ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/349569680

Deep orthogonal matrix factorization as a hierarchical clustering technique

Preprint · February 2021

CITATIONS
0

READS
98

2 authors:

Pierre De Handschutter
Université de Mons
8 PUBLICATIONS **4** CITATIONS

SEE PROFILE

Nicolas Gillis
Université de Mons
129 PUBLICATIONS **2,244** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Deep MF View project

Project Log-determinant constrained Non-negative Matrix Factorization View project

All content following this page was uploaded by Pierre De Handschutter on 24 February 2021.

The user has requested enhancement of the downloaded file.

Deep orthogonal matrix factorization as a hierarchical clustering technique

Pierre De Handschutter Nicolas Gillis
Department of Mathematics and Operational Research
University of Mons, Belgium

***Abstract*—Deep orthogonal nonnegative matrix factorization (deep ONMF) is a constrained deep low-rank matrix approximation model which decomposes a data matrix through several layers of factorizations. Deep ONMF imposes that each data point is assigned to a single cluster at each layer. In this paper, we first explain why deep ONMF can be interpreted as a bottom-up hiearchical clustering technique. Then our main contribution is to provide a simple yet effective greedy initialization strategy for deep ONMF. We show on synthetic data sets that it performs competitively with other initialization strategies, and apply it on the decomposition of a hyperspectral image into its constitutive materials.**

I. Introduction

Given a matrix $X \in \mathbb{R}^{m\times n}$ where each column is a data point lying in an m-dimensional space, a low-rank matrix approximation seeks for matrices $W \in \mathbb{R}^{m\times r}$ and $H \in \mathbb{R}^{r\times n}$ such that each data point $X(:,j)$ can be approximated as $X(:,j) \approx \sum_{k=1}^{r} W(:,k)H(k,j)$ for $j = 1,\dots,n$. This means that each data point is a linear combination of r basis vectors, where r is called the rank of the approximation. In matrix form, this approximation, also called a factorization, is written as $X \approx WH$, where each column of W corresponds to a basis vector and each column of H indicates the proportions in which each basis vector appears in each data point. The quality of the approximation is generally measured by the least squares criterion, that is, $\|X - WH\|_F^2$.

To ensure the interpretability and uniqueness of such models, constraints are typically imposed on the factors W and H, such as sparsity [1] and nonnegativity [2], leading to sparse component analysis and nonnegative matrix factorization (NMF), respectively. Adding orthogonality on top of nonnegativity for the factor H, we obtain orthogonal NMF (ONMF) [3] which can be formulated as follows

$$\min_{W\in\mathbb{R}_+^{m\times r}, H\in\mathbb{R}_+^{r\times n}} \|X - WH\|_F^2 \text{ such that } HH^T = I_r, \quad (1)$$

where I_r is the identity matrix of size r.

Recently, matrix factorizations (MFs) have been extended to the case where the input matrix is decomposed in more than two factors. More precisely, L layers of successive factorizations of ranks d_l ($l = 1,...,L$) are performed on X as follows: $X \approx W_1H_1, W_1 \approx W_2H_2, \dots, W_{L-1} \approx W_LH_L$, where $W_l \in \mathbb{R}^{m\times d_l}$ and $H_l \in \mathbb{R}_+^{d_l\times d_{l-1}}$ ($l = 1,\dots,L$) with $d_0 = n$, so that the matrix X is approximated as $X \approx W_LH_LH_{L-1}\cdots H_1$. This model is referred to as multilayer MF [4] or deep MF [5], depending on the way the optimization is performed. Multilayer MF performs the decomposition in a purely sequential way, that is, it successively minimizes $\|W_{i-1} - W_iH_i\|_F^2$ for $i = 1,2,\dots,L$ where $W_0 = X$. Deep MF considers a further backpropagation step. This consists in minimizing the loss function $\|X - W_LH_L\cdots H_1\|_F^2$ across the layers: after a sequential decomposition as in multilayer MF, the factors are iteratively updated in a block-coordinate descent fashion, see [6] and the references therein for more details. As for shallow MFs, additional constraints must be imposed on the factors to render it meaningful. Imposing nonnegativity and orthogonality on the H_l's leads to deep ONMF [7], the topic of this paper.

Organization of the paper: In Section II, we start by explaining why deep ONMF is a particular hierarchical clustering (HC) model. We then provide a greedy initialization for deep ONMF in Section III. In Section IV-A, we compare several initialization techniques on synthetic data, and in Section IV-B we illustrate the ability of deep ONMF combined with our greedy initialization to cluster the pixels of a hyperspectral image in a hierarchical way. Finally, in Section V, we briefly conclude and give perspectives of research.

II. Deep ONMF is equivalent to HC

It is well-known that standard NMF can be interpreted as a soft clustering technique. In particular, when the sum of the entries in each column of H is constrained to be equal to 1, $H(k,j)$ is the proportion in which the data point $X(:,j)$ is associated to the k-th basis vector $W(:,k)$. Due to the row-wise orthogonality constraint, ONMF is more restrictive: nonnegativity together with orthogonality implies that each column of H has at most a single non-zero entry. This follows from the fact that two nonnegative and orthogonal vectors must have disjoint supports. Hence ONMF associates each data point to a single basis vector and performs a hard clustering [3]. In fact, it can be proved that ONMF is equivalent to a weighted variant of spherical k-means [8]. Recall that spherical k-means minimizes the angles between the data points and their associated centroid, as opposed to k-means that minimizes their Euclidean distances.

Deep ONMF is the extension of ONMF (1) to several layers: for $l = 1,2,\dots,L$,

$$W_{l-1} \approx W_lH_l \text{ such that } (W_l,H_l) \geq 0 \text{ and } H_lH_l^T = I_{d_l}, \quad (2)$$

where $W_0 = X$. In deep ONMF, the ranks d_l's need to decrease as the factorization unfolds, that is, $d_l > d_{l+1}$ for

all l, otherwise we end up with trivial factorizations. In fact, if $d_l \leq d_{l+1}$ for some l, $W_l = W_{l+1}H_{l+1} = (W_l\, 0)\binom{I_{d_l}}{0}$ where 0 is the matrix of zeros of appropriate dimension. Under this assumption, deep ONMF can be interpreted as a hierarchical clustering (HC) model, as it successively merges the data points by aggregating clusters, which is referred to as bottom-up HC or agglomerative clustering [9]. This first layer splits the columns of X into d_1 clusters, the second layer splits the centroids W_1 into d_2 clusters, and so on. In particular, it is interesting to observe that when the ranks are such that $d_l = d_{l-1} - 1$ for all l, each layer merges two clusters of the previous layer into a single new cluster while keeping the others unchanged. Hence, deep ONMF is a HC technique where the criterion of the clustering is a weighted spherical k-means. It is closely related to deep k-means [10] which enforces each column of H_l to have a single nonzero entry equal to one, for all l (in ONMF, there is a scaling degree of freedom to approximate a data point with a centroid).

Note that other HC techniques are based on NMF ideas. This is for example the case in [11], [12], [13] where rank-2 NMF's are applied sequentially to split clusters in two. In contrast to deep ONMF, these techniques are top-down which means that they start by assigning all the data points to a single cluster and progressively split them in several clusters, hence leading to a different interpretation than deep ONMF.

III. Greedy initialization of deep ONMF

Usually, deep ONMF is applied using $d_l \ll d_{l-1}$ at each layer and, in particular, $d_1 \ll n$. However, we show in this section that it is possible to obtain a closed-form optimal solution of deep ONMF at the first layer when $d_1 = n-1$ or, equivalently, for ONMF with $r = n-1$; see Section III-A. Using this idea sequentially, we propose a greedy initialization for deep ONMF; see Section III-B. As most clustering strategies, deep ONMF relies on iterative methods and is very sensitive to initialization. This is the main motivation behind this work.

A. Solving ONMF for $r = n-1$

Let us consider a data matrix $X \in \mathbb{R}^{m\times n}$ and let us consider ONMF (1) with $r = n-1$. Because ONMF needs to assign n data points to $n-1$ clusters, only two data points need to be merged within the same cluster, say the ith and the jth. Assuming we know i and j, minimizing the reconstruction error requires finding the scalars h_i and h_j and the new centroid $w \in \mathbb{R}^m$ such that

$$e(i,j) = \|X(:,i) - h_i w\|_F^2 + \|X(:,j) - h_j w\|_F^2 \qquad (3)$$

is minimized, with $h_i^2 + h_j^2 = 1$ and $(h_i, h_j) \geq 0$. Using the first-order optimality conditions, it is easy to show that, at optimality, $w = h_i X(:,i) + h_j X(:,j)$ and $h_k = \frac{X(:,k)^T w}{\|w\|_2^2}$ for $k = i,j$. Combining both equalities and $h_i^2 + h_j^2 = 1$, and writing everything in terms of $\alpha := h_i$, we obtain

$$\frac{\alpha}{\sqrt{1-\alpha^2}} = \frac{\alpha\|X(:,i)\|^2 + \sqrt{1-\alpha^2}X(:,i)^T X(:,j)}{\alpha X(:,i)^T X(:,j) + \sqrt{1-\alpha^2}\|X(:,j)\|^2}.$$

Assume w.l.o.g. that $\|X(:,i)\|_2 \geq \|X(:,j)\|_2$. In the case $\|X(:,i)\|_2 = \|X(:,j)\|_2$, $\alpha = 1/\sqrt{2}$, otherwise $\alpha = \sqrt{\frac{L+\sqrt{L}}{2L}}$ where $L = 4K^2+1$ and $K = \frac{X(:,i)^T X(:,j)}{\|X(:,i)\|^2 - \|X(:,j)\|^2}$ (note that, as $\|X(:,i)\|_2 \to \|X(:,j)\|_2$, $L \to \infty$ and $\alpha \to 1/\sqrt{2}$). Finally, the optimal solution of ONMF with $r = n-1$ will merge the data points $X(:,i)$ and $X(:,j)$ for which $e(i,j)$ takes the smallest value. After permutation, let us assume w.l.o.g. that $i = n-1$ and $j = n$, the optimal ONMF has the form $X \approx \begin{pmatrix} X(:,1:n-2) & w \end{pmatrix} \begin{pmatrix} I_{n-2} & 0 & 0 \\ 0 & h_i & h_j \end{pmatrix}$, with error $e(i,j)$ given in (3).

We will denote by ONMF$(n-1)$ the exact procedure for ONMF for $r = n-1$. The two points leading to the smallest $e(i,j)$ (see Eq. 3) are merged according to Algorithm 1. Note that, given (i,j), computing α and w requires $\mathcal{O}(m)$ operations. Hence ONMF$(n-1)$ requires $\mathcal{O}(mn^2)$ operations to test all pairs (i,j) and pick the one that leads to the lowest error.

Algorithm 1 Exact orthogonal approximation of two points

Input: Two data points $x_i, x_j \in \mathbb{R}^m$.
Output: Basis vector $w \in \mathbb{R}^m$, coefficient α, error $e(i,j)$

perm$= 0$;
if $\|x_i\| < \|x_j\|$ **then**
  $(x_i, x_j) \leftarrow (x_j, x_i)$; perm $= 1$;
end if
if $\|x_i\| = \|x_j\|$ **then**
  $\alpha = 1/\sqrt{2}$
else
  $K = \frac{x_i^T x_j}{\|x_i\|^2 - \|x_j\|^2}$, $L = 4K^2+1$, $\alpha = \sqrt{\frac{L+\sqrt{L}}{2L}}$
end if
$w = \alpha x_i + \sqrt{1-\alpha^2}x_j$
$e(i,j) = \|x_i - \alpha w\|^2 + \|x_j - \sqrt{1-\alpha^2}w\|^2$
if perm $= 1$ **then**
  $\alpha \leftarrow \sqrt{1-\alpha^2}$.
end if

B. Application to deep ONMF

Based on ONMF$(n-1)$, we propose a simple greedy initialization of deep ONMF, which we refer to as the successive orthogonal decomposition algorithm (SODA).

Starting from the trivial decomposition $X = W_1 H_1$, with $W_1 = X$ and $H_1 = I_n$, the two data points $W_1(:,i)$ and $W_1(:,j)$ with smallest value for $e(i,j)$ are merged at layer 2 according to the closed-form solution described in the previous section. At each layer, two basis vectors are merged in a new one using the method described in Section III-A. When the current number of basis vectors is equal to the value of some inner rank d_l, they are used to initialize the corresponding W_l.

Let us analyze the computational cost of SODA. First, the reconstruction errors $e(i,j)$ of any two data points i and j are computed, which requires $\frac{n(n-1)}{2}$ times $\mathcal{O}(m)$ operations. SODA will have $n - d_L + 1$ steps to be able to construct W_L, each of them requires to compute the reconstruction error between the remaining basis vectors of the previous layer and

the new one, which requires $\mathcal{O}(nm)$ operations. Moreover, assuming an efficient sorting strategy of the array of errors $e(i,j)$'s, finding the couple of indices which generates the smallest $e(i,j)$ is $\mathcal{O}(n^2 \log(n^2))$. Hence, SODA requires in total $\tilde{\mathcal{O}}(n^2 m)$ operations (where $\tilde{}$ indicates that we removed logarithmic terms), which is not practical for large data sets. Usually, deep ONMF algorithms run in $\mathcal{O}(mnd_1)$ operations where $d_1 \ll n$.

However, for large data sets, such as hyperspectral images where n is the number of pixels (which can be of the order of millions), this greedy idea can be used as well. For example, we first compute an ONMF of rank larger than (or equal to) d_1, say $d_1' \geq d_1$ with any standard algorithm faster than SODA, and then "unfold" the remaining d_1' clusters through SODA. This requires $\mathcal{O}(mnd_1')$ operations for the first layer ONMF, and then $\tilde{\mathcal{O}}(d_1'^2 m)$ operations for the next ones computed by SODA. In practice, we recommend to use d_1' as a small multiple of d_1.

IV. Numerical experiments

In this section, we evaluate the performance of several initialization techniques for deep ONMF on synthetic data in Section IV-A, and show the hierarchical clustering produced by deep ONMF for a hyperspectral image in Section IV-B.

A. Synthetic data sets

Let us compare the effectiveness of the following initialization methods for the W_l's in deep ONMF:

- Random initialization (RAND): any W_l, $l = 1, ..., L$ is set up by randomly picking $d_l < d_{l-1}$ columns of W_{l-1}, with $W_0 = X$. This is the most standard approach in the literature.
- Successive nonnegative projection algorithm (SNPA) [6]: W_l is obtained with SNPA [14] applied on W_{l-1}.
- Our proposed greedy algorithm, SODA.
- RAND+SODA: Similarly to as described at the end of Section III-B, we randomly choose $d_1' \ll n$ points, and then apply SODA on this subset of points.

We compare these initializations when combined with the alternating optimization strategy that optimizes W_l's and H_l's alternatively by extending the multiplicative updates proposed for ONMF by [15] to deep ONMF.

We generate the synthetic data sets as follows, in $m = 3$ dimensions. We take $d_1 = 16$ and $d_2 = 4$ and generate the ground-truth (GT) basis vectors W_1 and W_2 whose columns have unit ℓ_1 norms in such a way that the 16 first layer basis vectors are clustered in 4 groups around 4 second layer basis vectors; see Fig. 1 for an illustration. As shown on Fig. 1, the columns of W_2 are the central basis vectors of 4 columns of W_1: more precisely, it is equal to their average, up to a scaling factor. We then pick $n = 1000$ points uniformly at random over the GT clusters, that is, each data point is equal to one of the columns of W_1, up to scaling factor and fix d_1' to 100. Finally, noise is added to the data such that

$$X = \max\left(0, \tilde{X} + \epsilon ||\tilde{X}||_F \frac{N}{||N||_F}\right),$$

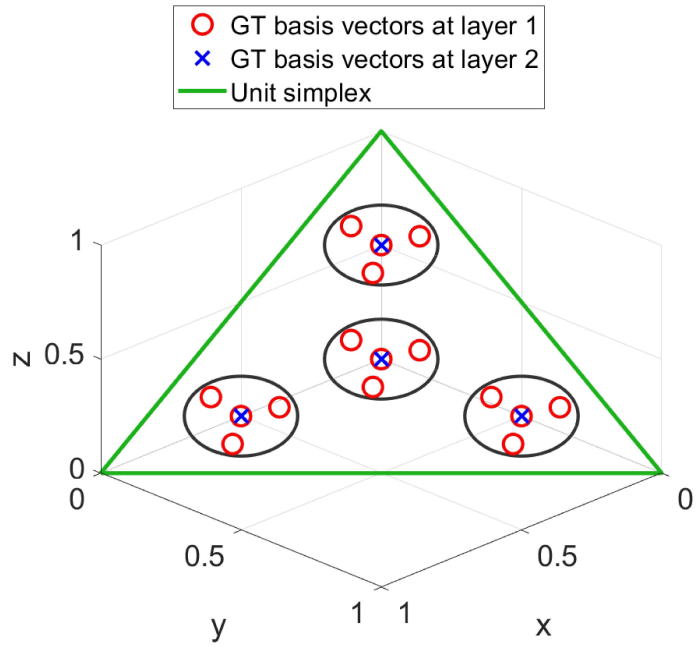

Fig. 1: Geometric illustration of the synthetic data sets.

where $\tilde{X} = W_1 H_1$, each entry of N follows a Gaussian distribution of mean 0 and standard deviation 1, and ϵ is the noise level. To assess the quality of the different initializations, 10 data sets are generated for each noise level and we report the mean and standard deviation of the clustering accuracy (ACC) at both layers for several noise levels. Given K estimated clusters G_k's and K ground truth clusters H_k's, the ACC is defined as

$$ACC(G, H) = \frac{1}{n} \max_{P \in [1 \cdots K]} \sum_{k=1}^{K} |G_k \cap H_{P(k)}| \qquad (4)$$

where P is any permutation of $\{1, 2, \ldots, K\}$.

The results are presented in terms of both reconstruction error and accuracy at Table I. More precisely, it reports the relative reconstruction error $\frac{\|X - W_2 H_2 H_1\|_F}{\|X\|_F}$, denoted *rel_err*, and the accuracy at the first and the second layers, denoted *ACC 1* and *ACC 2*, respectively.

Clearly, SODA outperforms RAND and SNPA in terms of clustering accuracy. When the noise is small, it always manages to reach a perfect clustering at both layers, contrary to the two other methods. Of course, this is at the expense of a larger computational cost, from $\mathcal{O}(mnr)$ for RAND and SNPA, to $\mathcal{O}(mn^2)$ for SODA. However, RAND+SODA performs almost as well as SODA at a reduced cost (see the end of Section III-B), showing that using the greedy procedure further in the decomposition is also worthwhile. Note that the accuracy of all algorithms is always a bit higher for the second layer since there are fewer clusters, which are better separated. The reason SNPA underperforms is because some clusters are contained in the convex cone of the others, while SNPA is designed to identify extreme rays of the cone generated by the columns of X.

Table I: Comparison of the clustering accuracies at layer 1 (*ACC 1*) and 2 (*ACC 2*) and final relative error (*rel_err*) of deep MF applied on synthetic data with several initialization strategies, as a function of the noise level. The average and standard deviation (if above 0.01) over 10 data sets are reported. The best method in terms of accuracy is highlighted in bold for each configuration.

	RAND			SNPA			SODA			RAND+SODA		
ϵ	ACC 1	ACC 2	rel_err (%)	ACC 1	ACC 2	rel_err (%)	ACC 1	ACC 2	rel_err (%)	ACC 1	ACC 2	rel_err
10^{-4}	0.54 ± 0.14	0.74 ± 0.21	9.26 ± 6.23	0.21 ± 0.03	0.69 ± 0.17	14.84 ± 3.91	**1**	**1**	7.49	**1**	**1**	7.50
10^{-3}	0.49 ± 0.17	0.66 ± 0.19	9.41 ± 6.19	0.18 ± 0.02	0.67 ± 0.14	15.49 ± 4.24	**1**	**1**	7.49	**1**	**1**	7.50
10^{-2}	0.48 ± 0.17	0.76 ± 0.16	10.31 ± 6.51	0.42 ± 0.09	0.72 ± 0.16	10.82 ± 4.21	**0.97**	**0.99**	7.51	**0.97**	**0.99**	7.52
10^{-1}	0.40 ± 0.07	0.70 ± 0.17	15.46 ± 4.57	0.38 ± 0.07	0.68 ± 0.09	14.27 ± 3.20	$\mathbf{0.69 \pm 0.01}$	$\mathbf{0.92 \pm 0.01}$	9.75	0.57 ± 0.07	$\mathbf{0.92 \pm 0.01}$	10.00 ± 0.67

Interestingly, the value of the relative reconstruction errors obtained by the different algorithms are close to one another (for example, for $\epsilon = 10^{-2}$, the average is 10.30% for RAND, 10.82% for SNPA, 7.51% for SODA, and 7.52% for RAND+SODA), although SODA-based algorithms have a significantly higher clustering accuracy (for example, for $\epsilon = 10^{-2}$ at the first layer, the average accuracy is 48% for RAND, 42% for SNPA, and 97% for SODA and RAND+SODA). This illustrates the fact that on challenging settings such as the one of Fig. 1 for which the maximum distance between two points belonging to the same cluster might be larger than the inter-cluster distance, different ways of splitting the data lead to comparable reconstruction errors but rather different clustering accuracies. In other words, ONMF has many local minima with similar objective function values, so it is important to use proper initialization and algorithms to identify good solutions. In fact, SODA extracts initial basis vectors located at the center of each cluster, which is not guaranteed by the other approaches. Hence SODA should be preferred as an initialization technique as it performs significantly better in terms of accuracy.

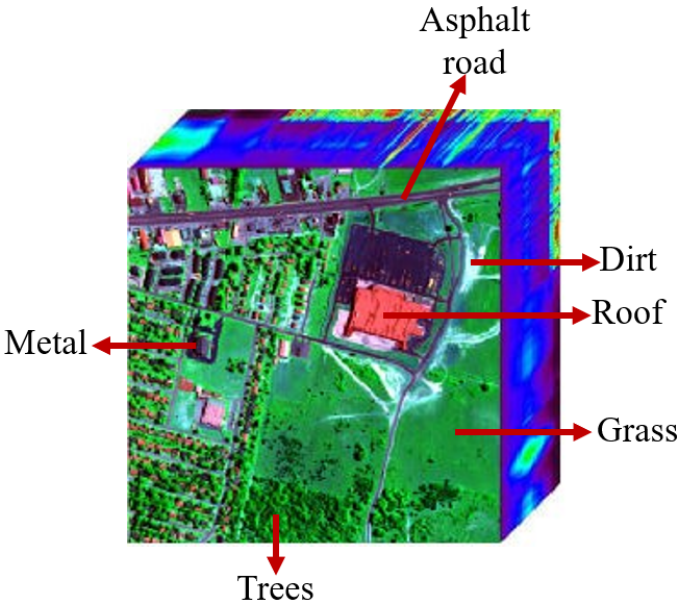

Fig. 2: Urban hyperspectral image.

B. Hyperspectral unmixing

A hyperspectral image (HI) contains the reflectance values of n pixels in m wavelength spectral bands and is generally represented by a matrix $X \in \mathbb{R}^{m \times n}$ where each column of X is the so-called spectral signature of each pixel. Hyperspectral unmixing (HU) consists in identifying the spectral signatures of r materials and under the linear mixing assumption, NMF is appropriate to solve HU [16]. Similarly, when deep ONMF is applied, the materials (also called endmembers) are extracted in a hierarchical bottom-up fashion.

The HYDICE Urban HI is made of $n = 307 \times 307$ pixels with $m = 162$ spectral bands; see Fig. 2. There are several versions of the ground truth depending on the number of materials considered [17].

The abundance maps, that is, the proportions in which every material appears in every pixel, extracted by deep ONMF, with $L = 6$, $d_l = 8 - l$ for all l are represented on Fig. 3. To initialize the factors of each layer, we first apply ONMF with $d_1 = 7$ with SNPA initialization and then apply SODA on W_1, while [6] used a multilayer ONMF with SNPA initialization of all layers. For conciseness, we gathered the representations of layer 3 and 4 as well as those of layer 5 and 6 in a single level as distinct clusters were merged at these layers. The first layer extracts two types of grass, trees, road, dirt, metal and roof. At layer 2, road and metal, which have similar spectral signatures, are merged in a single cluster. Then, the road/metal and dirt are merged to create a single cluster while the two kinds of grass are also merged. Finally, the road and roof are merged, while trees and grass are also gathered in a cluster made of vegetation.

Deep MF provides a richer decomposition than single-layer matrix factorization and the hybrid initialization combining SNPA with SODA is efficient to set up the factors.

V. Conclusion

In this paper, we explained why deep ONMF is equivalent to a particular bottom-up hierarchical clustering. We then proposed a greedy initialisation for deep ONMF, SODA, which was shown to outperform random initialization and SNPA on synthetic data sets, especially in situations with noise or when the clusters are quite close to each other. We emphasized the fact that similar (small) final reconstruction errors can be associated to various clustering accuracies hence a proper choice of the initialization technique is critical. We also showed that deep ONMF initialized with SODA-based algorithms are able to produce meaningful hierarchical decompositions in a hyperspectral image.

Future directions of research include to validate the proposed method on more data sets and other applications, such as topic modeling. Also, a thorough study of the robustness

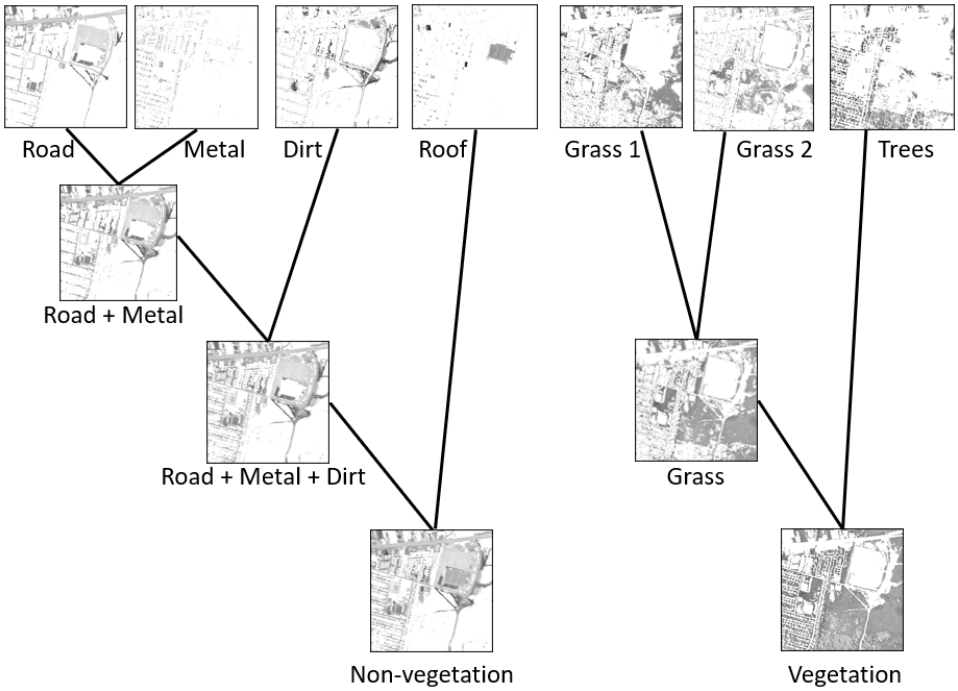

Fig. 3: Deep ONMF applied on the Urban HI.

to noise of SODA would be interesting. In fact, as long as the noise is sufficiently small, SODA provides an optimal clustering. This is obvious in the noiseless case, where all data points in the same cluster are multiples of one another, and should be quantified in noisy situations.

ACKNOWLEDGEMENT

This work was supported by the European Research Council (ERC starting grant n° 679515), and by the Fonds de la Recherche Scientifique - FNRS (F.R.S.-FNRS) and the Fonds Wetenschappelijk Onderzoek - Vlaanderen (FWO) under EOS Project no O005318F-RG47. Pierre De Handschutter is a research fellow of the F.R.S.-FNRS.

REFERENCES

[1] Pando Georgiev, Fabian Theis, and Andrzej Cichocki, "Sparse component analysis and blind source separation of underdetermined mixtures," *IEEE transactions on neural networks*, vol. 16, no. 4, pp. 992–996, 2005.

[2] Daniel D Lee and H Sebastian Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[3] Chris HQ Ding, Tao Li, Wei Peng, and Haesun Park, "Orthogonal nonnegative matrix t-factorizations for clustering," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 126–135.

[4] Andrzej Cichocki and Rafal Zdunek, "Multilayer nonnegative matrix factorisation," *Electronics Letters*, vol. 42, no. 16, pp. 947–948, 2006.

[5] George Trigeorgis, Konstantinos Bousmalis, Stefanos Zafeiriou, and Björn Schuller, "A deep matrix factorization method for learning attribute representations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 417–429, 2016.

[6] Pierre De Handschutter, Nicolas Gillis, and Xavier Siebert, "Deep matrix factorizations," *arXiv preprint arXiv:2010.00380*, 2020.

[7] Bensheng Lyu, Kan Xie, and Weijun Sun, "A deep orthogonal nonnegative matrix factorization method for learning attribute representations," in *International Conference on Neural Information Processing*. Springer, 2017, pp. 443–452.

[8] Filippo Pompili, Nicolas Gillis, P-A Absil, and François Glineur, "Two algorithms for orthogonal nonnegative matrix factorization with application to clustering," *Neurocomputing*, vol. 141, pp. 15–25, 2014.

[9] Athman Bouguettaya, Qi Yu, Xumin Liu, Xiangmin Zhou, and Andy Song, "Efficient agglomerative hierarchical clustering," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2785–2797, 2015.

[10] Shudong Huang, Zhao Kang, and Zenglin Xu, "Deep k-means: A simple and effective method for data clustering," in *International Conference on Neural Computing for Advanced Applications*. Springer, 2020, pp. 272–283.

[11] Da Kuang and Haesun Park, "Fast rank-2 nonnegative matrix factorization for hierarchical document clustering," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013, pp. 739–747.

[12] Yuqian Li, Diana M Sima, Sofie Van Cauter, Anca R Croitor Sava, Uwe Himmelreich, Yiming Pi, and Sabine Van Huffel, "Hierarchical nonnegative matrix factorization (hNMF): a tissue pattern differentiation method for glioblastoma multiforme diagnosis using MRSI," *NMR in Biomedicine*, vol. 26, no. 3, pp. 307–319, 2013.

[13] Nicolas Gillis, Da Kuang, and Haesun Park, "Hierarchical clustering of hyperspectral images using rank-two nonnegative matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2066–2078, 2015.

[14] Nicolas Gillis, "Successive nonnegative projection algorithm for robust nonnegative blind source separation," *SIAM Journal on Imaging Sciences*, vol. 7, no. 2, pp. 1420–1450, 2014.

[15] Seungjin Choi, "Algorithms for orthogonal nonnegative matrix factorization," in *2008 IEEE international joint conference on neural networks*. IEEE, 2008, pp. 1828–1832.

[16] José M Bioucas-Dias, Antonio Plaza, Nicolas Dobigeon, Mario Parente, Qian Du, Paul Gader, and Jocelyn Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 5, no. 2, pp. 354–379, 2012.

[17] Feiyun Zhu, "Spectral unmixing datasets with ground truths," *arXiv preprint arXiv:1708.05125*, 2017.

View publication stats