Algorithms for Independent Low-Rank Matrix Analysis

Daichi Kitamura[1,*]

[1]SOKENDAI (The Graduate University for Advanced Studies), Japan
[*]d-kitamura@nii.ac.jp, http://d-kitamura.sakura.ne.jp/en/index_en.html

ABSTRACT

This document summarizes an algorithm for independent low-rank matrix analysis, which was proposed as determined rank-1 multichannel nonnegative matrix factorization in the following published papers:
Daichi Kitamura, Nobutaka Ono, Hiroshi Sawada, Hirokazu Kameoka, and Hiroshi Saruwatari, "Efficient multichannel nonnegative matrix factorization exploiting rank-1 spatial model," Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015), pp. 276–280, April 2015.
Daichi Kitamura, Nobutaka Ono, Hiroshi Sawada, Hirokazu Kameoka, and Hiroshi Saruwatari, "Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 24, no. 9, pp. 1626–1641, September, 2016 (open access).
Two detailed algorithms for implementation and some empirical knowledges for the use of them are described in this document.

1 Introduction

Nonnegative matrix factorization (NMF) [1]–[5] is a low-rank approximation (dimensionality reduction) with nonnegative constraint, and many applications using NMF has been proposed, e.g., audio signal processing, text mining, image recognition, and bioinformatics. In particular, audio source separation is one of the most popular and successive research area of NMF [6]–[8]. For audio signals, NMF extracts some specific spectral patterns (bases) and their time-varying gains (activations) in the observed signal. The source separation using NMF is achieved by clustering such decomposed nonnegative parts into each source, and this clustering may require some prior knowledge about the sources in the mixture (e.g., semi- and full-supervised NMF methods [9, 10]).

As a blind source separation technique, multichannel NMF (MNMF) was proposed [11, 12], which simultaneously estimates the NMF variables (source model) and a spatial mixing matrix. The estimated bases and activations are attributed to sourcewise spatial features. These methods are generalized by Sawada et al. [13] exploiting more flexible spatial model called full-rank spatial covariance [14]. In this method, similarly to the original MNMF, the estimated source model is clustered based on the estimated spatial model to achieve the separation. However, it is reported that these algorithms are sensitive to the initial values of variables, and the computational cost is larger than those of conventional blind source separation methods.

For a determined or overdetermined blind source separation, independent component analysis (ICA) [15] and its extension to the frequency domain (frequency-domain ICA: FDICA) [16]–[20] are popular and traditional techniques. These ICA-based methods estimate a demixing matrix (an inverse matrix of the mixing matrix) in the frequency domain. In particular, independent vector analysis (IVA) [21]–[23] achieves better and stable source separation performance compared with FDICA, and an auxiliary-function-based efficient update rules for IVA were developed [24].

In [25, 26], a new algorithm for blind source separation was proposed, which is called *determined rank-1 MNMF*. Determined rank-1 MNMF has the same model as the conventional MNMF [11], but the demixing matrix is estimated instead of the mixing matrix. This approach is similar to the traditional ICA-based blind source separation, and it is revealed that determined rank-1 MNMF is a natural extension of IVA in terms of the flexibility of the source model (although the assumed prior distributions in IVA and determined rank-1 MNMF are different). Namely, the vector source model in IVA is extended to the low-rank matrix source model using NMF decomposition in determined rank-1 MNMF. To clarify this issue, in this document, we have renamed determined rank-1 MNMF as *independent low-rank matrix analysis (ILRMA)*. The cost function in ILRMA consists of those in IVA and simple NMF based on Itakura–Saito divergence. Therefore, all the variables including demixing matrix and NMF variables can efficiently be optimized by an iterative projection proposed in [24] and auxiliary-function-based multiplicative update rules [4]. However, since these update rules derived in the paper [26] are described in scalar notations, it is difficult to implement ILRMA in an efficient way using matrix operations. In this document, the efficient algorithms for ILRMA using matrix notations are described in Sects. 2.2 and 2.3.

2 Algorithms for ILRMA

There are two models in ILRMA: *ILRMA without partitioning function* (hereafter referred to as ILRMA1) and *ILRMA with partitioning function* (hereafter referred to as ILRMA2), where the partitioning function is used for determining the optimal number of NMF bases for each source. Therefore, in ILRMA1, we must set the fixed number of bases for each source. In ILRMA2, we only set the total number of bases, and the algorithm adaptively estimates the partition of all the bases by optimizing the partitioning function. The detailed formulations of them can be found in [26].

2.1 Definitions

In this subsection, some definitions of variables are described as preliminaries. Note that the lowercase Italic characters denote scalars, lowercase bold Italic characters denote column vectors, uppercase bold Italic characters denote matrices, and uppercase Sans-serif characters denote the third-order tensors. Also, $^{\mathrm{T}}$ and $^{\mathrm{H}}$ denote simple transpose and Hermitian transpose of vectors or matrices, respectively.

- M: number of channels (microphones) whose index is m
- N: number of sources whose index is n, where we assume $M = N$ (overdetermined situation) in this document
- I: number of frequency bins whose index is i
- J: number of time frames whose index is j
- L: number of bases for each source whose index is l
- K: total number of bases for all the sources whose index is k
- $\boldsymbol{x}_{ij} = (x_{ij,1}\ x_{ij,2}\ \cdots\ x_{ij,M})^{\mathrm{T}}$: complex-valued observed (mixture) signals in time-frequency domain
- X: complex-valued $I \times J \times M$ tensor whose element is $x_{ij,m}$
- $\boldsymbol{y}_{ij} = (y_{ij,1}\ y_{ij,2}\ \cdots\ y_{ij,N})^{\mathrm{T}}$: complex-valued estimated (separated) signals in time-frequency domain
- Y: complex-valued $I \times J \times N$ tensor whose element is $y_{ij,n}$
- $\boldsymbol{W}_i = (\boldsymbol{w}_{i,1}\ \boldsymbol{w}_{i,2}\ \cdots\ \boldsymbol{w}_{i,N})^{\mathrm{H}}$: complex-valued $N \times M$ demixing matrix for the ith frequency bin, which leads $\boldsymbol{y}_{ij} = \boldsymbol{W}_i \boldsymbol{x}_{ij}$
- $\boldsymbol{w}_{i,n}$: complex-valued $M \times 1$ vector in $\boldsymbol{W}_i$ called demixing filter for the nth source, where $y_{ij,n} = \boldsymbol{w}_{i,n}^{\mathrm{H}} \boldsymbol{x}_{ij}$
- $\boldsymbol{e}_n$: $N \times 1$ vector with only the nth element equal to unity and the other elements equal to zeros
- P: nonnegative $I \times J \times N$ tensor that corresponds to the power spectrograms of estimated source signals
- $p_{ij,n}$: nonnegative element of P
- R: nonnegative $I \times J \times N$ tensor that corresponds to the model spectrograms (time-frequency variances) for all sources (low-rank approximation of P using NMF decomposition)
- $\boldsymbol{T}_n$: nonnegative $I \times L$ basis matrix for the nth source, which is used in ILRMA1
- $\boldsymbol{V}_n$: nonnegative $L \times J$ activation matrix for the nth source, which is used in ILRMA1
- $t_{il,n}$: nonnegative element of $\boldsymbol{T}_n$.
- $v_{lj,n}$: nonnegative element of $\boldsymbol{V}_n$.
- $\boldsymbol{T}$: nonnegative $I \times K$ basis matrix for all sources, which is used in ILRMA2
- $\boldsymbol{V}$: nonnegative $K \times J$ activation matrix for all source, which is used in ILRMA2
- t_{ik}: nonnegative element of $\boldsymbol{T}$
- v_{kj}: nonnegative element of $\boldsymbol{V}$
- $\boldsymbol{Z} = (\boldsymbol{z}_1,\ \boldsymbol{z}_2,\ \cdots,\ \boldsymbol{z}_N)^{\mathrm{T}}$: $N \times K$ matrix called partitioning function used in ILRMA2, where all the elements are in the range $[0, 1]$

- z_n: $K \times 1$ vector in $\boldsymbol{Z}$
- z_{nk}: element of $\boldsymbol{Z}$, where $\sum_n z_{nk} = 1$
- $\mathbf{1}^{(\text{size})}$: matrix of ones whose size is denoted as the superscript, e.g., $\mathbf{1}^{(I \times J)}$
- ε: machine epsilon
- $\hat{\boldsymbol{y}}_{ij,n}$: complex-valued and scale-fitted $M \times 1$ vector, which corresponds to the estimated nth source (spatial) image

Note that the third-order tensor with a subscript denotes the sliced matrix or the fiber vector in the original tensor [27]. For example, since X is an $I \times J \times M$ tensor, $\mathsf{X}_{i::}$ denotes the $J \times M$ sliced matrix in X, $\mathsf{X}_{:j:}$ denotes the $I \times M$ sliced matrix in X, and $\mathsf{X}_{::m}$ denotes the $I \times J$ sliced matrix in X. Also, $\mathsf{X}_{ij:}$, $\mathsf{X}_{i:m}$, and $\mathsf{X}_{:jm}$ denote the $M \times 1$, $J \times 1$, and $I \times 1$ fiber (column) vectors, respectively. In the algorithm description, $\circ$ and the quotient symbol for matrices denote the elementwise multiplication and division, respectively. In addition, the absolute value $|\cdot|$ and the dotted exponent for matrices (e.g., $\boldsymbol{X}^{.2}$) denote the elementwise absolute value and the elementwise exponent, respectively. For example, $\mathsf{P}_{::n} = |\mathsf{Y}_{::n}|^{.2}$. Moreover, the maximum operator $\max(\cdot, \cdot)$ returns a matrix with the larger elements taken from two inputs in each entry.

2.2 Algorithm for ILRMA1

Since ILRMA1 does not utilize the partitioning function $\boldsymbol{Z}$, the optimization is simply carried out by an alternative iteration of iterative-projection-based IVA update rule [24] and simple NMF update rules [3, 4]. Algorithm 1 shows a detailed process flow in ILRMA1. For the implementation, we have to note the corresponding scalars, vectors, and matrices variables. For example, when we update $\boldsymbol{w}_{i,n}$ at the lines 13 and 14 in Algorithm 1, the nth row in $\boldsymbol{W}_i$ must be replaced to the new $\boldsymbol{w}_{i,n}^{\mathrm{H}}$ before it is used in the line 17.

2.3 Algorithm for ILRMA2

In ILRMA2, we only set the total number of bases. The K bases are shared (distributed) to each source by the partitioning function $\boldsymbol{Z}$. The optimization of $\boldsymbol{Z}$ is also carried out by an NMF-like multiplicative update rule. Algorithm 2 shows a detailed process flow in ILRMA2.

2.4 Post process of ILRMA

The ICA-based source separation methods cannot determine scales of the estimated sources. Therefore, after the separation, a back-projection technique [28, 29] must be applied to the estimated signal $\boldsymbol{y}_{ij}$. This technique ideally restores the scales of the estimated sources to their observed amplitudes in $\boldsymbol{x}_{ij}$, where the inverse matrix of the demixing matrix is utilized for the restoration. Algorithm 3 shows the detailed process flow of the back-projection technique. The output signal $\hat{\boldsymbol{y}}_{ij,n}$ is a scale-fitted estimated signal of the nth source, where $\hat{\boldsymbol{y}}_{ij,n}$ is a multichannel estimated signal of the nth source ($M \times 1$ vector), which is often called the source image.

3 Empirical knowledges

In this section, we describe some empirical knowledges about the implementation and use of ILRMA.

- When the input multichannel signal $\boldsymbol{x}_{ij}$ is overdetermined ($M > N$), principal component analysis should be applied in advance for reducing its dimension so that $M = N$.
- When the input signal $\boldsymbol{x}_{ij}$ includes zeros, the zero division might occur in NMF update rules. The zeros in $\boldsymbol{x}_{ij}$ should be replaced with the machine epsilons. Also, we must avoid to put zeros as the initial values of $\boldsymbol{T}_n$, $\boldsymbol{V}_n$, $\boldsymbol{T}$, $\boldsymbol{V}$, and $\boldsymbol{Z}$ because the multiplicative update rules do not work for the zero elements.
- For the initial values of $\boldsymbol{W}_i$, the identity matrix is preferable.
- When the number of time frames J is small, namely, the observed signal is too short, the matrix $\boldsymbol{W}_i \boldsymbol{U}_{i,n}$ at the lines 13 in Algorithm 1 and 25 in Algorithm 2 tends to be a singular matrix, and its inversion cannot be calculated. In such a case, pseudo-inverse should be used, but the computational cost greatly increases in many languages (e.g., Matlab). We can also avoid the inversion error by adding small values to the diagonal elements in the singular matrix.
- As a normalization coefficient, we described the root mean powers of estimated sources at the lines 20 in Algorithm 1 and 33 in Algorithm 2. However, any kind of coefficients can be used here. This normalization is processed to avoid a divergence of variables $\boldsymbol{W}_i$ or R to infinity. If the computational cost should be reduced, these normalization processes can be omitted although the algorithms become unstable in some cases.

Algorithm 1: ILRMA1

input : observed multichannel complex-valued signals $\boldsymbol{x}_{ij}$

output: estimated complex-valued sources $\boldsymbol{y}_{ij}$

Initialize $\boldsymbol{W}_i$ with identity matrix or complex-valued random values and $\boldsymbol{T}_n$ and $\boldsymbol{V}_n$ with nonnegative random values;

Calculate $\boldsymbol{y}_{ij} = \boldsymbol{W}_i \boldsymbol{x}_{ij}$ for all i and j; `// Initial estimated sources`

Calculate $\mathsf{P}_{::n} = |\mathsf{Y}_{::n}|^{\cdot 2}$ for all n; `// Initial power spectrograms of estimated sources`

Calculate $\mathsf{R}_{::n} = \boldsymbol{T}_n \boldsymbol{V}_n$ for all n; `// Initial model spectrograms`

repeat

    for $n = 1$ **to** N **do**

        $\boldsymbol{T}_n \leftarrow \max\left(\boldsymbol{T}_n \circ \left[\frac{(\mathsf{P}_{::n} \circ \mathsf{R}_{::n}^{\cdot -2}) \boldsymbol{V}_n^{\mathrm{T}}}{\mathsf{R}_{::n}^{\cdot -1} \boldsymbol{V}_n^{\mathrm{T}}}\right]^{\cdot \frac{1}{2}}, \varepsilon\right)$; `// Update of basis matrix`

        $\mathsf{R}_{::n} = \boldsymbol{T}_n \boldsymbol{V}_n$; `// New model spectrograms`

        $\boldsymbol{V}_n \leftarrow \max\left(\boldsymbol{V}_n \circ \left[\frac{\boldsymbol{T}_n^{\mathrm{T}} (\mathsf{P}_{::n} \circ \mathsf{R}_{::n}^{\cdot -2})}{\boldsymbol{T}_n^{\mathrm{T}} \mathsf{R}_{::n}^{\cdot -1}}\right]^{\cdot \frac{1}{2}}, \varepsilon\right)$; `// Update of activation matrix`

        $\mathsf{R}_{::n} = \boldsymbol{T}_n \boldsymbol{V}_n$; `// New model spectrograms`

        for $i = 1$ **to** I **do**

            $\boldsymbol{U}_{i,n} = \frac{1}{J}\left\{\mathsf{X}_{i::}^{\mathrm{H}}\left[\mathsf{X}_{i::} \circ \left(\mathsf{R}_{i:n}^{\cdot -1} \mathbf{1}^{(1 \times M)}\right)\right]\right\}^{\mathrm{T}}$; `//` $\mathsf{X}_{i::}$ `is` $J \times M$ `matrix,` $\mathsf{R}_{i:n}$ `is` $J \times 1$ `vector,` $\boldsymbol{U}_{i,n}$ `is` $M \times M$ `matrix`

            $\boldsymbol{w}_{i,n} \leftarrow (\boldsymbol{W}_i \boldsymbol{U}_{i,n})^{-1} \boldsymbol{e}_n$; `// Update of demixing filter`

            $\boldsymbol{w}_{i,n} \leftarrow \boldsymbol{w}_{i,n} (\boldsymbol{w}_{i,n}^{\mathrm{H}} \boldsymbol{U}_{i,n} \boldsymbol{w}_{i,n})^{-\frac{1}{2}}$; `// Normalization of demixing filter`

        end

    end

    Calculate $\boldsymbol{y}_{ij} = \boldsymbol{W}_i \boldsymbol{x}_{ij}$ for all i and j; `// New estimated sources`

    Calculate $\mathsf{P}_{::n} = |\mathsf{Y}_{::n}|^{\cdot 2}$ for all n; `// New power spectrograms of estimated sources`

    for $n = 1$ **to** N **do**

        $\lambda_n = \sqrt{\frac{1}{IJ} \sum_{i,j} p_{ij,n}}$; `// Normalization coefficient`

        for $i = 1$ **to** I **do**

            $\boldsymbol{w}_{i,n} \leftarrow \boldsymbol{w}_{i,n} \lambda_n^{-1}$; `// Normalization of demixing filter`

        end

        $\mathsf{P}_{::n} \leftarrow \mathsf{P}_{::n} \lambda_n^{-2}$; `// Normalization of separated power spectrogram`

        $\mathsf{R}_{::n} \leftarrow \mathsf{R}_{::n} \lambda_n^{-2}$; `// Normalization of model spectrogram`

        $\boldsymbol{T}_n \leftarrow \boldsymbol{T}_n \lambda_n^{-2}$; `// Normalization of basis matrix`

    end

until *converge*;

Algorithm 2: ILRMA2

input : observed multichannel complex-valued signals $\boldsymbol{x}_{ij}$

output: estimated complex-valued sources $\boldsymbol{y}_{ij}$

Initialize $\boldsymbol{W}_i$ with identity matrix or complex-valued random values, $\boldsymbol{T}$ and $\boldsymbol{V}$ with nonnegative random values, and $\boldsymbol{Z}$ with random values in range [0, 1];

$\boldsymbol{Z} \leftarrow \boldsymbol{Z} \circ \left(\mathbf{1}^{(N\times N)}\boldsymbol{Z}\right)^{.-1}$; // Ensuring $\sum_n z_{nk} = 1$

Calculate $\boldsymbol{y}_{ij} = \boldsymbol{W}_i \boldsymbol{x}_{ij}$ for all i and j; // Initial estimated sources

Calculate $\mathsf{P}_{::n} = |\mathsf{Y}_{::n}|^{.2}$ for all n; // Initial power spectrograms of estimated sources

Calculate $\mathsf{R}_{::n} = \left[\left(\mathbf{1}^{(I\times 1)}\boldsymbol{z}_n^{\mathrm{T}}\right) \circ \boldsymbol{T}\right]\boldsymbol{V}$ for all n; // Initial model spectrograms

repeat

    for $n = 1$ **to** N **do**

        $\boldsymbol{a}_n = \left(\frac{\{[\boldsymbol{T}^{\mathrm{T}}(\mathsf{P}_{::n} \circ \mathsf{R}_{::n}^{.-2})] \circ \boldsymbol{V}\}\mathbf{1}^{(J\times 1)}}{[(\boldsymbol{T}^{\mathrm{T}}\mathsf{R}_{::n}^{.-1}) \circ \boldsymbol{V}]\mathbf{1}^{(J\times 1)}}\right)^{.\frac{1}{2}}$; // $\boldsymbol{a}_m$ is $K \times 1$ vector

    end

    $\boldsymbol{Z} \leftarrow \max(\boldsymbol{Z} \circ \boldsymbol{A}, \varepsilon)$, where $\boldsymbol{A} = (\boldsymbol{a}_1 \ \cdots \ \boldsymbol{a}_N)^{\mathrm{T}}$; // Update of partitioning function, $\boldsymbol{A}$ is $N \times K$ matrix

    $\boldsymbol{Z} \leftarrow \boldsymbol{Z} \circ \left(\mathbf{1}^{(N\times N)}\boldsymbol{Z}\right)^{.-1}$; // Ensuring $\sum_n z_{nk} = 1$

    Calculate $\mathsf{R}_{::n} = \left[\left(\mathbf{1}^{(I\times 1)}\boldsymbol{z}_n^{\mathrm{T}}\right) \circ \boldsymbol{T}\right]\boldsymbol{V}$ for all n; // New model spectrogram

    for $i = 1$ **to** I **do**

        $\boldsymbol{b}_i = \left(\frac{\{[\boldsymbol{V}(\mathsf{P}_{i::} \circ \mathsf{R}_{i::}^{.-2})] \circ \boldsymbol{Z}^{\mathrm{T}}\}\mathbf{1}^{(N\times 1)}}{[(\boldsymbol{V}\mathsf{R}_{i::}^{.-1}) \circ \boldsymbol{Z}^{\mathrm{T}}]\mathbf{1}^{(N\times 1)}}\right)^{.\frac{1}{2}}$; // $\boldsymbol{b}_i$ is $K \times 1$ vector

    end

    $\boldsymbol{T} \leftarrow \max(\boldsymbol{T} \circ \boldsymbol{B}, \varepsilon)$, where $\boldsymbol{B} = (\boldsymbol{b}_1 \ \cdots \ \boldsymbol{b}_I)^{\mathrm{T}}$; // Update of basis matrix, $\boldsymbol{B}$ is $I \times K$ matrix

    Calculate $\mathsf{R}_{::n} = \left[\left(\mathbf{1}^{(I\times 1)}\boldsymbol{z}_n^{\mathrm{T}}\right) \circ \boldsymbol{T}\right]\boldsymbol{V}$ for all n; // New model spectrogram

    for $j = 1$ **to** J **do**

        $\boldsymbol{c}_j = \left(\frac{\left\{\left[\boldsymbol{T}^{\mathrm{T}}\left(\mathsf{P}_{:j:} \circ \mathsf{R}_{:j:}^{.-2}\right)\right] \circ \boldsymbol{Z}^{\mathrm{T}}\right\}\mathbf{1}^{(N\times 1)}}{\left[\left(\boldsymbol{T}^{\mathrm{T}}\mathsf{R}_{:j:}^{.-1}\right) \circ \boldsymbol{Z}^{\mathrm{T}}\right]\mathbf{1}^{(N\times 1)}}\right)^{.\frac{1}{2}}$; // $\boldsymbol{c}_j$ is $K \times 1$ vector

    end

    $\boldsymbol{V} \leftarrow \max(\boldsymbol{V} \circ \boldsymbol{C}, \varepsilon)$, where $\boldsymbol{C} = (\boldsymbol{c}_1 \ \cdots \ \boldsymbol{c}_J)$; // Update of activation matrix, $\boldsymbol{C}$ is $K \times J$ matrix

    Calculate $\mathsf{R}_{::n} = \left[\left(\mathbf{1}^{(I\times 1)}\boldsymbol{z}_n^{\mathrm{T}}\right) \circ \boldsymbol{T}\right]\boldsymbol{V}$ for all n; // New model spectrogram

    for $n = 1$ **to** N **do**

        for $i = 1$ **to** I **do**

            $\boldsymbol{U}_{i,n} = \frac{1}{J}\left\{\mathsf{X}_{i::}^{\mathrm{H}}\left[\mathsf{X}_{i::} \circ \left(\mathsf{R}_{i:n}^{.-1}\mathbf{1}^{(1\times M)}\right)\right]\right\}^{\mathrm{T}}$; // $\mathsf{X}_{i::}$ is $J \times M$ matrix, $\mathsf{R}_{i:n}$ is $J \times 1$ vector, $\boldsymbol{U}_{i,n}$ is $M \times M$ matrix

            $\boldsymbol{w}_{i,n} \leftarrow (\boldsymbol{W}_i \boldsymbol{U}_{i,n})^{-1}\boldsymbol{e}_n$; // Update of demixing filter

            $\boldsymbol{w}_{i,n} \leftarrow \boldsymbol{w}_{i,n}(\boldsymbol{w}_{i,n}^{\mathrm{H}}\boldsymbol{U}_{i,n}\boldsymbol{w}_{i,n})^{-\frac{1}{2}}$; // Normalization of demixing filter

        end

    end

    Calculate $\boldsymbol{y}_{ij} = \boldsymbol{W}_i \boldsymbol{x}_{ij}$ for all i and j; // New estimated sources

    Calculate $\mathsf{P}_{::n} = |\mathsf{Y}_{::n}|^{.2}$ for all n; // New power spectrograms of estimated sources

    for $n = 1$ **to** N **do**

        $\lambda_n = \sqrt{\frac{1}{IJ}\sum_{i,j} p_{ij,n}}$; // Normalization coefficient

        for $i = 1$ **to** I **do**

            $\boldsymbol{w}_{i,n} \leftarrow \boldsymbol{w}_{i,n}\lambda_n^{-1}$; // Normalization of demixing filter

        end

        $\mathsf{P}_{::n} \leftarrow \mathsf{P}_{::n}\lambda_n^{-2}$; // Normalization of separated power spectrogram

        $\mathsf{R}_{::n} \leftarrow \mathsf{R}_{::n}\lambda_n^{-2}$; // Normalization of model spectrogram

    end

    Calculate $t_{ik} \leftarrow t_{ik}\sum_n z_{nk}\lambda_n^{-2}$ for all i and k; // Normalization of basis matrix

    Calculate $z_{nk} \leftarrow z_{nk}\frac{\lambda_n^{-2}}{\sum_{n'} z_{n'k}\lambda_{n'}^{-2}}$ for all n and k; // Normalization of partitioning function

until *converge*;

Algorithm 3: Back-projection technique

input : estimated multichannel complex-valued sources $\boldsymbol{y}_{ij}$
output: scale-fitted estimated source images $\hat{\boldsymbol{y}}_{ij,n}$

for $i = 1$ **to** I **do**
  for $j = 1$ **to** J **do**
    for $n = 1$ **to** N **do**
      $\hat{\boldsymbol{y}}_{ij,n} = \boldsymbol{W}_i^{-1}\left(\boldsymbol{e}_n \circ \boldsymbol{y}_{ij}\right)$; // $M \times 1$ vector
    end
  end
end

- The separation result depends on the initial values. The error bars sometimes become large especially for the mixture of speech signals (see the experimental results in the paper [26]). The dependence on initial values becomes strong when the number of bases increases.

- Similarly to FDICA or IVA, ILRMA utilizes an assumption about the mixing system called the rank-1 spatial model (assumption of instantaneous mixture in the frequency domain). On the basis of this assumption, the reverberation time should be much shorter than the length of window function used in STFT. Also, the sources must spatially be stable (not moving) and must be assumed as point sources.

4 Conclusion

This document summarizes the algorithms for ILRMA. There are two models in ILRMA depending on the presence of partitioning function. I hope this document will help your happy source separation life.

5 Acknowledgements

I would like to thank Yoshiaki Bando and Yoshiki Mitsui for kindly implementing and checking the algorithms described in this document.

References

[1] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[2] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Proc. Advances in Neural Information Processing Systems*, vol. 13, pp. 556–562, 2000.

[3] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative matrix factorization with the Itakura–Saito divergence. With application to music analysis," *Neural Computation*, vol. 21, no. 3, pp. 793–830, 2009.

[4] M. Nakano, H. Kameoka, J. L. Roux, Y. Kitano, N. Ono, and S. Sagayama, "Convergence-guaranteed multiplicative algorithms for nonnegative matrix factorization with β-divergence," *Proc. IEEE International Workshop on Machine Learning for Signal Processing*, pp. 283–288, 2010.

[5] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari, "Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation," *John Wiley & Sons*, 2009.

[6] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[7] A. Ozerov, C. Févotte, and M. Charbit, "Factorial scaled hidden Markov model for polyphonic audio representation and source separation," *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 121–124, 2009.

[8] H. Kameoka, M. Nakano, K. Ochiai, Y. Imoto, K. Kashino, and S. Sagayama, "Constrained and regularized variants of non-negative matrix factorization incorporating music-specific constraints," *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 5365–5368, 2012.

[9] P. Smaragdis, B. Raj, and M. Shashanka, "Supervised and semi-supervised separation of sounds from single-channel mixtures," *Proc. International Conference on Independent Component Analysis and Signal Separation*, pp. 414–421, 2007.

[10] D. Kitamura, H. Saruwatari, K. Yagi, K. Shikano, Y. Takahashi, and K. Kondo, "Music signal separation based on supervised nonnegative matrix factorization with orthogonality and maximum-divergence penalties," *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, vol. E97-A, no. 5, pp. 1113–1118, 2014.

[11] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 550–563, 2010.

[12] H. Kameoka, T. Yoshioka, M. Hamamura, J. Le Roux, and K. Kashino, "Statistical model of speech signals based on composite autoregressive system with application to blind source separation," *Proc. International Conference on Latent Variable Analysis and Signal Separation*, pp. 245–253, 2010.

[13] H. Sawada, H. Kameoka, S. Araki, and N. Ueda, "Multichannel extensions of non-negative matrix factorization with complex-valued data," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 971–982, 2013.

[14] N. Q. K. Duong, E. Vincent, and R. Gribonval, "Under-determined reverberant audio source separation using a full-rank spatial covariance model," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1830–1840, 2010.

[15] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.

[16] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, pp. 21–34, 1998.

[17] S. Araki, R. Mukai, S. Makino, T. Nishikawa, and H. Saruwatari, "The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech," *IEEE Trans. Speech and Audio Processing*, vol. 11, no. 2, pp. 109–116, 2003.

[18] H. Sawada, R. Mukai, S. Araki, and S. Makino, "Convolutive blind source separation for more than two sources in the frequency domain," *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. Ⅲ-885–Ⅱ I-888, 2004.

[19] H. Buchner, R. Aichner, and W. Kellerman, "A generalization of blind source separation algorithms for convolutive mixtures based on second order statistics," *IEEE Trans. Speech and Audio Processing*, vol. 13, no. 1, pp. 120–134, 2005.

[20] H. Saruwatari, T. Kawamura, T. Nishikawa, A. Lee, and K. Shikano, "Blind source separation based on a fast-convergence algorithm combining ICA and beamforming," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 2, pp. 666–678, 2006.

[21] T. Kim, T. Eltoft, and T.-W. Lee, "Independent vector analysis: an extension of ICA to multivariate components," *Proc. International Conference on Independent Component Analysis and Blind Source Separation*, pp. 165–172, 2006.

[22] A. Hiroe, "Solution of permutation problem in frequency domain ICA using multivariate probability density functions," *Proc. International Conference on Independent Component Analysis and Blind Source Separation*, pp. 601–608, 2006.

[23] T. Kim, H. T. Attias, S.-Y. Lee, and T.-W. Lee, "Blind source separation exploiting higher-order frequency dependencies," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 70–79, 2007.

[24] N. Ono, "Stable and fast update rules for independent vector analysis based on auxiliary function technique," *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 189–192, 2011.

[25] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Efficient multichannel nonnegative matrix factorization exploiting rank-1 spatial model," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 276–280, April, 2015.

[26] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization," *IEEE/ACM Trans. Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1626–1641, September, 2016.

[27] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[28] N. Murata, S. Ikeda, and A. Ziehe, "An approach to blind source separation based on temporal structure of speech signals," *Neurocomputing*, vol. 41, no. 1–4, pp. 1–24, 2001.

[29] K. Matsuoka and S. Nakashima, "Minimal distortion principle for blind source separation," *Proc. International Conference on Independent Component Analysis and Blind Signal Separation*, pp. 722–727, 2001.