Xiu-rui Geng, Lu-yan Ji, Kang Sun, 2015. Non-negative matrix factorization based unmixing for principal component transformed hyperspectral data. *Frontiers of Information Technology & Electronic Engineering,* **17**(5):403-412. http://dx.doi.org/10.1631/FITEE.1600028

Non-negative matrix factorization based unmixing for principal component transformed hyperspectral data

Key words: Non-negative matrix factorization (NMF), Principal component analysis (PCA), Endmember, Hyperspectral

Corresponding author: Xui-rui Geng
E-mail: gengxr@sina.com
ORCID: http://orcid.org/0000-0003-0935-3753

Motivation

- Disadvantages of NMF:
  - NMF suffers from the large computational complexity for the popular multiplicative iteration rule;
  - NMF is sensitive to noise (outliers), and thus the corrupted data will make the results of NMF meaningless.

- Limitation of using PCA on NMF:

  the PCA transformed data will contain negative numbers, hindering the direct use of NMF's multiplicative iteration rule.

Background

- NMF

Given an $L \times M$ non-negative matrix $\boldsymbol{R}$ and a positive integer $p<L$, the task of non-negative matrix fraction is to find two non-negative matrices $\boldsymbol{E}_{L\times p}$ and $\boldsymbol{C}_{p\times M}$ such that

$$\boldsymbol{R} \approx \boldsymbol{E}\boldsymbol{C}.$$

The learning process for the multiplicative NMF is:

$$\boldsymbol{C} = \boldsymbol{C}. * \left(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{R}\right) ./ \left(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E}\boldsymbol{C}\right),$$

$$\boldsymbol{E} = \boldsymbol{E}. * \left(\boldsymbol{R}\boldsymbol{C}^{\mathrm{T}}\right) ./ \left(\boldsymbol{E}\boldsymbol{C}\boldsymbol{C}^{\mathrm{T}}\right).$$

Method

- Explore the impact of PCA to NMF
  - Impact of rotation

Let us denote $\widehat{R} = V^{\mathrm{T}}R$ and $\widehat{E} = V^{\mathrm{T}}E$ to be the rotated data and endmember matrix respectively, where V is an orthogonal matrix. Then we have

$$\begin{aligned}\widehat{C} &= \widehat{C}.*\left((V^{\mathrm{T}}E)^{\mathrm{T}}(V^{\mathrm{T}}R)\right)./\left((V^{\mathrm{T}}E)^{\mathrm{T}}V^{\mathrm{T}}E\widehat{C}\right)\\ &= \widehat{C}.*\left(E^{\mathrm{T}}VV^{\mathrm{T}}R\right)./(E^{\mathrm{T}}VV^{\mathrm{T}}E\widehat{C})\\ &= \widehat{C}.*(E^{\mathrm{T}}R)./(E^{\mathrm{T}}E\widehat{C}).\end{aligned}$$

Unchanged

$$\begin{aligned}\widehat{E} &= \widehat{E}.*(\widehat{R}C^{\mathrm{T}})./(\widehat{E}CC^{\mathrm{T}})\\ &= \widehat{E}.*(V^{\mathrm{T}}RC^{\mathrm{T}})./(\widehat{E}CC^{\mathrm{T}}).\end{aligned}$$

Changed

Method (Con'd)

- Explore the impact of PCA to NMF
  - Impact of translation

Suppose the data and endmembers contain negatives, we can select a vector $\boldsymbol{r}_0$, such that $(\boldsymbol{R} - r_0 1_M^{\mathrm{T}}) \geq 0$ and $(\boldsymbol{E} - r_0 1_p^{\mathrm{T}}) \geq 0$. Then the corresponding multiplicative updating rules becomes:

$$\boldsymbol{C} = \boldsymbol{C}. * \left((\boldsymbol{E} - r_0 1_p^{\mathrm{T}})^{\mathrm{T}} (\boldsymbol{R} - r_0 1_M^{\mathrm{T}})\right) ./ \left((\boldsymbol{E} - r_0 1_p^{\mathrm{T}})^{\mathrm{T}} (\boldsymbol{E} - r_0 1_p^{\mathrm{T}}) \boldsymbol{C}\right),$$

Changed

$$\boldsymbol{E} = (\boldsymbol{E} - r_0 1_p^{\mathrm{T}}). * \left((\boldsymbol{R} - r_0 1_M^{\mathrm{T}}) \boldsymbol{C}^{\mathrm{T}}\right) ./ \left((\boldsymbol{E} - r_0 1_p^{\mathrm{T}}) \boldsymbol{C}\boldsymbol{C}^{\mathrm{T}}\right) + r_0 1_p^{\mathrm{T}}.$$

Changed

Method (Con'd)

- PCNMF

Interestingly, based on our observation the maximum spectral angle between pixels in real hyperspectral data is mostly small (generally less than 45 degrees). In addition, the rotation operation will not change the spectral angles between data points. Both facts motivate us to apply the orthogonal procrustes (OP) technique to solve the non-negative problem of ***R*** and ***E*** in the PC space. That is to forcibly rotate all the data points into the first quadrant of the PC space, so that the multiplicitive update rule still works for ***E***.

Method (Con'd)

- PCNMF

Mathematically, the OP problem can be stated as follows:

$$\min_{Q} \; f(Q) = \|A - BQ\|_{\mathrm{F}}$$
$$\text{subject to} \quad Q^{\mathrm{T}}Q = QQ^{\mathrm{T}} = I.$$

And the optimal solution is:

$$Q = UW^{\mathrm{T}},$$

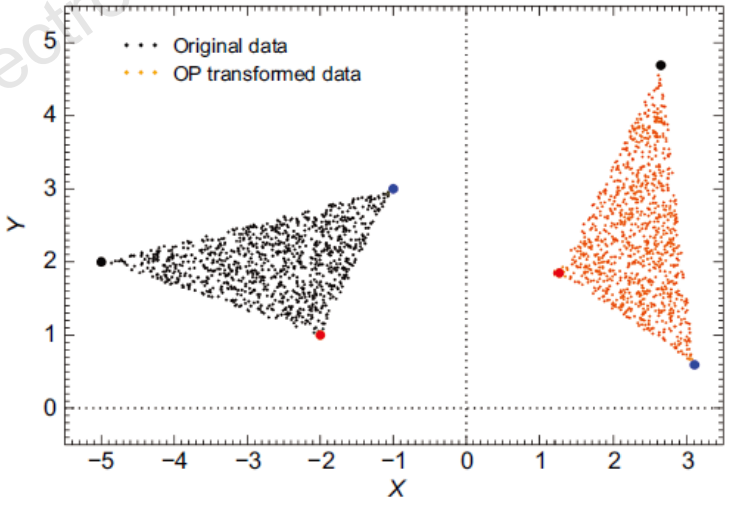


Fig. 1 Illustration of the orthogonal procrustes (OP) process in the 2D space

where the matrices ***U*** and ***W*** satisfy the singular value decomposition (SVD) equation $\boldsymbol{B}^{\mathrm{T}}\boldsymbol{A}=\boldsymbol{UDW}^{\mathrm{T}}$.

Method (Con'd)

- The pseudo-code of PCNMF is

Algorithm 1 Principal component non-negative matrix factorization

Input: data matrix, $\boldsymbol{R}_{L\times M}$; number of endmembers, p; maximum iteration number, maxiter; ASC weight, δ.

Output: endmember matrix, $\boldsymbol{E}_{L\times p}$; abundance matrix, $\boldsymbol{C}_{p\times M}$.

// PC-OP transformation

1: Calculate the autocorrelation matrix, $\boldsymbol{\Sigma}_{L\times L} = \boldsymbol{R}\boldsymbol{R}^{\mathrm{T}}/M$

2: Calculate the eigenvector matrix for $\boldsymbol{\Sigma}$ which satisfies $\boldsymbol{\Sigma} = \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^{\mathrm{T}}$, with the PC rotation matrix being $\boldsymbol{V} = [\boldsymbol{V}]_{:,1:(p-1)}$

3: Calculate the mean vector of the PC rotated data, $\bar{\boldsymbol{r}} = \boldsymbol{V}^{\mathrm{T}}\mathrm{mean}(\boldsymbol{R})$

4: Let $\boldsymbol{A}_{(p-1)\times 1} = [1, 1, \cdots, 1]^{\mathrm{T}}$, $\boldsymbol{B} = \bar{\boldsymbol{r}}$, and calculate the OP matrix $\boldsymbol{Q}$ based on Eq. (15)

5: Calculate the PC-OP transformed data, $\widetilde{\boldsymbol{R}} = (\boldsymbol{V}\boldsymbol{Q})^{\mathrm{T}}\boldsymbol{R}$

// NMF initialization

6: $\mathrm{index}_0 = \mathrm{FGDA}(\widetilde{\boldsymbol{R}}, p), \widetilde{\boldsymbol{E}}_0 = \widetilde{\boldsymbol{R}}_{:,\mathrm{index}_0}$

7: $\bar{\boldsymbol{R}} = \begin{bmatrix} \widetilde{\boldsymbol{R}} \\ \delta\mathbf{1}_M^{\mathrm{T}} \end{bmatrix}$, $\bar{\boldsymbol{E}}_0 = \begin{bmatrix} \widetilde{\boldsymbol{E}}_0 \\ \delta\mathbf{1}_p^{\mathrm{T}} \end{bmatrix}$

8: $\boldsymbol{C}_0 = \mathrm{lsqnonneg}(\bar{\boldsymbol{R}}, \bar{\boldsymbol{E}}_0)$

// NMF based endmember extraction

9: $[\boldsymbol{E}, \boldsymbol{C}] = \mathrm{NMF}(\bar{\boldsymbol{R}}, \bar{\boldsymbol{E}}_0, \boldsymbol{C}_0, \mathrm{maxiter})$

Major results

- Simulated data:

The spectra of three minerals (Alunite, Calcite, and Kaolinite) from U.S. Geological Survey (USGS) digital spectral library are selected as endmember signatures. Then 2000 mixture vectors are generated with abundance fractions following a Dirichlet distribution. In order to ensure that no pure pixel exist, all fractions are not allowed to be larger than 0.9.

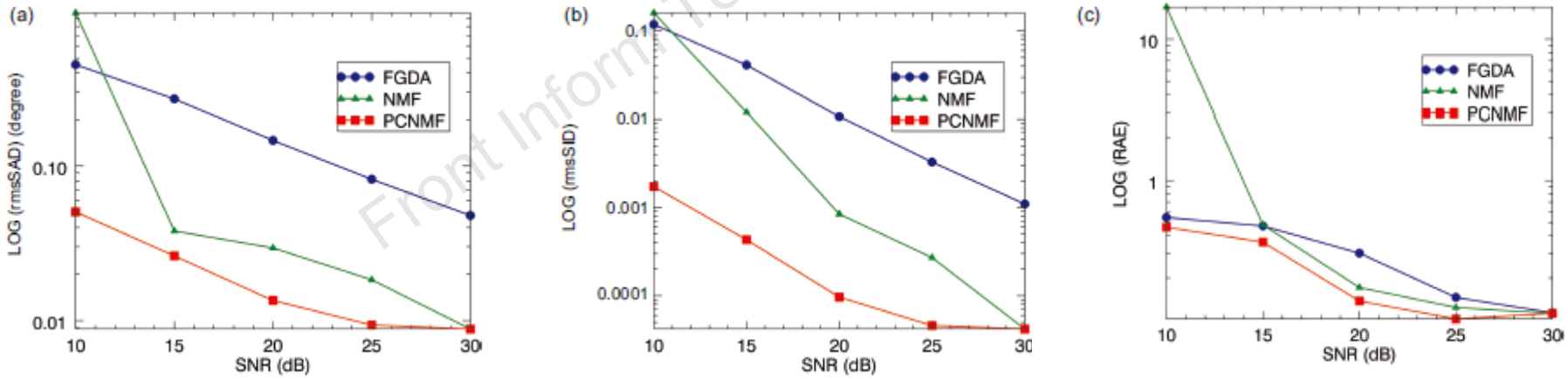
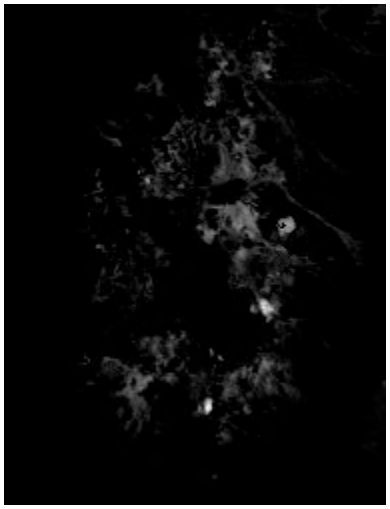


Fig. 4 The rmsSAD (a), rmsSID (b), and RAE (c) as functions of SNR for different methods with noise data

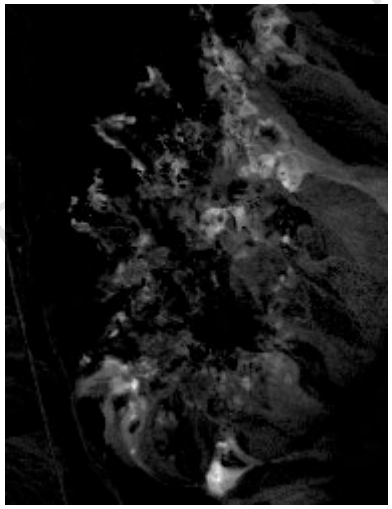
Major results (Con'd)

- Real data:

We evaluate the performance of PCNMF using the well-known AVIRIS Cuprite data set (191×250×187). The number of endmembers is set to 14.



Alunite



Kaolinite

Table 3 The SAD between USGS reference spectra and extracted endmembers by FGDA and PCNMF

No.	Substance	SAD (degree)	
		FGDA	PCNMF
1	Muscovite IL107	**5.3472**	5.9231
2	Desert Vanish GDS141	9.4029	**8.1041**
3	Alunite GDS84 Na03	4.6300	**3.9203**
4	Kaolin/Smect KLF508	**5.2682**	5.8254
5	Montmorillonite SWy-1	6.9298	**6.6978**
6	Kaolinite CM7	**4.8221**	4.8635
7	Buddingtonite NHB2301	5.4664	**5.2410**
8	Alunite GDS82 Na82	12.4593	**11.3928**
9	Montmorillonite+Illi CM42	6.2363	**5.6179**
10	Chalcedony CU91-6A	5.5653	**5.1812**
11	Alunite AL706	7.7288	**7.6880**
12	Montmorillonite+Illi CM37	4.8154	**4.7617**
13	Kaolin/Smect KLF511	**3.8877**	4.3025
14	Kaolin/Smect H89-FR-5	4.2887	**5.2605**
	Average	6.2034	**6.0557**

Bold text indicates the better SAD