Initialization for non-negative matrix factorization: a comprehensive review

Sajad Fathi Hafshejani · Zahra Moaberfard

Received: 9 April 2022 / Accepted: 23 October 2022 / Published online: 11 November 2022
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2022

Abstract
Non-negative matrix factorization (NMF) has become a popular method for representing meaningful data by extracting a non-negative basis feature from an observed non-negative data matrix. Some of the unique features of this method in identifying hidden data place this method among the powerful methods in the machine learning area. The NMF is a known non-convex optimization problem, and the initial point has a significant effect on finding an efficient local solution. In this paper, we investigate the most popular initialization procedures proposed for NMF so far. We describe each method and present some of their advantages and disadvantages. Finally, some numerical results to illustrate the performance of each algorithm are presented.

Keywords Non-negative matrix factorization · Initialization algorithms

Abbreviations
NMF Non-negative matrix factorization
LR Low-rank
SVD Singular value decomposition
PCA Principal component analysis
KL Kullback-Leibler
MU Multiplicative update
NPCA Non-negative PCA
ICA Independent component analysis
NICA Non-negative ICA
NNDSVD Non-negative double SVD
NNSVD-LRC Non-negative SVD LR correction
FCM Fuzzy C-means
DE Differential evolution
PSO Particle swarm optimization

1 Introduction
Over the last few years, the low-rank approximation, which is approximating a matrix by one whose rank is less than that of the original matrix, has been an important technique and highly popular method in data science. Low-rank approximations are fundamental and widely used tools for data analysis, dimensionality reduction, and data compression. This method appears in many applications such as image processing [1], text data-latent semantic indexing, text mining [2,3], machine learning [4,5], and metamorphic malware detection [6–8]. Low-rank approximations find two matrices of the much lower rank that approximate a high-dimensional matrix \( X \) such that:

\[
X_{m \times n} \approx W_{m \times r} H_{r \times n}.
\]

where

\[ r << \min(m, n) \]

in which \( r \) is so-called rank of the matrix. There commonly matrix decompositions give a low-rank approximation, for example, we can refer to singular value decompositions (SVD) [9–11]. It can provide the optimal rank and give the appropriate of low-rank approximation of a matrix [12–14]. Unfortunately, these approximations usually do not actualize eligible structural constraints such as element-wise non-negativity [15–17]. For this reason, other concepts based on convex optimization have been developed such as singular value decomposition (SVD) [18], semidiscrete decomposi-
NMF is an unsupervised data decomposition technique, akin to latent variable analysis, that can be used for feature learning, topics recovery, clustering, temporal segmentation, filtering, and source separation coding as with vector quantization. As a matter of fact, this method obtains parts-based, compression, and discriminant representation of the original data as well as enhancing the interpretability by using decomposes the main matrix into additive parts [28]. There have been some significant developments [29] in using NMF for computation of the linear part-based representation of non-negative data. Therefore, NMF can be considered as a method in the machine learning area which enhances the interpretability of the results. In fact, interpretability can be considered as one of the advantages of NMF. In addition, robustness is another property of NMF that can be applied to handle noise and estimates the missing values. Suppose that $X$ is a real and non-negative $m \times n$ matrix. NMF finds two real and non-negative matrices $W \in \mathbb{R}^{m \times r}$ and $H \in \mathbb{R}^{r \times n}$, such that:

$$X \approx WH.$$  

Various approaches to find matrices $W$ and $H$ have been proposed so far. An efficient way for this purpose is to apply optimization tools. To use optimization algorithms, a criterion for measuring the difference between the original matrix, that is, $X$, and the results, i.e., the matrix $WH$, is needed. This measure is called objective function and can be considered as a measure for denoting an error. Two common measures are:

- Square of Euclidian Distance (SED): In this case, the Frobenius norm between the origin matrix $X$ and its approximation $WH$ is used as the similarity measure to derive the following objective function:

$$\min_{W,H \geq 0} f(W,H) = \frac{1}{2} \|X - WH\|_F^2, \text{ such that } W \geq 0 \text{ and } H \geq 0.$$  

(3)

- Generalized Kullback-Leibler Divergence (GKLD): It is the most popular in real applications. The corresponding objective function that characterizes the similarity between matrix $X$ and matrix $WH$ (called divergence) is given by [28]:

$$\min_{W,H \geq 0} KL(X; WH) = \sum_{i,j} (X_{ij} \log \frac{X_{ij}}{(WH)_{ij}} - X_{ij}) + (WH)_{ij}.$$  

(4)

Note that the objective functions given by (3) and (4) are non-convex with respect to variables $W$ and $H$. So, iterative approaches suggested for solving these problems guarantee to converge to some local minimum (more precisely, stationary points), but require initialization mechanisms that can greatly affect their convergence rate. In addition, choosing a proper initial point can have a significant impact on the efficiency of the algorithm. “Good” initial values for NMF are defined as follows [30]:

- One that leads to rapid error reduction and faster convergence.
- One that leads to better overall error at convergence.

Despite that, several different schemes for initialization of the NMF have appeared in the literature. But each of these studies has only focused on one or two approaches for initialization of NMF and has stated the advantages of their strategy over some other methods. As pointed out in [31], the problem of initialization, which is an open issue for NMF algorithm, is still unsolved. However, it is crucial for the performance of NMF algorithm in data analysis. Lee and Seung [28] proved that the NMF algorithm can converge to a local minimum, which is affected greatly by the initial value of the matrices $W$ and $H$. As a result, given different initialization of NMF algorithm, the NMF bases may be totally different, and the experiments cannot therefore be repeated by others. In this regard, it seems that a systematic survey is of necessity and consequence. This review paper will summarize the most existing initialization strategies for NMF. We first present some common methods for solving NMF problem and then focus on the initialization approaches for this problem. We collect the most common initialization algorithms and investigate the advantages and disadvantages of them. Finally, we perform Lee’s algorithm to compare the efficiency of the initialization methods. We perform the algorithm on the ORL dataset consisting of face images and compare the results.

The rest of this paper is organized as follows. In Sect. 2, we briefly review the NMF problem and present some common approaches for solving it and classifies the existing initialization approaches. In Sect. 3, we have a comprehensive review of random initialization seeding methods and present their algorithms. We preset various clustering initialization strategies in Sect. 4. Heuristic schemes for initialization are invested in Sect. 5. In Sect. 6, we review some low-rank approximation methods. In Sect. 7, we demonstrate some

1 https://nimfa.biolaab.si/nimfa.examples.orl_-images.html.
numerical results of performing Lee’s algorithm on the ORL database to demonstrate the performance of each initialization strategy. We finally end up the paper by giving some concluding remarks in Sect. 8.

2 NMF methods

In this section, we review some common approaches for solving NMF and recall multiplicative update rules, i.e., the SED-MU and GKLD-MU proposed by Lee and Seung [28]. These methods have still been widely used as the baseline. The SED-MU updates the matrices $W$ and $H$ by using the following strategy:

$$W_{ia}^{k+1} = W_{ia}^{k} \frac{(X^H H^T)_{ia}}{(W^k H^k H^k^T)_{ia}}, \quad \forall i, a;$$  \(5\)

$$H_{bj}^{k+1} = H_{bj}^{k} \frac{(W^{k+1}^T X)_{bj}}{(W^{k+1}^T W^{k+1} H^{k})_{bj}}, \quad \forall b, j.$$  \(6\)

Moreover, the GKLD-MU can be formulated as:

$$W_{ia} \leftarrow W_{ia} \sum_j \frac{X_{ij}}{(WH)_{ij}} H_{aj},$$

$$W_{ia} \leftarrow \frac{W_{ia}}{\sum_j W_{ja}}, \quad \text{and} \quad H_{aj} \leftarrow H_{aj} \sum_i W_{ia} \frac{X_{ij}}{(WH)_{ij}}.$$  

It has been proven that the multiplicative update rule converges to a local minimum [28].

Another popular approach for solving NMF that applies SED as its objective function is so-called Alternating Non-negative Least Squares (ANLS). It is alternating least squares (ALS) modified under the non-negativity constraint and finds two matrices $W$ and $H$ by solving the following optimization problems:

$$W^{k+1} = \arg \min_{W \geq 0} f(W, H^k) = \frac{1}{2} \|X - W H^k\|_F^2, \quad (7)$$

$$H^{k+1} = \arg \min_{H \geq 0} \frac{1}{2} \|X - W^k H\|_F^2. \quad (8)$$

Although original problem (3) is non-convex and NP-hard with respect to variables $W$ and $H$, sub-problems (7) and (8) are convex problems. However, they may have multiple optimal solutions because they are not strictly convex problems [32].

To accelerate the convergence rate, one popular method is to apply the gradient descent approach with additive update rules. Other techniques such as conjugate gradient [33,34], projected gradient [32,35,36], interior point method [37], and more sophisticated second-order schemes like Newton and quasi-Newton methods are also in consideration [38]. To satisfy the non-negativity constraint, the updated matrices are brought back to the feasible region, i.e., the non-negative orthant, by additional projection, like simply setting all negative elements to zero. The next our goal in this section is to present a general framework for solving NMF in Algorithm 1, which starts with the given non-negative matrix $X$ and the initial matrices, that is, $W^0$ and $H^0$. Then it tries to find two non-negative matrices $W$ and $H$ such that the value of $\|X - WH\|$ is minimized. To do so, first, the algorithm checks the stop condition. If the stop condition is not true, then two matrices $W$ and $H$ will be updated with some common roles. The algorithm repeats this same process until the stop condition is true. In this case, an appropriate solution, i.e., two matrix $W$ and $H$ for origin problem, is obtained.

**Algorithm 1: The generic NMF Algorithm**

**Input:** $X$, $W^0$, $H^0$, $\epsilon$, and the rank of approximation $r$

**Output:** $W$ and $H$

1. Checking Stop Condition(s)
2. if $\|x^r - x^{r+1}\| < \epsilon$ or finish iterations, then
3. finish algorithm
4. else
5. Trial step calculation: Update $W$ and $H$

As mentioned above, in iterative methods for solving NMF, the matrices $W$ and $H$ are obtained in such a way that the value of the objective function is minimized. Based on the non-convexity property for NMF, it generally does not guarantee a unique solution and its solution is dependent on choosing initialization for $W$ and $H$ demonstrated as $W^0$ and $H^0$ in this paper. A good choice for initializing can significantly affect the rate of convergence of the algorithm and considerably reduces the value of the cost function. Therefore, the goal of this paper is to investigate the initialization methods for NMF. The existing initialization approaches for NMF can be classified into four categories, as shown in Fig. 1: Random schemes, which only use the random strategy; Clustering schemes which profit the clustering strategy; Heuristic schemes, which are based on Population-Based Algorithms (PBAs); Low-rank Approximation-Based schemes, which works based on decreasing the matrix rank.

Random strategy can be categorized into five subclasses:

- **Random**, which suggests initial matrix by using random.
- **Random Acol**, which calculates the initial matrix $W$ by getting an average of $q$ random columns of the matrix $X$.
- **Random C**, which calculates the initial matrix $W$ by getting an average of the chooses $q$ columns randomly from the longest (in the $2$-norm) columns of $X$ [39,40].
Co-Occurrence, which computes matrix $W$ by using $X^TX$ [40].

Gabor-based, which calculates the matrix $W$ by using Gabor wavelet, and it is suitable for image datasets [41].

Correspondingly, Clustering strategy is categorized into three subclasses:

- **K-means**, which uses the K-means algorithm for initialization matrix $W$ [42].
- **Fuzzy C-means**, which works based on the fuzzy roles [41,43,44].
- **Hierarchical Clustering**, which groups similar objects into groups called clusters [45].

Besides, Heuristic Schemes are categorized into four subclasses:

- **Genetic Algorithm** [46–48].
- **Particle Swarm Optimization** [49].
- **Differential Evolution** [49].
- **Fish School Search** [49].

Finally, Low-rank Approximation-Based is categorized into four subclasses.

- **Singular Value Decomposition**, which works based on SVD decomposition [30].
- **Non-negative Singular Value Decomposition with Low-Rank Correction**, which generates a positive matrix [50].
- **Non-negative PCA**, which works based on PCA algorithm [41,51].
- **Non-negative ICA**, which works based on ICA algorithm [52–54].

In the following sections, we will discuss in detail each of these algorithms.

### 3 Random schemes

Random initialization is a common approach that is used in the vast majority of NMF studies. Among several random initialization mechanisms, we consider five different random initialization strategies, namely, Random, Random C, Random ACOL, Co-Occurrence, and Gabor-based initialization. We describe them in the rest of this section.

#### 3.1 Random

Probabilistic concepts can be used as an effective method for initializing the NMF. Over the past two decades, probabilistic approaches have been established to compute matrix approximations, forming the field of randomized numerical linear algebra [55]. Random initialization is one of the common methods for initializing the NMF algorithm that is, elements in the matrix $W^0$ (and in $H^0$) are chosen as uniformly dis-
distributed numbers in the interval $[0, 1]$ or in the same range as the entries of the target matrix. However, the quality and reproducibility of the NMF result are rarely questioned when using random initialization. Different initialization of randomness and starting point will lead to different answers, so the algorithm should be run for several instances to select the best results of a local minimum. In addition, Random strategy can be used in many geometric initialization approaches for NMF, in which the columns are selected in a more sophisticated way [56–58].

In general, the randomized algorithms have shown their advantages for solving the linear least squares problem and low-rank matrix approximation. These methods have a low computational cost, but for some cases, the convergence rate to local minima and the qualitative solution is not guaranteed. However, randomness does not deliver reproducible results and does not generally provide a good first estimate for NMF algorithms.

In the standard NMF algorithm, $W^0$ and $H^0$ are two non-negative matrices, where they have drawn from a uniform distribution, usually within the same range as the target matrix entries. This strategy is inexpensive and sometimes provides a good first estimation for the NMF algorithm. Lee and Soung [28] used this approach to initialize the NMF algorithm for the first time. Later on, it has been applied for various NMF algorithms, such as classical matrix factorization [39,59,60]. Wang and Li [61] proposed an approach that based on random projections to efficiently compute the NMF. After that, Tepper and Sapiro [62] suggested a method that compressed the NMF algorithms based on the idea of bilateral random projections, while these compressed algorithms reduced the computational load considerably. However, this strategy was used in most of the NMF algorithms but has a drawback, that is, the algorithm needs multiple runs and in any performing, a different starting point is selected. This significantly increases the computation time needed to obtain the desired factorization. To tackle this problem, several strategies with different approaches for better seeking of NMF have been suggested, for example, computing a reasonable starting point from the target matrix itself. Their goal is to produce deterministic algorithms that need to run only once, still giving meaningful results (e.g., Clustering, SVD) that is the following we will discuss.

### 3.2 Random acol

Random Acol forms an initialization of each column of the basis matrix $W^0$ by averaging $q$ random columns of matrix $X$ [63]. Algorithm 2 presents a generic framework for initialization NMF based on Random Acol.

Random Acol initialization builds basis vectors from the given data matrix; hence, as observed in [40], when the matrix $X$ is sparse, this initialization scheme forms a sparse initial basis matrix $W^0$, which represents a more reasonable choice compared to the random initialization. However, the performance of NMF algorithms initialized by Random Acol scheme is comparable with those of random initialization [39]. Nevertheless, Random Acol has one clear advantage over random initialization, which means it is creating a very sparse $W^0$, but this method is also very inexpensive and easy to implement.

#### 3.3 Random C

Random C initialization is similar to Random Acol initialization, with only one main difference. In fact, it chooses $q$ columns randomly from the longest (in the 2-norm) columns of the matrix $X$, which generally means the densest columns since our text matrices are so sparse. This method is also fairly inexpensive and easy to implement. Moreover, it is summarized in Algorithm 3.

**Algorithm 2: Random ACOL Initialization algorithm**

| Input: $X, q, r$ |
| Output: $W$ |
| Set: $k = 0$ |
| 1 while $k \leq r$ do |
| 2 $k = k + 1$ |
| 3 Select $q$ columns of matrix $X$ as random |
| 4 $s = \text{mean of } q \text{ columns}$ |
| 5 $W_{ik} = s$ |

**Algorithm 3: Random C Initialization**

| Input: $X \in \mathbb{R}^{m \times n}$, $r$ and $q$ |
| Output: $W$ |
| Set: $k = 0$ |
| 1 while $k \leq r$ do |
| 2 $k = k + 1$ |
| 3 Find $q$ of the longest (in the 2-norm sense) columns of $X$ |
| 4 $s = \text{mean of } q \text{ columns}$ |
| 5 $W_{ik} = s$ |

It has been shown that Random C initialization yields better results than the Random Acol initialization for either the asymmetric or the symmetric formulations [64]. They performed their algorithm on the two standard document corpora: the Reuters document corpus, Volume 11 and the TDT5 one. Thus, for some datasets, the Random C initialization is more suitable compared to the Random Acol. Despite having a low computational cost and providing a more realistic first estimate of the sources compared to random initialization, these methods suffer from a lack of reproducibility.
3.4 Co-occurrence

Co-occurrence is a powerful tool for discovering the relationships between heterogeneous collections of attributes or events [65]. Typically, if two such features frequently co-occur throughout a database, it is assumed that they correspond to traits of the same object, concept, or process. The co-occurrence scheme first forms a term co-occurrence matrix $XX^T$. Next, this method randomly chooses the $k$ columns of the initial factor $W^0$ among the densest columns of the co-occurrence matrix and generates $H^0$ (when required) via the random initialization [55]. The co-occurrence scheme has the advantage of producing a basis matrix that includes some hidden information on the initial data (i.e., term-term similarities when a document clustering scenario is considered). However, it requires a higher computational cost than simple random initialization. It is extremely expensive for two reasons. First, if $m \gg n$, which means $C = XX^T$ is very large and often very dense too. Second, the algorithm for finding $W^0$ is extremely expensive, making this method impractical. As evidenced by some authors, the Random C and co-occurrence initialization approaches suffer from lack of diversity [40]. Algorithm 4 summarizes the co-occurrence approach for initializing the NMF that was first proposed in [55].

Algorithm 4: Co-Occurrence algorithm for initialization of NMF

\begin{algorithm}
\hspace*{0.02in} \textbf{Input:} $X \in \mathbb{R}^{m \times n}$ with $X \geq 0, r << \min (m, n)$, and parameter $t$
\hspace*{0.02in} \textbf{Output:} Matrix $W$
\hspace*{0.02in} Set: $C = X \times X^T$
\hspace*{0.02in} 1. Remove columns from $C$ corresponding to stop-words, or having $L_1$ norm less than $t$. Normalize (w.r.t $L_1$ norm) remaining columns.
\hspace*{0.02in} 2. Let ($w_1, w_2) = \text{argmax} ||C_{:,w_1} - C_{:,w_2}||_1$, set $W_{:,1}$ to $C_{:,w_1}$ and $W_{:,2}$ to $C_{:,w_2}$.
\hspace*{0.02in} 3. For each $c$ in between 3 and $r$ do
\hspace*{0.04in} 4. Find word $w$, such that column $C_{:,w}$ would maximize the optimization problem $\min_{1 \leq j \leq c} \|W(:,j) - C(:,w)\|$
\hspace*{0.04in} 5. Set $W_{:,c} = C_{:,w}$
\end{algorithm}

3.5 Gabor-based initialization

Gabor wavelet is a powerful tool in image feature extraction defined by Zheng et al. [41]:

$$\psi_{\mu,v}(z) = \left( \frac{k_{\mu,v}}{\sigma^2} \right)^2 e^{\left( \frac{-|k_{\mu,v}z|^2}{2\sigma^2} \right)} \left[ e^{-ik_{\mu,v}z} - e^{-z^2/2} \right]$$  \hspace{0.5cm} (9)

in which $\mu$ and $v$ denote the orientation and scale of the Gabor kernels and the wave vector $k_{\mu,v}$ is given by:

$$k_{\mu,v} = k_v e^{-i\phi_\mu}$$  \hspace{0.5cm} (10)

So, the Gabor feature representation of an image $I(z)$ is obtained by:

$$G_{\mu,v}(z) = I(z) * \psi_{\mu,v}(z)$$  \hspace{0.5cm} (11)

where $z = (x, y)$ and $*$ is the convolution operator. When an image convolves with Gabor wavelets, the image is transformed into a set of image features at certain scales and orientations. Therefore, the image can be reconstructed from these image features. Motivated by this point, Zheng et al. [41] applied the Gabor-based method to initialize NMF. The advantage of this method is that it is very suitable for image datasets.

4 Clustering schemes

The clustering-based method is one of the common approaches for initializing of the NMF. Since this method produces a summarized view of data helping the analyst to visualize data by means of compact and informative representations of large collections of samples [66]. The NMF as a clustering method can be traced back to work by Lee and Seung [28]. But, the first work that explicitly demonstrates that it was done by Xu et al. [67]. Typically, clustering algorithms are initialized by random strategy. Moreover, these methods have good results in environmental research in public health [68], signal and image processing [69]. If NMF is considered as a clustering process, the initialization strategy can be obtained based on results of clustering algorithms and fuzzy clustering. There are various types of clustering approaches, for example, supervised/unsupervised, hierarchical/partitional, hard/soft, and one way/many way (two-way clustering is known as co-clustering or bi-clustering) among others. Clustering-based initialization schemes will provide more realistic source estimates compared to low-rank approximation methods, but they can be computationally expensive. Furthermore, clustering methods usually require some initialization themselves. Most of the proposed initialization methods have been compared with random initialization in terms of convergence rate and/or quality of the solution. However, different random initialization approaches will lead to different NMF results, making it a questionable reference. It is unclear how previous studies have dealt with the lack of reproducibility. In this case that prototype-based clustering is a convenient method for the problem at hand, NMF could be a valid tool. NMF has been widely used in clustering applications [67,70], where the factors $W$ and $H$ have been interpreted in terms of cluster centroid and cluster membership, respectively. On the other hand, the divergence-based NMF algorithm is not utilization [31,71]. There are several initialization methods that work based on a clustering scheme. Most of these methods have
used the Euclidean distance between the input matrix and the NMF approximation.

Many different clustering methods exist in the literature, such as hierarchical clustering, prototype-based clustering, and density-based clustering. Hierarchical clustering yields a collection of nests of groups of data, while prototype-based clustering groups are represented in a compressed form through a prototype, i.e., an element belonging to the same domain of data. In density-based clustering, groups are formed in regions of data space where data are more crowded. The choice of the most appropriate method is up to the data analyst. In the following, we will concentrate on three well-known clustering schemes, i.e., K-means, Fuzzy C-means, and hierarchical clustering.

4.1 K-means

The K-mean method is a clustering technique used to grouped similar patterns in given features. The K-means (Grst introduced it in 1960 [72]) is the most widely used clustering technique [73]. This method represents points in the k-space that are the centers of clusters of nodes with the characteristic that they minimize the sum of squared distance deviations of the points in each cluster from the assigned cluster “centroid.” The K-means algorithm is an iterative algorithm for minimizing the sum of distance between each data point and its cluster center (centroid) and tries to minimize the sum-squared-error criterion. Generally, K-means method seeks to partition the dataset X into k disjoint clusters so that each point in the cluster is “closer” to the centroid associated with that cluster than it is to the other k - 1 centroids in the Euclidean sense.

As the K-means factor is added to NMF, it gives prominent importance in clustering with extracted features. The theoretical connection between factorization NMF with additional orthogonal constraints on its factors, K-means, and spectral clustering was demonstrated in [74], while the mathematical equivalence between orthogonal NMF and a weighted variant of spherical K-means was proved together with some indications about the cases in which orthogonal NMF should be preferred over K-means and spherical K-means.

The objective functions for K-means are defined as:

\[
M = \sum_{j=1}^{l} \sum_{i=1}^{n} \|x_i^{(j)} - c_j\|^2, \tag{12}
\]

where \(x\) is the feature vector, \(c_j\) denotes the center of the cluster, and \(j\) is the number of the cluster centers. The theoretical connection between K-means and NMF can be presented as:

\[
\min_{i=1}^{n} \|x_i - w_{c_i}\|^2 = \min \|X - WH\|_F^2, \tag{13}
\]

where \(W\) and \(H\) are two non-negative matrices. Moreover, \(x_1, ..., x_n\) denote the columns of \(X\) and \(w_1, ..., w_k\) are the \(k\) centroids and \(c_i = j\) when \(i\)-th point is assigned to \(j\)-th cluster \((j \in 1, ..., k)\). K-means methods find minimum of \(M\) often by applying iterative gradient descent approaches and usually converge to local minima.

There are several methods for initializing the NMF-based K-mean approach. We point out some of them here.

- The initial basis matrix \(W^0\) is constructed by using the K-means clustering approach and the initial matrix \(H^0\) is considered as a random matrix.
- The initial basis matrix \(W^0\) is constructed by using the K-means clustering strategy, and the initial matrix \(H^0\) is calculated by \(H^0 = (W^0)^T X\) and then the absolute value function is used for all elements in \(H^0\) in order to satisfy the initial constraint of NMF.
- The initial basis matrix \(W^0\) is obtained by using the cluster centroids obtained from K-means clustering. The initial matrix \(H^0\) is obtained by \(H^0 = (W^0)^T X\), and then all negative elements in \(H^0\) are transferred to zero in order to satisfy the initial the constraint of NMF.
- The initial basis matrix \(W^0\) is obtained by using the cluster centroids obtained from K-means clustering. The value of the membership degrees of each data point is calculated by:

\[
h_{kq} = \frac{1}{\left(\sum_{k=1}^{K} d(x_q,c_k)\right)^{1\over 2}},
\]

where \(d(.,.)\) denotes the Euclidean distance between the two points, \(x_q\) represents the \(q\)-th data point, and \(c_k\) represents the \(k\)-th cluster centroid. Moreover, the fuzzification parameter is denoted by \(m\). The initial matrix \(H^0\) is then obtained by using the membership degrees above.

Many random initialization methods for the K-means algorithm have been proposed so far. Most classical methods are random seed [72,75] and random partition [75]. Random seeds randomly select \(k\) instances (seed points) and assign each of the other instances to the cluster with the nearest seed point. Random partition assigns each data instance into one of the \(k\) clusters randomly. To escape from getting stuck at a local minimum, one can apply \(r\) random starts.

To improve the performance of divergence-based NMF algorithm that works based on Xue’s idea [42], a new method using the K-means and combination of normalizing technique with set divergence as the similarity measure in clustering to find the base vectors for NMF initialization and search of the centroids was first proposed by [42]. The authors used the \(L_1\) norm to normalize their algorithm. Their algorithm works based on the clustering method, and it stops...
when the number of clusters does not change. In addition, the spherical K-means clustering, which is a sparse K-means method, was first proposed to produce a structured initialization for NMF in [76].

4.2 Fuzzy C-means

The fuzzy set theory introduced by Zadeh et al. [77] provides a powerful analytical tool for the soft clustering method. The Fuzzy C-Means is the best-known approach for fuzzy clustering, based on optimizing an objective function. This concept has many applications as a convenient tool in clustering and has the most perfect algorithm theory. The FCM clustering algorithm can be considered as a variation and an extension of the traditional K-means clustering algorithm, in which for each data point a degree of membership or membership function of clusters is assigned. It is proven that the fuzzy clustering is an adaptation to noisy data and classes that are not well separated. By considering this property of fuzzy clustering, some research papers were done in this area. For example, Zheng et al. [41] proposed the FCM concept. They used their strategy to initialization of NMF. In another work, Rezaei et al. [44] applied FCM to initialize NMF as an efficient method to enhance NMF performance. [43] applied the FCM for initialization semi-non-negative matrix factorization and implemented their algorithm on the source signals. The source signals are not always sparse in the time domain.

Here, we present the FCM algorithm for initialization NMF [41].

Algorithm 5: Fuzzy C-means Algorithm

Input: $X \in \mathbb{R}^{m \times n}$ with $X \geq 0$ and $r << \min (m, n)$
Output: Matrix $W$
1. Fix the number of cluster $c$, fuzzier $q$, termination tolerance $\epsilon$ and any inner product norm metric for $\mathbb{R}^n$.
2. Initialize cluster center matrix $W$.
3. while $J \geq \epsilon$

4. Compute dissimilarity matrix $D_{i,j} = \|W_{i} - X_{j}\|$
5. Compute the membership matrix $U_{i,j} = \frac{1}{\sum_{k=1}^{c} (\frac{D_{i,k}}{D_{j,k}})^{-q}}$
6. Update cluster center $W$ by using: $W_{i} = \frac{1}{\sum_{j=1}^{n} U_{i,j}} X_{j}$
7. Calculate objective function $J = \sum_{i=1}^{c} \sum_{j=1}^{n} U_{i,j} D_{i,j}$

4.3 Hierarchical clustering

This method is motivated by common sense on “part,” which is the smallest unit that has some perceptual meaning. For example, a face image consists of various parts, including eyes, nose, eyebrows, cheek, lip, and so on. Metaphorically, a pixel corresponds to an atom, and then, a part can be considered as a molecule. As atoms in a molecule perform a chemical reaction together, pixels that build a part should be grouped together. They introduced a “closeness to rank-one” (CRO) measure in order to investigate whether row vectors in the sub-matrix show similar patterns or not. The CRO measure is defined by:

$$CRO(X_{(i,j),...}) = \frac{\sigma_1^2}{\sum_{j=1}^{c} \sigma_j^2} = \frac{\sigma_1^2}{\|X_{(i,j),...}\|_F^2}. \quad (14)$$

where $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r \geq 0$ are singular values of the sub-matrix $X_{(i,j),...}$, and $r$ denotes the rank of the $X_{(i,j),...}$. Algorithm 6 presents a general framework for the CRO algorithms. It was first proposed in [45]. They used this method to initialize the NMF algorithm and compared their results with random initialization strategy and investigated how goodness-of-fit (GOF) and sparseness change after the convergence of the standard NMF algorithm starting from these two different initialization methods.

Algorithm 6: CRO-based hierarchical clustering

Input: $X \in \mathbb{R}^{m \times n}$ with $X \geq 0$ and $r << \min (m, n)$
1. Assign each row vector $X_{1:}, X_{2:}, ..., X_{m:}$ into each own clusters $C_1, C_2, ..., C_m$
2. Calculate CRO in (14) between every pair of clusters
3. while $n$ clusters remains do
4. Find a pair of clusters with the largest CRO
5. Merge them into a single cluster
6. Compute CRO between the newly merged cluster and remaining clusters

5 Heuristic schemes

An important aspect which has not been deeply investigated yet is a proper initialization of the NMF factors in order to achieve a faster error reduction [78]. Thus, several heuristics algorithms have been proposed to solve NMF problem. However, only a few studies combined NMF and population-based algorithms (PBAs) and both of them are based on population-based optimization algorithms. Goldberg and Holland [79] presented genetic algorithms (GA) which are global search heuristics that operate on a population of solutions using techniques encouraged from evolutionary processes such as mutation, crossover, and selection. Stadthanner et al. [46] investigated the application of GA on sparse NMF for microarray analysis, while [80] proposed GA for Boolean matrix factorization, a variant of NMF for binary data based on Boolean algebra. The results in these two papers are promising, but barely connected to the initialization techniques introduced in this paper. In particle swarm
optimization (PSO) [81] each particle in the swarm adjusts its position in the search space based on the best position it has found so far as well as the position of the known best-fit particle of the entire swarm. In differential evolution (DE) [48] a particle is moved around in the search space using simple mathematical formulation if the new position is an improvement the position of the particle is updated; otherwise, the new position is discarded.

Algorithm 7 presents a pseudocode for NMF initialization using PBAs which was first proposed in [49]. They used their strategy to initialize the NMF and compared obtained results with some other algorithms such as random, NNDSVD and showed that their method has better results in terms of convergence.

6.1 Singular value decomposition

The potential impact of the NMF and its extensions on scientific advancements might be as great as the other popular matrix factorization technique, such as SVD, that is based on low-rank approximations. LR approximation using SVD has many applications over a wide spectrum of disciplines, for example, image compression, similarly, text data latent semantic indexing [82], event detection in streaming data, visualization of a document corpus, etc. In particular, Boutsidis and Gallopoulos [30] pointed out SVD-NMF has good properties under these two conditions:

- One that leads to rapid error reduction and faster convergence.
- One that leads to the overall error at convergence.

There exists a factorization with the following form:

\[ X = W \Sigma H^T \]  

(15)

Let us denote orthogonal matrices as \( W = [w_1, w_2, ...] \) and \( H = [h_1, h_2, ...] \) that include left and right singular vectors of \( X \), respectively. Moreover, the matrix \( \Sigma = diag(\sigma_1, \sigma_2, ..., \sigma_r) \) is a diagonal matrix containing the first \( r \) singular values of \( X \) and \((\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0)\).

The truncated SVD is the best rank-\( r \) approximate of matrix \( X \), in either spectral norm or Frobenius norm [83]. In particular, the singular values decrease quickly as \( i \) increasing in most instances [84], which means that some of the first singular values can contain almost all singular information of the input matrix.

There are two other papers that combine NMF and PBAs. In fact, both of them are based on GAs. [46] has investigated the application of GAs on sparse NMF for microarray analysis, while [80] have applied GAs for Boolean matrix factorization, a variant of NMF for binary data based on Boolean algebra.

6 Low-rank Approximation-Based

In this section, we focus on the most important low-rank (LR) approximation algorithms. Initialization schemes based on LR decomposition strategies do not require a randomization set. The LR methods include strategies using the Singular Value Decomposition (SVD), Non-negative Singular Value Decomposition with Low-Rank Correction (NNDSVD-LRC), Non-negative Principal Component Analysis (NPCA), and Non-negative Independent Component Analysis (NICA).
SVD does not necessarily produce the non-negative matrices. So, some algorithms in this area change negative elements to positive or zero.

This method has some drawbacks:

- The interpretability of the transformed features. The resulting orthogonal matrix factors generated by the approximation usually do not allow for direct interpretations in terms of the original features because they contain positive and negative coefficients [41].
- This method suffers from the fact that the approximation error \( \|X - WH\|_F^2 \) of the initial factors \((W, H)\) increases as the rank increases, which is not a desirable property for NMF initialization.

Non-negative Double Singular Value Decomposition (NND SVD) [30] is another method designed to enhance the initialization stage of the NMF. This method contains no randomization and is based on two SVD processes, one approximating the data matrix, the other approximating positive sections of the resulting partial SVD factors utilizing an algebraic property of unit rank matrices. NND SVD can readily be combined with existing NMF algorithms. This property leads to the NNSVD being considered as an efficient method for initializing the NMF. Algorithm 9 summarizes the NND SVD approach. This algorithm was first presented in [30]

\[ \text{Algorithm 9: NND SVD initialization of non-negative matrix} \]

**Input:** \( X \in \mathbb{R}^{m \times n} \) with \( X \geq 0 \) and \( r < \min(m, n) \)

**Output:** \( W \) and \( H \)

1. Compute the largest \( k \) singular triplets of \( X = [U, \Sigma, V] \)
2. Initialize \( W_{\perp} = \sqrt{\Sigma_{\perp}} \times U_{\perp} \) and \( H_{\perp} = \sqrt{\Sigma_{\perp}} \times V_{\perp}^T \)
3. for \( i = 2 \) to \( r \) do
   4. Define \( x := U_{\perp} \) and \( y := V_{\perp} \)
   5. Define \( xp := \text{pos}(x) \), \( xn := \text{neg}(x) \), \( yp := \text{pos}(y) \), and \( yn := \text{neg}(y) \)
   6. if \( \|xp\| \times \|yp\| > \|xn\| \times \|yn\| \) then
      7. Put \( u = \frac{xp}{\|xp\|} \), \( v = \frac{yp}{\|yp\|} \), and \( \sigma = \|xp\| \times \|yp\| \)
   8. else
      9. Put \( u = \frac{xn}{\|xn\|} \), \( v = \frac{yn}{\|yn\|} \), \( \sigma = \|xn\| \times \|yn\| \)
   10. \( W_{\perp} = \sqrt{\Sigma_{\perp}} \times \sigma \times u \) and \( H_{\perp} = \sqrt{\Sigma_{\perp}} \times \sigma \times v^T \)

6.2 Non-negative Singular Value Decomposition with Low-Rank Correction

The Non-negative Singular Value Decomposition with Low-Rank Correction (NNSVD-LRC) was first proposed in [50].

This method works based on the SVD, but it has some useful properties such as:

- This method generates sparse factors which not only provide storage efficiency but also provide better part-based representations and resilience to noise.
- It only requires a truncated SVD of rank \( \lceil \frac{r}{2} \rceil + 1 \).

Here, we describe the NNSVD-LCR framework in Algorithm 10 which was first proposed in [50]:

**Algorithm 10:** Non-negative Singular Value Decomposition with Low-Rank Correction (NNSVD-LRC)

| Input | \( X \in \mathbb{R}^{m \times n} \) with \( X \geq 0 \) and \( r < \min(m, n) \) |
|-------|--------------------------------------------------|
| Output | \( W \) and \( H \) |
| Set | \( p = \lceil \frac{r}{2} \rceil + 1 \) |
| \( U, \Sigma, V \) | truncated-SVD(X, p) |
| \( Y_p = U \Sigma_p^T \) | \( Z_p = \Sigma_p^T V^T \) |
| \( W(:,1) = \lceil Y_p(:,1) \rceil \) | \( H(:,1) = \lceil Z_p(:,1) \rceil \) |
| \( i = 2 \) | \( j = 2 \) |
| while \( i \leq r \) do | |
| if \( i \) is even then | |
| \( W(:,i) = \text{max}(Y_p(:,i), 0) = \text{max}(Z_p(:,i), 0) \) | |
| else | |
| \( j = j + 1 \) | |
| \( W(:,i) = \text{max}(-Y_p(:,i), 0) = \text{max}(-Z_p(:,i), 0) \) | |
| \( i = i + 1 \) | |

6.3 Non-negative PCA

Principal component analysis (PCA) is one of the best-known unsupervised feature extraction methods because of its conceptual simplicity and the existence of efficient algorithms that can implement it. Particularly in the face representation task, faces can be economically represented along with the eigenface coordinate space and approximately reconstructed using just a small collection of eigenfaces and their corresponding projections (coefficients). It is an optimal representation in the sense of mean-square error. However, it presents some drawbacks (such as the presence of mixed sign values), and several research papers demonstrated that it outperforms NMF in many applications such as face recognition [85,86].

Principal components sequentially capture the maximum variability among data, thus guaranteeing minimal information loss, and they are mutually uncorrelated. To clarify, consider the problem of human face recognition, where PCA has been largely adopted to obtain a set of basic images, the eigenfaces, that can be linearly combined to reconstruct images in the original dataset of face [87]. Here we describe this method. Given the \( m \times n \) matrix \( X \) as that of in NMF,
we define the average vector $\psi$ as:

$$\psi = \frac{1}{n} \sum_{i=1}^{n} X_i$$

The centered matrix can be calculated as:

$$\bar{X} = (X_1 - \psi, ..., X_n - \psi).$$

Using SVD to compute the eigenvectors of the $\bar{X}^T \bar{X}$, the eigenvector matrix $W$ is constructed by keeping only $r$ eigenvectors (corresponding to the $r$ largest eigenvalues $\lambda_i$) as column vectors, and $H$ is an $r \times n$ matrix containing the encoding coefficients. Moreover, the criterion of selecting $r$ is usually as follows

$$\frac{\sum_{i=1}^{r} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \geq \alpha,$$

in which $\alpha = 0.9$. As we know that the NMF seeks to find two non-negative matrices for initialization. So, the following nonlinear operator can be used to transfer negative elements to zero.

$$p(W) = p(W+) = \max(W_{ij}, 0) \quad p(H) = p(H+)$$

$$= \max(H_{ij}, 0). \quad (17)$$

There are several works in this area, for example, Zheng et al. [41] proposed PCA-based initialization method and, after obtaining $W$ and $H$, all negative elements in these two matrices change to zero. In another work, Zhao et al. [51] used the absolute value for all elements in the initial matrices $W$ and $H$ after PCA initialization. With these initialization methods, enhanced convergence rates as well as better accuracy were achieved. Geng et al. [88] pointed out the NMF is sensitive to noise (outliers) and used PCA to initialize the NMF.

Despite the popularity of PCA, this method has two key drawbacks:

- It lacks sparseness (i.e., factor loadings are linear combinations of all the input variables), yet sparse representations are generally desirable since they aid human understanding (e.g., with gene expression data), reduce computational costs and promote better generalization in learning algorithms [89,90]
- PCA is computationally expensive, and the size of the covariance matrix is proportional to the dimension of the data. As a result, the computation of the eigenvectors and eigenvalues might be impractical for high-dimensional data.

### 6.4 Non-negative ICA

Independent component analysis (ICA) is another mechanism that used to extract a set of statistically independent source variables from a collection of mixed signals without having information about the data source signals or the combination process. The initialization methods using PCA or SVD are based on the orthogonality between the bases representing the data matrix $X$. However, it has been shown that the optimal NMF bases are along the edges of a convex polyhedral cone, which is defined by the observed points in $X$, in an $M$-dimensional space [91,92]. Therefore, PCA and SVD may not be the best methods for the initialization in NMF. To avoid this situation, some researchers proposed the utilization of NICA bases and estimated independent sources as the initial values of the basis and weight matrices, respectively [52,53]. The numerical results provided faster and deeper convergence of the NMF cost function than the conventional methods. Benachir et al. [54] modified standard ICA taking into account the sum-to-one constraint and then eliminated some indeterminacies related to ICA using different strategies. Then, they used the outputs to initialize an NMF method. Another approach called non-negative IPCA which combines the advantage of both PCA and ICA was proposed by Yao et al. [93]. This approach can be considered as an initialization approach for NMF, which presented by Algorithm 11 [94].

#### Algorithm 11: Non-negative ICA

**Input**: $X \in \mathbb{R}^{m \times n}$ with $X \geq 0$ and $r \ll \min (m, n)$  
**Output**: $W$ and $H$

1. Set $A$ as a pseudoinverse of $X$
2. Apply PCA on the matrix $A$ to generate the basic matrix $W$
3. Whiten the basic matrix $W$ obtained above by using the eigenvalue decomposition of the covariance matrix of $W$.
4. Implement ICA algorithm on the whitened matrix $W$ and obtain the independent basic matrix $W^0$
5. Calculate the matrix $H^0$ by using $H^0 = (W^0)^T X$
6. Take the absolute value for all elements in $W^0$ and $H^0$.

### 7 Numerical results

In this section, we present some numerical simulations of performing Lee’s algorithm Lee and Seung [28] on the ORL dataset with some different initialization strategies. The goal of this section is to compare convergence and accuracy of the algorithm based on different initialization approaches. In all experiments, the following stopping condition is used:

$$\|W^k H^k - W^{k-1} H^{k-1}\| \leq \epsilon,$$

where $\epsilon = 10^{-10}$. 
7.1 Datasets and settings

All codes of the computer procedures were written in Python environment and carried out on a PC (CPU 2.60 GHz, 16G memory) with the Windows 10 operating system environment. To initialize the NMF, we choose 6 initialization approaches, which are popular, common and also easy to implement. These methods are Random, Random c and Co-occurrence approaches from random group, K-means from clustering group, NICA and SVD from low-rank group. For each approach, matrices \( W^0 \) and \( H^0 \) are determined as:

- **Random**: The Random approach is the easiest method in terms of implementation and uses frequently in the NMF algorithms. For this case, we initialize matrices \( W^0 \) and \( H^0 \) as random.

- **Co-Occurrence**: We initialize the matrix \( W^0 \) by using Algorithm 4 and the matrix \( H^0 = (W^0)^T X \).

- **Random C**: We apply Algorithm 3 to initialize matrix \( W^0 \). In addition, the matrix \( H^0 \) is obtained by \( H^0 = (W^0)^T X \).

- **SVD**: We perform Algorithm 8 to achieve matrices \( H^0 \) and \( W^0 \).

- **K-means**: The initial basis matrix \( W^0 \) is obtained by using the cluster centroids obtained from K-means clustering. The initial matrix \( H^0 \) is obtained by \( H^0 = (W^0)^T X \), and then all negative elements in \( H \) are transferred to zero.

- **NICA**: The initial basis matrix \( W^0 \) and \( H^0 \) are obtained by implementing Algorithm 11.

We perform the algorithm on the dataset ORL which has 400 images of 40 different classes that each of them has 10 images. The images were taken at different times, lighting and facial expressions. The faces are in an upright position in frontal view, with a slight left–right rotation. We performed the algorithm on original face images by constructing a matrix of shape, 10304 (pixels) x 400 (faces). It is clear that the number of images for training and test is very important in machine learning area. So, we run the algorithm with three different cases in terms of the number of data for training and testing and report the accuracy of the algorithm in each case. In fact, we perform the algorithm with number of training as \{320, 200, 120\}. The error for all experiments is calculated by:

\[
\text{Error} := \frac{\|X - WH\|_F}{\|X\|_F}. \quad (18)
\]

For the random group, we performed their algorithms 10 times and demonstrated the average results for each case. Figure 2 shows approximation error for the case where the number of training data is 320. As we see that, the NICA has the best result than the other algorithms. Moreover, the random strategy has the highest error. Figure 3 shows the error for the case where the number of training data is 200. Moreover, error for the case where the number of training data is 120 is demonstrated in Fig. 4.

As we see, the results of performing the algorithm with the NICA initialization approach are better than the other methods. In fact, this strategy improved the convergence results. This method has the best results for accuracy and the number of iterations. As shown in the figures, the NICA strategy has the best performance. In fact, the algorithm based on this method converges faster, and the value of error is the lowest.

In fact, as mentioned before, the NICA method is not suitable for some datasets due to the production of orthogonal matrices. These results also show that although the random
strategy is not expensive, its results are not as good as other methods.

Table 1 presents the performance of the NMF in terms of the accuracy of face recognition based on 6 different strategies for initialization. In addition, we measure the accuracy of the algorithm in face recognition by \( \text{Accuracy} := \frac{\text{The num of faces recognized correctly}}{\text{The total num of test data}} \times 100. \)

Based on Table 1, it is clear that the K-means and NICA approaches achieved the best results in terms of accuracy. The obtained results show that the Random strategy is not proper for ORL dataset; however, the Random C is a suitable choice between random group for initialization of NMF. The results obtained for SVD are not as good as K-means approach. One of the reasons for this issue is that the matrix \( W^0 \) is an orthogonal matrix, which causes the algorithm cannot classify new sample correctly.

### 8 Conclusion

In this paper, we studied various initialization strategies for NMF algorithms. In this direction, we first reviewed some common approaches for solving NMF. Then we classified the initialization approaches for NMF. In fact, these methods were divided into four classes, that is, Random, Clustering, Heuristic, and Low-rank approximation methods. We summarized the advantages and disadvantages for some of them in terms of computational complexity, difficult/easy to implement as well as some other characteristics in Table 2.

| Name          | Pros                               | Cons                                    |
|---------------|------------------------------------|-----------------------------------------|
| Random        | Easy                               | Dense \((W^0, H^0)\)                      |
|               | Cheap to compute                   | With no intuitive meaning               |
| Co-occurrence | Uses term-term similarities        | Very expensive                          |
| Random C      | Cheap                              | Sparse                                  |
| Random Aco    | Cheap                              | Only slight decrease number of iterations |
| k-means       | Reduces NMF iterations              | Dense                                   |
|               | Intuitive meaning of \( W^0 \)      | Expensive                               |
| FC-means      | Intuitive meaning of \((W^0, H^0)\) | Dense \((W^0, H^0)\)                     |
|               |                                     | expensive                               |
| NNDSVD        | No randomization;                   | Expensive                               |
| NNSVD-LRC     | No randomization;                   | Expensive                               |
|               | Non-negative;                       |                                        |
is very low. On the other hand, NNDSVD, FC-Means, Co-occurrence, and k-means are expensive methods that means the computational complexity of their algorithm is high. The NNSVD-LRC is another method for initialization that is not expensive than SVD. NICA and NPCA are two other methods that are not expensive as the K-means strategy, but they have better results. Numerical results presented in this paper showed that the approach NICA has better results than the other strategies in terms of error and accuracy. In fact, we performed Lee’s algorithm on ORL datasets with three different cases for the number of training data i.e., 120, 200, 320. The next goal is to investigate the behavior of the NMF algorithm on different types of datasets based on different strategies for initialization.

Acknowledgements The authors would like to thank the editors and anonymous reviewers for their constructive comments.

Funding No funding was received for conducting this study.

Code availability In this paper, the NMF library was employed. https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html

Declarations Conflict of interest The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

1. Mehrmann, V., Miedlar, A., Nkengla, M., Friedland, S.: Fast low rank approximations of matrices and tensors. Electronic J. Linear Algebra. 22, 1031–1048 (2011)
2. Eldén, Lars.: Numerical linear algebra and applications in data mining and it. (2003)
3. Skillcorn, D.: Understanding complex datasets: data mining with matrix decompositions. Chapman and Hall, London (2007)
4. Murphy, K.P.: Machine learning: a probabilistic perspective. MIT press, Cambridge (2012)
5. Lee, J., Kim, S., Lebanon, G., Singer, Y.: Matrix approximation under local low-rank assumption. http://arxiv.org/abs/1301.3192, (2013)
6. Ling, Y.T., Sani, N.F.M., Abdullah, M.A., Hamid, N.A.W.A.: Structural features with nonnegative matrix factorization for metamorphic malware detection. Computers Secur. 104, 102216 (2021)
7. Ling, Y.T., Sani, N.F.M., Abdullah, M.T., Hamid, N.A.W.A.: Nonnegative matrix factorization and metamorphic malware detection. J. Computer Virol. Hack. Tech. 15(3), 195–208 (2019)
8. Ling, Y.T., Sani, N.F.M., Abdullah, M.T., Hamid, N.A.W.A.: Metamorphic malware detection using structural features and nonnegative matrix factorization with hidden markov model. J. Computer Virol. Hack. Tech. 25, 1–21 (2021)
9. Golub, G., Loan, C.F.V.: Matrix computations, 3rd edn. The John Hopkins University, Baltimore (1996)
10. Alter, O.B., Patrick, O., Botstein, D.: Singular value decomposition for genome-wide expression data processing and modeling. Proc. Nat. Acad. Sci. 97(18), 10101–10106 (2000)
11. Wall, Michael E., Rechtsteiner, A., Rocha, Luis M.: Singular value decomposition and principal component analysis in a practical approach to microarray data analysis. Springer, Cham (2003)
12. Datta, BN.: Numerical linear algebra and applications, volume 116. Siam, (2010)
13. Sundarapandian, V.: Numerical linear algebra. PHI Learning Pvt. Ltd., NY (2008)
14. Trefethen, Lloyd N., Bau, David III.: Numerical linear algebra, volume 50. Siam, (1997)
15. Paruolo, Paolo.: Multivariate reduced rank regression, theory and applications, (2000)
16. Miller, D.N., Callahan, Raymond A.: Identification of linear time-invariant systems via constrained step-based realization. IFAC Proc 45(16), 1155–1160 (2012)
17. Chu, M.T., Funderlic, R.E.: Structured low rank approximation. Linear Algebra Appl. 366, 157–172 (2003)
18. Drineas, P., Kannan, R., Mahoney, M.: Fast monte carlo algorithms for matrices: approximating matrix multiplication. SIAM J. Comput. 36(1), 132–157 (2006)
19. Kolda, T.G., Oleary, D.P.: A semidiscrete matrix decomposition for latent semantic indexing information retrieval. ACM Trans. Inf. Syst. (TOIS) 16(4), 322–346 (1998)
20. Abdi, H., Williams, L.: Principal component analysis. Wiley Interdisciplinary Rev.: Comput. Stat. 2(4), 433–459 (2010)
21. Kwak, K.-C., Pedrycz, W.: Face recognition using an enhanced independent component analysis approach. IEEE Trans. Neural Netw. 18(2), 530–541 (2007)
22. Recht, B., Fazel, M., Parrilo, P.: Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM Rev. 52(3), 471–501 (2010)
23. Gander, W.: Algorithms for the QR decomposition. Res. Rep 80(02), 1251–1268 (1980)
24. Esposito, F., Del, B., Nicoletta, S.L.: Nonnegative matrix factorization models for knowledge extraction from biomedical and other real world data. PAMM 20(1), e202000032 (2021)
25. Peng, C., Zhang, Z., Chen, C., Kang, Z., Cheng, Q.: Two-dimensional semi-nonnegative matrix factorization for clustering. Inf. Sci. 590, 106–141 (2022)
26. Kang, Z., Lin, Z., Zhu, X., Xu, W.: Structured graph learning for scalable subspace clustering: From single view to multiview. IEEE Trans. Cybern. 26, 71 (2021)
27. Peng, C., Zhang, Z., Kang, Z., Chen, C., Cheng, Q.: Nonnegative matrix factorization with local similarity learning. Inf. Sci. 562, 325–346 (2021)
28. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. Nature 401(6755), 788–791 (1999)
29. Aggarwal, Charu C., Reddy, Chandan K.: Data clustering. Algorithms and applications. Chapman&Hall/CRC Data mining and Knowledge Discovery series, Londra, (2014)
30. Boutsidis, C., Gallopoulos, E.: Svd based initialization: a head start for nonnegative matrix factorization. Pattern Recogn. 41(4), 1350–1362 (2008)
31. Wild, S., Wild, W-S., Curry, J., Dougherty, A., Bettermont, M.: Seedng non-negative matrix factorizations with the spherical k-means clustering. PhD thesis, Citeeseer, (2003)
32. Dong, X., Zhang, W., Li, X.: Efficient nonnegative matrix factorization via projected newton method. Pattern Recognit. 45(9), 3557–3565 (2012)
33. Hafshejani, S-F., Gaur, D., Hossain, S., Benkoczi, R.; Barzilai and borwein conjugate gradient method equipped with a non-monotone line search technique and its application on non-negative matrix factorization. http://arxiv.org/abs/2109.05685, (2021)
34. Li, X., Zhang, W., Dong, X.: A class of modified fr conjugate gradient method and applications to non-negative matrix factorization. Computers Math. Appl. 73(2), 270–276 (2017)
75. Anderberg, Michael R.: Cluster analysis for applications: probability and mathematical statistics: a series of monographs and textbooks. Academic press, Cambridge (2014)

76. Wild, S., Curry, J., Dougherty, A.: Improving non-negative matrix factorizations through structured initialization. Pattern Recognit. 37(11), 2217–2232 (2004)

77. Anderberg, Michael R.: Cluster analysis for applications: probability and mathematical statistics: a series of monographs and textbooks. Academic press, Cambridge (2014)

78. Dong, B., Lin, Matthew M., Chu, Moody T.: Rank factorization—a heuristic approach via rank reduction. Numer. Algorith. 65(2), 251–274 (2014)

79. Goldberg, David E., Holland, J H.: Genetic algorithms and machine learning. (1988)

80. Snášel, Václav., Platoš, Jan., Krömer, Pavel.: Developing genetic algorithms for boolean matrix factorization. Databases, Texts, 61, (2008b)

81. Eberhart, R., Kennedy, J.: Particle swarm optimization. In Proceedings of the IEEE international conference on neural networks, volume 4, pages 1942–1948. Citeseer, (1995)

82. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. 41(6), 391–407 (1990)

83. Eckart, Carl, Young, Gale: The approximation of one matrix by another of lower rank. Psychometrika 1(3), 211–218 (1936)

84. Cichocki, A., Zdunek, R., Phan, A.H., Amari, S.-I.: Nonnegative matrix and tensor factorizations: applications to exploratory multivariate data analysis and blind source separation. Wiley, NY (2009)

85. Guillaumet, David, Vitria, Jordi: Evaluation of distance metrics for recognition based on non-negative matrix factorization. Pattern Recognit. Lett. 24(9–10), 1599–1605 (2003)

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.