T-LoHo: A Bayesian Regularization Model for Structured Sparsity and Smoothness on Graphs

Anonymous Author(s)
Affiliation
Address
email

Abstract

Many modern complex data can be represented as a graph. In models dealing with graph-structured data, multivariate parameters are not just sparse but have structured sparsity and smoothness in the sense that both zero and non-zero parameters tend to cluster together. We propose a new prior for high dimensional parameters with graphical relations, referred to as a Tree-based Low-rank Horseshoe(T-LoHo) model, that generalizes the popular univariate Bayesian horseshoe shrinkage prior to the multivariate setting to detect structured sparsity and smoothness simultaneously. The prior can be embedded in many hierarchical high dimensional models. To illustrate its utility, we apply it to regularize a Bayesian high-dimensional regression problem where the regression coefficients are linked on a graph. The resulting clusters have flexible shapes and satisfy the cluster contiguity constraint with respect to the graph. We design an efficient Markov chain Monte Carlo algorithm that delivers full Bayesian inference with uncertainty measures for model parameters including the number of clusters. We offer theoretical investigations of the clustering effects and posterior concentration results. Finally, we illustrate the performance of the model with simulation studies and real data applications such as anomaly detection in road networks. The results indicate substantial improvements over other competing methods such as sparse fused lasso.

1 Introduction

In high-dimensional models such as linear regressions where the number of parameters p may exceed the sample size n, it is often assumed that the p-dimensional parameter vector $\boldsymbol{\beta}$ has many zero components, namely the *sparsity assumption*. This sparsity assumption allows the unknown parameter $\boldsymbol{\beta}$ to lie on a low-dimensional subspace of $\mathbb{R}^p$, which addresses overfitting and leads to improved predictions and easier interpretations [17]. In many real-life applications, however, the parameter of interest $\boldsymbol{\beta}$ should be understood in a certain context where specific data structures exist, such as in time series, spatial and image data analysis, and network models. In many such cases, it is desirable to consider another type of low-dimensional structure, where $\boldsymbol{\beta}$ is assumed to have clustered patterns and have many possibly clustered zeros, which we shall call *sparse homogeneity assumption*.

One of the most popular models which assumes sparse homogeneity is fused lasso [40] in linear regression settings. It imposes an ℓ_1 penalty on the differences of time-neighboring coefficients as well as individual coefficients. The notion of 'time-neighboring' can be further generalized by considering a graph of regression coefficients $G=(V,E)$ with $|V|=p$ so that G represents the general context or structure in which $\boldsymbol{\beta}$ should be interpreted. It leads to the generalized lasso [41] where penalty term now involves $m := |E|$ number of pairwise differences of neighboring vertices with respect to G, and other similar approaches include grouping pursuit [43] and graph trend filtering [42]. However, when the number of edges m is large relative to p, regularization over the whole graph structure often faces

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

severe computational burden. [21, 38] avoided this problem by constructing a preliminary ranking of coefficients and performing a segmentation in 1-D. [28] used depth-first search(DFS) algorithm to establish an order and performed fused lasso on the chain graph. In a spatial setting, [24] used a Euclidean minimum spanning tree to form the fused lasso penalty which also enjoys computational benefits. Nevertheless, these fixed chain or tree orders may not be compatible with the true order with respect to its context G, and hence may lead to over-clustering. Besides, these optimization-based penalized estimators do not usually come with uncertainty measures.

Bayesian regularization methods have gained great popularity in high dimensional models due to its flexibility and convenience in quantifying estimation and prediction uncertainties. There is rich literature in Bayesian high-dimensional regression model under the sparsity assumption, which can be roughly summarized into two categories: (1) spike-and-slab priors [10, 11] and (2) global-local shrinkage priors (see [4] and references therein). A particular shrinkage prior, called the horseshoe prior [5], has gained a lot of attraction due to its tail-robustness and super-efficiency as well as substantial computational benefits of global-local shrinkage prior family [29]. However, few works have been done in the area of Bayesian high-dimensional regression model under the sparse homogeneity assumption. [23, 34] approached with Bayesian versions of fused lasso by introducing priors directly on the differences of neighboring coefficients. But similar to the generalized fused lasso, these approaches become computationally expensive to get posterior samples for general graphs with a large number of edges. Another more subtle limitation of these Bayesian fused lasso methods is that they fail to incorporate structural assumptions among local shrinkage parameters.

In this paper, we propose a Bayesian Tree-based Low-rank Horseshoe (T-LoHo) prior model to identify structured sparsity and smoothness of parameters represented as a graph. T-LoHo extends the univariate horseshoe shrinkage prior to a multivariate setting where $\boldsymbol{\beta}$ and its local shrinkage parameters are assumed to be piecewise constants on a graph. This low-rank structured model for local shrinkage parameters allows both clustering and sparsity effects to have strong local adaptivities. A random spanning forest(RSF)-based graph clustering prior is introduced to adaptively learn a compatible order to model graph partitions, which extends the recently developed random spanning tree partition models [39, 26] to allow for possibly unconnected graphs. We will show that this tree based prior improves computation efficiency without losing statistical efficiency as its support is flexible enough to accommodate all possible contiguous partitions. We introduce T-LoHo prior to model high dimensional linear regression coefficients, although it can be flexibly embedded in other high dimensional models. The resulting cluster estimates not only provide uncertainty measures but also have strong flexibility to accommodate sharp discontinuities. With state of the art computational strategies, we provide a highly efficient MCMC algorithm for the posterior inference. We also study theoretical aspects of our model including posterior consistency result and its effect on clustering. We demonstrate the efficacy of T-LoHo model by using synthetic data in the linear model example and its application to the real data analysis focusing on anomaly detection in road networks.

2 T-LoHo: Tree-based Low-rank Horseshoe Model

Consider a graph $G = (V, E)$ with $|V| = p$ and $|E| = m$ which represents the pre-known structure of our parameter of interest $\boldsymbol{\beta} \in \mathbb{R}^p$. For example, when $\boldsymbol{\beta}$ has a certain order (e.g. time) then we set G be a linear chain graph. When (vectorized) $\boldsymbol{\beta}$ lies on a 2-D image, we let G be a 2-D lattice. A *contiguous graph partition* $\Pi = \{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$ is a partition of V such that each $\mathcal{C}_k$ induces a connected subgraph of G (the term *graph partition* will always refer to a contiguous graph partition hereafter). Under the sparse homogeneity assumption on $\boldsymbol{\beta}$, the goal is to find a graph partition Π with unknown size K and corresponding $\boldsymbol{\beta}$ which contains many zeros. We first describe our sparsity-inducing model in section 2.1 conditional on a partition Π, and we describe the model for Π in section 2.2.

2.1 Low-Rank Multivariate Horseshoe Prior

First we review a horseshoe prior [5], which provides a sparse estimate of $\boldsymbol{\beta}$ by shrinking small effects towards zero while maintaining robustness to large signals:

$$\boldsymbol{\beta}|\sigma^2, \tau^2, \{\lambda_j\}_{j=1}^p \sim \mathcal{N}_p(\mathbf{0}, \sigma^2\tau^2 \text{diag}(\lambda_1^2, \ldots, \lambda_p^2))$$
$$\lambda_j \overset{iid}{\sim} C^+(0,1), \quad \tau \sim C^+(0, \tau_0), \quad p(\sigma^2) \propto 1/\sigma^2$$

where $\mathcal{N}_p(\boldsymbol{m}, \boldsymbol{V})$ denotes p-dimensional multivariate normal distribution with mean $\boldsymbol{m}$ and covariance $\boldsymbol{V}$, $C^+(0, \gamma)$ denotes half-Cauchy distribution with density $p(\lambda) = 2/[\pi\gamma(1 + \lambda^2/\gamma^2)]$ for $\lambda > 0$. Here τ is a global shrinkage parameter with hyperparameter τ_0 which provides substantial shrinkage towards zero, $\{\lambda_j\}$ are local shrinkage parameters with heavy tails which allow some of the β_j's to escape the shrinkage, and σ^2 is a scaling factor which is often assumed to be noise variance.

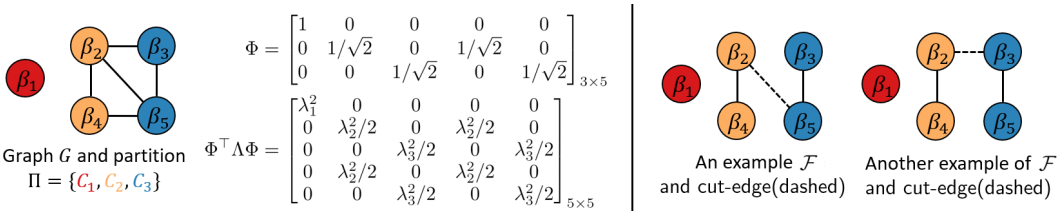

Figure 1: Illustrative example of graph partitioning and corresponding parameters when $\boldsymbol{\beta} \in \mathbb{R}^5$ forms $K = 3$ clusters, $\mathcal{C}_1 = \{1\}, \mathcal{C}_2 = \{2, 4\}, \mathcal{C}_3 = \{3, 5\}$.

Given a graph partition $\Pi = \{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$ of G, we seek a new joint prior distribution of $\boldsymbol{\beta} \in \mathbb{R}^p$ which reflects the clustering structure of Π to address homogeneity assumption. For example when $p = 5$ and $\Pi = \{\{1\}, \{2, 4\}, \{3, 5\}\}$ as shown in left panel of fig. 1, we seek a distribution which lies on a 3-dimensional subspace $\{\boldsymbol{\beta} \in \mathbb{R}^5 : \beta_2 = \beta_4, \beta_3 = \beta_5\}$. In general, given a graph partition Π with size K, we can construct a $K \times p$ matrix Φ from Π:

$$\Phi_{kj} = 1/\sqrt{|\mathcal{C}_k|} \text{ if } j \in \mathcal{C}_k \text{ and } 0 \text{ otherwise}, \quad k = 1, \ldots, K, \;\; j = 1, \ldots, p \tag{1}$$

This $K \times p$ matrix Φ represents a projection from a p dimensional to a reduced K-dimensional space. Note that rows of Φ are mutually orthonormal so that $\Phi\Phi^\top = \mathbf{I}_K$. Using Φ obtained from Π, we now introduce the low-rank horseshoe (LoHo) prior to impose sparse homogeneity assumption on $\boldsymbol{\beta}$:

$$\boldsymbol{\beta} \,|\, \sigma^2, \tau^2, \Lambda, \Pi \sim \mathcal{N}_p(\mathbf{0}, \sigma^2\tau^2\Phi^\top\Lambda\Phi), \;\; \Lambda := \mathrm{diag}(\lambda_1^2, \ldots, \lambda_K^2) \tag{2}$$

$$\lambda_k \overset{iid}{\sim} C^+(0, 1), \quad \tau \sim C^+(0, \tau_0), \quad p(\sigma^2) \propto 1/\sigma^2 \tag{3}$$

See left panel of fig. 1 for an example of Φ and a covariance matrix $\Phi^\top\Lambda\Phi$. Note LoHo introduces a covariance matrix capturing clustering dependence among β and its marginal local shrinkage for simultaneous sparsity and fusion, a key distinction from most existing methods [5, 23, 34] that assume independence or only dependence among $\boldsymbol{\beta}$ but not local shrinkage. Since K is assumed to be small relative to p, the covariance matrix $\Phi^\top\Lambda\Phi$ has low rank (i.e. $\mathrm{rank}(\Phi^\top\Lambda\Phi) = K \ll p$) and thus LoHo does not have a density with respect to Lebesgue measure on $\mathbb{R}^p$. Instead, its distribution lies on a row space of Φ with $\dim(\mathrm{rowsp}(\Phi)) = K$ and we can consider transformation $\tilde{\boldsymbol{\beta}} = \Phi\boldsymbol{\beta}$ so that $\tilde{\boldsymbol{\beta}}$ has a $\mathcal{N}_K(\mathbf{0}, \sigma^2\tau^2\Lambda)$ density with respect to Lebesgue measure on $\mathrm{rowsp}(\Phi)$. Observe that $\Phi^\top$ is the Moore–Penrose pseudoinverse of Φ, which implies $\Phi^\top\Phi$ is a projection onto $\mathrm{rowsp}(\Phi)$ so that we can recover $\boldsymbol{\beta} = \Phi^\top\tilde{\boldsymbol{\beta}}$.

By assigning a half-Cauchy prior, global shrinkage parameter τ creates a strong pull towards zero while *clusterwise* local shrinkage parameters $\{\lambda_k\}_{k=1}^K$ allow some of the *clusterwise* $\tilde{\beta}_k$'s to escape the shrinkage. Here Φ having orthonormal rows is important in the sense that the effect of local shrinkage parameters $\{\lambda_k\}_{k=1}^K$ remains same across the clusters with varying size. Thus using the projection onto the low-dimensional subspace, LoHo gives more parsimonious estimate of $\boldsymbol{\beta}$ by forming clusters of zero and nonzero parameters.

LoHo can be naturally incorporated into a linear model. With response vector $\boldsymbol{y} \in \mathbb{R}^n$ and column-standardized design matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ so that each column has unit ℓ_2 norm, we can write

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}_n(\mathbf{0}, \sigma^2\mathbf{I}_n)$$

Under this formulation, LoHo has a close connection with Bayesian compressed regression (BCR) [16]. BCR randomly projects predictors $\mathbf{X}_i \mapsto \Phi\mathbf{X}_i$ with a certain family of matrix Φ and performs model averaging, at the cost of losing interpretation of $\boldsymbol{\beta}$. But LoHo directly introduces a prior on $\boldsymbol{\beta}$ using the projection matrix Φ defined as (1) so that it induces clustered coefficient while maintaining interpretability of $\boldsymbol{\beta}$. Also, functional horseshoe prior (fHS) [35] shrinks $\mathbf{X}\boldsymbol{\beta}$ towards the subspace of $\mathrm{colsp}(\mathbf{X})$ while LoHo shrinks $\boldsymbol{\beta}$ towards $\mathbf{0}$ along $\mathrm{rowsp}(\Phi)$ to achieve sparse homogeneity.

We remark that, although it appears (τ, Λ) and Φ handle sparsity and homogeneity separately, shrinkage component in LoHo also affects clustering by facilitating cluster fusion when signal is low and improving cluster identification when signal is high. More details are discussed in section 3.2.

2.2 Tree-based Graph Partition Prior

Now we describe how we model the unknown partition Π. A graph partition can be defined as a collection of disjoint connect subgraphs such that the union of vertices is V. To bypass the need to handle a complex combinatorial graph partition problem, we consider an equivalent formulation of graph partition through cuts of spanning forests of G. Prop. 1 guarantees that for *any* choice of graph partition, there exist a spanning forest and corresponding set of cut-edges that *induces* the partition Π.

Proposition 1. *Let $G=(V,E)$ be a graph with n_c connected components and $\Pi=\{\mathcal{C}_1,\ldots,\mathcal{C}_K\}$ be a graph partition of G. Then there exists a spanning forest $\mathcal{F}=(V,E^F)$ with $|E^F|=|V|-n_c$, and a set of cut-edges $E^C \subset E^F$ with $|E^C| = K - n_c$ such that the induced cut of $\mathcal{F}$ is Π.*

Proof is deferred to Appendix A2. We will say a spanning forest $\mathcal{F}$ is *compatible* with Π if we can construct Π by cutting some of its edges. See right panel of fig. 1 for two examples of compatible $\mathcal{F}$ and cut-edge(s). Prop.1 suggests that to induce a graph partition prior with full support, it amounts to first assigning a prior model on all possible spanning forests of G, and then assigning a prior model on all possible cut-edges sets conditional on a spanning forest. When G is a connected spatial or spatial temporal graph, [39] considered a uniform prior on $\mathcal{F}$ which selects a spanning tree randomly from among all possible spanning trees with equal probability. But in this case, approximate method should be used to sample a spanning tree from its full conditional distribution due to serious inefficiency of the rejection sampler. In contrast, [26] considered a random minimum spanning tree approach by assigning iid uniform random weights to the edges, so it can generate any arbitrarily given spanning tree of G and enable an efficient posterior conditional sampling algorithm. Thus we follow a similar approach as in [26] which leads to the following prior on $\mathcal{F}$ with full support induced by random edge weights $\boldsymbol{W} = \{W_{ij}\}_{(i,j)\in E}$:

$$\mathcal{F} = \text{MSF}(G, \boldsymbol{W}), \quad W_{ij} \overset{iid}{\sim} \text{Unif}(0,1), (i,j) \in E \tag{4}$$

where $\text{MSF}(G, \boldsymbol{W})$ denotes the minimum spanning forest of the graph G with edge weights $\boldsymbol{W}$.

After $\mathcal{F}$ is given, selecting $K-n_c$ cut-edges forms a partition Π with size K. Following approaches of [22, 9, 26], we introduce a geometrically decaying prior distribution on K, and then select cut-edges uniformly at random given $(\mathcal{F}, K)$. The following prior specification completes the T-LoHo model:

$$\Pr(K=k) \propto (1-c)^k, \quad k = n_c, n_c+1, \ldots, p, \quad c \in [0,1) \tag{5}$$

$$p(\Pi \,|\, \mathcal{F}, K) \propto 1(\mathcal{F} \text{ is compatible with } \Pi \text{ and } |\Pi| = K) \tag{6}$$

T-LoHo involves two hyperparameters related to model complexity penalization. One is c in (5) which, if selected to be closer to 1, strongly penalizes models with larger numbers of clusters. Another is τ_0 in (3) controlling the strength of global shrinkage. As τ_0 reduces to 0, the posterior distribution of $\tilde{\boldsymbol{\beta}}$ tends to concentrate more at zero. More detailed hyperparameter sensitivity analysis and selection criteria are deferred to Appendix A5.

3 Posterior Inference and Theoretical Properties

3.1 Posterior Sampler and Computational Strategies

Here we briefly describe a reversible-jump Markov chain Monte Carlo algorithm (RJMCMC) [15] and discuss computational strategies therein. Denote the set of parameters $\Theta := (\tilde{\boldsymbol{\beta}}, \sigma^2, \Lambda, \tau, \Pi, K, \mathcal{F})$ and $\tilde{\mathbf{X}} := \mathbf{X}\Phi^\top$ so that $\tilde{\mathbf{X}}\tilde{\boldsymbol{\beta}} = \mathbf{X}\Phi^\top\Phi\boldsymbol{\beta} = \mathbf{X}\boldsymbol{\beta}$. The posterior $p(\Theta|\boldsymbol{y})$ becomes

$$p(\Theta|\boldsymbol{y}) \propto \quad \mathcal{N}_n(\boldsymbol{y}|\tilde{\mathbf{X}}\tilde{\boldsymbol{\beta}}, \sigma^2\mathbf{I}_n) \times \mathcal{N}_K(\tilde{\boldsymbol{\beta}}|\mathbf{0}, \sigma^2\tau^2\Lambda) \times 1/\sigma^2 \tag{7}$$

$$\times\, (1+\tau^2)^{-1} \textstyle\prod_{k=1}^K (1+\lambda_k^2)^{-1} \times \binom{p-n_c}{K-n_c}^{-1} \times (1-c)^K \times 1 \tag{8}$$

where line (8) is the product of priors $p(\tau)\prod_{k=1}^K p(\lambda_k) p(\Pi|K,\mathcal{F}) p(K) p(\boldsymbol{W})$. To perform posterior inference, we sample $\Theta|\boldsymbol{y}$ using collapsed RJMCMC posterior sampler as described in Algorithm 1.

Algorithm 1: One full iteration of RJMCMC posterior sampler

Step 1. Update $\Pi, K, \mathcal{F}$ using collapsed conditional $[\Pi, K, \mathcal{F} | \Lambda, \tau, \boldsymbol{y}]$ where $\tilde{\boldsymbol{\beta}}, \sigma^2$ are integrated out.† With probabilities (p_a, p_b, p_c, p_d) summing up to 1, perform one of the following substeps:

- 1-a. (*split*) Propose $(\Pi^\star, K^\star = K+1)$ compatible with $\mathcal{F}$, and accept with probability $\min\{1, \mathcal{A}_a \cdot \mathcal{P}_a \cdot \mathcal{L}_a\}$, where $\mathcal{A}_a$ is prior ratio, $\mathcal{P}_a$ is proposal ratio, $\mathcal{L}_a$ is likelihood ratio.
- 1-b. (*merge*) Propose $(\Pi^\star, K^\star = K-1)$ compatible with $\mathcal{F}$, and accept w.p. $\min\{1, \mathcal{A}_b \cdot \mathcal{P}_b \cdot \mathcal{L}_b\}$.
- 1-c. (*change*) Propose $(\Pi^\star, K^\star = K)$ compatible with $\mathcal{F}$, and accept w.p. $\min\{1, \mathcal{A}_c \cdot \mathcal{P}_c \cdot \mathcal{L}_c\}$.
- 1-d. (*hyper*) Update $\mathcal{F}^\star$ compatible with current Π.

Step 2. Jointly update $(\tau, \sigma^2, \tilde{\boldsymbol{\beta}})$ from $[\tau, \sigma^2, \tilde{\boldsymbol{\beta}} \mid \Lambda, \Pi, K, \mathcal{F}, \boldsymbol{y}]$, by performing:

- 2-1. Update τ from $[\tau \mid \Lambda, \Pi, K, \mathcal{F}, \boldsymbol{y}]$ using Metropolis-Hastings sampler,
- 2-2. Update σ^2 from $[\sigma^2 \mid \tau, \Lambda, \Pi, K, \mathcal{F}, \boldsymbol{y}]$ with an inverse gamma distribution,
- 2-3. Update $\tilde{\boldsymbol{\beta}}$ from $[\tilde{\boldsymbol{\beta}} \mid \sigma^2, \tau, \Lambda, \Pi, K, \mathcal{F}, \boldsymbol{y}]$ with a multivariate normal distribution.

Step 3. Update Λ from $[\Lambda \mid \tau, \sigma^2, \tilde{\boldsymbol{\beta}}, \Pi, K, \mathcal{F}, \boldsymbol{y}]$ using slice sampler.

† When $\mathbf{X} = \mathbf{I}_n$(i.e. normal means model), it is possible to integrate out Λ instead of σ^2; see Appendix A1.3.

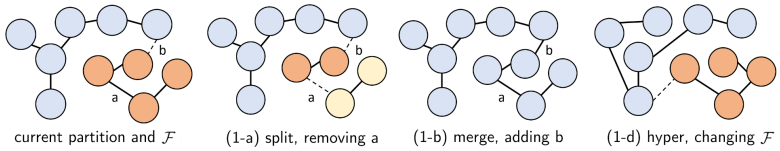

Figure 2: Illustration of step 1 in Alg.1. Step (1-c) corresponds to performing (1-a), (1-b) sequentially.

Step 1 corresponds to the 'jump' between the models by changing the cluster assignments; see fig. 2 for examples of such moves. Instead of using the full conditional distribution, we use the collapsed conditional distribution $[\Pi, K, \mathcal{F} | \Lambda, \tau, \boldsymbol{y}]$ to calculate likelihood ratio $\mathcal{L}$ which significantly improves mixing. The marginal likelihood after integrating out $\tilde{\boldsymbol{\beta}}$ and σ^2 is proportional to $|\boldsymbol{\Sigma}|^{-1/2}(\boldsymbol{y}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{y}/2)^{-n/2}$ where $\boldsymbol{\Sigma} = \mathbf{I}_n + \tau^2 \tilde{\mathbf{X}} \Lambda \tilde{\mathbf{X}}^\top$. In step 2, we jointly update $[\tau, \sigma^2, \tilde{\boldsymbol{\beta}} | -] = [\tau | -] \times [\sigma^2 | \tau, -] \times [\tilde{\boldsymbol{\beta}} | \tau, \sigma^2, -]$ following the approach of [18]. Finally in step 3, we update local shrinkage parameters Λ using slice sampler [27, 30].

One demanding computation bottleneck is the likelihood calculation that involves $\boldsymbol{\Sigma}^{-1}$ and $|\boldsymbol{\Sigma}|$. A notable advantage of T-LoHo is that it projects $\mathbf{X}$ to a low-rank space under the sparse homogeneity assumption, leading to an n by K transformed design matrix $\tilde{\mathbf{X}} = \mathbf{X}\Phi^\top$ where typically $K \ll n$. Applying Sherman-Woodbury-Morrison formula, we can easily invert $\boldsymbol{\Sigma}^{-1} = \mathbf{I}_n - \tilde{\mathbf{X}}(\tau^{-2}\Lambda^{-1} + \tilde{\mathbf{X}}^\top \tilde{\mathbf{X}})^{-1} \tilde{\mathbf{X}}^\top$ by reducing the rank of the inverting matrix. Especially, we utilize Cholesky decomposition $\mathbf{R}^\top \mathbf{R} = \tau^{-2}\Lambda^{-1} + \tilde{\mathbf{X}}^\top \tilde{\mathbf{X}}$ to efficiently calculate $\boldsymbol{\Sigma}^{-1}$ and $|\boldsymbol{\Sigma}|$. Once we have $\mathbf{R}$ which is right triangular, calculation of $\boldsymbol{\Sigma}^{-1}$ only involves backward substitution and also we can calculate $|\boldsymbol{\Sigma}|$ at no extra cost using the matrix determinant lemma. We can directly use $\boldsymbol{\Sigma}^{-1}$ in step (2-2) and $\mathbf{R}$ in step (2-3) via [33]'s algorithm, which are all precomputed in step (2-1).

A naive implementation of the above likelihood calculation would require to repeatedly update $\tau^{-2}\Lambda^{-1} + \tilde{\mathbf{X}}^\top \tilde{\mathbf{X}}$ and its Cholesky factor $\mathbf{R}$ for each MCMC step. For instance, step 1 has new $\tilde{\mathbf{X}}^\star = \mathbf{X}\Phi^{\star\top}$ and steps 2 and 3 have a new $\tau^\star$ or $\Lambda^\star$. Alternatively, we propose to directly update $\mathbf{R}$ from its previous value. For step 1, we use Cholesky rank-1 update/downdate [12, Sec. 6.5.4] which reduces computation cost from $O(nK^2 + K^3)$ to $O(K^2)$. For steps 2 and 3, we design a new tailored updating algorithm, *Cholesky diagonal update*, which reduces computation cost from $O(nK^2 + K^3)$ to $O(K^3)$. This Cholesky updating scheme proves to be a simple yet powerful computation strategy as MCMC typically requires many iterations. See Appendix A1 for its detailed explanation.

Excluding step (1-d), the total computation cost of Alg. 1 is $O(\max\{nK, K^3\})$, compared to the direct computation of $\boldsymbol{\Sigma}^{-1}$ and $|\boldsymbol{\Sigma}|$ which leads to $O(n^3)$. Step (1-d) takes $O(m \log p)$ to construct MSF where m is the number of edges in G. We note that for each iteration, step 1-d (hyper) is selected with probability p_d and we suggest a small value of p_d such as 0.05, considering that it would give RJMCMC enough time to explore the partition compatible with the current MSF $\mathcal{F}$.

3.2 Clustering Effect of T-LoHo

In this section, we investigate how T-LoHo prior differs from the normal prior in terms of the effect on clustering. The sparsity of difference of parameters $\beta_i - \beta_j$ plays an important role in clustering, examples include fused lasso [40] and ℓ_0 penalization [8]. In the Bayesian perspective, [36] put a t prior and most recently [2] put a horseshoe prior directly on the difference of coefficients to induce shrinkage. But the aforementioned Bayesian methods have several limitations: (1) An additional post-processing step is required to have cluster estimates, (2) Sparsity assumption cannot be easily incorporated into their method, and (3) Extension to general graph G is nontrivial. In contrast, T-LoHo overcomes all these limitations and effectively finds clusters by its flexible low-rank structure. Below, we show that the use of horseshoe prior not only introduces shrinkage of $\boldsymbol{\beta}$ but also has a less obvious but profound clustering effect to facilitate homogeneity pursuit.

To analyze the clustering effect of T-LoHo, we focus on the simple case when $\mathbf{X} = \mathbf{I}_n$ so that observations y_i are independent and normally distributed, $y_i|\beta_i, \sigma^2 \sim \mathcal{N}(\beta_i, \sigma^2)$, $i = 1, \ldots, n$. Here τ is assumed to be fixed. Then split step (1-a) in Alg. 1 corresponds to the comparison between the current model $\mathcal{M}_1 := (\Pi, K)$ and the proposed split model $\mathcal{M}_2 := (\Pi^\star, K^\star = K+1)$, with acceptance probability $\min\{1, \mathcal{A}_a \cdot \mathcal{P}_a \cdot \mathcal{L}_a\}$. Note the product of prior and proposal ratios $\mathcal{A}_a \cdot \mathcal{P}_a$ is proportional to $(1-c)$ which reflects the model size penalization prior $p(K) \propto (1-c)^K$. Here we will particularly focus on the likelihood ratio $\mathcal{L}_a = p(\boldsymbol{y}|\mathcal{M}_2)/p(\boldsymbol{y}|\mathcal{M}_1)$ where parameters $(\tilde{\boldsymbol{\beta}}, \sigma^2, \Lambda)$ are all integrated out (the likelihood ratio of merge step (1-b) is simply inverted, $\mathcal{L}_b = p(\boldsymbol{y}|\mathcal{M}_1)/p(\boldsymbol{y}|\mathcal{M}_2)$). Notice that $\mathcal{L}$ exactly corresponds to the Bayes factor of the Bayesian two-sample t test [13] which compares two means μ_1 and μ_2 of the clusters $\mathcal{C}_1$ and $\mathcal{C}_2$:

$$\mathcal{M}_1 : \text{mean of } \{y_i\}_{i\in\mathcal{C}_1} = \mu_1 = \mu_2 = \text{mean of } \{y_i\}_{i\in\mathcal{C}_2} \quad \text{v.s.} \quad \mathcal{M}_2 : \mu_1 \neq \mu_2$$

Following the formulation of Bayesian two-sample t test, priors are reparametrized as $(\delta, \bar{\mu}, \sigma^2) := ((\mu_1 - \mu_2)/\sigma, (\mu_1 + \mu_2)/2, \sigma^2)$ from (μ_1, μ_2, σ^2) because standardized difference $\delta = (\mu_1 - \mu_2)/\sigma$ is the parameter of interest. Noninformative prior on nuisance parameters $p(\bar{\mu}, \sigma^2) \propto 1/\sigma^2$ ensures that Bayes factor depends on the data only through the two-sample t statistics t:

$$t = (\bar{y}_1 - \bar{y}_2)/(s_p/\sqrt{n_\delta}), \quad s_p^2 = \left((n_1-1)s_1^2 + (n_2-1)s_2^2\right)/\nu, \quad n_\delta = (n_1^{-1} + n_2^{-1})^{-1}$$

where $n_k = |\mathcal{C}_k|$, $\nu = n_1 + n_2 - 2$ and $\bar{y}_k, s_k^2$ are the sample mean and variance of group k, $k = 1, 2$.

Now we compare how prior on δ makes difference on the Bayes factor. If we have independent normal priors on μ_1 and μ_2 with variance proportional to σ^2, it induces a normal prior on δ. Under the prior $\delta \sim \mathcal{N}(0, 1)$, also called unit information prior [20], the Bayes factor $B_{12}^n = p(\boldsymbol{y}|\mathcal{M}_1)/p(\boldsymbol{y}|\mathcal{M}_2)$ is

$$B_{12}^n = \frac{\int p(\boldsymbol{y}|\delta = 0, \bar{\mu}, \sigma^2) p(\bar{\mu}, \sigma^2) d(\bar{\mu}, \sigma^2)}{\int p(\boldsymbol{y}|\delta, \bar{\mu}, \sigma^2) p(\delta, \bar{\mu}, \sigma^2) d(\delta, \bar{\mu}, \sigma^2)} = \frac{\left(1 + t^2/\nu\right)^{-(\nu+1)/2}}{(1+n_\delta)^{-1/2}\left(1 + t^2/(\nu(1+n_\delta))\right)^{-(\nu+1)/2}}$$

If we have independent horseshoe priors on μ_1, μ_2, it induces the following heavy-tailed prior on δ:

Proposition 2. *Let $\pi_{HS}(\mu|\sigma, \tau) = \int_0^\infty \mathcal{N}(\mu|0, \sigma^2\tau^2\lambda^2) C^+(\lambda|0, 1) d\lambda$ be a horseshoe prior. If μ_1 and μ_2 independently follow horseshoe, then it induces a distribution of the standardized difference $\delta := (\mu_1 - \mu_2)/\sigma$ given τ, denoted as $\pi_\Delta(\cdot|\tau)$, which can be written as a scale mixture of normal:*

$$\pi_\Delta(\delta|\tau) = \int_0^\infty \mathcal{N}(\delta|0, v\tau^2) \frac{2}{\pi\sqrt{v+1}(v+2)} dv$$

Details are provided in Appendix A3. The density π_Δ is bounded above around the origin which is different from π_{HS}; i.e., $\pi_\Delta(0|\tau) = 2\cosh^{-1}(\sqrt{2})/(\pi^{3/2}\tau) < \infty$. Its tail is heavier than π_{HS} and behaves similar with the Strawderman-Berger prior [37, 3]. See left panel of fig. 3 for the graphical illustration of π_Δ. Under the prior induced by horseshoe $\delta \sim \pi_\Delta(\cdot|\tau = 1)$, the Bayes factor B_{12}^{hs} is

$$B_{12}^{hs} = \frac{\left(1 + t^2/\nu\right)^{-(\nu+1)/2}}{\int_0^\infty (1 + n_\delta v)^{-1/2} \left(1 + \frac{t^2}{\nu(1+n_\delta v)}\right)^{-(\nu+1)/2} \frac{2}{\pi\sqrt{v+1}(v+2)} dv}$$

We compare $\log B_{12}^n$ and $\log B_{12}^{hs}$ as a function of $|t|$ when the group sizes are (1) balanced, $n_1 = n_2$, (2) unbalanced, $n_1 : n_2 = 10 : 1$ with $\nu \in \{10, 20, 30\}$, and display results at right panel of fig. 3.

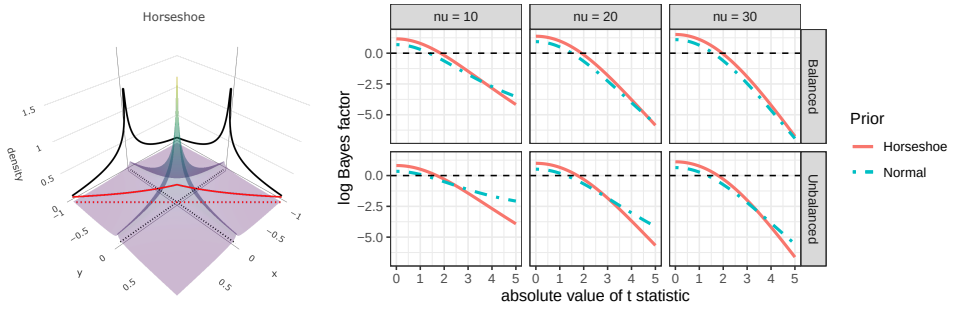

Figure 3: (Left) Joint density $f(x, y) = \pi_{HS}(x)\pi_{HS}(y)$ overlaid with marginal density $\pi_\Delta(x - y)$ shown as red, when $\tau = 1$. (Right) Comparison of $\log B_{12}^{hs}$(solid red) and $\log B_{12}^{n}$(broken blue) as a function of t under different (n_1, n_2) settings. Higher Bayes factor implies favoring $\mathcal{M}_1 : \mu_1 = \mu_2$.

When $|t|$ is small, we can see that $B_{12}^{hs} > B_{12}^{n}$ in all settings, which implies the prior $\delta \sim \pi_\Delta$ more strongly favors the one-group $\mathcal{M}_1$ over the two-group $\mathcal{M}_2$ compared to the normal prior $\delta \sim \mathcal{N}(0, 1)$. This can be explained by the fact that, under $\mathcal{M}_2$, heavy-tailed prior $\delta \sim \pi_\Delta$ anticipates large effect size *a priori* compared to the normal prior. Therefore T-LoHo facilitates cluster fusion when $|t|$ is small. Now when $|t|$ is large, we can see that π_Δ more strongly favors $\mathcal{M}_2$ compared to the normal prior, and the difference becomes more noticeable when group sizes are small and unbalanced. In fact, when (n_1, n_2) are fixed and $|t| \to \infty$, B_{12}^{n} never converges to 0 and is lower bounded by $(1 + n_\delta)^{-\nu/2} > 0$ which is also known as the *information paradox* [25]. In contrast, B_{12}^{hs} converges to 0 as $|t| \to \infty$ whenever $\nu \geq 1$. This finite sample consistency is important since it allows split step (1-a) to identify the (small, unbalanced) cluster with a high signal difference, which might not be possible under the usual normal prior because the penalty term $p(K) \propto (1 - c)^K$ may overwhelm it. In summary, compared to the normal prior, T-LoHo tends to reduce redundant cluster representations (when $|t|$ is small) while better captures the highly significant cluster differences (when $|t|$ is large).

3.3 Posterior Consistency Results

Notations. Let $(\boldsymbol{\beta}^*, \tilde{\boldsymbol{\beta}}^*, \sigma^*)$ denote the true $\boldsymbol{\beta}$, $\tilde{\boldsymbol{\beta}}$ and σ respectively. Let $\xi^* = \{j \in V : \beta_j^* \neq 0\}$ denote the true active set of indices. Let $\check{\Pi}$ denote an arbitrary partition of $V = \{1, \cdots, p\}$ whose corresponding partition of ξ^* is determined by removing those edges with $\beta_i^* - \beta_j^* > 0$ from the subgraph of any $\mathcal{F}$ compatible with $\check{\Pi}$ at vertex set ξ^*. We define $g_n^* = \max_{\check{\Pi}} |\check{\Pi}(\xi^*)|$ among all possible $\check{\Pi}$. Let P_n denote all unique partitions that have at most $g_n^*(1 + c_\delta)$ clusters and their corresponding partitions of ξ^* are nested in the true partition of ξ^* for some constant $c_\delta > 0$.

Below, we consider the case when p is much larger than n and establish posterior concentration results for the T-LoHo model as n goes to infinity. Our results rely on the following assumptions:

(A-1) The graph satisfies $g_n^* \prec n/\log p$, $n_c = o(g_n^*)$, and $\log |P_n| = O(g_n^* \log p)$.

(A-2) All the covariates are uniformly bounded. There exist some fixed constant $\lambda_0 > 0$, such that $\lambda_{\min}(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}) \geq n\lambda_0$ for any partition in P_n.

(A-3) $\max_j |\tilde{\beta}_j^*|/\sigma^* < L$, where $\log(L) = O(\log p)$.

(A-4) $-\log \tau = O(\log p)$, $\tau < p^{-(2+c_\tau)}\sqrt{g_n^* \log p/n}$, $1 - c \geq p^{-c_\alpha}$, and $\min_{\sigma^2 \in [\sigma^{*2},\, \sigma^{*2}(1+c_\sigma \varepsilon_n^2)]} \pi(\sigma^2) > 0$ for some positive constants c_τ, c_α and c_σ.

Assumption (A-1) is a regularity condition on G such that the resulting space of spanning forests is not too large. Assumption (A-2) is a commonly adopted condition on design matrix in high dimensional linear regressions. Assumption (A-3) bounds the growth rate of the standardized true coefficients. Assumption (A-4) is a condition on the prior distributions for prior distribution of τ, σ^2 and choice of hyperparameter c. The proof of theorem 1 is given in the Appendix A4.

Theorem 1. *(Posterior contraction) Under Assumptions (A-1) to (A-4), there exists a large enough constant $M_1 > 0$ and $\varepsilon_n \asymp \sqrt{g_n^* \log p/n}$ such that the posterior distribution satisfies $\pi_n(\|\boldsymbol{\beta} - \boldsymbol{\beta}^*\|_2 \geq M_1\sigma^*\varepsilon_n \mid \mathbf{y}) \leq \exp(-c_1 n\varepsilon_n^2)$ with probability $1 - \exp(-c_2 n\varepsilon_n^2)$ for some constants $c_1 > 0$ and $c_2 > 0$.*

4 Numerical Examples

4.1 Simulation Studies

We conduct a simulation study to demonstrate the utility of our model. Motivated by the scalar-on-image regression problem, we consider a similar simulation setting as [19]. We construct a 30×30 lattice graph which represents a structure of 2-D image. Column-standardized image predictors $\mathbf{X}_i \in \mathbb{R}^{900}, i = 1, \ldots, 100$ are generated from mean zero Gaussian process with kernel $K(x_j, x_l) = \exp(-d_{jl}/\vartheta)$ where d_{jl} is the distance between pixels j and l, ϑ is the range parameter with $\vartheta = 0$ indicating no dependence. True coefficient $\boldsymbol{\beta} \in \mathbb{R}^{900}$ is sparse (84% are zero) and has irregular cluster shapes with sharp discontinuities as shown in fig. 4(a). We let SNR $\in \{2, 4\}$ to set error variance $\sigma^2 = \mathrm{Var}(\mathbf{X}\boldsymbol{\beta})/\mathrm{SNR}$ and generate scalar responses $\boldsymbol{y} \in \mathbb{R}^{100}$ by $\boldsymbol{y} \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{I})$.

We compared our model with the soft-thresholded Gaussian process (STGP) [19, GPLv2] and fused lasso (FL) [40] with R package `genlasso`[1, GPLv2]. After collecting 4,000 posterior samples with 10^4 burn-in and 10 thin-in rate, we obtained posterior median estimate from T-LoHo with $(\tau_0, c) = (1, 0.5)$ and posterior mean estimate from STGP after thresholding. Here FL estimate is

$$\hat{\boldsymbol{\beta}}_{FL} = \mathrm{argmin}_{\boldsymbol{\beta}} \left\{ 0.5\|\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_{FL} \sum_{(j,l)\in E} |\beta_j - \beta_l| + \gamma_{FL} \cdot \lambda_{FL} \sum_{j=1}^{p} |\beta_j| \right\} \quad (9)$$

where we considered 3 candidates of $\gamma_{FL} \in \{0.2, 1, 5\}$ and selected λ_{FL} among 2000 number of steps using Bayesian information criteria. We used mean squared prediction error (MSPE) with test set size 1000 to measure the predictive power and Rand index (RI) [31] to measure the clustering accuracy. For T-LoHo, we used [7]'s method to obtain a cluster estimate among the MCMC draws. For STGP, thresholding posterior probabilities with cutoff 0.5 is used to classify zero/nonzero clusters. All computations were performed on Intel E5-2690 v3 CPU with 128GB of memory.

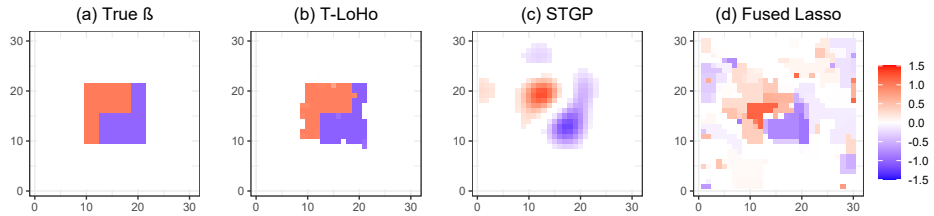

Figure 4: True and fitted result when $(\vartheta, \mathrm{SNR}) = (0, 4)$. (a) True coefficient image $\boldsymbol{\beta}$. (b) T-LoHo estimate with $(\tau_0, c) = (1, 0.5)$. (c) STGP estimate. (d) FL estimate with $\lambda_{FL} = 0.008$ ($\gamma_{FL} = 0.2$ which gives minimum MSPE). Results with other settings are available in Appendix A5.

Table 1: Performance comparison based on average MSPE and RI over 100 replicated simulations. Standard error is given in parentheses. RI $= 1$ indicates exact recovery of true cluster.

	ϑ	SNR	T-LoHo	STGP	Fused Lasso
MSPE	0	2	**68.48(30.04)**	93.4(17.09)	85.01(20.07)
	0	4	**24.39(19.57)**	86.27(15.75)	55.79(14.22)
	3	2	**250.94(112.08)**	277.76(52.56)	340.6(129.59)
	3	4	**59.73(23.24)**	163.89(21.6)	115.77(36.12)
Rand Index	0	2	**0.88(0.06)**	0.72(0.09)	0.75(0.03)
	0	4	**0.95(0.05)**	0.72(0.10)	0.79(0.03)
	3	2	**0.87(0.04)**	0.79(0.04)	0.80(0.03)
	3	4	**0.95(0.02)**	0.80(0.03)	0.86(0.02)
Time (in sec)	0	4	**107.92(3.77)**	339.88(16.70)	110.44(5.91)

From fig. 4, we can see that T-LoHo successfully captures the irregular shape of cluster boundaries and sharp discontinuities. STGP gives a much more smoothed estimate (which is expected because it does not assume homogeneity), and FL estimate contains many falsely identified nonzero clusters. The superior performance of T-LoHo over FL is partly attributed to (1) the use of horseshoe that reduces bias in FL, and (2) the use of RSF that more efficiently searches non-zero edge differences from a spanning forest instead of a larger graph G used in FL. Table 1 shows that in all (ϑ,SNR) settings, T-LoHo indeed outperforms other models in terms of both predictive and clustering accuracy.

4.2 Anomaly Detection in Road Networks

We apply T-LoHo to the problem of detecting anomalies in road graphs. NYC Pride March is an annual event held in every June in Manhattan, New York. As the march causes traffic congestion along the route, [42] considered the problem to detect clusters on road network which have different taxi pickup/dropoff patterns from usual. We constructed the road network graph using GRASS GIS [14] and got 3748 nodes (junctions) and 8474 edges (sections of roads). Following [42], we considered the event held in 2011 (12:00–2:00 pm, June 26th) and processed the number of taxi pickup/dropoff counts data [6, CC0] to the nearest nodes with log transformation. A log baseline seasonal average was calculated from the same time block 12:00–2:00 pm on the same day of the week across the nearest 8 weeks. Then it is plausible to assume that the difference between log counts on event day and log baseline seasonal average has many zeros and clustered patterns over the graph.

We fit T-LoHo and FL to compare the result. For T-LoHo, $(\tau_0, c) = (1, 0.8)$ were used to collect 5000 posterior samples with 1.5×10^5 burn-in and 10 thin-in rate. For the FL estimate (9), we followed [42]'s specification where λ_{FL} has 200 degrees of freedom and $\gamma_{FL} \cdot \lambda_{FL} = 0.2$.

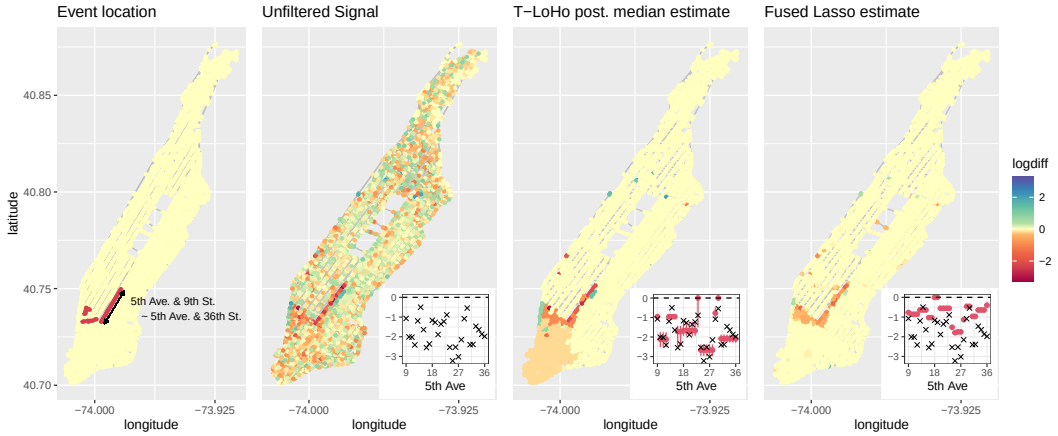

Figure 5: (Left two panels) 2011 NYC pride event route and unfiltered signal. Log-difference value below 0 indicates lower pickup/dropoff frequency than usual. (Right two panels) T-LoHo and FL estimates. (Bottom right subplots) Fitted value comparison zoomed along the parade route, 5th Ave.&9th St. to 5th Ave.&36th St. Here mark $\times$ indicates unfiltered signal value, red dot indicates estimated value, and red line indicates 90% credible interval for T-LoHo estimate.

Fig.5 shows the fitted result of two models. Both successfully capture the decreased taxi activity along the parade route and slightly increased taxi activity around the starting/ending point of the parade. From the subplots, we can see that FL is biased because of soft-thresholding [32] while T-LoHo appears to be less biased and also gives uncertainty measures. A notable difference is T-LoHo can capture the decreased taxi activity around the west lower Manhattan area while FL can't due to the bias. In summary, T-LoHo gives better insight on how taxi activity changes when such event occurs.

5 Concluding remarks

We propose a Tree-based Low-rank Horseshoe prior model to carry out Bayesian inference for graph-structured parameter which is assumed to be sparse and smooth. Accompanied with theoretical grounds and computational strategies, our simulation studies and real data example demonstrate that T-LoHo outperforms other competing methods such as fused lasso in a high dimensional linear regression context. Extensions to other types of high dimensional models are possible. Following a similar model construction spirit as T-LoHo, we can also build a general class of Tree-based Low-rank sparse homogeneity model extending other global-local shrinkage priors [29]. Another scenario not addressed in this paper is when we have a weighted graph $G = (V, E, w_0)$ as a parameter structure. In this case, incorporating *a priori* edge weight w_0 to T-LoHo is a nontrivial but interesting future research question which might be useful for many possible real data applications. This work does not present any foreseeable societal consequence, but users must be fully aware of the context represented as a graph G when giving interpretation on clustered parameters to avoid any misleading conclusions.

References

[1] Taylor B. Arnold and Ryan J. Tibshirani. *genlasso: Path Algorithm for Generalized Lasso Problems*, 2020. R package version 1.5.

[2] Sayantan Banerjee. Horseshoe shrinkage methods for Bayesian fusion estimation. *arXiv preprint arXiv:2102.07378*, 2021.

[3] James Berger et al. A robust generalized bayes estimator and confidence region for a multivariate normal mean. *Annals of Statistics*, 8(4):716–761, 1980.

[4] Anindya Bhadra, Jyotishka Datta, Nicholas G Polson, Brandon Willard, et al. Lasso meets horseshoe: A survey. *Statistical Science*, 34(3):405–427, 2019.

[5] Carlos M Carvalho, Nicholas G Polson, and James G Scott. The horseshoe estimator for sparse signals. *Biometrika*, 97(2):465–480, 2010.

[6] NYC Taxi & Limousine Commission. Tlc trip record data, 2021. https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page.

[7] David B Dahl. Model-based clustering for expression data via a dirichlet process mixture model. *Bayesian inference for gene expression and proteomics*, 4:201–218, 2006.

[8] Zhou Fan and Leying Guan. Approximate ℓ_0-penalized estimation of piecewise-constant signals on graphs. *The Annals of Statistics*, 46(6B):3217–3245, 2018.

[9] Wenning Feng, Chae Young Lim, Tapabrata Maiti, and Zhen Zhang. Spatial regression and estimation of disease risks: A clustering-based approach. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 9(6):417–434, 2016.

[10] Edward I George and Robert E McCulloch. Variable selection via gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993.

[11] Edward I George and Robert E McCulloch. Approaches for Bayesian variable selection. *Statistica sinica*, pages 339–373, 1997.

[12] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU press, 2013.

[13] Mithat Gönen, Wesley O Johnson, Yonggang Lu, and Peter H Westfall. The Bayesian two-sample t test. *The American Statistician*, 59(3):252–257, 2005.

[14] GRASS Development Team. *Geographic Resources Analysis Support System (GRASS GIS) Software*. Open Source Geospatial Foundation, USA, 2020.

[15] Peter J Green. Reversible jump markov chain monte carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[16] Rajarshi Guhaniyogi and David B Dunson. Bayesian compressed regression. *Journal of the American Statistical Association*, 110(512):1500–1514, 2015.

[17] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the lasso and generalizations*. CRC press, 2015.

[18] James E Johndrow, Paulo Orenstein, and Anirban Bhattacharya. Scalable approximate mcmc algorithms for the horseshoe prior. *Journal of Machine Learning Research*, 21(73):1–61, 2020.

[19] Jian Kang, Brian J Reich, and Ana-Maria Staicu. Scalar-on-image regression via the soft-thresholded Gaussian process. *Biometrika*, 105(1):165–184, 2018. code available at https://www4.stat.ncsu.edu/~bjreich/software.

[20] Robert E Kass and Adrian E Raftery. Bayes factors. *Journal of the american statistical association*, 90(430):773–795, 1995.

[21] Zheng Tracy Ke, Jianqing Fan, and Yichao Wu. Homogeneity Pursuit. *Journal of the American Statistical Association*, 110(509):175–194, 2015.

[22] Leonhard Knorr-Held and Günter Raßer. Bayesian detection of clusters and discontinuities in disease maps. *Biometrics*, 56(1):13–21, 2000.

[23] Minjung Kyung, Jeff Gill, Malay Ghosh, George Casella, et al. Penalized regression, standard errors, and Bayesian lassos. *Bayesian Analysis*, 5(2):369–411, 2010.

[24] Furong Li and Huiyan Sang. Spatial homogeneity pursuit of regression coefficients for large datasets. *Journal of the American Statistical Association*, 114(527):1050–1062, 2019.

[25] Feng Liang, Rui Paulo, German Molina, Merlise A Clyde, and Jim O Berger. Mixtures of g priors for Bayesian variable selection. *Journal of the American Statistical Association*, 103(481):410–423, 2008.

[26] Zhao Tang Luo, Huiyan Sang, and Bani Mallick. A Bayesian contiguous partitioning method for learning clustered latent variables. *Journal of Machine Learning Research*, 22(37):1–52, 2021.

[27] Radford M Neal. Slice sampling. *Annals of statistics*, pages 705–741, 2003.

[28] Oscar Hernan Madrid Padilla, James Sharpnack, James G Scott, and Ryan J Tibshirani. The DFS fused lasso: Linear-time denoising over general graphs. *J. Mach. Learn. Res.*, 18:176–1, 2017.

[29] Nicholas G Polson and James G Scott. Shrink globally, act locally: Sparse Bayesian regularization and prediction. *Bayesian statistics*, 9(501-538):105, 2010.

[30] Nicholas G Polson, James G Scott, and Jesse Windle. The Bayesian bridge. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pages 713–733, 2014.

[31] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

[32] Alessandro Rinaldo et al. Properties and refinements of the fused lasso. *Annals of Statistics*, 37(5B):2922–2952, 2009.

[33] Håvard Rue. Fast sampling of Gaussian markov random fields. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):325–338, 2001.

[34] Kaito Shimamura, Masao Ueki, Shuichi Kawano, and Sadanori Konishi. Bayesian generalized fused lasso modeling via neg distribution. *Communications in Statistics-Theory and Methods*, 48(16):4132–4153, 2019.

[35] Minsuk Shin, Anirban Bhattacharya, and Valen E Johnson. Functional horseshoe priors for subspace shrinkage. *Journal of the American Statistical Association*, 115(532):1784–1797, 2020.

[36] Qifan Song and Guang Cheng. Bayesian fusion estimation via t shrinkage. *Sankhya A*, 82(2):353–385, 2020.

[37] William E Strawderman. Proper bayes minimax estimators of the multivariate normal mean. *The Annals of Mathematical Statistics*, 42(1):385–388, 1971.

[38] Lu Tang and Peter XK Song. Fused lasso approach in regression coefficients clustering: learning parameter heterogeneity in data integration. *The Journal of Machine Learning Research*, 17(1):3915–3937, 2016.

[39] Leonardo V Teixeira, Renato M Assunção, and Rosangela H Loschi. Bayesian space-time partitioning by sampling and pruning spanning trees. *Journal of Machine Learning Research*, 20(85):1–35, 2019.

[40] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

[41] Ryan J Tibshirani, Jonathan Taylor, et al. The solution path of the generalized lasso. *The Annals of Statistics*, 39(3):1335–1371, 2011.

[42] Yu-Xiang Wang, James Sharpnack, Alexander J. Smola, and Ryan J. Tibshirani. Trend filtering on graphs. *Journal of Machine Learning Research*, 17(105):1–41, 2016.

[43] Yunzhang Zhu, Xiaotong Shen, and Wei Pan. Simultaneous grouping pursuit and feature selection over an undirected graph. *Journal of the American Statistical Association*, 108(502):713–725, 2013.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See introduction, section 1.
   (b) Did you describe the limitations of your work? [Yes] See introduction, section 1 and conclusion, section 5. Recall that our model assumes the sparse homogeneity assumption on $\boldsymbol{\beta}$.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See last part of the conclusion, section 5.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See section 2.2, section 3.2 and section 3.3.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix A2, A3 and A4.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See attached code in the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See section 4.1 and Appendix A5.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See section 4.1, parenthesis in table 1.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See section 4.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See section 4.1 and section 4.2.
   (b) Did you mention the license of the assets? [Yes] See section 4.1 and section 4.2.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See attached code in the supplementary material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] The real data we are using is the NYC taxi trip data distributed by NYC Taxi & Limousine Commission(TLC), see section 4.2. We checked that taxi drivers agreed to the TLC rule §66-24(f) regarding trip data collection and transmission.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] The real data we are using is the NYC taxi trip data(section 4.2). We checked this data does not contain any personally identifiable information or offensive content.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]