Graph-Valued Regression

Han Liu Xi Chen John Lafferty Larry Wasserman

Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Undirected graphical models encode in a graph G the dependency structure of a random vector Y. In many applications, it is of interest to model Y given another random vector X as input. We refer to the problem of estimating the graph $G(x)$ of Y conditioned on $X = x$ as "graph-valued regression". In this paper, we propose a semiparametric method for estimating $G(x)$ that builds a tree on the X space just as in CART (classification and regression trees), but at each leaf of the tree estimates a graph. We call the method "Graph-optimized CART", or Go-CART. We study the theoretical properties of Go-CART using dyadic partitioning trees, establishing oracle inequalities on risk minimization and tree partition consistency. We also demonstrate the application of Go-CART to a meteorological dataset, showing how graph-valued regression can provide a useful tool for analyzing complex data.

1 Introduction

Let Y be a p-dimensional random vector with distribution P. A common way to study the structure of P is to construct the undirected graph $G = (V, E)$, where the vertex set V corresponds to the p components of the vector Y. The edge set E is a subset of the pairs of vertices, where an edge between Y_j and Y_k is absent if and only if Y_j is conditionally independent of Y_k given all the other variables. Suppose now that Y and X are both random vectors, and let $P(\cdot \mid X)$ denote the conditional distribution of Y given X. In a typical regression problem, we are interested in the conditional mean $\mu(x) = \mathbb{E}(Y \mid X = x)$. But if Y is multivariate, we may be also interested in how the structure of $P(\cdot \mid X)$ varies as a function of X. In particular, let $G(x)$ be the undirected graph corresponding to $P(\cdot \mid X = x)$. We refer to the problem of estimating $G(x)$ as *graph-valued regression*.

Let $\mathcal{G} = \{G(x) : \ x \in \mathcal{X}\}$ be a set of graphs indexed by $x \in \mathcal{X}$, where $\mathcal{X}$ is the domain of X. Then $\mathcal{G}$ induces a partition of $\mathcal{X}$, denoted as $\mathcal{X}_1, \ldots, \mathcal{X}_m$, where x_1 and x_2 lie in the same partition element if and only if $G(x_1) = G(x_2)$. Graph-valued regression is thus the problem of estimating the partition and estimating the graph within each partition element.

We present three different partition-based graph estimators; two that use global optimization, and one based on a greedy splitting procedure. One of the optimization based schemes uses penalized empirical risk minimization, the other uses held-out risk minimization. As we show, both methods enjoy strong theoretical properties under relatively weak assumptions; in particular, we establish oracle inequalities on the excess risk of the estimators, and tree partition consistency (under stronger assumptions) in Section 4. While the optimization based estimates are attractive, they do not scale well computationally when the input dimension is large. An alternative is to adapt the greedy algorithms of classical CART, as we describe in Section 3. In Section 5 we present experimental results on both synthetic data and a meteorological dataset, demonstrating how graph-valued regression can be an effective tool for analyzing high dimensional data with covariates.

2 Graph-Valued Regression

Let $y_1, \ldots, y_n$ be a random sample of vectors from P, where each $y_i \in \mathbb{R}^p$. We are interested in the case where p is large and, in fact, may diverge with n asymptotically. One way to estimate G from the sample is the *graphical lasso* or *glasso* [13, 5, 1], where one assumes that P is Gaussian with mean μ and covariance matrix Σ. Missing edges in the graph correspond to zero elements in the precision matrix $\Omega = \Sigma^{-1}$ [12, 4, 7]. A sparse estimate of Ω is obtained by solving

$$\widehat{\Omega} = \underset{\Omega \succ 0}{\arg\min} \big\{ \operatorname{tr}(S\Omega) - \log|\Omega| + \lambda \|\Omega\|_1 \big\} \tag{1}$$

where Ω is positive definite, S is the sample covariance matrix, and $\|\Omega\|_1 = \sum_{j,k} |\Omega_{jk}|$ is the elementwise ℓ_1-norm of Ω. A fast algorithm for finding $\widehat{\Omega}$ was given by Friedman et al. [5], which involves estimating a single row (and column) of Ω in each iteration by solving a lasso regression. The theoretical properties of $\widehat{\Omega}$ have been studied by Rothman et al. [10] and Ravikumar et al. [9]. In practice, it seems that the glasso yields reasonable graph estimators even if Y is not Gaussian; however, proving conditions under which this happens is an open problem.

We briefly mention three different strategies for estimating $G(x)$, the graph of Y conditioned on $X = x$, each of which builds upon the glasso.

Parametric Estimators. Assume that $Z = (X, Y)$ is jointly multivariate Gaussian with covariance matrix $\Sigma = \begin{pmatrix} \Sigma_X & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_Y \end{pmatrix}$. We can estimate Σ_X, Σ_Y, and Σ_{XY} by their corresponding sample quantities $\widehat{\Sigma}_X$, $\widehat{\Sigma}_Y$, and $\widehat{\Sigma}_{XY}$, and the marginal precision matrix of X, denoted as Ω_X, can be estimated using the glasso. The conditional distribution of Y given $X = x$ is obtained by standard Gaussian formulas. In particular, the conditional covariance matrix of $Y \,|\, X$ is $\widehat{\Sigma}_{Y|X} = \widehat{\Sigma}_Y - \widehat{\Sigma}_{YX}\widehat{\Omega}_X\widehat{\Sigma}_{XY}$ and a sparse estimate of $\widehat{\Omega}_{Y|X}$ can be obtained by directly plugging $\widehat{\Sigma}_{Y|X}$ into glasso. However, the estimated graph does not vary with different values of X.

Kernel Smoothing Estimators. We assume that Y given X is Gaussian, but without making any assumption about the marginal distribution of X. Thus $Y \,|\, X = x \sim N(\mu(x), \Sigma(x))$. Under the assumption that both $\mu(x)$ and $\Sigma(x)$ are smooth functions of x, we estimate $\Sigma(x)$ via kernel smoothing:

$$\widehat{\Sigma}(x) = \sum_{i=1}^{n} K\left(\frac{\|x - x_i\|}{h}\right) (y_i - \widehat{\mu}(x))\,(y_i - \widehat{\mu}(x))^T \Big/ \sum_{i=1}^{n} K\left(\frac{\|x - x_i\|}{h}\right)$$

where K is a kernel (e.g. the probability density function of the standard Gaussian distribution), $\|\cdot\|$ is the Euclidean norm, $h > 0$ is a bandwidth and

$$\widehat{\mu}(x) = \sum_{i=1}^{n} K\left(\frac{\|x - x_i\|}{h}\right) y_i \Big/ \sum_{i=1}^{n} K\left(\frac{\|x - x_i\|}{h}\right).$$

Now we apply glasso in (1) with $S = \widehat{\Sigma}(x)$ to obtain an estimate of $G(x)$. This method is appealing because it is simple and very similar to nonparametric regression smoothing; the method was analyzed for one-dimensional X in [14]. However, while it is easy to estimate $G(x)$ at any given x, it requires global smoothness of the mean and covariance functions.

Partition Estimators. In this approach, we partition $\mathcal{X}$ into finitely many connected regions $\mathcal{X}_1, \ldots, \mathcal{X}_m$. Within each $\mathcal{X}_j$, we apply the glasso to get an estimated graph $\widehat{G}_j$. We then take $\widehat{G}(x) = \widehat{G}_j$ for all $x \in \mathcal{X}_j$. To find the partition, we appeal to the idea used in CART (classification and regression trees) [3]. We take the partition elements to be recursively defined hyperrectangles. As is well-known, we can then represent the partition by a tree, where each leaf node corresponds to a single partition element. In CART, the leaves are associated with the means within each partition element; while in our case, there will be an estimated undirected graph for each leaf node. We refer to this method as Graph-optimized CART, or Go-CART. The remainder of this paper is devoted to the details of this method.

3 Graph-Optimized CART

Let $X \in \mathbb{R}^d$ and $Y \in \mathbb{R}^p$ be two random vectors, and let $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ be n i.i.d. samples from the joint distribution of (X, Y). The domains of X and Y are denoted by $\mathcal{X}$ and $\mathcal{Y}$ respectively;

and for simplicity we take $\mathcal{X} = [0,1]^d$. We assume that

$$Y \mid X = x \sim N_p(\mu(x), \Sigma(x))$$

where $\mu : \mathbb{R}^d \to \mathbb{R}^p$ is a vector-valued mean function and $\Sigma : \mathbb{R}^d \to \mathbb{R}^{p\times p}$ is a matrix-valued covariance function. We also assume that for each x, $\Omega(x) = \Sigma(x)^{-1}$ is a sparse matrix, i.e., many elements of $\Omega(x)$ are zero. In addition, $\Omega(x)$ may also be a sparse function of x, i.e., $\Omega(x) = \Omega(x_R)$ for some $R \subset \{1,\ldots,d\}$ with cardinality $|R| \ll d$. The task of graph-valued regression is to find a sparse inverse covariance $\widehat{\Omega}(x)$ to estimate $\Omega(x)$ for any $x \in \mathcal{X}$; in some situations the graph of $\Omega(x)$ is of greater interest than the entries of $\Omega(x)$ themselves.

Go-CART is a partition based conditional graph estimator. We partition $\mathcal{X}$ into finitely many connected regions $\mathcal{X}_1,\ldots,\mathcal{X}_m$, and within each $\mathcal{X}_j$ we apply the glasso to estimate a graph $\widehat{G}_j$. We then take $\widehat{G}(x) = \widehat{G}_j$ for all $x \in \mathcal{X}_j$. To find the partition, we restrict ourselves to dyadic splits, as studied by [11, 2]. The primary reason for such a choice is the computational and theoretical tractability of dyadic partition based estimators.

Let $\mathcal{T}$ denote the set of dyadic partitioning trees (DPTs) defined over $\mathcal{X} = [0,1]^d$, where each DPT $T \in \mathcal{T}$ is constructed by recursively dividing $\mathcal{X}$ by means of axis-orthogonal dyadic splits. Each node of a DPT corresponds to a hyperrectangle in $[0,1]^d$. If a node is associated to the hyperrectangle $\mathcal{A} = \prod_{l=1}^d [a_l, b_l]$, then after being dyadically split along dimension k, the two children are associated with the sub-hyperrectangles $\mathcal{A}_L^{(k)} = \prod_{l<k}[a_l,b_l] \times [a_k, \frac{a_k+b_k}{2}] \times \prod_{l>k}[a_l,b_l]$ and $\mathcal{A}_R^{(k)} = \mathcal{A}\backslash\mathcal{A}_L^{(k)}$. Given a DPT T, we denote by $\Pi(T) = \{\mathcal{X}_1,\ldots,\mathcal{X}_{m_T}\}$ the partition of $\mathcal{X}$ induced by the leaf nodes of T. For a dyadic integer $N = 2^K$, we define $\mathcal{T}_N$ to be the collection of all DPTs such that no partition has a side length smaller than 2^{-K}. Let $I(\cdot)$ denote the indicator function. We denote $\mu_T(x)$ and $\Omega_T(x)$ as the piecewise constant mean and precision functions associated with T:

$$\mu_T(x) = \sum_{j=1}^{m_T} \mu_{\mathcal{X}_j} \cdot I\left(x \in \mathcal{X}_j\right) \text{ and } \Omega_T(x) = \sum_{j=1}^{m_T} \Omega_{\mathcal{X}_j} \cdot I\left(x \in \mathcal{X}_j\right),$$

where $\mu_{\mathcal{X}_j} \in \mathbb{R}^p$ and $\Omega_{\mathcal{X}_j} \in \mathbb{R}^{p\times p}$ are the mean vector and precision matrix for $\mathcal{X}_j$.

Before formally defining our graph-valued regression estimators, we require some further definitions. Given a DPT T with an induced partition $\Pi(T) = \{\mathcal{X}_j\}_{j=1}^{m_T}$ and corresponding mean and precision functions $\mu_T(x)$ and $\Omega_T(x)$, the negative conditional log-likelihood risk $R(T,\mu_T,\Omega_T)$ and its sample version $\widehat{R}(T,\mu_T,\Omega_T)$ are defined as follows:

$$R(T,\mu_T,\Omega_T) = \sum_{j=1}^{m_T} \mathbb{E}\left[\left(\mathrm{tr}\left[\Omega_{\mathcal{X}_j}\left((Y-\mu_{\mathcal{X}_j})(Y-\mu_{\mathcal{X}_j})^T\right)\right] - \log|\Omega_{\mathcal{X}_j}|\right)\cdot I\left(X \in \mathcal{X}_j\right)\right], \quad (2)$$

$$\widehat{R}(T,\mu_T,\Omega_T) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m_T}\left[\left(\mathrm{tr}\left[\Omega_{\mathcal{X}_j}\left((y_i-\mu_{\mathcal{X}_j})(y_i-\mu_{\mathcal{X}_j})^T\right)\right] - \log|\Omega_{\mathcal{X}_j}|\right)\cdot I\left(x_i \in \mathcal{X}_j\right)\right]. \quad (3)$$

Let $[[T]] > 0$ denote a prefix code over all DPTs $T \in \mathcal{T}_N$ satisfying $\sum_{T\in\mathcal{T}_N} 2^{-[[T]]} \le 1$. One such prefix code $[[T]]$ is proposed in [11], and takes the form $[[T]] = 3|\Pi(T)| - 1 + (|\Pi(T)| - 1)\log d/\log 2$. A simple upper bound for $[[T]]$ is

$$[[T]] \le (3 + \log d/\log 2)|\Pi(T)|. \quad (4)$$

Our analysis will assume that the conditional means and precision matrices are bounded in the $\|\cdot\|_\infty$ and $\|\cdot\|_1$ norms; specifically we suppose there is a positive constant B and a sequence $L_{1,n},\ldots,L_{m_T,n}$, where each $L_{j,n} \in \mathbb{R}^+$ is a function of the sample size n, and we define the domains of each $\mu_{\mathcal{X}_j}$ and $\Omega_{\mathcal{X}_j}$ as

$$\begin{aligned} M_j &= \left\{\mu \in \mathbb{R}^p : \|\mu\|_\infty \le B\right\}, \\ \Lambda_j &= \left\{\Omega \in \mathbb{R}^{p\times p} : \Omega \text{ is positive definite, symmetric, and } \|\Omega\|_1 \le L_{j,n}\right\}. \end{aligned} \quad (5)$$

With this notation in place, we can now define two estimators.

Definition 1. *The **penalized empirical risk minimization Go-CART estimator** is defined as*

$$\widehat{T}, \left\{\widehat{\mu}_{\widehat{\mathcal{X}}_j}, \widehat{\Omega}_{\widehat{\mathcal{X}}_j}\right\}_{j=1}^{m_{\widehat{T}}} = \mathrm{argmin}_{T\in\mathcal{T}_N, \mu_{\mathcal{X}_j}\in M_j, \Omega_{\mathcal{X}_j}\in\Lambda_j}\left\{\widehat{R}(T,\mu_T,\Omega_T) + \mathrm{pen}(T)\right\}$$

where $\widehat{R}$ is defined in (3) *and* $\mathrm{pen}(T) = \gamma_n \cdot m_T\sqrt{\frac{[[T]]\log 2 + 2\log(np)}{n}}$.

Empirically, we may always set the dyadic integer N to be a reasonably large value; the regularization parameter γ_n is responsible for selecting a suitable DPT $T \in \mathcal{T}_N$.

We also formulate an estimator that minimizes held-out risk. Practically, we could split the data into two partitions: $\mathcal{D}_1 = \{(x_1, y_1), \ldots, (x_{n_1}, y_{n_1})\}$ for training and $\mathcal{D}_2 = \{((x'_1, y'_1), \ldots, (x'_{n_2}, y'_{n_2}))\}$ for validation with $n_1 + n_2 = n$. The held-out negative log-likelihood risk is then given by

$$\widehat{R}_{\text{out}}(T, \mu_T, \Omega_T) = \frac{1}{n_2} \sum_{i=1}^{n_2} \sum_{j=1}^{m_T} \Big\{ \Big(\text{tr}\Big[\Omega_{\mathcal{X}_j} \left((y'_i - \mu_{\mathcal{X}_j})(y'_i - \mu_{\mathcal{X}_j})^T\right)\Big] - \log|\Omega_{\mathcal{X}_j}| \Big) \cdot I\left(x'_i \in \mathcal{X}_j\right) \Big\}. \tag{6}$$

Definition 2. *For each DPT T define*

$$\widehat{\mu}_T, \widehat{\Omega}_T = \text{argmin}_{\mu_{\mathcal{X}_j} \in M_j, \Omega_{\mathcal{X}_j} \in \Lambda_j} \widehat{R}(T, \mu_T, \Omega_T)$$

where $\widehat{R}$ is defined in (3) *but only evaluated on $\mathcal{D}_1 = \{(x_1, y_1), \ldots, (x_{n_1}, y_{n_1})\}$. The* ***held-out risk minimization Go-CART estimator*** *is*

$$\widehat{T} = \text{argmin}_{T \in \mathcal{T}_N} \widehat{R}_{\text{out}}(T, \widehat{\mu}_T, \widehat{\Omega}_T).$$

where $\widehat{R}_{\text{out}}$ is defined in (6) *but only evaluated on $\mathcal{D}_2$.*

The above procedures require us to find an optimal dyadic partitioning tree within $\mathcal{T}_N$. Although dynamic programming can be applied, as in [2], the computation does not scale to large input dimensions d. We now propose a simple yet effective greedy algorithm to find an approximate solution $(\widehat{T}, \widehat{\mu}_T, \widehat{\Omega}_T)$. We focus on the held-out risk minimization form as in Definition 2, due to its superior empirical performance. But note that our greedy approach is generic and can easily be adapted to the penalized empirical risk minimization form.

First, consider the simple case that we are given a dyadic tree structure T which induces a partition $\Pi(T)$=$\{\mathcal{X}_1, \ldots, \mathcal{X}_{m_T}\}$ on $\mathcal{X}$. For any partition element $\mathcal{X}_j$, we estimate the sample mean using $\mathcal{D}_1$:

$$\widehat{\mu}_{\mathcal{X}_j} = \frac{1}{\sum_{i=1}^{n_1} I\left(x_i \in \mathcal{X}_j\right)} \sum_{i=1}^{n_1} y_i \cdot I\left(x_i \in \mathcal{X}_j\right).$$

The glasso is then used to estimate a sparse precision matrix $\widehat{\Omega}_{\mathcal{X}_j}$. More precisely, let $\widehat{\Sigma}_{\mathcal{X}_j}$ be the sample covariance matrix for the partition element $\mathcal{X}_j$, given by

$$\widehat{\Sigma}_{\mathcal{X}_j} = \frac{1}{\sum_{i=1}^{n_1} I\left(x_i \in \mathcal{X}_j\right)} \sum_{i=1}^{n_1} \left(y_i - \widehat{\mu}_{\mathcal{X}_j}\right)\left(y_i - \widehat{\mu}_{\mathcal{X}_j}\right)^T \cdot I\left(x_i \in \mathcal{X}_j\right).$$

The estimator $\widehat{\Omega}_{\mathcal{X}_j}$ is obtained by optimizing $\widehat{\Omega}_{\mathcal{X}_j} = \arg\min_{\Omega \succ 0}\{\text{tr}(\widehat{\Sigma}_{\mathcal{X}_j}\Omega) - \log|\Omega| + \lambda_j\|\Omega\|_1\}$, where λ_j is in one-to-one correspondence with $L_{j,n}$ in (5). In practice, we run the full regularization path of the glasso, from large λ_j, which yields very sparse graph, to small λ_j, and select the graph that minimizes the held-out negative log-likelihood risk. To further improve the model selection performance, we refit the parameters of the precision matrix after the graph has been selected. That is, to reduce the bias of the glasso, we first estimate the sparse precision matrix using ℓ_1-regularization, and then we refit the Gaussian model without ℓ_1-regularization, but enforcing the sparsity pattern obtained in the first step.

The natural, standard greedy procedure starts from the coarsest partition $\mathcal{X} = [0,1]^d$ and then computes the decrease in the held-out risk by dyadically splitting each hyperrectangle $\mathcal{A}$ along dimension $k \in \{1, \ldots d\}$. The dimension k^* that results in the largest decrease in held-out risk is selected, where the change in risk is given by

$$\Delta\widehat{R}^{(k)}_{\text{out}}(\mathcal{A}, \widehat{\mu}_{\mathcal{A}}, \widehat{\Omega}_{\mathcal{A}}) = \widehat{R}_{\text{out}}(\mathcal{A}, \widehat{\mu}_{\mathcal{A}}, \widehat{\Omega}_{\mathcal{A}}) - \widehat{R}_{\text{out}}(\mathcal{A}^{(k)}_L, \widehat{\mu}_{\mathcal{A}^{(k)}_L}, \widehat{\Omega}_{\mathcal{A}^{(k)}_L}) - \widehat{R}_{\text{out}}(\mathcal{A}^{(k)}_R, \widehat{\mu}_{\mathcal{A}^{(k)}_R}, \widehat{\Omega}_{\mathcal{A}^{(k)}_R}).$$

If splitting any dimension k of $\mathcal{A}$ leads to an increase in the held-out risk, the element $\mathcal{A}$ should no longer be split and hence becomes a partition element of $\Pi(T)$. The details and pseudo code are provided in the supplementary materials.

This greedy partitioning method parallels the classical algorithms for classification and regression that have been used in statistical learning for decades. However, the strength of the procedures given in Definitions 1 and 2 is that they lend themselves to a theoretical analysis under relatively weak assumptions, as we show in the following section. The theoretical properties of greedy Go-CART are left to future work.

4 Theoretical Properties

We define the oracle risk R^* over $\mathcal{T}_N$ as

$$R^* = R(T^*, \mu_T^*, \Omega_T^*) = \inf_{T\in\mathcal{T}_N, \mu_{\mathcal{X}_j}\in M_j, \Omega_{\mathcal{X}_j}\in\Lambda_j} R(T, \mu_T, \Omega_T).$$

Note that T^*, $\mu_{T^*}^*$, and $\Omega_{T^*}^*$ might not be unique, since the finest partition always achieves the oracle risk. To obtain oracle inequalities, we make the following two technical assumptions.

Assumption 1. *Let $T \in \mathcal{T}_N$ be an arbitrary DPT which induces a partition $\Pi(T) = \{\mathcal{X}_1, \ldots, \mathcal{X}_{m_T}\}$ on $\mathcal{X}$, we assume that there exists a constant B, such that*

$$\max_{1\le j\le m_T} \|\mu_{\mathcal{X}_j}\|_\infty \le B \quad \text{and} \quad \max_{1\le j\le m_T} \sup_{\Omega\in\Lambda_j} \log|\Omega| \le L_n$$

where Λ_j is defined in (5) *and $L_n = \max_{1\le j\le m_T} L_{j,n}$, where $L_{j,n}$ is the same as in* (5)*. We also assume that*

$$L_n = o(\sqrt{n}).$$

Assumption 2. *Let $Y = (Y_1, \ldots, Y_p)^T \in \mathbb{R}^p$. For any $\mathcal{A} \subset \mathcal{X}$, we define*

$$\begin{aligned} Z_{k\ell}(\mathcal{A}) &= Y_k Y_\ell \cdot I(X \in \mathcal{A}) - \mathbb{E}(Y_k Y_\ell \cdot I(X \in \mathcal{A})) \\ Z_j(\mathcal{A}) &= Y_j \cdot I(X \in \mathcal{A}) - \mathbb{E}(Y_j \cdot I(X \in \mathcal{A})). \end{aligned}$$

We assume there exist constants M_1, M_2, v_1, and v_2, such that

$$\sup_{k,\ell,\mathcal{A}} \mathbb{E}|Z_{k\ell}(\mathcal{A})|^m \le \frac{m! M_2^{m-2} v_2}{2} \quad \text{and} \quad \sup_{j,\mathcal{A}} \mathbb{E}|Z_j(\mathcal{A})|^m \le \frac{m! M_1^{m-2} v_1}{2}$$

for all $m \ge 2$.

Theorem 1. *Let $T \in \mathcal{T}_N$ be a DPT that induces a partition $\Pi(T) = \{\mathcal{X}_1, \ldots, \mathcal{X}_{m_T}\}$ on $\mathcal{X}$. For any $\delta \in (0, 1/4)$, let $\widehat{T}, \widehat{\mu}_{\widehat{T}}, \widehat{\Omega}_{\widehat{T}}$ be the estimator obtained using the penalized empirical risk minimization Go-CART in Definition 1, with a penalty term $\mathrm{pen}(T)$ of the form*

$$\mathrm{pen}(T) = (C_1 + 1) L_n m_T \sqrt{\frac{[[T]] \log 2 + 2\log p + \log(48/\delta)}{n}}$$

where $C_1 = 8\sqrt{v_2} + 8B\sqrt{v_1} + B^2$. Then for sufficiently large n, the excess risk inequality

$$R(\widehat{T}, \widehat{\mu}_{\widehat{T}}, \widehat{\Omega}_{\widehat{T}}) - R^* \le \inf_{T\in\mathcal{T}_N} \left\{ 2\mathrm{pen}(T) + \inf_{\mu_{\mathcal{X}_j}\in M_j, \Omega_{\mathcal{X}_j}\in\Lambda_j} (R(T, \mu_T, \Omega_T) - R^*) \right\}$$

holds with probability at least $1 - \delta$.

A similar oracle inequality holds when using the held-out risk minimization Go-CART.

Theorem 2. *Let $T \in \mathcal{T}_N$ be a DPT which induces a partition $\Pi(T) = \{\mathcal{X}_1, \ldots, \mathcal{X}_{m_T}\}$ on $\mathcal{X}$. We define $\phi_n(T)$ to be a function of n and T such that*

$$\phi_n(T) = (C_2 + \sqrt{2}) L_n m_T \sqrt{\frac{[[T]] \log 2 + 2\log p + \log(384/\delta)}{n}}$$

where $C_2 = 8\sqrt{2v_2} + 8B\sqrt{2v_1} + \sqrt{2}B^2$ and $L_n = \max_{1\le j\le m_T} L_{j,n}$. Partition the data into $\mathcal{D}_1 = \{(x_1, y_1), \ldots, (x_{n_1}, y_{n_1})\}$ and $\mathcal{D}_2 = \{(x_1', y_1'), \ldots, (x_{n_2}', y_{n_2}')\}$ with sizes $n_1 = n_2 = n/2$. Let $\widehat{T}, \widehat{\mu}_{\widehat{T}}, \widehat{\Omega}_{\widehat{T}}$ be the estimator constructed using the held-out risk minimization criterion of Definition 2. Then, for sufficiently large n, the excess risk inequality

$$R(\widehat{T}, \widehat{\mu}_{\widehat{T}}, \widehat{\Omega}_{\widehat{T}}) - R^* \le \inf_{T\in\mathcal{T}_N} \left\{ 3\phi_n(T) + \inf_{\mu_{\mathcal{X}_j}\in M_j, \Omega_{\mathcal{X}_j}\in\Lambda_j} (R(T, \mu_T, \Omega_T) - R^*) \right\} + \phi_n(\widehat{T})$$

with probability at least $1 - \delta$.

Note that in contrast to the statement in Theorem 1, Theorem 2 results in a stochastic upper bound due to the extra $\phi_n(\widehat{T})$ term, which depends on the complexity of the final estimate $\widehat{T}$. Due to space limitations, the proofs of both theorems are detailed in the supplementary materials.

We now temporarily make the strong assumption that the model is correct, so that Y given X is conditionally Gaussian, with a partition structure that is given by a dyadic tree. We show that with high probability, the true dyadic partition structure can be correctly recovered.

Assumption 3. *The true model is*

$$Y \mid X = x \sim N_p(\mu^*_{T^*}(x), \Omega^*_{T^*}(x)) \tag{7}$$

where $T^* \in \mathcal{T}_N$ is a DPT with induced partition $\Pi(T^*) = \{\mathcal{X}^*_j\}_{j=1}^{m_{T^*}}$ and

$$\mu^*_{T^*}(x) = \sum_{j=1}^{m_{T^*}} \mu^*_j \, I(x \in \mathcal{X}^*_j), \quad \Omega^*_{T^*}(x) = \sum_{j=1}^{m_{T^*}} \Omega^*_j \, I(x \in \mathcal{X}^*_j).$$

Under this assumption, clearly

$$R(T^*, \mu^*_{T^*}, \Omega^*_{T^*}) = \inf_{T \in \mathcal{T}_N, \mu_T, \Omega_T \in \mathcal{M}_T} R(T, \mu_T, \Omega_T),$$

where $\mathcal{M}_T$ is given by

$$\mathcal{M}_T = \Big\{ \mu(x) = \sum_{j=1}^{m_T} \mu_{\mathcal{X}_j} \, I(x \in \mathcal{X}_j), \; \Omega(x) = \sum_{j=1}^{m_T} \Omega_{\mathcal{X}_j} \, I(x \in \mathcal{X}_j) : \; \mu_{\mathcal{X}_j} \in M_j, \; \Omega_{\mathcal{X}_j} \in \Lambda_j \Big\}.$$

Let T_1 and T_2 be two DPTs, if $\Pi(T_1)$ can be obtained by further split the hyperrectangles within $\Pi(T_2)$, we say $\Pi(T_2) \subset \Pi(T_1)$. We then have the following definitions:

Definition 3. *A tree estimation procedure $\widehat{T}$ is **tree partition consistent** in case*

$$\mathbb{P}\left(\Pi(T^*) \subset \Pi(\widehat{T})\right) \to 1 \;\; \textit{as } n \to \infty.$$

Note that the estimated partition may be finer than the true partition. Establishing a tree partition consistency result requires further technical assumptions. The following assumption specifies that for arbitrary adjacent subregions of the true dyadic partition, either the means or the variances should be sufficiently different. Without such an assumption, of course, it is impossible to detect the boundaries of the true partition.

Assumption 4. *Let $\mathcal{X}^*_i$ and $\mathcal{X}^*_j$ be adjacent partition elements of T^*, so that they have a common parent node within T^*. Let $\Sigma^*_{\mathcal{X}^*_i} = (\Omega^*_{\mathcal{X}^*_i})^{-1}$. We assume there exist positive constants c_1, c_2, c_3, c_4, such that either*

$$2 \log \left| \frac{\Sigma^*_{\mathcal{X}^*_i} + \Sigma^*_{\mathcal{X}^*_j}}{2} \right| - \log |\Sigma^*_{\mathcal{X}^*_i}| - \log |\Sigma^*_{\mathcal{X}^*_j}| \geq c_4$$

or $\|\mu^*_{\mathcal{X}^*_i} - \mu^*_{\mathcal{X}^*_j}\|_2^2 \geq c_3$. We also assume

$$\rho_{\min}(\Omega^*_{\mathcal{X}^*_j}) \geq c_1, \quad \forall j = 1, \ldots, m_{T^*},$$

where $\rho_{\min}(\cdot)$ denotes the smallest eigenvalue. Furthermore, for any $T \in \mathcal{T}_N$ and any $\mathcal{A} \in \Pi(T)$, we have $\mathbb{P}(X \in \mathcal{A}) \geq c_2$.

Theorem 3. *Under the above assumptions, we have*

$$\inf_{T \in \mathcal{T}_N, \, \Pi(T^*) \not\subseteq \Pi(T)} \; \inf_{\mu_T, \, \Omega_T \in \mathcal{M}_T} R(T, \mu_T, \Omega_T) - R(T^*, \mu^*_{T^*}, \Omega^*_{T^*}) > \min\{\frac{c_1 c_2 c_3}{2}, c_2 c_4\}$$

where c_1, c_2, c_3, c_4 are defined in Assumption 4. Moreover, the Go-CART estimator in both the penalized risk minimization and held-out risk minimization form is tree partition consistent.

This result shows that, with high probability, we obtain a finer partition than T^*; the assumptions do not, however, control the size of the resulting partition. The proof of this result appears in the supplementary material.

5 Experiments

We now present the performance of the greedy partitioning algorithm of Section 3 on both synthetic data and a real meteorological dataset. In the experiment, we always set the dyadic integer $N = 2^{10}$ to ensure that we can obtain fine-tuned partitions of the input space $\mathcal{X}$.

5.1 Synthetic Data

We generate n data points $x_1, \ldots, x_n \in \mathbb{R}^d$ with $n = 10,000$ and $d = 10$ uniformly distributed on the unit hypercube $[0,1]^d$. We split the square $[0,1]^2$ defined by the first two dimension of the unit

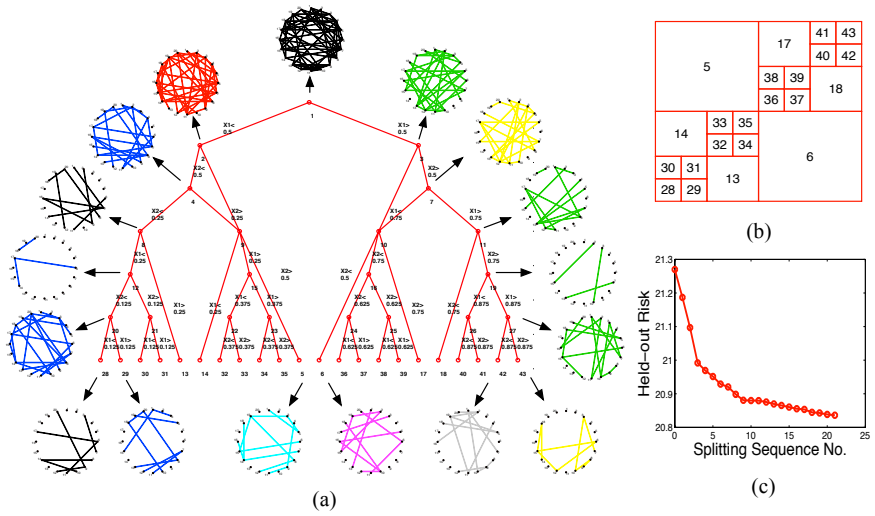

Figure 1: Analysis of synthetic data. (a) Estimated dyadic tree structure; (b) Ground true partition. The horizontal axis corresponds to the first dimension denoted as X_1 while the vertical axis corresponds to the second dimension denoted by X_2. The bottom left point corresponds to $[0, 0]$ and the upper right point corresponds to $[1, 1]$. It is also the induced partition on $[0, 1]^2$. The number labeled on each subregion corresponds to each leaf node ID of the tree in (a); (c) The held-out negative log-likelihood risk for each split. The order of the splits corresponds the ID of the tree node (from small to large).

hypercube into 22 subregions as shown in Figure 1 (b). For the t-th subregion where $1 \leq t \leq 22$, we generate an Erdös-Rényi random graph $G^t = (V^t, E^t)$ with the number of vertices $p = 20$, the number of edges $|E| = 10$ and the maximum node degree is four. Based on G^t, we generate the inverse covariance matrix Ω^t according to $\Omega^t_{i,j} = I(i = j) + 0.245 \cdot I((i, j) \in E^t)$, where 0.245 guarantees the positive definiteness of Ω^t when the maximum node degree is 4. For each data point x_i in the t-th subregion, we sample a 20-dimensional response vector y_i from a multivariate Gaussian distribution $N_{20}\left(0, \left(\Omega^t\right)^{-1}\right)$. We also create an equally-sized held-out dataset in the same manner based on $\{\Omega^t\}_{t=1}^{22}$.

The learned dyadic tree structure and its induced partition are presented in Figure 1. We also provide the estimated graphs for some nodes. We conduct 100 monte-carlo simulations and find that 82 times out of 100 runs our algorithm perfectly recover the ground true partitions on the X_1-X_2 plane and never wrongly split any irrelevant dimensions ranging from X_3 to X_{10}. Moreover, the estimated graphs have interesting patterns. Even though the graphs within each subregion are sparse, the estimated graph obtained by pooling all the data together is highly dense. As the greedy algorithm proceeds, the estimated graphs become sparser and sparser. However, for the immediate parent of the leaf nodes, the graphs become denser again. Out of the 82 simulations where we correctly identify the tree structure, we list the graph estimation performance for subregions 28, 29, 13, 14, 5, 6 in terms of precision, recall, and F1-score in Table 1.

Table 1: The graph estimation performance over different subregions

	Mean values over 100 runs (Standard deviation)					
subregion	region 28	region 29	region 13	region 14	region 5	region 6
Precision	0.8327 (0.15)	0.8429 (0.15)	0.9853 (0.04)	0.9821 (0.05)	0.9906 (0.04)	0.9899 (0.05)
Recall	0.7890 (0.16)	0.7990 (0.18)	1.0000 (0.00)	1.0000 (0.00)	1.0000 (0.00)	1.0000 (0.00)
$\mathrm{F1-score}$	0.7880 (0.11)	0.7923 (0.12)	0.9921 (0.02)	0.9904 (0.03)	0.9949 (0.02)	0.9913 (0.02)

We see that for a larger subregion (e.g. 13, 14, 5, 6), it is easier to obtain better recovery performance; while good recovery for a very small region (e.g. 28, 29) becomes more challenging. We also plot the held-out risk in the subplot (c). As can be seen, the first few splits lead to the most significant decreases of the held-out risk. The whole risk curve illustrates a diminishing return behavior. Correctly splitting the large rectangle leads to a significant decrease in the risk; in contrast, splitting the middle rectangles does not reduce the risk as much. We also conducted simulations where the true conditional covariance matrix is a continuous function of x; these are presented in the supplementary materials.

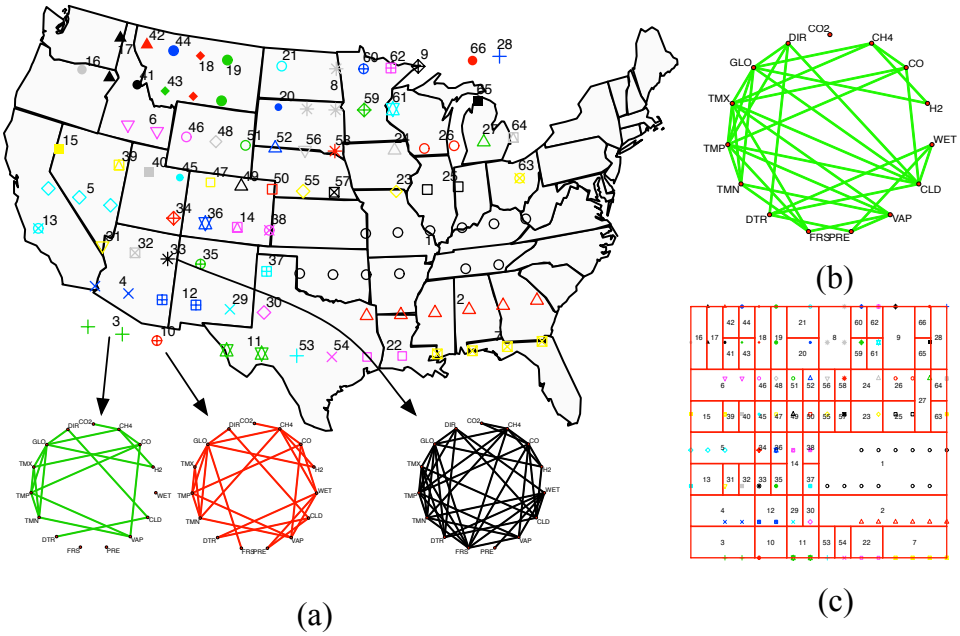

Figure 2: Analysis of climate data. (a) Learned partitions for the 100 locations and projected to the US map, with the estimated graphs for subregions 3, 10, and 33; (b) Estimated graph with data pooled from all 100 locations; (c) the re-scaled partition pattern induced by the learned dyadic tree structure.

5.2 Climate Data Analysis

In this section, we apply Go-CART on a meteorology dataset collected in a similar approach as in [8]. The data contains monthly observations of 15 different meteorological factors from 1990 to 2002. We use the data from 1990 to 1995 as the training data and data from 1996 to 2002 as the held-out validation data. The observations span 100 locations in the US between latitudes 30.475 to 47.975 and longitudes -119.75 to -82.25. The 15 meteorological factors measured for each month include levels of CO_2, CH_4, H_2, CO, average temperature (TMP) and diurnal temperature range (DTR), minimum temperate (TMN), maximum temperature (TMX), precipitation (PRE), vapor (VAP), cloud cover (CLD), wet days (WET), frost days (FRS), global solar radiation (GLO), and direct solar radiation (DIR).

As a baseline, we estimate a sparse graph on the data pooled from all 100 locations, using the glasso algorithm; the estimated graph is shown in Figure 2 (b). It is seen that the greenhouse gas factor CO_2 is isolated from all the other factors. This apparently contradicts the basic domain knowledge that CO_2 should be correlated with the solar radiation factors (including GLO, DIR), according to the IPCC report [6] which is one of the most authoritative reports in the field of meteorology. The reason for the missing edges in the pooled data may be that positive correlations at one location are canceled by negative correlations at other locations.

Treating the longitude and latitude of each site as two-dimensional covariate X, and the meteorology data of the $p = 15$ factors as the response Y, we estimate a dyadic tree structure using the greedy algorithm. The result is a partition with 66 subregions, shown in Figure 2. The graphs for subregions 3 and 10 (corresponding to the coast of California and Arizona states) are shown in subplot (a) of Figure 2. The graphs for these two adjacent subregions are quite similar, suggesting spatial smoothness of the learned graphs. Moreover, for both graphs, CO_2 is connected to the solar radiation factor GLO through CH_4. In contrast, for subregion 33, which corresponds to the north part of Arizona, the estimated graph is quite different. In general, it is found that the graphs corresponding to the locations along the coasts are sparser than those corresponding to the locations in the mainland.

Such observations, which require validation and interpretation by domain experts, are examples of the capability of graph-valued regression to provide a useful tool for high dimensional data analysis.

References

[1] O. Banerjee, L. E. Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation. *Journal of Machine Learning Research*, 9:485–516, March 2008.

[2] G. Blanchard, C. Schäfer, Y. Rozenholc, and K.-R. Müller. Optimal dyadic decision trees. *Mach. Learn.*, 66(2-3):209–241, 2007.

[3] L. Breiman, J. Friedman, C. J. Stone, and R. Olshen. *Classification and regression trees*. Wadsworth Publishing Co Inc, 1984.

[4] D. Edwards. *Introduction to graphical modelling*. Springer-Verlag Inc, 1995.

[5] J. H. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2007.

[6] IPCC. Climate Change 2007–The Physical Science Basis *IPCC Fourth Assessment Report*.

[7] S. L. Lauritzen. *Graphical Models*. Oxford University Press, 1996.

[8] A. C. Lozano, H. Li, A. Niculescu-Mizil, Y. Liu, C. Perlich, J. Hosking, and N. Abe. Spatial-temporal causal modeling for climate change attribution. In *ACM SIGKDD*, 2009.

[9] P. Ravikumar, M. Wainwright, G. Raskutti, and B. Yu. Model selection in Gaussian graphical models: High-dimensional consistency of ℓ_1-regularized MLE. In *Advances in Neural Information Processing Systems 22*, Cambridge, MA, 2009. MIT Press.

[10] A. J. Rothman, P. J. Bickel, E. Levina, and J. Zhu. Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, 2:494–515, 2008.

[11] C. Scott and R. Nowak. Minimax-optimal classification with dyadic decision trees. *Information Theory, IEEE Transactions on*, 52(4):1335–1353, 2006.

[12] J. Whittaker. *Graphical Models in Applied Multivariate Statistics*. Wiley, 1990.

[13] M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35, 2007.

[14] S. Zhou, J. Lafferty, and L. Wasserman. Time varying undirected graphs. *Machine Learning*, 78(4), 2010.