simulation study 1***. *The left panel shows the Normalised Mutual Information (NMI) index between the true clustering and the estimated clustering, for all combinations of K_{up} and N. The right panel shows the difference between the estimated and the true number of groups. In both panels, the boxplots are created using the values obtained across all generated datasets and time frames (the index is in fact evaluated for each $t = \{1, \dots, T\}$ independently).*

The right panel of Figure 3 focuses instead on the estimation of the number of groups. Since the transition probabilities Π and the allocations are both randomly generated for each dataset, it may be possible that the true number of groups becomes smaller than 3. Hence, the plot shows the difference between the estimated and the true K for each dataset and each of the time frames. It seems that the methodology tends towards an overestimation, at least in the larger datasets. This may be due to overfitting, in that the exact ICL criterion does not impose a sufficiently strong penalisation on the number of groups. Otherwise, it may be due to the greedy algorithm failing to converge to a better solution with fewer groups: this issue may potentially be overcome by restarting the algorithm a number of times with different initial configurations.

Finally, both plots highlight that the choice of K_{up} does not affect the performance by much. Note that a smaller K_{up} reduces the computing time, yet the optimal partition can only be found if K_{up} is greater than the optimal number of groups. Hence, in general, the higher K_{up} the better; nevertheless, smaller K_{up} values may be used to speed up the algorithm or to force it to return a solution with fewer groups.

7.2. Simulation study 2

The second simulated setting aims at highlighting that the model proposed is fundamentally different from other available methods, such as that of Matias and Miele (2017). In fact, this section shows that the method proposed in this paper can achieve better performances in datasets that exhibit strong time dependencies and persistence of edges or non-edges.

In this simulated setting, the number of time frames is again set to $T = 20$, whereas the number of nodes is set to 50. The three latent groups considered are characterised by the following edge

 ISSN: 2102-6238

probabilities:

$$\Theta = \begin{pmatrix} 0.9 & 0.1 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.1 & 0.9 \end{pmatrix}, \qquad \mathbf{P} = \begin{pmatrix} 0.9 & 0.1 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.1 & 0.1 \end{pmatrix}, \qquad \mathbf{Q} = \begin{pmatrix} 0.1 & 0.1 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.1 & 0.9 \end{pmatrix}. \quad (19)$$

The SBM-type probabilities simply follow a community structure with high within-groups probabilities. As concerns **P** and **Q**, if two nodes belong to the same group, the situation can be summarised as follows: in group 1 they tend to create edges frequently, but they seldom delete them; in group 2 they tend to create and delete edges very frequently; whereas in group 3 they delete edges frequently but create them seldom. Whenever the two nodes are in two different groups, they tend not to change the current state of their interaction. Regarding the transition probabilities, the nodes remain in the same group with probability 0.8 or can move to another group completely at random. However, the group of inactive nodes is non existent in this case, in that nodes cannot ever become inactive.

Using this parameter configuration, 500 networks are generated at random. On each of these, the `GreedyICL` procedure is run once with $K_{up} = 10$, and the `dynsbm` procedure of Matias and Miele (2017) is run once for every choice of $K = 1, \ldots, 6$. While the `GreedyICL` method chooses the number of groups in one run, in `dynsbm` only the run corresponding to the highest approximate ICL is retained as optimal, as advised in the related paper.

Figure 4 illustrates the results obtained in this experiment. Similarly to the previous simulation

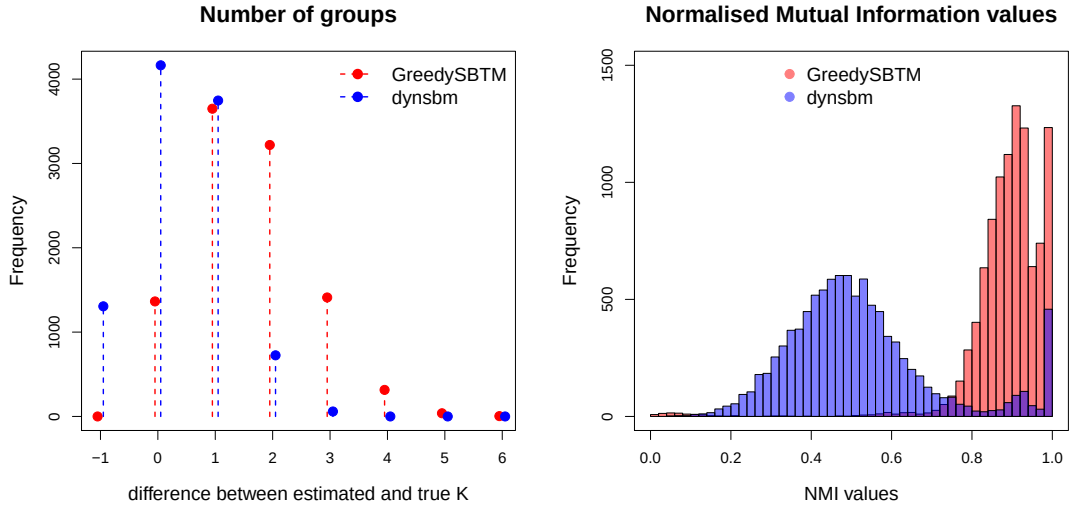

FIGURE 4. ***Simulation study 2***. *The left panel shows the aggregated number of time frames and datasets corresponding to the value of the difference between the estimated and true K, for the proposed greedy approach (in red) and for the algorithm* `dynsbm` *of Matias and Miele (2017) (in blue). The right panel shows the NMI indexes between the true and the estimated clusterings. The NMI values are evaluated for each time frame and each of the datasets independently.*

study, the `GreedyICL` tends towards an overestimation of the number of groups (see left panel of the figure), in that the correct value $K = 3$ is properly estimated in about 15% of the cases. The `dynsbm` seems to achieve better performance in this task. However, as documented in the plot

 ISSN: 2102-6238

on the right panel, this event is rather fortuitous, since the vast majority of the optimal solutions of `dynsbm` are fundamentally wrong, and they do not capture the essence of the generative mechanism of the data. In other words, the `dynsbm` method provides a different view on the data, which is not necessarily appropriate when high persistence of the edges is present.

The average computing times across all datasets for the two algorithms were 0.083 and 14.959 seconds, for `GreedySBTM` and `dynsbm` respectively.

7.3. Simulation study 3

The purpose of this study is to assess whether the hyperparameters of the model can have an effect on the final results. The collection of hyperparameters ϕ is separated in the sets $\left\{\delta_{gh}\right\}_{g,h}$, where $\delta = \delta_{gh}$ for all g and h, and $\left\{\eta^0_{gh}, \zeta^0_{gh}, a^{\mathbf{P}}_{gh}, b^{\mathbf{P}}_{gh}, a^{\mathbf{Q}}_{gh}, b^{\mathbf{Q}}_{gh}\right\}_{g,h}$, characterising the priors on the connection probabilities. If prior information on the parameters is available, it should be encoded in the model through the hyperparameters. However, as pointed out in a previous section, a non-informative setting is available, as fixing all of the hyperparameters to 0.5 corresponds to Jeffrey's priors. This setting is used throughout this paper with the exception of this section. More in general, the non-informative setting should be used whenever prior information on the parameters is not available.

Here, four different scenarios are considered, each corresponding to a type of informative prior distributions.

- **Scenario 1**: the hyperparameters are all set to 0.05. For the *Beta*-distributed connection probabilities, this implies a larger variance and thus more well-separated blocks. As concerns the *Dirichlet*-distributed transition probabilities, values closer to 0 and 1 are favoured more, hence leading to a more deterministic framework where migrations between groups follow similar patterns.
- **Scenario 2**: $\left\{\eta^0_{gh}, \zeta^0_{gh}, a^{\mathbf{P}}_{gh}, b^{\mathbf{P}}_{gh}, a^{\mathbf{Q}}_{gh}, b^{\mathbf{Q}}_{gh}\right\}_{g,h}$ are all set to 0.05, whereas $\delta = 5$. Differently from the previous scenario, this prior on the transition probabilities favours a more entropic structure, where migrations do not follow specific patterns.
- **Scenario 3**: $\delta = 0.05$ and the hyperparameters for the connection probabilities are all set to 5. In this case, migrations are expected to be rather deterministic, and groups not well separated.
- **Scenario 4**: all hyperparameters are fixed to 5, supporting a high-entropy structure for the transitions, and similar connection probabilities between blocks.

The data observations are generated following the same setup as in the first large simulation study. Hence: $T = 20$, $N = 250$, $K = 3$, all the hyperparameters are fixed to 0.5, and the number of generated datasets is 100. The true connection probabilities and transition probabilities are randomly generated for each dataset using the hierarchical structure described in Section 3. On each of these 100 datasets, the `GreedyICL` procedure is run once with $K_{up} = 20$. Table 1 shows the proportion of datasets for each value of the difference between the estimated number of groups and the true underlying K. The main finding, here, is that the estimated number of groups does not seem to be particularly sensitive to the choice of hyperparameters, since the results obtained are rather similar for all the scenarios considered. As in the first simulation study, the number of groups is most often overestimated by 1.

 ISSN: 2102-6238

	Difference between estimated K and true K										
Scenario	-2	-1	0	1	2	3	4	5	6	7	8
1	0	0.01	0.21	0.31	0.19	0.10	0.07	0.04	0.04	0.02	0
2	0	0.00	0.17	0.31	0.19	0.11	0.07	0.07	0.04	0.02	0
3	0	0.02	0.28	0.37	0.16	0.13	0.03	0.01	0.00	0.00	0
4	0	0.01	0.22	0.39	0.19	0.13	0.02	0.00	0.00	0.00	0

TABLE 1. ***Simulation study 3***. *For each value of the difference between the estimated and true K, the entries of the table show a proportion indicating the aggregated number of datasets and time frames where the value was obtained.*

Table 2 illustrates, instead, some summaries of the corresponding NMI values. Also in this

	Scenario			
	1	2	3	4
min	0.61	0.60	0.06	0.45
Q1	0.87	0.87	0.88	0.88
median	0.94	0.93	0.94	0.93
Q3	0.97	0.97	0.99	0.97
max	1	1	1	1

TABLE 2. ***Simulation study 3***. *For each scenario, the table reports summary statistics for the collection of NMI values obtained for each dataset and time frame.*

regard, the different prior settings do not have a relevant effect on the final results. This essentially demonstrates that, even in small network datasets, the likelihood part of the model plays a crucial role in determining the results.

It should be noted that, differently from the sparse finite mixture models studied by Rousseau and Mengersen (2011) and Malsiner-Walli et al. (2016), the modification of the hyperparameter δ does not necessarily affect the shrinkage properties for the prior on the number of groups; i.e. the regulation of the estimated K through δ seems not possible for the model considered here.

7.4. Simulation study 4

The last simulation study considers a particular structure resembling that of the real dataset which is analysed in the following section. This study considers 100 large artificial datasets where a high proportion of allocations are known to be 0, hence the nodes generally exhibit high levels of inactivity. The data observations are generated using $T = 1000$, $N = 100$, $K = 2$, and the connection probabilities are randomly sampled using hyperparameters equal to 0.2. As concerns the allocations, these are sampled for each node at each time frame independently from the set $\{0,1,2\}$, with probabilities $\{0.8,0.1,0.1\}$, respectively. On each of these 100 datasets, the `GreedyICL` procedure is run once with $K_{up} = 20$.

The left panel of Figure 5 illustrates the distribution of the difference between the estimated number of groups and its true value. The peak at 0 signals that the correct number of groups is recovered in a large number of datasets and time frames. The right panel of the same figure shows instead the NMI indexes, measuring the agreement between the true clusterings and the estimated ones. This plot shows that in the vast majority of cases a NMI of 1 (or nearly 1) is achieved.

 ISSN: 2102-6238

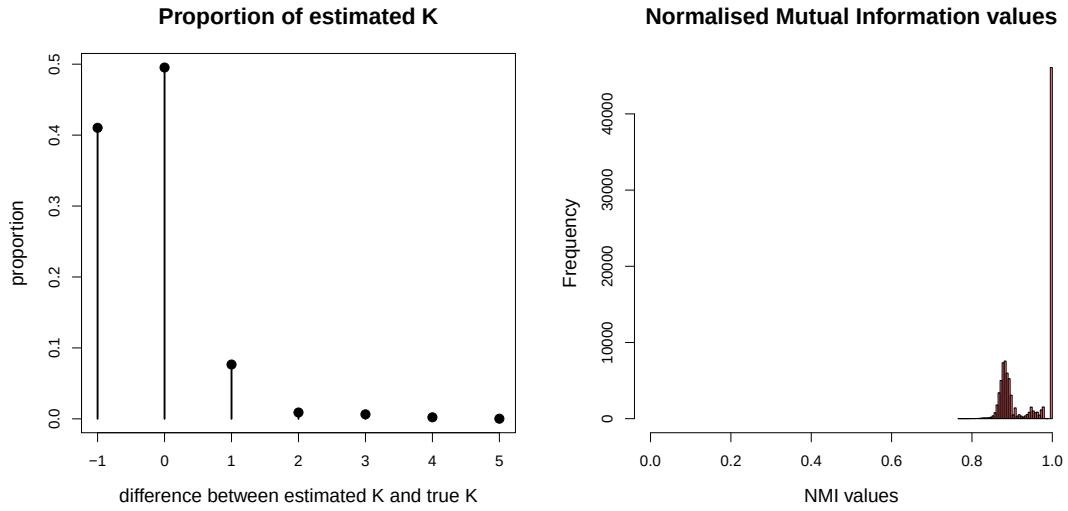

FIGURE 5. ***Simulation study 4***. *The left panel shows the overall proportion of datasets and time frames for each value of the difference between the estimated and true number of groups. The plot exhibits a peak at the correct value indicating that the correct model is selected in most cases. The right panel shows instead the NMI indexes between the true clusterings and the estimated clusterings, evaluated for every time frame and dataset.*

This signals that, even if the two partitions may have a different number of groups, the clustering solutions that they denote are very similar to one another.

8. Reality Mining dataset

The Reality Mining experiment was performed in 2004 as part of the Reality Commons project. The data was collected and first described by Eagle and Pentland (2006), and it includes human contacts between Massachusetts Institute of Technology (MIT) students, from 14 September 2004 to 5 May 2005. KONECT (the Koblenz Network Collection) provides a public version of a proximity network extracted from the Reality Mining data. The dataset describes proximity interactions of students through a list of undirected edges and their corresponding time stamp. The number of nodes having at least one interaction is $N = 96$, and the total number of interactions is 1,086,404. The 9 months were discretised in $T = 1392$ time frames of 4 hours each. Then, an adjacency cube $\mathcal{X}$ of size $N \times N \times T$ was created as follows:

$$x_{ij}^{(t)} = \begin{cases} 1 & \text{if nodes } i \text{ and } j \text{ had at least one interaction between } t-1 \text{ and } t, \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

The distribution of edges in the new representation is shown in Figure 6. The nodes were considered inactive in all of the time frames where they had zero interactions: as a consequence, approximately 83% of the allocation variables were set to zero, overall. The algorithm was then run once with $K_{up} = 20$, using k-means initialisation, and it converged after 15 iterations and 115 seconds.

 ISSN: 2102-6238

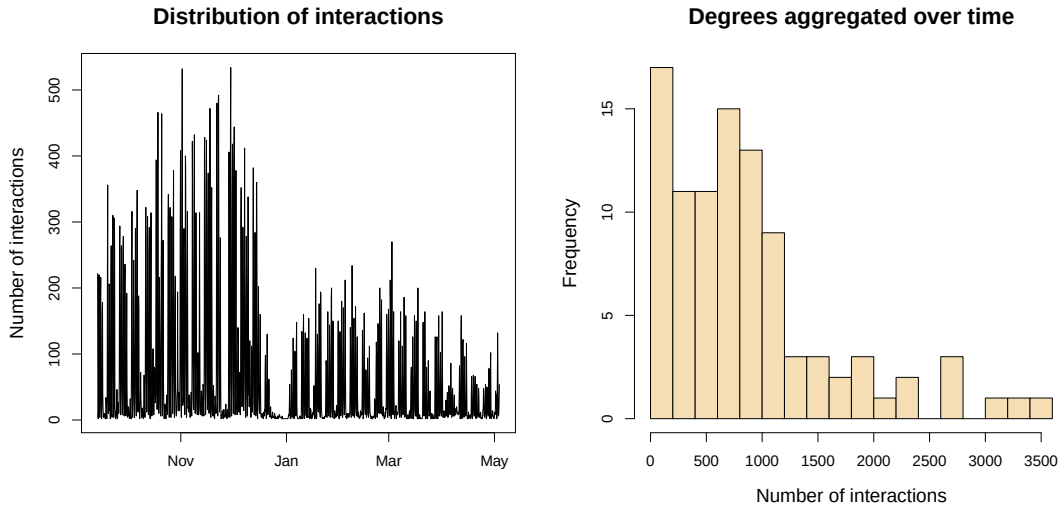

FIGURE 6. ***Reality mining dataset.*** *The plot on the left panel shows the number of edges at each of the time frames. The plot on the right panel shows instead the frequencies for the total number of edges incident to each node.*

The resulting number of groups is $K = 5$, meaning that if a node is active, it will select one of 5 different connectivity profiles at each time frame. The sizes of the groups of active nodes aggregated over time are $N_1 = 10{,}143$, $N_2 = 5{,}940$, $N_3 = 2{,}095$, $N_4 = 4{,}315$, and $N_5 = 547$. Figure 7 shows the frequencies of the number of groups across all of the time frames. These plots suggest that very often several of the groups are empty, meaning that the network is temporarily homogeneous. This is emphasised in Figure 8, where the size of all groups is shown for each time frame. The migrations between groups exhibit a clear temporal pattern, mostly following the day/night cycle. Additionally, a longer period of inactivity is observed at the end of December, where most nodes become inactive.

Plug-in estimators for the connection probabilities are available as follows:

$$\hat{P}_{gh} = \frac{U_{gh}^{01}}{U_{gh}^{01} + U_{gh}^{00}}; \quad \hat{Q}_{gh} = \frac{U_{gh}^{10}}{U_{gh}^{10} + U_{gh}^{11}}; \quad \hat{\theta}_{gh} = \frac{\eta_{gh}}{\eta_{gh} + \zeta_{gh}}; \quad \hat{\pi}_{gh} = \frac{R_{gh}}{\sum_{h=0}^{K} R_{gh}}. \tag{21}$$

For the Reality Mining dataset considered, these quantities are shown in Figure 9 through the matrices $\hat{\mathbf{P}}$, $\hat{\mathbf{Q}}$, $\hat{\Theta}$ and $\hat{\Pi}$, respectively. Overall, the matrices $\hat{\Theta}$ and $\hat{\mathbf{P}}$ exhibit high values on the leading diagonal suggesting assortative behaviour and the presence of community structure. The matrix $\hat{\mathbf{Q}}$ exhibits the opposite situation, suggesting that edges are deleted more frequently only if the nodes do not belong to the same group. This is reasonable, since it implies that edges between nodes in the same group are created more frequently and kept for a longer time, confirming the presence of communities and edge-persistence.

One can combine the information from these matrices to notice interesting disassortative patterns. In group number one, for example, the diagonal element in $\hat{\mathbf{P}}$ is small, and the nodes are more likely to connect with nodes in group five. Group five is also particularly connected with groups two and three, suggesting that the nodes in this group act like hubs in the network.

 ISSN: 2102-6238

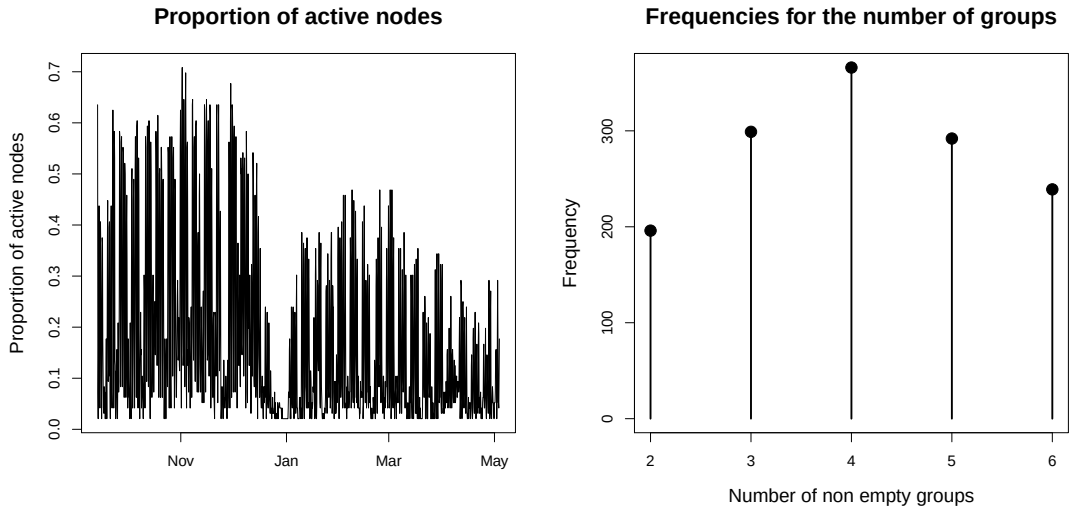

FIGURE 7. ***Reality mining dataset.*** *The plot on the left panel shows the proportion of nodes which are active in each time frame. The plot on the right panel shows instead the number of time frames where the number of non empty groups is equal to k, for* $k = 1, \ldots, 6$.

By contrast, group four is the only one exhibiting a very strong community structure, since the nodes in this groups interact almost exclusively with each other. This group may correspond to a particular community which displays isolation from the rest of the network.

As concerns the transition probabilities, the matrix $\hat{\Pi}$ exhibits high values on the diagonal which suggest high stability, since nodes tend not to change much their allocations over time. This is particularly true for group zero, containing the inactive nodes, which also continuously attracts nodes from all other groups.

9. Conclusions

This paper has introduced a new methodology to estimate the number of groups and the optimal clustering of the nodes in a Stochastic Block Transition Model. The criterion optimised is the exact Integrated Completed Likelihood, which has recently also been adopted in several other network modelling contexts. Such criterion is maximised using an iterative greedy procedure, which is known to be particularly computationally efficient. Although the framework is Bayesian, a non-informative set of prior distributions may be used, therefore resembling a black-box procedure.

One important advantage is that the method infers the number of latent groups within the same algorithmic framework, hence without requiring a grid search over all possible models. In the context considered, the number of groups reflects the number of different node behaviours that are observed at each time. One interesting feature of the method proposed is that groups may become relevant only in certain intervals and then remain empty for the remaining time frames. Hence, the number of groups may change at each time frame according to how heterogeneous the data is, or based on how many profiles are needed to represent the data.

The generative process considered allows nodes to become temporarily or permanently inactive,

 ISSN: 2102-6238

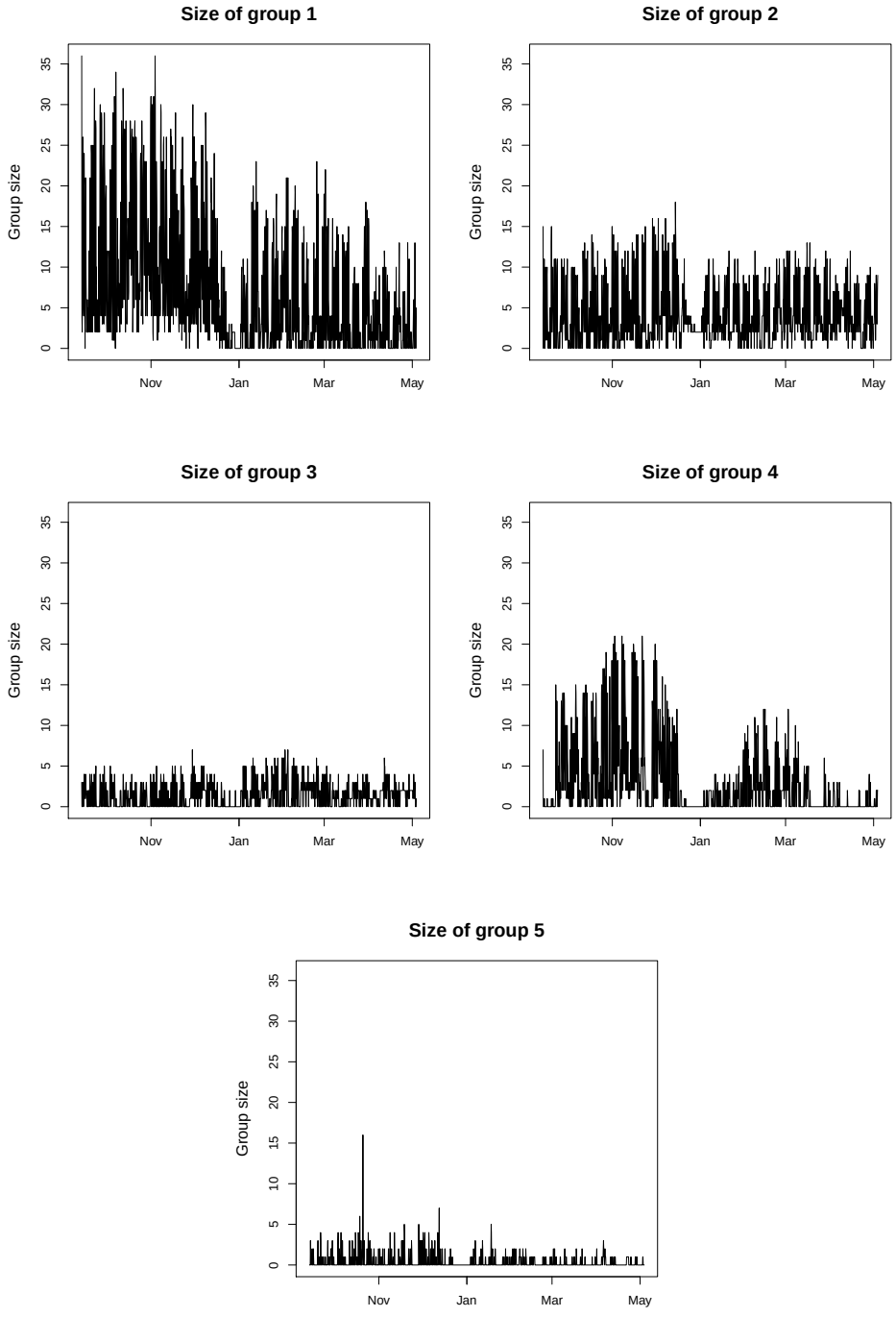

FIGURE 8. ***Reality mining dataset.*** *These plots show the number of nodes contained by each group at every time frame.*

 ISSN: 2102-6238

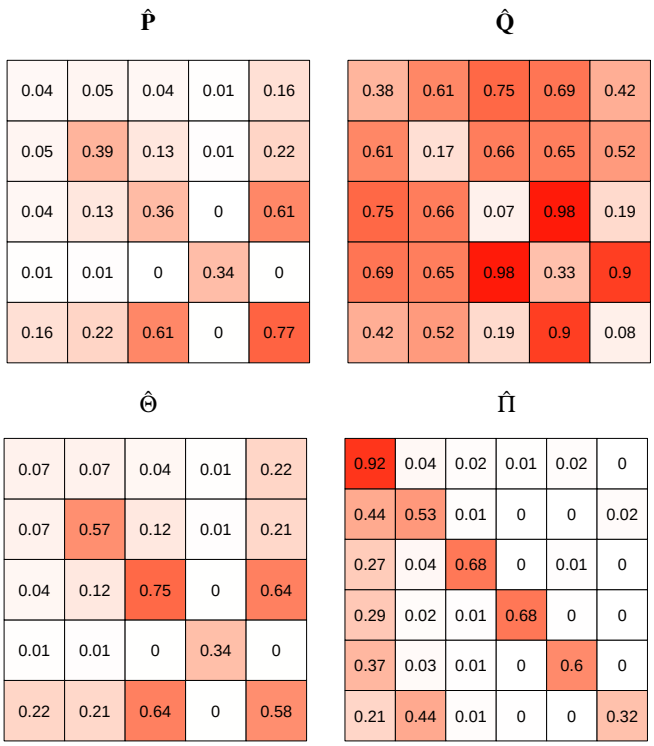

FIGURE 9. ***Reality mining dataset:*** *Estimated connection and transition probability matrices. The values for the group of inactive nodes are included only in* $\hat{\Pi}$*, in the bottom-right corner.*

making this approach appropriate for temporal networks with very many time frames. Crucially, the inactivity of the nodes is modelled in a very natural way, which can potentially lighten the computational burden substantially.

The procedure has been applied to both artificial and real dataset, showing that it can scale well with the size of the data. The simulation studies have shown that the method usually converges to excellent solutions, yet in larger datasets it may overestimate the number of groups. This seems to be a weak spot for the exact ICL method, since similar issues may be argued in other related works, such as Rastelli et al. (2018). The issue should be further addressed in future research to understand whether this may be a consequence of the heuristic estimation method, which fails at converging to solutions with fewer groups, or if this may be a limitation of the exact criterion itself, which may not be consistent for the framework considered. In addition, the simulations highlight that other available methods that do not account for edge persistence may fail to capture the true generative mechanisms of the data, and hence lead to qualitatively different clustering solutions and interpretations.

The application to the Reality Mining dataset offers a demonstration of the results that can be obtained. In this dataset, intense interaction periods appear to be distinctly fragmented due to recurring intervals of inactivity. The modelling approach proposed in this paper can handle this scenario in a natural way, and, more importantly, it can exploit the presence of inactive nodes to mitigate the computational burden.

 ISSN: 2102-6238

This paper has focused on undirected binary dynamic networks only. It may be interesting to extend this approach to the non-binary case, and to find a way to model the transitions on the edge values that allow for computationally efficient inferential procedures. Also, the discretisation of the time dimension may have a non-negligible effect on the data analysis: this is for example highlighted in Matias et al. (2018). Hence, another important future step would be to extend the Stochastic Block Transition Model principle to networks that evolve continuously on time.

Regarding the inferential procedure, several alternatives may be considered: similarly to Matias and Miele (2017), a variational Expectation-Maximisation algorithm may be employed to find the latent clustering and the model parameters within the same algorithmic framework; or, following the approaches of Wyse and Friel (2012), McDaid et al. (2013), and White et al. (2016), a collapsed Gibbs sampler may be used to sample the allocations from their marginal posterior distribution, hence obtaining an assessment of the uncertainty regarding both clustering and number of groups.

The initialisation of the algorithm remains a very central issue, since the procedure is known to be sensitive to initial conditions, and the final solutions may potentially differ a lot between various restarts. This paper uses the same initialisation method of Matias and Miele (2017) and Rastelli et al. (2018), however other possibilities (such as spectral clustering) may be explored.

The `R` package `GreedySBTM` accompanies this paper: it contains a `C++` implementation of the algorithms described in Section 6, and it includes the adapted version of the Reality Mining dataset used in Section 8. The package is publicly available on CRAN (R Core Team, 2017).

Acknowledgements

The author thanks Kevin Xu for the useful conversations, and the two anonymous reviewers for their thoughtful comments. This research was partially carried out while the author was affiliated to the Vienna University of Economics and Business, Austria, and funded through the Vienna Science and Technology Fund (WWTF) Project MA14-031.

References

Bartolucci, F., Marino, M. F., and Pandolfi, S. (2018). Dealing with reciprocity in dynamic stochastic block models. *Computational Statistics & Data Analysis*, 123:86–100.

Bertoletti, M., Friel, N., and Rastelli, R. (2015). Choosing the number of clusters in a finite mixture model using an exact integrated completed likelihood criterion. *Metron*, 73(2):177–199.

Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(7):719–725.

Côme, E. and Latouche, P. (2015). Model selection and clustering in stochastic block models based on the exact integrated complete data likelihood. *Statistical Modelling*, 15(6):564–589.

Corneli, M., Latouche, P., and Rossi, F. (2018). Multiple change points detection and clustering in dynamic networks. *Statistics and Computing*, 28(5):989–1007.

Daudin, J. J., Picard, F., and Robin, S. (2008). A mixture model for random graphs. *Statistics and Computing*, 18(2):173–183.

Eagle, N. and Pentland, A. S. (2006). Reality mining: sensing complex social systems. *Personal and ubiquitous computing*, 10(4):255–268.

Fortunato, S. (2010). Community detection in graphs. *Physics reports*, 486(3):75–174.

Friel, N., Rastelli, R., Wyse, J., and Raftery, A. E. (2016). Interlocking directorates in irish companies using a latent space model for bipartite networks. *Proceedings of the National Academy of Sciences*, 113(24):6629–6634.

Heaukulani, C. and Ghahramani, Z. (2013). Dynamic probabilistic models for latent feature propagation in social networks. In *International Conference on Machine Learning*, pages 275–283.

 ISSN: 2102-6238

Hoff, P. D., Raftery, A. E., and Handcock, M. S. (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098.

Holland, P. W., Laskey, K. B., and Leinhardt, S. (1983). Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137.

Ishiguro, K., Iwata, T., Ueda, N., and Tenenbaum, J. B. (2010). Dynamic infinite relational model for time-varying relational data analysis. In *Advances in Neural Information Processing Systems*, pages 919–927.

Karrer, B. and Newman, M. E. J. (2011). Stochastic blockmodels and community structure in networks. *Physical Review E*, 83(1):016107.

Latouche, P., Birmele, E., and Ambroise, C. (2012). Variational bayesian inference and complexity control for stochastic block models. *Statistical Modelling*, 12(1):93–115.

Malsiner-Walli, G., Frühwirth-Schnatter, S., and Grün, B. (2016). Model-based clustering based on sparse finite gaussian mixtures. *Statistics and computing*, 26(1-2):303–324.

Matias, C. and Miele, V. (2017). Statistical clustering of temporal networks through a dynamic stochastic block model. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(4):1119–1141.

Matias, C., Rebafka, T., and Villers, F. (2018). A semiparametric extension of the stochastic block model for longitudinal networks. *Biometrika*, 105(3):665–680.

Matias, C. and Robin, S. (2014). Modeling heterogeneity in random graphs through latent space models: a selective review. *ESAIM: Proceedings and Surveys*, 47:55–74.

McDaid, A. F., Murphy, T. B., Friel, N., and Hurley, N. J. (2013). Improved bayesian inference for the stochastic block model with application to large networks. *Computational Statistics & Data Analysis*, 60:12–31.

Nobile, A. and Fearnside, A. T. (2007). Bayesian finite mixtures with an unknown number of components: the allocation sampler. *Statistics and Computing*, 17(2):147–162.

Nowicki, K. and Snijders, T. A. B. (2001). Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rastelli, R., Latouche, P., and Friel, N. (2018). Choosing the number of groups in a latent stochastic blockmodel for dynamic networks. *Network Science (to appear)*. https://doi.org/10.1017/nws.2018.19.

Rousseau, J. and Mengersen, K. (2011). Asymptotic behaviour of the posterior distribution in overfitted mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(5):689–710.

Strehl, A. and Ghosh, J. (2002). Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617.

Wang, Y. J. and Wong, G. Y. (1987). Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82(397):8–19.

White, A., Wyse, J., and Murphy, T. B. (2016). Bayesian variable selection for latent class analysis using a collapsed gibbs sampler. *Statistics and Computing*, 26(1-2):511–527.

Wyse, J. and Friel, N. (2012). Block clustering with collapsed latent block models. *Statistics and Computing*, 22(2):415–428.

Wyse, J., Friel, N., and Latouche, P. (2017). Inferring structure in bipartite networks using the latent blockmodel and exact icl. *Network Science*, 5(1):45–69.

Xu, K. (2015). Stochastic block transition models for dynamic networks. In *Artificial Intelligence and Statistics*, pages 1079–1087.

Xu, K. S. and Hero, A. O. (2014). Dynamic stochastic blockmodels for time-evolving social networks. *Selected Topics in Signal Processing, IEEE Journal of*, 8(4):552–562.

Yang, T., Chi, Y., Zhu, S., Gong, Y., and Jin, R. (2011). Detecting communities and their evolutions in dynamic social networks – a bayesian approach. *Machine learning*, 82(2):157–189.

Zhang, X., Moore, C., and Newman, M. E. J. (2017). Random graph models for dynamic networks. *The European Physical Journal B*, 90(10):200.

 ISSN: 2102-6238