Learning Transforms With a Specified Condition Number

Subhadip Mukherjee and Chandra Sekhar Seelamantula
Department of Electrical Engineering, Indian Institute of Science, Bangalore 560012, India
Email: {subhadip, chandra.sekhar}@ee.iisc.ernet.in

***Abstract*—We address the problem of learning data-adaptive square sparsifying transforms with a condition number constraint. We adopt an alternating minimization (Alt. Min.) strategy and propose a projection approach, following the transform-update step within every iteration of Alt. Min., to enforce the condition number constraint by solving a quadratic program. The set of updated singular values of the transform can be expressed as an affine relaxation applied on the current ones. The proposed approach, referred to as *singular value relaxation* (SVR), is compared with two recently proposed transform learning techniques in terms of signal sparsification performance. The experimental results show that the transform learnt using SVR is in better agreement with the ground-truth and leads to competitive reconstruction performance with the state-of-the-art methods with easier tuning of parameters.**

I. Introduction

The advances in *compressed sensing* [1]–[3] have inspired the quest for efficient sparse representation of signals using appropriate dictionaries or transforms. The problem of learning data-dependent dictionaries has been addressed from three different perspectives: (i) synthesis [4], (ii) analysis [5], and (ii) transform models [6]. The advantage of the transform model over the other two lies in the computational simplicity of the sparse coding operation, which reduces to applying a threshold on the transform coefficients and has polynomial-time complexity. On the contrary, the same operation for the synthesis and analysis models is NP-hard.

We propose an Alt. Min. strategy for learning square sparsifying transforms with a direct control on their condition numbers. The proposed algorithm involves a projection applied on the singular values of the transform, referred to as *singular value relaxation* (SVR), following the transform-update step. The projection, performed by solving a quadratic program (QP), enforces the transform to satisfy a given condition number constraint. Since the constraint has a direct bearing on the condition number of the resulting transform, the SVR algorithm does not entail significant parameter tuning effort.

II. Problem Formulation and the SVR Algorithm

Given a training data matrix $Y = \{y_i \in \mathbb{R}^m, i = 1 : N\}$, the goal is to learn a well-conditioned sparsifying transform $W \in \mathbb{R}^{m\times m}$ such that $Wy_i = x_i + \xi_i$, where x_i is s-sparse and ξ_i is the modeling error, for all i. The objective is achieved by solving

$$\min_{W,x_i} \sum_{i=1}^{N} \|Wy_i - x_i\|_2^2 \text{ s.t. } \|x_i\|_0 \leq s, \|W\|_F = 1, \kappa(W) \leq \alpha, \quad (1)$$

where $\alpha \geq 1$ is a constant and $\kappa(W) = \frac{\sigma_{\max}(W)}{\sigma_{\min}(W)}$, the ratio of the maximum to the minimum singular values of W, denotes the condition number of W. The constraints in (1) help avoid the degenerate solution. To learn W satisfying condition number and norm constraints, Ravishankar and Bresler [6], [7] proposed to solve $\min_{W,x_i} \sum_{i=1}^{N} \|Wy_i - x_i\|_2^2 - \lambda \log|\det W| + \mu \|W\|_F^2$ s.t. $\|x_i\|_0 \leq s$, where λ and μ are appropriately chosen constants, using an Alt. Min. strategy. In their formalism, it is difficult to determine suitable values of λ and μ to enforce a certain condition number α on the learnt transform. On the contrary, the optimization posed in (1) offers a direct control on the desired condition number of the learnt transform.

To solve (1), we initialize with an estimate of W (typically the identity matrix) and alternate between finding the best X for a given W and vice versa. After obtaining an estimate of W, the constraint $\kappa(W) \leq \alpha$ is imposed on W by solving the QP

$$\tilde{\sigma} = \arg\min_{s_j} \frac{1}{2} \sum_{j=1}^{m} (s_j - \sigma_j)^2 \text{ s.t. } s_1 \geq \cdots \geq s_m \geq 0; s_1 \leq \alpha s_m, \quad (2)$$

where $\sigma_1 \geq \cdots \geq \sigma_m$, are the singular values of the current estimate of the transform such that $\frac{\sigma_1}{\sigma_m} > \alpha$. The Karush-Kuhn-Tucker (KKT) conditions for the QP reveal that solving (2) amounts to applying an affine relaxation of the form $\tilde{\sigma}_j = \sigma_j + \nu_j$ on the current singular values σ_j, to obtain the updated ones $\tilde{\sigma}_j$.

After the updated estimate $\tilde{W}$ is obtained, an optimal rotation Q_0 is applied on $\tilde{W}$ to minimize the error on the training set [8]:

$$Q_0 = \arg\min_{Q:Q^\top Q = I} \left\|Q\tilde{W}Y - X\right\|_F^2. \quad (3)$$

Premultiplying $\tilde{W}$ by the orthonormal Q_0 does not alter its condition number, but reduces the sparsification error. The optimization posed in (3) is famously known as the *orthogonal Procrustes problem* (OPP) and can be solved in closed-form [9] as $Q_0 = RP^\top$, where $C = P\Lambda R^\top$ is the singular-value decomposition (SVD) of $C \triangleq \tilde{W}YX^\top$. In the special case where $\alpha = 1$, that is, the transform matrix W is orthonormal, one can directly solve

$$W = \arg\min_{\hat{W}:\hat{W}^\top \hat{W} = I} \left\|\hat{W}Y - X\right\|_F^2, \quad (4)$$

to update W, instead of solving (2) and (3) separately, since both approaches lead to the same update as argued in the following:

Solving (4) leads to $W = R_1 P_1^\top$, where $P_1\Lambda_1 R_1^\top$ is the SVD of $YX^\top$. The matrix $\tilde{W}$, constructed using the solution of (2), is given by $\tilde{W} = \beta UV^\top$, where $\beta = \frac{1}{m}\sum_{i=1}^{m} \sigma_i$. Therefore, the matrix $C = \tilde{W}YX^\top$ takes the form $C = \beta UV^\top P_1\Lambda_1 R_1^\top = \left(UV^\top P_1\right)(\beta\Lambda_1) R_1^\top$, thereby leading to $Q_0 = R_1 P_1^\top VU^\top$. Consequently, the resulting updated transform after solving (2) and (3) becomes $W = Q_0\tilde{W} = \beta\left(R_1 P_1^\top VU^\top\right)UV^\top = \beta R_1 P_1^\top$, which is same as the update obtained from (4), up to a scale factor β. Notably, the problem of learning an orthonormal synthesis dictionary for sparse coding was recently considered in [10] and an alternating minimization approach was proposed, wherein the synthesis basis is updated via an OPP. The proposed SVR approach subsumes the special case of orthogonal basis considered in [10]. Following rotation, the transform W is rescaled to ensure that $\|W\|_F = 1$. The steps of SVR are summarized in Algorithm 1.

III. Numerical Experiments on Sparsification

A. Synthetic Signals

The sparsification performance of SVR is compared with the transform learning algorithms based on conjugate-gradient (TL-CG) [6] and closed-form updates (TL-CFU) [7]. For a fair comparison, we consider identical experimental settings as in [6]. A data matrix Y, containing $N = 200$ training examples, is generated by multiplying a synthesis dictionary of size 20×20 with a coefficient matrix A, having exactly four nonzero entries in every column. The synthesis

Algorithm 1 A singular-value relaxation (SVR) technique to learn a well-conditioned sparsifying square transform W.

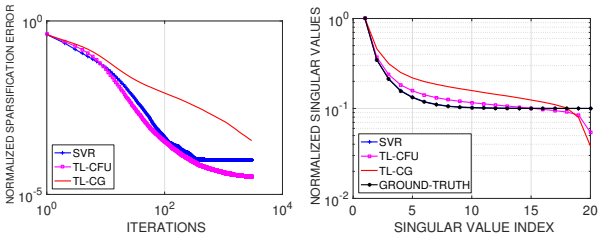
1. Input: Sparsity s, data matrix Y, α, and no. of iterations J_{iter}.
2. Initialization: Set $p \leftarrow 0$, $W^{(p)} \leftarrow I$.
3. Iterate J_{iter} times:
(i) *Sparse coding*: $x_i^{(p)} = T_s\left(W^{(p)} y_i\right)$, for all i, where $T_s(z)$ is the hard-thresholding operation that retains the top s entries (in magnitude) of z.
(ii) *Transform estimate*: $W^{(p)} \leftarrow X^{(p)} Y^{\dagger}$; where $\dagger$ is pseudoinverse.
(iii) *Affine relaxation*: If $\kappa\left(W^{(p)}\right) > \alpha$, compute the SVD of $W^{(p)} = UKV^T$, where $K = \text{diag}(\sigma_1, \cdots, \sigma_m)$. Calculate $\tilde{\sigma}_j$ by solving (2), and construct $\tilde{K} = \text{diag}(\tilde{\sigma}_1, \cdots, \tilde{\sigma}_m)$. Obtain the updated transform $\tilde{W} = U\tilde{K}V^{\top}$.
(iv) Set $W^{(p)} \leftarrow Q_0 \tilde{W}$, where Q_0 is the solution to (3).
(v) Perform scaling $W^{(p)} \leftarrow \frac{W^{(p)}}{\left\|W^{(p)}\right\|_F}$ and update $p \leftarrow p+1$.
5. Output: The learnt transform $W^{(p)}$.

dictionary is generated by drawing samples from the $\mathcal{N}(0,1)$ distribution, followed by enforcing a constraint such that its inverse, the ground-truth transform, has a condition number less than or equal to 10. The locations of the nonzero entries in the coefficient matrix are chosen uniformly at random, and their amplitudes are drawn from $\mathcal{N}(0,1)$. The parameter α in the SVR algorithm is taken as $\alpha = 10$, same as the condition number of the ground-truth. The maximum number of iterations is taken as $J_{\text{iter}} = 3000$. The parameters of TL-CG and TL-CFU are fixed exactly as recommended in [6]. The metrics for comparison are chosen to be the *normalized sparsification error*, defined as $\frac{\|WY - X\|_F^2}{\|WY\|_F^2}$ and the singular values (normalized by the maximum) of the resulting transform (cf. Figures 1(a) and 1(b), respectively). Since the training data is generated randomly, these performance metrics are averaged over 500 trials.

We observe that the normalized sparsification error for the SVR algorithm decays at a rate faster than TL-CG and competitive with TL-CFU, and attains a small value (about 10^{-4}) after 500 iterations. The normalized singular values of the transform learnt using SVR are in excellent agreement with those of the ground-truth transform. On the other hand, the TL-CG and TL-CFU algorithms learn transforms with slightly higher condition numbers.

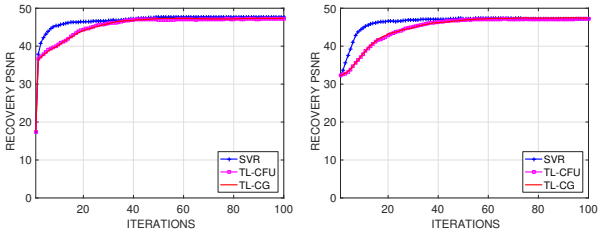
B. Sparsifying Image Patches

Non-overlapping patches of size 8×8 are extracted from the *Barbara* image, mean-subtracted, vectorized, and stacked as the columns of Y. For an input image of size 512×512, Y contains $N = 4096$ patches. As suggested in [6], the sparsity level is chosen as $s = 11$. The transform to be learnt is of size 64×64 and the value of α is chosen to be 1.1. All algorithms are iterated 100 times. The peak signal-to-noise ratio (PSNR), defined as $20 \log_{10} \frac{255\sqrt{P}}{\|Y - W^{-1}X\|_F}$ dB, where P is the number of pixels in the image, is chosen as the metric for comparison and shown in Figures 2(a) and 2(b) corresponding to identity and random initializations, respectively. The recovery PSNR in case of SVR increases faster than TL-CFU and TL-CG as the iterations progress. All the three techniques were found to be robust to initialization as the final recovery PSNR is nearly the same for both random and identity initializations.



(a) (b)

Fig. 1. (Color online) Comparison of SVR, TL-CFU, and TL-CG for noise-free data: (a) Normalized sparsification error and (b) normalized singular values, which are almost identical for SVR and the ground-truth.



(a) $W_{ij}^{(0)} \sim \mathcal{N}(0,1)$ (b) $W^{(0)} = I_m$

Fig. 2. (Color online) Image sparsification experiment: Recovery PSNR versus iterations for SVR, TL-CFU, and TL-CG, corresponding to random and identity matrix initializations.

IV. Conclusions

We have developed a new Alt. Min. algorithm for learning a data-dependent square sparsifying transform, where a condition number constraint is imposed by solving a QP in the singular-value domain, following the transform-update step. The constraint can be set directly depending on the target condition number, thereby leading to easier parameter fixing compared with the frameworks in [6] and [7]. The resulting SVR algorithm has a performance that is competitive with the state-of-the-art techniques and the singular values of the learnt transform exactly match those of the ground-truth.

References

[1] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer, 2010.
[2] D. L. Donoho, "Compressed sensing," *IEEE Trans. Info. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[3] E. J. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, pp. 21 –30, Mar. 2008.
[4] 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, Nov. 2006.
[5] R. Rubinstein, T. Peleg, and M. Elad, "Analysis K-SVD: A dictionary learning algorithm for the analysis sparse model," *IEEE Trans. Signal Process.*, vol. 61, no. 3, pp. 661–677, Feb. 2013.
[6] S. Ravishankar and Y. Bresler, "Learning sparsifying transforms," *IEEE Trans. Signal Process.*, vol. 61, no. 5, pp. 1072–1086, Mar. 2013.
[7] S. Ravishankar and Y. Bresler, "ℓ_0 sparsifying transform learning with efficient optimal updates and convergence guarantees," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2389–2404, May 2015.
[8] D. Barchiesi and M. D. Plumbley, "Learning incoherent dictionaries for sparse approximation using iterative projections and rotations," *IEEE Trans. Signal Process.*, vol. 61, no. 8, pp. 2055–2065, Apr. 2013.
[9] P. H. Schönemann, "A generalized solution of the orthogonal Procrustes problem," *Psychometrika*, vol. 31, issue 1, pp. 1–10, Mar. 1966.
[10] H. Schütze, E. Barth, and T. Martinetz, "Learning efficient data representations with orthogonal sparse coding," *IEEE Trans. Computational Imaging*, vol. 2, no. 3, pp.177–189, Sep. 2016.