Carnegie Mellon University
Machine Learning Department
School of Computer Science

Thesis Proposal

Uncovering Structure in High-Dimensions:
Networks and Multi-task Learning Problems

Mladen Kolar

May 2, 2012

Thesis Committee Members:
Eric P. Xing, Chair
Larry Wasserman
Aarti Singh
Francis Bach (INRIA)

Contents

Abstract

Extracting knowledge and providing insights into complex mechanisms underlying noisy high-dimensional data sets is of utmost importance in many scientific domains. Statistical modeling has become ubiquitous in the analysis of high-dimensional functional data in search of better understanding of cognition mechanisms, in the exploration of large-scale gene regulatory networks in hope of developing drugs for lethal diseases, and in prediction of volatility in stock market in hope of beating the market. Statistical analysis in these high-dimensional data sets is possible only if an estimation procedure exploits hidden structures underlying data.

This thesis develops flexible estimation procedures with provable theoretical guarantees for uncovering unknown hidden structures underlying data generating process. Of particular interest are procedures that can be used on high-dimensional data sets where the number of samples n much smaller than the ambient dimension p. Learning in high-dimensions is difficult due to the curse of dimensionality, however, the special problem structure makes inference possible. Due to its importance for scientific discovery, we put emphasis on consistent structure recovery throughout the thesis. Particular focus is given to two important problems, semi-parametric estimation of networks and feature selection in multi-task learning.

1 Introduction

In recent years, we have witnessed fast advancement of data-acquisition techniques in many areas, including biological domains, engineering and social sciences. As a result, new statistical and machine learning techniques are needed to help us develop a better understanding of complexities underlying large, noisy data sets.

Statistical inference in high-dimensions is challenging due to the curse of dimensionality. What makes the inference possible is that many real world systems have a special structure that can be represented with a much smaller number of parameters than the dimension of the ambient space. Even when a system cannot be represented exactly with few parameters, there are still good approximations that use few parameters and useful in providing insights into the system. This concept of parsimony commonly occurs in a number of scientific disciplines.

The main goal of this thesis is to develop flexible and principled statistical methods for uncovering hidden structure underlying high-dimensional, complex data sets with focus on scientific discovery. This thesis is naturally divided into two parts. In the first part, we focus on learning structure of time varying latent networks from nodal observations. The second part of the thesis focus on exploiting structure in multi-task learning.

1.1 Network Structure Estimation

Across the sciences, networks provide a fundamental setting for representing and interpreting information on the state of an entity, the structure and organization of communities, and changes in these over time. Traditional approaches to network analysis tend to make simplistic assumptions, such as assuming that there is only a single node or edge type, or ignoring the dynamics of the networks. Unfortunately, these classical approaches are not suitable for network data arising in contemporary applications. Modern network data can be large, dynamic, heterogeneous, noisy and incomplete. These characteristics add a degree of complexity to the interpretation and analysis of networks.

As a motivating example, let us consider estimation of cellular networks in systems biology. Studying biological networks is a difficult task, because in complex organisms, biological processes are often controlled by a large number of molecules that interact and exchange information in a spatial-temporally specific and context-dependent manner. Current approaches to studying biological networks have primarily focused on creating a descriptive analysis of macroscopic properties, which include degree distribution, path length and motif profiles of the networks, or using graph mining tools to identify clusters and subgraphs. Such simple analysis offer limited insights into the remarkably complex functional and structural organization of a biological system, especially in a dynamic context. Furthermore, it is often common to completely ignore the dynamic context in which the data are collected. For example, in the analysis of microarray data collected over a time course it is common to infer a single static gene network. As a solution to this problem, we develop a flexible framework

for inferring dynamic networks.

In this thesis, we develop flexible statistical procedures with rigorous theoretical guarantees for inferring unobservable dynamic network structure from nodal observations that are governed by the latent network. In particular, we build on the formalism of probabilistic graphical models in which we cast the problem of network learning as the problem of learning a graph structure from observational data. We develop methods for learning both undirected and directed graphical models. These estimation methods are developed for both gradually changing networks and networks with abrupt changes. Furthermore, we go beyond analysis dynamic systems only. Methods that are developed can be also used to learn conditional covariance structures, where a network depends on some other observed random variables.

Analysis of network data is an important problem in a number of disciplines [see, e.g., Kolaczyk, 2009, for a textbook treatment of the topic]. However, these methods assume availability of network structure for performing a statistical analysis. In this thesis, we develop techniques that learn network structure from only nodal observations. Once a network structure is learned, any of the existing network analysis tools can be used to further investigate properties of the underlying system. Therefore, this theses makes significant progress in advancing the boundary of what problems can be tackled using well developed network analysis tools.

1.2 Multi-task Learning

It has been empirically observed, on various data sets ranging from cognitive neuroscience Liu et al. (2009) to genome-wide association mapping studies Kim et al. (2009), that considering related estimation tasks jointly, improves estimation performance. Because of this, joint estimation from related tasks or multi-task learning has received much attention in the machine learning and statistics community.

In this thesis, we focus on a particular form of multi-task learning, in which the problem is to estimate the coefficients of several multiple regressions

$$\mathbf{y}_j = \mathbf{X}_j \boldsymbol{\beta}_j + \boldsymbol{\epsilon}_j, \qquad j \in [k] \tag{1}$$

where $\mathbf{X}_j \in \mathbb{R}^{n \times p}$ is the design matrix, $\mathbf{y}_j \in \mathbb{R}^n$ is the vector of observations, $\boldsymbol{\epsilon}_j \in \mathbb{R}^n$ is the noise vector and $\boldsymbol{\beta}_j \in \mathbb{R}^p$ is the unknown vector of regression coefficients for the j-th task, with $[k] = \{1, \ldots, k\}$.

Under the model in (1), we focus on variable selection under the assumption that the same variables are relevant for different regression problems. We sharply characterize the performance of different penalization schemes on the problem of selecting the relevant variables. Casting the problem of variable selection in the context of the Normal means, we are able to sharply characterize the sparsity patterns under which the Lasso procedure performs better than the group Lasso. Similarly, our results characterize how the group Lasso can perform better when each non-zero row is dense.

Next, we focus on efficient algorithms for screening relevant variables under the multi-task regression model. In particular, we analyze forward regression and marginal regression, which are extremely efficient in ultra-high dimensions. Common tool for variable selection in multi-task regression problems is the penalized least squares procedure, where the penalty biases solution to have many zero coefficients. Though efficient algorithms for these objectives exist, they still do not scale to million of input variables. Therefore, screening procedures are extremely useful for initial reduction of the dimensionality.

Part I

2 Network structure learning

Network models have become popular as a way to abstract complex systems and gain insights into relational patterns among observed variables. For example, in a biological study, nodes of the network can represent genes in one organism and edges can represent associations or regulatory dependencies among genes. In a social domain, nodes of a network can represent actors and edges can represent interactions between actors. Recent popular techniques for modeling and exploring networks are based on the structure estimation in the probabilistic graphical models, specifically, Markov Random Fields (MRFs). These models represent conditional independence between variables, which are represented as nodes. Once the structure of the MRF is estimated, the network is drawn by connecting variables that are conditionally dependent. The hope is that this graphical representation is going to provide additional insight into the system under observation, for example, by showing how different parts of the system interact.

In this part, we will focus on two types of Markov Random Fields: the Ising model, which represents a typical discrete MRF, and the Gaussian graphical model, which is a typical continuous MRF. We focus on these two models because they can be fully specified just with the first two moments. Even though they are quite simple, they are rich enough to be applicable in a number of domains, as we will see later, and they also provide an opportunity to succinctly present theoretical results. A statistical challenge in the framework of Markov Random Fields is to estimate reliably the graph structure from an observed sample.

2.1 Preliminaries

Let $G = (V, E)$ represent a graph, of which V denotes the set of vertices, and E denotes the set of edges over vertices. Depending on the specific application of interest, a node $u \in V$ can represent a gene, a stock, or a social actor, and an edge $(u, v) \in E$ can represent a relationship (e.g., correlation, influence, friendship) between actors u and v. Let $\mathbf{X} = (X_1, \ldots, X_p)'$, where $p = |V|$, be a random vector of nodal states following a probability distribution indexed by $\boldsymbol{\theta} \in \Theta$. Under a MRF, the nodal states X_u's are assumed to be

either discrete or continuous and the edge set $E \subseteq V \times V$ encodes certain conditional independence assumptions among components of the random vector $\mathbf{X}$, for example, the random variable X_u is conditionally independent of the random variable X_v given the rest of the variables if $(u, v) \notin E$. We focus on two types of MRFs: the Ising model and the Gaussian graphical models. We specify their forms below.

The Ising model arises as a special case of discrete MRFs, where each node takes binary nodal states. That is, under the Ising model, we have $X_u \in \mathcal{X} \equiv \{-1, 1\}$, for all $u \in V$ and the joint probability of $\mathbf{X} = \mathbf{x}$ can be expressed by a simple exponential family model:

$$\mathbb{P}_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{1}{Z} \exp\{\sum_{u<v} \theta_{uv} x_u x_v\}$$

where Z denotes the partition function that is usually intractable to compute and the weight potentials are given by θ_{uv} for all $(u, v) \in E$. Under the Ising model, the model is completely defined by the vector of parameters $(\theta_{uv})_{(u,v)\in V\times V}$. Furthermore, the parameters specify the graph structure, that is, we have that $\theta_{uv} = 0$ for all $(u, v) \notin E$.

The Gaussian graphical models will be used as the simplest continuous MRF as the probability distribution under the GGM can be fully specified with the first two moments. Let $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ be a p-dimensional multivariate Gaussian random variable with mean zero and covariance $\boldsymbol{\Sigma}$. Let $\boldsymbol{\Omega} \triangleq \boldsymbol{\Sigma}^{-1}$ be the precision matrix. The (a, b)-element, ω_{ab}, of the precision matrix is proportional to the partial correlation between random variables X_a and X_b, the ath and bth component of $\mathbf{X}$. Therefore X_a is conditionally independent of X_b given the rest of variables if and only if $\omega_{ab} = 0$. Again, this conditional dependence will be represented with a graph $G = (V, E)$, where the set of nodes V corresponds to the components of the random vector $\mathbf{X}$ and the edge set $E \subseteq V \times V$ includes edges between nodes only if the corresponding components are conditionally dependent, that is, an edge $e_{ab} \in E$ only if $\omega_{ab} \neq 0$. For a monograph on the Gaussian graphical models see Lauritzen [1996].

2.2 Literature Survey

The problem of learning conditional independence structure between random variables have been addressed extensively in the literature. Dempster [1972] identified the elements of the precision matrix as the canonical parameters of the exponential family of normal distributions. He also introduced the problem of covariance selection, in which one estimates covariance matrix by selecting zeros in its inverse. The work was done in the classical setting where p is smaller than n, however, when p is larger that n, the problem of identifying zeros in the precision matrix does not easily reduce to a hypothesis testing problem. In the seminal work, Meinshausen and Bühlmann [2006] relate the problem to variable selection in linear regression. Leveraging the lasso [Tibshirani, 1996] they efficiently estimate the non-zero pattern of the precision matrix. Another popular technique for estimating sparse precision matrix is based on ℓ_1-norm penalized maximum likelihood [Yuan and Lin, 2007]. Due to its

importance, the problem of covariance selection has drawn a lot of follow up work in both the machine learning and statistical community, which has led to remarkable progress in both computational [Banerjee et al., 2008, Friedman et al., 2008b, Hsieh et al., 2011, Duchi et al., 2008] and statistical issues [Ravikumar et al., 2008, Rothman et al., 2008, Fan et al., 2009b, Peng et al., 2009, Cai et al., 2011, 2010]. Other ways of estimating the covariance matrix in high-dimensions include Bickel and Levina [2008b], Bickel and Levina [2008a], Karoui [2008], and Wu and Pourahmadi [2009].

In the case of discrete MRFs, direct maximization of the log-likelihood is not tractable. Ravikumar et al. [2009] use a surrogate function which decomposes across different nodes and as a result can be maximized efficiently. Other work on graph structure estimation in discrete MRFs include score based searches [Srebro, 2001, Chow and Liu, 1968], which are limited to restricted classes of graphs due to the combinatorial explosion of the search space of graphs [Chickering, 1996], minimizing the Kullback-Leibler divergence [Abbeel et al., 2006] and other pseudo-likelihood methods [Bresler et al., 2008, Csiszar and Talata, 2006].

Almost all of the work have been driven by the simplifying assumption of static network structure. In the next few sections, we discuss more flexible estimation schemes.

3 Time Varying Networks

Prior to our work, literature mainly focused on estimating a single static network underlying a complex system. However, in reality, many systems are inherently dynamic and can be better explained by a dynamic network whose structure evolves over time. We develop statistical methodology of dealing with the following real world problems:

- *Analysis of gene regulatory networks.* Suppose that we have a set of n microarray measurements of gene expression levels, obtained at different stages during the development of an organism or at different times during the cell cycle. Given this data, biologists would like to get insight into dynamic relationships between different genes and how these relations change at different stages of development. The problem is that at each time point there is only one or at most a few measurements of the gene expressions; and a naive approach to estimating the gene regulatory network, which uses only the data at the time point in question to infer the network, would fail. To obtain a good estimate of the regulatory network at any time point, we need to leverage the data collected at other time points and extract some information from them.

- *Analysis of stock market.* In a finance setting, we have values of different stocks at each time point. Suppose, for simplicity, that we only measure whether the value of a particular stock is going up or down. We would like to find the underlying transient relational patterns between different stocks from these measurements and get insight into how these patterns change over time. Again, we only have one measurement at

each time point and we need to leverage information from the data obtained at nearby time points.

- *Understanding social networks.* There are 100 Senators in the U.S. Senate and each can cast a vote on different bills. Suppose that we are given n voting records over some period of time. How can one infer the latent political liaisons and coalitions among different senators and the way these relationships change with respect to time and with respect to different issues raised in bills just from the voting records?

The aforementioned problems have commonality in estimating a sequence of time-specific latent relational structures between a fixed set of entities (i.e., variables), from a time series of observation data of entities states; and the relational structures between the entities are time evolving, rather than being invariant throughout the data collection period.

3.1 Estimation Framework

In the following few section, we will assume that we are given a sequence of observations $\mathcal{D}_n = \{\mathbf{x}^t \sim \mathbb{P}^{(t)} | t \in \mathcal{T}_n\}$ where $\mathcal{T}_n$ is an index set. The goal is to estimate the parameters or more specifically conditional independence assumptions encoded by the sequence of probability distributions $\{\mathbb{P}^{(t)}\}_t$. We will use the estimation procedures of the form

$$\arg\min_{\boldsymbol{\theta}} L(\mathcal{D}_n; \boldsymbol{\theta}) + \text{pen}_\lambda(\boldsymbol{\theta}) \tag{2}$$

where $L(\cdot; \theta)$ is the convex loss function, $\text{pen}_\lambda(\cdot)$ is the regularization term and λ is a tuning parameter. The first term in the objective is measuring the fit to data, while the second one measures the complexity of the model. The regularization term is used to encode some prior assumptions about the model, e.g., sparsity of the graph structure or the way the graph structure changes over time. The loss functions that is used will be problem specific. For example, in the case of the Gaussian graphical models, we will use the negative log-likelihood, while in the case of discrete MRFs a surrogate to the negative log-likelihood will be used. In the next few sections, we will specialize the objective (2) to different estimation procedures.

3.2 Related Work

The work of Zhou et al. [2008] is the most relevant to our work on time varying networks, in which the authors develop a nonparametric method for estimation of time-varying Gaussian graphical model, under the assumption that the observations $\mathbf{x}^t \sim \mathcal{N}(0, \mathbf{\Sigma}^t)$ are independent, but not identically distributed, realizations of a multivariate distribution whose covariance matrix changes smoothly over time. In Zhou et al. [2008], the authors address the issue of consistent, in the Frobenius norm, estimation of the covariance and concentration matrix, however, the problem of consistent estimation of the non-zero pattern in the concentration matrix, which corresponds to the graph structure estimation, is not addressed there. Note

that the consistency of the graph structure recovery does not immediately follow from the consistency of the concentration matrix.

4 Smoothly Evolving Discrete Markov Random Fields (completed)

In this section, we discuss estimation of discrete MRFs when observations are coming from a smoothly changing probability distribution.

4.1 Model

We are given a sequence of n nodal states $\mathcal{D}_n = \{\mathbf{x}^t \sim \mathbb{P}_{\boldsymbol{\theta}^t} | t \in \mathcal{T}_n\}$, with the time index defined as $\mathcal{T}_n = \{1/n, 2/n, \ldots, 1\}$. For simplicity of presentation, we will assume that the observations are equidistant in time and only one observation is available at each time point from distribution $\mathbb{P}_{\boldsymbol{\theta}^t}$ indexed by $\boldsymbol{\theta}^t$. Specifically, we assume that the p-dimensional random vector $\mathbf{X}^t$ takes values in $\{-1, 1\}^p$ and the probability distribution takes the following form:

$$\mathbb{P}_{\boldsymbol{\theta}^t}(x) = \frac{1}{Z(\boldsymbol{\theta}^t)} \exp\left(\sum_{(u,v)\in E^t} \theta^t_{uv} x_u x_v \right), \quad \forall t \in \mathcal{T}_n, \tag{3}$$

where $Z(\boldsymbol{\theta}^t)$ is the partition function, $\boldsymbol{\theta}^t \in \mathbb{R}^{\binom{p}{2}}$ is the parameter vector and $G^t = (V, E^t)$ is an undirected graph representing certain conditional independence assumptions among subsets of the p-dimensional random vector $\mathbf{X}^t$. For any given time point $\tau \in [0, 1]$, we are interested in estimating the graph G^τ associated with $\mathbb{P}_{\boldsymbol{\theta}^\tau}$, given the observations $\mathcal{D}_n$.

Since we are primarily interested in a situation where the total number of observation n is small compared to the dimension p, our estimation task is going to be feasible only under some regularity conditions. We impose two natural assumptions: the *sparsity* of the graphs $\{G^t\}_{t\in\mathcal{T}_n}$, and the *smoothness* of the parameters $\boldsymbol{\theta}^t$ as functions of time. Intuitively, the smoothness assumption is required so that a graph structure at the time point τ can be estimated from samples close in time to τ. On the other hand, the sparsity assumption is required to avoid the curse of dimensionality and to ensure that a the graph structure can be identified from a small sample.

The model given in Eq. (3) can be thought of as a nonparametric extension of conventional MRFs, in the similar way as the varying-coefficient models [Cleveland, Grosse, and Shyu, 1991, Hastie and Tibshirani, 1993] are thought of as an extension to the linear regression models. The difference between the model given in Eq. (3) and an MRF model is that our model allows for parameters to change, while in MRF the parameters are considered fixed. Allowing parameters to vary over time increases the expressiveness of the model, and make it more suitable for longitudinal network data.

4.2 Estimation Procedure

Given a time point $\tau \in [0,1]$ and a sequence of observations $\mathcal{D}_n = \{\mathbf{x}^t \sim \mathbb{P}_{\boldsymbol{\theta}^t} | t \in \mathcal{T}_n\}$ with $\mathbb{P}_{\boldsymbol{\theta}^t}$ defined Eq. (3), the goal is to estimate the graph structure of the Markov random field associated with the distribution $\mathbb{P}_{\boldsymbol{\theta}^\tau}$. The parameter vector $\boldsymbol{\theta}^\tau$ is a $\binom{p}{2}$-dimensional vector, indexed by distinct pairs of nodes, of which an element is non-zero if and only if the corresponding edge $(u,v) \in E^\tau$. The problem of recovering the graph structure G^τ is equivalent to estimating the non-zero pattern of the vector $\boldsymbol{\theta}^\tau$, i.e., locations of non-zero elements of $\boldsymbol{\theta}^\tau$. A stronger notion of structure estimation is that of *signed edge recovery* in which an edge $(u,v) \in E^\tau$ is recovered together with the sign of the parameter $\text{sign}(\theta^\tau_{uv})$. We will show that the estimation procedure can consistently recover signed edges.

The estimation procedure is based on the neighborhood selection technique, where the graph structure is estimated by combining the local estimates of neighborhoods of each node. For each vertex $u \in V$, define the set of neighboring edges $S^\tau(u) := \{(u,v) \ : \ (u,v) \in E^\tau\}$ and the set of *signed neighboring edges* $S^\tau_\pm(u) := \{(\text{sign}(\theta^\tau_{uv}), (u,v)) \ : \ (u,v) \in S^\tau(u)\}$. The set of signed neighboring edges $S^\tau_\pm(u)$ can be determined from the signs of elements of the $(p-1)$-dimensional subvector of parameters $\boldsymbol{\theta}^\tau_u := \{\theta^\tau_{uv} \ : \ v \in V \backslash u\}$ associated with vertex u. Under the model (3), the conditional distribution of X^τ_u given other variables $\mathbf{X}^\tau_{\backslash u} := \{X^\tau_v \ : \ v \in V \backslash u\}$ takes the form

$$\mathbb{P}_{\boldsymbol{\theta}^\tau_u}(x^\tau_u | \mathbf{X}^\tau_{\backslash u} = \mathbf{x}^\tau_{\backslash u}) = \frac{\exp(2x^\tau_u \langle \boldsymbol{\theta}^\tau_u, \mathbf{x}^\tau_{\backslash u} \rangle)}{\exp(2x^\tau_u \langle \boldsymbol{\theta}^\tau_u, \mathbf{x}^\tau_{\backslash u} \rangle) + 1}, \tag{4}$$

where $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}'\mathbf{b}$ denotes the dot product. Observe that the model given in (4) can be viewed as expressing X^τ_u as the response variable in the generalized varying-coefficient models with $\mathbf{X}^\tau_{\backslash u}$ playing the role of covariates. For simplicity, we will write $\mathbb{P}_{\boldsymbol{\theta}^\tau_u}(x^\tau_u | \mathbf{X}^\tau_{\backslash u} = \mathbf{x}^\tau_{\backslash u})$ as $\mathbb{P}_{\boldsymbol{\theta}^\tau_u}(x^\tau_u | \mathbf{x}^\tau_{\backslash u})$.

Under the model given in Eq. (4) the log-likelihood, for one data-point $t \in \mathcal{T}_n$, can be written in the following form:

$$\begin{aligned} \gamma(\boldsymbol{\theta}_u; \mathbf{x}^t) &= \log \mathbb{P}_{\boldsymbol{\theta}_u}(x^t_u | \mathbf{x}^t_{\backslash u}) \\ &= x^t_u \langle \boldsymbol{\theta}_u, \mathbf{x}^t_{\backslash u} \rangle - \log \left(\exp(\langle \boldsymbol{\theta}_u, \mathbf{x}^t_{\backslash u} \rangle) + \exp(-\langle \boldsymbol{\theta}_u, \mathbf{x}^t_{\backslash u} \rangle) \right). \end{aligned} \tag{5}$$

For an arbitrary point of interest $\tau \in [0,1]$, the estimator $\hat{\boldsymbol{\theta}}^\tau_u$ of the sign-pattern of the vector $\boldsymbol{\theta}^\tau_u$ is defined as the solution to the following convex program:

$$\hat{\boldsymbol{\theta}}^\tau_u = \min_{\boldsymbol{\theta}_u \in \mathbb{R}^{p-1}} \left\{ \ell\left(\boldsymbol{\theta}_u; \mathcal{D}_n\right) + \lambda_n ||\boldsymbol{\theta}_u||_1 \right\} \tag{6}$$

where $\ell(\boldsymbol{\theta}_u; \mathcal{D}_n) = -\sum_{t \in \mathcal{T}_n} w^\tau_t \gamma(\boldsymbol{\theta}_u; \mathbf{x}^t)$ is the weighted logloss, with weights defined as

$$w^\tau_t = \frac{K_h(t-\tau)}{\sum_{t' \in \mathcal{T}_n} K_h(t'-\tau)}$$

Algorithm 1 Graph structure estimation

Input: Dataset $\mathcal{D}_n$, time point of interest $\tau \in [0,1]$, penalty parameter λ_n, bandwidth parameter h

Output: Estimate of the graph structure $\hat{G}^\tau$

1: **for all** $u \in V$ **do**
2: Estimate $\hat{\boldsymbol{\theta}}_u$ by solving the convex program (6)
3: Estimate the set of signed neighboring edges $\hat{S}^\tau_\pm(u)$ using (7)
4: **end for**
5: Combine sets $\{\hat{S}^\tau_\pm(u)\}_{u\in V}$ to obtain $\hat{G}^\tau$.

and $K_h(\cdot) = K(\cdot/h)$ is a symmetric nonnegative kernel. The regularization parameter $\lambda_n \geq 0$ is specified by a user and controls the sparsity of the solution. The program (6) is convex and a minimum over $\boldsymbol{\theta}_u$ is always achieved, as the problem can be cast as a constrained optimization problem over the ball $||\boldsymbol{\theta}_u||_1 \leq C(\lambda_n)$ and the claim follows from the Weierstrass theorem.

Let $\hat{\boldsymbol{\theta}}^\tau_u$ be a minimizer of (6). The convex program (6) does not necessarily have a unique optimum, but as we will prove shortly, in the regime of interest any two solutions will have non-zero elements in the same positions. Based on the vector $\hat{\boldsymbol{\theta}}^\tau_u$, we have the following estimate of the signed neighborhood:

$$\hat{S}^\tau_\pm(u) := \left\{(\text{sign}(\hat{\theta}^\tau_{uv}), (u,v)) \ : \ v \in V\backslash u, \ \hat{\theta}^\tau_{uv} \neq 0\right\}. \tag{7}$$

The structure of graph G^τ is consistently estimated if every signed neighborhood is recovered, i.e. $\hat{S}^\tau_\pm(u) = S^\tau_\pm(u)$ for all $u \in V$. A summary of the algorithm is given in Algorithm 1.

The convex program (6), can be solved using any general optimization solver. One particularly fast algorithm, based on the coordinate-wise descent method, for this type of a problem is described in Friedman, Hastie, and Tibshirani [2008a] and implemented as the R package *glmnet*. Note that the algorithm provides only an estimate of the graph structure at time point τ and in order to get insight into the dynamics of the graph changes, one needs to estimate the graph structure at multiple time points. Typically, in a real application task, one is interested in estimating G^τ for all $\tau \in \mathcal{T}_n$.

4.3 Theoretical results

Under suitable conditions it is possible to show that Algorithm 1 consistently recovers the graph structure, that is

$$\mathbb{P}[\forall u \ \hat{S}^\tau_\pm(u) = S^\tau_\pm(u)] \xrightarrow{n\to\infty} 1,$$

the property known as *sparsistency*. We are interested in the high-dimensional case, where the dimension $p = p_n$ is comparable or even larger than the sample size n. It is of great interest to understand the performance of the estimator under this assumption, since in many

real world scenarios the dimensionality of data is large. Our analysis is asymptotic and we consider the model dimension $p = p_n$ to grow at a certain rate as the sample size grows. This essentially allows us to consider more "complicated" models as we observe more data points. Another quantity that will describe the complexity of the model is the maximum node degree $s = s_n$, which is also considered as a function of the sample size. Under the assumption that the true-graph structure is sparse, we will require that the maximum node degree is small, $s \ll n$. The main result describes the scaling of the triple (n, p_n, s_n) under which the estimation procedure given in the previous section estimates the graph structure consistently. Another important quantity, appearing in the statement, is the minimum value of the parameter vector that is different from zero

$$\theta_{\min} = \min_{(u,v)\in E^\tau} |\theta^\tau_{uv}|.$$

Intuitively, the success of the recovery should depend on how hard it is to distinguish the true non-zero parameters from noise.

Theorem 1. *Under a suitable technical conditions (which can be found in Kolar and Xing [2009]), Algorithm 1 with the regularization parameter that satisfy*

$$\lambda_n \geq C\frac{\sqrt{\log p}}{n^{1/3}}$$

for a constant $C > 0$ independent of (n, p, s) is used to estimate the graph structure. Assume that the following conditions hold:

1. $h = \mathcal{O}(n^{-\frac{1}{3}})$
2. $s = o(n^{1/3})$, $\frac{s^3 \log p}{n^{2/3}} = o(1)$
3. $\theta_{\min} = \Omega(\frac{\sqrt{s \log p}}{n^{1/3}})$.

Then for a fixed $\tau \in [0, 1]$ the estimated graph $\hat{G}^\tau(\lambda_n)$ obtained through neighborhood selection satisfies

$$\mathbb{P}\left[\hat{G}^\tau(\lambda_n) \neq G^\tau\right] = \mathcal{O}\left(\exp\left(-C\frac{n^{2/3}}{s^3} + C' \log p\right)\right) \to 0,$$

for some constants C', C'' independent of (n, p, s).

This theorem guarantees that Algorithm 1 asymptotically recovers the sequence of graphs underlying all the nodal-state measurements in a time series, and the snapshot of the evolving graph at any time point during measurement intervals, under appropriate regularization parameter λ_n as long as the ambient dimensionality p and the maximum node degree s are not too large, and minimum $\boldsymbol{\theta}$ values do not tend to zero too fast.

In order to obtain insight into the network dynamics one needs to estimate the graph structure at multiple time points. A common choice is to estimate the graph structure for every $\tau \in \mathcal{T}_n$ and obtain a sequence of graph structures $\{\hat{G}^\tau\}_{\tau\in\mathcal{T}_n}$. We a have the following immediate consequence of Theorem 3.

Corollary 2. *Under the assumptions of Theorem 3, we have that*

$$\mathbb{P}\left[\forall \tau \in \mathcal{T}_n \ : \ \hat{G}^\tau(\lambda_n) = G^\tau\right] \xrightarrow{n\to\infty} 1. \tag{8}$$

4.4 Empirical results

The procedure described in the previous section was used to reverse engineer the gene regulatory networks of *Drosophila melanogaster* from a time series of gene expression data measured during its full life cycle. Over the developmental course of *Drosophila melanogaster*, there exist multiple underlying "themes" that determine the functionalities of each gene and their relationships to each other, and such themes are dynamical and stochastic. As a result, the gene regulatory networks at each time point are context-dependent and can undergo systematic rewiring, rather than being invariant over time. We used microarray gene expression measurements from Arbeitman et al. [2002] as our input data and we focused on 588 genes that are known to be related to developmental process based on their gene ontologies. Results were reported in Kolar et al. [2009a] and Song et al. [2009b].

5 Discrete Markov Random Fields With Jumps (completed)

In this section, we consider estimation of time-varying discrete Markov Random Fields that are not varying smoothly over time, but have structural changes.

5.1 Model

We are given a sequence of n nodal states $\mathcal{D}_n = \{\mathbf{x}^t \sim \mathbb{P}_{\boldsymbol{\theta}^t} | t \in \mathcal{T}_n\}$, with the time index defined as $\mathcal{T}_n = \{1/n, 2/n, \ldots, 1\}$. The probability distribution $\mathbb{P}_{\boldsymbol{\theta}^t}$ has the form given in (3). Unlike the previous section where the distribution was assumed to be changing smoothly over time, we assume that there are a number of change points at which the distribution generating samples changes abruptly. Formally, we assume that for each node u, there is a partition $\mathcal{B}_u = \{0 = B_{u,0} < B_{u,1} < \ldots < B_{u,k_u} = 1\}$ of the interval $[0, 1]$, such that each element of $\boldsymbol{\theta}_u^t$ is constant on each segment of the partition. At change points some of the elements of the vector $\boldsymbol{\theta}_u^t$ may become zero, while some others may become non-zero, which corresponds to a change in the graph structure. If the number of change points is small, i.e., the graph structure changes infrequently, then there will be enough samples at a segment of the partition to estimate the non-zero pattern of the vector $\boldsymbol{\theta}^\tau$.

5.2 Estimation Procedure

In this section, we give the estimation procedure of the non-zero pattern of $\{\boldsymbol{\theta}^t\}_{t\in\mathcal{T}_n}$ under the assumption that the elements of $\boldsymbol{\theta}_u^t$ are piecewise constant functions, with pieces defined by the partition $\mathcal{B}_u$. The estimation is performed node-wise. As opposed to the kernel smoothing estimator defined in Eq. (6), which gives the estimate at one time point τ, the procedure described below simultaneously estimates $\{\hat{\boldsymbol{\theta}}_u^t\}_{t\in\mathcal{T}_n}$. The estimators $\{\hat{\boldsymbol{\theta}}_u^t\}_{t\in\mathcal{T}_n}$ are defined as a minimizer of the following convex optimization objective:

$$\underset{\boldsymbol{\theta}_u^t\in\mathbb{R}^{p-1},\ t\in\mathcal{T}_n}{\operatorname{argmin}} \{\sum_{t\in\mathcal{T}_n}\gamma(\boldsymbol{\theta}_u^t;\mathbf{x}^t)+\lambda_1\sum_{t\in\mathcal{T}_n}||\boldsymbol{\theta}_u^t||_1+\lambda_{\text{TV}}\sum_{v\in V\setminus u}\text{TV}(\{\theta_{uv}^t\}_{t\in\mathcal{T}_n})\}, \tag{9}$$

where $\text{TV}(\{\theta_{uv}^t\}_{t\in\mathcal{T}_n}) := \sum_{i=2}^n |\theta_{uv}^{i/n}-\theta_{uv}^{(i-1)/n}|$ is the total variation penalty. We will refer to this approach of obtaining an estimator as `TV`. The penalty is structured as a combination of two terms. As mentioned before, the ℓ_1 norm of the parameters is used to regularize the solution towards estimators with lots of zeros and the regularization parameter λ_1 controls the number of non-zero elements. The second term penalizes the difference between parameters that are adjacent in time and, as a result, the estimated parameters have infrequent changes across time. This composite penalty, known as the "fused" Lasso penalty, was successfully applied in a slightly different setting of signal denoising (e.g., Rinaldo [2009]) where it creates an estimate of the signal that is piecewise constant.

The optimization problem given in Eq. (9) is convex and can be solved using off-the-shelf interior point solver (e.g., the `CVX` package by Grant and Boyd [2008]). However, for large scale problems (i.e., both p and n are large), interior point method can be computationally expensive. Therefore, we propose a block-coordinate descent procedure which is much more efficient than the existing off-the-shelf solvers for large scale problems. Observe that the loss function can be decomposed as $\mathcal{L}(\{\boldsymbol{\theta}_u^t\}_{t\in\mathcal{T}_n}) = f_1(\{\boldsymbol{\theta}_u^t\}_{t\in\mathcal{T}_n}) + \sum_{v\in V\setminus u} f_2(\{\theta_{uv}^t\}_{t\in\mathcal{T}_n})$ for a smooth differentiable convex function $f_1(\{\boldsymbol{\theta}_u^t\}_{t\in\mathcal{T}_n}) = \sum_{t\in\mathcal{T}_n}\gamma(\boldsymbol{\theta}_u^t;\mathbf{x}^t)$ and a convex function $f_2(\{\theta_{uv}^t\}_{t\in\mathcal{T}_n}) = \lambda_1\sum_{t\in\mathcal{T}_n}|\theta_{uv}^t| + \lambda_{\text{TV}}\,\text{TV}(\{\theta_{uv}^t\}_{t\in\mathcal{T}_n})$. Tseng [2001] established that the block-coordinate descent converges for loss functions with such structure. Based on this observation we propose the following algorithm:

1. Set initial values: $\hat{\boldsymbol{\theta}}_u^{t,0} \leftarrow \mathbf{0}, \quad \forall t\in\mathcal{T}_n$
2. For each $v\in V\setminus u$, set the current estimates $\{\hat{\theta}_{uv}^{t,iter+1}\}_{t\in\mathcal{T}_n}$ as a solution to the following optimization procedure:

$$\min_{\{\theta^t\in\mathbb{R}\}_{t\in\mathcal{T}_n}} \left\{ \begin{array}{c} \sum_{t\in\mathcal{T}_n}\gamma\left(\hat{\theta}_{u,1}^{t,iter+1},\ldots,\hat{\theta}_{u,v-1}^{t,iter+1},\theta^t,\hat{\theta}_{u,v+1}^{t,iter},\ldots,\hat{\theta}_{u,p-1}^{t,iter};\mathbf{x}^t\right) \\ +\lambda_1\sum_{t\in\mathcal{T}^n}|\theta^t| + \lambda_{\text{TV}}\,\text{TV}(\{\theta^t\}_{t\in\mathcal{T}_n}) \end{array} \right\} \tag{10}$$

3. Repeat step 2 until convergence

Using the proposed block-coordinate descent algorithm, we solve a sequence of optimization problems each with only n variables given in Eq. (10), instead of solving one big optimization problem with $n(n-1)$ variables given in Eq. (9). In our experiments, we find that the optimization in Eq. (9) can be estimated in an hour when the number of covariates is up to few hundreds and when the number of time points is also in hundreds. Here, the bottleneck is the number of time points. Observe that the dimensionality of the problem in Eq. (10) grows linearly with the number of time points. Again, the overall estimation procedure decouples to a collection of smaller problems which can be trivially parallelized. If we treat the optimization in Eq. (9) as an atomic operation, the overall algorithm scales linearly as a function of the number of covariates p, i.e. $\mathcal{O}(p)$.

5.3 Empirical results

Using the algorithm described in the previous section, we reverse engineered the latent sequence of temporally rewiring political networks between Senators from the US Senate voting records. The US senate data consists of voting records from 109th congress (2005 - 2006). There are 100 senators whose votes were recorded on the 542 bills. Each senator corresponds to a variable, while the votes are samples recorded as -1 for no and 1 for yes. This data set was analyzed in Banerjee et al. [2008], where a static network was estimated. We analyzed this data set in a time varying framework in order to discover how the relationship between senators changes over time. Results were reported in Kolar et al. [2010b].

6 Sparsistent Estimation Of Smoothly Varying Gaussian Graphical Models (completed)

In this section, we address the problem of consistent estimation of the sparsity pattern of a precision matrix when the underlying probability distribution changes smoothly.

6.1 Model

We use the framework of Zhou et al. [2008]. Let

$$\mathbf{x}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}^{t_i}), \quad i = 1, \ldots, n \tag{11}$$

be an independent sequence of p-dimensional observations distributed according to a multivariate normal distribution whose covariance matrix changes smoothly over time. Assume for simplicity that the time points are equidistant on a unit interval, that is, $t_i = i/n$. A graph $G^{t_i} = (V, E^{t_i})$ is associated with each observation $\mathbf{x}^i$ and it represents the non-zero elements of the precision matrix $\mathbf{\Omega}^{t_i} \triangleq (\mathbf{\Sigma}^{t_i})^{-1}$ (recall that $e_{ab} \in E^{t_i}$ only if $\omega_{ab}^{t_i} \neq 0$). With changing precision matrix $\mathbf{\Omega}^{t_i}$, the associated graphs change as well, which allows for modelling of dynamic networks. The model given in (11) can be thought of as a special case

of the varying coefficient models introduced in Hastie and Tibshirani [1993]. In particular, the model in (11), inherits flexibility and modelling power from the class of nonparametric models, but at the same time it retains interpretability of parametric models. Indeed, there are no assumptions on the parametric form of the elements of the covariance matrix $\mathbf{\Sigma}^t$ as a function of time.

Under the model (11), Zhou et al. [2008] studied the problem of the consistent recovery in the Frobenius norm of $\mathbf{\Omega}^\tau$ for some $\tau \in [0,1]$, as well as the predictive performance of the fitted model. While those results are very interesting and important in statistics, in many application areas, it is the graph structure that provides most insight into complex systems by allowing visualization of relational structures and mechanisms that explain the data. For example, in computational biology, a graph estimated from a gene expression microarray profile can reveal the topology of genetic regulation circuitry, while in sociocultural analysis, a graph structure helps identify communities and communication patterns among actors. Unfortunately, the consistent estimation of the graph structure does not follow immediately from the consistent estimation of the precision matrix $\mathbf{\Omega}$.

We address the problem of the consistent graph structure recovery under the model (11) in this section. We establish sufficient conditions for the penalized likelihood procedure, proposed in Zhou et al. [2008], to estimate the graph structure consistently. Furthermore, we modify the neighborhood selection procedure of Meinshausen and Bühlmann [2006] to estimate the graph structure under the model (11) and provide sufficient conditions for the graph recovery.

6.2 Penalized Likelihood Estimation

In this section, we show that, under some technical conditions, the procedure proposed in Zhou et al. [2008] is able to consistently estimate the set of non-zero elements of the precision matrix $\mathbf{\Omega}^\tau$ at a given time point $\tau \in [0,1]$. Under the model (11), an estimator of the precision matrix can be obtained by minimizing the following objective

$$\hat{\mathbf{\Omega}}^\tau = \underset{\mathbf{\Omega} \succ 0}{\operatorname{argmin}} \left\{ \operatorname{tr} \mathbf{\Omega} \hat{\mathbf{\Sigma}}^\tau - \log |\mathbf{\Omega}| + \lambda ||\mathbf{\Omega}^-||_1 \right\}. \tag{12}$$

where $\hat{\mathbf{\Sigma}}^\tau = \sum_i w_i^\tau \mathbf{x}^i (\mathbf{x}^i)'$ is the weighted sample covariance matrix, with weights defined as

$$w_i^\tau = \frac{K_h(t_i - \tau)}{\sum_i K_h(t_i - \tau)}, \tag{13}$$

$K : \mathbb{R} \mapsto \mathbb{R}$ being the kernel function and $K_h(\cdot) = K(\cdot/h)$. The tuning parameter λ controls the number of non-zero pattern of the estimated precision matrix, while the bandwidth parameter h controls the smoothness over time of the estimated precision matrix and the effective sample size. These tuning parameters depend on the sample size n, but we will omit this dependence in our notation.

Theorem 3. *Fix a time point of interest $\tau \in [0,1]$. Let $\{\mathbf{x}^i\}$ be an independent sample according to the model* (11)*. Under suitable technical conditions (which are provided in Kolar and Xing [2011]) there exists a constant $C > 0$ for which the following holds. Suppose that the weighted sample covariance matrix $\hat{\mathbf{\Sigma}}^\tau$ is estimated using the kernel with the bandwidth parameter satisfying $h = \mathcal{O}\left(n^{-1/3}\right)$. If the penalty parameter λ in* (12) *scales as $\lambda = \mathcal{O}\left(n^{-1/3}\sqrt{\log p}\right)$ and the sample size satisfies $n > Cd^3(\log p)^{3/2}$, then the minimizer $\hat{\mathbf{\Omega}}^\tau$ of* (12) *defines the edge set $\hat{E}^\tau$ which satisfies*

$$\begin{aligned}\mathbb{P}[\hat{E}^\tau \neq \{(a,b) \ : a \neq b, |\omega^\tau_{ab}| > \omega_{\min}\}] \\ = \mathcal{O}(\exp(-c\log p)) \to 0,\end{aligned}$$

for some constant $c > 0$, with $\omega_{\min} = M_\omega n^{-1/3}\sqrt{\log p}$ and M_ω being a sufficiently large constant.

The theorem states that all the non-zero elements of the population precision matrix $\mathbf{\Omega}^\tau$, which are larger in absolute value than $\omega_{\min}$, will be identified. Note that if the elements of the precision matrix are too small, then the estimation procedure is not able to distinguish them from zero. Furthermore, the estimation procedure does not falsely include zero elements into the estimated set of edges. The theorem guarantees consistent recovery of the set of sufficiently large non-zero elements of the precision matrix at the time point τ. In order to obtain insight into the network dynamics, the graph corresponding to $\mathbf{\Omega}^t$ needs to be estimated at multiple time points. Due to the slow rate of convergence of $\hat{\mathbf{\Omega}}^t$, it is sufficient to estimate a graph at each time point t_i, $i = 1, \ldots, n$.

6.3 Neighborhood Selection Estimation

In this section, we discuss the neighborhood selection approach to selection of non-zero elements of the precision matrix $\mathbf{\Omega}^\tau$ under the model (11). The neighborhood selection procedure was proposed in Meinshausen and Bühlmann [2006] as a way to estimate the graph structure associated to a GGM from an *i.i.d.* sample. As opposed to optimizing penalized likelihood, the neighborhood selection method is based on optimizing penalized pseudo-likelihood on each node of the graph, which results in local estimation of the graph structure.

We start by describing the neighborhood selection method under the model (11). Consider the following estimator for $\boldsymbol{\theta}^\tau_{\backslash a} \triangleq \{\theta^\tau_{ab}\}_{b \in V \backslash \{a\}}$,

$$\hat{\boldsymbol{\theta}}^\tau_{\backslash a} \triangleq \underset{\boldsymbol{\theta} \in \mathbb{R}^{p-1}}{\operatorname{argmin}} \ \sum_i (x^i_a - \sum_{b \neq a} x^i_b \theta_b)^2 w^\tau_i + \lambda ||\boldsymbol{\theta}||_1, \tag{14}$$

where the weight w^τ_i are defined in (13). The estimator $\hat{\boldsymbol{\theta}}^\tau_{\backslash a}$ defines the neighborhood of the node $a \in V$ at the time point τ as $\hat{N}^\tau_a \triangleq N(\hat{\boldsymbol{\theta}}^\tau_{\backslash a})$. By estimating the neighborhood of each node and combining them, the whole graph structure can be obtained.

We have the following result for the neighborhood selection procedure.

Theorem 4. *Fix a time point of interest $\tau \in [0,1]$. Let $\{\mathbf{x}^i\}$ be an independent sample according to the model* (11)*. Under suitable technical conditions (which are provided in Kolar and Xing [2011]) there exists a constant $C > 0$ for which the following holds. Suppose that the bandwidth parameter used in* (14) *satisfies $h = \mathcal{O}\left(n^{-1/3}\right)$. If the penalty parameter λ in* (14) *scales as $\lambda = \mathcal{O}\left(n^{-1/3}\sqrt{\log p}\right)$ and the sample size satisfies $n > Cd^{3/2}(\log p)^{3/2}$, then the neighborhood selection procedure defines the edge set $\hat{E}^\tau$, by solving* (14) *for all $a \in V$, which satisfies*

$$\begin{aligned}\mathbb{P}[\hat{E}^\tau \neq \{(a,b) \ : a \neq b, |\theta^\tau_{ab}| > \theta_{\min}\}] \\ = \mathcal{O}(\exp(-cn^{2/3}(d\log p)^{-1})) \to 0,\end{aligned}$$

for some constant $c > 0$, with $\theta_{\min} = M_\theta n^{-1/3}\sqrt{d\log p}$ and M_θ being a sufficiently large constant.

The theorem states that the neighborhood selection procedure can be used to estimate the pattern of non-zero elements of the matrix $\mathbf{\Omega}^\tau$ that are sufficiently large, as defined by $\theta_{\min}$. In order to gain insight into the network dynamics, the graph structure needs to be estimated at multiple time points.

The advantage of the neighborhood selection procedure over the penalized likelihood procedure is that it allows for very simple parallel implementation, since the neighborhood of each node can be estimated independently. Furthermore, the assumptions under which the neighborhood selection procedure consistently estimates the structure of the graph are weaker. Therefore, since the network structure is important in many problems, it seems that the neighborhood selection procedure should be the method of choice. However, in problems where the estimated coefficients of the precision matrix are also of importance, the penalized likelihood approach has the advantage over the neighborhood selection procedure. In order to estimate the precision matrix using the neighborhood selection, one needs first to estimate the structure and then fit the parameters subject to the structural constraints. However, it was pointed out by Breiman [1996] that such two step procedures are not stable.

7 Time Varying Gaussian Graphical Models With Jumps (in progress)

In this section, we discuss varying coefficient varying structure (VCVS) models for GGMs Kolar et al. [2009b]. We consider graph structure changing with time in a piece-wise constant manner.

7.1 Model

Let $\{\mathbf{x}_i\}_{i\in[n]} \in \mathbb{R}^p$ be a sequence of n independent observations from some p-dimensional multivariate normal distributions, not necessarily the same for every observation. Let $\{\mathcal{B}^j\}_{j\in[B]}$ be a disjoint partitioning of the set $[n]$ where each block of the partition consists of consecutive elements, that is, $\mathcal{B}^j \cap \mathcal{B}^{j'} = \emptyset$ for $j \neq j'$ and $\bigcup_j \mathcal{B}^j = [n]$ and $\mathcal{B}^j = [T_{j-1} : T_j] := \{T_{j-1}, T_{j-1}+1, \ldots, T_j - 1\}$. Let $\mathcal{T} := \{T_0 = 1 < T_1 < \ldots < T_B = n+1\}$ denote the set of partition boundaries. We consider the following model

$$\mathbf{x}_i \sim \mathcal{N}_p(\mathbf{0}, \mathbf{\Sigma}^j), \qquad i \in \mathcal{B}^j, \tag{15}$$

so that observations indexed by elements in $\mathcal{B}^j$ are p-dimensional realizations of a multivariate normal distribution with zero mean and the covariance matrix $\mathbf{\Sigma}^j = (\sigma^j_{ab})_{a,b\in[p]}$. Let $\mathbf{\Omega}^j := (\mathbf{\Sigma}^j)^{-1}$ denote the precision matrix with elements $(\omega^j_{ab})_{a,b\in[p]}$. With the number of partitions, B, and the boundaries of partitions, $\mathcal{T}$, unknown, we study the problem of estimating both the partition set $\{\mathcal{B}^j\}$ and the non-zero elements of the precision matrices $\{\mathbf{\Omega}^j\}_{j\in[B]}$ from the sample $\{\mathbf{x}_i\}_{i\in[n]}$. Note that in this section we study a particular case of the VCVS model, where the coefficients are piece-wise constant functions of time.

If the partitions $\{\mathcal{B}^j\}_j$ were known, the problem would be trivially reduced to the setting analyzed in the previous work. Dealing with the unknown partitions, together with the structure estimation of the model, is a much more challenging problem. In this section, we propose and analyze a method based on *time-coupled neighborhood selection*, where the model estimates are forced to stay similar across time using a fusion-type total variation penalty and the sparsity of each neighborhood is obtained through the ℓ_1 penalty.

7.2 Separate Estimation of Jumps and Networks

Consider the following regression model:

$$Y_i = \mathbf{X}_i'\beta(t_i) + \epsilon_i, \quad i = 1, \ldots, n, \tag{16}$$

where the design variables $\mathbf{X}_i \in \mathbb{R}^p$ are *i.i.d.* zero mean random variables sampled at some conditions indexed by $i = 1, \ldots, n$, such as the prices of a set of stocks at time i, or the signals from some sensors deployed at location i; the noise $\epsilon_1, \ldots, \epsilon_n$ are *i.i.d.* Gaussian variables with variance σ^2 independent of the design variables; and $\beta(t_i) = (\beta_1(t_i), \ldots, \beta_p(t_i))' : [0,1] \mapsto \mathbb{R}^p$ is a vector of unknown coefficient functions.

Coefficient functions can be related to the elements of the precision matrix in (15). This relationship can be exploited for a development of a neighborhood selection procedure, which was already shown in the previous chapter. Therefore, here we will focus on variable selection under the model (16).

We propose the following algorithm for estimating the time-varying structure of the varying-coefficient model in Eq. (16). The algorithm is a two-step procedure summarized as follows:

1. Estimate the block partition $\hat{\mathcal{T}}$, on which the coefficient vector is constant within each block. This can be obtained by minimizing the following objective:

$$\sum_{i=1}^{n} \left(Y_i - \mathbf{X}_i' \beta(t_i)\right)^2 + 2\lambda_2 \sum_{k=1}^{p} ||\beta_k||_{\mathrm{TV}}, \tag{17}$$

which we refer to as a *temporal difference* (TD) regression for reasons that will be clear shortly. We will employ a TD-transformation to Eq. (17) and turn it into an ℓ_1-regularized regression problem, and solve it using the randomized Lasso. Details of the algorithm and how to extract $\hat{\mathcal{T}}$ from the TD-estimate are given in Kolar et al. [2009b].

2. For each block of the partition, $\hat{\mathcal{B}}_j, 1 \leq j \leq \hat{B}$, estimate $\hat{\gamma}_j$ by minimizing the Lasso objective within the block:

$$\hat{\gamma}_j = \underset{\gamma \in \mathbb{R}^p}{\operatorname{argmin}} \sum_{t_i \in \hat{\mathcal{B}}_j} (Y_i - \mathbf{X}_i' \gamma)^2 + 2\lambda_1 ||\gamma||_1. \tag{18}$$

We name this procedure TDB-Lasso (or TDBL), after the two steps (TD randomized Lasso, and Lasso within Blocks) given above.

We have the following theoretical result for TDBL.

Theorem 5. *Under suitable technical conditions (given in Kolar et al. [2009b]) TDBL satisfies*

$$\lim_{n \to \infty} \mathbb{P}(\hat{B} = B) = 1, \tag{19}$$

$$\lim_{n \to \infty} \max_{1 \leq j \leq B} \mathbb{P}(||\hat{\gamma}_j - \gamma_j||_1 = 0) = 1, \tag{20}$$

$$\lim_{n \to \infty} \min_{1 \leq j \leq B} \mathbb{P}(\hat{S}_{\mathcal{B}_j} = S_{\mathcal{B}_j}) = 1. \tag{21}$$

Theorem 5 establishes that, under suitable conditions, TDBL consistently estimates correct number of blocks. Furthermore, on each block regression coefficients are consistently estimated and the relevant variables are selected consistently.

7.3 Joint Estimation of Jumps and Networks

In the previous section, we gave a two step procedure for estimation of change points and graph structure. That procedure can identify the correct structure under a very strong assumptions. In this section, we present a procedure that jointly estimates the jumps and network structure.

We build on the neighborhood selection procedure to estimate the changing graph structure in model (15). We use S_a^j to denote the neighborhood of the node a on the block $\mathcal{B}^j$ and

N_a^j to denote nodes not in the neighborhood of the node a on the j-th block, $N_a^j = V \backslash S_a^j$. Consider the following estimation procedure

$$\hat{\boldsymbol{\beta}}^a = \underset{\boldsymbol{\beta} \in \mathbb{R}^{p-1 \times n}}{\operatorname{argmin}} \ \mathcal{L}(\boldsymbol{\beta}) + \operatorname{pen}_{\lambda_1, \lambda_2}(\boldsymbol{\beta}) \tag{22}$$

where the loss is defined for $\boldsymbol{\beta} = (\beta_{b,i})_{b \in [p-1], i \in [n]}$ as

$$\mathcal{L}(\boldsymbol{\beta}) := \sum_{i \in [n]} \left(x_{i,a} - \sum_{b \in \backslash a} x_{i,b} \beta_{b,i} \right)^2 \tag{23}$$

and the penalty is defined as

$$\operatorname{pen}_{\lambda_1, \lambda_2}(\boldsymbol{\beta}) := 2\lambda_1 \sum_{i=2}^{n} ||\boldsymbol{\beta}_{\cdot,i} - \boldsymbol{\beta}_{\cdot,i-1}||_2 + 2\lambda_2 \sum_{i=1}^{n} \sum_{b \in \backslash a} |\beta_{b,i}|. \tag{24}$$

The penalty term is constructed from two terms. The first term ensures that the solution is going to be piecewise constant for some partition of $[n]$ (possibly a trivial one). The first term can be seen as a sparsity inducing term in the temporal domain, since it penalizes the difference between the coefficients $\boldsymbol{\beta}_{\cdot,i}$ and $\boldsymbol{\beta}_{\cdot,i+1}$ at successive time-points. The second term results in estimates that have many zero coefficients within each block of the partition. The estimated set of partition boundaries

$$\hat{\mathcal{T}} = \{\hat{T}_0 = 1\} \cup \{\hat{T}_j \in [2:n] \ : \ \hat{\boldsymbol{\beta}}^a_{\cdot,\hat{T}_j} \neq \hat{\boldsymbol{\beta}}^a_{\cdot,\hat{T}_j - 1}\} \cup \{\hat{T}_{\hat{B}} = n+1\}$$

contains indices of points at which a change is estimated, with $\hat{B}$ being an estimate of the number of blocks B. The estimated number of the block $\hat{B}$ is controlled through the user defined penalty parameter λ_1, while the sparsity of the neighborhood is controlled through the penalty parameter λ_2.

Based on the estimated set of partition boundaries $\hat{\mathcal{T}}$, we can define the neighborhood estimate of the node a for each estimated block. Let $\hat{\boldsymbol{\theta}}^{a,j} = \hat{\boldsymbol{\beta}}^a_{\cdot,i}$, $\forall i \in [\hat{T}_{j-1} : \hat{T}_j]$ be the estimated coefficient vector for the block $\hat{\mathcal{B}}^j = [\hat{T}_{j-1} : \hat{T}_j]$. Using the estimated vector $\hat{\boldsymbol{\theta}}^{a,j}$, we define the neighborhood estimate of the node a for the block $\hat{\mathcal{B}}^j$ as

$$\hat{S}_a^j := S(\hat{\boldsymbol{\theta}}^{a,j}) := \{b \in \backslash a \ : \ \hat{\theta}_b^{a,j} \neq 0\}.$$

Solving (22) for each node $a \in V$ gives us a neighborhood estimate for each node. Combining the neighborhood estimates we can obtain an estimate of the graph structure for each point $i \in [n]$.

We have the following results for the joint estimation.

Theorem 6. *Let $\{\mathbf{x}_i\}_{i \in [n]}$ be a sequence of observation according to the model in (15). Assume that suitable technical conditions hold (given in Kolar and Xing [2010a]). Suppose that the penalty parameters λ_1 and λ_2 satisfy*

$$\lambda_1 \asymp \lambda_2 = \mathcal{O}(\sqrt{\log(n)/n}). \tag{25}$$

Let $\{\hat{\boldsymbol{\beta}}_{\cdot,i}\}_{i\in[n]}$ *be any solution of* (22) *and let* $\hat{\mathcal{T}}$ *be the associated estimate of the block partition. Let* $\{\delta_n\}_{n\geq 1}$ *be a non-increasing positive sequence that converges to zero as* $n \to \infty$ *and satisfies* $\Delta_{\min} \geq n\delta_n$ *for all* $n \geq 1$*. Furthermore, suppose that* $(n\delta_n\xi_{\min})^{-1}\lambda_1 \to 0$*,* $\xi_{\min}^{-1}\sqrt{p}\lambda_2 \to 0$ *and* $(\xi_{\min}\sqrt{n\delta_n})^{-1}\sqrt{p\log n} \to 0$*, then if* $|\hat{\mathcal{T}}| = B + 1$ *the following holds*

$$\mathbb{P}[\max_{j\in[B]} |T_j - \hat{T}_j| \leq n\delta_n] \xrightarrow{n\to\infty} 1.$$

Theorem 6 states that if the number of jumps is known, then the positions of jumps can be consistently estimated. The following results shows that the correct model

Theorem 7. *Let* $\{\mathbf{x}_i\}_{i\in[n]}$ *be a sequence of observation according to the model in* (15)*. Assume that the conditions of theorem 6 are satisfied. Then, if* $|\hat{\mathcal{T}}| = B + 1$*, it holds that*

$$\mathbb{P}[S^k = S(\hat{\boldsymbol{\theta}}^k)] \xrightarrow{n\to\infty} 1, \qquad \forall k \in [B].$$

7.4 Empirical Results

The graph structure estimation using the TDB-Lasso is demonstrated on a real dataset of electroencephalogram (EEG) measurements. We used the brain computer interface (BCI) dataset IVa from Dornhege et al. [2004] in which the EEG data is collected from 5 subjects, who were given visual cues based on which they were required to imagine right hand or right foot for 3.5s. Results are reported in Kolar et al. [2009b].

7.5 Future work

In this chapter, we have studied estimation of both the change points and the network structure on each block where the network is not changing. Estimation is done using a surrogate loss function instead of the penalized maximum approach. We plan to investigate the problems of the form

$$\arg\min_{\{\boldsymbol{\Omega}^t\}_t} \sum_{t\in\mathcal{T}_n} \operatorname{tr}\boldsymbol{\Omega}^t\mathbf{x}\mathbf{x}' - \log|\boldsymbol{\Omega}^t| + \operatorname{pen}_{\lambda_1,\lambda_2}(\{\boldsymbol{\Omega}^t\}_t)$$

where the penalty function would enforce the resulting precision matrices to be piecewise constant and sparse.

Another future plan involves investigating different penalty functions than the one given in (24), such as

$$\tilde{\operatorname{pen}}_{\lambda_1,\lambda_2}(\boldsymbol{\beta}) := 2\lambda_1 \sum_{i=2}^{n} ||\boldsymbol{\beta}_{\cdot,i} - \boldsymbol{\beta}_{\cdot,i-1}||_\infty + 2\lambda_2 \sum_{i=1}^{n}\sum_{b\in\setminus a} |\beta_{b,i}|.$$

The ℓ_∞-norm penalization has been used previously to identify relevant variables in a multi-task learning setting. We hope that it could be used to identify change points more efficiently.

8 Conditional Estimation of Covariance Models (in progress)

In the previous sections, we discussed estimation of network structures as a function of time, however, in many applications, it is more natural to think of a network changing as a function of some other random variable. In this section, we focus on conditional estimation of network structures. We start by motivating the problem by few real world applications.

Consider the problem of gene network inference in systems biology, which is of increasing importance in drug development and disease treatment. A gene network is commonly represented as a fixed network, with edge weights denoting strength of associations between genes. Realistically, the strength of associations between genes can depend on many covariates such as blood pressure, sugar levels, and other body indicators; however, biologists have very little knowledge on how various factors affect strength of associations. Ignoring the influence of different factors leads to estimation procedures that overlook important subtleties of the regulatory networks. Consider another problem in quantitative finance, for which one wants to understand how different stocks are associated and how these associations vary with respect to external factors to help investors construct a diversified portfolio. The rule of *Diversification*, formalized by Modern Portfolio Theory Markowitz [1952], dictates that risk can be reduced by constructing a portfolio out of uncorrelated assets. However, it also assumes that the associations between assets are fixed (which is highly unrealistic) and a more robust approach to modeling assets would take into account how their associations change with respect to economic indicators, such as, gross domestic product (GDP), oil price or inflation rate. Unfortunately, there is very little domain knowledge on the exact relationship between economic indicators and associations between assets, which motivates the problem of *conditional covariance selection* we intend to investigate in this section.

8.1 Model

Let $\mathbf{X} \in \mathbb{R}^p$ denote a p-dimensional random vector representing genes or stock values, and $Z \in \mathbb{R}$ denote an index random variable representing some body factor or economic indicator of interest. Both of the above mentioned problems in biology and finance can be modeled as inferring non-zero partial correlations between different components of the random vector $\mathbf{X}$ conditioned on a particular value of the index variable $Z = z$. We assume that the value of partial correlations change with z, however, the set of non-zero partial correlations is constant with respect to z. Let $\mathbf{\Sigma}(z) := \mathrm{Cov}(\mathbf{X}|Z = z)$ denote the *conditional* covariance of $\mathbf{X}$ given Z, which we assume to be positive definite, and let $\mathbf{\Omega}(z) := \mathbf{\Sigma}(z)^{-1}$ denote the conditional precision matrix. The structure of non-zero components of the matrix $\mathbf{\Omega}(z)$ tells us a lot about associations between different components of the vector $\mathbf{X}$, since the elements of $\mathbf{\Omega}(z)$ correspond to partial correlation coefficients. In this section we address the challenge of selecting non-zero components of $\mathbf{\Omega}(z)$ from noisy samples. Usually, very little is known about the relationship between the index variable Z and associations between components of the random variable $\mathbf{X}$; so, we develop a nonparametric method for estimating the non-

zero elements of $\mathbf{\Omega}(z)$. Specifically, we develop a new method based on ℓ_1/ℓ_2 penalized kernel smoothing, that is able to estimate the functional relationship between the index Z and components of $\mathbf{\Omega}(z)$ with minimal assumptions on the distribution $(\mathbf{X}, Z)$ and only smoothness assumption on $z \mapsto \mathbf{\Omega}(z)$. In addition to developing an estimation procedure that works with minimal assumptions, we also focus on statistical properties of the estimator in the high-dimensional setting, where the number of dimensions p is comparable or even larger than the sample size. Ubiquity of high-dimensionality in many real world data forces us to carefully analyze statistical properties of the estimator, that would otherwise be apparent in a low-dimensional setting.

8.2 Estimation

Let $\mathbf{X} = (X_1, \ldots, X_p)^T \in \mathbb{R}^p$ be a p-dimensional random vector (representing gene expressions or stock values) and let random variable $Z \in [0, 1]$ be an associated univariate index (representing a body factor or an economy index). We will estimate associations between different components of $\mathbf{X}$ conditionally on Z. For simplicity of presentation, we assume that the index variable can be scaled into the interval $[0, 1]$ and, furthermore, we assume that it is a scalar variable. The kernel smoothing method, to be introduced, can be easily extended to multivariate Z. However, such an extension may only be practical in limited cases, due to the curse of dimensionality Li and Liang [2008]. Throughout the paper, we assume that $\mathbb{E}[\mathbf{X}|Z = z] = 0$ for all $z \in [0, 1]$. In practice, one can easily estimate the conditional mean of $\mathbf{X}$ given Z using local polynomial fitting Fan [1993] and subtract it from $\mathbf{X}$. We denote the conditional covariance matrix of $\mathbf{X}$ given Z as $\mathbf{\Sigma}(z) := \mathrm{Cov}(\mathbf{X}|Z = z) = (\sigma_{uv}(z))_{u,v\in[p]}$, where we use $[p]$ to denote the set $\{1, \ldots, p\}$. Assuming that $\mathbf{\Sigma}(z)$ is positive definite, for all $z \in [0, 1]$, the conditional precision matrix is given as $\mathbf{\Omega}(z) := \mathbf{\Sigma}(z)^{-1} = (\omega_{uv}(z))_{u,v\in[p]}$. Elements $(\omega_{uv}(z))_{u,v\in[p]}$ are smooth, but unknown functions of z.

With the notation introduced above, the problem of conditional covariance selection, e.g., recovering the strength of association between stocks as a function of oil price, or association between gene expressions as a function of blood pressure, can be formulated as estimating the non-zero elements in the conditional precision matrix $\mathbf{\Omega}(z)$. As mentioned before, association between different components of $\mathbf{X}$ can be expressed using the partial correlation coefficients, which are directly related to the elements of precision matrix as follows; the partial correlation $\rho_{uv}(z)$ between X_u and X_v $(u, v \in [p])$ given $Z = z$ can be computed as

$$\rho_{uv}(z) = -\frac{\omega_{uv}(z)}{\sqrt{\omega_{uu}(z)\omega_{vv}(z)}}. \tag{26}$$

The above equation confirms that the non-zero partial correlation coefficients can be selected by estimating non-zero elements of the precision matrix. Let $S := \{(u, v) : \int_{[0,1]} \omega_{uv}^2(z)dz > 0,\ u \neq v\}$ denote the set of non-zero partial correlation coefficients, which we assume to be constant with respect to z, i.e., we assume that the associations are fixed, but their strength can vary with respect to the index z. Furthermore, we assume that the number of non-zero partial correlation coefficients, $s := |S|$, is small. This is a reasonable assumption for

many problems, e.g., in biological systems a gene usually interacts with only a handful of other genes. In the following paragraphs, we relate the partial correlation coefficients to a regression problem, and present a computationally efficient method for estimating non-zero elements of the precision matrix based on this insight.

For each component X_u ($u \in [p]$) we set up a regression model, where X_u is the response variable, and all the other components are the covariates. Let $\mathbf{X}_{\setminus u} := \{X_v : v \neq u, v \in [p]\}$. Then we have the following regression model

$$X_u = \sum_{v \neq u} X_v b_{uv}(z) + \epsilon_u(z), \quad u \in [p], \tag{27}$$

with $\epsilon_u(z)$ being uncorrelated with $\mathbf{X}_{\setminus u}$ if and only if

$$b_{uv}(z) = -\frac{\omega_{uv}(z)}{\omega_{uu}(z)} = \rho_{uv}(z)\sqrt{\frac{\omega_{vv}(z)}{\omega_{uu}(z)}}. \tag{28}$$

We propose a locally weighted kernel estimator of the non-zero partial correlations. Let $\mathcal{D}^n = \{(\mathbf{x}^i, z^i)\}_{i \in [n]}$ be an independent sample of n realizations of $(\mathbf{X}, Z)$. For each $u \in [p]$, we define the loss function

$$\begin{aligned} \mathcal{L}_u(\mathbf{B}_u; \mathcal{D}^n) := & \\ \sum_{z \in \{z^j\}_{j \in [n]}} & \sum_{i \in [n]} \left(x_u^i - \sum_{v \neq u} x_v^i b_{uv}(z)\right)^2 K_h(z - z^i) \\ & + 2\lambda \sum_{v \neq u} ||b_{uv}(\cdot)||_2 \end{aligned} \tag{29}$$

where $\mathbf{B}_u = (\mathbf{b}_u(z^1), \ldots, \mathbf{b}_u(z^n))$, $\mathbf{b}_u(z^j) \in \mathbb{R}^{p-1}$, $K_h(z - z^i) = K(\frac{|z - z^i|}{h})$ is a symmetric density function with bounded support that defines local weights, h denotes the bandwidth, λ is the penalty parameter and $||b_{uv}(\cdot)||_2 := \sqrt{\sum_{z \in \{z^j\}_{j \in [n]}} b_{uv}(z)^2}$. Define $\hat{\mathbf{B}}_u$ as a minimizer of the loss

$$\hat{\mathbf{B}}_u := \underset{\mathbf{B} \in \mathbb{R}^{p-1 \times n}}{\operatorname{argmin}} \mathcal{L}_u(\mathbf{B}; \mathcal{D}^n). \tag{30}$$

Combing $\{\hat{\mathbf{B}}_u\}_{u \in [p]}$ gives an estimator

$$\hat{S} := \{(u, v) : \max\{||\hat{b}_{uv}(\cdot)||_2, ||\hat{b}_{vu}(\cdot)||_2\} > 0\} \tag{31}$$

of the non-zero elements of the precision matrix.

In Eq. (29), the ℓ_1/ℓ_2 norm is used to penalize model parameters. This norm is commonly used in the Group Lasso Yuan and Lin [2006]. In our case, since we assume the set of non-zero elements S, of the precision matrix, to be fixed with respect to z, the ℓ_2 norm is a natural way to shrink the whole group of coefficients $\{b_{uv}(z^i)\}_{i \in [n]}$ to zero. Note that the group consists of the same element, say (u, v), of the precision matrix for different values of z.

We can prove the following result for the estimation procedure outlined above.

Theorem 8. *Assume that the regularity conditions (given in Kolar et al. [2010a]) are satisfied. Furthermore, assume that* $\mathbb{E}[\exp(tX)|Z = z] \leq \exp(\sigma^2 t^2/2)$ *for all* $z \in [0,1]$, $t \in \mathbb{R}$ *and some* $\sigma \in (0, \infty)$. *Let* $h = \mathcal{O}(n^{-1/5})$, $\lambda = \mathcal{O}(n^{7/10}\sqrt{\log p})$ *and* $n^{-9/5}\lambda \to 0$. *If* $\frac{n^{11/10}}{\sqrt{\log p}} \min_{u,v \in S} ||b_{uv}(\cdot)||_2 \to \infty$, *then* $\mathbb{P}[\hat{S} = S] \to 1$.

8.3 Related Work

Estimation of high-dimensional time-varying graphical models discussed in previous chapters is closely related to the scenario discussed in this chapter. While the work on time-varying graphs could fit into this framework, there are a few mayor differences. The most notables difference is the dependence of network structure on time, which is not random. Although the estimation procedures are similar, the theoretical analysis is quite different.

There are only few references for work on nonparametric models for conditional covariance and precision matrices. Yin et al. [2008] develop a kernel estimator of the conditional covariance matrix based on the local-likelihood approach. Since their approach does not perform estimation of non-zero elements in the precision matrix, it is suitable in low-dimensions. Other related work includes nonparametric estimation of the conditional variance function in longitudinal studies [see Ruppert et al., 1997, Fan and Yao, 1998, and references within].

8.4 Empirical Results

We applied our method to analyzing relationships among stocks in the S&P 500. Such an analysis would be useful to an economist studying the effect of various indicators on the market, or an investor who is seeking to minimize his risk by constructing a diverse portfolio according to Modern Portfolio Theory Markowitz [1952]. Rather than assume static associations among stocks we believe it is more realistic to model them as a function of an economic indicator, such as oil price. The results were reported in Kolar et al. [2010a].

8.5 Future Work

We have proposed to estimate conditional covariance matrix using kernel smoothing techniques and have established convergence results for the estimation procedure. In the future, we plan to investigate information theoretic lower bounds for this problem, similar to Cai et al. [2010].

9 Time Varying Dynamic Bayesian Networks (in progress)

In the previous sections, we have focused on estimation of undirected graphical models. However, what has not been addressed so far is how to recover *directed* time-varying net-

works. In this section, we address the problem of learning time-varying dynamic Bayesian networks (TV-DBN) for modeling the directed time-evolving network structures underlying non-stationary biological time series. As before, to make this problem statistically tractable, we rely on the assumption that the underlying network structures are sparse and vary smoothly across time. We propose a kernel reweighted ℓ_1-regularized auto-regressive approach for learning this sequence of networks.

9.1 Model

We concern ourselves with stochastic processes in time or space domains, such as the dynamic control of gene expression during cell cycle, or the sequential activation of brain areas during cognitive decision making, of which the state of a variable at one time point is determined by the states of a set of variables at previous time points. Models describing a stochastic *temporal* processes can be naturally represented as *dynamic Bayesian networks* (DBN). Let $\boldsymbol{X}^t := (X_1^t, \ldots, X_p^t)^\top \in \mathbb{R}^p$ be an observation vector at time t. A stochastic dynamic process can be modeled by a "first-order Markovian transition model" $p(\boldsymbol{X}^t|\boldsymbol{X}^{t-1})$, which defines the probabilistic distribution at time t given those at time $t-1$. Under this assumption, the likelihood of a time series of T steps can be expressed as:

$$p(\boldsymbol{X}^1, \ldots, \boldsymbol{X}^T) = p(\boldsymbol{X}^1) \prod_{t=2}^{T} p(\boldsymbol{X}^t|\boldsymbol{X}^{t-1}) = p(\boldsymbol{X}^1) \prod_{t=2}^{T} \prod_{i=1}^{p} p(X_i^t|\boldsymbol{X}_{\pi_i}^{t-1}), \tag{32}$$

where we assume that the topology of the networks is specified by a set of regulatory relations $\boldsymbol{X}_{\pi_i}^{t-1} := \{X_j^{t-1} : X_j^{t-1} \text{ regulates } X_i^t\}$, and hence the transition model $p(\boldsymbol{X}^t|\boldsymbol{X}^{t-1})$ factors over individual nodes, *i.e.*, $\prod_i p(X_i^t|\boldsymbol{X}_{\pi_i}^{t-1})$.

A simple form of the transition model $p(\boldsymbol{X}^t|\boldsymbol{X}^{t-1})$ in a DBN is a *linear dynamics model*:

$$\boldsymbol{X}^t = \boldsymbol{A} \cdot \boldsymbol{X}^{t-1} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I}), \tag{33}$$

where $\boldsymbol{A} \in \mathbb{R}^{p \times p}$ is a matrix of coefficients relating the expressions at time $t-1$ to those of the next time point, and $\boldsymbol{\epsilon}$ is a vector of isotropic zero mean Gaussian noise with variance σ^2. In this case, the gate function that defines the conditional distribution $p(X_i^t|\boldsymbol{X}_{\pi_i}^{t-1})$ can be expressed as a univariate Gaussian, *i.e.*, $p(X_i^t|\boldsymbol{X}_{\pi_i}^{t-1}) = \mathcal{N}(X_i^t; \boldsymbol{A}_{i\cdot}\boldsymbol{X}^{t-1}, \sigma^2)$, where $\boldsymbol{A}_{i\cdot}$ denotes the i^{th} row of the matrix $\boldsymbol{A}$. This model is also known as an *auto-regressive* model.

The major reason for favoring DBNs over standard Bayesian networks (BN) or undirected graphical models is its enhanced semantic interpretability. An edge in a BN does not necessarily imply causality due to the *Markov equivalence* of different edge configurations in the network. In DBNs (of the type defined above), all directed edges only point from time $t-1$ to t, which bear a natural causal implication and are more likely to suggest regulatory relations. The auto-regressive model in (33) also offers an elegant formal framework for consistent estimation of the structures of DBNs; we can read off the edges between variables in $\boldsymbol{X}^{t-1}$ and $\boldsymbol{X}^t$ by simply identifying the nonzero entries in the transition matrix $\boldsymbol{A}$. For example, the non-zero entries of $\boldsymbol{A}_{i\cdot}$ represent the set of regulator $\boldsymbol{X}_{\pi_i}$ that directly lead to a response on X_i.

Contrary to the name of dynamic Bayesian networks may suggest, DBNs are *time-invariant* models and the underlying network structures do *not* change over time. That is, the dependencies between variables in $\boldsymbol{X}^{t-1}$ and $\boldsymbol{X}^t$ are fixed, and both $p(\boldsymbol{X}^t|\boldsymbol{X}^{t-1})$ and $\boldsymbol{A}$ are invariant over time. The term "dynamic" only means that the DBN can model dynamical systems. In the sequel, we will present a new formalism where the structures of DBNs are time-varying rather than invariant.

Formally, let graph $\mathcal{G}^t = (\mathcal{V}, \mathcal{E}^t)$ represents the conditional independence relations between the components of random vectors $\boldsymbol{X}^{t-1}$ and $\boldsymbol{X}^t$. The vertex set $\mathcal{V}$ is a common set of variables underlying $\boldsymbol{X}^{1:T}$, *i.e.*, each node in $\mathcal{V}$ corresponds to a sequence of variables $X_i^{1:T}$. The edge set $\mathcal{E}^t \subseteq \mathcal{V} \times \mathcal{V}$ contains directed edges from components of $\boldsymbol{X}^{t-1}$ to those of $\boldsymbol{X}^t$; an edge $(i,j) \notin \mathcal{E}^t$ if and only if X_i^t is conditionally independent of X_j^{t-1} given the rest of the variables in the model. Due to the time-varying nature of the networks, the transition model $p^t(\boldsymbol{X}^t|\boldsymbol{X}^{t-1})$ in (32) becomes time dependent. In the case of the auto-regressive DBN in (33), its time-varying extension becomes:

$$\boldsymbol{X}^t = \boldsymbol{A}^t \cdot \boldsymbol{X}^{t-1} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I}), \tag{34}$$

and our goal is to estimate the non-zero entries in the sequence of time dependent transition matrices $\{\boldsymbol{A}^t\}$ $(t = 1 \dots T)$. The directed edges $\mathcal{E}^t := \mathcal{E}^t(\boldsymbol{A}^t)$ in network $\mathcal{G}^t$ associated with each $\boldsymbol{A}^t$ can be recovered via $\mathcal{E}^t = \big\{(i,j) \in \mathcal{V} \times \mathcal{V} \mid i \neq j, \boldsymbol{A}_{ij}^t \neq 0\big\}$.

9.2 Estimation Procedure

We design a procedure that decomposes the problem of estimating the time-varying networks along two orthogonal axes. The first axis is along the time, where we estimate the network for each time point separately by reweighting the observations accordingly; and the second axis is along the set of genes, where we estimate the neighborhood for each gene separately and then join these neighborhoods to form the overall network. One benefit of such decomposition is that the estimation problem is reduced to a set of atomic optimizations, one for each node i $(i = 1 \dots |\mathcal{V}|)$ at each time point t^* $(t^* = 1 \dots T)$:

$$\hat{\boldsymbol{A}}_{i\cdot}^{t^*} = \underset{\boldsymbol{A}_{i\cdot}^{t^*} \in \mathbb{R}^{1\times n}}{\operatorname{argmin}} \frac{1}{T} \sum\nolimits_{t=1}^{T} w^{t^*}(t)(x_i^t - \boldsymbol{A}_{i\cdot}^{t^*} \boldsymbol{x}^{t-1})^2 + \lambda \left\| \boldsymbol{A}_{i\cdot}^{t^*} \right\|_1, \tag{35}$$

where λ is a parameter for the ℓ_1-regularization term, which controls the number of non-zero entries in the estimated $\hat{\boldsymbol{A}}_{i\cdot}^{t^*}$, and hence the sparsity of the networks; $w^{t^*}(t)$ is the weighting of an observation from time t when we estimate the network at time t^*. More specifically, it is defined as $w^{t^*}(t) = \frac{K_h(t-t^*)}{\sum_{t=1}^{T} K_h(t-t^*)}$, where $K_h(\cdot) = K(\frac{\cdot}{h})$ is a symmetric nonnegative kernel function and h is the kernel bandwidth. Note that multiple measurements at the same time point are considered as *i.i.d.* observations and can be trivially handled by assigning them the same weights.

We have the following result for the estimation procedure.

Theorem 9. *Assume that the conditions below hold:*

1. *Elements of $\boldsymbol{A}^t$ are smooth functions with bounded second derivatives, i.e. there exists a constant $L > 0$ s.t. $|\frac{\partial}{\partial t}\boldsymbol{A}^t_{ij}| < L$ and $|\frac{\partial^2}{\partial t^2}\boldsymbol{A}^t_{ij}| < L$.*

2. *The minimum absolute value of non-zero elements of $\boldsymbol{A}^t$ is bounded away from zero at observation points, and this bound tends to zero as we observe more and more samples, i.e., $a_{\min} := \min_{t\in\{1/T,2/T,\ldots,1\}} \min_{i\in[p],j\in S_i^t} |A^t_{ij}| > 0$.*

3. *Let $\boldsymbol{\Sigma}^t = \mathbb{E}[\boldsymbol{X}^t(\boldsymbol{X}^t)^T] = [\sigma_{ij}(t)]_{i,j=1}^p$ and let S_i^t denote the set of non-zero elements of the i-th row of the matrix $\boldsymbol{A}^t$, i.e. $S_i^t = \{j \in [p] \ : \ \boldsymbol{A}^t_{ij} \neq 0\}$. Assume that there exist a constant $d \in (0, 1]$ s.t. $\max_{j\in S_i^t, k\neq j} |\sigma_{jk}(t)| \leq \frac{d}{s}, \forall i \in [p], t \in [0, 1]$, where s is an upper bound on the number of non-zero elements, i.e. $s = \max_{t\in[0,1]} \max_{i\in[p]} |S_i^t|$.*

4. *The kernel $K(\cdot) : \mathbb{R} \mapsto \mathbb{R}$ is a symmetric function and has bounded support on $[0, 1]$. There exists a constant M_K s.t. $\max_{x\in\mathbb{R}} |K(x)| \leq M_K$ and $\max_{x\in\mathbb{R}} K(x)^2 \leq M_K$.*

Let the regularization parameter scale as $\lambda = \mathcal{O}(\sqrt{(\log p)/Th})$, the minimum absolute non-zero entry $a_{\min}$ of $\boldsymbol{A}^{t^*}$ be sufficiently large ($a_{\min} \geq 2\lambda$). If $h = \mathcal{O}(T^{1/3})$ and $\frac{s\log p}{Th} = o(1)$ then

$$\mathbb{P}[\text{supp}(\hat{\boldsymbol{A}}^{t^*}) = \text{supp}(\boldsymbol{A}^{t^*})] \to 1, \quad T \to \infty, \quad \forall t^* \in [0, 1]. \tag{36}$$

9.3 Empirical Results

We have applied the formalism of TV-DBNs to reverse engineer the time varying gene regulatory networks from time series of gene expression measured across two yeast cell cycles and to explore the interactions between brain regions in response to visual stimuli. Results are reported in Song et al. [2009a].

9.4 Future Work

Formalism of dynamic Bayesian Networks is very powerful in modeling Granger causality. We plan to investigate how TV-DBN can be used to estimate Granger causality from locally stationary processes. Furthermore, we plan to develop more sound theory behind the TV-DBN model.

10 Estimation From Data with Missing Values (proposed work)

In practice, we often have to analyze data that contains missing values [Little and Rubin, 1987]. Missing values may occur due to a number of reasons, for example, faulty machinery

that collects data, subjects not being available in subsequent experiments (longitudinal studies), limits from experimental design, etc. When missing values are present, they are usually imputed to obtain a complete data set on which standard methods can be applied. However, statistical inference is the goal of our analysis and direct methods that do not impute missing values as preferred. A systematic approach to missing values problem is based on likelihoods. However, with an arbitrary pattern of missing values, no explicit maximization of the likelihood is possible even for the mean values and covariance matrices. Expectation maximization algorithms, which are iterative methods, are commonly used in cases where explicit maximization of the likelihood is not possible, however, providing theoretical guarantees for such procedures is difficult. This approach was employed in Städler and Bühlmann [2009] to estimate sparse inverse covariance matrices, which we will review in the following section. In a recent work, Lounici [2012] deals with estimation of covariance matrices from data with missing values under the assumption that the true covariance matrix is approximately low rank. Loh and Wainwright [2011] recently studied high-dimensional regression problems when data contains missing values. Casting the estimation of a precision matrix as a sequence of regression problems, they obtain an estimator of the precision matrix without maximizing partially observed likelihood function using an EM algorithm.

We plan to develop a simple, principled method that directly estimates a large dimensional covariance matrix from data with missing values. We would form an unbiased estimator of the covariance matrix from available data, which is then plugged into the penalized maximum likelihood objective for a multivariate Normal distribution to obtain a sparse estimator of the inverse of the covariance matrix, called the precision matrix. Even though the initial estimator of the covariance matrix is not necessarily positive-definite, we would show that the final estimator of the precision matrix is positive definite.

We form a sample covariance matrix from the available samples containing missing values as follows. Let $\hat{\mathbf{S}} = [\hat{\sigma}_{ab}]_{ab}$ be the sample covariance matrix with elements

$$\hat{\sigma}_{ab} = \frac{\sum_{i=1}^{n} r_{ia} r_{ib} (x_{ia} - \hat{\mu}_a)(x_{ib} - \hat{\mu}_b)}{\sum_{i=1}^{n} r_{ia} r_{ib}} \tag{37}$$

where $\hat{\mu} = (\hat{\mu}_a)$ is the sample mean defined as $\hat{\mu}_a = (\sum_{i=1}^{n} r_{ia})^{-1} \sum_{i=1}^{n} r_{ia} x_{ia}$. Under the MCAR assumption, it is simple to show that $\hat{\mathbf{S}}$ is an unbiased estimator of $\mathbf{\Sigma}$, that is, $\mathbb{E}[\hat{\mathbf{S}}] = \mathbf{\Sigma}$.

Our estimator is formed by plugging $\hat{\mathbf{S}}$ into the objective

$$\hat{\mathbf{\Omega}} = \arg\max_{\mathbf{\Omega} \succeq 0} \{\log |\mathbf{\Omega}| - \operatorname{tr} \mathbf{\Omega} \hat{\mathbf{S}} - \lambda ||\mathbf{\Omega}^{-}||_1\}, \tag{38}$$

where $\mathbf{\Omega}^{-} := \mathbf{\Omega} - \operatorname{diag}(\mathbf{\Omega})$ and $||\mathbf{A}||_1 = \sum_{ij} |A_{ij}|$.

We have proposed a simple estimator for the precision matrix in high-dimensions from data with missing values. The estimator is based on a convex program that can be solved efficiently. Furthermore, the estimator does not require imputation of the missing values and can be found using existing numerical procedures. As such, we believe that it represents a viable alternative to the iterative EM algorithm.

There are two directions in which this work should be extended. First, the MCAR assumption is very strong and it is hard to check whether it holds in practice. However, we have observed in our simulation studies that under the MAR assumption, which is a more realistic assumption than MCAR, performance of the estimators does not degrade dramatically when estimating the support of the precision matrix. However, estimated parameters are quite far from the true parameters measured using the KL divergence. This could be improved by using a weighted estimator for the sample covariance matrix [see, e.g., Robins et al., 1994]. Second, it is important to establish sharp lower bounds for the estimation problem from data with missing values, which should reflect dependence on the proportion of observed entries γ [see, e.g., Lounici, 2012].

11 Estimation of Networks From Multi-attribute Data (proposed work)

In this chapter, we consider estimation of networks when for each node we have measurement of multiple attributes. Using standard methods for estimation of networks from nodal observations, we can estimate a network for each attribute individually. However, it is not clear how to combine estimated networks to obtain a single network reflecting the structure underlying a complex system.

Katenka and Kolaczyk [2011] study estimation of association networks from multi-attribute data using canonical correlation. If the canonical correlation is large enough, then a link between two nodes is formed. We will focus instead on estimation of partial correlation between different nodes.

Let $\mathbf{X}_1 \in \mathbb{R}^{k_1}, \ldots, \mathbf{X}_p \in \mathbb{R}^{k_p}$ be random vectors that jointly follow the following distribution

$$\begin{pmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_p \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \end{pmatrix}, \underbrace{\begin{pmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} & \cdots & \mathbf{\Sigma}_{1p} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} & \cdots & \mathbf{\Sigma}_{2p} \\ \vdots & & \ddots & \vdots \\ \mathbf{\Sigma}_{p1} & \cdots & & \mathbf{\Sigma}_{pp} \end{pmatrix}}_{\mathbf{\Sigma}} \right). \tag{39}$$

Each node in a graph is associated with the random vector and an edge between two nodes a and b exists if

$$\max_{\mathbf{u},\mathbf{v}} \ \mathrm{Corr}(\mathbf{u}'\mathbf{X}_a, \mathbf{v}'\mathbf{X}_b \mid \{\mathbf{X}_c \mid c \in [p] \backslash \{a, b\}\}) \neq 0. \tag{40}$$

A straight forward calculations shows that

$$\max_{\mathbf{u},\mathbf{v}} \ \mathrm{Cov}(\mathbf{u}'\mathbf{X}_a, \mathbf{v}'\mathbf{X}_b \mid \{\mathbf{X}_c \mid c \in [p] \backslash \{a, b\}\}) = \max_{\mathbf{u},\mathbf{v}} \ \mathbf{u}'\mathbf{\Omega}_{ab}\mathbf{v}$$

where $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$. Using standard results on the multivariate Normal distribution [Lauritzen,

1996], this shows that an edge between node a and node b exists if $\mathbf{X}_a$ and $\mathbf{X}_b$ are conditionally independent given all the remaining variables.

The above discussion naturally leads to the following estimation procedure

$$\min_{\mathbf{\Omega}} \ \operatorname{tr} \mathbf{S}\mathbf{\Omega} - \log |\mathbf{\Omega}| + \lambda_1 \sum_{a \neq b} ||\mathbf{\Omega}_{ab}||_2 \tag{41}$$

where $\mathbf{S}$ is the sample covariance matrix.

Alternatively, we can estimate an edge between nodes by relating the precision matrix to the coefficients of a regression. Let us partition the covariance matrix as

$$\mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Sigma}_{11} & \tilde{\mathbf{\Sigma}}_{12} \\ \tilde{\mathbf{\Sigma}}_{21} & \tilde{\mathbf{\Sigma}}_{22} \end{pmatrix}.$$

Then using formula for the inverse of a block matrix we have that

$$\mathbf{\Omega} = \begin{pmatrix} (\mathbf{\Sigma}_{11} - \tilde{\mathbf{\Sigma}}_{12}\tilde{\mathbf{\Sigma}}_{22}^{-1}\tilde{\mathbf{\Sigma}}_{21})^{-1} & -(\mathbf{\Sigma}_{11} - \tilde{\mathbf{\Sigma}}_{12}\tilde{\mathbf{\Sigma}}_{22}^{-1}\tilde{\mathbf{\Sigma}}_{21})^{-1}\tilde{\mathbf{\Sigma}}_{12}\tilde{\mathbf{\Sigma}}_{22}^{-1} \\ -\tilde{\mathbf{\Sigma}}_{22}^{-1}\tilde{\mathbf{\Sigma}}_{21}(\mathbf{\Sigma}_{11} - \tilde{\mathbf{\Sigma}}_{12}\tilde{\mathbf{\Sigma}}_{22}^{-1}\tilde{\mathbf{\Sigma}}_{21})^{-1} & (\tilde{\mathbf{\Sigma}}_{22} - \tilde{\mathbf{\Sigma}}_{21}\mathbf{\Sigma}_{11}^{-1}\tilde{\mathbf{\Sigma}}_{12})^{-1} \end{pmatrix}.$$

Recall that

$$\mathbf{X}_1 \mid \mathbf{X}_2 = \mathbf{x}_2, \ldots, \mathbf{X}_p = \mathbf{x}_p \sim \mathcal{N}(\tilde{\mathbf{\Sigma}}_{12}\tilde{\mathbf{\Sigma}}_{22}(\mathbf{x}_2', \ldots, \mathbf{x}_p')', \mathbf{\Sigma}_{11} - \tilde{\mathbf{\Sigma}}_{12}\tilde{\mathbf{\Sigma}}_{22}^{-1}\tilde{\mathbf{\Sigma}}_{21}),$$

which shows that estimating whether $\mathbf{\Omega}_{1b}$ is equal to zero or not is equivalent to estimating whether a corresponding block in $\tilde{\mathbf{\Sigma}}_{12}\tilde{\mathbf{\Sigma}}_{22}$ is zero or not.

We form the following regression problems

$$\arg\min \sum_{i \in [n]} (x_{i,1k} - \sum_{j=2}^{p} \mathbf{x}_{i,j}\boldsymbol{\beta}_{k,j})^2 + \lambda_1 \sum_{j=2}^{p} ||\boldsymbol{\beta}_{k,j}||_2, \quad k = 1, \ldots, k_1$$

and estimate an edge between nodes 1 and b if $\boldsymbol{\beta}_{k,b} = 0$ for all $k = 1, \ldots, k_1$.

Part II

12 Multi-task learning

12.1 Literature Survey

Multi-task learning has been an active research area for more than a decade [Baxter, 1995, Thrun and O'Sullivan, 1996, Caruana, 1997]. For an estimation procedure to benefit from multiple tasks, there need to be some connections between the tasks. One common assumption is that tasks share the feature structure. Along this direction, researchers have proposed

to select relevant variables that are predictive for all tasks [Turlach et al., 2005, Zou and Yuan, 2008, Zhang, 2006, Negahban and Wainwright, 2009, Obozinski et al., 2011, Lounici et al., 2009, Liu et al., 2009, Lounici et al., 2010, Kim et al., 2009] or to learn transformation of the original variables so that in the transformed space only few features are relevant [Argyriou et al., 2007, Argyriou et al., 2008, Obozinski et al., 2010].

The model in (1) under the joint sparsity assumption was analyzed in, for example, Obozinski et al. [2011], Lounici et al. [2009], Negahban and Wainwright [2009], Lounici et al. [2010] and Kolar and Xing [2010b]. Obozinski et al. [2011] propose to minimize the penalized least squares objective with a mixed $(2,1)$-norm on the coefficients as the penalty term. The authors focus on consistent estimation of the support set S, albeit under the assumption that the number of tasks k is fixed. Negahban and Wainwright [2009] use the mixed $(\infty,1)$-norm to penalize the coefficients and focus on the exact recovery of the non-zero pattern of the regression coefficients, rather than the support set S. For a rather limited case of $k=2$, the authors show that when the regression do not share a common support, it may be harmful to consider the regression problems jointly using the mixed $(\infty,1)$-norm penalty. In Lounici et al. [2009] and Lounici et al. [2010], the focus is shifted from the consistent selection to benefits of the joint estimation for the prediction accuracy and consistent estimation.

13 Multi-Normal Means Model (completed)

13.1 Motivation

Despite many previous investigations, the theory of variable selection in multi-task regression models is far from settled. A simple clear picture of when sharing between tasks actually improves performance has not emerged. In particular, to the best of our knowledge, there has been no previous work that sharply characterizes the performance of different penalization schemes on the problem of selecting the relevant variables in the multi-task setting.

In this chapter we study multi-task learning in the context of the *many Normal means model*. This is a simplified model that is often useful for studying the theoretical properties of statistical procedures. The use of the many Normal means model is fairly common in statistics but appears to be less common in machine learning. Our results provide a sharp characterization of the sparsity patterns under which the Lasso procedure performs better than the group Lasso. Similarly, our results characterize how the group Lasso (with the mixed $(2,1)$ norm) can perform better when each non-zero row is dense.

13.2 Model

We consider the following Normal means model. Let

$$Y_{ij} \sim \begin{cases} (1-\epsilon)\mathcal{N}(0,\sigma^2) + \epsilon\mathcal{N}(\mu_{ij},\sigma^2) & j \in [k], \quad i \in S \\ N(0,\sigma^2) & j \in [k], \quad i \in S^c \end{cases} \tag{42}$$

where $(\mu_{ij})_{i,j}$ are unknown real numbers, $\sigma = \sigma_0/\sqrt{n}$ is the variance with $\sigma_0 > 0$ known, $(Y_{ij})_{i,j}$ are random observations, $\epsilon \in [0,1]$ is the parameter that controls the sparsity of features across tasks and $S \subset [p]$ is the set of relevant features. Let $s = |S|$ denote the number of relevant features. Denote the matrix $M \in \mathbb{R}^{p \times k}$ of means

	Tasks			
	1	2	...	k
1	μ_{11}	μ_{12}	...	μ_{1k}
2	μ_{21}	μ_{22}	...	μ_{2k}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
p	μ_{p1}	μ_{p2}	...	μ_{pk}

and let $\boldsymbol{\theta}_i = (\mu_{ij})_{j \in [k]}$ denote the i-th row of the matrix M. The set $S^c = [p] \backslash S$ indexes the zero rows of the matrix M and the associated observations are distributed according to the Normal distribution with zero mean and variance σ^2. The rows indexed by S are non-zero and the corresponding observation are coming from a mixture of two Normal distributions. The parameter ϵ determines the proportion of observations coming from a Normal distribution with non-zero mean. The reader should regard each column as one vector of parameters that we want to estimate. The question is whether sharing across columns improves the estimation performance.

It is known from the work on the Lasso that in regression problems, the design matrix needs to satisfy certain conditions in order for the Lasso to correctly identify the support S [see van de Geer and Bühlmann, 2009, for an extensive discussion on the different conditions]. These regularity conditions are essentially unavoidable. However, the Normal means model (42) allows us to analyze the estimation procedure in (22) and focus on the scaling of the important parameters $(n, k, p, s, \epsilon, \mu_{\min})$ for the success of the support recovery. Using the model (42) and the estimation procedure in (22), we are able to identify regimes in which estimating the support is more efficient using the ordinary Lasso than with the multi-task Lasso and vice versa. Our results suggest that the multi-task Lasso does not outperform the ordinary Lasso when the features are not considerably shared across tasks; thus, practitioners should be careful when applying the multi-task Lasso without knowledge of the task structure.

An alternative representation of the model is

$$Y_{ij} = \begin{cases} \mathcal{N}(\xi_{ij}\mu_{ij}, \sigma^2) & j \in [k], \quad i \in S \\ N(0, \sigma^2) & j \in [k], \quad i \in S^c \end{cases}$$

where ξ_{ij} is a Bernoulli random variable with success probability ϵ. Throughout the paper, we will set $\epsilon = k^{-\beta}$ for some parameter $\beta \in [0,1)$; $\beta < 1/2$ corresponds to dense rows and $\beta > 1/2$ corresponds to sparse rows. Let $\mu_{\min}$ denote the following quantity $\mu_{\min} = \min |\mu_{ij}|$.

Under the model (42), we analyze penalized least squares procedures of the form

$$\hat{\boldsymbol{\mu}} = \underset{\boldsymbol{\mu} \in \mathbb{R}^{p \times k}}{\operatorname{argmin}} \ \frac{1}{2}||\mathbf{Y} - \boldsymbol{\mu}||_F^2 + \operatorname{pen}(\boldsymbol{\mu}) \tag{43}$$

where $||A||_F = \sum_{jk} A_{jk}^2$ is the Frobenious norm, $\text{pen}(\cdot)$ is a penalty function and $\boldsymbol{\mu}$ is a $p \times k$ matrix of means. We consider the following penalties:

1. the ℓ_1 penalty
$$\text{pen}(\boldsymbol{\mu}) = \lambda \sum_{i \in [p]} \sum_{j \in [k]} |\mu_{ij}|,$$
which corresponds to the Lasso procedure applied on each task independently, and denote the resulting estimate as $\hat{\boldsymbol{\mu}}^{\ell_1}$

2. the mixed $(2,1)$-norm penalty
$$\text{pen}(\boldsymbol{\mu}) = \lambda \sum_{i \in [p]} ||\boldsymbol{\theta}_i||_2,$$
which corresponds to the multi-task Lasso formulation in Obozinski et al. [2011] and Lounici et al. [2009], and denote the resulting estimate as $\hat{\boldsymbol{\mu}}^{\ell_1/\ell_2}$

3. the mixed $(\infty,1)$-norm penalty
$$\text{pen}(\boldsymbol{\mu}) = \lambda \sum_{i \in [p]} ||\boldsymbol{\theta}_i||_\infty,$$
which correspond to the multi-task Lasso formulation in Negahban and Wainwright [2009], and denote the resulting estimate as $\hat{\boldsymbol{\mu}}^{\ell_1/\ell_\infty}$.

For any solution $\hat{\boldsymbol{\mu}}$ of (43), let $S(\hat{\boldsymbol{\mu}})$ denote the set of estimated non-zero rows
$$S(\hat{\boldsymbol{\mu}}) = \{i \in [p] \ : \ ||\hat{\boldsymbol{\theta}}_i||_2 \neq 0\}.$$
We establish sufficient conditions under which $\mathbb{P}[S(\hat{\boldsymbol{\mu}}) \neq S] \leq \alpha$ for different methods. These results are complemented with necessary conditions for the recovery of the support set S.

13.3 Overview of the Main Results

Using the normal means model we can establish the following results.

1. We establish a lower bound on the parameter $\mu_{\min}$ as a function of the parameters (n, k, p, s, β). Our result can be interpreted as follows: for any estimation procedure there exists a model given by (42) with non-zero elements equal to $\mu_{\min}$ such that the estimation procedure will make an error when identifying the set S with probability bounded away from zero.

2. We establish the sufficient conditions on the signal strength $\mu_{\min}$ for the Lasso and both variants of the group Lasso under which these procedures can correctly identify the set of non-zero rows S.

By comparing the lower bounds with the sufficient conditions, we are able to identify regimes in which each procedure is optimal for the problem of identifying the set of non-zero rows S. Furthermore, we point out that the usage of the popular group Lasso with the mixed $(\infty, 1)$ norm can be disastrous when features are not perfectly shared among tasks. This is further demonstrated through an empirical study.

14 Feature Screening With Forward regression (in progress)

14.1 Motivation

Multiple output ultra-high dimensional regression problems commonly arise in a genome-wide association mapping studies. These studies aim to find a small set of causal single-nucleotide polymorphisms (SNP) (*variables*) that account for genetic variations of a large number of genes (*regression outputs*). However, this is a very challenging problem for current statistical methods since the number of variables is likely to reach millions. Genes in a biological pathway are co-expressed as a module and it is often assumed that a causal SNP affects multiple genes in one pathway, but not all of the genes in the pathway. In order to effectively reduce the dimensionality of the problem and to detect the causal SNPs, it is very important to look at how SNPs affect all genes in a biological pathway. Since the experimentally collected data is usually very noisy, regressing genes individually onto SNPs may not be sufficient to identify the relevant SNPs that are only weakly correlated with each gene. However, once the whole biological pathway is examined, it is much easier to find the causal SNPs. In this paper, we demonstrate that the Simultaneous Orthogonal Matching Pursuit (S-OMP) [Tropp et al., 2006] can be used to quickly reduce the dimensionality of the problem, without losing any of the relevant variables.

As the dimensionality of the problem and the number of outputs increase, it becomes computationally hard to solve the commonly used convex programs used to identify relevant variables in multiple output regression problems. Previous work Liu et al. [2009], Lounici et al. [2009], Kim et al. [2009], do not scale well to settings when the number of variables exceeds $\gtrsim 10000$ and the number of outputs exceeds $\gtrsim 1000$ as in genome-wide association studies. Furthermore, estimation error of the regression coefficients depends on the number of variables in the problem, so that the variable selection can improve convergence rates of estimation procedures. These concerns motivate us to propose and study the S-OMP as a fast way to remove many of the irrelevant variables.

14.2 Model

Consider the following model

$$\mathbf{y}_t = \mathbf{X}_t \boldsymbol{\beta}_t + \boldsymbol{\epsilon}_t, \quad t = 1, \ldots, T. \tag{44}$$

Assume that for all $t \in [T]$, $\mathbf{X}_t \in \mathbb{R}^{n \times p}$. For the design $\mathbf{X}_t$, we denote $\mathbf{X}_{t,j}$ the j-th column, $\mathbf{x}_{t,i}$ the i-th row and $x_{t,ij}$ the element at (i,j). Denote $\mathbf{\Sigma}_t = \mathrm{Cov}(\mathbf{x}_{t,i})$. Without loss of generality, we assume that $\mathrm{Var}(y_{t,i}) = 1$, $\mathbb{E}(x_{t,ij}) = 0$ and $\mathrm{Var}(x_{t,ij}) = 1$. The noise $\boldsymbol{\epsilon}_t$ is zero mean and $\mathrm{Cov}(\boldsymbol{\epsilon}_t) = \sigma^2 \mathbf{I}_{n \times n}$. We assume that the number of variables $p \gg n$ and that the regression coefficients $\boldsymbol{\beta}_t$ are jointly sparse. Let $\mathcal{M}_{*,t}$ denote the set of non-zero coefficients of $\boldsymbol{\beta}_t$ and $\mathcal{M}_* = \cup_{t=1}^T \mathcal{M}_{*,t}$ denote the set of all relevant variables. For an arbitrary set $\mathcal{M} = \{j_1, \ldots, j_k\}$, $\mathbf{X}_{t,\mathcal{M}}$ denotes the design with columns indexed by $\mathcal{M}$, $\mathbf{B}_{\mathcal{M}}$ denotes the rows of $\mathbf{B}$ indexed by $\mathcal{M}$ and $\mathbf{B}_j = (\boldsymbol{\beta}_{1,j}, \ldots, \boldsymbol{\beta}_{T,j})'$. The cardinality of the set $\mathcal{M}$ is denoted as $|\mathcal{M}|$. Let $s := |\mathcal{M}_*|$ denote the total number of relevant variables, so under the sparsity assumption we have $s < n$. For a square matrix $\mathbf{A}$, $\Lambda_{\min}(\mathbf{A})$ and $\Lambda_{\max}(\mathbf{A})$ are used to denote the minimum and the maximum eigenvalue, respectively. Lastly, we use $[p]$ to denote the set $\{1, \ldots, p\}$.

Before we continue, we give a few definitions that will facilitate the presentation of the algorithm and theoretical results.

Definition 10. *The union support recovery deals with estimation of $\mathcal{M}_*$, the set of all relevant variables.*

Definition 11. *The exact support recovery deals with estimation of $\{\mathcal{M}_{*,t}\}_{t \in [T]}$, the exact set of non-zero elements of $\mathbf{B}$.*

Definition 12. *An estimation procedure is said to have the sure screening property if it is able to find an estimator $\hat{\mathcal{M}}$ of the union support that satisfies $\mathbb{P}[\mathcal{M}_* \subseteq \hat{\mathcal{M}}] \to 1$ as $n \to \infty$.*

14.3 Estimation

The Simultaneous Orthogonal Matching Pursuit is outlined in Algorithm 2. Before describing the algorithm, we introduce some additional notation. For a model $\mathcal{M}$, let $\mathbf{H}_{t,\mathcal{M}}$ be the orthogonal projection onto $\mathrm{Span}(\mathbf{X}_{t,\mathcal{M}})$, i.e., $\mathbf{H}_{t,\mathcal{M}} = \mathbf{X}_{t,\mathcal{M}}(\mathbf{X}'_{t,\mathcal{M}}\mathbf{X}_{t,\mathcal{M}})^{-1}\mathbf{X}'_{t,\mathcal{M}}$, and define the residual sum of squares (RSS) as $\mathrm{RSS}(\mathcal{M}) = \sum_{t=1}^T \mathbf{y}'_t(\mathbf{I}_{n \times n} - \mathbf{H}_{t,\mathcal{M}})\mathbf{y}_t$.

The algorithm starts with an empty model $\mathcal{M}^{(0)} = \emptyset$. We recursively define the model $\mathcal{M}^{(k)}$ based on the model $\mathcal{M}^{(k-1)}$. The model $\mathcal{M}^{(k)}$ is obtained by adding a variable $\hat{j}_k$, which minimizes $\mathrm{RSS}(\mathcal{M}^{(k-1)} \cup j)$ over the set $[p] \backslash \mathcal{M}^{(k-1)}$, to the model $\mathcal{M}^{(k-1)}$. Repeating the algorithm for $n-1$ steps, a sequence of nested models $\{\mathcal{M}^{(k)}\}_{k=0}^{n-1}$ is obtained, with $\mathcal{M}^{(k)} = \{\hat{j}_1, \ldots, \hat{j}_k\}$.

To practically select one of the models from $\{\mathcal{M}^{(k)}\}_{k=0}^{n-1}$, we minimize the modified BIC criterion [Chen and Chen, 2008], which is defined as

$$\mathrm{BIC}(\mathcal{M}) = \log\left(\frac{\mathrm{RSS}(\mathcal{M})}{nT}\right) + \frac{|\mathcal{M}|(\log(n) + 2\log(p))}{n} \tag{45}$$

with $|\mathcal{M}|$ denoting the number of elements of the set $\mathcal{M}$. Let $\hat{s} = \mathrm{argmin}_{k \in \{0,\ldots,n-1\}} \mathrm{BIC}(\mathcal{M}^{(k)})$, so that the selected model is $\mathcal{M}^{(\hat{s})}$. Observe that $\mathcal{M}^{(\hat{s})}$ estimates only the union support and that further subselection is needed to estimate the exact support.

Algorithm 2 Group Forward Regression

Input: Dataset $\{\mathbf{X}_t, \mathbf{y}_t\}_{t=1}^T$
Output: Sequence of selected models $\{\mathcal{M}^{(k)}\}_{k=0}^{n-1}$

1: Set $\mathcal{M}^{(0)} = \emptyset$
2: **for** $k = 1$ to $n-1$ **do**
3: **for** $j = 1$ to p **do**
4: $\tilde{\mathcal{M}}_j^{(k)} = \mathcal{M}^{(k-1)} \cup \{j\}$
5: $\mathbf{H}_{t,j} = \mathbf{X}_{t,\tilde{\mathcal{M}}_j^{(k)}} (\mathbf{X}'_{t,\tilde{\mathcal{M}}_j^{(k)}} \mathbf{X}_{t,\tilde{\mathcal{M}}_j^{(k)}})^{-1} \mathbf{X}'_{t,\tilde{\mathcal{M}}_j^{(k)}}$
6: $\mathrm{RSS}(\tilde{\mathcal{M}}_j^{(k)}) = \sum_{t=1}^T \mathbf{y}'_t (\mathbf{I}_{n\times n} - \mathbf{H}_{t,j}) \mathbf{y}_t$
7: **end for**
8: $\hat{j}_k = \mathrm{argmin}_{j \in \{1,\ldots,p\} \setminus \mathcal{M}^{(k-1)}} \mathrm{RSS}(\tilde{\mathcal{M}}_j^{(k)})$
9: $\mathcal{M}^{(k)} = \mathcal{M}^{(k-1)} \cup \{\hat{j}_k\}$
10: **end for**

Remark: The S-OMP algorithm is outlined only conceptually in this section. The steps 5 and 6 of the algorithm can be implemented efficiently using the progressive Cholesky decomposition see, e.g., Cotter et al. [1999].

We have the following result for the S-OMP procedure

Theorem 13. *Assume that the suitable technical conditions (given in Kolar and Xing [2010b]) are satisfied. Furthermore, assume that*

$$\frac{n^{1-6\delta_s-6\delta_{\min}}}{\max\{\log(p), \log(T)\}} \to \infty, \ \ as\ n \to \infty. \tag{46}$$

Then there exists a number $m^*_{\max} = m^*_{\max}(n)$, so that in $m^*_{\max}$ all the relevant variables are included in the model, i.e., as $n \to \infty$

$$\begin{aligned} &\mathbb{P}[\mathcal{M}_* \subseteq \mathcal{M}^{(m^*_{\max})}] \\ &\quad \geq 1 - C_1 \exp\left(-C_2 \frac{n^{1-6\delta_s-6\delta_{\min}}}{\max\{\log p, \log T\}}\right), \end{aligned} \tag{47}$$

for some constants C_1, C_2. The exact value of $m^*_{\max}$ is given as

$$m^*_{\max} = \lfloor 2^4 \phi^{-2}_{\min} \phi_{\max} C^2_{\boldsymbol{\beta}} C^2_s c^{-2}_{\boldsymbol{\beta}} n^{2\delta_s + 2\delta_{\min}} \rfloor. \tag{48}$$

Under the assumptions of Theorem 13, $m^*_{\max} \leq n-1$, so that the procedure effectively reduces the dimensionality below the sample size. From the proof of the theorem, it is clear how multiple outputs help to identify the relevant variables. The crucial quantity in identifying all relevant variables is the minimum non-zero row norm of $\mathbf{B}$, which allows us to identify weak variables if they are relevant for a large number of outputs even though individual coefficients may be small.

14.4 Future work

In the future, we plan to investigate screening properties of forward regression under generalized linear models. A prototypical example would be selection in multi-task classification using logistic regression.

15 Marginal Regression For Multi-task Learning (proposed work)

In this section, we investigate marginal regression, also known as correlation learning, marginal learning and sure screening, which is one computationally superior alternative to the Lasso. This is a very old and simple procedure, which has recently gained popularity due to its desirable properties in high-dimensional setting [Wasserman and Roeder, 2009, Fan and Lv, 2008, Fan et al., 2009a, 2011]. See also Kerkyacharian et al. [2009] and Alquier [2008] for related procedures. Marginal regression is based on regressing the response variable on each variable separately

$$\hat{\mu}_j = (\mathbf{X}_j'\mathbf{X}_j)^{-1}\mathbf{X}_j'\mathbf{y}, \tag{49}$$

where $\mathbf{X}_j = (x_{1j}, \ldots, x_{nj})'$. Next, the values $\{|\hat{\mu}_j|\}$ are sorted in decreasing order, with $\{\hat{r}_j\}$ denoting the ranks, and the set of estimated variables is

$$\hat{S}(k) := \{1 \leq j \leq p : \hat{r}_j \leq k\}, \qquad 1 \leq k \leq p. \tag{50}$$

Note that in Eq. (49) we use the first n samples only to compute $\hat{\mu}_j$. Under a condition, related to the faithfulness conditions used in causal literature [Robins et al., 2003, Spirtes et al., 2000], it can be shown that the set $\hat{S}(k)$ correctly estimates the relevant variables $S := \{1 \leq j \leq p : \beta_j \neq 0\}$, see Wasserman and Roeder [2009].

Motivated by successful applications to variable selection in single task problems, we study properties of marginal regression in a multitask setting. We will consider the following multitask regression model

$$\mathbf{y}_t = \mathbf{X}\boldsymbol{\beta}_t + \boldsymbol{\epsilon}_t \qquad t = 1, \ldots, T \tag{51}$$

where $\mathbf{y}_t, \boldsymbol{\epsilon} \in \mathbb{R}^m$ and $\mathbf{X} \in \mathbb{R}^{m \times p}$. Again, we assume that $m = 2n$ and use half of the samples to rank the variables and the other half to select the correct number of relevant variables. The subscript t indexes tasks and $\boldsymbol{\beta}_t \in \mathbb{R}^p$ is the unknown regression coefficient for the t-th task. We assume that there is a shared design matrix $\mathbf{X}$ for all tasks, a situation that arises, for example, in genome-wide association studies. Alternatively, one can have one design matrix $\mathbf{X}_t$ for each task. We assume that the regression coefficients are jointly sparse. Let $S_t := \{1 \leq j \leq p : \beta_{tj} \neq 0\}$ be the set of relevant variables for the t-th task and let $S = \cup_t S_t$ be the set of all relevant variables. Under the joint sparsity assumption $s := |S| \ll n$.

To perform marginal regression in the multitask setting, one computes correlation between each variable and each task using the first half of the samples

$$\hat{\mu}_{tj} = (\mathbf{X}_j'\mathbf{X}_j)^{-1}\mathbf{X}_j'\mathbf{y}_t, \tag{52}$$

for each $t = 1, \ldots, T,\ j = 1, \ldots, p$. Let $\Phi : \mathbb{R}^T \mapsto \mathbb{R}_+$ be a scoring function, which is used to sort the values $\{\Phi(\{\hat{\mu}_{tj}\}_t)\}_j$ in decreasing order. Let $\{\hat{r}_{\Phi,j}\}$ denote the rank of variable j in the ordering, then the set of estimated variables is

$$\hat{S}_\Phi(k) := \{1 \le j \le p : \hat{r}_{\Phi,j} \le k\}, \qquad 1 \le k \le p. \tag{53}$$

For concreteness, we will use the norm $||\cdot||_1, ||\cdot||_2$ and $||\cdot||_\infty$ as our scoring functions and denote the sets of estimated variables $\hat{S}_{\ell_1}(k)$, $\hat{S}_{\ell_2}(k)$ and $\hat{S}_{\ell_\infty}(k)$ respectively.

16 Multi-task Learning With Sparse PCA (proposed work)

Let $\{\mathbf{y}^i\}_{i\in[n]}$ be a collection of samples in $\mathbb{R}^p$ generated as follows

$$\mathbf{y}^i = \sum_{j=1}^{r} \sqrt{\beta_j} v_j^i \mathbf{u}^j + \boldsymbol{\epsilon}^i, \quad i \in [n] \tag{54}$$

where $\{\boldsymbol{\beta}_j\}_{j\in[r]}$ is a collection of distinct, well separated positive numbers, $v_j^i \overset{iid}{\sim} \mathcal{N}(0,1)$ for all i, j, the vectors $\{\mathbf{u}^j\}_{j\in[r]}$ in $\mathbb{R}^p$ are sparse with disjoint supports and satisfy $||\mathbf{u}^j|| = 1$ and $\boldsymbol{\epsilon}^i \overset{iid}{\sim} \mathcal{N}_p(\mathbf{0}, \boldsymbol{\Gamma})$ with $\boldsymbol{\Gamma} = \sigma^2\mathbf{I}_p$ for all i. For every $j \in [r]$, define $\mathcal{J}_j = \{k \in [p] \ : \ u_k^j \neq 0\}$ and let d_j denote the number of non-zero components of $\mathbf{u}^j$, $d_j := |\mathcal{J}_j|$. Define $\mathcal{J}^C = \cap_{j\in[r]}(\mathcal{J}^j)^C$, the set of irrelevant dimensions. Let $\mathbf{s}^j = \text{sign}(\mathbf{u}^j)$ denote the sign vector associated with $\mathbf{u}^j$. For convenience, we will use $\beta_j = 0$ and $d_j = 0$ for $j > r$. Let $\mathbf{U}_1 = (\mathbf{u}^1, \ldots, \mathbf{u}^r)$ and let $\mathbf{U}_2$ be a matrix such that $\mathbf{U} := [\mathbf{U}_1\ \mathbf{U}_2]$ form a basis for $\mathbb{R}^p$. In particular, we can choose $\mathbf{U}_2$ such that, for each $j \in [r]$, there are $d_j - 1$ columns of $\mathbf{U}_2$ which are obtained by finding vectors orthogonal to $\mathbf{u}^j$ that have the same support $\mathcal{J}^j$ and the remaining $p - \sum_j d_j$ columns are chosen to have support on $\mathcal{J}^C$.

We are interested in the following optimization problem

$$\min_{\mathbf{W}\in\mathbb{R}^{p\times n}} \frac{1}{2n}||\mathbf{Y} - \mathbf{W}||_F^2 + \lambda\Theta_\nu(\mathbf{W}) \tag{55}$$

with

$$\Theta_\nu(\mathbf{W}) = \min_{\mathbf{Q}\succeq\mathbf{0},\ \mathbf{Q}\in\mathbb{R}^{p\times p}} \frac{1}{2}\,\text{tr}\,\mathbf{W}\mathbf{W}'\mathbf{Q}^{-1} + \frac{1}{2}\big[\nu\,\text{tr}\,\mathbf{Q} + (1-\nu)||\mathbf{Q}||_1\big] \tag{56}$$

and $\mathbf{Y} = (\mathbf{y}^i, \ldots, \mathbf{y}^n)$.

Matrix $\mathbf{Q}$ that minimizes (56) for $\hat{\mathbf{W}}$ is of particular interest. We plan to sharply characterize the conditions on the model (54) under which the sparsity of $\mathbf{Q}$ consistently recovers the sparsity pattern of the matrix $\mathbf{U}_1\mathbf{U}_1'$.

As shown in Bach et al. [2008], the optimization problem in (55) is related to dictionary learning and sparse matrix factorization. Furthermore, the same penalty function can be used in multi-task learning for learning structure between different tasks. For example, consider the following empirical risk minimization problem

$$\min_{\mathbf{W}\in\mathbb{R}^{p\times n}} \frac{1}{2n}||\mathbf{Y}-\mathbf{X}\mathbf{W}||_F^2 + \lambda\Theta_\nu(\mathbf{W}), \tag{57}$$

where $\mathbf{X}$ is the matrix of predictors.

17 Timeline

We summarize the completed work and provide a timeline for the future work in the following table.

Research tasks	Status and Schedule
Network estimation	
Time-varying Ising Model	[NIPS 2008, ISMB 2009, AOAS 2010]
Smoothly Varying Gaussian Graphical Models	[AISTATS 2011]
Gaussian Graphical Models with Jumps	[NIPS 2009] & Spring 2012
Conditional Covariance Estimation	[ICML 2010] & Summer 2012
Time Varying Dynamic Bayesian Networks	[NIPS 2009] & Summer 2012
Estimation from Data with Missing Values	Fall 2012
Estimation from Multi-attribute Data	Spring 2012
Multi-task Learning	
Multi-normal Means model	[JMLR 2011]
Screening with Forward Regression	[AISTATS 2010]
Marginal Regression	Fall 2012
Matrix factorization with Sparse PCA	Spring 2011
Other	
Thesis writing	Winter 2012 -

References

Pieter Abbeel, Daphne Koller, and Andrew Y. Ng. Learning factor graphs in polynomial time and sample complexity. *J. Mach. Learn. Res.*, 7:1743–1788, 2006. ISSN 1533-7928.

P. Alquier. Lasso, iterative feature selection and the correlation selector: Oracle inequalities and numerical performances. *Electronic Journal of Statistics*, 2:1129–1152, 2008.

M. Arbeitman, E. Furlong, F. Imam, E. Johnson, B. Null, B. Baker, M. Krasnow, M. Scott, R. Davis, and K. White. Gene expression during the life cycle of *Drosophila melanogaster*. *Science*, 297:2270–2275, 2002.

A. Argyriou, C.A. Micchelli, M. Pontil, and Y. Ying. A spectral regularization framework for multi-task structure learning. 2007.

Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, December 2008. doi: 10.1007/s10994-007-5040-8.

F. Bach, J. Mairal, and J. Ponce. Convex Sparse Matrix Factorizations. *ArXiv e-prints*, December 2008.

Onureena Banerjee, Laurent El Ghaoui, and Alexandre d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *J. Mach. Learn. Res.*, 9:485–516, 2008. ISSN 1533-7928.

J. Baxter. Learning internal representations. In *Proceedings of the eighth annual conference on Computational learning theory*, pages 311–320. ACM, 1995.

Peter J. Bickel and Elizaveta Levina. Covariance regularization by thresholding. *Annals of Statistics*, 36(6):2577–2604, 2008a.

Peter J. Bickel and Elizaveta Levina. Regularized estimation of large covariance matrices. *Annals of Statistics*, 36(1):199–227, 2008b.

Leo Breiman. Heuristics of instability and stabilization in model selection. *Annals of Statistics*, 24(6):2350–2383, 1996. ISSN 00905364. URL `http://www.jstor.org/stable/2242688`.

Guy Bresler, Elchanan Mossel, and Allan Sly. Reconstruction of markov random fields from samples: Some observations and algorithms. In *APPROX '08 / RANDOM '08: Proceedings of the 11th international workshop, APPROX 2008, and 12th international workshop, RANDOM 2008 on Approximation, Randomization and Combinatorial Optimization*, pages 343–356, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 978-3-540-85362-6. doi: http://dx.doi.org/10.1007/978-3-540-85363-3_28.

T. Cai, W. Liu, and X. Luo. A constrained l1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association, to appear*, 2011.

T.T. Cai, C.H. Zhang, and H.H. Zhou. Optimal rates of convergence for covariance matrix estimation. *The Annals of Statistics*, 38(4):2118–2144, 2010.

R. Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

Jiahua Chen and Zehua Chen. Extended Bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3):759–771, 2008. doi: 10.1093/biomet/asn034. URL http://biomet.oxfordjournals.org/cgi/content/abstract/95/3/759.

David Maxwell Chickering. Learning bayesian networks is np-complete. In *Learning from Data: Artificial Intelligence and Statistics V*, pages 121–130. Springer-Verlag, 1996.

C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *Information Theory, IEEE Transactions on*, 14(3):462–467, 1968. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1054142.

William S. Cleveland, Eric Grosse, and William M. Shyu. Local regression models. In John M. Chambers and Trevor J. Hastie, editors, *Statistical Models in S*, pages 309–376, 1991. ISBN 0-534-16765-9.

S.F. Cotter, R. Adler, R.D. Rao, and K. Kreutz-Delgado. Forward sequential algorithms for best basis selection. *Vision, Image and Signal Processing, IEE Proceedings* -, 146(5): 235–244, 1999. ISSN 1350-245X. doi: 10.1049/ip-vis:19990445.

Imre Csiszar and Zsolt Talata. Consistent estimation of the basic neighborhood of markov random fields. *Annals Of Statistics*, 34:123, 2006. URL doi:10.1214/009053605000000912.

A. P. Dempster. Covariance selection. *Biometrics*, 28(1):157–175, 1972. ISSN 0006341X. URL http://www.jstor.org/stable/2528966.

G. Dornhege, B. Blankertz, G. Curio, and K. Müller. Boosting bit rates in non-invasive EEG single-trial classifications by feature combination and multi-class paradigms. *IEEE Trans. Biomed. Eng.*, 51:993–1002, 2004.

J. Duchi, S. Gould, and D. Koller. Projected subgradient methods for learning sparse gaussians. In *Proceedings of the Twenty-fourth Conference on Uncertainty in AI (UAI)*, 2008.

J. Fan. Local Linear Regression Smoothers And Their Minimax Efficiencies. *The Annals of Statistics*, 21(1):196–216, 1993.

J. Fan and Q. Yao. Efficient Estimation of Conditional Variance Functions in Stochastic Regression. *Biometrika*, 85(3):645–660, 1998.

J. Fan, R. Samworth, and Y. Wu. Ultrahigh dimensional feature selection: beyond the linear model. *JMLR*, 10:2013–2038, 2009a.

J. Fan, Y. Feng, and R. Song. Nonparametric independence screening in sparse ultra-high-dimensional additive models. *JASA*, 106(495):544–557, 2011.

Jianqing Fan and Jinchi Lv. Sure independence screening for ultrahigh dimensional feature space. *Journal Of The Royal Statistical Society Series B*, 70(5):849–911, 2008. URL http://ideas.repec.org/a/bla/jorssb/v70y2008i5p849-911.html.

Jianqing Fan, Yang Feng, and Yichao Wu. Network exploration via the adaptive LASSO and SCAD penalties. *The Annals of Applied Statistics*, 3(2):521–541, 2009b. doi: 10.1214/08-AOAS215. URL http://projecteuclid.org/DPubS?service=UI&version=1.0&verb=Display&handle=euclid.aoas/1245676184.

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Department of Statistics, Stanford University, Tech. Rep*, 2008a.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostat*, 9(3):432–441, 2008b. doi: 10.1093/biostatistics/kxm045. URL http://biostatistics.oxfordjournals.org/cgi/content/abstract/9/3/432.

M. Grant and S. Boyd. Cvx: Matlab software for disciplined convex programming (web page and software), 2008. URL http://stanford.edu/~boyd/cvx.

Trevor Hastie and Robert Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 55(4):757–796, 1993. ISSN 00359246. URL http://www.jstor.org/stable/2345993.

Cho-Jui Hsieh, Matyas A. Sustik, Inderjit S. Dhillon, and Pradeep K. Ravikumar. Sparse inverse covariance matrix estimation using quadratic approximation. In *Advances in Neural Information Processing Systems 24*, pages 2330–2338. 2011.

Noureddine El Karoui. Operator norm consistent estimation of large-dimensional sparse covariance matrices. *Annals Of Statistics*, 36:2717, 2008. URL doi:10.1214/07-AOS559.

N. Katenka and E.D. Kolaczyk. Multi-attribute networks and the impact of partial information on inference and characterization. *Arxiv preprint arXiv:1109.3160*, 2011.

G. Kerkyacharian, M. Mougeot, D. Picard, and K. Tribouley. Learning out of leaders. *Multiscale, Nonlinear and Adaptive Approximation*, pages 295–324, 2009.

Seyoung Kim, Kyung-Ah Sohn, and Eric P. Xing. A multivariate regression approach to association analysis of a quantitative trait network. *Bioinformatics*, 25(12):i204–212, June 2009. doi: 10.1093/bioinformatics/btp218. URL http://bioinformatics.oxfordjournals.org/cgi/content/abstract/25/12/i204.

E.D. Kolaczyk. *Statistical analysis of network data: methods and models*. Springer Verlag, 2009.

M. Kolar and E.P. Xing. Estimating networks with jumps. *Arxiv preprint arXiv:1012.3795*, 2010a.

M. Kolar and E.P. Xing. On time varying undirected graphs. In *Proceedings of the 14th International Conference on Artifical Intelligence and Statistics AISTATS*, 2011.

M. Kolar, L. Song, A. Ahmed, and E.P. Xing. Estimating time-varying networks. *Annals of Applied Statistics (to appear)*, 2009a.

Mladen Kolar and Eric P Xing. Sparsistent estimation of Time-Varying discrete markov random fields. *0907.2337*, July 2009. URL `http://arxiv.org/abs/0907.2337`.

Mladen Kolar and Eric P. Xing. Ultra-high dimensional multiple output learning with simultaneous orthogonal matching pursuit: Screening approach. In *AISTATS 2010: Proceedings of the 13th International Conference on Artifical Intelligence and Statistics*, pages 413–420, 2010b.

Mladen Kolar, Le Song, and Eric Xing. Sparsistent learning of varying-coefficient models with structural changes. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1006–1014. 2009b.

Mladen Kolar, Ankur P. Parikh, and Eric P. Xing. On sparse nonparametric conditional covariance selection. In *ICML '10: Proceedings of the 27th Annual International Conference on Machine Learning*, 2010a.

Mladen Kolar, Le Song, Amr Ahmed, and Eric P. Xing. Estimating Time-Varying networks. *Annals of Applied Statistics*, 4(1):94—123, 2010b.

S. L. Lauritzen. *Graphical Models (Oxford Statistical Science Series)*. Oxford University Press, USA, July 1996.

R. Li and H. Liang. Variable Selection In Semiparametric Regression Modeling. *The Annals of Statistics*, 36(1):261–286, 2008.

R.J.A. Little and D.B. Rubin. *Statistical analysis with missing data*. Wiley, New York, 1987.

Han Liu, Mark Palatucci, and Jian Zhang. Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. In *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 649–656, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-516-1. doi: http://doi.acm.org/10.1145/1553374.1553458.

Po-Ling Loh and Martin J. Wainwright. High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. In *Advances in Neural Information Processing Systems 24*, pages 2726–2734. 2011.

K. Lounici. High-dimensional covariance matrix estimation with missing observations. *ArXiv e-prints*, January 2012.

Karim Lounici, Massimiliano Pontil, Alexandre B. Tsybakov, and Sara van de Geer. Taking advantage of sparsity in Multi-Task learning. In *Proceedings of the Conference on Learning Theory (COLT)*, 2009. URL `http://arxiv.org/abs/0903.1468`.

Karim Lounici, Massimiliano Pontil, Alexandre B Tsybakov, and Sara van de Geer. Oracle inequalities and optimal inference under group sparsity. *1007.1771*, July 2010. URL `http://arxiv.org/abs/1007.1771`.

H. Markowitz. Portfolio selection. *The journal of finance*, 7(1):77–91, 1952.

Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. *Annals of Statistics*, 34(3):1436–1462, 2006.

Sahand Negahban and Martin Wainwright. Phase transitions for high-dimensional joint support recovery. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1161–1168, 2009.

G. Obozinski, B. Taskar, and M.I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010.

Guillaume Obozinski, Martin J. Wainwright, and Michael I. Jordan. Support union recovery in high-dimensional multivariate regression. *The Annals of Statistics*, 39(1):1–47, February 2011. ISSN 0090-5364. doi: 10.1214/09-AOS776. URL `http://projecteuclid.org/euclid.aos/1291388368`.

Jie Peng, Pei Wang, Nengfeng Zhou, and Ji Zhu. Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association*, 104(486):735–746, 2009. doi: 10.1198/jasa.2009.0126. URL `http://pubs.amstat.org/doi/abs/10.1198/jasa.2009.0126`.

P. Ravikumar, M. J. Wainwright, G. Raskutti, and B. Yu. High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. Nov 2008.

P. Ravikumar, M. J. Wainwright, and J. D. Lafferty. High-dimensional ising model selection using ℓ_1 regularized logistic regression. *Annals of Statistics*, to appear, 2009.

Alessandro Rinaldo. Properties and refinements of the fused lasso. *The Annals of Statistics*, 37(5):2922–2952, 2009. doi: 10.1214/08-AOS665. URL `http://projecteuclid.org/DPubS?service=UI&version=1.0&verb=Display&handle=euclid.aos/1247836673`.

J. M. Robins, R. Scheines, P. Spirtes, and L. Wasserman. Uniform consistency in causal inference. *Biometrika*, 90(3):491–515, 2003.

James M. Robins, Andrea Rotnitzky, and Lue Ping Zhao. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American Statistical Association*, 89(427):pp. 846–866, 1994.

Adam J. Rothman, Peter J. Bickel, Elizaveta Levina, and Ji Zhu. Sparse permutation invariant covariance estimation. *Electronic Journal Of Statistics*, 2:494, 2008.

D. Ruppert, MP Wand, and U. Holst. Local polynomial variance-function estimation. *Technometrics*, 39(3):262–273, 1997.

L. Song, M. Kolar, and E.P. Xing. Time-varying dynamic bayesian networks. *Advances in Neural Information Processing Systems*, 22:1732–1740, 2009a.

Le Song, , Mladen Kolar, and Eric P. Xing. Keller: Estimating time-evolving interactions between genes. In *Proceedings of the 16th International Conference on Intelligent Systems for Molecular Biology*, 2009b.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, prediction, and search*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, second edition, 2000. ISBN 0-262-19440-6.

Nathan Srebro. Maximum likelihood bounded tree-width markov networks. In *UAI '01: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, pages 504–511, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1-55860-800-1.

N. Städler and P. Bühlmann. Missing values: sparse inverse covariance estimation and an extension to sparse regression. *Statistics and Computing*, pages 1–17, 2009.

S. Thrun and J. O'Sullivan. Discovering structure in multiple learning tasks: The tc algorithm. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*, pages 489–497. MORGAN KAUFMANN PUBLISHERS, INC., 1996.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58:267–288, 1996.

Joel A. Tropp, Anna C. Gilbert, and Martin J. Strauss. Algorithms for simultaneous sparse approximation. part i: Greedy pursuit. *Signal Processing*, 86(3):572 – 588, 2006. ISSN 0165-1684. doi: DOI:10.1016/j.sigpro.2005.05.030. URL `http://www.sciencedirect.com/science/article/B6V18-4GWC8JH-1/2/356d996af09dd87d495a94fba1f76a71`. Sparse Approximations in Signal and Image Processing.

P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *J. Optim. Theory Appl.*, 109(3):475–494, 2001. ISSN 0022-3239.

B.A. Turlach, W.N. Venables, and S.J. Wright. Simultaneous variable selection. *Technometrics*, 47(3):349–363, 2005. ISSN 0040-1706.

Sara A. van de Geer and Peter Bühlmann. On the conditions used to prove oracle results for the lasso. *Electronic Journal of Statistics*, 3:1360–1392, 2009. doi: 10.1214/09-EJS506.

L. Wasserman and K. Roeder. High dimensional variable selection. *Annals of statistics*, 37 (5A):2178, 2009.

W.B. Wu and M. Pourahmadi. Banding sample autocovariance matrices of stationary processes. *Statistica Sinica*, 19(4):1755–68, 2009.

Jianxin Yin, Zhi Geng, Runze Li, and Hansheng Wang. Nonparametric Covariance Model. *Statistica Sinica, Forthcoming*, 2008.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68:49–67, 2006. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.79.2062`.

Ming Yuan and Yi Lin. Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35, March 2007. doi: 10.1093/biomet/asm018. URL `http://biomet.oxfordjournals.org/cgi/content/abstract/94/1/19`.

J. Zhang. *A probabilistic framework for multitask learning*. PhD thesis, Carnegie Mellon University, 2006.

Shuheng Zhou, John Lafferty, and Larry Wasserman. Time varying undirected graphs. In Rocco A. Servedio and Tong Zhang, editors, *COLT*, pages 455–466. Omnipress, 2008.

H. Zou and M. Yuan. The f-infinity-norm support vector machine. *Stat. Sin*, 18:379–398, 2008.