Multi-Task Feature Selection on Multiple Networks via Maximum Flows*†

Mahito Sugiyama‡§ Chloé-Agathe Azencott‡ Dominik Grimm‡¶
Yoshinobu Kawahara‖ Karsten M. Borgwardt‡¶

Abstract

We propose a new formulation of multi-task feature selection coupled with multiple network regularizers, and show that the problem can be exactly and efficiently solved by maximum flow algorithms. This method contributes to one of the central topics in data mining: How to exploit structural information in multivariate data analysis, which has numerous applications, such as gene regulatory and social network analysis. On simulated data, we show that the proposed method leads to higher accuracy in discovering causal features by solving multiple tasks simultaneously using networks over features. Moreover, we apply the method to multi-locus association mapping with *Arabidopsis thaliana* genotypes and flowering time phenotypes, and demonstrate its ability to recover more known phenotype-related genes than other state-of-the-art methods.

Keywords: Multi-task learning; Feature selection; Network regularizer; Multi-locus association mapping

1 Introduction

Knowledge discovery from structured data is one of the central topics in data mining. In particular, *networks*, or *graphs*, have attracted considerable attention in the community, as they may represent molecular, biological, social, or other types of systems whose functionality and mechanisms are far from being completely understood. Large amounts of data are now available as networks across a wide range of domains, from biological pathways in KEGG to chemical compounds in PubChem and social networks on the web.

A crucial concern when studying such systems is to determine which part of the network is responsible for performing a particular function. For instance, medical researchers are interested in determining the subset of proteins in an interaction network that governs the response to a particular treatment. Structural biologists seek to find out which part of the 3D structure of a protein, which can be modeled by a graph, is correlated with a particular function. Neuroscientists search for subgraphs in brain connectivity networks from functional MRI screens that correlate with certain types of behavior or cognitive tasks. Geneticists are interested in sets of mutations in interacting genes that may be associated with heritable diseases.

Hence the general problem of *feature selection on networks* is of broad interest across disciplines. In these networks, features coincide with vertices (nodes) and the network topology can be viewed as *a priori* knowledge about relationships between features.

The common approach to this problem is to use Lasso-based regression [28] with an ℓ_1-regularizer of the weight vector and additional *structured regularizers* that represent relationships between features. Examples include supervised [3, 14, 15] and unsupervised [26] learning methods and applications in computer vision [9] and statistical genetics [6, 18, 19]. Many of those methods have also been proposed for multi-task learning, where the commonalities between related tasks are leveraged to improve the quality of models on individual tasks.

In spite of their success, we see a number of drawbacks to regression-based approaches in this context. First, they do not scale to millions or even hundreds of thousands of features, although such a setting is common, for instance, in genetics. Second, regression-based approaches concentrate on optimizing a prediction loss, while the problem to solve is often formulated in terms of finding features that are *relevant for*, *correlated to* or *associated with* a property of interest.

These two issues have been addressed by our recent work in statistical genetics, which proposes a new formulation of network-constrained feature selection called SConES [2]. This method directly maximizes a score of association rather than minimizing a prediction error. Its optimization scheme is exact and efficient, thanks to a minimum-cut reformulation, and it has been empirically shown to recover more causal features than

[*] Supplementary information is available at http://webdav.tuebingen.mpg.de/u/karsten/Veroeffentlichungen/papers/sdm14supp.pdf

[†] M.S. is funded by the Alexander von Humboldt Foundation.

[‡] Machine Learning and Computational Biology Research Group, Max Planck Institutes Tübingen, Germany

[§] Contact: mahito.sugiyama@tuebingen.mpg.de

[¶] Zentrum für Bioinformatik, Eberhard Karls Universität Tübingen, Germany

[‖] The Institute of Scientific and Industrial Research (ISIR), Osaka University, Japan its regression-based counterparts.

An additional issue arises in multi-task settings: Most current methods [16, 32, 33] assume that the same features should be selected across all tasks. While this is reasonable for some application domains, one can think of numerous examples where this assumption is violated. For instance, lung diseases such as asthma and chronic obstructive pulmonary disease may be linked to a set of common mutations, but there is no indication that the exact same mutations are causal in both diseases.

Moreover, to the best of our knowledge, none of the multi-task approaches incorporating structured regularizers make it possible to consider different structural constraints for different tasks. However, we may want to consider different biological pathways for different diseases, or to highlight different parts of brain connectivity networks for different correlated behaviors.

To address these two issues, we propose a new formulation of SConES for *multi-task* feature selection coupled with *multiple network regularizers* to improve feature selection in each task by combining and solving multiple tasks simultaneously. The key to extending SConES to multiple tasks, which is the main technical contribution in this paper, is the *unification* of multiple networks into a single network. This strategy enables us to solve the multiple tasks as a single task, and hence we still obtain the exact solution through a minimum cut reformulation. Thanks to the efficiency of the maximum flow algorithm that solves this problem, our approach is still tractable for large-scale networks. To the best of our knowledge, this is the first non-regression-based formulation of multi-task feature selection on multiple feature space networks.

The motivating application underlying this paper is to give an efficient method for *multi-locus association mapping* in genetics. This problem, which aims at explaining the genetic basis of diseases and other observed traits, is receiving growing interest in the context of genome-wide association studies (GWAS) [21]. The goal of GWAS is to find single-nucleotide polymorphisms (SNPs), single positions that differ in the genomes of different individuals, which are significantly associated with variance in phenotype (diseases or other observed traits). Since SNPs can be viewed as features, feature selection techniques have been widely developed and applied to this problem (e.g. [1]). The proposed method holds the following advantages over the typical GWAS setting: First, the method can exploit *a priori* biological knowledge about network structures over SNPs, derived for example from protein-protein interaction networks. Second, we often have not only one but several related phenotypes for each set of SNPs, and hence our method can treat them simultaneously as multiple tasks to increase the accuracy in retrieving phenotype-related SNPs. We confirm in this paper the efficacy of our approach compared to the state-of-the-art methods in feature selection on *Arabidopsis thaliana* SNPs and flowering time phenotypes.

This paper is organized as follows: We present our approach to multi-task feature selection across several networks in Section 2, discuss related work in Section 3, evaluate our method on synthetic and real data in Section 4, and summarize our contribution in Section 5.

2 Feature Selection on Networks

We directly perform feature selection on networks. Formally, we only require, for each task, a network over the features, that is, a weighted graph $G = (V, E)$ with a set of vertices (features) V and edges E, and a function $q : V \to \mathbb{R}$ that assigns to each vertex v a quantity $q(v)$, measuring its relevance for the problem at hand.

This is different from the typical setting of multivariate data analysis, where a design matrix $\boldsymbol{X} \in \mathbb{R}^{N\times|V|}$ and a response vector $\boldsymbol{y} \in \mathbb{R}^N$ about N individuals are given. In such a framework, feature selection is usually solved as a regularized linear regression problem, where one tries to determine a subset of features of $\boldsymbol{X}$ which minimizes a prediction error of $\boldsymbol{y}$. In this context a relevance score $q(v)$ can be easily obtained by measuring the association between $\boldsymbol{y}$ and each feature of $\boldsymbol{X}$. Many techniques are available for that purpose: Pearson's correlation coefficient or the cosine similarity for linear associations, and the Hilbert-Schmidt independence criterion (HSIC) [11], sequence kernel association test (SKAT) [29], or maximum information coefficient (MIC) [25] for non-linear associations.

2.1 Single Task. We first introduce the single-task formulation of SConES [2] for feature selection. Let $f : 2^V \to \mathbb{R}$ be *additive* in the sense that it is defined as

$$f(S) := \textstyle\sum_{v\in S} q(v). \tag{2.1}$$

This function measures the goodness of a subset S of features via $q(v)$ for each feature $v \in S$.

Our goal is to find a subset $S \subset V$ which maximizes $f(S)$ under the constraints that the cardinality of S is small and its elements tend to be connected to one another. As conducting an exhaustive search over all connected subnetworks is not feasible, we formulate the problem as follows by focusing on local connectivity:

$$\begin{aligned} &\operatorname*{argmax}_{S\subset V} f(S) - g(S), \\ &g(S) := \lambda \textstyle\sum_{e\in B} w(e) + \eta|S|, \end{aligned} \tag{2.2}$$

where $B = \{\, \{v, u\} \in E \mid v \in V \setminus S,\, u \in S \,\}$ is the set of edges located at the boundary of S and $w : E \to \mathbb{R}^+$

is a weighting function. The first term in the penalty function g enforces the *connectivity* of S, as it penalizes selecting a vertex without selecting all of its neighbors. The second term enforces its *sparsity* and the cardinality of S is penalized. The two real-valued parameters λ and η control these constraints.

A major advantage of this formulation is that Equation (2.2) can be exactly solved by *maximum flow algorithms* by adding source and sink nodes to the given network G (Supplementary Note A or [2]). The smallest known time complexity of these algorithms is $O(\,|V||E|\log(|V|^2/|E|)\,)$ [10] and the Boykov-Kolmogorov algorithm [4] is more efficient in practice.

Regularization Path. An interesting property of the regularization parameter η, which was not analyzed in [2], is its *anti-monotonicity* with respect to the number of selected features. Specifically, if we denote the selected features for each η by $S(\eta)$, we have $S(\eta) \subset S(\eta')$ if and only if $\eta > \eta'$. Moreover, we can easily check that our formulation satisfies all assumptions to apply the *parametric maximum flow algorithm* [8] (Supplementary Note B). With this algorithm, we can obtain the entire *regularization path* [13] along with the changes in η without increasing the time complexity.

In practice, this property of η is particularly interesting when we are given cardinality constraints *a priori* over the size of the set of selected features. Then we can directly pick from the regularization path the solutions that fulfill these constraints.

2.2 Multiple Tasks. Our main contribution is a new, generalized formulation of SConES (Equation (2.2)) to achieve feature selection for multiple tasks simultaneously. In what follows, we assume that the set of vertices (features) V is shared all over K tasks, and for each task i we have a network $G_i = (V, E_i)$ associated with a respective scoring function q_i. Given such a set of K networks $\mathcal{G} = \{G_1, G_2, \dots, G_K\}$, the multi-task feature selection is formulated as

$$(2.3)\quad \underset{S_1,\dots,S_K \subset V}{\operatorname{argmax}} \sum_{i=1}^{K} \big(\, f_i(S_i) - g_i(S_i)\,\big) - \sum_{i<j} h(S_i, S_j),$$

$$f_i(S_i) := \sum_{v \in S_i} q_i(v),\;\; g_i(S_i) := \lambda \sum_{e \in B_i} w_i(e) + \eta |S_i|.$$

We introduce a new penalty function $h : 2^V \times 2^V \to \mathbb{R}$ defined as $h(S, S') := \mu|\, S \bigtriangleup S' \,|$, where μ is a real-valued regularization parameter and $S \bigtriangleup S'$ is the symmetric difference between them, that is, $S \bigtriangleup S' = (S \cup S') \setminus (S \cap S')$. The penalty function h represents our belief that similar networks should be associated with related features, and the larger μ, the more we enforce this belief. A large μ is thus better when it is desirable to select the same features across tasks.

Here we show that this problem can be reduced to a single-task feature selection similar to that of Equation (2.2) and thus can also benefit from maximum flow algorithms. We show an example for $K = 2$ in Figure 1**a**. First we *replicate* the vertices of each network G_i so that all sets of vertices are disjoint, that is, $G'_i = (V'_i, E'_i)$ such that $V'_i \cap V'_j = \emptyset$ for every $i, j \in \{1, \dots, K\}$ with $i \neq j$. All edges are copied on the replicated set V'_i and assume that vertices are indexed from 1 to n in each network G_i, where vertices have the same index if they are identical in the original set V. The m-th vertex of a network G_i is denoted by v_i^m. We then construct a *unified network* $U(\mathcal{G}) = (\tilde{V}, \tilde{E})$ from the set of K networks $\mathcal{G} = \{G_1, \dots, G_K\}$ by connecting each pair of replicated vertices in the following manner:

$$\tilde{V} := \textstyle\bigcup_{i=1}^{K} V'_i,\;\; \tilde{E} := \bigcup_{i=1}^{K} E'_i \cup \bigcup_{m=1}^{n} A_m,\;\text{ where}$$
$$A_m := \big\{\, \{v_i^m, v_j^m\} \;\big|\; i, j \in \{1, \dots, K\}, i \neq j \,\big\}\,.$$

The weight $\tilde{w}$ of edges is given as $\tilde{w}(e) = w_i(e)$ if $e \in E'_i$ and $\tilde{w}(e) = \mu/\lambda$ otherwise. Thus $U(\mathcal{G})$ has $|\tilde{V}| = Kn$ vertices and $|\tilde{E}| = \sum_{i=1}^{K} |E_i| + nK(K-1)/2$ edges.

THEOREM 2.1. *Given a set of K networks $\mathcal{G} = \{G_1, \dots, G_K\}$. For every subset $S \subset \tilde{V}$ in the unified network $U(\mathcal{G})$, we have*

$$f(S) = \sum_{i=1}^{K} f_i(S_i),\quad g(S) = \sum_{i=1}^{K} g_i(S_i) + \sum_{i<j} h(S_i, S_j),$$

where f, g are defined over $U(\mathcal{G})$ as in Eq. (2.1)*,* (2.2)*.*

Proof. Suppose that $K = 2$ for simplicity. We can easily generalize the following proof for the case of $K > 2$ by considering sums over pairs.

The first equation $f(S) = f_1(S_1) + f_2(S_2)$ directly follows from the definition of S_1 and S_2. Thus we focus on the second equation $g(S) = g_1(S_1) + g_2(S_2) + h(S_1, S_2)$. Since $V'_1 \cap V'_2 = \emptyset$ and therefore $|S| = |S_1| + |S_2|$, all we have to prove is $\lambda \sum_{e \in B} w(e) = \lambda \sum_{e \in B_1} w_1(e) + \lambda \sum_{e \in B_2} w_2(e) + \mu |S_1 \bigtriangleup S_2|$. From the definition of the unified network, we have $B = B'_1 \cup B'_2 \cup B_A$, where $B'_i = \{\{v, u\} \in E'_i \mid v \in V'_i \setminus S,\; u \in S\}$, which is a set of edges on the unified network corresponding to B_i, and B_A is given as $B_A = \{\, \{v_i^m, v_j^m\} \mid v_i^m \in S \text{ and } v_j^m \notin S,\; i \neq j \,\} \subset \bigcup_{m=1}^{n} A_m$. We show an example of B in Figure 1**b**. We see that the cardinality of B_A is the same as that of the set $S_1 \bigtriangleup S_2$. Thus we have $\lambda \sum_{e \in B} w(e) = \lambda(\,\sum_{e \in B'_1} w_1(e) + \sum_{e \in B'_2} w_2(e) + (\mu/\lambda)|B_A|\,)$. □

The multiple task problem in (2.3) is therefore exactly equivalent to the single task feature selection problem (2.2) over the unified network $U(\mathcal{G})$, and can therefore be solved directly by the maximum flow algorithm.

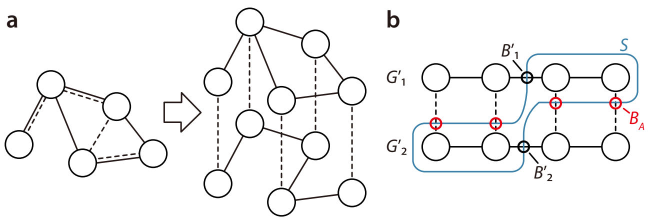

Figure 1: (**a**) Example of two networks (left) which share vertices and have different edges (solid and dotted lines), and the unified network (right), where vertices are duplicated and new edges (dotted lines) are added. (**b**) Example of two duplicated networks G'_1 and G'_2.

Note that this formulation applies even if some features are missing for some task. In that case V'_i contains only vertices corresponding to the features available for task i and A contains only edges $\{v_i^m, v_j^m\}$, where the feature m is available for both tasks i and j.

2.3 Spectral Analysis. We describe our formulation from a graph spectral point of view. We denote by $\boldsymbol{f} \in \{0,1\}^{|V|}$ the indicator vector of a subset $S \subset V$: $\boldsymbol{f}_v$ is set to 1 if $v \in S$ and 0 otherwise, and denote by $\boldsymbol{c} \in \mathbb{R}^{|V|}$ the vector composed of values $q(v)$.

The function f in Equation (2.1) can be rewritten as $f(S) = \boldsymbol{c}^{\mathrm{T}}\boldsymbol{f}$. Let $\boldsymbol{L}$ be the *Laplacian matrix* of a network G. Then $g(S) = \lambda \boldsymbol{f}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{f} + \eta\|\boldsymbol{f}\|_0$ holds for the penalty term in Equation (2.2) since we have $\boldsymbol{f}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{f} = \sum_{\{v,u\}\in E}(\boldsymbol{f}_v - \boldsymbol{f}_u)^2$ if all edge weights are 1, where $(\boldsymbol{f}_v - \boldsymbol{f}_u)^2$ is the XOR of $\boldsymbol{f}_v$ and $\boldsymbol{f}_u$ and coincides with the cardinality of B in Equation (2.2). The same relationship holds if the edges are weighted. The Laplacian graph regularizer $\boldsymbol{f}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{f}$ is often used in the literature for penalizing disconnected features (e.g. [18]). Thus we can state the formulation of single-task feature selection as: $\operatorname{argmax}_{\boldsymbol{f}\in\{0,1\}^{|V|}} \boldsymbol{c}^{\mathrm{T}}\boldsymbol{f} - \lambda\boldsymbol{f}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{f} - \eta\|\boldsymbol{f}\|_0$.

In the multi-task framework, the formulation is naturally extended to

$$\underset{\boldsymbol{f}_1,\dots,\boldsymbol{f}_K}{\operatorname{argmax}} \sum_{i=1}^{K}\big(\boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{f}_i - \lambda\boldsymbol{f}_i^{\mathrm{T}}\boldsymbol{L}_i\boldsymbol{f}_i - \eta\|\boldsymbol{f}_i\|_0\big) - \sum_{i<j}\mu\|\boldsymbol{f}_i - \boldsymbol{f}_j\|_2^2,$$

where $\|\boldsymbol{f} - \boldsymbol{f}'\|_2^2 = \sum_{v\in V}(\boldsymbol{f}_v - \boldsymbol{f}'_v)^2$. Suppose that $\tilde{\boldsymbol{f}}$ and $\tilde{\boldsymbol{c}}$ are concatenations of the vectors $\boldsymbol{f}_1\dots,\boldsymbol{f}_K$ and $\boldsymbol{c}_1\dots,\boldsymbol{c}_K$, respectively. We directly have $\sum_{i=1}^{K}(\boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{f}_i - \eta\|\boldsymbol{f}_i\|_0) = \tilde{\boldsymbol{c}}^{\mathrm{T}}\tilde{\boldsymbol{f}} - \eta\|\tilde{\boldsymbol{f}}\|_0$. Moreover, we can prove that $\sum_{i=1}^{K}\lambda\boldsymbol{f}_i^{\mathrm{T}}\boldsymbol{L}_i\boldsymbol{f}_i + \sum_{i<j}\mu\|\boldsymbol{f}_i - \boldsymbol{f}_j\|_2^2 = \lambda\tilde{\boldsymbol{f}}^{\mathrm{T}}\tilde{\boldsymbol{L}}\tilde{\boldsymbol{f}}$, where $\tilde{\boldsymbol{L}}$ is the Laplacian matrix of the unified network $U(\mathcal{G})$. Thus this becomes single-task feature selection:

$$\operatorname{argmax}_{\tilde{\boldsymbol{f}}\in\{0,1\}^{K|V|}} \tilde{\boldsymbol{c}}^{\mathrm{T}}\tilde{\boldsymbol{f}} - \lambda\tilde{\boldsymbol{f}}^{\mathrm{T}}\tilde{\boldsymbol{L}}\tilde{\boldsymbol{f}} - \eta\|\tilde{\boldsymbol{f}}\|_0.$$

3 Related Work

Current work on feature selection with network information typically concentrates on regularized linear regression (an excellent overview is given by [30]).

In a single-task context, the `Lasso` regression model [28] minimizes the prediction error together with the ℓ_1-norm of the regression parameter vector, which encourages sparse solutions. This fact popularized the use of Lasso for feature selection. Note however that the ℓ_1-norm is a convex relaxation of the cardinality constraint one would really want to enforce.

`Group Lasso` [31] partitions features into groups and encourages to select entire such groups via an ℓ_1/ℓ_2 penalty. If a graph over the features is available, groups can be defined as pairs of connected features [15]. However, the number of groups becomes prohibitively massive on large-scale networks, which grows exponentially once one starts to consider connected subsets of higher cardinality. A popular instance of Group Lasso is `Elastic Net` [34], in which all features belong to a single group; Elastic Net is particularly suited when the number of features is larger than that of samples and when several correlated features should be selected.

`Grace` (Graph-constrained estimation) [18, 19] adds a Laplacian graph regularizer analogous to that of SConES to the objective. While our method is an association-based approach, Grace aims at minimizing a prediction error in a Lasso framework. In practice, it is at least one order of magnitude slower than SConES on a network of the same size [2]. `aGrace` (adaptive Grace) [19] and `GOSCAR` [30] employ refined types of network regularizers which allow connected features to have effects of opposite directions.

Several multi-task versions of Lasso, in which related tasks are coupled with each other, have been proposed as well: `Multi-Task Lasso` [23] uses an ℓ_2-norm on each weight across all tasks to reward solutions, where the same features are selected for all tasks. `Graph-Guided Fused Lasso` [16] extends this idea by coupling the weight vectors of correlated tasks: the more correlated two tasks are, the more solutions in which they have similar weight vectors are rewarded. Recent developments combine multi-task learning with structured regularization on the input features (e.g. [6, 17]). Finally, [7] incorporates network-structured feature selection to multi-task learning. This Lasso approach has the advantage to integrate task covariance, which we leave to future work. However, it uses a single network over the features and its extension to multiple task-dependent networks is not straightforward.

Note that "network feature selection" [12] can also refer to the quite different class of problems where the *objects*, not their features, are connected over a graph.

4 Experiments

We evaluate the proposed method, which we refer to as Multi-SConES, on both synthetic and real data. Throughout all experiments, $q(v)$ is set to the absolute value of Pearson's correlation coefficient between a feature v and the response $\boldsymbol{y}$. This setting is slightly different from that in [2], where linear SKAT was used.

Environment. We used Mac OS X version 10.7.4 with 2 × 3 GHz Quad-Core Intel Xeon CPU and 16 GB of memory. SConES and Multi-SConES were implemented in R, version 2.15.1. All experiments were performed in the R environment.

Comparison partners. We evaluate our method in both single-task and multi-task contexts. We systematically use as a baseline the top-k features ranked by q alone, where k is set to be the same number of features as selected by SConES or its multi-task version. This allows us to evaluate the impact of the structured regularizer. We refer to this method as Correlation Ranking.

In single-task feature selection, we also compare SConES with two standard feature selection algorithms, Lasso [28] and Elastic Net [34], as well as three state-of-the-art structured regularizer methods, group Lasso [27] (with groups formed by edges as suggested by [15]), Grace, and aGrace [18, 19]. Grace and aGrace, which use a Laplacian graph regularizer, can be considered Lasso equivalents of SConES. In Lasso and Elastic Net, the network information was just ignored.

In multi-task settings, we compare Multi-SConES to multi-task Lasso [23] and multi-task Grace. Since Grace is for single-task feature selection, we construct an artificial dataset including a given network using the reformulation in Lemma 1 of [18], followed by applying multi-task Lasso to the dataset (Supplementary Note C). As we cannot determine sign changes in the objective function of aGrace [19, Section 2.2] over multiple tasks, we do not use a multi-task version of aGrace. Since these methods cannot treat different networks for different tasks, we are limited, for our experiments in this paper, to cases where all tasks share the same network.

In addition, as group Lasso clearly underperforms Grace in our single-task experiments, we keep Grace as the sole structured regularized Lasso comparison partner in the multi-task setting.

We used the `glmnet` package in R for Lasso, Elastic Net, and multi-task Lasso, and the `SGL` package in R for group Lasso. We implemented Grace and aGrace in R, based on the reformulation in Lemma 1 of [18]. As this method requires to compute the single value decomposition (SVD) of the network's Laplacian, it does not scale to large networks. We therefore adopt a new reformulation, replacing the matrix obtained by SVD with the *incidence matrix* of the network. It can easily be shown that this gives exactly the same solution as Grace. As the incidence matrix can be constructed in linear time in the number of vertices and edges, this is a much faster implementation.

Parameter selection. In feature selection, there is generally no ground truth to validate selected features in training. Thus one must use a proxy to evaluate the relative quality of the solutions given by different parameter values. One such proxy, used in [2], is the stability of the selection. Another possibility, which we used here, is to consider predictivity, which is still an indicator of the quality of the selected features although we do not wish to optimize it directly.

For every method, we performed 10-fold cross-validation and selected optimal parameters that yield the lowest mean squared error (MSE).

Evaluation criteria. Our main goal is to recover truly causal features, or, in other words, to accurately classify the features into causal and non-causal. As this binary classification problem is imbalanced, we evaluate performance using Matthews correlation coefficient (MCC [22]). MCC ranges from -1 to 1, 1 being best.

In experiments on synthetic data, we also evaluate the predictivity of the selected features and report MSE (ranging from 0 to 1, 0 being best) on a test set using ridge regression on the selected features.

4.1 Evaluation on Synthetic Data. We first evaluate (Multi-)SConES on synthetic data. We simulate four types of gene regulatory networks, models 1 to 4, which are exactly the same as described in [18] (Supplementary Note D) and use their combination in the multi-task setting. In each network there are $2,200$ features and the first 44 features are causal to the response. Models 1 and 3 (resp. 2 and 4) are positively (resp. negatively) correlated networks. Model 3 (resp. model 4) is identical to model 1 (resp. model 2), but the connection in models 3 and 4 is weaker than in models 1 and 2.

For each model, we generate training and test datasets of 100 samples each, and report the average MCC as well as the test MSE over 50 repetitions.

Efficiency. We first analyze the runtime of Multi-SConES with respect to the number of tasks. We create multi-task problems with varying number of tasks from 1 to 100 by repeatedly combining model 1 with itself, and report the runtimes of multi-task Lasso, Multi-Grace and Multi-SConES in Figure 2**a**.

Empirically, the runtime of Multi-SConES increases cubically with the number of tasks. While this is sub-optimal, in particular compared with Multi-Grace, we must remember that Multi-Grace cannot use different networks for different tasks. Moreover, Multi-SConES is still efficient enough to make it possible to analyze

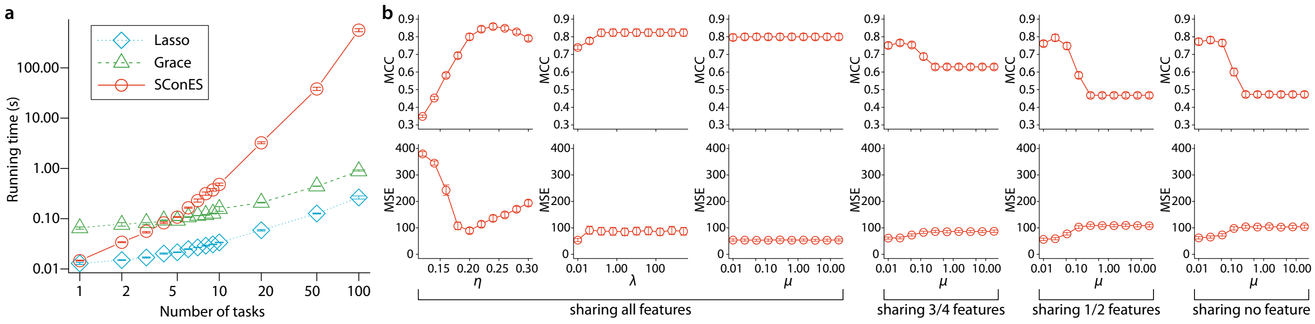

Figure 2: (**a**) Running time with respect to changes in number of tasks under fixed regularization parameters. Data are mean ± SEM. (**b**) Feature selection performance with respect to changes in regularization parameters η, λ, and μ. Note that x-axes for λ and μ have logarithmic scales. The effects of changes in μ are reported for various feature-sharing scenarios. Two parameters η and λ behave identically independently of the amount of true causal features shared by the tasks and corresponding plots are therefore not reported. Data are means ± SEM.

hundreds of thousands of features for the order of a dozen of tasks, which matches the statistical genetics setting that motivates our study.

In the particular case where we want to select the same features for all tasks and have a single network structure over the features, Multi-SConES reduces to a single-task problem over a network of same size: $q(v)$ is simply replaced with $\sum_{i=1}^{K} q_i(v)$ and μ set to zero. In that case Multi-SConES is as efficient as the single-task SConES, which is much more efficient than Grace over the same network (as corroborated by [2]).

Parameter sensitivity. Next, we analyze the behavior of Multi-SConES with respect to changes in the regularization parameters λ, η, and μ. For that purpose, we fix two of those parameters, and run Multi-SConES for two-task feature selection over models 1 and 2. To understand parameter sensitivity with respect to the amount of causal features shared across tasks, we perform experiments for four cases: models 1 and 2 share all, 3/4, half, or none of their features. We use $\lambda = 1$, $\mu = 1$, and $\eta = 0.2$ when they are fixed.

Results are shown in Figure 2**b**. Multi-SConES is sensitive to η, while more robust to μ and robust to λ if it is set large enough. The robustness of Multi-SConES with respect to λ can be understood as follows: once λ is large enough to cause the true causal features to be selected, if they form a subnetwork disconnected from the rest of the network, the corresponding penalty term becomes 0, and increasing λ will not affect the objective.

Similarly, if the true causal features are identical across all tasks, the penalty term controlled by μ is also 0, and varying μ will not affect the objective. However, if the causal features are not shared across all tasks, setting μ too large enforces the selection of too many identical features and leads to poor solutions. The behaviors of λ and η, however, remain unchanged across these different scenarios and is therefore not reported.

In contrast, MCC (resp. MSE) shows a concave (resp. convex) response to η, which fits to our theoretical analysis (Section 2.1, Regularization path), although here we evaluate solutions in terms of generalization error on independently generated test datasets.

In practice, this means that in cases well-behaved enough, we do not need to carefully tune the regularization parameters λ and μ. As the entire regularization path with respect to η can be obtained without increasing the time complexity, finding optimal parameters for Multi-SConES becomes attractively inexpensive.

Feature selection performance. We then evaluate the feature selection performance of Multi-SConES in both single-task and multi-task settings (Figure 3).

In the single-task setting, SConES shows much better performance than the baseline Correlation Ranking. This means that the penalty function g in Equation (2.2) works well to select connected features. Moreover, SConES outperforms all the other methods in terms of MCC, showing that it is better at recovering true causal features. Only Lasso (and, in one case, Elastic Net) outperforms SConES in terms of predictivity of the selected features. However the features it selects are too sparse and disconnected, resulting in notably worse MCC scores and difficulties in interpretability. These results are consistent with the behavior reported in [2].

To evaluate Multi-SConES, we create multi-task problems by combining the models. More precisely, for models 1 and 2, two-task problems are created by combining both models, and three-task problems by combining them with model 3; for models 3 and 4, two-task problems are creating by combining both models, and three-task problems by combining them with model 1. The four-task problem combines all four models.

Multi-SConES outperforms SConES in all cases.

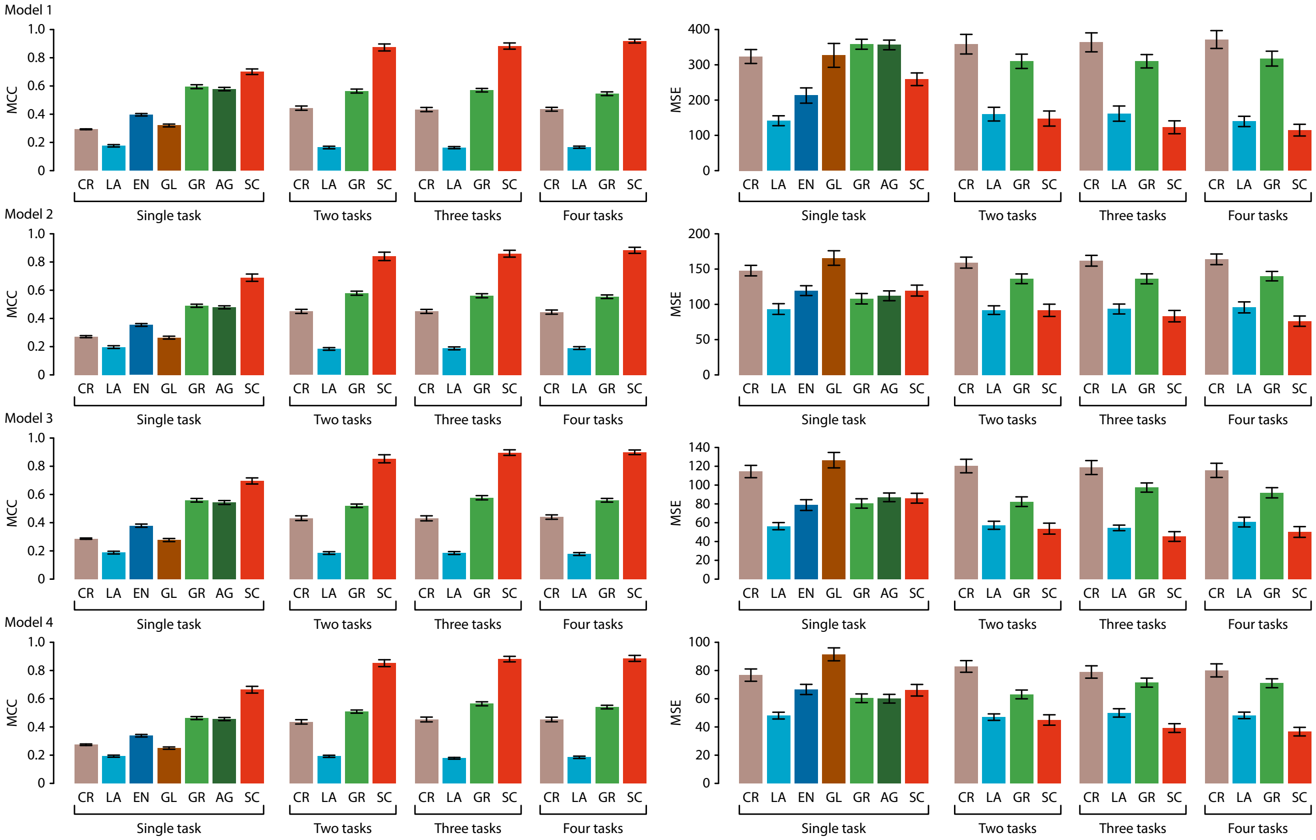

Figure 3: Feature selection performance for synthetic data. MCC (left column) should be maximized and MSE (right column) should be minimized. CR: the ranking of correlations (baseline), LA: Lasso, EN: the elastic net, GL: group Lasso, GR: Grace, AG: aGrace, and SC: SConES. Data are means ± SEM.

Moreover, performance (MCC and MSE) improves with the number of tasks. This confirms that our multi-task formulation on feature networks is effective compared to solving each task independently. Furthermore, Multi-SConES achieves significantly better MCC than all of its comparison partners and is also now superior in terms of predictivity. Our method is therefore effective for multi-task feature selection on networks.

We also examine Multi-SConES when causal features are not exactly shared. For each data model, we perform two-task feature selection (models $1+2$ and $3+4$) assuming that a fraction σ ($\sigma = 3/4$, $1/2$ or 0) of the causal features are shared between both tasks. Results are shown in Figure 4. Once again, Multi-SConES clearly outperforms all other methods in terms of MCC. The features it selects are also more predictive, except for those selected by multi-task Lasso when half of the features are shared between tasks. The more features are shared, the better Multi-SConES is at recovering causal, explanatory features. This holds for all multi-task methods and is typical in multi-task learning.

4.2 Multi-Locus Association Mapping.

As a real world application, we performed large-scale multi-locus mapping of *Arabidopsis thaliana* flowering time phenotypes. Our goal here is to uncover which SNPs are associated with flowering time, using a network over SNPs derived from biological properties.

Data preparation. We used the *Arabidopsis thaliana* GWAS data[1] collected by [1]. This dataset contains 216,130 SNPs (features) for 199 individuals, with missing values, and there are 23 flowering phenotypes in total. It is well known that *A. thaliana* is susceptible to population structure confounding, that is, the existence of subpopulations (clusters) of individuals due to, for instance, different demographic histories or diverse environmental influences, induces fake correlations between genotypes and phenotypes. We corrected it using principal component analysis as in [2, 24].

To evaluate the quality of the selected SNPs (features), we used the 282 candidate (causal) genes for flow-

[1] https://cynin.gmi.oeaw.ac.at/home/resources/atpolydb/genomic-polymorphism-data-in-arabidopsis-thaliana

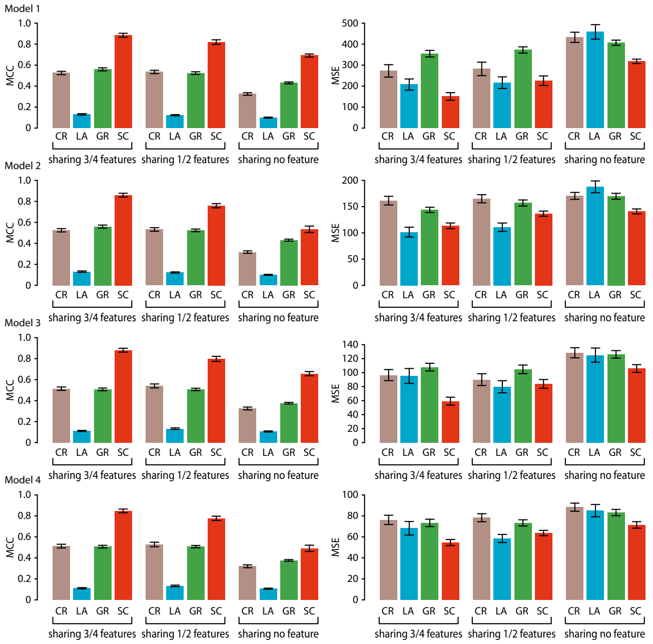

Figure 4: Feature selection performance in two tasks for synthetic data. A fraction σ ($= 3/4$, $1/2$ or 0) of the causal features are shared. Data are means ± SEM.

ering time listed in [5, (Table 12 in Dataset S1)], originally provided in [1], as an approximation of the gold standard. For each selected SNP, we checked whether or not it is located within 20 kb of one of the 282 candidate genes as in [2, 5]. If a SNP belongs to more than two genes, we assigned it to the closest gene.

We derived a network over SNPs from the *A. thaliana* protein-protein interaction network between genes from TAIR[2] (The Arabidopsis Information Resource). SNPs were connected with a weight of 1 if they belong to the same gene or connected genes. In addition, we connected each pair of SNPs adjacent on the genomic sequence with a small weight of 0.01.

Results. For each of two phenotypes (2W and LDV), which have high correlations to other phenotypes in average, we picked two additional phenotypes they are highly correlated with (4W and FTGH for 2W, 0W and FT10 for LDV) and checked whether or not these additional phenotypes improve performance of Multi-SConES and its competitors.

For each method, we determined the optimal parameters by 10-fold cross-validation and run it on the full data to get a final set of selected SNPs. We report MCC as well as the ratios of candidate SNPs (resp. genes) retrieved with respect to the number of selected SNPs (resp. genes) in Table 1.

Once again, Multi-SConES shows much better performance in terms of MCC than its competitors. In addition, the proportion of SNPs near candidate genes among the selected SNPs is higher for Multi-SConES than those for the other methods. Finally, combining several phenotypes helps recovering more candidate causal genes. Altogether, this means that our multi-task feature selection strategy can also be effectively employed for the important real-world problem of multi-locus association mapping in *A. thaliana*.

5 Conclusion

In this paper we have proposed Multi-SConES, a new formulation of multi-task feature selection with multiple network regularizers. We directly optimize feature relevance scores and exactly solve the formulation by maximum flow algorithms. Compared to the typical structured Lasso approaches, Multi-SConES shows improved ability to discover causal features in simulated and real-world experiments.

Unlike existing structured sparsity multi-task feature selection methods, Multi-SConES can use different networks for different tasks, and yields a clear, binary classification of features. Another attractive property of our approach is the possibility to incorporate cardinality constraints on the size of the solution set.

Currently, we model the relationship between tasks with a single parameter μ, which controls how coupled the solutions are. Some multi-task Lasso models can include more detailed structures of correlation between tasks (e.g. [16]). Others also consider relationships between tasks using a task covariance matrix [7]. In future work, we will study how to integrate these types of more complex task relationships into Multi-SConES.

Another interesting direction is to incorporate proximal methods with structured norms, which is also shown to be related to the maximum flow problem [20].

References

[1] Atwell, S., *et al.*: Genome-wide association study of 107 phenotypes in *Arabidopsis thaliana* inbred lines. Nature 465(7298), 627–631 (2010)

[2] Azencott, C.A., Grimm, D., Sugiyama, M., Kawahara, Y., Borgwardt, K.M.: Efficient network-guided multi-locus association mapping with graph cuts. Bioinformatics 29(13), i171–i179 (2013)

[3] Bach, F.: Structured sparsity-inducing norms through submodular functions. In: NIPS. pp. 118–126 (2010)

[4] Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. TPAMI 26(9), 1124–1137 (2004)

[5] Brachi, B., *et al.*: Linkage and association mapping of *Arabidopsis thaliana* flowering time in nature. PLoS Genetics 6(5), e1000940 (2010)

[2] http://www.arabidopsis.org/

Table 1: Results of multi-locus mapping for *Arabidopsis thaliana* genotypes and flowering time phenotypes.

Phenotype	MCC			Hit ratio of SNPs			Hit ratio of genes		
	Lasso	Grace	SConES	Lasso	Grace	SConES	Lasso	Grace	SConES
2W	0.001	−0.001	**0.014**	7/126	4/98	**42/338**	2/112	1/91	**7/124**
2W + 4W	−0.001	−0.003	**0.016**	7/175	6/198	**81/802**	2/163	2/191	**11/240**
2W + FT GH	0.001	0.000	**0.024**	9/173	7/146	**106/818**	9/162	7/135	**13/250**
2W + 4W + FT GH	0.005	0.002	**0.027**	15/183	16/265	**101/679**	6/174	3/256	**13/208**
LDV	0.001	0.000	**0.016**	6/116	7/144	**73/667**	2/107	2/131	**9/202**
LDV + 0W	0.005	0.007	**0.020**	16/196	19/206	**86/702**	2/183	2/187	**10/209**
LDV + FT10	0.001	0.001	**0.021**	12/214	10/191	**92/762**	1/199	1/181	**10/221**
LDV + 0W + FT10	0.003	0.002	**0.023**	18/283	19/323	**81/482**	2/265	1/307	**10/153**

[6] Chen, X., *et al.*: A two-graph guided multi-task lasso approach for eQTL mapping. In: AISTATS. pp. 208–217 (2012)

[7] Fei, H., Huan, J.: Structured feature selection and task relationship inference for multi-task learning. KAIS 35(2), 345–364 (2013)

[8] Gallo, G., Grigoriadis, M.D., Tarjan, R.E.: A fast parametric maximum flow algorithm and applications. SICOMP 18(1), 30–55 (1989)

[9] Gao, S., Tsang, I., Chia, L.: Laplacian sparse coding, hypergraph Laplacian sparse coding, and applications. TPAMI 35(1), 92–104 (2013)

[10] Goldberg, A.V., Tarjan, R.E.: A new approach to the maximum-flow problem. JACM 35(4), 921–940 (1988)

[11] Gretton, A., Bousquet, O., Smola, A., Schölkopf, B.: Measuring statistical dependence with Hilbert-Schmidt norms. In: ALT. pp. 63–77. Springer (2005)

[12] Gu, Q., Han, J.: Towards feature selection in network. In: CIKM. pp. 1175–1184 (2011)

[13] Hastie, T., Rosset, S., Tibshirani, R., Zhu, J.: The entire regularization path for the support vector machine. JMLR 5, 1391–1415 (2005)

[14] Huang, J., Zhang, T., Metaxas, D.: Learning with structured sparsity. JMLR 12, 3371–3412 (2011)

[15] Jacob, L., Obozinski, G., Vert, J.P.: Group lasso with overlap and graph lasso. In: ICML. pp. 433–440 (2009)

[16] Kim, S., Sohn, K.A., Xing, E.: A multivariate regression approach to association analysis of a quantitative trait network. Bioinformatics 25(12), i204–i212 (2009)

[17] Lee, S., Xing, E.P.: Leveraging input and output structures for joint mapping of epistatic and marginal eQTLs. Bioinformatics 28(12), i137–i146 (2012)

[18] Li, C., Li, H.: Network-constrained regularization and variable selection for analysis of genomic data. Bioinformatics 24(9), 1175–1182 (2008)

[19] Li, C., Li, H.: Variable selection and regression analysis for graph-structured covariates with an application to genomics. Ann Appl Stat 4(3), 1498–1516 (2010)

[20] Mairal, J., Jenatton, R., Obozinski, G., Bach, F.: Convex and network flow optimization for structured sparsity. JMLR 12, 2681–2720 (2011)

[21] Marchini, J., Donnelly, P., Cardon, L.R.: Genome-wide strategies for detecting multiple loci that influence complex diseases. Nat Genet 37(4), 413–417 (2005)

[22] Matthews, B.W.: Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochimica et Biophysica Acta 405(2), 442–451 (1975)

[23] Obozinski, G., Wainwright, M.J., Jordan, M.I.: High-dimensional union support recovery in multivariate regression. In: NIPS. pp. 1217–1224 (2008)

[24] Price, A.L., Patterson, N.J., Plenge, R.M., Weinblatt, M.E., Shadick, N.A., Reich, D.: Principal components analysis corrects for stratification in genome-wide association studies. Nat Genet 38(8), 904–909 (2006)

[25] Reshef, D.N., *et al.*: Detecting novel associations in large data sets. Science 334(6062), 1518–1524 (2011)

[26] Saha, B., Pham, D.S., Phung, D., Venkatesh, S.: Sparse subspace clustering via group sparse coding. In: SDM. pp. 130–138 (2013)

[27] Simon, N., Friedman, J., Hastie, T., Tibshirani, R.: A sparse-group lasso. J Comput Graph Stat 22(2), 231–245 (2013)

[28] Tibshirani, R.: Regression shrinkage and selection via the lasso. J Roy Stat Soc B 58(1), 267–288 (1996)

[29] Wu, M.C., Lee, S., Cai, T., Li, Y., Boehnke, M., Lin, X.: Rare-variant association testing for sequencing data with the sequence kernel association test. Am J Hum Genet 89(1), 82–93 (2011)

[30] Yang, S., Yuan, L., Lai, Y.C., Shen, X., Wonka, P., Ye, J.: Feature grouping and selection over an undirected graph. In: KDD. pp. 922–930 (2012)

[31] Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. J Roy Stat Soc B 68, 49–67 (2006)

[32] Zhang, Y., Yeung, D.Y., Xu, Q.: Probabilistic multi-task feature selection. In: NIPS. pp. 2559–2567 (2010)

[33] Zhou, Y., Jin, R., Hoi, S.C.H.: Exclusive lasso for multi-task feature selection. In: AISTATS. vol. 9, pp. 988–995 (2010)

[34] Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J Roy Stat Soc B 67(2), 301–320 (2005)