Learning Fast Orthonormal Sparsifying Transforms

Cristian Rusu and John Thompson
University of Edinburgh, UK
{c.rusu, john.thompson}@ed.ac.uk

The goal of this paper is to propose an algorithm that learns an orthonormal transform matrix (also called a dictionary in the sparse representation literature) of size $n \times n$ from a given training dataset that is numerically efficient, i.e., can be applied to data in $O(n \log n)$. We achieve this reduced complexity by factorizing the dictionary into a series of basic structured transformations that can be applied sequentially. We choose to focus on orthonormal transforms [1] since in the sparse approximation step these avoid the use of the numerically complex orthogonal matching pursuit (OMP) [2] or ℓ_1 [3] minimization, but still have complexity $O(n^2)$.

Given an N-sample dataset $\mathbf{Y} \in \mathbb{R}^{n \times N}$, the general orthonormal dictionary learning problem (which has been studied in the past and that we call here Q–DLA) [4] is formulated as:

$$\underset{\mathbf{U},\ \mathbf{X};\ \mathbf{U}\mathbf{U}^T=\mathbf{U}^T\mathbf{U}=\mathbf{I}}{\text{minimize}} \|\mathbf{Y}-\mathbf{U}\mathbf{X}\|_F^2 \text{ s. to } \|\mathbf{x}_i\|_0 \leq s,\ 1 \leq i \leq N. \quad (1)$$

This problem can be efficiently solved by alternating minimization: with $\mathbf{X}$ fixed, $\mathbf{U}$ is computed via the orthogonal Procrustes problem and with $\mathbf{U}$ fixed we have $\mathbf{X} = \mathcal{T}_s(\mathbf{U}^T\mathbf{Y})$ where $\mathcal{T}_s$ is an operator applied columnwise that keeps only the largest s entries in magnitude.

In this paper we propose to construct an orthonormal dictionary $\mathbf{U} \in \mathbb{R}^{n \times n}$ already factored as a product of m $\mathbf{G}_{ij}$ transforms:

$$\mathbf{U} = \mathbf{G}_{i_m j_m} \ldots \mathbf{G}_{i_2 j_2}\mathbf{G}_{i_1 j_1}. \quad (2)$$

The value of $m \ll n^2$ is a user choice. A G-transform is an orthonormal matrix with $c, d \in \mathbb{R}$ and indices $i \neq j$ as

$$\mathbf{G}_{ij} = \begin{bmatrix} \mathbf{I}_{i-1} & & & & \\ & * & & * & \\ & & \mathbf{I}_{j-i-1} & & \\ & * & & * & \\ & & & & \mathbf{I}_{n-j} \end{bmatrix} \in \mathbb{R}^{n \times n}, \quad (3)$$

where we have denoted $\mathbf{I}_i$ as the identity matrix of size i and $*$ stands for a non-zero entry. We denote the non-trivial part of $\mathbf{G}_{ij}$ as

$$\tilde{\mathbf{G}}_{ij} = \left\{ \begin{bmatrix} c & d \\ -d & c \end{bmatrix}, \begin{bmatrix} c & d \\ d & -c \end{bmatrix} \right\} \in \mathbb{R}^{2\times 2},\ c^2+d^2 = 1. \quad (4)$$

Notice that the matrix-vector multiplication $\mathbf{G}_{ij}\mathbf{y}$ takes only 6 operations and therefore $\mathbf{U}\mathbf{y}$ takes $6m$ with $\mathbf{U}$ from (2). Notice that a G-transform is a $(n+2)$-sparse matrix [5]. Consider now the dictionary learning problem in (1). Let us keep the sparse representations $\mathbf{X}$ fixed and consider a single G-transform as a dictionary. We reach the following

$$\underset{(i,j),\ \tilde{\mathbf{G}}_{ij}}{\text{minimize}} \ \|\mathbf{Y}-\mathbf{G}_{ij}\mathbf{X}\|_F^2. \quad (5)$$

For simplicity of exposition we define

$$\mathbf{Z} = \mathbf{Y}\mathbf{X}^T, \mathbf{Z}_{\{i,j\}} = \begin{bmatrix} Z_{ii} & Z_{ij} \\ Z_{ji} & Z_{jj} \end{bmatrix} \in \mathbb{R}^{2\times 2}, Z_{ij} = \mathbf{y}_i^T\mathbf{x}_j, \quad (6)$$

where $\mathbf{y}_i^T$ and $\mathbf{x}_i^T$ are the i^{th} rows of $\mathbf{Y}$ and $\mathbf{X}$, respectively. Therefore, the objective function of (5) is

$$\begin{aligned} &\|\mathbf{Y}-\mathbf{G}_{ij}\mathbf{X}\|_F^2 = \|\mathbf{Y}\|_F^2 + \|\mathbf{X}\|_F^2 - 2\text{tr}(\mathbf{Z}) - 2C_{ij}, \\ &\text{where } C_{ij} = \|\mathbf{Z}_{\{i,j\}}\|_* - \text{tr}(\mathbf{Z}_{\{i,j\}}). \end{aligned} \quad (7)$$

Algorithm 1 – $\mathbf{G}_m$–DLA. Fast Orthonormal Transform Learning.
Input: The dataset $\mathbf{Y} \in \mathbb{R}^{n\times N}$, the number of G-transforms m, the target sparsity s and the number of iterations K.
Output: The sparsifying orthonormal transform $\mathbf{U}$ as (2) and sparse representations $\mathbf{X}$ such that $\|\mathbf{Y}-\mathbf{U}\mathbf{X}\|_F^2$ is reduced.

Initialization:

1) Perform the singular value decomposition of the dataset $\mathbf{Y} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$.
2) Compute sparse representations $\mathbf{X} = \mathcal{T}_s(\mathbf{U}^T\mathbf{Y})$.
3) For $k = 1, \ldots, m$: with all previous $(k-1)$ G-transforms fixed, construct the new $\mathbf{G}_{i_k j_k}$ by (7) such that
$$\|\mathbf{Y}-\mathbf{G}_{i_k j_k}\mathbf{G}_{i_{k-1}j_{k-1}}\ldots\mathbf{G}_{i_1 j_1}\mathbf{X}\|_F^2 = \|\mathbf{Y}-\mathbf{G}_{i_k j_k}\mathbf{X}_k\|_F^2 \quad (10)$$
is minimized.

Iterations $1, \ldots, K$:

1) For $k = 1, \ldots, m$: update the new $\mathbf{G}_{i_k j_k}$, with all other transforms fixed, such that (9) is minimized.
2) Compute sparse representations $\mathbf{X} = \mathcal{T}_s(\mathbf{U}^T\mathbf{Y})$, where $\mathbf{U}$ is given by (2).

Since we want to minimize this quantity, the choice of indices needs to be made as follows

$$(i^\star, j^\star) = \underset{(i,j),\ j>i}{\arg\max}\ C_{ij}, \quad (8)$$

and then solve a Procrustes problem [6] of size 2 to construct $\tilde{\mathbf{G}}_{i^\star j^\star}$.

To construct the complete $\mathbf{U}$, we fix the representations $\mathbf{X}$ and all G-transforms in (2) except for the k^{th}, denoted as $\mathbf{G}_{i_k j_k}$. To optimize the dictionary $\mathbf{U}$ for this transform we reach the objective function

$$\begin{aligned} \|\mathbf{Y}-\mathbf{U}\mathbf{X}\|_F^2 &= \|\mathbf{Y}-\mathbf{G}_{i_m j_m}\ldots\mathbf{G}_{i_1 j_1}\mathbf{X}\|_F^2 \\ &= \|\mathbf{G}_{i_{k+1}j_{k+1}}^T \ldots \mathbf{G}_{i_m j_m}^T\mathbf{Y} - \mathbf{G}_{i_k j_k}\ldots\mathbf{G}_{i_1 j_1}\mathbf{X}\|_F^2 \\ &= \|\mathbf{Y}_k - \mathbf{G}_{i_k j_k}\mathbf{X}_k\|_F^2, \end{aligned} \quad (9)$$

where we have used the fact that multiplication by any orthonormal transform preserves the Frobenius norm. Matrices $\mathbf{Y}_k$ and $\mathbf{X}_k$ contain the accumulations of the G-transforms on $\mathbf{Y}$ and $\mathbf{X}$, respectively.

The full procedure, called G_m–DLA [7] is described in Algorithm 1 and the results on image data are shown in Figures 1 and 2. Figure 1 shows the converge of G_m–DLA while Figure 2 shows its capacity to build computationally efficient dictionaries whose representation performance is between that of the classical fast discrete cosine transform (DCT) and that of computationally complex learned orthonormal dictionaries.

Acknowledgment

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/K014277/1 and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

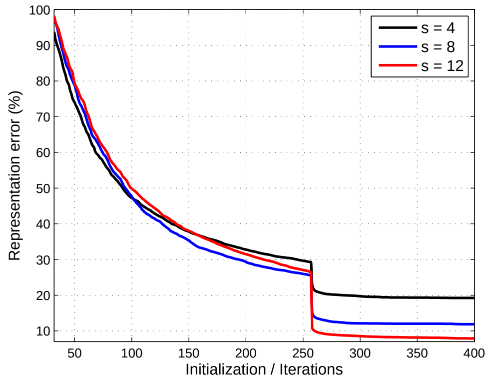

Fig. 1. For the proposed G_{256}–DLA we show the evolution of the relative representation error $\epsilon = \|\mathbf{Y} - \mathbf{U}\mathbf{X}\|_F^2 \|\mathbf{Y}\|_F^{-2}$ (%) for the dataset $\mathbf{Y}$ created from the patches of the images couple, peppers and boat with sparsity $s \in \{4, 8, 12\}$. The first 256 points in the plot are due to the initialization step ($m = 256$ transforms are initialized) and the other $K = 150$ are the regular iterations of G_{256}–DLA. The test dataset $\mathbf{Y} \in \mathbb{R}^{64\times 12288}$ consists of 8×8 non-overlapping patches with their means removed and normalized $\mathbf{Y} = \mathbf{Y}/255$.

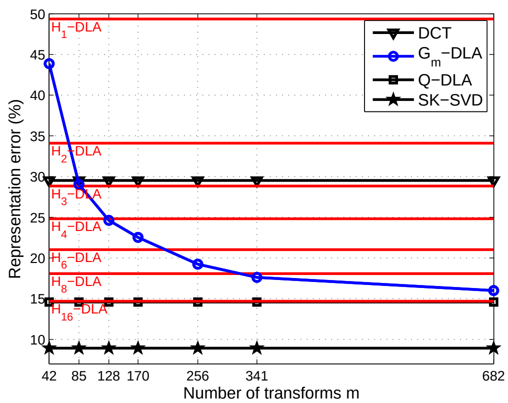

Fig. 2. For the same dataset as in Figure 1, we show comparisons, in terms of relative representation errors $\epsilon = \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \|\mathbf{Y}\|_F^{-2}$ (%), of G_m–DLA against the DCT [8], Q–DLA [4], SK–SVD [9][10][11] and Householder based orthonormal dictionaries [12] denoted here H_p–DLA where p is the number of reflectors in the factorization of the dictionary. The number of transforms m is chosen so that computational complexity comparisons against H_p–DLA is possible. Computational complexity approximately match between: H_1–DLA and G_{42}–DLA, H_2–DLA and G_{85}–DLA, H_3–DLA and G_{128}–DLA, H_4–DLA and G_{170}–DLA, H_6–DLA and G_{256}–DLA, H_8–DLA and G_{341}–DLA, H_{16}–DLA and G_{682}–DLA. The sparsity level is set to $s = 4$ for all methods.

References

[1] S. Ravishankar and Y. Bresler, "Learning sparsifying transforms," *IEEE Trans. Signal Process.*, vol. 61, no. 5, 2013.

[2] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, pp. 2231–2242, 2004.

[3] ——, "Just relax: Convex programming methods for subset selection and sparse approximation," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1030–1051, 2006.

[4] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, "Learning unions of orthonormal bases with thresholded singular value decomposition," in *Proc. IEEE ICASSP*, 2005, pp. 293–296.

[5] L. Le Magoarou and R. Gribonval, "Chasing butterflies: In search of efficient dictionaries," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on. IEEE*, 2015.

[6] P. Schonemann, "A generalized solution of the orthogonal Procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.

[7] C. Rusu and J. Thompson, "Learning fast sparsifying transforms," *https://arxiv.org/abs/1611.08230*, 2016.

[8] W.-H. Chen, C. H. Smith, and S. C. Fralick, "A fast computational algorithm for the discrete cosine transform," *IEEE Trans. Communications*, vol. 25, no. 9, pp. 1004–1009, 1977.

[9] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Sig. Proc.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[10] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," *CS Technion*, 2008.

[11] C. Rusu and B. Dumitrescu, "Stagewise K-SVD to design efficient dictionaries for sparse representations," *IEEE Signal Processing Letters*, vol. 19, no. 10, pp. 631–634, 2012.

[12] C. Rusu, N. Gonzalez-Prelcic, and R. Heath, "Fast orthonormal sparsifying transforms based on Householder reflectors," *IEEE Trans. Sig. Process.*, vol. 64, no. 24, pp. 6589–6599, 2016.