Inferring social network structure in ecological systems from spatio-temporal data streams

Electronic Supplementary Material

Ioannis Psorakis *[1,2], Stephen J. Roberts[1], Iead Rezek[1], and Ben C. Sheldon[2]

[1]Pattern Analysis and Machine Learning Research Group, University of Oxford
[2]Edward Grey Institute, University of Oxford

March 16, 2012

1 Network basics

Networks have been an important field of study since Euler and the solution of the Königsberg bridge problem (1735). During the 20th century and before the World Wide Web era, sociologists used the network paradigm to model human interactions with the most popular studies being Granovetter's "Strength of weak ties" (1973) [7] and Milgram's "Small-world" experiments (1967) [11]. In recent years, the field of network analysis has undergone an explosive growth [2] as theoretical and computational advances have allowed the study of large scale complex systems such as the World Wide Web, Social Media, scientific collaboration patterns, animal societies and protein interactions [4, 13]. One of the most fascinating findings of these studies is that real-world networks exhibit a staggering amount of similarity, allowing us to study a wide and diverse range of complex systems under a single conceptual framework [8].

In mathematics, a unipartite *graph* $\mathbf{G_1}(\mathbf{V}, \mathbf{D})$ is comprised of a set $\mathbf{V}$ of N nodes or vertices, connected together by a set $\mathbf{D}$ of M edges or links. The overall connectivity profile can be described by using the *adjacency matrix* $\mathbf{A} \in \mathbb{R}^{N \times N}$, so that if $a_{ij} \neq 0$ then nodes i and j are linked together [1]. An example graph is shown in Fig. 1.

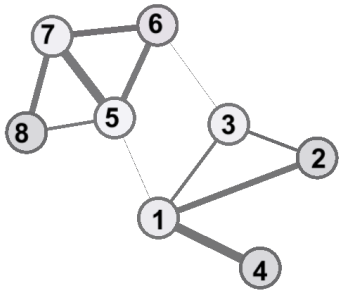

Figure 1: An example graph of $N = 8$ nodes and $M = 11$ edges. Edge widths represent varying connection strengths. The above network can be described by an 8×8 adjacency matrix $\mathbf{A}$, where each element a_{ij} denotes the connection strength between i and j.

Additionally, a two-mode or *bipartite graph* $\mathbf{G_2}(\mathbf{V}, \mathbf{U}, \mathbf{D})$ has two sets $\mathbf{V}, \mathbf{U}$ of nodes, N and K in number and the M links from $\mathbf{D}$ are allowed to connect vertices only of different type, as shown in Fig. 2. The connectivity profile is described by the *incidence matrix* $\mathbf{B} \in \mathbb{R}^{N \times K}$, so that if $b_{ik} \neq 0$ then the node i from $\mathbf{V}$ is connected to node k from $\mathbf{U}$. Similar to the unipartite case, the values of b_{ik} can be Boolean or quantify connection strength.

Very commonly we use the term *network* to describe the simplified version of the pattern of interactions in a system (for example an Online Social Network), where nodes are individual entities and edges represent some form of association, interaction, similarity, commodity flow or correlation between nodes. Similar to the way a map is a simplified (though useful) version of a landscape, a network describes the *topology* of a real-world system by focusing on the connectivity

*ioannis.psorakis@eng.ox.ac.uk

[1] a_{ij} can be a simple Boolean value (unweighted edge), a real value (weighted edge) or a signed value (directed edge). In this work we will not cover directed graphs, so $\mathbf{A} \in \mathbb{R}_{+}^{N \times N}$.

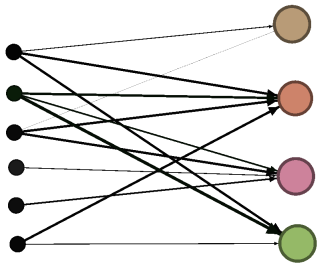

Figure 2: An example bipartite graph of $N = 6$ nodes of type 1 and $K = 4$ nodes of type 2. Edge widths represent varying connection strengths. The above network can be described by an 6×4 incidence matrix $\mathbf{B}$, where each element b_{ij} denotes the connection strength between i and j.

patterns of its individual components [16]. Although strictly speaking, the term "graph" denotes the abstract mathematical structure described by $\mathbf{G}(\mathbf{V}, \mathbf{D})$ and $\mathbf{A} \in \mathbb{R}^{N \times N}$, in some very influential works in the literature such as [13], the terms "graph" and "network" are used interchangeably.

The exponentially increasing popularity of network analysis in the scientific literature [2] is not only a result of the computational advances in data gathering, storage and processing technology of recent decades [16]. We have also realised that systems in nature are *complex*, i.e they are made up of a large number of entities interacting in such a way that their collective behaviour is not merely a simple combination of their individual behaviours [12]. The network paradigm is therefore a very appropriate and flexible tool to describe a system at a macroscopic scale, as nodes and edges can represent any sort of entity of association. This contrasts with the traditional reductionist viewpoint of breaking down a system into its parts and focusing on each component separately [16], as networks omit the individual characteristics of each node and describe our data in a *relational* form. The key idea of this framework is that the connectivity patterns in the data have a big effect on the behaviour of the network as a system [13].

The fact that networks have been so popular in describing real-world systems has led Buchanan and Caldarelli to ask *"why Nature is so fond of networks?"* [2]. This question is based on the relatively recent findings that networks that describe complex systems possess a significant amount of similar statistical and topological properties, regardless of the application domain.

2 Details on data collection and experiment set-up

The work described here is part of an ongoing study of social behaviour in a population of great tits *Parus major* at Wytham Woods, near Oxford, which have been the subject of a long-term population ecological study (e.g. see [10] and [1] for details). In each year from 2007-2009, each nestling great tit born on the study site, and each captured adult great tit breeding there, were ringed with a plastic colour ring (CoreRFID Ltd) as shown in Fig. 3, which contained a 125 Hz RFID tag.

Figure 3: Birds are "ringed" with a harmless RFID device that generates a sensor observation when the bird comes to the close proximity of one of the 16 loggers placed across Wytham woods, Oxfordshire.

The majority of the birds were marked in the breeding season, which occurs in May-June each year. Each autumn and winter (beginning in August and ending in early March), we deployed bird feeders, baited with sunflower seeds, at 67 locations, which were spaced regularly on a 250m grid throughout the 385 ha of the study site, shown in Fig. 4.

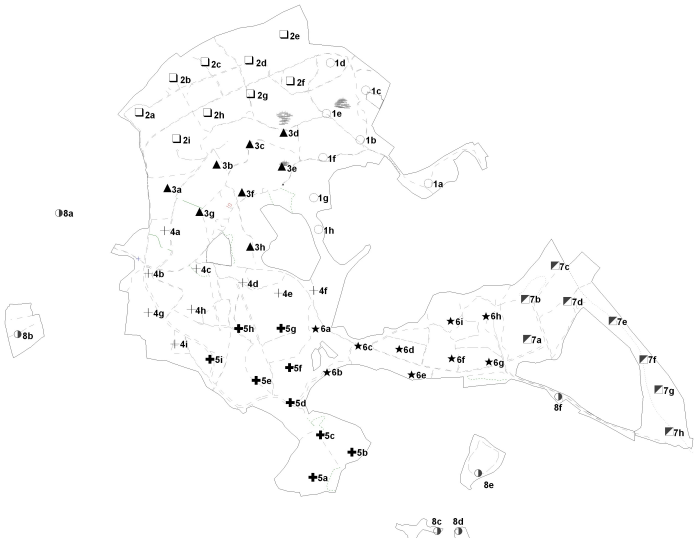

Figure 4: A grid of sixty-seven feeding location spread across Wytham woods, Oxford. At each one of the sixty-seven locations in the forest, there is a feeder that acts as an attraction point for foraging individuals. By placing appropriate logging hardware at the feeder, we are able to record the presence of each individual bird. Due to equipment constraints, there were only 16 loggers available at any time, and these were thus rotated around the 67 locations following a structured randomised design, so that each of 8 approximately equally-sized sections of the site always had two active loggers in it.

The feeders were equipped with two antennae (Francis Instruments Ltd, Cambridge) which logged visits of RFID-tagged great tits to collect sunflower seeds, and recorded the time of the visit to the nearest 15 seconds. The 16 loggers available at any time, were rotated around the 67 locations following a structured randomised design, so that each of 8 approximately equally-sized sections of the site always had two active loggers in it. Rotation happened on a 4 day schedule, and feeders were refilled with sunflowers each time they were moved. The data analysed here are taken from the first two winters of this project, 2007-8 and 2008-9, in which there were 548,709 records of 770 individuals and 484,088 records of 753 individuals respectively; in total over the two winters there were 1,032,797 records of 1,217 different individual great tits.

3 Details on the clustering scheme

3.1 Generative model

We consider the activity profile shown in Fig. 5, where each "burst" signifies a foraging event. We identify such regions of increased observation density by partitioning our data $\{b_z, t_z\}_{z=1}^{Z}$ based on their timestamp t_z, where a given subset $[Z_1, Z_2]$ denotes a series of bird appearances $\{b_z, t_z\}_{z=Z_1}^{Z_2}$ that occurred in close temporal proximity.

To address this one-dimensional *clustering* problem, we model each timestamp t_z in the data stream as a draw from a mixture of Gaussian distributions:

$$P(t_z) = \sum_{k=1}^{K} \pi_k \mathcal{N}(t_z | \mu_k, \beta_k^{-1}) \tag{1}$$

The above equation denotes that there are K "centres of mass" in the data stream, shown as vertical dashed lines in Fig. 5, around which observations are concentrated. Each k of those lines or *centroids* is placed in the data stream with a timestamp μ_k. The precision term β_k controls how "dense" each gathering event is in terms of the temporal distance of observations around it. Each cluster corresponds to a different Gaussian component k that is weighted by a mixing coefficient π_k, for which $\sum_{k=1}^{K} \pi_k = 1$. Our is to infer:

1. The effective number of clusters K in the data stream.
2. The position μ_k of each one of their respective centroids, along with the "density" parameter β_k.
3. The mixing coefficients π_k.
4. The assignment of each observation z to one of the K clusters or "events".

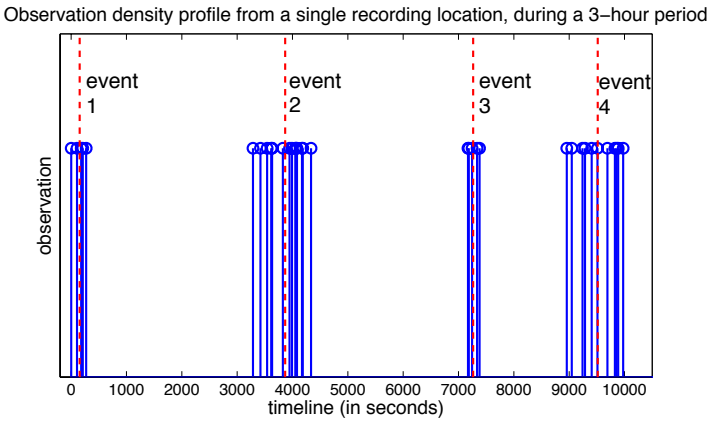

Figure 5: We plot bird arrivals as recorded at a specific location over the course of 3-hour period. We can see that the visitation profile is temporally focused, consisting of "bursts" of bird activity. Our goal is to identify such regions of increased observation density and examine which individuals participate in these gathering events.

Towards the above goals, we consider the generative model of Fig. 6, where the observed variable t_z denotes the timestamp of a given observation z. We assume that for each t_z in the data there is an associated "hidden" or *latent* binary vector $\mathbf{y}_z$, for which $\sum_{k=1}^{K} y_{zk} = 1$, where $y_{zk} = 1$ denotes the observation t_z was generated from mixture k. Thus we can write the likelihood of a single observation timestamp t_z, given the values of y_{zk}, μ_k, β_k, as:

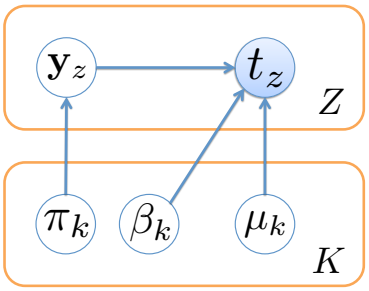

Figure 6: The graphical model denoting the generation of an observation t_z via a mixture of K Gaussians. The membership of t_z to a particular mixture is controlled by the latent $1 \times K$ binary vector $\mathbf{y}_z$. Constant terms that parameterise the priors have been omitted.

$$P(t_z|\mathbf{y}_z, \boldsymbol{\mu}, \boldsymbol{\beta}) = \prod_{k=1}^{K} \mathcal{N}(t_z|\mu_k, \beta_k^{-1})^{y_{zk}} \tag{2}$$

where the exponent y_{zk} denotes that mixture k is activated only if $y_{zk} = 1$. We consider each $\mathbf{y}_k$ as drawn from a multinomial distribution $P(\mathbf{y}_z|\boldsymbol{\pi}) = \mathcal{M}(\mathbf{y}_z; \boldsymbol{\pi})$ parameterised by the mixing coefficients π_k as seen in our model of Fig. 6. Because $\sum_{k=1}^{K} \pi_k = 1$ and based on our generative model, we consider these coefficients a draw from a Dirichlet distribution:

$$P(\boldsymbol{\pi}) = \mathbf{Dir}(\boldsymbol{\pi}; \lambda_0) = \frac{\Gamma(k\lambda_0)}{\Gamma(\lambda_0)^k} \prod_{k=1}^{K} \pi_k^{\lambda_0 - 1} \tag{3}$$

where $\Gamma(.)$ the gamma function. Additionally, we place a Gaussian distribution over each centroid μ_k:

$$P(\mu_k) = \mathcal{N}(\mu_k; m_0, \upsilon_0) = \frac{1}{\sqrt{2\pi\upsilon_0^{-2}}} \exp\left(\frac{-\upsilon_0^2}{2}(\mu_k - m_0)^2\right) \tag{4}$$

and a Gamma prior over the corresponding precisions β_k:

$$P(\beta_k) = \mathcal{G}(\beta_k; a_0, b_0) = \frac{b_0^{a_0}}{\Gamma(a_0)} \beta_k^{a_0 - 1} e^{-b_0 \beta_k} \tag{5}$$

where the hyper-hyperparameters $a_0, b_0, m_0, \upsilon_0$ that control the priors on Eq. (4) and (5) are fixed.

Let us now define a vector $\mathbf{t} \in \mathbb{R}^{Z\times 1}$ that contains all the observation timestamps from our data stream, along with a matrix $\mathbf{Y} \in \mathbb{R}^{Z\times K}$ where each row $\mathbf{y}_z$ is the latent variable denoting the mixture membership of observation z. Based on the generative model of Fig. 6, the joint distribution over all variables factorises as follows:

$$P(\mathbf{t}, \mathbf{Y}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi}) = P(\mathbf{t}|\mathbf{Y}, \boldsymbol{\mu}, \boldsymbol{\beta})P(\mathbf{Y}|\boldsymbol{\pi})P(\boldsymbol{\beta})P(\boldsymbol{\mu}) \tag{6}$$

what we are interested in, is the posterior distribution of the model parameters $\mathbf{Y}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi}$ given the data stream $\mathbf{t}$ and the prior structure denoted by our graphical model in Fig. 6.

3.2 Variational approximation

The proposed distribution $q(\mathbf{Y}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi})$ that approximates the posterior over the model parameters, is:

$$q(\mathbf{Y}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi}) = q(\mathbf{Y})q(\boldsymbol{\mu})q(\boldsymbol{\beta})q(\boldsymbol{\pi}). \tag{7}$$

The Variational Bayes (VB) framework seeks to maximise the negative free energy term:

$$F = \mathbb{E}_{\mathbf{Y},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\pi}}\left(\log \frac{P(\mathbf{t}, \mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi})}{q(\mathbf{Y})}\right) - KL(q(\boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi})||p(\boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi})) \tag{8}$$

where the first term corresponds to an average likelihood and the second term is the KL divergence between the priors and the posteriors. This objective function can be maximised via and EM scheme, where in the E-step the distribution over the mixture assignments $q(\mathbf{Y})$ is updated according to:

$$q(\mathbf{Y}) \propto \exp[I(\mathbf{Y})] \tag{9}$$

where

$$I(\mathbf{Y}) = \mathbb{E}_{\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\pi}}(\log P(\mathbf{t}, \mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\pi})) \tag{10}$$

For the precisions we have $q(\boldsymbol{\beta}) = \prod_k q(\beta_s)$ and Gamma densities:

$$q(\beta_k) = \mathcal{G}(\beta_k; a_k, b_k) \tag{11}$$

For the centroids μ_k, we have $q(\boldsymbol{\mu}) = \prod_k q(\mu_k)$ and Normal densities:

$$q(\mu_k) = \mathcal{N}(\mu_k; m_k, \upsilon_k) \tag{12}$$

and for the mixing coefficients we have a Dirichlet:

$$q(\boldsymbol{\pi}) = \Gamma\left(\sum_{k'} \lambda_{k'}\right) \prod_{k=1}^{K} \frac{\pi_k^{\lambda_k - 1}}{\Gamma(\lambda_k)} \tag{13}$$

For the indicator posteriors we have $q(\mathbf{Y}) = \prod_z q(\mathbf{y}_z)$ and we write $\gamma_{nk} = q(y_{nk} = 1)$. This quantity, termed the *responsibility* of cluster k for explaining data point t_z can be seen as a membership score of observation z to the event k.

3.2.1 E-step

The E-step consists of updating the indicator posterior according to:

$$\tilde{\gamma}_{zk} = \tilde{\pi}_k \tilde{\beta}_k^{1/2} \exp\left[-\frac{1}{2}\tilde{\beta}_k(t_z^2 + m_k^2 + \upsilon_k - 2m_k t_z)\right] \tag{14}$$

where

$$\log \tilde{\pi}_k = \Psi(\lambda_k) - \Psi(\sum_{k'} \lambda_{k'}) \;\text{ and }\; \log \tilde{\beta}_k = \Psi(b_k) + \log a_k \tag{15}$$

and $\Psi(.)$ is the digamma function. We normalise the responsibilities for a given data point t_z:

$$\gamma_{zk} = \frac{\tilde{\gamma}_{zk}}{\sum_{k'} \tilde{\gamma}_{zk'}} \tag{16}$$

thus obtaining is the probability that component k is responsible for explaining data point t_z.

3.2.2 M-step

In the M-step we define the following variables for our update equations:

$$\tilde{\pi}_k = \frac{1}{Z}\sum_{z=1}^{Z}\gamma_{zk} \quad (17)$$

$$\tilde{Z}_k = Z\tilde{\pi}_k \quad (18)$$

$$\tilde{t}_k = \frac{1}{Z}\sum_{z=1}^{Z}\gamma_{zk}t_z \quad (19)$$

$$\tilde{t}_k^2 = \frac{1}{Z}\sum_{z2=1}^{Z}\gamma_{zk}t_z^2 \quad (20)$$

in which $\tilde{\pi}_k$ is the proportion of data in component k and $\tilde{Z}_k$ is the number of observations associated with component k. The quantities $\tilde{t}_k$ and $\tilde{t}_k^2$ are the weighted data values and weighted squared data values respectively.

We then update the hyperparameters as follows. The mixing hyperparameters are updated by adding the data counts to the prior counts:

$$\lambda_k = \tilde{Z}_k + \lambda_0 \quad (21)$$

If we define the average variance of component k as:

$$\tilde{\sigma}_k^2 = \tilde{t}_k^2 + \tilde{\pi}_k(m_k^2 + \upsilon_k) - 2m_k\tilde{t}_k \quad (22)$$

then the hyperparameters for the precisions are updated as:

$$\frac{1}{a_k} = \frac{Z}{2}\tilde{\sigma}_k^2 + \frac{1}{a_0} \quad (23)$$

$$b_k = \frac{\tilde{Z}_k}{2} + b_0 \quad (24)$$

3.3 Initialisation

For the means μ_k governed by a Gaussian prior with mean m_0 and precision υ_0 we initialise using the corresponding global dataset statistics. For the precisions β_k we place uninformative priors by setting $a = 10^3$ and $b = 10^{-3}$. The mixing coefficients π_k, governed by a Dirichlet distribution, are parameterised with $\lambda_0 = 5$, once more giving an uninformative prior. The VB equations are applied iteratively until a consistent solution is reached. Convergence is measured by evaluating the negative free energy F, where minimal improvement ($< 10^{-3}$) defines our termination criterion.

4 GMMEvents as a clique percolation process

Let us assume that the ground truth network is known to us, and one of its communities is shown in Fig. 7. We seek to address the issue of how can the proposed model, which forces connections between all individuals that participate in the same gathering events, give rise to network communities where not all node pairs are connected.

Consider the data stream shown in Fig. 8, where our algorithm has identified various gathering events. As already discussed, in our model all individuals that participate in the same event are connected, therefore such observation-dense regions correspond to *fully-connected subgraphs* or *cliques* in the network. Due to the fact that many gathering events in the data stream can have *common members* (as individual birds do not have a fixed set of companions that join them during every single feeder visitation) our model naturally extracts a series of fully-connected subgraphs that share nodes. The whole network is then reconstructed as an aggregation of such partially overlapping or *adjacent* cliques.

Community structure in such process arises naturally, as there can be collections of fully-connected subgraphs that share members. In fact, Palla *et al.* in their 2005 paper [14] have shown that network communities can be seen as aggregations of adjacent cliques. The way our algorithm in Fig. 8 performs multiple "partial observations" of the community in Fig. 7 (one per gathering event), corresponds to what the authors in [14] describe as a "clique rolling process", which they use to identify communities.

In both cases, nodes do need to be directly connected in order to be assigned to the same community. What matters is their common position in a node neighbourhood that consists of many overlapping, strongly connected cliques.

Note that normally a data stream such as the one shown in Fig. 8 will be corrupted by noise; there can be individuals that appear in gathering events coincidentally, without having any social connection with any of the other members. Such

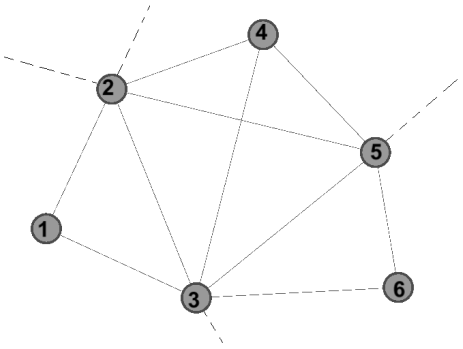

Figure 7: An example community of 6 nodes. Dashed lines represent links that allow connections with the rest of the network.

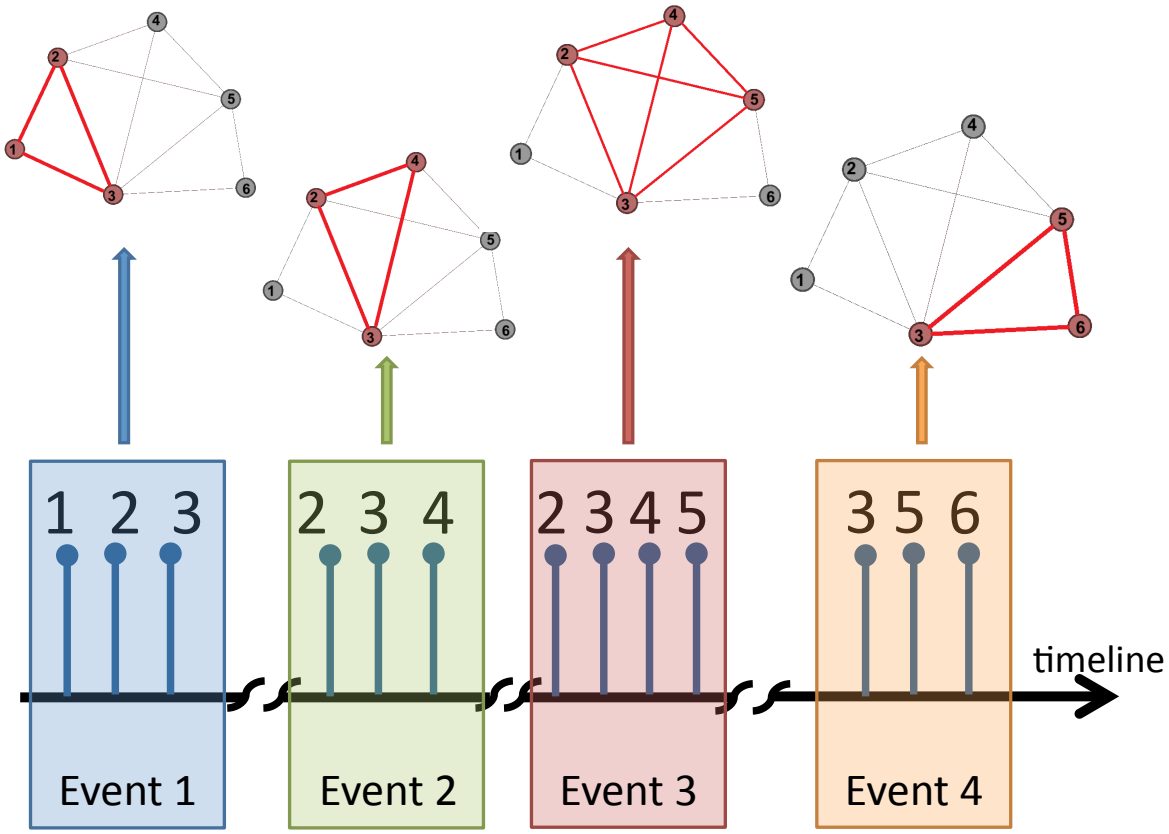

Figure 8: An example data stream of 4 gathering events. Each gathering event with k participants corresponds to a k-clique (fully connected subgraph with k nodes), which can be seen as a "partial view" of the overall community.

co-occurrences are usually removed by our significance testing scheme. In Fig. 8 we have chosen the noiseless case in order to illustrate how can individuals belong to the same community without a direct connection.

As a final note, by employing a probabilistic viewpoint we can regard communities not only as social circles, but also as node classes where individuals are most likely to interact. In fact, we have shown in previous work [15] that if nodes belong to the same social circles, the expected connectivity weight (under a Poisson noise model) between them increases. This implies that even if we have not observed a direct link between members of the same community, the fact that they are positioned in the same node neighbourhood increases the probability (our "posterior belief" in a Bayesian context) that either i) they are indeed connected, but we have missed the link due to incomplete data or ii) they are not directly linked yet, but there exists a pressure or *bias* for them to be connected because of their common social circle (this

is similar to transitivity [7] and graph densification [9]).

5 Application to artificial data streams

We performed a series of simple "sanity checks", in order to examine if, given a data stream $\{b_z, t_z\}_{z=1}^{Z}$, our approach captures the relationships among individuals that are in consistently close temporal proximity. This can be accomplished by generating artificial data streams from a fully observed graph and compare the extracted topology versus the original.

Consider a network of N nodes with a given adjacency matrix $\mathbf{A}$. We require that closely connected individuals in $\mathbf{A}$ will appear frequently together in the corresponding data stream. Additionally, we want our overall visitation profile to consist of clusters, such as the ones in Fig. 5, that denote "bursts" of social activity.

Based on the requirements stated above and given our adjacency matrix $\mathbf{A}$, we create our artificial data stream $\{b_z, t_z\}_{z=1}^{Z}$ as follows:

1. We set our first observation $z = 1$ and our clock $t_1 = 1$.
2. We set $b_z =$ one random node in the network.
3. We perform a *random walk* in the network neighbourhood around b_z. For each node i we visit, we increment our observation counter $z = z + 1$ and then set $b_z = i$ and $t_z = t_{z-1} + 1$.
4. The random walk stops after a certain number of visitations, controlled by a Poisson random variable λ_r. This sequence of observations defines a cluster or "burst" of social activity among closely connected individuals in the network.
5. We increment $z = z + 1$ and set $t_z = t_{z-1} + \lambda_t$, where λ_t a Poisson random variable that controls the temporal distance between clusters.
6. Return to step 2 or terminate if $z > z_{\max}$ and all graph nodes appear in the stream.

The above algorithm defines a scheme where nodes that form closely connected subgraphs (such as cliques) appear closely together in a data stream. In order to effectively explore such subgraphs, we have to define an appropriate random walk strategy:

- Consider a given node i in the network. We require that a random walker will traverse the network around this focal node, in a way that favours adjacent nodes who have a) strong connection weight and b) many common neighbours with i.
- Starting from i, we assign each adjacent node j a visitation probability $p(j|i) = a_{ij}/s_i$, where $s_i = \sum_{j=1}^{N} a_{ij}$ the *strength* of node i, so that strongly connected neighbours are *more likely* to be visited.
- While being in an adjacent node j of i, identify all their common neighbours using the row vector $A^{(c)} = A_{i*} \cdot A_{j*}$ where $(\cdot)$ denotes element-by-element multiplication and A_{i*} the i-th row of $\mathbf{A}$. Pick the next node h with probability $p(h|j) = A_h^{(c)} / \left(\sum_{n=1}^{N} A_n^{(c)}\right)$. If no common neighbours exist, return to focal node i. This is the inverse application of the scheme presented in Fig. 8.

We apply the above scheme on the Newman-Girvan random graph (NG) template [6] that consists of $N = 128$ nodes, observed solution of $C = 4$ communities (with $n = 32$ nodes each) and average degree of $\langle k \rangle = 16$. Additionally, the variable *inter-community* degree $\langle k_{out} \rangle$ controls the module cohesiveness of the network. Given an instance of such graph with adjacency matrix $\mathbf{A}$, we generate a data stream of an appropriate size in order for all nodes to appear. We then use GMMEvents to extract the agent connectivity in the stream, which leads us to a new adjacency matrix $\mathbf{A}'$.

We seek to compare $\mathbf{A}$ and $\mathbf{A}'$ and see how well the original topology of the NG graph is recovered in our new network. For that reason, an appropriate way of comparing the two networks would be in terms of their *community structure* and see how well their "topological signatures" match. This approach also avoids the problem of link weight magnitudes, which may be different between $\mathbf{A}$ and $\mathbf{A}'$ depending on the size of the data stream.

In Fig. 9 we plot the similarity of the original $\mathbf{A}$ versus the extracted $\mathbf{A}'$ NG graph at various levels of community cohesion. For each value of $\langle k_{out} \rangle$, which controls the tendency of nodes to link with members of other communities, we generate 100 networks. For each instance, we generate a data stream using the algorithm described above and use GMMEvents to recover the original adjacency matrix. We perform community extraction using Bayesian Non-negative Matrix Factorisation (Bayes NMF) [15] on each recovered adjacency matrix $\mathbf{A}'$ and use Normalised Mutual Information [3] to measure the mesoscopic similarity of the extracted versus the ground-truth graph. We can see from Fig. 9 that GMMEvents performs an accurate extraction of the original graph topology for most cases of community cohesion, while it fails only when the original network possesses a close to random mesoscopic organisation.

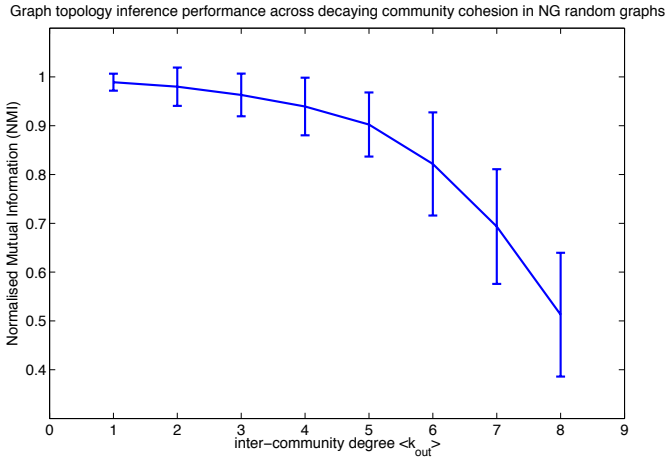

Figure 9: We plot the similarity, in terms of Normalised Mutual Information, of community structure between the ground truth Newman-Girvan random graph and the one extracted using GMMEvents . The different values of $\langle k_{out} \rangle$ represent various levels of "fuzziness" in terms of how easily communities are separated in the network.

6 Software

The methods presented in the paper were implemented using MATLAB R2010b, with the Statistics and Parallel toolboxes installed. The scripts are accessible through the Pattern Analysis and Machine Learning Research Group (PARG) webpage `http://www.robots.ox.ac.uk/~parg/software.html`.

The main script that performs graph extraction from spatio-temporal data is named `gmmevents.m`. The first input argument of the function is the actual data stream, which should be in a pre-specified format; a MATLAB matrix of dimensionality $[Z \times 3]$ where Z the total observations as shown in Fig. 10.

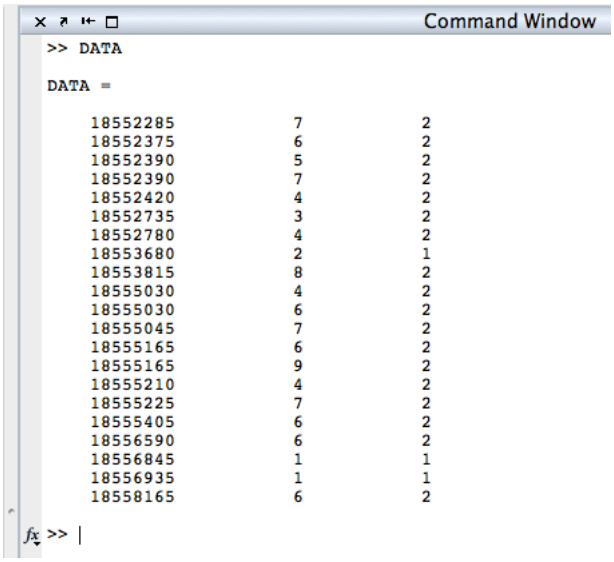

Figure 10:

In the above example, the input variable `DATA` represents a toy data stream with $Z = 21$ rows that represent 21 bird visits to feeding locations. The first column in Fig. 10 represents the timestamps t_z of each record. The second column represents the unique ID of individual birds. In our code we use integer IDs for identifying birds, so that there is a direct correspondence with the node numbering used in the adjacency matrix of the network. The third column specifies the location on which the observation took place. Again, the location IDs are integers.

The second argument is the total number of individuals in the stream. Although this can be skipped and automatically inferred by the second column of the `DATA` variable, there are cases where we want to show an individual's absence in the inferred network. The third argument is the number of randomisations for the null model. This can be set to 0 if the user does not which to perform a significance test on the link weights.

The script outputs are:

- `A` : the actual $N \times N$ adjacency matrix of the network. Each element a_{ij} is the number of co-occurrences of individuals i and j.
- `B` : is a L-cell array, where L is the number of locations. Each cell `B{l}` is an $N \times K_l$ matrix, with N individuals and K_l gathering events[2]. Such matrix (that defines a bipartite network) can be used for the extraction of various association indices such as the Half-Weight-Index [5].
- `X` : a $N \times 1$ vector, where each element x_i is the total appearances of individual i in the data stream.
- `Anull_mean` : the $N \times N$ adjacency matrix of the network under the null model. Each element `Anull_mean(i,j)` is the average number of co-occurrences of individuals i and j under the null hypothesis.
- `Anull_std` : an $N \times N$ matrix where each element `Anull_std(i,j)` denotes the standard deviation in the average number of `Anull_mean(i,j)` co-occurrences between individuals i and j, given the null hypothesis.

In cases were the dataset is very large (for example, it covers an entire year), in order to regulate computational complexity we would recommend that the algorithm is not applied on the whole stream directly. Instead, the user may want to partition it into chunks either denote some natural separation in the observation timeline (for example, days) or into chunks of size 10,000 records. The whole network can then be recovered by simply summing the subgraphs that correspond to each partition.

For bug reports and recommendations, the corresponding author is ioannis.psorakis@eng.ox.ac.uk.

References

[1] S. Bouwhuis, BC Sheldon, S. Verhulst, and A. Charmantier. Great tits growing old: selective disappearance and the partitioning of senescence to stages within the breeding cycle. *Proceedings of the Royal Society B: Biological Sciences*, 276(1668):2769, 2009.

[2] M. Buchanan and G. Caldarelli. A networked world. *Physics World*, 23(2):22–24, 2010.

[3] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas. Comparing community structure identification. *J. Stat. Mech.*, 2005(09):P09008, 2005.

[4] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, February 2010.

[5] J.R. Ginsberg and T.P. Young. Measuring association between individuals or groups in behavioural studies. *Animal Behaviour*, 1992.

[6] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *PNAS*, 99(12):7821–7826, 2002.

[7] M. Granovetter. The strength of weak ties. *American Journal of Sociology*, 78(6):1360–1380, 1973.

[8] R. Lambiotte. Multi-scale modularity in complex networks. *arXiv:1004.4268v1*, April 2010.

[9] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 177–187, New York, NY, USA, 2005. ACM.

[10] RH McCleery, RA Pettifor, P. Armbruster, K. Meyer, BC Sheldon, and CM Perrins. Components of variance underlying fitness in a natural population of the great tit parus major. *The American Naturalist*, 164(3):E62–E72, 2004.

[11] S. Milgram. The small world problem. *Phychology Today*, 1:61–67, 1967.

[12] M. E. J. Newman. The structure and function of complex networks. *SIAM Rev.*, 45(2):167–256, 2003.

[13] M. E. J. Newman. *Networks: an Introduction*. Oxford University Press, 2010.

[14] G. Palla, I. Derenyi, I Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature Letters*, 435(7043):814–818, 2005.

[15] Ioannis Psorakis, Stephen Roberts, Mark Ebden, and Ben Sheldon. Overlapping community detection using bayesian non-negative matrix factorization. *Phys. Rev. E*, 83:066114, Jun 2011.

[16] M. Rosvall. *Information Horizons in a Complex World*. PhD thesis, Department of Physics, Umea University, 2006. ISBN 91-7264-117-7.

[2] for each site $l = \{1, ..., L\}$ we usually have a different number of gathering events K_l