NETWORK CLASSIFICATION WITH APPLICATIONS TO BRAIN CONNECTOMICS[1]

BY JESÚS D. ARROYO RELIÓN[*], DANIEL KESSLER[†], ELIZAVETA LEVINA[†] AND STEPHAN F. TAYLOR[†,2]

Johns Hopkins University* and University of Michigan†

While statistical analysis of a single network has received a lot of attention in recent years, with a focus on social networks, analysis of a sample of networks presents its own challenges which require a different set of analytic tools. Here we study the problem of classification of networks with labeled nodes, motivated by applications in neuroimaging. Brain networks are constructed from imaging data to represent functional connectivity between regions of the brain, and previous work has shown the potential of such networks to distinguish between various brain disorders, giving rise to a network classification problem. Existing approaches tend to either treat all edge weights as a long vector, ignoring the network structure, or focus on graph topology as represented by summary measures while ignoring the edge weights. Our goal is to design a classification method that uses both the individual edge information and the network structure of the data in a computationally efficient way, and that can produce a parsimonious and interpretable representation of differences in brain connectivity patterns between classes. We propose a graph classification method that uses edge weights as predictors but incorporates the network nature of the data via penalties that promote sparsity in the number of nodes, in addition to the usual sparsity penalties that encourage selection of edges. We implement the method via efficient convex optimization and provide a detailed analysis of data from two fMRI studies of schizophrenia.

1. Introduction. Network data analysis has received a lot of attention in recent literature, especially unsupervised analysis of a single network which is thought of as generated from an exchangeable random graph model, for example Bickel and Chen (2009), Gao et al. (2017), Le, Levina and Vershynin (2017), Zhang and Zhou (2016) and many others. This setting is applicable to a number of

real life scenarios, such as social networks, but there are situations where the network nodes are labeled and therefore not exchangeable, and/or more than one network is available for analysis, which have received relatively less attention. Here we focus on the setting motivated by brain connectomics studies, where a sample of networks is available from multiple populations of interest (e.g., mentally ill patients and healthy controls). In this setting, each unit in the population (e.g., a patient) is represented by their own network, and the nodes (brain regions of interest) are labeled and shared across all networks through a registration process that maps all individual brains onto a common atlas. There are many classical statistical inference questions one can ask in this setting, for example, how to compare different populations (Tang et al. (2017b, 2017a)). The question we focus on in this paper is a classification problem: given a training sample of networks with labeled nodes drawn from multiple classes, the goal is to learn the rules for predicting the class of a given network, and just as importantly, interpret these rules.

Network methods are a popular tool in the neuroscience literature (Bullmore and Bassett (2011), Bullmore and Sporns (2009)). A brain network represents connectivity between different locations of an individual's brain. How connectivity is defined varies with the type of imaging technology used and the conditions under which data were collected. In this paper, we focus on functional connectivity, which is a measure of statistical association between each pair of locations in the brain, constructed from functional magnetic resonance imaging (fMRI) data, although the methods we develop are applicable to any sample of weighted networks with labeled nodes. In fMRI studies, BOLD (blood oxygen-level dependent) signal, a known correlate of underlying neural activity, is measured at a sequence of time points at many spatial locations in the brain, known as voxels, resulting in a 4-dimensional data array, with three spatial dimensions and a time index. Brain networks constructed from fMRI data have been successfully used for various tasks, such as differentiating between certain illnesses, or between types of external stimuli (Bullmore and Sporns (2009)), and contain enough information to identify individual subjects (Finn et al. (2015)). Extensive statistical literature has focused on the analysis of raw fMRI data (Lindquist (2008), Zhang et al. (2016), Zhou, Li and Zhu (2013)), usually aiming to characterize brain activation patterns obtained from task-based fMRI experiments. In this paper, we focus on resting-state fMRI data, where no particular task is performed and subjects are free to think about anything they want. Thus registering the time dimension across different subjects is not possible. The connectivity network approach, which averages over the time dimension in computing a measure of dependence between different voxels, is thus a natural choice, and has been widely used with multiple types of neuroimaging data.

Two different datasets are analyzed in this paper, both of resting state fMRI studies on schizophrenic patients and healthy controls. One dataset, COBRE (54 schizophrenics and 70 controls), is publicly available (Aine et al. (2017)); another,
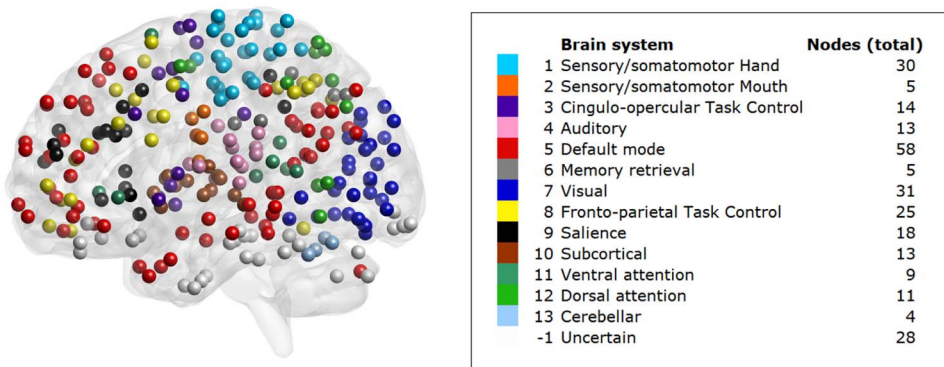
FIG. 1. *Regions of interest* (*ROIs*) *defined by Power et al.* (2011), *colored by brain systems, and the total number of nodes in each system.*

which we will refer to as UMich data (39 schizophrenics and 40 controls), was collected internally in the last author's lab. Having two datasets on the same disease allows us to cross-check models trained on one of them for classification on the other to check the robustness of our approach. The raw data arrays undergo preprocessing and registration steps, discussed in detail in the Supplementary Material (Arroyo Relión et al. (2019)), along with additional details on data collection. To construct a brain network from fMRI measurements, a set of nodes is chosen, typically corresponding to regions of interests (ROIs) from some predefined parcellation. In our analysis we use the parcellation of Power et al. (2011) (see Figure 1), which consists of 264 ROIs divided into 14 functional brain systems (in the CO-BRE data, node 75 is missing). A connectivity measure is then computed for every pair of nodes, resulting in an adjacency matrix of size 264×264. Many choices of connectivity measures are available (Smith et al. (2013)); perhaps the most commonly used one is the Pearson correlation coefficient between locations, computed by averaging over the time dimension. It has been argued that partial correlations are a better measure of connectivity (Varoquaux and Craddock (2013), Narayan, Allen and Tomson (2015)), but the choice depends on the final goal of analysis. In this paper we follow the vast majority of the connectomics literature and measure connectivity on each individual by using marginal correlations between the corresponding time series (see Figure 2). The correlations are then further rank-transformed and standardized; see supplement (Arroyo Relión et al. (2019)) for details. These transformations are intended to deal with subject-to-subject variability and the global signal regression issue (Gotts et al. (2013)), and although they lose some information, we observed that on our datasets classification based on standardized ranks of marginal correlations outperformed classification based on other connectivity measures, such as marginal correlations. The methods we develop here are applicable to networks that encode any type of connectivity measure.
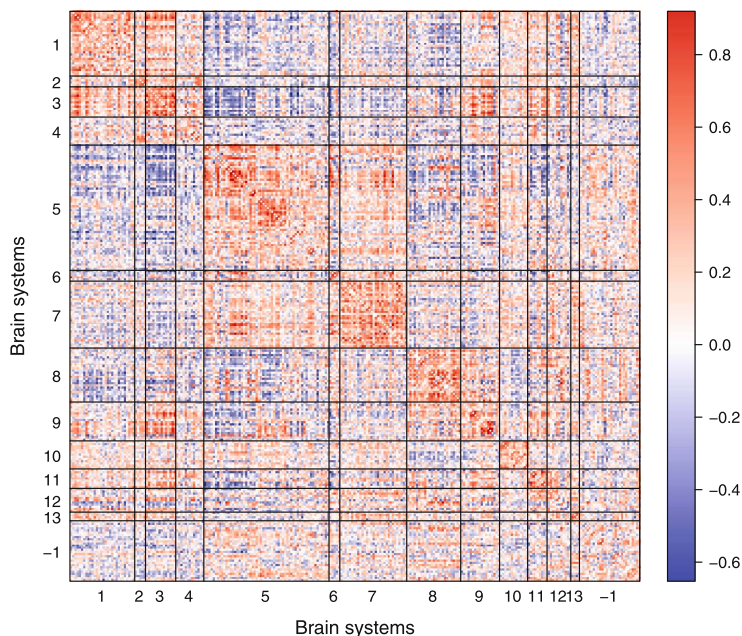
FIG. 2. *Brain network from one of the subjects*, *showing the value of the Fisher z-transformed correlations between the nodes*, *with the* 264 *nodes grouped into* 14 *brain systems*.

The problem of graph classification has been studied previously in other contexts, with a substantial literature motivated by the problem of classification of chemical compounds (Srinivasan et al. (1996), Helma et al. (2001)), where graphs represent the compound's molecular structure. This setting is very different, with small networks of about 20 nodes on average, binary or categorical edges recorded with no noise, and different nodes corresponding to different networks (Ketkar, Holder and Cook (2009)). Classification methods for chemical compounds are usually based on finding certain discriminative patterns in the graphs, like subgraphs or paths (Inokuchi, Washio and Motoda (2000), Gonzalez, Holder and Cook (2000)), and using them as features for training a standard classification method (Deshpande et al. (2005), Fei and Huan (2010), Kudo, Maeda and Matsumoto (2004)). Computationally, finding these patterns is only possible on small binary networks.

Another type of methods are based on graph kernels (Gärtner, Flach and Wrobel (2003), Vishwanathan et al. (2010)), which define a similarity measure between two networks. These kernels combined with support vector machines (SVMs) have been successfully used on small networks (Borgwardt et al. (2005), Kashima, Tsuda and Inokuchi (2003)), but the curse of dimensionality makes local kernel methods unsuitable for large scale networks (Bengio and Monperrus (2005)). On our datasets, graph kernel methods did not perform better than random guessing.

In the context of classifying large-scale brain networks, two main approaches have been followed. One approach is to reduce the network to its global summary measures such as the average degree, clustering coefficient, or average path length (Bullmore and Sporns (2009)), and use those measures as features for training a classification method. Previous studies have reported significant differences on some of these network measures for groups of patients with certain brain diseases compared with healthy controls (Supekar et al. (2008), Liu et al. (2008)), suggesting their usefulness as diagnostic biomarkers. However, global summary statistics collapse all local network information, which can harm the accuracy of classification and does not allow to identify local differences. In our data analysis, a method based on the network measures suggested in Prasad et al. (2015) performed poorly for classification (see Section 6).

An alternative approach to classification of large networks is to treat edge weights as a "bag of features," vectorizing the unique elements of the adjacency matrix and ignoring the network nature of the data. This approach can leverage many existing classification methods for vectors, and provides an interpretation at the edge level if variable selection is applied (Craddock et al. (2009), Richiardi et al. (2011), Zhang et al. (2012)). Spatial correlation between edges connecting neighboring nodes can be incorporated (Scott et al. (2015), Watanabe et al. (2014)), although the effectiveness of this regularization will depend on the parcellation used to define nodes. Alternatively, an individual test can be used for each edge to find significant differences between two populations, with a multiple testing correction and without constructing a classifier at all (Narayan, Allen and Tomson (2015)). While these methods can deliver good predictions, their interpretability is limited to individual edge selection, which is less scientifically interesting than identifying differentiating nodes or regions, and they cannot account for network structure.

Taking the network structure into account can have benefits for both testing and classification settings. Some methods perform inference over groups of edges based on the community assignments of the nodes to which they are incident. For example, Sripada et al. (2014a, 2014b) introduced *Network Contingency Analysis* which begins with massive univariate testing at each edge, and then counts the number of superthreshold connections in each *cell*, a group of edges that connect nodes in two functional systems. Nonparametric methods are then used to conduct inference on the count statistic for each cell, with multiple comparison correction for inference at the cell level. Power can be improved by applying a network-based multiple testing dependence correction (Zalesky, Fornito and Bullmore (2010)). For classification, better interpretability and potentially accuracy can be obtained if we focus on understanding which brain regions or interactions between them are responsible for the differences. In somewhat related work, Vogelstein et al. (2013) proposed to look for a minimal set of nodes which best explains the difference, though that requires solving a combinatorial problem. Hypothesis testing on a type of graph average has also been proposed (Ginestet et al. (2017)). Bayesian

nonparametrics approaches for modeling populations of networks allow to test for local edge differences between the groups (Durante and Dunson (2018)), but are computationally feasible only for small networks.

Our goal in this paper is to develop a high-dimensional network classifier that uses all the individual edge weights but also respects the network structure of the data and produces more interpretable results. To achieve this goal, we use structured sparsity penalties to incorporate the network information by penalizing both the number of edges and the number of nodes selected. Although our main application here is classification of brain connectivity networks, our methods are applicable to any weighted graphs with labeled nodes, and to general prediction problems, not just classification.

The rest of this paper is organized as follows. In Section 2, we introduce our classifier and the structured penalties. In Section 3 we show how to efficiently solve the resulting convex optimization problem by a proximal algorithm, each step of which is a further optimization problem which we solve by the alternating direction method of multipliers (ADMM). The performance of our method is evaluated and compared with other methods using simulations in Section 5. In Section 6, we analyze two brain connectivity datasets, each containing schizophrenic patients and healthy controls, and show that our regularization framework leads to state-of-the-art accuracy while providing interpretable results, some of which are consistent with previous findings and some are new. We conclude with a brief discussion in Section 7.

2. A framework for node selection in graph classification.

2.1. *A penalized graph classification approach.* We start from setting up notation. All graphs we consider are defined on the same set of N labeled nodes. A graph can be represented with its adjacency matrix $A \in \mathbb{R}^{N \times N}$. We focus on graphs that are undirected ($A_{ij} = A_{ji}$) and contain no self-loops ($A_{ii} = 0$). These assumptions are not required for the derivations below, but they match the neuroimaging setting and simplify notation. Our goal is predicting a class label Y from the graph adjacency matrix A; in this paper we focus on the binary classification problem where Y takes values $\{-1, 1\}$, although extensions from binary to multi-class classification or real-valued responses are straightforward. Throughout this paper, we use $\| \cdot \|_p$ to denote the entry-wise ℓ_p norm, that is, for a matrix $A \in \mathbb{R}^{m \times n}$, $\|A\|_p = (\sum_{i=1}^{m} \sum_{j=1}^{n} |A_{ij}|^p)^{1/p}$.

A standard general approach is to construct a linear classifier, which predicts the response Y from a linear combination of the elements of A, $\langle A, B \rangle = \text{Tr}(B^T A)$, where we arrange the coefficients in a matrix $B \in \mathbb{R}^{N \times N}$ to emphasize the network nature of the predictors. We focus on linear classifiers here because variable selection is at least as important as prediction itself in the neuroimaging application, and setting coefficients to 0 is a natural way to achieve this. The coefficients are typically estimated from training data by minimizing an objective consisting of a

loss function plus a penalty. The penalty can be used to regularize the problem to make the estimator well defined in high-dimensional problems, to select important predictors, and to impose structure, and many such penalties have been proposed, starting from the lasso (Tibshirani (1996)). Our focus is on designing a classifier in this framework that respects and utilizes the network nature of the predictors. In brain networks in particular, neuroscientists believe that edges are organized in subnetworks, also called brain systems (Power et al. (2011)), that carry out specific functions, and certain subnetworks are important for prediction (Bullmore and Sporns (2009)), although different studies tend to implicate different regions (Fornito et al. (2012)). Thus we aim to find nodes or subnetworks with good discriminative power, and hence select both the most informative nodes and edges.

Although the methods we develop here can be used on small networks, our main focus is on the more challenging case of medium to large brain networks. In brain connectivity studies dealing with multiple subjects, while raw images may contain hundreds of thousands of voxels, they are commonly down-sampled according to a parcellation scheme with a coarser resolution, usually resulting in networks with hundreds or thousands of nodes representing ROIs (see, e.g., Kiar et al. (2018)). This coarser resolution is essential for registration, as aligning different brains at a high resolution is much harder, but it still results in hundreds of thousands or millions of edges which serve as predictor variables. Given the typical data sizes in this area of application, we focus on methods based on convex formulations, which allow for efficient and scalable implementations with convergence guarantees.

Let $\{(A^{(1)}, Y_1), \ldots, (A^{(n)}, Y_n)\}$ be the training sample of undirected adjacency matrices with their class labels, and let $Y = (Y_1, \ldots, Y_n)$. A generic linear classifier described above is computed by finding the coefficients B defined by

$$(2.1) \qquad \hat{B} = \underset{B \in \mathcal{B}}{\arg\min}\{\ell(B) + \Omega(B)\},$$

where $\mathcal{B} = \{B \in \mathbb{R}^{N \times N} : B = B^T, \operatorname{diag}(B) = 0\}$, Ω is a penalty, and

$$\ell(B) = \frac{1}{n} \sum_{k=1}^{n} \tilde{\ell}(Y_k, A^{(k)}; B)$$

is a loss function evaluated on the training data. Our methodology can accommodate different choices of loss functions that can extend beyond classification problems (e.g., least squares or generalized linear models). The optimization algorithm presented in Section 3 can work with any convex and continuously differentiable loss function, and further assumptions are required for consistency (see Section 4). In this paper, for the purpose of classification we use the logistic loss function in the simulations and data analysis, which is defined as

$$\tilde{\ell}(Y_k, A^{(k)}; B, b) = \log\{1 + \exp(-Y_k \langle A^{(k)}, B \rangle + b)\}.$$

The threshold b is an additional parameter to be estimated.

To capture structural assumptions on important predictive edges, we focus on convex structured sparsity penalties (Bach et al. (2012)) that encourage a small number of active nodes, by which we mean nodes attached to at least one edge with a nonzero coefficient. One approach to finding a set of such nodes was proposed by Vogelstein et al. (2013), who called it a signal-subgraph, and proposed finding the minimal set of nodes (called signal vertices) which together are incident to all selected edges (but not every node connected to a selected edge is a signal vertex). Finding this set is a combinatorial optimization problem, and the set is not always uniquely defined. Instead, we focus on convex formulations that allow for efficient computation and encourage small active node sets indirectly.

Other convex penalties have been used for fMRI data as a way to enforce spatial smoothness in the solution (Grosenick et al. (2013), Hu and Allen (2015), Xin et al. (2014)). These methods assume that voxels are equally spaced in the brain, and neighboring voxels are highly correlated. In particular, Watanabe et al. (2014) proposed penalties for brain network classification using these spatial assumptions. Here, instead of enforcing a spatial regularization directly, we aim for a regularization that can be applied to any type of network data, and in particular to brain networks with coarse and/or uneven parcellations where enforcing spatial smoothness may not work as well. In any case, the flexibility of convex optimization algorithms allows one to easily incorporate additional spatially-informed penalties if needed.

2.2. *Selecting nodes and edges through group lasso.* To reflect the network structure of the predictors, we use a penalty that promotes a sparse classifier not only in the number of edges used, but also in the number of nodes. The group lasso penalty (Yuan and Lin (2006)) is designed to eliminate a group of variables simultaneously. Here we penalize the number of active nodes by treating all edges connected to one node as a group. Then eliminating this group (a row of coefficients in the matrix B) is equivalent to de-activating a node. The group penalty is defined as

$$(2.2) \qquad \Omega_{\lambda,\rho}(B) = \lambda \left(\sum_{i=1}^{N} \|B_{(i)}\|_2 + \rho \|B\|_1 \right),$$

where $B_{(i)}$ denotes the vector of edge weights incident to the ith node (or equivalently, the ith row or column of B), and $\lambda, \rho \geq 0$ are tuning parameters. Note that the constraint $B = B^T$ makes the groups overlap, since a coefficient B_{ij} belongs to groups associated with the nodes i and j, and therefore, the edge between nodes i and j would be selected only if both nodes are activated. The second term in the penalty $\rho \|B\|_1$ acts as the usual lasso penalty to promote sparsity inside the group (Friedman, Hastie and Tibshirani (2010a)), allowing to select a subset of edges for an active node. Due to the overlap in the groups, this lasso penalty is usually necessary in order to produce sparse solution (see Proposition 1). The constraint $\text{diag}(B) = 0$ in (2.1) is automatically enforced with this formulation.

REMARK 1.   An alternative to the constraint in the problem (2.1) is to optimize over the set

$$\tilde{\mathcal{B}} = \{B \in \mathbb{R}^{N \times N}, \text{diag}(B) = 0\}.$$

Without the symmetry constraint and assuming undirected graphs, the penalty (2.2) is equivalent to the overlapping group lasso formulation of Jacob, Obozinski and Vert (2009). This formulation has some advantages. Since it gives group lasso without overlaps, the lasso penalty $\rho \|B\|_1$ is not required to obtain sparse solutions, and more efficient optimization algorithms exist for this case. This approach would loosely correspond to the idea of selecting signal nodes as in Vogelstein et al. (2013), in the sense that an edge can be selected if at least one of its nodes is selected, and the second node could be inactive. The downside is that each edge now corresponds to two different coefficients B_{ij} and B_{ji}, the problem encountered by all variable selection methods that ignore symmetry, such as Meinshausen and Bühlmann (2006). The standard solution for this problem, as suggested by Jacob, Obozinski and Vert (2009), is to take the average of the coefficients. Intuitively, one would expect that the formulation using $\mathcal{B}$ would be better when the significant edges are incident to a small set of nodes, since both nodes have to be active for an edge to be selected, while using $\tilde{\mathcal{B}}$ may be better when for some nodes most of their edges are significant, creating "significant hubs." Since in our application we are primarily looking for discriminative brain subnetworks, we focus on the symmetrically constrained formulation for the rest of the paper. We also found that in practice this second formulation results in less accurate classifiers for the neuroimaging data discussed in Section 6.

REMARK 2.   The analogue to (2.2) for directed graphs would assign coefficients B_{ij} and B_{ij} to the same group, resulting in the penalty

$$(2.3) \qquad \Psi_{\lambda,\rho}(B) = \lambda \left(\sum_{i=1}^{N} \sqrt{\sum_{j}(B_{ij}^2 + B_{ji}^2)} + \rho \|B\|_1 \right),$$

where $B \in \tilde{\mathcal{B}}$. Alternatively, we can also use the formulation of Remark 1, by replicating the variables and estimating two matrices of coefficients, say $B^{(1)}$ and $B^{(2)}$, with the penalty

$$\tilde{\Psi}_{\lambda,\rho}(B^{(1)}, B^{(2)}) = \lambda \left[\sum_{i=1}^{N} \sqrt{\sum_{j}\{(B_{ij}^{(1)})^2 + (B_{ji}^{(2)})^2\}} + \rho(\|B^{(1)}\|_1 + \|B^{(2)}\|_1) \right],$$

with $B^{(1)}, B^{(2)} \in \tilde{\mathcal{B}}$, and set the coefficients matrix to $B = (B^{(1)} + B^{(2)})/2$. This formulation will again not directly select subnetworks as discussed in Remark 1.

Finally, for numerical stability we add an extra ridge penalty term $\frac{\gamma}{2}\|B\|_F^2 = \frac{\gamma}{2} \text{Tr}(B^T B)$, with γ a small and fixed constant. There are several benefits of combining ridge and lasso penalties (see, e.g., Zou and Hastie (2005)).

The parameter γ can be potentially considered as an additional tuning parameter, but here we only use a small fixed constant γ in order to avoid numerically degenerate solutions. In practice, the results are not sensitive to the exact value of γ.

Putting everything together, to fit our graph classifier, we solve the problem

$$(2.4) \quad (\hat{B}, \hat{b}) = \underset{B \in \mathcal{B}, b \in \mathbb{R}}{\arg\min} \left\{ \frac{1}{n} \sum_{k=1}^{n} \log\big(1 + \exp(-Y_k \langle B, A^{(k)} \rangle + b)\big) + \frac{\gamma}{2} \|B\|_F^2 \right.$$
$$\left. + \lambda \left(\sum_{i=1}^{N} \|B_i\|_2 + \rho \|B\|_1 \right) \right\}$$

for given values of λ and ρ, which will be chosen by cross-validation.

3. The optimization algorithm.
Our optimization algorithm to solve the problem (2.4) combines two common approaches to convex optimization: proximal algorithms and alternating direction method of multipliers (ADMM). We use an accelerated version of the proximal algorithm (Beck and Teboulle (2009)) to solve the main problem (2.4). In each step, we need to calculate a proximal operator, which is a further convex optimization problem solved with the ADMM algorithm.

The main optimization difficulty comes from the overlapping groups. Some algorithms have been proposed for this case, including a subgradient descent method (Duchi and Singer (2009)), which has a slow rate of convergence, or proximal algorithms based on smoothing the original problem (Chen et al. (2012), Yuan, Liu and Ye (2011)). Although smoothing yields fast algorithms, it is not clear that the sparsity pattern is preserved with those approximations. We follow an approach similar to Yuan, Liu and Ye (2011) and Chen et al. (2012), but solve the proximal operator for the penalty (2.2) directly using the ADMM method. This can potentially give a more accurate sparsity pattern, and the flexibility of the algorithm allows for additional penalties if desired, such as spatial smoothing similar to Watanabe et al. (2014) (see Remark 3).

The main problem (2.1) is solved with a proximal algorithm (see Parikh and Boyd (2013)). Recall that the proximal operator for a function f is defined as $\text{prox}_f(v) = \arg\min_x \{ f(x) + \frac{1}{2}\|x - v\|_2^2 \}$. Starting with an initial value $B^{(0)} \in \mathbb{R}^{N \times N}$, a proximal algorithm solves the optimization problem (2.1) by iteratively calculating the proximal operator of $\Omega = \Omega_{\lambda, \rho}$ for a descent direction of the differentiable loss function ℓ. We use an accelerated version of the algorithm (Beck and Teboulle (2009)), which for each $k = 2, \ldots$, until convergence, performs the updates

$$(3.1) \quad W^{(k)} = B^{(k-1)} + \frac{k-1}{k+2}\big(B^{(k-1)} - B^{(k-2)}\big),$$

$$B^{(k)} = \underset{t^{(k)}\Omega}{\text{prox}} \{ W^{(k)} - t^{(k)} \nabla \ell(W^{(k)}) \}$$

(3.2)

$$= \underset{B \in \mathcal{B}}{\arg\min} \left\{ \frac{1}{2} \| B - (W^{(k)} - t^{(k)} \nabla \ell(W^{(k)})) \|_2^2 + t^{(k)} \Omega(B) \right\},$$

where $\nabla \ell(W) \in \mathbb{R}^{N \times N}$ is the gradient of the loss function ℓ at W and $\{t^{(k)}\}$ is a sequence of positive values. If $\nabla \ell$ is Lipschitz continuous, with L its Lipschitz constant, the sequence of values $\ell(B^{(k)}) + \Omega(B^{(k)})$ converges to the optimal value at rate $O(1/k^2)$ if $t^{(k)} \in [0, 1/L)$. The value of $t^{(k)}$ can be chosen using a backtracking search (Beck and Teboulle (2009)), which decreases this value until the condition

$$(3.3) \qquad \ell(B^{(k)}) \leq \ell(W^{(k)}) + \langle \nabla \ell(W^{(k)}), B^{(k)} - W^{(k)} \rangle + \frac{1}{2t^{(k)}} \| B^{(k)} - W^{(k)} \|_2^2$$

is satisfied. This procedure ensures that step sizes $\{t^{(k)}\}$ become smaller as the algorithm progresses, until $t^{(k)} < 1/L$. In practice, L might be large, which can make the algorithm slow to converge. It has been observed in other sparse high-dimensional problems that search strategies for $t^{(k)}$ which allow for $t^{(k)} > 1/L$ when appropriate can actually speed up convergence (Scheinberg, Goldfarb and Bai (2014), Hastie, Tibshirani and Wainwright (2015)). We use a strategy of this type, allowing $t^{(k)}$ to increase by a factor of $\alpha \geq 1$ if the relative improvement in the loss function on iteration k becomes small. We observed that this strategy significantly reduces the number of iterations until convergence. The entire procedure is summarized in Algorithm 1 of the Supplementary Material (Arroyo Relión et al. (2019)).

The logistic loss function of (2.4) has an extra parameter b. Rather than including it as an unpenalized coefficient for a constant covariate, we use block coordinate descent and solve for b separately. This is convenient because the threshold b and the matrix of coefficients B may not be on the same scale. Thus, b can be updated by solving $b^{(k+1)} = \arg\min_{b \in \mathbb{R}} \ell(B^{(k)}, b)$, which is easy to compute via Newton's method.

The proximal algorithm requires solving the proximal operator (3.2), which has no closed form solution for the penalty (2.2) under the symmetry constraint. Strategies based on smoothing this penalty have been proposed (Chen et al. (2012), Yuan, Liu and Ye (2011)). However, to allow for variable selection which results from nondifferentiability of the penalty, we aim to solve the proximal operator directly using ADMM (see Boyd et al. (2011) for a review). Note that if the symmetric constraint is relaxed as in Remark 1, the proximal operator has a closed form solution (see Remark 4).

The ADMM works by introducing additional constraints and performing coordinate descent in the corresponding augmented Lagrangian function. Setting $Z = W^{(k)} - t^{(k)} \nabla \ell(W^{(k)})$ and $t = t^{(k)}$, and introducing the variables $Q, R \in \mathbb{R}^{N \times N}$,

we can formulate (3.2) as a convex optimization problem

$$
(3.4) \quad \min_{\tilde{B}, Q, R} \quad \frac{1}{2} \|\tilde{B} - Z\|_2^2 + t\lambda \left(\sum_{i=1}^{N} \|Q_{(i)}\|_2 + \rho \|R\|_1 \right)
$$

subject to $\tilde{B} = Q$, $\quad \tilde{B} = R$, $\quad \tilde{B} = \tilde{B}^T$, $\quad \mathrm{diag}(\tilde{B}) = 0$.

The ADMM algorithm introduces the multipliers $U, V \in \mathbb{R}^{N \times N}$ and a penalty parameter $\mu > 0$ to perform gradient descent on the Lagrangian of (3.4), given by $\mathcal{L}_\mu = \mathcal{L}_\mu(\tilde{B}, Q, R, U, V)$ as

$$
(3.5) \quad
\begin{aligned}
\mathcal{L}_\mu = {} & \frac{1}{2} \|\tilde{B} - Z\|_2^2 + t\lambda \left(\sum_{i=1}^{N} \|Q_{(i)}\|_2 + \rho \|R\|_1 \right) + \langle U, \tilde{B} - Q \rangle \\
& + \langle V, \tilde{B} - R \rangle \frac{\mu}{2} (\|\tilde{B} - Q\|_2^2 + \|\tilde{B} - R\|_2^2 + \|\tilde{B} - \tilde{B}^T\|_2^2).
\end{aligned}
$$

The value μ controls the gap between dual and primal feasibility. In practice, we observed that setting $\mu = 0.1$ gives a good balance between primal and dual feasibility, although other self-tuning methods are available (Parikh and Boyd (2013)). This function is optimized by coordinate descent, with each variable updated to minimize the value of $\mathcal{L}_\mu$ while all the other variables are fixed. This update has a closed form; the detailed steps of the ADMM are shown in Algorithm 2 of the Supplementary Material (Arroyo Relión et al. (2019)). These steps are performed until the algorithm converges within tolerance $\epsilon^{\mathrm{ADMM}} > 0$. Note that ADMM will be performed in each iteration of the algorithm to solve (2.4) and thus tolerance ϵ^{ADMM} can be decreased as the algorithm progresses. On the other hand, performing only one iteration of the algorithm gives a similar algorithm to the one of Chen et al. (2012).

REMARK 3. The ADMM makes it very easy to incorporate additional penalties. If Ψ is a new penalty, we can rewrite (3.4) by introducing an additional parameter $\tilde{Q}$ so it becomes

$$
\min_{\tilde{B}, Q, \tilde{Q}, R} \quad \frac{1}{2} \|\tilde{B} - Z^{(k)}\|_2^2 + t\lambda \left(\sum_{i=1}^{N} \|Q_{(i)}\|_2 + \rho \|R\|_1 \right) + t\Psi(\tilde{Q})
$$

subject to $\tilde{B} = Q$, $\quad \tilde{B} = \tilde{Q}$, $\quad \tilde{B} = R$, $\quad \tilde{B} = \tilde{B}^T$, $\quad \mathrm{diag}(\tilde{B}) = 0$.

We can obtain the Lagrangian formulation (3.5) in a similar manner, and include new parameters in the ADMM updates, which can be performed efficiently as long as the proximal operator of Ψ has a closed form solution. This is in fact the case for some other penalties of interest, such as the GraphNet penalty (Grosenick et al. (2013), Watanabe et al. (2014)), which can incorporate spatial location information.

REMARK 4. The alternative formulation for the graph penalty given in Remark 1 corresponds to standard sparse group lasso (Friedman, Hastie and Tibshirani (2010a)). In particular, we can still employ the proximal algorithms (3.1) and (3.2), but instead optimize over the set $\tilde{\mathcal{B}}$. Without the symmetric constraint on B, the overlap in the group lasso penalty disappears, and this vastly simplifies the problem. Using Theorem 1 of Yuan, Liu and Ye (2011), the update for $B^{(k)}$ has a closed form solution given by

$$(3.6) \qquad Y^{(k)} = B^{(k-1)} + \frac{k-2}{k}(B^{(k-1)} - B^{(k-2)}),$$

$$(3.7) \qquad Z_{ij}^{(k)} = \left(1 - \frac{\lambda\rho}{\|Y_{ij}^{(k)} - t_k\nabla_{ij}\ell(Y^{(k)})\|_2}\right)_+ (Y_{ij}^{(k)} - t_k\nabla_{ij}\ell(Y^{(k)})),$$

$$(3.8) \qquad B_{(i)}^{(k)} = \left(1 - \frac{\lambda}{\|Z_{(i)}^{(k)}\|_2}\right)_+ (Z_{(i)}^{(k)}), \quad i \in [N].$$

4. Theory. In this section, we show that the solution of the penalized problem (2.2) can recover the correct subgraph corresponding to the set of nonzero coefficients, and give its rates of convergence. The theory is a consequence of the results of Lee, Sun and Taylor (2015) for establishing model selection consistency of regularized M-estimators under geometric decomposability (see Supplementary Material (Arroyo Relión et al. (2019))). We present explicit conditions for our penalty to work well, which depend on the data as well as the tuning parameters.

Let $B^\star \subset \mathbb{R}^{N \times N}$ be the unknown parameter we seek to estimate, and we assume there is a set of active nodes $\mathcal{G} \subset [N]$ with $|\mathcal{G}| = G$, so that $B_{ij}^\star = 0$ if $i \in \mathcal{G}^C$ or $j \in \mathcal{G}^C$. We allow some edge weights inside the subgraph defined by $\mathcal{G}$ to be zero, but we focus on whether the set $\mathcal{G}$ is correctly estimated by the set $\hat{\mathcal{G}}$ of active nodes in $\hat{B}$. Denote by $\mathcal{M} \subseteq \mathbb{R}^{N \times N}$ the set of matrices where the only nonzero coefficients appear in the active subgraph, that is,

$$(4.1) \qquad \mathcal{M} = \{B \in \mathbb{R}^{N \times N} | B_{ij} = 0 \text{ for all } i \in \mathcal{G}^C \text{ or } j \in \mathcal{G}^C, B = B^T\}.$$

There are two main assumptions on the loss function ℓ required for consistent selection in high-dimensional models (Lee, Sun and Taylor (2015)). The first assumption is on the convexity of the loss function around $B^\star$, while the second assumption bounds the size of the entries in the loss Hessian between the variables in the active subgraph and the rest. Let the loss Hessian $\nabla^2\ell(B^\star) \in \mathbb{R}^{N \times N} \otimes \mathbb{R}^{N \times N}$ be defined by

$$\nabla_{(i,j),(k,l)}^2 \ell(B) = \frac{\partial^2 \ell(B)}{\partial B_{ij} \partial B_{kl}},$$

and define the matrix $H_{(i,j),\mathcal{G}} \in \mathbb{R}^{G \times G}$ with $(i,j) \in (\mathcal{G} \times \mathcal{G})^C$ such that

$$(4.2) \qquad (H_{(i,j),\mathcal{G}})_{k,l} = \mathrm{Tr}\{(\nabla^2_{(i,j),(\mathcal{G},\mathcal{G})}\ell(B^\star))\Lambda_{(k,l),(\cdot,\cdot)}\}, \quad 1 \le k,l \le G,$$

where $\Lambda \in \mathbb{R}^{G \times G} \otimes \mathbb{R}^{G \times G}$ is a tensor such that $\mathrm{Mat}(\Lambda)$ is a pseudoinverse of $\mathrm{Mat}(\nabla^2_{(\mathcal{G},\mathcal{G}),(\mathcal{G},\mathcal{G})}\ell(B^\star))$, and Mat is the operation that unfolds the entries of a tensor Λ into a $G^2 \times G^2$ matrix. The matrix $H_{(i,j),\mathcal{G}}$ measures how well the variable corresponding to the edge (i,j) can be represented by the variables in the active subgraph.

ASSUMPTION 1 (Restricted Strong Convexity). There exists a set $C \subset \mathbb{R}^{N \times N}$ with $B^\star \in C$, and constants $m > 0$, $\tilde{L} < \infty$ such that

$$\sum_{i,j} \Delta_{i,j} \mathrm{Tr}\{(\nabla^2_{(i,j),(\cdot,\cdot)}\ell(B))\Delta\} \ge m\|\Delta\|_2^2 \quad \forall B \in C \cap \mathcal{M}, \Delta \in C \cap \mathcal{M},$$

$$\|\nabla^2 \ell(B) - \nabla^2 \ell(B^\star)\|_2 \le \tilde{L}\|B - B^\star\|_2 \quad \forall B \in C.$$

ASSUMPTION 2 (Irrepresentability). There exists a constant $0 < \tau < 1$ such that

$$\max_{i \in \mathcal{G}^C}\left\|\left(\sum_{k=1}^{G}\|(H_{(i,j),\mathcal{G}})_{k\cdot}\|_2\right)_{j=1}^{N}\right\|_2 = 1 - \tau < 0.$$

This version of the irrepresentability condition corresponds to the one usually employed in group lasso penalties (Bach (2008)), but as we will see later, due to overlaps in the groups it further requires a lower bound on ρ to work for model selection.

The first two assumptions are stated directly as a function of the loss for a fixed design case, but they can be substituted with bounds in probability for the case of random designs. In order to obtain rates of convergence, we do require a distributional assumption on the first derivative of the loss. This assumption can be substituted with a bound on $\max_i \|\nabla\ell(B^\star)_{(i)}\|_2$.

ASSUMPTION 3 (Sub-Gaussian score function). Each pair in the sample (A, Y) is independent and comes from a distribution such that the entries of the matrix $\nabla\tilde{\ell}(Y, A; B^\star)$ are subgaussian. That is, for all $t > 0$ there is a constant $\sigma^2 > 0$ such that

$$\max_{i,j}\mathbb{P}(\|\nabla_{ij}\tilde{\ell}(Y, A; B^\star)\|_\infty > t) \le 2\exp(-t^2/\sigma^2).$$

With these assumptions, we establish consistency and correct model selection. The proof is given in the Supplementary Material (Arroyo Relión et al. (2019)).

PROPOSITION 1. *Suppose Assumptions 1 and 3 hold.*

(a) *Setting the penalty parameters as $\lambda = c_1 \sigma \sqrt{\frac{\log N}{n}} \min\{\frac{\sqrt{N}}{1+\rho}, \frac{1}{\rho}\}$ and $\rho \geq 0$ for some constant $c_1 > 0$, with probability at least $1 - 2/N$ the optimal solution of* (2.4) *is unique and satisfies*

$$\|\hat{B} - B^\star\|_2 = O_P\left(\sigma N \sqrt{\frac{\log N}{n}}\right). \tag{4.3}$$

(b) *Suppose Assumption 2 also holds. If $n > c_2 G^2 \sigma^2 \log N$ for a constant $c_2 > 0$, setting the penalty parameters as $\lambda = c_3 \sigma \sqrt{\frac{\log N}{n}} \min\{\frac{\sqrt{N}}{1+\rho}, \frac{1}{\rho}\}$ for some constant $c_3 > 0$, and*

$$\rho > \frac{1}{\tau} - \frac{1}{\sqrt{G}}, \tag{4.4}$$

then

$$\|\hat{B} - B^\star\|_2 = O_P\left(\sigma G \sqrt{\frac{\log N}{n}}\right), \tag{4.5}$$

$$\mathbb{P}(\hat{\mathcal{G}} \subseteq \mathcal{G}) = 1 - 2/N. \tag{4.6}$$

The part of the penalty associated with ρ causes the solution to be sparse. Due to the overlap in the groups, a small value of ρ will usually not result in zeros in the solution of the problem (2.4). The lower bound on ρ in (4.4) ensures that the irrepresentability condition of Lee, Sun and Taylor (2015) holds (see Lemma 2 in the supplement (Arroyo Relión et al. (2019))).

Proposition 1 ensures that, with high probability, all edges estimated to have nonzero weights are contained in the active subgraph, and quantifies the error in estimating the entries of B^*. To ensure that all active nodes are recovered, at least one edge corresponding to each active node needs to have a nonzero weight. A similar result can be obtained to guarantee recovery of all active nodes under a stronger assumption that the magnitude of the nonzero entries of $B^\star$ is bounded below by $|B_{ij}^\star| > c_4 G^2 \lambda$ for a constant c_4. The condition in part (b) requires a sample size that grows faster than the size of the active subgraph, which in practice can be much smaller than the size of the network, making the method suitable for our applications in which the sample size is limited and the number of nodes is large.

5. Numerical results on simulated networks. In this section, we evaluate the performance of our method using synthetic networks. We are interested in assessing the ability of the method to correctly identify predictive edges, its classification accuracy, and comparisons to benchmarks. We compare the different methods' edge selection performance in simulations using area under the curve (AUC). Brain

connectomic networks are characterized by organization of nodes into communities (Bullmore and Sporns (2009)), in which nodes within the same community tend to have stronger connections than nodes belonging to different communities. In order to generate synthetic networks that mimic this property, we introduce community structure using the stochastic block model (SBM) (Holland, Laskey and Leinhardt (1983)). Before generating edges, we assign each node a community label, C_i, where $C_i \in [K]$ for each $i \in [N]$. The node assignments are the same for all networks in the population. Given the community labels, network edges are generated independently from a distribution that only depends on the community labels of the nodes associated with each edge. Since fMRI networks are real-valued networks, we generate edge weights from a Gaussian distribution, rather than the standard Bernoulli distribution normally used with the SBM. Specifically, we draw each A_{ij} independently from $N(\mu_{C_i C_j}, \sigma^2)$, with $\mu \in \mathbb{R}^{K \times K}$ defined by

$$\mu_{kl} = \begin{cases} 0.3 & \text{if } k = l, \\ 0.1 & \text{if } k \neq l, \end{cases}$$

and $\sigma^2 = 0.18$. These values were chosen to approximately match the distribution of edge weights in our datasets (see Section 6). We set the number of nodes $N = 300$, with $K = 12$ communities of size 25 each. We work with undirected networks, so the adjacency matrices are symmetric, with 44,850 distinct edges. Although our method is able to scale to larger networks, this moderately sized setting is already highly computationally demanding for many of the comparison benchmarks.

To set up two different class distributions, we select a set of active nodes $\mathcal{G}$ first, defined by the nodes corresponding to some communities selected at random. Then, we alter a set of differentiating edges $\mathcal{E}$ selected at random from $\mathcal{G} \times \mathcal{G}$ with probability p. For each edge $(i, j) \in \mathcal{E}$, the distribution in class $Y = -1$ is $N(\mu_{C_i C_j}, \sigma^2)$, while the distribution in class $Y = 1$ is changed to $N(0.2, \sigma^2)$. Figure 3 shows example expected adjacency matrices for each class. We then generate 50 networks from each class, resulting in a sample size of $n = 100$. We vary $G = |\mathcal{G}|$ by changing the number of communities selected, and the value of p, to study the effect of the number of active nodes and the density of differentiating edges inside a subgraph. The number of communities selected is set to 1 ($|\mathcal{G}| = 25$), 2 ($|\mathcal{G}| = 50$) and all communities ($|\mathcal{G}| = 300$); note that in the last scenario all nodes are active and hence the node structure is not informative at all.

Since we are interested in identifying predictive edges and nodes, we use the AUC of the receiver operating characteristic (ROC) curve, for both edge and node selection. For each method, we calculate the ROC curve by changing its corresponding sparsity parameter to vary the number of edges selected. For a selection method $\mathcal{M}$ and a sparsity parameter η let $\hat{\mathcal{E}}(\mathcal{M}, \eta)$ be the set of edges selected by $\mathcal{M}$, and $\hat{\mathcal{G}}(\mathcal{M}, \eta)$ the set of active nodes corresponding to $\hat{\mathcal{E}}(\mathcal{M}, \eta)$. We calculate
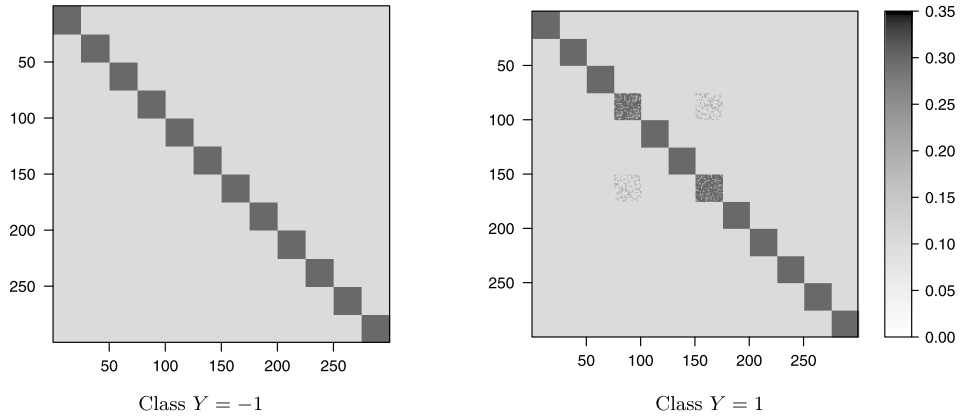
FIG. 3. *Expected adjacency matrices for each class. There are* 50 *active nodes* $\mathcal{G}$ *on communities 4 and 7, and edge weights on* 25% *of the edges within* $\mathcal{G} \times \mathcal{G}$ *have been altered for the second class of networks* ($Y = 1$).

the edge false positive rate (EFPR) and edge true positive rate (ETPR) as

$$\text{EFPR}(\mathcal{M}, \eta) = \frac{|\hat{\mathcal{E}}(\mathcal{M}, \eta) \cap \mathcal{E}^C|}{|\mathcal{E}^C|}, \qquad \text{ETPR}(\mathcal{M}, \eta) = \frac{|\hat{\mathcal{E}}(\mathcal{M}, \eta) \cap \mathcal{E}|}{|\mathcal{E}|}.$$

The node FPR and TPR are calculated similarly.

We also evaluate the prediction accuracy of the methods. For each method, we use 5-fold cross-validation to select the best tuning parameter using the training data, and then compute the test error on a different dataset simulated under the same settings. The AUC and test errors reported are averaged over 30 replications.

Methods for benchmark comparisons on simulated networks were selected based on their good performance on real data (see Section 6). For our method (GC), we vary the parameter ρ and compare results for two different values of λ, 0.05 (GC1) and 10^{-4} (GC2). For unstructured regularized logistic regression, we use the elastic net (Friedman, Hastie and Tibshirani (2010b)), with a fixed $\alpha = 0.02$ (ENet). The performance of elastic net is not very sensitive to different values of α, but the number of variables that the method is able to select with large values is limited (including the case of $\alpha = 1$ that corresponds to the lasso). A support vector machine with ℓ_1 penalty (Becker et al. (2009), Zhu et al. (2004)) is also included (SVML1) for comparison, and additionally we evaluate the classification error of the original support vector machines (SVM) (Cortes and Vapnik (1995)). For both SVMs, we use linear kernels, which performed better than nonlinear ones. We also consider an independent screening method for variable selection based on the two sample t-statistic (T-stat). Finally, we also compare with the signal-subgraph method (SS) (Vogelstein et al. (2013)) which is the only other method that takes into account the network structure of the predictor variables. Note that the signal subgraph is designed for binary networks, so in order to apply it we thresholded
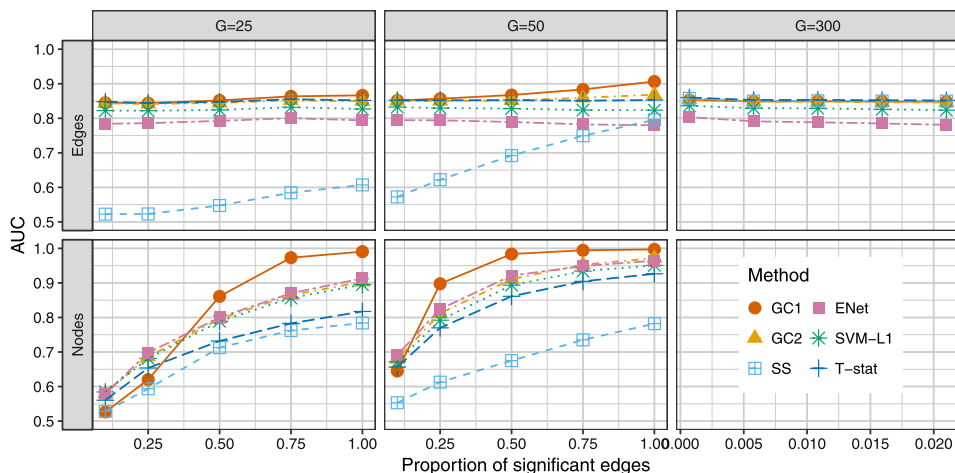
F_IG. 4. *Variable selection performance of different methods in terms of edge AUC (top) and node AUC (bottom) as a function of the fraction of differentiating edges in the subgraph induced by the active node set $\mathcal{G}$. As the proportion of active edges increases, methods that use network structure improve their performance when only a subset of the nodes is active.*

each edge at the population mean. For each method, we fit 10 different tuning parameters to change the sparsity of the solution.

Figure 4 shows the values of the average AUC for selecting edges (top) and nodes (bottom). For $G = 25$ and 50, as the proportion of differentiating edges in the active subgraph increases, methods that take into account network structure (GC1, GC2 and SS) slightly improve their edge AUC, since enforcing node selection also results in better edge selection, while the edge AUC remains constant for unstructured methods (ENet, T-stat and SVML1). On node selection, all methods improve the node AUC as the fraction of significant edges increases, but GC and SS have the largest gains. A similar trend is observed in classification error shown in Figure 5. All methods improve as the proportion of differentiating edges increases, but our method has the best performance overall. Our method performed the best with the larger value of λ (GC1) on variable selection, particularly when the set of active nodes is smaller, but both values of λ give very good classification performance. In the last scenario ($G = 300$), all nodes are active so the node AUC is undefined, and the node structure is not informative at all. Although the performance of our method is no longer the best, it performs comparably to state of the art methods that do not use network structure.

In terms of computing time, since there are many contributing factors including the software choice for implementation and the tuning parameters, a fair comparison is difficult. We can roughly say that elastic net is the fastest, taking about a minute to run a cross-validation instance, while our method takes about 10 minutes on average, and the signal-subgraph takes more than an hour.
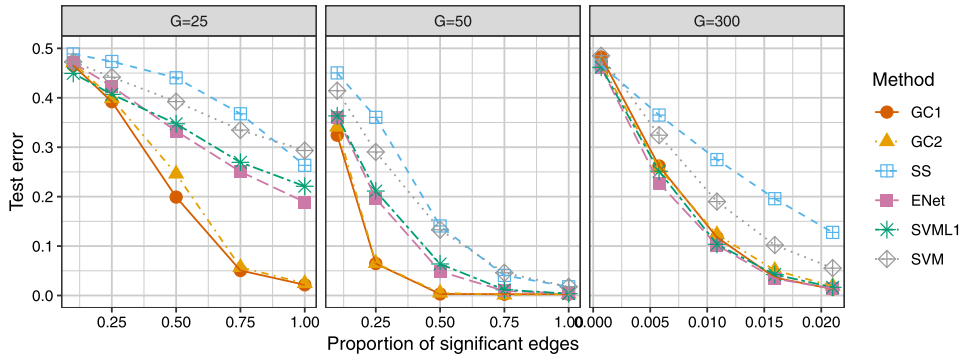
FIG. 5. *Classification error of different methods as a function of the fraction of differentiating edges in the subgraph induced by the active node set $\mathcal{G}$. Our method is more accurate when only a subset of the nodes is active.*

6. Application to schizophrenia data. We analyze the performance of the classifier on the two different brain fMRI datasets previously described in Section 1. The code of our classifier and the processed connectomics datasets are available in the Supplementary Material (Arroyo Relión et al. (2019)).

6.1. *Classification results.* First, we evaluate our method's classification accuracy. We use a nested 10-fold cross-validation to choose tuning parameters and estimate the test accuracy. The classifier is trained for a range of values of λ and ρ, with $\lambda \in \{10^{-7}, 10^{-6.5}, \ldots, 10^{-2}\}$ and $\rho \in \{10^{-3}, 10^{-2.5}, \ldots, 10^2\}$. The value of γ in (2.4) is set to 10^{-5}; we observed that setting γ to a small value speeds up convergence without affecting the accuracy or sparsity of the solution. Figure 6 shows the average cross-validated accuracy, sparsity (fraction of zero coefficients) and node sparsity (fraction of inactive nodes), as a heat map over the grid of tuning parameter values. We observe that λ has little influence on sparsity, which is primarily controlled by ρ. Moreover, as Proposition 1 suggests, values of $\rho < 1$ do not result in node selection. As expected, accuracy generally decreases as the solution becomes sparser, which is not uncommon in high-dimensional settings (Hastie, Tibshirani and Wainwright (2015)). However, we can still achieve excellent accuracy with a substantially reduced set of features. In the COBRE dataset, the best accuracy is obtained with only 1886 edges (5.4%) but almost all nodes are active (260). On the UMich data, 29,733 edges (85.6%) achieve the best performance, and all nodes are active. Choosing parameters by cross-validation often tends to include too many noise variables (Meinshausen (2007)), as we also observed in simulations. A commonly used technique to report solutions that still achieve good accuracy with a substantially reduced set of features is the so-called "one-standard-error rule" (Hastie, Tibshirani and Wainwright (2015)), in which one selects the most parsimonious classifier with cross-validation accuracy at most one standard error away from the best cross-validation accuracy. Figure 7 shows
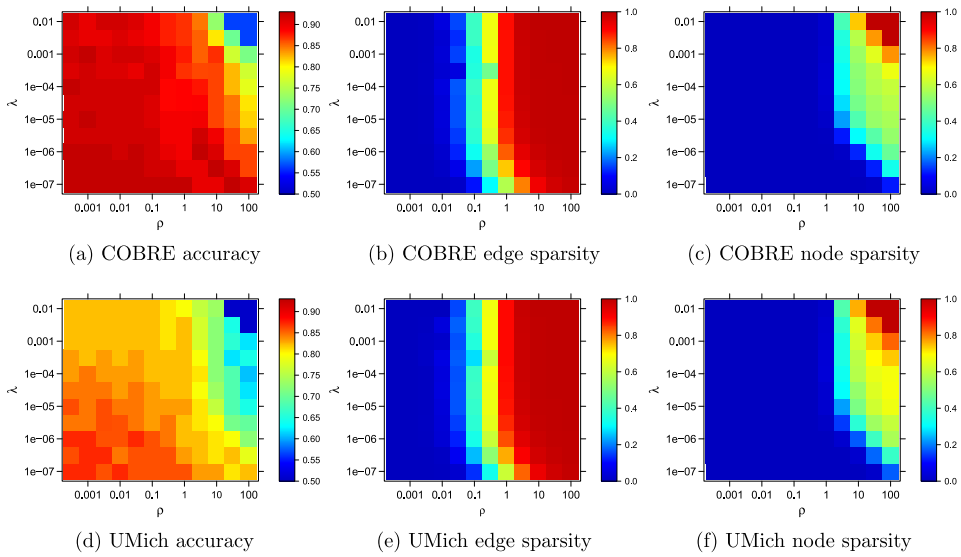
FIG. 6. *Cross-validated results for the two data sets. Classification accuracy (left), fraction of zero edge coefficients (middle), and fraction of inactive nodes (right).*

the solutions for each dataset obtained by this rule. Nodes are ordered by brain systems (see Figure 1). The fitted solution for COBRE has 549 nonzero coefficients (1.56%) and 217 active nodes (82.5%), while the UMich solution has 11,748 nonzero entries (33.8%), and all nodes are active. Note that when many variables
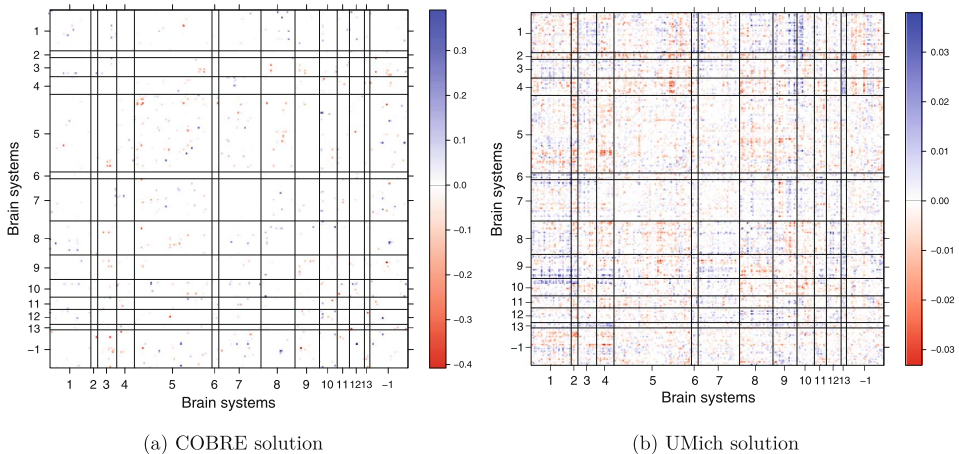

FIG. 7. *Fitted coefficients for COBRE and UMich datasets, with tuning parameters selected by the "one standard error rule." Positive coefficients corresponds to higher edge weights for schizophrenic patients.*

TABLE 1
Cross-validated accuracy (average and standard errors over 10 folds) for different methods

Method	Classification accuracy % (s.e.)	
	COBRE	UMich
With variable selection		
Our method (GC)	92.7 (2.6)	85.9 (3.6)
Elastic net	89.5 (1.8)	82.6 (4.7)
SVM-L1	87.9 (2.2)	86.2 (4.3)
Signal-subgraph	86.1 (3.3)	82.4 (3.3)
DLDA	84.6 (3.3)	73.4 (3.9)
Lasso	80.1 (5.6)	60.9 (5.6)
No variable selection		
SVM	93.5 (2.1)	89.8 (2.5)
Ridge penalty	91 (2.6)	80.9 (3.5)
Random forest	74.2 (2.6)	82.1 (3.9)
Network summaries	61.4 (3.1)	65 (7.2)

are selected, the magnitude of the coefficients becomes small due to the grouping effect of the penalty (Zou and Hastie (2005)).

We also compared our method to benchmarks (Table 1), using the same methods as in the previous section and training and evaluating all methods using the same nested 10-fold cross-validation. For SVM, we tested different kernels, including graph aware kernels (Gärtner, Flach and Wrobel (2003)), but in most cases local kernel methods were no better than random guessing. We additionally included random forests and a method based on global and local network summaries previously proposed as features for classifying brain data (Prasad et al. (2015)). For the latter, because our dataset is much larger, we only considered global and node features proposed in Prasad et al. (2015), which resulted in about 30,000 features per individual, and omitted edge features. Watanabe et al. (2014) evaluated their classifiers on a different parcellation of the COBRE data, and we do not include their methods since they are based on the assumption of equally spaced nodes and cannot be directly applied to our data. Their reported accuracy of 71.9% and 73.5% for the COBRE data is substantially lower than our method, although the results are not directly comparable.

Results in Table 1 show that most methods performed better on the COBRE dataset than on the UMich dataset, which can be partially explained by the different sample sizes and possibly noise levels. Besides differences in sample size and demographic characteristics (see Supplementary Material (Arroyo Relión et al. (2019))), the COBRE dataset is more homogeneous as it was collected using identical acquisition parameters, whereas the UMich dataset was pooled across five different experiments spanning seven years.
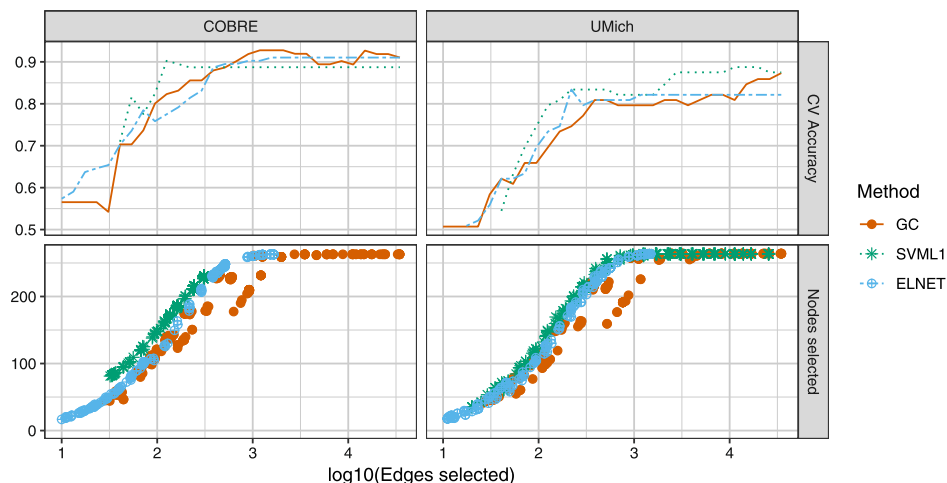
FIG. 8. *Cross-validated accuracy and number of nodes selected as a function of the number of edges used.*

Our method performs very well on both datasets, particularly among methods that do variable selection. SVMs, which use the hinge loss, perform well too, and generally outperform methods using the logistic loss. Our penalty can be combined with any loss, so we could also include our penalty combined with hinge loss which might potentially improve classification accuracy, but we do not pursue this direction, for two reasons: one, our method is close to SVM + L1 as it is (better on COBRE, slightly worse on UMich but the difference is within noise levels), and because solutions based on logistic loss are generally considered more stable and preferable for variable selection (Hastie, Tibshirani and Wainwright (2015)). In Figure 8, we plot cross-validated classification accuracy of these methods as a function of the number of variables selected. For the COBRE data, as we have observed before, good accuracy can be achieved with a fairly small number of edges, and the noisier UMich data requires more edges. In all cases, our method uses fewer nodes than the others, as it is designed to do so.

Ultimately, assessing significance of the selected variables is necessary, which is in general a difficult task in high-dimensional settings and an active area of research (see, e.g., Lee et al. (2016), Lockhart et al. (2014), Meinshausen and Bühlmann (2010), van de Geer et al. (2014)). In brain connectomics, it is particularly challenging to identify significant variables because of small sample sizes (Button et al. (2013)). Here we employ *stability selection* (Meinshausen and Bühlmann (2010)) which can be shown to control a type of false discovery rate by employing many rounds of random data splitting and calculating the probability of each variable being selected. Some versions of this method have been theoretically studied, and upper bounds on the expected number of variables with a low selection probability that are included in the final solution (i.e., errors) have

Edges with the top 15 *largest selection probabilities from stability selection. The first column shows the pair of nodes making the edge, the second column the brain systems the nodes belong to in the Power parcellation, and the third column the fitted coefficient of the edge*

	COBRE			UMich		
	Edge	Systems	Coefficient	Edge	Systems	Coefficient
1	(208, 85)	(9, −1)	−0.187	(110, 207)	(5, 9)	−0.013
2	(260, 11)	(12, −1)	0.183	(255, 113)	(1, 5)	0.014
3	(194, 140)	(8, −1)	0.136	(33, 218)	(1, 9)	0.016
4	(52, 186)	(3, 8)	−0.1	(46, 225)	(2, 10)	0.013
5	(160, 239)	(7, 11)	−0.082	(43, 90)	(2, 5)	−0.013
6	(120, 116)	(5, 5)	0.099	(23, 225)	(1, 10)	0.012
7	(57, 129)	(3, 5)	−0.128	(66, 118)	(4, 5)	−0.013
8	(24, 114)	(1, 5)	−0.148	(26, 145)	(1, 7)	0.013
9	(81, 179)	(5, 8)	−0.129	(186, 254)	(8, −1)	0.012
10	(193, 140)	(8, −1)	0.153	(15, 134)	(1, 6)	0.011
11	(178, 234)	(8, 10)	0.146	(76, 207)	(5, 9)	−0.012
12	(18, 194)	(1, 8)	0.116	(65, 84)	(4, −1)	−0.012
13	(215, 207)	(9, 9)	−0.076	(26, 122)	(1, 5)	0.012
14	(90, 224)	(5, 10)	0.123	(33, 145)	(1, 7)	0.012
15	(112, 253)	(5, −1)	0.136	(36, 224)	(1, 10)	0.011

been derived under mild conditions (Meinshausen and Bühlmann (2010), Shah and Samworth (2013)). We implemented the version of stability selection proposed by Shah and Samworth (2013), with values of λ and ρ obtained by cross-validation on the COBRE data, and by the "one standard error rule" on the UMich dataset, since stability selection is most relevant to sparse solutions. However, one of the advantages of stability selection is that it is not sensitive to the initial choice of tuning parameters, and changing tuning parameters only slightly alters the ordering of variables with the largest selection probabilities.

The edges with the 15 largest selection probabilities are reported in Table 2. Using the results of Shah and Samworth (2013) (equation 8), we estimated that the expected number of falsely selected variables (variables with a probability of selection smaller than the estimated) is bounded by 6.1 for the COBRE dataset and 9.7 for the UMich data, which also suggests that results on the UMich data might be less reliable. While the two datasets yield somewhat different patterns of edge selection, it is notable that the default mode network (5) was often selected in both. This network has been consistently implicated in schizophrenia (Ongür et al. (2010), Peeters et al. (2015), Whitfield-Gabrieli et al. (2009)), as well as other psychiatric disorders, possibly as a general marker of psychopathology (Broyd et al. (2009), Menon (2011)). In the COBRE dataset, edges were also selected from the fronto-parietal task control region (8), previously linked to schizophrenia (Bunney and Bunney (2000), Fornito et al. (2012)). These results coincide with the findings

TABLE 3
Classification accuracy (cross-validation average and standard error) of the classifier fitted on one dataset and evaluated on the other. The intercept (the mean) is fitted on the test data and the accuracy is estimated using 10-fold cross-validation on the test data

| | Test data | |
Training data	COBRE	UMich
COBRE	92.7 (2.6)	73.5 (3.4)
UMich	78.3 (3.0)	85.9 (3.6)

of Watanabe et al. (2014) on a different parcellation of the same data, which is an encouraging indication of robustness to the exact choice of node locations. Some of the variables with the highest estimated selection probabilities appear in the uncertain system (-1), in particular in the cell connecting it with salience system (9), which suggests that alternative parcellations that better characterize these regions may offer a better account of the schizophrenia-related changes. Additionally, sensory/somatomotor hand region (1) and salience system (9) also stand out in the UMich data, and these are networks that have also been implicated in schizophrenia (Dong et al. (2017)).

While results in Table 2 do not fully coincide on the two datasets, there are clear commonalities. Table 3 compares classification accuracy when the classifier is trained on one dataset and tested on the other (with the exception of the intercept, since the datasets are not centered in the same way, which is fitted on a part of the test data, and the test error is then computed via 10-fold cross-validation). While the accuracy is lower than when the same dataset is used for training and testing, as one would expect, it is still reasonably good and in fact better than some of the benchmark methods even when they train and test on the same data. We again observe that the COBRE dataset is easier to classify.

Figure 9 shows the active nodes in the COBRE dataset (marked in green), corresponding to the endpoints of the edges listed in Table 2. We also identified a set of 25 nodes that are not selected in any of the sparse solutions with cross-validation accuracy within one standard error from the best solution (marked in purple). These consistently inactive nodes are mostly clustered in two anatomically coherent regions.

7. Discussion. We have presented a method for classifying graphs with labeled nodes, motivated by brain connectomics but generally applicable to any setting with such graphs. The distinct feature of our method is that it is graph-aware, aiming to select a sparse set of *both* edges and nodes, but it is general in the sense that it does not rely on the spatial structure of the brain. The method is computationally efficient since the regularization we use is convex, and the solution is
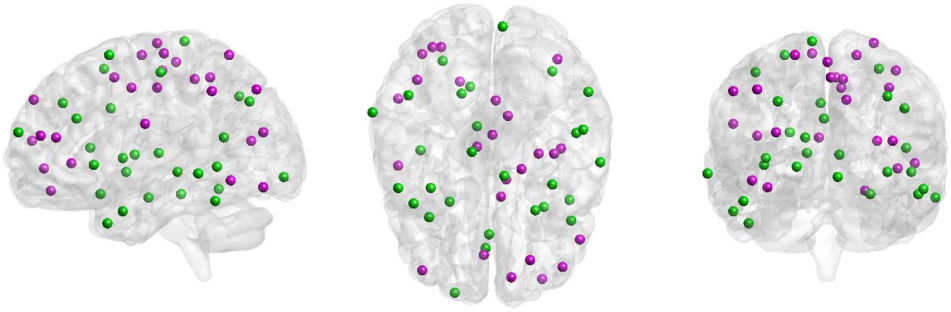
FIG. 9. *Nodes shown in green are endpoints of edges selected by stability selection shown in Table 2. Nodes shown in purple are nodes not selected by any of the sparse solutions within one standard error of the most accurate solution.*

implemented with efficient optimization algorithms. These properties guarantee fast convergence to the solution, making the methods scalable to networks with thousands of nodes, which is enough to deal with many of the brain atlases usually employed in neuroimaging (see, e.g., Kiar et al. (2018)). Statistically, the rate of convergence depends on the number of active nodes only, not the total number of nodes, which allows for accurate results with even moderate sample sizes if the active node set is small.

The results we obtained on the schizophrenia data are generally in agreement with previous studies. In particular, the default mode network has been consistently implicated in schizophrenia and many other psychiatric disorders (Broyd et al. (2009), Ongür et al. (2010)). While different subnetworks were implicated by the two different datasets, we are still able to predict the disease status fairly accurately by training on one dataset and testing on the other. The differences between the two datasets may reflect real differences in samples collected at different sites and in different experiments, as significant pathophysiological heterogeneity occurs for all psychiatric diagnoses, or they may simply reflect type 2 errors.

Our methods work very generally with a sample of networks with labeled nodes and associated responses. The many preprocessing steps inevitable when dealing with fMRI data always add some uncertainty, and preprocessing decisions can potentially affect downstream conclusions. We aimed to somewhat mitigate this by using ranks, which are more robust and showed a slightly better performance on our datasets. Another option, when practical, is to compare multiple preprocessing pipelines, and/or multiple measures of connectivity, to further validate results. Our method's independence of these particular choices and its computational efficiency make it an attractive option for such comparisons.

SUPPLEMENTARY MATERIAL

Supplement A: Algorithms, proofs, and data aquisition and preprocessing details (DOI: 10.1214/19-AOAS1252SUPPA; .pdf). In this supplementary mate-

rial, we provide the details of the optimization algorithms, proof of the theoretical results and a detailed description of the data aquisition and preprocessing.

Supplement B: Code and data (DOI: 10.1214/19-AOAS1252SUPPB; .zip). The .zip file contains source code of an R package that implements the methods described in this paper, as well as the post-processed connectomes used in the analysis.

REFERENCES

AINE, C. J., BOCKHOLT, H. J., BUSTILLO, J. R., CAÑIVE, J. M., CAPRIHAN, A., GAS-PAROVIC, C., HANLON, F. M., HOUCK, J. M., JUNG, R. E. et al. (2017). Multimodal neuroimaging in schizophrenia: Description and dissemination. *Neuroinformatics* **15** 343–364.

ARROYO RELIÓN, J. D., KESSLER, D., LEVINA, E. and TAYLOR, S. F. (2019). Supplement to "Network classification with applications to brain connectomics." DOI:10.1214/19-AOAS1252SUPPA, DOI:10.1214/19-AOAS1252SUPPB.

BACH, F. R. (2008). Consistency of the group lasso and multiple kernel learning. *J. Mach. Learn. Res.* **9** 1179–1225. MR2417268

BACH, F., JENATTON, R., MAIRAL, J. and OBOZINSKI, G. (2012). Structured sparsity through convex optimization. *Statist. Sci.* **27** 450–468. MR3025128

BECK, A. and TEBOULLE, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2** 183–202. MR2486527

BECKER, N., WERFT, W., TOEDT, G., LICHTER, P. and BENNER, A. (2009). penalizedSVM: A R-package for feature selection SVM classification. *Bioinformatics* **25** 1711–1712.

BENGIO, Y. and MONPERRUS, M. (2005). Non-local manifold tangent learning. In *Advances in Neural Information Processing Systems* 129–136.

BICKEL, P. J. and CHEN, A. (2009). A nonparametric view of network models and Newman–Girvan and other modularities. *Proc. Natl. Acad. Sci. USA* **106** 21068–21073.

BORGWARDT, K. M., ONG, C. S., SCHÖNAUER, S., VISHWANATHAN, S., SMOLA, A. J. and KRIEGEL, H.-P. (2005). Protein function prediction via graph kernels. *Bioinformatics* **21** i47–i56.

BOYD, S., PARIKH, N., CHU, E., PELEATO, B. and ECKSTEIN, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3** 1–122.

BROYD, S. J., DEMANUELE, C., DEBENER, S., HELPS, S. K., JAMES, C. J. and SONUGA-BARKE, E. J. (2009). Default-mode brain dysfunction in mental disorders: A systematic review. *Neurosci. Biobehav. Rev.* **33** 279–296.

BULLMORE, E. T. and BASSETT, D. S. (2011). Brain graphs: Graphical models of the human brain connectome. *Annu. Rev. Clin. Psychol.* **7** 113–140.

BULLMORE, E. and SPORNS, O. (2009). Complex brain networks: Graph theoretical analysis of structural and functional systems. *Nat. Rev., Neurosci.* **10** 186–198.

BUNNEY, W. E. and BUNNEY, B. G. (2000). Evidence for a compromised dorsolateral prefrontal cortical parallel circuit in schizophrenia. *Brains Res. Rev.* **31** 138–146.

BUTTON, K. S., IOANNIDIS, J. P., MOKRYSZ, C., NOSEK, B. A., FLINT, J., ROBINSON, E. S. and MUNAFÒ, M. R. (2013). Power failure: Why small sample size undermines the reliability of neuroscience. *Nat. Rev., Neurosci.* **14** 365–376.

CHEN, X., LIN, Q., KIM, S., CARBONELL, J. G. and XING, E. P. (2012). Smoothing proximal gradient method for general structured sparse regression. *Ann. Appl. Stat.* **6** 719–752. MR2976489

CORTES, C. and VAPNIK, V. (1995). Support-vector networks. *Mach. Learn.* **20** 273–297.

CRADDOCK, R. C., HOLTZHEIMER, P. E., HU, X. P. and MAYBERG, H. S. (2009). Disease state prediction from resting state functional connectivity. *Magn. Reson. Med.* **62** 1619–1628.

DESHPANDE, M., KURAMOCHI, M., WALE, N. and KARYPIS, G. (2005). Frequent substructure-based approaches for classifying chemical compounds. *IEEE Trans. Knowl. Data Eng.* **17** 1036–1050.

DONG, D., WANG, Y., CHANG, X., LUO, C. and YAO, D. (2017). Dysfunction of large-scale brain networks in schizophrenia: A meta-analysis of resting-state functional connectivity. *Schizophr. Bull.* **44** 168–181.

DUCHI, J. and SINGER, Y. (2009). Efficient online and batch learning using forward backward splitting. *J. Mach. Learn. Res.* **10** 2899–2934. MR2579916

DURANTE, D. and DUNSON, D. B. (2018). Bayesian inference and testing of group differences in brain networks. *Bayesian Anal.* **13** 29–58. MR3737942

FEI, H. and HUAN, J. (2010). Boosting with structure information in the functional space: An application to graph classification. In *Proceedings of the* 16*th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 643–652. ACM, New York.

FINN, E. S., SHEN, X., SCHEINOST, D., ROSENBERG, M. D., HUANG, J., CHUN, M. M., PAPADEMETRIS, X. and CONSTABLE, R. T. (2015). Functional connectome fingerprinting: Identifying individuals using patterns of brain connectivity. *Nat. Neurosci.* **18** 1664–1671.

FORNITO, A., ZALESKY, A., PANTELIS, C. and BULLMORE, E. T. (2012). Schizophrenia, neuroimaging and connectomics. *NeuroImage* **62** 2296–2314.

FRIEDMAN, J., HASTIE, T. and TIBSHIRANI, R. (2010a). A note on the group lasso and a sparse group lasso. Available at arXiv:1001.0736.

FRIEDMAN, J., HASTIE, T. and TIBSHIRANI, R. (2010b). Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **33** 1–22.

GAO, C., MA, Z., ZHANG, A. Y. and ZHOU, H. H. (2017). Achieving optimal misclassification proportion in stochastic block models. *J. Mach. Learn. Res.* **18** Paper No. 60, 45. MR3687603

GÄRTNER, T., FLACH, P. and WROBEL, S. (2003). On graph kernels: Hardness results and efficient alternatives. In *Learning Theory and Kernel Machines* 129–143. Springer, Berlin.

GINESTET, C. E., LI, J., BALACHANDRAN, P., ROSENBERG, S. and KOLACZYK, E. D. (2017). Hypothesis testing for network data in functional neuroimaging. *Ann. Appl. Stat.* **11** 725–750. MR3693544

GONZALEZ, J., HOLDER, L. B. and COOK, D. J. (2000). Graph based concept learning. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. AAAI Press* 1072. MIT Press, Cambridge, MA.

GOTTS, S., SAAD, Z., JO, H. J., WALLACE, G., COX, R. and MARTIN, A. (2013). The perils of global signal regression for group comparisons: A case study of autism spectrum disorders. *Front. Human Neurosci.* **7** 356.

GROSENICK, L., KLINGENBERG, B., KATOVICH, K., KNUTSON, B. and TAYLOR, J. E. (2013). Interpretable whole-brain prediction analysis with GraphNet. *NeuroImage* **72** 304–321.

HASTIE, T., TIBSHIRANI, R. and WAINWRIGHT, M. (2015). *Statistical Learning with Sparsity*: *The Lasso and Generalizations. Monographs on Statistics and Applied Probability* **143**. CRC Press, Boca Raton, FL. MR3616141

HELMA, C., KING, R. D., KRAMER, S. and SRINIVASAN, A. (2001). The predictive toxicology challenge 2000–2001. *Bioinformatics* **17** 107–108.

HOLLAND, P. W., LASKEY, K. B. and LEINHARDT, S. (1983). Stochastic blockmodels: First steps. *Soc. Netw.* **5** 109–137. MR0718088

HU, Y. and ALLEN, G. I. (2015). Local-aggregate modeling for big data via distributed optimization: Applications to neuroimaging. *Biometrics* **71** 905–917. MR3436716

INOKUCHI, A., WASHIO, T. and MOTODA, H. (2000). An apriori-based algorithm for mining frequent substructures from graph data. In *Principles of Data Mining and Knowledge Discovery* 13–23. Springer, Berlin.

JACOB, L., OBOZINSKI, G. and VERT, J.-P. (2009). Group lasso with overlap and graph lasso. In *Proceedings of the* 26*th Annual International Conference on Machine Learning* 433–440. ACM, New York.

KASHIMA, H., TSUDA, K. and INOKUCHI, A. (2003). Marginalized kernels between labeled graphs. In *International Conference of Machine Learning* 3 321–328.

KETKAR, N. S., HOLDER, L. B. and COOK, D. J. (2009). Empirical comparison of graph classification algorithms. In *Computational Intelligence and Data Mining*, 2009. *CIDM'09. IEEE Symposium on* 259–266. IEEE, New York.

KIAR, G., BRIDGEFORD, E., RONCAL, W. G., CHANDRASHEKHAR, V., MHEMBERE, D., RYMAN, S., ZUO, X.-N., MARGUILES, D. S., CRADDOCK, R. C. et al. (2018). A high-throughput pipeline identifies robust connectomes but troublesome variability. *BioRxiv* 188706.

KUDO, T., MAEDA, E. and MATSUMOTO, Y. (2004). An application of boosting to graph classification. In *Advances in Neural Information Processing Systems* 729–736.

LE, C. M., LEVINA, E. and VERSHYNIN, R. (2017). Concentration and regularization of random graphs. *Random Structures Algorithms* 51 538–561. MR3689343

LEE, J. D., SUN, Y. and TAYLOR, J. E. (2015). On model selection consistency of regularized M-estimators. *Electron. J. Stat.* 9 608–642. MR3331852

LEE, J. D., SUN, D. L., SUN, Y. and TAYLOR, J. E. (2016). Exact post-selection inference, with application to the lasso. *Ann. Statist.* 44 907–927. MR3485948

LINDQUIST, M. A. (2008). The statistical analysis of fMRI data. *Statist. Sci.* 23 439–464. MR2530545

LIU, Y., LIANG, M., ZHOU, Y., HE, Y., HAO, Y., SONG, M., YU, C., LIU, H., LIU, Z. et al. (2008). Disrupted small-world networks in schizophrenia. *Brain* 131 945–961.

LOCKHART, R., TAYLOR, J., TIBSHIRANI, R. J. and TIBSHIRANI, R. (2014). A significance test for the lasso. *Ann. Statist.* 42 413–468. MR3210970

MEINSHAUSEN, N. (2007). Relaxed Lasso. *Comput. Statist. Data Anal.* 52 374–393. MR2409990

MEINSHAUSEN, N. and BÜHLMANN, P. (2006). High-dimensional graphs and variable selection with the lasso. *Ann. Statist.* 34 1436–1462. MR2278363

MEINSHAUSEN, N. and BÜHLMANN, P. (2010). Stability selection. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* 72 417–473. MR2758523

MENON, V. (2011). Large-scale brain networks and psychopathology: A unifying triple network model. *Trends Cogn. Sci.* 15 483–506.

NARAYAN, M., ALLEN, G. I. and TOMSON, S. (2015). Two sample inference for populations of graphical models with applications to functional connectivity. Available at arXiv:1502.03853.

ONGÜR, D., LUNDY, M., GREENHOUSE, I., SHINN, A. K., MENON, V., COHEN, B. M. and RENSHAW, P. F. (2010). Default mode network abnormalities in bipolar disorder and schizophrenia. *Psychiatry Res.* 183 59–68.

PARIKH, N. and BOYD, S. (2013). Proximal algorithms. *Found. Trends Optim.* 1 123–231.

PEETERS, S. C., VAN DE VEN, V., GRONENSCHILD, E. H. M., PATEL, A. X., HABETS, P., GOEBEL, R., VAN OS, J. and MARCELIS, M. (2015). Default mode network connectivity as a function of familial and environmental risk for psychotic disorder. *PLoS ONE* 10 e0120030.

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. et al. (2011). Functional network organization of the human brain. *Neuron* 72 665–678.

PRASAD, G., JOSHI, S. H., NIR, T. M., TOGA, A. W., THOMPSON, P. M. and ADNI (2015). Brain connectivity and novel network measures for Alzheimer's disease classification. *Neurobiol. Aging* 36 121–131.

RICHIARDI, J., ERYILMAZ, H., SCHWARTZ, S., VUILLEUMIER, P. and VAN DE VILLE, D. (2011). Decoding brain states from fMRI connectivity graphs. *NeuroImage* 56 616–626.

SCHEINBERG, K., GOLDFARB, D. and BAI, X. (2014). Fast first-order methods for composite convex optimization with backtracking. *Found. Comput. Math.* 14 389–417. MR3201951

SCOTT, J. G., KELLY, R. C., SMITH, M. A., ZHOU, P. and KASS, R. E. (2015). False discovery rate regression: An application to neural synchrony detection in primary visual cortex. *J. Amer. Statist. Assoc.* **110** 459–471. MR3367240

SHAH, R. D. and SAMWORTH, R. J. (2013). Variable selection with error control: Another look at stability selection. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **75** 55–80. MR3008271

SMITH, S. M., VIDAURRE, D., BECKMANN, C. F., GLASSER, M. F., JENKINSON, M., MILLER, K. L., NICHOLS, T. E., ROBINSON, E. C., SALIMI-KHORSHIDI, G. et al. (2013). Functional connectomics from resting-state fMRI. *Trends Cogn. Sci.* **17** 666–682.

SRINIVASAN, A., MUGGLETON, S. H., STERNBERG, M. J. and KING, R. D. (1996). Theories for mutagenicity: A study in first-order and feature-based induction. *Artificial Intelligence* **85** 277–299.

SRIPADA, C., ANGSTADT, M., KESSLER, D., PHAN, K. L., LIBERZON, I., EVANS, G. W., WELSH, R. C., KIM, P. and SWAIN, J. E. (2014a). Volitional regulation of emotions produces distributed alterations in connectivity between visual, attention control, and default networks. *NeuroImage* **89** 110–121.

SRIPADA, C., KESSLER, D., FANG, Y., WELSH, R. C., PREM KUMAR, K. and ANGSTADT, M. (2014b). Disrupted network architecture of the resting brain in attention-deficit/hyperactivity disorder. *Hum. Brain Mapp.* **35** 4693–4705.

SUPEKAR, K., MENON, V., RUBIN, D., MUSEN, M. and GREICIUS, M. D. (2008). Network analysis of intrinsic functional brain connectivity in Alzheimer's disease. *PLoS Comput. Biol.* **4** e1000100.

TANG, M., ATHREYA, A., SUSSMAN, D. L., LYZINSKI, V., PARK, Y. and PRIEBE, C. E. (2017a). A semiparametric two-sample hypothesis testing problem for random graphs. *J. Comput. Graph. Statist.* **26** 344–354. MR3640191

TANG, M., ATHREYA, A., SUSSMAN, D. L., LYZINSKI, V. and PRIEBE, C. E. (2017b). A nonparametric two-sample hypothesis testing problem for random graphs. *Bernoulli* **23** 1599–1630. MR3624872

TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B* **58** 267–288. MR1379242

VAN DE GEER, S., BÜHLMANN, P., RITOV, Y. and DEZEURE, R. (2014). On asymptotically optimal confidence regions and tests for high-dimensional models. *Ann. Statist.* **42** 1166–1202. MR3224285

VAROQUAUX, G. and CRADDOCK, R. C. (2013). Learning and comparing functional connectomes across subjects. *NeuroImage* **80** 405–415.

VISHWANATHAN, S. V. N., SCHRAUDOLPH, N. N., KONDOR, R. and BORGWARDT, K. M. (2010). Graph kernels. *J. Mach. Learn. Res.* **11** 1201–1242. MR2645450

VOGELSTEIN, J. T., RONCAL, W. G., VOGELSTEIN, R. J. and PRIEBE, C. E. (2013). Graph classification using signal-subgraphs: Applications in statistical connectomics. *IEEE Trans. Pattern Anal. Mach. Intell.* **35** 1539–1551.

WATANABE, T., KESSLER, D., SCOTT, C., ANGSTADT, M. and SRIPADA, C. (2014). Disease prediction based on functional connectomes using a scalable and spatially-informed support vector machine. *NeuroImage* **96** 183–202.

WHITFIELD-GABRIELI, S., THERMENOS, H. W., MILANOVIC, S., TSUANG, M. T., FARAONE, S. V., MCCARLEY, R. W., SHENTON, M. E., GREEN, A. I., NIETO-CASTANON, A. et al. (2009). Hyperactivity and hyperconnectivity of the default network in schizophrenia and in first-degree relatives of persons with schizophrenia. *Proc. Natl. Acad. Sci. USA* **106** 1279–1284.

XIN, B., KAWAHARA, Y., WANG, Y. and GAO, W. (2014). Efficient generalized fused lasso and its application to the diagnosis of Alzheimer's disease. In *Twenty-Eighth AAAI Conference on Artificial Intelligence* 2163–2169.

YUAN, M. and LIN, Y. (2006). Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **68** 49–67. MR2212574

YUAN, L., LIU, J. and YE, J. (2011). Efficient methods for overlapping group lasso. In *Advances in Neural Information Processing Systems* 352–360.

ZALESKY, A., FORNITO, A. and BULLMORE, E. T. (2010). Network-based statistic: Identifying differences in brain networks. *NeuroImage* **53** 1197–1207.

ZHANG, A. Y. and ZHOU, H. H. (2016). Minimax rates of community detection in stochastic block models. *Ann. Statist.* **44** 2252–2280. MR3546450

ZHANG, J., CHENG, W., WANG, Z., ZHANG, Z., LU, W., LU, G. and FENG, J. (2012). Pattern classification of large-scale functional brain networks: Identification of informative neuroimaging markers for epilepsy. *PLoS ONE* **7** e36733.

ZHANG, L., GUINDANI, M., VERSACE, F., ENGELMANN, J. M. and VANNUCCI, M. (2016). A spatiotemporal nonparametric Bayesian model of multi-subject fMRI data. *Ann. Appl. Stat.* **10** 638–666. MR3528355

ZHOU, H., LI, L. and ZHU, H. (2013). Tensor regression with applications in neuroimaging data analysis. *J. Amer. Statist. Assoc.* **108** 540–552. MR3174640

ZHU, J., ROSSET, S., TIBSHIRANI, R. and HASTIE, T. J. (2004). 1-norm support vector machines. In *Advances in Neural Information Processing Systems* 49–56.

ZOU, H. and HASTIE, T. (2005). Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **67** 301–320. MR2137327

J. D. ARROYO RELIÓN
CENTER FOR IMAGING SCIENCE
JOHNS HOPKINS UNIVERSITY
BALTIMORE, MARYLAND 21218-2682
USA
E-MAIL: jarroyor@umich.edu

D. KESSLER
E. LEVINA
DEPARTMENT OF STATISTICS
UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN 48109–1107
USA
E-MAIL: kesslerd@umich.edu
         elevina@umich.edu

S. F. TAYLOR
DEPARTMENT OF PSYCHIATRY
UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN 48109–1107
USA
E-MAIL: sftaylor@umich.edu