A random effects stochastic block model for joint community detection in multiple networks with applications to neuroimaging

Subhadeep Paul, The Ohio State University
Yuguo Chen, University of Illinois at Urbana-Champaign

Abstract

Motivated by multi-subject experiments in neuroimaging studies, we develop a modeling framework for joint community detection in a group of related networks, which can be considered as a sample from a population of networks. The proposed random effects stochastic block model facilitates the study of group differences and subject-specific variations in the community structure. The model proposes a putative mean community structure which is representative of the group or the population under consideration but is not the community structure of any individual component network. Instead, the community memberships of nodes vary in each component network with a transition matrix, thus modeling the variation in community structure across a group of subjects. To estimate the quantities of interest we propose two methods, a variational EM algorithm, and a model-free “two-step” method based on either spectral or non-negative matrix factorization (NMF). Our NMF based method Co-OSNTF is of independent interest and we study its convergence properties to a stationary point. We also develop a resampling-based hypothesis test for differences in community structure in two populations both at the whole network level and node level. The methodology is applied to a publicly available fMRI dataset from multi-subject experiments involving schizophrenia patients. Our methods reveal an overall putative community structure representative of the group as well as subject-specific variations within each group. Using our network level hypothesis tests we are able to ascertain statistically significant difference in community structure between the two groups, while our node level tests help determine the nodes that are driving the difference.

KEY WORDS: Community detection; Neuroimaging; Non-negative matrix factorization; Population of networks; Random effects stochastic block model;

1 Introduction

Network analysis has received a plethora of multi-disciplinary interest in the last two decades due to its various scientific and industrial applications in a variety of fields including genetics, neuroscience, ecology, economics and social sciences. A relatively new but rapidly growing application area of network science is neuroimaging, where it is used in the analysis of anatomical and functional connectivity among the brain regions (See Rubinov and Sporns (2010); Bullmore and Sporns (2009); Hutchison et al. (2013); Sporns (2014) for reviews). In network neuroscience, a typical approach is to construct functional brain networks based on measures of inter-regional associations obtained from various sources of measurements, including the functional magnetic resonance imaging (fMRI)

---

1This work was supported in part by National Science Foundation grant DMS-1406455
2This work made use of the Illinois Campus Cluster, a computing resource that is operated by the Illinois Campus Cluster Program (ICCP) in conjunction with the National Center for Supercomputing Applications (NCSA), which is supported by funds from the University of Illinois at Urbana-Champaign.
3E-mail: paul.963@osu.edu
4E-mail: yuguo@illinois.edu
with blood oxygen level dependent (BOLD) signals (Bassett et al. 2011; Simpson et al. 2013a). Modern neuroimaging experiments typically involve multiple subjects and multiple trials across the subjects. Various properties of the functional network (e.g., community structure, modularity, resilience, connectivity, degree, rich club organization) are then investigated and contrasted among the subjects or groups of subjects (Bullmore and Sporns 2009; Van Den Heuvel and Pol 2010). The inter-subject and inter-group variations in many such network metrics have been related to cognitive ability and diseases in the literature (Bassett et al. 2011; Stevens et al. 2012; Hutchison et al. 2013; Jones et al. 2012; Wang et al. 2013; Braun et al. 2015; Yu et al. 2012a; Lynall et al. 2010; Alexander-Bloch et al. 2012).

However, due to the limitation of imaging instruments, physiological differences among the subjects, and responses to changing environmental conditions other than the experimental condition, the measured networks and consequently the network community structures vary from subject to subject within a group of subjects or even from trial to trial within a subject (Stevens et al. 2012; Simpson et al. 2013b; Moussa et al. 2012; Weber et al. 2013). In many neuroimaging studies, the researchers are interested in inferring and contrasting community structures between two populations, a healthy control population and a patient population with some condition. In a typical experimental setup, the subjects in the two groups serve as random samples from these two populations. Hence to facilitate comparison of populations beyond those in terms of single-value network summary measures (e.g., modularity), it is important to build statistical models for a random sample of networks from a population of networks. The related issue of consistency of network modules across different networks, both at module level and at node level, is investigated in Steen et al. (2011). A number of researchers have also investigated the consistency of brain modules across subjects and differences between two groups of subjects (Moussa et al. 2012; Alexander-Bloch et al. 2012; GadElkarim et al. 2012).

Unfortunately most of the literature on networks deal with a single instance of a network. This is primarily due to the fact that network data collection usually involves observing a network at one time point or tracking its evolution over time. However modern application of networks in neuroscience brings a unique challenge and opportunity in terms of multiple instances of interactions among the same set of nodes through measurements on multiple subjects. The central question then is how to quantify the uncertainty in community structure due to subject specific variations within a group of subjects, so that two groups (populations) of subjects can be statistically compared.

We approach this problem by developing methodology in the spirit of random effects linear models, which will help us separate systematic variations from variations due to sampling noise. In particular we propose a random effects stochastic block model (RESBM), parameterized by a putative mean community assignment matrix, a transition probability matrix, and appropriate block parameters. The putative mean community assignment matrix is a group level parameter, and the true community assignment matrices for each of the component networks are random variables generated from this putative mean community assignment through the transition probability matrix. We develop two estimation strategies to estimate the parameters and other quantities of interest, a variational EM algorithm and a model-free two-step approach based on non-negative matrix fac-
torization and spectral clustering. We also develop resampling based two-sample hypothesis tests to compare two populations of networks in terms of their network level and node level community structures.

The rest of the article is organized as follows. In Section 2 we describe a group of networks as a multi-layer network, and discuss existing statistical models for multi-layer networks with community structure. In Section 3 we describe the random effects stochastic block model. In Sections 4 and 5 we develop the estimation strategies. In Section 6 we develop the two-sample hypothesis tests. Section 7 studies the estimation and inference methods in simulated networks under several scenarios and several metrics. Finally in Section 8 we apply the methods to a real data set on Schizophrenia.

2 Models for multi-layer networks with community structure

A group of related interactions on the same set of nodes can be represented as a multi-layer network (Kivelä et al. 2014; Nicosia and Latora 2014; Boccaletti et al. 2014; Paul and Chen 2016a), where each network layer or type of edge represents a component network of the group. We define a multi-layer network on a set of $n$ nodes $V$, as a collection of graphs or networks $\mathcal{G} = \{G^{(1)}, \ldots, G^{(M)}\}$ with the common node set $V$. The component graphs of the collection, called the layers, may represent different types of interaction or the same type of interaction measured over multiple subjects or trials. We also assume the component graphs to be unweighted and undirected. Hence to each component layer $m$ we can associate an $n \times n$ square and symmetric adjacency matrix $A^{(m)}$, such that $A^{(m)}_{ij}$ is 1 if there is an edge between nodes $i$ and $j$ in layer $m$ of the multi-layer network and 0 otherwise. We will refer to the collection of the $M$ adjacency matrices as the adjacency tensor. For each layer, we can also define a normalized Laplacian matrix as $L^{(m)} = D^{(m)} - 1/2 A^{(m)} D^{(m)} - 1/2$, where $D^{(m)}$ is a diagonal matrix whose elements are the $m$th layer degrees of the nodes defined as $D^{(m)}_{ii} = \sum_j A^{(m)}_{ij}$. The $M$ normalized Laplacian matrices together form the $n \times n \times M$ three-way normalized Laplacian tensor.

There has been a recent surge in analysis of multi-layer and dynamic networks. A very common and widely investigated property of complex networks is community (or modular) structure. The multi-layer networks observed in the nature are also known to exhibit community structure (Mucha et al. 2010; Bazzi et al. 2014; Kivelä et al. 2014; Nicosia and Latora 2014; Boccaletti et al. 2014; Peixoto 2015). The multi-layer stochastic block model (MLSBM) is a statistical model for such multi-layer networks with community structure (Han et al. 2015; Valles-Catala et al. 2016; Paul and Chen 2016a; Stanley et al. 2016; Peixoto 2015; Barbillon et al. 2017; Paul and Chen 2016b, 2017). Recently De Bacco et al. (2017) proposed a more flexible multi-layer mixed membership stochastic block model that allows overlapping clusters. Most of the models described in the literature, with the exception of the strata-MLSBM of Stanley et al. (2016), are constrained by the fact that they assume the community structure to be the same across all layers. The estimation task is usually then to estimate this consensus community structure by fusing information from all layers. However, in some situations, e.g., in the multi-subject neuroimaging studies, it may be desirable to model
the variation in community structure in different layers along with finding a consensus clustering. The existing models are not flexible enough to model such data.

In a partial remedy of the situation, Stanley et al. (2016) introduced the strata-MLSBM where the community assignments vary across stratas but stay the same within the same strata. Within a strata, they further constraint the block model probability matrix to be identical across layers. A Bayesian nonparametric mixture model for jointly estimating community structure and identifying groups of networks with similar community structure in a collection of exchangeable networks was proposed in Reyes and Rodriguez (2016). A model similar in spirit to our proposed model is the Bayesian hierarchical mixed membership stochastic block model for an ensemble of networks proposed in Sweet et al. (2014).

In this paper we propose RESBM, a general and flexible modeling framework which contains the MLSBM as a special case. The model assumes the existence of a putative mean community structure which is representative of the group or the population, but is not necessarily the actual community structure in any of the subjects in the sample. Instead, each node in each of the subject networks deviates randomly from this putative community structure with an identical transition probability matrix. We formally define the model in the next section.

3 Random effects stochastic block model

To each node \(i\) of a network we associate a \(k\) dimensional community assignment vector \(x_i\), which takes value 1 in exactly one place and 0’s everywhere else. The location of 1 in the vector indicates the community the node belongs to. We call a matrix \(X \in \mathbb{R}^{n \times k}\) a community assignment matrix of the nodes of a network, if each row of the matrix is a community assignment vector for one of the \(n\) nodes in the network. Let \(\bar{Z} \in \mathbb{R}^{n \times k}\) be a putative mean community assignment matrix for a group or population of \(M\) networks. This is a fixed group or population level parameter. For each member network \(m\) in the sample, each node \(i\) can randomly switch its community label from this putative mean structure with an identical transition probability matrix. We formally define the model in the next section.

A unit vector, similar to the community assignment vector, is a vector whose components are all zeros except for one component that is one.

Formally we use the multivariate Bernoulli distribution and a transition probability matrix to describe the random deviation from the putative mean structure. A \(k\)-dimensional random vector \(Y\) follows a multivariate Bernoulli distribution with parameter \(p = (p_1, \ldots, p_k)\), \(\sum_{i=1}^{k} p_i = 1\), if for unit vector \(u = (u_1, \ldots, u_k)\),

\[
P(Y = u) = p_i, \quad \text{if } u_i = 1 \text{ and } u_j = 0 \text{ for } j \neq i.
\]

A unit vector, similar to the community assignment vector, is a vector whose components are all zeros except for one component that is one.

For each member network \(m\), the community assignment matrix of the network \(Z^{(m)}\) is drawn in the following way. For each node \(i\), the vector of community assignments is given by

\[
Z_i^{(m)} \sim \text{Multivariate-Bernoulli}(\bar{Z}_iT), \quad i = 1, \ldots, n, \quad m = 1, \ldots, M,
\]
where $Z_i^{(m)}$ and $\bar{Z}_i$ are the $i$th row of $Z^{(m)}$ and $\bar{Z}$, respectively. Here $T$ denotes the $k \times k$ non-negative transition probability matrix among the communities, with its diagonal elements being $\{\eta_1, \ldots, \eta_k\}$ and the off-diagonal elements in each row $q$ summing to $1 - \eta_q$. The vectors $Z_i^{(m)}$ are independent for all $i$ and $m$. However, the elements of the vector $Z_{iq}^{(m)}$, $q = 1, \ldots, k$, are dependent since $Z_i^{(m)} = 1$ mandates that $Z_{iq'}^{(m)} = 0$ for any $q' \neq q$.

We can also write the expectation and variance of the random vectors $Z_i^{(m)}$s as follows,

$$
E[Z_i^{(m)}] = \bar{Z}_iT = T_q, \quad Var[Z_i^{(m)}] = diag(\bar{Z}_iT) - T^T\bar{Z}_iT = diag(T_q) - T_q^TT_q,
$$

(3.2)

where $q$ is the putative mean community label of node $i$, $T_q$ is the $q$th row of $T$ and $diag(X)$ for any vector $X$ represents the diagonal matrix whose diagonal is the vector $X$. Here, and throughout the paper $T$ in superscript denotes the matrix transpose. Consequently $Z_i^{(m)}$ is a Multivariate-Bernoulli random unit vector with $T_q$ as the vector of parameters (probabilities). This implies that for each member network, a node $i$ that belongs to the mean community $q$, gets assigned to the same community as its mean community assignment with probability $\eta_q$ and a different community with probability $1 - \eta_q$. While we do not put any restriction on the transition probability matrix, we note that the model is most interesting when $\eta_q$ is large as compared to the remaining elements in the $q$th row. The model then can be interpreted in the context of multi-subject networks as follows. While most nodes retain their putative mean community memberships for the individual subject networks, a few randomly selected nodes change their memberships to another community according to probabilities from the transition probability matrix.

Given the community assignment matrices $\{Z^{(1)}, \ldots, Z^{(M)}\}$, the edges in the $M$ member networks are independently generated following a Bernoulli distribution,

$$
A_{ij}^{(m)}|Z^{(m)} \sim Bernoulli(P_{ij}^{(m)}), \quad i, j = 1, \ldots, n, \quad m = 1, \ldots, M.
$$

(3.3)
The Bernoulli probabilities can be modeled as a $k$ class stochastic block model (SBM) or a $k$ class degree-corrected stochastic block model with appropriate identifiability constraints. We focus only on the SBM in this paper. We have

$$A_{ij}^{(m)}|(Z_{iq}^{(m)} = 1, Z_{jl}^{(m)} = 1) \sim \text{Bernoulli}(\pi_{ql}^{(m)}), \quad q, l = 1, \ldots, k, \quad m = 1, \ldots, M.$$  

The model is schematically represented in Figure 1. However the model is not identifiable without further constraints. Similar to the discussion in Matias and Miele (2017) in the context of dynamic networks, the community labels might get switched between two layers and still give the same model, leading to incorrect inference. Hence we need certain constraints on the matrices $\{\pi^{(1)}, \ldots, \pi^{(M)}\}$ such that the communities are identifiable at all layers. We use the constraint that the diagonal elements of the matrices are identical in each layer, i.e., the vector $\{\pi_{11}^{(m)}, \ldots, \pi_{kk}^{(m)}\}$ is the same for all $m = 1, \ldots, M$ (Matias and Miele 2017).

We can also use the restricted multi-layer stochastic block model (RMLSBM) proposed in Paul and Chen (2016a) to jointly model the Bernoulli probabilities $P_{ij}^{(m)}$ in this framework as follows,

$$A_{ij}^{(m)}|(Z_{iq}^{(m)} = 1, Z_{jl}^{(m)} = 1) \sim \text{Bernoulli}\left(\frac{\exp(\pi_{ql}^{(m)} + \beta_m)}{1 + \exp(\pi_{ql}^{(m)} + \beta_m)}\right), \quad q, l = 1, \ldots, k, \quad m = 1, \ldots, M,$$

with the standard constraint $\sum_m \beta_m = 0$. We do not have the issue of non-identifiability from label switching in this case, since the matrix $\pi_{ql}^{(m)}$ does not change with layers and facilitates the identification of possibly permuted community labels in different layers. For the rest of the paper we primarily focus on the unrestricted multi-layer stochastic block model with the constraint of identical diagonal in each layer as the final step that generates $A$ in the schema shown in Figure 1.

Our estimation goals from this model include estimation of community assignments in each layer $Z_{iq}^{(m)}$, the putative mean community assignment matrix $\overline{Z}$, and the transition probability matrix $T$. Thus from this model we can infer an overall community assignment for the group of networks and obtain a quantification of the variability in this group putative community structure through the transition probability matrix.

### 4 A variational expectation-maximization estimator

In what follows we describe two methods to perform the estimation goals described earlier. The first method we describe is approximate maximum likelihood estimation (MLE) through variational expectation-maximization (EM) algorithm, first introduced in the context of standard SBM in Daudin et al. (2008) and later extended to multi-layer SBMs in Han et al. (2015), Paul and Chen (2016a) and Barbillon et al. (2017), and to dynamic SBM in Matias and Miele (2017). The asymptotic consistency and limiting distribution of the parameter estimates for the case of standard SBM are investigated in Celisse et al. (2012) and Bickel et al. (2013) respectively.

Below we derive the update rules for variational EM algorithm approximation to the MLE of the RESBM parameters. For this purpose we view the RESBM from a mixture model perspective and pose the parameters $Z_{iq}$s as random variables generated from a multinomial distribution with
parameters $\alpha = \{\alpha_1, \ldots, \alpha_k\}$. We denote the unobserved variables $\bar{Z}$ and $Z^{(m)}$ together as $X$ and the model parameters $T$, $\pi$ and $\alpha$ together as $\theta$. Further we denote the observed log-likelihood of the model as $l(A, \theta)$. The complete data log-likelihood, which is the joint likelihood of the observed data and the unobserved model community assignment variables, is given by

$$\log P(A, X, \theta) = \sum_{i=1}^{n} \sum_{q=1}^{k} \bar{Z}_{iq} \log \alpha_q + \sum_{m=1}^{M} \sum_{i=1}^{n} \sum_{1 \leq q, l \leq k} \bar{Z}_{iq} Z_{ql}^{(m)} \log T_{ql}$$

$$+ \sum_{m=1}^{M} \sum_{1 \leq i < j \leq n} \sum_{1 \leq q, l \leq k} \bar{Z}_{iq} Z_{jl}^{(m)} \{A_{ij}^{(m)} \log(\pi_{ql}^{(m)}) - (1 - A_{ij}^{(m)}) \log(1 - \pi_{ql}^{(m)})\}.$$

The likelihood of the observed data can be obtained by summing the complete data likelihood over all possible values of the unobserved variable in $X$, i.e., $l(A, \theta) = \log \int_X P(A, X, \theta)$. However, as with most mixture models, the number of all possible assignments grows exponentially and the sum quickly becomes computationally intractable even for moderate $n$. A remedy is to use the EM algorithm (Dempster et al. 1977). However, in network data the conditional distribution $P(X|A)$ is not tractable either (Daudin et al. 2008). Hence instead we use the variational EM algorithm, where for any distribution $R(X)$ over the unobserved variables, the observed likelihood is approximated by a function $J(R(X), \theta)$ involving expectation of the complete data log-likelihood under the distribution $R(X)$ and an entropy term.

The variational approximation to the EM algorithm that we employ here concentrates the search for optimal class assignments to a smaller set by assuming that the class assignments follow a product form multinomial distribution with parameters known as variational parameters. Let $R(X)$ be the variational approximating distribution. Then the function that approximates $l(A, \theta)$ is

$$J(R(X), \theta) = \int_X R(X) \log \left( \frac{P(A, X, \theta)}{R(X)} \right) = E_R[\log P(A, X, \theta)] + H(R),$$

where $H(R)$ is the entropy of $R$. The variational algorithm proceeds by iteratively maximizing $J(R(X), \theta)$ with respect to the parameters of $R$ and $\theta$. It can be shown that

$$J(R(X), \theta) = l(A, \theta) - KL[R(X), P(X|A)].$$

Hence the objective function is a lower bound on $l(A, \theta)$ and it can be shown that at each iteration of the optimization problem, the likelihood is non-decreasing. Further the variational distribution $R(X)$ has the interpretation of being an approximation of $P(X|A, \theta)$ in the Kullback-Leibler divergence.

Here we define $R(X)$ to have the following form of the product of multinomial densities:

$$R(X) = \prod_{i=1}^{n} \prod_{q=1}^{k} \bar{Z}_{iq} \frac{\tau_{iq}}{\epsilon_{iq}} \times \prod_{i=1}^{n} \prod_{m=1}^{M} \prod_{1 \leq q, l \leq k} (\epsilon_{iql}^{(m)}) Z_{ql}^{(m)} \bar{Z}_{iq},$$

where $\tau$ and $\epsilon$ are the variational parameters. For ease of exposition we further define variational parameters $\tau^{(m)}$ as $\tau_{iq}^{(m)} = \sum_{q=1}^{k} \tau_{iq} \epsilon_{iql}^{(m)}$ for $m = 1, \ldots, M$. Under the variational constraint on the
distribution of $X$, the objective function can be written as,

$$J_R(\theta, \bar{\tau}, \epsilon) = E_R[\log P(A, X, \theta)] + H(R)$$

$$= \sum_{i=1}^{n} \sum_{q=1}^{k} \tau_{iq} (\log \alpha_q - \log \bar{\tau}_{iq})$$

$$+ \sum_{m=1}^{M} \sum_{1 \leq i < j \leq n} \sum_{1 \leq l \leq k} \tau_{iq}^{(m)} \tau_{jl}^{(m)} \left\{ A_{ij}^{(m)} \log(\pi_{ql}^{(m)}) + (1 - A_{ij}^{(m)}) \log(1 - \pi_{ql}^{(m)}) \right\}$$

$$+ \sum_{m=1}^{M} \sum_{i=1}^{n} \sum_{1 \leq l \leq k} \tau_{iq} \epsilon_{iql}^{(m)} (\log T_{ql} - \log \epsilon_{iql})$$

(4.1)

with the constraints that $\sum_q \tau_{iq} = 1$ for all $i$, $\sum_q \tau_{iq}^{(m)} = 1$ for all $i$ and $m$, and $\tau_{il}^{(m)} = \sum_{q=1}^{k} \tau_{iq} \epsilon_{iql}^{(m)}$ for all $i, l, m$. Note that in the third line of the above equation, we have used the fact that $Z_{iq}^{(m)}$ and $Z_{jl}^{(m)}$ are independent random variables and hence the expectation of their product is the product of their expectations, while we have the fourth line since

$$E(Z_{iq}Z_{il}^{(m)}) = P(Z_{iq} = 1, Z_{il}^{(m)} = 1) = P(Z_{iq} = 1)P(Z_{il}^{(m)} = 1|Z_{iq} = 1) = \bar{\tau}_{iq} \epsilon_{iql}^{(m)}.$$

In what follows we will denote the term $\{ A_{ij}^{(m)} \log(\pi_{ql}^{(m)}) + (1 - A_{ij}^{(m)}) \log(1 - \pi_{ql}^{(m)}) \}$ as $\log(b_{ijlp}^{(m)})$.

In the variational expectation (VE) step of the following variational EM algorithm, we compute the variational parameters keeping the model parameters fixed, while in the M step we compute the model parameters. To derive the VE step we note that differentiating $J$ with respect to $\epsilon_{iql}^{(m)}$ and $\bar{\tau}_{iq}$ respectively yields,

$$\frac{\partial J}{\partial \epsilon_{iql}^{(m)}} \triangleq \bar{\tau}_{iq} \log T_{ql} - \bar{\tau}_{iq} (\log \epsilon_{iql}^{(m)} + 1) + \sum_{j \neq i}^{k} \sum_{p=1}^{k} \tau_{iq} \tau_{jp}^{(m)} \log(b_{ijlp}^{(m)}),$$

$$\frac{\partial J}{\partial \bar{\tau}_{iq}} \triangleq \log \alpha_q - \log \bar{\tau}_{iq} - 1 + \sum_{m=1}^{M} \sum_{l=1}^{k} \sum_{j \neq i}^{k} \epsilon_{iql}^{(m)} \tau_{jl}^{(m)} \log(b_{ijql}^{(m)}) + \sum_{m=1}^{M} \sum_{l=1}^{k} \epsilon_{iql}^{(m)} \log \left( \frac{T_{ql}^{(m)}}{\epsilon_{iql}^{(m)}} \right).$$

These two equations, along with the Lagrange multipliers for the constraints in (4.1), give the following fixed point equations which are used to update the variational parameters iteratively:

$$\epsilon_{iql}^{(m)} \propto \exp[\log(T_{ql}) + \sum_{j \neq i}^{k} \sum_{p=1}^{k} \tau_{jp}^{(m)} \log(b_{ijlp}^{(m)})],$$

(4.2)

$$\bar{\tau}_{iq} \propto \exp[\log \alpha_q + \sum_{m} \sum_{l} \epsilon_{iql}^{(m)} \log \left( \frac{T_{ql}^{(m)}}{\epsilon_{iql}^{(m)}} \right) + \sum_{m} \sum_{l} \sum_{j \neq i}^{k} \epsilon_{iql}^{(m)} \tau_{jl}^{(m)} \log(b_{ijql}^{(m)})],$$

(4.3)

and

$$\tau_{il}^{(m)} \propto \sum_{q=1}^{k} \bar{\tau}_{iq} \epsilon_{iql}^{(m)}.$$  

(4.4)
For the M step we have the following closed form update steps,

\[ T_{ql} \propto \sum_m \sum_i \bar{\tau}_{iq}^{(m)} c_{iql}^{(m)}, \quad (4.5) \]

\[ \bar{\pi}_{ql}^{(m)} = \frac{\sum_{1 \leq i < j \leq n} \tau_{iq}^{(m)} \tau_{jl}^{(m)} A_{ij}^{(m)}}{\sum_{1 \leq i < j \leq n} \tau_{iq}^{(m)} \tau_{jl}^{(m)}}, \quad q \neq l, \quad (4.6) \]

\[ \bar{\pi}_{qq}^{(m)} = \frac{\sum_m \sum_{1 \leq i < j \leq n} \tau_{iq}^{(m)} \tau_{jq}^{(m)} A_{ij}^{(m)}}{\sum_m \sum_{1 \leq i < j \leq n} \tau_{iq}^{(m)} \tau_{jq}^{(m)}}, \quad (4.7) \]

\[ \alpha_q = \frac{1}{n} \sum_i \bar{\tau}_{iq}. \quad (4.8) \]

The proportionalities in the above algorithm are turned into equalities through normalization using the constraints on the parameters.

5 Two-step spectral/NMF and ML algorithm

Next we describe a two-step approach to estimate the unknown quantities in RESBM. The first step solves an optimization problem involving a joint objective function to obtain the community assignment matrices in each layer \( Z^{(1)}, \ldots, Z^{(M)} \), and the putative mean community assignment matrix \( \bar{Z} \) simultaneously. This is a fully non-parametric step not dependent on any model. The transition probability matrix \( T \) is then obtained in the second step through conditional maximum likelihood (ML) conditioned on the estimated layer-wise and mean assignments. For the first step of the method we describe two algorithms involving a non-negative matrix factorization (NMF) and a spectral clustering based objective function respectively.

5.1 Co-regularized orthogonal symmetric non-negative matrix tri-factorization algorithm

We propose an algorithm for the first step of the two-step method based on orthogonal symmetric non-negative matrix tri-factorization (OSNTF) with smoothing. The proposed algorithm combines the ideas in linked matrix factorization (Tang et al. 2009) and co-regularized spectral clustering (Kumar et al. 2011). Similar to the co-regularized spectral clustering, the proposed method involves minimizing an objective function with two terms. The first term is the sum of Frobenius norm objective functions for single layer OSNTF (Paul and Chen 2016c) for each layer, and the second term is a smoothness penalty on the factor matrices obtained at each layer to make the subspaces spanned by the factor matrices closer to each other. The method is also in the same spirit as other non-negative matrix factorization based multi-view learning methods proposed in literature (Liu et al. 2013; Mankad and Michailidis 2013). However, the orthogonality constraints on the factor matrices in addition to the non-negativity constraints along with the smoothness penalty make the proposed method more appropriate for joint community detection in multiple networks.
Briefly, the technique is as follows. The KKT conditions for the objective in (5.2) are multiplicative updates described in Lee and Seung (2001), Ding et al. (2006) and Mirzal (2014). where \( \nabla \) is the gradient of the objective function with respect to a parameter \( \xi \) with respect to \( U^{(m)}, U^* \) are user-chosen tuning parameters (scalars). Note that this is a constraint optimization problem on Stiefel manifold with additional non-negativity constraints. To solve this optimization problem we employ the method of Lagrange multipliers for the orthogonality constraints and then derive multiplicative update rules. The Lagrangian objective function with the orthogonality constraints incorporated is,

\[
\xi \triangleq \sum_{m=1}^{M} \left\{ \| L^{(m)} - U^{(m)} S^{(m)} U^{(m)\top} \|_F^2 + \lambda_m (k - \| U^{(m)} T U^* \|_F^2) \right\} + \operatorname{tr} (\Lambda_{U^*} (U^* T U^* - I))
\]

\[
\equiv \operatorname{tr} \left( \sum_{m=1}^{M} \left\{ -2 U^{(m)\top} L^{(m)} U^{(m)} S^{(m)} + U^{(m)} S^{(m)} U^{(m)\top} U^{(m)} S^{(m)} U^{(m)\top} - 2 \lambda_m U^{(m)} T U^* U^* U^{(m)} \right. \right.
\]

\[
\left. \left. + \Lambda_{U^{(m)}} U^{(m)\top} U^{(m)} \right\} + \Lambda_{U^*} U^* T U^* \right) ,
\]

where \( \Lambda_{U^*} \geq 0 \) and \( \Lambda_{U^{(m)}} \geq 0 \) are \( k \times k \) non-negative symmetric matrices of Lagrange multiplier parameters. Here and throughout the paper \( \operatorname{tr}(\cdot) \) denotes the matrix trace. The goal is now to minimize this new objective function under the constraints that \( U^{(m)} \geq 0, U^* \geq 0, S^{(m)} \geq 0 \).

To derive an algorithm for the optimization problem, we follow the derivation techniques for multiplicative updates described in Lee and Seung (2001), Ding et al. (2006) and Mirzal (2014). Briefly, the technique is as follows. The KKT conditions for the objective in (5.2) are

\[
U^{(m)} \geq 0, \quad U^* \geq 0, \quad S^{(m)} \geq 0,
\]

\[
\nabla \xi |_{U^{(m)}} \geq 0, \quad \nabla \xi |_{U^*} \geq 0, \quad \nabla \xi |_{S^{(m)}} \geq 0,
\]

\[
U^{(m)} \odot \nabla \xi |_{U^{(m)}} = 0, \quad U^* \odot \nabla \xi |_{U^*} = 0, \quad S^{(m)} \odot \nabla \xi |_{S^{(m)}} = 0,
\]

where \( \nabla \xi |_{U^{(m)}}, \nabla \xi |_{U^*}, \nabla \xi |_{S^{(m)}} \) are gradients of the objective function (5.2) with respect to \( U^{(m)}, U^* \) and \( S^{(m)} \) and \( \odot \) represents the Hadamard (element-wise) product. The equations in the last line of the KKT conditions are known as the complimentary slackness conditions which can be used to derive the multiplicative update rules (MUR). If the gradient \( \nabla \xi \) with respect to a parameter can be written in the form \( \nabla \xi = [\nabla \xi]^+ - [\nabla \xi]^− \), where \( [\nabla \xi]^+, [\nabla \xi]^− \geq 0 \), then the multiplicative update rule would be

\[
X \leftarrow X \odot \left( \frac{([\nabla \xi]^−)_{ik}}{([\nabla \xi]^+)_{ik}} \right) ^\eta ,
\]
where \((\cdot)\)\(^{\eta}\) represents raising each element of the matrix to the power \(\eta\), and \(0 < \eta \leq 1\) is the learning rate. We take \(\eta = 1/2\). The inner division in the rightmost term is also element-wise.

The gradient with respect to \(U^{(m)}\) is given by,

\[
\nabla \xi|_{U^{(m)}} \triangleq -4L^{(m)}U^{(m)}S^{(m)} + 4U^{(m)}S^{(m)}U^{(m)\top}U^{(m)}S^{(m)} - 4\lambda_m U^{\ast T}U^{(m)} + 2U^{(m)}\Lambda_{U^{(m)}}. \tag{5.3}
\]

Hence the complementary slackness conditions imply the following multiplicative update rule for \(U^{(m)}:\)

\[
U^{(m)} \leftarrow U^{(m)} \odot \left( \frac{(L^{(m)}U^{(m)}S^{(m)} + \lambda_m U^{\ast T}U^{(m)})_{ik}}{(U^{(m)}(S^{(m)}U^{(m)\top}U^{(m)})S^{(m)} + \Lambda_{U^{(m)}})_{ik}} \right)^{1/2}. \tag{5.4}
\]

Similarly taking the gradient with respect to \(U^\ast\) is given by

\[
\nabla \xi|_{U^\ast} \triangleq -4 \sum_m \lambda_m U^{(m)\top}U^\ast + 2U^\ast \Lambda_{U^\ast}, \tag{5.5}
\]

and by the complimentary slackness conditions the multiplicative update rule for \(U^\ast\) is

\[
U^\ast \leftarrow U^\ast \odot \left( \frac{\left( \sum_m \lambda_m U^{(m)\top}U^\ast \right)_{ik}}{(U^\ast \Lambda_{U^\ast})_{ik}} \right)^{1/2}. \tag{5.6}
\]

Now taking the gradient with respect to \(S^{(m)}\) we have,

\[
\nabla \xi|_{S^{(m)}} = -2U^{(m)\top}L^{(m)}U^{(m)} + 2U^{(m)\top}U^{(m)}S^{(m)}U^{(m)\top U^{(m)}},
\]

from which we get the update rule as,

\[
S^{(m)} \leftarrow S^{(m)} \odot \left( \frac{(U^{(m)\top}L^{(m)}U^{(m)})_{ik}}{(U^{(m)\top}U^{(m)}S^{(m)}U^{(m)\top U^{(m)}})_{ik}} \right)^{1/2}. \tag{5.7}
\]

We use the multiplicative update rules in Equations (5.4), (5.6) and (5.7) to minimize the objective function \(\xi\). The convergence properties of these update rules are discussed in the next section. However, we note that the values of \(\Lambda_{U^{(m)}}\) and \(\Lambda_{U^\ast}\) in the previous update rules are unknown. They are approximated by equating the partial derivatives in the KKT conditions to zero ignoring the non-negativity constraints, but using the orthogonality constraints. Multiplying (5.3) by \(U^{(m)\top}\) and equating to 0, along with the relation \(U^{(m)\top}U^{(m)} = I\), we get,

\[
\Lambda_{U^{(m)}} = 2U^{(m)\top}L^{(m)}U^{(m)}S^{(m)} + 2S^{(m)}S^{(m)} - 2\lambda_m U^{(m)\top}U^\ast U^{\ast T}U^{(m)}. \tag{5.8}
\]

Substituting the value of \(\Lambda_{U^{(m)}}\) into (5.3) we get,

\[
\nabla \xi|_{U^{(m)}} = 2U^{(m)\top}U^{(m)\top}L^{(m)}U^{(m)}S^{(m)} - 2L^{(m)}U^{(m)}S^{(m)} - 2\lambda_m U^{\ast T}U^{(m)}(U^{(m)\top}U^\ast U^{\ast T}U^{(m)}),
\]

\[
= 2(U^{(m)\top}U^{(m)\top}L^{(m)}U^{(m)}S^{(m)} + \lambda_m U^{(m)\top}U^{(m)\top}U^\ast U^{\ast T}U^{(m)}) - 2(L^{(m)}U^{(m)}S^{(m)} + \lambda_m U^{\ast T}U^{(m)}).
\]

Hence the update rule for \(U^{(m)}\) is,

\[
U^{(m)} \leftarrow U^{(m)} \odot \left( \frac{(L^{(m)}U^{(m)}S^{(m)} + \lambda_m U^{\ast T}U^{(m)})_{ik}}{(U^{(m)\top}U^{(m)\top}L^{(m)}U^{(m)}S^{(m)} + \lambda_m U^{(m)\top}U^{(m)\top}U^\ast U^{\ast T}U^{(m)})_{ik}} \right)^{1/2}. \tag{5.9}
\]
Again in order to solve for $\Lambda U^*$, we multiply the right hand side of (5.5) by $U^T$, equate to 0 and use the relation $U^T U = I$. Then,

$$\Lambda U^* = 2 \sum_m \lambda_m U^{(m)} U^{(m)T} U^*.$$  \hspace{1cm} (5.10)

Substitution the value of $\Lambda U^*$ in (5.5), we get

$$\nabla \xi |_{U^*} = -4 \sum_m \lambda_m U^{(m)} U^{(m)T} U^* + 4 \sum_m \lambda_m U^* U^{(m)T} U^* U^{(m)T} U^*.$$

Hence the update rule for $U^*$ is,

$$U^* \leftarrow U^* \odot \left( \frac{\sum_m \lambda_m U^{(m)} U^{(m)T} U^*}{\sum_m \lambda_m U^* U^{(m)T} U^* U^{(m)T} U^*} \right)^{1/2}.$$  \hspace{1cm} (5.11)

Hence our final algorithm consists of iteratively updating $U^{(m)}$, $S^{(m)}$ and $U^*$ according to update rules (5.9), (5.7) and (5.11) respectively. Note that $\lambda_m$s are user specified parameters. For our numerical experiments and real data analysis in this paper we choose $\lambda_m$s to be 0.01 for all $m$. This choice of $\lambda_m$s works well in our synthetic experiments and is similar to the choice made for joint non-negative factorization in Liu et al. (2013).

5.1.1 Convergence and implementation details

In this section we address certain issues with the multiplicative update rules and provide a convergence guarantee. First, zero locking is a well known phenomenon in MUR algorithms, where if any entry of the parameter matrices is zero at any point during the updates, then the MUR algorithm will stop updating that entry irrespective of a stationary point is reached or not (Mirzal 2014). However, the next result describes a remedy to this issue.

**Lemma 1.** If none of the $L^{(m)}$s have any zero column or zero row, and all the initial solutions contain elements that are strictly positive (i.e., $(U^{(m)})^{(0)} > 0$, $(S^{(m)})^{(0)} > 0$, $(U^*)^{(0)} > 0$), then under the update rules in Equations (5.4), (5.6) and (5.7), $(U^{(m)})^{(t)} > 0$, $(S^{(m)})^{(t)} > 0$, $(U^*)^{(t)} > 0$ at the $t$-th iteration for all $t > 0$. Moreover, if $\{\tilde{U}^{(m)}, \tilde{U}^*, \tilde{S}^{(m)}\}$ is a stationary point, then the update rules in Equations (5.4), (5.6) and (5.7) will stop updating.

The proof of this lemma along with that of the following theorem appears in the Appendix. The lemma says that under certain conditions on the input Laplacian matrices, if the initial solutions contain all strictly positive entries, then the solution at the $t$-th iteration also contains all positive entries. From a practical implementation point of view, our Laplacian matrices do not contain any zero columns or rows, since that would imply an isolated node and we will simply drop the node. We start with an initial solution that has all strictly isolated nodes and the lemma guarantees that the solutions at each iteration remain strictly positive. However, the shortcoming of the procedure is that it would fail to detect any stationary point that contains zeros in some entries.
Theorem 1. If the conditions in Lemma 1 are satisfied, then under the multiplicative update rules defined in Equations (5.4), (5.6) and (5.7), the objective function $\xi$ defined in (5.2) is monotonically non-increasing.

This theorem suggests that the update rules can reach a stationary point of the objective function $\xi$ provided the solution lies on the positive orthant of the feasible region (all elements of all matrices in the solution contain strictly positive entries) and we start with an initial solution which is in the positive orthant.

5.2 Co-regularized spectral clustering

An alternative method to estimate the parameters in first step is the co-regularized spectral clustering method. The method was proposed in Kumar et al. (2011) and its theoretical properties under the MLSBM were studied in Paul and Chen (2017). The method, applied to the normalized Laplacian tensor, solves the following optimization problem,

$$\hat{[U]}(1),...,[U](m),\hat{U}^* = \arg\max_{U(m)\in\mathbb{R}^{n,k},\forall m, U^*\in\mathbb{R}^{n,k}} \sum_{m=1}^{M} \{\text{tr}(U(m)^T L(m) U(m)) + \gamma_m \text{tr}(U^* U^* U(m)^T U^*)\}$$

(5.12)

where $U(1),...,U(M)$ and $U^*$ are $n \times k$ matrices with orthonormal columns. The optimization problem is solved through a fast alternating eigen-decomposition of the matrices $L(m) - \gamma_m U^* U^* T$ and $\sum_m \gamma_m U(m)^T U(m)$ (Kumar et al. 2011). Kumar et al. (2011) noted that the objective function in (5.12) is non-decreasing under the alternating maximization and hence the solutions are guaranteed to be local maxima of the objective function. The first part of the objective function is the usual normalized cut spectral clustering objective function for different layers, while the second part is a distance based penalty function that attempts to make the subspaces spanned by the layer-wise eigenvector matrices more similar to each other by penalizing the subspace distances from a mean subspace (See Paul and Chen (2017) for details of this interpretation). The weights $\gamma_m$'s should be chosen to address the user’s preference in the trade-off between optimizing within a layer and increasing subspace cohesion as well as to reflect the relative importance of different layers. For our experiments in this paper we choose $\gamma_m$'s to be 0.05 for all $m$.

5.3 The conditional ML step

The first step of the two-step method obtains $\hat{U}(1),...,\hat{U}(m),\hat{U}^*$, which are matrices in the Grassmann manifold. In the second step, we first obtain community assignments from each of the $\hat{U}(m)$s and $\hat{U}^*$, and then the transition probability matrix is obtained from the estimated matrices through a conditional maximum likelihood (ML) estimator. For co-regularized spectral clustering community assignment is accomplished through the k-means clustering of the rows of the matrices, while for Co-OSNTF this is performed by assigning the rows to the community corresponding to the largest element in a row. Let $\hat{Z}(1),...,\hat{Z}(M)$ be the layer-wise community assignment matrices and
\( \hat{Z}^* \) be the mean community assignment matrix obtained in this way. Then the conditional ML estimate of \( T \) is given by

\[
\hat{T}_{ql} = \frac{n_{ql}}{\sum_l n_{ql}},
\]

(5.13)

where \( n_{ql} = \frac{1}{M} \sum_m \sum_i I\{\hat{Z}_{iq} = 1, \hat{Z}_{il}^{(m)} = 1\} \), with \( I(\cdot) \) being the indicator function. We note that the same estimator for \( T \) can be obtained by equating the first moments as well. We have,

\[
\frac{1}{M} \sum_{m=1}^M \hat{Z}^{(m)} = \hat{Z}^* \hat{T} \quad \Rightarrow \quad \hat{T} = (\hat{Z}^{*T} \hat{Z}^*)^{-1} \hat{Z}^{*T} \frac{1}{M} \sum_{m=1}^M \hat{Z}^{(m)},
\]

(5.14)

which is identical (written in matrix terms) to the one in (5.13).

We finish this section by noting that there is a conceptual similarity of this two-step approach with the modular allegiance matrix methodology (the spectral version of which is included in our simulation study later) in the context of studying flexibility of nodes in dynamic functional connectivity of brain networks in Bassett et al. (2011) and Braun et al. (2015).

6 Two-sample hypothesis testing: whole network-level and node-level tests

We next develop procedures for performing a two-sample hypothesis test between network community structures of two populations of subjects. The problem of statistically testing for differences in community structure has received considerable attention recently in the neuroimaging literature (Alexander-Bloch et al. 2012; GadElkarim et al. 2012; Fujita et al. 2014; Glerean et al. 2016; Kujala et al. 2016). These papers argue that differences between populations might not be well captured using a single network property measure like modularity, but it might be more meaningful to look at some measure of how different the module structures in the populations are. A permutation test using the average of NMI between pairs of network community assignments was proposed in Alexander-Bloch et al. (2012). GadElkarim et al. (2012) used a node-wise community consistency measure between two community partitions, called “Scaled Inclusivity (SI)” defined in Steen et al. (2011), to assess how consistent a node’s module is in a subject network with a mean module assignment for the control group (obtained from community detection in the mean connectivity matrix). GadElkarim et al. (2012) then proposed statistical hypothesis tests based on the scaled inclusivity vectors for the two groups. Glerean et al. (2016) first obtained a consensus partition across the group using a “clustering clusters” technique similar in function to module allegiance method mentioned earlier, and then determined Scaled Inclusivity of nodes across subjects with this consensus. The more general problem of hypothesis testing involving networks in functional neuroimaging has also been addressed in the literature (Ginestet et al. 2014, 2017; Narayan and Allen 2016).

Let the two populations to be compared be denoted as \( A \) and \( B \) (in neuroimaging experiments these correspond to a group of healthy controls and a group of patients), and the sample size from each group be \( M_1 \) and \( M_2 \) respectively. Further let the estimated model parameters for the two
groups be $\tilde{Z}_A(T_A), \{Z^{(1)}_A, \ldots, Z^{(M_1)}_A\}$ and $\tilde{Z}_B(T_B), \{Z^{(1)}_B, \ldots, Z^{(M_2)}_B\}$. There are two notions of “group mean” of community structure giving rise to two natural ways of testing the difference between the two populations in modular organization. The first notion of “group mean” is $\tilde{Z}$, the putative mean community assignment matrix of the group of networks. Accordingly the notion of group mean at the node level is $\tilde{Z}_i$ for node $i$. However, note that $\tilde{Z}$ is not the expectation of the community assignment matrices $Z^{(m)}_s$ in the group. Instead the expectation is $\tilde{Z}T$. This implies that for any node, its expected community assignment vector in any network within the group is $\tilde{Z}_iT$. Hence the second notion of “group mean” is represented by $\tilde{Z}T$ at the network level and $\tilde{Z}_iT$s at the node level.

The first network level test statistic we propose is the distance between $\mathcal{R}(\tilde{Z}_A(T_A))$ and $\mathcal{R}(\tilde{Z}_B(T_B))$, the subspaces spanned by the columns of $\tilde{Z}_A(T_A)$ and $\tilde{Z}_B(T_B)$ respectively. Note that the columns of both matrices $\tilde{Z}_A(T_A)$ and $\tilde{Z}_B(T_B)$ span subspaces in the Grassmann manifold $\mathcal{G}(k, n)$ (Edelman et al. 1998). A common notion of distance between subspaces in the Grassmann manifold is in terms of norms of the matrix of sines of canonical angles between the subspaces (Stewart and Sun 1990; Edelman et al. 1998; Dong et al. 2014). Formally, we define the “Sine test” statistic as the Grassmann subspace distance between the putative mean community assignment subspaces,

$$S = \|\sin(\Theta(\mathcal{R}(\tilde{Z}_A(T_A)), \mathcal{R}(\tilde{Z}_B(T_B))))\|_F^2 = \frac{1}{2} \|\tilde{Z}_A(T_A)D_AD_A^T - \tilde{Z}_B(T_B)D_B\tilde{Z}_B^T(T_B)\|_F^2$$

$$= k - \text{tr}(\tilde{Z}_A(T_A)D_AD_A^T)\tilde{Z}_B(T_B)D_B\tilde{Z}_B^T(T_B)),$$

where $\Theta(\mathcal{R}(\tilde{Z}_A(T_A)), \mathcal{R}(\tilde{Z}_B(T_B)))$ is the matrix canonical angles between subspaces spanned by the columns of $\tilde{Z}_A(T_A)$ and $\tilde{Z}_B(T_B)$, $D_A = (\tilde{Z}_A(T_A)^T\tilde{Z}_A(T_A))^{-1}$ and $D_B = (\tilde{Z}_B(T_B)^T\tilde{Z}_B(T_B))^{-1}$ are diagonal matrices whose elements are the sizes of the communities. The $(i, j)$th element of the matrix $\tilde{Z}_A(T_A)D_AD_A^T$ is either 0 or 1 depending upon whether the two nodes $i$ and $j$ are in the same community or not. We call this matrix the co-module matrix. Hence the elements of $\tilde{Z}_A(T_A)D_AD_A^T$ are just the scaled version of the elements of the co-module matrix. Intuitively, the test statistic looks at how many times two nodes are in the same community in one of the groups while they are not in the same community in the other group, with some scaling factors to make sure that contribution from large communities is properly penalized. Large values of the test statistic will then indicate that the mean module structures in the two groups are markedly different.

It is not always clear whether one should work with the co-module matrix or the scaled version of it. In our experiments since the community sizes are similar in networks from both groups, we do not observe any difference in the results from the scaled and unscaled versions, and hence proceed with the scaled version. Note that this statistic does not require the community labels in the two groups to be aligned to each other, neither does it require the number of communities to be identical. However if we want test at the node level, which we describe next, then we need both to have the same number of communities in the two mean module structures and the community labels need to be aligned by solving the linear sum assignment problem. The test statistic for testing at node $i$ is then simply $S(i) = (\tilde{Z}_i(A) - \tilde{Z}_i(B))^2$.

The second test statistic corresponds to the second notion of “group mean” as described earlier.
For each node $i$, define the node level “multivariate-Bernoulli unit vector” (MUV) test statistic

$$MUV(i) = (\bar{Z}_{(A)} i T_A - \bar{Z}_{(B)} i T_B)^T (\bar{Z}_{(A)} i T_A - \bar{Z}_{(B)} i T_B).$$

The network level MUV statistic is then:

$$MUV = ||\bar{Z}_{(A)} T_A T_A^T - \bar{Z}_{(B)} T_B T_B^T||_F^2.$$ 

Intuitively, $\bar{Z}T$ is a better measure of the group mean since it is the expectation of the subject’s community assignment matrices. Consequently, we expect MUV test to be a better test than Sine test, since it takes into account the variation in community structure present in the population. We make a comparison of the performance of the two tests with estimates from various methods in a simulated network in Section 7.5.

In all the above cases it is difficult to derive asymptotic distribution of the test statistics. Hence we construct p-value for the test through a permutation test based on re-sampling from the observed networks. We combine the network samples together and sample without replacement from the combined sample of $M_1 + M_2$ networks to create two samples of sizes $M_1$ and $M_2$ respectively. We fit the RESBM to both samples using variational EM and two-step methods and compute the Sine test and MUV test statistics in each case. We repeat the procedure many times to construct the empirical distribution of the test statistic under the null hypothesis. Comparing the observed value of the test statistics with the constructed empirical distribution yields the p-values. For the node level tests, we can perform the same procedures. However, when we make inference using the p-values we need to account for multiple comparisons through a Family Wise Error Rate (FWER) or False Discovery Rate (FDR) correction (Benjamini and Hochberg 1995).

7 Performance on simulated networks

In this section we numerically compare the performance of the proposed methods along with some already available or baseline methods in groups of networks simulated from the RESBM. We compare the performance under three metrics: the average performance in community detection across the individual components of the network sample, the performance in estimating the putative mean community structure, and the accuracy in estimating the transition probability matrix. The normalized mutual information (NMI), an information theoretic measure of similarity between two community structures, is used to assess the accuracy of the estimated community assignments against the true assignments. The NMI between two community assignment vectors measures the degree to which one community assignment vector can be obtained from the knowledge of the other vector. It takes values between 0 and 1, where 0 indicates a random assignment (no overlap of information) and 1 indicates a perfect match, and the higher the value, the better the match. The accuracy of the estimated transition probability matrix is measured in terms of difference in Frobenius norm.

We generate the multi-layer networks from the RESBM in the following way. We first generate the group putative community labels from a multinomial distribution with $k$ classes and equal
probability for each class. The community labels of a fraction $\kappa$ of nodes are then randomly changed to one of the communities other than its original community. Hence the transition probability matrix $T$ will have $1 - \kappa$ as the diagonal elements, while the $k - 1$ off-diagonal elements in each row sum to $\kappa$. We call the fraction $\kappa$ variation factor since it is an indicator of how much variation there is in the community structure among the layers. We generate the true community assignments for the $M$ layers of the multi-layer network by repeating the above process. For each of the layers the edges between the nodes are then drawn from a stochastic block model in the following fashion. We first generate the vector of $k$ diagonal elements which is common for all layers (required for the model to be identifiable) as $\lambda \sim U(a, b)$, where $U(a, b)$ denotes the continuous uniform distribution with parameters $a$ and $b$. Next in each layer the lower half of the $k^2 - k$ off-diagonal elements are generated from $U(a/\rho, b/\rho)$ while the upper half are identical to the lower half. The parameter $\rho$ controls the signal to noise ratio (SNR), and in all our experiments we keep the value close to 2 in all layers. The average density of the layers is controlled by another parameter called degree multiplier. Roughly speaking, increasing the degree multiplier by 1 corresponds to an increase of 2% of maximum degree in the degree density per layer. As an example if we have 500 nodes in a layer, then a degree multiplier of 3 corresponds to average degree density per layer being $500 \times 0.06 = 30$.

7.1 Methods compared

In our simulations, we compare the performance of three algorithms from the two proposed methods along with a number of baseline and other available methods. We summarize below the methods we compare.

- **VarEM**: The variational EM algorithm for computing approximate MLE in RESBM.
- **Co-Spectral**: The two-step co-regularized spectral clustering with conditional MLE.
- **Co-OSNTF**: The two-step co-regularized orthogonal non-negative matrix tri-factorization with conditional MLE.
- **Ind. Spectral**: Spectral clustering in each layer performed independently (Rohe et al. 2011). This method is used only for the comparison on performance in individual layers.
- **Mean Spectral**: Spectral clustering of the mean adjacency matrix (Han et al. 2015; Paul and Chen 2017).
- **SpectralK**: The spectral kernel method for mean community detection (Paul and Chen 2017), which is similar to the module allegiance matrix technique in Braun et al. (2015).
- **MLSBM**: The variational EM algorithm in MLSBM (Han et al. 2015; Paul and Chen 2016a; Barbillon et al. 2017). This method is used only for comparison in terms of putative mean community assignments.

In our comparisons we also include an estimate for the $T$ matrix obtained by using Ind. Spectral in the individual layers and SpectralK for the mean community assignments.
7.2 Notes on implementation

Note that a number of the above mentioned methods use the k-means algorithm at some point during their execution. With the usual implementation of the k-means algorithm available in common softwares, one needs to run the k-means algorithm multiple times with different starting clusters and take the solution that minimizes the k-means objective function. The initialization for the Co-OSNTF algorithm requires some care as well, since for the update rules to converge to a good solution it is crucial to start with a good initial solution. We perform spectral clustering in each layer separately and initialize $U^{(m)}$s with the clustering solution, while the $S^{(m)}$s are initialized by a diagonal matrix containing the absolute values of the eigenvalues. To avoid having zeros in the initial solution for $U^{(m)}$s, we assign $U_{iq}^{(m)} = 1 - ck$ if $i$ is in $q$th cluster in the $m$th network, and $U_{iq}^{(m)} = c$ for all other $q' \neq q$, where $c$ is a small constant chosen based on $k$. The columns are then normalized to make it a feasible initial solution. The parameter $U^*$ is initialized in a similar fashion with the clustering solution from the SpectralK algorithm.

![Figure 2](image-url)  

**Figure 2**: Performance of various methods across three metrics: (a) Average clustering performance across all layers (in NMI), (b) performance in detecting the mean community structure (in NMI) and (c) accuracy in estimating the transition probability matrix (in Frobenius norm) with increasing variation factor from 0.05 to 0.40. The number of nodes is 500, the number of communities is 3, the number of layers is 5 and the average degree per layer is 40.

7.3 Increasing variation across layers

Our first simulation setup fixes $n$ at 500, $M$ at 5, $k$ at 3, and the average degree in each layer at 40 (which is about 8% degree density), while it varies the variation factor across the layers from 0.05 to 0.40, in steps of 0.05. Figure 2 shows the results of this simulation across the three aforementioned metrics of comparison.

We note that in terms of community detection in individual layers, the performance of all methods, except Ind. Spectral, steadily falls as the variation factor increases (see Figure 2(a)).
This is because with increasing variation factor, the layers are increasingly dissimilar and hence information sharing across layers does not improve performance as much as it does for low variation factor. The VarEM consistently outperforms all other methods in this metric, while the performance of Co-Spectral and Co-OSNTF trails closely. The Ind. Spectral does not share information across layers and hence is agnostic to variation factor. Consequently, although it gives inferior performance initially, its performance almost catches up with Co-Spectral and Co-OSNTF with increasing variation factor. Assuming the same community structure in all layers and using the community assignments obtained from Mean Spectral for all layers gives considerably worse performance, especially when the variation factor is large, since the layers become very dissimilar.

For the putative mean community assignments all methods, except MLSBM, behave similarly (see Figure 2(b)). The two-step methods (Co-Spectral and Co-OSNTF) and SpectralK slightly outperform VarEM in this case. While the performance of all methods falls with increasing variation factor, the fall is slightly steeper for Mean Spectral as compared to the two-step methods. Finally the estimate of $T$ is most accurate with VarEM and stays almost flat with increasing variation factor. The performance of Co-Spectral is poor initially at low variation factor but quickly improves as the variation factor increases. The performance of Co-OSNTF is slightly worse than that of VarEM and also stays flat with increasing variation factor. The estimate of $T$ from the combination of Ind. Spectral and SpectralK (labeled as SpectralK in Figure 2(c)) performs poorly compared to the other three methods throughout.

![Figure 3: Performance of various methods across three metrics: (a) Average clustering performance across all layers (in NMI), (b) performance in detecting the mean community structure (in NMI) and (c) accuracy in estimating the transition probability matrix (in Frobenius norm) with increasing number of layers from 5 to 25. The number of nodes is 300, the number of communities is 3, the average degree per layer is 25, and the variation factor is 0.20.](image)
7.4 Increasing number of layers

This simulation is to assess the effect of increasing number of layers on the performance of the methods. We fix \( n \) at 300, \( k \) at 3, the average degree per layer at 25 (about 8% degree density), and the variation factor at 0.20, while we vary the number of layers from 5 to 25. The three plots in Figure 3 display the results on the three metrics of comparison. The VarEM outperforms all competing methods in terms of average performance in the individual layers (Figure 3(a)) and the accuracy of estimating the \( T \) matrix (Figure 3(c)). However, it underperforms in estimating the mean community assignments as compared to Mean Spectral, SpectralK, Co-Spectral and Co-OSNTF (Figure 3(b)). Among the two-step algorithms the newly proposed Co-OSNTF performs better than Co-Spectral algorithm in estimating both the mean community assignment and \( T \) matrix, while remaining competitive with Co-Spectral in layer-wise performance. The SpectralK algorithm is competitive to Co-Spectral in detecting the mean community structure. However its layer-wise counterpart, the Ind. Spectral, performs poorly in detecting the layer-wise community structures. Consequently the estimates of \( T \), derived from a combination of Ind. Spectral and SpectralK, are also less accurate compared to other methods.

![Figure 4](image.png)

Figure 4: Performance of various methods across three metrics: (a) Average clustering performance across all layers (in NMI), (b) performance in detecting the mean community structure (in NMI) and (c) accuracy in estimating the transition probability matrix (in Frobenius norm) with increasing average degree per layer from 30 to 70. The number of nodes is 500, the number of communities is 3, the number of layers is 5 and the variation factor is 0.20.

7.5 Increasing degrees

In the last simulation experiment we fix \( n \) at 500, \( k \) at 3, \( M \) at 5, and the variation factor at 0.20, while we vary the average degree per layer from 30 to 70 (about 6% to 14% degree density). The results of this simulation for the three metrics are displayed in Figure 4. Similar to the previous cases, the VarEM performs the best in individual layers but is outperformed by a number of methods.
in the mean community structure. As with the previous cases, the Co-Spectral and Co-OSNTF outperform SpectralK, Mean Spectral and MLSBM in mean community structure and outperform Ind. Spectral and Mean Spectral in layer-wise performance. In terms of estimating the $T$ matrix, the performance of Co-OSNTF, Co-Spectral and VarEM are similar and superior to that of Ind. Spectral - SpectralK combination.

7.6 Performance of hypothesis testing procedures on synthetic networks

We next numerically compare the Sine tests and MUV tests using the estimates from the various methods developed in this article in a hypothesis testing problem on synthetic networks generated from the RESBM. This comparison will help us choose both the test statistic we should use as well as the method we should use to compute the test statistics in real networks. We generate two samples of sizes 20 and 25 from a RESBM with 100 nodes, 3 communities and a $T$ matrix whose diagonal elements are 0.8 and the off diagonal elements are 0.2. Hence there is a 20% chance that a node will not retain its $\bar{Z}$ community in $Z^{(m)}$. The putative mean community assignment matrices for the two populations $\bar{Z}_A$ and $\bar{Z}_B$ are changed from being identical to being different in approximately 30% of nodes at intervals of 5%. A good test should not detect any difference either at network level or node level when the samples come from the same population RESBM, but detect differences as the populations from which the samples are drawn differ. The distribution of all the test statistics under the null hypothesis was computed by resampling from the combined sample 10000 times. The computation time was about 300 CPU core-hours (using HP X5650 2.66 GHz cores) for each of the seven cases. The results from the network level tests are presented in Table 1 and those from the node level tests are presented in Table 2.

Table 1: Network level tests: p-values of various test statistics on two synthetic network samples of sizes 20 and 25 drawn from a 100-node, 3-community RESBM. The columns represent results for different fraction of nodes that were changed to obtain the second $\bar{Z}$ from the first. 10000 resamples was used to compute the p-values, a * indicates significant at 1% level

| Test                      | 0     | 0.05 | 0.10  | 0.15  | 0.20  | 0.25  | 0.30  |
|--------------------------|-------|------|-------|-------|-------|-------|-------|
| Sine test (VarEM)        | 0.6939| 0.0813| 0.0001*| 0.0015*| 0.0000*| 0.0001*| 0.0001*| 0.0001*|
| Sine test (Co-Spectral)  | 1.0000| 0.2282| 0.0153| 0.0000*| 0.0001*| 0.0005*| 0.0004*| 0.0004*|
| Sine test (Co-OSNTF)     | 0.2552| 0.0865| 0.0100*| 0.0003*| 0.0014*| 0.0208| 0.0107| 0.0107|
| Sine test (SpectralK)    | 0.3953| 0.0385| 0.0294| 0.0004*| 0.0002*| 0.0005*| 0.0005*| 0.0005*|
| Sine test (Mean Spectral)| 0.2743| 0.4380| 0.0353| 0.0014*| 0.0004*| 0.0054*| 0.0004*| 0.0004*|
| MUV test (VarEM)         | 0.6797| 0.0013*| 0.0000*| 0.0000*| 0.0000*| 0.0000*| 0.0000*| 0.0000*|
| MUV test (Co-Spectral)   | 0.3437| 0.0087*| 0.0021*| 0.0000*| 0.0000*| 0.0000*| 0.0000*| 0.0000*|
| MUV test (Co-OSNTF)      | 0.1316| 0.0051*| 0.0012*| 0.0000*| 0.0000*| 0.0000*| 0.0000*| 0.0000*|
| MUV test (SpectralK)     | 0.2341| 0.0037*| 0.0061*| 0.0000*| 0.0000*| 0.0000*| 0.0000*| 0.0000*|
Table 2: Performance of node level tests (total errors and false positives) on testing between two samples of synthetic networks of sizes 20 and 25 drawn from a 100-node, 3-community RESBM at: (a) 0.05 FWER, (b) 0.05 FDR, and (c) 0.10 FDR thresholds. “Number of nodes changed” is the actual number of nodes whose putative mean community assignment varied between the two groups. The best performance in terms of least total errors in each column is indicated in bold.

| Number of nodes changed | False positives | Total errors |
|-------------------------|-----------------|--------------|
|                         | 0 6 7 12 23 24 32 | 0 6 7 12 23 24 32 |
| Sine test (VarEM)       | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (Co-Spectral) | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (Co-OSNTF)    | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (SpectralK)   | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| MUV test (VarEM)        | 0 0 0 1 1 4 10    | 0 4 10 7 5 2 2  |
| MUV test (Co-Spectral)  | 0 0 0 1 0 1 1     | 0 6 7 12 2 1 1  |
| MUV test (Co-OSNTF)     | 0 0 0 1 0 0 0     | 0 5 7 9 0 0 0 0 |
| MUV test (SpectralK)    | 0 0 0 1 0 2 1     | 0 5 6 12 2 2 2 2 |

(a) At 0.05 Family-Wise Error Rate threshold.

| Number of nodes changed | False positives | Total errors |
|-------------------------|-----------------|--------------|
|                         | 0 6 7 12 23 24 32 | 0 6 7 12 23 24 32 |
| Sine test (VarEM)       | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (Co-Spectral) | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (Co-OSNTF)    | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (SpectralK)   | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| MUV test (VarEM)        | 0 2 1 1 5 2       | 0 4 5 4 3 5 2 2 |
| MUV test (Co-Spectral)  | 0 0 0 1 0 1 1     | 0 6 7 10 0 1 1  |
| MUV test (Co-OSNTF)     | 0 0 0 1 0 0 0     | 0 5 7 7 0 0 0 0 |
| MUV test (SpectralK)    | 0 0 0 1 0 2 1     | 0 5 6 11 1 2 1  |

(b) At 0.05 False Discover Rate threshold.

| Number of nodes changed | False positives | Total errors |
|-------------------------|-----------------|--------------|
|                         | 0 6 7 12 23 24 32 | 0 6 7 12 23 24 32 |
| Sine test (VarEM)       | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (Co-Spectral) | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (Co-OSNTF)    | 0 0 0 0 0 0 0     | 0 6 7 12 23 24 32 |
| Sine test (SpectralK)   | 0 2 1 1 5 2       | 0 4 4 2 5 2 2  |
| MUV test (VarEM)        | 0 0 2 0 1 1 1     | 0 6 7 6 0 1 1  |
| MUV test (Co-Spectral)  | 0 0 0 1 0 0 0     | 0 5 2 5 0 0 0 0 |
| MUV test (Co-OSNTF)     | 0 0 0 1 0 2 1     | 0 5 6 8 0 2 1  |

(b) At 0.10 False Discover Rate threshold.
First we note from Table 1 that MUV tests generally perform better than the Sine tests in detecting small differences between two populations in these synthetic networks. All the MUV tests can detect statistically significant difference at the 1% level when the $\bar{Z}$ matrices differ by 5% nodes, while all the Sine tests fail to do that. When the $\bar{Z}$ matrices differ by 10% nodes, all MUV tests are significant, while only two Sine tests (VarEM and Co-OSNTF) are significant at 1% level. We also note that all tests correctly give a large p-value when there is no difference between the $\bar{Z}$ matrices. All tests, except Sine test (Co-OSNTF), also give a statistically significant p-value when the matrices differ by more than 20%, which is the amount by which each network community structure is expected to differ from its group mean.

Table 2 clearly shows the lack of power of Sine tests to detect node level differences. None of the Sine tests succeeded in detecting any node level differences that survived the multiple comparison corrected p-value threshold of 0.05 FWER, 0.05 FDR or 0.10 FDR. We note that all MUV procedures correctly fail to detect any node-level differences when no nodes were changed. However, the procedures start and continue to make errors as the number of nodes changed increases from 0 to 12, but then quickly reduce to almost no error as the number of nodes changed increases from 23 to 32. Among the MUV tests, both the VarEM based test and the Co-OSNTF based test perform particularly well. Moreover, the Co-OSNTF test detects all the node-level differences correctly when the number of nodes is 23 or more. We note that switching from 0.05 FDR to 0.10 FDR does not increase the number of false positives much but improves performance, especially when the number of nodes truly changed is low. Based on the observations from the performance of the competing methods and test statistics on the synthetic networks, we recommend the MUV test with VarEM and Co-OSNTF as the two best performing tests.

8 Application to a resting state fMRI neuroimaging study on schizophrenia

In this paper we have developed methods for multi-subject network analysis and comparison between groups of networks. In fMRI studies often one of these groups is a group of patients diagnosed with a certain disease and the other group is that of healthy controls. Several authors have uncovered intrinsic module structures in the functional organization of brain regions through network based analysis of spontaneous neuronal activity in resting state fMRI (He et al. 2009; Meunier et al. 2010; Power et al. 2011; Yu et al. 2012b; Moussa et al. 2012). The identified modules are consistent with several functionally connected subsystems generating spontaneous activities, e.g., motor functions, auditory, visual, attention and default mode. In what follows, we apply our methods to a publicly available dataset from resting state fMRI experiment performed on subjects diagnosed with schizophrenia along with healthy controls.
8.1 COBRE dataset

The dataset we analyze is the COBRE dataset publicly available to download from International Neuroimaging Data-sharing Initiative (INDI, 1000 Functional Connectomes project, http://fcon_1000.projects.nitrc.org/indi/retro/cobre.html), that consists of anatomical MR and resting state functional MR scans from 72 patients diagnosed with schizophrenia and 75 healthy controls with ages ranging from 18 to 65 years. A detailed description of experimental conditions and equipment is available in the aforementioned webpage. Network analysis is a key tool employed in analysis of functional connectivity in fMRI based experiments on schizophrenia (Lynall et al. 2010; Liu et al. 2008; Bassett et al. 2012; Yu et al. 2012a; van den Heuvel et al. 2010, 2013; Alexander-Bloch et al. 2010, 2012). Analysis of modular organization (community structure) of brain networks is of particular importance in understanding schizophrenia, since it has been hypothesized that schizophrenia is associated with neurodevelopment and evolution of brain, which in turn is influenced by modular organization (see Yu et al. (2012a) and references therein).

We used an automated preprocessing pipeline (Statistical Parametric Mapping’s (SPM) default preprocessing pipeline for volume based analyses (Penny et al. 2011)) implemented in Matlab toolbox CONN (Whitfield-Gabrieli and Nieto-Castanon 2012) with parameters similar to earlier studies in Lynall et al. (2010) and Bassett et al. (2012). In particular the steps included, deleting first three volumes, correcting for head motion by functional realignment and unwarping, functional slice timing correction, functional and structural image co-registration, structural segmentation and normalization, functional normalization to the standard Montreal Neurological Institute (MNI) space, functional outlier detection, and spatial smoothing with a Gaussian kernel with 6mm full width at half maximum (FWHM). Temporal filtering was performed with a high pass filter with cutoff at 0.008 Hz. The regions of interest (ROIs) were determined from a whole brain image parcellation into anatomically defined regions described in the Automated Anatomical Labeling (AAL) atlas (Tzourio-Mazoyer et al. 2002). We exclude the cerebellum and vermis, and concentrate on the remaining 90 cortical and subcortical ROIs, similar to previous studies (He et al. 2009; Lynall et al. 2010; Bassett et al. 2012). Mean ROI time series was obtained for each of the ROIs by averaging the time series in the voxels within the ROI.

To compute functional connectivity among the ROIs, we first decompose the mean time series in each ROI to a 4 scale maximal overlap discrete wavelet transform (Percival and Walden 2006), and then take the second scale which roughly corresponds to the frequency range 0.060-0.125 Hz. A ROI to ROI correlation matrix is subsequently constructed from the pairwise correlations among this scale 2 wavelet transformed time series. The second scale of the wavelet transformation is chosen to strike a balance between minimizing the impact of physiological noise that might confound the higher frequencies and not having enough samples to compute correlation matrix in lower frequency ranges (Lynall et al. 2010; Bassett et al. 2012; Alexander-Bloch et al. 2012). Binary correlation matrices were created for each subject in either case through thresholding at six levels, 0.1, 0.2, 0.25, 0.3, 0.4, and 0.5, resulting in graphs of different connection densities. For each of these thresholds, Table 3 presents the average of modularity values and number of
Table 3: Average modularity and average number of communities detected by modularity maximization (Spin-glass and Louvain algorithms) in the two groups of subjects for different thresholds. The columns of p-value indicate p-values obtained from Welch two sample t-test for difference in means. A * indicates statistically significant at 5% level.

| Threshold | Modularity       | Communities       |
|-----------|------------------|-------------------|
|           | Controls | Patients | p-value | Controls | Patients | p-value |
| 0.1       | 0.0690  | 0.0667    | 0.0015*  | 3.50     | 3.47     | 0.1972  |
| 0.2       | 0.1616  | 0.1556    | 0.0026*  | 4.20     | 4.23     | 0.5937  |
| 0.25      | 0.2213  | 0.2140    | 0.0176*  | 4.74     | 4.66     | 0.4466  |
| 0.3       | 0.2814  | 0.2717    | 0.0483*  | 5.35     | 5.14     | 0.1681  |
| 0.4       | 0.4681  | 0.4425    | 0.0263*  | 12.40    | 10.91    | 0.0291* |
| 0.5       | 0.6576  | 0.6204    | 0.0282*  | 38.83    | 35.38    | 0.0903  |

From Table 3, it is clear that the control group has significantly (at the 5% level) higher modularity value as compared to the patient group, irrespective of our choice of threshold. The average number of communities detected in each of the groups is not significantly different except at the threshold of 0.4. Both observations in terms of modularity and number of communities are consistent with previously reported results in the context of childhood-onset schizophrenia (Alexander-Bloch et al. 2012). We primarily focus our analysis on the threshold of 0.2. Although Alexander-Bloch et al. (2012) has shown that group differences in both modularity value and modular organization become more pronounced at higher values of the threshold, we use a smaller value since at higher values the network divides into too many communities which are difficult to interpret and visualize.

Note that the methods developed in this paper require the number of communities to be supplied as input. We obtain the number of communities in each case from the average number of communities detected in Table 3. For the threshold of 0.2, we observe that the average number of communities in both groups is about 4. Hence we fit RESBMs with 4 communities to the networks from the control group and patient group subjects using both the variational EM and the two-step methods. The group putative community structure obtained for each group from each of the three methods is illustrated on a brain surface template in Figure 5 using the BrainNet Viewer software (http://www.nitrc.org/projects/bnv/) (Xia et al. 2013).

We first note that community structure for the control group from Co-Spectral and Co-OSNTF almost agree with each other, and the four modules obtained roughly match the consistent modules
Figure 5: Group putative community structure of resting state network based on AAL ROIs in (a, c, e) healthy controls, and (b, d, f) patients with schizophrenia. Nodes are colored according to their group putative community obtained from the following methods: Co-Spectral for the first row (a, b), Co-OSNTF for the second row (c, d) and VarEM for the third row (e, f).
Table 4: Nodes of Interest: Nodes which are significant at 0.1 FDR correction.

| Node        | Uncorrected | FDR corrected | FWER   |
|-------------|-------------|---------------|--------|
| Pallidum.L  | 0.0004      | 0.0360        | 0.0360 |
| Amygdala.R | 0.0020      | 0.0774        | 0.1780 |
| Pallidum.R  | 0.0035      | 0.0774        | 0.3080 |
| Putamen.L   | 0.0039      | 0.0774        | 0.3393 |
| Thalamus.L  | 0.0043      | 0.0774        | 0.3698 |

Co-Spectral: Network level p-value: 0.0105

| Node        | Uncorrected | FDR corrected | FWER   |
|-------------|-------------|---------------|--------|
| Caudate.R   | 0.0013      | 0.0742        | 0.1170 |
| Temporal.Pole.Sup.L | 0.0017 | 0.0742 | 0.1513 |
| Hippocampus.R | 0.0031     | 0.0742      | 0.2728 |
| Pallidum.L  | 0.0033      | 0.0742        | 0.2871 |

Co-OSNTF: Network level p-value: 0.0042

detected in resting state in Moussa et al. (2012). Specifically, in the control group our modules blue (module 1), red (module 2), yellow (module 3) and green (module 4) are roughly equivalent to Moussa et al. (2012)’s default mode module, basal ganglia module, visual module and sensory/motor module respectively. For both Co-Spectral and Co-OSNTF the red group in controls gets disrupted in patients, and nodes belonging to the module get distributed to the yellow, green and blue modules in patients indicating disruption of the functional cohesion of these nodes. The blue module in controls gets divided into two modules in patients, while the yellow and green groups remain relatively intact.

To statistically test the disruption in community structure, we compute the p-values for the MUV test with 10000 permutation resamples. In both Co-Spectral and Co-OSNTF, the MUV test rejects the null hypothesis of no difference in the community structure between the controls and patients with p-values of 0.0102 and 0.0042 respectively (Table 4). The node level MUV test with Co-Spectral and Co-OSNTF found a number of ROIs to be statistically significantly altered at 0.1 FDR corrected p-values, which we call nodes of interest in Table 4. All the ROIs in cases of Co-Spectral and Co-OSNTF, except Temporal.Pole.Sup.L from Co-OSNTF, belong to the red module, as can be seen from Figure 5. Figure 6 displays a visualization of the locations of the ROI groups in a 8-view layout of the brain.

It is also crucially important to understand the variability of the different modules in the group putative community structure across the different subjects within the group. It can be assessed through the estimated transition matrices for control and patient groups in our model, as well as a module consistency matrix that measures the fraction of subjects for which two nodes or ROIs are
in the same community. The module consistency matrix is similar to the module allegiance matrix in Braun et al. (2015) and Scaled Inclusivity measure in Steen et al. (2011) and Moussa et al. (2012). Figure 7 displays the metrics for the results from VarEM method. In Figure 7(a) and (b) the ROIs are sorted according to their community label in increasing order (i.e., module 1 in bottom left corner and module 4 in top right corner). In Figure 7(c) and (d), the $4 \times 4$ module transition matrix is plotted with the putative modules arranged from 1 to 4 along the row and column. From Figure 7 it is clear that the module structure is more consistent in controls than in patients. The patients show greater variability in the functional connectivity between any two regions, leading them to be classified into different modules more often than controls. In particular, it can be seen from both the metrics in Figure 7 that the yellow module (module 3) is highly consistent in controls as has been observed in many previous resting state studies (see Moussa et al. (2012) and references therein), however, in patients it is much less consistent.

To better understand the community structure for the entire group, we visualize the structure obtained from Co-OSNTF algorithm in Figure 8 with the help of two measures: module consistency and mean connectivity. While the community structure is the same (the group putative community structure for controls and patients), the edges in Figure 8(a) represent the fraction of subjects for which two nodes are in the same community. The edges in Figure 8(b) represent the average connectivity between two nodes across all subjects. In both cases the edges are thresholded at a certain level (i.e., the edge appears if the quantity it represents exceeds a certain value) and are weighted, with thicker edges representing larger values. Similar to the observation in Moussa et al. (2012), we also have the visual module (yellow colored), containing both the primary and secondary visual cortices, as the most consistent module. We also observe that this module is relatively consistent in patients as well, confirming the observation from previous studies (Yu et al. 2012a). The red colored module in the control group, which roughly corresponds to the default
Figure 7: Stability of the group putative community assignments for VarEM: (a) and (b) are the matrices with elements as fraction of subjects for which two nodes (ROIs) are in the same community sorted according to the putative community structure for healthy controls and patients respectively; (c) and (d) are the estimated transition matrices among modules for healthy controls and patients respectively.

mode network in Moussa et al. (2012), is split in two parts with some nodes being part of another module. From Figure 8(a) it is clear that the blue group in controls is split in two groups in patients (blue and red) which are almost disjoint in terms of module consistency thresholded at 0.40. The nodes in the red group in controls have lost the tendency to be grouped together in the patients and instead they are more consistently grouped with different modules. The nodes that belonged to the yellow and green groups in controls appear to be unchanged in their co-module relation with the rest of the network.

9 Discussions

In this paper we have proposed a random effects model to jointly model the community structure in a sample from a population of networks. The proposed estimation and hypothesis testing methodology outperforms baseline and competing methods in simulated network samples. Our methods uncover meaningful differences between functional brain networks of healthy controls and schizophrenia patients. We hope the principled approach developed here will be useful in modeling and contrasting network valued samples in terms of their low dimensional latent structures.

We note that both the variational EM and Co-OSNTF algorithms have limitations. For Co-OSNTF, Theorem 1 only guarantees convergence to a stationary point for the multiplicative update rules with the unknown Lagrange multipliers $\Lambda_{U^{(m)}}$ and $\Lambda_{U^*}$, provided a stationary point exists in the positive orthant. In order to obtain a practical update rule, we made an assumption to approximate $\Lambda_{U^{(m)}}$ and $\Lambda_{U^*}$ and derived the final update rules. We also note that the convergence guarantee is only to a stationary point and not to a global optimum. Hence the stopping point of the update rule may not be the global maximum of the objective function. There is a similar issue with the variational EM algorithm as well. By design, the variational algorithm optimizes an
approximation of the likelihood objective function and the quality of the solution depends on the quality of that approximation.

Appendix

Proof of Lemma 1

The first statement can be seen very easily using the induction arguments in Lin (2007). By assumption on the initial solutions,

\[(U^{(m)}(t)) > 0, (U^*)(t) > 0, (S^{(m)}(t)) > 0,\]

for \(t = 0\). Let (9.1) be true for some \(t > 0\). Then, clearly, if there are no zero rows (or columns) in any of the \(L^{(m)}\)s, then the multiplier in (5.4) is strictly positive for all \(m, i, k\), and consequently, all elements of \((U^{(m)}(t+1))\) are strictly positive. The multiplier in (5.6) is strictly positive since by assumption (9.1) is true for \(t\), and hence \((U^*)(t+1) > 0\). Finally, no zero column or row in \(L^{(m)}\)s ensures the multiplier in (5.7) is also strictly positive and \((S^{(m)}(t+1)) > 0\).

For the second part, we have \(\{\tilde{U}^{(m)} > 0, \tilde{U}^* > 0, \tilde{S}^{(m)} > 0\}\) as a stationary point of the objective function \(\xi\). Since all the parameters at this point are in the positive orthant, the complimentary slackness conditions of the KKT conditions imply that \(\nabla \xi |_{U^{(m)}} = 0, \nabla \xi |_{U^*} = 0\) and \(\nabla \xi |_{S^{(m)}} = 0\).
Then the multipliers in update rules (5.4), (5.6) and (5.7) are all 1 and the update rules will stop updating.

**Proof of Theorem 1**

We will use the auxiliary function approach introduced in Lee and Seung (2001). For any parameter matrix $X$, a function $Y(X, X')$ is an auxiliary function of the objective function $\xi(X)$ if it satisfies the following conditions:

$$Y(X, X') \geq \xi(X), \quad Y(X, X) = \xi(X),$$

(9.2)

for any $X, X'$. Consider the update:

$$X^{(t+1)} = \arg \min_X Y(X, X^{(t)}).$$

(9.3)

It can be readily seen that

$$\xi(X^{(t)}) = Y(X^{(t)}, X^{(t)}) \geq Y(X^{(t+1)}, X^{(t)}) \geq \xi(X^{(t+1)}),$$

(9.4)

and consequently, $\xi(X)$ is monotonically non-increasing under the update (9.3). We define the following auxiliary function for $U^{(m)}$:

$$Y(U^{(m)}, U^{(m)'}) = -2 \sum_{ik} (L^{(m)}U^{(m)'}) S^{(m)} + \lambda_m U^* U^{T} U^{(m)'})_{ik} U^{(m)'}_{ik} \left(1 + \log \frac{U^{(m)'}_{ik}}{U^{(m)'}_{ik}} \right)$$

$$+ \sum_{ik} \left( U^{(m)'}(S^{(m)} U^{(m)'})^T U^{(m)'}) S^{(m)} + \Lambda U^{(m)})_{ik} U^{(m)2}_{ik} \right) \frac{1}{U^{(m)'}_{ik}} + \text{tr}(A U^* U^* U^*).$$

(9.5)

Next we show that the function $Y$ satisfies the conditions in (9.2). The first term in $Y$ is larger than the first and third terms of $\xi$ in (5.2) together since $z \geq 1 + \log z$, for all $z > 0$. The second term of $Y$ is also greater than the second term of $\xi$. To prove this we use the following proposition from Ding et al. (2005).

**Proposition 1.** For any matrices $A_{n \times n} \geq 0$, $B_{k \times k} \geq 0$, $S_{n \times k} \geq 0$, $S'_{n \times k} \geq 0$, the following inequality holds

$$\sum_i \sum_p \frac{(ASB)_{ip} S'_{ip}}{S'_{ip}} \geq \text{tr}(S^T A S B).$$

Letting $A = I$, $B = S^{(m)} U^{(m)} U^{(m)'}, S^{(m)} + \Lambda U^{(m)}$, $S = U^{(m)}$ and $S' = U^{(m)}'$, we have

$$\sum_{ik} \frac{(U^{(m)'}(S^{(m)} U^{(m)'})^T U^{(m)'}) S^{(m)} + \Lambda U^{(m)})_{ik} U^{(m)2}_{ik}}{U^{(m)'}_{ik}} \geq \text{tr}(U^{(m)} S^{(m)} U^{(m)} U^{(m)'}) S^{(m)} U^{(m)'} U^{(m)'},$$

and hence the second term of $Y$ is greater than the second term of $\xi$ in (5.2). The last term in both $Y$ and $\xi$ are identical. Hence $Y(U^{(m)}, U^{(m)'} \geq \xi(U^{(m)})$. Moreover, when $U^{(m)} = U^{(m)}'$, clearly, $Y(U^{(m)}, U^{(m)}) = \xi(U^{(m)})$. 

31
Then we just need to compute the update defined by (9.3) for \( U^{(m)} \). Differentiating \( Y \) with respect to \( U_{ik}^{(m)} \) (note \( U_{ik}^{(m)'} \) is fixed here) we have,

\[
0 = \frac{\partial Y(U^{(m)}, U^{(m)'})}{\partial U_{ik}^{(m)}} = -2(L^{(m)}U^{(m)'})ik U_{ik}^{(m)'} + \lambda_m U^{(m)'}U^{(m)'T}U_{ik}^{(m)'} + 2(U^{(m)'})ik (S^{(m)}U^{(m)'T}U^{(m)'})ik U_{ik}^{(m)}.
\]

Rearranging the terms we get the update rule in (5.4). Then by (9.4), \( \xi(U^{(m)}) \), keeping the other parameter matrices \( U^* \) and \( S^{(m)} \) fixed, is monotonically non-increasing under (5.4).

Similarly, for \( U^* \), we define the following auxiliary function

\[
Y(U^*, U^{'}) = -2 \sum_{ik} (\lambda_m U^{(m)}U^{(m)'T}U^{'})ik U_{ik}^{(m)'} \left( 1 + \log \frac{U_{ik}^{*}}{U_{ik}^{*'}} \right) + \sum_{ik} \frac{(U^*'U^*)ik U_{ik}^{*2}}{U_{ik}^{*'}} + f(U^{(m)}, S^{(m)}),
\]

where \( f(U^{(m)}, S^{(m)}) \) represents all terms of \( \xi \) that do not contain \( U^* \). We can verify that \( Y(U^*, U^{'}) \) in (9.6) is an auxiliary function of \( \xi \) in (5.2). The first term of \( Y \) is greater than the third term of \( \xi \) since \( z \geq 1 + \log z \) for all \( z > 0 \). From Proposition 1, letting \( A = I, B = \Lambda U^*, S = U^* \) and \( S' = U^{'} \), we have \( \sum_{ik} \frac{(U^*'U^*)ik U^{*2}}{U_{ik}^{*'}} \geq \text{tr}(\Lambda_U^*U^{*T}U^*) \). This implies the second term of \( Y \) is greater than the last term of \( \xi \). All the remaining terms in \( \xi \) are identical to \( f(U^{(m)}, S^{(m)}) \). Hence \( Y(U^*, U^{'}) \) satisfies the conditions for being an auxiliary function of \( \xi(U^*) \) in (9.2). Arguments in (9.4) imply that \( \xi(U^*) \), keeping \( U^{(m)} \) and \( S^{(m)} \) fixed, is monotonically non-increasing under the update \( (U^*)^{(t+1)} = \arg \min_{U^*} Y(U^*, (U^*)^{(t)}) \). To compute the update, we note that

\[
0 = \frac{\partial Y(U^*, U^{'})}{\partial U_{ik}^{*'}} = -2 \lambda_m (U^{(m)}U^{(m)'T}U^{'})ik U_{ik}^{*'} + \frac{2 (U^*'U^*)ik U_{ik}^{*}}{U_{ik}^{*'}}.
\]

Rearranging the terms we have the update rule in (5.6).

Finally, for \( S^{(m)} \), we have the following auxiliary function.

\[
Y(S^{(m)}, S^{(m)'} = -2 \sum_{ik} (U^{(m)'T}L^{(m)}U^{(m)})ik S_{ik}^{(m)'} \left( 1 + \log \frac{S_{ik}^{(m)}}{S_{ik}^{(m)'} \right) + \sum_{ik} \frac{(U^{(m)'T}U^{(m)})ik S_{ik}^{(m)2}}{S_{ik}^{(m)'}},
\]

where \( f(U^{(m)}, U^*) \) represents all terms of \( \xi \) that do not contain \( S^{(m)} \). The first term of \( Y(S^{(m)}, S^{(m)'} \) is greater than the first term of \( \xi \) in (5.2) since \( z \geq 1 + \log z \) for all \( z > 0 \). Next in Proposition 1, we let \( n = k \) and consequently in the notations of the proposition, \( A, B, S, S' \) are all \( k \times k \) matrices. Then letting \( A = B = U^{(m)'T}U^{(m)}, S = S^{(m)} \) and \( S' = S^{(m)'} \), we have

\[
\sum_{ik} \frac{(U^{(m)'T}U^{(m)})ik S_{ik}^{(m)2}}{S_{ik}^{(m)'}} \geq \text{tr}(U^{(m)'T}U^{(m)})U^{(m)'T}U^{(m)}U^{(m)'T}U^{(m)}).
\]

This implies the second term of \( Y \) in (9.7) is greater than the second term of \( \xi \) in (5.2). All the remaining terms in \( \xi \) are identical to \( f(U^{(m)}, U^*) \). Then under the update \( (S^{(m)})^{(t+1)} = \arg \min_{S^{(m)}} Y(S^{(m)}, (S^{(m)})^{(t)}) \), which leads to the rule in (5.7), \( \xi(S^{(m)}) \) is monotonically non-increasing keeping \( U^{(m)} \) and \( U^* \) fixed.
Hence alternatively updating $U^{(m)}, S^{(m)}, U^*$ according to the MUR update rules in (5.4), (5.6) and (5.7) will lead to the following chain of inequalities

$$
\xi((U^{(m)})^{(t)}, (S^{(m)})^{(t)}, (U^*)^{(t)}) \geq \xi((U^{(m)})^{(t+1)}, (S^{(m)})^{(t)}, (U^*)^{(t)}) \geq \xi((U^{(m)})^{(t+1)}, (S^{(m)})^{(t+1)}, (U^*)^{(t)}) \geq \ldots,
$$

and the function $\xi$ will be non-increasing.

References

Alexander-Bloch, A., Lambiotte, R., Roberts, B., Giedd, J., Gogtay, N. and Bullmore, E. (2012) The discovery of population differences in network community structure: new methods and applications to brain functional networks in schizophrenia. *Neuroimage*, 59, 3889–3900.

Alexander-Bloch, A. F., Gogtay, N., Meunier, D., Birn, R., Clasen, L., Lalonde, F., Lenroot, R., Giedd, J. and Bullmore, E. T. (2010) Disrupted modularity and local connectivity of brain functional networks in childhood-onset schizophrenia. *Frontiers in Systems Neuroscience*, 4, 147.

Barbillon, P., Donnet, S., Lazega, E. and Bar-Hen, A. (2017) Stochastic block models for multiplex networks: an application to a multilevel network of researchers. *Journal of the Royal Statistical Society: Series A*, 180, 295–314.

Bassett, D. S., Nelson, B. G., Mueller, B. A., Camchong, J. and Lim, K. O. (2012) Altered resting state complexity in schizophrenia. *Neuroimage*, 59, 2196–2207.

Bassett, D. S., Wymbs, N. F., Porter, M. A., Mucha, P. J., Carlson, J. M. and Grafton, S. T. (2011) Dynamic reconfiguration of human brain networks during learning. *Proceedings of the National Academy of Sciences*, 108, 7641–7646.

Bazzi, M., Porter, M. A., Williams, S., McDonald, M., Fenn, D. J. and Howison, S. D. (2014) Community detection in temporal multilayer networks, and its application to correlation networks. *arXiv preprint arXiv:1501.00040*.

Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the royal statistical society. Series B*, 289–300.

Bickel, P. J., Choi, D., Chang, X. and Zhang, H. (2013) Asymptotic normality of maximum likelihood and its variational approximation for stochastic blockmodels. *Ann. Statist.*, 41, 1922–1943.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R. and Lefebvre, E. (2008) Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008, P10008.

Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C. I., Gómez-Gardeñes, J., Romance, M., Sendina-Nadal, I., Wang, Z. and Zanin, M. (2014) The structure and dynamics of multilayer networks. *Physics Reports*, 544, 1–122.

Braun, U., Schäfer, A., Walter, H., Erk, S., Romanczuk-Seiferth, N., Haddad, L., Schweiger, J. I., Grimm, O., Heinz, A. and Tost, H. (2015) Dynamic reconfiguration of frontal brain networks during executive cognition in humans. *Proceedings of the National Academy of Sciences*, 112, 11678–11683.

Bullmore, E. and Sporns, O. (2009) Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10, 186–198.

Celisse, A., Daudin, J. J. and Pierre, L. (2012) Consistency of maximum-likelihood and variational estimators in the stochastic block model. *Electronic Journal of Statistics*, 6, 1847–1899.

Daudin, J. J., Picard, F. and Robin, S. (2008) A mixture model for random graphs. *Stat Comput.*, 18, 173–183.

De Bacco, C., Power, E. A., Larremore, D. B. and Moore, C. (2017) Community detection, link prediction and layer interdependence in multilayer networks. *arXiv preprint arXiv:1701.01369*.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977) Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B*, 39, 1–38.

Ding, C., Li, T., Peng, W. and Park, H. (2006) Orthogonal nonnegative matrix t-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 126–135. ACM.
Ding, C. H., He, X. and Simon, H. D. (2005) On the equivalence of nonnegative matrix factorization and spectral clustering. In *SDM*, vol. 5, 606–610. SIAM.

Dong, X., Frossard, P., Vandergeynst, P. and Nefedov, N. (2014) Clustering on multi-layer graphs via subspace analysis on Grassmann manifolds. *IEEE Transactions on Signal Processing*, 62, 905–918.

Edelman, A., Arias, T. A. and Smith, S. T. (1998) The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20, 303–353.

Fujita, A., Takahashi, D. Y., Patriota, A. G. and Sato, J. R. (2014) A non-parametric statistical test to compare clusters with applications in functional magnetic resonance imaging data. *Statistics in Medicine*, 33, 4949–4962.

Glerean, E., Pan, R. K., Salmi, J., Kujala, R., Lahmakoski, J. M., Roine, U., Nummenmaa, L., Leppämäki, S., Nieminen-von Wendt, T. and Tani, P. (2016) Reorganization of functionally connected brain subnetworks in high-functioning autism. *Human Brain Mapping*, 37, 1066–1079.

Han, Q., Xu, K. S. and Airoldi, E. M. (2015) Consistent estimation of dynamic and multi-layer block models. In *Proceedings of the 32nd International Conference on Machine Learning*, 1511–1520.

He, Y., Wang, J., Wang, L., Chen, Z. J., Yan, C., Yang, H., Tang, H., Zhu, C., Gong, Q. and Zang, Y. (2009) Uncovering intrinsic modular organization of spontaneous brain activity in humans. *PLoS ONE*, 4, e5226.

van den Heuvel, M. P., Mandl, R. C., Stam, C. J., Kahn, R. S. and Pol, H. E. H. (2010) Aberrant frontal and temporal complex network structure in schizophrenia: a graph theoretical analysis. *The Journal of Neuroscience*, 30, 15915–15926.

van den Heuvel, M. P., Sporns, O., Collin, G., Scheewe, T., Mandl, R. C., Cahn, W., Goñi, J., Pol, H. E. H. and Kahn, R. S. (2013) Abnormal rich club organization and functional brain dynamics in schizophrenia. *JAMA Psychiatry*, 70, 783–792.

Hutchison, R. M., Womelsdorf, T., Allen, E. A., Bandettini, P. A., Calhoun, V. D., Corbetta, M., Della Penna, S., Duyn, J. H., Glover, G. H. and Gonzalez-Castillo, J. (2013) Dynamic functional connectivity: promise, issues, and interpretations. *Neuroimage*, 80, 360–378.

Jones, D. T., Venmuri, P., Murphy, M. C., Gunter, J. L., Senjem, M. L., Machulda, M. M., Przybelski, S. A., Gregg, B. E., Kantarci, K. and Knopman, D. S. (2012) Non-stationarity in the resting brains modular architecture. *PLoS ONE*, 7, e39731.

Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y. and Porter, M. A. (2014) Multilayer networks. *Journal of Complex Networks*, 2, 203–271.

Kujala, R., Glerean, E., Pan, R. K., Jääskeläinen, I. P., Sams, M. and Saramäki, J. (2016) Graph coarse-graining reveals differences in the module-level structure of functional brain networks. *European Journal of Neuroscience*, 44, 2673–2684.

Kumar, A., Rai, P. and Daume, H. (2011) Co-regularized multi-view spectral clustering. In *Advances in Neural Information Processing Systems*, 1413–1421.

Lee, D. D. and Seung, H. S. (2001) Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, 556–562.

Lin, C.-J. (2007) Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19, 2756–2779.

Liu, J., Wang, C., Gao, J. and Han, J. (2013) Multi-view clustering via joint nonnegative matrix factorization. In *Proc. of SDM*, vol. 13, 252–260. SIAM.
Liu, Y., Liang, M., Zhou, Y., He, Y., Hao, Y., Song, M., Yu, C., Liu, H., Liu, Z. and Jiang, T. (2008) Disrupted small-world networks in schizophrenia. *Brain*, 131, 945–961.

Lynall, M.-E., Bassett, D. S., Kerwin, R., McKenna, P. J., Kitzbichler, M., Muller, U. and Bullmore, E. (2010) Functional connectivity and brain networks in schizophrenia. *The Journal of Neuroscience*, 30, 9477–9487.

Mankad, S. and Michailidis, G. (2013) Structural and functional discovery in dynamic networks with non-negative matrix factorization. *Physical Review E*, 88, 042812.

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, 1119–1141.

Meunier, D., Lambiotte, R. and Bullmore, E. T. (2010) Modular and hierarchically modular organization of brain networks. *Frontiers in Neuroscience*, 4, 200.

Mirjalil, A. (2014) A convergent algorithm for orthogonal nonnegative matrix factorization. *Journal of Computational and Applied Mathematics*, 260, 149–166.

Moussa, M. N., Steen, M. R., Laurienti, P. J. and Hayasaka, S. (2012) Consistency of network modules in resting-state fMRI connectome data. *PLoS ONE*, 7, e44428.

Mucha, P. J., Richardson, T., Macon, K., Porter, M. A. and Onnela, J. P. (2010) Community structure in time-dependent, multiscale, and multiplex networks. *Science*, 328, 876–878.

Narayan, M. and Allen, G. I. (2016) Mixed effects models for resampled network statistics improves statistical power to find differences in multi-subject functional connectivity. *Frontiers in Neuroscience*, 10.

Nicosia, V. and Latora, V. (2014) Measuring and modelling correlations in multiplex networks. *arXiv preprint arXiv:1403.1546*.

Paul, S. and Chen, Y. (2016a) Consistent community detection in multi-relational data through restricted multi-layer stochastic blockmodel. *Electronic Journal of Statistics*, 10, 3807–3870.

— (2016b) Null models and modularity based community detection in multi-layer networks. *arXiv preprint arXiv:1608.00623*.

— (2016c) Orthogonal symmetric non-negative matrix factorization under the stochastic block model. *arXiv preprint arXiv:1605.05349*.

— (2017) Consistency of community detection in multi-layer networks using spectral and matrix factorization methods. *arXiv Preprint arXiv:1704.07353*.

Peixoto, T. P. (2015) Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. *Physical Review E*, 92, 042807.

Penny, W. D., Friston, K. J., Ashburner, J. T., Kiebel, S. J. and Nichols, T. E. (2011) *Statistical parametric mapping: the analysis of functional brain images*. Academic press.

Percival, D. B. and Walden, A. T. (2006) *Wavelet Methods for Time Series Analysis*, vol. 4. Cambridge University Press.

Power, J. D., Cohen, A. L., Nelson, S. M., Wig, G. S., Barnes, K. A., Church, J. A., Vogel, A. C., Laumann, T. O., Miezin, F. M. and Schlaggar, B. L. (2011) Functional network organization of the human brain. *Neuron*, 72, 665–678.

Reichardt, J. and Bornholdt, S. (2006) Statistical mechanics of community detection. *Physical Review E*, 74, 016110.

Reyes, P. and Rodríguez, A. (2016) Stochastic blockmodels for exchangeable collections of networks. *arXiv preprint arXiv:1606.05277*.

Rohde, K., Chatterjee, S. and Yu, B. (2011) Spectral clustering and the high-dimensional stochastic blockmodel. *Ann. Statist.*, 39, 1878–1915.

Rubinov, M. and Sporns, O. (2010) Complex network measures of brain connectivity: Uses and interpretations. *Neuroimage*, 52, 1059–1069.

Simpson, S. L., Bowman, F. D. and Laurienti, P. J. (2013a) Analyzing complex functional brain networks: fusing statistics and network science to understand the brain. *Statistics Surveys*, 7, 1.

Simpson, S. L., Lyday, R. G., Hayasaka, S., Marsh, A. P. and Laurienti, P. J. (2013b) A permutation testing framework
to compare groups of brain networks. *Frontiers in Computational Neuroscience, 7*, 171.

Sporns, O. (2014) Contributions and challenges for network models in cognitive neuroscience. *Nature Neuroscience, 17*, 652–660.

Stanley, N., Shai, S., Taylor, D. and Mucha, P. J. (2016) Clustering network layers with the strata multilayer stochastic block model. *IEEE Transactions on Network Science and Engineering, 3*, 95–105.

Steen, M., Hayasaka, S., Joyce, K. and Laurienti, P. (2011) Assessing the consistency of community structure in complex networks. *Physical Review E, 84*, 016111.

Stevens, A. A., Tappon, S. C., Garg, A. and Fair, D. A. (2012) Functional brain network modularity captures inter- and intra-individual variation in working memory capacity. *PLoS ONE, 7*, e30468.

Stewart, G. W. and Sun, J.-G. (1990) *Matrix Perturbation Theory*. Academic Press, Boston, MA.

Sweet, T. M., Thomas, A. C. and Junker, B. W. (2014) Hierarchical mixed membership stochastic blockmodels for multiple networks and experimental interventions. *Handbook on Mixed Membership Models and Their Applications*, 463–488.

Tang, W., Lu, Z. and Dhillon, I. S. (2009) Clustering with multiple graphs. In *Proceedings of the Ninth IEEE International Conference on Data Mining*, 1016–1021. IEEE.

Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B. and Joliot, M. (2002) Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *Neuroimage, 15*, 273–289.

Valles-Catala, T., Massucci, F. A., Guimera, R. and Sales-Pardo, M. (2016) Multilayer stochastic block models reveal the multilayer structure of complex networks. *Physical Review X, 6*, 011036.

Van Den Heuvel, M. P. and Pol, H. E. H. (2010) Exploring the brain network: a review on resting-state fMRI functional connectivity. *European Neuropsychopharmacology, 20*, 519–534.

Weber, M. J., Detre, J. A., Thompson-Schill, S. L. and Avants, B. B. (2013) Reproducibility of functional network metrics and network structure: a comparison of task-related BOLD, resting ASL with BOLD contrast, and resting cerebral blood flow. *Cognitive, Affective, & Behavioral Neuroscience, 13*, 627–640.

Whitfield-Gabrieli, S. and Nieto-Castanon, A. (2012) CONN: a functional connectivity toolbox for correlated and anticorrelated brain networks. *Brain Connectivity, 2*, 125–141.

Xia, M., Wang, J. and He, Y. (2013) Brainnet viewer: a network visualization tool for human brain connectomics. *PLoS ONE, 8*, e68910.

Yu, Q., A Allen, E., Sui, J., R Arbabshirani, M., Pearlson, G. and D Calhoun, V. (2012a) Brain connectivity networks in schizophrenia underlying resting state functional magnetic resonance imaging. *Current Topics in Medicinal Chemistry, 12*, 2415–2425.

Yu, Q., Plis, S. M., Erhardt, E. B., Allen, E. A., Sui, J., Kiehl, K. A., Pearlson, G. and Calhoun, V. D. (2012b) Modular organization of functional network connectivity in healthy controls and patients with schizophrenia during the resting state. *Frontiers in Systems Neuroscience, 5*, 103.