Pairwise Approximate K-SVD

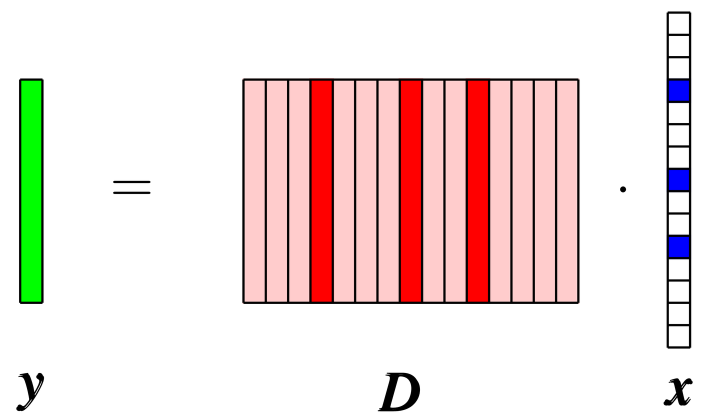
Paul Irofti, Bogdan Dumitrescu

The Research Institute of the University of Bucharest (ICUB) and Department of Computer Science, University of Bucharest, Romania

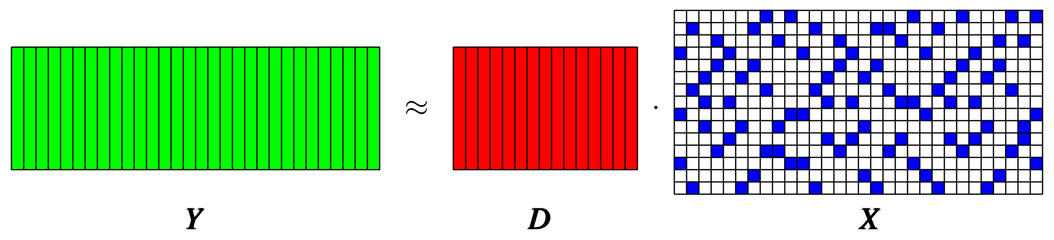
Department of Automatic Control and Computers, University Politehnica of Bucharest, Romania

Dictionary Learning

Sparse representation of vectorized signal $\boldsymbol{y} \in \mathbb{R}^{m^2}$

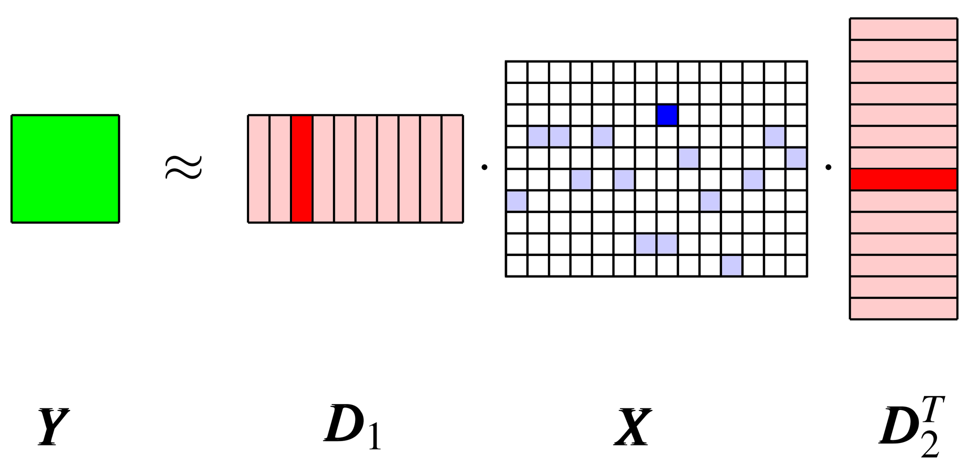


- $\boldsymbol{D} \in \mathbb{R}^{m \times n_1 n_2}$ is the dictionary
- $\boldsymbol{x} \in \mathbb{R}^{n_1 n_2}$ is the s-sparse representation
- dictionary columns are called atoms

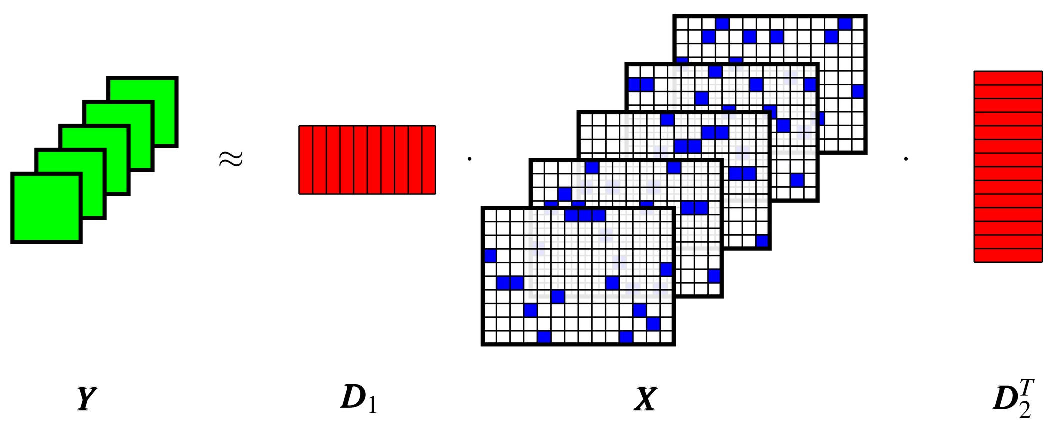


Separable Dictionary Learning

Sparse representation of a 2D signal $\boldsymbol{Y} \in \mathbb{R}^{m \times m}$



- $\boldsymbol{D}_1 \in \mathbb{R}^{m \times n_1}$, $\boldsymbol{D}_2 \in \mathbb{R}^{m \times n_2}$ left and right dictionaries
- $\boldsymbol{X} \in \mathbb{R}^{n_1 \times n_2}$ is the s-sparse representation



First algorithm SeDiL: highly complex algorithm based on manifold gradient search. Alternatives:

- rank-1 CP decomposition based (2DS), closer to K-SVD
- tensor orientated, using t-SVD (K-TSVD, TKSVD)
- non-separable focused on the Kronecker operation (SuKro, 2DNS)

Optimization Problem

Goal: Minimize the error

$$\min_{\boldsymbol{D}_1,\boldsymbol{D}_2,\boldsymbol{X}} \sum_{k=1}^{N} \|\boldsymbol{Y}_k - \boldsymbol{D}_1 \boldsymbol{X}_k \boldsymbol{D}_2^T\|_F^2 \quad \text{subject to } \|\boldsymbol{X}_k\|_0 \le s,\ k = 1:N;\quad \|\boldsymbol{d}_{1i}\|_2 = 1,\ i = 1:n_1;\quad \|\boldsymbol{d}_{2j}\|_2 = 1,\ j = 1:n_2 \tag{1}$$

Problem (1) is DL with dictionary $\boldsymbol{D}_2 \otimes \boldsymbol{D}_1$:

$$\|\boldsymbol{Y}_k - \boldsymbol{D}_1 \boldsymbol{X}_k \boldsymbol{D}_2^T\|_F^2 = \|\mathrm{vec}\,\boldsymbol{Y}_k - (\boldsymbol{D}_2 \otimes \boldsymbol{D}_1)\mathrm{vec}\,\boldsymbol{X}_k\|^2 \tag{2}$$

For the left and right atom indices $\mathcal{I} = \{i_1, \dots, i_s\}$ and $\mathcal{J} = \{j_1, \dots, j_s\}$, the approximation of $\boldsymbol{Y}_k$ is

$$\|\mathcal{E}(\boldsymbol{X}_k)\|_F = \|\boldsymbol{Y}_k - \sum_{\iota=1}^{s} x^{(k)}_{i_\iota j_\iota} \boldsymbol{d}_{1i_\iota} \boldsymbol{d}_{2j_\iota}^T\|_F = \|\mathrm{vec}\,\boldsymbol{Y}_k - \sum_{\iota=1}^{s} (\boldsymbol{d}_{2j_\iota} \otimes \boldsymbol{d}_{1i_\iota}) x^{(k)}_{i_\iota, j_\iota}\|. \tag{3}$$

TKSVD Dictionary Update

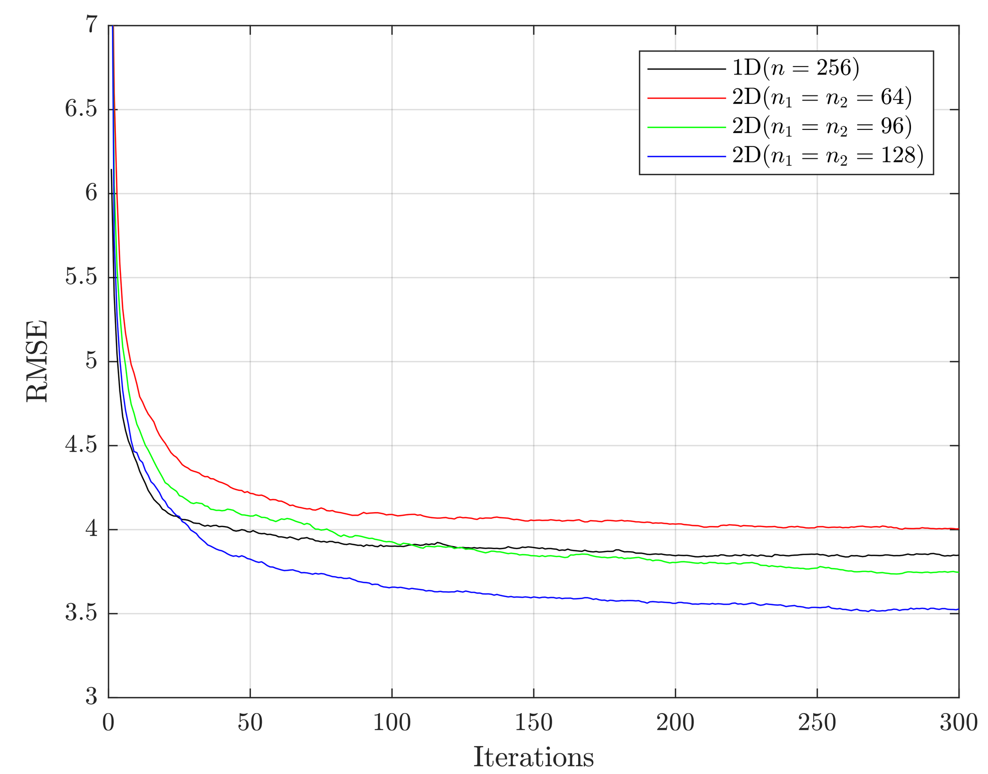
Simultaneously updates each (i, j) atoms pair

$$\boldsymbol{R}_k = \mathcal{E}(\boldsymbol{X}_k) + x^{(k)}_{i,j} \boldsymbol{d}_{1i} \boldsymbol{d}_{2j}^T \tag{4}$$

$$\min_{\boldsymbol{d}_{1i}, \boldsymbol{d}_{2j}} \sum_{k=1}^{N} \left\| \boldsymbol{R}_k - x^{(k)}_{i,j} \boldsymbol{d}_{1i} \boldsymbol{d}_{2j}^T \right\|_F^2 \quad \text{subject to } \|\boldsymbol{d}_{1i}\|_2 = 1, \|\boldsymbol{d}_{2j}\|_2 = 1. \tag{5}$$

- many $x^{(k)}_{i,j}$ are zero; rank-1 CP decomposition
- process iterated by TKSVD for all atom pairs
- diverges when two $\boldsymbol{d}_{2j}$ and $\boldsymbol{d}_{2j'}$ pair with $\boldsymbol{d}_{1i}$

RMSE



Pairwise Approximate K-SVD

Fix everything in (1) except for either atom i of $\boldsymbol{D}_1$

$$\min_{\boldsymbol{d}_{1i}} \sum_{k=1}^{N} \left\| \boldsymbol{R}_{1k} - \boldsymbol{d}_{1i} \boldsymbol{x}^{(k)}_{i,\mathcal{P}_k} \boldsymbol{D}_{2\mathcal{P}_k}^T \right\|_F^2 \text{ s.t. } \|\boldsymbol{d}_{1i}\|_2 = 1 \tag{6}$$

or atom j of $\boldsymbol{D}_2$

$$\min_{\boldsymbol{d}_{2j}} \sum_{k=1}^{N} \left\| \boldsymbol{R}_{2k} - \boldsymbol{D}_{1\mathcal{Q}_k} \boldsymbol{x}^{(k)}_{\mathcal{Q}_k,j} \boldsymbol{d}_{2j}^T \right\|_F^2 \text{ s.t. } \|\boldsymbol{d}_{2j}\|_2 = 1$$

When representing $\boldsymbol{Y}_k$

- $\mathcal{P}_k$ are the atoms in $\boldsymbol{D}_2$ pairing with atom i in $\boldsymbol{D}_1$
- $\mathcal{Q}_k$ are the atoms in $\boldsymbol{D}_1$ pairing with $\boldsymbol{d}_{2j}$

The k-th term of the sum from (6) becomes

$$\|\boldsymbol{R}_{1k}\|_F^2 - 2\mathrm{tr}(\boldsymbol{R}_{1k}^T \boldsymbol{d}_{1i} \boldsymbol{x}^{(k)}_{i,\mathcal{P}_k} \boldsymbol{D}_{2\mathcal{P}_k}^T) + \boldsymbol{x}^{(k)}_{i,\mathcal{P}_k} \boldsymbol{D}_{2\mathcal{P}_k}^T \boldsymbol{D}_{2\mathcal{P}_k} (\boldsymbol{x}^{(k)}_{i,\mathcal{P}_k})^T, \tag{7}$$

Minimized when the trace is maximized

$$\max_{\boldsymbol{d}_{1i}} \left(\sum_{k=1}^{N} \boldsymbol{x}^{(k)}_{i,\mathcal{P}_k} \boldsymbol{D}_{2\mathcal{P}_k}^T \boldsymbol{R}_{1k}^T \right) \boldsymbol{d}_{1i} \text{ s.t. } \|\boldsymbol{d}_{1i}\|_2 = 1 \tag{8}$$

Maximum when $\boldsymbol{d}_{1i}$ is collinear to the sum vector

$$\boldsymbol{d}_{1i} = \frac{\boldsymbol{t}_1}{\|\boldsymbol{t}_1\|},\ \boldsymbol{t}_1 = \sum_{k=1}^{N} \boldsymbol{R}_{1k} \boldsymbol{D}_{2\mathcal{P}_k} (\boldsymbol{x}^{(k)}_{i,\mathcal{P}_k})^T. \tag{9}$$

$$(\boldsymbol{x}^{(k)}_{i,\mathcal{P}_k})^T = (\boldsymbol{D}_{2\mathcal{P}_k}^T \boldsymbol{D}_{2\mathcal{P}_k})^{\dagger} \boldsymbol{D}_{2\mathcal{P}_k}^T \boldsymbol{R}_{1k}^T \boldsymbol{d}_{1i}. \tag{10}$$

For atom $\boldsymbol{d}_{2j}$ we update following

$$\boldsymbol{d}_{2j} = \frac{\boldsymbol{t}_2}{\|\boldsymbol{t}_2\|},\ \boldsymbol{t}_2 = \sum_{k=1}^{N} \boldsymbol{R}_{2k}^T \boldsymbol{D}_{1\mathcal{Q}_k} \boldsymbol{x}^{(k)}_{\mathcal{Q}_k,j}. \tag{11}$$

$$\boldsymbol{x}^{(k)}_{\mathcal{Q}_k,j} = (\boldsymbol{D}_{1\mathcal{Q}_k}^T \boldsymbol{D}_{1\mathcal{Q}_k})^{\dagger} \boldsymbol{D}_{1\mathcal{Q}_k}^T \boldsymbol{R}_{2k} \boldsymbol{d}_{2j}. \tag{12}$$

Pairwise Approximate K-SVD algorithm

Require: initial dictionaries $\boldsymbol{D}_1 \in \mathbb{R}^{m \times n_1}$, $\boldsymbol{D}_2 \in \mathbb{R}^{m \times n_2}$
signals set $\boldsymbol{Y}_k \in \mathbb{R}^{m \times m}$, $k = 1:N$
number of iterations $T \in \mathbb{N}$

Ensure: learned dictionaries $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$

1: **for** $t = 1$ **to** T **do**
2: Sparse coding: keeping $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ fixed, compute sparse representations $\boldsymbol{X}$ using 2D OMP
Dictionary update:
3: **for** $i = 1$ **to** n_1 **do**
4: Compute the new atom $\boldsymbol{d}_{1i}$ using (9)
5: Update $\boldsymbol{x}^{(k)}_{i,\mathcal{P}_k}$ using (10), $k = 1:N$
6: **for** $j = 1$ **to** n_2 **do**
7: Compute the new atom $\boldsymbol{d}_{2j}$ using (11)
8: Update $\boldsymbol{x}^{(k)}_{\mathcal{Q}_k,j}$ using (12), $k = 1:N$

Complexity:

- 2D OMP is $\mathcal{O}(m n_1 n_2)$, $1/m$ that of standard OMP
- dictionary update is $O(sm^2N)$, same as AK-SVD
- faster than CP-decomposition based algorithms
- considerably faster than SeDiL

References

[1] B. Dumitrescu and P. Irofti. *Dictionary Learning Algorithms and Applications*. Springer, 2018.

[2] Y. Fang, J. Wu, and B. Huang. 2D sparse signal recovery via 2D orthogonal matching pursuit. *Science China Information Sciences*, 55(4):889–897, 2012.

[3] Y. Fu, J. Gao, Y. Sun, and X. Hong. Joint multiple dictionary learning for tensor sparse coding. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, pages 2957–2964. IEEE, 2014.

[4] S. Hawe, M. Seibert, and M. Kleinsteuber. Separable dictionary learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 438–445, 2013.

Denoising

σ_{noise} / PSNR	Method	lena		barbara		boat		peppers		house	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
5 / 34.15	1D	38.46	0.942	37.94	0.962	36.98	0.935	37.60	0.925	39.27	0.953
	2D	38.22	0.940	37.69	0.961	36.63	0.929	37.34	0.920	38.71	0.948
10 / 28.13	1D	35.30	0.907	34.21	0.931	33.47	0.878	34.63	0.876	35.62	0.901
	2D	35.21	0.905	34.10	0.930	33.42	0.875	34.56	0.873	35.43	0.897
20 / 22.11	1D	32.02	0.857	30.21	0.867	29.89	0.789	31.88	0.833	32.47	0.857
	2D	31.94	0.855	30.08	0.863	29.88	0.787	31.88	0.831	32.25	0.854
30 / 18.58	1D	29.96	0.817	27.75	0.803	27.90	0.727	30.12	0.804	30.44	0.828
	2D	29.91	0.815	27.65	0.799	27.91	0.725	30.10	0.802	30.18	0.824
50 / 14.15	1D	27.37	0.754	24.66	0.685	25.47	0.642	27.66	0.755	27.47	0.762
	2D	27.32	0.752	24.57	0.681	25.41	0.638	27.72	0.754	27.35	0.758

https://cs.unibuc.ro/~pirofti, http://acse.pub.ro https://github.com/pirofti/pair_ksvd paul@irofti.net, bogdan.dumitrescu@acse.pub.ro