PIECEWISE CONSTANT NONNEGATIVE MATRIX FACTORIZATION

N. Seichepine[⋆] *S. Essid*[⋆] *C. Févotte*[†] *O. Cappé*[‡]

[⋆]Institut Mines-Télécom, Télécom ParisTech, CNRS LTCI
[†]Laboratoire Lagrange (CNRS, OCA & University of Nice) [‡]CNRS LTCI, Télécom ParisTech

ABSTRACT

In this paper we propose a non-negative matrix factorization (NMF) model with piecewise-constant activation coefficients. This structure is enforced using a total variation penalty on the rows of the activation matrix. The resulting optimization problem is solved with a majorization-minimization procedure. The proposed algorithm is well suited to analyze data explained by underlying piecewise-constant sequences of states. Its properties are first illustrated using synthetic data. We then use it to solve a video structuring problem that involves both segmentation and clustering tasks. An improvement over a state-of-the-art temporally smoothed NMF algorithm of both clustering and segmentation quality measures is observed.

Index Terms— Non-negative matrix factorization, temporal smoothing, total variation

1. INTRODUCTION

There exists numerous tasks where the data can be explained by locally invariant conditions: DNA sequences are formed with homogeneous sequences [1]; homogeneous region can be identified in images [2]; videos can be segmented into shots or more elaborate sequences [3], and a relatively stable spectral content is observed along one note in music transcription [4].

NMF is a standard tool to analyze non-negative data: its properties, assessed in [5], allow to identify recurrent *situations* (the *same* notes in audio recordings, *similar* sequences in videos), and to summarize data using only a few characteristic patterns grouped into a codebook, and associated activations. In its basic version, NMF makes no assumptions about the activations, which is suboptimal if data are temporally ordered and if we assume some specific knowledge: for the aforementioned problems, we expect *piecewise constant* activations (of some textures along one area or descriptors along time); it is therefore natural to encode this prior using a total variation penalty.

Building upon both the segmentation algorithms using total variation, and previous experimentations of NMF with temporal smoothing constraints, we propose a new kind of temporally regularized NMF. Section 2 describes our model, an effective algorithm is derived, and we discuss prior work. In Section 3, we study the properties of the proposed algorithm on both simulated data and a video structuring problem.

2. METHODOLOGY

After a brief reminder of NMF, we propose an optimization algorithm of a cost functional including a penalization of the total variation. We then discuss relations with prior work.

2.1. NMF and total variation penalty

The problem of NMF consists, given a matrix $V \in \mathbb{R}_+^{F\times N}$, in computing a matrix $W \in \mathbb{R}_+^{F\times K}$ and a matrix $H \in \mathbb{R}_+^{K\times N}$ such that $V \simeq WH$. The non-negativity of the coefficients of the matrices involved yields a non-subtractive, part-based representation [5]. The columns of W can be seen as patterns, or dictionary entries, while rows of H can be associated with regression coefficients, or activations. N corresponds to the number of observations (each column of V being one observation), K to the number of components, and F to the dimensionality of observations. For the tasks considered here, we look for solutions where $K \ll N$.

Considering some measure of fit D, the matrices W and H can be computed by looking at the optimization program $\min\ D(V\,|\,WH)$ with respect to W and H, under the non-negativity constraints $W \geq 0$ and $H \geq 0$. If we suppose that the columns of V correspond to regular, temporally ordered measurements, and try to favor temporal continuity by retaining solutions where rows of H tend to be piecewise constant, it is natural to formalize our problem via the following optimization problem, where h_{kn} denotes the coefficients of the matrix H:

$$\min_{W,H} D(V\,|\,WH) + \beta_s \sum_{k=1}^{K}\sum_{n=2}^{N} \left|h_{kn} - h_{k(n-1)}\right|, \qquad \text{s.t.} \quad W \geq 0,\, H \geq 0. \quad (1)$$

As illustrated below, the ℓ_1-norm used here will indeed favor solutions where the *derivative* of each row of H is sparse, and consequently piecewise constant activations.

Still, this simple implementation unfortunately leads to solutions where $\|H\|$ tends towards 0, since it is possible to reduce the penalty without affecting the "fit" part, by acting on the scale ambiguity between W and H. We consequently introduce the diagonal matrix $\Lambda \in \mathbb{R}^{K \times K}$, with k-th diagonal coefficient $\lambda_k = \sum_f w_{fk}$, which in fact corresponds to a rigorous variable change (see [3]), and consider the optimization program:

$$\min_{W,H} D(V \,|\, WH) + \beta_s \sum_{k=1}^{K} \sum_{n=2}^{N} \left| \lambda_k h_{kn} - \lambda_k h_{k(n-1)} \right| , \qquad \text{s.t.} \quad W \geq 0,\, H \geq 0. \tag{2}$$

The term β_s is simply a weighting hyperparameter, allowing us to choose solutions with arbitrarily settled temporal smoothness priors.

2.2. Optimization algorithm

We choose here to use a majorization-minimization (MM) algorithm [6, 7], that will sequentially update W and H. The MM algorithms consist in building, for each iterates, an auxiliary function G that majorizes the original cost function. G is sought to be easy to minimize and tight at the current values of W and H. Consequently, finding a minimizer of G will give a new iterate that corresponds to a lower value of the original cost function. The resulting algorithm – finding G for the current iterates, minimizing G with respect to W, finding G for the new iterates, minimizing G with respect to H, and iterate – can therefore be proven to return iterates corresponding to a strictly decreasing and convergent cost function.

The update rule is given below only for H, but can be derivated for W following the same process; moreover, in view of the application considered in Section 3, we treat here the situation where the measure of fit D corresponds to the sum over the coefficients of the element-wise generalized Kullback-Leibler divergence $D_{KL}(x|y) = x \log \left({}^{x}/_{y}\right) - x + y$; depending on the needs, it is possible to treat the situations where D is the Euclidian distance or the Itakura-Saito divergence, using a similar approach. Indeed, resorting to a convex-concave-constant decomposition, [7] propose a generic way to build the functions G when D is a β-divergence. We use similar notations: $\tilde{h}_{kn}$ are the coefficients of the current iterate of H, and denoting by $\psi_{kn} = \tilde{h}_{kn} \sum_{f=1}^{F} w_{fk} \left({}^{v_{fn}}/_{\sum_{k=1}^{K} (w_{fk} \tilde{h}_{kn})} \right)$ we get:

$$G(H|\tilde{H}) = \sum_{k=1}^{K} \sum_{n=1}^{N} \left(-\psi_{kn} \log h_{kn} + \lambda_k h_{kn} \right) + \beta_s \sum_{k=1}^{K} \sum_{n=2}^{N} \lambda_k \left| h_{kn} - h_{k(n-1)} \right| + \text{cst}. \tag{3}$$

$G(H|\tilde{H})$ is a mono-valued majorizing function, built at the point $\tilde{H}$. The dependency on $\tilde{H}$ is implicit below. We can take its subdifferential w.r.t. h_{kn} and obtain:

$$\partial_{h_{kn}} G(h_{kn}) = \frac{-\psi_{kn}}{h_{kn}} + \lambda_k + 2\beta_s \left|\lambda_k\right| \left(\operatorname{sign}\!\left(h_{kn} - h_{k(n-1)}\right) + \operatorname{sign}\!\left(h_{kn} - h_{k(n+1)}\right) \right) . \tag{4}$$

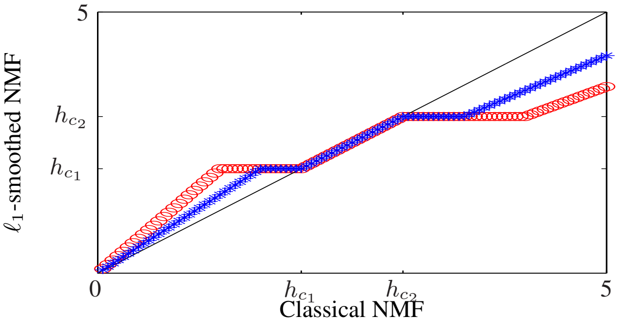

Fig. 1. Update $\bar{h}_{kn}$ returned by the proposed algorithm as a function of the update that would have been returned by a classical NMF in a similar situation, using respectively $\beta_s = 0.05$ (stars) and $\beta_s = 0.1$ (circles).

Where sign (x) is the sign function defined by sign $(x) = 1$ if $x > 0$, sign $(x) = -1$ if $x < 0$ and sign (0) is the interval $[-1, 1]$. From the equation 4, it is readily shown that there exists an unique positive value of h_{kn}, denoted by $\bar{h}_{kn}$, that cancels the subderivative and *minimizes* G. Denoting by $h_{c_1} = \min\left(h_{k(n-1)}, h_{k(n+1)}\right)$ and $h_{c_2} = \max\left(h_{k(n-1)}, h_{k(n+1)}\right)$, and using $^-$ and $^+$ to denote respectively their left and right limits, the solution can be summarized as follows:

$$\begin{cases} \bar{h}_{kn} = h_{c_1}, \text{ if } \partial_{h_{kn}} G(h_{c_1}^-) \leq 0 \text{ and } \partial_{h_{kn}} G(h_{c_1}^+) \geq 0 \\ \bar{h}_{kn} = h_{c_2}, \text{ if } \partial_{h_{kn}} G(h_{c_2}^-) \leq 0 \text{ and } \partial_{h_{kn}} G(h_{c_2}^+) \geq 0 \\ \bar{h}_{kn} = \frac{\psi_{\mathrm{kn}}/\lambda_{\mathrm{k}}}{1+4\beta_s\left(\mathbb{1}\left(\partial_{h_{kn}} G(h_{c_2}^+)<0\right) - \mathbb{1}\left(\partial_{h_{kn}} G(h_{c_1}^-)>0\right)\right)}, \text{ else.} \end{cases} \tag{5}$$

The update rule is illustrated in Figure 1, and can be interpreted as a soft thresholding: if the new iterate for h_{kn} is "naturally" (taking into account only the "fit" term) comprised between the values of the neighboring coefficients $h_{k(n-1)}$ and $h_{k(n+1)}$, each one "pulling" in opposite directions, the smoothing effect is canceled and the returned solution is exactly the same as for a classical NMF algorithm. Otherwise, if $\bar{h}_{kn}$ is far enough, depending on the value of β_s, from neighboring coefficients, it is simply shifted in comparison with a classical NMF solution; it is on the other hand *projected* onto the same value if one neighbor is close, resulting in constant segments.

It should be noted that each iteration has a linear cost of $\mathcal{O}(FKN)$. Using $K = 5$, $F = 128$ and $N = 12000$, a Matlab R2012b implementation needs around one minute to perform 300 iterations on a 2.8 GHz quad-core computer.

2.3. Related work

The idea to smooth activations in NMF has already been put into practice. Especially, [3] and [8] use an algorithm where an ℓ_2-norm is used in lieu of the proposed ℓ_1-norm. Temporal smoothness has also been enforced using other kind of penalties, related to Gamma chains, with both Kullback-Leibler [9] and Itakura-Saito [10] divergences, or explicitely modeling the ratio between short-term and long-term variance as in [11]. However, *all the proposed penalties* result in soft activations, only promoting a correlation between the columns of H, while we need *sharp* activations if we want to precisely identify and delimit areas in the data.

In a completely different context, it has been shown early in [12] that the total variation had a piecewise constant solutions-promoting property. This property has then been put to good use for segmentation tasks in [13], where a total variation penalty is used as a preprocessing step: the idea is to approximate available data by piecewise constant models, where the jumps are seen as a supplementary penalty.

This principle can be transposed to NMF; with the exception of the Euclidian distance as a measure of fit [14] the combination of NMF and total variation is new, yields sharp activations, and leads to an algorithm that is unsupervised.

3. EXPERIMENTAL VALIDATION

We now use the proposed algorithm, first on a synthetic example, then on a video structuring task, to illustrate its possibilities.

3.1. Illustration with simulated data

We build here a synthetic example, in order to illustrate the properties of the proposed algorithm: a matrix $V \in \mathbb{R}_+^{20\times 240}$ is computed, using a randomly generated matrix $W \in \mathbb{R}_+^{20\times 2}$ and a matrix $H \in \mathbb{R}_+^{2\times 240}$ with piecewise constant patterns of zeros and ones (see Figure 2). We then try to factorize $\tilde{V}$, a version of V corrupted by a Poisson noise, using different algorithms.

The results are presented in Figure 2, where the first row of the matrix H_{algo} returned by the algorithm is compared to the associated line of the generated ground truth H. We can see in (a) that the standard NMF algorithm gives noisy results, whereas the proposed algorithm returns piecewise constant ones in (b), as explained in Section 2.2. Finally, an algorithm where the ℓ_1-norm in the total variation is replaced by a ℓ_2-norm, as made in [3], also smooths the rows of H_{algo}, but the jumps are not entirely preserved, as illustrated in (c). Though the noise is not so high here that it would prevent a segmentation, it is clear that piecewise constant activations are of advantage in more delicate situations.

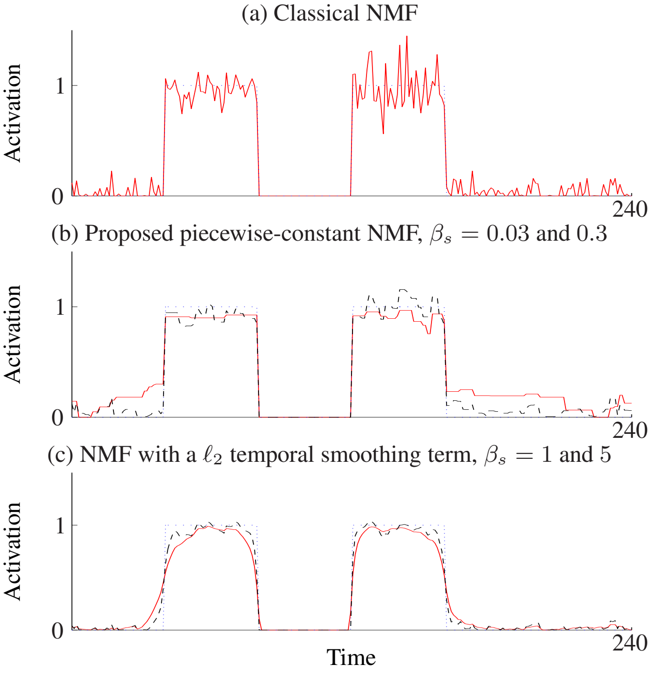

Fig. 2. Influence of the temporal smoothing terms. One row of H is displayed. The ground truth corresponds to dotted lines, the solutions given by the algorithms to dashed lines (first setting of β_s) and continuous lines (second setting).

3.2. Application to a video structuring task

The proposed algorithm is then tested on the 33 first videos of the *Canal9 political debates database* [15]. These videos feature different guests, and our objective is to identify the different phases: at each time, the algorithm must indicate who is onscreen, or answer that current images correspond to an establishing shot. The tests are made using 8 minute-long video sequences. One sequence is used per video, that starts at 3 minutes and 30 seconds.

The videos are first described as factorizable matrices V, following the process proposed by [3]: a codebook of visual words is first learned using PHOW descriptors and a K-means algorithm; then, the occurrences of the different words included in the images are aggregated over sliding windows. In the end, columns of V correspond to histograms of visual word occurrences. This justifies the use of a Kullback-Leibler divergence, since it is adapted to multinomial distributions [16, 17] and hence for the histograms data we use. Because of the properties of NMF, the factors W and H will respectively be formed with characteristic *video templates* and *template activations*, corresponding to onscreen speakers appearances. Simply thresholding these activations will therefore give the desired results.

We will compare our algorithm with a simple NMF, and the state of the art algorithm proposed in [3], that uses a ℓ_2 penalty to enforce the temporal smoothing prior. Choices must be made for the smoothing hyperparameter β_s for the

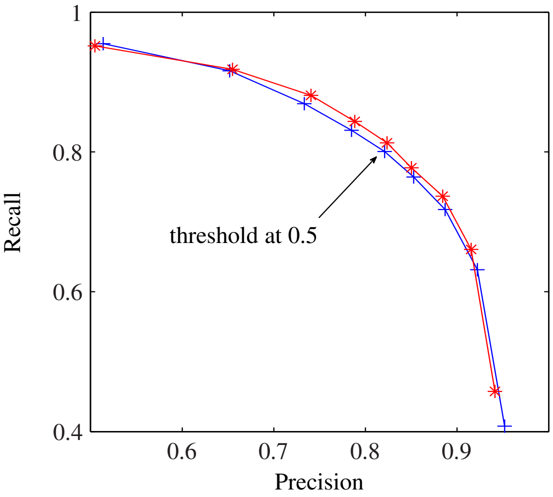

Fig. 3. ROC curves for the proposed total variation-based algorithm (stars) and an algorithm using a ℓ_2 penalty (crosses).

latter and the proposed algorithm, as well as for the used threshold. We retain the proposed value $\beta_s = 0.1$ for the ℓ_2-based algorithm, and available videos are separated into a development set of 10 videos and a test set of 23 videos. The development set is used to settle $\beta_s = 0.01$ for the studied ℓ_1-based algorithm. The thresholding is made as follows: each row of H is linearly rescaled to the unit interval, before a threshold of 0.5 (half the maximum) is used. This choice is retrospectively justified in Figure 3, where the mean precision and recall of shots identification are given, using different thresholds. The adopted value of 0.5 seems indeed to give the better trade-off for both algorithms.

Each algorithm is run 20 times, with random initializations; only the run corresponding to the final lowest cost functional is retained. The overall structuring errors, corresponding to the proportion of images wrongly labeled, are presented in Figure 4. The mean structuring error over the test set are respectively of 22.5% (standard NMF), 20.7% (ℓ_1) and 21.5% (ℓ_2), meaning that the proposed algorithm yields an improvement. By further investigating the results, we found that the mean rank correlations between the final cost functional and the structuring error, over the test set and along the different runs, were respectively of 0.3, 0.38 and 0.29. It is indeed expected that better optimization results will correspond to better solutions, and therefore that there exists a statistical *monotonic* relationship between the structuring error and the functional cost. The observed improvement of the structuring error is hence related with a better identifiability of the "good" results: the ℓ_2 penalty does not model optimally the considered problem, and leads to high costs for some solutions that were sensible from a structuring error viewpoint; on the other hand, the ℓ_1 norm does not penalize too much situations – corresponding to numerous segments in the activations – where (e.g.) speakers frequently interrupt each other, as long as they are well distinguished by the algorithm.

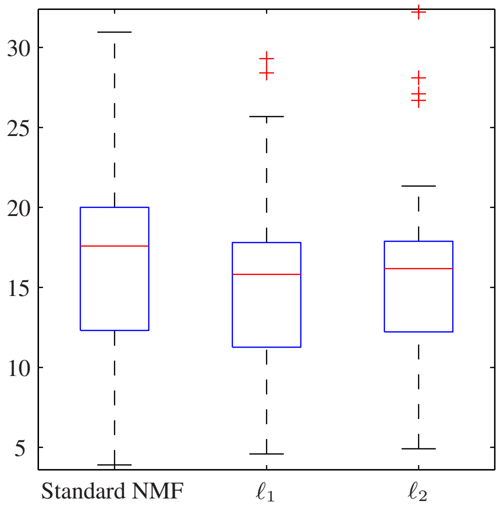

Fig. 4. Overall structuring error in % and box plots associated with per-video scores, respectively for a standard NMF algorithm, the proposed total variation-based algorithm, and using a ℓ_2 penalty.

Without even resorting to a threshold, we can perform a hard clustering of the shots, by labeling the data with the index of the row corresponding to the highest value in H at a given time. The quality of the clustering can be evaluated by computing the mean Rand index R [18] over the test set. Note that $R \in [0, 1]$, the highest values are the better. We obtain $R = 0.90$ and $R = 0.88$ for the ℓ_1 and ℓ_2 algorithms, respectively. Finally, the results can be evaluated from a segmentation point of view, by assessing the precision and the recall of the identification of jumps with a framewise accuracy. While we have a precision of 0.50 and a recall of 0.77 for the ℓ_1 algorithm, we obtain a precision of 0.48 and a recall of 0.74 for the ℓ_2 algorithm.

The proposed total variation-based algorithm systematically proves to be a preferable solution, using diversified evaluation metrics focusing on clustering or segmentation errors. It is furthermore very generic, and could be applied to all the situations mentioned in Section 1 where underlying segments can explain observations.

4. CONCLUSION

The penalization of total variation, usually restricted to denoising or segmentation problems, presents the interesting property to promote piecewise constant solutions. In this paper, we have shown that this penalization could also be used for NMF, with similar consequences. The proposed algorithm, taking advantage of the included temporal smoothness prior, facilitates clustering and segmentation tasks, as illustrated in Section 3. It could therefore be used for a wide variety of problems where we expect data to be explained by locally invariant conditions.

5. REFERENCES

[1] Richard Boys and Daniel Henderson, “A Bayesian approach to DNA sequence segmentation,” *Biometrics*, vol. 60, no. 3, pp. 573–81, Sept. 2004.

[2] Vivek Dey, Yun Zhang, and Ming Zhong, “A review on image segmentation techniques with remote sensing perspective,” in *International Society for Photogrammetry and Remote Sensing 7th Symposium*, 2010.

[3] Slim Essid and Cédric Févotte, “Smooth nonnegative matrix factorization for unsupervised audiovisual document structuring,” *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 415–425, Feb. 2013.

[4] Anssi Klapuri, *Signal processing methods for music transcription*, 2006.

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

[6] David Hunter and Kenneth Lange, “A Tutorial on MM Algorithms,” *The American Statistician*, vol. 58, no. 1, pp. 30–37, 2004.

[7] Cédric Févotte and Jérôme Idier, “Algorithms for Nonnegative Matrix Factorization with the β-Divergence,” *Neural Computation*, vol. 23, no. 9, pp. 2421–2456, Sept. 2011.

[8] Tuomas Virtanen, “Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 1066–1074, Mar. 2007.

[9] Tuomas Virtanen, Ali Taylan Cemgil, and Simon Godsill, “Bayesian extensions to non-negative matrix factorisation for audio signal modelling,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 1825–1828.

[10] Cédric Févotte, “Majorization-minimization algorithm for smooth Itakura-Saito nonnegative matrix factorization,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2011, pp. 1980–1983.

[11] Zhe Chen, Andrzej Cichocki, and Tomasz Rutkowski, “Constrained non-negative matrix factorization method for EEG analysis in early detection of Alzheimer’s disease,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 893–896.

[12] Leonid Rudin, Stanley Osher, and Emad Fatemi, “Nonlinear total variation based noise removal algorithms,” in *Proc. 11th International Conference of the Center for Nonlinear Studies on Experimental mathematics*, 1992, pp. 259–268.

[13] Kevin Bleakley and Jean-Philippe Vert, “The group fused Lasso for multiple change-point detection,” 2011, preprint hal-00602121.

[14] Haiqing Yin and Hongwei Liu, “Nonnegative matrix factorization with bounded total variational regularization for face recognition,” *Pattern Recognition*, vol. 31, no. 16, pp. 2468–2473, 2010.

[15] Alessandro Vinciarelli, Alfred Dielmann, Sarah Favre, and Hugues Salamin, “Canal9: A database of political debates for analysis of social interactions,” in *IEEE International Workshop on Social Signal Processing*, 2009.

[16] Jonathon Shlens, “Notes on Kullback-Leibler Divergence and Likelihood Theory,” 2007.

[17] Cédric Févotte and Ali Taylan Cemgil, “Nonnegative matrix factorizations as probabilistic inference in composite models,” in *Proc. 17th European Signal Processing Conference*, 2009, pp. 1913–1917.

[18] William Rand, “Objective criteria for the evaluation of clustering methods,” *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.