Clustered Sparse Bayesian Learning

Yu Wang * **David Wipf** † **Jeong-Min Yun** ♯ **Wei Chen*** **Ian Wassell** *

* University of Cambridge, Cambridge, UK
† Microsoft Research, Beijing, China
♯Pohang University of Science and Technology, Pohang, Republic of Korea
yw323@cam.ac.uk davidwip@microsoft.com azida@postech.ac.kr wc253@cam.ac.uk ijw24@cam.ac.uk

Abstract

Many machine learning and signal processing tasks involve computing sparse representations using an overcomplete set of features or basis vectors, with compressive sensing-based applications a notable example. While traditionally such problems have been solved individually for different tasks, this strategy ignores strong correlations that may be present in real world data. Consequently there has been a push to exploit these statistical dependencies by jointly solving a series of sparse linear inverse problems. In the majority of the resulting algorithms however, we must a priori decide which tasks can most judiciously be grouped together. In contrast, this paper proposes an integrated Bayesian framework for both clustering tasks together and subsequently learning optimally sparse representations within each cluster. While probabilistic models have been applied previously to solve these types of problems, they typically involve a complex hierarchical Bayesian generative model merged with some type of approximate inference, the combination of which renders rigorous analysis of the underlying behavior virtually impossible. On the other hand, our model subscribes to concrete motivating principles that we carefully evaluate both theoretically and empirically. Importantly, our analyses take into account all approximations that are involved in arriving at the actual cost function to be optimized. Results on synthetic data as well as image recovery from compressive measurements show improved performance over existing methods.

1 INTRODUCTION

Solving sparse linear inverse problems is a fundamental building block in numerous machine learning, computer vision, and signal processing applications related to compressive sensing and beyond (Elhamifar & Vidal 2013; Soltanolkotabi & Candes, 2012; Zhang & Rao, 2011; Hu et al., 2013). In its most basic form, sparse estimation algorithms are built upon the observation model

$$\boldsymbol{y} = \Phi \boldsymbol{x} + \boldsymbol{\epsilon}, \tag{1}$$

where $\Phi \in \mathbb{R}^{N \times M}$ is a dictionary of basis vectors that we assume to have unit ℓ_2 norm, $\boldsymbol{x} \in \mathbb{R}^M$ is a vector of unknown coefficients we would like to estimate, $\boldsymbol{y} \in \mathbb{R}^N$ is an observed measurement vector, and $\boldsymbol{\epsilon}$ is a noise vector distributed as $\mathcal{N}(0, \nu I)$. The objective is to estimate the unknown generative $\boldsymbol{x}$ under the assumption that it is maximally sparse, meaning that $\|\boldsymbol{x}\|_0$ is minimal. Here $\|\cdot\|_0$ represents the canonical ℓ_0 norm sparsity metric, or a count of the number of nonzero elements in a vector. This sparse linear inverse problem is compounded considerably by the additional assumption that $M > N$, meaning the dictionary Φ is *overcomplete*.

Now suppose that we have access to multiple measurement vectors from L different tasks of interest that are assembled as $Y = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_L] \in \mathbb{R}^{N \times L}$ and linked to a corresponding matrix of unknown coefficients $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_L] \in \mathbb{R}^{M \times L}$ via

$$\boldsymbol{y}_j = \Phi_j \boldsymbol{x}_j + \boldsymbol{\epsilon}_j, \quad \forall j = 1, \ldots, L. \tag{2}$$

If these measurement vectors and associated coefficients maintain some degree of dependency, for example the locations of zero-valued elements (or support sets) are statistically related, then it will generally be advantageous to jointly estimate X from Y as opposed to solving for each $\boldsymbol{x}_j$ individually (Obozinski et al., 2011). Perhaps the simplest example of this is frequently referred to as simultaneous sparse approximation (Tropp, 2006) or multi-task compressive sensing (Ji et al., 2009). In brief, these paradigms follow from the assumption that X is maximally *row sparse*, implying that each column of X is maximally sparse with a common support pattern. This model has been

Y. Wang is sponsored by the University of Cambridge Overseas Trust. Y. Wang and J. Yun are partially supported by sponsorship from Microsoft Research Asia. W. Chen is supported by EPSRC Research Grant EP/K033700/1 and the NSFC Research Grant 61401018.

applied to compressive sensing of images and video (Ji et al., 2009), tracking (Hong et al., 2013), and medical image analysis (Wan et al., 2012). Moreover, it can be naturally extended to handle a richer set of dependencies by applying a known graph structure that groups subsets of columns together for joint estimation (Cevher et al., 2008; Shi et al., 2014), as opposed to strict enforcement of row sparsity across all measurements. Other pre-defined structural assumptions can be found in (Archambeau et al., 2011; Jalali et al., 2013; Rao et al., 2013; Yang & Ravikumar, 2013).

The limitation of all of these strategies is that they are predicated on prior knowledge of how tasks should be grouped together to optimally facilitate subsequent sparse estimation. In contrast, here we intend to develop a principled multi-task learning algorithm that simultaneously clusters tasks blindly while optimally estimating sparse coefficient vectors informed by these clusters. We should state at the outset that multi-task compressive sensing has been merged with cluster learning before (Qi et al., 2008). However, this algorithm relies on a complex hierarchical model anchored with an approximate Dirichlet process prior distribution on X (Blei & Jordan, 2006). Subsequent model inference then requires an additional variational mean-field approximation. Overall the fundamental underlying mechanics of the model have not been carefully analyzed nor understood, nor is there any guarantee that sparsity will necessarily result. Other even more complex hierarhical models have been applied to somewhat-related multi-task learning problems, e.g., (Hernandez-Lobato & Hernandez-Lobato, 2013); however, these models require complex inference procedures that must ultimately be justified by the validity of the assumed prior distributions rather than provable properties of the resulting estimators.

The remainder of the paper is organized as follows. In Section 2 we first motivate our design principles. A specific Bayesian model and corresponding objective function are then developed in Section 3. Next we derive updates rules for optimization purposes, leading to our *clustered sparse Bayesian learning algorithm* (C-SBL) in Section 4. We theoretically and empirically analyze this framework in Sections 5 and 6 respectively, revealing that it is consistent with our original motivational principles. Moreover, estimation results on synthetic data and real images demonstrate improved estimation quality relative to existing algorithms when using compressive measurements. Overall, we summarize our contributions as follows:

- Analysis of specific, previously-unexamined theoretical principles that play a critical role in multi-task sparse estimation problems.
- Development of a robust sparse Bayesian algorithm that adheres to these principles to an extent not seen in any existing algorithm we are aware of.
- Although we employ a Bayesian entry point for our algorithmic strategy, final model justification is provided entirely based on rigorous properties of the underlying cost function that emerges, *including all approximations involved*, rather than any putative belief in the actual validity of assumed prior distributions.

2 MOTIVATING PRINCIPLES

When deriving an algorithm for joint clustering and multi-task sparse estimation, it is helpful to first define a few basic attributes that ideally any procedure might possess. Here we consider properties related to limiting behavior as the noise variance ν varies from zero towards large values for each task.

First assume $\nu \rightarrow 0$ and the following generative model. Let X^* denote the true coefficient matrix we would like to estimate using measurements $\boldsymbol{y}_j = \Phi_j \boldsymbol{x}_j^*$. We assume that the columns of X^* are partitioned into clusters with common sparsity profile or support within each cluster. Additionally let Ω_k denote the column indices of X^* associated with cluster $k = 1, \ldots, K \leq L$. For any matrix Z define Z_{Ω_k} as the sub-matrix of columns associated with the index set Ω_k. Then the relevant sparse linear inverse problem, assuming known clusters, becomes

$$\min_{\{X_{\Omega_k}\}} \sum_k |\Omega_k| \|X_{\Omega_k}\|_{row-\ell_0} \quad \text{s.t. } \boldsymbol{y}_j = \Phi_j \boldsymbol{x}_j, \ \forall j, \quad (3)$$

where $\| \cdot \|_{row-\ell_0}$ counts the number of nonzero *rows* of a matrix, which is then weighted by the cardinality of the set Ω_k in the objective function.[1]

Now assume the perturbation model

$$\bar{X}_k^* = A_k^x + \alpha^x R_k^x, \ \forall k, \quad (4)$$

where $\bar{X}_{\Omega_k}^*$ denotes the nonzero rows of $X_{\Omega_k}^*$ associated with cluster k, A_k^x is an arbitrary matrix of appropriate dimensions, $\alpha^x > 0$ is an arbitrarily small scalar, and R_k^x is a random matrix with iid, continuously-distributed elements. Likewise, assume that

$$\Phi_j = A_j^\phi + \alpha^\phi R_j^\phi, \forall j, \quad (5)$$

with analogous definitions to those in (4), albeit with obviously different dimensions and values. Then we have the following:

Lemma 1. Suppose we are given any Y generated with $\boldsymbol{y}_j = \Phi_j \boldsymbol{x}_j^*$ and $\|\boldsymbol{x}_j^*\|_0 < N \ \forall j$, where X^* satisfies (4) $\forall k$ and Φ_j satisfies (5) $\forall j$. Then X^* is the unique global minimum of both (3) and

$$\min_X \sum_j \|\boldsymbol{x}_j\|_0 \quad \text{s.t. } \boldsymbol{y}_j = \Phi_j \boldsymbol{x}_j, \forall j. \quad (6)$$

[1] Actually, this weighting factor is irrelevant to what follows in the noiseless case, but does play a role later when we consider noisy conditions.

The proof is relatively straightforward and comes from modifying Theorem 1 from (Baron et al., 2009). While perhaps notationally cumbersome to present, the message of this result is simple and widely applicable. In words, Lemma 1 implies that, under general conditions that apply to virtually any multi-task system of interest (since the sets $\{A_k^x\}$ $\{A_j^\phi\}$ are arbitrary and the perturbations applied to them can be infinitesimally small), the unique maximally row-sparse solution to the clustering problem is equivalent to the global solution obtained by simply evaluating each task individually. The cluster structure itself does not provide any direct advantage, and we could just as well solve (6), and without noise we can theoretically resolve any number of clusters between 1 and L.

With this in mind then, in the limit $\nu \to 0$ we would prefer to have a clustering algorithm whose global optimum is equivalent to (6). However, we need not solve this problem directly, which in general is NP-hard. Rather we favor an algorithm that can, to the extent possible, leverage cluster information to steer the algorithm towards the global solution of (6) while avoiding bad local optima.

Now consider larger values of noise, i.e., $\nu > 0$, where we would like to solve something akin to

$$\min_X \sum_j \|\boldsymbol{y}_j - \Phi_j \boldsymbol{x}_j\|_2^2 + \nu \sum_k |\Omega_k| \|X_{\Omega_k}\|_{row-\ell_0}. \quad (7)$$

In general, when the noise level is high we cannot hope to resolve a large number of clusters, and eventually we must merge to a single cluster as ν becomes sufficiently large. In this regime the issue is not so much one of avoiding bad local minima as it is enhancing the effective signal-to-noise ratio as much as possible. Note that the smaller K is, the more tasks per cluster, which has a substantial benefit in terms of signal-to-noise ratio. This can be easily visualized via the special case where $\Phi_j^\top \Phi_j = I \ \forall j$. Given this simplification, (7) has a closed-form solution given by

$$x^*_{i,j} = z_{i,j}\mathcal{I}\left[\sum_{j\in\Omega_{c(j)}} z^2_{i,j} > \nu|\Omega_{c(j)}|\right], \quad (8)$$

where $\boldsymbol{z}_j \triangleq \Phi_j^\top \boldsymbol{y}_j$, $c(j)$ denotes the cluster index of task j, and $\mathcal{I}$ is an indicator function. Thus the optimal solution represents a hard-thresholding operation, where the threshold is dictated by an average across tasks within each cluster. If we have only a single cluster, meaning $c(j) = 1 \ \forall j$ and $\Omega_1 = \{1, \ldots, L\}$, then this threshold value is maximally robust to noise given that all tasks are averaged together to increase the SNR of the threshold.

To conclude then, there are (at least) three important considerations:

1. At high SNR, local minima avoidance while finding maximally sparse solutions is paramount. We would also favor that, for a given clustering, maximally row-sparse solutions can be obtained by evading any suboptimal local extrema where possible.

2. At low SNR when it is impossible to resolve many clusters anyway, the issue is more about merging clusters to hopefully improve the implicit SNR.

3. In intermediate regimes we would like to accomplish a bit of both.

In Section 5 we provide theoretical evidence that our proposed algorithm is favorable with respect to points 1 and 2, while Section 6 presents empirical evidence in practical support of point 3.

3 MODEL DESCRIPTION

While perhaps not immediately obvious at first, this section will develop a Bayesian model that ultimately reflects the previously stated principles. Consistent with the observation model in (1), we adopt the Gaussian likelihood function

$$p(Y|X) \propto \prod_j \exp\left[-\frac{1}{2\nu}\|\boldsymbol{y}_j - \Phi_j \boldsymbol{x}_j\|_2^2\right]. \quad (9)$$

For present purposes we will assume that the noise variance ν is known (ultimately though this value can be learned from the data). For the prior distribution on each $\boldsymbol{x}_j$ we build upon the basic sparse Bayesian learning framework from (Tipping, 2001) which in the present context would involve a zero-mean Gaussian with an independent diagonal covariance; however, this would not allow for task clustering. Instead we assume the prior distribution

$$p(X|\Lambda, W) \propto \prod_j \exp\left[-\frac{1}{2}\boldsymbol{x}_j^\top \Gamma_j^{-1} \boldsymbol{x}_j\right], \quad (10)$$

where $\Lambda \in \mathbb{R}^{M\times K}$ and $W \in \mathbb{R}^{L\times K}$ are hyperparameter matrices; Λ is constrained to have all non-negative elements, $W \in \mathcal{S}$ is defined such that each row denoted as $\boldsymbol{w}^j$ is an element of the probability simplex, i.e.,

$$\mathcal{S} \triangleq \{\boldsymbol{w}^j : \sum_k w_{j,k} = 1, w_{j,k} \in [0,1]\}. \quad (11)$$

With some abuse of notation, we say that $W \in \mathcal{S}$ if every row $\boldsymbol{w}^j \in \mathcal{S}$. Finally, Γ_j is the diagonal covariance matrix produced via

$$\Gamma_j^{-1} = \sum_k w_{j,k}\Lambda_k^{-1}, \quad (12)$$

where Λ_k is defined as a diagonal matrix formed from the k-th column of matrix Λ.

Although the unknown $\boldsymbol{x}_j$ from each task are assumed to be independent via the above distributions, they will nonetheless become linked via the common set of hyperparameters that will subsequently be estimated from the data. Additionally, from (12) we are expressing what amounts to the j-th precision matrix as a linear combination of K diagonal precision matrix basis functions. Although we could have equally considered a linear basis expansion with respect to covariances, we chose precisions for algorithmic reasons detailed below. Additionally, the value of K can be viewed as an upper bound on the number of clusters we can expect in our data; for all experiments we simply choose $K = L$, the number of tasks.

Given this likelihood and prior, the posterior distribution $p(\boldsymbol{x}_j|\boldsymbol{y}_j; \Lambda, W)$ is also a Gaussian with mean

$$\hat{\boldsymbol{x}}_j = \Gamma_j \Phi_j^\top \left(\nu I + \Phi_j \Gamma_j \Phi_j^\top\right)^{-1} \boldsymbol{y}_j. \tag{13}$$

Thus if Λ and W were known, we have access to a simple closed-form estimator for $\boldsymbol{x}_j$. The most challenging responsibility then becomes estimating these unknown hyperparameters. The empirical Bayesian solution to this problem is to first apply hyperpriors to Λ and W, integrate out the unknown X, and then compute MAP estimates via

$$\max_{\Lambda > 0, W \in \mathcal{S}} \int p(Y|X) p(X; \Lambda, W) p(\Lambda) p(W) dX. \tag{14}$$

For the covariance bases we simply assume a flat hyperior $p(\Lambda) = 1$. In contrast, we assume $p(W) \propto \exp[-1/2 \sum_{j,k} f(w_{j,k})]$, where f is a function designed to promote a clustering effect as will be described shortly. Given the above, applying a $-2 \log$ transformation to (14) produces the equivalent problem

$$\min_{\Lambda > 0, W \in \mathcal{S}} \sum_j \left[\boldsymbol{y}_j \Sigma_{y_j}^{-1} \boldsymbol{y}_j + \log \left|\Sigma_{y_j}\right| \right] + \sum_{j,k} f(w_{j,k}), \tag{15}$$

where

$$\Sigma_{y_j} \triangleq \nu I + \Phi_j \Gamma_j \Phi_j^\top.$$

To facilitate later optimization, it will help to modify the log-det term in (15) as follows. First, using standard determinant identities we have

$$\begin{aligned} \log |\Sigma_{y_j}| \quad &\equiv \quad T1 \quad + \quad T2 \qquad (16) \\ &\triangleq \log \left| \sum_k w_{j,k} \Lambda_k^{-1} + \frac{1}{\nu} \Phi_j^\top \Phi_j \right| - \log \left| \sum_k w_{j,k} \Lambda_k^{-1} \right|, \end{aligned}$$

where irrelevant constants have been omitted. Using the fact that $\log |\cdot|$ is a concave function in the space of positive definite, symmetric matrices, $W \in \mathcal{S}$, and Jensen's inequality, it follows that

$$\sum_k w_{j,k} \log |\Lambda_k| \geq -\log \left| \sum_k w_{j,k} \Lambda_k^{-1} \right|. \tag{17}$$

This upper bound has the appeal that it is linear in elements of W which will facilitate the derivation of update rules to be presented shortly. We will henceforth be concerned with minimizing the new objective function

$$\begin{aligned} \mathcal{L}(\Lambda, W) \triangleq \sum_j \left[\boldsymbol{y}_j \Sigma_{y_j}^{-1} \boldsymbol{y}_j \right] + \sum_{j,k} f(w_{j,k}) \qquad (18) \\ + \sum_j \log \left| \sum_k w_{j,k} \Lambda_k^{-1} + \frac{1}{\nu} \Phi_j^\top \Phi_j \right| + \sum_{j,k} w_{j,k} \log |\Lambda_k|. \end{aligned}$$

Sparsity Promotion: The log-det term in the original cost (15) is a concave, non-decreasing function of each Γ_j, and hence it favors sparse diagonal elements, which in turn produces a sparse $\boldsymbol{x}_j$ estimate via the left multiplication in (13). But this sparsity can only be achieved if diagonal elements of the embedded basis functions Λ_k also converge to zero.[2] By virtue of the basis expansion (12) in terms of precisions, this then implies that the sparsity profile or support of Γ_j will mirror the sparsity profile of the *intersection* of all Λ_k associated with nonzero coefficients $w_{j,k}$. Typically this will encourage only a single unique basis function to be active for a given task j.

Note that the cost function modification using Jensen's inequality above does not interfere with this sparsity promotion agency. In fact, the upper bound gap has a minimal value of zero when either $\boldsymbol{w}^j$ equals an indicator vector (all zeros and a single one), or when all Λ_k are equal to one another. The former will lead to a maximal ℓ_0 norm solution, the latter a maximal row-sparse solution.

Cluster Promotion: We now turn to the related clustering issues and the role of f. For this purpose, it is instructive to elucidate exactly what we mean by a cluster. We define a cluster as a group of tasks that share a common diagonal support for Γ_j. Without f, it is easy to show that for any value of ν, if $K = L$ the globally optimal solution to (18) will involve the k-th column of W, $\boldsymbol{w}_k$, equal to a unique indicator vector for all k, each Γ_j will then be represented with a unique Λ_k, and no clustering will occur at all. In fact, there will be no clustering effect for either the purpose of avoiding local minima when ν is small, nor for improving the effective SNR when ν is large.

To mitigate this effect, f can be chosen to encourage W to have columns with multiple nonzero values, which is tantamount to requiring that groups of tasks must share one or more Λ_k basis matrices. However, because the support of any Γ_j will be the intersection of Λ_k supports associated with $w_{j,k} > 0$, these tasks will either share only a single Λ_k, or alternatively multiple different Λ_k will converge to the same basis matrix (or at least one with a common support). In either case, the net effect is hard clustering, where each task $j \in \Omega_k$ will be assigned some effective Λ_k, and the total number of unique such basis matrices will be some $\hat{K} < L$. Additionally, within each such cluster, it can be shown by extending the analysis in (Wipf et al., 2011) that the net effect on the final estimation step is as if there were

[2] While technically division by zero is undefined, we can still accommodate (12) and all attendant update rule derivations by considering the appropriate limiting cases along with judicious use of the Matrix Inversion Lemma and the Moore-Pensrose Pseudoinverve in place of direct inverses.

an explicit, concave and nondecreasing penalty on the ℓ_2 row norms of $\hat{X}_{\Omega_k}$, which naturally favors row-sparsity.

For these reasons we choose

$$f(w) = \beta w \log w, \tag{19}$$

where $\beta > 0$ is a constant. This f is convex over the domain $[0,1]$ and has a minimal value between zero and one, and therefore favors either sharing of basis functions along columns of W or merging different Λ_k values together via the mechanism outlined above. Importantly, many elements of W will still be pushed to exactly zero to shut off basis matrices from other clusters, provably so in certain circumstances although space here prevents a detailed treatment (Section 6 does provide empirical evidence for this however). While certainly other selections for f could potentially be more effective, this simple choice serves our purposes sufficiently well and leads to convenient update rules.

4 ALGORITHM DERIVATION

Optimization of (18) will involve expanding a majorization-minimization scheme suggested in (Wipf & Nagarajan, 2010) for single-task sparse estimation, where auxiliary variables are introduced to upper bound various terms in the objective function. First we use the bound

$$\frac{1}{\nu}\|\boldsymbol{y}_j - \Phi_j \boldsymbol{x}_j\|_2^2 + \boldsymbol{x}_j^\top \Gamma_j^{-1} \boldsymbol{x}_j \geq \boldsymbol{y}_j \Sigma_{y_j}^{-1} \boldsymbol{y}_j \tag{20}$$

for all $\boldsymbol{x}_j$, with equality iff $\boldsymbol{x}_j$ is given by (13). Now define $\boldsymbol{a}_j$ as a vector formed from the diagonal of $\sum_k w_{j,k}\Lambda_k^{-1}$. Because the term $T1$ in (16) is a concave, non-decreasing function of $\boldsymbol{a}_j$, we define $h^*(\mathbf{z})$ as the concave conjugate function (Boyd & Vandenberghe, 2004) of $h(\boldsymbol{a}_j) = \log\left|\sum_k w_{j,k}\Lambda_k^{-1} + \frac{1}{\nu}\Phi_j^T\Phi_j\right|$ defined as

$$h^*(\boldsymbol{z}_j) \triangleq \inf_{\boldsymbol{a}_j}(\boldsymbol{z}_j^T \boldsymbol{a}_j - h(\boldsymbol{a}_j)). \tag{21}$$

By construction we may then upper bound $T1$ via

$$\boldsymbol{z}_j^\top \boldsymbol{a}_j - h^*(\boldsymbol{z}_j) \geq \log\left|\sum_k w_{j,k}\Lambda_k^{-1} + \frac{1}{\nu}\Phi_j^\top \Phi_j\right| \tag{22}$$

for all $\boldsymbol{z}_j \geq 0$, with equality iff $\boldsymbol{z}_j$ is the gradient of $T1$ with respect to $\boldsymbol{a}_j$. This can be computed in closed form using

$$\boldsymbol{z}_j = \nabla_{\boldsymbol{a}_j}(T1) = \operatorname{diag}\left[\left(\sum_k w_{j,k}\Lambda_k^{-1} + \frac{1}{\nu}\Phi_j^\top \Phi_j\right)^{-1}\right]. \tag{23}$$

With these upper bounds fixed, we can then optimize over Λ and W. First, with W fixed, optimization over Λ decouples and we may consider each $\lambda_{i,k}$ individually. Collecting relevant terms we have

$$\min_{\lambda_{i,k}>0} \sum_j \frac{w_{j,k}}{\lambda_{i,k}}\left(x_{i,j}^2 + z_{i,j}\right) + w_{j,k}\log\lambda_{i,k}. \tag{24}$$

Computing derivatives, equating to zero, and checking first-order optimality conditions we arrive at the optimal solution

$$\lambda_{i,k}^{opt} = \frac{\sum_j w_{j,k}\left(x_{i,j}^2 + z_{i,j}\right)}{\sum_j w_{j,k}}, \quad \forall i,k. \tag{25}$$

Finally we fix Λ and optimize over W, solving separately for each row $\boldsymbol{w}^j$ via

$$\min_{\boldsymbol{w}^j \in \mathcal{S}} \sum_{i,k} w_{j,k}\left(\frac{x_{i,j}^2 + z_{i,j}}{\lambda_{i,k}}\right) + w_{j,k}\log\lambda_{i,k} + \beta w_{j,k}\log w_{j,k}. \tag{26}$$

There exist many strategies to perform the requisite convex optimization over $\boldsymbol{w}^j$. Since (26) can be computed in closed form without the constraint $\sum_k w_{j,k} = 1$, we simply solve without the constraint and then normalize the resulting solution, which is a form of projected gradient method. In our experiments we found this procedure to be adequate for obtaining good results, but certainly a more precise alternative could be substituted for this step. Additionally, although these updates can be implemented such that each step is guaranteed to reduce or leave (18) unchanged, this alone is insufficient to guarantee formal convergence to a stationary point. The latter requires, for example, that the additional conditions of Zangwill's Global Convergence Theorem hold (Zangwill, 1969). However, we have not encountered any convergence issues in practice.

We refer to the aggregation of these update rules as a clustered sparse Bayesian learning (C-SBL) algorithm. The basic algorithm flow-chart/summary can be found in the supplementary file. Finally, there are only two parameters to set when using C-SBL, specifically ν and β. The former can actually be learned from the data using an update rule originally proposed in (Tipping, 2001). In contrast, for β we adopt a simple heuristic to balance this value according to problem size. For all the simulations reported in Section 6, ν was learned and β was set using this fixed rule without any additional tuning as the problem settings change.

5 ANALYSIS

Low-Noise Cost Function Behavior: We now analyze some of the properties of the underlying C-SBL cost function from (18) that make it especially suitable for the clustered sparse estimation problem. First we examine the limiting case $\nu \to 0$, mirroring some of our observations from Section 2, where we discussed connections with maximally sparse solutions. We also define spark$[\Phi]$ as the smallest number of linearly dependent columns in some matrix Φ (Donoho and Elad, 2004). In this regard we have the following:

Theorem 1. *Assume that an optimal solution X^* to (6) exists with $\|\boldsymbol{x}_j^*\|_0 < N$ and spark$[\Phi_j] = N+1$ for all j. Additionally, let Λ^*, W^* denote any global solution* of $\lim_{\nu\to 0}\inf_{\Lambda>0,W\in\mathcal{S}}\mathcal{L}\left(\Lambda,W\right)$. Then the value of (13) as $\nu \to 0$ given by $\Gamma_j^*\Phi_j\left(\Phi_j\Gamma_j^*\Phi_j^\top\right)^\dagger \boldsymbol{y}_j$, when combined across all j with $\Gamma_j^* = \left(\sum_k w_{j,k}^*(\Lambda_k^*)^\dagger\right)^\dagger$, forms a globally optimal solution to (6).

This result can be proven by adapting Theorem 4 from (Wipf et al., 2011), which applies to single task compressive sensing models. Note that the spark assumption is very mild and will be satisfied almost surely by any dictionary constructed via (5). Therefore, the C-SBL cost function clearly favors maximally sparse solutions in the low-noise regime as desired. Importantly however, while the global optimum of C-SBL may be equivalent to (6), the entire cost function landscape is not identical, and exploiting the cluster structure, and row-sparsity within clusters, can be advantageous in avoiding distracting local minima. Two important distinctions play a role in this regard.

First, the inclusion of the penalty term $\sum_{j,k}f(w_{j,k})$, by favoring solutions in clusters, naturally steers away from unpromising areas of the parameter space without correlation structure among tasks. Secondly, if we are able to determine the correct cluster structure, then there is a natural mechanism embedded in (18) to leverage the resulting row-sparsity to avoid local solutions, sometimes provably so. For example, assume for simplicity that $\Phi_j = \Phi\ \forall j$, meaning the same dictionary is used for all tasks. Also define the condition number of any matrix A as $\kappa(A) = \|A^{-1}\|_2\|A\|_2$, where $\|\cdot\|_2$ is the spectral norm.

Now assume that our measurements have been partitioned into $\bar{K} \leq L$ clusters Ω_k, where Y_{Ω_k} are the columns of Y associated with cluster k. Such a clustering could be provided by an oracle, or alternatively can be viewed as an intermediate point during the optimization process whereby for every task $j \in \Omega_k$, $\Gamma_j = \Lambda_k$ for some unique Λ_k. We may then consider the remaining multi-task sparse estimation problems to estimate the corresponding maximally row-sparse $X_{\Omega_k}^*$ within each cluster, holding the cluster assignments fixed, similar to problem (3).

In this scenario, Jensen's inequality collapses to an equality, the C-SBL cost function (18) decouples, and we may equivalently consider each cluster k as a separate subproblem to minimize

$$\mathcal{L}_k(\Lambda_k) \triangleq \text{tr}\left[Y_{\Omega_k}Y_{\Omega_k}^\top\left(\Sigma_k\right)^{-1}\right] + |\Omega_k|\log|\Sigma_k|, \qquad (27)$$

where $\Sigma_k \triangleq \nu I + \Phi\Lambda_k\Phi^\top$. Then we have the following:

Theorem 2. Let spark$(\Phi) = N+1$. Also, let $X_{\Omega_k}^*$ be a maximally row-sparse feasible solution to $Y_{\Omega_k} = \Phi X_{\Omega_k}$ with $D \triangleq \|X_{\Omega_k}^*\|_{row-\ell_0}$. Define $\bar{X}_{\Omega_k}^*$ as the associated collection of nonzero rows. Then if $X_{\Omega_k}^*$ satisfies

$$\inf_{\Psi>0}\kappa\left(\Psi\bar{X}_{\Omega_k}^*(\bar{X}_{\Omega_k}^*)^\top\Psi\right) < \frac{N}{D} \qquad (28)$$

with $\Psi \in \mathbb{R}^{D\times D}$ diagonal, then $\lim_{\nu\to 0}\inf_{\Lambda_k>0}\mathcal{L}_k(\Lambda_k)$ has a unique stationary point Λ_k^*, and this point will satisfy $\Lambda_k^*\Phi^\top\left(\Phi\Lambda_k^*\Phi^\top\right)^\dagger Y_{\Omega_k} = X_{\Omega_k}^*$.

The supplementary file contains details of the proof. Theorem 2 dictates circumstances under which we are guaranteed to recover the maximally row-sparse solution within each cluster (assuming we are given an algorithm that converges to a stationary point), meaning we are guaranteed to solve (18) without resorting to brute-force optimization of the more challenging NP-hard problem (6). Moreover, the most relevant criteria under which this occurs depends only on the conditioning of the nonzero rows in $X_{\Omega_k}^*$. In words, if these rows contain complementary information regarding the true sparsity profile, as evidenced by a high condition number, no locally minimizing solutions exist. A weaker related result has already been known in the information theory community, but this result adapted to the present context would require that rows of $\bar{X}_{\Omega_k}^*$ be strictly orthogonal (Kim et al., 2012). Additionally, Theorem 2 is independent of any RIP conditions or other strong structural assumptions on Φ typical of compressive sensing recovery results.

Note that arguably the most common strategy for promoting row-sparse solutions is to solve problems of the form

$$\min_{X_{\Omega_k}}\sum_j h\left(\|\boldsymbol{x}_{\Omega_k}^j\|_2\right) \quad \text{s.t.} Y_{\Omega_k} = \Phi X_{\Omega_k}, \qquad (29)$$

where h is an arbitrary non-decreasing function, and $\boldsymbol{x}_{\Omega_k}^j$ denotes the j-th row of X_{Ω_k}. Interestingly though, specialized counter-examples can be used to show that, for any such h (including the selection $h(z) = z$ that leads to the convex $\ell_{1,2}$ mixed-norm (Obozinski et al., 2011) commonly used in compressive sensing), there will always exist a Φ and Y_{Ω_k}, consistent with the stipulations of Theorem 2 such that there is guaranteed to be a stationary point not equal to $X_{\Omega_k}^*$ when solving (29). Hence the C-SBL cost function maintains an inherent advantage at the cluster level from an optimization standpoint.

High-Noise Cost Function Behavior: Now we briefly consider the scenario where ν becomes large. We first observe that the data dependent term in (18) tends towards $\sum_j\|\boldsymbol{y}_j\|_2^2/\nu + O\left(\nu^{-1}\right)$ as ν increases. Likewise the remaining ν-dependent penalty term converges as $\log\left|\Gamma_j^{-1} + (1/\nu)\Phi_j^\top\Phi_j\right| \to \log\left|\sum_k w_{j,k}\Lambda_k^{-1}\right| + O\left(\nu^{-1}\right)$. By Jensen's inequality, the resulting combined factor

$$\sum_{j,k} w_{j,k}\log|\Lambda_k| + \sum_j \log\left|\sum_k w_{j,k}\Lambda_k^{-1}\right| \qquad (30)$$

has a minimal value of zero when either $\boldsymbol{w}^j$ equals an indicator vector for all j, or when Λ_k equals some Λ' for all k. The former scenario will cause the weight penalty

$\sum_{j,k} f(w_{j,k})$ to become large. However, if all $\Lambda_k = \Lambda'$, then all of these penalty factors can effectively be minimized. Assuming the contribution from $O\left(\nu^{-1}\right)$ terms is small, this will then minimize the overall objective function. Moreover, with all $\Lambda_k = \Lambda'$, we by definition collapse to a single cluster, multi-task sparse estimation model as was motivated in Section 2 at low SNR.

6 EXPERIMENTS

This section provides empirical validation for the proposed C-SBL algorithm. We compare performance against the traditional convex ℓ_1 penalized regression estimator commonly using in compressive sensing, as well as three related sparse Bayesian algorithms that have previously been applied to similar problems. These include the original sparse Bayesian learning (SBL) (Tipping, 2001), a multiple measurement vector (MMV) extension of SBL (Ji et al., 2009), and the Dirichlet Process (DP) prior adaptation of multi-task Bayesian compressive sensing (Qi et al., 2008). The latter is arguably the closest competitor to C-SBL given its ability to learn clusters with sparse support. An additional sparse Bayesian algorithm from (Zhang & Rao, 2011) also addresses a multi-task sparse learning setting based upon related variational principles; however, this method cannot learn clusters, our central purpose, nor does code appear to be available for handling different sensing matrices Φ_j for different tasks. Therefore we do not include comparisons here. We will begin with synthetic data simulations to demonstrate model properties followed by efforts to reconstruct image sequences from compressive measurements.

Synthetic Data: For the first experiment we generate data from $\bar{K} = 5$ clusters. Within every cluster are 5 tasks each for a total of $L = 25$ tasks. Each corresponding $X^*_{\Omega_k}$ is generated with a random row-sparsity pattern distinct from one another, and with nonzero rows distributed as $\bar{X}^*_{\Omega_k} \sim \mathcal{N}(0,1)$ (i.e., each task has its own independent nonzero coefficients. The associated task-specific dictionaries are generated via $\Phi_j \sim \mathcal{N}(0, 1/N)$; we set $M = 256$, $D \triangleq \|X^*_{\Omega_k}\|_{row-\ell_0} = 30$, and the number of measurements per task N is varied. We then compute $\boldsymbol{y}_j = \Phi_j \boldsymbol{x}^*_j \; \forall j$ in each instance and run the respective algorithms to compare the recovery performance, averaging across 50 trials.

Results are presented in Figure 1(a), where we display the normalized mean-squared error metric given by $< \|\hat{X} - X^*\|_2^2 / \|X^*\|_2^2 >$. We observe that C-SBL has the lowest reconstruction error among all the methods. Additionally based on Lemma 1, X^* will almost surely be the globally optimal solution to (6). While it has been proven that regular SBL also has the same global optimum to (6) (Wipf et al., 2011), this algorithm is blind to any structure between tasks and therefore may become trapped at suboptimal local minima, leading to relatively poorer performance. On the other hand, C-SBL is more likely to reach the global optima by exploiting cluster information. Interestingly, the DP algorithm, which also putatively leverages these clusters, does not perform significatively better than SBL, suggesting that it is non-trivial to optimally use the additional structure.

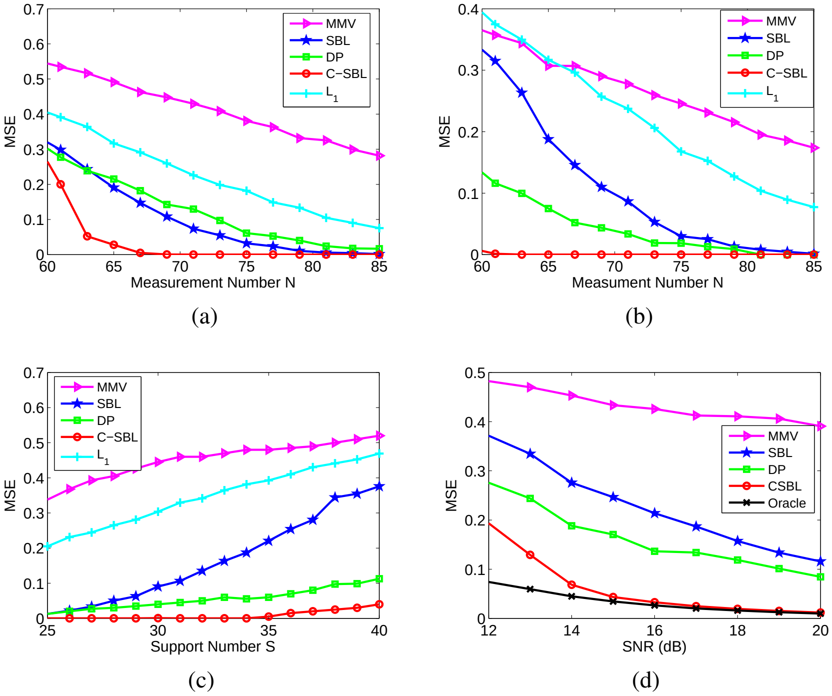
In contrast, with a different data generation mechanism, DP has demonstrated improvement over SBL but not C-SBL. Here we recreate a close approximation to experiments conducted in (Qi et al., 2008).[3] We begin with $D = 27$ nonzero rows but with both amplitudes and supports shared across tasks. An innovations component is then added, whereby an additional 3 elements of each task are given random nonzero values, with task-specific, randomly generated support. Figures 1(b) and 1(c) display results as different parameters are varied. Indeed in this revised scenario DP does significantly outperform SBL; however, C-SBL retains its advantage over all algorithms.

Finally we consider reconstructions in the presence of noise. For this purpose we generate data in the same manner as was used to generate Figure 1(a), and fix $N = 75$, $M = 256$, and $D = 30$ while varying the SNR using additive Gaussian white noise. We also include an ideal oracle estimator that knows the true clusters. Results are displayed in Figure 1(d), where again C-SBL is observed to perform well, and in excess of $15dB$ SNR nearly matches even the oracle.

Image Clustering and Reconstruction: Here we consider a real-world application motivated in (Qi et al., 2008) that involves simultaneously reconstructing multiple images from different dynamic scenes using compressive measurements. In this scenario, tasks are images and each cluster represents a group of snapshots taken from a given dynamic scene that are likely to have a similar sparsity profile in the wavelet domain. Moreover, we may expect to have different cluster sizes and noise levels across snapshots, and moving objects behave like the innovations applied in producing Figures 1(b) and 1(c).

For this experiment we choose 5 dynamic scenes (5 clusters), each having $\{5, 3, 3, 4, 4\}$ tasks respectively. Images have a resolution of 64×64, although the supplementary file contains higher resolution examples. Data are sampled using the 'db4' 2D wavelet transform using 4 scales as provided by Matlab. Each Gaussian sensing matrix Φ_j is $N = 2275 \times M = 5986$, with iid elements generated as before. Although undoubtedly better performance could be obtained by selecting different transforms and/or applying different sampling rates to different scales, this is not our primary focus here. Overall we are merely adopting an established benchmark and inserting C-SBL into this pipeline

[3] Note that certain simulation specifics needed to exactly reproduce the results from (Qi et al., 2008) were missing (e.g. SNR), and we were unfortunately unable to obtain code from the authors.



(a)

(b)

(c)

(d)

Figure 1: Synthetic data reconstruction performance comparisons; in all cases $M = 256$. (a) MSE versus N, with $D = 30$. Tasks belonging to the same cluster share the same support; nonzero coefficients are independent. (b) MSE versus N. Tasks belonging to the same cluster share the same support and coefficients over $D = 27$ nonzero rows. Each task then has an additional 3 randomly positioned, independent nonzero elements (innovations). (c) Same as (b), only now $N = 70$ and the total support cardinality $S \triangleq D + 3$ is varied. (d) MSE versus SNR. Data generated as in Figure 1(b), with $N = 75$, $D = 30$, and additive Gaussian white noise applied to achieve the desired SNR.

unaltered or specially tuned.

Figure 2 shows example reconstruction results of four out of five of the different scenes. For space consideration we only show a single reconstructed image frame from each scene cluster and compare three algorithms: C-SBL, D-P, and MMV. The supplementary file contains the full results and other details. Figure 3(a) shows the normalized MSE trajectory as a function of iteration number up to convergence. In terms of both MSE (Figure 3(a)) and visual inspection (Figure 2 and supplementary), C-SBL outperforms other algorithms. In terms of per-iteration computational complexity all algorithms are approximately equal, scaling quadratically in M, and linearly in N and L with the proper implementation.

Finally, Figures 3(b) and 3(c) display the beneficial hard clustering effect of C-SBL with regard to ground truth as revealed through heat-maps of the estimated cluster matrices W. Here column permutations are irrelevant as the column labels are arbitrary. By employing C-SBL, tasks within the same group (as partitioned by the ground truth in Figure 3(b)) return nonzeros along the same columns of the estimated W (Figure 3(c)). In this way, C-SBL uses multiple bases Λ_k to model the clusters (different scenes in Figure 2) as evidenced by multiple nonzeros in the rows of W. However, this is the artifact of many different Λ_k fusing together within a true cluster, and all of these bases within a cluster must eventually share the same support (and typically magnitudes as well) by virtue of the support intersection property described in Section 3. Consequently, we can infer that C-SBL correctly learns the correct five clusters ultimately leading to the best performance (see supplementary file for DP clustering results, which fail to mirror the ground truth).

7 CONCLUSION

In this paper we have derived a novel Bayesian model and attendant analyses for solving multi-task sparse linear inverse problems by exploiting unknown cluster structure among the tasks. Although Bayesian models have been deployed for solving related problems, these often involve organizing postulated distributional assumptions into a complex hierarchy such that approximate inference techniques must be applied that are difficult to unpack and rationalize. In contrast, herein we rely only on a simple empirical prior and then justify this parameterization using rigorous properties of the underlying cost function that emerges. This ‘semi-Bayesian’ strategy promotes understanding of the central mechanisms at work in producing a successful algorithm, including all approximations involved, and potentially suggests targeted enhancements.

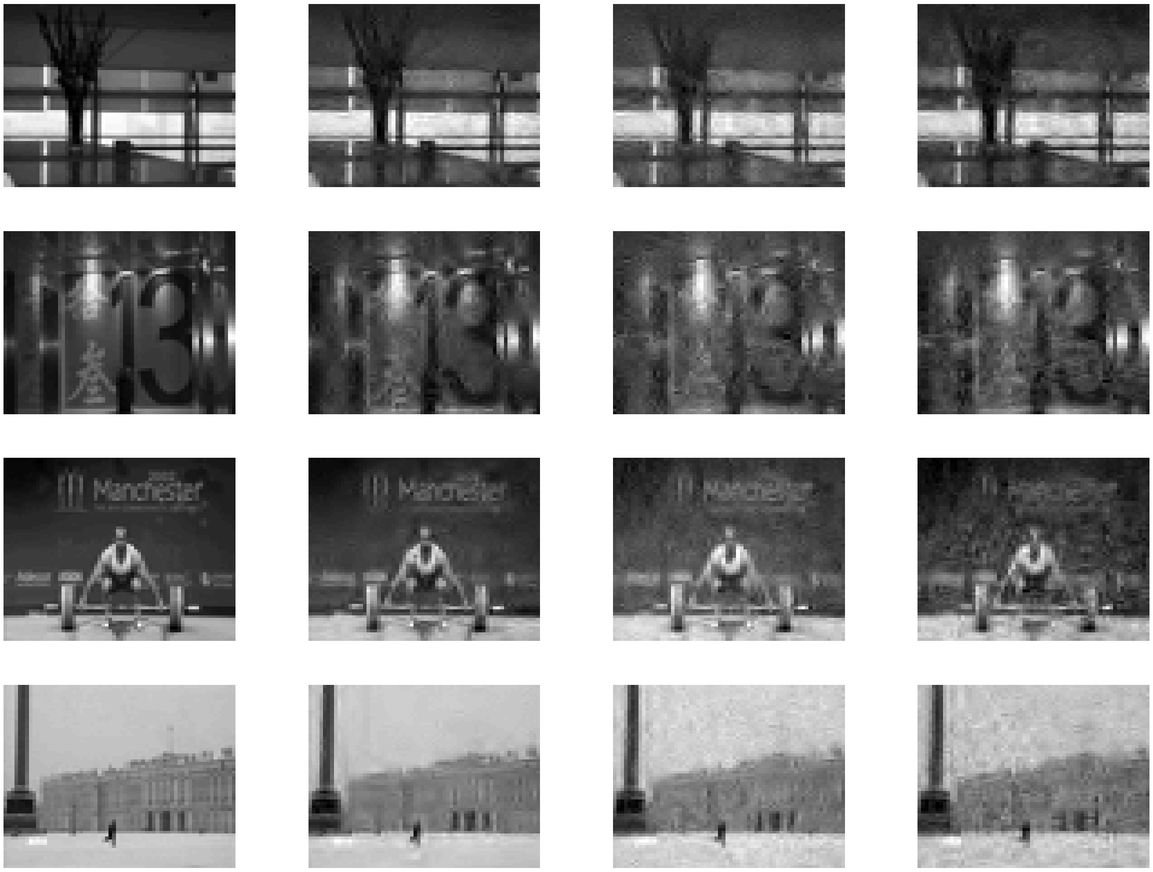
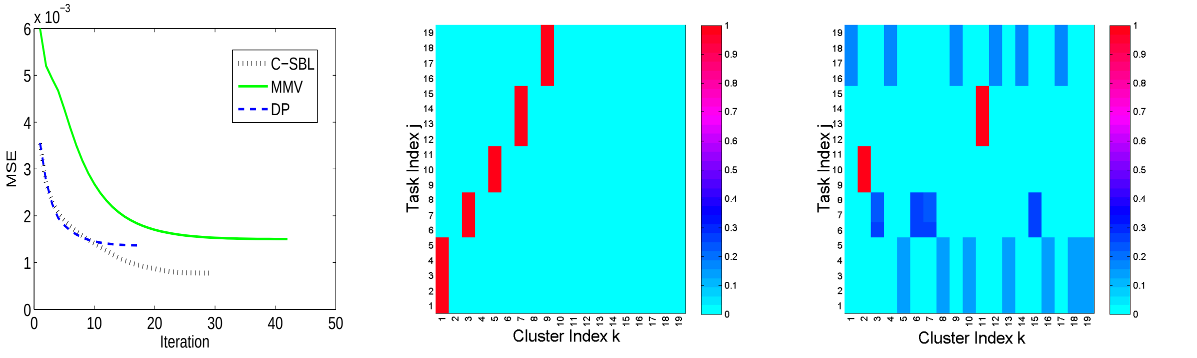


Figure 2: Reconstructions of 64×64 images from four of the five dynamic scenes (the fifth scene would not fit owing to space considerations, but is contained in the supplementary file). From left to right: Original image, C-SBL, DP, MMV. Sampling rate is $N/M = 0.38$. See supplementary file for full data, higher resolution, lower sampling rate examples.



(a) MSE Trajectories

(b) Ground Truth Cluster Pattern of W

(c) Estimated Weight Matrix W by C-SBL

Figure 3: (a) MSE versus iteration for image reconstruction. (b) Ground truth cluster patterns (c) Estimated clustering W matrix by C-SBL.

References

C. Archambeau, S. Guo, and O. Zoeter (2011), Sparse Bayesian multi-task learning. *Advances in Neural Information Processing Systems*, 1755-1763, Dec.

D. Baron, M. F. Duarte, and M. B. Wakin (2009). Distributed compressive sensing. *arXiv:0901.3403v1.4729v2*.

D. M. Blei, and M. I. Jordan (2006). Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1), Mar.

S. Boyd and L. Vandenberghe (2004). *Convex optimization*. Cambridge University Press, New York.

V. Cevher, C. Hegde, M. F. Duarte, and R. G. Baraniuk (2008). Sparse signal recovery using markov random fields. *Advances in Neural Information Processing Systems*, 257-264, Dec.

D. L. Donoho, and M. Elad (2003). Optimally sparse representation in general (non-orthogonal) dictionaries via ℓ_1 minimization. *Proceedings of The National Academy of Sciences of the United States of America*, 100(5), 2197-2202, Mar.

E. Elhamifar, and R. Vidal (2013). Sparse subspace clustering: algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11), 2765-2780, Nov

D. Hernández-lobato, and J. M. Hernández-Lobato (2013). Learning feature selection dependencies in multi-task learning. *Advances in Neural Information Processing Systems*, 746-754, Dec.

Z. Hong, X. Mei, D. Prokhorov, and D. Tao (2013). Tracking via robust multi-task multi-view joint sparse representation. *Computer Vision, 2013 IEEE International Conference on*, 649-656, Dec.

A. Jalali, P. Ravikumar, and S. Sanghavi (2013). A dirty model for multiple sparse regression *IEEE Trans. Information Theory*, 59(12), 7947-7968, Dec.

S. Ji, D. Dunson, and L. Carin (2009). Multi-task compressive sensing. *IEEE Trans. Signal Processing*, 57(1), 92-106, Jan.

J. Kim, O. Lee, and J. Ye (2012). Compressive music: revisiting the link between compressive sensing and array signal processing. *IEEE Trans. Information Theory*, 58(1), 278-301, Jan.

G. Obozinski, M. J. Wainwright, and M. I. Jordan (2011). Support union recovery in high-dimensional multivariate regression *The Annals of Statistics*, 39(1), 1-47, Feb.

Y. Qi, D. Liu, D. Dunson, and L. Carin (2008). Multi-task compressive sensing with Dirichlet process priors. *Proceedings of the 25th International Conference on Machine Learning*, 768-775, July.

N. Rao, C. Cox, R. Nowak, and T. Rogers (2013). Sparse overlapping sets lasso for multitask learning and its application to fMRI analysis. *Advances in Neural Information Processing Systems*, 2202-2210, Dec.

T. Shi, D. Tang, L. Xu, and T. Moscibroda (2014). Correlated compressive sensing for networked data. *Conference on Uncertainty in Artificial Intelligence*, July.

M. Soltanolkotabi, and E. J. Candès (2012). A geometric analysis of subspace clustering with outliers. *The Annals of Statistics* , 40(4), 2195-2238, July.

M. Tipping (2001). Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1, 211-244, June.

J. Tropp (2006). Just relax: convex programming methods for identifying sparse signals. *IEEE Trans. Information Theory*, 52(3), 1030-1051, March.

J. Wan, Z. Zhang, J. Yan, T. Li, B. Rao, S. Fang, S. Kim, S. Risacher, A. Saykin, and L. Shen (2012). Sparse Bayesian multi-task learning for predicting cognitive outcomes from neuroimaging measures in alzheimer's disease. *Computer Vision and Pattern Recognition, 2012 IEEE Conference on*, 940-947, June.

D. Wipf and S. Nagarajan (2010). Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions. *Journal of Selected Topics in Signal Processing (Special Issue on Compressive Sensing)*, 4(2), April.

D. Wipf, B. Rao, and S. Nagarajan (2011). Latent variable Bayesian models for promoting sparsity. *IEEE Trans. Information Theory*, 57(9), Sept.

E. Yang, and P. D. Ravikumar (2013). Dirty statistical models. *Advances in Neural Information Processing Systems*, 611-619, Dec.

W. Zangwill (1969). *Nonlinear programming: A unified approach*. Prentice Hall, New Jersey.

Z. Zhang and B. Rao (2011). Sparse signal recovery with temporally correlated source vectors using sparse Bayesian learning. *IEEE Journal of Selected Topics in Signal Processing*, 5(5), 912-926, Nov.