DEPARTMENT OF STATISTICS
University of Wisconsin
1300 University Ave.
Madison, WI 53706

Learning Higher-Order Graph Structure with Features by Structure Penalty

Shilin Ding[1]
Department of Statistics
University of Wisconsin, Madison

Grace Wahba[1]
Department of Statistics, Department of Computer Sciences
and Department of Biostatistics and Medical Informatics
University of Wisconsin, Madison

Xiaojin Zhu
Department of Computer Sciences
University of Wisconsin, Madison

Learning Higher-Order Graph Structure with Features by Structure Penalty

Shilin Ding   SDING@STAT.WISC.EDU
Department of Statistics
University of Wisconsin-Madison
Madison, WI 53706, USA

Grace Wahba   WAHBA@STAT.WISC.EDU
Department of Statistics, Department of Computer Sciences
and Department of Biostatistics and Medical Informatics
University of Wisconsin-Madison
Madison, WI 53706, USA

Xiaojin Zhu   JERRYZHU@CS.WISC.EDU
Department of Computer Sciences
University of Wisconsin-Madison
Madison, WI 53706, USA

Abstract

In discrete undirected graphical models, the conditional independence of node labels Y is specified by the graph structure. We study the case where there is another input random vector X (e.g. observed features) such that the distribution $P(Y \mid X)$ is determined by functions of X that characterize the higher-order interactions among the Y's. The main contribution of this paper is to learn the graph structure and the functions conditioned on X at the same time. We prove that discrete undirected graphical models with feature X are equivalent to multivariate discrete models. The reparameterization of the potential functions in graphical models by conditional log odds ratios of the latter offers advantages in the representation of the conditional independence structure in the model. The functional spaces can be flexibly determined by kernels. Additionally, we impose a structure penalty on groups of functions to learn the graph structure. These groups with overlaps are designed to enforce hierarchical function selection. In this way, we are able to shrink higher order interactions to obtain a sparse graph structure.

1. Introduction

In undirected graphical models (UGMs), a graph is defined as $G = (V, E)$, where $V = \{1, \cdots, K\}$ is the set of nodes and $E \subset V \times V$ is the set of edges between the nodes. UGMs have been widely used in computer vision, natural language processing and other applications (Szeliski et al., 2007; Banerjee et al., 2008; Schnitzspan et al., 2009). The Conditional Random Fields (CRFs) (Lafferty et al., 2001) achieve great success by modeling the effects of features X on the labels (responses) Y of the nodes.

The graph structure specifies the conditional independence among nodes. Much prior work has focused on graphical model structure learning without conditioning on X. For instance, Meinshausen and Buhlmann (2006) and Peng *et al.* (2009) studied sparse covariance

estimation of Gaussian Markov Random Fields. The covariance matrix fully determines the dependence structure in the Gaussian distribution and thus specifies the linkage. However, this is not the case for non-elliptical distributions, such as those in discrete UGMs. Ravikumar *et al.* (2010) discussed consistent structure selection of Ising models based on l_1-regularized logistic regression, while Höfling and Tibshirani (2009) proposed pseudo-likelihood with l_1 penalty for Ising models. Ising models are special cases of discrete UGMs with (usually) only pairwise interactions, and without features. We focused on discrete UGMs with both higher order interactions and features. It is important to note that the graph structure may change conditioned on different X's, thus our approach may lead to better estimates.

In addressing the problem of structure learning with features, Liu *et al.* (2010) assumed Gaussian distributed Y given X, and they partitioned the space of X into bins. Schmidt *et al.* (2008) proposed a framework to jointly learn pairwise CRFs and parameters with block-l_1 regularization. Bradley and Guestrin (2010) learned tree CRF that recovers a max spanning tree of a complete graph based on heuristic pairwise link scores. These methods utilize only pairwise information to scale to large graphs. The closest work is Schmidt and Murphy (2010), which examined the higher-order graphical structure learning problem without considering features. They used an active set method to learn higher order interactions in a greedy manner. Their model is over-parameterized, and the hierarchical assumption is sufficient but not necessary for conditional independence in the graph.

To the best of our knowledge, no previous work addressed the issue of graph structure learning of all orders while conditioning on input features. Our contributions include a reparemeterization of UGMs with bivariate outcomes into multivariate Bernoulli (MVB) models[1]. The set of conditional log odds ratios in MVB models are complete to represent the effects of features on responses and their interactions at all levels. The sparsity in this set of functions are sufficient and necessary for the conditional independence in the graph. Therefore, the relation between graph structure and sparsity in the set of interactions is straightforward: Two nodes are conditionally independent iff the pairwise interaction is constant zero. And some non-zero higher order interaction means all the nodes involved are not conditionally independent.

To obtain a sparse graph structure, we impose structure penalty on groups of functions with overlaps. Our approach can be viewed as group lasso with overlaps. Group lasso (Yuan and Lin, 2006) leads to selection of variables in groups. It is consistent when the groups are exclusive and cover the whole set. Jacob *et al.* (2009) considered the penalty on groups with arbitrary overlaps. Zhao *et al.* (2009) set up the general framework for hierarchical variable selection with overlapping groups, which we adopt here for the functions. Our groups are designed to enforce the sparsity on the functions and shrink higher order interactions similar to hierarchical inclusion restriction in Schimdt and Murphy (2010). We give a proximal linearization algorithm that efficiently learns the complete model where the normalization factor is calculated by the junction tree algorithm (Koller and Friedman, 2009). Global convergence is guaranteed (Wright, 2010). It can be used in applications where the number of responses is small, such as the Census Bureau data in Section 5.2, or the biomedical data in which the effects of environmental variables (smoking) on multiple correlated clinical

---

1. It can be easily extended to general discrete UGMs as shown in the end of Section 2.

responses (diabetes) are interesting. We then propose a greedy search algorithm to scale our method up to large graphs as the number of parameters grows exponentially.

2. Conditional Independence in Discrete Undirected Graphical Models

Table 1: Notations

Symbol	Description
$\|\cdot\|$	Euclidean l_2 norm
n	Sample size
p	Number of covariates
K	Number of Response/Output
Y	K dimensional Response/Output
X	Covariates/Feature/Input, $n \times p$ matirx
Ω	Set of $\{1, 2, \ldots, K\}$
Ψ_K	Power set of Ω except the empty set
ω, κ, υ	Element of Ψ_K used for indexing
$y^\omega(i)$	$y^\omega(i) = \prod_{k \in \omega} y_k(i)$
$\mathcal{Y}(i)$	Augmented responses $(y^1(i), \ldots, y^\Omega(i))$ where $y^\omega(i) = \prod_{k \in \omega} y_k(i)$
c	Model parameters
$\tilde{p}$	$(p+1) \cdot |\Psi_k|$, dimension of c
T_υ	$T_\upsilon = \{\omega | \upsilon \subset \omega\}$ is the subgraph rooted at υ containing all its descendants
f^{T_υ}	$f^{T_\upsilon} = (f^\omega), \omega \in T_\upsilon$, where f^ω is the conditonal log odds ratio
$J(f^{T_\upsilon})$	Penalty on f^{T_υ}
p_υ	Weight for penalty on structure T_υ
s_υ, r_υ	Subgradient of $\lambda J(f^{T_\upsilon})$ of the υth group
$S^\omega(y; x)$	$S^\omega(y; x) = \sum_{\kappa \in T^\omega} y^\kappa f^\kappa$

The notations in this paper are summarized in Table 1. In this section, we first discuss the relationship between the multivariate Bernoulli (MVB) model and the UGM whose nodes are binary, i.e. $Y_i = 0$ or 1. At the end, we will give the representation of the general discrete UGM where Y_i takes value in $\{0, \cdots, m-1\}$. In UGMs, the distribution of multivariate discrete random variables $Y_1, \ldots, Y_K$ given X is:

$$P(Y_1 = y_1, \ldots, Y_K = y_K | X) = \frac{1}{Z(X)} \prod_{C \in \mathcal{C}} \Phi_C(y_C; X) \tag{1}$$

where $Z(X)$ is the normalization factor. The distribution is factorized according to the cliques in the graph. A clique $C \subset \Omega = \{1, \ldots, K\}$ is the set of nodes that are fully connected. $\Phi_C(y_C; X)$ is the potential function on C, indexed by $y_C = (y_i)_{i \in C}$. For the purpose of efficient computation, $\mathcal{C}$ is usually the set of all maximal cliques, which are not properly contained in any other cliques. This factorization follows from the Markov property: any two nodes not in a clique are conditionally independent given others. Different representations by non-maximal cliques can be converted to maximal clique representation by reformulation of the potential functions (Wainwright and Jordan, 2008). So $\mathcal{C}$ does not

3

have to comply with the graph structure, as long as it is sufficient. For example, the most general choice for any given graph is $\mathcal{C} = \{\Omega\}$. See Theorem 2.2 and Example 2.1 for details.

(a) Graph 1

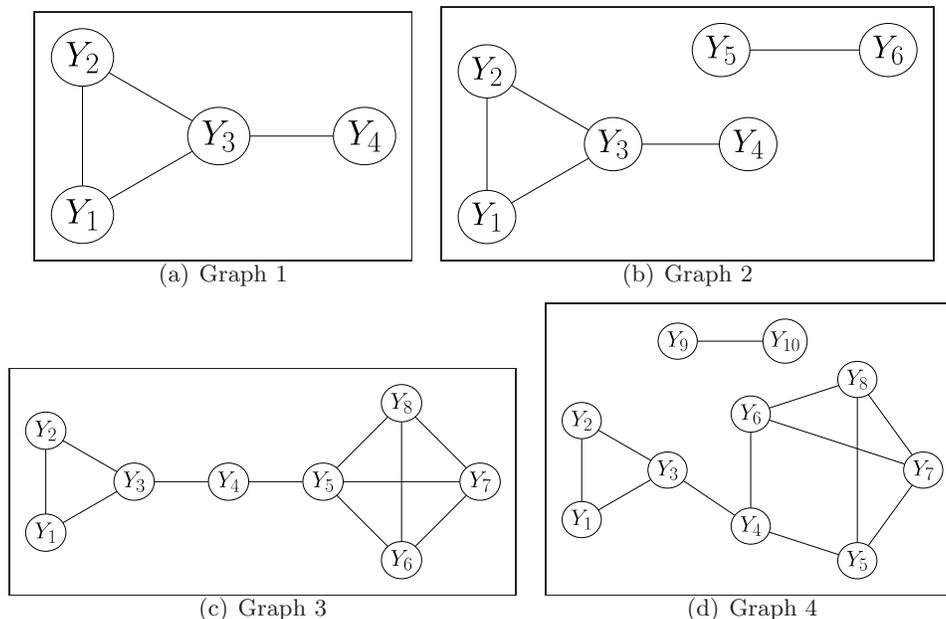(b) Graph 2

(c) Graph 3

(d) Graph 4

Figure 1: Graphical model examples.

Given the graph structure, the potential functions characterize the distribution on the graph. But if the graph is unknown in advance, estimating the potential functions on all possible cliques tends to be over-parameterized (Schmidt and Murphy, 2010). Furthermore, $\log \Phi_C(y_C; X) = 0$ is sufficient for the conditional independence between the nodes but not necessary (see Example 2.1). To avoid these problems, we introduce the MVB model that is equivalent to (1) with binary nodes, i.e. $Y_i = 0$ or 1. The MVB distribution is:

$$P(Y_1 = y_1, \ldots, Y_K = y_k | X = x) = \exp\Big\{ \sum_{\omega=1}^{\Omega} y^\omega f^\omega - b(f) \Big\} \tag{2}$$

$$= \exp\big\{ y_1 f^1(x) + \cdots + y_K f^K(x) + \cdots + y_1 y_2 f^{1,2}(x) + \cdots + y_1 \ldots y_K f^{1,\ldots,K}(x) - b(f) \big\}$$

Here, we use the following notations. Let $\overline{\Psi}_K$ be the power set of $\Omega = \{1, \ldots, K\}$, and use $\Psi_K = \overline{\Psi}_K - \{\emptyset\}$ to index the $2^K - 1$ f^ω's in (2). Let ω denotes a set in Ψ_K, define $\mathcal{Y} = (y^1, \cdots, y^\omega, \cdots, y^\Omega)$ be the augmented response with $y^\omega = \prod_{i \in \omega} y_i$. And $f = (f^1, \ldots, f^\omega, \ldots, f^\Omega)$ is the vector of conditional log odds ratios (Gao et al., 2001). We assume f^ω is in a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}^\omega$ with kernel K^ω (Wahba, 1990). For example, in our simulation we choose f^ω to be B-spline (see Appendix). We focus on estimating the set of $f^\omega(x)$ with feature x where the sparsity in the set specifies the graph structure.

We present the following lemma and theorem which show the equivalence between UGM and MVB:

Lemma 2.1 *In a MVB model, define the odd-even partition of the power set of ω as:* $\Psi_{odd}^{\omega} = \{\kappa \subset \omega \mid |\kappa| = |\omega| - k, \text{where } k \text{ is odd}\}$, *and* $\Psi_{even}^{\omega} = \{\kappa \subset \omega \mid |\kappa| = |\omega| - k, \text{ where } k \text{ is even}\}$. *Note* $|\Psi_{odd}^{\omega}| = |\Psi_{even}^{\omega}| = 2^{|\omega|-1}$. *The following property holds:*

$$f^{\omega} = \log \frac{\prod_{\kappa \in \Psi_{even}^{\omega}} P(Y_i = 1, i \in \kappa; Y_j = 0, j \in \Omega - \kappa | X)}{\prod_{\kappa \in \Psi_{odd}^{\omega}} P(Y_i = 1, i \in \kappa; Y_j = 0, j \in \Omega - \kappa | X)}, \quad b(f) = \log \frac{Z(x)}{\prod_{C \in \mathcal{C}} \Phi_C(0; x)} \quad (3)$$

Theorem 2.2 *A UGM of the general form (1) with binary nodes is equivalent to a MVB model of (2). In addition, the following are equivalent:*

    1) There is no $|C|$-order interaction in $\{Y_i, i \in C\}$;

    2) There is no clique $C \in \Psi_K$ in the graph;

    3) $f^{\omega} = 0$ for all ω such that $C \subset \omega$.

A proof is given in Appendix A. It states that there is a clique C in the graph, iff there is $\omega \supset C, f^{\omega} \neq 0$ in MVB model. The advantage of modeling by MVB is that the sparsity in f^{ω}'s is sufficient and necessary for the conditional independence in the graph, thus fully specifying the graph structure. Specifically, Y_i, Y_j are conditionally independent iff $f^{\omega} = 0, \omega \supset \{i, j\}$. This showed the interaction is non-zero iff all the nodes involved are not conditionally independent.

Example 2.1 *When $K = 2$, $\Omega = \{1, 2\}, \mathcal{C} = \{\Omega\}$, denote $\Phi_{\Omega}(Y_1 = 1, Y_2 = 1; X)$ as Φ_{11} for simplicity, then $P(Y_1 = 1, Y_2 = 1|X) = \frac{1}{Z}\Phi_{11}$. Define $\Phi_{10}, \Phi_{01}, \Phi_{00}$ similarly, then the distribution with UGM parameterization is determined. The relation between UGM and MVB is*

$$f^1 = \log \frac{\Phi_{10}}{\Phi_{00}}, \quad f^2 = \log \frac{\Phi_{01}}{\Phi_{00}}, \quad f^{1,2} = \log \frac{\Phi_{11} \cdot \Phi_{00}}{\Phi_{01} \cdot \Phi_{10}}$$

Note, the independence between Y_1 and Y_2 implies:

$$f^{1,2} = 0 \quad or \quad \log \frac{\Phi_{11} \cdot \Phi_{00}}{\Phi_{01} \cdot \Phi_{10}} = 0$$

Therefore, $f^{1,2}$ being zero in MVB model is sufficient and necessary for the conditional independence in the model. On the other hand, $\log \Phi_C = 0$ is a sufficient condition but not necessary.

The distribution of a general discrete UGM where $Y_k \in \bar{V} = \{0, \cdots, m - 1\}$ can be extended from (2) naturally. Let $V = \{1, \ldots, m - 1\}, y_{\omega} = (y_i)_{i \in \omega}$, then

$$P(Y_1 = y_1, \cdots, Y_K = y_K | X) = \exp \Big\{ \sum_{\omega=1}^{\Omega} \sum_{v \in V^{|\omega|}} I(y_{\omega} = v) f_v^{\omega} - b(f) \Big\} \quad (4)$$

where I is an indicator function and V^n is the tensor product of n V's. Each f^{ω} is a $|V|^{|\omega|}$ vector.

3. Structure Penalty

In many applications, the assumption is that the graph has very few large cliques. Similar to the hierarchical inclusion restriction in Schmidt and Murphy (2010), we will include a higher order interaction only when all its subsets are included. Our model is very flexible in that $f^\omega(x)$ can be in an arbitrary RKHS. But in this section and next, we will discuss the special case where f^ω is linear for simplicity. The extension to non-linear or non-parametric models is straightforward.

Let $y(i) = (y_1(i), \ldots, y_K(i)), x(i) = (x_1(i), \ldots, x_p(i))$ be the ith data point. The augmented representation of the multivariate responses is:

$$\mathcal{Y}(i) = (y^1(i), \ldots, y^\omega(i), \ldots, y^\Omega(i)) \tag{5}$$

There are $|\Psi_K| = 2^K - 1$ functions in total. We first consider learning the full model when K is small, and later propose a greedy search algorithm to scale the method to large graphs. The general penalized log likelihood model is:

$$\min I_\lambda(f) = L(f) + \lambda J(f) \tag{6}$$
$$= \sum_{i=1}^{n} \left(-\mathcal{Y}(i)^T f(x(i)) + b(f) \right) + \lambda J(f)$$

where $L(f)$ is the negative log likelihood and $J(\cdot)$ is the structure penalty. The objective is to obtain sparse estimation of the cliques by the structure penalty on f. The hierarchical assumption is that if there is no interaction on clique C, then all f^ω should be zero, for $\omega \supset C$. The penalty is designed to shrink such f^ω toward zero. We consider the penalty guided by the lattice in Figure 2. The lattice T has $2^K - 1$ nodes: $1, \ldots, \omega, \ldots, \Omega$. There is an edge from ω_1 to ω_2 if and only if $\omega_1 \subset \omega_2$ and $|\omega_1| + 1 = |\omega_2|$. Jenatton *et al.* (2009) discussed how to define the groups to achieve different nonzero patterns.
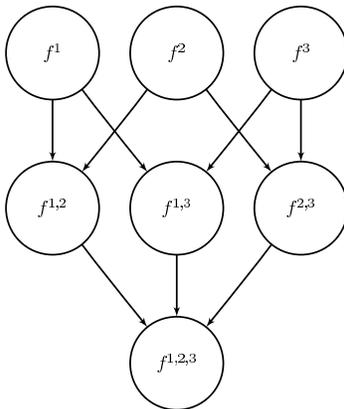

Figure 2: Hierarchical lattice for penalty

Let $T_v = \{\omega \in \Psi_K | v \subset \omega\}$ be the subgraph rooted at v in T, including all the descendants of v. Denote $f^{T_v} = (f^\omega)_{\omega \in T_v}$. All the functions are categorized into groups with

6

overlaps as $(T_1, \ldots, T_\Omega)$. The structure penalty on the group T_v is:

$$J(f^{T_v}) = p_v \sqrt{\sum_{\omega \in T_v} \|f^\omega\|_{\mathcal{H}^\omega}^2}$$

where p_v is the weight for the penalty on T_v, empirically chosen as $\frac{1}{|T_v|}$. Then, the objective is:

$$\min_f \quad I_\lambda(f) = L(f) + \lambda \sum_v p_v \sqrt{\sum_{\omega \in T_v} \|f^\omega\|_{\mathcal{H}^\omega}^2} \tag{7}$$

The following theorem shows that by minimizing the objective (7), f^{ω_1} will enter the model before f^{ω_2} if $\omega_1 \subset \omega_2$. That is to say, if f^{ω_1} is zero, there will be no higher order interactions on ω_2. It is an extension of Theorem 1 in Zhao $et\ al.$ (2009) and the proof is given in Appendix A.

Theorem 3.1 *Objective (7) is convex, thus the minimal is attainable. Let $\omega_1, \omega_2 \in \Psi_K$ and $\omega_1 \subset \omega_2$. If $\hat{f}$ is the minimizer of (7) given the observations, that is, $0 \in \partial I_\lambda(\hat{f})$ which is the subgradient of I_λ at $\hat{f}$, then $\hat{f}^{\omega_2} = 0$ almost surely if $\hat{f}^{\omega_1} = 0$.*

Example 3.1 *If $K = 3$, $f = (f^1, f^2, f^3, f^{1,2}, f^{1,3}, f^{2,3}, f^{1,2,3})$. The group at node 1 in Figure 2 is $f^{T_1} = (f^1, f^{1,2}, f^{1,3}, f^{1,2,3})$ and $J(f^{T_1}) = p_1 \sqrt{\|f^1\|^2 + \|f^{1,2}\|^2 + \|f^{1,3}\|^2 + \|f^{1,2,3}\|^2}$.*

4. Parameter Estimation

In this section, we discuss parameter estimation where the ωth function space is linear as $\mathcal{H}^\omega = \{1\} \oplus \mathcal{H}_1^\omega$. $\{1\}$ refers to the constant function space, and $\mathcal{H}_1^\omega$ is a RKHS with a linear kernel. The function $f^\omega \in \mathcal{H}^\omega$ has the form $f^\omega(x) = c_0^\omega + \sum_{j=1}^p c_j^\omega x_j$. Its norm is $\|f^\omega\|_{\mathcal{H}^\omega}^2 = \|c^\omega\|^2$, where $\|\cdot\|$ stands for Euclidean l_2 norm. Here, we denote $c^\omega = (c_0^\omega, \ldots, c_p^\omega)^T \in \mathbb{R}^{p+1}$ as a vector of length $p + 1$ and $c = (c^\omega)_{\omega \in \Psi_K} \in \mathbb{R}^{\tilde{p}}$ is the concatenated vector of all parameters of length $\tilde{p} = (p+1) \cdot |\Psi_K|$. Hence, the objective (7) is now:

$$\min_c \quad I_\lambda(c) = L(c) + \lambda \sum_v p_v \|c^{T_v}\| \tag{8}$$

where $c^{T_v} = (c^\omega)_{\omega \in T_v}$ is a $(p+1) \cdot |T^v|$ vector.

4.1 Estimating the complete model on small graphs

Many applications do not involve a large amount of responses, e.g., the Census Bureau data in our experiment. It is desirable to learn the complete model when the graph is small for consistency reasons. We propose a method to optimize the objective (8) of the complete model with all interaction levels by iteratively solving the following proximal linearization problem as discussed in Wright (2010):

$$\min_c L_k + \nabla L_k^T(c - c_k) + \frac{\alpha_k}{2}\|c - c_k\|^2 + \lambda J(c) \tag{9}$$

Algorithm 1 Proximal Linearization Algorithm

Input: $c_0, \alpha_0, \zeta > 1, tol > 0$
repeat
    Choose $\alpha_k \in [\alpha_{min}, \alpha_{max}]$
    Solve Eq (9) for d_k
    while $\delta_k = I_\lambda(c_k) - I_\lambda(c_k + d_k) < \|d_k\|^3$ **do**
      // Insufficient decrease
      Set $\alpha_k = \max(\alpha_{min}, \zeta\alpha_k)$
      Solve Eq (9) for d_k
    end while
    Set $\alpha_{k+1} = \alpha_k/\zeta$
    Set $c_{k+1} = c_k + d_k$
until $\delta_k < tol$

---

where $L_k = L(c_k)$, and α_k is a positive scalar chosen adaptively at kth step. With slight abuse of notation, we use c_k to denote the value of c at kth step. Algorithm 1 summarized the framework of solving (8). Following the analysis in Wright (2010), we can ensure that the proximal linearization algorithm will converge for negative log-likelihood loss function plus group lasso type penalties with overlaps.

However, solving group lasso with overlaps is not trivial due to the non-smoothness at the singular point. In recent years, several papers have addressed this problem. Jacob *et al.* (2009) duplicated the variables in the design matrix that appear in group overlaps, then solved the problem as group lasso without overlaps. Kim and Xing (2010) reparameterized the group norm with additional dummy variables. They alternatively optimized the model parameters and the dummy variables at each step. It is efficient for the quadratic loss function on Gaussian data, but might not scale well in our case. Instead, we solve (9) by its smooth and convex dual problem (Liu and Ye, 2010; Mairal et al., 2010). The details are presented in the Appendix.

4.2 Estimating large graphs

The above algorithm is efficient on small graphs. It usually terminates within 20 iterations in our experiments. However, the issue of estimating a complete model is the exponential number of f^ω's and the same amount of groups involved in objective (8). It is intractable when the graph becomes large. The hierarchical assumption and the structure penalty lend themselves naturally to a greedy search algorithm:

1. Start from the set of main effects as $A_0 = \{f^1, \cdots, f^K\}$.

2. In step i, remove the nodes that are not in A_i from the lattice in Figure 2. Obtain a sparse estimation of the functions in A_i by algorithm (1). Denote the resulting sparse set A_i'.

3. Let $A_{i+1} = A_i'$. Keep adding a higher order interaction into A_{i+1} if all its subsets of interactions are included in A_i'. And also add this node into the lattice in Figure 2.

Iterate step 2 and 3 until convergence. The algorithm is similar to the active set method in Schmidt and Murphy (2010). It has multiple runs of algorithm (1) to enforce the hierarchical

8

assumption. It is not guaranteed to converge to the global optimum. Nonetheless, our empirical experiments show its ability to scale to large graphs.

5. Experiments

5.1 Toy Data

In the simulation, we create 6 toy graphs. The first four graphs are depicted in Figure 1. Graph 5 has 100 nodes where the first 8 nodes have the same structure as in Figure 1(c) and the other nodes are independent. Graph 6 also has 100 nodes where the first 10 nodes have the same connection as in Figure 1(d) and the others are independent. We generate 100 datasets for each graph structure in Figure 1 to evaluate the learning accuracy. The sample size in each dataset is 1000. Here is how the first data set is generated:

The length of the feature vector, p, is set to 5 in our experiment, i.e. $X = (X_1, \ldots, X_5)$. Each $f^\omega(x) = c_0^\omega + \sum_{j=1}^{5} g_j^\omega(x_j)$ where $g_j^\omega(x_j) = \sum_{k=1}^{D} c_{jk}^\omega B_k(x_j)$ is spanned by the B-spline basis functions $\{B_k(\cdot)\}_{k=1,\cdots,D}$ (see the Appendix for details). The number of basis functions, D, is chosen to be 5. The true set of the model parameters, c_{jk}^ω, $j = 1, \cdots, 5, k = 1, \cdots, 5$, is uniformly sampled from $\{-5, -4, \cdots, 5\}$. We set the intercepts c_0^ω in main effects to 1, and those in second or higher order interactions to 2. Then 1000 samples of Y are generated as follows: For each sample, each component of the feature vector $X_j, j = 1, \ldots, 5$ is independently generated from a uniform distribution on [-1, 1]. Then each Y is sampled according to the probability in equation (2) (Gibbs sampling is applied in graphs 5 and 6), and the model is fit on the 1000 samples. To get the second and each subsequent data set, a new true set of the c's is generated and the 1000 samples of Y are regenerated according to the probability in equation (2), after generating a new X for each sample.

We use GACV (generalized approximate cross validation) (Xiang and Wahba, 1996; Ma, 2010) and BGACV (B-type GACV) (Shi et al., 2008) to choose the regularization parameter λ for the complete model (graphs 1-4). We use AIC for greedy search in graphs 5 and 6 due to computational consideration. The range of λ is chosen according to Koh *et al.* (2007). The details of the tuning methods are discussed in the Appendix. The R package, BMN, is used as a baseline (Höfling and Tibshirani, 2009).

In Table 2, we count, for each function f^ω, the number of runs out of 100 where f^ω is recovered ($\|c^\omega\| \neq 0$). If a recovered function is in the true model, it is considered a true positive, otherwise a false positive. The main effects are always detected correctly, thus are not listed in the table. Our structure penalty is more effective compared to BMN which only considers pairwise interactions.

In Figure 3, we show the learning results in terms of true positive rate (TPR) as sample size increases from 100 to 1000. The experimental setting is the same as before. The TPRs improve with increasing sample size. Compared to graphs 1 and 2, the algorithm needs substantially larger sample size to achieve high TPR on graphs 3-6. GACV achieves better TPR, but higher FPR compared to BGACV. Our method outperforms BMN in all six graphs.

Table 2: Number of true positive and false positive functions

Graph	Method	$f^{1,2}$	$f^{1,3}$	$f^{2,3}$	$f^{3,4}$	$f^{1,2,3}$	$f^{5,7,8}$	$f^{5,6,7,8}$	FP
1	BMN	60	76	70	60	0	-	-	162
	Complete-GACV	100	100	100	94	84	-	-	136
	Complete-BGACV	86	83	83	72	14	-	-	11
2	BMN	44	50	38	58	0	-	-	412
	Complete-GACV	100	99	100	99	83	-	-	341
	Complete-BGACV	88	91	88	78	33	-	-	64
3	BMN	72	64	60	60	0	0	0	830
	Complete-GACV	91	87	81	92	62	71	33	412
	Complete-BGACV	36	22	23	93	0	39	0	162
4	BMN	48	34	37	29	0	0	-	774
	Complete-GACV	92	98	94	90	54	45	-	693
	Complete-BGACV	68	68	71	62	0	0	-	144
5	BMN	38	28	26	22	0	0	0	9476
	Greedy-AIC	99	99	98	97	22	21	0	1997
6	BMN	28	26	14	26	0	0	-	9672
	Greedy-AIC	100	100	100	99	24	15	-	3458

5.2 Case Study: Census Bureau County Data

We use the county data from U.S. Census Bureau[2] to validate our method. The data includes demographic information for all counties in the United States, covering population, employment, votes (Scammon et al., 2005), etc. We delete the counties that have missing values in the columns we are interested in to obtain 2668 entries in total. The outcomes of this study are summarized in Table 3. "Vote" is coded as 1 if the Republican candidate won in the 2004 presidential election. To dichotomize the remaining response variables, the national mean is selected as a threshold. The third column in the table gives the percentage of positive subjects (counties). The data is standardized to mean 0 and variance 1. The following features are included in the model: Housing unit change in percent from 2000-2006, percent of each of three ethnic groups (White, African, Asian), percent foreign born, percent people over 65, percent people under 18, percent people with a high school education, percent people with a bachelors degree; birth rate, death rate, per capita government expenditure in dollars.

Adjusting the regularization parameter λ from 0.3289 to 0.1389, we observe new interactions enter the model gradually. The graph structure of $\lambda = 0.1559$ is shown in Figure 4(a). The first number on each edge is the order when this edge enters the model, and the second number is the corresponding λ. The unemployment rate plays an important role as a hub in the graph. It is strongly related to poverty and crimes. Population change is negatively related to violent crimes, as well as to unemployment rate. In Figure 4(b) obtained by BMN, unemployment rate is not a hub.

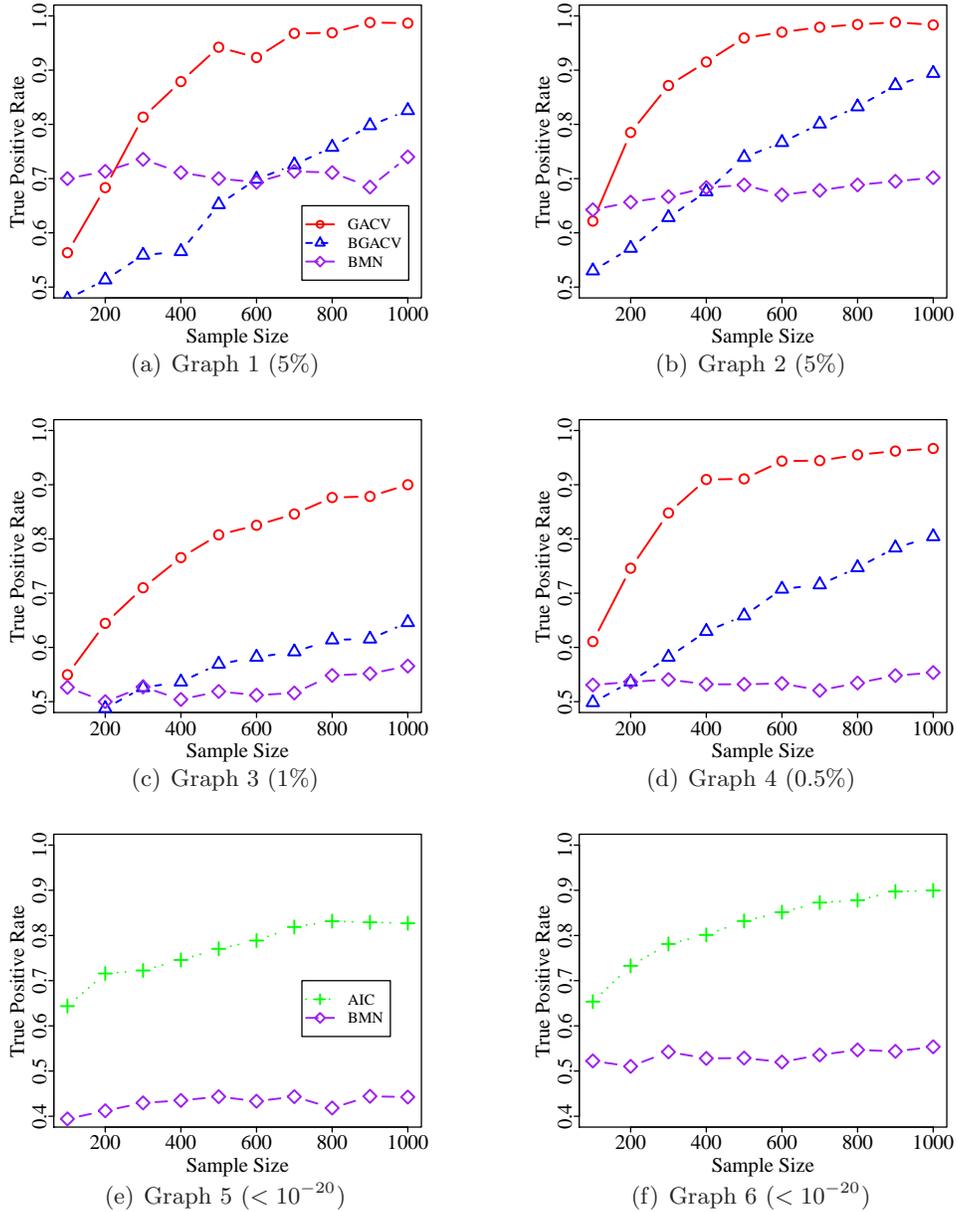2. http://www.census.gov/statab/www/ccdb.html

Figure 3: The True Positive Rate (TPR) of graph structure learning with increasing sample size by different methods. The percentage in the bracket is the upper bound of False Positive Rate (FPR) in each experiment. BMN always has larger FPR compared to GACV/BGACV/AIC.

We analyze the link between "Vote" and "PChange", which is recovered by our method before any other links. The marginal correlation between "Vote" and "PChange" (without X) is only 0.0389, which is the second smallest absolute value in the correlation matrix.

11

Table 3: Selected response variables

Response	Description	Positive%
Vote	2004 votes for Republican presidential candidate	81.11
Poverty	Poverty Rate	52.70
VCrime	Violent Crime Rate, eg. murder, robbery	23.09
PCrime	Property Crime Rate, eg. burglary	6.82
URate	Unemployment Rate	51.35
PChange	Population change in percent from 2000 to 2006	64.96

However, this link becomes the first one discovered by our method. It has been suggested that there is indeed a connection between population change and voting behavior[3]. This shows that after taking features into account, the dependence structure of response variables may change and hidden relations could be discovered. The main factors in this case are "percentage of housing unit change" (X_1) and "population percentage of people over 65" (X_2). The part of the fitted model shown below suggests that as housing units increase, the counties are more likely to have both positive results for "Vote" and "PChange". But this tendency will be counteracted by the increase of people over 65: the responses are less likely to take both positive values.

$$\hat{f}^{Vote} = 0.0463 \cdot X_1 + 0.0877 \cdot X_2 + \cdots$$
$$\hat{f}^{PChange} = 0.2315 \cdot X_1 - 0.0942 \cdot X_2 + \cdots$$
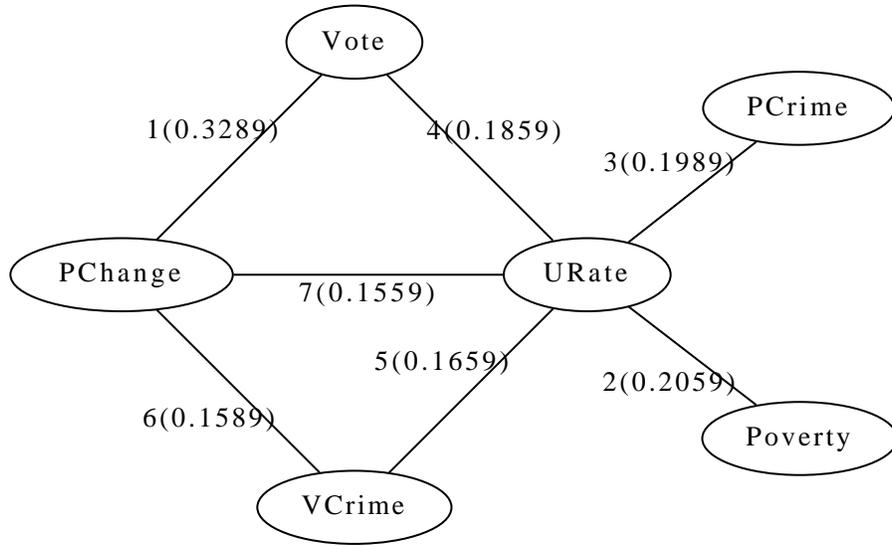$$\hat{f}^{Vote,PChange} = 0.0211 \cdot X_1 - 0.0115 \cdot X_2 + \cdots$$

6. Conclusions and Future Work

Our method can learn the graph structure that is specified by the conditonal log odds ratios conditioned on input features X. An efficient algorithm is given to estimate the complete model. A greedy search is applied when the graph is large. This method can be extended to model a general discrete UGM, where Y_k takes value in $\{0, \ldots, m-1\}$. Another extension will be the selection of features for each f^ω, that is the sparsity within each group. But the graph structure would change with different selected features. We will also be interested in investigating the consistency property of the proposed method.
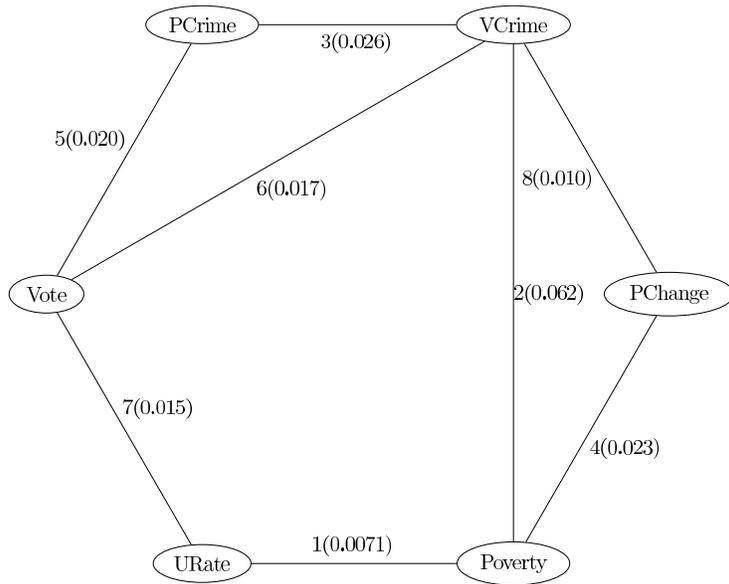
Acknowledgments

---

3. http://www.ipsos-mori.com/researchpublications/researcharchive/2545/Analysis-Population-change-turnout-the-election.aspx

(a) Complete

(b) BMN

Figure 4: Interactions of response variables in the Census Bureau data. The first number on the edge is the order at which the link is recovered. The number in bracket is the corresponding λ.

Appendix A. Proof

A.1 Proof of Theorem 2.2

Proof Given UGM (1), the corresponding parameterization in MVB model is shown in (3) of Lemma 2.1. Conversely, given the MVB model of (2), the cliques can be determined by the nonzero f^ω: clique C exists if $C = \omega$ and $f^\omega \neq 0$. Then the maximal cliques can be inferred from the graph structure. And suppose they are $C_1, \ldots, C_m$. Let $\omega_i = C_i$, for $i = 1, \ldots, m$, and $\kappa_1 = \emptyset$, $\kappa_i = C_i \cap (C_{i-1} \cup \cdots \cup C_1)$, $i = 2, \ldots, m$. Then the parameterization is:

$$\Phi_{C_i}(y_{C_i}; x) = \exp\left(S^{\omega_i}(y; x) - S^{\kappa_i}(y; x)\right) \quad \text{and} \quad Z(x) = \exp(b(f)) \tag{10}$$

where $S^\omega(y; x) = \sum_{\kappa \subset \omega} y^\kappa f^\kappa(x)$. Thus, UGM (1) with bivariate nodes is equivalent to MVB (2).

In the latter part of the theorem, $1 \Rightarrow 2$ and $3 \Rightarrow 1$ follow naturally from the Markov property of graphical models. To show $2 \Rightarrow 3$, let y_C^ω be a realization of y_C such that $y_C^\omega = (y_i^\omega)_{i \in C}$ where $y_i^\omega = 1$ if $i \in \omega$ and $y_i^\omega = 0$ otherwise. Notice that whenever $\kappa \cap C = \kappa' \cap C$, we have $y_C^\kappa = y_C^{\kappa'}$. For any possible $v = \kappa \cap C$, $\kappa' \in \{\kappa | \kappa = v \cup u, \text{ s.t. } u \subset \omega - v\}$ will satisfy the condition: $\kappa' \cap C = v$. There are $2^{|\omega - v|}$ such κ' in total due to the choice of u. Also, they appear in the nominator and denominator of equation (3) equally. So, for any $C \in \mathcal{C}$,

$$\prod_{\kappa \in \Psi_{even}^\omega} \Phi_C(y_C^\kappa; x) = \prod_{\kappa \in \Psi_{odd}^\omega} \Phi_C(y_C^\kappa; x) \tag{11}$$

It follows that $f^\omega = 0$ by (3). ∎

A.2 Proof of Theorem 3.1

Proof We give the proof for the linear case. The convexity of I_λ is easy to check, since L and $J(f^{T_v})$ are all convex in c. Suppose there is some $\omega_2 \supset \omega_1$ s.t. $\hat{c}^{\omega_2} \neq 0$ and $\hat{c}^{\omega_1} = 0$, by the groups constructed through Figure 2, $\|\hat{c}^{T_v}\| = \|(\hat{c}^\omega)_{v \subset \omega}\| \neq 0$ for all $v \subset \omega_1$. So the partial derivative of the objective (8) with respect to c^{ω_1} at $\hat{c}^{\omega_1}$ is

$$\left.\frac{\partial L}{\partial c^{\omega_1}}\right|_{c^{\omega_1} = \hat{c}^{\omega_1}} + \lambda \sum_{v \subset \omega_1} p_v \frac{\hat{c}^{\omega_1}}{\|\hat{c}^{T_v}\|} = 0 \tag{12}$$

Thus, the probability of $\{\hat{c}^{\omega_2} \neq 0\}$ equals to the probability of $\{\left.\frac{\partial L}{\partial c^{\omega_1}}\right|_{c^{\omega_1} = \hat{c}^{\omega_1}} = 0\}$, which is 0. ∎

Appendix B. Multivariate Bernoulli model

The multivariate Bernoulli (MVB) model of K random variables has $2^K - 1$ natural parameters (Whittaker, 1990). Given the predictive variable X, these parameters are functions

of X, called conditional log odds ratios. From the distribution of the MVB, $f^\omega(X)$ can be written as:

$$f^\omega(x) = \log OR(Y_i, i \in \omega | Y_j = 0, j \notin \omega; X = x) \tag{13}$$

Here, the odds ratios are calculated recursively

$$OR(Y_i | X = x) = \frac{P(Y_i = 1 | X = x)}{1 - P(Y_i = 1 | X = x)}, \tag{14}$$

$$OR(Y_i, i \in \omega \cup \{k\} | X = x) = \frac{OR(Y_i, i \in \omega | Y_k = 1, X = x)}{OR(Y_i, i \in \omega | Y_k = 0, X = x)}, \text{ with } k \notin \omega \tag{15}$$

The following two notations are useful in optimization and parameter tuning:

$$S^\omega(y; x) = \sum_{\kappa \subset \omega} y^\kappa f^\kappa(x); \quad S^\omega(x) = \sum_{\kappa \subset \omega} f^\kappa(x); \tag{16}$$

It follows from the definition of the conditional log odds ratio in (13) that

$$\exp(S^\omega(x)) = \frac{P(Y_i = 1, i \in \omega, \text{and } Y_j = 0, j \in \Omega - \omega | X = x)}{P(Y_i = 0, i \in \Omega | X = x)} \tag{17}$$

Then the normalization factor is:

$$\exp(b(f(x))) = 1 + \sum_{\omega \in \Psi_K} \exp(S^\omega(x)) \tag{18}$$

In practice, the $\exp(b(f(x)))$ is calculated by the junction tree algorithm to avoid enumerating 2^K possible values of Y, which is intractable in large graphs.

Appendix C. Dual of the proximal linearization problem

To solve the following objective of the proximal linearization problem

$$\min_c L_k + \nabla L_k^T(c - c_k) + \frac{\alpha_k}{2} \|c - c_k\|^2 + \lambda J(c) \tag{19}$$

we solve its dual problem as suggested in Liu and Ye (2010). Let $Z = \{v \in \Psi_K | \|c^{T_v}\| = 0\}$, and $\bar{Z} = \Psi_K - Z$ be the complement. Define $s_v, v \in \Psi_K$ as:

$$s_v \in \mathbb{S}_v = \{s = (s^\omega)_{\omega \in \Psi_K} \mid s \in \mathbb{R}^{\tilde{p}}, \|s\| \le \lambda p_v, s^\omega = 0 \text{ if } \omega \in T_v\} \tag{20}$$

then the subgradient of (9) is:

$$\nabla L + \alpha_k(c - c_k) + \sum_{v \in Z} s_v + \sum_{u \in \bar{Z}} r_u \tag{21}$$

where s_v is the subgradient of $\lambda p_v \|c^{T_v}\|$ for $v \in Z$ and r_u is the subgradient for $u \in \bar{Z}$:

$$r_u = \arg\max_{s_u} \langle s_u, c \rangle, \text{ for } u \in \bar{Z} \tag{22}$$

15

The subgradient s_v is in a unit ball of certain subspace of $\mathbb{R}^{\tilde{p}}$. These subspaces are not perpendicular to each other. Thus, s_v's are not separable, and closed form solution of (9) cannot be obtained. We solve the proximal subproblem (9) by its smoothing and convex dual problem. Note (9) is equivalent to:

$$\min_{c \in \mathbb{R}^{\tilde{p}}} \max_{S \in \mathbb{S}} \phi(c, S) = \nabla L_k^T (c - c_k) + \frac{\alpha_k}{2} \|c - c_k\|^2 + \sum_{v \in \Omega} \langle s_v, c \rangle \tag{23}$$

where S is a $\tilde{p} \times |\Psi_K|$ matrix whose columns are s_v. $\mathbb{S} = \{S | S = (s_1, \ldots, s_v, \ldots, s_\Omega), s_v \in \mathbb{S}_v$ for $v \in \Psi_K \}$ is the feasible region of S. Since $\phi(\cdot, S)$ is lower semicontinuous and $\phi(c, \cdot)$ is upper semicontinuous, there exists a saddle point and the max and min are exchangeable. The solution of minimizing $\phi(c, S)$ is:

$$\tilde{c} = \arg\min_c \phi(c, S) = c_k - \frac{1}{\alpha_k} \nabla L_k - \frac{1}{\alpha_k} \sum_v s_v \tag{24}$$

Substitute $\tilde{c}$ back into (23), we have the dual problem of (9) as:

$$\max_{S \in \mathbb{S}} \eta(S) = -\frac{1}{2} \| \sum_v s_v \|^2 + (\alpha_k c_k - \nabla L_k)^T \sum_v s_v \tag{25}$$

Following the proof in Liu and Ye (2010), we can show that $\eta(S)$ is convex and Lipschitz continuous. The differential is $\alpha_k \tilde{c} e^T$ where $e \in \mathbb{R}^{\tilde{p}}$ is a vector of ones. Hence, (25) can be solved by existing gradient methods.

Appendix D. B-spline

Given m knots, $t_0 \leq t_1 \leq \cdots \leq t_{m-1}$, the B-spline basis functions of degree d are defined recursively (De Boor, 1978):

$$b_{k,0} = \begin{cases} 1; & \text{if } t_k \leq t < t_{k+1} \\ 0; & \text{otherwise} \end{cases} \quad, \text{ for } k = 0, \cdots, m - 2$$

$$b_{k,l} = \frac{t - t_k}{t_{k+l} - t_k} b_{k,l-1}(t) + \frac{t_{k+l+1} - t}{t_{k+l+1} - t_{k+1}} b_{k+1,l-1}(t), \text{ for } k = 0, \cdots, m - d - 2; l = 0, \cdots, d$$

Let $B_k(\cdot) = b_{k,d}(\cdot)$, then $\{B_k, k = 0, \cdots, m - d - 2\}$ are $m - d - 1$ basis functions, which span the functional space $\mathcal{B}$. The B-spline curve in $\mathcal{B}$ is:

$$g(t) = \sum_{k=0}^{m-d-2} c_k B_k(t) \tag{26}$$

where c_k's are the control points to be estimated. In our simulation studies, c_k's are assumed to be one dimensional scalers for simplicity.

We let each $f^\omega(x)$ where $x = (x_1, \cdots, x_p)'$ be in $\mathcal{B}_0 \oplus \mathcal{B}_1 \oplus \cdots \oplus \mathcal{B}_p$. Here, $\mathcal{B}_0$ is a space of constant functions and $\mathcal{B}_j, j = 1 \cdots, p$ is a B-spline functional space on domain $x_j \in \mathcal{X}_j$. Therefore,

$$f^\omega(x) = c_0^\omega + \sum_{j=1}^{p} g_j(x_j) \tag{27}$$

where $g_j(x_j) \in \mathcal{B}_j$ are defined in (26).

Appendix E. Tuning

For i-th data point $(y(i), x(i))$, denote $S_i^\omega = S^\omega(x(i))$, then the normalization factor of the i-th data is $b_i = b(f(x(i))) = \log\left(1 + \sum_\omega \exp S_i^\omega\right)$. The mean of the augmented response $\mathcal{Y}(i)$ in the MVB model is:

$$\mu(i) = E[\mathcal{Y}(i)|x(i), f] = (\mu^1(i), \cdots, \mu^\kappa(i), \cdots, \mu^\Omega(i)) \tag{28}$$

$$\text{where } \mu^\kappa(i) = \frac{\partial b_i}{\partial f^\kappa} = \frac{\sum_{\omega \in T_\kappa} \exp S_i^\omega}{\exp b_i} \tag{29}$$

The $|\Psi_K| \times |\Psi_K|$ covariance matrix of the augmented response is:

$$W(i) = var(\mathcal{Y}(i)|x(i), f) \tag{30}$$

where the (α, β)-th element of $W(i)$ is:

$$W_{\alpha,\beta}(i) = \frac{\partial^2 b_i}{\partial f^\alpha (\partial f^\beta)^T} = \frac{\sum_{\omega \in T_\alpha \cap T_\beta} \exp S_i^\omega}{\exp b_i} - \mu^\alpha(i) \cdot \mu^\beta(i) \tag{31}$$

Let R_v be a $\tilde{p} \times \tilde{p}$ diagonal matrix whose (i,i)-th element is 1 if $c_i \neq 0$. Then, the v-th group penalty $J(f^{T_v})$ can be written as:

$$J(f^{T_v}) = p_v \sqrt{\sum_{\omega \in T_v} \|f^\omega\|^2} = p_v \|R_v c\| \tag{32}$$

Note R_v is symmetric and $R_v \cdot R_v = R_v$, direct calculation yields the derivative and Hessian of the penalty term:

$$\frac{\partial J}{\partial c} = \sum_{v:R_v c \neq 0} p_v \frac{R_v c}{\|R_v c\|} \tag{33}$$

$$\frac{\partial^2 J}{\partial c \partial c^T} = \sum_{v:R_v c \neq 0} p_v J_v = \sum_{v:R_v c \neq 0} p_v \frac{R_v (\|R_v c\|^2 I - c \cdot c^T) R_v}{\|R_v c\|^3} \tag{34}$$

where $J_v \doteq (R_v(\|R_v c\|^2 I - c \cdot c^T) R_v)/\|R_v c\|^3$. Denote the grand design matrix as:

$$D = \left(D(1)^T \quad \cdots \quad D(n)^T\right)^T \tag{35}$$

$$\text{where } D(i) = \begin{pmatrix} x(i)^T & 0 & \cdots & 0 \\ 0 & x(i)^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x(i)^T \end{pmatrix} \tag{36}$$

Suppose there are N non-zero elements of c at location $\{a_1, \ldots, a_N\}$. Let $\tilde{D}$ be the matrix composed by the $a_1, \ldots, a_N$th column of D. Then, the Hessian matrix of I_λ is:

$$\frac{\partial^2 I_\lambda}{\partial c \partial c^T} = \frac{\partial^2 L}{\partial c \partial c^T} + \lambda \frac{\partial^2 J}{\partial c \partial c^T} = \tilde{D}^T W \tilde{D} + \lambda \sum_{v:R_v c \neq 0} p_v J_v \tag{37}$$

Let H be the $n|\Psi_K| \times n|\Psi_K|$ influence matrix that implies

$$f_{\lambda,\epsilon} - f_\lambda \approx H\epsilon \tag{38}$$

where ϵ is a small perturbation on $\mathcal{Y}$; $f_\lambda = Dc_\lambda$ is the estimated function value with tuning parameter λ; and $f_{\lambda,\epsilon}$ is the estimated function value with the perturbation. Then, the analysis of the first order Taylor expansion of $\frac{\partial I_\lambda}{\partial c}(\mathcal{Y} + \epsilon, c_{\lambda,\epsilon})$ leads to the formulation of H as follows (refer to Xiang and Wahba (1996) and Ma (2010) Chapter 3 for more details)

$$H = \tilde{D}\left(\frac{\partial^2 I_\lambda}{\partial c \partial c^T}\right)^{-1}\tilde{D}^T = \tilde{D}\left(\tilde{D}^T W \tilde{D} + \lambda \sum_{v:R_v c \neq 0} p_v J_v\right)^{-1}\tilde{D}^T \tag{39}$$

The (i,j)-th $q \times q$ submatrix of H is

$$H(i,j) = \tilde{D}(i)^T\left(\frac{\partial^2 I_\lambda}{\partial c \partial c^T}\right)^{-1}\tilde{D}(j) \tag{40}$$

Let $Q(i) = I - H(i,i)W(i)$ for $i = 1, \ldots, n$, define the generalized average matrix, denoted as $\bar{Q}$, of $\{Q(i), i = 1, \ldots, n\}$ as follows

$$\bar{Q} = (\delta - \gamma)I_{q \times q} + \gamma \cdot ee^T = \begin{pmatrix} \delta & \gamma & \cdots & \gamma \\ \gamma & \delta & \cdots & \gamma \\ \vdots & \vdots & \ddots & \vdots \\ \gamma & \gamma & \cdots & \delta \end{pmatrix} \tag{41}$$

where e is the unit vector of length q and

$$\delta = \frac{1}{nq\sum_{i=1}^n tr(Q(i))}, \quad \gamma = \frac{1}{nq(q-1)}\left[e^T Q(i)e - tr(Q(i))\right] \tag{42}$$

Let $\bar{H}$ be the generalized average of $\{H(i,i), i = 1, \cdots, n\}$, the GACV score is

$$GACV(\lambda) = OBS(\lambda) + \frac{1}{n}\sum_{i=1}^n \mathcal{Y}(i)^T \bar{Q}^{-1}\bar{H}\big(\mathcal{Y}(i) - \mu(i)\big) \tag{43}$$

where

$$OBS(\lambda) = \frac{1}{n}\left[-\mathcal{Y}(i)^T f_\lambda(x(i)) + b(f_\lambda(x(i)))\right] \tag{44}$$

is the observed log-likelihood.

The degrees of freedom of multivariate Bernoulli data is generally difficult to obtain. But we can have a good approximation from GACV (Shi et al., 2008) as

$$\hat{df}(\lambda) = \sum_{i=1}^n \mathcal{Y}(i)^T \bar{Q}^{-1}\bar{H}\big(\mathcal{Y}(i) - \mu(i)\big) \tag{45}$$

So the BGACV score can be defined as

$$BGACV(\lambda) = OBS(\lambda) + \frac{1}{n}\frac{\log n}{2}\sum_{i=1}^n \mathcal{Y}(i)^T \bar{Q}^{-1}\bar{H}\big(\mathcal{Y}(i) - \mu(i)\big) \tag{46}$$

For the model selection criteria AIC, the degree of freedom is approximated by the number of non-zero c_{jk}'s in the group penalty.

18

References

O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516, 2008. ISSN 1532-4435.

J.K. Bradley and C. Guestrin. Learning tree conditional random fields. In *Proceedings of the 27th International Conference on Machine learning*, pages 127–134, 2010.

C. De Boor. *A practical guide to splines.* Applied Mathematical Sciences, 1978.

F. Gao, G. Wahba, R. Klein, and B. Klein. Smoothing Spline ANOVA for multivariate Bernoulli observations, with application to ophthalmology data. *Journal of the American Statistical Association*, 96(453):127, 2001.

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

L. Jacob, G. Obozinski, and J.P. Vert. Group Lasso with overlap and graph Lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 433–440, 2009.

R. Jenatton, J.Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. *arXiv:0904.3523*, 2009.

S. Kim and E.P. Xing. Tree-guided group lasso for multi-task regression with structured sparsity. In *Proceedings of 27th International Conference on Machine Learning*, pages 543–550, Haifa, Israel, 2010.

K. Koh, S.J. Kim, and S. Boyd. An interior-point method for large-scale l1-regularized logistic regression. *Journal of Machine learning research*, 8(8):1519–1555, 2007.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques.* The MIT Press, 2009. ISBN 0262013193.

J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289, 2001.

H. Liu, X. Chen, J. Lafferty, and L. Wasserman. Graph-valued regression. *arXiv:1006.3972*, 2010.

J. Liu and J. Ye. Fast overlapping group lasso. *arXiv:1009.0306v1*, 2010.

Xiwen Ma. *Penalized Regression in Reproducing Kernel Hilbert Spaces With Randomized Covariate Data.* PhD thesis, Department of Statistics, University of Wisconsin-Madison, 2010.

Julien Mairal, Rodolphe Jenatton, Guillaume Obozinski, and Francis Bach. Network flow algorithms for structured sparsity. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1558–1566. 2010.

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

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

P. Ravikumar, M.J. Wainwright, and J. Lafferty. High-dimensional Ising model selection using l1-regularized logistic regression. *Annals of Statistics*, 38(3):1287–1319, 2010.

R.M. Scammon, A.V. McGillivray, and R. Cook. *America Votes 26: 2003-2004, Election Returns By State*. CQ Press, 2005. ISBN 9781568029740.

M. Schmidt and K. Murphy. Convex structure learning in log-linear models: Beyond pairwise potentials. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

M. Schmidt, K. Murphy, G. Fung, and R. Rosales. Structure learning in random fields for heart motion abnormality detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

P. Schnitzspan, M. Fritz, S. Roth, and B. Schiele. Discriminative structure learning of hierarchical representations for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2238–2245, 2009.

W. Shi, G. Wahba, S. Wright, K. Lee, R. Klein, and B. Klein. LASSO-Patternsearch algorithm with application to ophthalmology and genomic data. *Statistics and its Interface*, 1(1):137, 2008.

R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:1068–1080, June 2007. ISSN 0162-8828.

G. Wahba. *Spline Models for Observational Data*. Society for Industrial Mathematics, 1990.

M.J. Wainwright and M.I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1:1–305, 2008. ISSN 1935-8237.

J. Whittaker. *Graphical models in applied multivariate statistics*. Wiley (Chichester England and New York), 1990.

S.J. Wright. Accelerated block-coordinate relaxation for regularized optimization. Technical report, Department of Computer Science, University of Wisconsin-Madison, 2010.

D. Xiang and G. Wahba. A generalized approximate cross validation for smoothing splines with non-Gaussian data. *Statistica Sinica*, 6:675–692, 1996. ISSN 1017-0405.

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

P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. *Annals of Statistics*, 37(6A):3468–3497, 2009.