Journal of Machine Learning Research 12 (2011) 3371-3412 Submitted 6/10; Revised 12/10; Published 11/11

Learning with Structured Sparsity

Junzhou Huang JZHUANG@UTA.EDU
Department of Computer Science and Engineering
University of Texas at Arlington
Arlington, TX, 76019 USA *

Tong Zhang TZHANG@STAT.RUTGERS.EDU
Department of Statistics
Rutgers University
Piscataway, NJ, 08854 USA †

Dimitris Metaxas DNM@CS.RUTGERS.EDU
Department of Computer Science
Rutgers University
Piscataway, NJ, 08854 USA

Editor: Francis Bach

Abstract

This paper investigates a learning formulation called *structured sparsity*, which is a natural extension of the standard sparsity concept in statistical learning and compressive sensing. By allowing arbitrary structures on the feature set, this concept generalizes the group sparsity idea that has become popular in recent years. A general theory is developed for learning with structured sparsity, based on the notion of coding complexity associated with the structure. It is shown that if the coding complexity of the target signal is small, then one can achieve improved performance by using coding complexity regularization methods, which generalize the standard sparse regularization. Moreover, a structured greedy algorithm is proposed to efficiently solve the structured sparsity problem. It is shown that the greedy algorithm approximately solves the coding complexity optimization problem under appropriate conditions. Experiments are included to demonstrate the advantage of structured sparsity over standard sparsity on some real applications.

Keywords: structured sparsity, standard sparsity, group sparsity, tree sparsity, graph sparsity, sparse learning, feature selection, compressive sensing

1. Introduction

We are interested in the sparse learning problem under the fixed design condition. Consider a fixed set of p basis vectors $\{\mathbf{x}_1,\ldots,\mathbf{x}_p\}$ where $\mathbf{x}_j \in \mathbb{R}^n$ for each j. Here, n is the sample size. Denote by X the $n \times p$ data matrix, with column j of X being $\mathbf{x}_j$. Given a random observation $\mathbf{y} = [\mathbf{y}_1,\ldots,\mathbf{y}_n] \in \mathbb{R}^n$ that depends on an underlying coefficient vector $\bar{\beta} \in \mathbb{R}^p$, we are interested in the problem of estimating $\bar{\beta}$ under the assumption that the target coefficient $\bar{\beta}$ is sparse. Throughout the paper, we consider fixed design only. That is, we assume X is fixed, and randomization is with respect to the noise in the observation $\mathbf{y}$.

*. An extended abstract of the paper was presented at the international conference of machine learning, 2009.

†. This author is partially supported by NSF DMS-1007527, NSF IIS-1016061, and AFOSR-10097389.

We consider the situation that the true mean of the observation $\mathbb{E}\mathbf{y}$ can be approximated by a sparse linear combination of the basis vectors. That is, there exists a target vector $\beta \in \mathbb{R}^p$ such that either $\mathbb{E}\mathbf{y} = X\beta$ or $\mathbb{E}\mathbf{y} - X\beta$ is small. Moreover, we assume that β is sparse. Define the support of a vector $\beta \in \mathbb{R}^p$ as

$$\mathrm{supp}(\beta) = \{j : \beta_j \neq 0\},$$

and $\|\beta\|_0 = |\mathrm{supp}(\beta)|$. A natural method for sparse learning is L_0 regularization:

$$\hat{\beta}_{L0} = \arg\min_{\beta \in \mathbb{R}^p} \hat{Q}(\beta) \quad \text{subject to } \|\beta\|_0 \leq s, \tag{1}$$

where s is the desired sparsity. For simplicity, unless otherwise stated, the objective function considered throughout this paper is the least squares loss

$$\hat{Q}(\beta) = \|X\beta - \mathbf{y}\|_2^2,$$

where $\|\cdot\|_2$ denotes the Euclidean norm.

Since this optimization problem is generally NP-hard, in practice, one often considers approximate solutions. A standard approach is convex relaxation of L_0 regularization to L_1 regularization, often referred to as Lasso (Tibshirani, 1996). Another commonly used approach is greedy algorithms, such as the orthogonal matching pursuit (OMP) (Tropp and Gilbert, 2007).

In practical applications, one often knows a structure on the coefficient vector β in addition to sparsity. For example, in group sparsity, one assumes that variables in the same group tend to be zero or nonzero simultaneously. The purpose of this paper is to study the more general estimation problem under structured sparsity. If meaningful structures exist, we show that one can take advantage of such structures to improve the standard sparse learning. Specifically, we study the following natural extension of L_0 regularization to structured sparsity problems. It replaces the L_0 constraint in (1) by a more general term $c(\beta)$, which we call *coding complexity*. The precise definition will be given later in Section 2, and some concrete examples will be given later in Section 4.

$$\hat{\beta}_{constr} = \arg\min_{\beta \in \mathbb{R}^p} \hat{Q}(\beta) \quad \text{subject to } c(\beta) \leq s. \tag{2}$$

In this formulation, s is a tuning parameter. Alternatively, we may also consider the penalized formulation

$$\hat{\beta}_{pen} = \arg\min_{\beta \in \mathbb{R}^p} \left[\hat{Q}(\beta) + \lambda c(\beta)\right], \tag{3}$$

where $\lambda > 0$ is a regularization parameter that can be tuned. Since (2) and (3) penalize the coding complexity $c(\beta)$, we shall call this approach *coding complexity regularization*.

The optimization of either (2) or (3) is generally hard. For related problems, there are two common approaches to alleviate this difficulty. One is convex relaxation (L_1 regularization to replace L_0 regularization for standard sparsity); the other is forward greedy selection (also called orthogonal matching pursuit or OMP). We do not know any extensions of L_1 regularization like convex relaxation methods that can handle general structured sparsity formulations with provable performance guarantees. In particular, the theoretical analysis in our companion paper (Huang and Zhang, 2010) for group Lasso fails to yield meaningful bounds for more complex convex relaxation methods that are proposed for general structured sparsity formulations considered in this paper. For this reason, we present an extension of the standard greedy OMP algorithm that can be applied to general structured sparsity problems, and more importantly, meaningful sparse recovery bounds can be obtained for this algorithm. We call the resulting procedure *structured greedy algorithm* or StructOMP, which approximately solves (2). The details will be described later in Section 3.

1.1 Related Work

The idea of using structure in addition to sparsity has been explored before. An example is group structure, which has received much attention recently. For example, group sparsity has been considered for simultaneous sparse approximation (Wipf and Rao, 2007) and multi-task compressive sensing and learning (Argyriou et al., 2008; Ji et al., 2008) from the Bayesian hierarchical modeling point of view. Under the Bayesian hierarchical model framework, data from all sources contribute to the estimation of hyper-parameters in the sparse prior model. The shared prior can then be inferred from multiple sources. He et al. recently extend the idea to the tree sparsity in the Bayesian framework (He and Carin, 2009a,b). Although the idea can be justified using standard Bayesian intuition, there are no theoretical results showing how much better (and under what kind of conditions) the resulting algorithms perform. In the statistical literature, Lasso has been extended to the group Lasso when there exist group/block structured dependencies among the sparse coefficients (Yuan and Lin, 2006).

However, none of the above mentioned work was able to show advantage of using group structure. Although some theoretical results were developed in Bach (2008) and Nardi and Rinaldo (2008), neither showed that group Lasso is superior to the standard Lasso. Koltchinskii and Yuan (2008) showed that group Lasso can be superior to standard Lasso when each group is an infinite dimensional kernel, by relying on the fact that meaningful analysis can be obtained for kernel methods in infinite dimension. Obozinski et al. (2008) considered a special case of group Lasso in the multi-task learning scenario, and showed that the number of samples required for recovering the exact support set is smaller for group Lasso under appropriate conditions. Huang and Zhang (2010) developed a theory for group Lasso using a concept called strong group sparsity, which is a special case of the general structured sparsity idea considered here. It was shown in Huang and Zhang (2010) that group Lasso is superior to standard Lasso for strongly group-sparse signals, which provides a convincing theoretical justification for using group structured sparsity. Related results can also be found in Chesneau and Hebiri (2008) and Lounici et al. (2009).

While group Lasso works under the strong group sparsity assumption, it doesn't handle the more general structures considered in this paper. Several limitations of group Lasso were mentioned by Huang and Zhang (2010). For example, group Lasso does not correctly handle overlapping groups (in that overlapping components are over-counted); that is, a given coefficient should not belong to different groups. This requirement is too rigid for many practical applications. To address this issue, a method called composite absolute penalty (CAP) is proposed in Zhao et al. (2009) which can handle overlapping groups. A satisfactory theory remains to be developed to rigorously demonstrate the effectiveness of the approach. In a related development, Kowalski and Torresani (2009) generalized the mixed norm penalty to structured shrinkage, which can identify structured significance maps and thus can handle the case of the overlapping groups. However, there were no additional theory to justify their methods.

It is also worth pointing out that independent of this paper, two recent work (Jacob et al., 2009; Jenatton et al., 2009) considered structured sparsity in the convex relaxation setting, and extended group Lasso to more complicated sparse regularization conditions. These work complement the idea considered in this paper, which focuses on a natural non-convex formulation of general structured sparsity, as well as its greedy approximation. Again, since convex relaxation methods are more difficult to analyze in the structured sparsity setting with overlapping groups, a satisfactory theoretical justification remains an open challenge. For example the analysis in our companion work (Huang

and Zhang, 2010) on group Lasso does not correctly generalize to the above mentioned convex relaxation formulations because a straight-forward application leads to a bound proportional to the number of overlapping groups covering a true variable. Unfortunately, at least for some of the structures considered in this paper (such as hierarchical tree structure), in order to show the effectiveness of using the extra structural information, we need $\Omega(\log_2(p))$ groups to cover each variable, which leads to a bound showing no benefits over standard Lasso if we directly apply the analysis of Huang and Zhang (2010). It is worth noting that the lack of analysis doesn't mean that formulations in Jacob et al. (2009) and Jenatton et al. (2009) are ineffective. For example, some algorithmic techniques are employed by Jenatton et al. (2009) to address the over-counting issue we mentioned above, but the resulting procedures are non-trivial to analyze. In comparison the greedy algorithm is easier to analyze and (being non-convex) doesn't suffer from the above mentioned problem. Therefore this paper focuses on developing a direct generalization of the popular OMP algorithm to handle structured sparsity.

In addition to the above mentioned work, other structures have also been explored in the literature. For example, so-called tonal and transient structures were considered for sparse decomposition of audio signals in Daudet (2004). Grimm et al. (2007) investigated positive polynomials with structured sparsity from an optimization perspective. The theoretical result there did not address the effectiveness of such methods in comparison to standard sparsity. The closest work to ours is a recent paper by Baraniuk et al. (2010). In that paper, model based sparsity was considered and the structures comes from the predefined models. It is important to note that some theoretical results were obtained there to show the effectiveness of their method in compressive sensing. Moreover a generic algorithmic template was presented for structured sparsity. A drawback of the template is that it relies on finding the pruning of residue or signal estimates to a subset of variables with small structured complexity. These steps have to be specifically designed for different data models under specialized assumptions. In this regard, while the algorithmic template is generic, the actual implementation for the pruning steps will be quite different for different types of structures (for example, see Cevher et al., 2009a,b). In other words, it does not provide a common scheme to represent their "models" for different structured sparsity data. Different structure representation schemes have to be built for different "models". It thus remains as an open issue how to develop a general theory for structured sparsity, together with a general algorithm based on a generic structure representation scheme that can be applied to a wide class of such problems. The Structured OMP algorithm, which is proposed in this paper, is an attempt to address this issue. Although each type of structures requires an appropriately chosen block set (see Section 3 and Section 4), the algorithmic implementation based on a generic structure representation scheme is the same for different structures. We note that in general it is much easier to pick an appropriate block set than to design a new pruning algorithm.

We see from the above discussion that there exists extensive literature on combining sparsity with structured priors, with empirical evidence showing that one can achieve better performance by imposing additional structures. However, it is still useful to establish a general theoretical framework for structured sparsity that can quantify its effectiveness, as well as an efficient algorithmic implementation. The goal of this paper is to develop such a general theory that addresses the following issues, where we pay special attention to the benefit of structured sparsity over the standard non-structured sparsity:

- quantifying structured sparsity;

- the minimal number of measurements required in compressive sensing;
- estimation accuracy under stochastic noise;
- an efficient algorithm that can solve a wide class of structured sparsity problems with meaningful sparse recovery performance bounds.

2. Coding Complexity Regularization

In structured sparsity, not all sparse patterns are equally likely. For example, in group sparsity, coefficients within the same group are more likely to be zeros or nonzeros simultaneously. This means that if a sparse coefficient vector's support set is consistent with the underlying group structure, then it is more likely to occur, and hence incurs a smaller penalty in learning. One contribution of this work is to formulate how to define structure on top of sparsity, and how to penalize each sparsity pattern. We then develop a theory for the corresponding penalized estimators (2) and (3).

2.1 Structured Sparsity and Coding Complexity

In order to formalize the idea of structured sparsity, we denote by $I = \{1,\ldots,p\}$ the index set of the coefficients. Consider any sparse subset $F \subset \{1,\ldots,p\}$, we assign a cost $\mathrm{cl}(F)$. In structured sparsity, the cost of F is an upper bound of the coding length of F (number of bits needed to represent F by a computer program) in a pre-chosen prefix coding scheme. It is a well-known fact in information theory (e.g., Cover and Thomas, 1991) that mathematically, the existence of such a coding scheme is equivalent to

$$\sum_{F\subset I} 2^{-\mathrm{cl}(F)} \leq 1.$$

From the Bayesian statistics point of view, $2^{-\mathrm{cl}(F)}$ can be regarded as a lower bound of the probability of F. The probability model of structured sparse learning is thus: first generate the sparsity pattern F according to probability $2^{-\mathrm{cl}(F)}$; then generate the coefficients in F.

Definition 1 *A cost function* $\mathrm{cl}(F)$ *defined on subsets of* I *is called a coding length (in base-2) if*

$$\sum_{F\subset I, F\neq\emptyset} 2^{-\mathrm{cl}(F)} \leq 1.$$

We give $\emptyset$ *a coding length 0. The corresponding structured sparse coding complexity of* F *is defined as*

$$c(F) = |F| + \mathrm{cl}(F).$$

A coding length $\mathrm{cl}(F)$ *is sub-additive if*

$$\mathrm{cl}(F\cup F') \leq \mathrm{cl}(F) + \mathrm{cl}(F'),$$

and a coding complexity $c(F)$ *is sub-additive if*

$$c(F\cup F') \leq c(F) + c(F').$$

Clearly if $\mathrm{cl}(F)$ is sub-additive, then the corresponding coding complexity $c(F)$ is also sub-additive. Note that for simplicity, we do not introduce a trade-off between $|F|$ and $\mathrm{cl}(F)$ in the definition of $c(F)$. However, in real applications, such a trade-off may be beneficial: for example we may define $c(F) = \gamma|F| + \mathrm{cl}(F)$, where γ is considered a tuning parameter in the algorithm.

Based on the structured coding complexity of subsets of I, we can now define the structured coding complexity of a sparse coefficient vector $\bar{\beta} \in \mathbb{R}^p$.

Definition 2 *Giving a coding complexity $c(F)$, the structured sparse coding complexity of a coefficient vector $\bar{\beta} \in \mathbb{R}^p$ is*

$$c(\bar{\beta}) = \min\{c(F) : \mathrm{supp}(\bar{\beta}) \subset F\}.$$

We will later show that if a coefficient vector $\bar{\beta}$ has a small coding complexity $c(\bar{\beta})$, then $\bar{\beta}$ can be effectively learned, with good in-sample prediction performance (in statistical learning) and reconstruction performance (in compressive sensing). In order to see why the definition requires adding $|F|$ to $\mathrm{cl}(F)$, we consider the generative model for structured sparsity mentioned earlier. In this model, the number of bits to encode a sparse coefficient vector is the sum of the number of bits to encode F (which is $\mathrm{cl}(F)$) and the number of bits to encode nonzero coefficients in F (this requires $O(|F|)$ bits up to a fixed precision). Therefore the total number of bits required is $\mathrm{cl}(F) + O(|F|)$. This information theoretical result translates into a statistical estimation result: without additional regularization, the learning complexity for least squares regression within any fixed support set F is $O(|F|)$. By adding the model selection complexity $\mathrm{cl}(F)$ for each support set F, we obtain an overall statistical estimation complexity of $O(\mathrm{cl}(F) + |F|)$. We would like to mention that the coding complexity approach in this paper is related to but extends the Union-of-Subspaces model of Lu and Do (2008), which corresponds to a hard assignment of $\mathrm{cl}(F)$ to be either a constant c or $+\infty$.

While the idea of using coding based penalization is clearly motivated by the minimum description length (MDL) principle, the actual penalty we obtain for structured sparsity problems is different from the standard MDL penalty for model selection. Moreover, our analysis differs from some other MDL based analysis (such as Haupt and Nowak, 2006) that only deals with minimization over a countably many candidate coefficients $\bar{\beta}$ (the candidates are chosen a priori). This difference is important in sparse learning, and analysis as in Haupt and Nowak (2006) cannot be applied to the estimators of (2) or (3). Therefore in order to prevent confusion, we avoid using MDL in our terminology. Nevertheless, one may consider our framework as a natural combination of the MDL idea and the modern sparsity analysis. We will consider detailed examples of $\mathrm{cl}(F)$ in Section 4.

2.2 Theory of Coding Complexity Regularization

We assume sub-Gaussian noise as follows.

Assumption 1 *Assume that $\{\mathbf{y}_i\}_{i=1,\ldots,n}$ are independent (but not necessarily identically distributed) sub-Gaussians: there exists a constant $\sigma \geq 0$ such that $\forall i$ and $\forall t \in \mathbb{R}$,*

$$\mathbb{E}_{\mathbf{y}_i} e^{t(\mathbf{y}_i - \mathbb{E}\mathbf{y}_i)} \leq e^{\sigma^2 t^2/2}.$$

Both Gaussian and bounded random variables are sub-Gaussian using the above definition. For example, if a random variable $\xi \in [a,b]$, then $\mathbb{E}_\xi e^{t(\xi - \mathbb{E}\xi)} \leq e^{(b-a)^2 t^2/8}$. If a random variable is Gaussian: $\xi \sim N(0,\sigma^2)$, then $\mathbb{E}_\xi e^{t\xi} \leq e^{\sigma^2 t^2/2}$.

The following property of sub-Gaussian noise is important in our analysis. Our simple proof yields a sub-optimal choice of the constants.

Proposition 3 *Let $P \in \mathbb{R}^{n\times n}$ be a projection matrix of rank k, and $\mathbf{y}$ satisfies Assumption 1. Then for all $\eta \in (0,1)$, with probability larger than $1-\eta$:*

$$\|P(\mathbf{y}-\mathbb{E}\mathbf{y})\|_2^2 \le \sigma^2[7.4k + 2.7\ln(2/\eta)].$$

We also need to generalize sparse eigenvalue condition, used in the modern sparsity analysis. It is related to (and weaker than) the RIP (restricted isometry property) assumption (Candes and Tao, 2005) in the compressive sensing literature. This definition takes advantage of coding complexity, and can be also considered as (a weaker version of) structured RIP. We introduce a definition.

Definition 4 *For all $F \subset \{1,\ldots,p\}$, define*

$$\rho_-(F) = \inf\left\{\frac{1}{n}\|X\beta\|_2^2/\|\beta\|_2^2 : \mathrm{supp}(\beta)\subset F\right\},$$
$$\rho_+(F) = \sup\left\{\frac{1}{n}\|X\beta\|_2^2/\|\beta\|_2^2 : \mathrm{supp}(\beta)\subset F\right\}.$$

Moreover, for all $s > 0$, define

$$\rho_-(s) = \inf\{\rho_-(F) : F \subset I, c(F) \le s\},$$
$$\rho_+(s) = \sup\{\rho_+(F) : F \subset I, c(F) \le s\}.$$

In the theoretical analysis, we need to assume that $\rho_-(s)$ is not too small for some s that is larger than the signal complexity. Since we only consider eigenvalues for submatrices with small cost $c(\bar{\beta})$, the sparse eigenvalue $\rho_-(s)$ can be significantly larger than the corresponding ratio for standard sparsity (which will consider all subsets of $\{1,\ldots,p\}$ up to size s). For example, for random projections used in compressive sensing applications, the coding length $c(\mathrm{supp}(\bar{\beta}))$ is $O(k\ln p)$ in standard sparsity, but can be as low as $c(\mathrm{supp}(\bar{\beta})) = O(k)$ in structured sparsity (if we can guess $\mathrm{supp}(\bar{\beta})$ approximately correctly. Therefore instead of requiring $n = O(k\ln p)$ samples, we require only $O(k + \mathrm{cl}(\mathrm{supp}(\bar{\beta})))$. The difference can be significant when p is large and the coding length $\mathrm{cl}(\mathrm{supp}(\bar{\beta})) \ll k\ln p$. An example for this is group sparsity, where we have p/k_0 even sized groups, and variables in each group are simultaneously zero or nonzero. The coding length of the groups are $(k/k_0)\ln(p/k_0)$, which is significantly smaller than $k\ln p$ when p is large (see Section 4 for details).

More precisely, we have the following random projection sample complexity bound for the structured sparse eigenvalue condition. The theorem implies that the structured RIP condition is satisfied with sample size $n = O(k + (k/k_0)\ln(p/k_0))$ in group sparsity (where $s = O(k + (k/k_0)\ln(p/k_0))$) rather than $n = O(k\ln(p))$ in standard sparsity (where $s = O(k\ln p)$). For hierarchical tree sparsity (see Section 4 for details), it requires $n = O(k)$ examples (with $s = O(k)$), which matches the result of Baraniuk et al. (2010). Therefore Theorem 6 shows that in the compressive sensing applications, it is possible to reconstruct signals with fewer number of random projections by using structured sparsity.

Proposition 5 (Structured-RIP) *Suppose that elements in X are iid standard Gaussian random variables $N(0,1)$. For any $t > 0$ and $\delta \in (0,1)$, let*

$$n \ge \frac{8}{\delta^2}[\ln 3 + t + s\ln(1+8/\delta)].$$

Then with probability at least $1-e^{-t}$, *the random matrix* $X \in \mathbb{R}^{n\times p}$ *satisfies the following structured-RIP inequality for all vector* $\bar{\beta} \in \mathbb{R}^p$ *with coding complexity no more than* s:

$$(1-\delta)\|\bar{\beta}\|_2 \le \frac{1}{\sqrt{n}}\|X\bar{\beta}\|_2 \le (1+\delta)\|\bar{\beta}\|_2. \tag{4}$$

Although in the theorem, we assume Gaussian random matrix in order to state explicit constants, it is clear that similar results hold for other sub-Gaussian random matrices. Note that the proposed generalization of RIP extends related results in compressive sensing and statistics (Baraniuk et al., 2010; Huang and Zhang, 2010).

The following result gives a performance bound for constrained coding complexity regularization in (2). The 2-norm parameter estimation bound $\|\hat{\beta}-\bar{\beta}\|_2$ requires that $\rho_-(\cdot) > 0$ (otherwise, the bound becomes trivial). For random design matrix X, the lower-bound in (4) is thus needed.

Theorem 6 *Suppose that Assumption 1 is valid. Consider any fixed target* $\bar{\beta} \in \mathbb{R}^p$. *Then with probability exceeding* $1-\eta$, *for all* $\varepsilon \ge 0$ *and* $\hat{\beta} \in \mathbb{R}^p$ *such that:* $\hat{Q}(\hat{\beta}) \le \hat{Q}(\bar{\beta})+\varepsilon$, *we have*

$$\|X\hat{\beta}-\mathbb{E}\mathbf{y}\|_2 \le \|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2 + \sigma\sqrt{2\ln(6/\eta)} + 2(7.4\sigma^2 c(\hat{\beta}) + 4.7\sigma^2\ln(6/\eta)+\varepsilon)^{1/2}.$$

Moreover, if the coding scheme $c(\cdot)$ *is sub-additive, then*

$$n\rho_-(c(\hat{\beta})+c(\bar{\beta}))\|\hat{\beta}-\bar{\beta}\|_2^2 \le 10\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 37\sigma^2 c(\hat{\beta}) + 29\sigma^2\ln(6/\eta) + 2.5\varepsilon.$$

This theorem immediately implies the following result for (2): $\forall\bar{\beta}$ such that $c(\bar{\beta}) \le s$,

$$\frac{1}{\sqrt{n}}\|X\hat{\beta}_{constr}-\mathbb{E}\mathbf{y}\|_2 \le \frac{1}{\sqrt{n}}\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2 + \frac{\sigma}{\sqrt{n}}\sqrt{2\ln(6/\eta)} + \frac{2\sigma}{\sqrt{n}}(7.4s+4.7\ln(6/\eta))^{1/2},$$

$$\|\hat{\beta}_{constr}-\bar{\beta}\|_2^2 \le \frac{1}{\rho_-(s+c(\bar{\beta}))n}\left[10\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 37\sigma^2 s + 29\sigma^2\ln(6/\eta)\right].$$

Although for simplicity this paper does not consider the problem of estimating $\rho_-(s+c(\bar{\beta}))$, it is possible to estimate it approximately (for example, using ideas of d'Aspremont et al., 2008). We can generally expect $\rho_-(s+c(\bar{\beta})) = O(1)$ by assuming that the sample size is sufficiently large according to Proposition 5. The result immediately implies that as sample size $n\to\infty$ and $s/n \to 0$, the root mean squared error prediction performance $\|X\hat{\beta}-\mathbb{E}\mathbf{y}\|_2/\sqrt{n}$ converges to the optimal prediction performance $\inf_{c(\bar{\beta})\le s}\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2/\sqrt{n}$. This result is agnostic in that even if $\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2/\sqrt{n}$ is large, the result is still meaningful because it says the performance of the estimator $\hat{\beta}$ is competitive to the best possible estimator in the class $c(\bar{\beta}) \le s$.

In compressive sensing applications, we take $\sigma = 0$, and we are interested in recovering $\bar{\beta}$ from random projections. For simplicity, we let $X\bar{\beta} = \mathbb{E}\mathbf{y} = \mathbf{y}$, and our result shows that the constrained coding complexity penalization method achieves exact reconstruction $\hat{\beta}_{constr} = \bar{\beta}$ as long as $\rho_-(2c(\bar{\beta})) > 0$ (by setting $s = c(\bar{\beta})$). According to Proposition 5, this is possible when the number of random projections (sample size) reaches $n = O(c(\bar{\beta}))$. This is a generalization of corresponding results in compressive sensing (Candes and Tao, 2005). As we have pointed out earlier, this number can be significantly smaller than the standard sparsity requirement of $n = O(\|\bar{\beta}\|_0\ln p)$, if the structure imposed in the formulation is meaningful.

As an example, for group sparsity (see Section 4), we consider m pre-defined groups, each of size k_0. If the support of $\bar{\beta}$ is covered by g of the m groups, we know from Section 4 that the

complexity can be defined as $s = g\log_2(2m) + gk_0$. In comparison, the standard sparsity complexity is given by $s = \|\bar{\beta}\|_0 \log_2(2p)$, which may be significantly larger if $g \ll \|\bar{\beta}\|_0$ (that is, the group structure is meaningful). It can be shown that the group-Lasso estimator may also achieve the group sparsity complexity of $s = g\log_2(2m) + gk_0$ (Huang and Zhang, 2010; Lounici et al., 2009), but the result for group-Lasso requires a stronger condition involving structured-RIP. Note that the first bound in Theorem 6 does not require any RIP assumption, while the second bound only requires a very weak dependency of the form $\rho_-(\cdot) > 0$. In contrast, the required dependency for group Lasso is significantly stronger, and details can be seen in Huang and Zhang (2010), Lounici et al. (2009) and Nardi and Rinaldo (2008). Although the result for the coding complexity estimator (2) is better due to weaker RIP dependency, we shall point out that it doesn't mean that for group sparsity, we should use (2) instead of group-Lasso in practice. This is because solving (2) requires non-convex optimization, while group-Lasso is a convex formulation. This is why we will consider an efficient algorithm to approximately solve (2) in Section 3.

Similar to Theorem 6, we can obtain the following result for (3). A related result for standard sparsity under Gaussian noise can be found in Bunea et al. (2007).

Theorem 7 *Suppose that Assumption 1 is valid. Consider any fixed target* $\bar{\beta} \in \mathbb{R}^p$. *Then with probability exceeding* $1-\eta$, *for all* $\lambda > 7.4\sigma^2$ *and* $a \geq 7.4\sigma^2/(\lambda - 7.4\sigma^2)$, *we have*

$$\|X\hat{\beta}_{pen} - \mathbb{E}\mathbf{y}\|_2^2 \leq (1+a)^2\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2^2 + (1+a)\lambda c(\bar{\beta}) + \sigma^2(10 + 5a + 7a^{-1})\ln(6/\eta).$$

Unlike the result for (2), the prediction performance $\|X\hat{\beta}_{pen} - \mathbb{E}\mathbf{y}\|_2$ of the estimator in (3) is competitive to $(1+a)\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2$, which is a constant factor larger than the optimal prediction performance $\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2$. By optimizing λ and a, it is possible to obtain a similar result as that of Theorem 6. However, this requires tuning λ, which is not as convenient as tuning s in (2). Note that both results presented here, and those in Bunea et al. (2007) are superior to the more traditional least squares regression results with λ explicitly fixed (for example, theoretical results for AIC). This is because one can only obtain the form presented in Theorem 6 by tuning λ. Such tuning is important in real applications.

3. Structured Greedy Algorithm

In this section, we describe a generalization of the OMP algorithm for standard sparsity. Our generalization, which we refer to as structured greedy algorithm or simply StructOMP, takes advantage of block structures to approximately solve the structured sparsity formulation (2). It would be worthwhile to mention that the notion of block structures here is different from block sparsity in model-based compressive sensing (Baraniuk et al., 2010).

Note that in this algorithm, we assume that $c(F)$ is relatively easy to compute (up to a constant) for any given F. For this purpose, we may use a relatively easy to compute upper bound of $c(F)$. For example, for graph structured sparsity described later in Section 4, we may simply use the right hand side of Proposition 11 as the definition of $c(F)$. If the maximum degree of a graph is small, we can simply use $c(F) = g\ln(p) + |F|$, where g is the number of connected components in F. For practical purposes, a multiplicative constant in the definition of $c(F)$ is not important because it can be absorbed into the tuning parameter s.

3.1 Algorithm Description

The main idea of StructOMP is to limit the search space of the greedy algorithm to small blocks. We will show that if a coding scheme can be approximated with blocks, then StructOMP is effective. Additional discussion of block approximation can be found in Section 4.

Formally, we consider a subset $\mathcal{B} \subset 2^I$. That is, each element (which we call a block or a base block) of $\mathcal{B}$ is a subset of I. We call $\mathcal{B}$ a block set if $I = \cup_{B\in\mathcal{B}} B$ and all single element sets $\{j\}$ belong to $\mathcal{B}$ ($j \in I$). Note that $\mathcal{B}$ may contain additional non single-element blocks. The requirement of $\mathcal{B}$ containing all single element sets is for notational convenience, as it implies that every subset $F \subset I$ can be expressed as the union of blocks in $\mathcal{B}$. Mathematically this requirement is non-important because we may simply assign ∞ coding length to single-element blocks, which is equivalent to excluding these single element sets.

Input: $(X, \mathbf{y})$, $\mathcal{B} \subset 2^I$, $s > 0$
Output: $F^{(k)}$ and $\beta^{(k)}$
let $F^{(0)} = \emptyset$ and $\beta^{(0)} = 0$
for $k = 1, 2, \ldots$
  select $B^{(k)} \in \mathcal{B}$ to maximize progress $\qquad (*)$
  let $F^{(k)} = B^{(k)} \cup F^{(k-1)}$
  let $\beta^{(k)} = \arg\min_{\beta\in\mathbb{R}^p} \hat{Q}(\beta)$ subject to $\mathrm{supp}(\beta) \subset F^{(k)}$
  if $(c(\beta^{(k)}) > s)$ **break**
end

Figure 1: Structured Greedy Algorithm

In Figure 1, we are given a set of blocks $\mathcal{B}$ that contains subsets of I. Instead of searching all subsets $F \subset I$ up to a certain complexity $|F| + c(F)$, which is computationally infeasible, we search only the blocks restricted to $\mathcal{B}$. It is assumed that searching over $\mathcal{B}$ is computationally manageable. In practice, the computational cost is linear in the number of base blocks $|\mathcal{B}|$.

At each step $(*)$, we try to find a block from $\mathcal{B}$ to maximize progress. It is thus necessary to define a quantity that measures progress. Our idea is to approximately maximize the gain ratio:

$$\frac{\hat{Q}(\beta^{(k-1)}) - \hat{Q}(\beta^{(k)})}{c(\beta^{(k)}) - c(\beta^{k-1})},$$

which measures the reduction of objective function per unit increase of coding complexity. This greedy criterion is a natural generalization of the standard greedy algorithm, and essential in our analysis. For least squares regression, we can define the gain ratio as follows:

$$\phi(B) = \frac{\|P_{B-F^{(k-1)}}(X\beta^{(k-1)} - \mathbf{y})\|_2^2}{c(B \cup F^{(k-1)}) - c(F^{(k-1)})}, \tag{5}$$

where

$$P_F = X_F (X_F^\top X_F)^+ X_F^\top$$

is the projection matrix to the subspaces generated by columns of X_F. Here $(X_F^\top X_F)^+$ denotes the Moore-Penrose pseudo-inverse.

More precisely, for least squares regression, at each step $(*)$ of Figure 1, we select a block $B^{(k)}$ that satisfies the condition

$$\phi(B^{(k)}) \geq \nu \max_{B \in \mathcal{B}} \phi(B) \tag{6}$$

for some $\nu \in (0,1]$. We may regard ν as a fixed approximation ratio (to ensure the quality of approximate optimization) that will appear in our analysis, although the algorithm does not have to pick ν a priori.

The reason to allow approximate maximization in (6) is that our practical implementation of StructOMP maximizes a simpler quantity

$$\tilde{\phi}(B) = \frac{\|X^\top_{B-F^{(k-1)}}(X\beta^{(k-1)} - \mathbf{y})\|_2^2}{c(B \cup F^{(k-1)}) - c(F^{(k-1)})}, \tag{7}$$

which is more efficient to compute (especially when blocks are overlapping). Since the ratio

$$\|X^\top_{B-F^{(k-1)}}\mathbf{r}\|_2^2 / \|P_{B-F^{(k-1)}}\mathbf{r}\|_2^2$$

is bounded between $\rho_+(B)$ and $\rho_-(B)$ (these quantities are defined in Definition 4), we know that maximization of $\tilde{\phi}(B)$ leads to an approximate maximization of $\phi(B)$ with $\nu \geq \rho_-(B)/\rho_+(B)$. That is, maximization of (7) in our practical StructOMP implementation corresponds to an approximate maximization in (6). Moreover, ν only appears in our analysis, and it does not appear explicitly in our implementation.

Note that we shall ignore $B \in \mathcal{B}$ such that $B \subset F^{(k-1)}$, and just let the corresponding gain to be 0. Moreover, if there exists a base block $B \not\subset F^{(k-1)}$ but $c(B \cup F^{(k-1)}) \leq c(F^{(k-1)})$, we can always select B and let $F^{(k)} = B \cup F^{(k-1)}$ (this is because it is always beneficial to add more features into $F^{(k)}$ without additional coding complexity). We assume this step is always performed if such a $B \in \mathcal{B}$ exists. The non-trivial case is $c(B \cup F^{(k-1)}) > c(F^{(k-1)})$ for all $B \in \mathcal{B}$; in this case both $\phi(B)$ and $\tilde{\phi}(B)$ are well defined.

3.2 Convergence Analysis

It is important to understand that the block structure is only used to limit the search space in the structured greedy algorithm. However, our theoretical analysis shows that if in addition, the underlying coding scheme can be approximated by block coding using base blocks employed in the greedy algorithm, then the algorithm is effective in minimizing (2). Although one does not need to know the specific approximation in order to use the greedy algorithm, knowing its existence (which can be shown for the examples discussed in Section 4) guarantees the effectiveness of the algorithm. It is also useful to understand that our result does not imply that the algorithm won't be effective if the actual coding scheme cannot be approximated by block coding.

We shall introduce a definition before stating our main results.

Definition 8 *Given $\mathcal{B} \subset 2^I$, define*

$$\rho_0(\mathcal{B}) = \max_{B \in \mathcal{B}} \rho_+(B), \qquad c_0(\mathcal{B}) = \max_{B \in \mathcal{B}} c(B)$$

and

$$c(\bar{\beta}, \mathcal{B}) = \min \left\{ \sum_{j=1}^{b} c(\bar{B}_j) : \mathrm{supp}(\bar{\beta}) \subset \bigcup_{j=1}^{b} \bar{B}_j \quad (\bar{B}_j \in \mathcal{B}); b \geq 1 \right\}.$$

The following theorem shows that if $c(\bar{\beta}, \mathcal{B})$ is small, then one can use the structured greedy algorithm to find a coefficient vector $\beta^{(k)}$ that is competitive to $\bar{\beta}$, and the coding complexity $c(\beta^{(k)})$ is not much worse than that of $c(\bar{\beta}, \mathcal{B})$. This implies that if the original coding complexity $c(\bar{\beta})$ can be approximated by block complexity $c(\bar{\beta}, \mathcal{B})$, then we can approximately solve (2).

Theorem 9 *Suppose the coding scheme is sub-additive. Consider* $\bar{\beta}$ *and* ε *such that*

$$\varepsilon \in (0, \|\mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2]$$

and

$$s \geq \frac{\rho_0(\mathcal{B}) c(\bar{\beta}, \mathcal{B})}{\nu \rho_-(s + c(\bar{\beta}))} \ln \frac{\|\mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2}{\varepsilon}.$$

Then at the stopping time k, we have

$$\hat{Q}(\beta^{(k)}) \leq \hat{Q}(\bar{\beta}) + \varepsilon.$$

By Theorem 6, the result in Theorem 9 implies that

$$\|X\beta^{(k)} - \mathbb{E}\mathbf{y}\|_2 \leq \|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2 + \sigma\sqrt{2\ln(6/\eta)} + 2\sigma\sqrt{7.4(s + c_0(\mathcal{B})) + 4.7\ln(6/\eta) + \varepsilon/\sigma^2},$$
$$\|\beta^{(k)} - \bar{\beta}\|_2^2 \leq \frac{10\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2^2 + 37\sigma^2(s + c_0(\mathcal{B})) + 29\sigma^2\ln(6/\eta) + 2.5\varepsilon}{\rho_-(s + c_0(\mathcal{B}) + c(\bar{\beta}))n}.$$

The result shows that in order to approximate a signal $\bar{\beta}$ up to accuracy ε, one needs to use coding complexity $O(\ln(1/\varepsilon))c(\bar{\beta}, \mathcal{B})$. Now, consider the case that $\mathcal{B}$ contains small blocks and their sub-blocks with equal coding length, and the actual coding scheme can be approximated (up to a constant) by block coding generated by $\mathcal{B}$; that is, $c(\bar{\beta}, \mathcal{B}) = O(c(\bar{\beta}))$. In this case we need $O(s\ln(1/\varepsilon))$ to approximate a signal with coding complexity s. For this reason, we will extensively discuss block approximation in Section 4.

In order to improve forward greedy procedures, backward greedy strategies can be employed, as shown in various recent works such as Zhang (2011). For simplicity, we will not analyze such strategies in this paper. It is worth mentioning that in practice, greedy algorithm is often adequate. In particular the $O(\ln(1/\varepsilon))$ factor vanishes for a weakly sparse target signal $\bar{\beta}$, where the magnitude of its coefficients gradually decrease to zero. This concept has been considered in previous work such as Donoho (2006) and Baraniuk et al. (2010). In such case, we may choose an appropriate optimal stopping point to avoid the $O(\ln(1/\varepsilon))$ factor. In fact, practitioners often observe that OMP can be more effective than Lasso for weakly sparse target signals (in spite of stronger theoretical results for Lasso with strongly sparse target signals). This will be confirmed in our experiments as well. Without cluttering the main text, we leave the detailed analysis of StructOMP for weakly sparse signals to Appendix F. Our analysis is the first theoretical justification of this empirical phenomenon.

4. Structured Sparsity Examples

Before giving detailed examples, we describe a general coding scheme called *block coding*, which is an expansion of Definition 8. The basic idea of block coding is to define a coding scheme on

a small number of base blocks (a block is a subset of I), and then define a coding scheme on all subsets of I using these base blocks.

Consider block set $\mathcal{B} \subset 2^I$. We assume that every subset $F \subset I$ can be expressed as the union of blocks in $\mathcal{B}$. Let cl_0 be a code length on $\mathcal{B}$:

$$\sum_{B \in \mathcal{B}} 2^{-\mathrm{cl}_0(B)} \leq 1,$$

we define $\mathrm{cl}(B) = \mathrm{cl}_0(B) + 1$ for $B \in \mathcal{B}$. It not difficult to show that the following cost function on $F \subset I$ is a coding length

$$\mathrm{cl}_{\mathcal{B}}(F) = \min \left\{ \sum_{j=1}^{b} \mathrm{cl}(B_j) : F = \bigcup_{j=1}^{b} B_j \quad (B_j \in \mathcal{B}) \right\}.$$

This is because

$$\sum_{F \subset I, F \neq \emptyset} 2^{-\mathrm{cl}(F)} \leq \sum_{b \geq 1} \sum_{B_\ell \in \mathcal{B}: 1 \leq \ell \leq b} 2^{-\sum_{\ell=1}^{b} \mathrm{cl}(B_\ell)} \leq \sum_{b \geq 1} \prod_{\ell=1}^{b} \sum_{B_\ell \in \mathcal{B}} 2^{-\mathrm{cl}(B_\ell)} \leq \sum_{b \geq 1} 2^{-b} = 1.$$

We call the coding scheme $\mathrm{cl}_{\mathcal{B}}$ block coding. It is clear from the definition that block coding is sub-additive.

From Theorem 9 and the discussions thereafter, we know that under appropriate conditions, a target coefficient vector with a small block coding complexity can be approximately learned using the structured greedy algorithm. This means that the block coding scheme has important algorithmic implications. That is, if a coding scheme can be approximated by block coding with a small number of base blocks, then the corresponding estimation problem can be approximately solved using the structured greedy algorithm.

For this reason, we shall pay special attention to block coding approximation schemes for examples discussed below. In particular, a coding scheme $\mathrm{cl}(\cdot)$ can be polynomially approximated by block coding if there exists a block coding scheme $\mathrm{cl}_{\mathcal{B}}$ with polynomial (in p) number of base blocks in $\mathcal{B}$, such that there exists a positive constant $C_{\mathcal{B}}$ independent of p:

$$\mathrm{cl}_{\mathcal{B}}(F) \leq C_{\mathcal{B}} \mathrm{cl}(F).$$

That is, up to a constant, the block coding scheme $\mathrm{cl}_{\mathcal{B}}()$ is dominated by the coding scheme $\mathrm{cl}()$.

While it is possible to work with blocks with non-uniform coding schemes, for simplicity examples provided in this paper only consider blocks with uniform coding, which is similar to the representation used in the Union-of-Subspaces model of Lu and Do (2008).

4.1 Standard Sparsity

A simple coding scheme is to code each subset $F \subset I$ of cardinality k using $k \log_2(2p)$ bits, which corresponds to block coding with $\mathcal{B}$ consisted only of single element sets, and each base block has a coding length $\mathrm{cl}_0 = \log_2 p$. This corresponds to the complexity for the standard sparse learning.

A more general version is to consider single element blocks $\mathcal{B} = \{\{j\} : j \in I\}$, with a non-uniform coding scheme $\mathrm{cl}_0(\{j\}) = c_j$, such that $\sum_j 2^{-c_j} \leq 1$. It leads to a non-uniform coding length on I as

$$\mathrm{cl}(B) = |B| + \sum_{j \in B} c_j.$$

In particular, if a feature j is likely to be nonzero, we should give it a smaller coding length c_j, and if a feature j is likely to be zero, we should give it a larger coding length. In this case, a subset $F \subset I$ has coding length $\mathrm{cl}(F) = \sum_{j \in F}(1 + c_j)$.

4.2 Group Sparsity

The concept of group sparsity has appeared in various recent work, such as the group Lasso in Yuan and Lin (2006) or multi-task learning in Argyriou et al. (2008). Consider a partition of $I = \cup_{j=1}^{m} G_j$ into m disjoint groups. Let $\mathcal{B}_G$ contain the m groups $\{G_j\}$, and $\mathcal{B}_1$ contain p single element blocks. The strong group sparsity coding scheme is to give each element in $\mathcal{B}_1$ a code-length cl_0 of ∞, and each element in $\mathcal{B}_G$ a code-length cl_0 of $\log_2 m$. Then the block coding scheme with blocks $\mathcal{B} = \mathcal{B}_G \cup \mathcal{B}_1$ leads to group sparsity, which only looks for signals consisted of the groups. The resulting coding length is: $\mathrm{cl}(B) = g \log_2(2m)$ if B can be represented as the union of g disjoint groups G_j; and $\mathrm{cl}(B) = \infty$ otherwise.

Note that if the support of the target signal F can be expressed as the union of g groups, and each group size is k_0, then the group coding length $g \log_2(2m)$ can be significantly smaller than the standard sparsity coding length of $|F| \log_2(2p) = g k_0 \log_2(2p)$. As we shall see later, the smaller coding complexity implies better learning behavior, which is essentially the advantage of using group sparse structure. It was shown by Huang and Zhang (2010) that strong group sparsity defined above also characterizes the performance of group Lasso. Therefore if a signal has a pre-determined group structure, then group Lasso is superior to the standard Lasso.

An extension of this idea is to allow more general block coding length for $\mathrm{cl}_0(G_j)$ and $\mathrm{cl}_0(\{j\})$ so that

$$\sum_{j=1}^{m} 2^{-\mathrm{cl}_0(G_j)} + \sum_{j=1}^{p} 2^{-\mathrm{cl}_0(\{j\})} \leq 1.$$

This leads to non-uniform coding of the groups, so that a group that is more likely to be nonzero is given a smaller coding length. If feature set F can be represented as the union of g groups $G_{j_1}, \ldots, G_{j_g}$, then its coding length is $\mathrm{cl}(F) = g + \sum_{j=1}^{g} \mathrm{cl}_0(G_j)$.

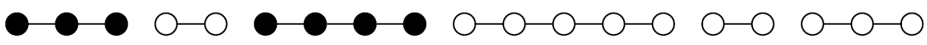


Figure 2: Group sparsity: nodes are variables, and black nodes are selected variables

Group sparsity is a special case of graph sparsity discussed below. Figure 2 shows an example of group sparsity, where the variables are represented by nodes, and the selected variables are represented by black nodes. Each pre-defined group is represented as a connected components in the graph, and the example contains six groups. Two groups, the first and the third from the left, are selected in the example. The number of selected variables (black nodes) is seven. Therefore we have $g = 2$ and $|F| = 7$. If we encode each group uniformly, then the coding length is $\mathrm{cl}(F) = 2\log_2(12)$.

4.3 Hierarchical Sparsity

One may also create a hierarchical group structure. A simple example is wavelet coefficients of a signal (Mallat, 1999). Another simple example is a binary tree with the variables as leaves, which we describe below. Each internal node in the tree is associated with three options: only left child, only right child, or both children; each option can be encoded in $\log_2 3$ bits.

Given a subset $F \subset I$, we can go down from the root of the tree, and at each node, decide whether only left child contains elements of F, or only right child contains elements of F, or both children contain elements of F. Therefore the coding length of F is $\log_2 3$ times the total number of internal nodes leading to elements of F. Since each leaf corresponds to no more than $\log_2 p$ internal nodes, the total coding length is no worse than $\log_2 3 \log_2 p|F|$. However, the coding length can be significantly smaller if nodes are close to each other or are clustered. In the extreme case, when the nodes are consecutive, we have $O(|F| + \log_2 p)$ coding length. More generally, if we can order elements in F as $F = \{j_1, \ldots, j_q\}$, then the coding length can be bounded as $\mathrm{cl}(F) = O(|F| + \log_2 p + \sum_{s=2}^{q} \log_2 \min_{\ell<s} |j_s - j_\ell|)$.

If all internal nodes of the tree are also variables in I (for example, in the case of wavelet decomposition), then one may consider feature set F with the following property: if a node is selected, then its parent is also selected. This requirement is very effective in wavelet compression, and often referred to as the zero-tree structure (Shapiro, 1993). Similar requirements have also been applied in statistics (Zhao et al., 2009) for variable selection and in compressive sensing (Baraniuk et al., 2010). The argument presented in this section shows that if we require F to satisfy the zero-tree structure, then its coding length is at most $O(|F|)$, without any explicit dependency on the dimensionality p. This is because one does not have to reach a leave node. Figure 3 shows an example of hierarchical sparsity, where the nodes of the tree are variables, and black nodes indicate those variables that are selected. The total number of selected variables (number of black nodes) is $|F| = 8$. This example obeys the requirement that if a node is selected, then its parent is also selected. Therefore the complexity is measured by $O(|F|)$.

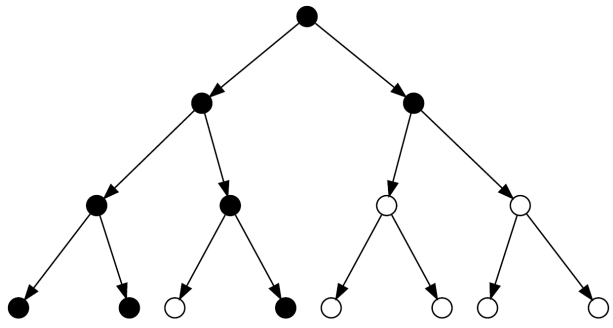


Figure 3: Hierarchical sparsity: nodes are variables, and black nodes are selected variables

The tree-based coding scheme discussed in this section can be polynomially approximated by block coding using no more than $p^{1+\delta}$ base blocks ($\delta > 0$). The idea is similar to that of the image coding example in the more general graph sparsity scheme which we discuss next.

4.4 Graph Sparsity

We consider a generalization of the hierarchical and group sparsity ideas by employing a (directed or undirected) graph structure G on I. To the best of our knowledge, this general structure has not been considered in any previous work.

In graph sparsity, each variable (an element of I) is a node of G but G may also contain additional nodes that are not variables. In order to take advantage of the graph structure, we favor connected regions (that is, nodes that are grouped together with respect to the graph structure). The following result defines a coding length on graphs based on the underlying graph structure. We leave its analysis to Appendix A.

Proposition 10 *Let G be a graph with maximum degree d_G. There exists a constant $C_G \leq 2\log_2(1+d_G)$ such that for any probability distribution q on G ($\sum_{v\in G} q(v) = 1$ and $q(v) \geq 0$ for $v \in G$), the following quantity (which we call graph coding) is a coding length on 2^G:*

$$\mathrm{cl}(F) = C_G|F| + g - \sum_{j=1}^{g} \max_{v\in F_j} \log_2(q(v)),$$

where $F \subset 2^G$ can be decomposed into the union of g connected components $F = \cup_{j=1}^{g} F_j$.

Note that graph coding is sub-additive. As a concrete example, we consider image processing, where each image is a rectangle of pixels (nodes); each pixel is connected to four adjacent pixels, which forms the underlying graph structure. We may take $q(v) = 1/p$ for all $v \in G$, where $p = |G|$ is the number of variables. Proposition 10 implies that if F is composed of g connected regions, then the coding length is $g\log_2(2p) + 2\log_2(5)|F|$, which can be significantly better than standard sparse coding length of $|F|\log_2(2p)$. For example, Figure 4 shows an image grid, where nodes are variables and selected variables are denoted by black nodes. In this example, the selected variables have two connected components (that is, $g = 2$): one in the top-left part, and the other in the bottom-right part of the grid. The total number of selected variables (the number of black nodes) is $|F| = 11$.

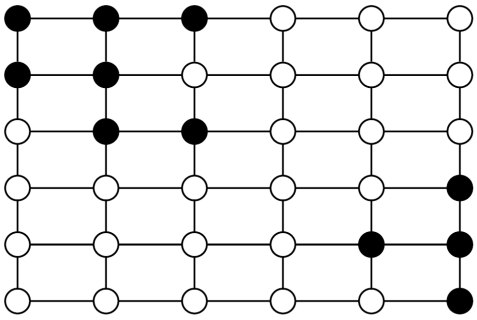


Figure 4: Graph sparsity: nodes are variables, and black nodes are selected variables

Note that group sparsity is a special case of graph sparsity, where each group is one connected region, as shown in Figure 2. We may also link adjacent groups to form the more general line-structured sparsity as in Figure 2. The advantage of line structure over group structure is that we do

not need to know the specific group divisions a priori as in Figure 2. From Proposition 10, similar coding complexity can be obtained as long as F can be covered by a small number of connected regions. Tree-structured hierarchical sparsity is also a special case of graph sparsity with a single connected region containing the root (we may take $q(\text{root}) = 1$). In fact, one may generalize this concept as follows. We consider a special case of sparse sparsity where we limit F to be a connected region that contains a fixed starting node v_0. We can simply let $q(v_0) = 1$, and the coding length of F is $O(|F|)$, which is independent of the dimensionality p. This generalizes the similar claim for the zero-tree structure described earlier.

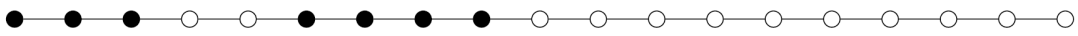


Figure 5: Line-structured sparsity: nodes are variables, and black nodes are selected variables

The following result shows that under uniform encoding of the nodes $q(v) = 1/p$ for $v \in G$, general graph coding schemes can be polynomially approximated with block coding. The idea is to consider relatively small sized base blocks consisted of nodes that are close together with respect to the graph structure, and then use the induced block coding scheme to approximate the graph coding.

Proposition 11 *Let G be a graph with maximum degree d_G, and $p = |G|$. Consider any number $\delta > 0$ such that $L = \delta \log_2 p$ is an even integer. Let $\mathcal{B}$ be the set of connected nodes of size up to L; that is, $B \in \mathcal{B}$ is a connected region in G such that $|B| \leq L$. Then there exists a constant $C_G \leq 2\log_2(1+d_G)$, such that $|\mathcal{B}| \leq p^{1+C_G\delta}$. If we consider the uniform code-length $\text{cl}_0(B) = (1+C_G\delta)\log_2 p$ for all $B \in \mathcal{B}$, then the induced block-coding scheme $\text{cl}_{\mathcal{B}}$ satisfies*

$$\text{cl}_{\mathcal{B}}(F) \leq g(1+C_G\delta)\log_2 p + 2(C_G+\delta^{-1})|F|.$$

where g is the number of connected regions in F.

The result means that graph sparsity can be polynomially approximated with a block coding scheme if we let $q(v) = 1/p$ for all $v \in G$. As we have pointed out, block approximation is useful because the latter is required in the structured greedy algorithm which we propose in this paper.

Note that a refined result holds for hierarchical sparsity (where we have $q(\text{root}) = 1$) using block approximation that does not explicitly depend on $\log_2 p$. In this case, for each tree depth $\ell = 1, 2, 3, \ldots$, we can restrict the underlying tree upto depth ℓ, and apply Proposition 11 on the restricted tree. Using this idea, the coding length for F depends explicitly on the maximum depth of F in the tree instead of $\log_2 p$.

4.5 Random Field Sparsity

Let $z_j \in \{0,1\}$ be a random variable for $j \in I$ that indicates whether j is selected or not. The most general coding scheme is to consider a joint probability distribution of $z = [z_1, \ldots, z_p]$. The coding length for F can be defined as $-\log_2 p(z_1, \ldots, z_p)$ with $z_j = I(j \in F)$ indicating whether $j \in F$ or not.

Such a probability distribution can often be conveniently represented as a binary random field on an underlying graph. In order to encourage sparsity, on average, the marginal probability $p(z_j)$

should take 1 with probability close to $O(1/p)$, so that the expected number of j's with $z_j = 1$ is $O(1)$. For disconnected graphs (z_j are independent), the variables z_j are iid Bernoulli random variables with probability $1/p$ being one. In this case, the coding length of a set F is $|F|\log_2(p) - (p-|F|)\log_2(1-1/p) \approx |F|\log_2(p)+1$. This is essentially the probability model for the standard sparsity scheme. In a more sophisticated situation, one may also let $E(z_j)$ to grow with sample size n. This is useful in non-parametric statistics.

We note that random field model has been considered in Cevher et al. (2009a). For many such models, it is possible to approximate a general random field coding scheme with block coding by using approximation methods in the graphical model literature. However, such approximations are problem specific, and the details are beyond the scope of this paper.

5. Experiments

The purpose of these experiments is to demonstrate the advantage of structured sparsity over standard sparsity. We compare the proposed StructOMP to OMP and Lasso, which are standard algorithms to achieve sparsity but without considering structure (Tibshirani, 1996; Tropp and Gilbert, 2007). For graph sparsity, the choice of $c(F)$ is simply $c(F) = g\log_2 p + |F|$, where g is the number of connected regions of F. This is adequate based on the discussion in Section 3. However, as pointed out after Definition 1, a better method is to use $c(F) = g\log_2 p + \gamma|F|$, where we tune γ appropriately. We observe that in practice, such tuning often improves performance. Nevertheless, in our experiments, we only report results with fixed $\gamma = 1$ for simplicity. This also means our experiments only demonstrate the advantage of StructOMP very conservatively without fine-tuning. The base blocks used in StructOMP are described in each experiment. Parameters (such as s in StructOMP or λ in Lasso) are tuned by cross-validation on the training data. We test various aspects of our theory to check whether the experimental results are consistent with the theory. Although in order to fully test the theory, one should also verify the RIP (or structured RIP) assumptions, in practice this is difficult to check precisely (however, it is possible to verify it approximately using ideas of d'Aspremont et al., 2008). Therefore in the following, we shall only study whether the experimental results are consistent with what can be expected from our theory, without verifying the detailed assumptions. The experimental protocols follow the setup of compressive sensing, where the original signals are projected using random projections, with noise added. Our goal is to recover the original signals from the noise corrupted projections.

In the experiments, we use Lasso-modified least angle regression (LARS/Lasso) as the solver of Lasso (B. Efron and Tibshirani, 2004). In order to quantitatively compare performance of different algorithms, we use recovery error, defined as the relative difference in 2-norm between the estimated sparse coefficient vector $\hat{\bar{\beta}}_{est}$ and the ground-truth sparse coefficient $\bar{\beta}$: $\|\hat{\bar{\beta}}_{est} - \bar{\beta}\|_2 / \|\bar{\beta}\|_2$. Our experiments focus on graph sparsity, with several different underlying graph structures. Note that graph sparsity is more general than group sparsity; in fact connected regions may be regarded as dynamic groups that are not pre-defined. However, for illustration, we include a comparison with group Lasso using some 1D simulated examples, where the underlying structure can be more easily approximated by pre-defined groups. Since additional experiments involving more complicated structures are more difficult to approximate by pre-defined groups, we exclude group-Lasso in those experiments.

All experiments were conducted on a 2.4GHz PC in Matlab. The code for our implementation of StructOMP can be obtained from `http://ranger.uta.edu/~huang/Downloads.htm`. In the

simulation experiments, we use k to denote the sparsity (number of nonzeros) of the true signal, and this should not be confused with the number of iterations k which we used earlier in the description of the StructOMP algorithm.

5.1 Simulated 1D Signals with Line-Structured Sparsity

In the first experiment, we randomly generate a $1D$ structured sparse signal with values ± 1, where data dimension $p = 512$, sparsity number $k = 64$ and group number $g = 4$. The support set of these signals is composed of g connected regions. Here, each component of the sparse coefficient is connected to two of its adjacent components, which forms the underlying graph structure. The graph sparsity concept introduced earlier is used to compute the coding length of sparsity patterns in StructOMP. The projection matrix X is generated by creating an $n \times p$ matrix with i.i.d. draws from a standard Gaussian distribution $N(0,1)$. For simplicity, the rows of X are normalized to unit magnitude. Zero-mean Gaussian noise with standard deviation $\sigma = 0.01$ is added to the measurements. Our task is to compare the recovery performance of StructOMP to those of OMP, Lasso and group Lasso for these structured sparsity signals under the framework of compressive sensing.

Figure 6 shows one instance of generated signal and the corresponding recovered results by different algorithms when $n = 160$. Since the sample size n is not big enough, OMP and Lasso do not achieve good recovery results, whereas the StructOMP algorithm achieves near perfect recovery of the original signal. We also include group Lasso in this experiment for illustration. We use predefined consecutive groups that do not completely overlap with the support of the signal. Since we do not know the correct group size, we just try group Lasso with several different group sizes (gs=2, 4, 8, 16). Although the results obtained with group Lasso are better than those of OMP and Lasso, they are still inferior to the results with StructOMP. As mentioned, this is because the pre-defined groups do not completely overlap with the support of the signal, which reduces the efficiency. In StructOMP, the base blocks are simply small connected line segments of size gs=3: that is, one node plus its two neighbors. This choice is only for simplicity, and it already produces good results in our experiments. If we include larger line segments into the base blocks (e.g., segments of size gs=4,5, etc), one can expect even better performance from StructOMP.

To study how the sample size n effects the recovery performance, we vary the sample size and record the recovery results by different algorithms. To reduce the randomness, we perform the experiment 100 times for each sample size. Figure 7(a) shows the recovery performance in terms of Recovery Error and Sample Size, averaged over 100 random runs for each sample size. As expected, StructOMP is better than the group Lasso and far better than the OMP and Lasso. The results show that the proposed StructOMP can achieve better recovery performance for structured sparsity signals with less samples. Figure 7(b) shows the recovery performance in terms of CPU Time and Sample Size, averaged over 100 random runs for each sample size. The computation complexities of StructOMP and OMP are far lower than those of Lasso and Group Lasso.

It is worth noting that the performance of StructOMP is less stable than the other algorithms when the sample size n is small. This is because for randomly generated design matrix, the structured RIP condition is only satisfied probabilistically. For small n, the necessary structured RIP condition can be violated with relatively large probability, and in such case StructOMP does not have much advantage (at least theoretically). This implies the relatively large variance. The effect is much less noticeable with weakly sparse signal in Figure 11(a) because the necessary structured

RIP condition is easier to satisfied for weakly sparse signals (based on our theory). Therefore the experimental results are consistent with our theory.

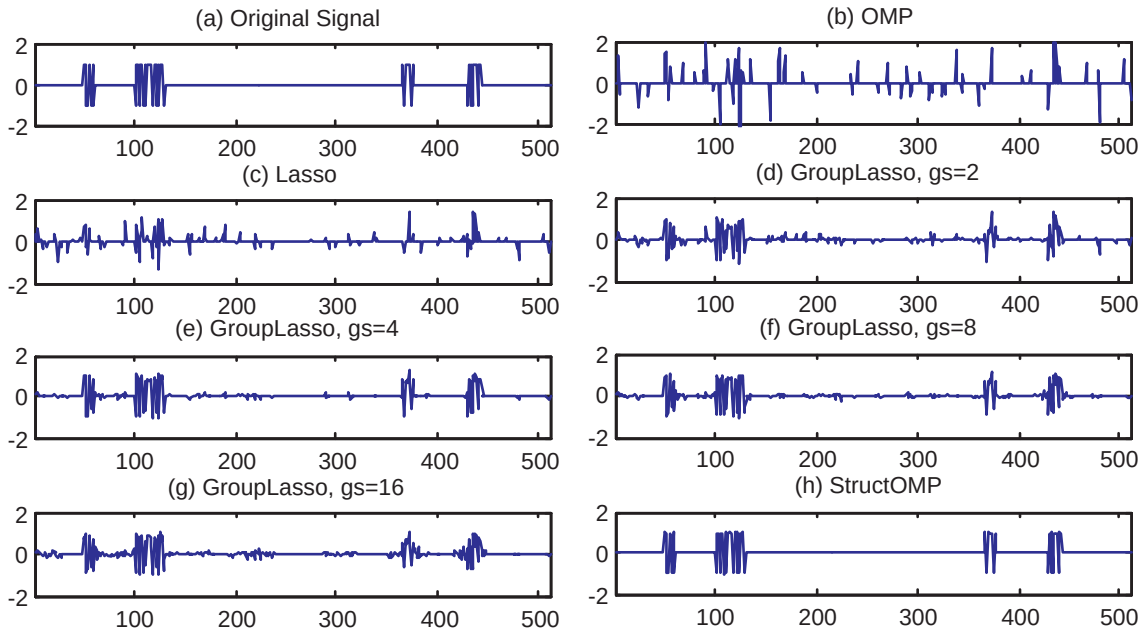
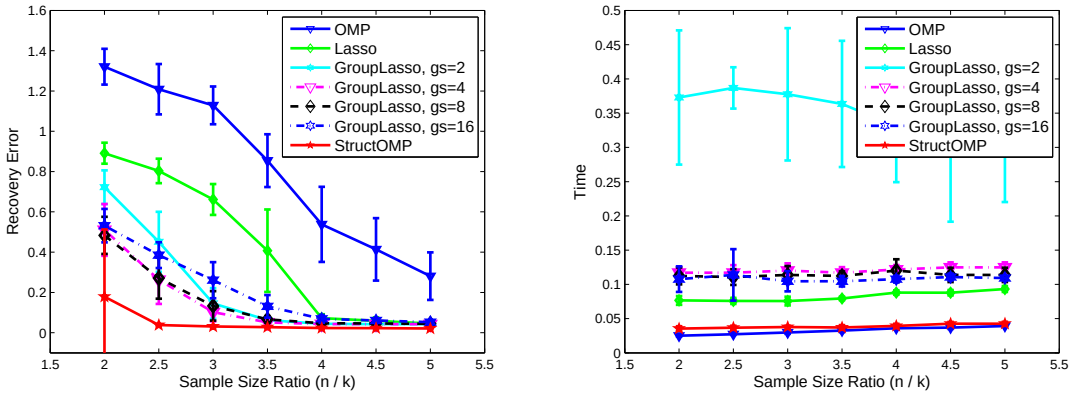


Figure 6: Recovery results of 1D signal with strongly line-structured sparsity. (a) original data; (b) recovered results with OMP (error is 0.9921); (c) recovered results with Lasso (error is 0.8660);; (d) recovered results with Group Lasso (error is 0.4832 with group size gs=2); (e) recovered results with Group Lasso (error is 0.4832 with group size gs=4);(f) recovered results with Group Lasso (error is 0.2646 with group size gs=8);(g) recovered results with Group Lasso (error is 0.3980 with group size gs=16); (h) recovered results with StructOMP (error is 0.0246).

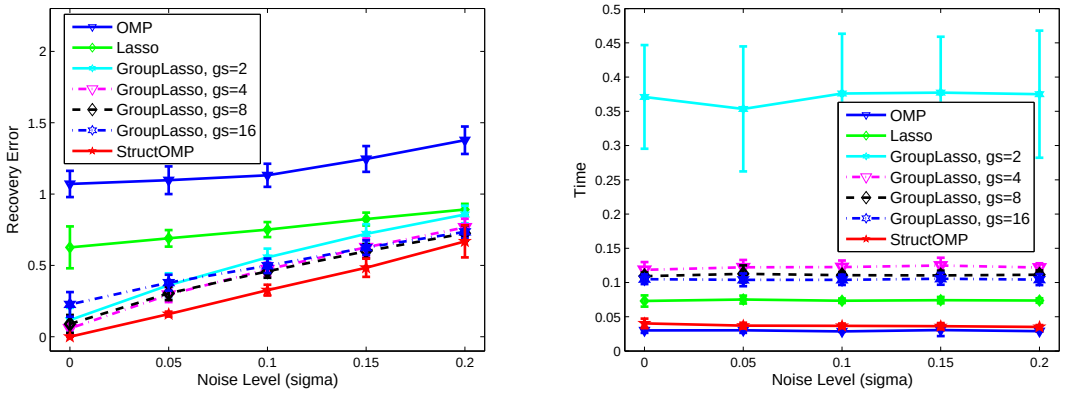
To study how the additive noise affects the recovery performance, we adjust the noise power σ and then record the recovery results by different algorithms. In this case, we fix the sample size at $n = 3k = 192$, and perform the experiment 100 times for each noise level tested. Figure 8(a) shows the recovery performance in terms of Recovery Error and Noise Level, averaged over 100 random runs for each noise level. As expected, StructOMP is also better than the group Lasso and far better than the OMP and Lasso. Figure 8(b) shows the recovery performance in terms of CPU Time and Noise Level, averaged over 100 random runs for each sample size. The computational complexities of StructOMP and OMP are lower than those of Lasso and Group Lasso.

To further study the performance of the StructOMP, we also compare it to two other methods for structured sparsity including OverlapLasso (Jacob et al., 2009) and ModelCS (Baraniuk et al., 2010) using the implementations available from the web. For fair comparisons, the same structures are used in OverlapLasso, ModelCS and StructOMP. As mentioned before, in these experiments, we use small connected line segments of size gs=3 (including one node plus its two neighbors) as base blocks in StructOMP. Therefore in OverlapLasso, the groups are also connected line segments of size gs=3; in ModelCS, this structure leads to the model assumption that if one node is nonzero, then its two neighbors has a high probability of being nonzeros. Figure 9(a) shows the recovery



(a) (b)

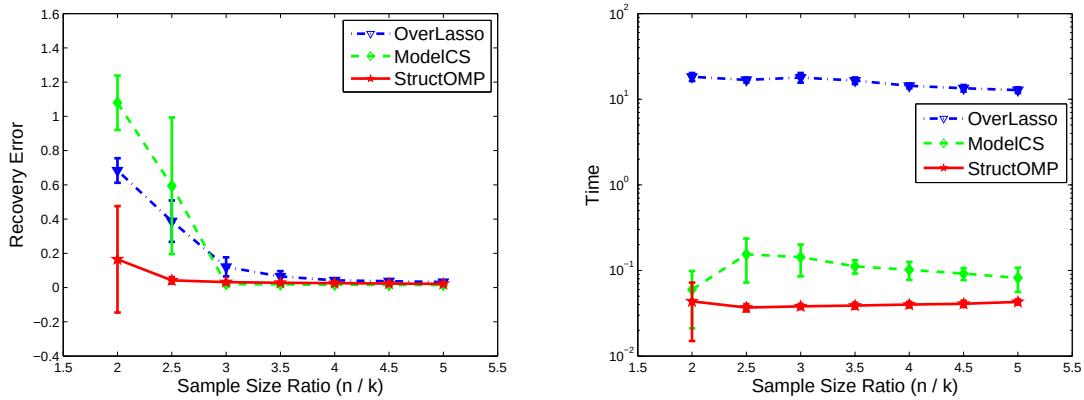
Figure 7: Recovery performance: (a) Recovery Error vs. Sample Size Ratio (n/k); (b) CPU Time vs. Sample Size Ratio (n/k)



(a) (b)

Figure 8: Recovery performance in terms of Noise Levels: (a) Recovery Error vs. Noise Level; (b) CPU Time vs. Noise Level

performance in terms of Recovery Error and Sample Size, averaged over 100 random runs for each sample size. At least for this problem, StructOMP achieves better performance than OverlapLasso and ModelCS, which shows that the proposed StructOMP algorithm can achieve better recovery performance than other structured sparsity algorithms for some problems. Figure 9(b) shows the recovery performance in terms of CPU Time and Sample Size, averaged over 100 random runs for each sample size. Although it is difficult to see from the figure, the computational complexity of StructOMP is lower than that of ModelCS (about half CPU time) and are far lower than that of OverlapLasso, at least based on the implementation of Jacob et al. (2009).



(a) (b)

Figure 9: Performance Comparisons between methods related with structured sparsity(OverlapLasso (Jacob et al., 2009), ModelCS (Baraniuk et al., 2010), StructOMP): (a) Recovery Error vs. Sample Size Ratio (n/k); (b) CPU Time vs. Sample Size Ratio (n/k)

Note that Lasso performs better than OMP in the first example. This is because the signal is strongly sparse (that is, all nonzero coefficients are significantly different from zero). In the second experiment, we randomly generate a $1D$ structured sparse signal with weak sparsity, where the nonzero coefficients decay gradually to zero, but there is no clear cutoff. One instance of generated signal is shown in Figure 10 (a). Here, $p = 512$ and all coefficient of the signal are not zeros. We define the sparsity k as the number of coefficients that contain 95% of the image energy. The support set of these signals is composed of $g = 2$ connected regions. Again, each element of the sparse coefficient is connected to two of its adjacent elements, which forms the underlying 1D line graph structure. The graph sparsity concept introduced earlier is used to compute the coding length of sparsity patterns in StructOMP. The projection matrix X is generated by creating an $n \times p$ matrix with i.i.d. draws from a standard Gaussian distribution $N(0,1)$. For simplicity, the rows of X are normalized to unit magnitude. Zero-mean Gaussian noise with standard deviation $\sigma = 0.01$ is added to the measurements.

Figure 10 shows one generated signal and its recovered results by different algorithms when $k = 32$ and $n = 48$. Again, we observe that OMP and Lasso do not achieve good recovery results, whereas the StructOMP algorithm achieves near perfect recovery of the original signal. As we do not know the predefined groups for group Lasso, we just try group Lasso with several different

group sizes (gs=2, 4, 8, 16). Although the results obtained with group Lasso are better than those of OMP and Lasso, they are still inferior to the results with StructOMP. In order to study how the sample size n effects the recovery performance, we vary the sample size and record the recovery results by different algorithms. To reduce the randomness, we perform the experiment 100 times for each of the sample sizes.

Figure 11(a) shows the recovery performance in terms of Recovery Error and Sample Size, averaged over 100 random runs for each sample size. As expected, StructOMP algorithm is superior in all cases. What's different from the first experiment is that the recovery error of OMP becomes smaller than that of Lasso. This result is consistent with our theory, which predicts that if the underlying signal is weakly sparse, then the relatively performance of OMP becomes comparable to Lasso. Figure 11(b) shows the recovery performance in terms of CPU Time and Sample Size, averaged over 100 random runs for each sample size. The computational complexities of StructOMP and OMP are far lower than those of Lasso and Group Lasso.

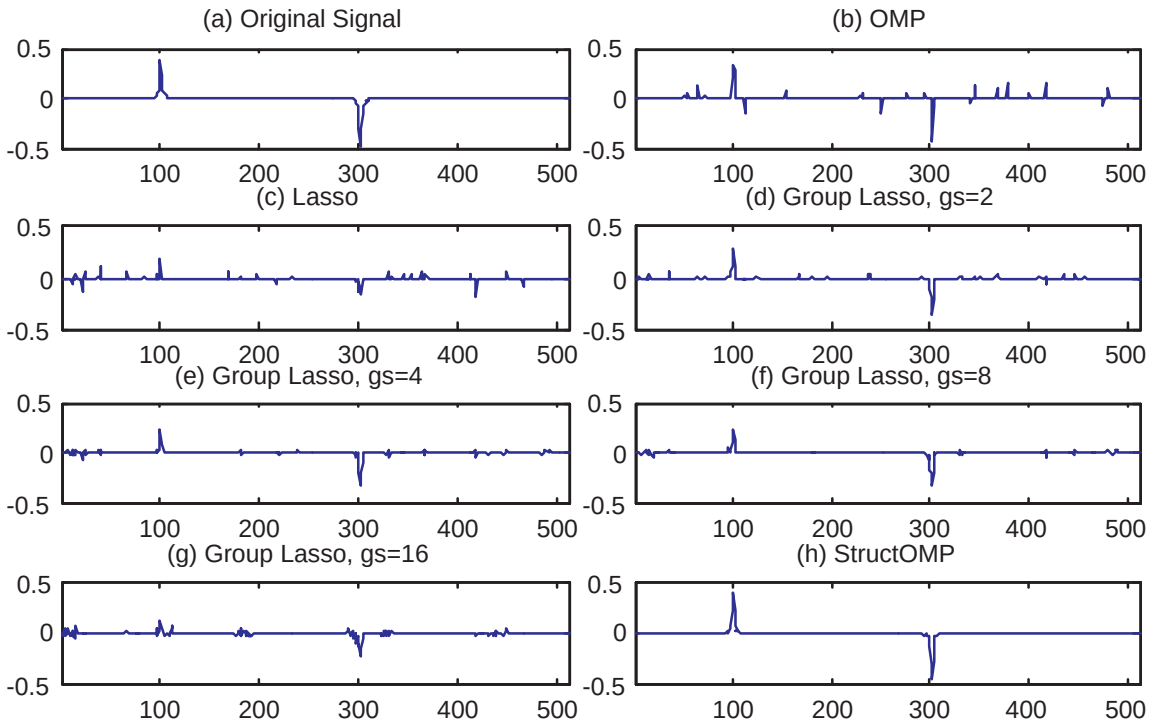
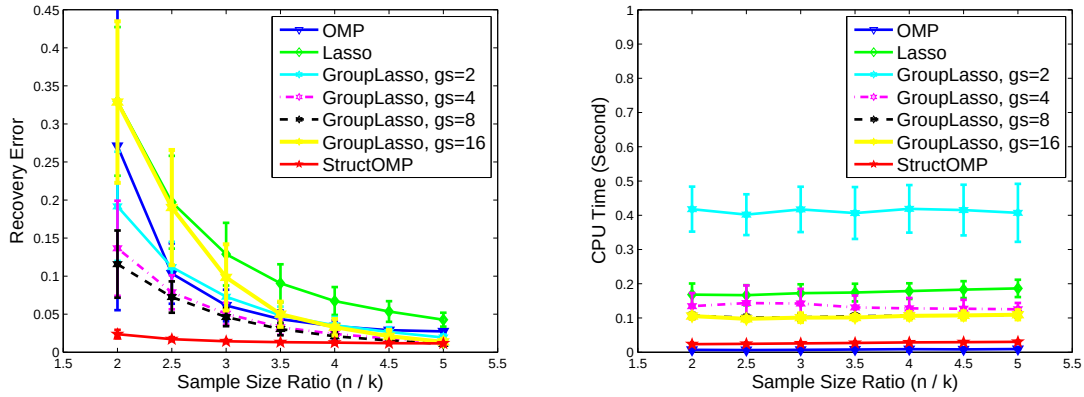


Figure 10: Recovery results of 1D weakly sparse signal with line-structured sparsity. (a) original data; (b) recovered results with OMP (error is 0.5599); (c) recovered results with Lasso (error is 0.6686); (d) recovered results with Group Lasso (error is 0.4732 with group size gs=2); (e) recovered results with Group Lasso (error is 0.2893 with group size gs=4);(f) recovered results with Group Lasso (error is 0.2646 with group size gs=8);(g) recovered results with Group Lasso (error is 0.5459 with group size gs=16); (h) recovered results with StructOMP (error is 0.0846).



(a) (b)

Figure 11: Recovery performance for 1D Weak Line-Sparsity: (a) Recovery Error vs. Sample Size Ratio (n/k); (b) CPU Time vs. Sample Size Ratio (n/k)

5.2 2D Image Compressive Sensing with Tree-structured Sparsity

It is well known that 2D natural images are sparse in a wavelet basis. Their wavelet coefficients have a hierarchical tree structure, which is widely used for wavelet-based compression algorithms (Shapiro, 1993). Figure 12(a) shows a widely used example image with size 64×64: *cameraman*. Note that we use a reduced image instead of the original for computational efficiency since the experiments is run many times with different random matrices. This reduction should not affect the relative performance among various algorithms.

In this experiment, each 2D wavelet coefficient of this image is connected to its parent coefficient and child coefficients, which forms the underlying hierarchical tree structure (which is a special case of graph sparsity). In our experiment, we choose Haar-wavelet to obtain its tree-structured sparsity wavelet coefficients. The projection matrix X and noises are generated with the same method as that for 1D structured sparsity signals. OMP, Lasso and StructOMP are used to recover the wavelet coefficients from the random projection samples respectively. Then, the inverse wavelet transform is used to reconstruct the images with these recovered wavelet coefficients. Our task is to compare the recovery performance of the StructOMP to those of OMP and Lasso under the framework of compressive sensing.

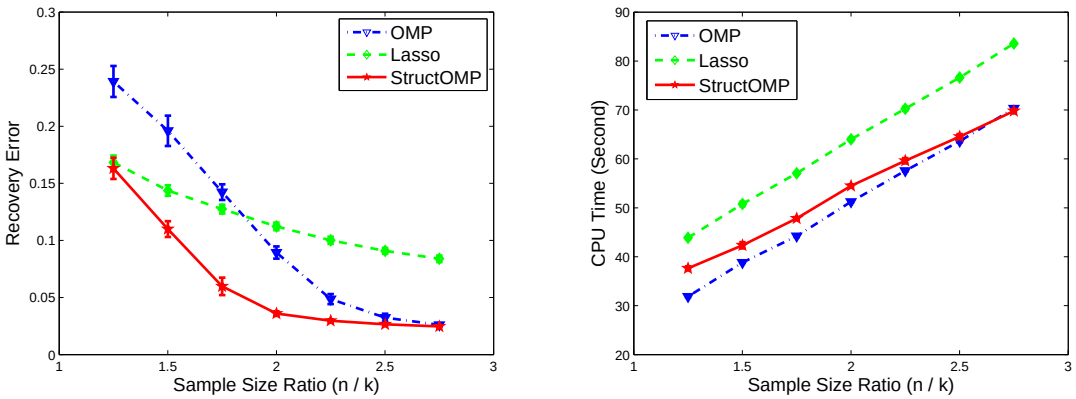
For Lasso, we use identical regularization parameter for all coefficients (without varying regularization parameters based on bands or tree depth). For StructOMP, a simple block-structure is used, where each block corresponds to a node in the tree, plus its ancestors leading to the root. This corresponds to setting $\delta = 0$ in Proposition 11. We use this block set for efficiency only because the number of blocks is only linear in p.

Figure 12 shows one example of the recovered results by different algorithms with sparsity number $k = 1133$ and sample size $n = 2048$. It shows that StructOMP obtains the best recovered result. Figure 13(a) shows the recovery performance in terms of Sample Size and Recovery Error, averaged over 100 random runs for each sample size. The StructOMP algorithm is better than both Lasso and OMP in this case. Since real image data are weakly sparse, the performance of standard

OMP (without structured sparsity) is similar to that of Lasso. Figure 13(b) shows the recovery performance in terms of Sample Size and CPU Time, averaged over 100 random runs for each sample size. The computational complexity of StructOMP is comparable to that of OMP and lower than that of Lasso.

(a) (b) (c) (d)

Figure 12: Recovery results with sample size $n = 2048$: (a) cameraman image, (b) recovered image with OMP (error is 0.1886; CPU time is 46.16s), (c) recovered image with Lasso (error is 0.1670; CPU time is 60.26s) and (d) recovered image with StructOMP (error is 0.0375; CPU time is 48.99s)



(a) (b)

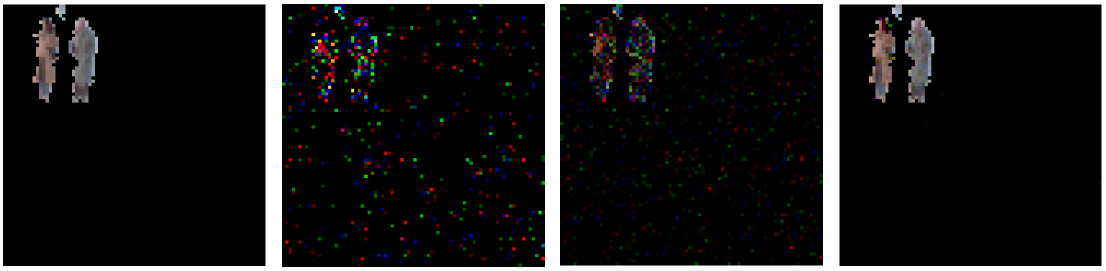
Figure 13: Recovery performance for 2D wavelet tree sparsity: (a) Recovery Error vs. Sample Size; (b) CPU Time vs. Sample size

5.3 Background Subtracted Images for Robust Surveillance

Background subtracted images are typical structure sparsity data in static video surveillance applications. They generally correspond to the foreground objects of interest. Unlike the whole scene, these images are not only spatially sparse but also inclined to cluster into groups, which correspond to different foreground objects. Thus, the StructOMP algorithm can obtain superior recovery

from compressive sensing measurements that are received by a centralized server from multiple and randomly placed optical sensors. In this experiment, the testing video is downloaded from `http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/`. The background subtracted images are obtained with the software (Zivkovic and Heijden, 2006). One sample image frame is shown in Figure 14(a). The support set of 2D images is thus composed of several connected regions. Here, each pixel of the 2D background subtracted image is connected to four of its adjacent pixels, forming the underlying graph structure in graph sparsity. We randomly choose 100 background subtracted images as test images.

Note that color images have three channels. We can consider three channels separately and perform sparse recovery independently for each channel. On the other hand, since in this application, three channels of the color background subtracted image share the same support set, we can enforce group sparsity across the color channels for each pixel. That is, a pixel in the color image can be considered as a triplet with three color intensities. We will thus consider both cases in our comparisons. In the latter case, we simply replace OMP and Lasso by Group OMP (which has also been studied by Lozano et al., 2009) and Group Lasso respectively.

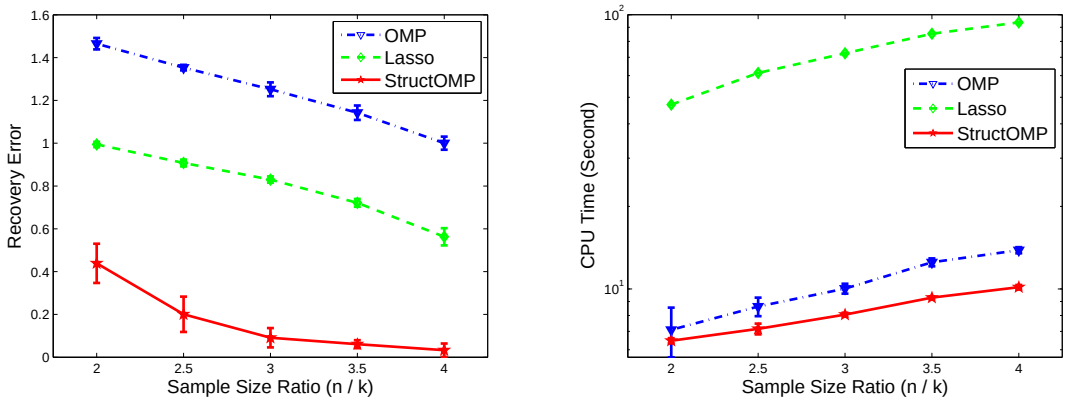


(a) (b) (c) (d)

Figure 14: Recovery results with sample size $n = 900$: (a) the background subtracted image, (b) recovered image with OMP (error is 1.1833), (c) recovered image with Lasso (error is 0.7075) and (d) recovered image with StructOMP (error is 0.1203)

In this experiment, we firstly consider the 3 color channel independently, and use OMP, Lasso and StructOMP to separately recover each channel. The results shown in Figure 14 demonstrates that the StructOMP outperforms both OMP and Lasso in recovery. Figure 15(a) shows the recovery performance as a function of increasing sample size ratios. It demonstrates again that StructOMP significantly outperforms OMP and Lasso in recovery performance on video data. Comparing to the image compression example in the previous section, the background subtracted images have a more clearly defined sparsity pattern where nonzero coefficients are generally distinct from zero (that is, stronger sparsity); this explains why Lasso performs better than the OMP on this particular data. The results is again consistent with our theory. Figure 17(b) shows the recovery performance in terms of Sample Size and CPU Time, averaged over 100 random runs for each sample size. The computational complexity of StructOMP is again comparable to that of OMP and lower than that of Lasso.

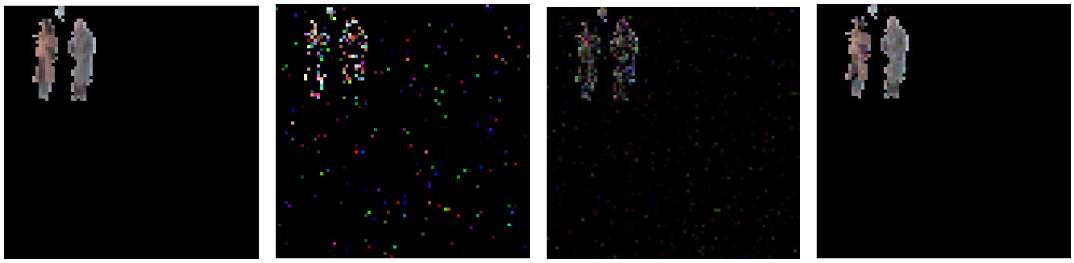
If we consider a pixel as a triplet in the background subtracted image, we replace OMP and Lasso by Group OMP and Group Lasso (across the color channels), and compare their performance to StructOMP. The results in Figure 16 indicate that StructOMP is still superior, although



(a) (b)

Figure 15: Recovery performance: (a) Recovery Error vs. Sample Size; (b) CPU Time vs. Sample size

as expected, the recovery performance of Group OMP (or Group Lasso) improves that of OMP (or Lasso). Figure 17(a) shows the recovery performance as a function of increasing sample size ratios. It demonstrates again that StructOMP outperforms Group OMP and Group Lasso in this application. Figure 17(b) shows the recovery performance in terms of Sample Size and CPU Time, averaged over 100 random runs for each sample size. The computational complexity of StructOMP is again comparable to that of Group OMP and lower than that of Group Lasso.

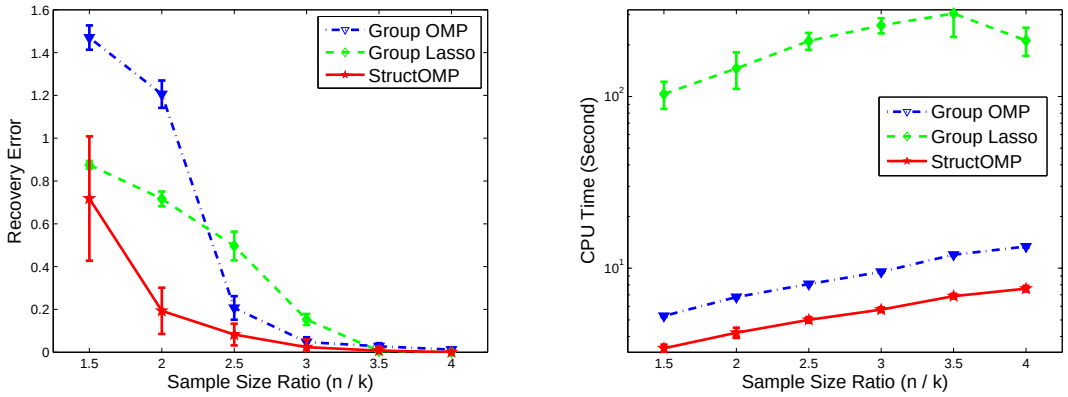


(a) (b) (c) (d)

Figure 16: Recovery results with sample size $n = 600$: (a) the background subtracted image, (b) recovered image with Group OMP (error is 1.1340), (c) recovered image with Group Lasso (error is 0.6972) and (d) recovered image with StructOMP (error is 0.0808)

6. Discussion

This paper develops a theory for structured sparsity where prior knowledge allows us to prefer certain sparsity patterns to others. Some examples are presented to illustrate the concept. The



(a) (b)

Figure 17: Recovery performance: (a) Recovery Error vs. Sample Size; (b) CPU Time vs. Sample size

general framework established in this paper includes the recently popularized group sparsity idea as a special case.

In structured sparsity, the complexity of learning is measured by the coding complexity $c(\bar{\beta}) \leq \|\bar{\beta}\|_0 + \mathrm{cl}(\mathrm{supp}(\bar{\beta}))$ instead of $\|\bar{\beta}\|_0 \ln p$ which determines the complexity in standard sparsity. Using this notation, a theory parallel to that of the standard sparsity is developed. The theory shows that if the coding length $\mathrm{cl}(\mathrm{supp}(\bar{\beta}))$ is small for a target coefficient vector $\bar{\beta}$, then the complexity of learning $\bar{\beta}$ can be significantly smaller than the corresponding complexity in standard sparsity. Experimental results demonstrate that significant improvements can be obtained on some real problems that have natural structures.

The structured greedy algorithm presented in this paper is the first efficient algorithm proposed to handle the general structured sparsity learning. It is shown that the algorithm is effective under appropriate conditions. Future work include additional computationally efficient methods such as convex relaxation methods (e.g. L_1 regularization for standard sparsity, and group Lasso for strong group sparsity) and backward greedy strategies to improve the forward greedy method considered in this paper.

Appendix A. Proof of Proposition 10 and Proposition 11

Proof of Proposition 10.

First we show that we can encode all connected regions F (that is, with $g = 1$) using no more than

$$C_G|F| - \max_{v \in F} \log_2 q(v) \tag{8}$$

bits. We consider the following procedure to encode F: first, we pick a node v_* from F achieving $-\max_{v \in F} \log_2 q(v)$, which requires $-\max_{v \in F} \log_2 q(v)$ bits. We then push v_* into a stack S. We encode the remaining nodes in F using the following algorithm: until the stack S is empty, we take the top element v out of the stack S, and do the following

(a) Encode the number of neighbors of v in F that has not been visited so far, with no more than $\log_2(1+d_G)$ bits.

(b) For each neighbor v' of v in F that has not been visited, we encode it (i.e., the associated edge between v and v') with no more than $\log_2 d_G$ bits. We then push v' into the stack S.

Since F is connected, after this procedure finishes (the stack becomes empty), we have visited all nodes in F. Since step (a) can be invoked only $|F|$ times, the total number of bits in step (a) is no more than $|F|\log_2(1+d_G)$. The number of bits in step (b) is no more than the number of nodes in F (except for node v_*) times the bits to encode each node, which is no more than $(|F|-1)\log_2(1+d_G)$. Therefore the total number of bits in step (a) and (b) is less than $C_G|F|$. This proves (8).

For $g > 1$, we may encode each connected component F_j of F sequentially, using number of bits according to (8). Then after encoding each connected region F_j, we use 1 bit to encode whether $j = g$ or not (that is, whether we should stop or encode an additional connected component). This gives the formula in Proposition 10.

Proof of Proposition 11.

We first prove the following two lemmas.

Lemma 12 *Given a positive even integer L. Let F be a connected region of G such that $|F| \geq L+1$. Then it is possible to partition F as the union of two connected regions F_1 and F_2 such that: $F = F_1 \cup F_2$, $|F_1 \cap F_2| = 1$, and*

- *either* $\min(|F_1|, |F_2|) \geq 0.5L+1$*;*
- *or* $0.5L+1 \leq |F_1| \leq L$*.*

Proof We consider the following algorithm. Start with a node v of F and set $F_1 = \{v\}$ and let $u_1 = v$. Repeat the following procedure

(a) If $|F_1| \geq 0.5L+1$, then exit the procedure with the current F_1 and $F_2 = (F - F_1) \cup \{u_1\}$.

(b) If $F - F_1$ is connected: let v be a node in $F - F_1$ that is connected to F_1. We add v to F_1, and set $u_1 = v$. We then repeat the procedure (a)(b)(c).

(c) If $F - F_1$ is not connected: $(F - F_1) \cup \{u_1\}$ is connected by construction. Merge the smallest connected component of $F - F_1$ into F_1. Repeat the procedure (a)(b)(c).

Clearly the procedure eventually will end at step (a) because each iteration $|F_1|$ is increased by at least 1. When it ends, $F_1 \cap F_2 = \{u_1\}$. Moreover, there were two possible scenarios in the previous iteration:

(1) Step (b) was invoked. That is, $|F_1|$ was increased by 1 in the previous iteration, and hence $|F_1| = 0.5L+1 \leq L$. Moreover, F_2 is connected.

(2) Step (c) was invoked. Still, F_2 is connected by the construction of u_1. If $|F_1|$ was increased by no more than $L/2$ in the previous step (c), then $|F_1| \leq L$ and the lemma holds. Otherwise, $F - F_1$ has more than $L/2$ nodes because this scenario implies that even the smallest connected component has more than $L/2$ nodes in the previous step (c). Therefore in this case we have $|F_2| > 0.5L+1$.

■

Lemma 13 *Given a positive even integer L. Any connected region F such that $|F| \geq 0.5L+1$ can be covered by at most $2(|F|-1)/L$ connected regions, each of size no more than L.*

Proof We fix L and prove the claim by induction on $|F|$. If $0.5L+1 \leq |F| \leq L$, then F is covered by itself, and the claim is trivial. If $L < |F| \leq 1.5L$, then by Lemma 12, we can partition F into two connected regions, each $\leq L$. Therefore the claim also holds trivially.

Now assume that the claim holds for $|F| \leq k$ with $k \geq 1.5L$. For F such that $|F| = k+1$, we apply Lemma 12 and partition it into two regions $F = F_1 \cup F_2$ such that $\min(|F_1|, |F_2|) \geq 0.5L+1$ and $|F_1| + |F_2| = |F| + 1$. Therefore by the induction hypothesis, we can cover each F_j $(j = 1,2)$ by $2(|F_j|-1)/L$ connected regions, each of size no more than L. It follows that the total number of connected regions to cover both F_1 and F_2 is no more than $2(|F_1|+|F_2|-2)/L = 2(|F|-1)/L$, which completes the induction. ■

We are now ready to prove Proposition 11. First, from (8), we know that $C_G|B| + \log_2 p$ is a coding-length for connected regions $B \in \mathcal{B}$. Therefore

$$2^{-(C_G L + \log_2 p)}|\mathcal{B}| \leq \sum_{B \in \mathcal{B}} 2^{-(C_G|B| + \log_2 p)} \leq 1.$$

This implies that $|\mathcal{B}| \leq p^{1+C_G\delta}$.

Since Lemma 13 implies that each connected component F_j of F can be covered by $1+2(|F_j|-1)/L$ connected regions from $\mathcal{B}$, we have $\mathrm{cl}_{\mathcal{B}}(F_j) \leq (1+2(|F_j|-1)/L)(1+C_G\delta)\log_2 p$ under the uniform coding on $\mathcal{B}$. By summing over the connected components, we obtain the desired bound.

Appendix B. Proof of Proposition 3

Lemma 14 *Consider a fixed vector $\mathbf{x} \in \mathbb{R}^n$, and a random vector $\mathbf{y} \in \mathbb{R}^n$ with independent sub-Gaussian components: $\mathbb{E}e^{t(\mathbf{y}_i - \mathbb{E}\mathbf{y}_i)} \leq e^{\sigma^2 t^2/2}$ for all t and i, then $\forall \varepsilon > 0$:*

$$\Pr\left(\left|\mathbf{x}^\top \mathbf{y} - \mathbb{E}\mathbf{x}^\top \mathbf{y}\right| \geq \varepsilon\right) \leq 2e^{-\varepsilon^2/(2\sigma^2\|\mathbf{x}\|_2^2)}.$$

Proof Let $s_n = \sum_{i=1}^n (\mathbf{x}_i\mathbf{y}_i - \mathbb{E}\mathbf{x}_i\mathbf{y}_i)$; then by assumption, $\mathbb{E}(e^{ts_n} + e^{-ts_n}) \leq 2e^{\sum_i \mathbf{x}_i^2\sigma^2 t^2/2}$, which implies that $\Pr(|s_n| \geq \varepsilon)e^{t\varepsilon} \leq 2e^{\sum_i \mathbf{x}_i^2\sigma^2 t^2/2}$. Now let $t = \varepsilon/(\sum_i \mathbf{x}_i^2\sigma^2)$, we obtain the desired bound. ■

The following lemma is taken from Pisier (1989).

Lemma 15 *Consider the unit sphere $S^{k-1} = \{x : \|x\|_2 = 1\}$ in $\mathbb{R}^k$ $(k \geq 1)$. Given any $\varepsilon > 0$, there exists an ε-cover $Q \subset S^{k-1}$ such that $\min_{q \in Q}\|x-q\|_2 \leq \varepsilon$ for all $\|x\|_2 = 1$, with $|Q| \leq (1+2/\varepsilon)^k$.*

B.1 Proof of Proposition 3

According to Lemma 15, given $\varepsilon_1 > 0$, there exists a finite set $Q = \{q_i\}$ with $|Q| \leq (1+2/\varepsilon_1)^k$ such that $\|Pq_i\|_2 = 1$ for all i, and $\min_i \|P\mathbf{z} - Pq_i\|_2 \leq \varepsilon_1$ for all $\|P\mathbf{z}\|_2 = 1$. To see the existence of Q,

we consider a rotation of the coordinate system (which does not change 2-norm) so that $P\mathbf{z}$ is the projection of $\mathbf{z} \in \mathbb{R}^n$ to its first k coordinates in the new coordinate system. Lemma 15 can now be directly applied to the first k coordinates in the new system, implying that we can pick q_i such that $Pq_i = q_i$.

For each i, Lemma 14 implies that $\forall \varepsilon_2 > 0$:

$$\Pr\left(\left|q_i^\top P(\mathbf{y} - \mathbb{E}\mathbf{y})\right| \geq \varepsilon_2\right) \leq 2e^{-\varepsilon_2^2/(2\sigma^2)}.$$

Taking union bound for all $q_i \in Q$, we obtain with probability exceeding $1 - 2(1+2/\varepsilon_1)^k e^{-\varepsilon_2^2/2\sigma^2}$:

$$\left|q_i^\top P(\mathbf{y} - \mathbb{E}\mathbf{y})\right| \leq \varepsilon_2$$

for all i.

Let $\mathbf{z} = P(\mathbf{y} - \mathbb{E}\mathbf{y})/\|P(\mathbf{y} - \mathbb{E}\mathbf{y})\|_2$, then there exists i such that $\|P\mathbf{z} - Pq_i\|_2 \leq \varepsilon_1$. We have

$$\begin{aligned}\|P(\mathbf{y} - \mathbb{E}\mathbf{y})\|_2 =& \mathbf{z}^\top P(\mathbf{y} - \mathbb{E}\mathbf{y}) \\ \leq& \|P\mathbf{z} - Pq_i\|_2 \|P(\mathbf{y} - \mathbb{E}\mathbf{y})\|_2 + |q_i^\top P(\mathbf{y} - \mathbb{E}\mathbf{y})| \\ \leq& \varepsilon_1 \|P(\mathbf{y} - \mathbb{E}\mathbf{y})\|_2 + \varepsilon_2.\end{aligned}$$

Therefore

$$\|P(\mathbf{y} - \mathbb{E}\mathbf{y})\|_2 \leq \varepsilon_2/(1-\varepsilon_1).$$

Let $\varepsilon_1 = 2/15$, and $\eta = 2(1+2/\varepsilon_1)^k e^{-\varepsilon_2^2/2\sigma^2}$, we have

$$\varepsilon_2^2 = 2\sigma^2[(4k+1)\ln 2 - \ln \eta],$$

and thus

$$\|P(\mathbf{y} - \mathbb{E}\mathbf{y})\|_2 \leq \frac{15}{13}\sigma\sqrt{2(4k+1)\ln 2 - 2\ln\eta}.$$

This simplifies to the desired bound.

Appendix C. Proof of Proposition 5

We use the following lemma from Huang and Zhang (2010).

Lemma 16 *Suppose X is generated according to Proposition 5. For any fixed set $F \subset I$ with $|F| = k$ and $0 < \delta < 1$, we have with probability exceeding $1 - 3(1+8/\delta)^k e^{-n\delta^2/8}$:*

$$(1-\delta)\|\beta\|_2 \leq \frac{1}{\sqrt{n}}\|X_F\beta\|_2 \leq (1+\delta)\|\beta\|_2$$

for all $\beta \in \mathbb{R}^k$.

C.1 Proof of Proposition 5

Since $\mathrm{cl}(F)$ is a coding length, we have (for any fixed $\gamma < 1$)

$$\sum_{F:|F|+\mathrm{cl}(F)\leq s}(1+8/\delta)^{|F|} \leq \sum_{F:|F|+\gamma\mathrm{cl}(F)\leq s}(1+8/\delta)^{|F|}$$
$$\leq \sum_{F}(1+8/\delta)^{s-\gamma\mathrm{cl}(F)} = (1+8/\delta)^{s}\sum_{F}2^{-\mathrm{cl}(F)} \leq (1+8/\delta)^{s},$$

where in the above derivation, we take $\gamma = 1/\log_2(1+8/\delta)$.

For each F, we know from Lemma 16 that for all β such that $\mathrm{supp}(\beta)\subset F$:

$$(1-\delta)\|\beta\|_2 \leq \frac{1}{\sqrt{n}}\|X\beta\|_2 \leq (1+\delta)\|\beta\|_2$$

with probability exceeding $1-3(1+8/\delta)^{|F|}e^{-n\delta^2/8}$.

We can thus take the union bound over $F : |F|+\mathrm{cl}(F)\leq s$, which shows that with probability exceeding

$$1-\sum_{F:|F|+\mathrm{cl}(F)\leq s}3(1+8/\delta)^{|F|}e^{-n\delta^2/8},$$

the structured RIP in Equation (4) holds. Since

$$\sum_{F:|F|+\mathrm{cl}(F)\leq s}3(1+8/\delta)^{|F|}e^{-n\delta^2/8} \leq 3(1+8/\delta)^{s}e^{-n\delta^2/8} \leq e^{-t},$$

we obtain the desired bound.

Appendix D. Proof of Theorem 6 and Theorem 7

Lemma 17 *Suppose that Assumption 1 is valid. For any fixed subset $F\subset I$, with probability $1-\eta$, $\forall\beta$ such that $\mathrm{supp}(\beta)\subset F$, and $a>0$, we have*

$$\|X\beta-\mathbb{E}\mathbf{y}\|_2^2 \leq (1+a)[\|X\beta-\mathbf{y}\|_2^2-\|\mathbf{y}-\mathbb{E}\mathbf{y}\|_2^2]+(2+a+a^{-1})\sigma^2[7.4|F|+4.7\ln(4/\eta)].$$

Proof Let

$$P_F = X_F(X_F^\top X_F)^+X_F^\top$$

be the projection matrix to the subspace generated by columns of X_F. Here X_F may not be full-rank, and $(X_F^\top X_F)^+$ denotes the Moore-Penrose pseudo-inverse. Since $X\beta$ belongs to this subspace, we have $P_F X\beta = X\beta$.

Let $\mathbf{z} = (I - P_F)\mathbb{E}\mathbf{y}/\|(I - P_F)\mathbb{E}\mathbf{y}\|_2$, $\delta_1 = \|P_F(\mathbf{y} - \mathbb{E}\mathbf{y})\|_2$ and $\delta_2 = |\mathbf{z}^\top(\mathbf{y} - \mathbb{E}\mathbf{y})|$, we have

$$\begin{aligned}
&\|X\beta - \mathbb{E}\mathbf{y}\|_2^2 \\
=&\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2 + 2(\mathbf{y} - \mathbb{E}\mathbf{y})^\top(X\beta - \mathbb{E}\mathbf{y}) \\
=&\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2 + 2(\mathbf{y} - \mathbb{E}\mathbf{y})^\top(X\beta - P_F\mathbb{E}\mathbf{y}) - 2\mathbf{z}^\top(\mathbf{y} - \mathbb{E}\mathbf{y})\|(I - P_F)\mathbb{E}\mathbf{y}\|_2 \\
=&\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2 + 2(\mathbf{y} - \mathbb{E}\mathbf{y})^\top P_F(X\beta - P_F\mathbb{E}\mathbf{y}) - 2\mathbf{z}^\top(\mathbf{y} - \mathbb{E}\mathbf{y})\|(I - P_F)\mathbb{E}\mathbf{y}\|_2 \\
\le&\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2 + 2\delta_1\|X\beta - P_F\mathbb{E}\mathbf{y}\|_2 + 2\delta_2\|(I - P_F)\mathbb{E}\mathbf{y}\|_2 \\
\le&\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2 + 2\sqrt{\delta_1^2 + \delta_2^2}\sqrt{\|X\beta - P_F\mathbb{E}\mathbf{y}\|_2^2 + \|(I - P_F)\mathbb{E}\mathbf{y}\|_2^2} \\
=&\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2 + 2\sqrt{\delta_1^2 + \delta_2^2}\|X\beta - \mathbb{E}\mathbf{y}\|_2.
\end{aligned}$$

Note that in the above derivation, the first two equalities are simple algebra. The third equality uses the fact that $P_F X\beta = X\beta$. The first inequality uses the Cauchy-Schwartz inequality and the definitions of δ_1 and δ_2. The second inequality uses the Cauchy-Schwartz inequality of the form $\delta_1 a_1 + \delta_2 a_2 \le \sqrt{\delta_1^2 + \delta_2^2}\sqrt{a_1^2 + a_2^2}$. The last equality uses the fact that $\|X\beta - P_F\mathbb{E}\mathbf{y}\|_2^2 + \|(I - P_F)\mathbb{E}\mathbf{y}\|_2^2 = \|X\beta - \mathbb{E}\mathbf{y}\|_2^2$, which is a consequence of the fact that P_F is a projection matrix and $P_F X\beta = X\beta$.

Now, by solving the above displayed inequality with respect to $\|X\beta - \mathbb{E}\mathbf{y}\|_2$, we obtain

$$\begin{aligned}
\|X\beta - \mathbb{E}\mathbf{y}\|_2^2 \le& \left[\sqrt{\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2 + \delta_1^2 + \delta_2^2} + \sqrt{\delta_1^2 + \delta_2^2}\right]^2 \\
\le&(1 + a)[\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2] + (2 + a + 1/a)(\delta_1^2 + \delta_2^2).
\end{aligned}$$

The desired bound now follows easily from Proposition 3 and Lemma 14, where we know that with probability $1 - \eta/2$,

$$\delta_1^2 = (\mathbf{y} - \mathbb{E}\mathbf{y})^\top P_F(\mathbf{y} - \mathbb{E}\mathbf{y}) \le \sigma^2(7.4|F| + 2.7\ln(4/\eta)),$$

and with probability $1 - \eta/2$,

$$\delta_2^2 = |\mathbf{z}^\top(\mathbf{y} - \mathbb{E}\mathbf{y})|^2 \le 2\sigma^2\ln(4/\eta).$$

We obtain the desired result by substituting the above two estimates and simplify. ■

Lemma 18 *Suppose that Assumption 1 is valid. Then we have with probability* $1 - \eta$, $\forall \beta \in \mathbb{R}^p$ *and* $a > 0$*:*

$$\|X\beta - \mathbb{E}\mathbf{y}\|_2^2 \le (1 + a)\left[\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2\right] + (2 + a + 1/a)\sigma^2[7.4c(\beta) + 4.7\ln(4/\eta)].$$

Proof Note that for each F, with probability $2^{-\mathrm{cl}(F)}\eta$, we obtain from Lemma 17 that $\forall \mathrm{supp}(\beta) \in F$,

$$\|X\beta - \mathbb{E}\mathbf{y}\|_2^2 \le (1 + a)\left[\|X\beta - \mathbf{y}\|_2^2 - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2\right] + (2 + a + 1/a)\sigma^2[7.4(|F| + \mathrm{cl}(F)) + 4.7\ln(4/\eta)].$$

Since $\sum_{F \subset I, F \neq \emptyset} 2^{-\mathrm{cl}(F)}\eta \le \eta$, the result follows from the union bound. ■

Lemma 19 *Consider a fixed subset $\bar{F} \subset I$. Given any $\eta \in (0,1)$, we have with probability $1-\eta$:*

$$|\|X\bar{\beta}-\mathbf{y}\|_2^2 - \|\mathbf{y}-\mathbb{E}\mathbf{y}\|_2^2| \le \|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 2\sigma\sqrt{2\ln(2/\eta)}\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2.$$

Proof Let $\tilde{a} = (X\bar{\beta}-\mathbb{E}\mathbf{y})/\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2$, we have

$$\begin{aligned}
&|\|X\bar{\beta}-\mathbf{y}\|_2^2 - \|\mathbf{y}-\mathbb{E}\mathbf{y}\|_2^2| \\
=&|-2(X\bar{\beta}-\mathbb{E}\mathbf{y})^\top(\mathbf{y}-\mathbb{E}\mathbf{y}) + \|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2| \\
\le& 2\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2|\tilde{a}^\top(\mathbf{y}-\mathbb{E}\mathbf{y})| + \|\mathbb{E}\mathbf{y}-X\bar{\beta}\|_2^2.
\end{aligned}$$

The desired result now follows from Lemma 14. ■

Lemma 20 *Suppose that Assumption 1 is valid. Consider any fixed target $\bar{\beta} \in \mathbb{R}^p$. Then with probability exceeding $1-\eta$, for all $\lambda \ge 0, \varepsilon \ge 0, \hat{\beta} \in \mathbb{R}^p$ such that: $\hat{Q}(\hat{\beta}) + \lambda c(\hat{\beta}) \le \hat{Q}(\bar{\beta}) + \lambda c(\bar{\beta}) + \varepsilon$, and for all $a > 0$, we have*

$$\begin{aligned}
\|X\hat{\beta}-\mathbb{E}\mathbf{y}\|_2^2 \le& (1+a)[\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 2\sigma\sqrt{2\ln(6/\eta)}\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2] \\
&+ (1+a)\lambda c(\bar{\beta}) + a' c(\hat{\beta}) + b'\ln(6/\eta) + (1+a)\varepsilon,
\end{aligned}$$

where $a' = 7.4(2+a+a^{-1})\sigma^2 - (1+a)\lambda$ and $b' = 4.7\sigma^2(2+a+a^{-1})$. Moreover, if the coding scheme $c(\cdot)$ is sub-additive, then

$$n\rho_-(c(\hat{\beta})+c(\bar{\beta}))\|\hat{\beta}-\bar{\beta}\|_2^2 \le 10\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 2.5\lambda c(\bar{\beta}) + (37\sigma^2 - 2.5\lambda)c(\hat{\beta}) + 29\sigma^2\ln(6/\eta) + 2.5\varepsilon.$$

Proof We obtain from the union bound of Lemma 18 (with probability $1-\eta/3$) and Lemma 19 (with probability $1-2\eta/3$) that with probability $1-\eta$:

$$\begin{aligned}
&\|X\hat{\beta}-\mathbb{E}\mathbf{y}\|_2^2 \\
\le& (1+a)\left[\|X\hat{\beta}-\mathbf{y}\|_2^2 - \|\mathbf{y}-\mathbb{E}\mathbf{y}\|_2^2\right] + (2+a+a^{-1})[7.4\sigma^2 c(\hat{\beta}) + 4.7\sigma^2\ln(6/\eta)] \\
\le& (1+a)\left[\|X\bar{\beta}-\mathbf{y}\|_2^2 - \|\mathbf{y}-\mathbb{E}\mathbf{y}\|_2^2 + \lambda c(\bar{\beta}) + \varepsilon\right] + a' c(\hat{\beta}) + b'\ln(6/\eta) \\
\le& (1+a)[\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 2\sigma\sqrt{2\ln(6/\eta)}\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2] + (1+a)\lambda c(\bar{\beta}) + a' c(\hat{\beta}) \\
&+ b'\ln(6/\eta) + (1+a)\varepsilon.
\end{aligned}$$

This proves the first claim of the lemma.

The first claim with $a = 1$ implies that

$$\begin{aligned}
&\|X\hat{\beta}-X\bar{\beta}\|_2^2 \le [\|X\hat{\beta}-\mathbb{E}\mathbf{y}\|_2 + \|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2]^2 \\
\le& 1.25\|X\hat{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 5\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 \\
\le& 7.5\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 5\sigma\sqrt{2\ln(6/\eta)}\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2 + 2.5\lambda c(\bar{\beta}) + 1.25(29.6\sigma^2 - 2\lambda)c(\hat{\beta}) \\
&+ 1.25\times 18.8\sigma^2\ln(6/\eta) + 2.5\varepsilon \\
\le& 10\|X\bar{\beta}-\mathbb{E}\mathbf{y}\|_2^2 + 2.5\lambda c(\bar{\beta}) + (37\sigma^2 - 2.5\lambda)c(\hat{\beta}) + 29\sigma^2\ln(6/\eta) + 2.5\varepsilon.
\end{aligned}$$

Since $c(\hat{\beta}-\bar{\beta}) \le c(\hat{\beta}) + c(\bar{\beta})$, we have $\|X\hat{\beta}-X\bar{\beta}\|_2^2 \ge n\rho_-(c(\hat{\beta})+c(\bar{\beta}))\|\hat{\beta}-\bar{\beta}\|_2^2$. This implies the second claim. ■

D.1 Proof of Theorem 6

We take $\lambda = 0$ in Lemma 20, and obtain:

$$\begin{aligned}\|X\hat{\beta} - \mathbb{E}\mathbf{y}\|_2^2 \leq & (1+a)[\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2^2 + 2\sigma\sqrt{2\ln(6/\eta)}\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2] \\ & + 7.4(2+a+a^{-1})\sigma^2 c(\hat{\beta}) + 4.7\sigma^2(2+a+a^{-1})\ln(6/\eta) + (1+a)\varepsilon \\ = & (\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2 + \sigma\sqrt{2\ln(6/\eta)})^2 + 14.8\sigma^2 c(\hat{\beta}) + 7.4\sigma^2\ln(6/\eta) + \varepsilon \\ & + a[(\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2 + \sigma\sqrt{2\ln(6/\eta)})^2 + 7.4\sigma^2 c(\hat{\beta}) + 2.7\sigma^2\ln(6/\eta) + \varepsilon] \\ & + a^{-1}[7.4\sigma^2 c(\hat{\beta}) + 4.7\sigma^2\ln(6/\eta)].\end{aligned}$$

Now let $z = \|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2 + \sigma\sqrt{2\ln(6/\eta)}$, and we choose a to minimize the right hand side as:

$$\begin{aligned}\|X\hat{\beta} - \mathbb{E}\mathbf{y}\|_2^2 \leq & z^2 + 14.8\sigma^2 c(\hat{\beta}) + 7.4\sigma^2\ln(6/\eta) + \varepsilon \\ & + 2[z^2 + 7.4\sigma^2 c(\hat{\beta}) + 2.7\sigma^2\ln(6/\eta) + \varepsilon]^{1/2}[7.4\sigma^2 c(\hat{\beta}) + 4.7\sigma^2\ln(6/\eta)]^{1/2} \\ \leq & [(z^2 + 7.4\sigma^2 c(\hat{\beta}) + 2.7\sigma^2\ln(6/\eta) + \varepsilon)^{1/2} + (7.4\sigma^2 c(\hat{\beta}) + 4.7\sigma^2\ln(6/\eta))^{1/2}]^2 \\ \leq & [z + 2(7.4\sigma^2 c(\hat{\beta}) + 4.7\sigma^2\ln(6/\eta) + \varepsilon)^{1/2}]^2.\end{aligned}$$

This proves the first inequality. The second inequality follows directly from Lemma 20 with $\lambda = 0$.

D.2 Proof of Theorem 7

The desired bound is a direct consequence of Lemma 20, by noticing that

$$2\sigma\sqrt{2\ln(6/\eta)}\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2 \leq a\|X\bar{\beta} - \mathbb{E}\mathbf{y}\|_2^2 + a^{-1}2\sigma^2\ln(6/\eta),$$

$a' \leq 0$, and

$$b' + a^{-1}2\sigma^2 \leq (10 + 5a + 7a^{-1})\sigma^2.$$

Appendix E. Proof of Theorem 9

The following lemma is an adaptation of a similar result in Zhang (2011) on greedy algorithms for standard sparsity.

Lemma 21 *Suppose the coding scheme is sub-additive. Consider any $\bar{\beta}$, and a cover of $\bar{\beta}$ by $\mathcal{B}$:*

$$\mathrm{supp}(\bar{\beta}) \subset \bar{F} = \cup_{j=1}^{b}\bar{B}_j \quad (\bar{B}_j \in \mathcal{B}).$$

Let $c(\bar{\beta}, \mathcal{B}) = \sum_{j=1}^{b} c(\bar{B}_j)$. Let $\rho_0 = \max_j \rho_+(\bar{B}_j)$. Then consider F such that $\forall j : c(\bar{B}_j \cup F) \geq c(F)$, we define

$$\beta = \arg\min_{\beta' \in \mathbb{R}^p} \|X\beta' - \mathbf{y}\|_2^2 \quad \textit{subject to} \quad \mathrm{supp}(\beta') \subset F.$$

If $\|X\beta - \mathbf{y}\|_2^2 \geq \|X\bar{\beta} - \mathbf{y}\|_2^2$, we have

$$\max_j \phi(\bar{B}_j) \geq \frac{\rho_-(F \cup \bar{F})}{\rho_0 c(\bar{\beta}, \mathcal{B})}[\|X\beta - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2],$$

where as in (5), we define

$$\phi(B) = \frac{\|P_{B-F}(X\beta - \mathbf{y})\|_2^2}{c(B \cup F) - c(F)}.$$

Proof For all $\ell \in F$, $\|X\beta + \alpha X\mathbf{e}_\ell - \mathbf{y}\|_2^2$ achieves the minimum at $\alpha = 0$ (where $\mathbf{e}_\ell$ is the vector of zeros except for the ℓ-th component, which is one). This implies that

$$\mathbf{x}_\ell^\top (X\beta - \mathbf{y}) = 0$$

for all $\ell \in F$. Therefore we have

$$\begin{aligned}
&(X\beta - \mathbf{y})^\top \sum_{\ell \in \bar{F} - F} (\bar{\beta}_\ell - \beta_\ell)\mathbf{x}_\ell \\
=&(X\beta - \mathbf{y})^\top \sum_{\ell \in \bar{F} \cup F} (\bar{\beta}_\ell - \beta_\ell)\mathbf{x}_\ell = (X\beta - \mathbf{y})^\top (X\bar{\beta} - X\beta) \\
=&-\frac{1}{2}\|X(\bar{\beta} - \beta)\|_2^2 + \frac{1}{2}\|X\bar{\beta} - \mathbf{y}\|_2^2 - \frac{1}{2}\|X\beta - \mathbf{y}\|_2^2.
\end{aligned}$$

Now, let $\bar{B}'_j \subset \bar{B}_j - F$ be disjoint sets such that $\cup_j \bar{B}'_j = \bar{F} - F$. The above inequality leads to the following derivation $\forall \eta > 0$:

$$\begin{aligned}
&-\sum_j \phi(\bar{B}_j)(c(\bar{B}_j \cup F) - c(F)) \\
\leq&\sum_j \left[\left\| X\beta + \eta \sum_{\ell \in \bar{B}'_j} (\bar{\beta}_\ell - \beta_\ell)\mathbf{x}_\ell - \mathbf{y} \right\|_2^2 - \|X\beta - \mathbf{y}\|_2^2 \right] \\
\leq&\eta^2 \sum_{\ell \in \bar{F} - F} (\bar{\beta}_\ell - \beta_\ell)^2 \rho_0 n + 2\eta (X\beta - \mathbf{y})^\top \sum_{\ell \in \bar{F} - F} (\bar{\beta}_\ell - \beta_\ell)\mathbf{x}_\ell \\
\leq&\eta^2 \sum_{\ell \in \bar{F} - F} (\bar{\beta}_\ell - \beta_\ell)^2 \rho_0 n - \eta\|X(\bar{\beta} - \beta)\|_2^2 + \eta\|X\bar{\beta} - \mathbf{y}\|_2^2 - \eta\|X\beta - \mathbf{y}\|_2^2.
\end{aligned}$$

Note that we have used the fact that $\|P_{B-F}(X\beta - \mathbf{y})\|_2^2 \geq \|X\beta - \mathbf{y}\|_2^2 - \|X\beta - \mathbf{y} + X\Delta\beta\|_2^2$ for all $\Delta\beta$ such that $\mathrm{supp}(\Delta\beta) \subset B - F$. By optimizing over η, we obtain

$$\begin{aligned}
\max_j \phi(\bar{B}_j) \sum_j c(\bar{B}_j) \geq& \sum_j \phi(\bar{B}_j)(c(\bar{B}_j \cup F) - c(F)) \\
\geq& \frac{[\|X(\bar{\beta} - \beta)\|_2^2 + \|X\beta - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2]^2}{4\sum_{\ell \in \bar{F} - F} (\bar{\beta}_\ell - \beta_\ell)^2 \rho_0 n} \\
\geq& \frac{4\|X(\bar{\beta} - \beta)\|_2^2 [\|X\beta - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2]}{4\sum_{\ell \in \bar{F} - F} (\bar{\beta}_\ell - \beta_\ell)^2 \rho_0 n} \\
\geq& \frac{\rho_-(F \cup \bar{F})}{\rho_0} [\|X\beta - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2].
\end{aligned}$$

This leads to the desired bound. In the above derivation, the first inequality is simple algebra; the second inequality is by optimizing over η mentioned earlier; the third inequality is of the form $[a_1 + a_2]^2 \geq 4a_1 a_2$. The last inequality uses the definition of $\rho_-(\cdot)$. ■

E.1 Proof of Theorem 9

Let

$$\nu' = \frac{\nu\rho_-(s+c(\bar{F}))}{\rho_0(\mathcal{B})c(\bar{\beta},\mathcal{B})}.$$

By Lemma 21, we have at any step $k > 0$:

$$\|X\beta^{(k-1)} - \mathbf{y}\|_2^2 - \|X\beta^{(k)} - \mathbf{y}\|_2^2 \geq \nu'[\|X\beta^{(k-1)} - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2](c(\beta^{(k)}) - c(\beta^{(k-1)}),$$

which implies that

$$\max[0, \|X\beta^{(k)} - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2] \leq \max[0, \|X\beta^{(k-1)} - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2]e^{-\nu'(c(\beta^{(k)})-c(\beta^{(k-1)})}.$$

Therefore at stopping, we have

$$\begin{aligned}
&\|X\beta^{(k)} - \mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2 \\
\leq &[\|\mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2]e^{-\nu' c(\beta^{(k)})} \\
\leq &[\|\mathbf{y}\|_2^2 - \|X\bar{\beta} - \mathbf{y}\|_2^2]e^{-\nu' s} \leq \varepsilon.
\end{aligned}$$

This proves the theorem.

Appendix F. Performance of StructOMP for Weakly Sparse Signals

Theorem 22 *Suppose the coding scheme is sub-additive. Given a sequence of targets $\bar{\beta}_j$ such that $\hat{Q}(\bar{\beta}_0) \leq \hat{Q}(\bar{\beta}_1) \leq \cdots$ and $c(\bar{\beta}_j, \mathcal{B}) \leq c(\bar{\beta}_0, \mathcal{B})/2^j$. If*

$$s \geq \frac{\rho_0(\mathcal{B})}{\nu \min_j \rho_-(s+c(\beta_j))} c(\bar{\beta}_0, \mathcal{B}) \left[3.4 + \sum_{j=0}^{\infty} 2^{-j} \ln \frac{\hat{Q}(\bar{\beta}_{j+1}) - \hat{Q}(\bar{\beta}_0) + \varepsilon}{\hat{Q}(\bar{\beta}_j) - \hat{Q}(\beta_0) + \varepsilon}\right]$$

for some $\varepsilon > 0$. Then at the stopping time k, we have

$$\hat{Q}(\beta^{(k)}) \leq \hat{Q}(\bar{\beta}_0) + \varepsilon.$$

Proof For simplicity, let $f_j = \hat{Q}(\bar{\beta}_j)$. For each $k = 1, 2, \ldots$ before the stopping time, let j_k be the largest j such that

$$\hat{Q}(\beta^{(k)}) \geq f_j + f_j - f_0 + \varepsilon.$$

Let $\nu' = (\nu \min_j \rho_-(s + c(\bar{\beta}_j)))/(\rho_0(\mathcal{B})c(\bar{\beta}_0, \mathcal{B}))$.

We prove by contradiction. Suppose that the theorem does not hold, then for all k before stopping, we have $j_k \geq 0$.

For each $k > 0$ before stopping, if $j_k = j_{k-1} = j$, then we have from Lemma 21 (with $\bar{\beta} = \bar{\beta}_j$)

$$c(\beta^{(k)}) \leq c(\beta^{(k-1)}) + \nu'^{-1} 2^{-j} \ln \frac{\|X\beta^{(k-1)} - \mathbf{y}\|_2^2 - f_j}{\|X\beta^{(k)} - \mathbf{y}\|_2^2 - f_j}.$$

Therefore for each $j \geq 0$, we have:

$$\sum_{k: j_k = j_{k-1} = j} [c(\beta^{(k)}) - c(\beta^{(k-1)})] \leq \nu'^{-1} 2^{-j} \ln \frac{2(f_{j+1} - f_0 + \varepsilon)}{f_j - f_0 + \varepsilon}.$$

Moreover, for each $j \geq 0$, Lemma 21 (with $\bar{\beta} = \bar{\beta}_j$) implies that

$$\sum_{k: j_k = j, j_{k-1} > j} [c(\beta^{(k)}) - c(\beta^{(k-1)})] \leq \nu'^{-1} 2^{-j}.$$

Therefore we have

$$\sum_{k: j_k = j} [c(\beta^{(k)}) - c(\beta^{(k-1)})] \leq \nu'^{-1} 2^{-j} \left[1.7 + \ln \frac{f_{j+1} - f_0 + \varepsilon}{f_j - f_0 + \varepsilon} \right].$$

Now by summing over $j \geq 0$, we have

$$c(\beta^{(k)}) \leq 3.4\nu'^{-1} + \nu'^{-1} \sum_{j=0}^{\infty} 2^{-j} \ln \frac{f_{j+1} - f_0 + \varepsilon}{f_j - f_0 + \varepsilon} \leq s.$$

This is a contradiction because we know at stopping, we should have $c(\beta^{(k)}) > s$. ■

In the above theorem, we can see that if the signal is only weakly sparse, in that $(\hat{Q}(\bar{\beta}_{j+1}) - \hat{Q}(\bar{\beta}_0) + \varepsilon)/(\hat{Q}(\bar{\beta}_j) - \hat{Q}(\bar{\beta}_0) + \varepsilon)$ grows sub-exponentially in j, then we can choose $s = O(c(\underline{\beta}_0, \mathcal{B}))$. This means that we can find $\beta^{(k)}$ of complexity $s = O(c(\beta_0, \mathcal{B}))$ to approximate a signal β_0. The worst case scenario is when $\hat{Q}(\bar{\beta}_1) \approx \hat{Q}(0)$, which reduces to the $s = O(c(\bar{\beta}_0, \mathcal{B}) \log(1/\varepsilon))$ complexity in Theorem 9.

As an application, we introduce the following concept of weakly sparse compressible target that generalizes the corresponding concept of compressible signal in standard sparsity from the compressive sensing literature (Donoho, 2006). A related extension has also appeared in Baraniuk et al. (2010).

Definition 23 *The target $\mathbb{E}\mathbf{y}$ is $(a, \underline{q})$-compressible with respect to block $\mathcal{B}$ if there exist constants $a, q > 0$ such that for each $s > 0$, $\exists \bar{\beta}(s)$ such that $c(\bar{\beta}(s), \mathcal{B}) \leq s$ and*

$$\frac{1}{n} \|X\bar{\beta}(s) - \mathbb{E}\mathbf{y}\|_2^2 \leq a s^{-q}.$$

Corollary 24 *Suppose that the target is (a, q)-compressible with respect to $\mathcal{B}$. Then with probability $1 - \eta$, at the stopping time k, we have*

$$\hat{Q}(\beta^{(k)}) \leq \hat{Q}(\bar{\beta}(s')) + 2na/s'^q + 2\sigma^2[\ln(2/\eta) + 1],$$

where

$$s' \leq \frac{s\nu}{(10 + 3q)\rho_0(\mathcal{B})} \min_{u \leq s'} \rho_-(s + c(\bar{\beta}(u))).$$

Proof Given s', we consider $f_j = \min_{\ell \geq j} \hat{Q}(\bar{\beta}(s'/2^\ell))$. We also assume that f_0 is achieved at $\ell_0 \geq 0$. Note that by Lemma 19, we have with probability $1 - 2^{-j-1}\eta$:

$$\begin{aligned} |\hat{Q}(\bar{\beta}(s'/2^j)) - \|\mathbf{y} - \mathbb{E}\mathbf{y}\|_2^2| \leq & 2\|X\bar{\beta}(s'/2^j) - \mathbb{E}\mathbf{y}\|_2^2 + 2\sigma^2[j + 1 + \ln(2/\eta)] \\ \leq & 2an2^{qj}/s'^q + 2\sigma^2[j + 1 + \ln(2/\eta)]. \end{aligned}$$

This means the above inequality holds for all j with probability $1-\eta$. Therefore

$$\begin{aligned} f_{j+1} - f_0 \leq & \hat{Q}(\bar{\beta}(s'/2^{j+1})) - \hat{Q}(\bar{\beta}(s')) \\ \leq & |\hat{Q}(\bar{\beta}(s'/2^{j+1})) - \|\mathbf{y}-\mathbb{E}\mathbf{y}\|_2^2| + |\hat{Q}(\bar{\beta}(s')) - \|\mathbf{y}-\mathbb{E}\mathbf{y}\|_2^2| \\ \leq & 4an2^{q(j+1)}/s'^q + 4\sigma^2[0.5j+1+\ln(2/\eta)]. \end{aligned}$$

Now, by taking $\varepsilon = 2an/s'^q + 2\sigma^2[\ln(2/\eta)+1]$ in Theorem 22, we obtain

$$\begin{aligned} \sum_{j=0}^{\infty} 2^{-j} \ln \frac{f_{j+1}-f_0+\varepsilon}{f_j - f_0 + \varepsilon} &\leq \sum_{j=\ell_0}^{\infty} 2^{-j} \ln(1+(f_{j+1}-f_0)/\varepsilon) \\ &\leq \sum_{j=\ell_0}^{\infty} 2^{-j} \ln(4+2(0.5j+2^{q(j+1)})) \\ &\leq \sum_{j=\ell_0}^{\infty} 2^{-j}(2+0.5j+\ln 2+q(j+1)\ln 2) \leq 4.4+4(0.5+q\ln 2), \end{aligned}$$

where we have used the simple inequality $\ln(\alpha+2\beta) \leq 0.5\alpha + \ln(2\beta)$ when $\alpha, \beta \geq 1$. Therefore,

$$\begin{aligned} s &\geq \frac{\rho_0(\mathcal{B})s'}{\nu \min_{u \leq s'} \rho_-(s+c(\bar{\beta}(u)))}(10+3q) \\ &\geq \frac{\rho_0(\mathcal{B})s'}{\nu \min_{u \leq s'} \rho_-(s+c(\bar{\beta}(u)))} \left[3.4 + \sum_{j=0}^{\infty} 2^{-j} \ln \frac{f_{j+1}-f_0+\varepsilon}{f_j-f_0+\varepsilon}\right]. \end{aligned}$$

This means that Theorem 22 can be applied to obtain the desired bound. ■

If we assume the underlying coding scheme is block coding generated by $\mathcal{B}$, then we have $\min_{u \leq s'} \rho_-(s+c(\bar{\beta}(u))) \leq \rho_-(s+s')$. The corollary shows that we can approximate a compressible signal of complexity s' with complexity $s = O(qs')$ using greedy algorithm. This means the greedy algorithm obtains optimal rate for weakly-sparse compressible signals. The sample complexity suffers only a constant factor $O(q)$. Combine this result with Theorem 6, and take union bound, we have with probability $1-2\eta$, at stopping time k:

$$\begin{aligned} &\frac{1}{\sqrt{n}}\|X\beta^{(k)} - \mathbb{E}\mathbf{y}\|_2 \leq \sqrt{\frac{a}{s'^q}} + \sigma\sqrt{\frac{2\ln(6/\eta)}{n}} + 2\sigma\sqrt{\frac{7.4(s+c_0(\mathcal{B}))+6.7\ln(6/\eta)}{n} + \frac{2a}{\sigma^2 s'^q}}, \\ &\|\beta^{(k)} - \bar{\beta}(s')\|_2^2 \leq \frac{1}{\rho_-(s+s'+c_0(\mathcal{B}))}\left[\frac{15a}{s'^q} + \frac{37\sigma^2(s+c_0(\mathcal{B}))+34\sigma^2\ln(6/\eta)}{n}\right]. \end{aligned}$$

Given a fixed n, we can obtain a convergence result by choosing s (and thus s') to optimize the right hand side. The resulting rate is optimal for the special case of standard sparsity, which implies that the bound has the optimal form for structured q-compressible targets. In particular, in compressive sensing applications where $\sigma = 0$, we obtain when sample size reaches $n = O(qs')$, the reconstruction performance is

$$\|\bar{\beta}^{(k)} - \bar{\beta}\|_2^2 = O(a/s'^q),$$

which matches that of the constrained coding complexity regularization method in (2) up to a constant $O(q)$. Since many real data involve weakly sparse signals, our result provides strong theoretical justification for the use of OMP in such problems. Our experiments are consistent with the theory.

References

A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning Journal*, 73:243–272, 2008.

I. Johnstone B. Efron, T. Hastie and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32: 407–499, 2004.

F. R. Bach. Consistency of the group lasso and multiple kernel learning. *Journal of Machine Learning Research*, 9:1179–1225, 2008.

R. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde. Model based compressive sensing. *IEEE Transactions on Information Theory*, 56:1982–2001, 2010.

F. Bunea, A. B. Tsybakov, and M. H. Wegkamp. Aggregation for Gaussian regression. *Annals of Statistics*, 35:1674–1697, 2007.

E. J. Candes and T. Tao. Decoding by linear programming. *IEEE Transaction on Information Theory*, 51:4203–4215, 2005.

V. Cevher, M. F. Duarte, C. Hegde, and R. Baraniuk. Sparse signal recovery using markov random fields. In *Proceeding of NIPS*, 2009a.

V. Cevher, P. Indyk, C. Hegde, and R. G. Baraniuk. Recovery of clustered sparse signals from compressive measurements. In *Proceeding of the 8th International Conference on Sampling Theory and Applications*, 2009b.

C. Chesneau and M. Hebiri. Some theoretical results on the grouped variables Lasso. *Mathematical Methods of Statistics*, 17:317–326, 2008.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.

A. d'Aspremont, F. Bach, and L. El Ghaoui. Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9:1269–1294, 2008.

L. Daudet. Sparse and structured decomposition of audio signals in overcomplete spaces. In *Proceeding of International Conference on Digital Audio Effects*, 2004.

D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52:1289–1306, 2006.

D. Grimm, T. Netzer, and M. Schweighofer. A note on the representation of positive polynomials with structured sparsity. *Archiv der Mathematik*, 89:399–403, 2007.

J. Haupt and R. Nowak. Signal reconstruction from noisy projections. *IEEE Trans. Information Theory*, 52:4036–4048, 2006.

L. He and L. Carin. Exploiting structure in wavelet-based bayesian compressive sensing. *IEEE Transaction on Signal Processing*, 57:3488–3497, 2009a.

L. He and L. Carin. Exploiting structure in compressive sensing with a jpeg basis. In *Preprint*, 2009b.

J. Huang and T. Zhang. The benefit of group sparsity. *Annals of Statistics*, 38(4):1978–2004, 2010.

L. Jacob, G. Obozinski, and J. Vert. Group lasso with overlap and graph lasso. In *Proceedings of ICML*, 2009.

R. Jenatton, J. Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. Technical report, Tech Report: arXiv:0904, 2009.

S. Ji, D. Dunson, and L. Carin. Multi-task compressive sensing. *IEEE Transactions on Signal Processing*, 2008. Accepted.

V. Koltchinskii and M. Yuan. Sparse recovery in large ensembles of kernel machines. In *Proceeding of COLT*, 2008.

M. Kowalski and B. Torresani. Structured sparsity: from mixed norms to structured shrinkage. In *Workshop on Signal Processing with Adaptive Sparse Representations*, 2009.

K. Lounici, M. Pontil, A. B. Tsybakov, and S. A. Van De Geer. Taking advantage of sparsity in multi-task learnng. In *Proceeding of COLT*, 2009.

A. Lozano, G. Swirszcz, and N. Abe. Group orthogonal matching pursuit for variable selection and prediction. In *Proceedings of NIPS*, 2009.

Y. M. Lu and M. N. Do. A theory for sampling signals from a union of subspaces. *IEEE Transactions on Signal Processing*, 56(6):2334–2345, 2008.

S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1999.

Y. Nardi and A. Rinaldo. On the asymptotic properties of the group lasso estimator for linear models. *Electronic Journal of Statistics*, 2:605–633, 2008.

G. Obozinski, M. J. Wainwright, and M. I. Jordan. Union support recovery in high-dimensional multivariate regression. Technical Report 761, UC Berkeley, 2008.

G. Pisier. *The Volume of Convex Bodies and Banach Space Geometry*. Cambridge University Press, 1989.

J. M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41:3445–3462, 1993.

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

J. Tropp and A. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.

D. Wipf and B. Rao. An empirical Bayesian strategy for solving the simultaneous sparse approximation problem. *IEEE Transactions on Signal Processing*, 55(7):3704–3716, 2007.

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

T. Zhang. Adaptive forward-backward greedy algorithm for learning sparse representations. *IEEE Transactions on Information Theory*, 57:4689–4708, 2011.

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

Z. Zivkovic and F. Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27(7):773–780, 2006.