A JUMP START FOR NMF WITH N-FINDR AND NNLS

Joachim Ganseman[*]
IBBT-Visionlab, Department of Physics
University of Antwerp
Antwerp, Belgium
joachim.ganseman@ua.ac.be

Paul Scheunders
IBBT-Visionlab, Department of Physics
University of Antwerp
Antwerp, Belgium
paul.scheunders@ua.ac.be

ABSTRACT

Nonnegative Matrix Factorization is a popular tool for the analysis of audio spectrograms. It is usually initialized with random data, after which it iteratively converges to a local optimum. In this paper we show that N-FINDR and NNLS, popular techniques for dictionary and activation matrix learning in remote sensing, prove useful to create a better starting point for NMF. This reduces the number of iterations necessary to come to a decomposition of similar quality. Adapting algorithms from the hyperspectral image unmixing and remote sensing communities, provides an interesting direction for future research in audio spectrogram factorization.

1. INTRODUCTION

1.1. NMF

Nonnegative matrix factorization (NMF) has seen lots of popular use since the publication of multiplicative update algorithms in [1]. It was first used on audio spectrograms in [2]. Many variations of the algorithm exist, depending on the cost function or minimization approach used. An overview can be found in [3].

Given a matrix $V_{m\times n}$ and a strictly positive integer p, NMF seeks an approximate decomposition of V into two matrices $W_{m\times p}$ and $H_{p\times n}$, such that $WH \approx V$. This is accomplished through iterative minimization of a norm $||WH-V||$. Multiplicative update rules derived from a gradient descent or expectation-maximization approach are often used. Consider as norm e.g. the β-divergence, which encompasses the Itakura-Saito divergence ($\beta = 0$), the generalized Kullback-Leibler divergence ($\beta = 1$) and the Euclidian distance ($\beta = 2$). Given these, [4] formulates generic update rules, for which they also prove convergence, as follows ($\circ$ denotes element-wise matrix multiplication):

$$W \leftarrow W \circ \left(\frac{(V \circ (WH)^{\beta-2})H^T}{(WH)^{\beta-1}H^T} \right)^{\phi(\beta)} \quad (1)$$

$$H \leftarrow H \circ \left(\frac{W^T(V \circ (WH)^{\beta-2})}{W^T(WH)^{\beta-1}} \right)^{\phi(\beta)} \quad (2)$$

with $\phi(\beta)$ chosen as:

$$\phi(\beta) = \begin{cases} \frac{1}{2-\beta} & \text{if} \quad \beta < 1 \\ 1 & \text{if} \quad 1 \leq \beta \leq 2 \\ \frac{1}{\beta-1} & \text{if} \quad \beta > 2 \end{cases} \quad (3)$$

[*] The author is supported by a specialization grant of the Institute for innovation through Science and Technology (IWT-Flanders)

NMF is generally initialized randomly, and converges to a local optimum. The result is a decomposition of V in a *dictionary matrix* W and an *activation matrix* H. Applied to a magnitude or power audio spectrogram, W is composed of p spectra and H indicates their weights at a given time. Note that here we use an example (from [2]) where the resulting W and H are also musically meaningful, but that is certainly not generally guaranteed.

1.2. Hyperspectral Image Unmixing

Finding a sparse approximate decomposition of spectral data is a problem that also arises in remote sensing [5, 6]. As an example, an earth surveying satellite produces *hyperspectral images* covering a wide range of the electromagnetic spectrum. Each pixel in such an image is considered a linear combination of different elements that each have their own spectral signature: water, vegetation, copper and other minerals, ... These constituting elements are called *endmembers*, and upon plotting the contribution of a single endmember to the entire image, we obtain an *abundance* map.

Hyperspectral image data is nonnegative by nature. The unmixing problem is usually split up in 2 subproblems that are treated separately:

- Endmember Extraction: find the endmembers W that are present in the image,
- Abundance Estimation: compute the contribution of each endmember in every pixel (H).

NMF computes solutions to both problems simultaneously through iterative updates. Having 2 independent subproblems makes it easier to develop algorithms that iterate over the datapoints, instead of requiring the entire dataset to be in memory as with NMF. As very large datasets are common in that field, this approach is usually preferred over NMF in hyperspectral image unmixing.

One key difference is that many hyperspectral image unmixing methods enforce that all abundancies should sum to one (assuming that an observed pixel cannot contain any "nothingness"), whereas a frame in an audio spectrogram can contain silence. When applying image unmixing methods to audio spectrograms, we can circumvent this by adding an all-zero *shadow* endmember to the dictionary before the abundance estimation process, acting as a component representing silence. [6] mentions that it is unclear whether this sum-to-one constraint significantly influences unmixing results.

An accessible tutorial text on hyperspectral image unmixing is provided by [5], while [6] contains an extensive overview of the developments over the last decade.

1.3. A geometric view of unmixing

Hyperspectral image unmixing is often tackled from a geometric perspective. Each datapoint is considered to be lying within a simplex, with the vertices formed by the endmember spectra. Simplices have the following properties, which can be easily verified for the 2-simplex in the toy example provided in figure 1:

- Every n-simplex has $n+1$ vertices and $n+1$ edges,
- Every point within the space defined by the vertices has $n+1$ barycentric coordinates, e.g. $M = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$,
- Normalized, the barycentric coordinates of every point sum to 1,
- If a point has all positive coordinates, it lies within the convex hull formed by the vertices.
- Every point within a simplex can be expressed as a nonnegative linear combination of the vertices, e.g. $M = \frac{1}{3}A + \frac{1}{3}B + \frac{1}{3}C$.

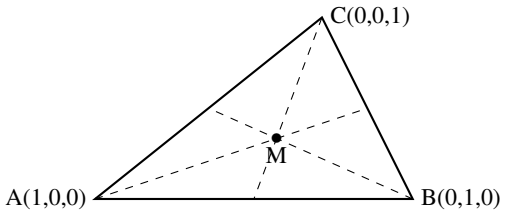

Figure 1: *The standard 2-simplex (a.k.a. triangle) and its centroid.*

All hyperspectral pixels (or spectrogram frames) are datapoints within a space that has a dimensionality equal to the number of frequency bins. Now assume that all datapoints can be written as a linear combination of only a few spectra. Then all our datapoints lie on a hyperplane within this space, bounded by these few spectra. The problem of finding a sparse decomposition then translates to: find a low-dimensional simplex within our dataspace that encompasses as much of the data as possible with minimal error. Figure 2 gives an impression of such configuration, where data in a 3D space is well-approximated within a lower-dimensional plane (in this case 2D).

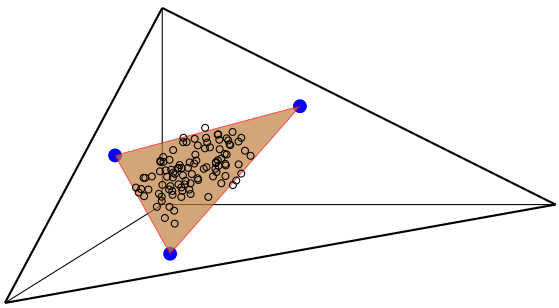

Figure 2: *Finding a simplex covering the datacloud in a higher-dimensional space.*

The link between unmixing, with possible audio applications, and this geometric simplex interpretation, has been observed before [7, 8, 9]. Notably, [10] proposes a method based on Single-Class Support Vector Machines to solve the related problem of finding a conic hull around the data. Here, we employ methods that manipulate simplices, and show how sparse spectrogram decompositions can be obtained when combining these with NMF.

2. INITIALIZING NMF WITH N-FINDR

2.1. N-FINDR

Algorithms that search for a low-dimensional simplex within the data can be roughly subdivided into 2 classes:

- Find a maximum volume simplex inscribed in the data.
- Find a minimum volume simplex encompassing the data.

N-FINDR [11] is an algorithm of the maximum volume inscribed category. It therefore assumes that the endmembers are present in the data, or otherwise formulated: the obtained dictionary elements that we wish that explain the data, must be present in the data themselves. For audio, this translates to the assumption that there are frames present within the spectrogram that only contain a single component, not mixed with any other components.

Suppose we wish to find p extreme datapoints and use them as endmembers. N-FINDR selects them as follows:

- Reduce dimensionality to $p-1$ using PCA or similar
- Select p random points
- For all other datapoints:
  - Try out the datapoint as replacement for each of the selected p
  - Calculate the simplex volume: Cayley-Menger determinant, or simpler: $\frac{1}{(p-1)!}|det\binom{1\ldots1}{p}|$
  - Keep the set of points resulting in maximum simplex volume.

This algorithm inflates a simplex within the datacloud. Both the final results and the runtime of N-FINDR depend on the initial set of endmembers and the order in which the datapoints are evaluated. To increase the likelihood of growing the simplex fast, we can e.g. avoid spatial correlation in neighbouring pixels in images (or temporal correlation in neighbouring frames in audio spectrograms) by randomizing datapoint evaluation. This and several other possible optimizations are described in [12].

2.2. Abundance estimation

After obtaining the simplex, we add an all-0 component to account for possible silence. By doing this, we can use abundance estimation methods that rely on the sum-to-one constraint. A range of literature is available that covers the abundance estimation by solving the Fully Constrained Least Squares Unmixing (FCLSU) problem [13]. For our purpose, we keep it simple and use the traditional Lawson-Hanson Nonnegative Least Squares (NNLS) algorithm. It is implemented in MATLAB as the `lsqnonneg` function.

3. RESULTS

As example data, we use a 5-note solo piano excerpt originally coming from [2]. There are 4 distinct pitches present in the signal, so we first try to find 4 different components. A second test signal is created by overlapping the first one several times with itself at different positions. This highly mixed scenario has several notes being struck at the same time in most frames.

We choose to compute amplitude spectrograms and minimize the generalized Kullback-Leibler divergence as NMF method, with W and H initialized either randomly or with values obtained through N-FINDR and NNLS. Extracting 4 components, the convergence behaviour is shown in figure 3. All curves depict mean data, computed over 100 runs of the algorithm. The x-axis shows the number of iterations, on the y-axis the residual norm is plotted. N-FINDR+NNLS initialized NMF delivers better results when the number of iterations is kept low. Given the solo data, it also stays better, but with more mixed data, the randomly initialized NMF takes and keeps a slight lead after about the 25th iteration.

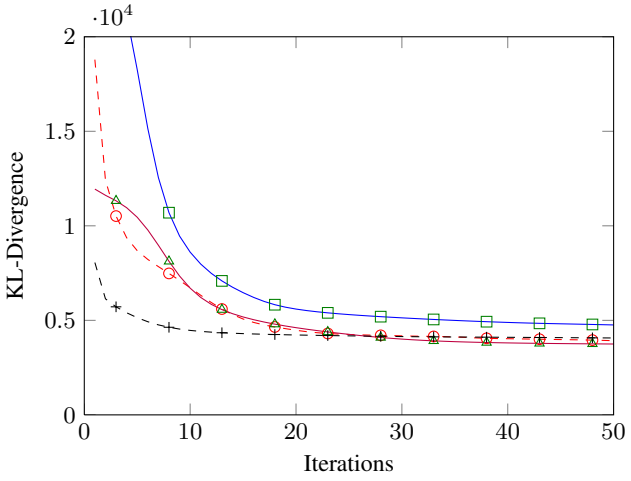

Figure 3: *Mean convergence of NMF initialized randomly on solo data (squares) and mixed data (triangles), and initialized with N-FINDR and NNLS on solo data (bullets) and mixed data (crosses). Extracting 4 components, 50 iterations.*

When we don't know the dimensionality of the data, we can overestimate the number of components as this keeps at least the reconstruction error down. We run the same examples, but now try to find 40 components in the data where only 4 are enough to explain most of it. In the solo data, most of the additional components will only be low energy as a result. In the mixed data, some additional components are likely to explain not only single pitches but also entire chords separately.

The effect is shown in figure 4. N-FINDR and NNLS provide an excellent starting point to begin with, up to the point that subsequent NMF seems hardly necessary. On the other hand, the randomly initialized NMF takes more time to converge, but eventually overtakes the other after a while. This happens at about the 26th iteration for the solo data, and slightly earlier at around the 22nd iteration for the mixed data.

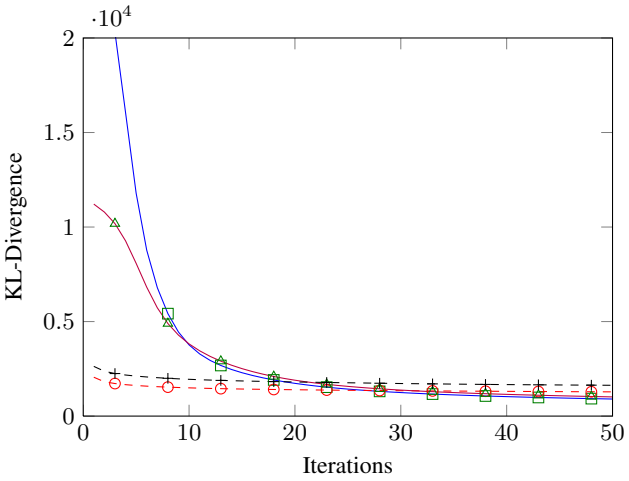

Figure 4: *Mean convergence of NMF initialized randomly on solo data (squares) and mixed data (triangles), and initialized with N-FINDR and NNLS on solo data (bullets) and mixed data (crosses). Extracting 40 components, 50 iterations.*

The difference is better visible when computing the Signal to Distortion Ratio (SDR) of the reconstructed signal, as defined in the BSS_EVAL toolkit [14], in function of the number of iterations. Figure 5 shows results for the mixed data, with both 4 and 40 components computed. Given 4 components, N-FINDR and NNLS provide an initialization that already boasts a good SDR, which can be slightly improved upon when followed by up to 8 NMF iterations. After that there is a minimal decay, but the SDR remains better than wat can be obtained with random initialization. When the number of components is increased to 40, N-FINDR and NNLS obtain a good result that is hardly improved by subsequent NMF, while random-initialized NMF eventually gets better when run for enough iterations.

With p components and n datapoints, N-FINDR calculates np times a $p \times p$ determinant, resulting in a time complexity of $\mathcal{O}(p^4 n)$ when a naive implementation is used. It is only really beneficial with a small number of components: on our machine finding 4 components took 0.12 seconds and finding 40 took 5.5 seconds. NNLS computes their abundancies in respectively 0.30 and 0.97 seconds. NMF's multiplicative updates have an $\mathcal{O}(mnpi)$ complexity where i is the number of iterations. NMF with 50 iterations with 4 components took 2.0 seconds on our machine.

4. CONCLUSIONS AND FUTURE DIRECTIONS

We began by pointing out the equivalence between the problems of nonnegative audio spectrogram factorization and hyperspectral image unmixing. Whereas the former has until now been mostly tackled with Nonnegative Matrix Factorization, literature about the latter has concentrated on Simplex Decompositions.

Using N-FINDR and NNLS, two proven traditional methods used in hyperspectral image unmixing, we showed that we can obtain dictionary and activation matrices that can serve as good

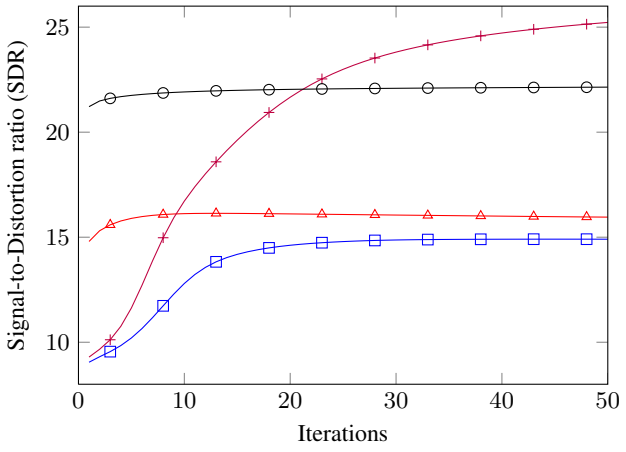

Figure 5: *Reconstruction Signal-to-Distortion ratio in function of the number of iterations of a highly mixed signal extracting 4 components: initialized randomly (squares) or with N-FINDR+NNLS (triangles); and extracting 40 components: initialized randomly (crosses) or with N-FINDR+NNLS (bullets).*

initialization for NMF. This results in less iterations necessary to come to similar results as randomly-initialized NMF would. If the number of components or iterations is large, the latter still catches up. This may be related to the particular minimization approach employed in the implementation of NMF. More specifically, when starting from a good starting position, where the gradient is small, other algorithms than the rescaled gradient descent from [1] may be better able to escape from a possible bad path. This is an interesting direction for future work.

Similarly, more methods from the field of remote sensing can be readily added to the toolset for audio spectrogram analysis. Vertex Component Analysis [15] comes to mind as alternative to N-FINDR with better time complexity. Of particular interest are methods that do not assume dictionary elements to be present in the data [16], e.g. SISAL [17]. Finally, emerging topics in the remote sensing literature that are also of interest to the audio community are sparsity, on-line or real-time algorithms, data dimensionality estimation, correlations between datapoints, etc. Care needs to be taken regaring the specific properties of audio data though, as there are the presence of silence and the dynamic range.

5. REFERENCES

[1] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Neural Information Processing Systems (NIPS)*, Denver, USA, 2000, pp. 556–562.

[2] P. Smaragdis and J.C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, oct. 2003, pp. 177 – 180.

[3] A. Cichocki, R. Zdunek, and S.-i. Amari, "Nonnegative matrix and tensor factorization [lecture notes]," *Signal Processing Magazine, IEEE*, vol. 25, no. 1, pp. 142 –145, 2008.

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

[5] N. Keshava and J.F. Mustard, "Spectral unmixing," *Signal Processing Magazine, IEEE*, vol. 19, no. 1, pp. 44 –57, jan 2002.

[6] J.M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical and sparse regression-based approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, , no. 2, pp. 354–379, 2012.

[7] M. Shashanka, B. Raj, and P. Smaragdis, "Probabilistic latent variable models as nonnegative factorizations," *Computational Intelligence and Neuroscience*, 2008.

[8] P. Smaragdis, M. Shashanka, and B. Raj, "A sparse non-parametric approach for single channel separation of known sounds," in *Neural Information Processing Systems*, Vancouver, Canada, dec. 2009, pp. 722–730.

[9] M. Shashanka, "Simplex decompositions for real-valued datasets," in *IEEE Int. Workshop on Machine Learning for Signal Processing (MLSP)*, sept. 2009, pp. 1 –6.

[10] Slim Essid, "A single-class svm based algorithm for computing an identifiable nmf," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Kyoto, Japan, march 2012.

[11] M.E. Winter, "N-findr: an algorithm for fast autonomous spectral end-member determination in hyperspectral data," *Image Spectrometry V, Proc. SPIE*, vol. 3753, pp. 266–277, 1999.

[12] M. Zortea and A. Plaza, "A quantitative and comparative analysis of different implementations of n-findr: A fast end-member extraction algorithm," *IEEE Geoscience and Remote Sensing Letters*, vol. 6, no. 4, pp. 787 –791, oct. 2009.

[13] R. Heylen, D. Burazerovic, and P. Scheunders, "Fully constrained least squares spectral unmixing by simplex projection," *IEEE Trans. Geoscience and Remote Sensing*, vol. 49, no. 11, pp. 4112 –4122, nov. 2011.

[14] E. Vincent, C. Févotte, and R. Gribonval, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[15] J.M.P. Nascimento and J.M. Bioucas-Dias, "Vertex component analysis: a fast algorithm to unmix hyperspectral data," *IEEE Trans. Geoscience and Remote Sensing*, vol. 43, no. 4, pp. 898 – 910, april 2005.

[16] J. Plaza, E.M.T. Hendrix, I. Garcia, G. Martin, and A. Plaza, "On endmember identification in hyperspectral images without pure pixels: A comparison of algorithms," *J. Mathematical Imaging and Vision*, vol. 42, no. 2-3, pp. 163–175, february 2012.

[17] J.M. Bioucas-Dias, "A variable splitting augmented lagrangian approach to linear spectral unmixing," in *1st Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, aug. 2009, pp. 1 –4.