A PENALIZED REGRESSION MODEL FOR THE JOINT ESTIMATION OF EQTL ASSOCIATIONS AND GENE NETWORK STRUCTURE[1]

BY MICOL MARCHETTI-BOWICK[*,2], YAOLIANG YU[†], WEI WU[*] AND ERIC P. XING[*]

Carnegie Mellon University [*] *and University of Waterloo* [†]

In this work, we present a new approach for jointly performing eQTL mapping and gene network inference while encouraging a transfer of information between the two tasks. We address this problem by formulating it as a multiple-output regression task in which we aim to learn the regression coefficients while simultaneously estimating the conditional independence relationships among the set of response variables. The approach we develop uses structured sparsity penalties to encourage the sharing of information between the regression coefficients and the output network in a mutually beneficial way. Our model, *inverse-covariance-fused lasso*, is formulated as a biconvex optimization problem that we solve via alternating minimization. We derive new, efficient optimization routines to solve each convex sub-problem that are based on extensions of state-of-the-art methods. Experiments on both simulated data and a yeast eQTL dataset demonstrate that our approach outperforms a large number of existing methods on the recovery of the true sparse structure of both the eQTL associations and the gene network. We also apply our method to a human Alzheimer's disease dataset and highlight some results that support previous discoveries about the disease.

1. Introduction. A critical task in the study of biological systems is understanding how gene expression is regulated within the cell. Although this problem has been studied extensively over the past few decades, it has recently gained momentum due to rapid advancements in techniques for high-throughput data acquisition. Within this broad task, two problems that have received significant attention in recent years are (a) identifying the genetic loci that regulate gene expression, a problem known as eQTL mapping [Rockman and Kruglyak (2006)], and (b) determining which genes have a direct influence on the expression of other genes, a problem known as gene network estimation [Gardner and Faith (2005)]. Prior work on learning regulatory associations has largely treated eQTL mapping and gene network estimation as completely separate problems.

In this work we pursue a holistic approach to discovering the patterns of gene regulation in the cell by integrating eQTL mapping and gene network estimation into a single model. Specifically, given a dataset that contains genotype information for a set of single nucleotide polymorphisms (SNPs) along with mRNA expression measurements for a set of genes, we aim to simultaneously learn the SNP–gene and gene–gene relationships. The key element of our approach is that we transfer knowledge between these two tasks in order to yield more accurate solutions to both problems.

In order to share information between tasks, we assume that two genes that are tightly linked in a regulatory network are likely to be associated with similar sets of SNPs in an eQTL map, and vice versa. Our assumption is motivated by the observation that genes participating in the same biological pathway or module are usually co-expressed or co-regulated, and therefore linked in a gene network [Barabasi and Oltvai (2004)]. Because of this, when the expression of one gene is perturbed, it is likely that the expression of the entire pathway will be affected. In the case of eQTL mapping, this means that any genetic locus that is associated with the expression of one gene will influence the expression of the entire subnetwork to which the gene belongs. By explicitly encoding these patterns into our model, we can take advantage of this biological knowledge to boost our statistical power for detecting eQTLs. Ultimately, this allows us to leverage information about gene–gene relationships to learn a more accurate set of eQTL associations, and similarly to leverage information about SNP–gene relationships to learn a more accurate gene network.

Based on these key assumptions, we construct a unified statistical model by formulating the problem as a multiple-output regression task in which we jointly estimate the regression coefficients and the inverse covariance structure among the response variables. Specifically, given SNPs $x = (x_1, \ldots, x_p)$ and genes $y = (y_1, \ldots, y_q)$, our goal is to regress y on x and simultaneously estimate the inverse covariance of y. In this model, the matrix of regression coefficients encodes the SNP–gene relationships in the eQTL map, whereas the inverse covariance matrix captures the gene–gene relationships in the gene network. In order to ensure that information is transferred between the two components, we incorporate a regularization penalty that explicitly encourages pairs of genes that have a high weight in the inverse covariance matrix to have similar regression coefficient values. This structured penalty enables the two estimates to learn from one another as well as from the data.

2. Related work. A large number of techniques have been developed to address eQTL mapping and gene network estimation in isolation. The traditional approach to eQTL mapping is to examine each SNP–gene pair independently and perform a univariate statistical test to determine whether an association exists between the two. Recently, a series of more complex multivariate models have been applied to this task in order to jointly capture the effects of several SNPs on one

gene [Michaelson et al. (2010)]. Similarly, the simplest method for gene network inference is to construct a graph in which each edge is weighted by the marginal correlation between the corresponding genes. Another related method that has recently gained popularity, known as Gaussian graphical model estimation or equivalently as inverse covariance estimation, produces a network in which the edges are instead weighted by the conditional correlations [Dempster (1972)].

In this work we will use a multiple-output regression model to perform eQTL mapping and incorporate a Gaussian graphical model over the genes in order to simultaneously infer the gene network. However, it is well known that model estimation is challenging in the high-dimensional setting in which we have more variables in the model than samples from which to learn. This is the precisely the setting frequently encountered in genomics, where we see datasets with at most a few hundred samples but as many as 1 million SNPs and tens of thousands of genes. To combat this problem, we will incorporate regularization in order to reduce variance in the model estimate.

The challenge of high-dimensional data has already inspired a remarkable number of useful methods for performing regularized regression [Kim and Xing (2009, 2012), Tibshirani (1996), Yuan and Lin (2006)] and regularized inverse covariance estimation [Banerjee, El Ghaoui and d'Aspremont (2008), Friedman, Hastie and Tibshirani (2008), Ravikumar et al. (2011), Tan et al. (2014)]. Our work will build on existing penalty terms that have been carefully designed to induce sparsity and structure.

The main stumbling block to using structured regularization in multi-task regression is that it typically requires prior knowledge of the relationships among the outputs, which is not always readily available. To circumvent this, another class of models have been developed that jointly learn the regression coefficients along with the output dependency structure [Lee and Liu (2012), Rai, Kumar and Daume (2012), Rothman, Levina and Zhu (2010), Sohn and Kim (2012), Zhang and Yeung (2010)]. However, none of these approaches use a regularization penalty to explicitly encourage shared structure between the estimate of the regression parameters and the output network. Furthermore, the majority of these methods do not learn the covariance structure of the outputs y but rather the conditional covariance of the outputs given the inputs $y \mid x$. This can be interpreted as the covariance of the noise matrix, and does not capture the true relationships between outputs.

Finally, several other approaches have been developed to boost the power of eQTL mapping by incorporating prior knowledge in other ways, often using Bayesian methods. However, the vast majority of these techniques consider each candidate SNP–gene association independently [Nica et al. (2010), Stephens (2013)] or can only be applied with a very limited number of SNPs or genes due to computational limitations [Banerjee, Yandell and Yi (2008), Flutre et al. (2013)].

Here we present a new approach that jointly performs eQTL mapping and gene network inference by leveraging prior biological knowledge about the structure of the problem. In particular, we construct a novel multiple-output regression model

that jointly estimates the regression coefficients and the output network via transfer learning. To our knowledge, we are the first to address this problem by incorporating regularization penalties to encourage the explicit sharing of information between the two estimates.

3. Background. Before presenting our approach, we provide some background on the problems of penalized multiple-output regression and sparse inverse covariance estimation, which will form the building blocks of our unified model.

In what follows, we assume X is an n-by-p dimensional matrix of SNP genotypes, which we also call *inputs*, and Y is an n-by-q dimensional matrix of gene expression values, which we also call *outputs*. Here n is the number of samples, p is the number of SNPs, and q is the number of genes. The element $x_{ij} \in \{0, 1, 2\}$ represents the genotype value of sample i at SNP j, encoded as 0 for two copies of the major allele, 1 for one copy of the minor allele, and 2 for two copies of the minor allele. Similarly $y_{ik} \in \mathbb{R}$ represents the expression value of sample i in gene k. We assume that the expression values for each gene are mean-centered.

3.1. *Multiple-output lasso.* Given input matrix X and output matrix Y, the standard ℓ_1-penalized multiple-output regression problem, also known as the multi-task lasso [Tibshirani (1996)], is given by

$$(1) \qquad \min_{B} \frac{1}{n} \|Y - XB\|_F^2 + \lambda \|B\|_1,$$

where B is a p-by-q dimensional matrix and β_{jk} is the regression coefficient that maps SNP x_j to gene y_k. Here $\| \cdot \|_1$ is an ℓ_1 norm penalty that induces sparsity among the estimated coefficients, and λ is a regularization parameter that controls the degree of sparsity. The objective function given above is derived from the penalized negative log likelihood of a multivariate Gaussian distribution, assuming $y \mid x \sim \mathcal{N}(x^T B, \varepsilon^2 I)$ where we let $\varepsilon^2 = 1$ for simplicity. Although this problem is formulated in a multiple-output framework, the ℓ_1 norm penalty merely encourages sparsity, and does not enforce any shared structure between the regression coefficients of different outputs. As a result, the objective function given in (1) decomposes into q independent regression problems.

3.2. *Graph-guided fused lasso.* Given a weighted graph $G \in \mathbb{R}^{q \times q}$ that encodes a set of pairwise relationships among the outputs, we can modify the regression problem by imposing an additional fusion penalty that encourages genes y_k and y_m to have similar parameter vectors $\beta_{\cdot k}$ and $\beta_{\cdot m}$ when the weight of the edge connecting them is large. This problem is known as the graph-guided fused lasso [Chen et al. (2010), Kim, Sohn and Xing (2009), Kim and Xing (2009)] and

is given by

$$\min_{B} \frac{1}{n} \|Y - XB\|_F^2 + \lambda \|B\|_1$$

(2)

$$+ \gamma \sum_{k,m} |g_{km}| \cdot \|\beta_{\cdot k} - \text{sgn}(g_{km})\beta_{\cdot m}\|_1.$$

Here the ℓ_1 norm penalty again encourages sparsity in the estimated coefficient matrix. In contrast, the second penalty term, known as a graph-guided fusion penalty, encourages similarity among the regression parameters for all pairs of outputs. The weight of each term in the fusion penalty is dictated by $|g_{km}|$, which encodes the strength of the relationship between y_k and y_m. Furthermore, the sign of g_{km} determines whether to encourage a positive or negative relationship between parameters; if $g_{km} > 0$ (i.e., genes y_k and y_m are positively correlated), then we encourage $\beta_{\cdot k}$ to be equal to $\beta_{\cdot m}$, but if $g_{km} < 0$ (i.e., genes y_k and y_m are negatively correlated), we encourage $\beta_{\cdot k}$ to be equal to $-\beta_{\cdot m}$. If $g_{km} = 0$, then genes y_k and y_m are unrelated, and so we don't fuse their respective regression coefficients.

3.3. *Sparse inverse covariance estimation.* In the graph-guided fused lasso model defined in (2), the graph G must be known ahead of time. However, it is also possible to learn a network over the set of genes. One way to do this is to estimate their pairwise conditional independence relationships. If we assume $y \sim \mathcal{N}(\mu, \Sigma)$, where we let $\mu = 0$ for simplicity, then these conditional independencies are encoded in the inverse covariance matrix, or precision matrix, defined as $\Theta = \Sigma^{-1}$. We can obtain a sparse estimate of the precision matrix using the graphical lasso [Friedman, Hastie and Tibshirani (2008)] given by

(3)

$$\min_{\Theta} \frac{1}{n} \text{tr}(Y^T Y \Theta) - \log \det(\Theta) + \lambda \|\Theta\|_1.$$

This objective is again derived from the penalized negative log likelihood of a Gaussian distribution, where this time the ℓ_1 penalty term encourages sparsity among the entries of the precision matrix.

4. Method. We now introduce our approach for jointly estimating the coefficients in a multiple-output regression problem and the edges of a network over the regression outputs. We apply this technique to the problem of simultaneously learning an eQTL map and a gene regulatory network from genome (SNP) data and transcriptome (gene expression) data. Although we focus exclusively on this application, the same problem formulation appears in other domains as well.

4.1. *A joint regression and network estimation model.* Given SNPs $x \in \mathbb{R}^p$ and genes $y \in \mathbb{R}^q$, in order to jointly model the n-by-p regression parameter matrix

B and the q-by-q inverse covariance matrix Θ, we begin with two core modeling assumptions:

$$x \sim \mathcal{N}(0, T), \tag{4}$$

$$y \mid x \sim \mathcal{N}(x^T B, E), \tag{5}$$

where T is the covariance of x and E is the conditional covariance of $y \mid x$. Given the above model, we can also derive the marginal distribution of y. To do this, we first use the fact that the marginal distribution $p(y)$ is Gaussian.[3] We can then derive the mean and covariance of y, as follows:

$$\mathbb{E}_y(y) = \mathbb{E}_x\left(\mathbb{E}_{y|x}(y|x)\right) = 0;$$

$$\mathbb{C}\mathrm{ov}_y(y) = \mathbb{E}_x\left(\mathbb{C}\mathrm{ov}_{y|x}(y|x)\right) + \mathbb{C}\mathrm{ov}_x\left(\mathbb{E}_{y|x}(y|x)\right) = E + B^T T B.$$

Using these facts, we conclude that the distribution of y is given by

$$y \sim \mathcal{N}(0, \Theta^{-1}), \tag{6}$$

where $\Theta^{-1} = E + B^T T B$ denotes the marginal covariance of y. This allows us to explicitly relate Θ, the inverse covariance of y, to B, the matrix of regression parameters. Lastly, we simplify our model by assuming $T = \tau^2 I_{p \times p}$ and $E = \varepsilon^2 I_{q \times q}$. With this change, the relationship between B and Θ^{-1} can be summarized as $\Theta^{-1} \propto B^T B$ because B is now the only term that contributes to the off-diagonal entries of Θ and hence to the inverse covariance structure among the genes.[4]

4.2. *Estimating model parameters with a fusion penalty.*

Now that we have a model that captures B and Θ, we want to jointly estimate these parameters from the data while encouraging the relationship $\Theta^{-1} \propto B^T B$. To do this, we formulate our model as a convex optimization problem with an objective function of the form

$$\mathrm{loss}_{y|x}(B) + \mathrm{loss}_y(\Theta) + \mathrm{penalty}(B, \Theta), \tag{7}$$

where $\mathrm{loss}_{y|x}(B)$ is a loss function derived from the negative log likelihood of $y \mid x$, $\mathrm{loss}_y(\Theta)$ is a loss function derived from the negative log likelihood of y, and $\mathrm{penalty}(B, \Theta)$ is a penalty term that encourages shared structure between the estimates of B and Θ.

Given n i.i.d. observations of x and y, let X be a matrix that contains one observation of x per row and let Y be a matrix that contains one observation of y per

---

[3] See equation B.44 of Appendix B in Bishop (2006).

[4] Although we make this simplifying assumption in our model, we later demonstrate via simulation experiments that ICLasso still performs well in practice when these constraints are violated, namely when the dimensions of x are not independent and the dimensions of y have residual covariance structure after the effect of $x^T B$ is removed.

row. Then we define the inverse-covariance-fused lasso (ICLasso) optimization problem as

$$\min_{B,\Theta} \frac{1}{n}\|Y - XB\|_F^2 + \frac{1}{n}\operatorname{tr}(Y^T Y \Theta) - \log\det(\Theta)$$

(8)
$$+ \lambda_1\|B\|_1 + \lambda_2\|\Theta\|_1$$

$$+ \gamma \sum_{k,m} |\theta_{km}| \cdot \|\beta_{\cdot k} + \operatorname{sgn}(\theta_{km})\beta_{\cdot m}\|_1.$$

From a statistical perspective, the above formulation is unusual because we aim to simultaneously optimize the marginal and conditional likelihood functions of y. However, when we consider it simply as an optimization problem and divorce it from the underlying model, we see that it boils down to a combination of the objectives from the multiple-output lasso and the graphical lasso problems, with the addition of a graph-guided fused lasso penalty to encourage transfer learning between the estimates of B and Θ.

When Θ is fixed, our objective reduces to the graph-guided fused lasso with the graph given by $G = -\Theta$. When B is fixed, our objective reduces to a variant of the graphical lasso in which the ℓ_1 norm penalty has a different weight for each element of the inverse covariance matrix, that is, the standard penalty term $p(\Theta) = \lambda \sum_{k,m} |\theta_{km}|$ is replaced by $p(\Theta) = \sum_{k,m} w_{km}|\theta_{km}|$ where the weights are given by $w_{km} = \lambda_2 + \gamma\|\beta_{\cdot k} + \operatorname{sgn}(\theta_{km})\beta_{\cdot m}\|$.

We further deconstruct the ICLasso objective by describing the role of each term in the model:

- The first term $\frac{1}{n}\|Y - XB\|_F^2$ is the regression loss, and is derived from the conditional log likelihood of $y \mid x$. Its role is to encourage the coefficients B to map X to Y, that is, to obtain a good estimate of the eQTL map from the data.
- The second term $\frac{1}{n}\operatorname{tr}(Y^T Y \Theta) - \log\det(\Theta)$ is the inverse covariance loss, and is derived from the marginal log likelihood of y. Its role is to encourage the network Θ to reflect the partial correlations among the outputs, that is, to obtain a good estimate of the gene network from the data.
- The third term $\lambda_1\|B\|_1$ is an ℓ_1 norm penalty over the matrix of regression coefficients that induces sparsity among the SNP–gene interactions encoded in B.
- The fourth term $\lambda_2\|\Theta\|_1$ is an ℓ_1 norm penalty over the precision matrix that induces sparsity among the gene–gene interactions encoded in Θ.
- The final term $\gamma \sum_{k,m} |\theta_{km}| \cdot \|\beta_{\cdot k} + \operatorname{sgn}(\theta_{km})\beta_{\cdot m}\|_1$ is a graph-guided fusion penalty that encourages similarity between the coefficients of closely related outputs; specifically, when genes y_k and y_m have a positive partial correlation, it fuses β_{jk} towards β_{jm} for all SNPs x_j, and when genes y_k and y_m have a negative partial correlation, it fuses β_{jk} towards $-\beta_{jm}$ for all SNPs x_j.[5]

---

[5]Note that θ_{km} is negatively proportional to the partial correlation between y_k and y_m, meaning that a negative value of θ_{km} indicates a positive partial correlation and vice versa [see, e.g., Peng,

In the above objective, the loss functions come directly out of the modeling assumptions given in (5) and (6). The sparsity-inducing ℓ_1 norm penalties make estimation feasible in the high-dimensional setting where $p, q > n$, and contribute to the interpretability of the eQTL map and gene network.

4.3. *Sparse structure in B and Θ.* In this section, we describe how the ICLasso model captures sparse structure that is shared between the eQTL map B and the gene network Θ, and in doing so enables transfer learning.

We first show that the graph-guided fused lasso penalty encourages the structure $\Theta^{-1} \propto B^T B$, thereby linking the two estimates. Consider the optimization problem $\hat{\Theta} = \arg\min_\Theta f(\Theta) \equiv \operatorname{tr}(B^T B \Theta) - \log\det(\Theta)$. We can solve this problem in closed form by taking the gradient $\nabla_\Theta f(\Theta) = B^T B - \Theta^{-1}$ and setting it to 0, which yields the solution $\hat{\Theta}^{-1} = B^T B$. This suggests that the penalty $\operatorname{tr}(B^T B \Theta)$ encourages the desired structure, while the log determinant term enforces the constraint that Θ be positive semidefinite, which is necessary for Θ to be a valid inverse covariance matrix.

However, instead of directly using this penalty in our model, we demonstrate that it encourages similar structure as the graph-guided fused lasso penalty. We compare the trace penalty, denoted TRP, and the graph-guided fused lasso penalty, denoted GFL, below.

$$(9) \qquad \operatorname{TRP}(B, \Theta) = \operatorname{tr}(B^T B \Theta) = \sum_{k=1}^{q} \sum_{m=1}^{q} \theta_{km} \cdot \beta_{\cdot k}^T \beta_{\cdot m},$$

$$(10) \qquad \operatorname{GFL}(B, -\Theta) = \sum_{k=1}^{q} \sum_{m=1}^{q} |\theta_{km}| \cdot \| \beta_{\cdot k} + \operatorname{sgn}(\theta_{km}) \beta_{\cdot m} \|_1.$$

We show that these penalties are closely related by considering three cases.

- When $\theta_{km} = 0$, the relevant terms in both TRP and GFL go to zero. In this case, nothing links $\beta_{\cdot k}$ and $\beta_{\cdot m}$ in either penalty.
- When $\theta_{km} < 0$, the relevant term in TRP is minimized when $\beta_{\cdot k}^T \beta_{\cdot m}$ is large and positive, which occurs when $\beta_{\cdot k}$ and $\beta_{\cdot m}$ point in the same direction. Similarly, the corresponding term in GFL is minimized when $\beta_{\cdot k} = \beta_{\cdot m}$. In this case, both penalties encourage similarity between $\beta_{\cdot k}$ and $\beta_{\cdot m}$ with strength proportional to the magnitude of θ_{km}.
- When $\theta_{km} > 0$, the relevant term in TRP is minimized when $\beta_{\cdot k}^T \beta_{\cdot m}$ is large and negative, which occurs when $\beta_{\cdot k}$ and $\beta_{\cdot m}$ point in opposite directions. Similarly, the corresponding term in GFL is minimized when $\beta_{\cdot k} = -\beta_{\cdot m}$. In this case, both penalties encourage similarity between $\beta_{\cdot k}$ and $-\beta_{\cdot m}$ with strength proportional to the magnitude of θ_{km}.

Zhou and Zhu (2009)]. This explains why the sign is flipped in the fusion penalty in (8) relative to the one in (2).

We choose to use the graph-guided fused lasso penalty instead of the trace penalty because it more strictly enforces the relationship between B and Θ^{-1} by fusing the regression parameter values of highly correlated genes.

Next, we describe a set of conditions under which our assumptions on B and Θ are compatible with one another. Although we do not provide theoretical guarantees on what type of structure will be learned by our method, we illustrate via a toy example that certain biologically realistic scenarios will naturally lead to sparsity in both B and $\Theta = (I + B^T B)^{-1}$.

Consider a gene network that is organized into a set of densely connected subnetworks corresponding to functional gene modules (e.g., pathways). In this case, we might expect the true Θ to be block diagonal, meaning that there exist blocks $C_1, \ldots, C_d$ such that any pair of genes belonging to two different blocks are not connected in the gene network, that is, $\theta_{km} = 0$ for any $y_k \in C_a$ and $y_m \notin C_a$. Furthermore, suppose our central assumption on the relationship between B and Θ is satisfied, namely genes that are linked in the gene network are also associated with similar sets of SNPs in the eQTL map. Then we might expect that any pair of genes belonging to the same block will have the same SNP–gene associations, that is, $\beta_{jk} = \beta_{jm} \ \forall j$ for any $y_k, y_m \in C_a$. Since we also assume that the true B is sparse, this would lead to a block sparse pattern in B in which each gene module is associated with only a subset of the SNPs.

A simple example of this type of sparse structure is shown in Figure 1. Note that such a pattern in B would lead to block diagonal structure in $\Sigma = I + B^T B$ that preserves the blocks defined by $C_1, \ldots, C_d$. Furthermore, since the inverse of a block diagonal matrix is also block diagonal with the same blocks, this implies that $\Theta = \Sigma^{-1}$ will be block diagonal as well.

This provides an example of a scenario that occurs naturally in biological networks and satisfies our modeling assumptions. However, we note that our model is flexible enough to handle other types of sparse structure as well. In fact, one of the main advantages of our approach is that the sparsity pattern is learned from the data rather than specified in advance.
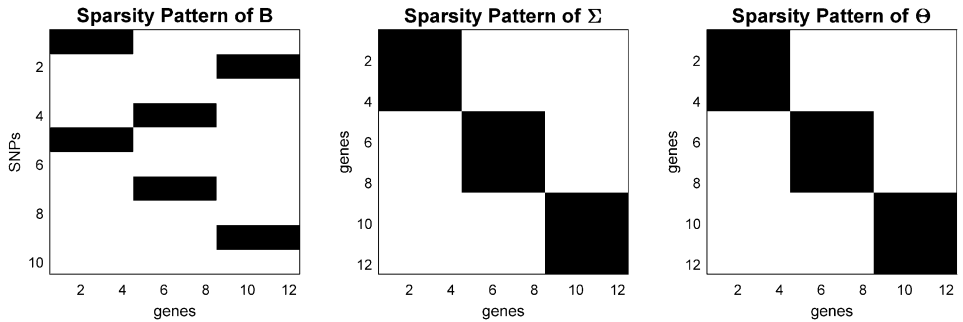

FIG. 1.   *A toy example with 10 SNPs and 12 genes grouped into 3 modules. When B exhibits a certain type of sparse structure, $\Sigma = I + B^T B$ and $\Theta = \Sigma^{-1}$ will also be sparse.*

4.4. *Relationship to other methods.* There are currently two existing approaches that jointly estimate regression coefficients and network structure: multivariate regression with covariance estimation (MRCE), from Rothman, Levina and Zhu (2010), and conditional Gaussian graphical models (CGGM), originally from Sohn and Kim (2012) and further developed by Wytock and Kolter (2013) and Yuan and Zhang (2014). In this section, we describe how our approach differs from these others.

All three methods, including ours, assume that the inputs X and outputs Y are related according to the basic linear model $Y = XB + E$, where E is a matrix of Gaussian noise. However, each approach imposes a different set of additional assumptions on top of this, which we discuss below.

MRCE: This method assumes that $E \sim \mathcal{N}(0, \Omega^{-1})$, which leads to $Y \mid X \sim \mathcal{N}(XB, \Omega^{-1})$. MRCE estimates B and Ω by solving the following objective:

$$(11) \quad \min_{B,\Omega} \frac{1}{n} \mathrm{tr}\big((Y - XB)^T (Y - XB)\, \Omega\big)$$

$$- \log \det(\Omega) + \lambda_1 \|B\|_1 + \lambda_2 \|\Omega\|_1.$$

It is important to note that Ω is the conditional inverse covariance of $Y \mid X$, which actually corresponds to the inverse covariance of the noise matrix E rather than the inverse covariance of the output matrix Y. We therefore argue that Ω doesn't capture any patterns that are shared with the regression coefficients B, since by definition Ω encodes the structure in Y that cannot be explained by XB.

CGGM: This approach makes an initial assumption that X and Y are jointly Gaussian with the following distribution:

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \Gamma & \Lambda \\ \Lambda^T & \Omega \end{bmatrix} \right).$$

In this formulation, the distribution of $Y \mid X$ is given by $\mathcal{N}(-X\Lambda\Omega^{-1}, \Omega^{-1})$. This corresponds to the reparameterization of B as $-\Lambda\Omega^{-1}$, where Ω is the conditional inverse covariance matrix and Λ represents the "direct" influence of X on Y. CGGM estimates Λ and Ω by solving the following optimization problem, where sparsity penalties are applied to Λ and Ω instead of B and Ω as was the case in the previous model:

$$(12) \quad \min_{\Lambda,\Omega} \frac{1}{n} \mathrm{tr}\big((Y + X\Lambda\Omega^{-1})^T (Y + X\Lambda\Omega^{-1})\, \Omega\big)$$

$$- \log \det(\Omega) + \lambda_1 \|\Lambda\|_1 + \lambda_2 \|\Omega\|_1.$$

Here the meaning of Ω has not changed, and it once again represents the inverse covariance of the noise matrix.

ICLasso: Our method implicitly assumes two underlying models: $Y \mid X \sim \mathcal{N}(XB, I)$ and $Y \sim \mathcal{N}(0, \Theta^{-1})$. In this case, Θ represents the marginal inverse

covariance of Y rather than the conditional inverse covariance of $Y \mid X$, which was captured by Ω in (11) and (12). The optimization problem in (8) is obtained by combining the loss functions derived from the log likelihood of each model and then incorporating sparsity penalties over B and Θ and an additional fusion penalty to encourage shared structure.

Both MRCE and CGGM have two important drawbacks that are not shared by our approach. First, both of these methods estimate Ω, the precision matrix of the noise term, rather than Θ, the precision matrix of the outputs Y. Second, neither method incorporates a structured sparsity penalty that explicitly encourages shared structure between the network and the regression coefficients. In fact, it would not make sense for these methods to apply a joint penalty over B and Ω because, as discussed above, we wouldn't expect these parameters to have any shared structure. By comparison, our method learns the true output network Θ and uses a graph-guided fused lasso penalty to explicitly encourage outputs that are closely related in Θ to have similar parameter values in B.

4.5. *Optimization via alternating minimization.* Finally, we present an efficient algorithm to solve the inverse-covariance-induced fused lasso problem defined in (8). We start by rewriting the fusion penalty as follows:

$$
\begin{aligned}
\text{GFL}(B, -\Theta) &= \gamma \sum_{k,m} |\theta_{km}| \cdot \| \beta_{\cdot k} + \text{sgn}(\theta_{km})\beta_{\cdot m} \|_1 \\
&= \gamma \sum_{k,m} \max\{\theta_{km}, 0\} \cdot \| \beta_{\cdot k} + \beta_{\cdot m} \|_1 \\
&\quad + \gamma \sum_{k,m} \max\{-\theta_{km}, 0\} \cdot \| \beta_{\cdot k} - \beta_{\cdot m} \|_1,
\end{aligned}
$$

from which it is clear that GFL is biconvex in B and Θ. Thus, upon defining

$$
g(B) = \frac{1}{n} \| Y - XB \|_F^2 + \lambda_1 \| B \|_1,
$$

$$
h(\Theta) = \frac{1}{n} \text{tr}(Y^T Y \Theta) - \log \det(\Theta) + \lambda_2 \| \Theta \|_1,
$$

we can rewrite the original optimization problem as

$$
(13) \qquad \min_{B,\Theta} \; g(B) + h(\Theta) + \text{GFL}(B, -\Theta).
$$

Here $g(B)$ is the usual lasso formulation in (1), $h(\Theta)$ is the usual graphical lasso formulation in (3), and the graph-guided fusion penalty couples the two problems. Since GFL is biconvex, we can solve the joint problem (13) using an alternating minimization strategy. We describe how to solve each convex sub-problem in the Supplementary Material [Marchetti-Bowick et al. (2019)].

5. Experiments. In this section, we present the results from a series of experiments on both synthetic and real data. We compare our method to several baselines and demonstrate that it achieves better recovery of the underlying structure of B and Θ than existing methods.

5.1. *Simulation study.* We begin by evaluating our model on synthetic data so that we can directly measure how accurately the sparse structure of the eQTL map and the gene network are recovered. We compare our eQTL map estimates $\hat{B}$ to several baselines, including traditional pairwise linear regression (LinReg), standard multi-task lasso (Lasso), graph-guided fused lasso using a sparse covariance matrix as its graph (GFLasso1), graph-guided fused lasso using a sparse precision matrix as its graph (GFlasso2), sparse multivariate regression with covariance estimation (MRCE), and the conditional Gaussian graphical model (CGGM). We compare our network estimates $\hat{\Theta}$ to a traditional pairwise correlation network (Corr) and the graphical lasso (GLasso). A detailed explanation of how we select hyperparameter values for all methods is provided in the Supplementary Material [Marchetti-Bowick et al. (2019)].

In our synthetic data experiments, we focus on recovering block-structured networks in which the genes are divided into a set of modules, or groups. In order to generate data according to our model, we assume that the genes within each module only regulate one another and are associated with the same set of eQTLs. Specifically, this means that if genes k and m belong to the same module, we will have $\theta_{km} \neq 0$ and $\beta_{\cdot k} \approx \beta_{\cdot m}$. Although we focus on this data setting because it makes intuitive biological sense and satisfies our modeling assumptions, we note that our approach is flexible enough to handle other types of structure among the SNPs and genes.

An example of one of our synthetic datasets is shown in Figure 2. The ground truth for both B and Θ is given in the far left panel. The next three columns show the estimated values of B for three competing methods, and the results of our method are shown on the far right. In this example, the drawbacks of each of the baseline methods are evident. The covariance matrix used for the network structure in GFLasso1 captures many spurious patterns in Y that don't correspond to true patterns in the regression map, which confuses the estimate of B. The precision matrix used for the network structure in GFLasso2 does not accurately capture the true inverse covariance structure because of the low signal-to-noise ratio in Y. This prevents the fusion penalty from effectively influencing the estimate of B. Finally, although CGGM gets a reasonable estimate of the network, despite the fact that it learns the conditional inverse covariance Ω instead of the marginal inverse covariance Θ, this structure is not explicitly enforced in B, which still leads to a poor estimate of the regression parameters. In contrast, the cleanest estimate of both $\hat{B}$ and $\hat{\Theta}$ comes from ICLasso.

The complete procedure we use for generating synthetic data is described in the Supplementary Material [Marchetti-Bowick et al. (2019)]. To investigate the behavior of our method under a wide range of conditions, we compare four different
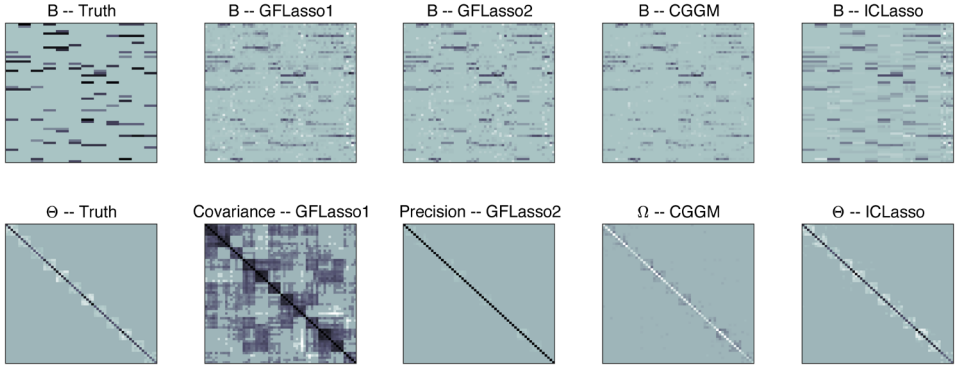
FIG. 2. *A qualitative comparison of results on a single synthetic dataset with $p = 60$ and $q = 60$. The far left panel contains the ground truth for B and Θ. The remaining panels show the estimates of the regression coefficients for each method (top) along with the graph structure that was used or estimated by the method (bottom).*

covariance settings. In the first setting, we maintain our modeling assumption that both T and E are are proportional to the identity matrix. In the second setting, we keep $E \propto I_{q \times q}$ but assume that T has some off-diagonal covariance structure. This captures a situation in which the SNP genotypes are correlated, which could easily arise from linkage disequilibrium. In the third setting, we keep $T \propto I_{p \times p}$ but assume that E has some off-diagonal covariance structure. This captures a situation in which the residual gene expression values are correlated even after removing SNP effects, which might come from environmental or epigenetic factors. Finally, in the fourth setting, we assume that both T and E are nonidentity.

The main results of our synthetic experiments are shown in Figure 3. Within each covariance setting, we jointly vary the number of SNPs and genes, keeping the ratio fixed. We evaluate our approach according to three metrics. In the top two rows, we show the F1 score on the recovery of the true nonzero elements of B and Θ, respectively. This reflects the ability of each method to learn the correct structure of the eQTL map and the gene network. In the bottom row, we show the prediction error of Y on an out-of-sample test set. We note that this test set is completely separate from both the training set (used to estimate the model parameters) and the validation set (used to select the best values of the hyperparameters). Our results show that ICLasso achieves the best performance even when we violate our modeling assumptions by introducing covariance among the SNPs, conditional covariance among the genes, or both.

We also perform a second experiment in which we vary the density of B, the density of Θ, and the number of SNPs while keeping the number of genes fixed. These results are shown in Supplementary Figure S1 [Marchetti-Bowick et al. (2019)]. Overall, our results clearly demonstrate that ICLasso outperforms all baselines in nearly all of the settings we consider.
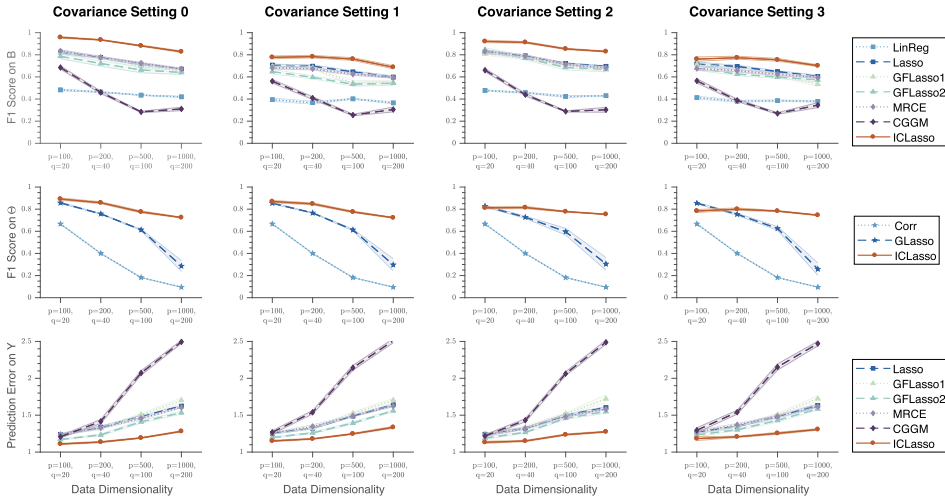
FIG. 3. *A comparison of results on synthetic data generated with each of the four different types of covariance structure and with several different values of p and q. We fix the group size to $g = 10$ and the number of SNP associations per gene to $s = 5$. All results are averaged over 20 simulations, and the shaded regions show the standard error.*

Finally, we perform a timing experiment to analyze the computational complexity of our approach. These results are shown in Supplementary Figure S2 [Marchetti-Bowick et al. (2019)]. We show how the run time grows as a function of both p and q. Although our method is the slowest amongst all baselines for large values of q, it scales nicely in p. We argue that the cost of the extra computation time is a worthwhile tradeoff for the boost in accuracy that ICLasso provides.

5.2. *Yeast eQTL study.* In order to evaluate our approach in a real-world setting and provide a proof of concept for our model, we applied ICLasso to a yeast eQTL dataset from Brem and Kruglyak (2005) that consists of 2,956 SNP genotypes and 5,637 gene expression measurements across 114 yeast samples. To preprocess the data, we removed SNPs with duplicate genotypes and retained only the 25% of genes with the highest variance in expression, leaving $p = 1,157$ SNPs and $q = 1,409$ genes in our analysis.

We used our approach to jointly perform eQTL mapping and gene network inference on the yeast dataset, treating the SNPs as inputs X and the genes as outputs Y. We trained our model on 91 samples and used the remaining 23 samples as a validation set for tuning the hyperparameters. Given the trained model, we read the eQTL associations from the regression coefficient matrix $\hat{B}$, which encodes SNP–gene relationships, and obtained the gene network from the inverse covariance matrix $\hat{\Theta}$, which encodes gene–gene relationships. In addition to ICLasso, we ran Lasso and GFlasso on the yeast data to obtain two additional estimates of B,

	Density	Training error	Validation error
Lasso	1.65%	0.502	0.718
GFLasso	2.87%	0.392	0.715
ICLasso	6.88%	0.395	0.703

and ran GLasso1 to obtain another estimate of Θ. Note that we chose not to compare to MRCE and CGGM because these methods performed worse than the other baselines in the most realistic data settings that we tested in our simulation experiments. Furthermore, we did not compare to GFLasso2 because the performance of the two variants of GFLasso that we evaluated were comparable.

Table 1 shows the density of $\hat{B}$ obtained with each method, along with the prediction error of Y on the training set and on the held-out validation set, which were calculated using $\|Y_{\text{train}} - X_{\text{train}}\hat{B}\|_F^2$ and $\|Y_{\text{valid}} - X_{\text{valid}}\hat{B}\|_F^2$, respectively. We chose not to sacrifice any data for a test set, but these results indicate that ICLasso achieves an equivalent or better fit to the training and validation sets than Lasso and GFLasso.

5.2.1. *Quantitative analysis.* Because the true yeast eQTLs and gene network structure are not known, there is no ground truth for this problem. We instead analyzed the output of each method by performing a series of enrichment analyses that together provide a comprehensive picture of the biological coherence of the results. An enrichment analysis uses gene annotations to identify specific biological processes, functions, or structures that are over-represented among a group of genes relative to the full set of genes that is examined [Subramanian et al. (2005)]. To evaluate our yeast data results, we performed three types of enrichment analyses: biological process and molecular function enrichment using annotations from the Gene Ontology (GO) database [Ashburner et al. (2000)] and pathway enrichment using annotations from the Kyoto Encyclopedia of Genes and Genomes (KEGG) database [Kanehisa and Goto (2000)]. We used a hypergeometric test to compute a p-value for each term, and then adjusted these values to account for multiple hypothesis testing. Significance was determined using an adjusted p-value cutoff of 0.01.

We first analyzed $\hat{B}$ by performing a per-SNP enrichment analysis. For each SNP j, we used the nonzero elements in $\beta_{j\cdot}$ to identify the set of genes associated with the SNP. Next we performed GO and KEGG enrichment analyses on this group of genes by comparing their annotations to the full set of 1409 genes that we included in our study. We repeated this procedure for each SNP, and calculated the total number of terms that were enriched over all SNPs to obtain a global measure of enrichment for $\hat{B}$. In addition, we calculated the total number of SNPs that

TABLE 2
GO and KEGG enrichment analysis on yeast eQTL map

	Number of enriched terms			Avg.	Number of enriched SNPs			Avg.
	GO-BP	GO-MF	KEGG	change	GO-BP	GO-MF	KEGG	change
Lasso	1862	804	205	—	198	132	127	—
GFLasso	3499	1528	312	+**77**%	286	211	172	+**47**%
ICLasso	8046	3147	1025	+**155**%	590	453	441	+**126**%

were enriched for at least one term in each category. These results are summarized in Table 2. It is evident that ICLasso outperforms both GFLasso and Lasso on estimating the regression coefficients, since it has more than twice as many enriched terms in GO biological process, GO molecular function, and KEGG than either baseline.

Next we used a similar approach to evaluate the structure present in $\hat{\Theta}$. We first obtained groups of genes by using spectral clustering to perform community detection among the genes using the inferred network structure. After clustering the genes into 100 groups,[6] we performed GO and KEGG enrichment analyses on each cluster and calculated the total number of enriched terms along with the total number of clusters that were enriched for at least one term. These results are summarized in Table 3. Once again, our approach has more enrichment than the baseline in every category, which implies that the gene network estimated by ICLasso has a much more biologically correct structure than the network estimated by GLasso.

5.2.2. *Qualitative analysis.* The quantitative results in Tables 2 and 3 indicate that, compared to other methods, our approach identifies more eQTLs that are associated with genes significantly enriched in certain biological processes and pathways. A more detailed examination of our results revealed that many of the

TABLE 3
GO and KEGG enrichment analysis on yeast gene network

	Number of enriched terms			Avg.	Number of enriched ckusters			Avg.
	GO-BP	GO-MF	KEGG	change	GO-BP	GO-MF	KEGG	change
GLasso	173	77	31	—	14	12	11	—
ICLasso	321	127	41	+**61**%	29	26	22	+**108**%

---

[6]We also clustered with 25, 50, and 200 groups and obtained similar results.

enriched terms correspond to metabolic pathways, and that the eQTLs we identified agree with those discovered in a previous study that analyzed the effect of genetic variations on the yeast metabolome.

Breunig et al. (2014) identified the metabolite quantitative trait loci (mQTLs) for 34 metabolites and then examined each mQTL for the presence of metabolic genes in the same pathway as the linked metabolite. We found that 10 of these 34 metabolites were linked to metabolic genes where our identified eQTLs reside. For example, Breunig et al. determined that the metabolite valine is linked to an mQTL in a region spanned by the ILV6 gene, which encodes a protein involved in valine biosynthesis. In our study, we also identified an eQTL located in ILV6. Moreover, we found that the eQTL in ILV6 is associated with 365 genes that are significantly enriched for pathways involved in the metabolism and biosynthesis of various amino acids. This is consistent with the fact that the metabolism and biosynthesis of amino acids in the cell needs to be coordinated.

Furthermore, our enrichment analysis shows that the eQTL-associated genes we identified are enriched for various metabolic pathways (e.g. sulfur, riboflavin, protein, starch, and sucrose metabolism; oxidative phosphorylation; glycolysis), as well as more general pathways, such as cell cycle pathways and MAPK pathways. This is consistent with the roles of the mQTLs identified by Breunig et al. Interestingly, among these genes, SAM1, encoding an S-adenosylmethionine synthetase, is also among the eQTLs in our list. Our results show that the eQTL we found in SAM1 is associated with 252 genes that are enriched for cytoplasmic translation and ribosome functions, consistent with the fact that SAM is the methyl donor in most methylation reactions and is essential for DNA methylation of proteins, nucleic acids, and lipids [Roberts and Selker (1995)].

Finally, to illustrate our results, we visualized the SNP–gene associations discovered by GFLasso and ICLasso by plotting a binary heatmap of the two estimates of B in Figure 4. Within each heatmap, both the SNPs and genes are sorted

$\hat{B}$ estimated with GFlasso $\hat{B}$ estimated with ICLasso
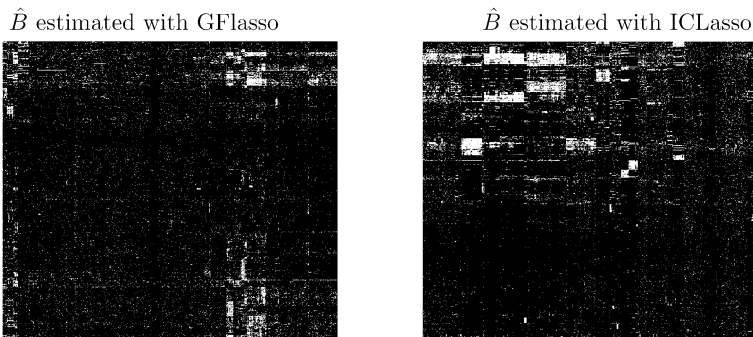
FIG. 4. *Binary heatmap of associations between SNPs (one per row) and genes (one per column), estimated with GFLasso and ICLasso. In each image, the SNPs and genes are ordered to maximize the visual clustering of associations.*

TABLE 4
Known Alzheimer's disease genes

Gene symbol

APP, APOE, PLD3, TREM2, SORL1, GAB2, BIN1, CLU, CD33, CR1, PICALM, ABCA7, CD2AP, MS4A6A, MS4A4E

to maximize the clustering of associations. From these plots, it is clear that the associations discovered by ICLasso contain more interesting block structure than those discovered by GFLasso.

5.3. *Human eQTL study.* Lastly, we applied our method to a human eQTL dataset in order to identify a set of interesting genomic loci that may play a role in Alzheimer's disease. For this study, we used a dataset from Zhang et al. (2013) that contains $n = 540$ case and control samples of patients with Alzheimer's disease, genotypes of $p = 555,091$ SNPs across all chromosomes, and mRNA expression values of $q = 40,638$ gene probes measured in the cerebellum, a region of the brain that governs motor control and some cognitive functions.

We preprocessed this data by selecting a subset of interesting SNPs and genes to include in our analysis. To filter genes, we calculated the marginal variance of the expression of each gene, the fold change in each gene's expression between the case and control samples, and the p-value of a t-test with the case-control status. We then selected all genes with variance in the top 10%, fold change in the top 10%, or p-value in the bottom 10%, along with a set of 15 genes known to be associated with Alzheimer's disease. These genes are listed in Table 4. To filter SNPs, we calculated the p-value of a chi-square test with the case-control status. We then selected all SNPs with uncorrected p-value < 0.05, along with all SNPs located within 500 kb of any of the Alzheimer's genes. This filtering yielded $p = 24,643$ SNPs and $q = 9,692$ genes.

Applying ICLasso to this dataset yielded an estimate of $\hat{B}$ with 4.07% density and an estimate of $\hat{\Theta}$ with 1.70% density. To analyze the results, we first constructed a set of candidate SNPs comprised of the top 10 SNPs associated with each of the Alzheimer's genes based on association strength. Since some genes are represented by multiple probes in the dataset, there are 25 gene expression values corresponding to the 15 Alzheimer's genes. From these, we identified 185 candidate eQTLs for further analysis.

5.3.1. *Enrichment analysis.* We performed an enrichment analysis for each of these SNPs by looking at the set of genes linked to each SNP in the eQTL map and determining whether these are enriched for any GO biological process terms relative to the full universe of 9,692 genes. Among these, 58 (31%) are enriched for at least one term using a corrected p-value cutoff of 0.01. When analyzing the

results, we noticed three categories of candidate eQTLs that might play a role in Alzheimer's disease: SNPs associated with genes enriched for immune functions, SNPs associated with genes enriched for metabolic functions, and SNPs associated with genes enriched for neural functions. A selected set of interesting results from each of these categories are highlighted in Supplementary Tables S1, S2 and S3 [Marchetti-Bowick et al. (2019)]. One particularly interesting observation is that many of the SNPs in the first category are associated with genes implicated in myeloid cell activated immune response. This is notable because Alzheimer's disease has previously been linked to acute myeloid leukemia [Malik et al. (2015)].

5.3.2. *Comparison to baselines.* Next, we compared our results to the two simple pairwise baselines (LinReg and Corr). These methods are commonly used for QTL mapping and gene network estimation, and are often favored for large datasets due to their efficient run time. For the purposes of this experiment, we selected significance thresholds for the baselines that yielded estimates of B and Θ that had the same density as the estimates produced by ICLasso.

The results of our comparative analysis are summarized in Figure 5. This plot shows the number of SNP associations per gene sorted by the degree of the gene
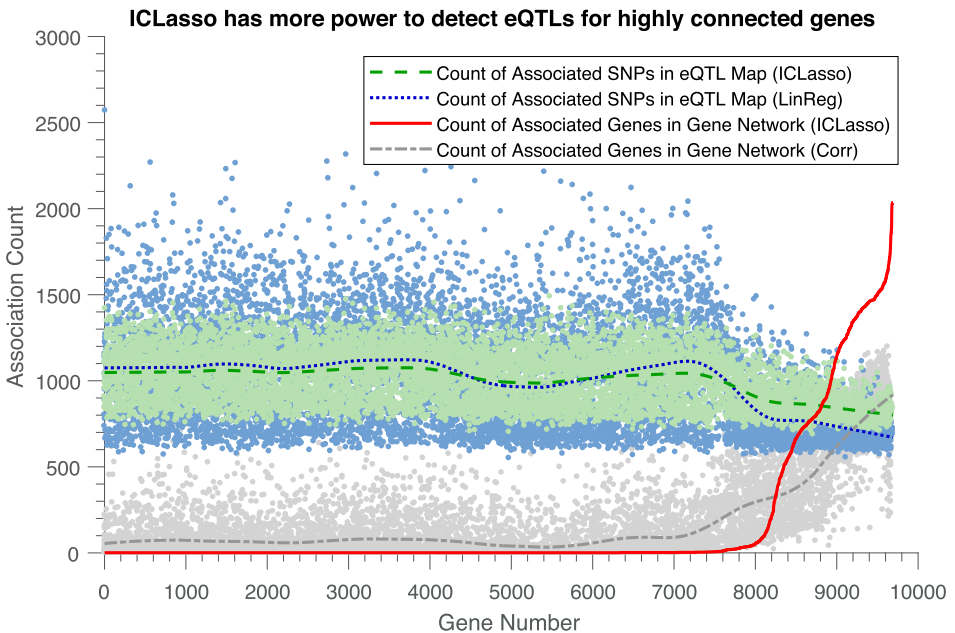

FIG. 5. *A comparison of the results obtained by ICLasso and the pairwise methods on Alzheimer's data. The blue and green lines show the number of SNP associations per gene, and the red and gray lines show the number of gene associations per gene. The genes on the horizontal axis are sorted according to their degree in the network estimated by ICLasso. The three dashed and dotted lines are smoothed versions of the corresponding scatter plots.*

in the network estimated by ICLasso. The trend lines emphasize that ICLasso has more power to detect eQTL associations for genes that are highly connected in the estimated gene network. This result matches our intuition about the structural prior encoded in the ICLasso penalty term, which enables the model to detect SNP–gene associations that exhibit a weak signal in the data by leveraging information about other related genes. We also note that the gene connectivity estimated by ICLasso correlates well with the connectivity estimated by the correlation network.

To provide an additional view of the results, we plotted the distributions of the SNP and gene association counts in Supplementary Figure S3 [Marchetti-Bowick et al. (2019)]. One significant difference between ICLasso and the baseline in the distribution of the number of gene associations (shown in the right panel) is that the ICLasso distribution clearly has 3 modes (one peaked at 0, one peaked at 750, and a third peaked at 1500). This suggests that the ICLasso estimate of the gene network identified at least two large interconnected sub-networks.

5.3.3. *Stability analysis.* Finally, we performed experiments to assess the stability of our results. First, because our eQTL dataset is somewhat unusual in that it contains both case and control samples, we wanted to check whether any of the associations could be due to batch effects in the data. To investigate, we experimented with including a few of the top principal components from the SNP genotypes X and the gene expression values Y as additional covariates in our model. The results of this experiment are given in Supplementary Figure S4 [Marchetti-Bowick et al. (2019)]. They show that including these principal components barely alters the results. In particular, including only one PC from Y yields nearly identical results, and including one PC from X only slightly changes the set of associations. We also experimented with including more than one PC from each dataset, but the results were the same.

Second, because this dataset is very high-dimensional, with p and q both much larger than the sample size n, we wanted to evaluate how sensitive the results are to the set of samples used to fit our model. To do that, we refit the model to several random subsamples of the data and also refit the model using only control samples and only case samples. All of these subsamples have a fixed sample size of $n = 270$. The results of these experiments are summarized in Supplementary Figure S4 [Marchetti-Bowick et al. (2019)]. They show that the set of associations we identify is very sensitive to the set of samples used for training, with only 15% overlap on average between each pair of random samples. This is almost certainly because of the noisiness inherent to high-dimensional datasets. The overlap between the separate case and control estimates is even lower at 3%. We believe this is largely due to the fact that we based our initial filtering on correlation with the case/control status.

Note that in all of the stability results, the overlap in the set of associations given by two estimates of B is measured as the number of associations in the union divided by the number of associations in the intersection.

6. Conclusion. In this work, we propose a new model called the inverse-covariance-induced fused lasso which jointly estimates regression coefficients B and an output network Θ while using a graph-guided fused lasso penalty to explicitly encourage shared structure. Our model is formulated as a biconvex optimization problem, and we derive new, efficient optimization routines for each convex sub-problem based on existing methods.

Our results on both synthetic and real data unequivocally demonstrate that our model achieves significantly better performance on recovery of the structure of B, recovery of the structure of Θ, and prediction error than all baselines we evaluated. In this paper, we demonstrated that our approach can be used to perform joint eQTL mapping and gene network estimation on a yeast dataset, yielding more biologically coherent results than previous work. However, the same problem setting appears in many different applications, and the inverse-covariance-induced fused lasso model can therefore be used in a wide range of domains.

The primary disadvantage of our proposed method is that it is not scalable in the number of genes. One promising direction for future work would be to explore an approximation in the style of Meinshausen and Bühlmann (2006) that performs neighborhood selection for estimating the gene network instead of solving for the exact value of Θ. Furthermore, the screening rules for the graphical lasso proposed in Danaher, Wang and Witten (2014) can be directly extended to our model, and would likely provide a significant speedup when working with block sparse gene networks.

SUPPLEMENTARY MATERIAL

Supplement to "A penalized regression model for the joint estimation of eQTL associations and gene network structure." (DOI: 10.1214/18-AOAS1186 SUPP; .pdf). We provide a supplementary document [Marchetti-Bowick et al. (2019)] that contains additional details about the optimization algorithm and additional results for both the synthetic and real data experiments.

REFERENCES

ASHBURNER, M., BALL, C. A., BLAKE, J. A., BOTSTEIN, D., BUTLER, H., CHERRY, J. M., DAVIS, A. P., DOLINSKI, K., DWIGHT, S. S., EPPIG, J. T. et al. (2000). Gene ontology: Tool for the unification of biology. *Nat. Genet.* **25** 25–29.

BANERJEE, O., EL GHAOUI, L. and D'ASPREMONT, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *J. Mach. Learn. Res.* **9** 485–516. MR2417243

BANERJEE, S., YANDELL, B. S. and YI, N. (2008). Bayesian quantitative trait loci mapping for multiple traits. *Genetics* **179** 2275–2289.

BARABASI, A.-L. and OLTVAI, Z. N. (2004). Network biology: Understanding the cell's functional organization. *Nat. Rev. Genet.* **5** 101–113.

BISHOP, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, New York. MR2247587

BREM, R. B. and KRUGLYAK, L. (2005). The landscape of genetic complexity across 5700 gene expression traits in yeast. *Proc. Natl. Acad. Sci. USA* **102** 1572–1577.

BREUNIG, J. S., HACKETT, S. R., RABINOWITZ, J. D. and KRUGLYAK, L. (2014). Genetic basis of metabolome variation in yeast. *PLoS Genet.* **10** e1004142.

CHEN, X., KIM, S., LIN, Q., CARBONELL, J. G. and XING, E. P. (2010). Graph-structured multitask regression and an efficient optimization method for general fused lasso. ArXiv preprint. Available at arXiv:1005.3579.

DANAHER, P., WANG, P. and WITTEN, D. M. (2014). The joint graphical lasso for inverse covariance estimation across multiple classes. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **76** 373–397. MR3164871

DEMPSTER, A. P. (1972). Covariance selection. *Biometrics* **28** 157–175.

FLUTRE, T., WEN, X., PRITCHARD, J. and STEPHENS, M. (2013). A statistical framework for joint eQTL analysis in multiple tissues. *PLoS Genet.* **9** e1003486.

FRIEDMAN, J., HASTIE, T. and TIBSHIRANI, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **9** 432–441.

GARDNER, T. S. and FAITH, J. J. (2005). Reverse-engineering transcription control networks. *Phys. Life Rev.* **2** 65–88.

KANEHISA, M. and GOTO, S. (2000). KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.* **28** 27–30.

KIM, S., SOHN, K.-A. and XING, E. P. (2009). A multivariate regression approach to association analysis of a quantitative trait network. *Bioinformatics* **25** i204–i212.

KIM, S. and XING, E. P. (2009). Statistical estimation of correlated genome associations to a quantitative trait network. *PLoS Genet.* **5** Article ID e1000587.

KIM, S. and XING, E. P. (2012). Tree-guided group lasso for multi-response regression with structured sparsity, with an application to EQTL mapping. *Ann. Appl. Stat.* **6** 1095–1117. MR3012522

LEE, W. and LIU, Y. (2012). Simultaneous multiple response regression and inverse covariance matrix estimation via penalized Gaussian maximum likelihood. *J. Multivariate Anal.* **111** 241–255. MR2944419

MALIK, M., CHILES III, J., XI, H. S., MEDWAY, C., SIMPSON, J., POTLURI, S., HOWARD, D., LIANG, Y., PAUMI, C. M., MUKHERJEE, S. et al. (2015). Genetics of CD33 in Alzheimer's disease and acute myeloid leukemia. *Hum. Mol. Genet.* **24** 3557–3570.

MARCHETTI-BOWICK, M., YU, Y., WU, W. and XING, E. P. (2019). Supplement to "A penalized regression model for the joint estimation of eQTL associations and gene network structure." DOI:10.1214/18-AOAS1186SUPP.

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

MICHAELSON, J. J., ALBERTS, R., SCHUGHART, K. and BEYER, A. (2010). Data-driven assessment of eQTL mapping methods. *BMC Genomics* **11** 502.

NICA, A. C., MONTGOMERY, S. B., DIMAS, A. S., STRANGER, B. E., BEAZLEY, C., BARROSO, I. and DERMITZAKIS, E. T. (2010). Candidate causal regulatory effects by integration of expression QTLs with complex trait genetic associations. *PLoS Genet.* **6** e1000895.

PENG, J., ZHOU, N. and ZHU, J. (2009). Partial correlation estimation by joint sparse regression models. *J. Amer. Statist. Assoc.* **104** 735–746. MR2541591

RAI, P., KUMAR, A. and DAUME, H. (2012). Simultaneously leveraging output and task structures for multiple-output regression. In *Advances in Neural Information Processing Systems* (*NIPS*) 3185–3193.

RAVIKUMAR, P., WAINWRIGHT, M. J., RASKUTTI, G. and YU, B. (2011). High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electron. J. Stat.* **5** 935–980. MR2836766

ROBERTS, C. J. and SELKER, E. U. (1995). Mutations affecting the biosynthesis of S-adenosylmethionine cause reduction of DNA methylation in Neurospora crassa. *Nucleic Acids Res.* **23** 4818–4826.

ROCKMAN, M. V. and KRUGLYAK, L. (2006). Genetics of global gene expression. *Nat. Rev. Genet.* **7** 862–872.

ROTHMAN, A. J., LEVINA, E. and ZHU, J. (2010). Sparse multivariate regression with covariance estimation. *J. Comput. Graph. Statist.* **19** 947–962. MR2791263

SOHN, K.-A. and KIM, S. (2012). Joint estimation of structured sparsity and output structure in multiple-output regression via inverse-covariance regularization. In *International Conference on Artificial Intelligence and Statistics* (*AISTATS*) 1081–1089.

STEPHENS, M. (2013). A unified framework for association analysis with multiple related phenotypes. *PLoS ONE* **8** e65245.

SUBRAMANIAN, A., TAMAYO, P., MOOTHA, V. K., MUKHERJEE, S., EBERT, B. L., GILLETTE, M. A., PAULOVICH, A., POMEROY, S. L., GOLUB, T. R., LANDER, E. S. et al. (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. USA* **102** 15545–15550.

TAN, K. M., LONDON, P., MOHAN, K., LEE, S.-I., FAZEL, M. and WITTEN, D. (2014). Learning graphical models with hubs. *J. Mach. Learn. Res.* **15** 3297–3331. MR3277170

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

WYTOCK, M. and KOLTER, Z. (2013). Sparse Gaussian conditional random fields: Algorithms, theory, and application to energy forecasting. In *International Conference on Machine Learning* (*ICML*) 1265–1273.

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, X.-T. and ZHANG, T. (2014). Partial Gaussian graphical model estimation. *IEEE Trans. Inform. Theory* **60** 1673–1687. MR3168429

ZHANG, Y. and YEUNG, D.-Y. (2010). A convex formulation for learning task relationships in multi-task learning. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence* (*UAI*'10) 733–742. AUAI Press, Arlington, VA.

ZHANG, B., GAITERI, C., BODEA, L.-G., WANG, Z., MCELWEE, J., PODTELEZHNIKOV, A. A., ZHANG, C., XIE, T., TRAN, L., DOBRIN, R. et al. (2013). Integrated systems approach identifies genetic nodes and networks in late-onset Alzheimer's disease. *Cell* **153** 707–720.

M. MARCHETTI-BOWICK
W. WU
E. P. XING
SCHOOL OF COMPUTER SCIENCE
CARNEGIE MELLON UNIVERSITY
PITTSBURGH, PENNSYLVANIA 15213
USA
E-MAIL: micolmb@cs.cmu.edu
         weiwu2@cs.cmu.edu
         epxing@cs.cmu.edu

Y. YU
DAVID R. CHERITON SCHOOL
   OF COMPUTER SCIENCE
UNIVERSITY OF WATERLOO
WATERLOO, ONTARIO N2L 3G1
CANADA
E-MAIL: yaoliang.yu@uwaterloo.ca