Pattern Recognition Letters 197 (2025) 353–358

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Orthogonal nonnegative matrix factorization with the Kullback–Leibler divergence

Jean Pacifique Nkurunziza [a,1], Fulgence Nahayo [a], Nicolas Gillis [b,*]

[a] LURMISTA/ISTA, University of Burundi, Boulevard de l'Indépendance 11, Bujumbura, 6934, Burundi
[b] University of Mons, Rue de Houdain 9, 7000 Mons, Belgium

ARTICLE INFO

Editor: Alexandru C. Telea

Keywords:
Orthogonal nonnegative matrix factorization
Clustering
Alternating optimization
Document classification
Hyperspectral imaging

ABSTRACT

Orthogonal nonnegative matrix factorization (ONMF) has become a standard approach for clustering. As far as we know, most works on ONMF rely on the Frobenius norm to assess the quality of the approximation. This paper presents a new model and algorithm for ONMF that minimizes the Kullback–Leibler (KL) divergence. As opposed to the Frobenius norm which assumes Gaussian noise, the KL divergence is the maximum likelihood estimator for Poisson-distributed data, which can model better sparse vectors of word counts in document data sets and photo counting processes in imaging. We develop an algorithm based on alternating optimization, KL-ONMF, and show that it performs favorably with the Frobenius-norm based ONMF for document classification and hyperspectral image unmixing.

1. Introduction

Given a data matrix $X \in \mathbb{R}^{m\times n}$ where each column, $X_{:j} \in \mathbb{R}^m$ for $j = 1, 2, \ldots, n$, corresponds to a data point, and a factorization rank r, orthogonal nonnegative matrix factorization (ONMF) [1] aims to find $W \in \mathbb{R}^{m\times r}$ and $H \in \mathbb{R}^{r\times n}$ such that

$$X \approx WH, \quad H \geq 0, \quad \text{and } HH^\top = I_r, \tag{1}$$

where I_r is the r-b-r identity matrix, and $(\cdot)^\top$ denotes the transpose of a matrix. The constraints on the matrix H make ONMF a clustering problem [1]: if a matrix H is component-wise nonnegative (that is, $H \geq 0$) and orthogonal (that is, $HH^\top = I_r$), then H has at most a single non-zero entry in each column (that is, in each $H_{:j}$). Hence each data point is approximated using a single column of W. ONMF has attracted a lot of attention and has been shown to perform well for various clustering tasks; see [1–9] and the references therein. As far as we know, most previous works on ONMF focused on the Frobenius norm to assess the quality of the approximation, that is, they focused on the following optimization problem: given $X \in \mathbb{R}^{m\times n}$ and r, solve

$$\min_{W\in\mathbb{R}^{m\times r}, H\in\mathbb{R}^{r\times n}} \|X - WH\|_F^2 \text{ s.t. } H \geq 0 \text{ and } HH^\top = I_r,$$

where $\|X\|_F^2 = \sum_{i,j} X_{i,j}^2$ is the squared Frobenius norm, and $X_{i,j}$ is the entry of matrix X at position (i,j). The only exceptions we were able to find are the papers [10,11] in which the authors consider the KL and Bregman divergences but they used a regularization term to promote orthogonality, leading to soft clusterings; in this paper we impose orthogonality as a hard constraint. The underlying assumption when using the Frobenius norm is that the noise follows a Gaussian distribution. However, in situations when Gaussian noise is not meaningful, other objective functions should be used. In particular, and this will be the focus of this paper, if the entries of the data matrix follow a Poisson distribution of parameter given by WH, then one should minimize the Kullback–Leibler (KL) divergence between X and WH, which is defined as

$$D_{\mathrm{KL}}(X, WH) = \sum_{i,j} D_{\mathrm{KL}}(X_{i,j}, (WH)_{i,j}),$$

where $D_{\mathrm{KL}}(x, y) = y - x + x\log\frac{x}{y}$, and, by convention, $D_{\mathrm{KL}}(0, y) = y$, while the KL divergence is not defined for $x < 0$. The KL divergence is for example much more meaningful than the Frobenius norm for sparse document data sets where each column of X is a vector of word count [12], and in some imaging applications [13,14].

difference between ONMF and k-means. Let us denote $\mathcal{K}_k$ the set that contains the non-zero indices of the kth row of H in (1), that is, $\mathcal{K}_k = \{j \mid H_{k,j} \neq 0\}$. We have

$$X_{:j} \approx W_{:k} H_{k,j} \quad \text{for all} \quad j \in \mathcal{K}_k.$$

* Corresponding author.
E-mail addresses: pacifiquenkuru997@gmail.com (J.P. Nkurunziza), fulgence.nahayo@ub.edu.bi (F. Nahayo), nicolas.gillis@umons.ac.be (N. Gillis).
URL: https://sites.google.com/site/nicolasgillis/ (N. Gillis).
[1] JPS is associated to the National Doctoral School of Burundi.

https://doi.org/10.1016/j.patrec.2025.08.012
Received 7 October 2024; Received in revised form 8 July 2025; Accepted 17 August 2025
Available online 15 September 2025

The data points are separated into r clusters, $\{\mathcal{K}_k\}_{k=1}^r$, whose centroids are given by the columns of W. Standard k-means (and variants that use other distances than the squared Euclidean distance [15]) approximates each data point with the cluster centroid, that is, $X_{:j} \approx W_{:k}$ for $j \in \mathcal{K}_k$. ONMF approximates each data point as a *multiple* of each centroid, and hence the angle between the centroids and the data points plays a more important role than their distances. In fact, ONMF was shown to be equivalent to a weighted variant of spherical k-means [6]; see also [16, pp. 188–189]. Note that (i) X and W are not required to be nonnegative, although this is often the case in practice, and (ii) if X is nonnegative, then the optimal W also is.

Outline and contribution of the paper. In this paper, we propose an algorithm for ONMF with the KL divergence, using alternating optimization. As for the Frobenius norm which is described in Section 2, we will be able to derive closed-form updates, and hence devise a simple, yet effective, algorithm for ONMF with the KL divergence, which we will refer to as KL-ONMF; see Section 3. We will show that it compares favorably with ONMF with the Frobenius norm for document classification and hyperspectral image unmixing; see Section 4.

2. Alternating optimization for ONMF with the Frobenius norm

Many algorithms have been developed for ONMF; see the references in Section 1. Let us recall the standard algorithm for ONMF based on alternating optimization proposed in [6], which we refer to as Fro-ONMF. It follows the same scheme as other clustering algorithms, in particular k-means that alternatively updates the centroids and the clusters; see also, e.g., [15] for generalizations to any Bregman divergences. Fro-ONMF updates W and H alternatively with closed-form expressions which can be derived as follows.

For H fixed, since W is unconstrained, the optimal solution must have its gradient equal to zero:

$$2(WH - X)H^\top = 0 \quad \Rightarrow \quad W = XH^\top,$$

since $HH^\top = I_r$. It is interesting to interpret this closed-form expression. Since H is orthogonal, there is a single non-zero entry per column, and recall $\mathcal{K}_k$ is the set that contains the non-zero indices of the kth row of H. We have

$$W_{:k} = X(:, \mathcal{K}_k)H(k, \mathcal{K}_k)^\top = \sum_{j \in \mathcal{K}_k} X_{:j} H_{k,j}, \tag{2}$$

where, given sets of indices $\mathcal{I}$ and $\mathcal{J}$, $X(\mathcal{I}, \mathcal{J})$ denotes the submatrix of X with row (resp. column) indices in $\mathcal{I}$ (resp. $\mathcal{J}$). This means that $W_{:k}$ is a *weighted* average of the data points (that is, the columns of X) belonging to the kth cluster corresponding to the kth row of H. The weight for each data point will depend on its norm since we will see that $H_{k,j}$ is equal to $\frac{X_{:j}^\top W_{:k}}{\|W_{:k}\|_2^2}$.

For W fixed, assume we know the position where $H_{:j}$ is different from zero, say $H_{k,j} \neq 0$. We must minimize $\|X_{:j} - W_{:k}H_{k,j}\|_2^2$ which is equal to

$$\|X_{:j}\|_2^2 - 2X_{:j}^\top W_{:k} H_{k,j} + H_{k,j}^2 \|W_{:k}\|_2^2.$$

If $X_{:j}^\top W_{:k} \geq 0$ (this will always hold when $X \geq 0$ and $W \geq 0$), the optimal $H_{k,j}^* = \frac{X_{:j}^\top W_{:k}}{\|W_{:k}\|_2^2}$, otherwise, $H_{k,j}^* = 0$. Hence $\|X_{:j} - W_{:k}H_{k,j}^*\|_2^2$ is equal to $\|X_{:j}\|_2^2 - \left(\frac{X_{:j}^\top W_{:k}}{\|W_{:k}\|_2}\right)^2$. Therefore, the non-zero entry of $H_{:j}$ will be the entry k that maximizes $\frac{W_{:k}^\top X_{:j}}{\|W_{:k}\|_2}$. Hence, to update H, we first normalize W so that its columns have unit ℓ_2 norm, then compute $W^\top X \in \mathbb{R}^{r \times n}$ and the maximum in each column will correspond to the non-zero entry in each column of H. Finally we update these entries with the above closed-form expression.

Note that, after this update, the rows of H might not have norm 1; which is required by the constraint $HH^\top = I_r$. This can be fixed, without changing the objective function value, by rescaling the solution WH since there is a scaling degree of freedom: for any $\alpha > 0$, $W_{:k}H_{k:} = (\alpha W_{:k})(H_{k:}/\alpha)$. If W is updated after H, the scaling of W is not necessary since it will be automatically scaled with the optimal closed-form expression provided in (2).

Algorithm 1 summarizes alternating optimization for ONMF with the Frobenius norm, which we refer to as Fro-ONMF. Note that X and W do not need to be nonnegative, ONMF in the Frobenius norm can be used to cluster data points with negative entries.

Algorithm 1 Fro-ONMF - alternating optimization for ONMF with the Frobenius norm [6]

Input: Data matrix: $X \in \mathbb{R}^{m \times n}$, Initialization: $W \in \mathbb{R}^{m \times r}$, factorization rank: $r \ll n$, maximum number of iterations: maxiter, convergence criterion: $\delta \ll 1$.
Output: Matrices $W \in \mathbb{R}^{m \times r}$ and $H \in \mathbb{R}_+^{r \times n}$ with $HH^\top = I_r$ such that $\|X - WH\|_F^2$ is minimized.

$t = 1$, $H = 1$, $H^{(p)} = 0$.
while $t \leq$ maxiter and $\|H - H^{(p)}\|_F \geq \delta$ **do**
  % Update H for W fixed
  $H^{(p)} \leftarrow H$. *% $H^{(p)}$ is the previous iterate to monitor convergence.*
  Normalized W: $W_n(:, k) = W_{:k}/\|W_{:k}\|_2$ for all k.
  Compute $A = W_n^\top X \in \mathbb{R}^{r \times n}$.
  Let $\mathcal{K}_k = \{j \mid A(k, j) > A(k', j) \text{ for all } k' \neq k\}$.
  Update $H(k, j) \leftarrow \frac{W_{:k}^\top X_{:j}}{\|W_{:k}\|_2^2}$ for all k and $j \in \mathcal{K}_k$.
  Scale H: $H_{k:} \leftarrow \frac{H_{k:}}{\|H_{k:}\|_2}$ for all k.
  % Update W for H fixed
  $W_{:k} \leftarrow X(:, \mathcal{K}_k)H(k, \mathcal{K}_k)^\top$ for all k.
  $t \leftarrow t + 1$
end while

Let us discuss three important aspects of Algorithm 1.

Computational cost. The main cost of Algorithm 1 is the matrix–matrix product $W_n^\top X$ which requires $O(\text{nnz}(X)r)$ operations, where $\text{nnz}(X)$ is the number of non-zero entries of X (it is equal to mn if X is dense). The other operations requires $O(\text{nnz}(X))$ operations or less, assuming $\text{nnz}(X) \gg \max(m, n)r$ (which should be the case otherwise the number of non-zero entries of X is smaller than the number of parameters in ONMF). This makes Algorithm 1 very fast and scalable.

Note that the update of W can be computed directly as $W \leftarrow XH^\top$. However, if using a dense matrix to represent H, this would require $O(\text{nnz}(X)r)$ operations, which would make this step r times slower. This is a new improvement of the implementation of Fro-ONMF whose previous version used the update $W \leftarrow XH^\top$, requiring $O(\text{nnz}(X)r)$ operations, making the total cost per iteration almost double.

Convergence. By construction, the objective function, $\|X - WH\|_F^2$, which is bounded below, decreases at each step, and hence objective function values converge. Since the feasible set for H is compact, and the level sets are compact (because $\|X - WH\|_F^2$ is coercive in W as H is normalized with non-zero rows, meaning that $\|X - WH\|_F^2$ goes to infinity as any entry of W goes to infinity), there exists a converging subsequence of the iterates.

Stopping criterion. Since H is normalized ($\|H_{k:}\|_2 = 1$ for all k), it makes sense to use the (cheap) stopping criterion $\|H - H^{(p)}\|_F < \delta$, where δ is a parameter smaller than 1. We will use $\delta = 10^{-6}$. We will also use a maximum of 100 iterations (which is never reached in our numerical experiments). Another important aspect of ONMF algorithms is the initialization of W which will be discussed in Section 4.

3. Alternating optimization for ONMF with the KL divergence

As explained in the introduction, we focus in this paper on ONMF with the KL divergence, which can be formulated as follows: given

$X \in \mathbb{R}^{m\times n}$ and r, solve

$$\min_{W\in\mathbb{R}^{m\times r}, H\in\mathbb{R}^{r\times n}} D_{\mathrm{KL}}(X, WH) \text{ s.t. } H \geq 0, HH^{\top} = I_r. \quad (3)$$

Note that, as opposed to ONMF with the Frobenius norm, the KL divergence requires X to be component-wise nonnegative, because the KL divergence is defined only for $X \geq 0$.

In the next two sections, we detail the closed-form expressions for W when H is fixed in (3), and vice versa.

3.1. Update W when H is fixed

Since H is orthogonal and non-negative, each column of W needs only to be optimized over a subset of the columns of X, as for Fro-ONMF. Recall the notation $\mathcal{K}_k = \{j | H_{k,j} > 0\}$. To find the optimal $W_{:k}$, we need to solve the following problem:

$$\min_{W_{:k}} \sum_{j\in\mathcal{K}_k} D_{\mathrm{KL}}(X_{:j}, W_{:k}H_{k,j}). \quad (4)$$

Lemma 1. *The optimal solution of the problem (4) is given by $W^*_{:k} = \frac{X(:,\mathcal{K}_k)e}{H_{k:}e}$, where e is the vector of all ones of appropriate dimension.*

Proof. The derivative w.r.t. $W_{:k}$ of $D_{\mathrm{KL}}(X, WH)$ is given by $ee^{\top}H^{\top}_{k:} - \frac{[X]}{[WH]}H^{\top}_{k:}$ [12]. Using the structure of (4) to simplify this expression, and setting the gradient to zero for optimality (as we will see, the unconstrained solution is nonnegative as long as X and H are), we obtain that $ee^{\top}H^{\top}_{k:}$ is equal to

$$ee^{\top}H(k,\mathcal{K}_k)^{\top} = \frac{[X(:,\mathcal{K}_k)]}{[W_{:k}H(k,\mathcal{K}_k)]}H(k,\mathcal{K}_k)^{\top} = \frac{[X(:,\mathcal{K}_k)e]}{[W_{:k}]}.$$

Since $e^{\top}H(k,\mathcal{K}_k)^{\top} = H(k,\mathcal{K}_k)e$, this gives $W^*_{:k} = \frac{X(:,\mathcal{K}_k)e}{H(k,\mathcal{K}_k)e}$ at optimality. □

It is interesting to note that the optimal $W_{:k}$ is a scaled average of the columns of X that are associated with the cluster k defined by the kth row of H. This is rather different than Fro-ONMF where $W_{:k}$ is a weighted average where the weights depend linearly on the norm of the data points; see the discussion around Equation (2) in Section 2. This implies that the centroids depend *quadratically* on the norms of the data points. Hence KL-ONMF will be less sensitive to outliers, and will give more importance to data points with smaller norm, as opposed to Fro-ONMF. The reason is that the KL divergence $\mathrm{KL}(x, y)$ grows linearly with y for y sufficiently large (as opposed to quadratically for the Frobenius norm). This will be illustrated in Section 4; in particular for hyperspectral images where KL-ONMF will be able to identify materials with small spectral signatures, while Fro-ONMF will not be able to do so.

3.2. Update H when W is fixed

As for the Frobenius norm, let us first assume we know the position at which $H_{:j}$ is non-zero, we will then select the entry that minimizes $D_{\mathrm{KL}}(X_{:j}, W_{:k}H_{k,j})$ the most.

Lemma 2. *Given $X_{:j} \geq 0$ and $W_{:k} \geq 0$, we have*

$$H^*_{k,j} = \operatorname*{argmin}_{H_{k,j}} D_{KL}(X_{:j}, W_{:k}H_{k,j}) = \frac{e^{\top}X_{:j}}{e^{\top}W_{:k}}.$$

Proof. Let us consider the problem $\min_{\alpha} D_{\mathrm{KL}}(X_{:j}, \alpha W_{:k})$, where α represents $H_{k,j}$. We need to minimize

$$\begin{aligned} f(\alpha) &= D_{\mathrm{KL}}(X_{:j}, \alpha W_{:k}) \\ &= \alpha e^{\top}W_{:k} - \sum_{i=1}^{m} X_{i,j}\log(\alpha W_{i,k}) \\ &= \alpha e^{\top}W_{:k} - e^{\top}X_{:j}\log\alpha - \sum_{i=1}^{m} X_{i,j}\log W_{i,k}. \end{aligned}$$

The derivative with respect to α is $f'(\alpha) = e^{\top}W_{:k} - \frac{1}{\alpha}e^{\top}X_{:j}$, so the optimal solution is given by $H^*_{k,j} = \alpha^* = \frac{e^{\top}X_{:j}}{e^{\top}W_{:k}}$. □

This means that $H_{k,j}$ is the ratio between the average entries in the corresponding columns of X and W.

Then, to decide which column of W is best for $X_{:j}$, let us compute the error we would get for each choice of $H^*_{k,j}$:

$$\begin{aligned} KL(X_{:j}, H^*_{k,j}W_{:k}) &= \frac{e^{\top}X_{:j}}{e^{\top}W_{:k}}e^{\top}W_{:k} - \sum_{i=1}^{m} X_{i,j}\log\left(\frac{e^{\top}X_{:j}}{e^{\top}W_{:k}}W_{i,k}\right) \\ &= e^{\top}X_{:j} - \sum_{i=1}^{m} X_{i,j}\log\left(\frac{W_{i,k}}{e^{\top}W_{:k}}\right) - e^{\top}X_{:j}\log(e^{\top}X_{:j}). \end{aligned}$$

The entry of the column of $H_{:j}$ that should be set to a positive value is the one that minimizes the quantity $KL(X_{:j}, H^*_{k,j}W_{:k})$, hence it corresponds to the index k such that $\sum_{i=1}^{m} X_{i,j}\log\left(\frac{W_{i,k}}{e^{\top}W_{:k}}\right)$ is maximized. That is, to select the column of W that best approximates the column $X_{:j}$ in the KL divergence, we pick the column $W_{:k}$ maximizing $\sum_{i=1}^{m} X_{i,j}\log\left(\frac{W_{i,k}}{e^{\top}W_{:k}}\right)$. Numerically, we construct W_n by normalizing the columns of W to have unit ℓ_1 norm, and then compute $A = \log(W_n + \epsilon)^{\top}X$, where ϵ is a small constant to avoid numerical issues, and the log is taken component-wise: the locations of the non-zero entries of the optimal H are given by the maximum of each column of A.

3.3. Algorithm KL-ONMF

Algorithm 2 summarizes the alternating optimization scheme, which we refer to as KL-ONMF.

Algorithm 2 KL-ONMF - alternating optimization for ONMF with the KL divergence

Input: Nonnegative matrix: $X \in \mathbb{R}^{m\times n}_{+}$, Initialization: $W \in \mathbb{R}^{m\times r}_{+}$, factorization rank: r, maximum number of iterations: maxiter, convergence criterion: $\delta \ll 1$, small regularization parameter: $\epsilon \ll 1$.

Output: Matrices $W \in \mathbb{R}^{m\times r}_{+}$ and $H \in \mathbb{R}^{r\times n}_{+}$ with $HH^{\top} = I_r$ such that $D_{\mathrm{KL}}(X, WH)$ is minimized.

```
t = 1, H = 1, H^(p) = 0.
while t ≤ maxiter and ‖H − H^(p)‖_F ≥ δ do
    % Update H for W fixed
    H^(p) ← H. % H^(p) is the previous iterate to monitor convergence.
    Normalized W_n: W_{:k} ← W_{:k}/e^T W_{:k} for all k.
    Compute A = log(W_n + ε)^T X ∈ R^{r×n}.
    Let K_k = {j | A(k, j) > A(k', j) for all k' ≠ k}.
    Update H(k, j) ← e^T X_{:j} / e^T W_{:k} for all k and j ∈ K_k.
    Scale H: H_{k:} ← H_{k:} / ‖H_{k:}‖_2 for all k.
    % Update W for H fixed
    W_{:k} ← X(:,K_k)e / H_{k:}e for all k.
    t ← t + 1
end while
```

Computational cost. As for Fro-ONMF (Algorithm 1), the main cost is a matrix–matrix product, namely computing $\log(W + \epsilon)^{\top}X$, which requires $O(\mathrm{nnz}(X)r)$ operations. The other operations requires $O(\mathrm{nnz}(X))$

operations or less, assuming $\text{nnz}(X) \gg \max(m, n)r$ as explained previously. This makes Algorithm 2 have essentially the same computational cost as Fro-ONMF, hence being very fast and scalable. We have observed in practice that KL-ONMF is slightly faster than Fro-ONMF, the reason is the update of W: KL-ONMF only needs to perform $\text{nnz}(X)+nr$ sums to compute $W_{:k} = X(:, \mathcal{K}_k)e/H_{k:}e$, while Fro-ONMF requires $\text{nnz}(X)$ sums and products for the update of $W_{:k} = X(:, \mathcal{K}_k)H(k, \mathcal{K}_k)^\top$.

Convergence. The same observations as for Fro-ONMF apply: the objective function values will converge while there is a converging subsequence of iterates.

Stopping criterion. As for Fro-ONMF, we use the stopping criterion $\|H - H^{(p)}\|_F < \delta \ll 1$, and we will use $\delta = 10^{-6}$. We will also use a maximum of 100 iterations, which is never reached in our numerical experiments. The parameter ϵ that allows to take the log of W is set to 10^{-3}.

4. Numerical experiments

In this section, we compare the performance of Fro-ONMF (Algorithm 1) and KL-ONMF (Algorithm 2) for clustering documents (Section 4.1) and pixels in hyperspectral images (Section 4.2). As mentioned in the introduction, it is meaningful to use the KL divergence for such data sets: documents are represented as vector of word counts for which a Poisson counting process makes more sense than additive Gaussian noise [12], while images might be thought of as a photon counting process [13,14]. The code in MATLAB is available from https://gitlab.com/ngillis/kl-onmf, and can be used to rerun all experiments. All experiments were run on a LAPTOP Intel(R) Core(TM) i7-8850H CPU @ 2.60 GHz 16,0Go RAM.

Initialization. There are many ways to initialize ONMF algorithms, as they are for k-means. To simplify the presentation, we use the approach proposed in [16]. It initializes W with the successive nonnegative projection algorithm (SNPA) [17] that identifies a subset of r columns of X that represent well-spread data points in the data set.

4.1. Document data sets

We first use ONMF to cluster the 15 document data sets from [18]. Table 1 reports the name of the data sets as well as their dimensions (m is the number of words, n the number of documents) and the number of clusters (r). For each data set, Table 1 reports the accuracy in percent of the clustering obtained by Fro-ONMF and KL-ONMF, denoted acc-F and acc-KL, respectively. Given the true disjoint clusters $C_i \subset \{1, 2, \dots, n\}$ for $1 \le i \le r$ and given a computed disjoint clustering $\{\tilde{C}_i\}_{i=1}^r$, the clustering *accuracy* is defined as

$$\text{accuracy}\left(\{\tilde{C}_i\}_{i=1}^r\right) = \max_{\pi \in [1,2,\dots,r]} \frac{1}{n} \sum_{i=1}^{r} |C_i \cap \tilde{C}_{\pi(i)}|,$$

where $[1, 2, \dots, r]$ is the set of permutations of $\{1, 2, \dots, r\}$. The last row of Table 1 reports the weighted average accuracy, where the weight is the number of documents to be clustered. Table 1 also reports the average run times in seconds over 20 runs, denoted time-F and time-KL, respectively, as well as the number of iterations needed for the two algorithms to converge, denoted it-F and it-KL, respectively. For comparison, we also report the best accuracy for 6 ONMF algorithms reported in [6] which we denote acc-best (only for 12 out of the 15 data sets, because [6] did not consider the largest data sets to reduce the computational load). We observe the following:

- On average, KL-ONMF provides significantly better clustering, with 59,2% accuracy compared to 41,2% for Fro-NMF. However, for some data sets, Fro-NMF sometimes provides significantly better solutions (for k1b and tr23).
- Compared to the 6 other ONMF algorithms (which used different initializations), KL-ONMF still performs best on average.

Table 1
Fro-ONMF vs. KL-ONMF for the clustering of 15 document data sets: m is the number of words, n is the number of documents, r is the number of clusters, acc-F and acc-KL are the accuracies in percent for Fro-ONMF and KL-ONMF, resp., acc-best is the best results among 6 ONMF algorithms reported in [6], time-F and time-KL are the run times in seconds for Fro-ONMF and KL-ONMF, resp., and it-F and it-KL are the number of iterations needed to converge for Fro-ONMF and KL-ONMF, resp. The best result between Fro-ONMF and KL-ONMF is highlighted in bold.

	n	m	r	acc-F	acc-KL	acc-best	time-F	time-KL	it-F	it-KL
NG20	19 949	43 586	20	26.5	**48.8**	NA	5.5	**2.3**	38	**34**
ng3sim	2998	15 810	3	38.0	**71.4**	NA	0.3	**0.1**	24	**11**
classic	7094	41 681	4	55.9	**85.4**	58.8	1.1	**0.3**	97	**37**
ohscal	11 162	11 465	10	29.1	**47.7**	39.2	1.3	**0.6**	**31**	36
k1b	2340	21 819	6	**74.4**	58.5	79.0	0.5	**0.1**	40	**10**
hitech	2301	10 080	6	**39.6**	38.5	48.7	0.4	**0.1**	28	**18**
reviews	4069	18 483	5	51.4	**72.4**	63.7	0.4	**0.2**	14	**12**
sports	8580	14 870	7	48.9	**66.6**	50.0	1.2	**0.6**	**24**	29
la1	3204	31 472	6	56.2	**61.2**	65.8	0.7	**0.3**	28	**25**
la12	6279	31 472	6	45.7	**67.5**	NA	2.7	**0.7**	59	**29**
la2	3075	31 472	6	**62.2**	59.9	52.8	0.6	**0.2**	27	**13**
tr11	414	6424	9	50.5	**54.1**	50.2	0.1	**0.0**	19	**9**
tr23	204	5831	6	**43.1**	34.3	41.2	0.0	0.0	**8**	16
tr41	878	7453	10	44.2	**48.6**	53.2	0.2	**0.1**	25	**15**
tr45	690	8261	10	42.2	**59.6**	41.4	0.1	**0.1**	13	**10**
Averages				41.2	**59.2**	54.0	1.0	**0.4**	32	**20**

Table 2
Fro-ONMF vs. KL-ONMF for the clustering of 3 HSIs: m is the number of spectral bands, n is the number of pixels, r is the number of endmembers, MRSA-F and MRSA-KL are the average MRSA compared to the ground truth for Fro-ONMF and KL-ONMF, resp., time-F and time-KL are the run times in seconds for Fro-ONMF and KL-ONMF, resp., and it-F and it-KL are the number of iterations needed to converge for Fro-ONMF and KL-ONMF, resp. The best result is highlighted in bold.

	m	n	r	MRSA-F	MRSA-KL	time-F	time-KL	it-F	it-KL
Moffet	159	2500	3	26.1	**7.77**	**0.09**	0.11	10	10
Samson	156	9025	3	**2.74**	2.75	0.22	0.22	7	**6**
Jasper	198	10 000	4	19.6	**3.6**	**0.49**	0.50	22	**16**

- KL-ONMF requires on average fewer iterations than Fro-ONMF to converge while its cost per iteration is slightly cheaper (see the discussion in Section 3.3), leading to faster computational times, on average more than 2 times faster.

4.2. Hyperspectral images

A hyperspectral image (HSI) can be represented as an m-by-n matrix where m is the number of wavelength measured which is typically between 100 and 200, as opposed to 3 bands for color images (red, green and blue), and n is the number of pixels. When most pixels in a HSI contain a single material, referred to as an endmember in the HSI literature, it makes sense to cluster them according to the endmember they contain; see, e.g., [19] and the references therein.

Let us compare Fro-ONMF and KL-ONMF on three widely used data sets: Moffet, Samson and Jasper. For these data sets, we have good estimates of the ground-truth endmembers; see [20] and the references therein. Hence we can evaluate the quality of the endmembers extracted by an ONMF algorithm by comparing it to the ground truth. The standard metric to do so is the mean removed spectral angle (MRSA) between two vectors $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^m$:

$$\text{MRSA}(x, y) = \frac{100}{\pi} \cos^{-1}\left(\frac{(x - \bar{x}e)^\top (y - \bar{y}e)}{\|x - \bar{x}e\|_2 \|y - \bar{y}e\|_2}\right),$$

where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$. Note that $\text{MRSA}(x, y) \in [0, 100]$, and the smaller the MRSA, the better approximation y is of x. Table 2 reports the

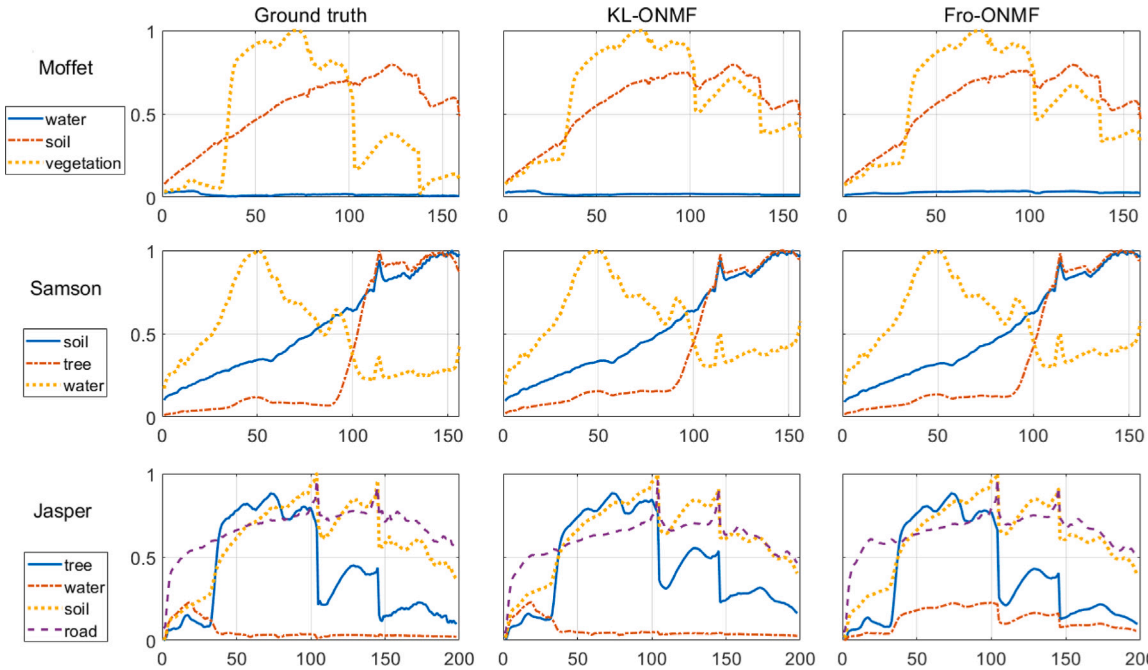

Fig. 1. Spectral signatures extracted with KL-ONMF and Fro-ONMF on hyperspectral images (these correspond to the columns of the computed W factor); from top to bottom: Moffet, Samson and Jasper. The x-axis represent the band number. These curves correspond to the cluster centroids: the more similar, the harder the clustering problem.

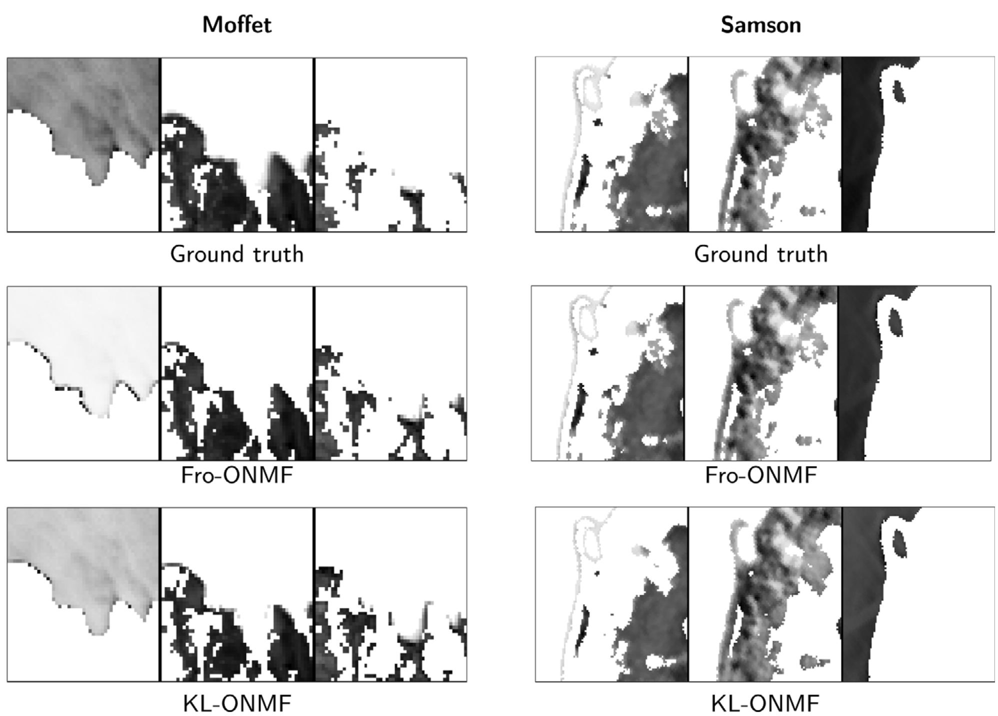

Fig. 2. Clustering of the Moffet HSI (from left to right: water, soil and vegetation), and the Samson HSI (from left to right: soil, tree, water). The first row corresponds to the ground truth. Here KL-ONMF performs better than Fro-NMF to extract the water in the Moffet data set (see also Fig. 1).

average MRSA between the columns of the estimated W's and the ground-truth W_t, using an optimal permutation. We observe the following:

- For the Samson data set, Fro-ONMF and KL-ONMF provide very similar results.
- For the Moffet and Jasper data sets, KL-ONMF outperforms Fro-ONMF, with significantly smaller MRSA. The reason is that these two data sets contains an endmember with small norm (that is, a column of W with small norm) corresponding to the water; see Fig. 1. Since Fro-NMF tends to favor endmembers with large

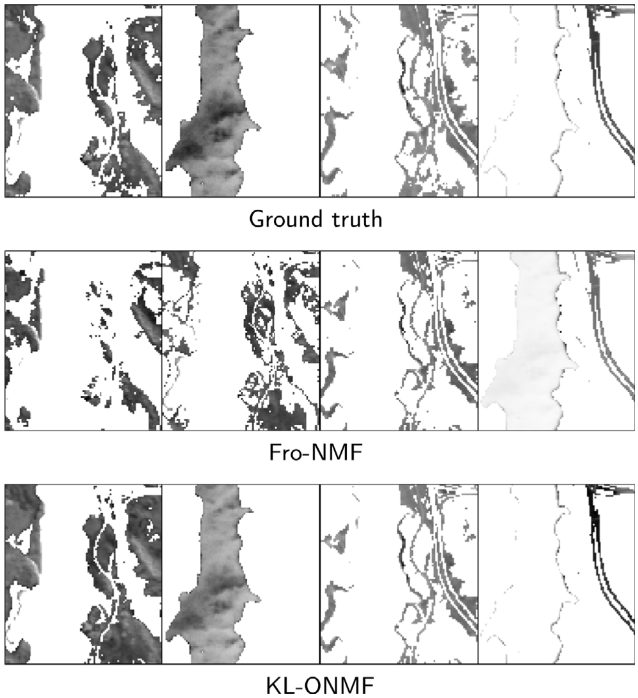

Fig. 3. Clustering of the Jasper HSI. From left to right: tree, water, soil and road.

norms, it is unable to extract the water properly in these two data sets, and hence has a significantly larger MRSA than KL-ONMF. Figs. 2 and 3 display the clustering of the pixels for the three HSIs for Fro-ONMF and KL-ONMF. We observe in fact that KL-ONMF perfectly extracts the water on the Moffet and Jasper data sets, while Fro-ONMF completely fails to do so.

- In terms of runtime and number of iterations, Fro-ONMF and KL-ONMF have similar performances.

5. Conclusion

In this paper, we have proposed a new clustering model for nonnegative data, namely orthogonal NMF with the KL divergence. We designed an alternating optimization algorithm, which is simple but effective and highly scalable, running in $O(\text{nnz}(X)r)$ operations where $\text{nnz}(X)$ is the number of non-zero entries in the data matrix, and r is the number of clusters. We showed on documents and hyperspectral images that KL-ONMF performs favorably with ONMF with the Frobenius norm, as it provides, on average, better clustering results, while running faster on average. Further work include the generalization of Algorithm 2 to any Bregman divergence [15].

CRediT authorship contribution statement

Jean Pacifique Nkurunziza: Writing – review & editing, Writing – original draft, Software, Methodology, Investigation, Conceptualization. **Fulgence Nahayo:** Writing – review & editing, Supervision, Conceptualization. **Nicolas Gillis:** Writing – review & editing, Writing – original draft, Supervision, Software, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

NG acknowledges the support by the European Union (ERC consolidator, eLinoR, no 101085607). The authors are grateful to the anonymous reviewers whose feedback helped us improve our paper.

Data availability

Data will be made available on request.

References

[1] C. Ding, T. Li, W. Peng, H. Park, Orthogonal nonnegative matrix t-factorizations for clustering, in: Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., ACM, 2006, pp. 126–135.
[2] S. Choi, Algorithms for orthogonal nonnegative matrix factorization, in: Proc. of the Int. Joint Conf. on Neural Networks, 2008, pp. 1828–1832.
[3] J. Yoo, S. Choi, Orthogonal nonnegative matrix factorization: Multiplicative updates on stiefel manifolds, in: Intelligent Data Engineering and Automated Learning, 2008, pp. 140–147.
[4] Z. Yang, E. Oja, Linear and nonlinear projective nonnegative matrix factorization, IEEE Trans. Neural Netw. 21 (2010) 734–749.
[5] B. Li, G. Zhou, A. Cichocki, Two efficient algorithms for approximately orthogonal nonnegative matrix factorization, IEEE Signal Process. Lett. 22 (2014) 843–846.
[6] F. Pompili, N. Gillis, P.A. Absil, F. Glineur, Two algorithms for orthogonal nonnegative matrix factorization with application to clustering, Neurocomputing 141 (2014) 15–25.
[7] A. Mirzal, A convergent algorithm for orthogonal nonnegative matrix factorization, J. Comput. Appl. Math. 260 (2014) 149–166.
[8] M. Asteris, D. Papailiopoulos, A.G. Dimakis, Orthogonal NMF through subspace exploration, in: Advances in Neural Information Processing Systems, NIPS, 2015, pp. 343–351.
[9] S. Wang, T.H. Chang, Y. Cui, J.S. Pang, Clustering by orthogonal non-negative matrix factorization: A sequential non-convex penalty approach, in: ICASSP, 2019, pp. 5576–5580.
[10] H. Li, T. Adal, W. Wang, D. Emge, A. Cichocki, A. Cichocki, Non-negative matrix factorization with orthogonality constraints and its application to Raman spectroscopy, J. Signal Process. Syst. 48 (2007) 83–97.
[11] K. Kimura, M. Kudo, Y. Tanaka, A column-wise update algorithm for nonnegative matrix factorization in bregman divergence with an orthogonal constraint, Mach. Learn. 103 (2016) 285–306.
[12] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.
[13] W.H. Richardson, Bayesian-based iterative method of image restoration, J. Opt. Soc. Am. 62 (1972) 55–59.
[14] L.B. Lucy, An iterative technique for the rectification of observed distributions, Astron. J. 79 (745) (1974).
[15] A. Banerjee, S. Merugu, I.S. Dhillon, J. Ghosh, J. Lafferty, Clustering with Bregman divergences, J. Mach. Learn. Res. 6 (2005).
[16] N. Gillis, Nonnegative Matrix Factorization, SIAM, Philadelphia, 2020.
[17] N. Gillis, Successive nonnegative projection algorithm for robust nonnegative blind source separation, SIAM J. Imaging Sci. 7 (2014) 1420–1450.
[18] S. Zhong, J. Ghosh, Generative model-based document clustering: A comparative study, Knowl. Inf. Syst. 8 (2005) 374–384.
[19] N. Gillis, D. Kuang, H. Park, Hierarchical clustering of hyperspectral images using rank-two nonnegative matrix factorization, IEEE Trans. Geosci. Remote Sens. 53 (2014) 2066–2078.
[20] F. Zhu, Hyperspectral unmixing: ground truth labeling, datasets, benchmark performances and survey, 2017, arXiv preprint arXiv:1708.05125.