



Accelerated SVD-based initialization for nonnegative matrix factorization

Flavia Esposito¹ · Syed Muhammad Atif² · Nicolas Gillis³

Received: 29 February 2024 / Revised: 23 July 2024 / Accepted: 12 August 2024
© The Author(s) 2024

Abstract

Nonnegative matrix factorization (NMF) is a popular dimensionality reduction technique. NMF is typically cast as a non-convex optimization problem solved via standard iterative schemes, such as coordinate descent methods. Hence the choice of the initialization for the variables is crucial as it will influence the factorization quality and the convergence speed. Different strategies have been proposed in the literature, the most popular ones rely on singular value decomposition (SVD). In particular, Atif et al. (Pattern Recognit Lett 122:53–59, 2019) have introduced a very efficient SVD-based initialization, namely NNSVD-LRC, that overcomes the drawbacks of previous methods, namely, it guarantees that (i) the error decreases as the factorization rank increases, (ii) the initial factors are sparse, and (iii) the computational cost is low. In this paper, we improve upon NNSVD-LRC by using the low-rank structure of the residual matrix; this allows us to obtain NMF initializations with similar quality to NNSVD-LRC (in terms of error and sparsity) while reducing the computational load. We evaluate our proposed solution over other NMF initializations on several real dense and sparse datasets.

Keywords Nonnegative matrix factorization · Singular value decomposition · Nesterov accelerated gradient · Initialization

Mathematics Subject Classification 15A23 · 68Q17 · 65F55 · 65D17

✉ Flavia Esposito
flavia.esposito@uniba.it

Syed Muhammad Atif
syed.atif@zu.edu.pk

Nicolas Gillis
nicolas.gillis@umons.ac.be

¹ Department of Mathematics, Università degli Studi di Bari Aldo Moro, Bari, Italy

² College of Computing and Software Engineering, Ziauddin University, Karachi, Pakistan

³ Department of Mathematics and Operational Research, Faculté Polytechnique, Université de Mons, Mons, Belgium

1 Introduction

Nonnegative matrix factorization (NMF) decomposes an input nonnegative data matrix, $X \in \mathbb{R}_+^{m \times n}$, into the product of two nonnegative matrices $W \in \mathbb{R}_+^{m \times k}$ and $H \in \mathbb{R}_+^{k \times n}$ such that $X \approx WH$, where k is the rank of the factorization. The nonnegativity constraints of NMF allow one to reconstruct data using a purely additive model, with the advantage of a natural interpretability. This makes NMF useful in many applications (Cichocki et al. 2009; Gillis 2020) in various research fields like image analysis (Ensari 2016; Prajapati and Jadhav 2015; Luce et al. 2016; Shiga et al. 2016), text mining (Du et al. 2019), signal processing (Yoshii et al. 2016), and computational biology (Maruyama et al. 2014).

NMFs are in most cases computed by solving the following optimization problem:

$$\min_{W \geq 0, H \geq 0} \|X - WH\|_F^2, \quad (1)$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Most algorithms for solving (1) use standard non-linear optimization schemes such as block coordinate descent methods, updating the factors in an iterative way; see, e.g., Cichocki et al. (2009), Gillis (2020) and the references therein. Hence, they require initialization mechanisms for the variables, W and H , that can greatly influence the factorization results. They can affect both (i) the number of iterations needed for an algorithm to converge (in fact, if the initial point is closer to a local minimum, it will require a reduced number of iterations to converge to it), and (ii) the final solution the algorithm will converge to. Different strategies have been proposed for NMF initialization (Esposito 2021), such as simple schemes (e.g., random initializations) (Langville et al. 2006), structured schemes (e.g., SVD- or clustering-based (Wild et al. 2004; Rezaei et al. 2011)), evolutionary-based approaches (e.g., genetics algorithms, particle swarm optimization) (Janecek and Tan 2011), and geometric-based methods. The latter class of methods tries to locate the vertices of the convex polytope formed by the columns of X ; see, e.g., Zdunek (2012); Araújo et al. (2001); Nascimento and Dias (2005); Liu and Tan (2018) and [Gillis (2020) Chapter 7]. To the best of our knowledge, some of the most popular initialization methods for NMF are based on deterministic low-rank decompositions such as rank-1 decomposition (Liu and Tan 2018) and the SVD (Esposito 2021), and geometric methods such as vertex component analysis (Nascimento and Dias 2005) and the successive projection algorithm (Araújo et al. 2001; Sauwen et al. 2017). In this paper, we focus on these latter. In this class, two of the most widely used methods, introduced in Boutsidis and Gallopoulos (2008) and Qiao (2015), have the drawback to increasing the approximation error, $\|X - WH\|_F^2$, as the rank increases; see Sect. 2 for more details. More recently, a variant of these approaches, referred to as Nonnegative SVD with Low-Rank Correction (NNSVD-LRC), makes use of a low-rank correction to address this issue (Atif et al. 2019), while reducing the computational load by using a truncated SVD of smaller rank (roughly half). In this paper, we improve upon NNSVD-LRC by proposing a new initialization scheme, referred to as accelerated Nonnegative SVD with Progressive Residual Projection (accNNSVD-PRP), that reduces further the computational load of NNSVD-LRC by keeping track of the residual using its low-rank property. The paper is organized as follows. Section 2 briefly reviews the background of SVD-based initialization for NMF, with an emphasis on NNSVD-LRC. Section 3 details our proposed solution, and highlights the differences with existing SVD-based initializations. In Sect. 4, we show that the new proposed initialization, accNNSVD-PRP, compares favorably against standard structured and random initializations on several real dense and sparse data sets. Section 5 concludes the paper and discusses some future directions of research.

2 SVD-based NMF initializations: previous works

As mentioned in the introduction, SVD-based initializations are the most popular and effective for NMF. Recall that the rank- p truncated SVD approximates a data matrix $X \in \mathbb{R}_+^{m \times n}$ with a lower-rank matrix composed by p rank-one terms as follows:

$$X \approx X_p = U_p \Sigma_p V_p^T, \tag{2}$$

where X_p is the best rank- p approximation of X in the Frobenius norm, $U_p \in \mathbb{R}^{m \times p}$ and $V_p \in \mathbb{R}^{n \times p}$ contain left and right singular vectors, and $\Sigma_p \in \mathbb{R}^{p \times p}$ contains the singular values on its diagonal in nonincreasing order. Let us write the truncated SVD as the product of only two matrices, as follows

$$X \approx X_p = \sum_{i=1}^p y_i z_i = YZ, \tag{3}$$

where $Y = U_p \Sigma_p^{1/2} \in \mathbb{R}^{m \times p}$ and $Z = \Sigma_p^{1/2} V_p^T \in \mathbb{R}^{p \times n}$, so that $y_i = \sqrt{\sigma_i} U_p(:, i)$ and $z_i = \sqrt{\sigma_i} V_p(:, i)^T$ for $1 \leq i \leq p$. (Note that z_i 's are row vectors, which will simplify the notation by avoiding using the transpose sign.) Due to their negative entries,¹ Y and Z cannot be used directly for NMF initialization.

Let us define, for a generic vector $b \in \mathbb{R}^q$, its nonnegative part, $b^{(\geq 0)} = \max(0, b)$, and its nonpositive part, $b^{(\leq 0)} = \max(0, -b)$, so that $b = b^{(\geq 0)} - b^{(\leq 0)}$. In this way, (3) can be rewritten as:

$$X_p = \sum_{i=1}^p \left(y_i^{(\geq 0)} z_i^{(\geq 0)} + y_i^{(\leq 0)} z_i^{(\leq 0)} \right) - \sum_{i=1}^p \left(y_i^{(\geq 0)} z_i^{(\leq 0)} + y_i^{(\leq 0)} z_i^{(\geq 0)} \right). \tag{4}$$

The negative sign of the second term in (4) poses challenges for NMF initialization, since it can lead to negative initialization values, which are not suitable for NMF purposes. In the literature, there are different methods designed to effectively manage and mitigate the impact of negative values.

The two most popular and widely used approaches are the following:

- (1) *Nonnegative Double SVD* (NDSVD) (Boutsidis and Gallopoulos 2008) discards the second term (with the minus sign) in (4), and selects p product terms (among the $2p$) from the first term according to the following criterion: for each i , it chooses $y_i^{(\geq 0)} z_i^{(\geq 0)}$ if $\|y_i^{(\geq 0)} z_i^{(\geq 0)}\|_F > \|y_i^{(\leq 0)} z_i^{(\leq 0)}\|_F$, otherwise it opts for $y_i^{(\leq 0)} z_i^{(\leq 0)}$. This takes advantage of the sign ambiguity of the SVD (Boutsidis and Gallopoulos 2008; Bro et al. 2008).
- (2) *SVD-NMF* (Qiao 2015) adopts as an initialization the component-wise absolute values of Y and Z , that is, $W = |Y|$ and $H = |Z|$.

Unfortunately, these two methods have several drawbacks. First, the reconstruction error of the initialization, $\|X - WH\|_F^2$, increases as the rank of the factorization increases. This is because all negative terms are discarded. Second, a lot of information embedded into the first term of (4) is lost. To alleviate these drawbacks, another method, called Nonnegative SVD with Low-Rank Correction (NNSVD-LRC), was proposed more recently in Atif et al. (2019) that splitting (4) in different parts, it considers just the nonnegative side of the rank p -approximation.

¹ Roughly half of them, except for the first rank-one factor, by the Perron-Frobenius theorem (Berman and Plemmons 1994).

2.1 Nonnegative SVD with low-rank correction

NNSVD-LRC is composed by two phases: a nonnegative SVD (NNSVD) initialization, followed a Low-Rank Correction (LRC) that allows NNSVD-LRC to decrease the error as the factorization rank increases. The pseudo-code of NNSVD-LRC is detailed in Algorithm 1.

Algorithm 1 NNSVD-LRC

Require: A nonnegative matrix X , factorization rank k , the stopping parameter $\delta \in (0, 1)$ for the low-rank correction.

Ensure: Nonnegative factors $W \in \mathbb{R}^{m \times k}$ and $H \in \mathbb{R}^{k \times n}$ such that $X \approx WH$.

Phase 1 - Nonnegative Singular Value Decomposition

```

1:  $p = \lfloor \frac{k}{2} + 1 \rfloor$ ;
2:  $[U_p, \Sigma_p, V_p] = \text{truncated-SVD}(X, p)$ ;
3:  $Y = U_p \Sigma_p^{1/2}$ ;  $Z = \Sigma_p^{1/2} V_p^T$ ;
4:  $W(:, 1) = |Y(:, 1)|$ ;  $H(1, :) = |Z(1, :)|$ ;
5:  $i = 2$ ;  $j = 2$ ;
6: while  $j \leq k$  do
7:   if  $j$  is even then
8:      $W(:, j) = \max(Y(:, i), 0)$ ;
9:      $H(j, :) = \max(Z(i, :), 0)$ ;
10:  else
11:     $W(:, j) = \max(-Y(:, i), 0)$ ;
12:     $H(j, :) = \max(-Z(i, :), 0)$ ;
13:     $i = i + 1$ ;
14:  end if
15:   $j = j + 1$ ;
16: end while

```

Phase 2 - Low rank correction

```

17:  $X_p = YZ$ ;
18:  $e_0 = \|X_p - WH\|_F$ ;  $t = 0$ ;
19: while  $t = 0$  or  $e_t - e_{t-1} \geq \delta e_0$  do
20:   Perform one iteration of A-HALS Gillis and Glineur (2012) on  $X_p$  starting from  $(W, H)$  to obtain an improved solution  $(W, H)$ .
21:    $e_{t+1} = \|X_p - WH\|_F$ ;
22:    $t = t + 1$ ;
23: end while

```

The first phase, NNSVD, computes the nonnegative factors W and H from the rank- p truncated SVD of X , using $p = \lfloor k/2 + 1 \rfloor$ as in (2), and then construct YZ as in (3). By the Perron-Frobenius theorem, $|y_1||z_1|$ is an optimal rank-one approximation of X in the Frobenius norm, hence we set $W(:, 1) = |y_1|$ and $H(1, :) = |z_1|$.

The remaining $k - 1$ columns of W and rows of H are completed with the next $\lfloor p/2 \rfloor$ factors of the truncated SVD as follows:

$$W(:, j) = \begin{cases} y_i^{(\geq 0)} & \text{if } j \text{ is even,} \\ y_i^{(\leq 0)} & \text{otherwise,} \end{cases} \quad H(j, :) = \begin{cases} z_i^{(\geq 0)} & \text{if } j \text{ is even,} \\ z_i^{(\leq 0)} & \text{otherwise,} \end{cases} \quad (5)$$

where $j = 2, 3, \dots, k$ and $i = \lfloor \frac{j}{2} + 1 \rfloor$.

The second phase will fed the preliminary factors W and H into LRC to reduce the error induced during the NNSVD phase due to loss of information in the second term of (4).

This is done by using a state-of-the-art NMF algorithm, namely the Accelerated Hierarchical Alternating Least Squares (A-HALS)² algorithm (Gillis and Glineur 2012), to solve

$$\min_{W \geq 0, H \geq 0} \|X_p - WH\|_F^2. \tag{6}$$

The reason to use X_p instead of X is that $X_p = YZ$ has a low rank representation and hence each iteration of the subsequent NMF algorithm can be reduced from $O(mnk)$ operations per iteration to $O((m+n)k^2)$ operations. Although NNSVD-LRC performs a few relatively cheap A-HALS iterations, it is found empirically that the time taken by these iterations is not negligible compared to the truncated SVD computation.

This motivates us to introduce a new SVD-based initialization for NMF, namely accNNSVD-PRP described in the next section.

3 Accelerated nonnegative SVD with progressive residual projection

In this section, we detail the new proposed SVD-based initialization for NMF, namely Accelerated NNSVD with Progressive Residual Projection (accNNSVD-PRP); see Algorithm 2.

Algorithm 2 Accelerated Nonnegative SVD with Progressive Residual Projection (accNNSVD-PRP)

Require: A nonnegative matrix X factorization rank k , extrapolation parameter α_{init} , tolerance for the stopping criterion tol , and maximum number of iterates $t^{(max)}$.

Ensure: Nonnegative factors $W \in \mathbb{R}_+^{m \times k}$ and $H \in \mathbb{R}_+^{k \times n}$ such that $X \approx WH$.

1: $[W, H, \bar{H}] = \text{mNNSVD}(X, k)$; %See Algorithm 3.

2: $[H] = \text{PRP}(W, H, \bar{H}, \alpha_{init}, tol, t^{(max)})$; %See Algorithm 4.

It differs from NNSVD-LRC in the following ways:

- (1) It modifies the first phase, NNSVD, to keep track of the second term of (4) which is discarded by NNSVD-LRC; see Algorithm 3.
- (2) It replaces the second phase, LRC, by a cheaper one, which we refer to as Progressive Residual Projection (PRP). In contrast to NNSVD-LRC, it only updates the factor H to decrease the error.

Let us describe these two phases in more details.

Phase 1: modified NNSVD (mNNSVD) This phase is very similar to NNSVD, the first phase of NNSVD-LRC, see Algorithm 1. The generated factors (W, H) are the same since it splits in different parts (5), but mNNSVD keeps track of the discarded term in (4), denoted \bar{H} , such that

$$X_p = WH - W\bar{H}, \quad \text{where } (W, H, \bar{H}) \geq 0,$$

that is,

$$\bar{H}(1, :) = \mathbf{0}, \quad \text{and} \quad \bar{H}(j, :) = \begin{cases} z_i^{(<0)} & \text{if } j \text{ is even,} \\ z_i^{(>0)} & \text{otherwise,} \end{cases}$$

² A-HALS is an accelerated variant of the HALS algorithm (Cichocki et al. 2007) that uses a block coordinate descent method to solve the nonnegative least squares subproblems in order to update the factor matrices W and H .

where $j = 2, 3, \dots, k$, $i = \lfloor \frac{j}{2} + 1 \rfloor$, and $\mathbf{0}$ denotes the vector of zeros of appropriate dimension. Algorithm 3 described this procedure.

Algorithm 3 Modified NNSVD (mNNSVD)

Require: A nonnegative matrix X and factorization rank k .

Ensure: Nonnegative factors $W \in \mathbb{R}_+^{m \times k}$, $H \in \mathbb{R}_+^{k \times n}$ and $\overline{H} \in \mathbb{R}_+^{k \times n}$ such that $X \approx X_p = W(H - \overline{H})$.

```

1:  $p = \lfloor \frac{k}{2} + 1 \rfloor$ ;
2:  $[U_p, \Sigma_p, V_p] = \text{truncated-SVD}(X, p)$ ;
3:  $Y_p = U_p \Sigma^{1/2}$ ;  $Z_p = \Sigma^{1/2} V^T$ ;
4:  $W(:, 1) = |Y_p(:, 1)|$ ;  $H(1, :) = |Z_p(1, :)|$ ;  $\overline{H}(1, :) = \mathbf{0}$ ;
5:  $i = 2$ ;  $j = 2$ ;
6: while  $j \leq k$  do
7:   if  $j$  is even then
8:      $W(:, j) = \max(Y_p(:, i), 0)$ ;
9:      $H(j, :) = \max(Z_p(i, :), 0)$ ;
10:     $\overline{H}(j, :) = \max(-Z_p(i, :), 0)$ ;
11:   else
12:      $W(:, j) = \max(-Y_p(:, i), 0)$ ;
13:      $H(j, :) = \max(-Z_p(i, :), 0)$ ;
14:      $\overline{H}(j, :) = \max(Z_p(i, :), 0)$ ;
15:      $i = i + 1$ ;
16:   end if
17:    $j = j + 1$ ;
18: end while

```

Phase 2: Progressive Residual Projection (PRP)

From the output of Algorithm 3, we directly use the basis matrix W as initialization. We will only update H to reduce the approximation error.

To this aim, similarly as in LRC, we want to solve

$$\min_{H \geq 0} F(H) := \|X_p - WH\|_F^2. \tag{7}$$

Let us denote $H^{(0)}$ and $\overline{H}^{(0)}$ the output of mNNSVD. We have $X_p = WH^{(0)} - W\overline{H}^{(0)}$. To solve (7), let us resort to Nesterov accelerated gradient descent (Nesterov 1983, 2018), and let us denote the iterates $H^{(t)}$ for $t = 0, 1, \dots$. At every iteration, we will have

$$X_p = WH^{(0)} - W\overline{H}^{(0)} \approx WH^{(t)}.$$

To progressively keep track of the residual, let us define

$$\overline{H}^{(t)} = \overline{H}^{(0)} + H^{(t)} - H^{(0)}.$$

This implies that, for all t , the following equality holds

$$X_p = W(H^{(t)} - \overline{H}^{(t)}). \tag{8}$$

This can be used to compute the gradient efficiently, as follows:

$$\nabla F(H^{(t)}) = -2W^T(X_p - WH^{(t)}) = 2W^TW\overline{H}^{(t)}. \tag{9}$$

It is worthy to note that the formulation of the gradient in (9) implicitly depends on $H^{(t)}$. Since Nesterov accelerated gradient method does not ensure the objective function to decrease

at every iteration, we embed the code with a restarting scheme. If the objective function increases, the algorithm abandons the extrapolated sequence and takes a standard gradient step (O’donoghue B, Candes E, 2015). The cost for each iteration of this phase consists of $O(k^2(m+n))$ operations. In contrast, although LRC has the same computational cost per iteration, it alternates between optimizing W and H , necessitating a few iterations each time to update both factors, which results in a comparatively slower process. On the other hand, PRP exclusively optimizes H and typically requires less iterations (as it updates only H), it operates significantly faster than LRC.

Algorithm 4 provides the pseudo-code of PRP, where $S^{(t)}$ denotes the extrapolated sequence of Nesterov gradient method (Nesterov 1983). Recall that Nesterov accelerated gradient method performs a projected gradient step (step 3 in Algorithm 4) from an extrapolated sequence, denoted $S^{(t)}$, computed via a so-called extrapolation, see step 5 in Algorithm 4, where the so-called extrapolation parameters (step 4 in Algorithm 4) are chosen to guarantee an optimal convergence rate.

Algorithm 4 Progressive Residual Projection

Require: Nonnegative factors $W \in \mathbb{R}_+^{m \times k}$, $H^{(0)} \in \mathbb{R}_+^{k \times n}$, and $\overline{H}^{(0)} \in \mathbb{R}_+^{k \times n}$, extrapolation parameter α_{init} , tolerance for the stopping criterion tol , and maximum number of iterates $t^{(max)}$.

Ensure: Nonnegative factor $H \in \mathbb{R}_+^{k \times n}$.

- 1: Define: $S^{(0)} = H^{(0)}$; $\overline{S}^{(0)} = \overline{H}^{(0)}$; $\alpha^{(0)} = \alpha_{init}$; $\Delta S = H^{(0)} - \overline{H}^{(0)}$; $e^{(0)} = \|W\overline{S}^{(0)}\|_F$; $e^{(1)} = e^{(0)}$;
 $L = \|W^T W\|_2$; $t = 1$;
 - 2: **while** $t \leq t^{(max)}$ and $(t == 1 \text{ or } \alpha^{(t)} == 1 \text{ or } (e^{(t-1)} - e^{(t)}) \geq tol \cdot \|X_p\|_F)$ **do**
 - 3: $H^{(t)} = \max\left(0, S^{(t-1)} - \left(\frac{2}{L}(W^T W)\overline{S}^{(t-1)}\right)\right)$
 - 4: $\alpha^{(t)} = \frac{1 + \sqrt{4(\alpha^{(t-1)})^2 + 1}}{2}$;
 - 5: $S^{(t)} = H^{(t)} + \frac{\alpha^{(t-1)} - 1}{\alpha^{(t)}}(H^{(t)} - H^{(t-1)})$;
 - 6: $\overline{S}^{(t)} = S^{(t)} - \Delta S$;
 - 7: $e^{(t)} = \|W\overline{S}^{(t)}\|_F$
 - 8: **if** $(e^{(t-1)} - e^{(t)}) < 0$ **then**
 - 9: $\alpha^{(t)} = 1$; $S^{(t)} = H^{(t-1)}$; $\overline{S}^{(t)} = S^{(t)} - \Delta S$;
 - 10: **end if**
 - 11: $t = t + 1$;
 - 12: **end while**
 - 13: $H = H^{(t-1)}$
-

Sparsity of factors in accNNSVD-PRP Since W is initialized via the SVD in accNNSVD-PRP, its columns will be roughly 50% sparse (except for the first one, by the Perron-Frobenius theorem). On the other hand, since H is optimized, it is hard to predict the sparsity it will achieve, and it will depend on the data set. On the dense data sets presented in Sect. 4.1, the average sparsity of H was about 50% (between 45% and 56% for all data sets and tested ranks). For the sparse data sets (see Sect. 4.1), the sparsity was between 64% and 74%.

4 Numerical experiments

In this section, we show some numerical results to corroborate the goodness of our proposed method as initialization for NMF problems. The code of accNNSVD-PRP is available on

GitHub at <https://github.com/5y3datif/accNNSVD-PRP>. We compare the proposed method with the scaled random initialization (SRI) from Gillis and Glineur (2008, 2012), and with five structure-based NMF initializations: three are SVD-based, namely NNDSVD, SVD-NMF and NNSVD- LRC (Atif et al. 2019), and two clustering-based, namely a recent hybrid method combining clustering and the computation of rank-one SVDs called CR1- NMF (Liu and Tan 2017), and SPKM based on spherical k -means (Wild et al. 2004). The code for CR1- NMF and SPKM are available from <https://github.com/zhaoqiangliu/cr1-nmf>. Note that, because of the non-deterministic nature of CR1-NMF, SPMK and SRI, we run the methods 100 times and report mean results along with corresponding standard deviations. All tests are preformed using Matlab R2021b on a laptop Intel(R) Core(TM) i5-9300 H CPU @ 2.40GHz empowered by NVIDIA GEFORCE GTX 1650, CUDA toolkit 10.0 and cuDNN 7.4. We take the advantage of available GPU (whenever possible) during simulation. We will compare the various methods using the following two quantities:

1. the relative error(X, WH) = $\frac{\|X-WH\|_F}{\|X\|_F}$,
2. the computational time which is computed as the CPU time in Matlab.

4.1 Datasets description

We conduct experiments on the following dense and sparse datasets.

Dense datasets

The following three widely used and real dense datasets:

- AT&T Faces is a dataset of face images taken in 1992–1994 and it is one of the widely used face images datasets by the research community. There are ten different images of 40 distinct subjects. The size of each image is 92×112 pixels, with 256 grey levels per pixel. The dataset is freely available from <http://cam-orl.co.uk/facedatabase.html>.
- Faces95 is a facial images dataset that consists of 72 subjects such that there is a sequence of 20 images with a resolution of 180×200 pixels. The dataset is freely accessible from <https://www.essex.ac.uk/mv/allfaces/faces95.zip>.
- PaviaU is a Hyperspectral image dataset acquired by the ROSIS sensor during a flight campaign over Pavia, Italy. Each image has resolution of 610×610 with a spatial resolution of 1.3m, and 103 spectral bands. The dataset is freely available from <http://www.ehu.es/ccwintco/uploads/e/ee/PaviaU.mat>, http://www.ehu.es/ccwintco/uploads/5/50/PaviaU_gt.mat.

Sparse datasets

We also tested our approach on sparse document datasets. The datasets were derived from the San Jose Mercury newspaper articles that are distributed as part of the TREC collection (TIPSTER Vol. 3) and are accessible from <https://catalog.ldc.upenn.edu/LDC93T3D> (Zhong and Ghosh 2005):

- Sports: it consists of documents about seven different sports (such as baseball, basketball, football, hockey, boxing, bicycle, and golf). It contains 8580 documents with 14870 words. Its sparsity is 99.14%.
- Reviews: it consists documents about five topics (such as food, movies, music, radio, and restaurants). It contains of 4069 documents and 18483 words. Its sparsity is 98.99%.

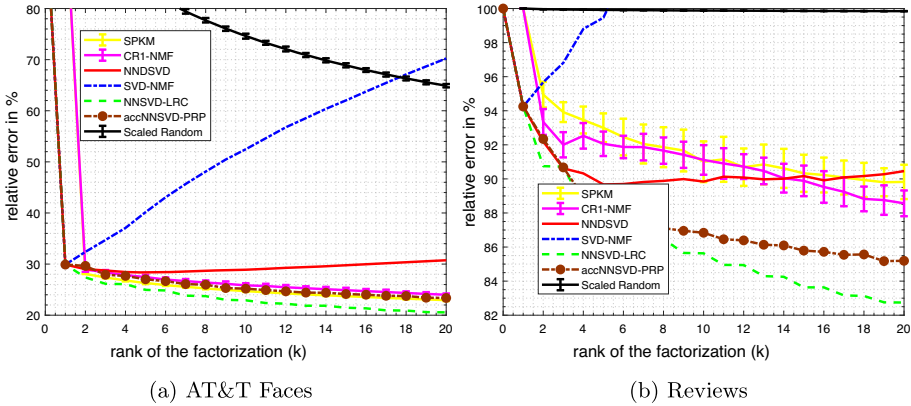


Fig. 1 Relative errors of the SVD-based NMF initializations for different values of the rank k , for the data set AT&T Faces and Reviews. Similar results are observed for the other datasets

- **Hitech:** it consists documents about six technologies (such as computers, electronics health, medical, research, and technology). It contains 2301 documents with 10080 words. Its sparsity is 99.14%.

4.2 Results

First, as for NNSVD- LRC and as opposed to NNSVD and SVD-NMF, our proposed method, accNNSVD-PRP, decreases the relative error as the rank increase; see Fig. 1 for an example on the AT&T Faces and Reviews data sets. We report representative examples for both classes of data sets studied: dense and sparse; moreover for the Reviews data set it can be noted that the relative error still decreases for all the rank values tested (from $k = 1, \dots, 20$) that could represent an over-estimation of the $k = 5$ topics in the data sets, as explained in Zhong and Ghosh (2005).

We also observe on Fig. 1 that accNNSVD-PRP generates solutions with initial error larger than that of NNSVD- LRC, but this is expected since accNNSVD-PRP only optimizes the factor H in the second phase. Table 1 reports the relative errors after a few iterations (namely, 5, 25, 125) of the HALS algorithm for NMF (Cichocki et al. 2007). We observe that acc-NNSVDPRP provides similar relative error values compared to NNSVD-LRC, this is even more evident especially after enough iterations of HALS. Compared to the other initialization strategies, accNNSVD-PRP and NNSVD-LRC perform on average better, as already reported in Atif et al. (2019).

However, although accNNSVD-PRP does not outperform NNSVD-LRC in terms of relative errors, it runs significantly faster since its second phase is much cheaper. Table 2 reports the computational times of the tested NMF initialization for different over-estimated rank values i.e., $k = 15, 20, 25, 35, 50$.

We observe that, except for random initializations which are very fast to generate, accNNSVD-PRP runs among the fastest in all cases (always at least second best, and very close to the fastest when not the best). In particular, accNNSVD-PRP runs significantly faster than NNSVD-LRC: on average 4.5 times faster on dense data sets, and 1.7 times faster on sparse data sets.

Table 1 Relative error in percent of different initialization methods after few iterations (namely 5, 25, 125) of HALS for different over-estimated rank values i.e., $k = 15, 20, 25$. Cases when accNNSVD-PRP performs better than NNSVD-LRC are indicated in bold

		(a) Dense dataset								
		AT&T Faces		Faces95		PaviaU				
		$k = 15$	$k = 20$	$k = 25$	$k = 15$	$k = 20$	$k = 25$			
5	SPKM	22.91	22.23	21.56	32.40	31.30	30.25	3.25	2.46	2.19
	CRI-NMF	23.55	22.56	22.08	32.58	31.42	30.43	2.99	2.59	2.33
	NNDSVD	20.81	20.09	19.54	29.22	27.97	27.10	3.82	3.52	3.28
	SVD-NMF	21.04	20.15	19.44	29.57	28.22	27.27	3.30	2.98	2.71
	NNSVD-LRC	20.51	19.56	18.73	28.47	27.11	26.00	2.68	2.25	1.98
	accNNSVD-PRP	20.97	19.98	19.27	29.16	27.95	26.86	2.92	2.57	2.37
	Scaled Random	23.69	22.39	21.28	31.88	30.47	28.82	5.19	4.95	4.44
	SPKM	20.81	20.18	19.34	28.55	27.50	26.16	2.89	2.19	1.93
	CRI-NMF	20.87	20.10	19.42	28.32	26.72	25.88	2.65	2.32	2.15
	NNDSVD	19.40	18.44	17.71	27.40	25.99	24.86	2.00	1.73	1.45
25	SVD-NMF	19.57	18.60	17.84	27.64	26.21	25.14	2.12	1.83	1.59
	NNSVD-LRC	19.54	18.59	17.73	27.42	25.97	24.83	2.09	1.75	1.52
	accNNSVD-PRP	19.70	18.65	17.81	27.57	26.22	24.98	1.91	1.64	1.45
	Scaled Random	19.99	18.89	18.10	27.75	26.29	25.03	3.00	2.63	2.32
	SPKM	19.20	18.27	17.51	27.11	25.70	24.40	2.55	2.04	1.76
	CRI-NMF	19.20	18.22	17.48	27.16	25.58	24.43	2.07	2.03	1.91
	NNDSVD	19.11	18.06	17.25	27.06	25.51	24.30	1.70	1.42	1.20
	SVD-NMF	19.14	18.14	17.30	27.05	25.53	24.40	1.71	1.42	1.25
	NNSVD-LRC	19.12	18.10	17.28	27.10	25.55	24.28	1.76	1.46	1.26
	accNNSVD-PRP	19.15	18.07	17.28	27.16	25.53	24.35	1.64	1.39	1.21
Scaled Random	19.26	18.16	17.34	27.18	25.58	24.37	2.07	1.80	1.61	

Table 1 continued

Sparse dataset	Hitech			Reviews			Sports			
	$k = 15$	$k = 20$	$k = 25$	$k = 15$	$k = 20$	$k = 25$	$k = 15$	$k = 20$	$k = 25$	
	5	SPKM	88.18	87.09	86.61	82.82	81.83	80.52	83.76	82.78
	CRI-NMF	88.55	87.07	85.71	82.87	81.65	80.82	83.39	81.89	80.72
	NNDSVD	87.92	86.57	85.43	82.32	81.12	80.26	83.27	81.68	80.28
	SVD-NMF	88.21	86.95	85.56	82.83	81.28	80.38	82.99	81.46	80.02
	NNSVD-LRC	88.15	86.57	85.31	82.14	81.21	80.04	83.04	81.22	80.21
	accNNSVD-PRP	88.32	86.89	85.38	82.54	81.25	80.19	83.09	81.44	80.11
	Scaled Random	88.43	87.28	85.99	83.72	82.53	80.72	83.17	81.66	80.86
25	SPKM	87.80	86.29	85.00	82.20	80.76	79.72	82.66	80.91	79.51
	CRI-NMF	88.03	86.20	84.89	81.98	80.92	79.62	82.53	81.15	79.67
	NNDSVD	87.71	86.28	84.86	82.17	80.82	79.89	82.80	81.07	79.59
	SVD-NMF	87.75	86.17	84.81	82.04	80.75	79.57	82.64	81.11	79.48
	NNSVD-LRC	87.76	86.29	84.87	81.96	80.96	79.68	82.55	80.90	79.64
	accNNSVD-PRP	87.78	86.36	84.81	82.03	80.86	79.66	82.85	80.89	79.60
	Scaled Random	87.80	86.35	84.97	82.54	81.08	79.73	82.96	80.96	79.93
125	SPKM	87.69	86.14	84.89	81.89	80.65	79.55	82.62	80.86	79.43
	CRI-NMF	87.70	86.12	84.77	81.89	80.73	79.56	82.52	80.90	79.51
	NNDSVD	87.66	86.19	84.72	81.90	80.56	79.50	82.62	80.88	79.43
	SVD-NMF	87.67	86.15	84.75	81.89	80.57	79.52	82.63	80.89	79.43
	NNSVD-LRC	87.70	86.20	84.76	81.95	80.76	79.65	82.46	80.86	79.52
	accNNSVD-PRP	87.66	86.20	84.74	81.95	80.78	79.65	82.65	80.84	79.46
	Scaled Random	87.78	86.35	84.84	82.52	80.90	79.85	82.89	80.95	79.65

Table 2 CPU time (in s.) taken by different NMF initializations for the different data sets according to several over-estimated rank values i.e., $k = 15, 20, 25, 35, 50$

	SPKM	CRI-NMF	NNDSVD	SVD-NMF	NNSVD-LRC	accNNSVD-PRP	Scaled Random
AT&T Faces	$k = 15$	0.27 ± 0.08	0.09 ± 0.02	0.16	0.14	0.91	0.02 ± 0.01
	$k = 20$	0.28 ± 0.09	0.11 ± 0.02	0.21	0.18	1.19	0.02 ± 0.02
	$k = 25$	0.31 ± 0.08	0.14 ± 0.02	0.28	0.23	1.43	0.02 ± 0.02
	$k = 35$	0.31 ± 0.07	0.20 ± 0.03	0.39	0.31	1.88	0.02 ± 0.02
	$k = 50$	0.32 ± 0.08	0.27 ± 0.03	0.52	0.50	2.61	0.02 ± 0.01
PaviaU	$k = 15$	1.45 ± 0.24	0.24 ± 0.06	0.46	0.42	0.93	0.09 ± 0.02
	$k = 20$	1.55 ± 0.24	0.33 ± 0.07	0.64	0.58	1.68	0.11 ± 0.03
	$k = 25$	1.67 ± 0.31	0.37 ± 0.07	0.84	0.77	2.07	0.13 ± 0.03
	$k = 35$	1.69 ± 0.36	0.48 ± 0.09	2.36	1.94	3.33	0.17 ± 0.04
	$k = 50$	1.76 ± 0.42	0.62 ± 0.09	2.31	2.02	4.81	0.24 ± 0.04
Faces95	$k = 15$	2.75 ± 0.12	0.51 ± 0.05	0.67	0.65	1.21	0.07 ± 0.02
	$k = 20$	2.78 ± 0.11	0.53 ± 0.05	0.90	0.87	1.64	0.07 ± 0.02
	$k = 25$	2.97 ± 0.10	0.55 ± 0.05	1.15	1.11	2.00	0.08 ± 0.03
	$k = 35$	3.43 ± 0.13	0.61 ± 0.05	1.61	1.55	2.61	0.09 ± 0.02
	$k = 50$	3.83 ± 0.14	0.72 ± 0.05	2.36	2.25	3.58	0.10 ± 0.03
Sports	$k = 15$	1.59 ± 0.07	2.11 ± 1.64	0.19	0.13	0.30	0.01 ± 0.02
	$k = 20$	1.68 ± 0.08	2.04 ± 1.73	0.22	0.18	0.29	0.02 ± 0.02
	$k = 25$	1.09 ± 0.05	1.73 ± 1.26	0.26	0.23	0.35	0.02 ± 0.02
	$k = 35$	2.05 ± 0.09	1.72 ± 1.31	0.39	0.30	0.69	0.03 ± 0.03
	$k = 50$	2.33 ± 0.11	1.51 ± 0.13	0.50	0.44	0.84	0.05 ± 0.03
$k = 15$	1.18 ± 0.06	1.71 ± 0.35	0.17	0.12	0.25	0.01 ± 0.02	

Table 2 continued

	SPKM	CRI-NMF	NNDSVD	SVD-NMF	NNSVD-LRC	accNNSVD-PRP	Scaled Random
$k = 20$	1.23±0.07	1.61±0.38	0.22	0.16	0.31	0.24	0.02 ± 0.02
$k = 25$	0.31±0.03	1.62±0.45	0.27	0.21	0.45	0.27	0.02 ± 0.02
$k = 35$	1.39±0.08	1.50±0.39	0.50	0.41	0.66	0.55	0.03 ± 0.02
Reviews	1.65±0.09	1.47±0.36	0.52	0.41	1.05	0.64	0.04 ± 0.02
$k = 15$	0.31±0.03	0.61±0.11	0.23	0.18	0.30	0.15	0.01 ± 0.01
$k = 20$	0.36±0.04	0.54±0.12	0.26	0.20	0.35	0.17	0.01 ± 0.02
$k = 25$	0.37±0.04	0.52±0.11	0.32	0.25	0.41	0.20	0.01 ± 0.02
$k = 35$	0.44±0.04	0.52±0.13	0.38	0.33	0.59	0.30	0.01 ± 0.02
Hitech	0.51±0.03	0.45±0.14	0.50	0.38	0.73	0.41	0.02 ± 0.02

5 Conclusion

In this paper, we have proposed a new SVD-based initialization for NMF, namely accNNSVD-PRP. It is inspired by NNSVD-LRC, a state-of-the-art algorithm, but we adapted the two steps of NNSVD-LRC, by reformulating the NNSVD to introduce a factor \bar{H} that takes into account some discarded terms that will be updated later with a low-rank correction. This procedure allows the proposed algorithm to run significantly faster, while keeping its nice properties, in particular that the approximation error decreases as the factorization rank increases. As a remark we would like to note that this approach can be adapted to switch between H and W factors and to include W in the LRC correction phase as well, to further improve performance of accNNSVD-PRP and this is the object of our future works. To conclude, we have shown over various sparse and dense datasets that accNNSVD-PRP runs significantly faster than NNSVD-LRC while performing similarly as an NMF initialization scheme in terms of relative errors. Moreover, it outperforms other state-of-the-art algorithms such as NNSVD and SVD-NMF.

Acknowledgements The author F.E. is a member of the Gruppo Nazionale Calcolo Scientifico - Istituto Nazionale di Alta Matematica (GNCS-INdAM) and she would like to thank Prof. Nicoletta Del Buono from the University of Bari Aldo Moro for scientific discussion during this project.

Author Contributions All authors contribute equally to the work.

Funding Open access funding provided by Università degli Studi di Bari Aldo Moro within the CRUI-CARE Agreement. The author F.E. is supported by ERC Seeds Uniba project “Biomes Data Integration with Low-Rank Models” (CUP H93C23000720001), Piano Nazionale di Ripresa e Resilienza (PNRR), Missione 4 “Istruzione e Ricerca”-Componente C2 Investimento 1.1, “Fondo per il Programma Nazionale di Ricerca e Progetti di Rilevante Interesse Nazionale”, Progetto PRIN-2022 PNRR, P2022BLN38, Computational approaches for the integration of multi-omics data. CUP: H53D23008870001.

Availability of data and materials Data are publicly available at the links in the text.

Declaration

Conflict of interest Not applicable.

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication All authors consent to the publication of the work.

Code availability The code of accNNSVD-PRP is available on GitHub at <https://github.com/5y3datif/accNNSVD-PRP>.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Araújo MCU, Saldanha TCB, Galvao RKH, Yoneyama T, Chame HC, Visani V (2001) The successive projections algorithm for variable selection in spectroscopic multicomponent analysis. *Chemometr Intell Lab Syst* 57(2):65–73
- Atif SM, Qazi S, Gillis N (2019) Improved svd-based initialization for nonnegative matrix factorization using low-rank correction. *Pattern Recognit Lett* 122:53–59
- Berman A, Plemmons RJ (1994) Nonnegative matrices in the mathematical sciences. SIAM
- Boutsidis C, Gallopoulos E (2008) Svd based initialization: a head start for nonnegative matrix factorization. *Pattern Recognit* 41(4):1350–1362
- Bro R, Acar E, Kolda TG (2008) Resolving the sign ambiguity in the singular value decomposition. *J Chemometr* 22(2):135–140
- 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
- Cichocki A, Zdunek R, Amari SI (2007) Hierarchical ALS algorithms for nonnegative matrix and 3d tensor factorization. In: international conference on independent component analysis and signal separation. Springer, Berlin, pp 169–176
- Du R, Drake B, Park H (2019) Hybrid clustering based on content and connection structure using joint nonnegative matrix factorization. *J Glob Optim* 74(4):861–877
- Ensari T (2016) Character recognition analysis with nonnegative matrix factorization. *Int J Comput I*
- Esposito F (2021) A review on initialization methods for nonnegative matrix factorization: towards omics data experiments. *Mathematics* 9(9):1006
- Geng X, Ji L, Sun K (2016) Non-negative matrix factorization based unmixing for principal component transformed hyperspectral data. *Front Inf Technol Electron Eng* 403–412
- Gillis N (2020) Nonnegative matrix factorization. SIAM, Philadelphia
- Gillis N, Glineur F (2012) Accelerated multiplicative updates and hierarchical als algorithms for nonnegative matrix factorization. *Neural Comput* 24(4):1085–1105
- Gillis N, Glineur F (2008) Nonnegative factorization and the maximum edge biclique problem. [arXiv:0810.4225](https://arxiv.org/abs/0810.4225)
- Janecek A, Tan Y (2011) Using population based algorithms for initializing nonnegative matrix factorization. In: International conference in swarm intelligence. Springer, Berlin, pp 307–316
- Langville AN, Meyer CD, Albright R, Cox J, Duling D (2006) Initializations for the nonnegative matrix factorization. In: Proceedings of the twelfth ACM SIGKDD international conference on knowledge discovery and data mining, Citeseer, pp 23–26
- Liu Z, Tan VY (2017) Rank-one nmf-based initialization for nmf and relative error bounds under a geometric assumption. *IEEE Trans Signal Process* 65(18):4717–4731
- Liu Z, Tan V (2018) Rank-one NMF-based initialization for NMF and relative error bounds under a geometric assumption. *Information Theory And Applications Workshop (ITA)*, pp. 1–15
- Luce R, Hildebrandt P, Kuhlmann U, Liesen J (2016) Using separable nonnegative matrix factorization techniques for the analysis of time-resolved raman spectra. *Appl Spectrosc* 70(9):1464–1475
- Maruyama R, Maeda K, Moroda H, Kato I, Inoue M, Miyakawa H, Aonishi T (2014) Detecting cells using non-negative matrix factorization on calcium imaging data. *Neural Netw* 55:11–19
- Nascimento J, Dias J (2005) Vertex component analysis: a fast algorithm to unmix hyperspectral data. *IEEE Trans Geosci Remote Sens* 898–910
- Nesterov Y (1983) A method of solving a convex programming problem with convergence rate $O(1/k^2)$. In: *Sov. Math. Dokl*, vol 27, pp 372–376
- Nesterov Y et al (2018) Lectures on convex optimization, vol 137. Springer, Berlin
- O’donoghue B, Candes E (2015) Adaptive restart for accelerated gradient schemes. *Found Comput Math* 15(3):715–732
- Prajapati SJ, Jadhav KR (2015) Brain tumor detection by various image segmentation techniques with introduction to non negative matrix factorization. *Brain* 4(3):600–3
- Qiao H (2015) New svd based initialization strategy for non-negative matrix factorization. *Pattern Recognit Lett* 63:71–77
- Rezaei M, Boostani R, Rezaei M (2011) An efficient initialization method for nonnegative matrix factorization. *J Appl Sci* 11(2):354–359
- Sauwen N, Acou M, Bharath HN, Sima DM, Veraart J, Maes F, Himmelreich U, Achten E, Van Huffel S (2017) The successive projection algorithm as an initialization method for brain tumor segmentation using non-negative matrix factorization. *Plos One* 12(8):e0180268
- Shiga M, Tatsumi K, Muto S, Tsuda K, Yamamoto Y, Mori T, Tanji T (2016) Sparse modeling of eels and edx spectral imaging data by nonnegative matrix factorization. *Ultramicroscopy* 170:43–59

- Wang X, Xie X, Lu L (2012) An effective initialization for orthogonal nonnegative matrix factorization. *J Comput Math* 34–46
- Wild S, Curry J, Dougherty A (2004) Improving non-negative matrix factorizations through structured initialization. *Pattern Recognit* 37(11):2217–2232
- Yoshii K, Itoyama K, Goto M (2016) Student's t nonnegative matrix factorization and positive semidefinite tensor factorization for single-channel audio source separation. In: 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, pp 51–55
- Zdunek R (2012) Initialization of nonnegative matrix factorization with vertices of convex polytope. In: International conference on artificial intelligence and soft computing, pp 448–455
- Zhong S, Ghosh J (2005) Generative model-based document clustering: a comparative study. *Knowl Inf Syst* 8(3):374–384

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.