BIOINFORMATICS ORIGINAL PAPER
Vol. 25 no. 11 2009, pages 1390–1396
doi:10.1093/bioinformatics/btp177

Systems biology

Network-based multiple locus linkage analysis of expression traits

Wei Pan
Division of Biostatistics, School of Public Health, University of Minnesota, A460 Mayo Building (MMC 303), Minneapolis, MN 55455-0378, USA

Received on November 16, 2008; revised on March 24, 2009; accepted on March 26, 2009
Advance Access publication March 31, 2009
Associate Editor: Jonathan Wren

ABSTRACT

Motivation: We consider the problem of multiple locus linkage analysis for expression traits of genes in a pathway or a network. To capitalize on co-expression of functionally related genes, we propose a penalized regression method that maps multiple expression quantitative trait loci (eQTLs) for all related genes simultaneously while accounting for their shared functions as specified a priori by a gene pathway or network.

Results: An analysis of a mouse dataset and simulation studies clearly demonstrate the advantage of the proposed method over a standard approach that ignores biological knowledge of gene networks.

Contact: weip@biostat.umn.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Genomic locations that control the expression levels of transcripts are called expression quantitative trait loci (eQTL) (Brem *et al.,* 2002; Schadt *et al.,* 2003). In eQTL analysis, treating the expression level of a gene as a quantitative trait, one aims to identify genomic loci associated with the trait; see Kendziorski and Wang (2006) for a recent review on statistical methods. It is likely that a gene's expression is associated with multiple regulators, hence a multiple locus linkage analysis is perhaps most suitable. A straightforward approach is to treat each expression trait independently, then apply a standard QTL mapping technique to each individual trait. For a single trait, a general approach to mapping multiple QTL is essentially to identify non-zero regression coefficients in a regression analysis (Sen and Churchill, 2001). Traditionally, such a variable selection problem can be approached by model comparison via a model selection criterion (Bogdan *et al.,* 2004; Broman and Speed, 2002), or to save computing time, by a sequential, e.g. a stepwise forward, procedure for variable selection (Storey *et al.,* 2005; Zou and Zeng, 2007). In addition to high-computational cost, a severe problem associated with these approaches is the reduced power due to multiple tests. Furthermore, such model selection is separated from parameter estimation and is inherently unstable (Breiman, 1996). Hence, simultaneous variable selection and parameter estimation has become increasingly popular as implemented in either a fully Bayesian (e.g. Xu, 2003; Yi *et al.,* 2003) or a penalization (e.g. Zhang and Xu, 2005) framework for QTL mapping. In spite of their much different implementations, the Bayesian framework and the penalization one are closely related to each other due to the connection between penalization and Bayesian modeling (Tibshirani, 1996). Although the penalized likelihood approach is computationally simpler and faster than the fully Bayesian approach, Zhang and Xu (2005) found that both approaches performed well and similarly in their simulation study for QTL mapping, as confirmed by Xu (2007), in which an empirical Bayesian method with good performance and fast computation was proposed.

As for multivariate traits, due to correlated expressions among the genes, separate analyses on individual expression traits would not be efficient. On the other hand, standard multivariate QTL techniques (Jiang and Zeng, 1995) cannot be directly applied to a relatively small sample (with sample size typically less than or around 100) with a large number of expression traits, which is in the order of thousands; furthermore, because some genes' expression levels are uncorrelated, pooling over all the genes together in a multivariate QTL analysis, even if computationally feasible, would again have reduced power. To overcome these problems while realizing the potential of combining correlated expression traits, a common strategy (Lan *et al.,* 2003; Li *et al.,* 2006, 2007) is to apply dimension reduction to a group of relevant genes based on either a gene functional annotation system, such as the Gene Ontology (GO) (Ashburner *et al*., 2000) or KEGG (Kanehisa and Goto, 2000), or on clustering or co-expression analysis (Tseng, 2007). Albeit useful, there are various issues associated with these methods: how to select gene groups, how to conduct dimension reduction and how to interpret results, etc.; in addition, they may fail to take into account specific roles of the genes in a group or cluster. An extreme example is to use the average expression of the genes in a pathway or subnetwork to infer their common eQTL (Kliebenstein *et al.,* 2006), which will be shown to be a special case of our approach. Alternatively, another line of research is to analyze genome-wide expression traits simultaneously without depending on any gene grouping or dimension reduction. For example, Jia and Xu (2007) extended a popular Bayesian variable selection scheme for a single regression model (George and McCulloch, 1993) to multiple regression models, one for each expression trait. We argue that, although this approach takes advantage of existing multiple traits by borrowing information across them, as a generic method, by treating all the genes equally a priori, it fails to utilize biological knowledge that the genes function in different pathways. Our goal here is to incorporate biological knowledge on genes in the form of gene networks while maintaining the strength of borrowing information across the genes.

We would like to distinguish two aims of eQTL analyses. The first aim, focus of most existing works, is to identify candidate regulator–target gene pairs with eQTL suggesting locations of regulators while the genes whose expressions are analyzed as targets (Liu *et al*., 2008). We restrict our attention to this first step in this article. Based on the inferred regulator–target pairs, a preliminary gene regulatory network can be constructed, which however may not distinguish direct and indirect effects of a regulator. The second aim is to refine a preliminary regulatory network that is either inferred from aim one or given a priori. Some typical examples are to infer causal relationships (such that network edges showing indirect effects of a regulator can be removed) or orienting edges (Aten *et al*., 2008; Liu *et al*., 2008; Neto *et al.,* 2008) among the genes, possibly also including some phenotypes (such as clinical traits). Focusing on aim one, we require a gene network to be given a priori; this given network may or may not be related to the network to be inferred. In our example, we use a co-expression network while inferring a transcriptional regulatory network. Any network can be used in our approach as long as it implies our key prior assumption: any two neighboring genes in the network are more likely (than a random pair of genes) to have their expression co-regulated by some common loci. For example, it is reasonable to use KEGG pathways (Kanehisa and Goto, 2000), protein–protein interaction networks, transcriptional regulatory networks such as available from RegulonDB (Salgado *et al*., 2004), and even some computationally predicted networks, such as functional linkage (Lee *et al*., 2004) or regulatory (Faith *et al*., 2007) networks.

It is helpful to review a main theme of Lan *et al.* (2006) that partly motivated this research. In an eQTL study of mice related to obesity and diabetes, Lan *et al.* discovered that, by separate QTL mappings on the individual expression traits, the expression of a few, but only a minority of, GPCR genes, was statistically significantly linked to a region on chromosome two; by correlating these significant genes' expression with other genes, they found that the expanded list of co-expressed genes included all 194 GPCR genes, many of which showed secondary linkage peaks, though not statistically significant, in the same region of chromosome two. They argued that combining eQTL analysis with co-expression analysis would yield higher statistical power for biologically more meaningful discoveries. We completely agree with them; in fact, we did like to go further along the line by proposing a unified framework: rather than having two separate analysis steps of eQTL mapping and expression clustering, respectively, we would like to directly incorporate biological knowledge of GPCR genes into eQTL analysis and see whether it can improve statistical power to discover common eQTL for these functionally related genes. For example, we may first construct a co-expression network (possibly by clustering analysis), then incorporate the network information into eQTL mapping. More generally, because genes work coordinately as dictated by some pathways or networks, any two genes in the same pathway are expected to have correlated expression levels, leading to similar associations (or non-association) with a marker. Hence, to capitalize on the correlation among the genes as suggested by a gene network a priori, we construct a proper penalty function to realize the smoothness of the association parameters for neighboring genes in a general framework of penalized regression. Such an idea has been explored in the context of a single regression model by Li and Li (2008) and Pan *et al.* (2009). Here, we extend the idea to the scenario with multiple regression models, in which special characteristics derived from multiple models demand special treatments, such as in selecting penalization parameters. We will demonstrate the advantage of this approach over a standard approach that treats individual expression traits separately.

2 METHODS

2.1 Penalized regression

We first consider the most common situation with a single linear regression model:

$$Y = X\beta + \epsilon, \quad E(\epsilon) = 0, \qquad (1)$$

where $Y = (y_1, y_2, \ldots, y_n)'$ is a vector of trait values, $X = (x_{ik})$ is a design matrix, $\beta = (\beta_1, \ldots, \beta_p)'$ is the vector of unknown regression coefficients and $\epsilon = (\epsilon_1, \ldots, \epsilon_n)'$ is a vector of noise or error terms. We assume throughout that both Y and each predictor have been standardized to have mean 0 and variance 1. To estimate β, it is popular to use the ordinary least square estimator (OLSE)

$$\tilde{\beta} = \arg\min_{\beta} L(\beta) = \arg\min_{\beta} \sum_{i=1}^{n} (y_i - \sum_{k=1}^{p} x_{ik}\beta_k)^2.$$

For variable selection, it is common to follow a stepwise-type procedure based on, e.g. P-values of $\tilde{\beta}_k$, which however is unstable (Breiman, 1996). Furthermore, if $p \approx n$ or $p > n$, it may be difficult to estimate the variance, and thus P-value for any β_k; in fact, even $\tilde{\beta}_k$ would be unstable with large variability.

It has become increasingly popular to take a penalization approach that realizes variable selection and parameter estimation *simultaneously*, especially for 'large p, small n' problems, as arising from eQTL analysis. A penalized least square estimator (PLSE) is defined to be

$$\hat{\beta} = \arg\min_{\beta} L_P(\beta) = \arg\min_{\beta} L(\beta) + p_\lambda(\beta),$$

where $p_\lambda(\beta)$ is a penalty function with penalization or tuning parameter λ, which has to be determined based on some model selection criterion such as cross-validation (CV). There have been quite a few penalty functions proposed in the literature. For variable selection, the most popular is the L_1 or Lasso (Tibshirani, 1996) penalty:

$$p_\lambda(\beta) = \lambda \sum_{k=1}^{p} |\beta_k|.$$

A nice feature of the Lasso penalty is its capability for variable selection: with a sufficiently large λ, some $\hat{\beta}_k$ will be exactly 0, effectively excluding the corresponding predictor x_k from the model.

The whole solution path $\hat{\beta}(\lambda)$, as a function of penalization parameter λ, can be efficiently obtained by a slightly modified Lars algorithm (Efron *et al.,* 2004). Furthermore, one can take the Lars estimator as an alternative to the Lasso estimator, though the two are often similar.

In spite of many successful applications of the above methods, they are generic, possibly failing to take full advantage of prior knowledge of existing structures among the predictors. For example, in eQTL analysis, they ignore various gene functions and thus the relationships among the genes, such as represented in gene networks. To incorporate biological knowledge of gene networks, Li and Li (2008) proposed a new penalty that uses the Laplacian of a network. Specifically, given a network that describes the relationships among the predictors, denote d_i as the degree of predictor (or exchangeably, node) i in the network; that is, d_i is the number of direct neighbors of node i in the network. Li and Li's network-based penalty is

$$p_\lambda(\beta) = \lambda_1 \sum_{i=1}^{p} |\beta_i| + \lambda_2 \sum_{i \sim j} \left(\frac{\beta_i}{\sqrt{d_i}} - \frac{\beta_j}{\sqrt{d_j}} \right)^2, \qquad (2)$$

where $i \sim j$ means that nodes i and j are direct neighbors on the network. Similar to the elastic net penalty (Zou and Hastie, 2005), the first term is used for variable selection while the second to smooth the parameters over the network. A limitation of penalty (2) is its high computational cost, which can be prohibitive for large p: a straightforward implementation as suggested by Li and Li (2008) converts the model fitting to a Lasso problem with $n+p$ observations and p predictors. In addition, determining two tuning parameters (λ_1, λ_2) is computationally more intensive than choosing just one as in Lasso or Lars.

Pan *et al.* (2009) proposed a new network-based penalty. Similar to the second term of (2), their penalty contains multiple terms, each of which is for an edge in a network and has a form of grouped penalty (Yuan and Lin, 2006; Zhao *et al.,* 2006); it differs from existing grouped penalties for its specific reference to a network. This penalty also allows the user to choose a weight for each node, which realizes different types of shrinkages, enforcing various prior relationships among β_i's. For each node i with degree d_i, define $w_i = g(d_i, \gamma)$ as its weight, possibly dependent on d_i and γ. The penalty is

$$\begin{aligned} p_\lambda(\beta;\gamma,w) &= \lambda 2^{1/\gamma'} \sum_{i\sim j} p(\beta_i,\beta_j) \\ &= \lambda 2^{1/\gamma'} \sum_{i\sim j} \left(\frac{|\beta_i|^\gamma}{w_i} + \frac{|\beta_j|^\gamma}{w_j} \right)^{1/\gamma}, \end{aligned} \quad (3)$$

where $\gamma > 1$ and $1/\gamma' + 1/\gamma = 1$. Each penalty term $p(\beta_i, \beta_j)$ is essentially a weighted L_γ norm of vector $(\beta_i, \beta_j)'$, and hence $p_\lambda(\beta; \lambda, w)$ is convex in β. A generalized boosted Lasso algorithm (Zhao and Yu, 2004) can be applied to obtain an approximate solution path $\hat{\beta}(\lambda)$.

Pan *et al.* (2009) studied some specific choices of the weights: for example, if $w_i = d_i$, under some conditions, it is shown that $|\beta_i|$'s for neighboring genes are smoothed and shrunken to each other, which is often desired. Other choices of weights may result in different shrinkage schemes. Each term of (3) is a (weighted) grouped penalty, encouraging both β_i and β_j to be equal to 0 *simultaneously* (Yuan and Lin, 2006; Zhao *et al.,* 2006), which incorporates our prior assumption that two neighboring genes in a network should be more similar to each other and thus more likely to be both associated with the trait. Furthermore, a larger γ is chosen to impose a stronger shrinkage effect for two neighboring genes $i \sim j$.

The numerical studies in Pan *et al.* (2009) indicated some complex relationships between the choices of γ and w and the resulting predictive performance, while the performance in variable selection was more robust to the choices. Because the goal here is more for variable selection, we will simply use $\gamma = 2$ and $w_i = d_i$ throughout.

2.2 Adapting penalized regression to eQTL analysis

In eQTL analysis, in contrast to standard regression (1), we have multiple regression models, one for each expression trait g:

$$Y_g = X\beta_g + \epsilon_g, \quad E(\epsilon_g) = 0, \quad (4)$$

for $g = 1, \ldots, G$. Note that the design matrix X is related to the marker data drawn from the individuals, independent of gene g. Here we assume that, for simplicity, (i) eQTLs are located on markers; (ii) there are only additive effects: each marker is coded as -1, 0 and 1. It is possible to relax the above two assumptions: for the first one, we can take the general approach of Sen and Churchill (2001) by first imputing pseudo-genotypes between two neighboring markers (as shown in Section 3.4) and then proceeding as proposed here by treating pseudo-genotypes as observed markers; for the second one, we only need to create another dummy variable for the dominance effect and its associated regression coefficient for each marker, then proceed as proposed here.

To estimate gene- or trait-specific $\beta_g = (\beta_{g1}, \ldots, \beta_{gp})'$, a standard approach would apply a penalized method, such as Lasso or Lars, to each of the above model, which however ignores possible relationships among the genes. In particular, to capitalize on the association between co-expression and co-regulation, it is advantageous to smooth the parameters for functionally related or co-expressed genes. Here, we wish to incorporate into analysis gene function information in a general form as gene networks. As in a functional group analysis, if two genes' expression profiles are strongly correlated, then there is a high chance that the two genes are co-regulated and share some common eQTLs. Here, we assume that the co-expression or co-regulation relationships among the genes are described by a gene network a priori. Specifically, if two genes $g \sim h$ are linked in a network, we assume a priori that their regression coefficients are close: $|\beta_g| \approx |\beta_h|$; that is, two neighboring genes in the network are more likely to share common eQTLs. We can combine the above multiple regression models into a single one with $Y_c = (Y_1', \ldots, Y_G')'$, $X_c = \mathrm{diag}(X, \ldots, X)$ and $\beta = (\beta_1', \ldots, \beta_G')'$. The penalty function is

$$p_\lambda(\beta;\gamma,w) = \lambda\sqrt{2} \sum_{g\sim h} \sum_{k=1}^{p} \sqrt{\frac{\beta_{gk}^2}{d_g} + \frac{\beta_{hk}^2}{d_h}}. \quad (5)$$

Then the same generalized boosted Lasso algorithm can be applied to obtain an approximate solution path $\hat{\beta}(\lambda)$. Note that, if the λ is large enough, then $\beta_g \approx \beta_h$, corresponding to the network-averaging approach of Kliebenstein *et al.* (2006).

Although we obtain an approximate solution path $\hat{\beta}_g(\lambda)$ as a function of λ simultaneously for each g, rather than selecting a common $\hat{\lambda}_0$ for all the genes, we choose gene-specific tuning parameters $\hat{\lambda}_g$ to account for possible heterogeneity across the models; this partially acknowledges that two different pairs of neighboring genes in the network may share their mutual similarity to varying degrees. Only gene-specific data (Y_g, X) are used in evaluation to select $\hat{\lambda}_g$. Specifically, for a V-fold CV, first, we randomly partition the data (Y_g, X) into V subsets of about an equal size: (Y_g^v, X^v), $v = 1, \ldots, V$; denote by (Y_g^{-v}, X^{-v}) the data excluding partition v. Second, we fit a network-based (or other penalized) regression model to training data $\{(Y_g^{-v}, X^{-v}) : g = 1, \ldots, G\}$, and use (Y_g^v, X^v) as test data to calculate the prediction mean squared error (PMSE), $\mathrm{PMSE}_g^v(\lambda)$, of the trait for each λ and g; iterate the process for each v. Third, an overall PMSE for trait g is $\mathrm{PMSE}_g(\lambda) = \mathrm{Ave}_v \mathrm{PMSE}_g^v(\lambda)$. We then choose the penalization parameter for trait or model g as $\hat{\lambda}_g = \mathrm{argmin}_\lambda \mathrm{PMSE}_g(\lambda)$. At the end, the final network-based PLSE (Net) for trait g is $\hat{\beta}_g = \hat{\beta}_g(\hat{\lambda}_g)$. Note that, due to the choice of gene-specific penalization parameter, the penalty function used can be also regarded as gene specific; that is, for any gene g, the penalization parameter λ in the penalty function (5) is replaced by a gene-specific λ_g.

A standard approach would fit each model separately, e.g. by the Lars algorithm. Again, CV can be used to estimate an optimal penalization parameter $\hat{\lambda}_g$ and the final Lars-PLSE $\hat{\beta}_g = \hat{\beta}_g(\hat{\lambda}_g)$. Note that by the use of network information, the solution path $\hat{\beta}_g(\lambda)$ obtained by the network-based approach differs from that by the Lars method, as to be shown in the real data example. Hence, although gene-specfic penalization parameters are used by both methods, their final estimates are in general different (even with the same penalization parameter values).

Once we obtain a final PLSE $\hat{\beta}_g$, we examine the components of $\hat{\beta}_g$: a non-zero component suggests a possible linkage between trait g and the corresponding marker; of course, a claimed linkage is either a true or a false positive. For any penalization method, these suggested linkages by the non-zero components of the PLSE are taken as our estimated eQTLs. Finally, we note that, for either method, one can parameterize the penalization parameter as a fraction parameter s: for any given λ, suppose that $\hat{\beta}(\lambda)$ is the PLSE; then there is a corresponding fraction $s = p_\lambda(\hat{\beta}(\lambda))/p_\lambda(\hat{\beta}(0))$, in which the denominator refers to an estimate without penalization. Throughout the below results, we use fraction s, which is always between 0 and 1, facilitating its use and comparison across various models. In particular, we note that for our example data, we found that for either the Lars or our network-based method, if a common penalization parameter s_0 was imposed for all the traits, then $\hat{s}_0 = 0$ would be selected, leading to intercept-only models and suggesting no eQTL at all.

3 RESULTS

3.1 Example data

We analyzed a published mouse dataset deposited at the gene expression omnibus (GEO) by Lan *et al.* (2006). The data contained 60 mice in an F2 sample from the C57BL/6J (B6) and BTBR founder strains; B6 and BTBR strains, when made obese, showed different susceptibility to diabetes: B6-ob/ob mice are diabetes resistant while BTBR-ob/ob mice are not (Lan *et al.,* 2006). About 45 000 gene expression traits were obtained from Affymetrix `Moe430 Set` arrays, processed by the robust multi-array average (RMA) method (Irizarry *et al.,* 2003). Genotypes for 194 markers were distributed across 19 chromosomes with an average marker interval of ∼10 cM. There were about 7% missing genotypes. We applied the imputation method of Sen and Churchill (2001), as implemented in R/qtl (Broman *et al.,* 2003), to replace any missing genotype by an imputed value; all the work was done with this imputed dataset.

Because our goal is to investigate whether and how to incorporate gene function information into analysis, for illustration, we will consider only the genes in the G protein-coupled receptor (GPCR) protein signaling pathway. For the purpose, we constructed a gene co-expression network. Although it is possible to use the same dataset to do so, to mimic the practical situation with a network given a priori, we used another mouse dataset with liver gene expression of 135 female mice from an sample of F2 intercross between inbred strains of C3H/HeJ and C57BL/6J (Ghazalpour *et al.* 2006). The data were derived from some custom ink-jet arrays; the authors provided on their web site a subset of the data consisting of the top 8000 genes with the most varying expression levels across 135 samples. By gene names, we identified 17 GPCR genes appearing on both datasets; one of the genes, Rgs3, appeared twice (with two probe sets) and were denoted as Rgs3 and Rgs3(2), respectively. Using the second dataset, we calculated pairwise Pearson correlation coefficients between any two of the 17 genes; using a cutoff at 0.4, we obtained a co-expression network (Fig. 1). We conducted an eQTL analysis for these 17 GPCR genes throughout.

3.2 Analysis results

We applied Lasso/Lars implemented in R package `lars` (Efron *et al*., 2002) to each expression trait separately; there seemed to be some problems with CV for Lasso, so we used Lars throughout. Supplementary Figure 1 showed the solution paths for the genes, and the selected tuning parameters in terms of fraction s by a 5-fold CV; note quite different $\hat{s}_g$ across the models. The Lars-PLSEs of β_g's from the final models are shown in Supplementary Figure 2; a non-zero component of $\hat{\beta}_g$ suggested a possible linkage between trait g and the corresponding locus. For 10 genes, $\hat{s}_g = 0$ was selected by CV, leading to an intercept-only model, hence no eQTL was detected; for the other ones, four (Ccr5, Dok4, Rps6ka4 and 1200007D18Rik) showed a linkage at marker locus *D2Mit148* on chromosome 2, which was in agreement with Lan *et al*. (2006) who found a significant linkage peak on chromosome 2 for several GPCR genes. Nonetheless, because most of the genes did not show a linkage on chromosome 2, we would like to see whether the network-based method could improve the detection by borrowing information across the genes connected on the network. Note that the four genes whose expression traits were identified by the Lars to be linked to marker *D2Mit148* on chromosome 2 are mostly disconnected to each other in the co-expression network (Fig. 1).

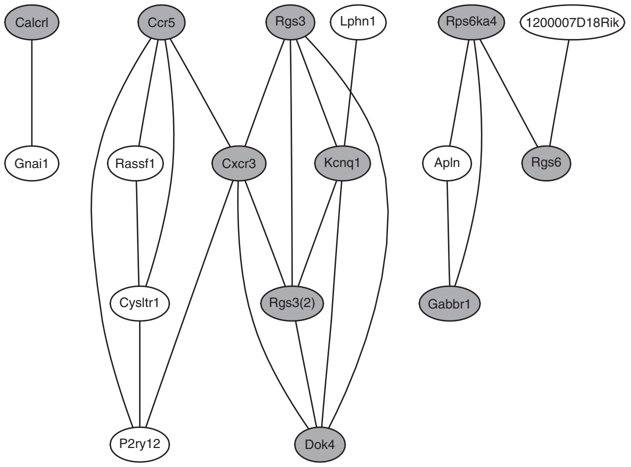

Fig. 1. A co-expression network derived from Ghazalpour *et al.*'s data. The dark nodes were the genes with their expression traits linked to a marker (D2Mit148) locus on chromosome 2 as suggested by the Net, while Lars suggested only four genes linked to the locus: Ccr5, Rps6ka4, Dok4 and 1200007D18Rik.

The solution paths and selected penalization parameters $\hat{s}_g$ by our network-based method (Net) with a 5-fold CV are shown in Supplementary Figure 3. It is clear that the solution paths are different from that of Lars. As a result, more linkages, either true or false positives, were identified by the network-based method (Supplementary Fig. 4). In particular, in addition to the three of the four genes identified by Lars (except 1200007D18Rik), the network-based method also identified seven other genes whose expression was linked to marker *D2Mit148* on chromosome 2: Calcr1, Cxcr3, Kcnq1, Rgs6, gs3, Rgs3(2) and Gabbr1. In contrast to the four largely disconnected genes identified by the Lars, all the genes except Calcr1 identified by the Net were well connected to each other in the co-expression network.

We also applied the network-averaging approach of Kliebenstein *et al.* (2006). It took the average expression of all 17 genes in the network as the expression trait for each individual, then applied the Lars. By a 5-fold CV, the penalization parameter was selected at 0, hence all the regression coefficient estimates were 0; that is, no eQTL was detected. The solution paths are shown in Supplementary Figure 5.

Next, we used simulated data to confirm that the network-based method could indeed improve statistical power to detecting eQTL in practical situations.

3.3 Simulation I

3.3.1 Simulation setups To mimic real data, we used the same genotype data and same number of individuals (i.e. $n = 60$ and $p =$ 194 for each gene); the network was the subnetwork for the real data containing five genes: Rps6ka4, 1200007D18Rik, Apln, Rgs6 and Gabbr1. The true models for simulated data assumed that all the five genes were linked to the last marker on chromosomes 2 (*D2Mit148*), 4 (except 1200007D18Rik) linked to the first marker on chromosome 10 (*D10Mit16*), and only one (Gabbr1) linked to the first marker on chromosome 13 (*D13Mit16*). Specifically, the

Table 1. Simulation I: sample means (SDs) of the numbers of the true positives (q_1) and false positives (q_0) by the two methods from 100 simulated datasets

Case	Gene	Rps6ka4		120007D18Rik		Apln		Rgs6		Gabbr1	
		q_0	q_1	q_0	q_1	q_0	q_1	q_0	q_1	q_0	q_1
	True	0	2	0	1	0	2	0	2	0	3
1	Lars	9.58*	1.98	7.48	0.97	10.23*	1.98	10.35	1.96	14.17	2.96
		(5.75)	(0.14)	(6.76)	(0.17)	(5.98)	(0.14)	(6.41)	(0.20)	(5.74)	(0.20)
	Net	7.06	2.00	7.59	0.99	8.62	2.00	9.09	2.00	13.69	2.99
		(5.18)	(0.00)	(7.15)	(0.10)	(6.67)	(0.00)	(6.57)	(0.00)	(6.81)	(0.10)
2	Lars	8.23*	1.52*	6.62	0.77*	8.94	1.66*	8.89	1.50*	12.32	2.20*
		(5.54)	(0.69)	(6.77)	(0.42)	(5.58)	(0.57)	(6.47)	(0.64)	(6.37)	(0.90)
	Net	6.79	1.93	6.98	0.88	8.31	1.91	8.48	1.86	12.18	2.63
		(5.27)	(0.29)	(6.37)	(0.33)	(6.48)	(0.35)	(6.41)	(0.38)	(7.13)	(0.53)
3	Lars	6.35	1.06*	5.02*	0.45*	6.86	1.05*	6.74	1.04*	9.21	1.46*
		(5.36)	(0.79)	(6.41)	(0.50)	(5.58)	(0.74)	(5.99)	(0.83)	(7.02)	(1.01)
	Net	6.69	1.55	6.44	0.68	7.50	1.61	7.44	1.49	10.11	1.97
		(5.83)	(0.72)	(6.65)	(0.47)	(6.25)	(0.62)	(6.65)	(0.70)	(7.80)	(0.87)
4	Lars	4.99*	0.70*	4.10*	0.21*	5.23	0.67*	5.16	0.70*	6.71*	0.82*
		(5.29)	(0.73)	(6.10)	(0.41)	(5.52)	(0.77)	(6.11)	(0.78)	(6.51)	(0.87)
	Net	5.81	1.11	5.25	0.37	6.06	1.08	5.42	0.91	8.05	1.30
		(5.63)	(0.80)	(7.42)	(0.49)	(6.04)	(0.80)	(6.23)	(0.83)	(7.31)	(0.94)

*P-value <0.01 from a paired t-test to compare the mean difference of q_1 (or q_0) from that of the Net.

expression trait of gene g for individual i was simulated from a linear model

$$y_{gi}=x_{1i}\beta_{g,1}+x_{2i}\beta_{g,2}+x_{3i}\beta_{g,3}+\epsilon_{gi},\quad \epsilon_{gi}\overset{iid}{\sim}N(0,\sigma_e),$$

where x_{1i}, x_{2i} and x_{3i} are the genotypes of individual i at the three possibly linked markers, and the true coefficients were $\beta_{g,1}=r_{g1j}\times(-0.2)$ for $g=1,...,5$, $\beta_{g,2}=r_{g2j}\times(0.2)$ for $g\neq 2$ and $\beta_{g,2}=0$ for $g=2$, and $\beta_{g,3}=0.2$ for $g=5$ and $\beta_{g,2}=0$ for $g\neq 5$; $r_{gkj}\overset{iid}{\sim}U(0.8,1.2)$ was a scaling factor used to perturb the effect size of marker k on trait g in dataset j. Four cases were considered with the noise SD $\sigma_e=0.3$, 0.5, 0.7 and 0.9, respectively; for each case, 100 simulated datasets were generated independently.

To save computing time, for tuning parameter selection, we used an independent tuning dataset of an equal size (i.e. $n=60$). The idea was similar to CV except that we only needed to fit a model once with the training data, then used the tuning data to calculate PMSE and thus selected the tuning parameter $\hat{s}_g$ for each trait g.

The simulation setups reflected practical situations. First, there were multiple eQTLs, some of which were common for all the genes while others were not. Second, although the effect sizes of a common eQTL on its targets were close, they were not exactly the same. Finally, the prior belief as modeled in the penalty function was largely, but not completely, correct: some markers associated with a gene were not associated with the gene's neighbor(s). Note that, although there were at most three eQTLs for each trait, which was unknown, as usual, we fitted a model with all 194 markers for each trait.

3.3.2 Simulation results The simulation results are included in Table 1. For each gene, we considered the number of false positives (q_0) and number of true positives (q_1); ideally, we would like to have q_0 as small and q_1 as large as possible. From Table 1, it is clear that the network-based method in general gave a higher number of true positives while often maintaining an either smaller or comparable number of false positives as compared with the Lars, a standard approach that treated each trait separately. One can even argue that, for each trait with only 1–3 true eQTLs but with >190 non-eQTL markers, it is affordable to have a few extra false positives as long as there is a significant improvement in detecting a true eQTL. Note that for many cases the mean differences of q_1's between the two methods were substantial and statistically significant, as judged by the P-values given by paired t-tests. Hence, as expected, by borrowing information across the genes in a network, the network-based method gained statistical power of detecting eQTL.

3.4 Simulation II

The second set of simulation setups were the same as that in Simulation I except the following three modifications. First, based on the same genotype data, we used R/qtl to impute a pseudo marker every 8 cM on each chromosome, resulting in a total of 408 original or imputed markers. Second, we added two pseudogenes, say genes 6 and 7, to the original subnetwork of the five genes: gene 6 was connected to both gene 120007D18Rik and gene 7; there was no any other change with the subnetwork. Third, in addition to the three previous eQTLs, we added a new eQTL at a pseudo-marker at 80 cM on chromosome 3 (between markers D3Mit44 and D3Mit19) for the five genes with a regression coefficient of −0.2, while the two pseudo genes were not linked to any loci. In addition to the previous two methods, we also considered the network-averaging (Ave) approach of Kliebenstein *et al.* (2006): for each individual, the average expression level of the seven genes was used as the trait, then a linear model was fitted using Lars; it was implicitly assumed that all the genes in the network shared the same eQTL. Otherwise, we followed exactly the same procedures as before, yielding four setups and the corresponding results (Table 2).

It is evident that our network-based method always detected more eQTLs while giving fewer false positives than the Lars method.

Table 2. Simulation II: sample means (SDs) of the numbers of the true positives (q_1) and false positives (q_0) by the three methods from 100 simulated datasets

Case	Gene	Rps6ka4		120007D18Rik		Apln		Rgs6		Gabbr1		Gene 6		Gene 7	
		q_0	q_1	q_0	q_1	q_0	q_1	q_0	q_1	q_0	q_1	q_0	q_1	q_0	q_1
	True	0	3	0	2	0	3	0	3	0	4	0	0	0	0
1	Lars	14.05*	2.70*	13.98*	1.54*	14.77*	2.71*	14.69*	2.62*	19.14*	3.63*	2.68	0.00	3.27	0.00
		(6.05)	(0.56)	(7.52)	(0.58)	(6.45)	(0.50)	(5.95)	(0.56)	(6.67)	(0.56)	(5.02)	(0.00)	(5.73)	(0.00)
	Net	10.30	2.93	11.38	1.93	12.03	2.94	12.44	2.96	14.79	3.91	2.60	0.00	3.62	0.00
		(4.63)	(0.29)	(6.00)	(0.29)	(5.64)	(0.28)	(5.05)	(0.20)	(5.51)	(0.32)	(4.23)	(0.00)	(5.52)	(0.00)
	Ave	15.98*	2.93	16.98*	1.93	15.98*	2.93	15.98*	2.93	15.79	3.12*	18.91*	0.00	18.91*	0.00
		(6.83)	(0.26)	(6.83)	(0.26)	(6.83)	(0.26)	(6.83)	(0.26)	(6.79)	(0.48)	(6.82)	(0.00)	(6.82)	(0.00)
2	Lars	11.86*	1.83*	11.01*	0.87*	13.10*	1.85*	12.33*	1.69*	15.84*	2.26*	2.68	0.00	3.27	0.00
		(6.22)	(0.92)	(7.89)	(0.73)	(6.36)	(0.83)	(6.11)	(0.91)	(7.39)	(0.97)	(5.02)	(0.00)	(5.73)	(0.00)
	Net	9.16	2.34	9.37	1.27	10.81	2.50	11.22	2.40	13.01	2.97	2.73	0.00	3.28	0.00
		(4.43)	(0.78)	(6.82)	(0.69)	(5.55)	(0.66)	(5.13)	(0.67)	(6.12)	(0.85)	(4.56)	(0.00)	(5.11)	(0.00)
	Ave	15.37*	2.59*	16.27*	1.69*	15.37*	2.59	15.37*	2.59	15.27*	2.69*	17.96*	0.00	17.96*	0.00
		(6.90)	(0.55)	(6.88)	(0.49)	(6.90)	(0.55)	(6.90)	(0.55)	(6.87)	(0.63)	(6.79)	(0.00)	(6.79)	(0.00)
3	Lars	9.10*	1.07*	8.35*	0.48*	10.16*	1.11*	9.76*	1.01*	11.12	1.14*	2.68	0.00	3.27	0.00
		(6.81)	(0.90)	(8.18)	(0.67)	(6.68)	(0.79)	(6.43)	(0.81)	(7.79)	(1.01)	(5.05)	(0.00)	(5.73)	(0.00)
	Net	7.23	1.43	7.31	0.62	9.11	1.63	8.73	1.42	10.30	1.70	2.60	0.00	3.37	0.00
		(4.50)	(0.91)	(7.12)	(0.71)	(5.44)	(0.86)	(5.08)	(0.81)	(6.35)	(0.99)	(4.25)	(0.00)	(5.50)	(0.00)
	Ave	14.65*	2.14*	15.41*	1.38*	14.65*	2.14*	14.65*	2.14*	14.58*	2.21*	16.79*	0.00	16.79*	0.00
		(7.23)	(0.75)	(7.23)	(0.65)	(7.23)	(0.75)	(7.23)	(0.75)	(7.22)	(0.81)	(7.11)	(0.00)	(7.11)	(0.00)
4	Lars	6.42*	0.55*	6.01	0.27	7.72	0.66*	7.57	0.66*	9.09*	0.74*	2.68	0.00	3.27	0.00
		(5.68)	(0.73)	(7.89)	(0.49)	(6.62)	(0.68)	(6.80)	(0.76)	(7.55)	(0.87)	(5.03)	(0.00)	(5.73)	(0.00)
	Net	5.42	0.72	5.53	0.32	6.99	0.84	7.11	0.84	7.94	0.93	2.51	0.00	3.57	0.00
		(4.46)	(0.79)	(6.51)	(0.55)	(5.63)	(0.75)	(6.05)	(0.83)	(5.64)	(0.88)	(4.25)	(0.00)	(5.58)	(0.00)
	Ave	13.06*	1.72*	13.69*	1.09*	13.06*	1.72*	13.06*	1.72*	13.02*	1.76*	14.78*	0.00	14.78*	0.00
		(7.18)	(0.87)	(7.24)	(0.73)	(7.18)	(0.87)	(7.18)	(0.87)	(7.13)	(0.91)	(7.28)	(0.00)	(7.28)	(0.00)

*P-value <0.01 from a paired t-test to compare the mean difference of q_1 (or q_0) from that of the Net.

The network-averaging method discovered more eQTLs but at the high cost of much larger number of false positives (with statistical significance for most of them).

4 DISCUSSION

We have proposed a gene network-based regression approach to multiple linkage analysis of expression traits. Because the genes in the same functional group or pathway tend to be co-expressed and co-regulated, it makes sense to combine eQTL analyses for the functionally related genes; this point has been increasingly recognized (e.g. Lan *et al.,* 2006). However, there is yet any consensus on how to do so effectively; here we have outlined a novel penalized regression approach that incorporates into a penalty the prior knowledge of gene functions embedded in a network. In particular, we have formulated multiple regression models for individual traits as a single, expanded regression model so that an existing penalization technique for a single model can be applied. Although this formulation largely simplifies the problem, there are some special features associated with multiple models that distinguish the problem from the one with only a single model. For example, although the regression coefficient solution paths for all the traits are obtained as functions of a common penalization parameter, because of possibly varying effect sizes of a common eQTL across the traits or of varying degrees of co-expression for neighboring genes in the network, we allow trait-specific penalization parameters to be selected at the end; for the mouse data, the choice of such trait-specific penalization parameters seemed to be necessary.

Our work is related to the ongoing efforts of incorporating biological knowledge into genomic data analysis. For example, Wei and Pan (2008b) considered the use of gene functional groups in regression analysis of gene expression on DNA–protein binding data [or similarly on DNA sequence data as shown by Conlon *et al.* (2003)] to infer transcription factor–target relationships. The basic idea is that the genes within the same functional group share some relevant characteristics, and thus it may be beneficial to borrow information across them. Their methods can be adapted to the current context. However, a limitation of the methods is their dependence on gene group selection: for example, since there are thousands of GO categories, which one to use? There is a trade-off between group size and functional specificity: a more specific and functionally homogeneous functional group contains necessarily fewer genes, introducing estimation difficulties with a smaller sample size. We feel that a gene network is both flexible and powerful to capture biological knowledge, as compared with gene functional groups; in fact, some even argued that the former should be more suitable (Fraser and Marcotte, 2004). For gene functional groups, in addition to which group to use, there are also other issues, such as how to handle genes annotated in multiple groups and why treating the genes inside a group equally a priori.

Although we have focused on regression analysis for multiple eQTL mapping, the idea of incorporating biological knowledge can be equally applied to single eQTL analysis. For instance, to identify possible associations between any expression trait and any genomic marker, Kendziorski *et al.* (2006) proposed a mixture model (of transcripts) over markers (MOM), in which the transcripts are treated equally a priori as in a standard mixture model; Gelfond *et al.* (2007) proposed incorporating genomic location information into MOM. As developed for differential expression analysis, incorporating gene functional groups as defined by either functional annotations or clustering analysis (Pan, 2006), or by gene networks (Wei and Li, 2007; Wei and Pan, 2008a), into a mixture model for eQTL analysis is in principle straightforward, though computational challenge remains due to a large number of components in the mixture model. Finally, our current implementation based on the generalized boosted Lasso algorithm can handle dozens of genes in a pathway; for genome-wide expression traits, one may deal with individual pathways separately. Although considering pathways one by one is often acceptable, if one wants to combine thousands of expression traits simultaneously with a genome-wide network, more efficient algorithms still need to be developed. These are all interesting topics to be studied.

ACKNOWLEDGEMENTS

The author thanks the reviewers' for helpful comments.

Funding: National Institutes of Health (GM081535 and HL65462).

Conflict of Interest: none declared.

REFERENCES

Ashburner,M. *et al*. (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, **25**, 25–29.

Aten,J.E. *et al.* (2008) Using genetic markers to orient the edges in quantitative trait networks: the NEO software. *BMC Syst. Biol.*, **2**, 34.

Bogdan,M. *et al.* (2004) Modifying the Schwarz Bayesian information criterion to locate multiple interacting quantitative trait loci. *Genetics*, **167**, 989–999.

Breiman,L. (1996) Heuristics of instability and stabilization in model selection. *Ann. Stat.*, **24**, 2350–2383.

Brem,R.B. *et al.* (2002) Genetic dissection of transcriptional regulation in budding yeast. *Science*, **296**, 752–755.

Broman,K.W. and Speed,T.P. (2002) A model selection approach for the identification of quantitative trait loci in experimental crosses (with discussion). *J. R. Stat. Soc. Ser. B*, **64**, 641–656.

Broman,K.W. *et al.* (2003) R/qtl: QTL mapping in experimental crosses. *Bioinformatics*, **19**, 889–890.

Conlon,E.M. *et al.* (2003) Integrating regulatory motif discovery and genome-wide expression analysis. *Proc. Natl Acad. Sci. USA*, **100**, 3339–3344.

Efron,B. *et al.* (2004) Least angle regression. *Ann. Stat.*, **32**, 407–499.

Faith,J.J. *et al.* (2007) Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol.*, **5**, e8.

Fraser,A.G. and Marcotte,E.M. (2004) A probabilistic view of gene function. *Nat. Genet.*, **36**, 559–564.

Gelfond,J.A.L. *et al.* (2007) Proximity model for expression quantitative trait loci (eQTL) detection. *Biometrics*, **63**, 1108–1116.

George,E.I. and McCulloch,R.E. (1993) Variable selection via Gibbs sampling. *J. Am. Stat. Assoc.*, **88**, 881–889.

Ghazalpour,A. *et al.* (2006) Integrating genetic and network analysis to characterize genes related to mouse weight. *PLoS Genet.*, **2**, e130.

Irizarry,R.A. *et al.* (2003) Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, **4**, 249–264.

Jia,Z. and Xu,S. (2007) Mapping quantitative trait loci for Expression Abundance. *Genetics*, **176**, 611–623.

Jiang,C. and Zeng,Z.B. (1995) Multiple traits analysis of genetic mapping for quantitative trait loci. *Genetics*, **140**, 1111–1127.

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

Kendziorski,C.M. and Wang,P. (2006). A review of statistical methods for expression quantitative trait loci mapping. *Mamm. Genome*, **17**, 509–517.

Kendziorski,C.M. *et al.* (2006) Statistical methods for expression quantitative trait loci (eQTL) mapping. *Biometrics*, **62**, 19–27.

Kliebenstein,D.J. *et al.* (2006) Identification of QTLs controlling gene expression networks defined a priori. *BMC Bioinformatics*, **7**, 308.

Lan,H. *et al.* (2003) Dimension reduction for mapping mRNA abundance as quantitative traits. *Genetics*, **164**, 1607–1614.

Lan,H. *et al*. (2006) Combined expression trait correlations and expression quantitative trait locus mapping. *PLoS Genet.*, **2**, 51–61.

Lee,I. *et al.* (2004) Probabilistic functional network of yeast genes. *Science*, **306**, 1555–1558.

Li,C. and Li,H. (2008) Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, **24**, 1175–1182.

Li,H. *et al.* (2006) Integrative genetic analysis of transcription modules: towards filling the gap between genetic loci and inherited traits. *Hum. Mol. Genet.*, **15**, 481–492.

Li,N. *et al.* (2007) Functional group-based linkage analysis of gene expression trait loci. *BMC Proceedings*, **1** (Suppl. 1), S117.

Liu,B. *et al.* (2008) Gene network inference via structured equation modeling in genetical genomics experiments. *Genetics*, **178**, 1763–1776.

Neto,E.C. *et al.* (2008) Inferring causal phenotype networks from segregating populations. *Genetics*, **179**, 1089–1100.

Pan,W. (2006) Incorporating gene functional annotations in detecting differential gene expression. *Appl. Stat.*, **55**, 301–316.

Pan,W. *et al.* (2009) Incorporating predictor network in penalized regression with application to microarray data. *Biometrics*, in press (Available as Research Report 2009-001, Division of Biostatistics, University of Minnesota. at `http://www.biostat.umn.edu./rrs.php` last accessed date on February 15, 2009).

Salgado,H. *et al.* (2004) RegulonDB (version 4.0): transcriptional regulation, operon organization and growth conditions in Escherichia coli K-12. *Nucleic Acids Res.*, **32**, D303–D306.

Schadt,E.E. *et al*. (2003) Genetics of gene expression surveyed in maize, mouse and man. *Nature*, **422**, 297–302.

Sen,S. and Churchill,G.A. (2001) A statistical framework for quantitative trait mapping. *Genetics*, **159**, 371–387.

Storey,J.D. *et al.* (2005) Multiple locus linkage analysis of genomewide expression in yeast. *PLoS Biol.*, **3**, e267.

Tibshirani,R. (1996) Regression shrinkage and selection via the LASSO. *J. R. Stat. Soc. B*, **58**, 267–288.

Tseng,G.C. (2007) Penalized and weighted K-means for clustering with scattered objects and prior information in high-throughput biological data. *Bioinformatics*, **23**, 2247–2255.

Wei,Z. and Li,H. (2007) A Markov random field model for network-based analysis of genomic data. *Bioinformatics*, **23**, 1537–1544.

Wei,P. and Pan,W. (2008a) Incorporating gene networks into statistical tests for genomic data via a spatially correlated mixture model. *Bioinformatics*, **24**, 404–411.

Wei,P. and Pan,W. (2008b) Incorporating gene functions into regression analysis of DNA-protein binding data and gene expression data to construct transcriptional networks. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **5**, 401–415.

Xu,S. (2003) Estimating polygenic effects using markers of the entire genome. *Genetics*, **163**, 789–801.

Xu,S. (2007) An empirical Bayes method for estimating epistatic effects of quantitative trait loci. *Biometrics*, **63**, 513–521.

Yi,N. *et al.* (2003) Stochastic search variable selection for identifying quantitative trait loci. *Genetics*, **164**, 1129–1138.

Yuan,M. and Lin,Y. (2006) Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. B* **68**, 49–67.

Zhang,Y.-M. and Xu,S. (2005) A penalized maximum likelihood method for estimating epistatic effects of QTL. *Heredity* **95**, 96–104.

Zhao,P. and Yu,B. (2004) Boosted Lasso. Technical Report #678, Department of Statistics, UC-Berkeley.

Zhao,P. *et al.* (2006) Grouped and hierarchical model selection through composite absolute penalties. *Ann. Stat.*, in press.

Zou,H. and Hastie,T. (2005) Regularization and variable selection via the elastic net. *J. R. Stat. Soc. B*, **67**, 301–320.

Zou,W. and Zeng,Z.B. (2007) Multiple interval mapping for gene expression QTL analysis. Available at statgen.ncsu.edu/zeng/MIM-eQTL.pdf (last accessed date on Feburary 15, 2009).