Applied Network Science

RESEARCH Open Access

Model selection for network data based on spectral information

Jairo Iván Peña Hidalgo[1] and Jonathan R. Stewart[1*]

*Correspondence:
jrstewart@fsu.edu

[1] Department of Statistics, Florida State University, 117 N Woodward Ave, Tallahassee 32306-4330, FL, USA

Abstract
In this work, we explore the extent to which the spectrum of the graph Laplacian can characterize the probability distribution of random graphs for the purpose of model evaluation and model selection for network data applications. Network data, often represented as a graph, consist of a set of pairwise observations between elements of a population of interests. The statistical network analysis literature has developed many different classes of network data models, with notable model classes including stochastic block models, latent node position models, and exponential families of random graph models. We develop a novel methodology which exploits the information contained in the spectrum of the graph Laplacian to predict the data-generating model from a set of candidate models. Through simulation studies, we explore the extent to which network data models can be differentiated by the spectrum of the graph Laplacian. We demonstrate the potential of our method through two applications to well-studied network data sets and validate our findings against existing analyses in the statistical network analysis literature.

Keywords: Statistical network analysis, Network data, Model selection, Social network analysis

Introduction

Network data have witnessed a surge of interest across a variety of fields and disciplines in recent decades, including the study of social networks (Lusher et al. 2013), network epidemiology (involving the spread of disease through networks of contacts) (Morris 2004), covert networks of criminal activity and terrorism (Coutinho et al. 2020), brain networks (Obando and de Vico Fallani 2017), financial markets (Finger and Lux 2017), and more. Network data, as a data structure, is typically represented as a graph, consisting of a set of nodes representing the elements of a population of interest (e.g., researchers in a collaboration network) and a set of pairwise observations or measurements between nodes represented as edges between nodes (e.g., co-authorship on a paper). Many classes of models have been proposed and developed to study and model network data. A non-exhaustive review includes exponential families of random graph models (ERGMs) (e.g., Lusher et al. 2013; Schweinberger et al. 2020), stochastic block models (SBMs) (e.g., Holland et al. 1983; Anderson et al. 1992), latent position models (LPMs) (e.g., Hoff et al. 2002; Sewell and Chen 2015; Athreya et al. 2018), and more. Each class offers a unique mathematical platform for constructing statistical models of networks, with respective strengths and weaknesses.

A persistent challenge in statistical network analysis applications is how to compare different models and select models for specific network data sets. The literature has primarily focused on model selection problems within specific classes of models (SBMs: Wang and Bickel (2017); Latouche et al. (2014); ERGMs: Hunter et al. (2008); Yin et al. (2019); LSMs: Ryan et al. (2017); Loyal and Chen (2023)). At present, the literature which explores methods for comparing model fit or performing model selection across models from different mathematical foundations is underdeveloped.

The main contributions of this work include:

- Conducting extensive simulation studies that explore the extent to which statistical models for network data can be differentiated based on the spectrum of the graph Laplacian, the results of which demonstrate that the empirical properties of the spectrum of the graph Laplacian hold great potential for predicting model classes of networks.
- Elaborating a novel non-parametric methodology under weak assumptions in network data settings that facilitates comparing models and performing model selection across models with different mathematical foundations. We demonstrate the potential of this proposed methodology with two real-world applications that have been previously studied, validating our findings against established findings.

The rest of the paper is organized as follows. Section "Spectral properties of the graph Laplacian" reviews spectral properties of the graph Laplacian for networks and motivates the use of spectral information in the model selection problem for network data. Our proposed methodology is introduced in Sect. "Methodology". We present experimental studies and simulations in Sect. "Simulation studies", and two applications of our methodology in Sect. "Applications". We conclude with a discussion in Sect. "Conclusion".

Spectral properties of the graph Laplacian

Eigenvalues of the graph Laplacian encode many well-known properties of a network. The multiplicity of the eigenvalue 0 corresponds to the number of connected components in a network (Brouwer and Haemers 2011). The second smallest eigenvalue is known as the algebraic connectivity (Fiedler 1973), and measures the overall connectivity of a graph (de Abreu 2007). It is used to establish Cheeger inequalities (Donetti et al. 2006), which have applications in image segmentation (Shi and Malik 2000), graph clustering (Kwok et al. 2013) and expander graphs (Hoory et al. 2006). The subsequent eigenvalues of the graph Laplacian have been used to establish inequalities in the minimal number of cuts (edge deletions) required to partition a network into independent subnetworks (Bollobás and Nikiforov 2004). In the case of isomorphic networks, the corresponding graph Laplacian matrices will be similar, so their eigenvalue decomposition will be the same. In our context, this means one can always differentiate two non-isomorphic networks if their eigenvalues differ. The reverse result is not generally true as there are graphs possessing the same eigenvalue decomposition (cospectral) that are not isomorphic (Cvetković et al. 1980). However, numerical evidence suggests that the fraction of (non-isomorphic) cospectral graphs tends to zero as the number of nodes in a graph grows (Brouwer and Haemers 2011).

Several applications of spectral decomposition of the graph Laplacian have been proposed in the network analysis literature. For example, Lei and Rinaldo (2015) established the consistency of the spectral clustering method for stochastic block models. Another example is in Newman (2006), where a family of community detection algorithms were proposed for networks based on the spectral decomposition of the graph Laplacian. Athreya et al. (2018) provides an extensive survey of results in consistency, asymptotic normality, hypothesis tests, and inference for random dot product graphs based on the spectral embedding of the Laplacian. As a last example, Shore and Lubin (2015) proposed a spectral based statistic for evaluating goodness-of-fit for network models reminiscent of the R^2 statistic in regression settings. This statistic compares the eigenvalues of the graph Laplacian in a fitted model to the corresponding eigenvalues from a prespecified *null* model (typically taken to be a Bernoulli random graph model, referred to as a density-only model).

In light of these results, it is natural to regard the vector of eigenvalues of the graph Laplacian as a signature of a network, containing important information about its nature and structure, which can then be exploited for the purposes of model evaluation and selection. The methodology introduced in this work is, therefore, motivated by the following considerations:

1. If the true data-generating process is in the list of candidate models, the observed eigenvalues (derived from an observed network) are expected to fall within the spectral distribution of the data-generating process. If, in practice, none of the proposed models are the true generating process, candidate models can still be assessed by their ability to capture the spectrum of the observed graph Laplacian, providing a means for developing a method for model selection.
2. We can obtain a relative measure of fit among competing models depending on how well the spectrum of the observed graph Laplacian is captured by candidate models, providing a means to not only select a best-fitting model, but also to compare the fit of the best-fitting model to unselected alternatives.
3. Our methodology requires no parametric assumptions on the data-generating process and can compare models across different mathematical platforms, including models which do not have a well-defined likelihood function or which are constructed through a stochastic process, examples of which include agent-based models (e.g., Snijders et al. 2010; Jackson and Watts 2002) and generative algorithms based on preferential attachment models (e.g., Barabasi and Albert 1999; Zeng et al. 2013).

Methodology

We consider simple undirected networks defined on a set of $N \geq 3$ nodes which we denote by $\mathcal{N} := \{1, \ldots, N\}$. The corresponding adjacency matrix is denoted by $\boldsymbol{X} \in \{0,1\}^{N \times N}$, where $X_{i,j} = 1$ corresponds to the event that there is an edge between nodes i and j and $X_{i,j} = 0$ otherwise. We adopt the standard conventions for undirected networks and assume that $X_{i,j} = X_{j,i}$ (for all $\{i,j\} \subset \mathcal{N}$) and $X_{i,i} = 0$ (for all $i \in \mathcal{N}$);

extensions to directed networks are discussed Simulation study 2. Extensions to networks with valued edges are possible, but beyond the scope of this work. The degree of node $i \in \mathscr{N}$ is defined to be

$$d_i \quad := \quad \deg_i(\boldsymbol{X}) = \sum_{j=1}^{N} X_{i,j},$$

defining $\boldsymbol{d} \quad := (d_1, \ldots, d_N) \in \{0, 1, \ldots, N-1\}^N$ to be the vector of node degrees of the network. The graph Laplacian is defined as $\boldsymbol{L}(\boldsymbol{X}) \quad := \mathrm{diag}(\boldsymbol{d}) - \boldsymbol{X}$, where $\mathrm{diag}(\boldsymbol{d})$ is the $N \times N$ diagonal matrix with diagonal $\boldsymbol{d}$. Since $\boldsymbol{L}(\boldsymbol{X})$ is symmetric and positive semi-definite (Brouwer and Haemers 2011), the eigenvalues of $\boldsymbol{L}(\boldsymbol{X})$ are real and non-negative. Throughout, let $\boldsymbol{\lambda} \in \mathbb{R}^N$ denote the vector of ordered eigenvalues (from smallest to largest) of the graph Laplacian matrix $\boldsymbol{L}(\boldsymbol{X})$. The vector $\boldsymbol{\lambda}$ will depend on the adjacency matrix $\boldsymbol{X}$ through $\boldsymbol{L}(\boldsymbol{X})$, however, for ease of presentation, we do not make this dependence explicit notationally, as it will be clear contextually.

We outline a methodology for model selection in network data settings that exploits the spectral properties of the graph Laplacian, motivated by the considerations discussed in the previous section. We assume there is a fully observed network denoted by its observed adjacency matrix $\boldsymbol{X}_{\mathrm{obs}}$. The corresponding observed vector of eigenvalues of the observed graph Laplacian $\boldsymbol{L}(\boldsymbol{X}_{\mathrm{obs}})$ is denoted by $\boldsymbol{\lambda}_{\mathrm{obs}}$. Our inferential goal is to select a best fitting model for the observed network $\boldsymbol{X}_{\mathrm{obs}}$ from a set of candidate models $\{\mathscr{M}_1, \ldots, \mathscr{M}_M\}$ $(M \geq 2)$, which typically will consist of models already fit to the observed network $\boldsymbol{X}_{\mathrm{obs}}$. We frame the problem as a classification problem and aim to construct a classifier $\mathcal{P} : \mathbb{R}^N \mapsto \{1, \ldots, M\}$ which will predict a model class for an observed network. This classifier is trained on the graph Laplacian spectrum of simulated networks from each of the candidate model and predicts a model $m^\star \in \{1, \ldots, M\}$ from the set of candidate models $\{\mathscr{M}_1, \ldots, \mathscr{M}_M\}$.

We present our model selection method algorithm in Table 1. Our methodology aims to exploit the information contained in the empirical distribution of the eigenvalues of the graph Laplacian matrices to select the most appropriate class for the observed vector of eigenvalues. We do this by training a classifier $\mathcal{P} : \mathbb{R}^N \mapsto \{1, \ldots, M\}$ to differentiate candidate models $\{\mathscr{M}_1, \ldots, \mathscr{M}_M\}$ based on the spectral distribution of their

Table 1 Description of the model selection algorithm

Model selection procedure:
1. Simulate K networks $\boldsymbol{X}^{(m,1)}, \ldots, \boldsymbol{X}^{(m,K)}$ from each of the candidate models $\mathscr{M}_m \in \{\mathscr{M}_1, \ldots \mathscr{M}_M\}$.
2. For each $\boldsymbol{X}^{(m,k)}$, compute its graph Laplacian matrix $\boldsymbol{L}(\boldsymbol{X}^{(m,k)})$ and the corresponding vector of eigenvalues $\boldsymbol{\lambda}^{(m,k)} \in \mathbb{R}^N$.
3. Construct a design matrix $\boldsymbol{D} \in \mathbb{R}^{(KM) \times N}$ by stacking the KM vectors of eigenvalues $\boldsymbol{\lambda}^{(m,k)}$ to form the rows of $\boldsymbol{D}$.
4. Train a classifier $\mathcal{P} : \mathbb{R}^N \mapsto \{1, \ldots, M\}$ to predict a model $m^\star \in \{1, \ldots, M\}$ using the K simulated vectors of eigenvectors $\boldsymbol{\lambda}^{(m,k)}$ for each class $m \in \{1, \ldots, M\}$ contained in the design matrix $\boldsymbol{D}$.
5. Compute the graph Laplacian matrix $\boldsymbol{L}(\boldsymbol{X}_{\mathrm{obs}})$ for the observed network $\boldsymbol{X}_{\mathrm{obs}}$ and the corresponding vector of eigenvalues $\boldsymbol{\lambda}_{\mathrm{obs}}$.
6. Predict a class $m^\star = \mathcal{P}(\boldsymbol{\lambda}_{\mathrm{obs}})$ for the observed network using the trained classifier from Step 4 and set $\mathscr{M}^\star = \mathscr{M}_{m^\star}$.

corresponding graph Laplacians. If the observed vector of eigenvalues is an outlier compared to the simulated distribution of eigenvalues from a particular model, we have evidence to reject said model as the true data-generating process in favor of a different model for which the observed vector of eigenvalues is more likely. Naturally, one might wonder if different models may give rise to the same distribution of eigenvalues. As remarked in the previous section, numerical evidence suggests the fraction of non-isomorphic graphs that share the same eigenvalue decomposition of its Laplacian tends to zero as the number of nodes tends to infinity (Brouwer and Haemers 2011). Because of this, if two models result in networks with similar eigenvalue distributions, we may consider both models to fit the observed network equally well in this regard. On the issue of model misspecification (which we take to mean that the true data-generating model is not a candidate model in the set $\{\mathscr{M}_1, \ldots, \mathscr{M}_M\}$), we can still utilize the proposed method to identify the candidate model out of the list of proposed candidate models which is closest to or most plausible for the observed network with respect to the distribution of the eigenvalues of the graph Laplacian.

Selection and training of the classifier

Real-life networks can possess hundreds, thousands, or even millions of nodes. As the dimension of the vector of eigenvalues of the graph Laplacian matrices is equal to the number of nodes in the network, classification methods based on eigenvalues of the graph Laplacian matrix will be prone to the usual challenges of high dimensional classification. Since the literature on classification methods is quite extensive, it may seem that the choice of classifier is a critical step in our methodology. However, our results demonstrate that the choice of classifier may not significantly affect the results of our methodology under certain conditions. We briefly discuss some practical considerations of selecting the classifier.

There are several supervised algorithms to choose from when training a classification rule. Linear discriminant analysis, perhaps one of the oldest classifiers, requires the computation of the inverse of the covariance matrix of features, and as such, suffers a decay in performance as the number of features grows (Bickel and Levina 2004). In general, linear classifiers based on projections of features (such as Principal Components Analysis or Partial Least Squares) can perform poorly for large dimensional classification tasks, except under stringent specific circumstances (Cai and Chen 2010). A second common classification approach are distance-based clustering algorithms, which include k-nearest neighbor and nearest-centroids methods as classic examples. Such methods in high-dimensional settings are reviewed by Cai and Chen (2010), whose findings suggest that, overall, a large number of noisy features (with low classification power) deteriorates the performance of such algorithms. The last broad class of classifiers are classifiers based on minimizing loss functions. These methods include support vector machines, neural networks, and boosting algorithms, and are generally regarded as the best-performing class of algorithms for minimizing a classification error. Succinctly, these methods learn a decision rule by minimizing a loss function plus a regularization term to help control the effects of over-fitting. Within this class, eXtreme Gradient Boosting (XGBoost) may be considered a state-of-the-art algorithm, and has gained notoriety for being one of the most prominent choices of classifiers in machine learning competitions (Chen and

Guestrin 2016). As one of its outstanding characteristics, XGBoost is virtually unparalleled in terms of scalability (in both sample size and number of features) and accuracy.

In the rest of this paper, we use exclusively XGBoost, except in one simulation study, where we compare the performance of our methodology under different classifiers. Regarding training, we advise including a feature selection step, which may involve filtering eigenvalues with low importance scores, as well as potentially adding new features from the observed eigenvalues. An example in the latter case is the sum of eigenvalues, which corresponds to twice the number of edges in an undirected graph (Brouwer and Haemers 2011), providing relevant network information useful to distinguish between models. We follow this approach by both filtering the specific eigenvalues we use in the classifier through subset selection, which is a standard step of the XGBoost algorithm, as well as including the sum of eigenvalues as a measure of the density of the network, as discussed. We provide additional details about using XGBoost in the proposed methodology in the supplement.

Model comparison

Many classification algorithms provide more than just a predicted class, often returning a vector of propensity scores or probabilities for each class $\boldsymbol{s} = (s_1, \ldots, s_M)$ with the property that $\|\boldsymbol{s}\|_1 = 1$. This measure is not invariant to the number of models being considered (here, the number of classes in the classifier). If more models are added to the set of candidate models, the propensity scores could shrink simply because more models are being considered under the condition that $\|\boldsymbol{s}\|_1 = 1$, in which case the interpretation of the raw propensity scores $s_1, \ldots, s_M$ will depend on the number of models M. To overcome this issue and facilitate the comparison of models, we propose to normalize the propensity scores to obtain a relative measure of each models performance, relative to the best performing model. To this end, we define

$$\tilde{s}_i \quad := \quad \frac{s_i}{\|\boldsymbol{s}\|_\infty}, \qquad \text{for each } i = 1, \ldots, M,$$

to be the normalized score, which is equal to 1 for the highest scoring model. By rescaling all propensity scores in this manner, the number of models M considered in the candidate set of models does not affect the interpretation of the relative propensity scores $\tilde{s}_1, \ldots, \tilde{s}_M$.

Selection of the number of simulated networks *K*

The sample size KM of the design matrix $\boldsymbol{D} \in \mathbb{R}^{(KM)\times N}$ from which to build a classifier depends on the number K of simulated networks $\boldsymbol{X}^{(m,1)}, \ldots, \boldsymbol{X}^{(m,K)}$ drawn from each candidate model $\mathscr{M}_m \in \{\mathscr{M}_1, \ldots \mathscr{M}_M\}$. We assessed the effect that choosing different values of K has on the performance of our proposed methodology by comparing the classification accuracy rate under different values of K, network sizes, and model parameters. To this end, we partitioned the design matrix into training and test sets to avoid a fictitiously high accuracy rate from potential overfitting. After experimenting with different values of K in the range of 100 to 10000, we found that this choice was not determinant in the success of our procedure. More concretely, we found small variations (around 1%) in test set accuracy as K varies between 100 and 10000. As such, we use

$K = 100$ throughout this work but note that different settings may require a larger training set to differentiate models successfully.

Model classification with overfit models

One challenge in model selection lies in the fact that certain models may overfit the network, reproducing the observed network (or networks close to the observed network) with a higher propensity than might be plausible. A case in point lies in the following maximally overfit model. Let $\boldsymbol{x}_{\text{obs}}$ be the observed network within a support $\mathbb{X}$. Define

$$\delta_{\boldsymbol{x}_{\text{obs}}}(\boldsymbol{x}) \quad := \quad \mathbb{1}(\boldsymbol{x} = \boldsymbol{x}_{\text{obs}}) = \begin{cases} 1 & \boldsymbol{x} = \boldsymbol{x}_{\text{obs}} \\ 0 & \boldsymbol{x} \neq \boldsymbol{x}_{\text{obs}} \end{cases}, \qquad \text{for all } \boldsymbol{x} \in \mathbb{X}.$$

With probability 1, any sequence of networks $\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(m)}$ generated from the distribution $\delta_{\boldsymbol{x}_{\text{obs}}}$ will possess a sequence of vectors of eigenvalues $\boldsymbol{\lambda}^{(1)}, \ldots, \boldsymbol{\lambda}^{(m)}$ which are identical to the observed $\boldsymbol{\lambda}_{\text{obs}}$ for $\boldsymbol{x}_{\text{obs}}$. Any classifier should systematically prefer this class of overfitted models over any alternative. Although our methodology does not penalize for overfitting, a case could be made that in the case of relatively similar normalized scores, the model with fewer parameters should be preferred at the expense of not capturing the observed eigenvalue distribution as well as other models.

This challenge is not unique to our method, however. When it comes to comparing models from different classes or when there are not theoretical guarantees supporting particular methods for model selection, a general approach in the literature is to compare the distribution of simulated network statistics (such as degree distribution, geodesic distance, edgewise shared partners distribution, etc.) and the observed values of those statistics based on the observed network (Hunter et al. 2008). In this situation, the same pitfalls due to overfitting are shared with our methodology, evidenced by the same example above given by $\delta_{x_{\text{obs}}}$. In practice, one either chooses model specifications which are believed to not be overfitted, or one employs out-of-sample measures for model fit in order to study whether a particular model is representative of the observed network or is perhaps overfitted (Stewart et al. 2019; Yin et al. 2019).

As discussed in the introduction, there exist methods and theory for performing model selection within many prominent classes of models, despite the sparse literature on comparing models from different classes which motivates this work. One approach to countering overfitting within our methodology is to leverage the methods and procedures for model selection within different classes of models and then to move forward into our proposed methodology the best candidate model from each class. For example, Loyal and Chen (2023) elaborate a Bayesian model selection algorithm for latent space models for networks which provably controls overfitting with respect to the dimension of the latent space.

Simulation studies

We conduct several simulation studies in order to demonstrate the potential of our proposed methodology. Specifically, we aim to examine the extent to which the signature of a network is contained within the spectrum of the graph Laplacian. Simulation studies permit knowledge of the true data-generating model, which facilitates empirical studies which aim to clarify the conditions under which our proposed methodology is able to successfully differentiate different network models and structural properties of networks.

Simulation study 1: curved exponential families

We study the performance of our methodology on curved exponential families, which have gained popularity in the social network analysis community (e.g., Snijders et al. 2006; Hunter and Handcock 2006), as well as other applications (e.g., Obando and de Vico Fallani 2017; Schweinberger et al. 2020; Stivala and Lomi 2021). The prominence of curved exponential family parameterizations for random graph models emerged out of a desire to solve challenges related to degeneracy and fitting of early and ill-posed model specifications (Snijders et al. 2006). Additionally, curved exponential family parameterizations are able to parsimoniously model complex sequences of graph statistics, such as degree sequences and shared partner sequences, without sacrificing interpretability (Hunter 2007; Stewart et al. 2019). A prototypical example used in the social network analysis literature is the geometrically-weighted edgewise shared partner model, which models transitivity through the shared partner sequence (Snijders et al. 2006; Hunter 2007; Stewart et al. 2019).

We simulate networks according to the following model:

$$\mathbb{P}(\boldsymbol{X} = \boldsymbol{x}) \propto \exp\left(\theta_1 \sum_{i<j}^{N} x_{i,j} + \sum_{t=1}^{N-2} \sum_{i<j}^{N} \eta_t(\theta_2, \theta_3)\, \mathrm{SP}_t(\boldsymbol{x})\right), \tag{1}$$

where $\theta_1 \in \mathbb{R}$ controls the baseline propensity for edge formation, and

$$\eta_t(\theta_2, \theta_3) = \theta_2\, \exp(\theta_3) \left[1 - (1 - \exp(-\theta_3))^t\right], t \in \{1, \ldots, N-2\},$$

parameterizes the sequence of shared partner statistics

$$\mathrm{SP}_t(\boldsymbol{x}) = \sum_{i<j}^{N} x_{i,j}\, \mathbb{1}\left(\sum_{h \neq i,j}^{N} x_{i,h}\, x_{h,j} \; = \; t\right), t \in \{1, \ldots, N-2\}.$$

In words, $\mathrm{SP}_t(\boldsymbol{x})$ counts the number of edges in the network between nodes which have exactly t mutual connections, commonly called shared partners in the social network analysis literature. While $\theta_2 \in \mathbb{R}$, in typical applications $\theta_2 \geq 0$ and $\theta_3 \in (0, \infty)$, as values of $\theta_3 < -\log 2$ correspond to models which are unstable in the sense of Schweinberger (2011), and empirical evidence suggests that $\theta_3 \in (0, \infty)$ in many applications (Schweinberger 2011; Stewart et al. 2019). The effect that the GWESP model specified by (1) has on the degree and shared partner distributions of networks is visualized in Fig. 1, where positive values of θ_2 stochastically encourage network formations with more transitive edges, i.e., edges between nodes with at least one shared partner, relative to the Bernoulli random graph model with $\theta_2 = 0$. This is evidenced by the rightward shift in the ESP distribution of the GWESP model, relative to the Bernoulli model.

We take the true data-generating model $\mathscr{M}^\star$ to be the curved exponential family specified by (1) with parameter vector $\boldsymbol{\theta}^\star = (-2.5, \theta_2, 1)$, with θ_2 on a grid covering the interval [0, 0.5]. Note that when $\theta_2 = 0$, the model reduces to a Bernoulli random graph model with edge probability $p = [1 + \exp(-2.5)]^{-1}$. We consider the problem of

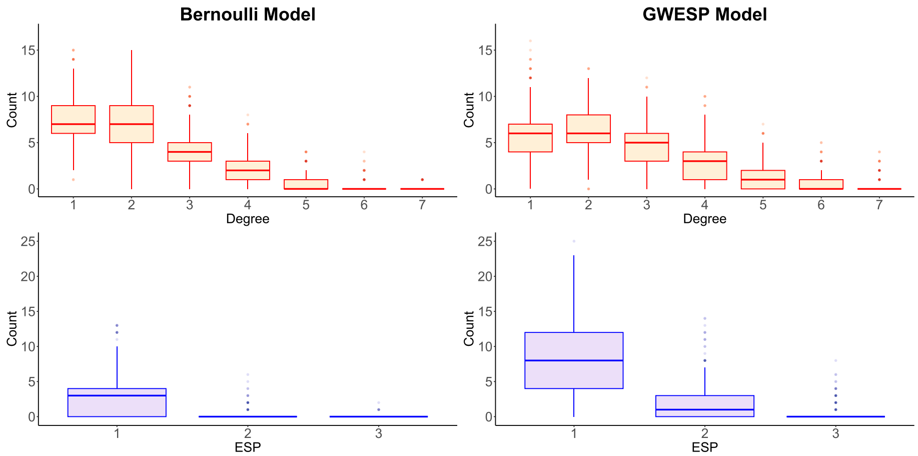

Fig. 1 We visualize the degree and ESP distributions for the Bernoulli and GWESP models with network size $N = 25$. We simulate 1000 networks from (1) with data-generating parameters $(\theta_1, \theta_2, \theta_3) = (-2.5, 0, 1)$ (Bernoulli) and $(\theta_1, \theta_2, \theta_3) = (-2.5, .3, 1)$ (GWESP). By increasing θ_2, the term of shared partner counts in (1) carries more weight. This favors networks with a larger number of shared partners relative to the Bernoulli model. In consequence, denser networks are more likely to be observed as well. Because of these two considerations, we observe a rightward shift in the degree distribution and an upward shift in the ESP distribution in the GWESP model relative to the Bernoulli model

selecting between two models $\mathscr{M}_1$ and $\mathscr{M}_2$, where $\mathscr{M}^\star = \mathscr{M}_1$ and $\mathscr{M}_2$ is the Bernoulli random graph model with edge probability $p = [1 + \exp(2.5)]^{-1}$. By varying θ_2 we are able to study the threshold of effect size (θ_2) for which we are able to correctly detect the presence of transitivity in the network, as modeled by the geometrically-weighted edgewise shared partner model in (1).

We vary the network size $N = 25, 50, 75, 100, 200, 300$, performing 5000 replicates for each network size. The results of this simulation study are summarized in Fig. 2. When θ_2 is close to 0, the point at which $\mathscr{M}_1 = \mathscr{M}_2$, as discussed above, our methodology tends to select $\mathscr{M}_1$ and $\mathscr{M}_2$ with equal probability. However, once θ_2 is sufficiently large (relative to the network size N), our methodology correctly selects $\mathscr{M}_1$ in almost every replicate. The effect of the size of the network is seen as we vary N from 25 to 300. When the network size is larger ($N = 100, 200, 300$), we are able to correctly find the data-generating model $\mathscr{M}_1$ with high probability for smaller values of θ_2. In contrast, we require $\theta_2 \geq .25$ before we are able to have a high confidence in correctly selecting the data-generating model in networks of size $N = 75$, requiring $\theta_2 \geq .5$ for networks of size $N = 25$.

Simulation study 2: reciprocity in directed networks

When the adjacency matrix $\boldsymbol{X}$ is undirected, the corresponding graph Laplacian matrix $\boldsymbol{L}(\boldsymbol{X})$ will be positive semidefinite (Brouwer and Haemers 2011), resulting in a real-valued vector of eigenvalues $\boldsymbol{\lambda} \in \mathbb{R}^N$. However, when $\boldsymbol{X}$ is the adjacency matrix of a directed network, the graph Laplacian, as defined for undirected networks, may not be positive semidefinite, and may involve complex valued eigenvalues. A common adaptation for directed networks in the literature is to consider the incidence matrix $\boldsymbol{B} \in \{0, 1, -1\}^{N \times |E|}$, where $|E|$ is the total number of edges in the network. On each column of the incidence matrix exactly one element will be -1, indicating the

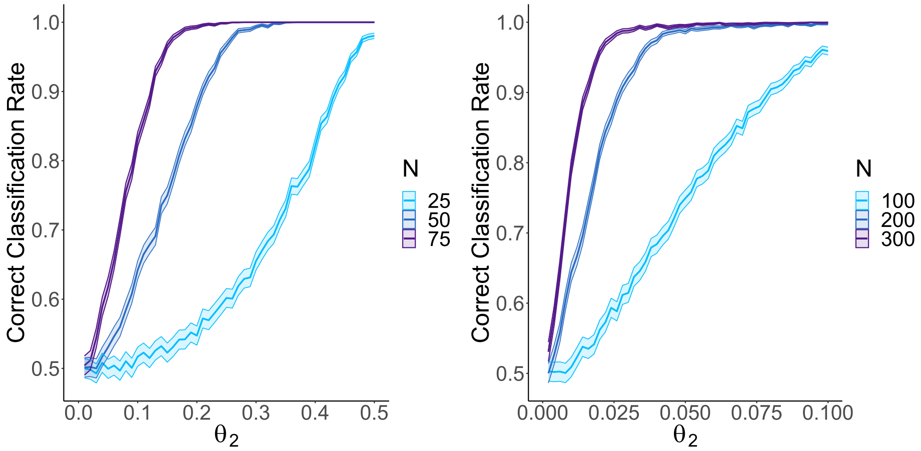

Fig. 2 Results of Simulation study 1. (left) Estimate of the correct classification rate with 95% confidence bands for networks of sizes $N = 25, 50, 75$. We observe that for any fixed θ_2 in the range of (0, 0.5), a larger network size corresponds to a significative larger accuracy rate. (right) Estimate of the correct classification rate with 95% confidence bands for networks of sizes $N = 100, 200, 300$. For this larger scale of network sizes, differences in the θ_2 parameter are now significantly detected as θ_2 varies in the even smaller range of (0, 0.1). Our numerical evidence suggests that for any given fixed value of θ_2, a large enough network size will be likely to detect said difference in θ_2. Conversely, for any fixed network size, differences in θ_2 are almost guaranteed to be detected if the difference is large enough

node where an edge begins, and exactly one element will be 1, indicating the node where said edge ends. Every other entry is zero. In this manner, a directed network is completely specified by listing all existing edges as columns that indicate which nodes are connected and an orientation between them. We can adapt our proposed methodology to directed networks by considering the symmetric graph Laplacian defined by $\mathbf{L} := \mathbf{B}^t\mathbf{B}$ (Brouwer and Haemers 2011).

We simulate directed networks from the probability mass function

$$\mathbb{P}(\boldsymbol{X} = \boldsymbol{x}) \propto \prod_{i<j}^{N} \exp\left(\theta_1 \left(x_{i,j} + x_{j,i}\right) + \frac{\theta_2}{2}\, x_{i,j}\, x_{j,i}\right), \tag{2}$$

We apply our methodology taking $\mathscr{M}_1$ to be the density only model with fixed $\theta_2 = 0$ in (2). We take $\mathscr{M}^\star = \mathscr{M}_2$ to be the general model specified via (2) with unrestricted parameters. We conduct a simulation study by taking $\theta_1 = -2.5$ in both $\mathscr{M}_1$ and $\mathscr{M}_2$, taking $\theta_2 = 0$ in $\mathscr{M}_1$, and varying θ_2 on a uniform grid of 100 values in [0, 1] for $\mathscr{M}_2$. The simulation results in Fig. 3 are based on 1000 replications in each case, reconfirming findings in the previous simulation study which suggested that the ability of our methodology to detect the true data-generating model depends on how far θ_2 is from 0, the point at which $\mathscr{M}_1 = \mathscr{M}_2$, and the size of the network. In other words, larger network sizes seem to allow for earlier detection of a reciprocity term in the case of an exponential model for directed networks. Alternatively, evidence suggests that for any value of a reciprocity effect, our methodology will be able to correctly pick up on that term if the network is large enough. Moreover, this study uniquely demonstrates that our methodology can be applied successfully to directed networks.

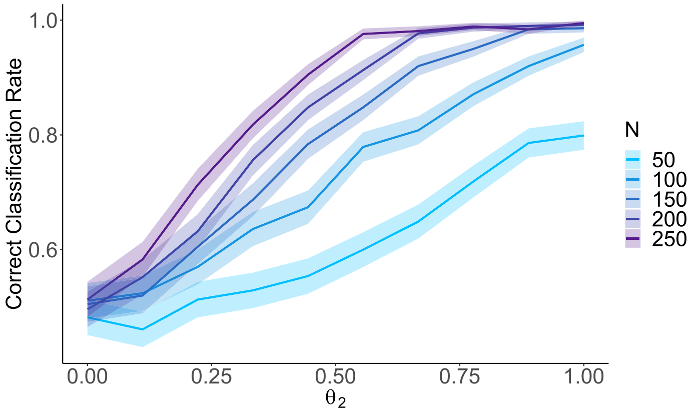

Fig. 3 Results of Simulation study 2. Estimates of the correct classification rate with 95% confidence band for various network sizes *N*. Larger network sizes allow for earlier detection of a reciprocity term in the case of an exponential model for directed networks. Alternatively, evidence suggests that for any value of a reciprocity effect, our methodology will be able to correctly pick up on that term if the network is large enough

Simulation study 3: latent position models

Latent variable models for networks, especially latent position models, have witnessed increased popularity and attention since the seminal work of Hoff et al. (2002). In this class of models, nodes are given a latent position $\boldsymbol{z}_i \in \boldsymbol{Z}$ $(i = 1, \ldots, N)$ in a latent space, typically taken to be the Euclidean space (i.e., $\boldsymbol{Z} = \mathbb{R}^k$), although alternative spaces and geometries have been proposed as well, as is the case of ultrametric spaces (Schweinberger and Snijders 2003), dot product similarity resulting in bilinear forms (Hoff et al. 2002; Athreya et al. 2018), as well as hyperbolic (Krioukov et al. 2010) and elliptic geometries (Smith et al. 2019). Edges in the network are assumed to be conditionally independent given the latent positions of nodes. Following Hoff et al. (2002), we simulate networks in this study accordingly:

$$\log \frac{\mathbb{P}(X_{i,j} = 1 \,|\, \boldsymbol{z}_i, \boldsymbol{z}_j)}{\mathbb{P}(X_{i,j} = 0 \,|\, \boldsymbol{z}_i, \boldsymbol{z}_j)} = \theta - \|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2, \tag{3}$$

where $\theta \in \mathbb{R}$ and $\boldsymbol{z}_i, \boldsymbol{z}_j \in \mathbb{R}^k$. Under this specification, the odds of two nodes forming an edge decreases in the Euclidean distance $\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2$ between the positions of the two nodes in the latent metric space.

We explore the ability of our methodology to detect the true dimension of a latent space by generating networks from the latent Euclidean model described above, varying the dimension of the latent metric space $k \in \{1, 2, 3, 4, 5\}$. Latent positions of nodes are randomly generated from a multivariate normal distribution in dimension $k \in \{1, 2, 3, 4, 5\}$ with zero mean vector and identity covariance matrix. The candidate competings models are generated in the same fashion across dimensions $1, \ldots, 5$. We set $\theta = -2.5$ to ensure a low baseline probability of edge formation, reflecting the sparsity of many real-world networks, and vary the network size $N \in \{50, 100, 150, 200, 250\}$. We apply our model selection methodology in each case and compute the percentage of times our methodology selects each of the candidate latent space models.

We summarize the results of the simulation study in Fig. 4, which demonstrates that our methodology is able to correctly identify the true dimension of the

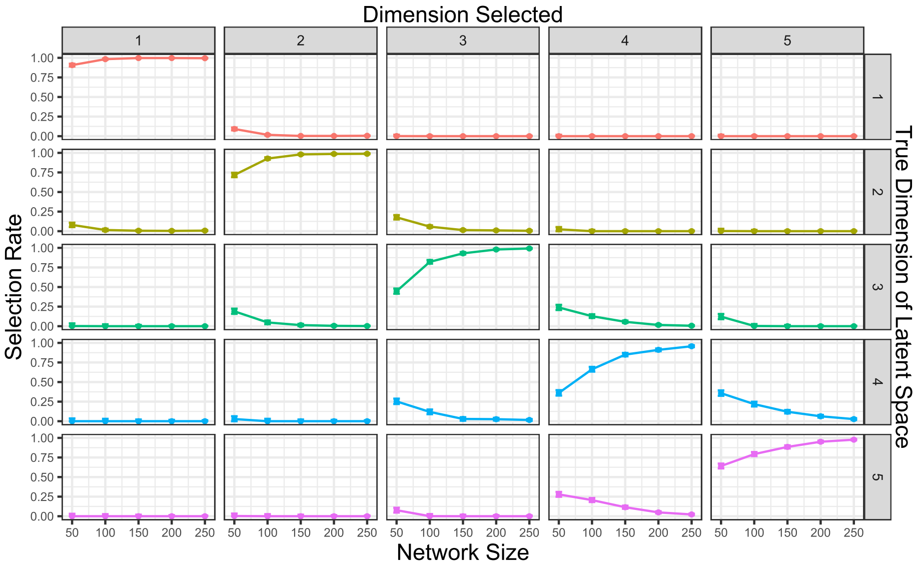

Fig. 4 Results of Simulation study 3. Estimates of the correct classification rate with 95% confidence intervals for network sizes $N = 50, 100, 150, 200, 250$ and across latent space dimensions $k = 1, 2, 3, 4, 5$. The diagonal panels correspond to a correct classification where the selection rate is desired to be the highest. We observe that larger network sizes are associated with a higher accuracy rate. However, as the dimension of the latent space grows, only networks large enough can be confidently classified

data-generating latent space model provided the network size is sufficiently large. The diagonal panels in Fig. 4 correspond to correct selection of the dimension of the latent space. Of particular note, the problem becomes more challenging as the dimension of the latent space grows, but this effect is mitigated as the network size increases, with most correct selection rates in this study close to 1 for networks of size $N = 250$.

Simulation study 4: comparing different latent mechanisms

We next study whether our proposed methodology is capable of distinguishing different latent mechanisms for edge formation in a latent position model. The first one is the same latent space model specified in (3), while the second one replaces the Euclidean distance term $-\|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2$ with the dot product $\boldsymbol{z}_i^t \boldsymbol{z}_k$, commonly referred to as a bilinear form. A related class of latent position models which utilize bilinear forms of latent node positions are random dot product graphs (Athreya et al. 2018). As in the previous simulation study, latent positions of nodes are randomly generated from a multivariate normal distribution with zero mean vector but this time with covariance matrix $\sigma^2 \boldsymbol{I}$, with $\boldsymbol{I}$ being the identity matrix (of appropriate dimension) and $\sigma^2 \in \{0.1, 0.2, \ldots, 1.0\}$ a scale factor. As the scale factor tends to zero, both models converge to a density-only model so detecting the true generating process becomes more difficult. We summarize the results of the simulation study in Fig. 5, which demonstrates that our methodology is able to correctly identify the true model (distance based) when compared to a bilinear (similarity based) model. Of particular note, performance improves as the dimension of the latent space increases and as the network size increases, as in the previous studies conducted.

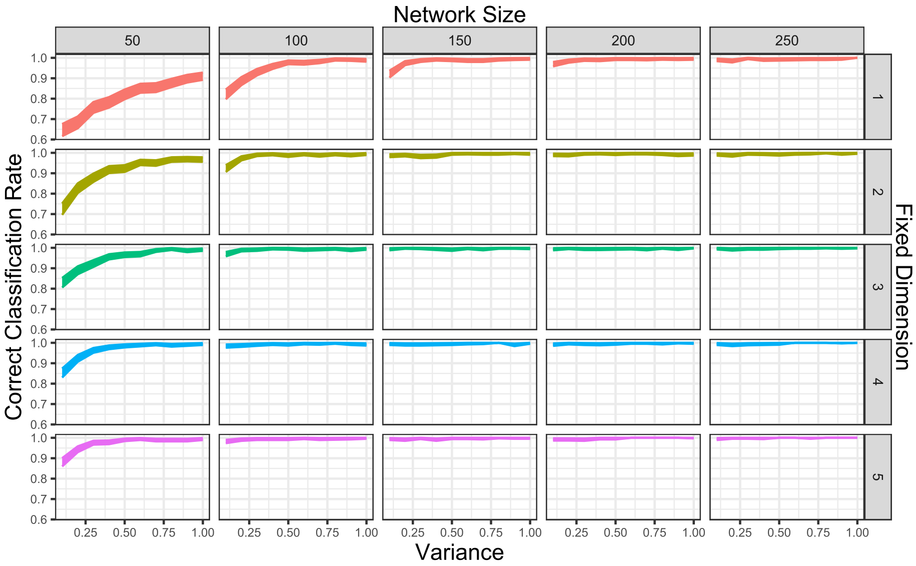

Fig. 5 Results of Simulation study 4 comparing a distance-based model (true model) to a similarity based model. Estimates of the correct classification rate with 95% confidence band for different networks at different sizes and across different dimensions of latent spaces. Our procedure is significantly more accurate in detecting differences in the latent space mechanism as the network size and dimension of the latent space increase. As the variance coefficient is closer to zero, the two model configurations collapse to a density-only model so correct classification becomes more difficult

Simulation study 5: stochastic block models

Following Wang and Bickel (2017), we simulate networks in this study according to a stochastic block model with $K \in \{1, 2, \ldots\}$ blocks:

$$\mathbb{P}(X_{i,j} = x_{i,j} \,|\, Z_i = z_i,\, Z_j = z_j) = \pi_{z_i,z_j}, x_{i,j} \in \{0,1\}, \quad \{i,j\} \subset \mathscr{N}, \tag{4}$$

where $z_i \in \{1, \ldots, K\}$ denotes the block membership of node $i \in \mathscr{N}$ and $\pi_{k,l} = \pi_{l,k} \in (0,1)$ $(1 \leq k \leq l \leq K)$. The block membership variables $Z_1, \ldots, Z_N$ can either be fixed (modeled as degenerate random variables) or can follow a probability distribution (e.g., the Multinomial distribution). In the following simulation study, we will assume that each Z_i $(i \in \mathscr{N})$ is drawn independently from a discrete uniform defined on $\{1, \ldots, K\}$.

This simulation study explores the extent to which our proposed methodology is able to distinguish networks which were generated from stochastic block models with different numbers of blocks, i.e., exploring the methods' fitness for identifying the correct number of clusters. We generate networks from the stochastic block model described above, varying the true number of clusters $K \in \{2, 3, 4, 5\}$ and assigning node memberships to each block randomly based on a discrete uniform distribution defined on $\{1, \ldots, K\}$ as mentioned above. Each of the candidate models is generated in the same way, with the candidate models being defined to follow a stochastic block model with a number of clusters in the range of $\{2, \ldots, 5\}$. We fix all within-block probabilities to be a value $p_{\text{within}} \in \{.05, .1, .15, .2\}$ and similarly fix all between-block probabilities to be a constant $p_{\text{between}} = (2/3)\, p_{\text{within}}$, and consider networks of size $N \in \{200, 500\}$. Under this setup, our simulation study ensures a form of relative sparsity among the blocks (in the sense that the within-block subgraphs will be relatively more dense than the between-block subgraphs), as well as making our parameter values comparable to the simulation studies conducted by Wang and Bickel (2017). We apply our model selection methodology in each case and compute the percentage of times our methodology selects each of the candidate stochastic block models.

We summarize the results of the simulation study in Fig. 6, which demonstrates that our methodology is able to correctly distinguish networks generated from stochastic block models with differing numbers of blocks, i.e., we are able to identify the data-generating number of clusters, provided the within probability is sufficiently large, with a high selection rate. The diagonal panels in Fig. 6 correspond to the correct selection of the number of clusters. Interestingly, we notice the problem becomes more challenging as the true number of clusters grows, but this effect is mitigated as the within-block probability increases or the network size increases.

Simulation study 6: effect of the choice of classifier

In this study, we repeat Simulation study 1 using three different classifiers, XGBoost (Chen and Guestrin 2016), Random Forest (Ho 1995; Liaw and Wiener 2002) and Naive Bayes (Hand and Yu 2001; Majka 2019). Doing so allows us to examine the effect that the choice of classifier has on the results of this simulation study, as well as to explore the relative effectiveness of each classifier in this simulation study. Figure 7 shows a similar performance for all classifiers in this simulation study, with the notable exception being the naive Bayes classifier when networks are size 25, suggesting that the choice of classifier has a weak effect on the performance of our proposed methodology, provided the network is sufficiently large. In line with conclusions in the previous simulation studies, larger network sizes result in more pronounced model signatures. In light of these

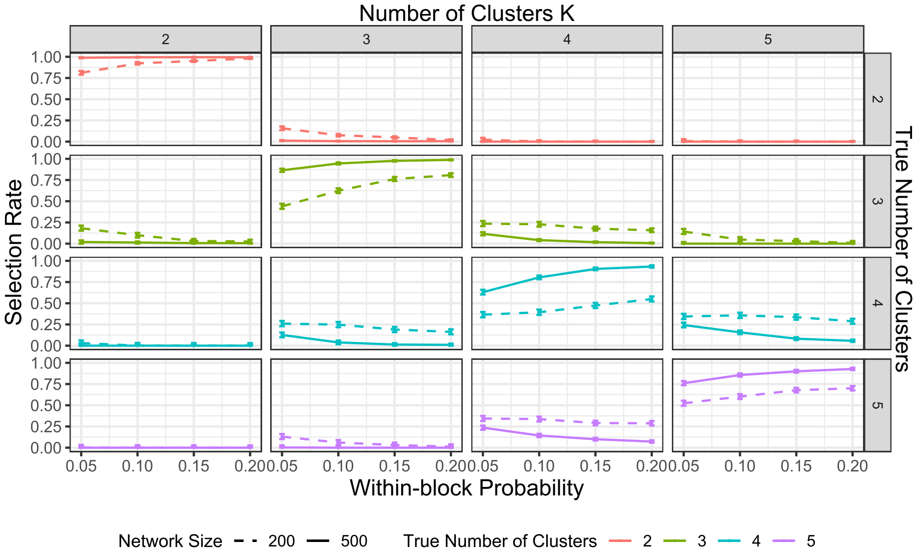

Fig. 6 Results of Simulation study 5. Estimates of the correct classification rate with 95% confidence intervals for network size $N \in \{200, 500\}$, within-block probability $p_{\text{within}} \in \{.05, .1, .15, .2\}$, and true number of clusters $K \in \{2, 3, 4, 5\}$. The diagonal panels correspond to a correct classification where the selection rate is desired to be the highest. We observe that accuracy increases as network size or within-block probability increases. However, as the number of clusters grows, only when the within probability is high enough we can be confident in accurately identifying the true number of clusters

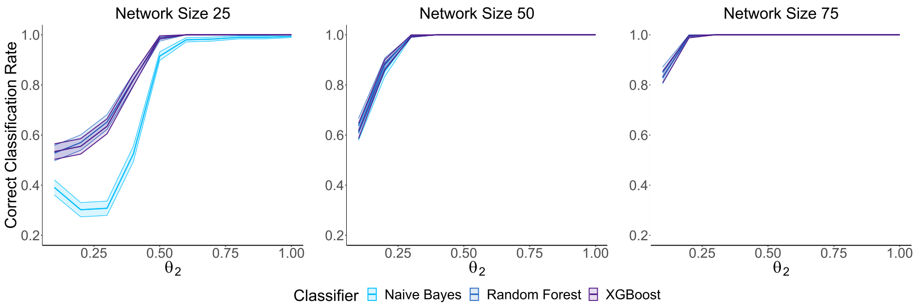

Fig. 7 Results of Simulation study 6. Estimates of the correct classification rate with 95% confidence band for different classifiers. As the network size increases, the choice of the classifier becomes less important, since eventually all classifiers agree and exhibit the same accuracy rates across the range of alternative models being considered. For lower network sizes and a range of low values in the effect size θ_2, differences in the classifier employed might emerge

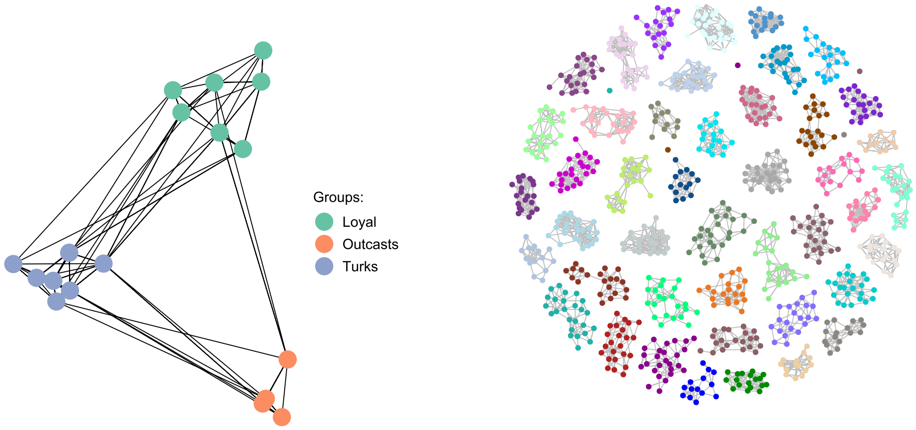

Fig. 8 (left) Sampson's monastery network, with node colors corresponding to group. (right) School classes friendship network based on 44 third grade classes, with node colors corresponding to class

results, the effect of the choice of classifier appears to diminish if the model signal is sufficiently strong.

Applications

In order to study the performance of our proposed model selection methodology in applications to real-world network data, we study two network data sets which have previously been studied in the literature, in order to have a baseline for evaluating whether our methodology confirms existing results and knowledge about these networks. The first is Sampson's monastery network (Sampson 1968), whereas the second is a friendship network consisting of third grade classes (Stewart et al. 2019). We visualize each network in Fig. 8, discuss each data set in further detail in their respective sections.

Application 1: Sampson's monastery network

We apply our model selection methodology to the Sampson's monastery network data on social relationships (likeness) among 18 monk novices in a New England monastery in 1968 (Sampson 1968). Based on the existing literature studying this network, we propose different model structures for this network which are well-designed to capture the community structure known to be a critical component of the network. In order to model this structure, stochastic block models have been applied to the network (Airoldi et al. 2008), as well as latent position models with a hierarchical group-based prior distribution structure on the latent positions (Handcock et al. 2007).

We consider the following models:

- SBM: $\mathscr{M}_1$–$\mathscr{M}_4$ correspond to stochastic block models with $K = 1, 2, 3, 4$ blocks ($\mathscr{M}_1$ being equivalent to a density only model).
- LPM: $\mathscr{M}_5$–$\mathscr{M}_8$ correspond to latent position models with model terms for density and reciprocity and latent space dimensions $K = 1, 2, 3, 4$.
- GLPM: $\mathscr{M}_9$–$\mathscr{M}_{20}$ combine the two previous specifications by utilizing the hierarchical group-based prior distribution structure of Handcock et al. (2007), considering all combinations of group number $K = 2, 3, 4$ and latent space dimension $d = 1, 2, 3, 4$.

Each model was fit and our model selection methodology was applied to choose the best fitting model among the candidate models. The latent space models were fit with Krivitsky and Handcock (2014) and the stochastic block models were fit with Leger (2016). Table 2 presents the results. The model with the highest propensity score is $\mathscr{M}_4$, the stochastic block model with $K = 4$ blocks. We can interpret this as model $\mathscr{M}_4$ displaying a better agreement (relative to the alternative models) between its simulated graph Laplacian eigenvalues and the observed eigenvalues in Sampson's graph Laplacian.

It has been well-established in the literature that the Sampson's monastery network features strong community structure (Handcock et al. 2007; Airoldi et al. 2008),

Table 2 Propensity scores s_i and normalized propensity scores $\tilde{s}_i$ for models $\mathscr{M}_1$–$\mathscr{M}_{20}$ for the Sampson's monastery network

Model	s_i	$\tilde{s}_i$	Model	s_i	$\tilde{s}_i$
$\mathscr{M}_1$ (SBM, $K = 1$)	0.002	0.004	$\mathscr{M}_2$ (SBM, $K = 2$)	0.003	0.007
$\mathscr{M}_3$ (SBM, $K = 3$)	0.032	0.077	$\mathscr{M}_4$ (SBM, $K = 4$)	**0.410**	**1**
$\mathscr{M}_5$ (LPM, $d = 1$)	0.028	0.068	$\mathscr{M}_6$ (LPM, $d = 2$)	0.028	0.069
$\mathscr{M}_7$ (LPM, $d = 3$)	0.005	0.013	$\mathscr{M}_8$ (LPM, $d = 4$)	0.003	0.008
$\mathscr{M}_9$ (GLPM, $K = 2, d = 1$)	0.023	0.055	$\mathscr{M}_{10}$ (GLPM, $K = 3, d = 1$)	0.044	0.108
$\mathscr{M}_{11}$ (GLPM, $K = 4, d = 1$)	0.043	0.104	$\mathscr{M}_{12}$ (GLPM, $K = 2, d = 2$)	0.060	0.147
$\mathscr{M}_{13}$ (GLPM, $K = 3, d = 2$)	0.083	0.202	$\mathscr{M}_{14}$ (GLPM, $K = 4, d = 2$)	0.036	0.089
$\mathscr{M}_{15}$ (GLPM, $K = 2, d = 3$)	0.020	0.050	$\mathscr{M}_{16}$ (GLPM, $K = 3, d = 3$)	0.041	0.101
$\mathscr{M}_{17}$ (GLPM, $K = 4, d = 3$)	0.061	0.148	$\mathscr{M}_{18}$ (GLPM, $K = 2, d = 4$)	0.012	0.029
$\mathscr{M}_{19}$ (GLPM, $K = 3, d = 4$)	0.030	0.074	$\mathscr{M}_{20}$ (GLPM, $K = 4, d = 4$)	0.035	0.085

Bold font identify the model with the highest propensity score, i.e, the model selected by our proposed method

Our methodology identifies model $\mathscr{M}_4$ as the most appropriate to describe Sampson's network. This is based on a higher predicted probability for the observed eigenvalues in Sampson's graph Laplacian in belonging to $\mathscr{M}_4$'s class of eigenvalues featuring three labeled groups. However, statistical analyses have revealed the presence of a potential fourth group, evidenced in analysis which employ mixed membership stochastic block models (Airoldi et al. 2008), as well as evidence in studies which employ latent position models which suggests certain nodes may have strong connections to two or more labeled groups (Handcock et al. 2007). Within the context of the models we considered here, the choice of a stochastic block model with $K = 4$ blocks appears to be sufficient to capture the mixing patterns of the communities as well as the reciprocity from the inclusion of a reciprocity term. We hold the opinion that the expression of transitivity is not sufficiently strong in this network, otherwise the latent position model with $K = 4$ groups would potentially serve as a better model, as latent position models are able to capture network transitivity through the latent metric space. Figure 9 supports this claim by simulating networks from $\mathscr{M}_4$ and comparing the empirical triangle count distribution of these simulated networks to the observed number of triangles in the network, demonstrating good model fit in this regard.

Application 2: third grade school classes friendship network

We end the section with an application to a multilevel network consisting of 6,607 third grade students over 306 classes across 176 primary schools in Poland in the 2010/2011 academic year. A complete description of the data set can be found in Stewart et al. (2019). Multilevel network data have become a focal point of attention in many applications and come in many different types (Snijders 2016). In this application, third grade students are nested within classes which are themselves nested within schools. The network is then a multilevel network where the first level consists of the students, the second consists of the classes, and the third level consists of the schools. Stewart et al. (2019) has extensively studied this data set, providing the closest we can get to a data-generating model.

The network contains 306 classes, but features a significant portion of non-response resulting in a large percentage of missing edge data in the network. The issues of missing data require careful consideration and are beyond the scope of this work. As

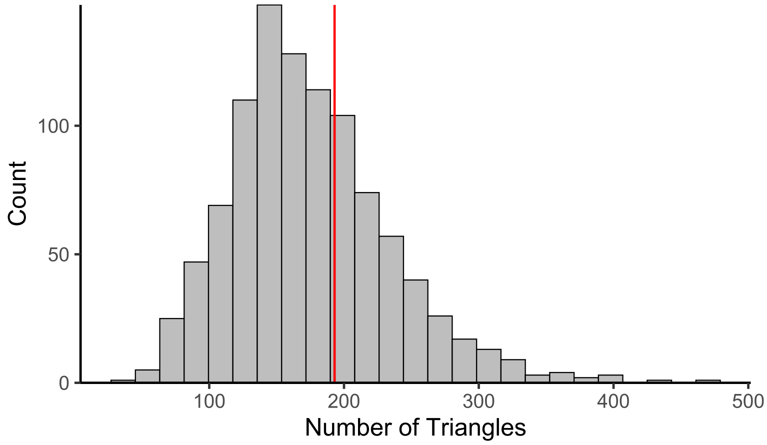

Fig. 9 Fit of the observed number of triangles in the Sampson network (indicated in red) relative to the triangle distribution in simulated networks from $\mathscr{M}_4$. The consistency between this distribution and the real observed statistic suggests that $\mathscr{M}_4$ is a good choice to model Sampson's network

such, we restrict our study in this work to the 44 classes within the multilevel network that did not feature any missing edge data. This multilevel network data set naturally fits into the local dependence framework of Schweinberger and Handcock (2015), for which class based sampling is justified under the local dependence assumption (Proposition 3& Theorem 2, Schweinberger and Stewart 2020); additional details of the data set can be found in Stewart et al. (2019). The data set employed is a directed network of 906 nodes corresponding to the individual students within the 44 classes without missing edge data, where a directed edge $i \to j$ implies that person i stated they were friends person j. Part of the data collected included the sex of each student (recorded as male or female).

In this application, we study whether our proposed methodology for model selection coincides with published findings for this network by studying Models 1–4 published in Stewart et al. (2019), which we summarize in Table 3. The first three model terms (edges, mutual, and out-degree terms) control for structural effects within the network, including density, reciprocity, and fitting the degree distribution. The next three model terms adjust for different sex-based edge effects and homophily. The last three model terms correspond to the geometrically-weighted shared partner (GWESP) term specified in (1) that was studied in Simulation study 1. The inclusion of this model term is aimed at capturing a stochastic tendency towards network transitivity and triad formations based on values of the base parameter (θ_2 in (1)) and the decay parameter (θ_3 in (1)). Model 1 includes no GWESP term, whereas Model 2 and Model 3 fix the decay parameter at specific values found in the literature, reducing the curved exponential family to a canonical exponential family (see discussions in Hunter (2007) and Stewart et al. (2019)). Model 4 estimates the decay parameter.

We fit each of the four models $\mathscr{M}_1, \mathscr{M}_2, \mathscr{M}_3, \mathscr{M}_4$ and apply our model selection methodology, which selects model $\mathscr{M}_4$ (propensity $= 0.9967$) above all other candidate models (Table 4). This coincides with the findings of Stewart et al. (2019), who explored the fit of various models to the data set with respect to common-place heuristic measures (Hunter et al. 2008), as well as out-of-sample measures and through the Bayesian Information Criterion (BIC). Figure 10 demonstrates the model fit to relevant network features in the statistical network analysis literature.

Table 3 Descriptions of models 1–4 found in Stewart et al. (2019)

Model Term	$\mathscr{M}_1$	$\mathscr{M}_2$	$\mathscr{M}_3$	$\mathscr{M}_4$
Edges	✓	✓	✓	✓
Mutual	✓	✓	✓	✓
Out-degrees (1–6)	✓	✓	✓	✓
Out-degree (Female)	✓	✓	✓	✓
In-degree (Female)	✓	✓	✓	✓
Sex-match	✓	✓	✓	✓
GWESP (decay parameter fixed at 0)		✓		
GWESP (decay parameter fixed at .25)			✓	
GWESP (decay parameter estimated)				✓

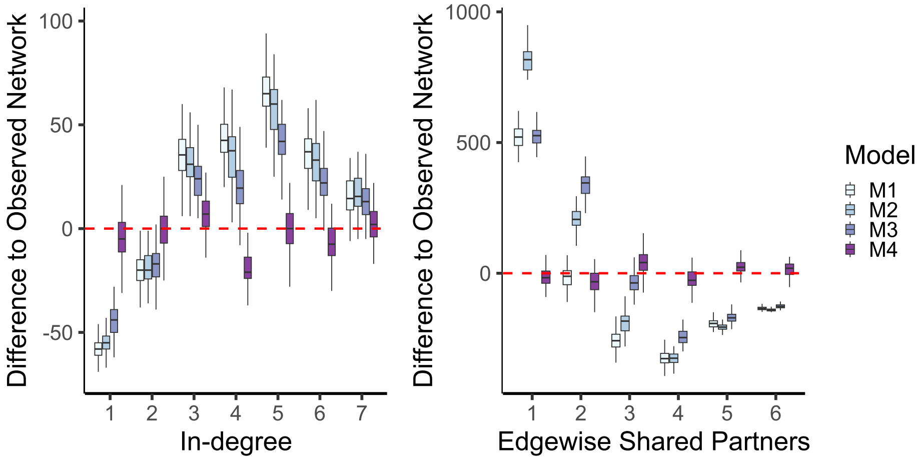

Fig. 10 Difference in selected statistics between fitted models and the observed Polish school network. Values closer to the red horizontal line show better agreement with the observed statistic. (left) Difference between observed in-degree statistic and predicted distribution under different candidate models. (right) Difference between the observed edgewise shared partner statistic sequence and predicted distribution under different candidate models. We observe better agreement for model $\mathscr{M}_4$ compared to the alternative models in both statistics. This agreement is consistent with the result of our methodology that selects model $\mathscr{M}_4$ as the best-fitting model for the Polish school network data

Table 4 Propensity scores s_i and normalized propensity scores $\tilde{s}_i$ for models $\mathscr{M}_1$–$\mathscr{M}_4$ for the multilevel school network

Model	s_i	$\tilde{s}_i$	Model	s_i	$\tilde{s}_i$
$\mathscr{M}_1$ (No decay)	0.0004	0.0004	$\mathscr{M}_2$ (Decay fixed at 0)	0.0006	0.0006
$\mathscr{M}_3$ (Decay fixed at 0.25)	0.0023	0.0023	$\mathscr{M}_4$ (Decay estimated)	**0.9967**	**1**

Bold font identify the model with the highest propensity score, i.e, the model selected by our proposed method

Conclusion

We introduced a novel non-parametric methodology for model selection for network data. This methodology can be applied to a wide class of network models under very weak assumptions (namely, simulating networks from a fitted model), and it effectively allows the comparison of models under very different mathematical foundations. Our method is based on two key ideas. The first one is to exploit the topographical information in the spectrum of the graph Laplacian in order to distinguish between different network models. All our simulation experiments confirm that different model specifications lead to distinguishably different empirical distributions of the spectrum of the graph Laplacian. Although there are no theoretical results in our work that can offer a guarantee for this methodology, we believe our experimental evidence shows merit in this approach and contributes to further its discussion. Our second key idea is to capitalize on the significant advancements in the literature of supervised classification, and delegate to it the job of assessing whether the predicted values of a model correctly fit a set of observations. We believe this approach is warranted given the intrinsic high-dimensional nature of our spectral approach, although our own evidence shows this choice becomes less critical under certain conditions. At its output, our methodology provides a relative measure of fit that ranks how well a set of candidate models describes an observed graph Laplacian. This allows not only to choose the best-fitting model but also to assess its fit among competing alternatives.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1007/s41109-024-00640-4.

Additional file1

Author Contributions

JIPH conceived the idea of the study, created software implementations for it, programmed and analyzed the simulation and application studies, and edited the manuscript. JRS designed the simulation studies, proposed application studies, and drafted the manuscript.

Funding

This work was partially supported by the National Science Foundation (NSF award MMS-2345043), the Department of Defense Test Resource Management Center under contract FA8075-18-D-0002/FA8075-21-F-0074, and Florida State University.

Availability of data

The datasets generated and/or analyzed during the current study along with code used for simulation studies and analyses are available at https://jrstew.github.io/files/HidalgoStewart2024_data_repo.zip.

Declarations

Competing of interests

The authors declare they have no Competing of interests.

Received: 16 February 2024 Accepted: 25 June 2024
Published online: 10 July 2024

References

Airoldi E, Blei D, Fienberg S, Xing E (2008) Mixed membership stochastic blockmodels. J Mach Learn Res 9(65):1981–2014

Anderson CJ, Wasserman S, Faust K (1992) Building stochastic blockmodels. Soc Netw 14(1–2):137–161

Athreya A, Fishkind DE, Tang M, Priebe CE, Park Y, Vogelstein JT, Levin K, Lyzinski V, Qin Y, Sussman DL (2018) Statistical inference on random dot product graphs: a survey. J Mach Learn Res 18(226):1–92

Barabasi A-L, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512

Bickel PJ, Levina E (2004) Some theory for fisher's linear discriminant function, 'naive bayes', and some alternatives when there are many more variables than observations. Bernoulli 10(6):989–1010

Bollobás B, Nikiforov V (2004) Graphs and hermitian matrices: eigenvalue interlacing. Discr Math 289(1–3):119–127

Brouwer AE, Haemers WH (2011) *Spectra of graphs*. Springer Science & Business Media

Cai T, Chen X (2010) *Highdimensional Data Analysis*. Higher Education Press Limited Company

Chen T, Guestrin C (2016) Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp 785–794

Coutinho JA, Diviák T, Bright D, Koskinen J (2020) Multilevel determinants of collaboration between organised criminal groups. Social Netw 63:56–69

Cvetković DM, Doob M, Sachs H (1980) Spectra of graphs: theory and application. Academic Press, New York

de Abreu NMM (2007) Old and new results on algebraic connectivity of graphs. Linear Algeb Appl 423(1):53–73

Donetti L, Neri F, Muñoz MA (2006) Optimal network topologies: expanders, cages, ramanujan graphs, entangled networks and all that. J Statist Mchan Theory Exper 2006(8):P08007

Fiedler M (1973) Algebraic connectivity of graphs. Czechosl Math J 23(2):298–305

Finger K, Lux T (2017) Network formation in the interbank money market: an application of the actor-oriented model. Social Netw 48:237–249

Hand DJ, Yu K (2001) Idiot's Bayes—not so stupid after all? Int Stat Rev 69(3):385–398

Handcock MS, Raftery AE, Tantrum JM (2007) Model-based clustering for social networks. J Royal Stat Soc Ser A 170:301–354

Hastie T, Tibshirani R, Friedman J (2011) The elements of statistical learning, 2nd edn. Springer-Verlag, New York

Ho TK (1995) Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, vol 1, pp 278–282. IEEE

Hoff PD, Raftery AE, Handcock MS (2002) Latent space approaches to social network analysis. J Am Stat Assoc 97(460):1090–1098

Holland PW, Laskey KB, Leinhardt S (1983) Stochastic block models: some first steps. Social Netw 5:109–137

Hoory S, Linial N, Wigderson A (2006) Expander graphs and their applications. Bull Am Math Soc 43(4):439–561

Hunter DR (2007) Curved exponential family models for social networks. Social Netw 29:216–230

Hunter DR, Handcock MS (2006) Inference in curved exponential family models for networks. J Computat Graph Stat 15:565–583

Hunter DR, Goodreau SM, Handcock MS (2008) Goodness of fit of social network models. J Am Stat Assoc 103:248–258

Jackson MO, Watts A (2002) The evolution of social and economic networks. J Econom Theory 106(2):265–295

Krioukov D, Papadopoulos F, Kitsak M, Vahdat A, Boguñá M (2010) Hyperbolic geometry of complex networks. Phys Rev E 82:036106

Krivitsky PN, Handcock MS (2014) *latentnet: Latent position and cluster models for statistical networks*. The Comprehensive R Archive Network

Kwok TC, Lau LC, Lee YT, Oveis Gharan S, Trevisan L (2013) Improved cheeger's inequality: Analysis of spectral partitioning algorithms through higher order spectral gap. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 11–20

Latouche P, Birmelé E, Ambroise C (2014) Model selection in overlapping stochastic block models. Electron J Stat 8(1):762–794

Leger J-B (2016) Blockmodels: A r-package for estimating in latent block model and stochastic block model, with various probability functions, with or without covariates. *arXiv preprint*arXiv:1602.07587

Lei J, Rinaldo A (2015) Consistency of spectral clustering in stochastic block models. Ann Stat 43(1):215–237

Liaw A, Wiener M (2002) Classification and regression by randomforest. R News 2(3):18–22

Loyal JD, Chen Y (2023) A spike-and-slab prior for dimension selection in generalized linear network eigenmodels. *arXiv preprint*arXiv:2309.11654

Lusher D, Koskinen J, Robins G (2013) Exponential random graph models for social networks. Cambridge University Press, Cambridge, UK

Majka M (2019) *naivebayes: high performance implementation of the naive bayes algorithm in R*. The comprehensive r archive network

Morris M (2004) *Network epidemiology: a handbook for survey design and data collection*. Oxford University Press on Demand

Newman MEJ (2006) Finding community structure in networks using the eigenvectors of matrices. Phys Rev E 74(3):036104

Obando C, de Vico Fallani F (2017) A statistical model for brain networks inferred from large-scale electrophysiological signals. J Royal Soc Interface 14(128):20160940

Ryan C, Wyse J, Friel N (2017) Bayesian model selection for the latent position cluster model for social networks. Netw Sci 5(1):70–91

Sampson S (1968) *A novitiate in a period of change: an experimental and case study of relationships*. PhD thesis, Department of Sociology, Cornell University

Schweinberger M (2011) Instability, sensitivity, and degeneracy of discrete exponential families. J Am Stat Assoc 106(496):1361–1370

Schweinberger M, Handcock MS (2015) Local dependence in random graph models: characterization, properties and statistical inference. J Royal Stat Soc Ser B 77:647–676

Schweinberger M, Snijders TA (2003) Settings in social networks: a measurement model. Sociol Methodol 33(1):307–341

Schweinberger M, Stewart J (2020) Concentration and consistency results for canonical and curved exponential-family models of random graphs. Ann Stat 48:374–396

Schweinberger M, Krivitsky PN, Butts CT, Stewart J (2020) Exponential-family models of random graphs: Inference in finite, super, and infinite population scenarios. Stat Sci 35:627–662

Sewell DK, Chen Y (2015) Latent space models for dynamic networks. J Am Stat Assoc 110:1646–1657

Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Patt Anal Mach Intell 22(8):888–905

Shore J, Lubin B (2015) Spectral goodness of fit for network models. Soc Netw 43:16–27

Smith AL, Asta DM, Calder CA (2019) The geometry of continuous latent space models for network data. Stat Sci 34(3):428–453

Snijders TA (2016) The multiple flavours of multilevel issues for networks. In *Multilevel network analysis for the social sciences*, pages 15–46. Springer

Snijders TA, van de Bunt GG, Steglich CE (2010) Introduction to stochastic actor-based models for network dynamics. Soc Netw 32(1):44–60

Snijders TAB, Pattison PE, Robins GL, Handcock MS (2006) New specifications for exponential random graph models. Sociol Methodol 36:99–153

Stewart J, Schweinberger M, Bojanowski M, Morris M (2019) Multilevel network data facilitate statistical inference for curved ERGMs with geometrically weighted terms. Soc Netw 59:98–119

Stivala A, Lomi A (2021) Testing biological network motif significance with exponential random graph models. Appl Netw Sci 6(1):1–27

Wang YXR, Bickel PJ (2017) Likelihood-based model selection for stochastic block models. Ann Stat 45(2):500–528

Yin F, Phillips NE, Butts CT (2019) Selection of exponential-family random graph models via held-out predictive evaluation (hope). *arXiv preprint*arXiv:1908.05873

Zeng R, Sheng QZ, Yao L, Xu T, Xie D (2013) A practical simulation method for social networks. Proc First Austral Web Conf Vol 144:27–34

Publisher's Note