MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Non-negative Matrix Factor Deconvolution; Extracation of Multiple Sound Sources from Monophonic Inputs

Paris Smaragdis

TR2004-104 September 2004

Abstract

In this paper we present an extension to the Non-Negative Matrix Factorization algorithm which is capable of identifying components with temporal structure. We demonstrate the use of this algorithm in the magnitude spectrum domain, where we employ it to perform extraction of multiple sound objects from a single channel auditory scene.

International Congress on Independent Component Analysis and Blind Signal Separation (ICA)

Non-negative Matrix Factor Deconvolution; Extraction of Multiple Sound Sources from Monophonic Inputs

Paris Smaragdis

Mitsubishi Electric Research Laboratories
201 Broadway, Cambridge MA, 02139, USA
paris@merl.com

Abstract. In this paper we present an extension to the Non-Negative Matrix Factorization algorithm which is capable of identifying components with temporal structure. We demonstrate the use of this algorithm in the magnitude spectrum domain, where we employ it to perform extraction of multiple sound objects from a single channel auditory scene.

1 Introduction

Non-Negative Matrix Factorization (NMF), was introduced as a concept independently by Paatero (1997) as the Positive Matrix Factorization, and by Lee and Seung (1999) who also proposed some very efficient algorithms for its computation. Since its inception NMF has been applied successfully to a variety of problems despite a hazy statistical underpinning. In this paper we will introduce an extension of NMF for time series, which is useful for problems akin to source separation for single channel inputs.

2 Non-negative Matrix Factorization

The original formulation of NMF is defined as follows. Starting with a non-negative $M \times N$ matrix $\mathbf{V} \in \mathbb{R}^{\geq 0, M \times N}$ the goal is to approximate it as a product of two non-negative matrices $\mathbf{W} \in \mathbb{R}^{\geq 0, M \times R}$ and $\mathbf{H} \in \mathbb{R}^{\geq 0, R \times N}$ where $R \leq M$, such that we minimize the error of reconstruction of $\mathbf{V}$ by $\mathbf{W} \cdot \mathbf{H}$. The success of the reconstruction can be measured using a variety of cost functions, in this paper we will use a cost function introduced by Lee and Seung (1999):

$$D = \left\| \mathbf{V} \otimes \ln(\frac{\mathbf{V}}{\mathbf{W} \cdot \mathbf{H}}) - \mathbf{V} + \mathbf{W} \cdot \mathbf{H} \right\|_F \tag{1}$$

where $\| \cdot \|_F$ is the Frobenius norm and $\otimes$ is the Hadamard product (an element-wise multiplication); the division is also element-wise. Lee and Seung (2000) also introduced an efficient multiplicative update algorithm to optimize this function without the need for constraints to enforce non-negativity:

$$\mathbf{H} = \mathbf{H} \otimes \frac{\mathbf{W}^\top \cdot \frac{\mathbf{V}}{\mathbf{W} \cdot \mathbf{H}}}{\mathbf{W}^\top \cdot \mathbf{1}}, \quad \mathbf{W} = \mathbf{W} \otimes \frac{\frac{\mathbf{V}}{\mathbf{W} \cdot \mathbf{H}} \cdot \mathbf{H}^\top}{\mathbf{1} \cdot \mathbf{H}^\top} \tag{2}$$

C.G. Puntonet and A. Prieto (Eds.): ICA 2004, LNCS 3195, pp. 494–499, 2004.

where $\mathbf{1}$ is a $M \times N$ matrix with all its elements set to unity, and the divisions are again element-wise. The variable R corresponds to the number of basis functions to extract. It is usually set to a small number so that NMF results into a low-rank approximation.

2.1 NMF for Sound Object Extraction

It has been shown (Casey and Westner 2000, Smaragdis 2001) that sequentially applying PCA and ICA on magnitude short-time spectra results in decompositions which permits extraction of multiple simple sounds from single-channel inputs. A similar NMF formulation is developed here. Consider a sound scene $s(t)$, and its short-time Fourier transform packed into a $M \times N$ matrix:

$$\mathbf{F} = DFT \begin{bmatrix} s(t_1) & s(t_2) & & s(t_N) \\ \vdots & \vdots & \dots & \vdots \\ s(t_1 + M - 1) & s(t_2 + M - 1) & & s(t_N + M - 1) \end{bmatrix} \quad (3)$$

where M is the DFT size and N the overall number of frames computed[1]. From the matrix $\mathbf{F} \in \mathbb{R}^{M \times N}$ we can extract the magnitude of the transform $\mathbf{V} = |\mathbf{F}|, \mathbf{V} \in \mathbb{R}^{\geq 0, M \times N}$ and then apply NMF on it. To better understand the point of this operation consider the spectrogram in figure 1.

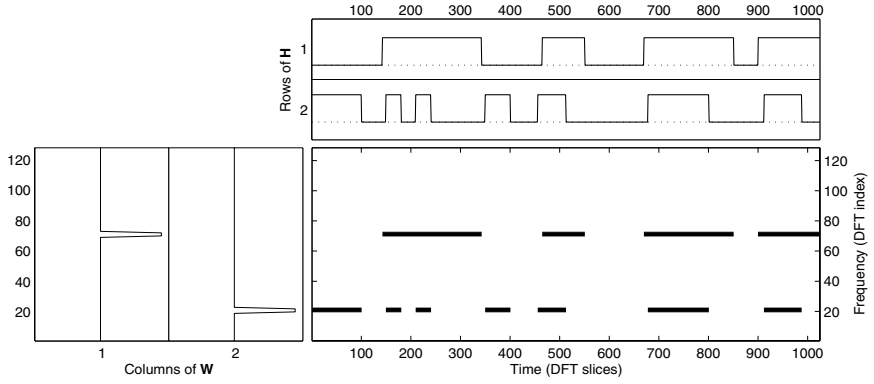

Fig. 1. NMF on spectrograms. The lower right plot is the input magnitude spectrogram, it represents two sinusoids with randomly gated amplitudes. The two columns of $\mathbf{W}$, interpreted as spectral bases, are shown in the leftmost plot. The rows of $\mathbf{H}$, depicted at the top plot, are the time weights corresponding to the two spectral bases.

It it easily seen that this spectrogram defines a scene that it composed out of sinusoids of two frequencies beeping in and out in some random manner.

[1] Ideally we would also apply a window function to the input sound to improve the spectral estimation. Since it this isn't a crucial addition to the process, we omit it for notational simplicity.

Applying a two-component NMF on this signal we obtain the two factors $\mathbf{W}$ and $\mathbf{H}$ also shown in figure 1. If we examine the two columns of $\mathbf{W}$, shown at the leftmost plots of the figure, we notice that they have energy only at the two frequencies that are present in the input spectrogram. We can interpret these two columns as basis functions for the spectra contained in the spectrogram. Likewise the rows of $\mathbf{H}$, shown at the top of the figure, only have energy at the time points where the two sinusoids do. We can interpret the rows of $\mathbf{H}$ as the weights of the spectral bases at each time. The bases and the weights have a one-to-one correspondence. The first basis describes the spectrum of one of the sinusoids and the first weight vector describes its time envelope. Likewise the other sinusoid is described in both time and frequency by the set of the second basis and second weight vector. In effect we can say that we have performed a rudimentary sound scene description. Although we presented a simplistic scenario this method is powerful enough to dissect even a piece of complex piano music to a set of weights and spectral bases describing each note played and its position in time, effectively performing musical transcription (Smaragdis 2003).

3 Non-negative Matrix Factor Deconvolution

The process we described above works well for many audio tasks. It is however a weak model since it does not take into account the relative positions of each spectrum thereby discarding temporal information. In this section we will introduce an extended version of NMF which deals with this issue. In the previous section we used the model $\mathbf{V} \approx \mathbf{W} \cdot \mathbf{H}$. In this section we will extend it to:

$$\mathbf{V} \approx \sum_{t=0}^{T-1} \mathbf{W}_t \cdot \overset{t\rightarrow}{\mathbf{H}} \tag{4}$$

where $\mathbf{V} \in \mathbb{R}^{\geq 0, M \times N}$ is the input we wish to decompose, and $\mathbf{W}_t \in \mathbb{R}^{\geq 0, M \times R}$ and $\mathbf{H} \in \mathbb{R}^{\geq 0, R \times N}$ are the bases and weights matrices. The $\overset{i\rightarrow}{(\cdot)}$ operator shifts the columns of its argument by i spots to the right. So that:

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}, \overset{0\rightarrow}{\mathbf{A}} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}, \overset{1\rightarrow}{\mathbf{A}} = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 0 & 5 & 6 & 7 \end{bmatrix}, \overset{2\rightarrow}{\mathbf{A}} = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 0 & 0 & 5 & 6 \end{bmatrix}, \ldots \tag{5}$$

The leftmost columns of the matrix are appropriately set to zero so as to maintain the original size of the input. Likewise we define the inverse operation $\overset{\leftarrow i}{(\cdot)}$, which shifts columns to the left.

Just as before our objective is to find a set of $\mathbf{W}_t$ and a $\mathbf{H}$ to approximate $\mathbf{V}$ as best as possible. We set $\mathbf{\Lambda} = \sum_{t=0}^{T-1} \mathbf{W}_t \cdot \overset{t\rightarrow}{\mathbf{H}}$ and define the cost function:

$$D = \left\| \mathbf{V} \otimes \ln(\frac{\mathbf{V}}{\mathbf{\Lambda}}) - \mathbf{V} + \mathbf{\Lambda} \right\|_F \tag{6}$$

To optimize this model we can use a strategy akin to the one presented above, only this time we will have to optimize more than just two matrices. The update

rules for this case will be the same as when performing NMF for each iteration of t, plus some shifting to appropriately line up the arguments:

$$\mathbf{H} = \mathbf{H} \otimes \frac{\mathbf{W}_t^\top \cdot \overset{\leftarrow t}{\left[\frac{\mathbf{V}}{\mathbf{\Lambda}}\right]}}{\mathbf{W}_t^\top \cdot \mathbf{1}} \quad \text{and} \quad \mathbf{W}_t = \mathbf{W}_t \otimes \frac{\frac{\mathbf{V}}{\mathbf{\Lambda}} \cdot \overset{t\rightarrow}{\mathbf{H}}^\top}{\mathbf{1} \cdot \overset{t\rightarrow}{\mathbf{H}}^\top}, \quad \forall t \in [0...T-1] \tag{7}$$

In every training iteration for each t we update $\mathbf{H}$ and each $\mathbf{W}_t$. That way we can optimize the factors in parallel and account for their interplay. In complex cases it is often useful to average the updates of $\mathbf{H}$ over all t's. Due to the rapid convergence properties of the multiplicative rules there is the danger that $\mathbf{H}$ has been more influenced by the last $\mathbf{W}_t$ used for its update, rather than the entire ensemble of $\mathbf{W}_t$. To gain some intuition on the form of the factors $\mathbf{W}_t$ and $\mathbf{H}$, consider the data in figure 2.

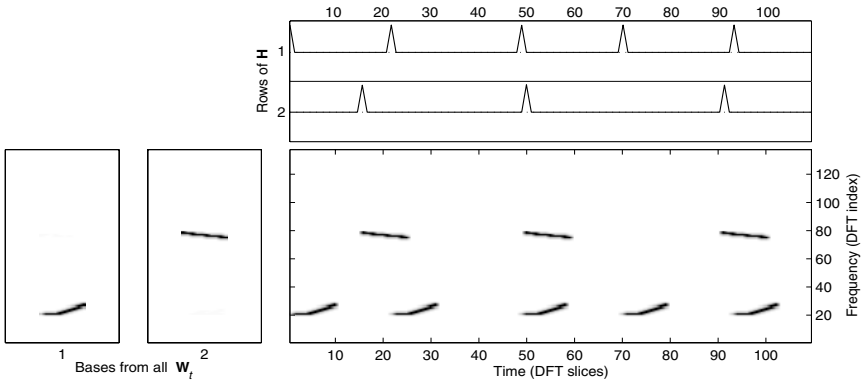

Fig. 2. A spectrogram and the extracted NMFD bases and weights. The lower right plot is the magnitude spectrogram that we used as an input to NMFD. The two leftmost plots are derived from $\mathbf{W}$, and are interpreted as temporal-spectral bases. The rows of $\mathbf{H}$, depicted at the top plot, are the time weights corresponding to the two bases. Note that the leftmost plots have been zero-padded in these figures from left and right so as to appear in the same scale as the input plot.

Just like the previous example the scene contains two randomly repeating elements, however they exhibit a temporal structure which cannot be expressed by spectral bases spanning a single time unit. We perform a two-component NMFD with $T = 10$. This results into a $\mathbf{H}$ and T $\mathbf{W}_t$ matrices of size $M \times 2$. The n^{th} column of the t^{th} $\mathbf{W}_t$ matrix is the n^{th} basis offset by t spots in the left-right dimension (time in our case). In other words the $\mathbf{W}_t$ matrices contain bases that extend in both dimensions of the input. $\mathbf{H}$, like in regular NMF, holds the weights of these functions. Examining figure 2 we see that the bases in $\mathbf{W}_t$ contain the finer temporal information in the present patterns, while $\mathbf{H}$ localizes them in time.

3.1 NMFD for Sound Object Extraction

Using the above formulation of NMFD we analyze a sound snippet which contains a set of drum sounds. In this example the drum sounds exhibit some overlap at both time and frequency. The input was sampled at 11.025 Hz and analyzed with 256-point DFTs which were overlapping by 128-points. A hanning window was applied to the input to improve the spectral estimate. NMFD was performed for 3 basis functions each with a time extend of 10 DFT frames ($R = 3$ and $T = 10$). The results are shown in figure 3. There are three types of drum sounds present into the scece; four instances of the bass drum (the low frequency element), two instances of a snare drum (the two loud wideband bursts), and the hi-hat the repeating high-band burst. Upon analysis we extract a set of spectral/temporal basis functions from $\mathbf{W}_t$. The weights from $\mathbf{H}$ show us how these bases are placed in time. Examining the bases we see that they have encapsulated the short-time spectral evolution of each drum. For example the second basis has adapted to the bass drum structure. Note how the main frequency of the basis drops with time and is preceded from a wide-band element just like the bass drum sound. Likewise the snare drum basis is wide-band with denser energy at the mid-frequencies, and the hi-hat basis is mostly high-band.

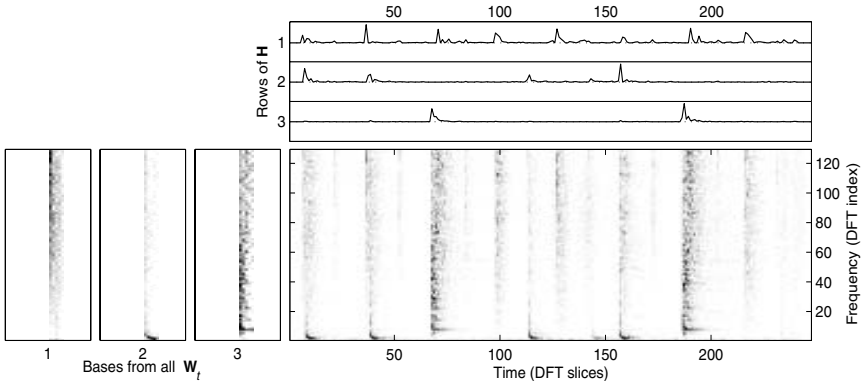

Fig. 3. NMFD bases and weights for drum example. The lower right plot is the magnitude spectrogram that we used as an input. The three leftmost plots are the temporal-spectral bases from $\mathbf{W}_t$. Their corresponding weights and rows of $\mathbf{H}$ are depicted at the top plot. Note how the extracted bases encapsulate the temporal/spectral structure of the three drum sounds in the spectrogram.

Having this description is a valuable guide to perform separation. We can do partial reconstructions of the input spectrogram using one basis function at a time. For example to extract the bass drum which was mapped to the j^{th} basis we do:

$$\hat{\mathbf{V}}_j = \sum_{t=0}^{T-1} \mathbf{W}_t^{(j)} \cdot \overrightarrow{\mathbf{H}}^t \tag{8}$$

where the $(\cdot)^{(j)}$ operator selects the jth column of the argument. This gives us the magnitude spectrogram of one component. We apply this to the original phase of the spectrogram and invert the result to obtain a time series. Subjectively we have found that the extracted elements consistently sound like the elements of the input sound scene. Unfortunately it is very hard to come up with a useful and intuitive measure that otherwise describes the quality of separation due to various non-linear distortions and lost information, problems inherent in the mixing and the analysis processes.

4 Conclusions

In this paper we presented an convolutional version of NMF. We have pinpointed some of the shortcomings of conventional NMF when analyzing temporal patterns and presented an extension which results in the extraction of more expressive basis functions. We have also shown how these basis functions can be used in the same way spectral bases have been used on spectrograms to extract sound objects from single channel sound scenes.

References

Casey, M.A. and A. Westner (2000) "Separation of Mixed Audio Sources by Independent Subspace Analysis", in *Proceedings of the International Computer Music Conference*, Berlin, Germany, August, 2000.

Lee, D.D. and H.S. Seung. (1999) "Learning the parts of objects with nonnegative matrix factorization". In *Nature*, 401:788 791, 1999.

Lee, D.D. and H.S. Seung (2000) "Algorithms for Non-Negative Matrix Factorization". In *Neural Information Processing Systems* 2000, pp. 556-562.

Paatero, P. (1997) "Least Squares Formulation of Robust Non-Negative Factor Analysis", in *Chemometrics and Intelligent Laboratory Systems* **37**, pp. 23-35, 1997.

Smaragdis, P. (2001) "Redundancy Reduction for Computational Audition, a Unifying Approach", *Doctoral Dissertation*, MAS Dept. Massachusetts Institute of Technology, Cambridge MA, USA.

Smaragdis, P. and J.C. Brown. (2003) "Non-Negative Matrix Factorization for Polyphonic Music Transcription", in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. New Paltz, NY, October 2003.