COLLABORATIVE HIERARCHICAL SPARSE MODELING

Pablo Sprechmann, Ignacio Ramirez and Guillermo Sapiro — University of Minnesota

Yonina Eldar — Technion I. I. T.

ABSTRACT

Sparse modeling is a powerful framework for data analysis and processing. Traditionally, encoding in this framework is done by solving an ℓ_1-regularized linear regression problem, usually called *Lasso*. In this work we first combine the sparsity-inducing property of the Lasso model, at the individual feature level, with the block-sparsity property of the *group Lasso* model, where sparse groups of features are jointly encoded, obtaining a sparsity pattern hierarchically structured. This results in the *hierarchical Lasso*, which shows important practical modeling advantages. We then extend this approach to the collaborative case, where a set of simultaneously coded signals share the same sparsity pattern at the higher (group) level but not necessarily at the lower one. Signals then share the same active groups, or classes, but not necessarily the same active set. This is very well suited for applications such as source separation. An efficient optimization procedure, which guarantees convergence to the global optimum, is developed for these new models. The underlying presentation of the new framework and optimization approach is complemented with experimental examples and preliminary theoretical results.

1. INTRODUCTION AND MOTIVATION

In addition to being very attractive at the theoretical level, sparse signal modeling has been shown to lead to numerous state-of-the-art results in signal processing. The standard model assumes that a signal can be efficiently represented by a sparse linear combination of atoms from a given or learned dictionary. The selected atoms form what is usually referred to as the *active set*, whose cardinality is significantly smaller than the size of the dictionary and the dimension of the signal. In recent years, it has been shown that adding structural constraints to this active set has value both at the level of representation robustness and at the level of signal interpretation (in particular when the active set indicates some physical properties of the signal), see [1] and references therein. This leads to *group* or *structured* sparse coding, where instead of considering the atoms as singletons, the atoms are grouped, and a few groups are active at a time. An alternative way to add structure (and robustness) to the problem is to consider the simultaneous encoding of multiple signals, requesting that they all share the same active set. This is a natural collaborative filtering approach to sparse coding, see [2] and references therein.

In this work we extend these models in a number of directions. First, we present a hierarchical sparse model, where not only a few (sparse) groups of atoms are active at a time, but also each group enjoys internal sparsity.[1] At the conceptual level, this means that the signal is represented by a few groups (classes), and inside each group only a few members are active at a time. A simple example of this is a piece of music (numerous applications in genomics), where only a few instruments are active at a time (each instrument is a group), and the actual music played by the instrument is efficiently represented by a few atoms of the sub-dictionary/group corresponding to it. Thereby, this proposed hierarchical sparse coding framework permits to efficiently perform source separation, where the individual sources (classes/groups) that generated the signal are identified at the same time as their efficient representation is reconstructed (the sparse code inside the group). An efficient optimization procedure is proposed to solve this hierarchical sparse coding framework.

Then, we go a step beyond this. Imagine now that we have multiple recordings of the same two instruments (or different time windows of the same recording), each time playing different songs. Then, if we apply this new hierarchical sparse coding approach collaboratively, we expect that the different recordings will share the same groups (since they are of the same instruments), but each will have its unique sparsity pattern inside the group (since each recording is a different melody). We propose a collaborative hierarchical sparse coding framework addressing exactly this.[2] An efficient optimization procedure for this case is derived as well.

In the remainder of this paper, we introduce these new models and their corresponding optimization, present examples illustrating them, and provide possible directions of research opened by these new frameworks, including some theoretical ones.

2. COLLABORATIVE HIERARCHICAL CODING

2.1. Background: Lasso and group Lasso

Assume we have a set of data samples $\mathbf{x}_j \in \mathbb{R}^m, j = 1, \dots, n$, and a dictionary of p atoms, assembled as a matrix $\mathbf{D} \in \mathbb{R}^{m \times p}$, $\mathbf{D} = [\mathbf{d}_1 \mathbf{d}_2 \dots \mathbf{d}_p]$. Each sample $\mathbf{x}_j$ can be written as $\mathbf{x}_j = \mathbf{D}\mathbf{a}_j + \epsilon$, $\mathbf{a}_j \in \mathbb{R}^p$, $\epsilon \in \mathbb{R}^m$, that is, as a linear combination of the atoms in the dictionary $\mathbf{D}$ plus some perturbation ϵ. The

IR and PS contributed equally to this work.

[1] While we here consider only 2 levels of sparsity, the proposed framework is easily extended to multiple levels.

[2] Note that different recordings can also have different instruments, so some of them will share the same groups while not necessarily all of them will be exactly the same.

978-1-4244-7417-2/10/$26.00 ©2010 IEEE

basic underlying assumption in sparse coding is that, for all or most j, the optimal reconstruction $\mathbf{a}_j$ has only a few nonzero elements. Formally, if we define the cost ℓ_0 as the pseudo-norm counting the number of nonzero elements of $\mathbf{a}_j$, $\|\mathbf{a}_j\|_0 := |\{k : a_{kj} \neq 0\}|$, we expect that $\|\mathbf{a}_j\|_0 \ll p$ for all or most j. The ℓ_0 optimization is non-convex and known to be NP-hard, so a convex approximation to it is considered instead, which uses the ℓ_1 norm cost,

$$\min_{\mathbf{a}} \|\mathbf{a}\|_1 \quad \text{s.t.} \quad \|\mathbf{x}_j - \mathbf{D}\mathbf{a}\|_2^2 \leq \epsilon. \tag{2.1}$$

The above approximation is known as the Lasso [3]. A popular variant is to use the unconstrained version

$$\min_{\mathbf{a}} \frac{1}{2} \|\mathbf{x}_j - \mathbf{D}\mathbf{a}\|_2^2 + \lambda \|\mathbf{a}\|_1, \tag{2.2}$$

where λ is a parameter usually found by cross-validation.

The $\|\cdot\|_1$ regularizer induces sparsity in the solution $\mathbf{a}_j$. This is desirable not only from a regularization point of view, but also from a model selection point, where one wants to identify the relevant features or factors (atoms) that conform each sample $\mathbf{x}_j$. In many situations, however, one wants to represent the relevant factors not as single atoms but as groups of atoms. Given a dictionary of p atoms, we define groups through their indexes, $g \subseteq \{1, \ldots, p\}$. Given a group g, we define the subset of atoms of $\mathbf{D}$ belonging to it as $\mathbf{D}_g$, and the corresponding set of linear reconstruction coefficients as $\mathbf{a}_g$. Define $\mathcal{G} = \{g_1, \ldots, g_{|\mathcal{G}|}\}$ to be a partition of $\{1, \ldots, p\}$. The group Lasso problem was introduced in [4] as

$$\min_{\mathbf{a}} \frac{1}{2} \|\mathbf{x}_j - \mathbf{D}\mathbf{a}\|_2^2 + \lambda \psi_{\mathcal{G}}(\mathbf{a}), \tag{2.3}$$

where $\psi_{\mathcal{G}}$ is the group Lasso regularizer defined in terms of $\mathcal{G}$ as $\psi_{\mathcal{G}}(\mathbf{a}) = \sum_{g \in \mathcal{G}} \|\mathbf{a}_g\|_2$. Note that $\psi_{\mathcal{G}}$ can be seen as an ℓ_1 on Euclidean norms of the vectors formed by coefficients belonging to the same group $\mathbf{a}_g$. This is a generalization of the ℓ_1 regularizer, as the latter arises from the special case $\mathcal{G} = \{1, 2, \ldots, p\}$, and, as such its effect on the groups of $\mathbf{a}$ is also a natural generalization of the one obtained with the Lasso: it "turns on" or "off" atoms in groups.

2.2. The Hierarchical Lasso

The group Lasso trades sparsity at the single-coefficient level with sparsity at a group level, while, inside each group, the solution is dense (actually it reduces to a least squares within the group). As we are interested in maintaining the sparsity at the coefficient level, we simply re-introduce the ℓ_1 regularizer together with the group regularizer, leading to the proposed *Hierarchical Lasso (HiLasso)* model,[3]

$$\min_{\mathbf{a}} \frac{1}{2} \|\mathbf{x}_j - \mathbf{D}\mathbf{a}\|_2^2 + \lambda_2 \psi_{\mathcal{G}}(\mathbf{a}) + \lambda_1 \|\mathbf{a}\|_1. \tag{2.4}$$

We refer to this regularizer as the $\ell_2 + \ell_1$.[4] In Section 3 we propose an efficient optimization for (2.4), while in Section 4 we experimentally show the virtues of this model.

[3]While preparing the camera ready version of this work we leaned of a simultaneously developed paper, [5], that also proposed this model, with a different optimization approach. The collaborative framework presented next is not developed in [5]. See also [6].

[4]We can similarly define a hierarchical sparsity model based on ℓ_0.

2.3. Collaborative Hierarchical Lasso

In numerous applications, one expects that certain collections of samples $\mathbf{x}_j$ share the same active components from the dictionary, that is, that the indexes of the nonzero coefficients in $\mathbf{a}_j$ are the same for all the samples in the collection. Imposing such dependency in the ℓ_1 regularized regression problem gives rise to the so called collaborative (also called "multitask" or "simultaneous") sparse coding problem [2, 7].

More specifically, if we consider the matrix of coefficients $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_n]$ associated to the reconstruction of the samples $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$, the collaborative sparse coding model is given by

$$\min_{a} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \sum_{k=1}^{p} \left\|\mathbf{a}^k\right\|_2, \tag{2.5}$$

where $\mathbf{a}^k$ is the k-th row of $\mathbf{A}$, that is, the vector of the n different values that the coefficient associated to the k-th atom takes for each sample j. If we now extend this idea to the group Lasso, we obtain a collaborative group Lasso formulation,

$$\min_{a} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \psi_{\mathcal{G}}(\mathbf{A}), \tag{2.6}$$

where the regularizer $\psi_{\mathcal{G}}$ for a matrix is defined as $\psi_{\mathcal{G}}(\mathbf{A}) = \sum_{g \in \mathcal{G}} \|\mathbf{A}_g\|_F$, being $\mathbf{A}_g$ the submatix formed by all the rows belonging to group g.[5] We chose this notations since this regularizer is the natural extension of the regularizer in (2.3) for the collaborative case.

To the best of our knowledge, this combination has not yet been investigated in the literature. In this paper we are moving one step forward and treat this together with the hierarchical extension. The combined model that we propose for this problem (*C-HiLasso*) can be written as follows

$$\min_{\mathbf{A}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_2 \psi_{\mathcal{G}}(\mathbf{A}) + \sum_{j=1}^{n} \lambda_1 \|\mathbf{a}_j\|_1. \tag{2.7}$$

The collaborative group Lasso is a particular case of our model when λ_1 is zero. On the other hand, one can obtain independent Lasso for each $\mathbf{x}_i$ by setting λ_2 to zero. This new formulation is particularly well suited when the vectors have missing components. In this case combining the information from all the samples is very important in order to lead to a correct representation and model (group) selection. This can be done by slightly changing the data term in (2.6). For each data vector $\mathbf{x}_j$ one computes the reconstruction error using only the observed elements. Note that the missing components do not affect the other terms of the equation.

3. OPTIMIZATION

3.1. Single-signal problem: HiLasso

In the last decade, optimization of problems of the form of (2.2) and (2.3) have been deeply studied and there exist very efficient

[5]While the introduced collaborative HiLasso model is more general, we consider the separable case for the optimization here developed.

algorithms for solving them. Recently, Wright et. al [8] proposed a framework, SpaRSA, for solving the general problem

$$\min_{\mathbf{a}} f(\mathbf{a}) + \lambda\psi(\mathbf{a}), \tag{3.8}$$

under reasonable assumptions. To guarantee convergence f needs to be a smooth and convex function while ψ only needs to be finite in $\mathbb{R}^n$. When the regularizer, ψ, is group separable, the optimization can be subdivided into smaller problems, one per group. The framework becomes powerful when these subproblems can be solved efficiently. This is the case of the Lasso and group Lasso settings but is not immediate when the regularizer is the proposed $\ell_1 + \ell_2$ norm. In this work we combine the SpaRSA with the Alternating Direction Method of Multipliers [9] (ADMOM), to efficiently solve the HiLasso problem.

The SpaRSA algorithm generates a sequence of iterates $\{\mathbf{x}^t\}_{t\in\mathbb{N}}$ that, under certain conditions, converges to the solution of (3.8). At each iteration, $\mathbf{x}^{t+1}$ is obtained solving

$$\min_{\mathbf{z}} (\mathbf{z}-\mathbf{x}^t)^T\nabla f(\mathbf{x}^t) + \frac{\alpha^t}{2}\left\|\mathbf{z}-\mathbf{x}^t\right\|_2^2 + \lambda\psi(\mathbf{z}), \tag{3.9}$$

for some sequence of parameters $\{\alpha^t\}_{t\in\mathbb{N}}$ with $\alpha^t \in \mathbb{R}^+$. The conditions for which the algorithm converges depend on the choice of α^t, see [8] for details.

It is easy to show that (3.9) is equivalent to

$$\min_{\mathbf{z}} \frac{1}{2}\left\|\mathbf{z}-\mathbf{u}^t\right\|_2^2 + \frac{\lambda}{\alpha^t}\psi(\mathbf{z}), \tag{3.10}$$

where $\mathbf{u}^t = \mathbf{x}^t - \frac{1}{\alpha^t}\nabla f(\mathbf{x}^t)$. In this new formulation, it is clear that the first term in the cost function can be separated elementwise. Thus when the regularization function $\psi(\mathbf{z})$ is group separable, so is the overall optimization, and one can solve (3.10) independently for each group,

$$\min_{\mathbf{z}_g} \frac{1}{2}\left\|\mathbf{z}_g-\mathbf{u}_g^t\right\|_2^2 + \frac{\lambda}{\alpha^t}\psi_g(\mathbf{z}_g),$$

which in the case of HiLasso, this becomes,

$$\min_{\mathbf{b}\in\mathbb{R}^{|g|}} \frac{1}{2}\left\|\mathbf{b}-\mathbf{w}\right\|_2^2 + \frac{\lambda_2}{\alpha^t}\left\|\mathbf{b}\right\|_2 + \frac{\lambda_1}{\alpha^t}\left\|\mathbf{b}\right\|_1, \tag{3.11}$$

where $\mathbf{w} = \mathbf{u}_g^t$ and $\mathbf{u}^t = \mathbf{a}^t - \frac{1}{\alpha^t}\mathbf{D}^T(\mathbf{D}\mathbf{a}^t - \mathbf{x})$. This is a SOCP for which one could use generic solvers. However, this subproblem needs to be solved many times within the SpaRSA iterations, so it is crucial to solve it efficiently. For this we use the ADMOM method [9]. The idea is to solve the artificially constrained equivalent problem,

$$\min_{b} \frac{1}{2}\left\|\mathbf{b}-\mathbf{w}\right\|_2^2 + \tilde{\lambda}_2\left\|\beta\right\|_2 + \tilde{\lambda}_1\left\|\mathbf{b}\right\|_1, \quad \text{s.t.} \quad \mathbf{b}=\beta,$$

where $\tilde{\lambda}_i = \lambda_i/\alpha^t$. The algorithm generates a set of iterates $\{\mathbf{b}^t, \beta^t, \mathbf{p}^t\}_{t\in\mathbb{N}^+}$ which converges to the minimum of the Augmented Lagrangian of the problem

$$\begin{aligned} L_c(\mathbf{b},\beta,\mathbf{p}) =& \frac{1}{2}\left\|\mathbf{b}-\mathbf{w}\right\|_2^2 + \tilde{\lambda}_2\left\|\mathbf{b}\right\|_2 + \tilde{\lambda}_1\left\|\mathbf{b}\right\|_1 \\ &+ \mathbf{p}^T(\mathbf{b}-\beta) + \frac{c}{2}\left\|\mathbf{b}-\beta\right\|_2^2, \end{aligned}$$

where the elements of $\mathbf{p}$ are the so called Lagrangian multipliers, and c is a fixed constant. At each iteration, the variables $\mathbf{b}$ and β are updated, one at a time, by minimizing the Augmented Lagrangian while letting the remaining fixed:

$$\begin{aligned} \mathbf{b}^{t+1} =& \operatorname*{argmin}_{\mathbf{b}} \frac{1}{2}\left\|\mathbf{b}-\mathbf{w}\right\|_2^2 + \tilde{\lambda}_1\left\|\mathbf{b}\right\|_1 + \mathbf{b}^T\mathbf{p} \\ &+ \frac{c}{2}\left\|\mathbf{b}-\beta\right\|_2^2, \qquad (3.12) \\ \beta^{t+1} =& \operatorname*{argmin}_{\beta} \tilde{\lambda}_2\left\|\beta\right\|_2 - \beta^T\mathbf{p} + \frac{c}{2}\left\|\mathbf{b}^{t+1}-\beta\right\|_2^2, \\ \mathbf{p}^{t+1} =& \mathbf{p} + c(\mathbf{b}^{t+1}-\beta^{t+1}). \end{aligned}$$

For convenience in the notation we omitted the super-indexes for the iterates at step t, just explicitly indexing them at step $t+1$. The update for $\mathbf{b}$ is separable into scalar subproblems on the coordinates of $\mathbf{b}$. The optimality conditions on the subgradient of each of this scalar problems leads to a simple variant of the well known soft-thresholding operator, $\mathcal{S}(w_i,\lambda) = \operatorname{sgn}(w_i)\max\{0, |w_i|-\lambda\}$. For convenience, we use the notation $\mathcal{S}(\mathbf{w},\lambda)$ to denote the vector obtained when applying the soft-thresholding operator (with parameter λ) to each element of $\mathbf{w}$. On the other hand, the update for β is not separable into scalar subproblems. However its optimality condition is given by $\beta' + \tilde{\lambda}_2\partial\left\|\beta'\right\|_2 - \mathbf{b}' \ni \mathbf{0}$, which is exactly the one leading to the vector shrinkage operator, $\mathcal{S}_v$, described in [4] for the group Lasso (actually much simpler, since there is no matrix multiplication involved):

$$\mathcal{S}_v(\mathbf{b},\tilde{\lambda}_2) = \left[1 - \frac{\tilde{\lambda}_2}{\left\|\mathbf{b}\right\|_2}\right]_+ \mathbf{b}.$$

Then both updates can be written in closed form and computed very efficiently:

$$\mathbf{b} = \frac{1}{c+1}\mathcal{S}(\mathbf{w}+c\beta-\mathbf{p},\tilde{\lambda}_1), \;\; \beta = \frac{1}{c}\mathcal{S}_v(\mathbf{p}+c\mathbf{b},\tilde{\lambda}_2).$$

The algorithm is very robust and converges in very few iterations to its optimum, thereby obtaining a very efficient approach to solve the subproblem (3.11). The SpaRSA framework then becomes a very interesting approach for the proposed HiLasso. The complete algorithm is summarized in Algorithm 1. An additional speed up is obtained by bypassing ADMOM when a whole group is not active. From the optimality conditions of (3.11) it follows that, if $\mathbf{0}$ is a solution when $\lambda_1 = 0$ (standard group Lasso), it is also a solution in the general case. This can be simply checked by evaluating $\mathcal{S}_v(\mathbf{w},\tilde{\lambda}_2) > \mathbf{0}$.

3.2. Optimization of the Collaborative HiLasso

We now propose an optimization algorithm to efficiently solve the collaborative HiLasso. The main idea is to use ADMOM to divide the overall problem into two subproblems: one that breaks the multi-signal problem into n single-signal ℓ_1 regressions, and another that treats the multi-signal case as a single group Lasso-like problem. In this way we take advantage of the separability of each term as shown in Figure 1. We define a constrained optimization problem,

$$\min \frac{1}{2}\left\|\mathbf{X}-\mathbf{D}\mathbf{A}\right\|_F^2 + \lambda_1\sum_j\left\|\mathbf{a}_j\right\|_1 + \lambda_2\psi_{\mathcal{G}}(\mathbf{B}) \;\; \text{s.t. } \mathbf{A}=\mathbf{B}.$$

Result: The optimal point $\mathbf{x}^*$
Set $t := 0$;
Choose a factor $\eta > 1$ and constants $c > 0$ and $0 < \alpha_{\min} < \alpha_{\max}$;
Choose an initial $\mathbf{x}(0) = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{|\mathcal{G}|})$;
while *stopping criterion is not satisfied* **do**
  Choose $\alpha^t \in [\alpha_{\min}, \alpha_{\max}]$;
  Set $\mathbf{u}^t \leftarrow \mathbf{x}^t - \frac{1}{\alpha^t}\nabla f(\mathbf{x}^t)$;
  while *stopping criterion is not satisfied* **do**
    % Here we use the group separability of (3.10) and
    % solve (3.11) for each group;
    for $i = 1$ *to* $|\mathcal{G}|$ **do**
      if $\mathcal{S}_v(\mathbf{w}, \tilde{\lambda}_2) > \mathbf{0}$ **then**
        Set $r := 0$;
        Choose an initial $\mathbf{p}^0, \beta^0, \mathbf{b}^0$;
        while *stopping criterion is not satisfied* **do**
          $\mathbf{b}^{r+1} = \frac{1}{c+1}\mathcal{S}(\mathbf{u_i^t} + c\beta^r - \mathbf{p^r}, \tilde{\lambda}_1)$;
          $\beta^{r+1} = \frac{1}{c}\mathcal{S}_v(\mathbf{p}^r + c\,\mathbf{b}^{r+1}, \tilde{\lambda}_2)$;
          $\mathbf{p}^{r+1} = \mathbf{p}^r + c(\mathbf{b}^{r+1} - \beta^{r+1})$;
          Set $r \leftarrow r + 1$;
        end
        Set $\mathbf{x}_g^{t+1} := \mathbf{b}^{r+1}$;
      else
        Set $\mathbf{x}_g^{t+1} := \mathbf{0}$;
      end
    end
    Set $\alpha^t \leftarrow \eta\alpha^t$;
  end
  Set $t \leftarrow t + 1$;
end

Algorithm 1: HiLasso optimization algorithm.

The ADMOM iterations are given by (we omitted the super-index for variables at iteration t for notation convenience).

$$\begin{aligned}\mathbf{A}^{t+1} =& \underset{\mathbf{A}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_1 \sum_j \|\mathbf{a}_j\|_1 + \\ & \operatorname{Tr}(\mathbf{A}^T\mathbf{P}^{t+1}) + \frac{c}{2}\|\mathbf{B} - \mathbf{A}\|_F^2, \qquad (3.13)\\ \mathbf{B}^{t+1} =& \underset{\mathbf{B}}{\operatorname{argmin}} \frac{c}{2}\left\|\mathbf{B} - \mathbf{A}^{t+1}\right\|_F^2 + \operatorname{Tr}(\mathbf{B}^T\mathbf{P}^{t+1}) \\ & + \lambda_2\psi_{\mathcal{G}}(\mathbf{B}), \qquad (3.14)\\ \mathbf{P}^{t+1} =& \mathbf{P} + c(\mathbf{A} - \mathbf{B}).\end{aligned}$$

Solving for $\mathbf{A}^{t+1}$: Problem (3.13) can be separated into n single-signal subproblems by updating one column of the matrix $\mathbf{A}$ at a time,

$$\min_{\mathbf{a}_j} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\mathbf{a}_j\|_2^2 + \mathbf{p}_j^T\mathbf{a}_j + \frac{c}{2}\|\mathbf{a}_j - \mathbf{b}\|_2^2 + \lambda_1\|\mathbf{a}_j\|_1 .$$

This problem can be solved using the SpaRSA framework. The idea is to consider the first three terms of the cost as $f(\cdot)$ in Equation (3.8). The associated computational cost is equivalent to the one of the Lasso, since the regularizer is the standard ℓ_1 norm.

Solving for $\mathbf{B}^{t+1}$: The problem given by (3.14) is group separable, as a direct consequence of the separability of $\psi_{\mathcal{G}}$. Thus, we need to solve $|\mathcal{G}|$ optimization problems of the form,

$$\min_{\mathbf{B}_g} \frac{c}{2}\left\|\mathbf{B}_g - \mathbf{A}_g^{t+1}\right\|_F^2 + \operatorname{Tr}(\mathbf{P}_g^{t+1}\mathbf{B}_g^T) + \lambda_2\|\mathbf{B}_g\|_F ,$$

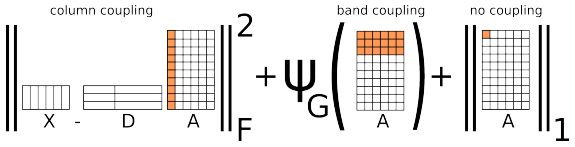

Fig. 1. Structure of the problem in terms of coupling.

where $\mathbf{A}_g$, $\mathbf{B}_g$ and $\mathbf{P}_g$ are the $|g| \times n$ sub-matrices of $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{P}$ associated with the group g respectively. We express them as column vectors (each with $|g| \times n$ components) by concatenating their columns, obtaining $\mathbf{b}_g$, β_g and $\mathbf{p}_g$ respectively, and rewrite the optimization problem in vectorial form as

$$\min_{\mathbf{b}} \lambda_2\|\mathbf{b}\|_2 - \mathbf{p}_g^T\mathbf{b} + \frac{c}{2}\left\|\mathbf{a}_g^{t+1} - \mathbf{b}\right\|_2^2 . \qquad (3.15)$$

This problem is identical to (3.12) and can be reduced to a group Lasso problem by simply changing variables and thus, it is solved using vectorial thresholding.

4. EXPERIMENTAL RESULTS

We start by comparing our model with the standard Lasso and group Lasso using synthetic data. We created $|\mathcal{G}|$ dictionaries, $\mathbf{D}_i$, with 64 atoms of dimension 64, with i.i.d. Gaussian entries. The columns were normalized to have unit ℓ_2 norm. Then we randomly chose two groups to be active at each time (on all the signals). Sets of $N = 200$ testing signals were generated, one per active group, as linear combinations of $k \ll 64$ elements of the dictionaries, $\mathbf{x}_j^i = \mathbf{D}_i\mathbf{a}_j^i$. These signals were also normalized. The mixtures were created by summing these signals and (eventually) adding gaussian noise of standard deviation σ. The generated testing signals have a hierarchical sparsity structure and while they share groups, they do not necessarily share the sparsity pattern inside the groups.

We built a single dictionary by concatenating the sub-dictionaries, $\mathbf{D} = [\mathbf{D}_1, \ldots, \mathbf{D}_{|\mathcal{G}|}]$, and use it to solve the Lasso, group Lasso, HiLasso and C-HiLasso problems. Table 1 summarizes the Mean Square Error (MSE) and Hamming distance of the recovered coefficient vectors. We observe that our model is able to exploit the hierarchical structure of the data as well as the collaborative structure. From a modeling point of view, we observe that the group Lasso selects in general the correct blocks but it does not give a sparse solution within them. On the other hand, Lasso gives a solution that has nonzero elements belonging to groups that were not active in the original signal, leading to a wrong model selection. HiLasso gives a sparse solution that picks atoms form the correct groups but still presents some minor mistakes. For the collaborative case, in all the tested cases, no coefficients were selected outside the correct active groups and the recovered coefficients are consistently the best ones. This robustness comes from the fact that the active groups are collaboratively found using the information present in all the signals. We consider the USPS digits dataset that has been shown to be well represented in the sparse modeling framework [10]. Here the signals are vectors containing the unwrapped gray intensities of 16×16 images. We chose two digits and summed them up to create a mixture image. We

σ	Lasso	Glasso	HiLasso	C-HiLasso
0.1	41.7/ 22.0	117.3 / 361.6	33.0 / 19.8	**16.3 / 13.3**
0.2	56.4 / 21.6	118.2 / 378.3	39.9 / 22.7	**24.9 /17.1**
0.4	96.5 / 22.7	137.8/ 340.3	65.6 / **19.5**	**59.5** /27.4
k	Lasso	Glasso	HiLasso	C-HiLasso
8	38.8 / 22.0	118.4 / 318.2	27.2 / 19.5	**9.6 / 16.2**
12	120.0 / 36.2	116.6 /350.4	70.4 / **26.5**	**41.3** / 29.1
16	164.1 / 43.9	109.3 / 338.6	110.0 / **32.2**	**55.1** / 35.0
$\|\mathcal{G}\|$	Lasso	Glasso	HiLasso	C-HiLasso
4	108.0 / 27.8	191.6 /221.7	100.9 / **29.8**	**74.2** / 30.2
8	120.0 / 36.2	116.6 /350.4	70.4 / **26.5**	**41.3** / 29.1

Table 1. Active sets MSE (we show them multiplied by 10^3) and Hamming distance (MSE / Hamming) for the tested methods. In the first case we vary the noise level while we keep $|\mathcal{G}| = 8$ and $k = 8$ fixed. In the two other tables the signals are noise free and we first set $|\mathcal{G}| = 8$ while changing k, and then set $k = 12$ while changing the number of groups. For each method the regularization parameters were the ones for which the best results where obtained.

created 200 random mixture images and then analyzed them with the different methods. In this case there is no ground truth active set, and we used as a measure of performance the separating error defined as $\frac{1}{NR}\sum_{i=1}^{R}\sum_{j=1}^{N}\left\|\mathbf{x}_j^i - \hat{\mathbf{x}}_j^i\right\|_2^2$, where $\mathbf{x}_j^i$ is the component corresponding to source i in the signal j, and $\hat{\mathbf{x}}_j^i$ is the recovered one.

Using the usual training-testing split for USPS we first learned a dictionary for each digit. We then created a single dictionary by concatenating them. In Figure 2 we show the separation error obtained in different situations. As in the synthetic case, only the collaborative method was able to successfully detect the true active sources. We show in Figure 2 some examples of the recovered active sets for each method.

We also used the digits dataset to experiment with missing data. We randomly discarded an average of 60% of the pixels per mixed image and then applied the C-Hilasso. The algorithm is capable of correctly detect which digits are present in the images. In Figure 3 we show some examples. Note that this is a quite different problem than the one commonly addressed in the matrix completion literature. Here we do not aim to recover signals that all belong to a unique unknown sub-space, but signals that are the combination of two non-unique spaces to be automatically selected from the available dictionary. Such unknown spaces have common models/groups for all the signals in question (the coarse level of the hierarchy), but not necessarily the exact same atoms and therefore not necessarily belong to the same sub-spaces. Both levels of the hierarchy are automatically detected, e.g., that the groups are those corresponding to "3" and "5," and the exact atoms (sub-spaces) in each group, these last ones possibly different for each signal in the set. While we consider that the possible sub-spaces are to be selected from the provided dictionary, in Section 5 we discuss learning such dictionaries as well. In such case, the standard matrix completion problem becomes a particular case of the C-HiLasso framework (with a single group and all the signals having the same active set, sub-space, in the group), naturally opening numerous theoretical questions for this new more general model.[6] Finally,

[6]Prof. Carin and collaborators have new results on the case of a single group and signals in possible different sub-spaces of the group, an intermediate model between standard matrix completion and C-HiLasso (personal communication).

Digits	Lasso	Glasso	HiLasso	C-HiLasso
3+5	74.1	80.1	68.6	**63.4**
3+5+n	87.9	95.4	92.9	**77.3**
2+7	61.1	60.8	58.7	**42.6**
2+7+n	75.4	65.2	64.7	**53.7**

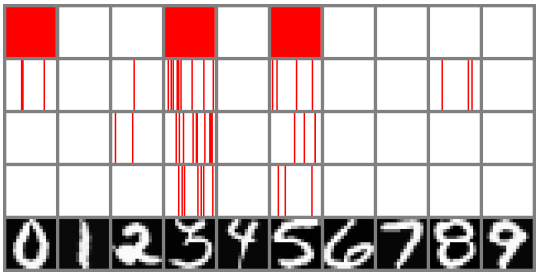

Fig. 2. . (Top) The table shows the separating errors (we show them multiplied by 10^3) for the digits dataset. We show the results for separating digits 3 and 5, and 2 and 7, with and without additive noise of standard deviation $\sigma = 0.1$. We used sets of 200 copies. (Bottom) Active sets recovered for the group Lasso, Lasso, HiLasso and C-HiLasso for a given example. Each block corresponds to the coefficients associated with the digits displayed bellow. The active coefficients are displayed in read. Only C-HiLasso manages to perfectly recover the correct models (with the lowest separating error), while HiLasso performs very well also.

we used C-HiLasso to separate overlapping textures in an image. We chose 8 textures form the Brodatz dataset and trained one dictionary for each one of them (these form the 8 groups of the dictionary). Then we created an image as the sum of two textues (the testing images were not used in the training stage). In Figure 4 we show results. The overall group Hamming distance obtained for C-HiLasso is 0.003, showing that the correct groups, and only them, were practically selected all the time.

5. DISCUSSION

In this paper we have introduced a new framework of collaborative hierarchical sparse coding, where multiple signals collaborate in their encoding, sharing code groups (models) and having (possible disjoint) sparse representations inside the corresponding groups. An efficient optimization approach was developed, which guarantees convergence to the global minimum, and examples illustrating the power of this framework were presented. At the practical level, we are currently working on the applications of this proposed framework in a number of directions, including collaborative instruments separation in music; and signal classification, following the demonstrated capability to collectively select the correct groups/models.

At the theoretical level, a whole family of new problems is opened by this proposed framework. A critical one is the overall capability of selecting the correct groups and thereby of performing correct model selection and source separation. Let us consider for example the case of only two groups (so no sparsity at the group level) and a single signal composed by the linear combination of atoms from each group. Then, it is easy to show that the cross-mutual coherence between the groups plays a critical role. Let us call μ_i, $i = 1, 2$, the internal

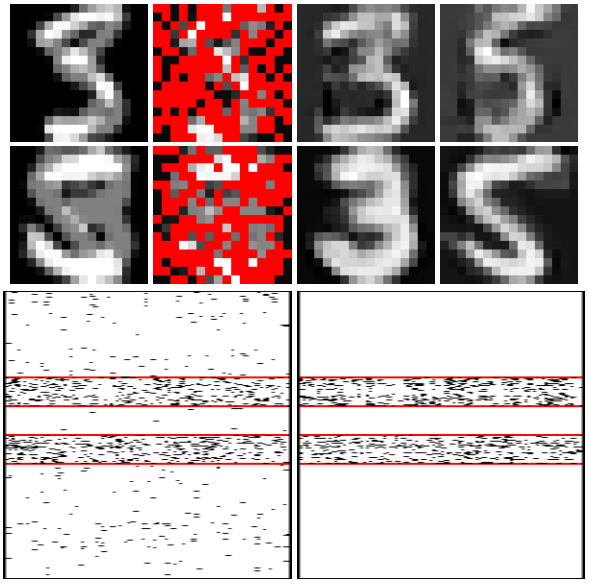

Fig. 3. . (Top) We show two examples (one per row) of the recovered digits from a mixture with 60% of missing components. We first show the original mixture image, then the image with the missing pixels highlighted in red, and finally the digits recovered. (Bottom) Here we show a comparision of the active sets recovered using the Lasso (left) and the C-HiLasso (right) methods. The active sets for the set of signals (as shown in Figure 2) are placed as columns. The coefficients corresponding to digits 3 and 5 fall inside the area delimited by the red horizontal lines. While C-HiLasso recovers the correct sources in all the cases, the Lasso method makes several mistakes.

coherence of the atoms of the group i, and $\mu_{1,2}$ the one between the groups (maximal normalized correlation between an atom of group 1 with an atom of group 2). Then it is easy to show that uniqueness of the separation can be guaranteed if $(2k_1 - 1)\mu_1 + 2k\mu_{1,2} < 1$ and $(2k_2 - 1)\mu_2 + 2k_1\mu_{1,2} < 1$, with k_i the respective sparsity levels inside each group (this is a weaker bound that the more stringent one developed by [11]).

This needs to be extended to actual sparsity at the group level and to the collaborative case. Note of course that considering a single active group is a particular case of our model (see [10] for works in this case), thereby an overall theoretical framework for our proposed collaborative hierarchical framework will automatically include numerous of the existing results in sparse coding.

Finally, we have also developed a framework for learning the dictionary for collaborative hierarchical sparse coding, meaning the optimization is simultaneously on the dictionary and the code. As it is the case with standard dictionary learning, this is expected to lead to significant performance improvements (again, see [10] for the particular case of this with a single group active at a time).

Acknowledgments: Work partially supported by NSF, ONR, NGA, and ARO. We thank Dr. Tristan Nguyen, when we presented him this model, he motivated us to think in a hierarchical fashion and to look at this as just the particular case of a fully hierarchical sparse coding framework. We also thank Prof. Larry Carin, Dr. Guoshen Yu, and Alexey Castrodad for very stimulating conversations and for the fact that their own work also motivated the example with missing information.

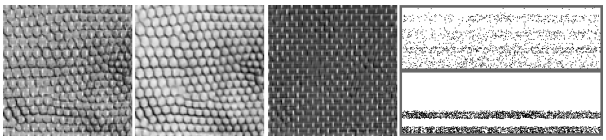

Fig. 4. . Results for the texture segmentation. One example of the mixture and the C-HiLasso separated textures are shown. This is followed by the active set diagram (as in Figure 3), Lasso on top (with class selection wrongly all over the 8 textures) and C-HiLasso on bottom, where only the 2 corresponding groups are selected.

6. REFERENCES

[1] R. Jenatton, J. Audibert, and F. Bach, "Structured variable selection with sparsity-inducing norms," Tech. Rep. arXiv:0904.3523v1, INRIA, 2009.

[2] J. Tropp, "Algorithms for simultaneous sparse approximation. part ii:convex relaxation," *Signal Processing*, vol. 86, no. 3, pp. 589–602, 2006.

[3] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[4] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society, Series B*, vol. 68, pp. 49–67, 2006.

[5] J. Friedman, T. Hastie, and R. Tibshirani, "A note on the group lasso and a sparse group lasso," preprint (2010), available at `http://www-stat.stanford.edu/ tibs`.

[6] J. Peng, J. Zhu, A. Bergamaschi, W. Han, D. Noh, J. Pollack, and P. Wang, "Regularized multivariate regression for identifying master predictors with application to integrative genomics study of breast cancer," *Annals of Applied Statistics.To appear.*

[7] B. Turlach, W. Venables, and S. Wright, "Simultaneous variable selection," *Technometrics*, vol. 27, pp. 349–363, 2004.

[8] J. Wright, R. Nowak, and M. Figueiredo, "Sparse reconstruction by separable approximation," *Trans. Sig. Proc.*, vol. 57, no. 7, pp. 2479–2493, 2009.

[9] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Comptutation: Numerical Methods*, Prentice Hall, 1989.

[10] P. Sprechmann, I. Ramirez, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence," in *CVPR*, 2010.

[11] J. Starck, M. Elad, and D. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Transactions on Image Processing*, vol. 14, pp. 1570–1582, 2004.