DOI
https://doi.org/10.1007/s00285-019-01355-2

Orthogonal joint sparse NMF for microarray data analysis

Flavia Esposito[1,2] · **Nicolas Gillis**[3] · **Nicoletta Del Buono**[1,2]

Abstract
The 3D microarrays, generally known as gene-sample-time microarrays, couple the information on different time points collected by 2D microarrays that measure gene expression levels among different samples. Their analysis is useful in several biomedical applications, like monitoring dose or drug treatment responses of patients over time in pharmacogenomics studies. Many statistical and data analysis tools have been used to extract useful information. In particular, nonnegative matrix factorization (NMF), with its natural nonnegativity constraints, has demonstrated its ability to extract from 2D microarrays relevant information on specific genes involved in the particular biological process. In this paper, we propose a new NMF model, namely Orthogonal Joint Sparse NMF, to extract relevant information from 3D microarrays containing the time evolution of a 2D microarray, by adding additional constraints to enforce important biological proprieties useful for further biological analysis. We develop multiplicative updates rules that decrease the objective function monotonically, and compare our approach to state-of-the-art NMF algorithms on both synthetic and real data sets.

Keywords NMF · Microarray · Sparsity · Orthogonal · Gene expression · Metagene

Mathematics Subject Classification MSC 65 · MSC 92

✉ Flavia Esposito
flavia.esposito@uniba.it

Nicolas Gillis
nicolas.gillis@umons.ac.be

Nicoletta Del Buono
nicoletta.delbuono@uniba.it

[1] Department of Mathematics, University of Bari Aldo Moro, via E. Orabona 4, 70125 Bari, Italy

[2] INDAM Research Group GNCS, Roma, Italy

[3] Department of Mathematics and Operational Research, Université de Mons, Rue de Houdain 9, 7000 Mons, Belgium

1 Introduction

Microarray data analysis (MDA) aims to analyze gene expression data obtained using microarray experiments to extract information among genes, accross different conditions and different samples (Moschetta et al. 2013). Depending on the goal of the biomedical investigation, microarray experiments produce different type of data sets. Generally, these data sets can be divided into two classes, namely 2D and 3D microarrays. The 2D microarrays are the most used: they generate gene-sample and gene-time data sets; the first being a static set of data simultaneously recording gene expression levels on different samples; the latter registering the evolution of gene expression levels measured on one sample over different time points. The 3D microarrays, generally known as gene-sample-time microarrays, couple the information collected by 2D microarrays and measure gene expression levels among different samples on different time points. They are characteristic of some specific biomedical contexts such as monitoring drug activities on stabilized tumor cells (Zhang 2006; Borgwardt et al. 2006).

From a mathematical point of view, gene expression levels can be represented as vectors in a specific vector space. In this way, a 2D microarray is recorded as a real nonnegative matrix $X \in \mathbb{R}_+^{n \times m}$ for gene-sample microarray or $Y \in \mathbb{R}_+^{n \times T}$ for a gene-time microarray where n is the number of genes, m the number of samples, and T the number of time points. A third-order tensor $\chi \in \mathbb{R}_+^{n \times m \times T}$ can be used as a multilinear algebra representation of a 3D microarray. Figure 1 illustrates the structure of a third-order gene-sample-time tensor.

To extract information from this large amount of data and to reveal the dense network of knowledge embedded in this structured representation, different approaches have been used in the literature (Kouskoumvekaki et al. 2013; Glaab et al. 2011; Kong et al. 2011; Yang and Michailidis 2015). Among them, dimensionality reduction is a key and powerful technique (Dai et al. 2006; Boccarelli et al. 2018; Nikulin

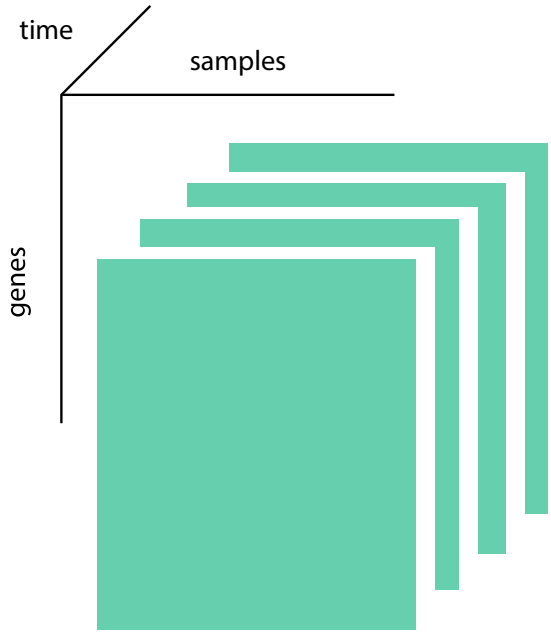


Fig. 1 Structure of a gene-sample-time data set

et al. 2012). Several algebraic matrix decompositions were adopted to tackle 2D microarrays; examples include the singular value decomposition (SVD) and principal component analysis (PCA) (Alter et al. 2000; Wall et al. 2003), independent component analysis (ICA) (Kong et al. 2008), network component analysis (Liao et al. 2003) and non-negative matrix factorization (NMF) (Brunet et al. 2004; Liu et al. 2008; Li and Ngom 2010). Among them, NMF reveals to be one of the most suitable techniques for managing 2D microarrays. In fact, NMF naturally fits the non-negativity of the microarray data providing a useful instrument for learning part-based representations (Del Buono et al. 2016). Multilinear algebra decompositions have also been applied to reduce the dimensionality and extract features from microarray tensors. For example, different gene-time microarrays were combined to construct and study an artificial 3D microarray via the higher-order SVD (HOSVD) (Omberg et al. 2007), the tensor version of the SVD. Multilinear ICA has been studied to classify integrated tumor gene expression data obtained in different scenarios (Du et al. 2009) and high-order NMF has been used to predict positive or negative responders to Interferon beta (IFNβ) treatments (Li and Ngom 2011).

NMF is able to extract from 2D microarrays relevant information on specific genes involved in the particular biological process (Del Buono et al. 2016). In this paper, we propose a new NMF model, namely Orthogonal Joint Sparse NMF (OJSNMF), to extract relevant information from 3D microarrays containing the time evolution of a 2D microarray.

The paper is organized as follows. In Sect. 2, we provide a brief introduction of the NMF approach applied to gene sample microarrays. This allows us to motivate the proposed OJSNMF model and its application to the layers of 3D microarrays. Section 3 is devoted to the design of specific updates rules to tackle OJSNMF. Section 4 reports numerical results of OJSNMF applied on synthetic data sets and to the data set from (Baranzini et al. 2004; Li and Ngom 2011) to study the response to IFNβ. Section 5 discusses future works, addressing open problems related with the applications of the proposed algorithm to real data and to the biological interpretation of the obtained results.

2 Orthogonal joint sparse NMF

In this section, we briefly review the use of NMF to analyze 2D microarray, and present our new proposed model to tackle 3D microarrays.

2.1 2D microarray analysis: NMF approach

In 2D microarrays, gene expressions are collected in a non-negative matrix $X \in \mathbb{R}_+^{n\times m}$, in which rows correspond to different genes and columns to samples (which may represent distinct tissues, experiments, patients, conditions or time points). Hence the (i, j)th entry of X, denoted x_{ij}, indicates the expression level of the ith gene in the jth sample (Brunet et al. 2004; Kim and Park 2007a). NMF approximates the data matrix X as the product of two non-negative matrices $W \in \mathbb{R}_+^{n\times r}$ and $H \in \mathbb{R}_+^{r\times m}$ so

that $X \approx WH$. Each column $X_{:,j}$ of the matrix X is reconstructed via a non-negative linear combination of the columns of the basis matrix W, weighted with coefficients of the matrix H:

$$X_{:,j} \approx \sum_{k=1}^{r} W_{:,k} H_{kj} \quad j = 1, \ldots, m.$$

The r columns of the basis matrix W are called *metagenes*. Each metagene is a basis vector whose entries indicate the importance of each gene in this particular metagene (e.g., $W_{ik} = 0$ means that the ith gene is not part of the kth metagene). These metagenes are such that the subspace they generate represents the most significant information hidden in the data. The scalar H_{kj} reveals the effect of the kth metagene on the jth sample.

The number of metagenes, that is, the rank r of the factorization, is problem dependent and usually specified by the user. In biomedical fields (especially in MDA), this value is usually chosen by some empirical technique (Del Buono et al. 2016; Brunet et al. 2004; Hutchins et al. 2008). It is worth observing that the decomposition $X \approx WH$ can also be interpreted row-wise, where rows of H are metasamples.
The nonnegativity of the factors make the heatmap representations of the low-rank approximation a useful tool for understanding the factorization results. Particularly, heatmap reports individual values of X, W and H as colors, as illustrated in Fig. 2 where a rank-2 reduction of some data matrix is shown.

NMF is usually written as a non-linear constrained optimization problem:

$$\min_{W \geq 0, H \geq 0} \mathrm{D}(X, WH), \tag{1}$$

where $\mathrm{D}(\cdot, \cdot)$ denotes some divergence function

$$D : \mathbb{R}_{+}^{n \times m} \times \mathbb{R}_{+}^{n \times m} \to \mathbb{R}_{+},$$

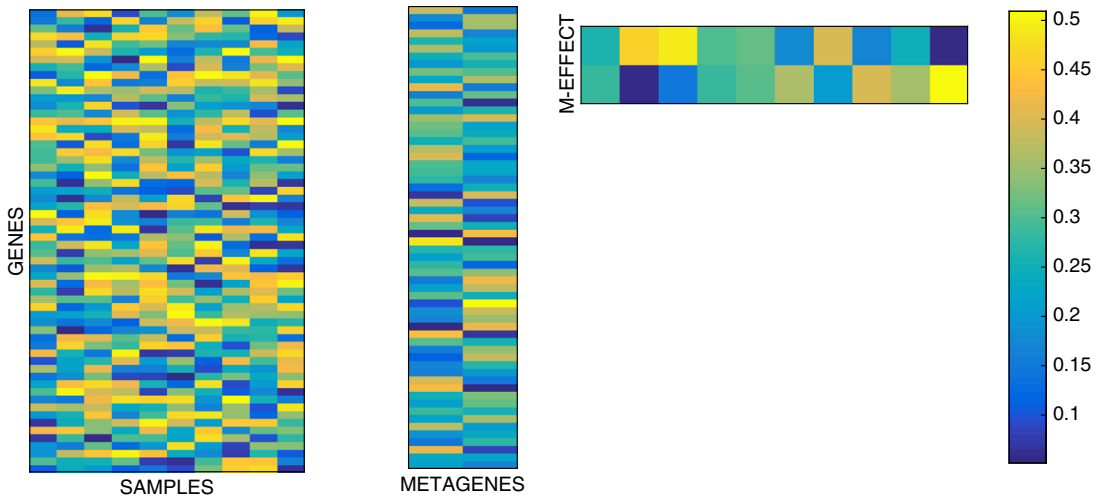


Fig. 2 Heatmap of a rank-2 NMF

typically satisfying the following properties: (i) it is continuously differentiable (at least once) in both variables, (ii) it is individually convex in W and H, and (iii) it equals 0 if and only if $X = WH$ (Dhillon et al. 2005). One of the most used divergence in MDA is the generalized Kullback-Leibler (KL) divergence:

$$KL(X, WH) = \sum_{i,j} X_{ij} \log\left(\frac{X_{ij}}{(WH)_{ij}}\right) - X_{ij} + (WH)_{ij},$$

which corresponds to the maximum likelihood estimation under independent Poisson noise. Note that the terms i and j for which $X_{ij} = 0$ are not well defined. An equivalent definition of the KL divergence, which explicitly assumes $0 \log(0) = 0$, is

$$KL(X, WH) = \sum_{(i,j)\,|\,X_{ij}>0} X_{ij} \log\left(\frac{X_{ij}}{(WH)_{ij}}\right) - \sum_{i,j}\left(X_{ij} - (WH)_{ij}\right).$$

Over the years, many constrained NMF problems have been proposed in the literature; see, e.g., (Cichocki et al. 2009). In the next section, we propose a new constrained NMF model, dubbed orthogonal joint sparse NMF, that is particularly well suited to analyze gene-sample-time 3D microarrays.

2.2 Orthogonal joint sparse NMF for 3D microarray analysis

Let us denote $X_t \in \mathbb{R}_+^{n\times m}$, where $t = 1, \ldots, T$ represents the time, the slices of a third-order gene-sample-time tensor χ that represent microarrays, where n denotes the number of genes and m the number of samples. In this paper, our goal is to find factor matrices $W_t \in \mathbb{R}_+^{n\times r}$ and $H_t \in \mathbb{R}_+^{r\times m}$ for each $t = 1, \ldots, T$ such that $X_t \approx W_t H_t$, where r is the number of latent factors. Recall that the columns of W_t represent metagenes, while each column of H_t provides the activation of these metagenes in the corresponding column of X_t. In this section, we propose the following model to analyze such data sets, which we refer to as orthogonal joint sparse NMF (OJSNMF):

$$\min_{\substack{H_t\geq 0\\ W_t\geq 0}} \sum_{t=1}^{T}\left(KL(X_t, W_t H_t) + \lambda\|W_t\|_1 + \frac{1}{2}\alpha\left\|\overline{W}_t - W_t\right\|_F^2 + \frac{1}{2}\gamma\left\|W_t^\top W_t - \mathbb{I}_r\right\|_F^2\right) \tag{2}$$

where λ, α and γ are positive penalty parameters, $\|\cdot\|_1$ is the component-wise ℓ_1 norm of a matrix, $\|\cdot\|_F$ is the Frobenius norm of a matrix and $\overline{W}_t$ is defined as follows:

$$\overline{W}_t = \begin{cases} W_{t+1} & \text{for } t = 1,\\ \frac{W_{t+1}+W_{t-1}}{2} & \text{for } 2 \leq t \leq T-1,\\ W_{t-1} & \text{for } t = T. \end{cases} \tag{3}$$

The objective function in (2) contains four terms. The first term is the KL divergence between the data matrix X_t and its approximation $W_t H_t$, for all time $t = 1, \ldots, T$. The

remaining penalization terms on the metagene matrices W_t's are related to biological and operational constraints in order to improve the decomposition:

- The second term in (2), $\|W_t\|_1$, will increase the sparsity of W_t. In fact, the ℓ_1 norm is a well-known surrogate for the ℓ_0 norm; see, e.g., (Mairal et al. 2014). The sparsity of W_t ensures the selection of a few genes in each metagene, which is a desirable property when treating a huge amount of genes simultaneously (Kim and Park 2007b). Although the orthogonality term will already enforce sparsity of W_t, the ℓ_1 penalty will enhance this property, so that some rows of W_t could be equal to zero, that is, some genes could be absent of all metagenes, which is typical in practice.
- The third term in (2), $\left\|\overline{W}_t - W_t\right\|_F^2$, ensures that the metagene matrices W_ts are similar, and evolve smoothly over time. In fact, the matrices W_ts computed for the different time steps are linked together by enforcing similar patterns between different time points t of the same metagenes. These assumptions are frequent with time-series data in the biomedical field. In fact, a similar approach to smooth coefficients on the factors matrices was used for analyzing time-series electromyography data with a post-processing 2D-interpolation (Cheung et al. 2015). However, to the best of our knowledge, this is the first time that this hypothesis emerges in MDA context with an algorithmic approach that incorporates the assumption in the minimization process as part of the objective function.
- The fourth term in (2), $\left\|W_t^\top W_t - \mathbb{I}_r\right\|_F^2$, promotes the columns of W_t to be orthogonal. This term is in accordance to particular gene extraction techniques proposed in the literature (Carmona-Saez et al. 2006; Kim and Park 2007a), where *the extracted metagenes have no or small overlap*. In fact, the columns of a nonnegative matrix are orthogonal if and only if the sparsity pattern of its columns are disjoint (Ding et al. 2005; Pompili et al. 2014). Moreover, this term enforces the columns of W_t to have their ℓ_2 norms close to one which provides some normalization and prevents the columns of W_t to have different scales or get close to zero.

It is worth noting that the OJSNMF (2) generalizes previous NMF models: sparse NMF (when using $T = 1$ and $\alpha = \gamma = 0$) and orthogonal NMF (when using $T = 1$ and $\lambda = \gamma = 0$), and we will compare OJSNMF to these in Sect. 4.

3 Multiplicative updates for OJSNMF

In this section, we propose an algorithm based on multiplicative updates to tackle (2). As for most NMF algorithms, we will use an alternating strategy, that is, we optimize alternatively over the factor matrices W_t and H_t for $t = 1, 2, \ldots, T$. Algorithm 1 provides a pseudocode to tackle (2).

Since updating (W_t, H_t) for different t's is the same optimization problem, we focus in the next section to update a single pair (W_t, H_t), that we will denote (W, H) for simplicity.

Algorithm 1: Alternating multiplicative updates for OJSNMF (2)

Data: $X_t \in \mathbb{R}_+^{n\times m}$, $t = 1, \ldots, T$, factorization rank r, penalty parameters λ (sparsity), α (coherence between time steps), γ (orthogonality)
Result: $W_t \in \mathbb{R}_+^{n\times r}$, $H_t \in \mathbb{R}_+^{r\times m}$, $t = 1, \ldots, T$
begin
  Choose some initial matrices $W_t \in \mathbb{R}_+^{n\times r}$, $H_t \in \mathbb{R}_+^{r\times m}$, $t = 1, \ldots, T$
  while *some stopping criterion is met* **do**
    for $t = 1, 2, \ldots, T$ **do**
      Update H_t using update from (5)
      Update W_t using update from (6)

3.1 Optimizing over a single (W_t, H_t)

In this section, we consider the minimization of $F(W, H)$ defined as

$$F(W, H) = KL(X, WH) + \lambda||W||_1 + \frac{1}{2}\alpha \left\|\overline{W} - W\right\|_F^2 + \frac{1}{2}\gamma \left\|W^\top W - \mathbb{I}\right\|_F^2. \quad (4)$$

The corresponding optimization problem is non-convex in both unknowns W and H. However, it is convex with respect to the variable H, and we can use the original updates of Lee and Seung (2000) that are guaranteed to decrease $F(W, H)$ for W fixed. In fact, H only appears in the first term of F, namely, $KL(X, WH)$, and the updates are the following

$$H_{aj} \leftarrow H_{aj} \frac{\sum_b W_{ba} X_{bj}/(WH)_{bj}}{\sum_b W_{ba}} \qquad \forall a, j. \quad (5)$$

Note that non-increasingness of $F(W, H)$ under the above updates is guaranteed whether all entries of H are updated sequentially or simultaneously. The reason is that these updates have been derived by minimizing exactly an auxiliary function for F (a function that is equal to F at the current iterate H and larger everywhere else; see Definition 1 below) which is separable in the entries of H –the auxiliary function has the form $\sum_{a,j} f(H_{aj})$. The update we will develop for W (Theorem 1) will share the same property. Note also that, as for all multiplicative updates, the above updates cannot modify zero entries which may prevent them to converge to a stationary point. A possible fix to this issue is to use a lower bound, say 10^{-16} for the entries of W and H, which guarantees convergence to stationary points (Gillis and Glineur 2012; Takahashi and Hibi 2014). We adopt this strategy for the numerical experiments in Sect. 4.

3.1.1 Optimizing over W

The problem for W is more complicated than for H, due to the non-convex orthogonality penalty term $\left\|W^\top W - \mathbb{I}\right\|_F^2$. We now derive multiplicative updates for W that

are guaranteed to decrease the objective function $F(W, H)$ for H fixed. The main theoretical contribution of this work is to prove the following theorem.

Theorem 1 *The function* (4) *is non increasing under the following update rule:*

$$W_{ia} \leftarrow W_{ia}\sqrt{\frac{-B_{ia}+\sqrt{\Delta_{ia}}}{2A^*_{ia}}} \qquad \forall i, a, \tag{6}$$

where

- $\Delta_{ia} = B_{ia}^2 - 4\frac{A^*_{ia}}{W_{ia}^2}C_{ia}$,
- $A^*_{ia} = 2\left[2\alpha + \gamma\left(\frac{(WW^\top W)_{ia}}{W_{ia}} + 2\right)\right]$,
- $B_{ia} = \frac{\sum_{j=1}^m H_{aj}+\lambda}{W_{ia}} - 3\alpha - 6\gamma$,
- $C_{ia} = -\sum_{j=1}^m X_{ij}\frac{W_{ia}H_{aj}}{(WH)_{ij}} - \alpha\overline{W}_{ia}W_{ia}$.

and $W_{ia} \geq 0$. *Non-increasingness is guaranteed whether these updates are performed simultaneously (that is, all* i, a *are updated independently) or sequentially.*

Note that for $\alpha = \gamma = 0$, the update are not well defined since $A^*_{ia} = 0$. However, computing the limit of the update for $(\alpha, \gamma) \to (0, 0)$, we obtain

$$\lim_{(\alpha,\gamma)\to(0,0)} W_{ia}\sqrt{\frac{-B_{ia}+\sqrt{\Delta_{ia}}}{2A^*_{ia}}} = W_{ia}\sqrt{\frac{\sum_j X_{ij}H_{aj}/(WH)_{ij}}{\sum_j H_{aj}+\lambda}},$$

so that the update becomes

$$W_{ia} \leftarrow W_{ia}\sqrt{\frac{\sum_j X_{ij}H_{aj}/(WH)_{ij}}{\sum_j H_{aj}+\lambda}}.$$

Interestingly, these updates are the same as in (Liu et al. 2003), but with a square root factor. These updates can be used to tackle sparse NMF. In order to prove Theorem 1, we will break up the objective function in different terms and, for each term, we will provide an auxiliary function. The sum of these auxiliary functions will be an auxiliary function for the original objective function and allows us to derive the multiplicative updates from Theorem 1. For simplicity, in the following, we will denote $F(W, H) = F(W)$ for H fixed, which can be written as the sum of the following terms

$$F(W) = F_1(W) + F_2(W) + \alpha F_3(W) + \frac{1}{2}\alpha F_4(W) + \gamma F_5(W) + \frac{1}{2}\gamma F_6(W) + \frac{1}{2}\gamma r + \frac{1}{2}\alpha \operatorname{Tr}(\overline{WW}^\top),$$

where the functions F_i's are defined as follows:

- $F_1(W) = \sum_{i=1}^n \sum_{j=1}^m -X_{ij}\log(WH)_{ij}$,

- $F_2(W) = \sum_{i=1}^{n} \sum_{a=1}^{r} W_{ia} \left(\sum_{j=1}^{m} H_{aj} + \lambda \right)$,
- $F_3(W) = -\sum_{i=1}^{n} \sum_{a=1}^{r} W_{ia} \overline{W}_{ia}$,
- $F_4(W) = \mathrm{Tr}\,(WW^\top)$,
- $F_5(W) = -\mathrm{Tr}\,(WW^\top)$,
- $F_6(W) = \mathrm{Tr}\,(WW^\top WW^\top)$.

In fact, we have

$$\frac{1}{2}\gamma \left\| W^\top W - \mathbb{I} \right\|_F^2 = \frac{1}{2}\gamma\, \mathrm{Tr}(WW^\top WW^\top) - \gamma\, \mathrm{Tr}(WW^\top) + \frac{1}{2}\gamma r, \text{ and}$$
$$\frac{1}{2}\alpha \left\| \overline{W} - W \right\|_F^2 = \frac{1}{2}\alpha\, \mathrm{Tr}\,(WW^\top) - \alpha\, \mathrm{Tr}\,(W\overline{W}^\top) + \frac{1}{2}\alpha\, \mathrm{Tr}\,(\overline{WW}^\top).$$

Let us define formally an auxiliary function.

Definition 1 Let $F : \mathbb{R}^{m\times r} \to \mathbb{R} : W \to F(W)$. Given $W^s \in \mathbb{R}^{m\times r}$, an auxiliary function $f : (\mathbb{R}^{m\times r})^2 \to \mathbb{R} : (W, W^s) \to f(W, W^s)$ for $F(W)$ at W_s is a function which satisfies the following two conditions:

1. $f(W^s, W^s) = F(W^s)$, and
2. $f(W, W^s) \geq F(W)\ \forall W$.

The aim of such an auxiliary function is to provide an upper approximation of F that matches F at the point W^s (which will be the current iterate in our algorithm). Hence, the matrix $W^{s+1} = \mathrm{argmin}_W\, f(W, W^s)$ will satisfy

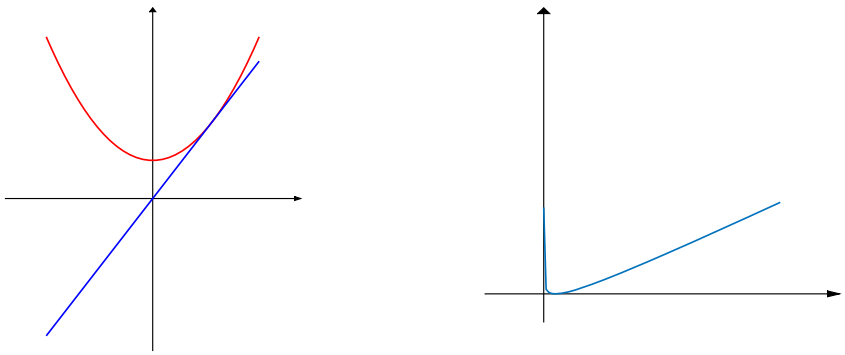
$$F(W^{s+1}) \leq f(W^{s+1}, W^s) \leq f(W^s, W^s) = F(W^s).$$

Given that the minimum of f can be computed efficiently, this procedure therefore provides us with a monotonic algorithm to minimize F: $W^1 \to W^2 \to \ldots$ such that $F(W^{s+1}) \leq F(W^s)$ for all $s \geq 1$. In the following lemmas, we provide an auxiliary function for F by deriving an auxiliary function for each F_i $(1 \leq i \leq 6)$. It has to be noted that these auxiliary functions were chosen so that f can be minimized easily. In fact, we designed these auxiliary functions so that the zeros of the derivative of f satisfy an equation of the type $ax^4 + bx^2 + c = 0$ whose roots can be computed easily, which in turn will allow us to minimize f exactly over the nonnegative orthant. For the first term, we use the same auxiliary function as in (Lee and Seung 2000).

Lemma 1 (Lee and Seung (2000)) *The function*

$$f_1(W, W^s) = -\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{a=1}^{r} X_{ij} \frac{W^s_{ia} H_{aj}}{\sum_{b=1}^{r} W^s_{ib} H_{bj}} \times \left(\log\left(W_{ia} H_{aj}\right) - \log\left(\frac{W^s_{ia} H_{aj}}{\sum_{b=1}^{r} W^s_{ib} H_{bj}} \right) \right)$$

is an auxiliary function for $F_1(W)$.



(a) Parabola of f_2 **(b)** Positive term of f_3

Fig. 3 Auxiliary functions

Lemma 2 gives a convex parabola greater than the straight line, suitable to our purposes; see Fig. 3a for an illustration.

Lemma 2 *The function*

$$f_2(W, W^s) = \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{c_a}{2W_{ia}^s} W_{ia}^2 + \frac{c_a}{2} W_{ia}^s,$$

where $c_a = \sum_{j=1}^{m} H_{aj} + \lambda \geq 0$, *is an auxiliary function for* $F_2(W)$.

Proof The first condition for f_2 to be an auxiliary function for F_2 can be checked easily. For the second condition, we have

$$\begin{aligned} f_2(W, W^s) - F_2(W) &= \sum_{i}^{n}\sum_{a=1}^{r} \frac{c_a}{2W_{ia}^s} W_{ia}^2 + \frac{c_a}{2} W_{ia}^s - c_a W_{ia} \\ &= \frac{1}{2}\sum_{i}^{n}\sum_{a=1}^{r} \frac{c_a}{W_{ia}^s} \left(W_{ia}^2 - 2W_{ia}W_{ia}^s + (W_{ia}^s)^2 \right) \\ &= \frac{1}{2}\sum_{i}^{n}\sum_{a=1}^{r} \frac{c_a}{W_{ia}^s} \left(W_{ia} - W_{ia}^s \right)^2 \geq 0. \end{aligned}$$

□

The following Lemma gives us a logarithmic auxiliary function for $F_3(W)$.

Lemma 3 *The function*

$$f_3(W, W^s) = \sum_{i=1}^{n}\sum_{a=1}^{r} -\overline{W}_{ia} W_{ia}^s \log\left(\frac{e}{W_{ia}^s} W_{ia} \right)$$

is an auxiliary function for $F_3(W)$.

Proof The first condition is straightforward. For the second condition, we have

$$\begin{aligned} f_3(W, W^s) - F_3(W) &= \sum_{i,a} -\overline{W}_{ia} W^s_{ia} \log\left(\frac{e}{W^s_{ia}} W_{ia}\right) + W_{ia}\overline{W}_{ia} \\ &= \sum_{i,a} \overline{W}_{ia} W^s_{ia} \left[-\log\left(\frac{e}{W^s_{ia}} W_{ia}\right) + \frac{W_{ia}}{W^s_{ia}}\right] \\ &= \sum_{i,a} \overline{W}_{ia} W^s_{ia} \left[-1 - \log\left(\frac{W_{ia}}{W^s_{ia}}\right) + \frac{W_{ia}}{W^s_{ia}}\right] \geq 0. \end{aligned}$$

Figure 3b displays the function $-1 - \log(x) + x$ for $x \geq 0$ which is nonnegative for $x \geq 0$ because the function $x - 1$ is the tangent of the concave function $\log(x)$ at $x = 1$, which completes the proof since $\frac{W_{ia}}{W^s_{ia}} \geq 0$. □

Lemmas 4 and 5provide two four degree polynomials as auxiliary functions for $F_4(W)$ and $F_5(W)$, respectively

Lemma 4 *The function:*

$$f_4(W, W^s) = \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{2}{(W^s_{ia})^2} W^4_{ia} - 3W^2_{ia} + 2(W^s_{ia})^2$$

is an auxiliary function for $F_4(W)$.

Proof The first condition to prove that f_4 is an auxiliary function for $F_4(W)$ is straightforward. For the second condition we have

$$\begin{aligned} f_4(W, W^s) - F_4(W) &= \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{2}{(W^s_{ia})^2} W^4_{ia} - 3W^2_{ia} + 2(W^s_{ia})^2 - W^2_{ia} \\ &= \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{2}{(W^s_{ia})^2} \left(W^4_{ia} - 2W^2_{ia}(W^s_{ia})^2 + (W^s_{ia})^4\right) \\ &= \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{2}{(W^s_{ia})^2} \left(W^2_{ia} - (W^s_{ia})^2\right)^2 \geq 0. \end{aligned}$$

□

Lemma 5 *The function*

$$f_5(W, W^s) = \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{W^4_{ia}}{(W^s_{ia})^2} - 3W^2_{ia} + (W^s_{ia})^2$$

is an auxiliary function for $F_5(W)$.

Proof Again, the first condition can be checked easily. For the second condition, we have

$$\begin{aligned} f_5(W, W^s) - F_5(W) &= \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{W_{ia}^4}{(W_{ia}^s)^2} - 3W_{ia}^2 + (W_{ia}^s)^2 + W_{ia}^2 \\ &= \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{1}{(W_{ia}^s)^2}\left(W_{ia}^4 - 2W_{ia}^2(W_{ia}^s)^2 + (W_{ia}^s)^4\right) \\ &= \sum_{i=1}^{n}\sum_{a=1}^{r} \frac{1}{(W_{ia}^s)^2}\left(W_{ia}^2 - (W_{ia}^s)^2\right)^2 \geq 0. \end{aligned}$$

□

Before providing the auxiliary function for F_6, let us recall a result by (He et al. 2011) in the case of symmetric NMF.

Lemma 6 (He et al. 2011) *Let be $A \in \mathbb{R}^{n\times n}$ a non-negative and symmetric matrix and $W^s \in \mathbb{R}^{n\times r}$. Then, for all k and $W \in \mathbb{R}^{n\times r}$, we have*

$$\sum_{i,j} A_{ij} W_{ik}^2 W_{jk}^2 \leq \sum_{i} \frac{\sum_j A_{ij}(W_{jk}^s)^2}{(W_{ik}^s)^2} W_{ik}^4.$$

Using this lemma, we propose the last auxiliary function for F_6.

Lemma 7 *The function*

$$f_6(W, W^s) = \sum_{i,a} \frac{(W^s(W^s)^\top W^s)_{ia}}{(W^s)_{ia}^3} W_{ia}^4$$

is an auxiliary function for $F_6(W)$.

Proof The first condition of an auxiliary function is easy to show:

$$f_6(W, W) = \sum_{i,j} \frac{(WW^\top W)_{ij}}{W_{ij}^3} W_{ij}^4 = \sum_{i,j} (WW^\top W)_{ij} W_{ij} = \mathrm{Tr}(WW^\top WW^\top).$$

For the second condition, we have

$$\begin{aligned} F_6(W) &= \sum_{i,j} (WW^\top)_{ij}^2 = \sum_{ij} \left(\sum_k W_{ik} W_{jk}\right)^2 \\ &= \sum_{ij} \left(\sum_k c_k \frac{W_{ik} W_{jk}}{c_k}\right)^2 \leq \sum_{ij} \left(\sum_k c_k \left(\frac{W_{ik} W_{jk}}{c_k}\right)^2\right), \end{aligned}$$

where $\sum_k c_k = 1$. The last inequality holds because of the convexity of the quadratic function. Choosing $c_k = \frac{W^s_{ik} W^s_{jk}}{(W^s (W^s)^\top)_{ij}}$, we have

$$F_6(W) \leq \sum_{i,j,k} \frac{(W^s (W^s)^\top)_{ij}}{W^s_{ik} W^s_{jk}} W^2_{ik} W^s_{jk}.$$

The matrix A defined with $A_{ij} = \frac{(W^s (W^s)^\top)_{ij}}{W^s_{ik} W^s_{jk}}$ for all i, j is symmetric and non negative. By Lemma 6, we have

$$F_6(W) \leq \sum_k \sum_{ij} A_{ij} W^2_{ik} W^2_{jk} \leq \sum_{i,k} \frac{\sum_j A_{ij} (W^s_{jk})^2}{(W^s_{ik})^2} W^4_{ik}.$$

□

The following corollary follows from the previous lemmas since F is a linear combination of the F_i's with nonnegative coefficients.

Corollary 1 *The function*

$$\begin{aligned} f(W, W^s) = & f_1(W, W^s) + f_2(W, W^s) + \alpha f_3(W, W^s) + \frac{1}{2}\alpha f_4(W, W^s) \\ & + \gamma f_5(W, W^s) + \frac{1}{2}\gamma f_6(W, W^s) + \frac{1}{2}\gamma r + \frac{1}{2}\alpha \operatorname{Tr}(\overline{W W}^\top) \end{aligned}$$

is an auxiliary function for $F(W)$.

In order to prove Theorem 1, it remains to show that the minimum of f is attained by the update given in (6).

Proof (Theorem 1) First, note that the function $f(W, W^s)$ is separable in the entries of W, that is, there is no interaction between these variables. Hence, as for the original MU, $f(W, W^s)$ can be optimized in each variable individually. Let us denote $g(W_{ia})$ the univariate polynomial corresponding to the terms of $f(W, W^s)$ where W_{ia} appears so that $f(W, W^s) = \sum_{i,a} g(W_{ia})$. Hence, we need to solve $\min_{W_{ia} \geq 0} g(W_{ia})$ for each i, a. Since $\lim_{W_{ia} \to \infty} g(W_{ia}) = \infty$, if the derivative of g has a zero at a single positive point $W^*_{ia} > 0$, then it will be an optimal solution of $\min_{W_{ia} \geq 0} g(W_{ia})$. Let us find the points W_{ia} such that $g'(W_{ia}) = 0$, that is,

$$\begin{aligned} \frac{\partial f(W, W^s)}{\partial W_{ia}} = & -\sum_j X_{ij} \frac{W^s_{ia} H_{aj}}{\sum_b W^s_{ib} H_{bj}} \frac{1}{W_{ia}} + \frac{W_{ia}}{W^s_{ia}} \left(\sum_j H_{aj} + \lambda \right) - \alpha \overline{W}_{ia} W^s_{ia} \frac{1}{W_{ia}} \\ & + \frac{1}{2}\alpha \left(\frac{8}{(W^s_{ia})^2} W^3_{ia} - 6 W_{ia} \right) + \gamma \left(\frac{4}{(W^s_{ia})^2} W^3_{ia} - 6 W_{ia} \right) \\ & + \frac{1}{2}\gamma \frac{4 \left(W^s (W^s)^\top W^s \right)_{ia}}{(W^s_{ia})^3} W^3_{ia} = 0, \end{aligned}$$

which, for $W_{ia} \neq 0$, can be simplified to

$$A_{ia} W_{ia}^4 + B_{ia} W_{ia}^2 + C_{ia} = 0 \tag{7}$$

where

- $A_{ia} = \frac{2}{(W_{ia}^s)^2} \left[2\alpha + \gamma \left(\frac{(W^s (W^s)^\top W^s)_{ia}}{W_{ia}^s} + 2 \right) \right]$,
- $B_{ia} = \frac{\sum_{j=1}^m H_{aj} + \lambda}{W_{ia}^s} - 3\alpha - 6\gamma$, and
- $C_{ia} = -\sum_{j=1}^m X_{ij} \frac{W_{ia}^s H_{aj}}{(W^s H)_{ij}} - \alpha \overline{W}_{ia} W_{ia}^s$.

The polynomial (7) has four roots, but only one non-negative root, given by

$$W_{ia}^* = W_{ia} \sqrt{\frac{-B_{ia} + \sqrt{\Delta_{ia}}}{2A_{ia}^*}},$$

where $\Delta_{ia} = B_{ia}^2 - 4A_{ia}C_{ia} \geq B_{ia}^2$ since $C_{ia} \leq 0$ and $A_{ia} \geq 0$.
Note that if $W_{ia} = 0$ then $W_{ia}^* = 0$ hence the above update does not modify this entry: this is the so-called zero-locking phenomenon of the multiplicative updates in the NMF literature. In practice, the entries of W should be initialized with positive entries hence will remain positive (although they could converge to zero). We should observe that, even if the objective function in (4) is different from typical NMF cost function, the zero-locking phenomenon is still applicable to the update rule (6). In fact, it is easy to show, that the other factor of the update rule, converges to a real number when $W_{ia} \to 0$:

$$\frac{-B_{ia} + \sqrt{\Delta_{ia}}}{A_{ia}^*} \to \frac{-\left(\sum_j H_{aj} + \lambda\right) + \sqrt{\left(\sum_j H_{aj} + \lambda\right)^2 + 8\gamma \left(\sum_j X_{ij} \frac{H_{aj}}{\sum_{l \neq a} W_{il} H_{lj}}\right) \left(\sum_{l \neq a} \sum_{b \neq i} W_{il} W_{bl} W_{ba}\right)}}{2\gamma \sum_{l \neq a} \sum_{b \neq i} W_{il} W_{bl} W_{ba}}.$$

This prove the zero-locking phenomenon for (6) since no indeterminate form can be achieved. □

The computational cost of the proposed algorithm is $\mathcal{O}(nmr)$ operations per iteration. The most costly operations are matrix products on factors of order $n \times m$, $n \times r$ and $r \times m$. It should be pointed out that the most NMF algorithms run in $\mathcal{O}(nmr)$ hence scale linearly in the dimensions of the input matrix and the factorization rank; in particular the multiplicative updates of (Lee and Seung 2000). Hence, although our multiplicative updates require some additional computations, the asymptotic computational cost is the same as standard NMF algorithms hence scales similarly in practice. For example, on the synthetic data sets presented in the next section, the original multiplicative updates of (Lee and Seung 2000) require about 6 s while OJSNMF updates require in average 10 s to perform 1000 iterations (we used Matlab 2016a and run them on a i-7 Core machine with a capacity of memory of 12 GB RAM).

4 Experimental results

In this section, we perform numerical experiments on several synthetic data sets and on a real biological data set from (Baranzini et al. 2004; Li and Ngom 2011).

4.1 Choice of the parameters

Choosing the parameters λ, α and γ related to the regularization terms in OJSNMF (2) is crucial for the proposed model. Fine tuning parameters is a difficult issue that is intrinsic in problems where the importance of several objectives need to be balanced. Unfortunately, there does not exist a single procedure to manage this aspect because choosing good parameters is application dependent. Hence, in practice, one should rely on the users' feedbacks to produce meaningful results.

Recall that the parameter λ regulates the degree of sparsity of the basis matrices W_t's, α penalizes their dissimilarity, and γ regulates their orthogonality. In this paper we use a parameter selection assuming the three regularization terms along with the data fitting term should be given the same importance in the objective function. This is done as follows: the initial matrices W_t^0 and H_t^0 for each slice ($t = 1, \ldots, T$) are used to compute the different terms in the objective function, namely $a_0 = \sum_{t=1}^{T} KL(X_t, W_t^0 H_t^0)$, $b_0 = \sum_{t=1}^{T} \left\| W_t^0 \right\|_1$, $c_0 = \frac{1}{2} \sum_{t=1}^{T} \left\| W_t^0 - \overline{W}_t \right\|_F^2$ and $d_0 = \frac{1}{2} \sum_{t=1}^{T} \left\| (W_t^0)^\top W_t^0 - \mathbb{I}_r \right\|_F^2$. Then, the parameters are chosen so that each term in the objective function is equal to a_0, that is, $\lambda = \frac{a_0}{b_0}$, $\alpha = \frac{a_0}{c_0}$, and $\gamma = \frac{a_0}{d_0}$. These values λ, α and γ are used during all the minimization procedure.

Note that a similar strategy can be used to balance differently the importance between the different terms in the objective function. In fact, the values of these parameters is user dependent as it relates to some information which could be in principle provided by the domain expert or connected to some a priori knowledge about the specific data set. These parameters could also be tuned during the optimization process in order to achieve some degree of sparsity (λ), orthogonality (γ) and coherence between basis matrices W_ts (α). For simplicity, we use in this paper the simple approach outlined above.

4.2 Comparison with other methods

The proposed algorithm is compared in terms of its global behavior and performances with other NMF algorithms based on the KL divergence:

- the baseline method proposed in (Lee and Seung 2000), namely MU, which solves problem (2) with $\lambda = \alpha = \gamma = 0$. This means that there is no coupling between the variables W_t for $t = 1, \ldots, T$, hence this is equivalent to solve T independent NMF problems.
- the orthogonal NMF algorithm presented in (Li et al. 2010) which we refer to as NMFOS-KL. This method solves $\sum_t KL(X_t, W_t H_t) + \gamma KL(I, W_t^\top W_t)$ whose second term is used to make the matrices W_t close to orthogonal.

– the sparse NMF algorithm proposed in (Liu et al. 2003; Esposito et al. 2017), which we refer to as sparse-NMF and solves (2) with $\alpha = \gamma = 0$. As for the original MU, this means that there is no coupling between the variables W_t.

To illustrate the flexibility of our approach and perform a meaningful comparison with the other algorithms, we compare MU with OJSNMF with parameters $\lambda = 0$ and $\alpha = \gamma = 10^{-5}$, NMFOS-KL with OJSNMF with parameters $\lambda = \alpha = 0$, and with sparse-NMF with OJSNMF with parameters $\alpha = \gamma = 10^{-5}$. For NMFOS-KL and sparse-NMF, we use the same penalty parameter as OJSNMF.

It should be noted that we have not included a comparison with tensor decompositions. The reason is that our model has additional constraints that are usually not taken into account altogether in standard tensor-based approaches. In fact, we believe that standard tensor methods are not suitable for our purpose. In particular, (nonnegative) CPD/PARAFAC would not be appropriate because the basis vectors corresponding to the time dimension would only allow to scale metagenes hence a gene present in a metagene would have to be active in all time steps, while imposing orthogonality on the basis vectors conrresponding to metagene is also not standard (we are not aware of a tensor method taking nonnegativity and orthogonality constraints into account). Also, higher-order/multilnear SVD would not be (easily) interpretable and is mostly used for compression purposes. Instead, we adopted a common comparison approach used in the literature, as for instance adopted in (Farias et al. 2016): we have considered our model as T NMF problems with flexible couplings, this allows to meaningfully comparing to the MU applied to the unfolded tensor $X \in \mathbb{R}^{n\times(m\cdot T)}$. In other words, we apply MU on the augmented matrix $[X_1 X_2 \dots X_T]$ so that it is approximated by $W[H_1 H_2 \dots H_T]$ which explicitly imposes that all W_ts are equal to W. We will refer to this technique as MU-unfolding and it is equivalent to the OJSNMF model with $\lambda = \gamma = 0$ and α very large.

4.3 Quality measures

In order to evaluate the solutions generated by the different algorithms, we will use the following measures averaged over each slice:

– Relative approximation error: the last value of the relative objective function of the KL-divergence scaled appropriately:

$$\text{Error} = \frac{\sum_{ij} X_{ij} \log\left(\frac{X_{ij}}{(WH)_{ij}}\right) - X_{ij} + (WH)_{ij}}{\sum_{ij} X_{ij} \log\left(X_{ij}\right)}.$$

– Two sparsity measures for the W_ts:

(i) the one proposed by Hoyer (Hoyer et al. 2004) which estimate the sparsity of a vector $x \in \mathbb{R}^n$ as

$$sparsness(x) = \frac{\sqrt{n} - \frac{\|x\|_1}{\|x\|_2}}{\sqrt{n} - 1},$$

(ii) the proportion of the elements smaller than 10^{-3} times the maximal value in the corresponding column.

– Relative distance between a slice W_t and its local average $\overline{W}_t$, that is, $\left\|\overline{W}_t - W_t\right\|_F / \left\|\overline{W}_t\right\|_F$.
– Relative orthogonality measure: $\left\|W_t^\top W_t - \mathbb{I}_r\right\|_F / \sqrt{r}$ since $\|\mathbb{I}_r\|_F = \sqrt{r}$.

4.4 Synthetic data sets

We first perform experiments on synthetic data sets to investigate the performance of the proposed algorithm. The data sets used for the analysis are constructed imposing following the OJSNMF model. Each slice is the product of $W_t^{true} \in \mathbb{R}^{90\times 7}$ and $H_t^{true} \in \mathbb{R}^{7\times 30}$ that are randomly generated as follows.

The matrix W_t^{true} is the same for each $t = 1, \ldots, T$. It is obtained as a normally distributed perturbation of a normalized row monomial matrix[1] W_{mono} mixed with an uniform random noise matrix. In summary, each slice W_t is generated randomly around the monomial matrix W_{mono} hence all slices are close to one another, not just consecutive slices.

This choice agrees with the theoretical situation described in Sect. 2.2. The entries of H_t^{true} are randomly generated from the standard uniform distribution in the interval [0,1]. The pseudo code 2 details the dataset generation in Matlab notation. We generate 100 such data sets.

Algorithm 2: Construction of the Synthetic Data Set

Data: Set the parameters of the normal and uniform distribution $\sigma_1 = 0.05$, $\mu = 1$, $\sigma_2 = 0.01$, compute the row monomial matrix $W_{mono} \in \mathbb{R}_+^{90\times 7}$ normalized by columns.
Result: $W_t^{true} \in \mathbb{R}_+^{90\times 7}$ and $H_t^{true} \in \mathbb{R}_+^{7\times 30}$ for $t = 1, \ldots, 5$
begin
  for $t = 1, 2, \ldots, 5$ **do**
    $A_{1t} = randn(90, 7)$;
    $A_{2t} = rand(90, 7)$;
    $W_t^{true} = W_{mono} .* (\mu + \sigma_1 * A_{1t}) + \sigma_2 * A_{2t}$;
    $H_t^{true} = randn(7, 30)$;

To initialize the different algorithms, we use the same initial matrices randomly generated using the uniform distribution to which we applied one iteration of the MU; this allows the initial matrices to be scaled well compared to the data X_t hence make the choice of the parameter described in Sect. 4.1 meaningful. Note that for the considered 100 synthetic data sets, the average values of the parameters are $\overline{\lambda} = 2.31$, $\overline{\alpha} = 203.87$ and $\overline{\gamma} = 110.13$.

Figure 4 shows the evolution of the average relative error for the different algorithms. Quite naturally, OJSNMF decreases the relative error slower than MU and converges to a higher value (except when only the sparsity parameter of OJSNMF

[1] A monomial matrix has exactly one non-zero element in each row

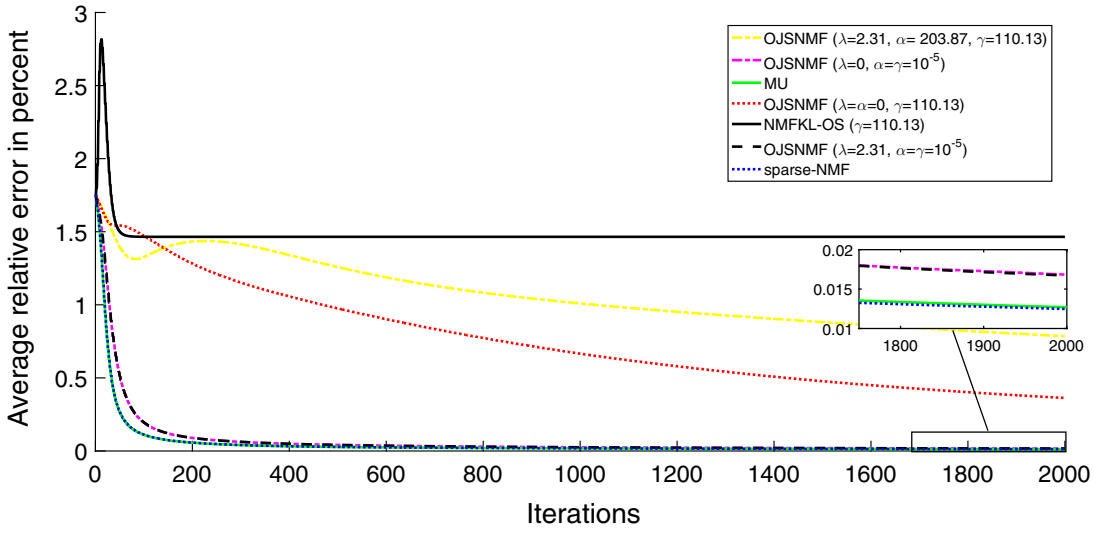


Fig. 4 Comparisons between the average relative objective function values over the 100 data sets

is postivie, namely, λ, in which case both methods perform similarly) since it needs to take into account other terms in the objective function (namely, orthogonality and smoothness over time, which are more involved constraints than sparsity). We observe that OJSNMF outperforms NMFKL-OS in terms of relative error.

Table 1 reports the average results of the quality measures detailed in Sect. 4.3 for the 100 synthetic data sets, along with the standard deviations. The following observations can be made:

- When used to match the parameters of MU and sparse-NMF, OJSNMF performs similarly as these methods. Note that MU and sparse-NMF provide very similar results: to obtain sparser solutions, the parameter λ should be chosen larger. OJSNMF only has a slightly larger error (+0.01%) because it converges slightly slower (see Fig. 4) as it uses a square root in the multiplicative updates. (The square root could be removed but we would loose the convergence result of Theorem 1 for the general case.) This illustrates the flexibility of OJSNMF.
- When compared to NMFOS-KL, OJSNMF has much smaller error (0.36% vs. 1.47%) but larger orthogonality (3.73% vs. 0%). This is explained in part by the fact that these two algorithms are not using the same model: NMFOS-KL uses $KL(I, W^\top W)$ and OJSNMF uses $||I_r - W^\top W||_F$.
- OJSNMF with all the parameters positive (the ninth row of Table 1) is able to identify solutions with small error (0.79% in average) while all the penalty terms have good values, with 81.62% sparsity, 2.05% coherence between the columns of the W_ts and 4.87% orthogonality. In particular, it has quality measures close to the other algorithms, while being able to produce very similar W_ts (average distance between slices of 2.05%), while the other approaches completely fail to do so ($>$ 100% relative error) except MU-unfolding. This illustrates the fact that NMF is highly non-unique (Gillis et al. 2012) and that using prior information from domain expert is a key aspect when using NMF.
- When compared to MU-unfolding, OJSNMF has the same error, but has less orthogonal slices (recall we chose $\gamma = 0$ to compare with MU-unfolding). This

Table 1 Average and standard deviation of the different quality measures in percent among 100 data sets

Methods	Error	Sparsity Hoyer	Sparsity less 10^{-3}	$\frac{\|\overline{W}_t - W_t\|_F}{\|\overline{W}_t\|_F}$	$\frac{\|W_t^\top W_t - \mathbb{I}_7\|_F}{\sqrt{r}}$
MU	0.01± 0.01	58.32±2.18	39.66±4.49	134.56±7.70	43.26±4.71
OJSNMF ($\lambda = 0$, $\alpha = \gamma = 10^{-5}$)	0.02±0.01	56.95±2.11	36.91±4.18	133.59±7.79	47.34±4.21
NMFOS-KL ($\overline{\gamma} = 110.13$)	1.47±0.08	71.18±0.35	$85.71 \pm 1.43 \cdot 10^{-13}$	143.47±2.19	$10^{-13}(1 \pm 9 \cdot 10^{-2})$
OJSNMF ($\lambda = \alpha = 0$, $\overline{\gamma} = 110.13$)	0.36±0.08	69.30±0.30	78.50±0.76	142.29±5.22	3.73±0.58
Sparse-NMF ($\overline{\lambda} = 2.31$)	0.01±0.01	58.37±2.18	39.74± 4.34	134.03±7.74	100.00±0.0003
OJSNMF ($\overline{\lambda} = 2.31$, $\alpha = \gamma = 10^{-5}$)	0.02±0.01	56.93±1.97	37.07±4.01	133.11±7.47	99.99±0.001
MU unfolding	0.01± 0.01	66.81±1.64	36.26±11.38	0	20.21±7.99
OJSNMF ($\overline{\lambda} = 0$, $\overline{\alpha} = 203.87$, $\overline{\gamma} = 0$)	0.01±0.01	63.03±3.86	35.55±8.99	5.53±6.48	88.91±6.37
OJSNMF ($\overline{\lambda} = 2.31$, $\overline{\alpha} = 203.87$, $\overline{\gamma} = 110.13$)	0.79±0.23	69.82±0.60	81.62±1.01	2.05±0.38	4.87±0.35

illustrates the fact that NMF is a highly ill-posed problem with non-unique solutions.

4.5 Real data set

In accordance to some previous studies about 3D microarrays (Li and Ngom 2011; Baranzini et al. 2004), we analyzed the data set provided in (Baranzini et al. 2004)[2]. The data set collects gene expression levels about patients affected by multiple-sclerosis (MS) disease and treated during different time steps with the protein Interferon beta (IFNβ). This kind of biological experiment was performed to clarify the medical responses, at molecular levels, to IFNβ treatments. The real data set was used to predict good or bad responders to these treatments (Li and Ngom 2011). Originally, the data set combined information about 73 genes, 53 patients and 7 time points (consisting of a first treatment, a quarterly treatment during the first year and two more treatments during last therapy year). However, since the data set contained some missing values in correspondence to the expression levels of some genes and patients, we pre-processed it to obtain a final complete data set, removing the associated rows and columns. The final

Table 2 Quality measures in percent for different values of the factorization rank

Rank and parameters	Error	Sparsity Hoyer	Sparsity less 10^{-3}	$\frac{\|\overline{W}_t - W_t\|_F}{\|\overline{W}_t\|_F}$	$\frac{\|W_t^\top W_t - \mathbb{I}_7\|_F}{\sqrt{r}}$
$r = 2$	0.91	86.44	45.06	0.78	1.27
$\lambda = 1.27$, $\alpha = 115.20$, $\gamma = 103.54$					
$r = 3$	0.69	93.83	64.29	0.61	1.47
$\lambda = 1.18$, $\alpha = 97.66$, $\gamma = 59.15$					
$r = 4$	0.59	94.08	73.35	0.51	1.79
$\lambda = 1.19$, $\alpha = 79.14$, $\gamma = 48.08$					
$r = 5$	0.67	95.26	78.24	0.66	1.51
$\lambda = 1.24$, $\alpha = 68.43$, $\gamma = 56.60$					
$r = 6$	0.65	96.01	81.96	0.64	1.42
$\lambda = 1.29$, $\alpha = 58.47$, $\gamma = 59.31$					

tensor $\chi \in \mathbb{R}_+^{52\times 27\times 7}$ collects gene expression levels of 52 genes, measured among 27 patients during a treatment of 7 time steps. For this experiment, instead of using randomly generated initial matrices, we initialize the algorithms using the non-negative double singular value decomposition (NNSVD) (Casalino et al. 2014; Boutsidis and Gallopoulos 2008) which provides better results than random initializations.

Let us observe the variations in the quality measures for various values of the factorization rank ($r = 2, \ldots, 6$); see Table 2.

We observe that the largest gap in the quality measures is from $r = 2$ to $r = 3$ (in particular the error and the sparsity), while they remain closer for $r \geq 3$. Interestingly, this value of r matches the value chosen in (Li and Ngom 2011). Hence, we will choose $r = 3$ to compare OJSNMF with the other NMF algorithms as in the previous section; see Table 3 for the numerical results.

We observe the following:

- As for the synthetic data set, MU, sparse-NMF and OJSNMF with the corresponding parameters behave very similarly. This is because the solution of MU is already rather sparse, and the penalty parameter λ is not large enough to obtain sparser solutions with sparse-NMF.
- NMFOS-KL and OJSNMF provide very similar results, as opposed to the synthetic data set. NMFOS-KL has only slightly larger orthogonality and OJSNMF slightly better coherence between the slices.

Table 3 Quality measures in percent for the real data set from (Li and Ngom 2011)

Method	Error	Sparsity Hoyer	Sparsity less 10^{-3}	$\frac{\|\overline{W}_t - W_t\|_F}{\|\overline{W}_t\|_F}$	$\frac{\|W_t^\top W_t - \mathbb{I}_7\|_F}{\sqrt{r}}$
MU	0.21	72.78	10.53	56.99	87.04
OJSNMF ($\lambda = 0$, $\alpha = \gamma = 10^{-5}$)	0.21	72.50	10.53	54.86	79.24
NMFOS-KL ($\gamma = 59.15$)	0.39	77.10	66.58	69.65	0.001
OJSNMF ($\lambda = \alpha = 0$, $\gamma = 59.15$)	0.39	73.20	57.05	61.09	0.92
Sparse-NMF ($\lambda = 1.18$)	0.21	72.70	10.62	59.76	100.00
OJSNMF ($\lambda = 1.18$, $\alpha = \gamma = 10^{-5}$)	0.21	72.52	10.35	58.09	100.00
MU unfolding	0.25	73.12	20.51	0	67.72
OJSNMF ($\lambda = 0$, $\alpha = 97.66$, $\gamma = 0$)	0.24	73.70	19.04	0.74	58.00
OJSNMF ($\lambda = 1.18$, $\alpha = 97.66$, $\gamma = 59.15$)	0.69	93.83	64.29	0.61	1.47

– When compared to the MU-unfolding, OJSNMF gives very similar results although in this case, it generated a solution with smaller error (0.24% vs. 0.25%) while having the slices almost identical (0.75%), and having also a better orthogonality (58% vs. 67.72%). This shows that introducing flexibility in the model is important as there are in practice evolution of the basis matrix W_ts over time.
– OJSNMF with all parameters activated presents good results respecting all the constraints in the model while the error is not significantly increased (note that the sparsity is much larger than for the other algorithms). Figure 5 displays the heatmap representation of the concatenated W_ts, which shows the sparsity, orthogonality and similar patterns of the different slices.

Interpretation of the OJSNMF metagenes

As explained in Sect. 2.1 a metagene codifies the weights or importance of each gene (row element) in the analyzed process (Brunet et al. 2004). To extract relevant genes within metagenes, different techniques have been developed when NMF is applied on 2D microarrays. The simplest approach sets *a priori* the number of genes within a metagene and extracts the corresponding number of elements with the largest magnitude (Crescenzi and Giuliani 2001). Another strategy is proposed in (Kim and Tidor 2003) and identifies features as meaningful when the corresponding coefficients in the basis matrix exceed a fixed threshold value. These mechanisms can be also adapted to analyze results obtained by OJSNMF, since it has the ability to construct

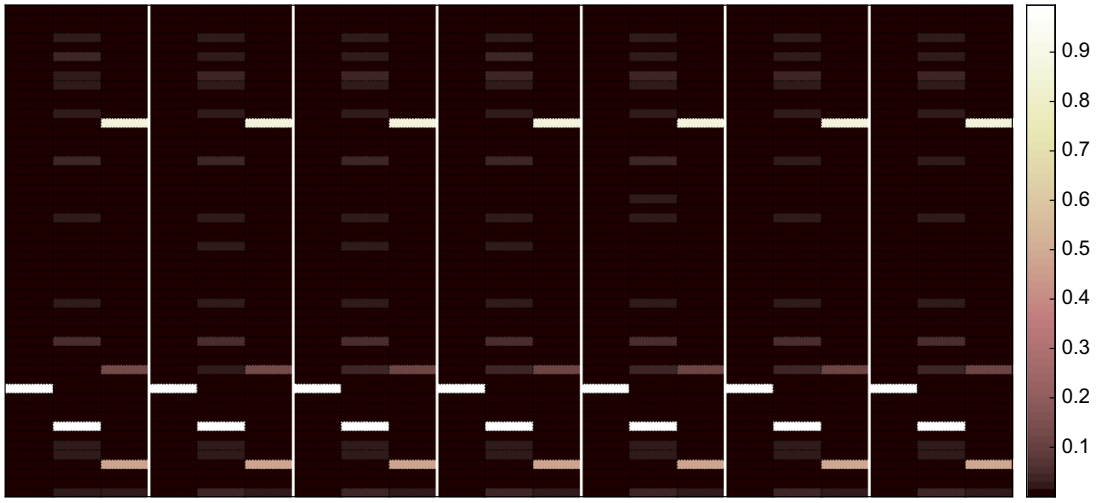


Fig. 5 Heatmap of the W_t slices for $t = 1, \ldots, 7$ concatenated. As reported in Table 3, the slices W_ts are very similar, with an average relative difference between slices of 0.61%. This explains why is it not possible to observe with the naked eye any significant difference between the different slices on this figure

similar slices W_t for $t = 1, \ldots, T$. Figure 5 displays the metagenes obtained by OJSNMF when applied to our numerical example and illustrates this similarity.

In this paper, since no preliminary knowledge on the dataset is available, we adopt a simple threshold strategy to decide whether a gene belongs or not to a metagene. Particularly, when the value in the column of the W_ts is at least 5% of the largest value in that column, the gene is extracted. The first metagene contains only one gene (MIP1a gene), the second metagene contains two genes (RANTES and CD86), and the third metagene contains three genes (Tbet, CD69 and IRF5). All these gene are widley present in the literature panorama of the multiple-sclerosis disease studies (Boven et al. 2000; Gade-Andavolu et al. 2004; Boivin et al. 2015; Racke et al. 2014; Vandenbroeck et al. 2011; Marckmann et al. 2004; Wiesemann et al. 2008; Huang et al. 2001).

5 Conclusion and future works

In this paper, we have proposed a new model, dubbed orthogonal joint sparse NMF (OJSNMF), to extract relevant information from 3D microarrays containing the time evolution of a 2D microarray. Our model is based on the KL divergence and is very flexible as it can incorporate three penalties: sparsity, orthogonality and coherence between the basis matrices of different time steps. We have developed multiplicative updates for OJSNMF and proved they monotonically decrease the objective function. We have shown that our approach competes favorably with state-of-the-art NMF algorithms based on the KL divergence on both synthetic and real data sets.

Further work The analysis of microarray data requires a constant dialogue with the domain experts to provide a biological interpretation of the mathematical models. For this reason, although the construction of our proposed model was supervised

from experts of the domain, OJSNMF still requires to be validated in real conditions. Another interesting aspect to be further investigated is the parameter choice. In fact, although in the experimental section we adopted a simple heuristic for the selection of the parameters, others options could be considered (e.g., tuning the parameters in order to achieve some desired value of the penalty terms).

Finally, it would be interesting to apply our model in other situations. For example, it would be useful to analyse hyperspectral images of a scene taken at different time points. In a few words, in that model, there will be one abundance matrix for each hyperspectral image. Each column of these abundance matrices record the abundances of a material in the pixels of the image. These matrices, correspond to the W_t's in our model, will be sparse, close to being orthogonal (most pixels in such images contain mosty one material) and smooth over time since the materials present in a scene usually evolve smoothly over time; see, e.g., (Veganzones et al. 2016) for more details on this application and an example on the evolution of the snow coverage in the French Alps during the snow season.

Acknowledgements This work has been supported in part by the GNCS (*Gruppo Nazionale per il Calcolo Scientifico*) of Istituto Nazionale di Alta Matematica Francesco Severi, P.le Aldo Moro, Roma, Italy. NG acknowledges the support of the European Research Council (ERC starting Grant No. 679515). We thank Angelina Boccarelli (Department of Biomedical Science and Human Oncology, Medical School, University of Bari, Italy) for her useful biological support during the construction of the model and her precious suggestions for the biological interpretation. We also thank Jérémy Cohen (IRISA, Rennes) for his suggestion to compare OJSNMF with NMF applied on the unfolded tensor.

References

Alter O, Brown PO, Botstein D (2000) Singular value decomposition for genome-wide expression data processing and modeling. Proc Natl Acad Sci 97(18):10101–10106

Baranzini SE, Mousavi P, Rio J, Caillier SJ, Stillman A, Villoslada P, Wyatt MM, Comabella M, Greller LD, Somogyi R et al (2004) Transcription-based prediction of response to IFNβ using supervised computational methods. Plos Biol 3(1):e2

Boccarelli A, Esposito F, Coluccia M, Frassanito MA, Vacca A, Del Buono N (2018) Improving knowledge on the activation of bone marrow fibroblasts in mgus and mm disease through the automatic extraction of genes via a nonnegative matrix factorization approach on gene expression profiles. J Transl Med 16(1):217

Boivin N, Baillargeon J, Doss PMIA, Roy AP, Rangachari M (2015) Interferon-β suppresses murine th1 cell function in the absence of antigen-presenting cells. PLOS ONE 10(4):1–17

Borgwardt KM, Vishwanathan S, Kriegel HP (2006) Class prediction from time series gene expression profiles using dynamical systems kernels. Biocomputing. World Scientific, Singapore, pp 547–558

Boutsidis C, Gallopoulos E (2008) SVD based initialization: a head start for nonnegative matrix factorization. Pattern Recognit 41(4):1350–1362

Boven L, Montagne L, Nottet H, De Groot C (2000) Macrophage inflammatory protein-1α (MIP-1α), MIP-1β, and RANTES mRNA semiquantification and protein expression in active demyelinating multiple sclerosis (MS) lesions. Clin Exp Immunol 122(2):257–263

Brunet JP, Tamayo P, Golub TR, Mesirov JP (2004) Metagenes and molecular pattern discovery using matrix factorization. Proc Natl Acad Sci 101(12):4164–4169

Carmona-Saez P, Pascual-Marqui RD, Tirado F, Carazo JM, Pascual-Montano A (2006) Biclustering of gene expression data by non-smooth non-negative matrix factorization. BMC Bioinform 7(1):1

Casalino G, Del Buono N, Mencar C (2014) Subtractive clustering for seeding non-negative matrix factorizations. Inf Sci 257:369–387

Cheung VC, Devarajan K, Severini G, Turolla A, and Bonato P (2015) Decomposing time series data by a non-negative matrix factorization algorithm with temporally constrained coefficients. In 2015 37th annual international conference of the IEEE on engineering in medicine and biology society (EMBC), pp 3496–3499

Cichocki A, Zdunek R, Phan AH, Amari SI (2009) Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation. Wiley, New York

Crescenzi M, Giuliani A (2001) The main biological determinants of tumor line taxonomy elucidated by a principal component analysis of microarray data. FEBS Lett 507(1):114–118

Dai JJ, Lieu L, Rocke D (2006) Dimension reduction for classification with gene expression microarray data. Stat Appl Genet Mol Biol 5(1):1–21

Del Buono N, Esposito F, Fumarola F, Boccarelli A, Coluccia M (2016) Breast cancer's microarray data: pattern discovery using nonnegative matrix factorizations. Machine learning, optimization, and big data. Springer, Berlin, pp 281–292

Dhillon IS and Sra S (2005) Generalized nonnegative matrix approximations with Bregman divergences. In NIPS, vol 18

Ding C, He X, and Simon H (2005) On the equivalence of nonnegative matrix factorization and spectral clustering. In Proceedings of the 2005 SIAM international conference on data mining, pp 606–610. SIAM

Du Mg, Zhang SW, and Wang H (2009) Tumor classification using high-order gene expression profiles based on multilinear ICA. Adv Bioinform. https://doi.org/10.1155/2009/926450

Esposito F, Del Buono N (2017) Exploring hidden information in sparse NMF. Technical Report 8, University of Bari, Department of Mathematics

Farias RC, Cohen JE, Comon P (2016) Exploring multimodal data fusion through joint decompositions with flexible couplings. IEEE Trans Signal Process 64(18):4830–4844

Gade-Andavolu R, Comings DE, MacMurray J, Vuthoori RK, Tourtellotte WW, Nagra RM, Cone LA (2004) RANTES: a genetic risk marker for multiple sclerosis. Mult Scler J 10(5):536–539

Gillis N (2012) Sparse and Unique nonnegative matrix factorization through data preprocessing. J Mach Learn Res 13:3349–3386

Gillis N, Glineur F (2012) Accelerated multiplicative updates and hierarchical ALS algorithms for nonnegative matrix factorization. Neural Comput 24(4):1085–1105

Glaab E, Garibaldi JM, Krasnogor N (2011) Integrative analysis of large-scale biological data sets. Nat Precedings. https://doi.org/10.1038/npre.2011.5598.1

He Z, Xie S, Zdunek R, Zhou G, Cichocki A (2011) Symmetric nonnegative matrix factorization: algorithms and applications to probabilistic clustering. IEEE Trans Neural Netw 22(12):2117–2131

Hoyer PO (2004) Non-negative Matrix factorization with sparseness constraints. J Mach Learn Res 457–1469

Huang YM, Hussien Y, Jin YP, Söderstrom M, Link H (2001) Multiple sclerosis: deficient in vitro responses of blood mononuclear cells to IFN-β. Acta Neurol Scand 104(5):249–256

Hutchins LN, Murphy SM, Singh P, Graber JH (2008) Position-dependent motif characterization using non-negative matrix factorization. Bioinformatics 24:2684–2690

Kim H, Park H (2007a) Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. Bioinformatics 23(12):1495–1502

Kim H, Park H (2007b) Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. Bioinformatics 23(12):1495–1502

Kim PM, Tidor B (2003) Subsystem identification through dimensionality reduction of large-scale gene expression data. Genome Res 13(7):1706–1718

Kong W, Mou X, Hu X (2011) Exploring matrix factorization techniques for significant genes identification of Alzheimer's disease microarray gene expression data. BioMed Cent BMC Bioinform 12:S7

Kong W, Vanderburg CR, Gunshin H, Rogers JT, Huang X (2008) A review of independent component analysis application to microarray gene expression data. BioTechniques 45(5):501–520

Kouskoumvekaki I, Shublaq N, Brunak S (2013) Facilitating the use of large-scale biological data and tools in the era of translational bioinformatics. Brief Bioinform 15(6):942–952

Lee DD and Seung HS (2000) Algorithms for non-negative matrix factorization. In Proceedings of the advances in neural information processing systems conference, vol 3, pp 556–562. MIT Press

Li Y and Ngom A (2010) Non-negative matrix and tensor factorization based classification of clinical microarray gene expression data. In 2010 IEEE international conference on bioinformatics and biomedicine (BIBM), pp 438–443. IEEE

Li Y and Ngom A (2011) Classification of clinical gene-sample-time microarray expression data via tensor decomposition methods. In: Rizzo R, Lisboa PJG (eds) Computational intelligence methods for bioinformatics and biostatistics. Springer, Berlin, pp 275–286

Li Z, Wu X, Peng H (2010) Nonnegative matrix factorization on orthogonal subspace. Pattern Recognit Lett 31(9):905–911

Liao JC, Boscolo R, Yang YL, Tran LM, Sabatti C, Roychowdhury VP (2003) Network component analysis: reconstruction of regulatory signals in biological systems. Proc Natl Acad Sci 100(26):15522–15527

Liu W, Yuan K, Ye D (2008) Reducing microarray data via nonnegative matrix factorization for visualization and clustering analysis. J Biomed Inform 41(4):602–606

Liu W, Zheng N, and Lu X (2003) Non-negative matrix factorization for visual coding. In Proceedings of 2003 IEEE international conference on acoustics, speech, and signal processing, 2003 (ICASSP'03), vol 3, pp 3–293. IEEE

Mairal J, Bach F, and Ponce J (2014) Sparse Modeling for Image and Vision Processing. arXiv preprint arXiv:1411.3230

Marckmann S, Wiesemann E, Hilse R, Trebst C, Stangel M, Windhagen A (2004) Interferon-β up-regulates the expression of co-stimulatory molecules CD80, CD86 and CD40 on monocytes: significance for treatment of multiple sclerosis. Clin Exp Immunol 138(3):499–506

Moschetta M, Basile A, Ferrucci A, Frassanito MA, Rao L, Ria R, Solimando AG, Giuliani N, Angelina B, Fumarola F, Coluccia M, Rossini B, Ruggieri S, Nico B, Maiorano E, Ribatti D, Roccaro AM, Vacca A (2013) Novel targeting of phospho-cMET overcomes drug resistance and induces antitumor activity in multiplle myeloma. Clin Cancer Res 19(16):4371–82

Nikulin V and Huang TH (2012) Unsupervised dimensionality reduction via gradient-based matrix factorization with two adaptive learning rates. In Proceedings of ICML workshop on unsupervised and transfer learning, pp. 181–194

Omberg L, Golub GH, Alter O (2007) A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies. Proc Natl Acad Sci 104(47):18371–18376

Pompili F, Gillis N, Absil PA, Glineur F (2014) Two algorithms for orthogonal nonnegative matrix factorization with application to clustering. Neurocomputing 141:15–25

Racke MK, Yang Y, Lovett-Racke AE (2014) Is T-bet a potential therapeutic target in multiple sclerosis? J Interferon Cytokine Res 34(8):623–632

Takahashi N, Hibi R (2014) Global convergence of modified multiplicative updates for nonnegative matrix factorization. Comput Optim Appl 57(2):417–440

Vandenbroeck K, Alloza I, Swaminathan B, Antigüedad A, Otaegui D, Olascoaga J, Barcina MG, De Las Heras V, Bartolomé M, Fernández-Arquero M et al (2011) Validation of IRF5 as multiple sclerosis risk gene: putative role in interferon beta therapy and human herpes virus-6 infection. Genes Immun 12(1):40

Veganzones MA, Cohen JE, Farias RC, Chanussot J, Comon P (2016) Nonnegative tensor cp decomposition of hyperspectral data. IEEE Trans Geosci Remote Sens 54(5):2577–2588

Wall ME, Rechtsteiner A, and Rocha LM (2003) Singular value decomposition and principal component analysis. In: Berrar DP, Dubitzky W, Granzow M (eds) A practical approach to microarray data analysis. Springer, Berlin, pp 91–109

Wiesemann E, Deb M, Trebst C, Hemmer B, Stangel M, Windhagen A (2008) Effects of interferon-β on co-signaling molecules: upregulation of CD40, CD86 and PD-l2 on monocytes in relation to clinical response to interferon-β treatment in patients with multiple sclerosis. Multiple Scler J 14(2):166–176

Yang Z, Michailidis G (2015) A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data. Bioinformatics 32(1):1–8

Zhang A (2006) Advanced analysis of gene expression microarray data, vol 1. World Scientific, Singapore