Learning the Structure of Mixed Graphical Models

Jason D. Lee[*] and **Trevor J. Hastie**[†,‡]

[*]Institute of Computational and Mathematical Engineering, Stanford University.

[†]Department of Statistics, Stanford University.

[‡]An earlier version of this paper was published at AISTATS 2013. This version is significantly expanded with new experimental results, comparisons, and theoretical results.

Abstract

We consider the problem of learning the structure of a pairwise graphical model over continuous and discrete variables. We present a new pairwise model for graphical models with both continuous and discrete variables that is amenable to structure learning. In previous work, authors have considered structure learning of Gaussian graphical models and structure learning of discrete models. Our approach is a natural generalization of these two lines of work to the mixed case. The penalization scheme involves a novel symmetric use of the group-lasso norm and follows naturally from a particular parametrization of the model. Supplementary materials for this paper are available online.

1 Introduction

Many authors have considered the problem of learning the edge structure and parameters of sparse undirected graphical models. We will focus on using the l_1 regularizer to promote sparsity. This line of work has taken two separate paths: one for learning continuous valued data and one for learning discrete valued data. However, typical data sources contain both continuous and discrete variables: population survey data, genomics data, url-click pairs etc. For genomics data, in addition to the gene expression values, we have attributes attached to each sample such as gender, age, ethnicity etc. In this work, we consider learning mixed models with both continuous Gaussian variables and discrete categorical variables.

For only continuous variables, previous work assumes a multivariate Gaussian (Gaussian graphical) model with mean 0 and inverse covariance Θ. Θ is then estimated via the graphical lasso by minimizing the regularized negative log-likelihood $l(\Theta) + \lambda \|\Theta\|_1$. Several e cient methods for solving this can be found in Friedman et al. (2008a); Banerjee et al. (2008). Because the graphical lasso problem is computationally challenging, several authors considered methods related to the pseudolikelihood (PL) and nodewise regression (Meinshausen and Bühlmann, 2006; Friedman et al., 2010; Peng et al., 2009). For discrete

---

11 Supplementary Materials

Code. MATLAB Code that implements structure learning for the mixed graphical model.

Supplementary Material. Technical appendices on sampling from the mixed model, maximum likelihood, calibration weights, model selection consistency, and convexity.

models, previous work focuses on estimating a pairwise Markov random field of the form $p\left(y\right) \propto exp \sum_{r \le j} \phi_{rj}\left(y_r, y_i\right)$, where φ_{rj} are pairwise potentials. The maximum likelihood problem is intractable for models with a moderate to large number of variables (high-dimensional) because it requires evaluating the partition function and its derivatives. Again previous work has focused on the pseudolikelihood approach (Guo et al., 2010; Schmidt, 2010; Schmidt et al., 2008; Höfling and Tibshirani, 2009; Jalali et al., 2011; Lee et al., 2006; Ravikumar et al., 2010).

Our main contribution here is to propose a model that connects the discrete and continuous models previously discussed. The conditional distributions of this model are two widely adopted and well understood models: multiclass logistic regression and Gaussian linear regression. In addition, in the case of only discrete variables, our model is a pairwise Markov random field; in the case of only continuous variables, it is a Gaussian graphical model. Our proposed model leads to a natural scheme for structure learning that generalizes the graphical Lasso. Here the parameters occur as singletons, vectors or blocks, which we penalize using group-lasso norms, in a way that respects the symmetry in the model. Since each parameter block is of different size, we also derive a calibrated weighting scheme to penalize each edge fairly. We also discuss a conditional model (conditional random field) that allows the output variables to be mixed, which can be viewed as a multivariate response regression with mixed output variables. Similar ideas have been used to learn the covariance structure in multivariate response regression with continuous output variables Witten and Tibshirani (2009); Kim et al. (2009); Rothman et al. (2010).

In Section 2, we introduce our new mixed graphical model and discuss previous approaches to modeling mixed data. Section 3 discusses the pseudolikelihood approach to parameter estimation and connections to generalized linear models. Section 4 discusses a natural method to perform structure learning in the mixed model. Section 5 presents the calibrated regularization scheme, Section 6 discusses the consistency of the estimation procedures, and Section 7 discusses two methods for solving the optimization problem. Finally, Section 8 discusses a conditional random field extension and Section 9 presents empirical results on a census population survey dataset and synthetic experiments.

2 Mixed Graphical Model

We propose a pairwise graphical model on continuous and discrete variables. The model is a pairwise Markov random field with density $p(x, y; \Theta)$ proportional to

$$exp\left(\sum_{s=1}^{p}\sum_{t=1}^{p} -\frac{1}{2}\beta_{st}x_s x_t + \sum_{s=1}^{p}\alpha_s x_s + \sum_{s=1}^{p}\sum_{j=1}^{q}\rho_{sj}\left(y_i\right)x_s + \sum_{j=1}^{q}\sum_{r=1}^{q}\phi_{rj}\left(y_r, y_j\right)\right). \quad (1)$$

Here x_s denotes the sth of p continuous variables, and y_j the jth of q discrete variables. The joint model is parametrized by $\Theta = [\{\beta_{st}\}, \{\alpha_s\}, \{\rho_{sj}\}, \{\varphi_{rj}\}]$. The discrete y_r takes on L_r states. The model parameters are β_{st} continuous-continuous edge potential, α_s continuous node potential, $\rho_{sj}(y_j)$ continuous-discrete edge potential, and $\varphi_{rj}(y_r, y_j)$ discrete-discrete edge potential. $\rho_{sj}(y_j)$ is a function taking L_j values $\rho_{sj}(1), \ldots, \rho_{sj}(L_j)$. Similarly, $\varphi_{rj}(y_r, y_j)$ is

a bivariate function taking on $L_r \times L_j$ values. Later, we will think of $\rho_{sj}(y_j)$ as a vector of length L_j and $\varphi_{rj}(y_r, y_j)$ as a matrix of size $L_r \times L_j$.

The two most important features of this model are:

1. the conditional distributions are given by Gaussian linear regression and multiclass logistic regressions;

2. the model simplifies to a multivariate Gaussian in the case of only continuous variables and simplifies to the usual discrete pairwise Markov random field in the case of only discrete variables.

The conditional distributions of a graphical model are of critical importance. ==The absence of an edge corresponds to two variables being conditionally independent.== The conditional independence can be read o from the conditional distribution of a variable on all others. ==For example in the multivariate Gaussian model, x_s is conditionally independent of x_t i the partial correlation coefficient is 0.== The partial correlation coefficient is also the regression coefficient of x_t in the linear regression of x_s on all other variables. Thus the conditional independence structure is captured by the conditional distributions via the regression coefficient of a variable on all others. Our mixed model has the desirable property that the two type of conditional distributions are simple Gaussian linear regressions and multiclass logistic regressions. This follows from the pairwise property in the joint distribution. In more detail:

1. The conditional distribution of y_r given the rest is multinomial, with probabilities defined by a multiclass logistic regression where the covariates are the other variables x_s and $y_{\backslash r}$ (denoted collectively by z in the right-hand side):

$$p\left(y_r = k \mid y_{\backslash r}, x; \Theta\right) = \frac{exp\left(\omega_k^T z\right)}{\sum_{l=1}^{L_r} exp\left(\omega_l^T z\right)} = \frac{exp\left(\omega_{0k} + \sum_j \omega_{kj} z_j\right)}{\sum_{l=1}^{L_r} exp\left(\omega_{0l} + \sum_j \omega_{lj} + z_j\right)} \quad (2)$$

Here we use a simplified notation, which we make explicit in Section 3.1. The discrete variables are represented as dummy variables for each state, e.g. $z_j = 1\left[y_u = k\right]$, and for continuous variables $z_s = x_s$.

2. The conditional distribution of x_s given the rest is Gaussian, with a mean function defined by a linear regression with predictors $x_{\backslash s}$ and y_r.

$$\begin{aligned} E\left(x_s \mid x_{\backslash s}, y_r; \Theta\right) &= \omega^T z = \omega_0 + \sum_j z_j \omega_j \\ p\left(x_s \mid x_{\backslash r}, y_r; \Theta\right) &= \frac{1}{\sqrt{2\pi}\sigma_s} exp\left(-\frac{1}{2\sigma_s^2}\left(x_s - \omega^T z\right)^2\right). \end{aligned} \quad (3)$$

As before, the discrete variables are represented as dummy variables for each state $z_j = 1\left[y_u = k\right]$ and for continuous variables $z_s = x_s$.

The exact form of the conditional distributions (2) and (3) are given in (11) and (10) in Section 3.1, ==where the regression parameters w_j are defined in terms of the parameters Θ.==

The second important aspect of the mixed model is the two special cases of only continuous and only discrete variables.

1. Continuous variables only. The pairwise mixed model reduces to the familiar multivariate Gaussian parametrized by the symmetric positive-definite inverse covariance matrix $B = \{\beta_{st}$ and mean $\mu = B^{-1}\alpha$,

$$p(x) \propto exp\left(-\frac{1}{2}\left(x - B^{-1}\alpha\right)^{T} B \left(x - B^{-1}\alpha\right)\right).$$

2. Discrete variables only. The pairwise mixed model reduces to a pairwise discrete (second-order interaction) Markov random field,

$$p(y) \propto exp\left(\sum_{j=1}^{q}\sum_{r=1}^{q}\phi_{rj}(y_r, y_j)\right).$$

Although these are the most important aspects, we can characterize the joint distribution further. The conditional distribution of the continuous variables given the discrete follow a multivariate Gaussian distribution, $p(x|y) = \mathcal{N}\left(\mu(y), B^{-1}\right)$. Each of these Gaussian distributions share the same inverse covariance matrix B but differ in the mean parameter, since all the parameters are pairwise. By standard multivariate Gaussian calculations,

$$p(x|y) = \mathcal{N}\left(B^{-1}\gamma(y), B^{-1}\right) \quad (4)$$

$$\{\gamma(y)\}_s = \alpha_s + \sum_j \rho_{sj}(y_j) \quad (5)$$

$$p(y) \propto exp\left(\sum_{j=1}^{q}\sum_{r=1}^{j}\phi_{rj}(y_r, y_j) + \frac{1}{2}\gamma(y)^T B^{-1}\gamma(y)\right) \quad (6)$$

Thus we see that the continuous variables conditioned on the discrete are multivariate Gaussian with common covariance, but with means that depend on the value of the discrete variables. The means depend additively on the values of the discrete variables since $\{\gamma(y)\}_s = \sum_{j=1}^{r}\rho_{sj}(y_j)$. The marginal $p(y)$ has a known form, so for models with few number of discrete variables we can sample efficiently.

2.1 Related work on mixed graphical models

Lauritzen (1996) proposed a type of mixed graphical model, with the property that conditioned on discrete variables, $p(x|y) = \mathcal{N}(\mu(y), \Sigma(y))$. The homogeneous mixed graphical model enforces common covariance, $\Sigma(y) \equiv \Sigma$. Thus our proposed model is a special case of Lauritzen's mixed model with the following assumptions: common covariance, additive mean assumptions and the marginal $p(y)$ factorizes as a pairwise

discrete Markov random field. With these three assumptions, the full model simplifies to the mixed pairwise model presented. Although the full model is more general, the number of parameters scales exponentially with the number of discrete variables, and the conditional distributions are not as convenient. For each state of the discrete variables there is a mean and covariance. Consider an example with q binary variables and p continuous variables; the full model requires estimates of 2^q mean vectors and covariance matrices in p dimensions. Even if the homogeneous constraint is imposed on Lauritzen's model, there are still 2^q mean vectors for the case of binary discrete variables. The full mixed model is very complex and cannot be easily estimated from data without some additional assumptions. In comparison, the mixed pairwise model has number of parameters $O((p+q)^2)$ and allows for a natural regularization scheme which makes it appropriate for high dimensional data.

An alternative to the regularization approach that we take in this paper, is the limited-order correlation hypothesis testing method Tur and Castelo (2012). The authors develop a hypothesis test via likelihood ratios for conditional independence. However, they restrict to the case where the discrete variables are marginally independent so the maximum likelihood estimates are well-defined for $p > n$.

There is a line of work regarding parameter estimation in undirected mixed models that are decomposable: any path between two discrete variables cannot contain only continuous variables. These models allow for fast exact maximum likelihood estimation through node-wise regressions, but are only applicable when the structure is known and $n > p$ (Edwards, 2000). There is also related work on parameter learning in directed mixed graphical models. Since our primary goal is to learn the graph structure, we forgo exact parameter estimation and use the pseudolikelihood. Similar to the exact maximum likelihood in decomposable models, the pseudolikelihood can be interpreted as node-wise regressions that enforce symmetry.

After we proposed our model[1], the independent work of Cheng et al. (2013) appeared, which considers a more complicated mixed graphical model. Their model includes higher order interaction terms by allowing the covariance of the continuous variables to be a function of the categorical variables y, which results in a larger model similar to Lauritzen's model. This results in a model with $O(p^2 q + q^2)$ parameters, as opposed to $O(p^2 + q^2)$ in our proposed pairwise model. We believe that in the high-dimensional setting where data sparsity is an issue a simpler model is an advantage.

To our knowledge, this work is the first to consider convex optimization procedures for learning the edge structure in mixed graphical models.

3 Parameter Estimation: Maximum Likelihood and Pseudolikelihood

Given samples $(x_i, y_i)_{i=1}^n$, we want to find the maximum likelihood estimate of Θ. This can be done by minimizing the negative log-likelihood of the samples:

---

[1]Cheng et al. (2013), http://arxiv.org/abs/1304.2810, appeared on arXiv 11 months after our original paper was put on arXiv, http://arxiv.org/abs/1205.5012.

$$\ell\left(\Theta\right)=-\sum_{i=1}^{n}log\,p\left(x_i,y_i;\Theta\right)\quad\text{where}\quad(7)$$

$$log\,p\left(x,y;\Theta\right)=\sum_{s=1}^{p}\sum_{t=1}^{p}-\frac{1}{2}\beta_{st}x_sx_t+\sum_{s=1}^{p}\alpha_sx_s+\sum_{s=1}^{p}\sum_{j=1}^{q}\rho_{sj}\left(y_j\right)x_s+\sum_{j=1}^{q}\sum_{r=1}^{j}\phi_{rj}\left(y_r,y_j\right)-log\,Z\left(\Theta\right)\quad(8)$$

The negative log-likelihood is convex, so standard gradient-descent algorithms can be used for computing the maximum likelihood estimates. The major obstacle here is $Z(\Theta)$, which involves a high-dimensional integral. Since the pairwise mixed model includes both the discrete and continuous models as special cases, maximum likelihood estimation is at least as difficult as the two special cases, the first of which is a well-known computationally intractable problem. We defer the discussion of maximum likelihood estimation to the supplementary material.

3.1 Pseudolikelihood

The pseudolikelihood method Besag (1975) is a computationally efficient and consistent estimator formed by products of all the conditional distributions:

$$\tilde{\ell}\left(\Theta|x,y\right)=-\sum_{s\setminus1}^{p}log\,p\left(x_s|x_{=s},y;\Theta\right)-\sum_{r=1}^{q}log\,p\left(y_r|x,y_{\setminus r};\Theta\right)\quad(9)$$

The conditional distributions $p(x_s|x_{\setminus s}, y; \theta)$ and $p(y_r = k|y_{\setminus r}, x; \theta)$ take on the familiar form of linear Gaussian and (multiclass) logistic regression, as we pointed out in (2) and (3). Here are the details:

- The conditional distribution of a continuous variable x_s is Gaussian with a linear regression model for the mean, and unknown variance.

$$p\left(x_s|x_{\setminus s},y;\Theta\right)=\frac{\sqrt{\beta_{ss}}}{\sqrt{2\pi}}exp\left(\frac{-\beta_{ss}}{2}\left(\frac{\alpha_s+\sum_j\rho_{sj}\left(y_j\right)-\sum_{t\neq s}\beta_{st}x_t}{\beta_{ss}}-x_s\right)^2\right)\quad(10)$$

- The conditional distribution of a discrete variable y_r with L_r states is a multinomial distribution, as used in (multiclass) logistic regression. Whenever a discrete variable is a predictor, each of its levels contribute an additive effect; continuous variables contribute linear effects.

$$p\left(y_r|y_{\setminus r},x;\Theta\right)=\frac{exp\left(\sum_s\rho_{sr}\left(y_r\right)x_s+\phi_{rr}\left(y_r,y_r\right)+\sum_{j\neq r}\phi_{rj}\left(y_r,y_j\right)\right)}{\sum_{l=1}^{L_r}exp\left(\sum_s\rho_{sr}\left(l\right)x_s+\phi_{rr}\left(l,l\right)+\sum_{j\neq r}\phi_{rj}\left(l,y_j\right)\right)}\quad(11)$$

Taking the negative log of both gives us

$$-log\ p\left(x_s|x_{\backslash s},y;\Theta\right)=-\frac{1}{2}log\ \beta_{ss}=\frac{\beta_{ss}}{2}\left(\frac{\alpha_s}{\beta_{ss}}+\sum_j\frac{\rho_{sj}\left(y_j\right)}{\beta_{ss}}-\sum_{t\neq s}\frac{\beta_{st}}{\beta_{ss}}x_t-x_s\right)^2 \quad (12)$$

$$-log\ p\left(y_r|y_{\backslash r},x;\Theta\right)=-log\frac{exp\left(\sum_s\rho_{sr}\left(y_r\right)x_s+\phi_{rr}\left(y_r,y_r\right)+\sum_{j\neq r}\phi_{rj}\left(y_r,y_j\right)\right)}{\sum_{l=1}^{L_r}exp\left(\sum_s\rho_{sr}\left(l\right)x_s+\phi_{rr}\left(l,l\right)+\sum_{j\neq r}\phi_{rj}\left(l,y_j\right)\right)} \quad (13)$$

A generic parameter block, θ_{uv}, corresponding to an edge (u, v) appears twice in the pseudolikelihood, once for each of the conditional distributions $p(z_u|z_v)$ and $p(z_v|z_u)$.

Proposition 1. *The negative log pseudolikelihood in* (9) *is jointly convex in all the parameters* $\{\beta_{ss}, \beta_{st}, \alpha_s, \varphi_{rj}, \rho_{sj}\}$ *over the region* $\beta_{ss} > 0$.

We prove Proposition 1 in the Supplementary Materials.

3.2 Separate node-wise regression

A simple approach to parameter estimation is via separate node-wise regressions; a generalized linear model is used to estimate $p(z_s|z_{\backslash s})$ for each s. Separate regressions were used in Meinshausen and Bühlmann (2006) for the Gaussian graphical model and Ravikumar et al. (2010) for the Ising model. The method can be thought of as an asymmetric form of the pseudolikelihood since the pseudolikelihood enforces that the parameters are shared across the conditionals. Thus the number of parameters estimated in the separate regression is approximately double that of the pseudolikelihood, so we expect that the pseudolikelihood outperforms at low sample sizes and low regularization regimes. The node-wise regression was used as our baseline method since it is straightforward to extend it to the mixed model. As we predicted, the pseudolikelihood or joint procedure outperforms separate regressions; see top left box of Figures 5 and 6. Liu and Ihler (2012, 2011) confirm that the separate regressions are outperformed by pseudolikelihood in numerous synthetic settings.

Concurrent work of Yang et al. (2012, 2013) extend the separate node-wise regression model from the special cases of Gaussian and categorical regressions to generalized linear models, where the univariate conditional distribution of each node $p(x_s|x_{\backslash s})$ is specified by a generalized linear model (e.g. Poisson, categorical, Gaussian). By specifying the conditional distributions, Besag (1974) show that the joint distribution is also specified. Thus another way to justify our mixed model is to define the conditionals of a continuous variable as Gaussian linear regression and the conditionals of a categorical variable as multiple logistic regression and use the results in Besag (1974) to arrive at the joint distribution in (1). However, the neighborhood selection algorithm in Yang et al. (2012, 2013) is restricted to models of the form $p(x) \propto \exp(\sum_s \theta_s x_s + \sum_{s,t} \theta_{st} x_s x_t + \sum_s C(x_s))$. In particular, this procedure cannot be applied to edge selection in our pairwise mixed model in (1) or the categorical model in (2) with greater than 2 states. Our baseline method of separate regressions is closely related to the neighborhood selection algorithm they proposed; the baseline can be considered as a generalization of Yang et al. (2012, 2013) to allow for more general pairwise

interactions with the appropriate regularization to select edges. Unfortunately, the theoretical results in Yang et al. (2012, 2013) do not apply to the baseline nodewise regression method, nor the joint pseudolikelihood.

4 Conditional Independence and Penalty Terms

In this section, we show how to incorporate edge selection into the maximum likelihood or pseudolikelihood procedures. In the graphical representation of probability distributions, the absence of an edge $e = (u, v)$ corresponds to a conditional independency statement that variables x_u and x_v are conditionally independent given all other variables (Koller and Friedman, 2009). We would like to maximize the likelihood subject to a penalization on the number of edges since this results in a sparse graphical model. In the pairwise mixed model, there are 3 type of edges

1. β_{st} is a scalar that corresponds to an edge from x_s to x_t. $\beta_{st} = 0$ implies x_s and x_t are conditionally independent given all other variables. This parameter is in two conditional distributions, corresponding to either x_s or x_t is the response variable, $p(x_s|x_{\backslash s}, y; \Theta)$ and $p(x_t|x_{\backslash t}, y; \Theta)$.

2. ρ_{sj} is a vector of length L_j. If $\rho_{sj}(y_j) = 0$ for all values of y_j, then y_j and x_s are conditionally independent given all other variables. This parameter is in two conditional distributions, corresponding to either x_s or y_j being the response variable: $p(x_s|x_{\backslash s}, y; \Theta)$ and $p(y_j|x, y_{\backslash j}; \Theta)$.

3. φ_{rj} is a matrix of size $L_r \times L_j$. If $\varphi_{rj}(y_r, y_j) = 0$ for all values of y_r and y_j, then y_r and y_j are conditionally independent given all other variables. This parameter is in two conditional distributions, corresponding to either y_r or y_j being the response variable, $p(y_r|x, y_{\backslash r}; \Theta)$ and $p(y_j|x, y_{\backslash j}; \Theta)$.

For conditional independencies that involve discrete variables, the absence of that edge requires that the entire matrix φ_{rj} or vector ρ_{sj} is 0.[2] The form of the pairwise mixed model motivates the following regularized optimization problem

$$\underset{\Theta}{minimize} \quad \ell_\lambda (\Theta) = \ell (\Theta) = \lambda \left(\sum_{s<t} 1\left[\beta_{st} \neq 0 \right] + \sum_{sj} 1\left[\rho_{sj} \not\equiv 0 \right] + \sum_{r<j} 1\left[\phi_{rj} \not\equiv 0 \right] \right). \quad (14)$$

All parameters that correspond to the same edge are grouped in the same indicator function. This problem is non-convex, so we replace the l_0 sparsity and group sparsity penalties with the appropriate convex relaxations. For scalars, we use the absolute value (l_1 norm), for vectors we use the l_2 norm, and for matrices we use the Frobenius norm. This choice corresponds to the standard relaxation from group l_0 to group l_1/l_2 (group lasso) norm (Bach et al., 2011; Yuan and Lin, 2006),

---

[2]If $\rho_{sj}(y_j) = constant$, then x_s and y_j are also conditionally independent. However, the unpenalized term will absorb the constant, so the estimated $\rho_{sj}(y_j)$ will never be constant for $\lambda > 0$.

$$minimize_{\Theta} \quad \ell_\lambda(\Theta) = \ell(\Theta) + \lambda \left(\sum_{s=1}^{p}\sum_{t=1}^{s-1}|\beta_{st}| + \sum_{s=1}^{p}\sum_{j=1}^{q}\|\rho_{sj}\|_2 + \sum_{j=1}^{q}\sum_{r=1}^{j-1}\|\phi_{rj}\|_F \right). \quad (15)$$

5 Calibrated regularizers

In (15) each of the group penalties are treated as equals, irrespective of the size of the group. We suggest a calibration or weighting scheme to balance the load in a more equitable way. We introduce weights for each group of parameters and show how to choose the weights such that each parameter set is treated equally under p_F, the fully-factorized independence model.[3]

$$minimize_{\Theta} \quad \ell(\Theta) + \lambda \left(\sum_{t=1}^{p}\sum_{t=1}^{s-1}w_{st}|\beta_{st}| + \sum_{s=1}^{p}\sum_{j=1}^{q}w_{sj}\|\rho_{sj}\|_2 + \sum_{j=1}^{q}\sum_{r=1}^{j-1}w_{rj}\|\phi_{rj}\|_F \right) \quad (16)$$

Based on the KKT conditions (Friedman et al., 2007), the parameter group θ_g is non-zero if

$$\left\|\frac{\partial\ell}{\partial\theta_g}\right\| > \lambda w_g$$

where θ_g and w_g represents one of the parameter groups and its corresponding weight. Now $\frac{\partial\ell}{\partial\theta_g}$ can be viewed as a generalized residual, and for different groups these are different dimensions—e.g. scalar/vector/matrix. So even under the independence model (when all terms should be zero), one might expect some terms $\|\frac{\partial\ell}{\partial\theta_g}\|$ to have a better random non-zero of being non-zero (for example, those of bigger dimensions). Thus for all parameters to be on equal footing, we would like to choose the weights w_g such that

$$w_g = E_{p_F}\left\|\frac{\partial\ell}{\partial\theta_g}\right\| \times \text{constant}, \quad (17)$$

where p_F is the fully factorized (independence) model. We will refer to these as the exact weights. We do not have a closed form expression for computing them, but they can be easily estimated by a simple null-model simulation. We also propose an approximation that can be computed exactly. It is straightforward to compute $E_{p_F}\|\frac{\partial\ell}{\partial\theta_g}\|^2$ in closed form, which leads to approximate weights

$$w_g \propto \sqrt{E_{p_F}\left\|\frac{\partial\ell}{\partial\theta_g}\right\|^2} \quad (18)$$

---

[3]Under the independence model p_F is fully-factorized $p(x, y) = \prod_{s=1}^{p}p(x_s)\prod_{r=1}^{q}p(y_r)$.

In the supplementary material, we show that for the three types of edges this leads to the expressions

$$
\begin{aligned}
w_{st} &= \sigma_s \sigma_t, \\
w_{sj} &= \sigma_s \sqrt{\sum_a p_a (1 - p_a)}, \quad \text{and} \\
w_{rj} &= \sqrt{\sum_a p_a (1 - p_a) \sum_b q_b (1 - q_b)},
\end{aligned}
\tag{19}
$$

where σ_s is the standard deviation of the continuous variable x_s, $p_a = Pr(y_r = a)$ and $q_b = Pr(y_j = b)$. For all 3 types of parameters, the weight has the form of

$w_{uv} = \mathbf{tr}\left(\mathbf{cov}\left(z_u\right)\right) \mathbf{tr}\left(\mathbf{cov}\left(z_v\right)\right)$, where z represents a generic variable and $\mathbf{cov}(z)$ is the variance-covariance matrix of z.

We conducted a small simulation study to show that calibration is needed. Consider a model with 4 independent variables: 2 continuous with variance 10 and 1, and 2 discrete variables with 10 and 2 levels. There are 6 candidate edges in this model and from row 1 of Table 1 we can see the sizes of the gradients are different. In fact, the ratio of the largest gradient to the smallest gradient is greater than 4. The edges ρ_{11} and ρ_{12} involving the first continuous variable with variance 10 have larger edge weights than the corresponding edges, ρ_{21} and ρ_{22} involving the second continuous variable with variance 1. Similarly, the edges involving the first discrete variable with 10 levels are larger than the edges involving the second discrete variable with 2 levels. This reflects our intuition that larger variance and longer vectors will have larger norm.

The approximate weights from Equation (19) are a very good approximation to $\|\nabla l\|$. Since the weights are only defined up to a proportionality constant, the cosine similarity is an appropriate measure of the quality of approximation. For this simulation, the cosine similarity is

$$
sim\left(w, \|\nabla \ell\|\right) = .993,
$$

which is extremely close to 1.

Using the weights from Table 1, we conducted a second simulation to record which edge would enter first when the four variables are independent. The results are shown in Table 2. Both exact and approximate calibration perform much better than no calibration, but neither deliver the ideal 1/6th probabilities one might desire in a situation like this. Of course, calibrating to the expectations of the gradient does not guarantee equal entry probability, but it brings them much closer.

The exact weights do not have simple closed-form expressions, but they can be easily computed via Monte Carlo. This can be done by simulating independent Gaussians and multinomials with the appropriate marginal variance σ_s and marginal probabilities p_a, then approximating the expectation in (17) by an average. The computational cost of this procedure is negligible compared to fitting the mixed model, so in practice either the exact or approximate weights can be used.

6 Model Selection Consistency

In this section, we study the model selection consistency, whether the correct edge set is selected and the parameter estimates are close to the truth, of the pseudolikelihood and maximum likelihood estimators. Consistency can be established using the framework first developed in Ravikumar et al. (2010) and later extended to general m-estimators by Lee et al. (2013). Instead of stating the full results and proofs, we will illustrate the type of theorems that can be shown and defer the rigorous statements to the Supplementary Material.

First, we define some notation. Recall that Θ is the vector of parameters being estimated $\{\beta_{ss}, \beta_{st}, \alpha_s, \varphi_{rj}, \rho_{sj}\}$, Θ^* be the true parameters that estimated the model, and $Q = \nabla^2 l(\Theta^*)$. Both maximum likelihood and pseudolikelihood estimation procedures can be written as a convex optimization problem of the form

$$minimize \quad \ell(\Theta) + \lambda \sum_{g \in G} \|\Theta_g\|_2 \quad (20)$$

where $l(\theta) = \{l_{ML}, l_{PL}\}$ is one of the two log-likelihoods. The regularizer

$$\sum_{g \in G} \|\Theta_g\| = \lambda \left(\sum_{s=1}^{p} \sum_{t=1}^{s-1} |\beta_{st}| + \sum_{s=1}^{p} \sum_{j=1}^{q} \|\rho_{sj}\|_2 + \sum_{j=1}^{q} \sum_{r=1}^{j-1} \|\phi_{rj}\|_F \right).$$

The set G indexes the edges $_{st}$, ρ_{sj}, and φ_{rj}, and Θ_g is one of the three types of edges. Let A and I represent the active and inactive groups in Θ, so $\Theta_g^\star \neq 0$ for any $g \in A$ and $\Theta_g^\star = 0$ for any $g \in I$.

Let $\hat{\Theta}$ be the minimizer to Equation (20). Then $\hat{\Theta}$ satisfies,

1.
   $$\|\hat{\Theta} - \Theta^\star\|_2 \leq C \sqrt{\frac{|A|\log|G|}{n}}$$

2. $\hat{\Theta}_g = 0$ for $g \in I$ The exact statement of the theorem is given in the Supplementary Material.

7 Optimization Algorithms

In this section, we discuss two algorithms for solving (15): the proximal gradient and the proximal newton methods. This is a convex optimization problem that decomposes into the form $f(x) + g(x)$, where f is smooth and convex and g is convex but possibly non-smooth. In our case f is the negative log-likelihood or negative log-pseudolikelihood and g are the group sparsity penalties.

Block coordinate descent is a frequently used method when the non-smooth function g is the l_1 or group l_1. It is especially easy to apply when the function f is quadratic, since each block coordinate update can be solved in closed form for many different non-smooth g (Friedman et al., 2007). The smooth f in our particular case is not quadratic, so each block update cannot be solved in closed form. However in certain problems (sparse inverse covariance),

the update can be approximately solved by using an appropriate inner optimization routine (Friedman et al., 2008b).

7.1 Proximal Gradient

Problems of this form are well-suited for the proximal gradient and accelerated proximal gradient algorithms as long as the proximal operator of *g* can be computed (Combettes and Pesquet, 2011; Beck and Teboulle, 2010)

$$prox_t\left(x\right) = \operatorname*{argmin}_{u} \frac{1}{2t} \|x - u\|^2 + g\left(u\right) \quad (21)$$

For the sum of l_2 group sparsity penalties considered, the proximal operator takes the familiar form of soft-thresholding and group soft-thresholding (Bach et al., 2011). Since the groups are non-overlapping, the proximal operator simplifies to scalar soft-thresholding for β_{st} and group soft-thresholding for ρ_{sj} and φ_{rj}.

The class of proximal gradient and accelerated proximal gradient algorithms is directly applicable to our problem. These algorithms work by solving a first-order model at the current iterate x_k

$$\operatorname*{argmin}_{u} \quad f\left(x_k\right) + \nabla f(x_k)^T \left(u - x_k\right) + \frac{1}{2t} \|u - x_k\|^2 + g\left(u\right) \quad (22)$$

$$= \operatorname*{argmin}_{u} \frac{1}{2t} \|u - \left(x_k - t\nabla f\left(x_k\right)\right)\|^2 + g\left(u\right) \quad (23)$$

$$= prox_t\left(x_k - t\nabla f\left(x_k\right)\right) \quad (24)$$

$$\operatorname*{argmin}_{u} \quad f\left(x_k\right) + \nabla f(x_k)^T \left(u - x_k\right) + \frac{1}{2t}\left(u - x_k\right)^T H\left(u - x_k\right) + g\left(u\right) \quad (25)$$

The proximal gradient iteration is given by $x_{k+1} = prox_t\left(x_k - t\nabla f(x_k)\right)$ where *t* is determined by line search. The theoretical convergence rates and properties of the proximal gradient algorithm and its accelerated variants are well-established (Beck and Teboulle, 2010). The accelerated proximal gradient method achieves linear convergence rate of $O(c^k)$ when the objective is strongly convex and the sublinear rate $O(1/k^2)$ for non-strongly convex problems.

The TFOCS framework (Becker et al., 2011) is a package that allows us to experiment with 6 different variants of the accelerated proximal gradient algorithm. The TFOCS authors found that the Auslender-Teboulle algorithm exhibited less oscillatory behavior, and proximal gradient experiments in the next section were done using the Auslender-Teboulle implementation in TFOCS.

7.2 Proximal Newton Algorithms

The class of proximal Newton algorithms is a 2nd order analog of the proximal gradient algorithms with a quadratic convergence rate (Lee et al., 2012; Schmidt, 2010; Schmidt et al., 2011). It attempts to incorporate 2nd order information about the smooth function f into the model function. At each iteration, it minimizes a quadratic model centered at x_k

$$= \underset{u}{argmin} \frac{1}{2t} \left(u - x_k + tH^{-1}\nabla f(x_k)\right)^T H \left(u - x_k + tH^{-1}\nabla f(x_k)\right) + g(u) \quad (26)$$

$$= \underset{u}{argmin} \frac{1}{2t} \left\| u - \left(x_k - tH^{-1}\nabla f(x_k)\right) \right\|_H^2 + g(u) \quad (27)$$

$$:= Hprox_t \left(x_k - tH^{-1}\nabla f(x_k)\right) \quad \text{where} \quad H = \nabla^2 f(x_k) \quad (28)$$

$$f(x_k + tp_k) + g(x_k + tp_k) \leq f(x_k) + g(x_k) - \frac{t\alpha}{2}\|p_k\|^2$$

The *Hprox* operator is analogous to the proximal operator, but in the $\|\cdot\|_H$-norm. It simplifies

Algorithm 1

Proximal Newton

---

repeat

    Solve subproblem $p_k = Hprox_t\left(x_k - tH_k^{-1}\nabla f(x_k)\right) - x_k$ using TFOCS.

    Find t to satisfy Armijo line search condition with parameter a

        $f(x_k + tp_k) + g(x_k + tp_k) \quad f(x_k) + g(x_k) - \frac{t\alpha}{2}\| p_k \|^2$

    Set $x_{k+1} = x_k + tp_k$

    $k = k + 1$

until $\dfrac{\| x_k - x_{k+1}\|}{\| x_k \|} < tol$

---

to the proximal operator if $H = I$, but in the general case of positive definite H there is no closed-form solution for many common non-smooth $g(x)$ (including l_1 and group l_1). However if the proximal operator of g is available, each of these sub-problems can be solved efficiently with proximal gradient. In the case of separable g, coordinate descent is also applicable. Fast methods for solving the subproblem $Hprox_t(x_k - tH^{-1}\nabla rf(x_k))$ include coordinate descent methods, proximal gradient methods, or Barzilai-Borwein (Friedman et al., 2007; Combettes and Pesquet, 2011; Beck and Teboulle, 2010; Wright et al., 2009). The proximal Newton framework allows us to bootstrap many previously developed solvers to the case of arbitrary loss function f.

Theoretical analysis in Lee et al. (2012) suggests that proximal Newton methods generally require fewer outer iterations (evaluations of *Hprox*) than first-order methods while providing higher accuracy because they incorporate 2nd order information. We have confirmed empirically that the proximal Newton methods are faster when *n* is very large or the gradient is expensive to compute (e.g. maximum likelihood estimation). Since the objective is quadratic, coordinate descent is also applicable to the subproblems. The hessian matrix *H* can be replaced by a quasi-newton approximation such as BFGS/L-BFGS/SR1. In our implementation, we use the **PNOPT** implementation (Lee et al., 2012).

7.3 Path Algorithm

Frequently in machine learning and statistics, the regularization parameter λ is heavily dependent on the dataset. λ is generally chosen via cross-validation or holdout set performance, so it is convenient to provide solutions over an interval of $[\lambda_{min}, \lambda_{max}]$. We start the algorithm at $\lambda_1 = \lambda_{\max}$ and solve, using the previous solution as warm start, for $\lambda_2 > \ldots > \lambda_{\min}$. We find that this reduces the cost of fitting an entire path of solutions (See Figure 4). λ_{max} can be chosen as the smallest value such that all parameters are 0 by using the KKT equations (Friedman et al., 2007).

8 Conditional Model

In addition to the variables we would like to model, there are often additional features or covariates that affect the dependence structure of the variables. For example in genomic data, in addition to expression values, we have attributes associated to each subject such as gender, age and ethnicity. These additional attributes affect the dependence of the expression values, so we can build a conditional model that uses the additional attributes as features. In this section, we show how to augment the pairwise mixed model with features.

Conditional models only model the conditional distribution $p(z|f)$, as opposed to the joint distribution $p(z, f)$, where z are the variables of interest to the prediction task and f are features. These models are frequently used in practice Lafferty et al. (2001).

In addition to observing x and y, we observe features f and we build a graphical model for the conditional distribution $p(x, y|f)$. Consider a full pairwise model $p(x, y, f)$ of the form (1). We then choose to only model the joint distribution over only the variables x and y to give us $p(x, y|f)$ which is of the form

$$
\begin{aligned}
&p\left(x,y|f;\Theta\right) \\
&= \frac{1}{Z\left(\Theta|f\right)} exp \left(\sum_{s=1}^{p}\sum_{t=1}^{p} -\frac{1}{2}\beta_{st}x_s x_t + \sum_{s=1}^{p}\alpha_s x_s + \sum_{s=1}^{p}\sum_{j=1}^{q}\rho_{sj}\left(y_j\right)x_s + \sum_{j=1}^{q}\sum_{r=1}^{j}\phi_{rj}\left(y_r,y_j\right) + \sum_{l=1}^{F}\sum_{s=1}^{p}\gamma_{ls}x_s f_l + \sum_{l=1}^{F}\sum_{r=1}^{q}\eta_{lr}\left(y_r\right)f_l \right)
\end{aligned}
\tag{29}
$$

We can also consider a more general model where each pairwise edge potential depends on the features

$$
p\left(x,y|f;\Theta\right) = \frac{1}{Z\left(\Theta|f\right)} exp \left(\sum_{s=1}^{p}\sum_{t=1}^{p} -\frac{1}{2}\beta_{st}\left(f\right)x_s x_t + \sum_{s=1}^{p}\alpha_s\left(f\right)x_s + \sum_{s=1}^{p}\sum_{j=1}^{q}\rho_{sj}\left(y_j,f\right)x_s + \sum_{j=1}^{q}\sum_{r=1}^{j}\phi_{rj}\left(y_r,y_j,f\right) \right)
\tag{30}
$$

(29) is a special case of this where only the node potentials depend on features and the pairwise potentials are independent of feature values. The specific parametrized form we consider is $\varphi_{rj}(y_r, y_j, f) \equiv \varphi_{rj}(y_r, y_j)$ for $r \quad j$, $\rho_{sj}(y_j, f) \equiv \rho_{sj}(y_j)$, and $\beta_{st}(f) = \beta_{st}$. The node potentials depend linearly on the feature values, $\alpha_s(f) = \alpha_s + \sum_{l=1}^{F} \gamma_{ls} x_s f_l$, and $\varphi_{rr}(y_r, y_r, f) = \varphi_{rr}(y_r, y_r) + \Sigma_l \eta_{lr}(y_r)$.

9 Experimental Results

We present experimental results on synthetic data, survey data and on a conditional model.

9.1 Synthetic Experiments

In the synthetic experiment, the training points are sampled from a true model with 10 continuous variables and 10 binary variables. The edge structure is shown in Figure 2a. λ is chosen proportional to $\sqrt{\frac{log(p+q)}{n}}$ as suggested by the theoretical results in Section 6. We experimented with 3 values $\lambda = \{1, 5, 10\} \sqrt{\frac{log(p+q)}{n}}$ and chose $\lambda = 5 \sqrt{\frac{log(p+q)}{n}}$ so that the true edge set was recovered by the algorithm for the sample size $n = 2000$. We see from the experimental results that recovery of the correct edge set undergoes a sharp phase transition, as expected. With $n = 1000$ samples, the pseudolikelihood is recovering the correct edge set with probability nearly 1. The maximum likelihood was performed using an exact evaluation of the gradient and log-partition. The poor performance of the maximum likelihood estimator is explained by the maximum likelihood objective violating the irrepresentable condition; a similar example is discussed in (Ravikumar et al., 2010, Section 3.1.1), where the maximum likelihood is not irrepresentable, yet the neighborhood selection procedure is. The phase transition experiments were done using the proximal Newton algorithm discussed in Section 7.2.

We also run the proximal Newton algorithm for a sequence of instances with $p = q = 10, 50, 100, 500, 1000$ and $n = 500$. The largest instance has 2000 variables and takes 12.5 hours to complete. The timing results are summarized in Figure 3.

9.2 Survey Experiments

The census survey dataset we consider consists of 11 variables, of which 2 are continuous and 9 are discrete: age (continuous), log-wage (continuous), year(7 states), sex(2 states),marital status (5 states), race(4 states), education level (5 states), geographic region(9 states), job class (2 states), health (2 states), and health insurance (2 states). The dataset was assembled by Steve Miller of OpenBI.com from the March 2011 Supplement to Current Population Survey data. All the evaluations are done using a holdout test set of size 100, 000 for the survey experiments. The regularization parameter λ is varied over the interval $[5 \times 10^{-5}, 0.7]$ at 50 points equispaced on log-scale for all experiments. In practice, λ can be chosen to minimize the holdout log pseudolikelihood.

9.2.1 Model Selection—In Figure 4, we study the model selection performance of learning a graphical model over the 11 variables under different training samples sizes. We

see that as the sample size increases, the optimal model is increasingly dense, and less regularization is needed.

9.2.2 Comparing against Separate Regressions—A sensible baseline method to compare against is a separate regression algorithm. This algorithm fits a linear Gaussian or (multiclass) logistic regression of each variable conditioned on the rest. We can evaluate the performance of the pseudolikelihood by evaluating $-\log p(x_s|x_{\backslash s}, y)$ for linear regression and $-\log p(y_r|y_{\backslash r}, x)$ for (multiclass) logistic regression. Since regression is directly optimizing this loss function, it is expected to do better. The pseudolikelihood objective is similar, but has half the number of parameters as the separate regressions since the coefficients are shared between two of the conditional likelihoods. From Figures 5 and 6, we can see that the pseudolikelihood performs very similarly to the separate regressions and sometimes even outperforms regression. The benefit of the pseudolikelihood is that we have learned parameters of the joint distribution $p(x, y)$ and not just of the conditionals $p(x_s|y, x_{\backslash s})$. On the test dataset, we can compute quantities such as conditionals over arbitrary sets of variables $p(y_A, x_B|y_A{}^C, x_B{}^C)$ and marginals $p(x_A, y_B)$ (Koller and Friedman, 2009). This would not be possible using the separate regressions.

9.2.3 Conditional Model—Using the conditional model (29), we model only the 3 variables logwage, education(5) and jobclass(2). The other 8 variables are only used as features. The conditional model is then trained using the pseudolikelihood. We compare against the generative model that learns a joint distribution on all 11 variables. From Figure 7, we see that the conditional model outperforms the generative model, except at small sample sizes. This is expected since the conditional distribution models less variables. At very small sample sizes and small λ, the generative model outperforms the conditional model. This is likely because generative models converge faster (with less samples) than discriminative models to its optimum.

9.2.4 Maximum Likelihood vs Pseudolikelihood—The maximum likelihood estimates are computable for very small models such as the conditional model previously studied. The pseudolikelihood was originally motivated as an approximation to the likelihood that is computationally tractable. We compare the maximum likelihood and maximum pseudolikelihood on two different evaluation criteria: the negative log likelihood and negative log pseudolikelihood. In Figure 8, we find that the pseudolikeli-hood outperforms maximum likelihood under both the negative log likelihood and negative log pseudolikelihood. We would expect that the pseudolikelihood trained model does better on the pseudolikelihood evaluation and maximum likelihood trained model does better on the likelihood evaluation. However, we found that the pseudolikelihood trained model outperformed the maximum likelihood trained model on both evaluation criteria. Although asymptotic theory suggests that maximum likelihood is more efficient than the pseudolikeli-hood, this analysis is applicable because of the finite sample regime and misspecified model. See Liang and Jordan (2008) for asymptotic analysis of pseudolikelihood and maximum likelihood under a well-specified model. We also observed the pseudolikelihood slightly outperforming the maximum likelihood in the synthetic experiment of Figure 2b.

10 Conclusion

This work proposes a new pairwise mixed graphical model, which combines the Gaussian graphical model and discrete graphical model. Due to the introduction of discrete variables, the maximum likelihood estimator is computationally intractable, so we investigated the pseudolikelihood estimator. To learn the structure of this model, we use the appropriate group sparsity penalties with a calibrated weighing scheme. Model selection consistency results are shown for the mixed model using the maximum likelihood and pseudolikelihood estimators. The extension to a conditonal model is discussed, since these are frequently used in practice.

We proposed two efficient algorithms for the purpose of estimating the parameters of this model, the proximal Newton and the proximal gradient algorithms. The proximal Newton algorithm is shown to scale to graphical models with 2000 variables on a standard desktop. The model is evaluated on synthetic and the current population survey data, which demonstrates the pseudolikelihood performs well compared to maximum likelihood and nodewise regression.

For future work, it would be interesting to incorporate other discrete variables such as poisson or binomial variables and non-Gaussian continuous variables. This would broaden the scope of applications that mixed models could be used for. Our work is a first step in that direction.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgements

References

Bach F, Jenatton R, Mairal J, Obozinski G. Optimization with sparsity-inducing penalties. Foundations and Trends in Machine Learning. 2011; 4:1–106. URL http://dx.doi.org/10.1561/2200000015.

Banerjee O, El Ghaoui L, d'Aspremont A. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. The Journal of Machine Learning Research. 2008; 9:485–516.

Beck A, Teboulle M. Gradient-based algorithms with applications to signal recovery problems. Convex Optimization in Signal Processing and Communications. 2010:42–88.

Becker SR, Candès EJ, Grant MC. Templates for convex cone problems with applications to sparse signal recovery. Mathematical Programming Computation. 2011:1–54.

Besag J. Spatial interaction and the statistical analysis of lattice systems. Journal of the Royal Statistical Society. Series B (Methodological). 1974:192–236.

Besag J. Statistical analysis of non-lattice data. The statistician. 1975:179–195.

Cheng, Jie; Levina, Elizaveta; Zhu, Ji. High-dimensional mixed graphical models. arXiv preprint arXiv:1304.2810. 2013

Combettes PL, Pesquet JC. Proximal splitting methods in signal processing. Fixed-Point Algorithms for Inverse Problems in Science and Engineering. 2011:185–212.

Edwards, D. Introduction to graphical modelling. Springer; 2000.

Friedman J, Hastie T, Höfling H, Tibshirani R. Pathwise coordinate optimization. The Annals of Applied Statistics. 2007; 1(2):302–332.

Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. Biostatistics. 2008a; 9(3):432–441. [PubMed: 18079126]

Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. Biostatistics. 2008b; 9(3):432–441. [PubMed: 18079126]

Friedman, J.; Hastie, T.; Tibshirani, R. Applications of the lasso and grouped lasso to the estimation of sparse graphical models. Technical report, Technical Report. Stanford University; 2010.

Guo, J.; Levina, E.; Michailidis, G.; Zhu, J. Joint structure estimation for categorical markov networks. 2010. Submitted. Available at http://www.stat.lsa.umich.edu/~elevina

Höfling H, Tibshirani R. Estimation of sparse binary pairwise markov networks using pseudo-likelihoods. The Journal of Machine Learning Research. 2009; 10:883–906.

Jalali A, Ravikumar P, Vasuki V, Sanghavi S, UT ECE; UT CS. On learning discrete graphical models using group-sparse regularization. Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS). 2011

Kim, Seyoung; Sohn, Kyung-Ah; Xing, Eric P. A multivariate regression approach to association analysis of a quantitative trait network. Bioinformatics. 2009; 25(12):i204–i212. [PubMed: 19477989]

Koller, D.; Friedman, N. Probabilistic graphical models: principles and techniques. The MIT Press; 2009.

Lafferty, John; McCallum, Andrew; Pereira, Fernando CN. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001

Lauritzen, SL. Graphical models. Vol. 17. Oxford University Press; USA: 1996.

Lee, Jason D.; Sun, Yuekai; Taylor, Jonathan. On model selection consistency of mestimators with geometrically decomposable penalties. arXiv preprint arXiv:1305.7477. 2013

Lee JD, Sun Y, Saunders MA. Proximal newton-type methods for minimizing convex objective functions in composite form. arXiv preprint arXiv:1206.1623. 2012

Lee SI, Ganapathi V, Koller D. Efficient structure learning of markov networks using l1regularization. NIPS. 2006

Liang, P.; Jordan, MI. Proceedings of the 25th international conference on Machine learning. ACM; 2008. An asymptotic analysis of generative, discriminative, and pseudolikelihood estimators.; p. 584-591.

Liu Q, Ihler A. Learning scale free networks by reweighted l1 regularization. Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS). 2011

Liu Q, Ihler A. Distributed parameter estimation via pseudo-likelihood. Proceedings of the International Conference on Machine Learning (ICML). 2012

Meinshausen N, Bühlmann P. High-dimensional graphs and variable selection with the lasso. The Annals of Statistics. 2006; 34(3):1436–1462.

Peng J, Wang P, Zhou N, Zhu J. Partial correlation estimation by joint sparse regression models. Journal of the American Statistical Association. 2009; 104(486):735–746. [PubMed: 19881892]

Ravikumar P, Wainwright MJ, Lafferty JD. High-dimensional ising model selection using l1-regularized logistic regression. The Annals of Statistics. 2010; 38(3):1287–1319.

Rothman, Adam J.; Levina, Elizaveta; Zhu, Ji. Sparse multivariate regression with covariance estimation. Journal of Computational and Graphical Statistics. 2010; 19(4):947–962. [PubMed: 24963268]

Schmidt, M. PhD thesis. University of British Columbia; 2010. Graphical Model Structure Learning with l1-Regularization..

Schmidt M, Murphy K, Fung G, Rosales R. Structure learning in random fields for heart motion abnormality detection. CVPR. IEEE Computer Society. 2008

Schmidt M, Kim D, Sra S. Projected newton-type methods in machine learning. 2011

Tur, Inma; Castelo, Robert. Learning mixed graphical models from data with p larger than n. arXiv preprint arXiv:1202.3765. 2012

Witten, Daniela M.; Tibshirani, Robert. Covariance-regularized regression and classification for high dimensional problems. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2009; 71(3):615–636.

Wright SJ, Nowak RD, Figueiredo MAT. Sparse reconstruction by separable approximation. Signal Processing, IEEE Transactions on. 2009; 57(7):2479–2493.

Yang E, Ravikumar P, Allen G, Liu Z. Graphical models via generalized linear models. Advances in Neural Information Processing Systems 25. 2012:1367–1375.

Yang E, Ravikumar P, Allen GI, Liu Z. On graphical models via univariate exponential family distributions. arXiv preprint arXiv:1301.4183. 2013

Yuan M, Lin Y. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2006; 68(1):49–67.
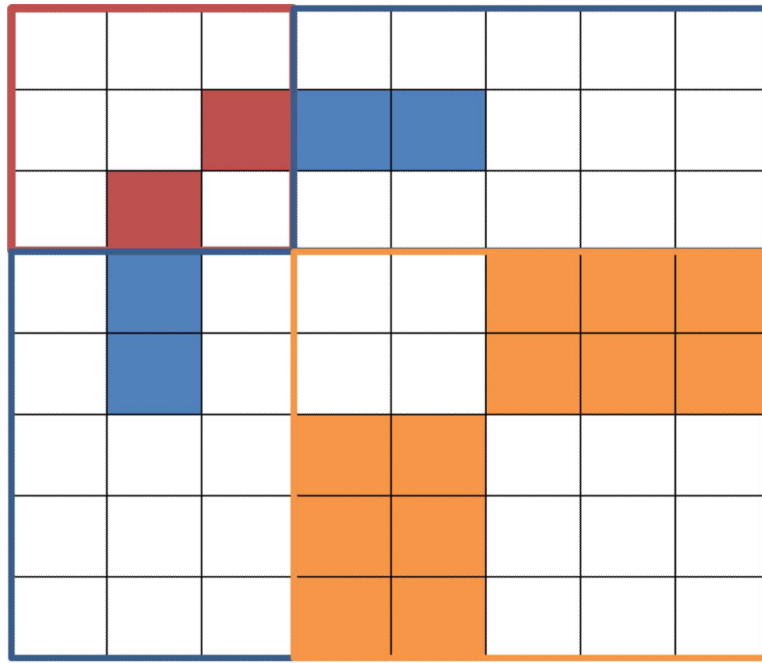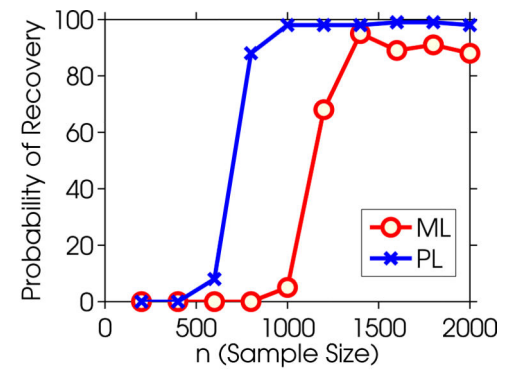
Figure 1.

Symmetric matrix represents the parameters Θ of the model. This example has $p = 3$, $q = 2$, $L_1 = 2$ and $L_2 = 3$. The red square corresponds to the continuous graphical model coefficients B and the solid red square is the scalar β_{st}. The blue square corresponds to the coefficients ρ_{sj} and the solid blue square is a vector of parameters $\rho_{sj}(\cdot)$. The orange square corresponds to the coefficients φ_{rj} and the solid orange square is a matrix of parameters $\varphi_{rj}(\cdot, \cdot)$. The matrix is symmetric, so each parameter block appears in two of the conditional probability regressions.
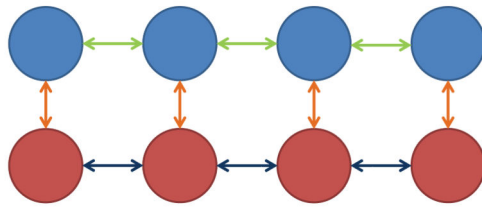
(a) Graph Structure with 4 continuous and 4 discrete variables.


(b) Probability of recovery

Figure 2.
Figure 2a shows the graph used in the synthetic experiments for p = q = 4; the experiment actually used p=10 and q=10. Blue nodes are continuous variables, red nodes are binary variables and the orange, green and dark blue lines represent the 3 types of edges. Figure 2b is a plot of the probability of correct edge recovery, meaning every true edge is selected and no non-edge is selected, at a given sample size using Maximum Likelihood and Pseudolikelihood. Results are averaged over 100 *trials*.

| $p+q$	Time per Iteration (sec)	Total Time (min)	Number of Iterations
20	.13	.003	13
100	4.39	1.32	18
200	18.44	6.45	21
1000	245.34	139	34
2000	1025.6	752	44

Figure 3.
Timing experiments for various instances of the graph in Figure 2a. The number of variables range from 20 to 2000 with $n = 500$.
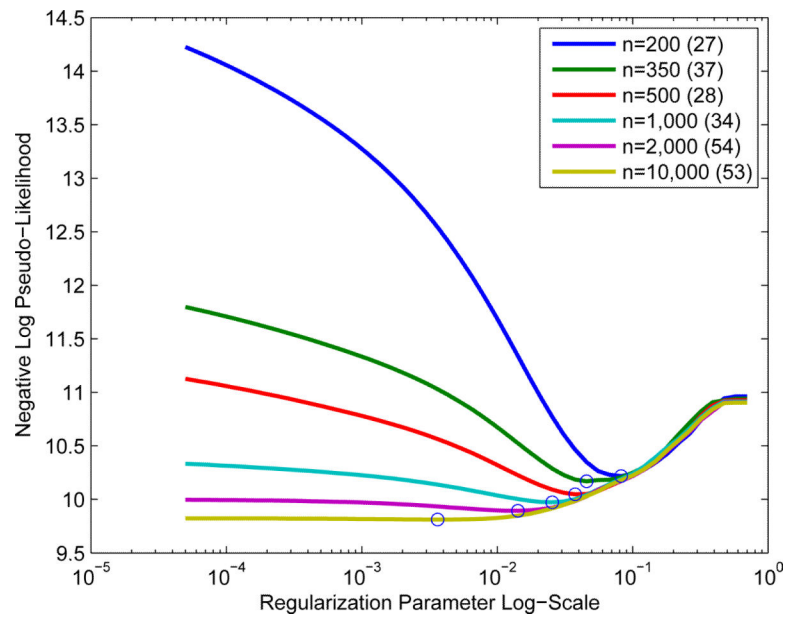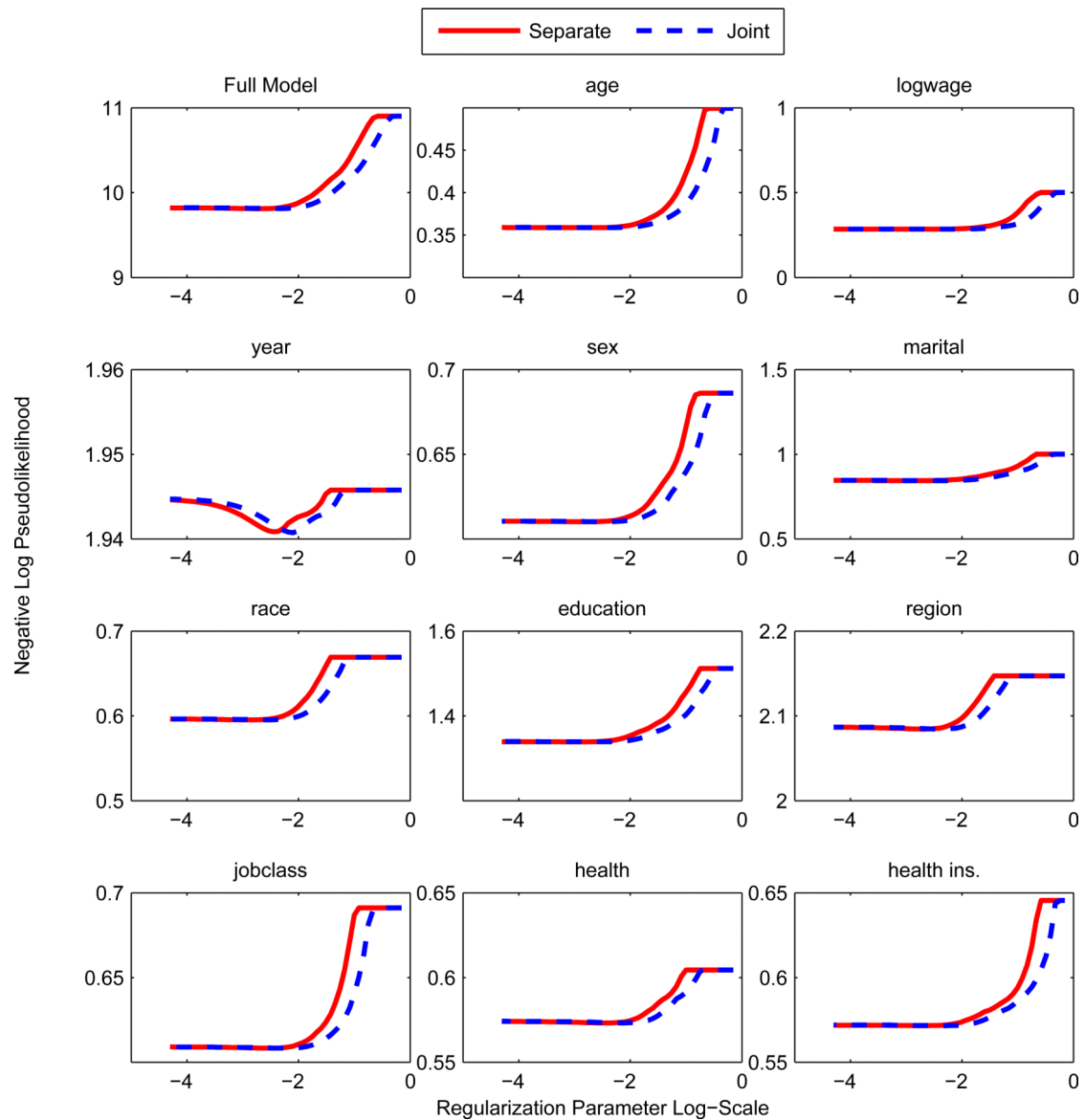
Figure 4.
Model selection under different training set sizes. Circle denotes the lowest test set negative log pseudolikelihood and the number in parentheses is the number of edges in that model at the lowest test negative log pseudolikelihood. The saturated model has 55 *edges*.
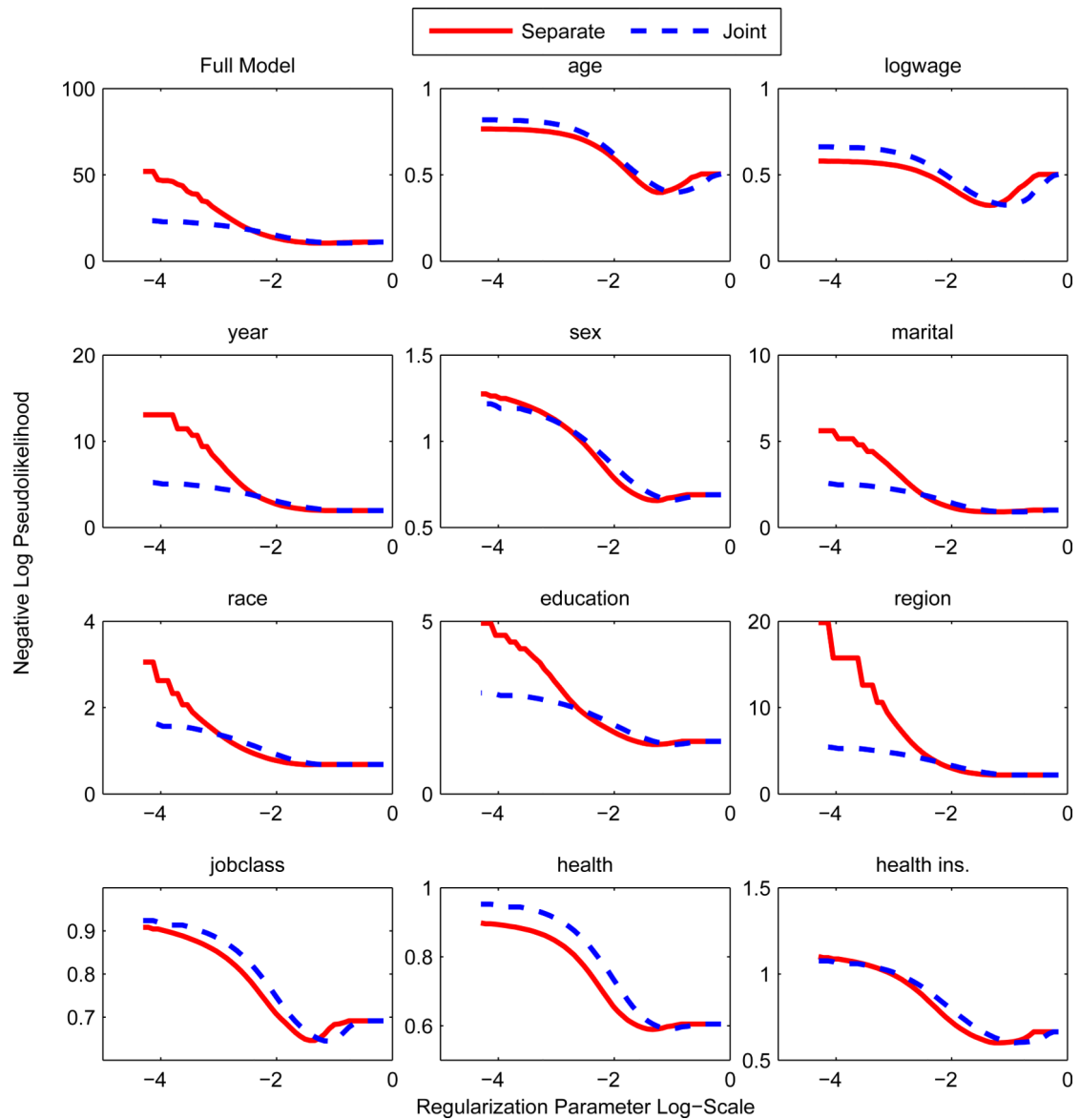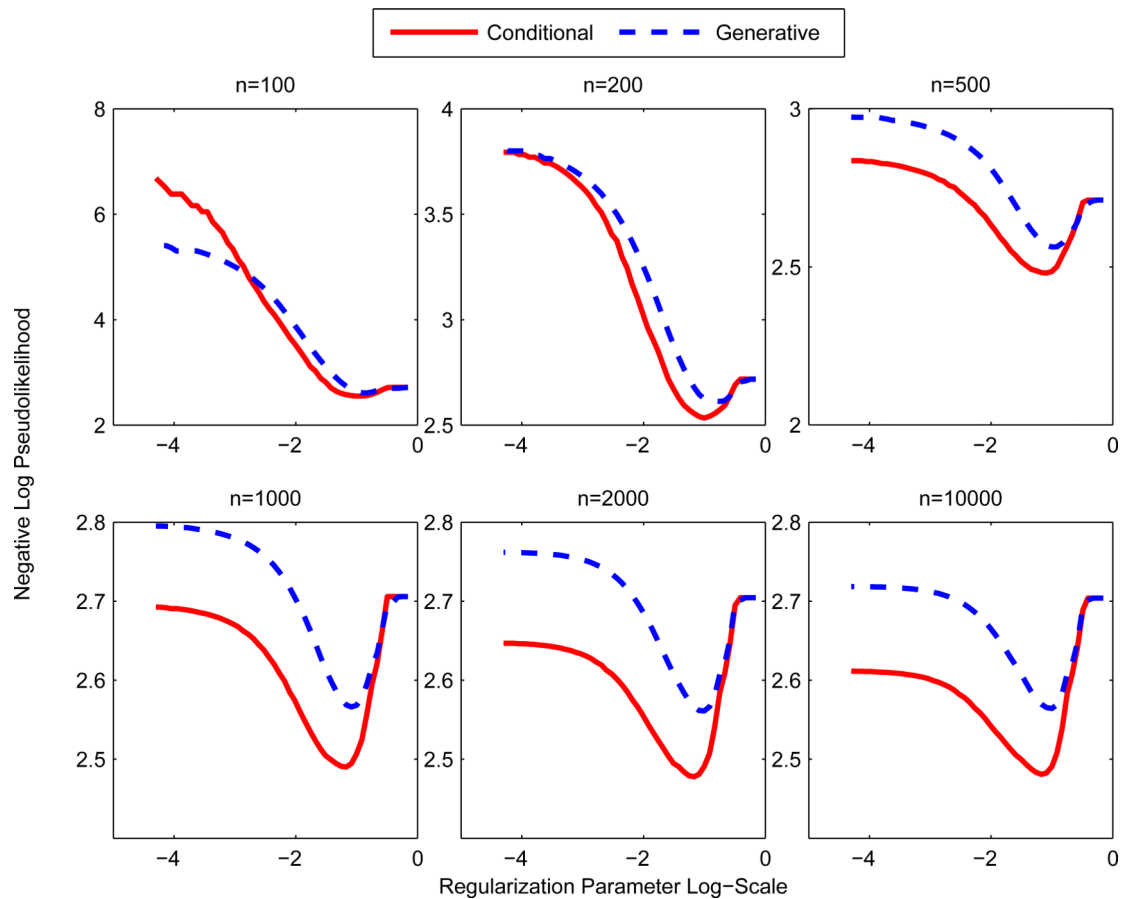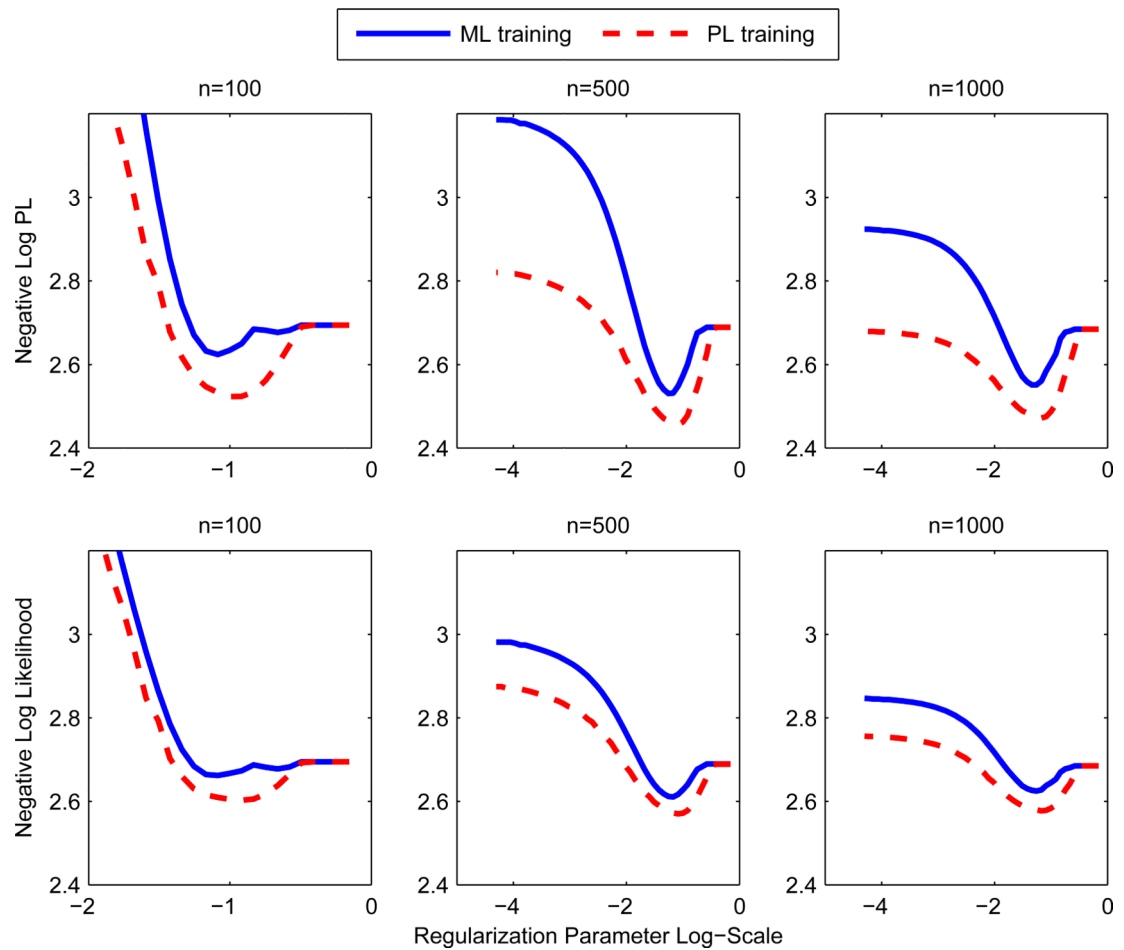
Figure 5.
Separate Regression vs Pseudolikelihood n = 100. y-axis is the appropriate regression loss for the response variable. For low levels of regularization and at small training sizes, the pseudolikelihood seems to overfit less; this may be due to a global regularization effect from fitting the joint distribution as opposed to separate regressions.

Figure 6.
Separate Regression vs Pseudolikelihood n = 10, 000. y-axis is the appropriate regression loss for the response variable. At large sample sizes, separate regressions and pseudolikelihood perform very similarly. This is expected since this is nearing the asymptotic regime.

Figure 7.
Conditional Model vs Generative Model at various sample sizes. y-axis is test set
performance is evaluated on negative log pseudolikelihood of the conditional model. The
conditional model outperforms the full generative model at except the smallest sample size n
= 100.

Figure 8.
Maximum Likelihood vs Pseudolikelihood. y-axis for top row is the negative log pseudolikelihood. y-axis for bottom row is the negative log likelihood. Pseudolikelihood outperforms maximum likelihood across all the experiments.

Table 1

Penalty weights in a six-edge model.

	$\left\lVert \frac{\ell}{\phi_{12}} \right\rVert_F$	$\left\lVert \frac{\ell}{\rho_{11}} \right\rVert_2$	$\left\lVert \frac{\ell}{\rho_{21}} \right\rVert_2$	$\left\lVert \frac{\ell}{\rho_{12}} \right\rVert_2$	$\left\lVert \frac{\ell}{\rho_{22}} \right\rVert_2$	$\left\lvert \frac{\ell}{\beta_{12}} \right\rvert$
Exact weights w_g (17)	0.18	0.63	0.19	0.47	0.15	0.53
Approximate weights w_g (19)	0.13	0.59	0.18	0.44	0.13	0.62

Row 1 shows the exact weights w_g computed via (17) using Monte Carlo simulation. Row 2 shows the approximate weights computed via (19). Note that the weights are far from uniform, and the approximate weights are close to the exact weights.

Table 2

Fraction of times an edge is the first selected by the group lasso regularizer, based on 1000 simulation runs.

	φ_{12}	ρ_{11}	ρ_{21}	ρ_{12}	ρ_{22}	β_{12}
No Calibration $w_g = 1$	0.000	0.487	0.000	0.163	0.000	0.350
Exact w_g (17)	0.101	0.092	0.097	0.249	0.227	0.234
Approximate w_g (19)	0.144	0.138	0.134	0.196	0.190	0.198

Ideally each edge should be first 1/6th of the time (0.167), with a standard error of 0.012. The group lasso with equal weights (first row) is highly unbalanced. Using the exact weights from (17) is quite good (second row), while the approximate weighing scheme of (19) (third row) appears to perform the best.