Learning Dictionary-Based Unions of Subspaces for Image Denoising

David Hong, Robert P. Malinas, Jeffrey A. Fessler, Laura Balzano
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI, 48105, USA
Email: {dahong, rmalinas, fessler, girasole}@umich.edu

***Abstract*—Many signals of interest are well-approximated by sparse linear combinations of atomic signals from a dictionary. Equivalently, they are well-approximated by low-dimensional subspaces in a union of subspaces generated by the dictionary. A given sparsity level has an associated union of subspaces (UoS) generated by sparse combinations of correspondingly many atoms. When considering a sequence of sparsity levels, we have a sequence of unions of subspaces (SUoS) of increasing dimension. This paper considers the problem of learning such an SUoS from data. While each UoS is combinatorially large with respect to sparsity level, our learning approach exploits the fact that sparsity is structured for many signals of interest, i.e., that certain collections of atoms are more frequently used together than others. This is known as group sparsity structure and has been studied extensively when the structure is known a priori. We consider the setting where the structure is unknown, and we seek to learn it from training data. We also adapt the subspaces we obtain to improve representation and parsimony, similar to the goal of adapting atoms in dictionary learning. We illustrate the benefits of the learned dictionary-based SUoS for the problem of denoising; using a more parsimonious and representative SUoS results in improved recovery of complicated structures and edges.**

***Index Terms*—unions of subspaces, structured sparsity, dictionary learning.**

I. Introduction

In many applications ranging from medical imaging [1] to multi-band signal processing [2] and genetics [3], to name just a few, signals of interest are often modeled as sparse linear combinations of a large (often over-complete) set of dictionary atoms. Namely, we assume our signals of interest $x \in \mathbb{C}^m$ satisfy

$$\min_{z:\|z\|_0 \leq k} \|x - Dz\|_2 \leq \varepsilon\sqrt{m} \tag{1}$$

where

- $D \in \mathbb{C}^{m\times n}$ is a *dictionary* with n unit norm columns $d_1, \ldots, d_n \in \mathbb{C}^m$ referred to as *atoms*
- k is the *sparsity level* (typically much smaller than n)
- ε is the approximation root mean square error (RMSE)

Any signal that is exactly k-sparse in the dictionary D, i.e., that satisfies (1) with $\varepsilon = 0$, lies in the span of the atoms identified by the support of its k-sparse coefficient vector z. Hence, it has often been noted that such signals lie in the union of $\binom{n}{k}$ subspaces, each of dimension k.

Since the results of [4], [5] showed that it is possible to efficiently recover these signals from only $O(k \log n)$ measurements using ℓ_1 optimization, this model has been applied widely for signal denoising and inverse problems. It has also been widely recognized that not all $\binom{n}{k}$ possible k-sparse supports are equally likely, resulting in an extensive literature on group sparsity or structured sparsity constraints for signal representation and recovery. Identifying groups corresponds exactly to selecting a subset in the union of k-dimensional subspaces (2) and hence this model is also called a *structured* union of subspaces [6], [7]. In the vast majority of research, however, the group structure is assumed known. This paper attempts to learn group structure from data.

In general, learning which of the combinatorially many possible supports are most relevant for a given dataset is challenging. Our key insight is that the lowest-dimensional models represent the bulk of the signals for some datasets. Hence, we first learn 1-sparse supports, then 2-sparse supports and so on, where at each stage we seek to represent only the data not already well approximated. The training data associated with each support can then be collected and used to learn an even lower-dimensional subspace. We can also discard subspaces associated with only a few signals; doing so further simplifies the model and increases overall representation error only slightly. Organizing the remaining subspaces by dimension yields a sequence of unions of subspaces (SUoS) of increasing dimension.

This paper proposes an algorithm based on this intuition for learning a parsimonious and representative SUoS from data. We demonstrate the benefit of the learned model over unstructured sparsity by applying it to image denoising.

II. Related Work

Hidden Markov Models for Wavelet coefficients. Motivated by the observation that wavelet coefficients are typically correlated within and across scales, [8], [9] propose learning hidden Markov models with tree-structure to capture the correlations among the coefficients then using them to improve signal estimation and classification. Capturing these correlations provides rich information about the sparse coefficients and their relationships. In contrast, our proposed approach

J. A. Fessler was supported by NIH U01 EB018753. L. Balzano was supported by DARPA grant 16-43-D3M-FP-03.

ISBN 978-90-827970-1-5 © EURASIP 2018

learns only the structure of supports, but immediately applies without extension to dictionaries without tree-structure.

Structured Sparsity and Group Lasso. Numerous applications ranging from multi-band signal processing [2] to genetics [3] motivated extensive work in the past decade on both theory and algorithms that exploit known group structure in supports to improve signal/subspace recovery from compressive measurements [6], [7], [10]–[12] and classification [13]. Example structures include non-overlapping groups [14], overlapping groups [15], [16], tree-structured groups [17], and even groups with internal sparsity [18]. This paper is largely inspired by the benefits of capturing structured sparsity that they demonstrate, but we focus instead on how to learn unknown structures from data.

Learning the structure for structured sparsity. The authors of [19] propose a statistical model for structured sparsity and an inference scheme for its hyperparameters. In contrast to [19] and the works discussed above, our proposed approach learns new subspaces that need not be generated from atoms of the dictionary and so may provide more parsimonious representations. Namely, we focus more on learning arbitrary sequences of unions of subspaces than on sparsity structure for a given dictionary. Still, we use sparsity structure to first cluster the data; incorporating ideas from [19] in that step would be an interesting direction for future work.

Subspace clustering. Subspace clustering [20] groups data using a nearest-subspace cost function and can be used to learn a union of subspaces by simply learning a subspace for each cluster. Likewise, our proposed approach clusters data then learns a subspace for each cluster. However, it generally differs from other subspace clustering approaches by exploiting structured sparsity with respect to an initial dictionary to select the number of clusters. Additionally, while many subspace clustering techniques can learn subspaces of different dimensions, an SUoS may further have higher-dimensional subspaces that entirely contain lower-dimensional subspaces. Subspace clustering techniques do not typically learn this type of structure.

Dictionary learning. Dictionary learning adapts dictionary atoms to more parsimoniously represent data [21], [22] and our proposal shares this trait. As with any collection of subspaces, one can also obtain a learned dictionary from our proposal by using the subspace basis vectors as atoms and assuming the corresponding non-overlapping group sparsity. In contrast to dictionary learning approaches, we first cluster the data and then learn subspaces for them. A notable consequence is that the (effective) number of atoms and the sparsity structure are learned from data.

III. Learning a Sequence of Unions of Subspaces

The set of all k-sparse signals in a given dictionary D forms a union of $\binom{n}{k}$ many k-dimensional subspaces, defined as:

$$\mathcal{F}_k(D) := \{Dz : z \in \mathbb{C}^n, \|z\|_0 \leq k\} = \bigcup_{\mathcal{I}\in\Omega_k} \mathcal{R}(D_\mathcal{I}) \quad (2)$$

where $\mathcal{R}(\cdot)$ gives the range of the columns of its input, $D_\mathcal{I}$ is a matrix formed from the columns of D indexed by $\mathcal{I}$, and the union is carried out over the $\binom{n}{k}$ index sets in

$$\Omega_k := \{\mathcal{I} \subset \{1,\dots,n\} : |\mathcal{I}| = k\}.$$

Taking all sparsity levels yields a sequence of unions of subspaces (SUoS) $\mathcal{F}_0(D),\dots,\mathcal{F}_n(D)$ of increasing dimension. This is distinct from the union of unions; $\mathcal{F}_0(D)\cup\cdots\cup\mathcal{F}_n(D)$ is actually just $\mathcal{F}_n(D)$ since it contains the rest.

A. Goal: Learn a "parsimonious" SUoS from data

We aim to learn a new sequence of unions of subspaces

$$\mathcal{U}_0,\mathcal{U}_1,\dots,\mathcal{U}_m \subseteq \mathbb{C}^m \quad (3)$$

that well-approximate a given collection of training vectors $x_1,\dots,x_T \in \mathbb{C}^m$. Each $\mathcal{U}_k$ is a (potentially empty) union of N_k many k-dimensional subspaces

$$\mathcal{U}_k := \bigcup_{i=1}^{N_k} \mathcal{R}(U_{k,i}) = \bigcup_{i=1}^{N_k} \{U_{k,i}z : z \in \mathbb{C}^k\} \quad (4)$$

where the columns of $U_{k,i} \in \mathbb{C}^{m\times k}$ span a k-dimensional subspace. We consider $\{0\}$ to be a zero-dimensional subspace.

Note that $\mathcal{U}_0$ must be either $\{0\}$ or $\emptyset$, and likewise $\mathcal{U}_m$ must be either $\mathbb{C}^m$ or $\emptyset$. Beyond that, however, there are infinitely many choices for each of $\mathcal{U}_1,\dots,\mathcal{U}_{m-1}$ that produce perfect representation of the training vectors. Two such choices are always:

$$\mathcal{U}_0,\mathcal{U}_2,\dots,\mathcal{U}_m = \emptyset \qquad \mathcal{U}_1 = \bigcup_{i=1}^{T} \mathcal{R}(x_i) \quad (5)$$

$$\mathcal{U}_0,\dots,\mathcal{U}_{m-1} = \emptyset \qquad \mathcal{U}_m = \mathbb{C}^m. \quad (6)$$

However, (5) is undesirable because it does not generalize from the data; only scaled training vectors appear in the SUoS. It is not parsimonious in that $\mathcal{U}_1$ contains many subspaces. (6) has only one subspace but is not parsimonious in that it does not yield low-dimensional representations of vectors.

We seek low-dimensional subspaces, where each represents nontrivially many training vectors. This requires balancing the trade-off between using low-dimensional subspaces and using subspaces expressive enough to represent diverse data vectors. Formulating this goal precisely and cleanly is challenging and ongoing work.

B. Proposal: A dictionary-based SUoS learning algorithm

We propose learning an SUoS $\mathcal{U}_0,\dots,\mathcal{U}_m$ where each subspace approximates a subset of the training vectors $x_1,\dots,x_N$ identified by their structured sparsity in a dictionary. Given a dictionary D, approximation tolerances $\varepsilon_\mathrm{s},\varepsilon_\mathrm{u}$ and a threshold number of training vectors τ, the proposed method has the following steps:

1) Sparsely approximate each training vector x_t with the dictionary D by sparse coding: for $t = 1,\dots,T$ solve

$$\hat{z}_t = \operatorname*{argmin}_{z_t\in\mathbb{C}^n} \|z_t\|_0 \quad \text{s.t.} \quad \|x_t - Dz_t\|_2 \leq \varepsilon_\mathrm{s}\sqrt{m}. \quad (7)$$

ISBN 978-90-827970-1-5 © EURASIP 2018

Under mild conditions on the dictionary [23], orthogonal matching pursuit (OMP) solves (7) efficiently and reliably. It solves (7) exactly for orthogonal atoms.

2) Cluster the training vectors by the atoms in their sparse approximation, i.e., by the supports $\mathrm{supp}(\hat{z}_t)$.
3) Discard clusters containing fewer than τ training vectors, obtaining clusters $\mathcal{X}_1, \ldots, \mathcal{X}_L \subset \{x_1, \ldots, x_T\}$.
4) Learn an orthonormal subspace basis $U_\ell \in \mathbb{C}^{m \times k_\ell}$ for each cluster $\mathcal{X}_\ell$ by minimizing the approximation error

$$\rho(U_\ell, \mathcal{X}_\ell) := \sqrt{\frac{1}{|\mathcal{X}_\ell|} \sum_{x \in \mathcal{X}_\ell} \|x - U_\ell U_\ell^T x\|_2^2} \qquad (8)$$

where k_ℓ is the smallest dimension that results in a U_ℓ within the approximation tolerance $\rho(U_\ell, \mathcal{X}_\ell) \leq \varepsilon_{\mathrm{u}} \sqrt{m}$.

We find the dimension k_ℓ and associated $U_\ell \in \mathbb{C}^{m \times k_\ell}$ via the singular value decomposition by noting that k_ℓ is the smallest value for which

$$\frac{1}{\sqrt{|\mathcal{X}_\ell|}} \sqrt{\sum_{j > k_\ell} \sigma_j^2(X_\ell)} \leq \varepsilon_{\mathrm{u}} \sqrt{m} \qquad (9)$$

where $\sigma_j(X_\ell)$ is the jth singular value of the matrix $X_\ell \in \mathbb{C}^{m \times |\mathcal{X}_\ell|}$ whose columns are the $|\mathcal{X}_\ell|$ training vectors in $\mathcal{X}_\ell$. The columns of U_ℓ are simply the first k_ℓ left singular vectors of X_ℓ [24].

5) Collect the subspace bases $U_1, \ldots, U_L$ by their dimensions $k_1, \ldots, k_L$, obtaining the unions of subspaces:

$$\mathcal{U}_k = \bigcup_{\ell : k_\ell = k} \mathcal{R}(U_\ell) \qquad k = 0, \ldots, m. \qquad (10)$$

Steps 1-3 exploit structured sparsity to form clusters of training signals that we hope lie near low-dimensional subspaces that are learned in steps 4-5. In this way, the approach combines learning sparsity structure like [19] with adaptation to the data like in dictionary learning [21].

Note that the approach automatically chooses how many subspaces of each dimension to include, and encourages a parsimonious SUoS with low-dimensional subspaces that all represent nontrivially many training vectors. Furthermore, the approach is efficient; the primary sources of computational cost are sparse coding, which is done efficiently via OMP, and the singular value decomposition of each cluster.

IV. Denoising with a general sequence of unions of subspaces

This section describes how to use an SUoS for denoising. Denoising a vector $y \in \mathbb{C}^m$ using (unstructured) sparsity can be accomplished by solving the sparse coding problem:

$$\hat{z} = \underset{z \in \mathbb{C}^n}{\mathrm{argmin}} \|z\|_0 \quad \text{s.t.} \quad \|y - Dz\|_2 \leq \varepsilon \sqrt{m} \qquad (11)$$

then returning the "denoised" vector $\hat{x} = D\hat{z}$. We propose a generalization of this scheme to arbitrary SUoS models as follows: given an SUoS $\mathcal{U}_0, \ldots, \mathcal{U}_m$ solve the minimization

$$\hat{k} = \min_k k \quad \text{s.t.} \quad \min_{x \in \mathcal{U}_k} \|x - y\|_2 \leq \varepsilon \sqrt{m} \qquad (12)$$

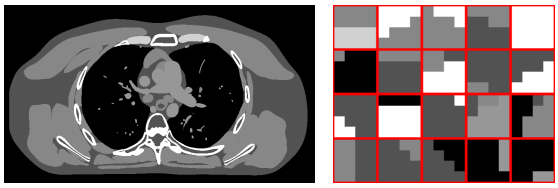

Fig. 1: Training slice (475×835) of the XCAT digital phantom [25], [26] and a set of randomly selected 4×4 patches. The display window for both is $[900, 1100]$ HU.

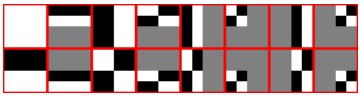

Fig. 2: Atoms of the 2D orthogonal Haar wavelet dictionary.

to select a dimension then return the denoised vector

$$\hat{x} = \underset{x \in \mathcal{U}_{\hat{k}}}{\mathrm{argmin}} \|x - y\|_2. \qquad (13)$$

In essence, the scheme is to project y onto the lowest dimensional subspace that approximates it with RMSE within ε. For the dictionary-generated SUoS $\mathcal{F}_0(D), \ldots, \mathcal{F}_n(D)$, it is precisely a restatement of the sparsity approach; the subspace dimension k in (12) corresponds to the sparsity $\|z\|_0$ in (11).

Thus we have the following simple procedure for denoising with a general SUoS:

1) Initialize $k = 0$.
2) Select the subspace basis U among those in $\mathcal{U}_k$ that maximizes the projection length $\|U^T y\|_2 = \|UU^T y\|_2$.
3) If $\|y - UU^T y\|_2 \leq \varepsilon \sqrt{m}$, return $UU^T y$. Otherwise, increment k and go back to step 2.

The exhaustive search in step 2 may appear worrisome, but for parsimonious SUoS we hope to have relatively few subspaces in each union. Moreover, we hope that most signals are close to low-dimensional subspaces and can exit early in the algorithm.

Varying ε trades off between model error and noise; larger choices allow approximation by lower-dimensional subspaces that further suppress noise but that are also less likely to be representative. Adapting the dimension like this is desirable for diverse signal classes such as image patches where some are nearly constant while others may be highly textured.

V. Experiments on an X-ray CT digital phantom

This section illustrates learning an SUoS for patches of an axial slice of the XCAT digital phantom [25], [26] then using it for denoising.

A. Learning an SUoS

We learn an SUoS for 4×4 patches extracted from a 475×835 slice of the XCAT phantom, shown in Figure 1 with a display window of 900 to 1100 modified Hounsfield units (HU). Extracting all overlapping 4×4 patches yields $T = 392704$ training samples in $\mathbb{R}^{16}$, 53444 of which are not constant. We use 2D orthogonal Haar wavelets (Figure 2) as the input dictionary $D \in \mathbb{R}^{16 \times 16}$ to match the piecewise

ISBN 978-90-827970-1-5 © EURASIP 2018

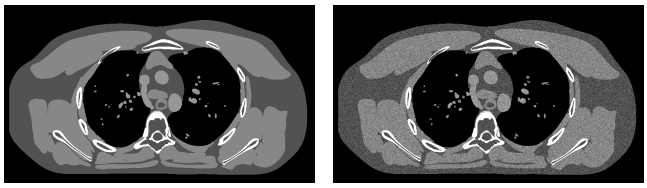

Fig. 3: Test slice (475×835) of the XCAT digital phantom [25], [26] on left with a noisy version on right (noise std. dev. of 20 HU). Display window is $[900, 1100]$ HU.

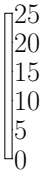

Fig. 4: Absolute error maps in $[0, 25]$ HU range for images denoised using unstructured sparse coding (left) and the learned SUoS (right) with a tolerance of $\varepsilon = 27$ HU.

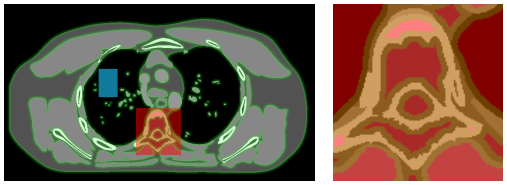

Fig. 5: Color overlays (zoomed in on right), showing locations of the regions of interest: edge vicinity (green), spine (red), their intersection (yellow), and lung (cyan).

constant nature of the XCAT phantom, and set the approximation tolerances to $\varepsilon_{\mathrm{s}} = \varepsilon_{\mathrm{u}} = 5$ HU based on a rough desired precision. The threshold number of training vectors $\tau = 25$ is chosen to remove sufficiently rare subspaces; note that $\tau/T = 25/392704 \approx 0.006\%$ of the training data.

Table I shows the number of unique supports obtained at each sparsity level k after step 3 of the learning algorithm for both $\tau = 1$ (i.e., no clusters discarded) and $\tau = 25$, in addition to the number of possible supports, $\binom{16}{k}$. Discarding small clusters ($\tau = 25$) discards 2113 patches (approximately 0.5% of the training data) and reduces the number of unique supports from 652 to 104, but even before discarding any ($\tau = 1$), there are already many fewer supports than the $2^{16} = 65536$ possible. The patches are also sparsely representable by the 2D Haar wavelets overall, with an average of $(1/T)\sum_{t=1}^{T} \|z_t\|_0 = 1.6$ nonzero coefficients per patch. However, a nontrivial number of patches are not easily represented by sparse combinations of these wavelets, evidenced by the dense supports containing over eight atoms found both when $\tau = 1$ and when $\tau = 25$. The second stage of learning finds lower-dimensional subspace representations for these patches.

Table II shows the number of subspaces obtained at each dimension after completion of the learning algorithm for both $\tau = 1$ and $\tau = 25$. As each subspace is formed from a cluster identified in steps 1–3, there are once again 652 subspaces when $\tau = 1$ and 104 when $\tau = 25$. Compared with the sparsity of supports in Table I, however, the dimensions of the final learned subspaces tend to be significantly smaller. Adapting a subspace to each cluster allows for low-dimensional subspaces when the cluster contains signals that are similar but not sparse in the input dictionary, and seeking an *average* approximation error within ε_{u} allows for even lower-dimensional subspaces that only approximate the cluster overall.

The learning algorithm automatically avoids the trivial solutions (5) and (6) by exploiting structured sparsity in the 2D Haar wavelets to cluster and adapting the subspaces to obtain 104 generally low-dimensional subspaces that all represent nontrivially many training vectors. Using a laptop with an Intel Core i5-6300U CPU (2.40 GHz, 2.50 GHz) and 8 GB of RAM, learning the subspaces from the 392704 patches takes around 15 seconds with unoptimized code written in Julia.

B. Denoising using the learned SUoS

We denoised a 475×835 test slice extracted from another portion of the XCAT phantom that has additive zero-mean Gaussian noise with a standard deviation of 20 HU as shown in Figure 3. We first denoised all 4×4 patches extracted from the noisy image and then combined the denoised patches back into a denoised image, averaging where patches overlap. Figure 4 shows absolute error maps for the denoised images obtained when patches are denoised by: a) solving (unstructured) sparse coding (11) with 2D Haar wavelets, or b) using a learned SUoS. We chose a tolerance of $\varepsilon = 27$ HU for both; it seemed to produce the best sparse coding performance in our experiments. Using a laptop with an Intel Core i5-6300U CPU (2.40 GHz, 2.50 GHz) and 8 GB of RAM, denoising the 475×835 test slice takes around 4 seconds for both sparse coding and the learned SUoS with unoptimized Julia code.

Comparing against the true (noiseless) test slice, sparse coding obtains an overall RMSE of 5.1 HU and the learned SUoS obtains a slight improvement to 4.6 HU. However, edge detail is important for these images, and the error maps reveal that the learned SUoS generally recovers edges more accurately, especially around the spine. To investigate further, we consider four regions of interest (ROI) shown in Figure 5: a) an edge ROI obtained by dilating the edge map provided by a Canny edge detector, b) a spine ROI, c) their intersection, and d) a lung ROI. The RMSE's (in HU) on all ROI's are:

	Edge	Spine	Intersection	Lung
Sparse coding	8.6	8.5	11.1	**3.6**
Learned SUoS	**7.5**	**6.1**	**7.5**	**3.6**

There is practically no improvement in the lung ROI, where the image is constant and the two models provide equally parsimonious representations. However, the learned SUoS better recovers detailed regions, most significantly seen in the intersection ROI, i.e., around edges in the spine.

Choosing ε can be a challenge in practice. Denoising again but with a larger tolerance $\varepsilon = 50$ HU yields error maps shown in Figure 6 and the following ROI RMSE's (in HU):

ISBN 978-90-827970-1-5

TABLE I: Number of unique supports obtained at each sparsity level from XCAT phantom patches ($\varepsilon_{\mathrm{s}} = 5$ HU).

Sparsity	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Total
$\tau = 1$	1	1	7	10	11	15	29	50	67	89	98	101	63	74	32	4	0	652
$\tau = 25$	1	1	2	4	1	0	8	11	11	16	6	16	11	10	2	4	0	104
# possible	1	16	120	560	1820	4368	8008	11440	12870	11440	8008	4368	1820	560	120	16	1	65536

TABLE II: Number of subspaces learned at each dimension from XCAT phantom patches ($\varepsilon_{\mathrm{s}} = \varepsilon_{\mathrm{u}} = 5$ HU).

Dimension	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Total
$\tau = 1$	1	291	106	79	40	49	22	16	10	11	13	8	4	0	2	0	0	652
$\tau = 25$	1	1	11	13	10	15	11	4	6	11	7	8	4	0	2	0	0	104

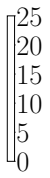

Fig. 6: Absolute error maps in $[0, 25]$ HU range for images denoised using unstructured sparse coding (left) and the learned SUoS (right) with a larger tolerance of $\varepsilon = 50$ HU.

	Edge	Spine	Intersection	Lung
Sparse coding	10.7	12.7	16.9	**3.4**
Learned SUoS	**8.4**	**7.0**	**8.9**	**3.4**

Since the learned SUoS captures more of the structure, it is significantly more robust to choosing ε too large.

VI. Conclusions and Future work

This paper proposes a method for learning a dictionary-based sequence of unions of subspaces (SUoS) and showed the benefits of learning such a model for image denoising. Interesting avenues of future work include understanding how to choose the approximation tolerances and threshold number of atoms in learning as well as understanding the impact of the dictionary used (e.g., if it was learned and overcomplete). Developing a precise and clean formulation of the learning objective is also an interesting and important open problem. A final avenue of future work is the application of this model to inverse problems such as image reconstruction.

References

[1] M. Lustig, D. Donoho, J. Santos, and J. Pauly, "Compressed Sensing MRI," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 72–82, 2008.

[2] M. Mishali and Y. Eldar, "Blind Multiband Signal Reconstruction: Compressed Sensing for Analog Signals," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 993–1009, 2009.

[3] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, and J. P. Mesirov, "Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles," *Proc. National Academy of Sciences*, vol. 102, no. 43, pp. 15 545–15 550, 2005.

[4] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[5] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[6] Y. C. Eldar and M. Mishali, "Robust Recovery of Signals From a Structured Union of Subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5302–5316, 2009.

[7] T. Wimalajeewa, Y. C. Eldar, and P. K. Varshney, "Subspace Recovery From Structured Union of Subspaces," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 2101–2114, 2015.

[8] M. Crouse, R. Nowak, and R. Baraniuk, "Wavelet-based statistical signal processing using hidden Markov models," *IEEE Trans. Signal Process.*, vol. 46, no. 4, pp. 886–902, 1998.

[9] J. Romberg, H. Choi, and R. Baraniuk, "Bayesian tree-structured image modeling using wavelet-domain hidden Markov models," *IEEE Trans. Image Process.*, vol. 10, no. 7, pp. 1056–1068, 2001.

[10] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-Based Compressive Sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, 2010.

[11] M. F. Duarte and Y. C. Eldar, "Structured Compressed Sensing: From Theory to Applications," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4053–4085, 2011.

[12] N. Rao, B. Recht, and R. Nowak, "Universal Measurement Bounds for Structured Sparse Signal Recovery," in *Proc. 15th Int. Conf. on Artificial Intelligence and Statistics*, 2012.

[13] N. Rao, R. Nowak, C. Cox, and T. Rogers, "Classification With the Sparse Group Lasso," *IEEE Trans. Signal Process.*, vol. 64, no. 2, pp. 448–463, 2016.

[14] J. Huang and T. Zhang, "The benefit of group sparsity," *Annals of Statistics*, vol. 38, no. 4, pp. 1978–2004, 2010.

[15] L. Jacob, G. Obozinski, and J.-P. Vert, "Group lasso with overlap and graph lasso," in *Proc. 26th Annu. Int. Conf. on Machine Learning*, 2009, pp. 1–8.

[16] D. Percival, "Theoretical properties of the overlapping groups lasso," *Electronic Journal of Statistics*, vol. 6, pp. 269–288, 2012.

[17] S. Kim and E. P. Xing, "Tree-guided group lasso for multi-response regression with structured sparsity, with an application to eQTL mapping," *Annals of Applied Statistics*, vol. 6, no. 3, pp. 1095–1117, 2012.

[18] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A Sparse-Group Lasso," *J. Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013.

[19] N. Shervashidze and F. Bach, "Learning the Structure for Structured Sparsity," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4894–4902, 2015.

[20] R. Vidal, "Subspace clustering," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 52–68, 2011.

[21] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[22] S. Li, H. Yin, and L. Fang, "Group-Sparse Representation With Dictionary Learning for Medical Image Denoising and Fusion," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 12, pp. 3450–3459, 2012.

[23] J. Tropp, "Greed is Good: Algorithmic Results for Sparse Approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[24] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.

[25] W. P. Segars, M. Mahesh, T. J. Beck, E. C. Frey, and B. M. W. Tsui, "Realistic CT simulation using the 4D XCAT phantom," *Medical Physics*, vol. 35, no. 8, pp. 3800–3808, 2008.

[26] W. P. Segars, G. Sturgeon, S. Mendonca, J. Grimes, and B. M. W. Tsui, "4D XCAT phantom for multimodality imaging research," *Medical Physics*, vol. 37, no. 9, pp. 4902–4915, 2010.

ISBN 978-90-827970-1-5 © EURASIP 2018