BIOINFORMATICS

Vol. 29 ISMB/ECCB 2013, pages i171–i179
doi:10.1093/bioinformatics/btt238

Efficient network-guided multi-locus association mapping with graph cuts

Chloé-Agathe Azencott[1,*], Dominik Grimm[1], Mahito Sugiyama[1], Yoshinobu Kawahara[2] and Karsten M. Borgwardt[1,3]

[1]Machine Learning and Computational Biology Research Group, Max Planck Institute for Developmental Biology & Max Planck Institute for Intelligent Systems Spemannstr. 38, 72076 Tübingen, Germany, [2]The Institute of Scientific and Industrial Research (ISIR) Osaka University 8-1 Mihogaoka, Ibaraki-shi, Osaka 567-0047, Japan and [3]Zentrum für Bioinformatik, Eberhard Karls Universität Tübingen, 72076 Tübingen, Germany

ABSTRACT

Motivation: As an increasing number of genome-wide association studies reveal the limitations of the attempt to explain phenotypic heritability by single genetic loci, there is a recent focus on associating complex phenotypes with sets of genetic loci. Although several methods for multi-locus mapping have been proposed, it is often unclear how to relate the detected loci to the growing knowledge about gene pathways and networks. The few methods that take biological pathways or networks into account are either restricted to investigating a limited number of predetermined sets of loci or do not scale to genome-wide settings.

Results: We present SConES, a new efficient method to discover sets of genetic loci that are maximally associated with a phenotype while being connected in an underlying network. Our approach is based on a minimum cut reformulation of the problem of selecting features under sparsity and connectivity constraints, which can be solved exactly and rapidly.

SConES outperforms state-of-the-art competitors in terms of runtime, scales to hundreds of thousands of genetic loci and exhibits higher power in detecting causal SNPs in simulation studies than other methods. On flowering time phenotypes and genotypes from *Arabidopsis thaliana*, SConES detects loci that enable accurate phenotype prediction and that are supported by the literature.

Availability: Code is available at http://webdav.tuebingen.mpg.de/u/karsten/Forschung/scones/.

Contact: chloe-agathe.azencott@tuebingen.mpg.de

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

1 INTRODUCTION

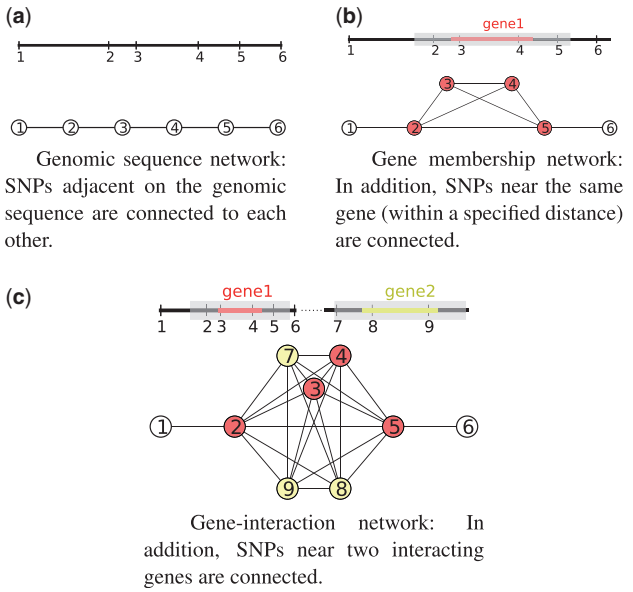
Twin and family/pedigree studies make it possible to estimate the heritability of observed traits, that is to say the amount of their variability that can be attributed to genetic differences. In the past few years, genome-wide association studies (GWAS), in which several hundreds of thousands to millions of single nucleotide polymorphisms (SNPs) are assayed in up to thousands of individuals, have made it possible to identify hundreds of genetic variants associated with complex phenotypes (Zuk *et al.*, 2012). Unfortunately, although studies associating single SNPs with phenotypic outcomes have become standard, they often fail to explain much of the heritability of complex traits (Manolio *et al.*, 2009). Investigating the joint effects of multiple loci by mapping sets of genetic variants to the phenotype has the potential to help explain part of this missing heritability (Marchini *et al.*, 2005). Although efficient multiple linear regression approaches (Cho *et al.*, 2010; Rakitsch *et al.*, 2012; Wang *et al.*, 2011) make the detection of such multivariate associations possible, they often remain limited in power and hard to interpret. Incorporating biological knowledge into these approaches could help boosting their power and interpretability. However, current methods are limited to predefining a reasonable number of candidate sets to investigate (Cantor *et al.*, 2010; Fridley and Biernacka, 2011; Wu *et al.*, 2011), for instance by relying on gene pathways. They consequently run the risk of missing biologically relevant loci that have not been included in the candidate sets. This risk is made even likelier by the incomplete state of our current biological knowledge.

For this reason, our goal here is to use prior knowledge in a more flexible way. We propose to use a biological network, defined between SNPs, to guide a multi-locus mapping approach that is both efficient to compute and biologically meaningful: *We aim to find a set of SNPs that (i) are maximally associated with a given phenotype and (ii) tend to be connected in a given biological network. In addition, this set must be computed efficiently on genome-wide data.* In this article, we assume an additive model to characterize multi-locus association. The network constraint stems from the assumption that SNPs influencing the same phenotype are biologically linked. However, the diversity of the type of relationships that this can encompass, together with the current incompleteness of biological knowledge, makes providing a network in which all the relevant connections are present unlikely. For this reason, although we want to encourage the SNPs to form a subnetwork of the network, we also do not want to enforce that they *must* form a single connected component. Finally, we stress that the method must scale to networks of hundreds of thousands or millions of nodes. Approaches by Chuang *et al.* (2007) or Li and Li (2008); Nacu *et al.* (2007) developed to analyze gene networks containing hundreds of nodes do therefore not apply.

Although our method can be applied to any network between genetic markers, we explore three special types of networks (Fig. 1):

- *GS network*: SNPs adjacent on the genomic sequence (GS) are linked together. In this setting, we aim at recovering subsequences of the genomic sequence that correlate with the phenotype.

*To whom correspondence should be addressed.



Genomic sequence network: SNPs adjacent on the genomic sequence are connected to each other.

Gene membership network: In addition, SNPs near the same gene (within a specified distance) are connected.

Gene-interaction network: In addition, SNPs near two interacting genes are connected.

Fig. 1. Small examples of the three types of networks considered

- *GM (gene membership) network*: SNPs are connected as in the sequence network described earlier in the text; in addition, SNPs near the same gene are linked together as well. Usually, a SNP is considered to belong to a gene if it is either located inside said gene or within a predefined distance of this gene. In this setting, we aim more particularly at recovering genes that correlate with the phenotype.
- *GI (gene interaction) network*: SNPs are connected as in the GM network described earlier in the text. In addition, supposing we have a gene–gene interaction network (derived, for example, from protein–protein interaction data or gene expression correlations), SNPs belonging to two genes connected in the gene network are linked together. In this setting, we aim at recovering potential pathways that explain the phenotype.

Our task is a feature selection problem in a graph-structured feature space, where the features are the SNPs, and the selection criterion should be related to their association with the phenotype considered. Our problem is different from subgraph selection problems such as those encountered in chemoinformatics, where each object is a graph and each feature is a subgraph of its own (Tsuda, 2011).

Several approaches have already been developed for selecting graph-structured features. A number of them (Le Saux and Bunke, 2005; Jie *et al.*, 2012) only use the graph over the features to build the learners evaluating their relevance, but do not enforce that the selected features should follow this underlying structure. Indeed, they can be applied to settings where the features connectivity varies across examples, whereas here, all individuals share the same network.

The overlapping group Lasso (Jacob *et al.*, 2009; Liu *et al.*, 2012) is a sparse linear model designed to select features that belong to the union of a small number of predefined groups. If a graph over the features is given, defining those groups as all pairs of features connected by an edge or as all linear subgraphs of a given size yields the so-called graph Lasso. A similar approach is taken by Huang *et al.* (2009): their structured sparsity penalty encourages selecting a small number of base blocks, where blocks are sets of features defined so as to match the structure of the problem. In the case of a graph-induced structure, blocks are defined as small connected components of that graph. As shown in Mairal and Yu (2011), the overlapping group Lasso aforementioned is a relaxation of this binary problem. As the number of linear subgraphs or connected components of a given size grows exponentially with the number of nodes of the graph, which can reach millions in the case of whole-genome SNP data, only the edge-based version of the graph Lasso can be applied to our problem. It is however unclear whether it is sufficient to capture long-range connections between graph nodes.

Li and Li (2008) propose a network-constrained version of the Lasso that imposes the type of graph connectivity we deem desirable. However, their approach has been developed with networks of genes (rather than of SNPs) in mind and does not scale easily to the datasets we envision. Indeed, the implementation they propose relies on a singular value decomposition of the Laplacian of the network, which is intensive to compute and cannot be stored in memory.

Chuang *et al.* (2007) also searched subnetworks of protein–protein interaction networks that are maximally associated with a phenotype; however, their greedy approach requires fixing beforehand a (necessarily small) upper-limit on the size of the subnetworks considered.

In the case of directed acyclic graphs, Mairal and Yu (2011) propose a minimum flow formulation that makes it possible to use for groups (or blocks) the set of all paths of the network. Unfortunately, the generalization to undirected graphs with cycles, such as the SNP networks we consider, requires randomly assigning directions to edges and pruning those in cycles without any biological justification. Although this can work reasonably well in practice (Mairal and Yu, 2011), this is akin to artificially removing more than half of the network connections without any biological justification.

In what follows, we formulate the network-guided SNP selection problem as a minimum cut problem on a graph derived from the SNP network in Section 2 and evaluate the performance of our solution both in simulations and on actual *Arabidopsis thaliana* data in Section 3.

2 METHODS

2.1 Problem formulation

Let n be the number of SNPs and m the number of individuals. The SNP–SNP network is described by its adjacency matrix $\boldsymbol{W}$ of size $n \times n$. A number of statistics based on covariance matrices, such as the Hilbert-Schmidt Independence Criterion (HSIC), (Gretton *et al.*, 2005) or the Sequence Kernel Association Test (SKAT) (Wu *et al.*, 2011), can be used to compute a measure of dependence $\boldsymbol{c} \in \mathbb{R}^n$ between each single SNP and the phenotype. Under the common assumption that the joint effect of several SNPs is additive (which corresponds to using linear kernels in those methods), $\boldsymbol{c}$ is such that the association between a group of SNPs and the phenotype can be quantified as the sum of the scores of the SNPs belonging to this group. That is, given an indicator vector $\boldsymbol{f} \in \{0,1\}^n$ such that, for any $p \in \{1, \ldots, n\}$, f_p is set to 1 if the p-th

SNP is selected and 0 otherwise, the score of the selected SNPs is given by $Q(\boldsymbol{f}) = \sum_{p=1}^{n} c_p f_p = \boldsymbol{c}^\top \boldsymbol{f}$.

We want to find the indicator vector $\boldsymbol{f}$ that maximizes $Q(\boldsymbol{f})$ while ensuring that the solution is made of connected components of the SNP network. However, in general, it is difficult to find a subset of SNPs that satisfies the above two properties. Given a positive integer k, the problem of finding a connected subgraph with k vertices that maximize the sum of the weights on the vertices, which is equivalent to $Q(\boldsymbol{f})$ of our case, is known to be a strongly **NP**-complete problem (Lee and Dooly, 1996). Therefore, this problem is often addressed based on enumeration-based algorithms, whose runtime grows exponentially with k. To cope with this problem, we consider an approach based on a graph-regularization scheme, which allows us to drastically reduce the runtime.

2.2 Feature selection with graph regularization

Rather than searching through all subgraphs of a given network, we reward the selection of adjacent features through graph regularization. As it is also desirable for biological interpretation, and to avoid selecting large numbers of SNPs in linkage disequilibrium, that the selected subnetworks are small in size, we reward sparse solutions. The first requirement can be addressed by means of a smoothness regularizer on the network (Ando and Zhang, 2007; Smola and Kondor, 2003), whereas the second one can be enforced with an l_0 constraint:

$$\underset{\boldsymbol{f}\in\{0,1\}^n}{\arg\max} \underbrace{\boldsymbol{c}^\top \boldsymbol{f}}_{\text{association}} - \underbrace{\lambda \boldsymbol{f}^\top \boldsymbol{L}\boldsymbol{f}}_{\text{connectivity}} - \underbrace{\eta ||\boldsymbol{f}||_0}_{\text{sparsity}} \tag{1}$$

where $\boldsymbol{L}$ is the Laplacian of the SNP network. $\boldsymbol{L}$ is defined as $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D}$ is the diagonal matrix where $\boldsymbol{D}_{p,p}$ is the degree of node p. Here, we directly minimize the number of nonzero entries in f and do not require the proxy of an l_1 constraint to achieve sparsity (of course in the case of binary indicators, l_1 and l_0 norms are equivalent). Positive parameters λ and η control the importance of the connectedness of selected features and the sparsity regularizer, respectively.

As $W_{p,q} = 1$ if q is a neighbor of p (also written as $p \sim q$), and 0 otherwise, if we denote by $\mathcal{N}(p)$ the neighborhood of p, then the degree of p can be rewritten $D_{p,p} = \sum_{q\in\mathcal{N}(p)} 1$. The second term in Equation (1) can therefore be rewritten as

$$\boldsymbol{f}^\top \boldsymbol{L}\boldsymbol{f} = \sum_{p\sim q} (f_p - f_q)^2, \tag{2}$$

and the problem in Equation (1) is equivalent to

$$\underset{\boldsymbol{f}\in\{0,1\}^n}{\arg\min} \sum_{p=1}^{n} f_p(c_p - \eta) - \lambda \sum_{p\sim q} (f_p - f_q)^2. \tag{3}$$

As $(f_p - f_q)^2$ is 1 if $f_p \neq f_q$ and 0 otherwise, it can be seen that the connectivity term in Equation (1) penalizes the selection of SNPs not connected to one another, as well as the selection of only subnetworks of connected components of the SNP network. It does not prohibit the selection of several disconnected subnetworks. In particular, solutions may include individual SNPs fully disconnected from the other selected SNPs. Also, as $||\boldsymbol{f}||_0 = \mathbb{1}_n^\top \boldsymbol{f}$ in our case, the sparsity term in Equation (1) is equivalent to reducing the individual SNP scores $\boldsymbol{c}$ by a constant $\eta > 0$.

2.3 Min-Cut solution

A *cut* on a weighted graph over vertices $V := \{1, \ldots, n\}$ is a partition of V in a nonempty set S and its complementary $V \setminus S$. The *cut-set* of the cut is the set of edges whose end vertices belong to different sets of the partition. The *minimum cut* of the graph is the cut such that the sum of the weights of the edges belonging to its cut-set is minimum. If $\boldsymbol{A}$ is the adjacency matrix of the graph, finding the minimum cut is equivalent to finding $S \subset V$ that minimizes the *cut-function* $\sum_{p\in S} \sum_{q\notin S} \boldsymbol{A}_{p,q} = \sum_{p=1}^{n} \sum_{q=1}^{n} \boldsymbol{f}_p(1 - \boldsymbol{f}_q)\, A_{p,q}$ where $\boldsymbol{f}_p$ is 1 if $p \in S$ and 0 otherwise. Given two vertices s and t, an *s/t-cut* is a cut such that $s \in S$ and $t \in V \setminus S$. According to the max-flow min-cut theorem (Papadimitriou and Steiglitz, 1982), a minimum s/t-cut can be efficiently computed with the maximum flow algorithm (Goldberg and Tarjan, 1988).

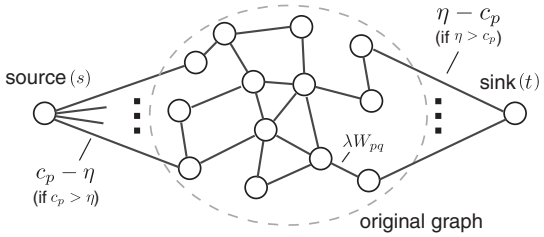


Fig. 2. Graph for the s/t-min-cut formulation of the selection of networks of genetic markers

PROPOSITION 1. *Given a graph $\mathcal{G}$ of adjacency matrix $\boldsymbol{W}$, solving the graph-regularized feature selection problem formalized in* Equation (1) *is equivalent to finding an s/t min-cut on the graph, depicted in Figure 2, whose vertices are that of $\mathcal{G}$, augmented by two additional nodes s and t, and whose edges are given by the adjacency matrix $\boldsymbol{A}$, where $\boldsymbol{A}_{p,q} = \lambda \boldsymbol{W}_{p,q}$ for $1 \le p,q \le n$ and*

$$\boldsymbol{A}_{s,p} = \begin{cases} c_p - \eta & \text{if } c_p > \eta \\ 0 & \text{otherwise} \end{cases} \text{ and } \boldsymbol{A}_{t,p} = \begin{cases} \eta - c_p & \text{if } c_p < \eta \\ 0 & \text{otherwise} \end{cases} \quad (p = 1, \ldots, n).$$

PROOF. The problem in Equation (1) is equivalent to

$$\underset{\boldsymbol{f}\in\{0,1\}^n}{\arg\min} (\eta \mathbb{1}_n - \boldsymbol{c})^\top \boldsymbol{f} + \lambda \boldsymbol{f}^\top \boldsymbol{L}\boldsymbol{f}. \tag{4}$$

The second term of the objective is a cut-function over $\mathcal{G}$:

$$\boldsymbol{f}^\top \boldsymbol{L}\boldsymbol{f} = \sum_{p=1}^{n} f_p \left(D_{p,p} - \sum_{q=1}^{n} W_{p,q} f_q \right) = \sum_{p=1}^{n} \sum_{q=1}^{n} W_{p,q} f_p (1 - f_q).$$

The first term can also be encoded as a cut-function by introducing to artificial nodes s and t:

$$\begin{aligned} \sum_{p=1}^{n} (\eta - c_p)\boldsymbol{f}_p &= \sum_{\substack{p\in S \\ c_p < \eta}} (\eta - c_p) + \sum_{\substack{p\in V \\ c_p \ge \eta}} (\eta - c_p) - \sum_{\substack{p\notin S \\ c_p \ge \eta}} (\eta - c_p) \\ &= \sum_{p=1}^{n} \boldsymbol{A}_{s,p} \boldsymbol{f}_s (1 - \boldsymbol{f}_p) + \sum_{p=1}^{n} \boldsymbol{A}_{p,t} \boldsymbol{f}_p (1 - \boldsymbol{f}_t) + \mathcal{C} \end{aligned}$$

where $\mathcal{C} = \sum_{p\in V; c_p \ge \eta} (\eta - c_p)$ is a constant, $\boldsymbol{f}_s = 1$, $\boldsymbol{f}_t = 0$ and $\boldsymbol{A}$ is defined as aforementioned. As $\boldsymbol{f}_s = 1$ and $\boldsymbol{f}_t = 0$ enforce that $s \in S$ and $t \notin S$, it follows that Equation (1) is an s/t min-cut problem on the transformed graph defined by the adjacency matrix $\boldsymbol{A}$ over the vertices of $\mathcal{G}$ augmented by s and t. The aforementioned still holds if $\boldsymbol{W}$ is a weighted adjacency matrix, and therefore the min-cut reformulation can also be applied to a weighted network. ■

It is therefore possible to use maximal flow algorithms to efficiently optimize the objective function defined in Equation (1) and select a small number of connected SNPs maximally associated with a phenotype. In our implementation, we use the Boykov–Kolmogorov algorithm (Boykov and Kolmogorov, 2004). Although its worst case complexity is in $\mathcal{O}(n^2 n_E n_C)$, where n_E is the number of edges of the graph and n_C the size of the minimum cut, it performs much better in practice, particularly when the graph is sparse. We refer to this method as SConES, for Selecting CONnected Explanatory SNPs.

3 RESULTS

We evaluate the ability of SConES to detect networks of trait-associated SNPs on simulated datasets and on datasets from an association mapping study in *A.thaliana*.

3.1 Experimental settings

For all of our experiments, we consider the three SNP networks defined in Section 1: the GS network, the GM network and the GI network. For SConES, the association term $\boldsymbol{c}$ is derived from Linear SKAT (Wu *et al.*, 2011), which makes it possible to correct for covariates (and therefore population structure). SKAT has been devised to address rare variants association problems by grouping SNPs to achieve statistical significance, but it can equally be applied to common variants.

Univariate linear regression: As a baseline for comparisons, we run a linear regression-based single-SNP search for association and select those SNPs that are significantly associated with the phenotype (Bonferroni-corrected P-value ≤ 0.05).

Linear mixed model: Similarly, we run a linear mixed model (LMM) single-SNP search for association (Lippert *et al.*, 2011) and select those SNPs that are significantly associated with the phenotype (Bonferroni-corrected P-value ≤ 0.05).

Lasso: To compare SConES to a method that also considers all additive effects of SNPs simultaneously with a sparsity constraint, but without any network regularization, we also run a Lasso regression (Tibshirani, 1994), using the SLEP implementation (http://www.public.asu.edu/~jye02/Software/SLEP) of the Lasso.

ncLasso: In addition, we compare SConES to the network-constrained Lasso ncLasso (Li and Li, 2008), a version of the Lasso with sparsity and graph-smoothing constraints equivalent to that of SConES. Given a genotype matrix $\boldsymbol{G}$ and a phenotype $\boldsymbol{r}$, ncLasso solves the following relaxed problem ($\boldsymbol{f} \in \mathbb{R}^n$):

$$\arg\min_{\boldsymbol{f}\in\mathbb{R}^n} \frac{1}{2}\left|\left|\boldsymbol{G}\boldsymbol{f}-\boldsymbol{r}\right|\right|_2^2+\lambda\boldsymbol{f}^\top\boldsymbol{L}\boldsymbol{f}+\eta\left|\left|\boldsymbol{f}\right|\right|_1 \quad (5)$$

The solution for ncLasso proposed by Li and Li (2008) requires to compute and store a single value decomposition of $\boldsymbol{L}$ and is therefore not applicable when its sizes exceeds $100\,000 \times 100\,000$ by far. However, a similar solution can be obtained by decomposing $\boldsymbol{L}$ as the product of the network's incidence matrix with its transpose, an approach that is much faster (particularly when the network is sparse).

groupLasso and graphLasso: Eventually, we also compare our method to the nonoverlapping group Lasso (Jacob *et al.*, 2009). The nonoverlapping group Lasso solves the following relaxed problem:

$$\arg\min_{\boldsymbol{f}\in\mathbb{R}^n} \frac{1}{2}\left|\left|\boldsymbol{G}\boldsymbol{f}-\boldsymbol{r}\right|\right|_2^2+\lambda\sum_{g\in\mathcal{G}}\left|\left|\boldsymbol{f}^{\mathcal{G}}\right|\right|_2 \quad (6)$$

where $\mathcal{G}$ is a set of (possibly overlapping) predefined groups of SNPs. We consider the following two versions:

- graphLasso, for which the groups are directly defined from the same networks as considered for SConES as all pairs of vertices connected by an edge;
- groupLasso, for which the groups are defined sensibly as follows:
  - GS groups: pairs of adjacent SNPs (this gives raise to the same groups as for graphLasso with the sequence network);
  - GM groups: SNPs near the same gene;
  - GI groups: SNPs near either member of two interacting genes. Here, SNPs near genes that are not in the interaction network get grouped by gene.

We use the SLEP implementation of the nonoverlapping group Lasso, combined with the trick described by Jacob *et al.* (2009) to compute the overlapping group Lasso by replicating features in nonoverlapping groups.

Setting the parameters: All methods considered, except for the univariate linear regression, have parameters (e.g. λ and η in the case of SConES) that need to be optimized. In our experiments, we run 10-fold cross-validation grid-search experiments over ranges of values of the parameters: seven values of λ and η each for SConES and ncLasso and seven values of the parameter λ for the Lasso and the nonoverlapping group Lasso (ranging from 10^{-3} to 10^3). We then pick as optimal the parameters leading to the most stable selection and report as finally selected the features selected in all folds. More specifically, we define stability according to a consistency index similar to that of Kuncheva (2007). The consistency index between two feature sets S and S' is defined as $I_C(S,S') = \frac{n|S\cap S'|-|S||S'|}{n\min(|S|,|S'|)-|S||S'|}$ (Details can be found in the Supplementary Materials). For an experiment with k folds, the consistency is computed as the average of the $k(k-1)/2$ pairwise consistencies between the sets of features selected over each fold.

3.2 Runtime

We first compare the CPU runtime of SConES with that of the linear regression, ncLasso and graphLasso. To assess the performance of our methods, we simulate from 100 to 200 000 SNPs for 200 individuals and generate exponential random networks with a density of 2% (chosen as an upper limit on the density of currently available gene-gene interaction networks) between those SNPs.

We report the real CPU runtime of one cross-validation, for set parameters, over a single AMD Opteron CPU (2048 KB, 2600 MHz) with 512 GB of memory, running Ubuntu 12.04 (Fig. 3). Across a wide range of numbers of SNPs, SConES is at least two orders of magnitude faster than graphLasso and one order of magnitude faster than ncLasso.

3.3 Simulations

To assess the performance of our methods, we simulate phenotypes for $m = 500$ real *A.thaliana* genotypes (214 051 SNPs), chosen at random among those made available by Horton *et al.* (2012), and the *A.thaliana* protein–protein interaction information from The Arabidopsis Internet Resource (TAIR, http://www.arabidopsis.org/portals/proteome/proteinInteract.jsp, resulting in 55 584 646 SNP–SNP connections). We use a window size of 20 000 bp to define proximity of a SNP to a gene, in accordance with the threshold used for the interpretation of GWAS results in Atwell *et al.* (2010). Restricting ourselves to 1000 randomly chosen SNPs with minor allele

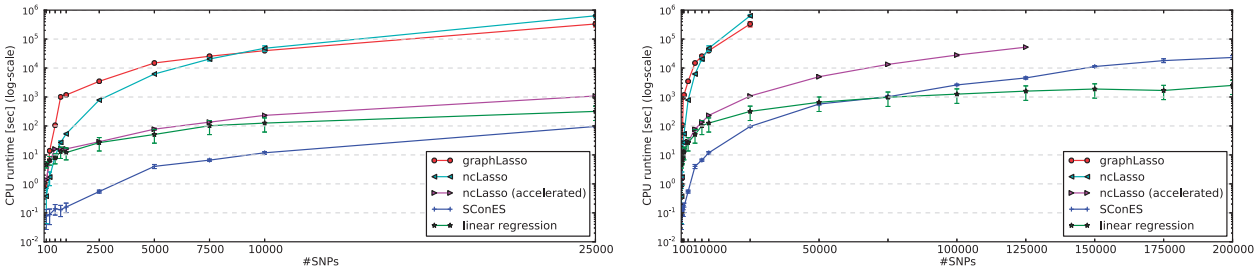


Fig. 3. Real CPU runtime comparison between univariate linear regression, ncLasso, nonoverlapping group Lasso and SConES, from 100 to 25 000 SNPs (left) and from 100 to 200 000 SNPs (right). 'ncLasso' refers to the original implementation suggested by Li and Li (2008) and 'ncLasso (accelerated)' to the incidence-matrix-based implementation we use here. After 3 weeks, nonoverlapping group Lasso and ncLasso had not finished running for 50 000 SNPs. The accelerated version of ncLasso ran out of memory for ≥150 000 SNPs

Table 1. F-scores of SConES, compared with state-of-the-art Lasso algorithms and a baseline univariate linear regression, in six different data simulation scenarios

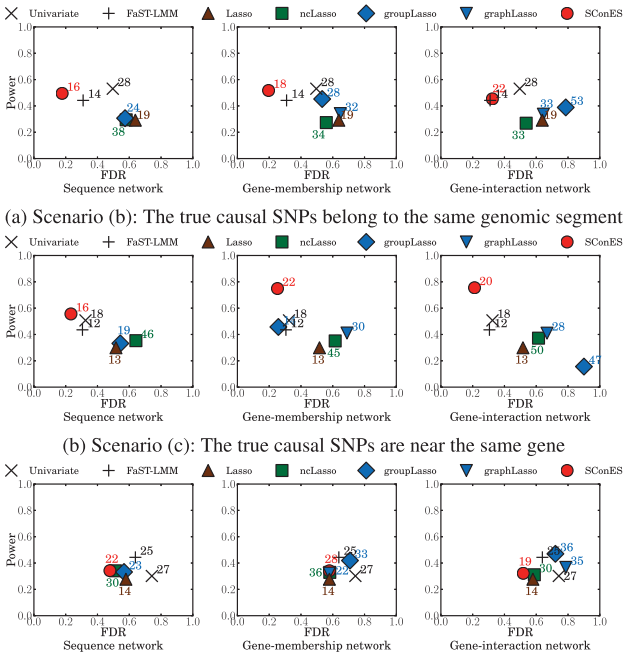
Method	(a)	(b)	(c)	(d)	(e)	(f)
Univariate	0.26 ± 0.07	0.29 ± 0.12	0.28 ± 0.14	0.27 ± 0.07	0.26 ± 0.07	0.23 ± 0.08
LMM	*0.32 ± 0.01*	0.35 ± 0.01	0.33 ± 0.01	0.36 ± 0.02	0.38 ± 0.01	*0.33 ± 0.01*
Lasso	**0.35 ± 0.01**	0.32 ± 0.02	0.36 ± 0.01	0.36 ± 0.01	0.37 ± 0.01	0.32 ± 0.01
ncLasso						
GS	0.17 ± 0.01	0.25 ± 0.02	0.25 ± 0.01	0.45 ± 0.01	0.38 ± 0.02	0.30 ± 0.01
GM	0.17 ± 0.01	0.26 ± 0.02	0.26 ± 0.02	0.38 ± 0.01	0.29 ± 0.01	0.27 ± 0.01
GI	0.19 ± 0.01	0.26 ± 0.02	0.26 ± 0.02	0.43 ± 0.02	0.34 ± 0.02	0.28 ± 0.01
groupLasso						
GS	0.23 ± 0.01	0.30 ± 0.01	0.34 ± 0.01	0.37 ± 0.01	0.36 ± 0.02	0.32 ± 0.01
GM	0.12 ± 0.00	0.44 ± 0.02	0.55 ± 0.01	*0.50 ± 0.01*	*0.40 ± 0.01*	*0.33 ± 0.01*
GI	0.09 ± 0.00	0.26 ± 0.02	0.11 ± 0.01	**0.54 ± 0.01**	*0.40 ± 0.01*	**0.34 ± 0.01**
graphLasso						
GS	0.23 ± 0.01	0.30 ± 0.01	0.34 ± 0.01	0.37 ± 0.01	0.36 ± 0.02	0.32 ± 0.01
GM	0.23 ± 0.01	0.28 ± 0.01	0.33 ± 0.01	0.36 ± 0.01	0.31 ± 0.01	0.31 ± 0.01
GI	0.22 ± 0.01	0.28 ± 0.01	0.34 ± 0.01	0.33 ± 0.01	0.30 ± 0.01	0.27 ± 0.01
SConES						
GS	0.21 ± 0.01	*0.55 ± 0.04*	0.57 ± 0.04	*0.50 ± 0.01*	**0.43 ± 0.02**	*0.33 ± 0.02*
GM	0.19 ± 0.02	**0.58 ± 0.03**	*0.75 ± 0.03*	0.49 ± 0.01	*0.40 ± 0.02*	0.32 ± 0.02
GI	0.20 ± 0.02	0.48 ± 0.03	**0.78 ± 0.03**	0.49 ± 0.01	0.39 ± 0.01	**0.34 ± 0.02**

Note: The true causal SNPs are (a) unconnected; (b) adjacent on the GS; (c) near the same gene; (d) near either of the same two connected genes; (e) near either of the same three connected genes; (f) near either of the same five connected genes. Best performance in bold and second best in italics.

frequency larger than 10%, we pick 20 of the SNPs to be causal, and generate phenotypes $y_i = w^\top g_i + \epsilon$, where both the support weights w and the noise ε are normally distributed. We consider the following scenarios: (a) the causal SNPs are randomly distributed in the network; (b) the causal SNPs are adjacent on the genomic sequence; (c) the causal SNPs are near the same gene; (d–f) the causal SNPs are near either of two, three and five interacting genes, respectively. We then select SNPs using univariate linear regression, Lasso, ncLasso, the two flavors of nonoverlapping group Lasso and SConES as described in Section 3.1. We repeat each experiment 30 times and compare the selected SNPs of either approach with the true causal ones in terms of power (fraction of causal SNPs selected) or false discovery rate (FDR, fraction of selected SNPs that are not causal). We summarize the results with F-scores (harmonic mean of power and one minus FDR) in Table 1.

As SConES returns a binary feature selection rather than a feature ranking, it is not possible to draw FDR curves or compare powers at same FDR as is often done when evaluating such methods. Figure 4 presents the average FDR and power of the different algorithms under three of the scenarios, depending on the network used. The closer the FDR power point representing an algorithm to the upper-left corner, the better this algorithm at maximizing power while minimizing FDR. As it is easy to get better power by selecting more SNPs, we also report on the same figure the number of SNPs selected by each algorithm and show that it remains reasonably close to the true value of 20 causal SNPs.

SConES is systematically better than its state-of-the-art comparison partners at leveraging structural information to retrieve the connected SNPs that were causal. Only when the groups perfectly match the causal structure [Scenario (d)] can



(a) Scenario (b): The true causal SNPs belong to the same genomic segment

(b) Scenario (c): The true causal SNPs are near the same gene

(c) Scenario (f): The true causal SNPs are near any of five interacting genes

Fig. 4. Power and FDR of SConES, compared with state-of-the-art Lasso algorithms and a baseline univariate linear regression, in three different data simulation scenarios. Best methods are closest to the upper-left corner. Numbers denote the number of SNPs selected by the method

groupLasso outperform SConES. Although the performance of SConES and ncLasso does depend on the network, the nonoverlapping group Lasso is much more sensitive to the definition of its groups. Furthermore, we observe that removing a small fraction (1–15%) of the edges between causal features does not harm the performance of SConES (Supplementary Table S1). This means that SConES is robust to missing edges, an important point when the biological network used is likely to be incomplete. Nevertheless, the performance of SConES, as that of all other network-regularized approaches, is strongly negatively affected when the network is entirely inappropriate [Scenario (a)]. In addition, the decrease in performance from Scenario (c) to Scenario (f), when the number of interacting genes near which the causal SNPs are located increases from 1 to 5, indicates that SConES, like its structure-regularized comparison partners, performs better when the causal SNPs are less spread out in the network. Finally, ncLasso is both slower and less performing than SConES. This indicates that solving the feature selection problem we pose directly, rather than its relaxed version, allows for better recovery of true causal features.

3.4 *Arabidopsis* flowering time phenotypes

We then apply our method to a large collection of 17 *A.thaliana* flowering times phenotypes from Atwell *et al.* (2010) (up to 194 individuals, 214 051 SNPs). The groups and networks are again derived from the TAIR protein–protein interaction data. We filter out SNPs with a minor allele frequency lower than 10%, as is typical in *A.thaliana* GWAS studies. We use the first principal components of the genotypic data as covariates to correct for population structure (Price *et al.*, 2006): the number of principal components is chosen by adding them one by one until the genomic control is close to 1 (see Supplementary Figure S1).

The direct competitors of SConES on this problem are the methods that also impose graph constraints on the SNPs they select, namely, graphLasso and ncLasso. However, graphLasso does not scale to datasets such as ours with >200 k SNPs (see Fig. 3). Hence, we had to exclude it from our experiments. While even our accelerated implementation of ncLasso could not be run on >125 000 SNPs in our simulations, the networks derived for *A.thaliana* are sparser than that used in the simulations, which makes it possible to run ncLasso on this data.

Instead, we compare SConES to ncLasso and groupLasso, which uses pairs of neighboring SNPs, SNPs from the same gene or SNPs from interacting genes as predefined groups. The groupLasso on sequence-neighboring SNPs is identical to graphLasso on the sequence network, which is the only instance of graphLasso whose computation is practically feasible on this dataset. We run Lasso, ncLasso, groupLasso and SConES on the flowering time phenotypes as described in Section 3.1. However, for many of the phenotypes, the Lasso approaches select large number of SNPs (>10 000), which makes the results hard to interprete. Using cross-validated predictivity, as is generally done for Lasso, still does not entirely solve this issue, particularly for large group sizes (see Supplementary Tables S2 and S3). We therefore filter out solutions containing >1% of the total number of SNPs before using consistency to select the optimal parameters.

To evaluate the quality of the SNPs selected, we perform ridge regression on each phenotype in a cross-validation scheme that uses only the selected SNPs and report its average Pearson's squared correlation coefficient in Figure 5. We also report, as an additional baseline, the cross-validated predictivity of a standard best linear unbiased prediction (BLUP) (Henderson, 1975). Although the features selected by groupLasso + GS achieve higher predictivity than SConES + GS on most phenotypes, the features selected by SConES + GM are at least as predictive as those selected by groupLasso + GM in two thirds of the phenotypes; the picture is the same for SConES + GI, whose selected SNPs are on average more predictive than those of groupLasso + GI. The superiority of groupLasso in that respect is to be expected, as predicitivity is directly optimized by the regression. Also in 80% of the cases, if any of the feature selection methods achieves high predictivity ($R^2 > 0.6$), SConES outperforms all other methods including BLUP.

Next, we checked whether the selected SNPs from the three methods coincide with flowering time genes from the literature. We report in Table 2 the number of SNPs selected by each of the methods and the proportion of these SNPs that are near flowering time candidate genes listed by Segura *et al.* (2012). Here, the picture is reversed: SConES + GS and groupLasso + GI retrieve the highest ratio of SNPs near candidate genes, whereas groupLasso + GS, SConES + GI and SConES + GM show lower ratios. At first sight, it seems surprising that the methods with highest predictive power retrieve the least SNPs near candidate genes.

To further investigate this phenomenon, we record how many distinct flowering time candidate genes are retrieved on average by the various methods. A gene is considered retrieved if the

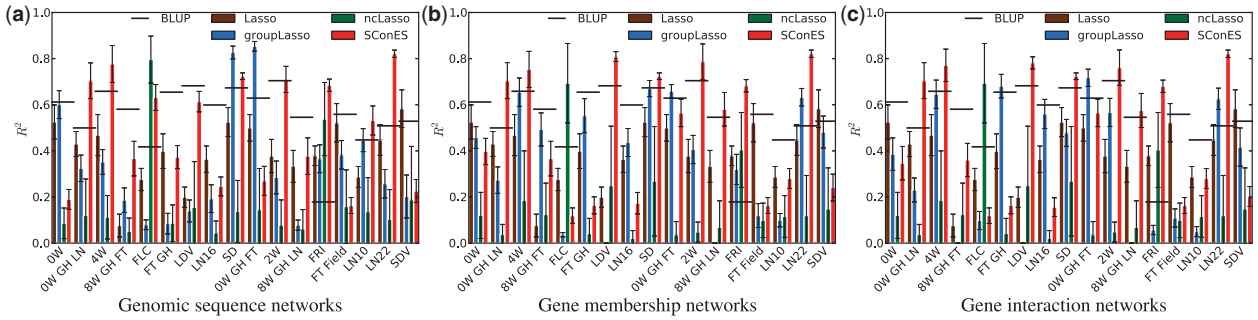


Fig. 5. Cross-validated predictivity (measured as Pearson's squared correlation coefficient between actual phenotype and phenotype predicted by a ridge-regression over the selected SNPs) of SConES compared with that of Lasso, groupLasso and ncLasso. Horizontal bars indicate cross-validated BLUP predictivity

Table 2. Associations detected close to known candidate genes, for all flowering time phenotypes of *Arabidopsis thaliana*

Phenotype	Univariate	LMM	Lasso	groupLasso			ncLasso			SConES		
				GS	GM	GI	GS	GM	GI	GS	GM	GI
0 W	0/3	0/0	1/29	33/288	59/706	144/547	40/1077	14/318	14/318	**123/271**	0/85	0/69
0 W GH LN	0/0	0/0	2/20	13/205	54/478	**128/321**	31/981	11/320	11/320	92/1251	92/1252	92/1253
4 W	1/8	1/2	15/129	7/52	48/1489	**80/436**	2/238	6/298	6/298	104/1670	66/1078	42/859
8 W GH FT	0/5	0/1	10/143	**5/16**	66/1470	0/0	14/427	11/398	11/398	26/322	26/322	26/319
FLC	0/1	0/1	1/31	2/95	0/101	0/214	4/135	1/35	1/35	**115/1592**	0/2	0/2
FT GH	0/1	2/10	**7/46**	8/106	90/841	177/1417	37/1434	42/1709	42/1709	0/626	0/59	0/59
LDV	0/4	**1/2**	10/80	8/32	0/0	0/0	14/437	7/177	7/177	39/674	86/1381	54/1091
LN16	0/5	0/0	9/222	0/95	138/957	89/1307	22/1094	33/1323	33/1323	**73/73**	0/3	0/4
SD	0/2	0/1	3/36	36/569	51/863	84/721	20/466	10/224	10/224	**7/59**	**7/59**	**7/59**
0 W GH FT	0/9	**1/3**	20/194	49/654	52/898	241/1258	63/1597	84/1997	84/1997	0/6	29/317	29/317
2 W	0/12	0/6	4/36	7/79	93/610	**126/810**	28/1006	43/1256	43/1256	76/756	78/1185	25/892
8 W GH LN	0/2	0/3	8/122	13/168	0/0	0/0	19/493	21/501	21/501	**11/73**	75/776	68/757
FRI	6/11	5/9	6/18	8/64	8/20	**10/10**	2/9	2/4	2/4	101/1266	101/1271	101/1274
FT Field	2/4	0/0	1/79	5/37	51/221	**52/72**	18/709	5/238	5/238	4/8	4/8	4/8
LN10	0/1	0/0	0/12	2/34	**18/121**	0/202	12/644	12/649	12/649	165/1921	0/91	0/91
LN22	2/14	0/0	6/65	0/12	33/894	81/1023	23/501	26/506	26/506	**140/1378**	**140/1378**	140/1378
SDV	0/5	0/1	4/208	3/94	1/721	105/936	14/379	15/384	15/384	**53/454**	0/8	0/8

Note: We report the number of selected SNPs near candidate genes, followed by the total number of selected SNPs. Largest ratio in bold.

method selects a SNP near it. Our results are shown in Table 3. Methods retrieving a large fraction of SNPs near candidate genes do not necessarily retrieve the largest number of distinct candidate genes. Good predictive power, as shown in Figure 5, however, seems to correlate with the number of distinct candidate genes selected by an algorithm, not with the percentage of selected SNPs near candidate genes. groupLasso + GI has the highest fraction of candidate gene SNPs among all methods but detects only three distinct candidate genes. This is probably due to groupLasso selecting entire genes or gene pairs; if groupLasso detects a candidate gene, it will pick most of the SNPs near that gene, which leads to its high candidate SNP ratio in Table 2.

We also compare the selected SNPs to those deemed significant by a LMM ran on the full data (see Supplementary Table S4). SConES systematically recovers more of those SNPs than the Lasso approaches.

To summarize, SConES is able to select SNPs that are highly predictive of the phenotype. Among all methods, SConES + GM discovers the largest number of distinct genes whose involvement in flowering time is supported by the literature.

4 DISCUSSION AND CONCLUSIONS

In this article, we defined SConES, a novel approach to multi-locus mapping that selects SNPs that tend to be connected in a given biological network without restricting the search to predefined sets of loci. As the optimization of SConES can be solved by maximum flow, our solution is computationally efficient and scales to whole-genome data. Our experiments show that our method is one to two orders of magnitude faster than the state-of-the-art Lasso-based comparison partners and can therefore easily scale to hundreds of thousands of SNPs.

Table 3. Summary statistics, averaged over the *Arabidopsis thaliana* flowering time phenotypes: average total number of selected SNPs ('No of SNPs'), average proportion of selected SNPs near candidate genes ('near candidate genes') and average number of different candidate genes recovered ('candidate genes hit')

Method	No of SNPs	Near candidate genes	Candidate genes hit
Univariate	5	0.09	0.35
LMM	2	0.12	0.35
Lasso	86	0.09	3.82
groupLasso GS	153	0.10	4.35
groupLasso GM	611	0.09	1.35
groupLasso GI	546	0.20	2.65
ncLasso GS	684	0.04	4.88
ncLasso GM	608	0.06	4.59
ncLasso GI	608	0.06	4.59
SConES GS	729	0.18	11.53
SConES GM	546	0.08	14.82
SConES GI	496	0.07	12.24

In simulations, SConES is better at leveraging the structure of the biological network to recover causal SNPs.

On real GWAS data from *A.thaliana*, the predictive ability of the features selected by SConES is superior to that of groupLasso on two of the three network types we consider. When using more biological information (gene membership or interactions), SConES tends to recover more distinct explanatory genes than groupLasso, resulting in better phenotypic prediction.

The constraints imposed by groupLasso and SConES are different: although the groups given to groupLasso and the networks passed to SConES come from the same information, the groups force many more SNPs to be selected simultaneously when they may not bring much more information. This gives SConES more flexibility and makes it less vulnerable to ill-defined groups or networks, which is especially desirable in the light of the current noisiness and incompleteness of biological networks. Our results on the GS network actually indicate that graphLasso, using pairs of network edges as groups, may achieve the same flexibility as SConES; unfortunately, it is too computationally demanding to be run on the most informative networks.

We currently derive the SNP networks from neighborhood along the genome sequence, closeness to a same gene or proximity to interacting proteins. Refining those networks and exploring other types of networks as well as understanding the effects of their topology and density is one of our next projects.

Although we do not explicitly consider linkage disequilibrium, the l_0 sparsity constraint of SConES should enforce that when several correlated SNPs are associated with a phenotype, a single one of them is picked. On the other hand, if SConES is given a GS network such as the one we describe, the graph smoothness constraint will encourage nearby SNPs to be selected together, leading to the selection of subsequences that are likely to be haplotype blocks. Such a network should therefore only be used when the goal of the experiment is to detect consecutive sequences of associated SNPs.

For now, SConES considers an additive model between genetic loci. Future work includes taking pairwise multiplicative effects into account. Replacing the association term in Equation (1) by a sum over pairs of SNPs rather than over individual SNPs results in a maximum flow problem over a fully connected network of SNPs, which cannot be solved straightforwardly, if only because the resulting adjacency matrix is too large to fit in memory on a regular computer. It might be possible, however, to leverage some of the techniques used for two-locus GWAS (Achlioptas *et al.*, 2011; Kam-Thong *et al.*, 2012) to help solve this problem.

Extensions of SConES to other models include the use of mixed models to account for population structure and other confounders. This is currently a challenge, as it is unclear how to derive additive test statistics from such models.

An interesting extension to study would replace the Laplacian by a random-walk-based matrix, derived from powers of the adjacency matrix, so as to treat disconnected SNPs that are close-by in the networks differently from those that are far apart. Although we already observe that SConES is robust to edge removal, this would likely make it more resistant to missing edges.

Another important extension of SConES is to devise a way to evaluate the statistical significance of the set of selected SNPs. Regularized feature selection approaches such as SConES or its Lasso comparison partners do not lend themselves well to the computation of *P*-values. Permutation tests could be an option, but the number of permutations to run is difficult to evaluate as is that of hypotheses tested. Another possibility would be to implement the multiple-sample splitting approach proposed by Meinshausen *et al.* (2009). However, the loss of power from performing selection on only subsets of the samples is too large, given the sizes of current genomic datasets, to make this feasible. Therefore, evaluating statistical significance and controlling FDRs of Lasso and SConES approaches alike remain a challenge for the future.

Finally, further exciting research topics include applying SConES to larger datasets from human disease consortia (we estimate it would require less than a day to run on a million of SNPs) and extending it to the detection of shared networks of markers between multiple phenotypes.

ACKNOWLEDGEMENTS

The authors thank Recep Colak, Barbara Rakitsch and Nino Shervashidze for fruitful discussions.

Funding: C.A. is funded by an Alexander von Humboldt fellowship. This work was partially funded by the DFG project Kernels for Large, Labeled Graphs (LaLa).

Conflict of Interest: none declared.

REFERENCES

Achlioptas,P. *et al.* (2011) *Two-Locus Association Mapping In Subquadratic Time. KDD '11.* ACM, New York, NY, USA, pp. 726–734.

Ando,R.K. and Zhang,T. (2007) Learning on graph with Laplacian regularization. In: Schölkopf,B. and Hoffman,T. (eds.) *Advances in Neural Information Processing Systems 19.*

Atwell,S. *et al.* (2010) Genome-wide association study of 107 phenotypes in *Arabidopsis thaliana* inbred lines. *Nature*, **465**, 627–631.

Boykov,Y. and Kolmogorov,V. (2004) An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE T. Pattern Anal.*, **26**, 1124–1137.

Cantor,R.M. *et al.* (2010) Prioritizing GWAS results: a review of statistical methods and recommendations for their application. *Am. J. Hum. Genet.*, **86**, 6–22.

Cho,S. *et al.* (2010) Joint identification of multiple genetic variants via elastic-net variable selection in a genome-wide association analysis. *Ann. Hum. Genet.*, **74**, 416–428.

Chuang,H.Y. *et al.* (2007) Network-based classification of breast cancer metastasis. *Mol. Syst. Biol.*, **3**, 140.

Fridley,B.L. and Biernacka,J.M. (2011) Gene set analysis of SNP data: benefits, challenges, and future directions. *Eur. J. Hum. Genet.*, **19**, 837–843.

Goldberg,A.V. and Tarjan,R.E. (1988) A new approach to the maximum-flow problem. *J. ACM*, **35**, 921–940.

Gretton,A. *et al.* (2005) Measuring statistical dependence with Hilbert-Schmidt norms. In: Sanjay,J. *et al.* (eds.) *Algorithmic Learning Theory, 16th International Conference, ALT 2005, Singapore, October 8-11, 2005, Proceedings. Lecture Notes in Computer Science 3734 Springer 2005*. ALT. Springer-Verlag, pp. 63–77.

Henderson,C.R. (1975) Best linear unbiased estimation and prediction under a selection model. *Biometrics*, **31**, 423–447.

Horton,M.W. *et al.* (2012) Genome-wide patterns of genetic variation in worldwide *Arabidopsis thaliana* accessions from the RegMap panel. *Nat. Genet.*, **44**, 212–216.

Huang,J. *et al.* (2009) *Learning with Structured Sparsity*. In: Andrea,P.D. *et al.* (eds.) *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*. ACM, New York, NY, USA, pp. 417–424.

Jacob,L. *et al.* (2009) *Group Lasso with Overlap and Graph Lasso*. In: Andrea,P.D. *et al.* (eds.) *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*. ACM, New York, NY, USA, pp. 433–440.

Jie,B. *et al.* (2012) Structural feature selection for connectivity network-based MCI diagnosis. In: Yap,P.T. *et al.* (ed.) *Multimodal Brain Image Analysis,* Volume 7509 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg, pp. 175–184.

Kam-Thong,T. *et al.* (2012) GLIDE: GPU-based linear regression for detection of epistasis. *Hum. Hered.*, **73**, 220–236.

Kuncheva,L.I. (2007) A stability index for feature selection. In: Vladan,D. (ed.) *Proceedings of the 25th IASTED International Multi-Conference: artificial intelligence and applications*. IASTED/ACTA Press, Innsbruck, Austria.

Le Saux,B. and Bunke,H. (2005) Feature selection for graph-based image classifiers. In: Marques,J. *et al.* (ed.) *Pattern Recognition and Image Analysis,* Volume 3523 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg, pp. 147–154.

Lee,H.F. and Dooly,D.R. (1996) Algorithms for the constrained maximum-weight connected graph problem. *Nav. Res. Logist.*, **43**, 985–1008.

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

Lippert,C. *et al.* (2011) FaST linear mixed models for genome-wide association studies. *Nat. Meth.*, **8**, 833–835.

Liu,J. *et al.* (2012) Incorporating group correlations in genome-wide association studies using smoothed group lasso. *Biostatistics*, **14**, 205–219.

Mairal,J. and Yu,B. (2011) Path coding penalties for directed acyclic graphs. In: *Proceedings of the 4th NIPS Workshop on Optimization for Machine Learning (OPT'11)*.

Manolio,T.A. *et al.* (2009) Finding the missing heritability of complex diseases. *Nature*, **461**, 747–753.

Marchini,J. *et al.* (2005) Genome-wide strategies for detecting multiple loci that influence complex diseases. *Nat. Genet.*, **37**, 413–417.

Meinshausen,N. *et al.* (2009) *P*-values for high-dimensional regression. *J. Am. Stat. Assoc.*, **104**, 1671–1681.

Nacu,Ş. *et al.* (2007) Gene expression network analysis and applications to immunology. *Bioinformatics*, **23**, 850–858.

Papadimitriou,C.H. and Steiglitz,K. (1982) *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall Inc, Englewood Cliffs, NJ, USA.

Price,A.L. *et al.* (2006) Principal components analysis corrects for stratification in genome-wide association studies. *Nat. Genet.*, **38**, 904–909.

Rakitsch,B. *et al.* (2012) A lasso multi-marker mixed model for association mapping with population structure correction. *Bioinformatics.*, **29**, 206–214.

Segura,V. *et al.* (2012) An efficient multi-locus mixed-model approach for genome-wide association studies in structured populations. *Nat. Genet.*, **44**, 825–830.

Smola,A. and Kondor,R. (2003) Kernels and regularization on graphs. In: Schölkopf,B. and Warmuth,M. (eds.) *Learning Theory and Kernel Machines,* Volume 2777 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg, pp. 144–158.

Tibshirani,R. (1994) Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Series B*, **58**, 267–288.

Tsuda,K. (2011) Graph classification methods in chemoinformatics. In: Lu,H.H.S. *et al.* (ed.) *Handbook of Statistical Bioinformatics, Springer Handbooks of Computational Statistics*. Springer, Berlin Heidelberg, pp. 335–351.

Wang,D. *et al.* (2011) Identifying QTLs and epistasis in structured plant populations using adaptive mixed lasso. *J. Agric. Biol. Environ. Stat.*, **16**, 170–184.

Wu,M.C. *et al.* (2011) Rare-variant association testing for sequencing data with the sequence kernel association test. *Am. J. Hum. Genet.*, **89**, 82–93.

Zuk,O. *et al.* (2012) The mystery of missing heritability: Genetic interactions create phantom heritability. *Proc. Natl Acad. Sci. USA*, **109**, 1193–1198.